ALARIC: An algorithm for constructing arbitrarily complex initial density distributions with low particle noise for SPH/SPMHD applications.

Luis Vela Vela[a,*], Raul Sanchez[a], Joachim Geiger[b]

[a]Departamento de Física. Universidad Carlos III, Leganés, Spain
[b]Max-Planck-Institut für Plasmaphysik, Greifswald, Germany

Abstract

A method is presented to obtain initial conditions for Smoothed Particle Hydrodynamic (SPH) scenarios where arbitrarily complex density distributions and low particle noise are needed. Our method, named ALARIC, tampers with the evolution of the internal variables to obtain a fast and efficient profile evolution towards the desired goal. The result has very low levels of particle noise and constitutes a perfect candidate to study the equilibrium and stability properties of SPH/SPMHD systems. The method uses the iso-thermal SPH equations to calculate hydrodynamical forces under the presence of an external fictitious potential and evolves them in time with a 2nd-order symplectic integrator. The proposed method generates tailored initial conditions that perform better in many cases than those based on purely crystalline lattices, since it prevents the appearance of anisotropies.

Keywords: Smoothed Particle Hydrodynamics, SPH, Smoothed Particle Magneto-hydrodynamics, SPMHD, Initial Conditions, Low Noise.

1. Introduction

1.1. Context

Smoothed Particle Hydrodynamics, SPH for short, was first introduced in the late 70's by Gingold and Monaghan [1] together with Lucy [2] as a La-

[*] luvelav@fis.uc3m.es

grangian numerical method to solve the equations of hydrodynamics [3]. Since then, the method has gained both popularity and robustness [4] and is widely used today to simulate a wide variety of scenarios, ranging from its original hydrodynamic cradle ([5], [6]) to modern fracture/elasticity studies ([7], [8]), turbulence ([9], [10]), magneto-hydrodynamics ([3], [11], [12]) and relativistic flows [13], not to mention a myriad of applications in industry including visual effects in cinema [7]. Historically, the philosophy of SPH could best be identified with the ideas behind the early PIC codes developed at LANL and maybe even traced back to the beginnings of Weighed Residual Methods in the late 30's as a Collocation Method [14].

In the process of becoming the successful numerical method it is today, SPH encountered many problems on its way, many of which were resolved: Considerable time has been invested in studying the numerical properties of different kernels ([15], [16], [17]), contructing a conservative formulation of the SPH equations not prone to numerical instabilities ([3], [11], [18]), dealing with arbitrary boundaries ([19], [20]) or with the correct discretization of the Laplacian operator -which is the mechanism responsible for dissipative effects- ([21], [22]). However, little discussion has taken place around the creation of arbitrarily complex, low-noise, initial conditions. We see now that as the method expands in its scope, the need to initialise the particles with arbitrary mass density distributions becomes indispensable. In this manuscript we describe, examine and offer a solution to this problem which is both straightforward and efficient.

1.2. Basics of Smoothed Particle Hydrodynamics

As mentioned before, SPH is a Lagrangian numerical method to solve the equations of hydrodynamics. In their continuum form, the equations describing the motion of a moving fluid are usually stated as:

$$
\begin{aligned}
\frac{\mathrm{d}\rho}{\mathrm{d}t} &= -\rho(\nabla\cdot\mathbf{v}) && (1)\\
\rho\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} &= -\nabla p && (2)\\
\frac{\mathrm{d}u}{\mathrm{d}t} &= -(\gamma-1)u(\nabla\cdot\mathbf{v}) && (3)
\end{aligned}
$$

where $\mathrm{d}/\mathrm{d}t$ is the material derivative, γ is the specific heat ratio, and the state variables $\rho, \mathbf{v}$ and u correspond to the mass density of the fluid, its velocity and its thermal energy per unit mass, respectively. The equation of state linking the pressure p with these variables is usually chosen to be the ideal gas law:

$$p = (\gamma - 1)\ u\ \rho \tag{4}$$

The SPH method replaces the continuous system with a set of point particles with specific values of the fields $\rho_a, \mathbf{v}_a$ and u_a with -a- the numbering index for all the particles in the simulation. The smoothing approximation in SPH means that the values of these fields are not Dirac-δ functions centred at $\mathbf{r}_a$, but they rather spread out *around* it: They are smooth particles. The spreading is not homogeneous, in fact it is concentrated at the centre and decreases radially outwards. This radial dependence is given by the weight function W (with support radius H_a) known as the interpolating kernel:

$$W(|\mathbf{r}_a - \mathbf{r}_b|, H_a) = \frac{C}{(H_a)^d}\mathcal{K}\left(\frac{|\mathbf{r}_a - \mathbf{r}_b|}{H_a}\right) \tag{5}$$

where C is a constant which guarantees that W is normalised to unity, d is the dimensionality of the problem, and $\mathcal{K}$ is usually chosen from a family of functions known as the Wendland Kernels [17]. Throughout this manuscript we use the 4th order Wendland Kernel (WC4 for short)[1] defined as:

$$\mathcal{K}_{\mathrm{WC4}}(q) = \Theta(1-q)\begin{cases} (1-q)^5\left(8q^2+5q+1\right) & d=1 \\ (1-q)^6\left(\frac{35q^2}{3}+6q+1\right) & d=2 \bigvee d=3 \end{cases} \tag{6}$$

[1] For the WC4 kernel, the values of C are $\{3/2, 9/\pi, 495/32\pi\}$ for the 1D, 2D and 3D cases respectively.

where $\Theta(q)$ the Heaviside-Step function:

$$\Theta(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{7}$$

A system of equations analogous to 1, 2 and 3 can be derived for the set of smooth particles. The equations obtained are discrete and conserve the total mass of the system, linear and angular momenta and the total energy down to the precision of the time-integration algorithm. A formal derivation of the equations can be found in [3], [7] or [23], we summarise the main results:

$$\begin{aligned}
\rho_a &= \sum_{b \in \mathcal{N}_a} m_b W_{ab}(H_a) &(8)\\
\frac{\mathrm{d}\mathbf{v}_a}{\mathrm{d}t} &= -\sum_{b \in \mathcal{N}_a} m_b \left(\frac{\mathcal{S}_a}{\rho_a^2} \frac{F_{ab}(H_a)}{\Omega_a} + \frac{\mathcal{S}_b}{\rho_b^2} \frac{F_{ab}(H_b)}{\Omega_b} \right) \cdot \mathbf{r}_{ab} &(9)\\
\frac{\mathrm{d}u_a}{\mathrm{d}t} &= -\sum_{b \in \mathcal{N}_a} m_b \frac{p_a}{\rho_a^2} (\mathbf{v}_{ab} \cdot \mathbf{r}_{ab}) \frac{F_{ab}(H_a)}{\Omega_a} &(10)
\end{aligned}$$

with $\mathbf{r}_{ab} = \mathbf{r}_a - \mathbf{r}_b$, $W_{ab}(H_a) = W(|\mathbf{r}_a - \mathbf{r}_b|, H_a)$ and $F_{ab}(H_a) = F(|\mathbf{r}_a - \mathbf{r}_b|, H_a)$. Also, F, Ω and the tensor $\mathcal{S}$ are defined as usual:

$$\begin{aligned}
\nabla W_{ab}(H_a) &= -\mathbf{r}_{ab} F_{ab}(H_a) &(11)\\
\mathcal{S}_a &= -p_a \mathbb{I} &(12)\\
\Omega_a &= 1 + \frac{H_a}{d\rho_a} \sum_{b \in \mathcal{N}_a} m_b \frac{\partial W_{ab}(H_a)}{\partial H} &(13)
\end{aligned}$$

Equations 1, 2 and 3 do not, however, include dissipative effects, which is why equations 8, 9 and 10 exclude such dynamics too. The inclusion of dissipative terms comes usually via viscosity ν, friction χ and conductivity κ:

$$\left.\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}\right|_{\mathrm{diss}} = \nu \nabla^2 \mathbf{v} - \chi \mathbf{v} \qquad \left.\frac{\mathrm{d}u}{\mathrm{d}t}\right|_{\mathrm{diss}} = \kappa \nabla^2 u \tag{14}$$

A discrete version of these terms is also available in SPH:

$$
\begin{aligned}
\left(\nu\nabla^2\mathbf{v}\right)_a &= -\sum_{b\in\mathcal{N}_a} m_b\ \alpha_b^v\ v_{sig}^v\ \left(\mathbf{v}_{ab}\cdot\hat{\mathbf{r}}_{ab}\right)\ \frac{r_{ab}\overline{F_{ab}}}{\overline{\rho_{ab}}}\ \hat{\mathbf{r}}_{ab} && (15)\\
\left(\chi\mathbf{v}\right)_a &= \chi\mathbf{v}_a && (16)\\
\left(\kappa\nabla^2 u\right)_a &= \sum_{b\in\mathcal{N}_a} m_b \left(\alpha_b^v\ v_{sig}^v\ \frac{(\mathbf{v}_{ab}\cdot\hat{\mathbf{r}}_{ab})^2}{2} - \alpha_b^u\ v_{sig}^u u_{ab}\right)\frac{r_{ab}\overline{F_{ab}}}{\overline{\rho_{ab}}} && (17)
\end{aligned}
$$

where:

$$\overline{\rho_{ab}} = \mathrm{avg}\{\rho_a,\rho_b\} \qquad \overline{F_{ab}} = \mathrm{avg}\{F_{ab}(H_a), F_{ab}(H_b)\} \tag{18}$$

and $\hat{\mathbf{r}}_{ab}$ is the unit vector joining particles a and b. The speed at which signals propagate in the fluid v_{sig} is taken as the fastest velocity present in the system, and the coefficients α^v and α^u play the roles of viscosity and conductivity (ν and κ resp.) and can be kept constant or can be set to evolve, change and adapt as required by the problem. A good model for the temporal evolution of these α-coefficients can be found in [12], [24], or [25][2]. Equations 8, 9 and 10 together with the dissipative terms 15, 16 and 17 and the evolution equations for the α-coefficients constitute the main result in the SPH model.

[2] The evolution is usually given by a source-sink model:

$$\frac{d\alpha}{dt} = S - \frac{\alpha-\alpha_{\min}}{\tau} \tag{19}$$

where τ is a characteristic decay time, $\alpha_{\min}$ is the minimum value towards which α relaxes, and S is a source terms indicating where the dissipation should be applied. Usual choices for this terms are:

$$S_a^v = \max_b\{0, -\nabla\cdot\mathbf{v}_a\} \qquad S_a^u = \frac{H_a|\nabla^2 u|_a}{2\sqrt{u_a+\epsilon}} \tag{20}$$

with the divergence, and laplacian operator given by their SPH counterpart (See [21] for details):

$$
\begin{aligned}
(\nabla\cdot\mathbf{v})_a &= \sum_b \frac{m_b}{\rho_b}\mathbf{v}_{ab}\cdot\mathbf{r}_{ab}F_{ab}(H_a) && (21)\\
(\nabla^2 u)_a &= -2\sum_b \frac{m_b}{\rho_b}(u_a-u_b)\frac{F_{ab}(H_a)}{r_{ab}} && (22)
\end{aligned}
$$

1.3. The problem of Initial Conditions in SPH

The SPH equations lay down the rules for the time evolution of the state variables but say little about the initial condition of such fields. Some fields, like the velocity or the thermal energy, can be defined at will, but when it comes to the mass density ρ, subtleties appear and the problem becomes non-trivial, at least in its most general formulation where geometries and density profiles are arbitrarily complex. These subtleties can be better captured by looking at the SPH density estimator in Eq.8:

$$\rho_a = \sum_{b \in \mathcal{N}_a} m_b W_{ab}(H_a) \tag{23}$$

where the summation runs over the set $\mathcal{N}_a$ formed by all the particles b at a distance H_a or less from particle a - i.e. over its neighbours.

Using this estimator, we see how ρ_a does not only depend on its own mass m_a, but also on the masses of the particles around, and more importantly perhaps, on the relative positions[3] from its neighbours.

Thus, the problem of finding an initial distribution of particles consistent with a prescribed density field translates into an inverse problem where both the mass *and* the position of every particle have to be found knowing only the desired density profile ρ_a; solving this -efficiently- is not a trivial task.

1.4. Short review of solutions to the density initialisation problem

Over the years several methods have been proposed that, one way or another, circumvent this issue. These approaches either take advantage of symmetries inherent in the system, or rely on approximations that might not strictly hold, but lie within the approximation errors of the SPH formalism.

[3] The dependence of ρ on H is only apparent since H is completely defined by the particle positions. Details will be discussed later, but can also be found in [3], or [12].

1.4.1. Variable mass methods

The first kind of attempt is to make use of the m_a coefficients in the density estimator in Eq.23 to obtain any desired profile. The idea is to begin with a crystalline structure with homogeneous particle density, that is, a structure where:

$$n_a = \sum_{b \in \mathcal{N}_a} W_{ab}(H_a) = n = \text{constant} \tag{24}$$

and simply choose the appropriate mass m_a for each particle to obtain the desired mass density profile: $\rho_a = m_a n_a$. This approach, although intuitively correct, has a few disadvantages: First, it implicitly assumes the following approximation to hold:

$$\rho_a = \sum_{b \in \mathcal{N}_a} m_b W_{ab}(H_a) \approx m_a \sum_{a \in \mathcal{N}_a} W_{ab}(H_a) \tag{25}$$

which is only valid in the case of slowly varying, or constant, mass distributions; Secondly, it can lead to interactions between particles of different masses, which, as noted by others (See [26] for example), produces undesired numerical artefacts. For that reason, most modern implementations avoid the "different mass" approach.

With this in mind, our task transforms into *placing N particles of equal mass such that the SPH density estimator best approximates a density distribution of our choice.*

1.4.2. Remapping or Stretching methods

The second kind of solutions (see e.g.[26] and [27]) can be used in problems with high degrees of symmetry: usually cylindrical, spherical, or in general, a density profile depending on only one variable (see [12]). The idea is to create a radial mapping from an homogeneous, crystalline configuration (where ρ_0 is constant), to a profile with a definite radial dependence $\rho^\star(r)$. The crystal positions r_0 are mapped into the desired positions r via a radial transformation

$r = \xi(r_0)$ defined by:

$$\frac{d\xi}{dr_0} \propto \frac{\rho_0}{\rho^\star}\left(\frac{r_0}{r}\right)^{d-1} \tag{26}$$

with d the dimensionality of the problem and subjected to the boundary conditions: $\xi(0) = 0$ and $\xi(R) = R$. The integration of Eq.26 can be carried out both analytically or numerically given the complexity of the problem at hand.

A different implementation can be found in [12] and [28] where, starting on one end of domain $L = [x_{\text{min}}, x_{\text{max}}]$, the interval is scanned to find the adequate position for each particle while ensuring one thing: *the particles should maintain their relative position in the mass distribution function.* The implementation of this scanning comes down to finding the root of the function:

$$f(x) = \frac{\int_{x_{\text{min}}}^{x} \rho^\star(x)\ dx}{\int_{x_{\text{min}}}^{x_{\text{max}}} \rho^\star(x)\ dx} - \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \tag{27}$$

where the solution x will determine the position of the particle. It must be noticed that in spherical or cylindrical problems where the initial configuration is a cartesian crystalline structure, the remapping creates numerical artefacts and the results are not adequate as a suitable initial condition.

The stretching method does not work in geometries with fewer symmetries. However, within the limited cases where the method does works it seems to give fast results whose errors scale down with the number of particles as shown by Fig.1.

1.4.3. Relaxation methods

Another alternative that has been used throughout the history of SPH is that of a relaxation "stage" prior to the simulation as documented in [29] and [30]. The idea behind is to simulate usual SPH dynamics under two artificial effects: That of an external potential and heavy damping. The first allows the particles to replicate arbitrarily complex density profiles in virtually any geometry, and the second is put forth to accelerate that process by damping the dynamics.

These relaxation techniques can deliver low-noise results but are well known to be time consuming and usually only viable when the net forces are expected

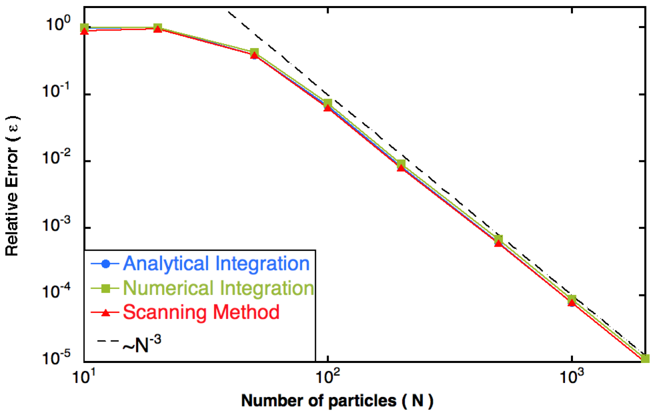

Figure 1: Scaling of the relative error ε with the number of particles N. The target density profile used was $\rho^{\star}(r) = \exp(-5r)$ in the interval $[0, 1]$. The stretching map was calculated by integrating analytically and numerically Eq.26 and using the scanning method as well. We see an N^{-3} scaling in all three methods for values of N up to 2000 particles where it attains a relative error of $\varepsilon = 10^{-5}$. The precise definition of ε is given in section.3.

to vanish at $t = 0$. The ALARIC algorithm, which will be presented shortly, encompasses the ideas of these typical relaxation methods and builds on them so as to retain the low-noise results and reduce the computational time. In a later section a comparison is offered to demonstrate the relative improvement of ALARIC with respect to typical relaxation schemes.

1.4.4. Inertia-Free methods

Another option is the Weighted Voronoi Tessellation, or WVT, method introduced by Diehl et. al. [28]. The idea behind the method is to calculate particle displacements *directly* from the proximity of each particle to its neighbours. The idea relies on using the forces acting over the particles to construct the immediate velocity instead of the acceleration[4]

[4] This can be identified with the assumption of a low Reynolds number where viscous effects are overwhelmingly dominant (See [31] for example) or from a more classical perspective,

1.5. Introducing ALARIC: An ALgorithm for ARbitrary Initial Conditions.

The ALARIC method presented in this manuscript builds on the relaxation algorithms and the ideas behind the inertia-free methods to deliver a robust tool to construct low-noise and arbitrarily complex density distribution in a relatively short time. ALARIC emerges almost naturally from the standard set of SPH equations which, as a result, can be easily included into any code without having to redesign it entirely.

The main idea behind our algorithm is to use the very same SPH equations described in the previous section to calculate hydrodynamic forces but includes 1.) an external potential to drive the particles to the desired density configuration, 2.) friction and viscosity, to strongly damp their motion towards the sought final distribution, 3.) sensible criteria for artificially nullifying the velocities of all the particles - which effectively stops low-k oscillations and reduces the inertial forces of the system to zero, 4.) a particle-splitting algorithm (reminiscent of a multi-grid approach in mesh-based numerical methods) which efficiently converges the slow low-k features first, when the number of particles is small, and the fast high-k features last, when the number of particles is big and viscous mechanisms dominate their motion.

This paper is organised as follows: Section 2 describes the equations be-

as a world where Aristotelian Physics is correct.. Hence our classification as an Inertia-Free method. The particle displacement is directly calculated by accumulating the weighted forces due to its closest neighbours:

$$\Delta \mathbf{r}_i = \mu H_i \sum_{j \in \mathcal{N}_i} W(|\mathbf{r}_i - \mathbf{r}_j|, H_i) \tag{28}$$

Where μ is a free[5] parameter that represents the strength of the interaction and the function $W(r, H)$ represents the weight of each contribution. Dielh et. al. worked with a weight that resembled a repulsive r^{-2}-type of force with the suggestion to replace it with the SPH kernel. This suggestion however was never implemented in their paper.

The method was tested thoroughly and offered converged results in about 100 iterations. The results were tested by setting 8000 particles of mass 1 and equal H-field to reproduce uniform and non-uniform density profiles. They measured the interpolation accuracy and the particle noise showing the WVT outperforms all the other methods considered in the paper[6].

hind ALARIC, including a detailed exposition of its main ingredients. Section 3 touches certain topics related to the quality of the converged solution. Numerical results that illustrate the efficiency of the proposed methodology are presented in Section 4. Section 5 tests ALARIC against state-of-the-art methods, and includes a discussion on how the initial conditions obtained with ALARIC compare against purely crystalline lattices in a point-explosion test. Finally, conclusions are presented in Section 6.

2. ALARIC's main ingredients

2.1. External Potential

The particle initialization method proposed by us introduces a fictitious potential ψ that comes into the equations as a pressure field. The effective force felt by the particles is:

$$\nabla p_{\text{eff}} = \nabla (p - \psi) \tag{29}$$

The fictitious potential is constructed from the *desired* density profile $\rho^\star$ through the assumed equation of state:

$$\psi(\mathbf{r}) = (\gamma - 1)\ u\ \rho^\star(\mathbf{r}) \tag{30}$$

The best results for our algorithm have been obtained by using an iso-thermal model of the fluid. This means the internal energy of each particle u_a will remain constant throughout the run and is usually initialised to be constant in space. Since we are restricting ourselves to the "equal mass" approach the mass density ρ_a can be computed from the node density n_a. Under the "equal mass" approach, both fields H_a and n_a (in a d-dimensional scenario) are usually coupled via the equations:

$$\begin{aligned} n_a \times (H_a)^d &= \text{const.} && (31) \\ n_a &= \sum_{b \in \mathcal{N}_a} W_{ab}(H_a) && (32) \end{aligned}$$

where the constant in equation 31 is usually a measure of the number of neighbors that every particle interacts with. This system of equations can be solved self-consistently once the positions of the particles are known, meaning the fields H and n should not be regarded as new variables, but rather as intrinsic, or natural, properties of any set of points.

2.2. Friction and Viscosity

If friction and viscosity are present in the system, the appropriate SPH equation of motion are given by:

$$\frac{\mathrm{d}\mathbf{v}_a}{\mathrm{d}t} = \left(\frac{\mathrm{d}\mathbf{v}_a}{\mathrm{d}t}\right)_{\text{ideal}} + \left(\frac{\mathrm{d}\mathbf{v}_a}{\mathrm{d}t}\right)_{\text{visc}} + \left(\frac{\mathrm{d}\mathbf{v}_a}{\mathrm{d}t}\right)_{\text{fric}} \tag{33}$$

where each individual term is given by:

$$\begin{aligned}
\left(\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}\right)_{\text{ideal}} &= -\sum_{b\in\mathcal{N}_a} m\left(\frac{\mathcal{S}_a}{\rho_a^2}\frac{F_{ab}(H_a)}{\Omega_a} + \frac{\mathcal{S}_b}{\rho_b^2}\frac{F_{ab}(H_b)}{\Omega_b}\right)\cdot\mathbf{r}_{ab} && (34)\\
\left(\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}\right)_{\text{visc}} &= -\sum_{b\in\mathcal{N}_a} m\ \alpha_b\ v_{sig}\ (\mathbf{v}_{ab}\cdot\hat{\mathbf{r}}_{ab})\ \frac{r_{ab}\overline{F_{ab}}}{\overline{\rho_{ab}}}\ \hat{\mathbf{r}}_{ab} && (35)\\
\left(\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}\right)_{\text{fric}} &= -\chi\mathbf{v}_a && (36)
\end{aligned}$$

Where the field α has been set to evolve according to the equations presented in [24] and [25][7]. Finally, the numerical value for the friction coefficient χ that offered optimal results was $\chi = 2$.

2.3. Sensible criteria for nullifying the velocities

Its important to keep in mind that the main goal of ALARIC is to obtain a suitable state to be used later as an initial condition in an SPH simulation. It is altogether unimportant how that solution is reached. This means we are free to temper with the evolution of all and every state variable at will, as long

[7] With the numerical value for $\alpha_{min} = 0.0$

as we can make our simulation converge faster. With this in mind, we have found that nullifying the velocities of the particles every certain time interval τ helps to avoid low-k oscillations which would be otherwise essentially immune to viscous dynamics.

An adequate interval τ can be found by choosing the shortest of the following two characteristic times:

Crossing time (τ_{cross}). This characteristic time is estimated using a characteristic speed, like the speed of sound for example, and a characteristic length coming from the geometry of the computational domain:

$$\tau_{\text{cross}} = \frac{L}{c_s} = \frac{L}{\sqrt{\gamma \frac{P}{\rho}}} \tag{37}$$

Near Equilibrium (τ_{NE}). A second characteristic time[8] is estimated as the time interval $[0, \tau_{\text{NE}}]$ such that:

$$E_k(\tau_{\text{NE}}) > E_k(\tau_{\text{NE}} + \Delta t) \tag{38}$$

where E_k is the total kinetic energy of the system:

$$E_k = \sum_{a=1}^{N} \frac{m v_a^2}{2} \tag{39}$$

The shortest of these two characteristic times is used to provide a sensible time interval τ at the end of which we will bring all the particles to a sudden stop by setting their velocities to zero.

$$\tau = \min\{\tau_{\text{cross}}, \tau_{\text{NE}}\} \tag{40}$$

In spite of the use of velocity nullification to accelerate the convergence, there might be scenarios that still take excessively long to converge. Some of

[8] This choice was inspired by a swinging pendulum whose kinetic energy is maximum *precisely* at the point where the pendulum position coincides with its equilibrium position. Although the degrees of freedom in a typical SPH system are considerably larger than that of a simple pendulum, this criteria has offered relatively good results, as will be shown in the following section.

them seem to approach a regime characterised by oscillations of the density distribution around the desired profile whose period can be close to τ and therefore the constant nullification of the velocities becomes redundant and ineffective.

To avoid this, a small change to the characteristic time has been implemented: to increase τ every time the particles are brought to a sudden stop:

$$\tau_{new} = (1 + \theta)\tau_{old} \tag{41}$$

Optimal results have been obtained in the examined cases, using $\theta = 0.1$.

2.4. Particle Splitting

The last ingredient in ALARIC is inspired by multi-grid approaches commonly used in mesh-based numerical methods. The idea is to first pursue a quick convergence of the coarse, low-k, features of $\rho^\star$ with a fraction of the desired number of particles N, denoted by N_0. Once this simulation has converged to our desired profile $\rho^\star$, each particle is then replaced by a set of p new particles positioned at $\mathbf{r}_a + \mathbf{q}_{a,i}$ with $i \in \{1, .., p\}$. We are free to choose the precise geometry of the newly created p particles. However, in our experience, good results have been obtained using the geometric arrays depicted in Fig. 2 where the arrows depict the displacement vectors $\mathbf{q}_{a,i}$. They start at the position of the old particle and end at the positions of the new particles.

Their *orientation* follows neatly organised patters: left/center/right in 1D, a triangle with its center occupied in 2D and a tetrahedron, also with its center occupied (similar to a methane molecule), in the three-dimensional case[9].

The *magnitude* of such displacements is chosen so as to avoid overlapping with other, newly-created, particles that arise from neighbouring old particles. This can be accomplished by setting the magnitude of every $\mathbf{q}_{a,i}$ proportional to the local inter-particle-spacing which can be obtained by scaling down the support radius H_a of the old particle. For a d-dimensional scenario we propose:

[9] These arrangements imply the values for $p = \{3, 4, 5\}$ for the 1, 2 and 3 dimensional cases respectively.

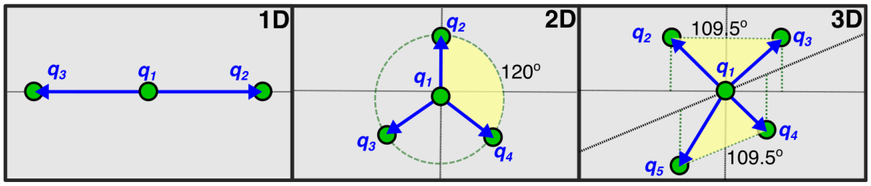

Figure 2: Schematic representation of the geometric arrays used during the particle-splitting algorithm. Shown arrays for 1, 2 and 3 dimensional cases.

$$\|\mathbf{q}_{a,i}\| = \frac{2H_a}{\sqrt[d]{p}\ N_H} \qquad \forall i \in \{1, \cdots, p\} \tag{42}$$

where N_H is the approximate number of neighbours each particle interacts with[10]. This splitting can be continued, i-times for example, until the particles are abundant enough to meet our initialization requirements.

3. Quality of solution

In this brief section we address two important questions about the quality of the density profile that we encounter at the end of the simulation. First, we propose a measure of the agreement between the current density profile and the desired density profile, and secondly, we present the necessary conditions to ensure that such measure will, indeed, tend to zero as the two profiles approach each other.

3.1. Ensuring convergence

The use of ALARIC will drive our system to a state of equilibrium, that is, a state where the velocities are zero, and the forces between the particles are balanced: $\nabla p_{\mathrm{eff}} = (\gamma - 1)\ u\ \nabla(\rho - \rho^{\star}) = 0$. This condition is fulfilled whenever the following condition holds true:

$$\rho - \rho^{\star} = \rho_0 = \text{constant} \tag{43}$$

[10] This formula can be easily obtained by considering the volume element assigned to the old particle, and splitting it into p, smaller, fragments.

Unless ρ_0 equals zero our density will not be equal to the desired density. To make sure this *is* the case, the mass m of the particles is being re-calculated every time the velocities are nullified. We set it to be equal to:

$$m = \frac{1}{N} \sum_a \frac{\rho_a}{n_a} \tag{44}$$

This choice for m ensures $\rho_0 = 0$ in the final state and allows our system to evolve to a state of equilibrium characterised by a mass density $\rho = \rho^\star$.

3.2. Monitoring the Errors: ε and $\dot{\varepsilon}$

Evolving the system using the equations implemented in ALARIC will, in due time, converge to the sought initial state characterised by $\rho = \rho^\star$. The quality of the converged state can be monitored in time using:

$$\varepsilon = \frac{\int \|\rho(\mathbf{r},t) - \rho^\star(\mathbf{r})\|^2 dV}{\int \|\rho^\star(\mathbf{r})\|^2 dV} = \frac{\sum_{a=1}^{N} (\rho_a - \rho^\star(\mathbf{r}_a))^2 / n_a}{\sum_{a=1}^{N} \rho^\star(\mathbf{r}_a)^2 / n_a} \tag{45}$$

Ideally, the simulation must be run until ε falls below a certain tolerance level. In some cases, however, our limited number of particles impedes us to reach such tolerances, hence it is wise to consider an alternative threshold derived from the derivative of the error: $\dot{\varepsilon} = \mathrm{d}\varepsilon/\mathrm{d}t$. This threshold helps detect cases where the time trace of ε stagnates and becomes flat ($\dot{\varepsilon} \approx 0$). The reasons behind this flattening will be discussed in the following sections and in the Appendix.

It must be noticed that the geometric patterns used during the particle splitting shown in Fig.2 are not perfect and can introduce a small amount of noise into our ε-trace. This can set back our error estimate several orders of magnitude trashing our efforts for a faster convergence and making the use of particle-splitting counterproductive. It is thus wise, not to use too many stages during the simulation and not to set error tolerances too low during the intermediate stages of the multi-grid simulation.

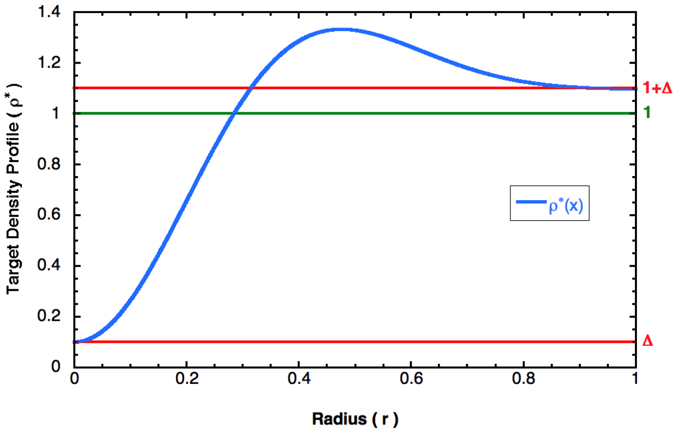

Figure 3: Illustration of the target function $\rho^\star$ and the meaning of the parameter Δ. The value $\Delta = 0.1$ was used for this plot.

4. Numerical Results

We proceed now to discuss several examples that illustrate the power of the proposed methodology and explore some of its more salient features.

4.1. Impact of ALARIC's ingredients on the convergence.

To illustrate the performance of our method in a one-dimensional case, we will use 300 particles to recreate the following density profile:

$$\rho^\star(r) = (1 + \Delta) - \cos(5r)\exp(-5r^2) \tag{46}$$

The shape of $\rho^\star$ and the meaning of Δ are both shown in Fig.3. We proceed by initialising the particles with a flat and homogeneous density and we let them evolve using our modified SPH dynamics. The threshold on the density error has been set to $\varepsilon_{\text{min}} = \dot{\varepsilon}_{\text{min}} = 10^{-8}$.

Fig.4 shows the time traces of the density error in eight different runs with $\Delta = 0.0$. Each run includes a new combination of ALARIC's ingredients to its dynamics: The first run is a reference run, it simply shows the evolution of the system under the effect of the external potential, but no viscosity is present, no

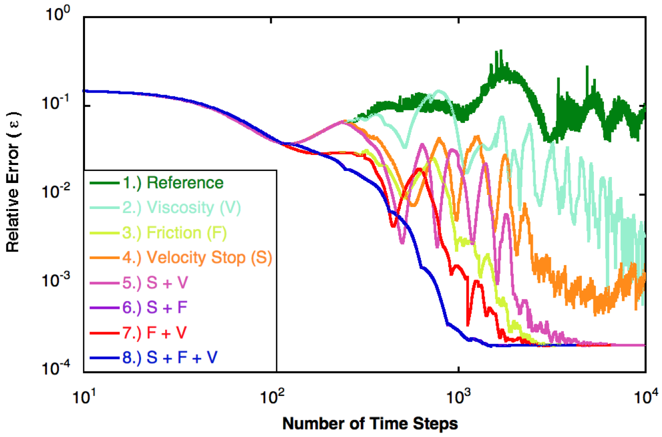

Figure 4: A log-log plot of the time traces of ε for 8 different simulations. Every simulation represents a different combination of the available ingredients.

friction, and the velocities are not being nullified. The impact of ALARICs last ingredient, the particle splitting, will be studied in subsection 4.2.

Runs 2, 3 and 4 include viscosity, friction and velocity stopping separately. We see how the time trace of the density error in each case decreases much faster than the reference case. In particular, we can see how viscosity alone softens the ε-trace while friction alone, although faster, has a noisier ε-trace. Run 3 corresponds to the implementation of a typical relaxation algorithm as described in [29].

Runs 5, 6 and 7 combine the ingredients in pairs, while run 8 combines all the ingredients at once. We see how runs 6 and 8 completely overlap, in fact, its quite difficult to notice their individual traces. Runs 6 and 8 performs better than any other combination so far. The reason behind their similarity comes from the fact that the added viscosity in run 8 softens the time-trace of ε, but has little effect on the overall convergence.

Table 1 summarises the number of time-steps required in each case to converge, the real computational time spent in each simulation, and the residual error ε left at the end of the run. The simulations have been ranked from fastest to slowest.

Run	**TimeSteps**	**CPU time** [s]	**Residual** ε
Reference	> 10000	> 100	0.1700
Viscosity (V)	> 10000	> 100	0.0034
Velocity Stop (S)	> 10000	> 100	0.0013
S + V	> 10000	> 100	0.0002
Friction (F)	6841	64	0.0002
F + V	6486	78	0.0002
S + F + V	4218	49	0.0002
S + F	4219	36	0.0002

Table 1: Performance comparison between the 8 runs.

We can clearly see how the extra computation of the viscosity term in run 8 makes every time-step a little bit slower than run 6. On the other hand, the use of viscosity softens, in general, the trace of the density error making easier the identification of its convergence through the $\dot{\varepsilon}$ threshold. To better appreciate this effect Fig.5 presents the traces of the time derivative of the error traces ($\dot{\varepsilon} = \mathrm{d}\varepsilon/\mathrm{d}t$) for runs 6 and 8.

A question remains: Is the $(S+F+V)$ combination worth the extra computational time compared to the (S+F) approach? In our experience, the presence of dissipative dynamics, although slower, prevents the development of fast oscillations which do not crash the run but create excessive amount of noise which disrupts the convergence towards a relaxed state. We will use the $(S+F+V)$ dynamics on every run to come.

4.2. Impact of the particle-splitting on the convergence

In this section we illustrate the effect of splitting particles on the overall convergence process. To do so we will replicate the density profile $\rho^{\star}$ from equation 46 with 909 particles and $\Delta = 0.0$. We will do that in three different ways: Firstly, we will initialise 909 particles, and let them evolve with the (S+F+V)-

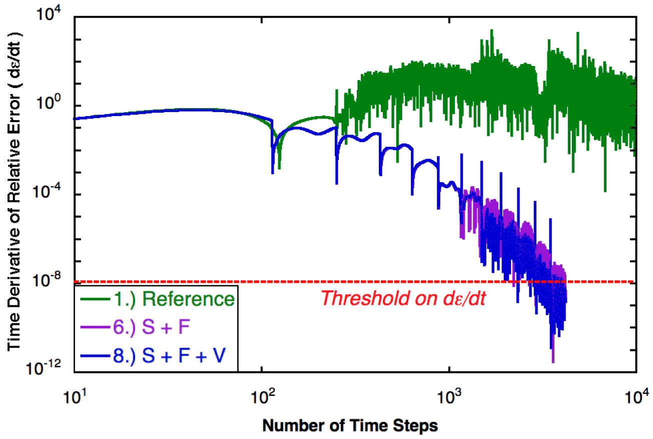

Figure 5: A log-log plot of the time traces of $\dot{\varepsilon}$ for 3 different simulations. The viscosity present in run 8 (S + F + V) makes the time trace of $\dot{\varepsilon}$ fall a little bit faster than that in run 6 (S + F). The effects of viscosity are more visible in the last stages of the simulation, where big displacements have already taken place, and only small inter-particle motion remains. For comparison, the time trace of $\dot{\varepsilon}$ in run 1 (Reference) is also offered. The time traces does not stop immediately after falling below the threshold because the condition for convergence is not evaluated at every time step, but every τ seconds.

dynamics. Secondly, we run a simulation that starts with 303 particles and is equipped with a 1-stage particle splitting. Thirdly, we initialise 101 particles but perform a 2-stage particle splitting. All three runs will end up with 909 particles in their final stages, and all of them are run until convergence is obtained. Fig.6 shows the time-trace of the three simulations.

An important phenomenon is illustrated here, the 1-stage run is indeed faster than the 0-stage one. Partially due to the longer time step allowed by a smaller number of particles before the splitting, and partially because viscosity is more efficient after the splitting when the number of particles is greater.

We can also see the effect of the noise introduced by the particle splitting. In fact, the noise created in the 2-stage run is big enough to send the ε-trace above the trace of the 1-stage run making it effectively slower. In conclusion, the noise created by the particle splitting is acceptable once: after the low-k features of the profile have been replicated, but not when the particles are approaching the delicate equilibrium characteristic of a converged run. We will run our scenarios with 1-stage particle splitting at most.

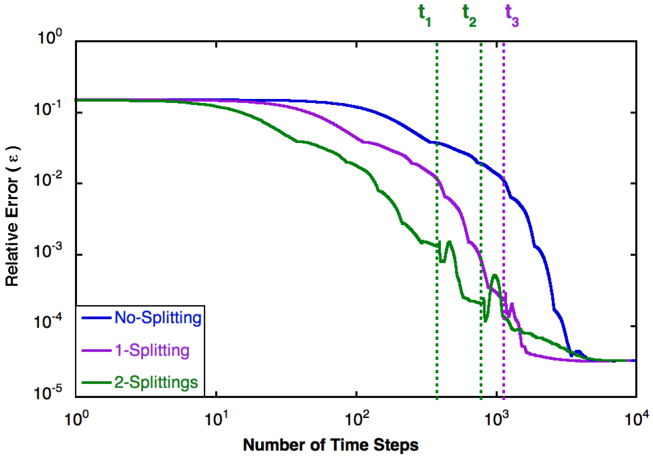

Figure 6: ε-traces for 3 different runs, each with different number of particle splitting stages. The dotted lines at times t_1 and t_2 help identify the precise moment when the particles multiply in the 2-Splitting simulation. The 1-Splitting simulation has only one splitting event at $t = t_3$. Although the fact that $t_2 < t_3$ presages a better performance of the 2-Splitting run in comparison with the 1-Splitting run, as the simulations evolve, we see how the time trace of the latter decays more rapidly. In this example, the use of more than one splitting event is counterproductive.

4.3. Impact of the parameter Δ on convergence.

An important feature of the desired density profile is the strong depression around the origin where it attains the value $\rho(0) = \Delta$. In this section we will use a 1-stage particle splitting approach to illustrate the effect that Δ has over the convergence. To do so, we run 6 different simulations with the following values for Δ:

$$\Delta = \left\{ 0,\ \frac{1}{100},\ \frac{5}{100},\ \frac{1}{10},\ \frac{5}{10},\ 1 \right\} \tag{47}$$

The runs begin with $N_0 = 101$ in the initial stage[11], and 303 in the final stage. The results are summarised in Fig.7.

By looking at the shape of the individual time traces we can categorise the time traces into two, well-defined, groups: The first group, composed by the runs with $\Delta \geq 0.1$ seems to converge using the ε criteria; This means that here the error estimator falls below the proposed threshold on $\varepsilon = 10^{-8}$. The second group, that where $\Delta < 0.1$, did not converge on ε but on $\dot{\varepsilon}$, meaning that the density error could not go further down and it flattened.

This behaviour hints at the existence of a lowest-possible error that the system can attain, and more importantly, it shows that this value seems to increase as Δ decreases - that is, it becomes harder to replicate scenarios with ever low-density regions.

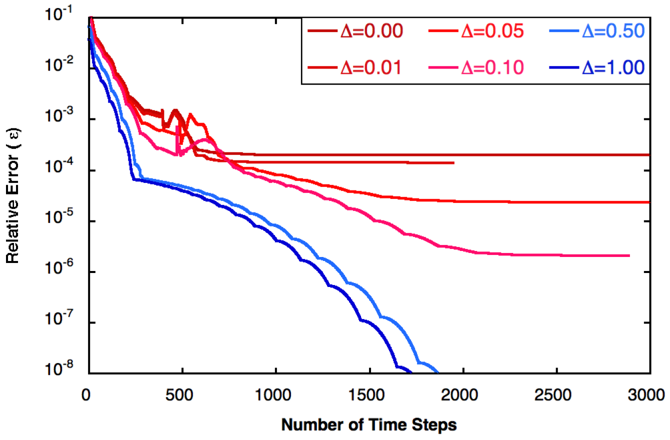

Figure 7: Time traces correspond to simulations with equal configurations but different values of the parameter Δ. The time traces show different behaviours which can be grouped into two distinct categories: Convergence in ε (For $\Delta > 0.1$) and convergence in $\dot{\varepsilon}$ (For $\Delta < 0.1$).

The smallest possible value of ε can be estimated using only a few assump-

[11] The reason for the odd choice of number of particles (101) has to do with ensuring that one particle always sits at the bottom of the density well, at $x = 0$.

tions (the details of the calculation can be found in the Appendix). Table 2 compares this theoretical estimate with the saturated values obtained form the flat lines in Fig.7.

Parameter Δ	$\varepsilon_{\text{teo.min}}$	$\varepsilon_{\text{stag}}$
0.00	$1 \cdot 10^{-4}$	$2 \cdot 10^{-4}$
0.01	$7 \cdot 10^{-5}$	$1 \cdot 10^{-4}$
0.05	$9 \cdot 10^{-6}$	$2 \cdot 10^{-5}$
0.10	$8 \cdot 10^{-7}$	$2 \cdot 10^{-6}$
0.50	$1 \cdot 10^{-9}$	$< 10^{-8}$
0.10	$8 \cdot 10^{-11}$	$< 10^{-8}$

Table 2: For different values of Δ we compare the values of the density error at which the simulation stagnates $\varepsilon_{\text{stag}}$ with a theoretical estimation of the lowest possible error attainable in such simulation $\varepsilon_{\text{teo.min}}$. Notice how the measured stagnation value of ε always falls above the theoretical minimum.

Table 2 clearly shows how the measured error is always above our theoretical minimum. The cases where a convergence in ε was achieved ($\Delta > 0.1$) the saturation value for the error must, logically, lie below the error threshold used, i.e. must be somewhere below 10^{-8} - Incidentally, the theoretical estimation for these cases also falls below this threshold.

As a final illustration of the behaviour of the relaxed particles around the low-density region we present in Fig.8 the radial dependence of the density profiles for 1, 2 and 3-dimensional runs. In every case we see that a single particle resides at the bottom of the density well. This peculiar final state is key in understanding our estimation of the lowest possible error estimation presented in the appendix and included in many of the graphs presented so far.

The theoretical lower limit on ε should intuitively be a function of the number of particles N and the parameter Δ. The calculation of the theoretical minimum for ε can be done through the following formula, whose derivation can be found

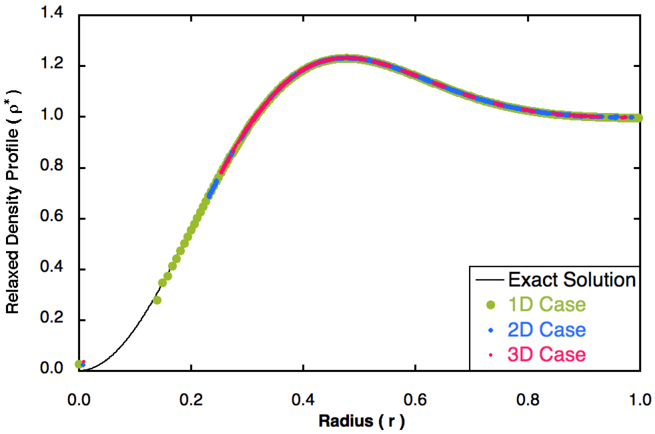

Figure 8: Radial Density profile for multidimensional simulations. The number of particles in the d-dimensional case obeys a scaling formula of the form $N = N_0 \cdot l^d$ where the coefficients N_0 and l have been arbitrarily set to $N_0 = 77$ and $l = 7$.

on the Appendix:

$$\varepsilon_{\text{teo.min}} = \frac{\mathcal{V}\lambda^d}{\mathcal{Q}} \left(\frac{m}{\mathcal{V}\lambda^d} - \Delta \right)^2 \tag{48}$$

Where $\mathcal{V}$ is a constant, $\mathcal{Q}$ depends on Δ, and m and λ depend both on Δ and N. To see the direct dependance of ε on N and Δ Fig.9 shows contour plots of equal ε for different values of N and Δ for the 1, 2 and 3-dimensional cases.

Fig.9 depicts a behaviour of ε that agrees with our preconceptions of the theoretical minimum: For a fixed value of Δ, increasing the number of particles allows us to lower the error in the estimation, and analogously, for a fixed number of particles, increasing the numerical value of Δ makes the density profile easier to be replicated, thus lowering the error estimate.

4.4. Flexibility of the external potential formulation

The use of an external potential finds its most salient advantage in its flexibility; We are not required to limit our expectations to density profiles with

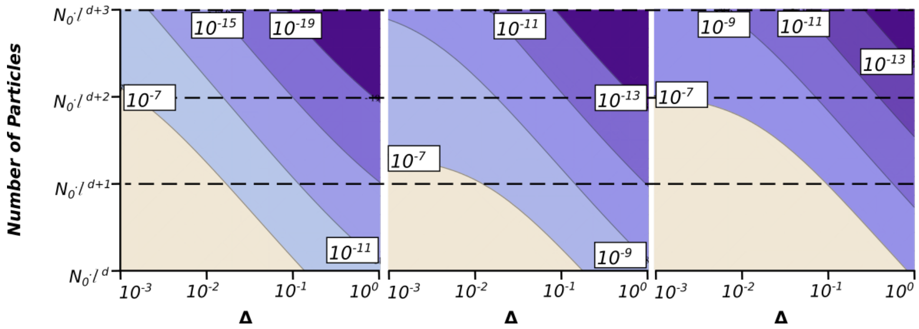

Figure 9: 1D(left), 2D(center) and 3D(right) contour plots of the theoretical minimum estimation $\varepsilon_{\text{tep.min}}(N, \Delta)$ in the domain $\{N_0 \cdot l^d, N_0 \cdot l^{d+3}\} \times \{10^{-3}, 10^0\}$. The coefficients N_0 and l have been arbitrarily set to $N_0 = 77$ and $l = 7$. We can see how $\varepsilon_{\text{teo.min}}$ decreases for higher number of particles, and increases for lower values of Δ, as expected.

high degrees of symmetry, but we are free to set our external potential using *any* desired density profile. As an example of this we will use 10K particles in a 2D unit-ball domain to replicate a desired mass density $\rho^\star$ given by:

$$\rho^\star(r, \theta \mid p) = \frac{1}{2} + 2r^2 \exp\left[-4\left(r - \frac{1}{5}\right)^2\right] \sin\left[p\left(\theta + \log r\right)\right] \tag{49}$$

with p a flexible parameter. The resulting scenarios for $p = \{1, 2, 3\}$ are shown in Fig.10 and Fig.11. Specifically, Fig.10 depicts the set of particles and illustrate how they gather around certain regions, and deplete others, to recreate the high and low density regions characteristic of $\rho^\star$.

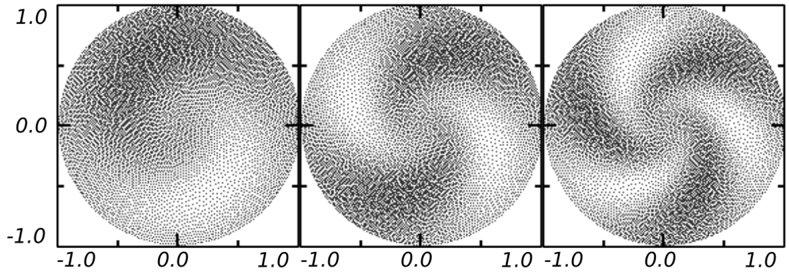

Figure 10: Particle mesh for the scenarios corresponding to $\rho^\star(r, \theta| \ 1)$, $\rho^\star(r, \theta| \ 2)$ and $\rho^\star(r, \theta| \ 3)$, respectively.

Fig.11 shows the resulting density field associated with the particle distribution presented in Fig.10. Here, the colour palette has been chosen to best separate the 'valleys' form the 'hills' of $\rho^\star$.

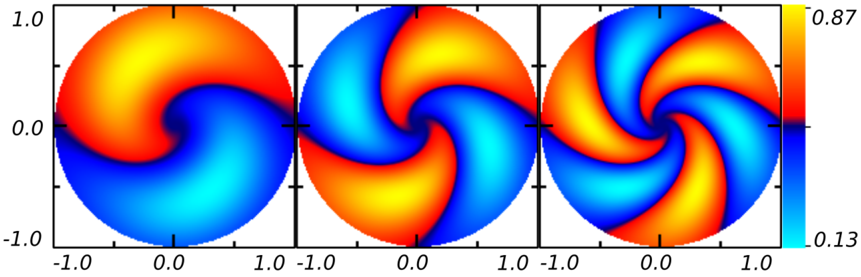

Figure 11: Density field for the scenarios corresponding to $\rho^\star(r,\theta|\,1)$, $\rho^\star(r,\theta|\,2)$ and $\rho^\star(r,\theta|\,3)$.

5. Discussion

5.1. Performance of ALARIC against inertia-free methods.

This section has the only purpose to compare ALARIC against what can be considered the state-of-the-art methods: The WVT method. We propose the test used in their paper [12] to illustrate their method. We use 2000 particles of equal mass to replicate a uniform density profile in a circular domain and compare the outcome of the WVT method and of ALARIC at different stages of the run. Initially, the particles are arranged in the same random configuration for each method. The comparison shots will take place at $TS = 1, 10, 100, 1000, 10000$ where TS is the number of time-steps. As mentioned before, Diehl et al. only implemented a version of their algorithm where the kernel used was of the r^{-2}-kind, here we will also present a version of the WVT algorithm that uses the SPH kernel. Finally we will compare also against a naively constructed inertia-free method coined AFC as in Away-From-Closest. The AFC method simply displaces each particle an amount $\mathbf{\Delta r}$ away from its closest neighbour. The strength of the displacement is also regulated by a free parameter μ which is linearly brought down to zero in the course of the run.

[12] In Fig.3 of Dielh et. al.[28]

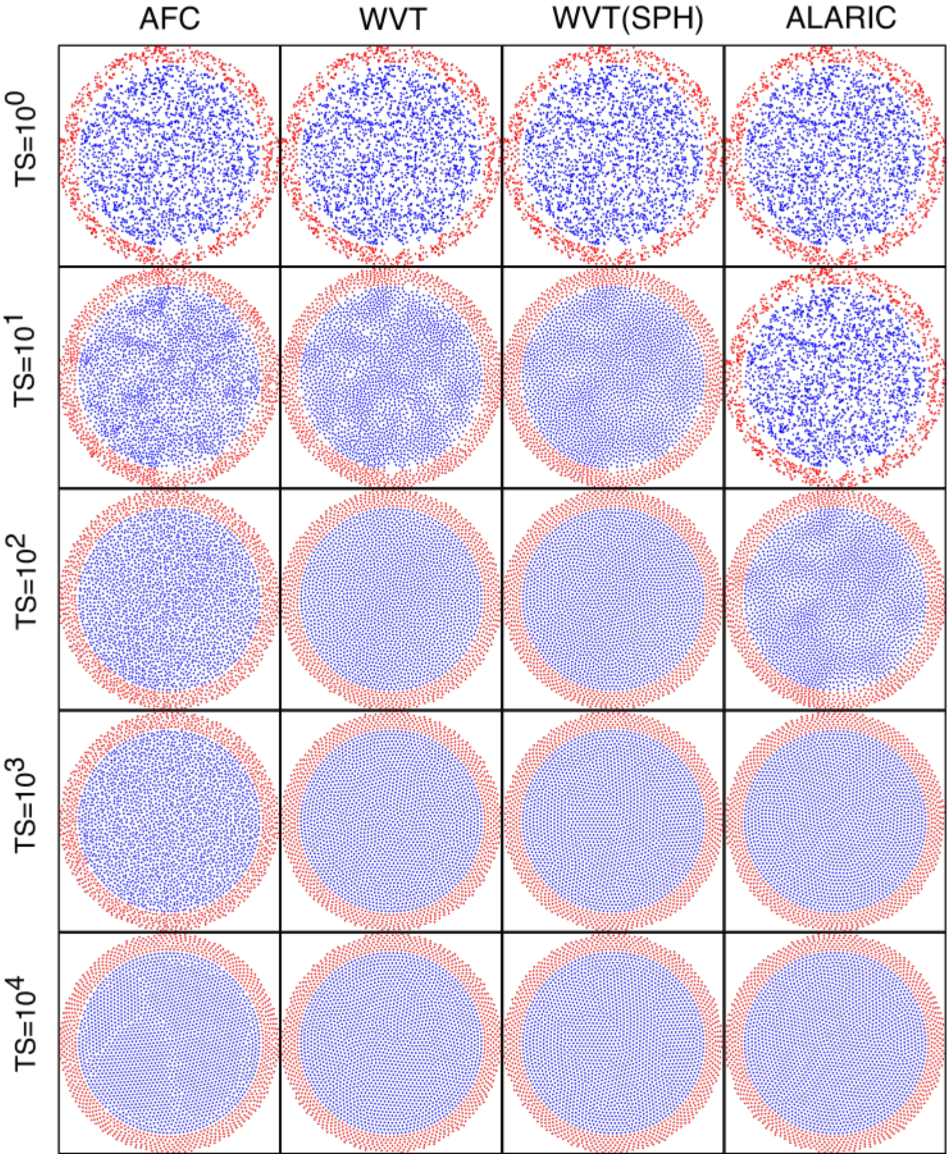

Figure 12: Particle configuration at different stages of the run ($TS = 1, 10, 100, 1000, 10000$) for the AFC method, the WVT method using both the r^{-2}-weight and the SPH kernel and ALARIC. It can be seen how, in less than 100 TS, the WVT couple already delivers reasonably homogeneous configurations. It is not until TS=1000 that ALARIC obtains similar results. The naively constructed AFC method is quite noisy and performs the poorest.

Fig.12 shows the successive snapshots of the particle configuration at different time-steps. It is clear to the naked eye how the WVT method delivers visibly homogeneous configurations as soon as TS=100. The AFC method is quite noisy and does not offer a sensible solution until the very end of the run. ALARIC obtains a reasonably homogeneous profile somewhere between $TS = 100$ and $TS = 1000$.

The end results of all the methods at $TS = 10000$ are almost indistinguishable to the naked eye with the exception perhaps of the AFC method where patches of hexagonally packed regions are separated by what can be considered grain boundaries.

In pursue of a more formal approach, Fig.13 offers the time-traces the relative error ε for each of the candidates throughout the whole run. We can clearly see the decisive advantages of each method. The WVT method is capable of reducing the relative error ε quite early in the run and reaches a minimum error of about 10^{-3} by the end of the run. The time-trace reveals how the minimum value of ε is actually achieved at around $TS = 1000$ and then it seems to stagnate, even though the parameter μ still decreases towards zero. ALARICs time-trace on the other hand takes some time to start decreasing, but by $TS = 100$ it has already matched the relative error of the AFC method, and by $TS = 200$ that of the WVT couple. ALARICs time-trace seems to stagnate after the first thousand time-steps with the difference that the error is much lower, approximately 10^{-5}, that is two orders of magnitude lower than the others. As expected the AFC offers a noisy time-trace which is always outperformed by the WVT methods.

This comparison shines light on the pros and cons of each method, we conclude the WVT to be a fast way of achieving reasonable results, while ALARIC offers a low-noise solution in twice or three times the number of time steps. In real computational time however, every time-step of ALARIC is almost twice as longer than in the WVT run since more terms have to be added and a proper time integrations takes place. All in all, ALARIC takes roughly ten times longer that the WVT method to converge but offers results whose relative error is one

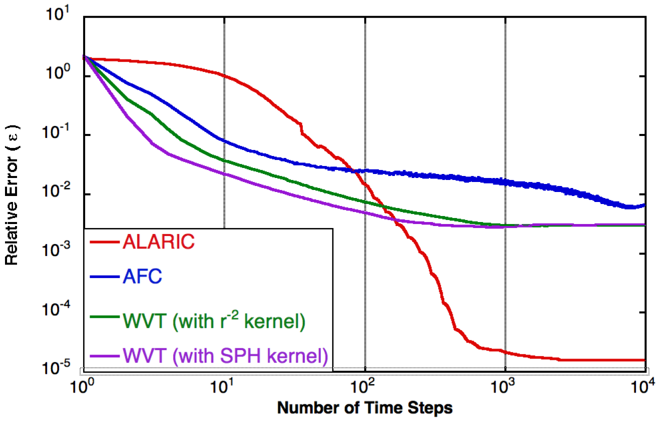

Figure 13: Time-traces of ε for all the methods.

hundred times smaller.

5.2. Performance of ALARIC-produced IC in a point-explosion test

This section will try to explore and compare how an initial condition obtained using ALARIC performs with respect to initial conditions created using perfect crystal lattices. To do so, we will run a point-like explosion using our full SPH engine and see how the shock-wave is affected by the inherent asymmetries of crystal lattices. We will use approximately 40K particles in a two dimensional square $[-0.5, 0.5] \times [-0.5, 0.5]$ with periodic boundary conditions, and will place a total amount of energy E_0 in the system distributed as follows: Half of the energy will be placed in particle $-a-$ which is chosen to be the closest to the origin $(0, 0)$, and the other half will be evenly distributed among *all* the other remaining particles. The pressure in the central particle p_a is much greater than its surroundings and it creates a shock-wave as it propagates radially outwards. The wave-front ought to be circular in shape since the surrounding media is *assumed* to be homogeneous and isotropic.

The initial condition consists of a static medium with homogeneous density $\rho = 1$. We take three approaches to replicate this configuration: The first two options place the particles along Cubic and Hexagonal Close-Packing lattices. The third option uses ALARIC to position 40K particles into a desired density $\rho^\star = 1$. Fig.14 shows the wave fronts of the point-like explosions at time $0.09s$. To better capture the asymmetry in the explosion we have decided to plot the velocity divergence field.

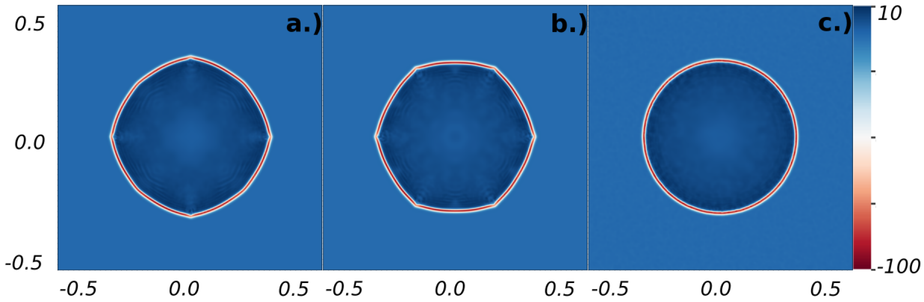

Figure 14: Snapshot at time $t = 0.09s$ of the point-like explosion test. We plot $(\nabla \cdot \mathbf{v})$ to better capture the shape of the wave-front. The three graphs correspond to the use of three different underlying crystal lattice in the initial condition: Cubic Lattice (a), Hexagonal-Close-Package (b) and a crystal lattice obtained by using ALARIC with a flat target density profile (c). We see how the crystalline structures (a) and (b) seem to have preferred directions of propagation while (c) is the only one who offers an homogeneous and isotropic wave propagation.

We can clearly see that the wavefront seems to propagate slightly faster along the planes of the crystalline structures (cases a and b) while retaining a much circular shape in the third case (case c).

Fig.15 shows the radial projection of both fields: the velocity divergence, and the mass density. In every case, horizontal broadening is due to asymmetries in the radial propagation of the wave, while vertical broadening is due to noisy fields.

We can see from both variables, how at the shock wave as well as behind it, a big horizontal broadening takes place in the first two cases (a and b), while the third case, case c, it remains 'thin' - that is, remains circular.

It must also be noticed that there is absolutely no vertical broadening throughout the space in front of the shockwave in cases (a and b) because the perfect crystalline structure remains untouched and delivers a clean estimate of

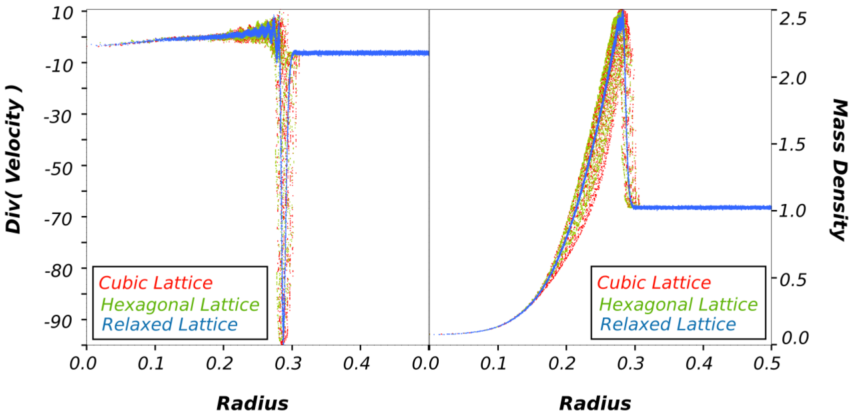

Figure 15: Radial projection of the $(\nabla \cdot \mathbf{v})$ and ρ fields, left and right respectively. The big, radially local, negative divergence (convergence) at the wave-front is a key aspect in capturing the shape and isotropic properties of the propagation.

the field, but there is a small noise in case *c* which overlaps and hides the radial profiles of the crystalline cases. The reason for this noise has to do with the quasi-crystalline state obtained with ALARIC.

6. Conclusions

ALARIC has been introduced as a simple and versatile tool to aid existing SPH codes in constructing tailored initial conditions of arbitrary complexity and exquisitely low-noise. The method consists of various ingredients which can be used individually if necessary. However, internal testing suggests that using the a.) iso-thermal SPH equations, b.) including dissipative dynamics through the introduction of friction and viscosity (with $\chi = 2$ and $\alpha_{\rm min} = 0.0$) terms, c.) nullifying the velocities every τ-seconds (with a 10% increase in τ every time its used) and d.) a one-stage particle-splitting event after the coarse features have been obtained, gave optimal results for -*all*- the scenarios presented in this manuscript.

Few new parameters were introduced during the exposition of the inner workings of ALARIC. Although specific numerical values were used in the tests presented here, the spirit of this paper is by no means limiting in this sense. Were the method proved slow or inefficient in future applications, the reader is encouraged to change and manipulate these values as he/she sees fit.

As an example, section 4.2 was devoted to the 1-stage vs. multi-stage particle splitting matter. It was concluded that more than one stage proved, not only inefficient, but to be even counterproductive to our end. The reason lies partially behind the amount of noise introduced by the splitting patterns presented in Fig.2. The specific shapes of these patterns were arbitrarily chosen and do not, in general, tesselate the d-dimensional space around them. This geometrical imperfection gives rise to an immediate reorganisation of the particles which introduces noise in the time trace of ε. The splitting patterns are different for the 1, 2 and 3-dimensional cases, and the amount of noise introduced in each case is different. In three dimensional simulations, for example, the noise introduced after the splitting is much bigger than in the 1D case. With this in mind, our intention has not been to limit the use of the particle splitting stages, but rather to warn the reader about some of its limitations - we do not rule out the existence of scenarios where 2 or more stages of particle splitting will be, indeed, the better option.

On the same token, we have seen ALARIC being initilised from many different particle configurations: Crystalline (1D-Equispaced in Sec.4.1 and Sec.4.2), quasi-crystalline (Concentric shells in Sec.4.4) and Random (Sec.5.1). In each case ALARIC performed quite well but we can not conclude which initialisation is in fact the most efficient. It has also been suggested (See [29] for example) to initialise the relaxation using a Monte Carlo method which will intuitively place the system close to the desired profile. Finally, with the introduction of the particle splitting scheme in ALARIC we can initialise the system with a significantly smaller number of particles which will replicate the broad features of the system quite readily. In fact, this number can be small enough that choosing to initialise it in a crystalline, quasi-crystalline, random or Monte Carlo becomes almost unimportant. There is also the possibility of coupling inertia-free methods with ALARIC to improve even further the computational speed of the method; These questions however, lie outside the scope of this paper.

For a particular choice of target density (Eq.46) we presented a theoretical estimation of the lowest possible error attainable by the system using only a few assumptions about the behaviour of the system. The estimation method, although conceived with this particular density profile in mind, has the potential to be extended to more general density distributions.

ALARIC was also tested against the state-of-the-art methods. The comparison helped illuminate the strong points and the weaknesses of our algorithm. Where WVT methods offer reasonable results with relatively short number of time steps, ALARIC delivers solutions whose errors are two orders of magnitude smaller at 10 times the computational cost.

Finally, by comparing the performance of initial conditions obtained with ALARIC with those created by pure crystalline structures, we emphasised the ability of the method to create isotropic initial conditions without preferred directions of propagation - which, as shown in section 5, can introduce spurious effects on the physical behaviour of the system.

7. Aknowledgments

The authors would like to thank J.M.Reynolds, D.Price and F.Stasyszyn for the fruitful conversations which, although brief in duration, were rich in content and helped shape the contents of this paper. This research was sponsored in part by DGICYT (Dirección General de Investigaciones Científicas y Tecnológicas) of Spain under Project No. ENE2015-68265 and the Erasmus Mundus Program: International Doctoral College in Fusion Science and Engineering FUSION-DC. The authors thank the editor and referees for the constructive comments and their patience.

Appendix 1. Analytical estimation of lowest possible error

To shed some light into the effect that Δ has on the overall convergence process (with $\rho^{\star}$ defined in Eq.46) we proceed as follows: we use a computational domain which correspondes to an d-dimensional ball, centred at the origin with unit radius: $\mathbb{B}^d_{[0,1]}$; This corresponds to the interval $[-1, 1]$ in 1D, the unit circle in 2D, and a unit sphere in 3D.

Let's now consider ε as defined in Eq.45. We see how this expression can be split into the individual contributions from all the particles in the domain:

$$\varepsilon = \sum_i \varepsilon_i \tag{50}$$

Fig.8 shows how all the simulations run for this density target evolved towards a state where one, and only one, particle was left alone sitting at the bottom of the density well. This observation allows us to assume that the biggest contribution to the error estimate comes from this particle:

$$\varepsilon_a = \frac{(mn_a - \Delta)^2}{n_a} \left(\int_{\mathbb{B}^d_{[0,1]}} [\rho^\star(\mathbf{r})]^2 d\mathbf{r} \right)^{-1} \tag{51}$$

The denominator can be evaluated analytically:

$$\mathcal{Q} \equiv \int_{\mathbb{B}^d_{[0,1]}} [\rho^\star(\mathbf{r})]^2 d\mathbf{r} = \mathcal{A} \int_0^1 [\rho^\star(r)]^2 r^{d-1} dr \tag{52}$$

with $\mathcal{A}$ the 'angular' part of the integral[13]. Next we integrate our density target over $\mathbb{B}^d_{[0,\lambda]}$, where the radial distance λ has been precisely chosen so as to ensure that the mass contained inside such ball, equals the mass of one particle - the very same particle sitting at the bottom of the density well; That is, the distance λ is *defined* via:

$$\mathcal{A} \int_0^\lambda \rho^\star(r) r^{d-1} dr = m \tag{53}$$

[13] Since our target function happens to be a radial function, an integration over $\mathbb{B}^d_{[0,1]}$, in spherical coordinates allows us to separate the integrals and directly evaluate the angular part. The values of $\mathcal{A}$ for $d = \{1, 2, 3\}$ are $\{2, 2\pi, 4\pi\}$ respectively.

This choice of λ also implies the choice for n_a:

$$n_a = \frac{1}{\mathcal{V}\lambda^d} \tag{54}$$

where the denominator simply represents the volume of a d-dimensional ball of radius λ. Under this interpretation, the constant $\mathcal{V}$, represents the volume of a d-dimensional *unit* sphere[14]. With these assumptions in mind, the error estimate on particle a is given by:

$$\varepsilon_a = \frac{\mathcal{V}\lambda^d}{\mathcal{Q}} \left(\frac{m}{\mathcal{V}\lambda^d} - \Delta \right)^2 \tag{55}$$

The mass of each particle can be readily obtained from the number of particles N ($m = \int_{\mathbb{B}^d_{[0,1]}} \rho^\star(\mathbf{r}) d\mathbf{r}/N$). And λ can be obtained numerically[15] from Eq.53.

In summary, for a given number of particles N, and a definite value of Δ, the minimum possible error the system can strive to, is obtained by assuming that particle a is the biggest source of error, and therefore, the best case scenario will come from neglecting the contributions of all the other particles in the system, that is:

$$\varepsilon_{\text{teo.min}} \approx \varepsilon_a \tag{56}$$

To illustrate the validity of the assumptions we initialise 51 equi-spaced particles in the 1D interval $[-1, 1]$ and run 5 consecutive simulation each with an extra multi-grid stage.

We compare 5 time traces coming from simulations with 0, 1, 2, 3 and 4 multi-grid stages, respectively. As mentioned before, every particle-splitting event triplicates (x3) the number of particles: meaning that the simulation with 0 stages ends up with the same 51 particles at the end, and in comparison, the

[14] The corresponding values of $\mathcal{V}$ for $d = \{1, 2, 3\}$ are $\{2, \pi, 4\pi/3\}$ respectively.

[15] An analytically solution is permitted in the one and two-dimensional cases. The process involves finding the root of an $(n + 2)$-degree polynomial meaning the three dimensional case must be treated numerically in any case.

simulation with 4 stages will end up with $51 \cdot 3^4$ particles, that is, with 4131 particles.

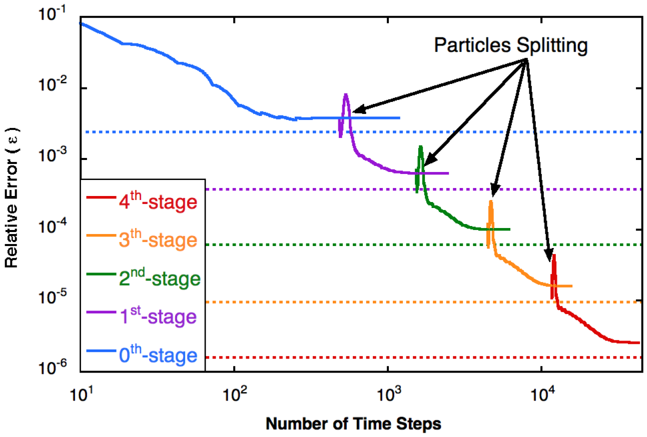

Figure 16: Time traces corresponding to 4 different simulations. Each run has an extra intermediate multi-grid stage. The time traces overlap with each other until the particle-splitting event occurs. Each event triplicates the number of particles and, consequently, the density error decrease even further - without ever going below the theoretical minimum. The time traces are color-coded with their respective theoretical minimum.

Fig.16 shows all 5 time traces. We see how the error estimate in general decreases, but there is a small oscillation following the splitting events. The noise introduced by the particle splitting and the consequent reorganisation of the particles are responsible for this oscillation in the time traces. In the graph is also shown the theoretical minimum for each run in the corresponding colour.

The time traces overlap for most of their paths until the algorithm finds it pertinent to perform a particle-splitting, at this point the time traces bifurcate. We see how the difference between the value, where the time trace saturates, and the theoretical minimum seem to get closer together as the number of particles increases - this is in accordance with Eq.56 and the assumption that the main source of error comes form the one particle sitting at the bottom of the density well, whose contribution becomes more marginal as the number of particles grows.

References

[1] R. A. Gingold, J. J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Monthly notices of the royal astronomical society 181 (3) (1977) 375–389.

[2] L. B. Lucy, A numerical approach to the testing of the fission hypothesis, The astronomical journal 82 (1977) 1013–1024.

[3] D. J. Price, Smoothed particle hydrodynamics and magnetohydrodynamics, Journal of Computational Physics 231 (3) (2012) 759–794.

[4] M. Steinmetz, E. Müller, On the capabilities and limits of smoothed particle hydrodynamics, Astronomy and Astrophysics 268 (1993) 391–410.

[5] R. F. Stellingwerf, Smooth particle hydrodynamics, in: Advances in the Free-Lagrange Method Including Contributions on Adaptive Gridding and the Smooth Particle Hydrodynamics Method, Springer, 1991, pp. 239–247.

[6] J. J. Monaghan, Smoothed particle hydrodynamics, Annual review of astronomy and astrophysics 30 (1992) 543–574.

[7] J. J. Monaghan, Smoothed particle hydrodynamics, Reports on progress in physics 68 (8) (2005) 1703.

[8] P. Michel, W. Benz, D. C. Richardson, Catastrophic disruption of asteroids and family formation: a review of numerical simulations including both fragmentation and gravitational reaccumulations, Planetary and Space Science 52 (12) (2004) 1109–1117.

[9] H. Gotoh, Sub-particle-scale turbulence model for the mps method-lagrangian flow model for hydraulic engineering, Comp. Fluid Dyn. J. 9 (4) (2001) 339–347.

[10] E. Y. Lo, S. Shao, Simulation of near-shore solitary wave mechanics by an incompressible sph method, Applied Ocean Research 24 (5) (2002) 275–286.

[11] R. Stellingwerf, R. Peterkin Jr, Smooth particle magnetohydrodynamics (invited review), Memorie della Societa Astronomica Italiana 65 (1994) 991.

[12] D. J. Price, J. Wurster, C. Nixon, T. S. Tricco, S. Toupin, A. Pettitt, C. Chan, G. Laibe, S. Glover, C. Dobbs, et al., Phantom: A smoothed particle hydrodynamics and magnetohydrodynamics code for astrophysics, arXiv preprint arXiv:1702.03930.

[13] S. Rosswog, Astrophysical smooth particle hydrodynamics, New Astronomy Reviews 53 (4) (2009) 78–104.

[14] Z. Meglicki, Verification and accuracy of smoothed particle magnetohydrodynamics, Computer Physics Communications 81 (1–2) (1994) 91 – 104. doi:http://dx.doi.org/10.1016/0010-4655(94)90113-9.
URL //www.sciencedirect.com/science/article/pii/0010465594901139

[15] D. A. Fulk, D. W. Quinn, An analysis of 1-d smoothed particle hydrodynamics kernels, Journal of Computational Physics 126 (1) (1996) 165–180.

[16] J. Hongbin, D. Xin, On criterions for smoothed particle hydrodynamics kernels in stable field, Journal of Computational Physics 202 (2) (2005) 699–709.

[17] W. Dehnen, H. Aly, Improving convergence in smoothed particle hydrodynamics simulations without pairing instability, Monthly Notices of the Royal Astronomical Society 425 (2) (2012) 1068–1082.

[18] S. Rosswog, D. Price, Magma: a three-dimensional, lagrangian magnetohydrodynamics code for merger applications, Monthly Notices of the Royal Astronomical Society 379 (3) (2007) 915–931.

[19] J. J. Monaghan, J. B. Kajtar, Sph particle boundary forces for arbitrary boundaries, Computer Physics Communications 180 (10) (2009) 1811–1820.

[20] S. Tavakkol, A. R. Zarrati, M. Khanpour, Curvilinear smoothed particle hydrodynamics, International Journal for Numerical Methods in Fluids 83 (2) (2017) 115–131.

[21] L. Brookshaw, A method of calculating radiative heat diffusion in particle simulations, in: Proceedings of the Astronomical Society of Australia, Vol. 6, 1985, pp. 207–210.

[22] P. Espanol, M. Revenga, Smoothed dissipative particle dynamics, Physical Review E 67 (2) (2003) 026705.

[23] V. Springel, L. Hernquist, Cosmological smoothed particle hydrodynamics simulations: the entropy equation, Monthly Notices of the Royal Astronomical Society 333 (3) (2002) 649–664.

[24] S. Rosswog, D. Price, 3d meshfree magnetohydrodynamics, in: Meshfree Methods for Partial Differential Equations IV, Springer, 2008, pp. 247–275.

[25] D. Price, J. Monaghan, Smoothed particle magnetohydrodynamics–i. algorithm and tests in one dimension, Monthly Notices of the Royal Astronomical Society 348 (1) (2004) 123–138.

[26] M. Herant, Dirty tricks for sph, Memorie della Societa Astronomica Italiana 65 (1994) 1013.

[27] D. Price, J. Monaghan, An energy-conserving formalism for adaptive gravitational force softening in smoothed particle hydrodynamics and n-body codes, Monthly Notices of the Royal Astronomical Society 374 (4) (2007) 1347–1358.

[28] S. Diehl, G. Rockefeller, C. L. Fryer, D. Riethmiller, T. S. Statler, Generating optimal initial conditions for smoothed particle hydrodynamics simulations, Publications of the Astronomical Society of Australia 32 (2015) e048.

[29] W. H. Zurek, W. Benz, Redistribution of angular momentum by nonaxisymmetric instabilities in a thick accretion disk, The astrophysical journal 308 (1986) 123–133.

[30] M. Nagasawa, T. Nakamura, S. M. Miyama, Three-dimensional hydrodynamical simulations of type ii supernova-mixing and fragmentation of ejecta, Publications of the Astronomical Society of Japan 40 (1988) 691–708.

[31] E. M. Purcell, Life at low reynolds number, American journal of physics 45 (1) (1977) 3–11.