Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Journal of Computational Physics 208 (2005) 345–367

JOURNAL OF
COMPUTATIONAL
PHYSICS

www.elsevier.com/locate/jcp

Regularized smoothed particle hydrodynamics with improved multi-resolution handling

S. Børve [a,b,d,*], M. Omang [b,c], J. Trulsen [b,d]

[a] *Centre of Mathematics for Applications, P.O. Box 1053 Blindern, N-0316 Oslo, Norway*
[b] *Institute of Theoretical Astrophysics, University of Oslo, P.O. Box 1029 Blindern, N-0315 Oslo, Norway*
[c] *Norwegian Defence Estates Agency, P.O. Box 405 Sentrum, N-0103 Oslo, Norway*
[d] *Centre for Advanced Study, Drammensveien 78, N-0271 Oslo, Norway*

Received 5 January 2004; received in revised form 15 November 2004; accepted 17 February 2005
Available online 14 April 2005

Abstract

Regularized Smoothed Particle Hydrodynamics (RSPH) has been proposed as an extension to Smoothed Particle Hydrodynamics (SPH). Its development has been motivated by the desire to improve the accuracy of the latter method by allowing the resolution to vary in time and space by orders of magnitude and by providing the means to avoid particle penetration. In the current work, an alternative technique for handling stepwise, spatially varying resolution, is proposed. A comparison is made between the original and new RSPH techniques. This includes test results from 2D simulations of linear sound waves as well as weak shocks. In addition, simulation results of two interacting, cylindrical blast waves are presented that provide a comparison between standard SPH with variable smoothing length and RSPH. The test results show that, using the new technique, a more accurate and robust description near steps in the resolution is obtained and that a significant improvement compared to standard SPH is achieved.

Keywords: Simulation; Particle methods and lattice-gas methods; Gas dynamics, general

---

1. Introduction

Smoothed Particle Hydrodynamics (SPH), originally developed for studying astrophysical problems [11,17], has been applied to a wide range of applications [6,14,20,22,29]. It is a Lagrangian method where a continuous fluid is represented by a discrete set of interacting particles. Each particle has attributes such

---

[*] Corresponding author. Tel.: +47 22630583; fax: +47 22856505.
E-mail address: steinar.borve@astro.uio.no (S. Børve).

0021-9991/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2005.02.018

as mass, velocity, and thermal pressure, and as the particles move with the fluid flow, the particle distribution will in general become more irregular in the course of a simulation. Due to the lack of reproducibility, solutions obtained from an increasingly more irregular particle distribution will exhibit an increasing amount of numerical errors. In an attempt to produce particle methods less sensitive to the relative position of individual particles, a number of SPH related methods have been proposed [1,7,13,16,27].

In high Mach number flows, a highly irregular particle distribution can in some cases cause the numerical description to break down. If the distance between neighbouring particles becomes too large, interaction between particles ceases to exist altogether. To reduce the risk of this happening, the particle mass profile should be restricted to being slowly varying in space. In 1D, a slowly varying mass profile is equivalent to the particle spacing being roughly proportional to the local variation in the density. Unfortunately, this tends to give inefficient numerical descriptions for applications involving large variations in the density. In regions of strong compression, a different problem might arise. When particles of different velocities are forced together, particle penetration, a situation where the velocity field no longer is single valued, can occur. The problem can be reduced either by adding sufficient amounts of artificial viscosity [19], or by using the XSPH prescription for updating the position of the particles [18]. These solutions alter the underlying physics by either introducing dissipation or increasing the dispersion, respectively.

To improve the accuracy of SPH calculations, the smoothing length h, the parameter determining the resolution in the continuous limit, is commonly allowed to vary in time and space. The smoothing length is then normally determined by the local number density of particles [12,25]. Unfortunately, the inclusion of a variable h requires additional terms in the equations of motion in order to satisfy conservation requirements [23]. To be able to safely neglect these extra terms, h is usually restricted to being slowly varying. This approach leaves small possibilities for tailoring the numerical description and makes it difficult to vary the resolution by orders of magnitude. Also, it is our experience that maintaining a slowly varying h can be difficult, especially in shock type problems.

Regularized Smoothed Particle Hydrodynamics (RSPH) has been proposed as an extension to SPH in an attempt to address some of the shortcomings of standard SPH [3]. The primary aim of the work has been to develop a numerical tool for efficiently producing accurate simulation results of shocks in general, and MHD shocks in particular. To achieve this, RSPH allows the particle distribution to be redefined at temporal intervals through a mass, momentum, and total energy conserving process. Highly irregular particle distributions, and therefore also particle penetration, can be avoided as a result of this feature. Furthermore, the smoothing length is allowed to vary both in time and space by orders of magnitude based on flexible, problem-specific criteria. Contrary to previous work, the smoothing length is chosen to be piecewise constant in space and time, with each step in space representing a factor of 2 change in the smoothing length value. For particles positioned sufficiently close to a step in the h-profile, a standard SPH calculation would produce large errors. An alternative summation technique, referred to as cell summation (CS), was therefore proposed for particles of this category [3].

Promising results using RSPH for hydrodynamic and magnetohydrodynamic shocks in 1D and 2D have been presented [3,4,24]. In these tests, steps in the smoothing length profile have been placed at distances larger than the local interaction range from regions where shock phenomena are occurring. The amount of artificial viscosity needed in order for the shocks to evolve are quite large so that any small amplitude disturbances generated by the shocks are typically strongly damped before reaching any step in the h-profile (hereafter referred to as step in h). In this paper it will be demonstrated that when either of these specifications are not sufficiently met, CS might give a less than satisfactory treatment of steps in h. Instead, a new technique, hereafter referred to as auxiliary particle interpolation (API), is proposed. The new technique replaces CS and is meant to provide a more accurate and robust treatment of steps in h.

In Section 2 some basic features of SPH are presented together with a brief discussion on the limitations of the CS technique. In Section 3 a detailed description of API, the new technique proposed for handling steps in the smoothing length profile, is given. In order to illustrate the difference in performance between

CS and API, a number of 2D tests have been performed where the smoothing length varies in space but is fixed in time. Linear sound wave tests are presented and discussed in Section 4.2. In these tests, we take a closer look at how wave properties are effected by the introduction of steps in the smoothing length profile. In Sections 4.3 and 4.4, we present simulations of plane and cylindrical weak shock waves in 2D. Finally, a detailed comparison is made in Section 4.5 between the performance of a standard SPH code with variable h and RSPH using either CS or API, this time allowing h to vary both in time and space. The chosen test problem describes the interaction of two cylindrical blast waves. Section 5 provides a concluding discussion of the results.

2. Standard SPH and first generation RSPH

SPH is based on interpolation theory. The equations of motion are expressed using integral interpolants, which in turn are approximated by summation interpolants. In the current work, the following set of equations are used to describe a hydrodynamic fluid [19]:

$$\frac{\mathrm{d}\mathbf{r}_a}{\mathrm{d}t} = \boldsymbol{v}_a, \tag{1}$$

$$\rho_a = \sum_b m_b \mathscr{W}_{ab}, \tag{2}$$

$$\frac{\mathrm{d}\boldsymbol{v}_a}{\mathrm{d}t} = -\sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} + \Pi_{ab} \right) \cdot \nabla_a \mathscr{W}_{ab}, \tag{3}$$

$$\frac{\mathrm{d}e_a}{\mathrm{d}t} = \sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{1}{2}\Pi_{ab} \right) \boldsymbol{v}_{ab} \cdot \nabla_a \mathscr{W}_{ab}, \tag{4}$$

$$P_a = (\gamma - 1) e_a \rho_a. \tag{5}$$

The mass, position, velocity, density, thermal pressure, and internal energy of particle a is denoted by m_a, $\boldsymbol{r}_a$, $\boldsymbol{v}_a$, ρ_a, P_a, and e_a, respectively. The difference in velocity between particle a and b, $\boldsymbol{v}_a - \boldsymbol{v}_b$, is denoted $\boldsymbol{v}_{ab}$. The interpolating kernel $\mathscr{W}_{ab} = \mathscr{W}(\boldsymbol{r}_a - \boldsymbol{r}_b, h_{ab})$ has the characteristic scale length $h_{ab} = (h_a + h_b)/2$ and should normalize to unity when integrated over all space. The interpolating kernel chosen in this work is the commonly used third-order B-spline function [19]

$$\mathscr{W}_{ab} = \frac{C(\delta)}{h^\delta} \begin{cases} 1 - \frac{3}{2}v^2 + \frac{3}{4}v^3 & \text{if } 0 \leqslant v \leqslant 1, \\ \frac{1}{4}(2 - v)^3 & \text{if } 1 < v \leqslant 2, \\ 0 & \text{if } v > 2, \end{cases} \tag{6}$$

where $v = |\boldsymbol{r}_a - \boldsymbol{r}_b|/h$. The normalization constant $C(\delta)$ equals 2/3, $10/(7\pi)$ or $1/\pi$ depending on the number of spatial dimensions δ, and $\nabla_a \mathscr{W}_{ab}$ is the derivative of $\mathscr{W}_{ab}$ with respect to $\boldsymbol{r}_a$. The artificial viscosity term, Π_{ab}, is needed when modelling shocks [19]. The ratio of specific heats at constant volume and constant pressure is given by γ.

2.1. Standard SPH with variable h

In Section 4.5 a comparison is made between the performance of RSPH using either CS or API and an SPH code using a standard approach to incorporating a variable smoothing length. Following the technique outlined in [12], the smoothing length h_a of a given particle a is updated so as to keep the number of particle interactions $\mathscr{N}_a$ involving particle a roughly constant. To achieve this, h_a is updated at time t by an iterative process:

$$\tilde{h}_a^n(t) = \frac{1}{2}\tilde{h}_a^{n-1}(t)\left[1 + \left(\frac{\mathscr{N}_s}{\mathscr{N}_a^{n-1}(t)}\right)^{1/2}\right], \tag{7}$$

where $\mathscr{N}_s$ is an input parameter indicating the optimal number of interactions involving one particle and $\tilde{h}_a^0(t) = h_a(t - \Delta t)$, the smoothing length of particle a at the previous time step. When the error $\delta\mathscr{N}_a^n(t) = |\mathscr{N}_a^n(t) - \mathscr{N}_s|$ falls below a predefined limit, the iterative process is terminated and $h_a(t)$ is set equal to $\tilde{h}_a^n(t)$. In [12] a few percent is indicated as a reasonable tolerance level for the ratio $\delta\mathscr{N}_a/\mathscr{N}_s$. However, we found that for the test performed in Section 4.5 such a low tolerance level was not achievable. We set the tolerance level to 22%, which in this case corresponds to allowing $\mathscr{N}_a$ for vary between 23 and 34. To achieve even this modest tolerance level for all particles at every time step, the number of required iterations of Eq. (7) per time step were typically 1–10, at critical time steps becoming as large as 50–100. Since each iteration requires updating the list of interacting pairs of particles, updating the smoothing length represent a major contributor to the total CPU time.

In order for the particle interactions to be symmetric in the case of variable h, it is proposed in [12] to choose $\mathscr{W}_{ab} = [\mathscr{W}(\boldsymbol{r}_a - \boldsymbol{r}_b, h_a) + \mathscr{W}(\boldsymbol{r}_a - \boldsymbol{r}_b, h_b)]/2$. For the blast wave test presented in Section 4.5, we found however the form $\mathscr{W}_{ab} = \mathscr{W}(\boldsymbol{r}_a - \boldsymbol{r}_b, h_{ab})$ given in [8], the same form used in the RSPH codes, to be preferable. The standard SPH code differs from the RSPH codes by only using particles of equal mass, even when large density variations are present. This is done to prevent the particle number density from getting too small when rarefaction waves propagate into originally high-density regions. Furthermore, the initial position of the particles is determined statistically by regarding the initial density profile as a probability function. This is a common approach to avoid systematic errors in the initial condition. As argued in [12], the initial statistical noise should only be of importance when simulating more or less time stationary problems.

2.2. Cell summation

The smoothing length profile used in the RSPH method is piecewise constant, with all spatial steps representing a factor of 2 change in the value of h. Applying standard SPH summations to particles within interaction range of such a step will create large numerical errors. An alternative summation method, referred to as cell summation (CS), was therefore suggested for particles found sufficiently close to steps in h [3]. The general idea behind the CS technique is that each particle should interpret the local particle distribution according to its own smoothing length. For a given particle a this is achieved by projecting the local particle distribution onto a lattice of template particles, or summation cells, symmetrically placed around particle a. The initial template particle spacing is determined by an estimate of the size of a. The projection of a neighbouring particle b onto the template particles of a is split into $N_s = (h_b/h_a)^\delta$ parts, referred to as subparticles, if $N_s > 1$. The reader is referred to [3] for details.

Although promising results have been produced using CS [3,4,24], there are a number of problems connected with this approach. First, the individual subparticles have to be assigned attributes based only on the information provided by the two interacting particles a and b. In multidimensional applications this information is not sufficient to determine subparticle attributes that are consistent with the spatial variation of the global field quantities. Secondly, the method requires the interpolating kernel of particle a to be modified by reducing the interaction range if a small, neighbouring particle b becomes enclosed by the interaction space of particle a. Template particles situated partially outside the original interaction range might contain incomplete information and should therefore be excluded from the summations. This is also achieved by reducing the interaction range of the kernel. A modified kernel should naturally meet the normalization requirement put on any SPH interpolating kernel. Still, the difference in discretization error between the original interpolating kernel and any modified interpolating kernel might lead to the generation of additional noise.

3. Auxiliary particle interpolation

The problems associated with CS is largely due to the fact that the template particles, or summation cells, need to be replaced every time step. This makes it difficult to enforce consistency conditions. A new method called auxiliary particle interpolation (API) is therefore proposed. It is based on the idea that particles should only interact directly with other particles having the same value of h. A particle whose interaction sphere is intersected by a step in h would then typically experience a particle deficiency. To prevent this from happening, additional particles referred to as auxiliary particles (as opposed to the regular particles) are generated in a sufficiently thick layer, typically $3h$ to $6h$ wide, around the step in h. The auxiliary particles used in this method play a similar role to that of the summation cells in the CS technique. The main difference being that the auxiliary particles are not replaced every time step. Instead, the auxiliary particles have the same life cycle as the regular particles, existing between two consecutive redefinitions of the particle distribution. The idea of having more than one type of particles taking care of different computational tasks is not new [15,27], but the characteristic feature of the auxiliary particles is that they are typically only needed in well-defined sub-regions, and not in the entire computational domain.

In Fig. 1, the use of auxiliary particles in 2D is illustrated in a situation where a step in h is indicated by the solid, vertical line. The filled circles represent regular SPH particles while the open circles represent auxiliary particles. The radii of the circles are proportional to the smoothing length of the corresponding

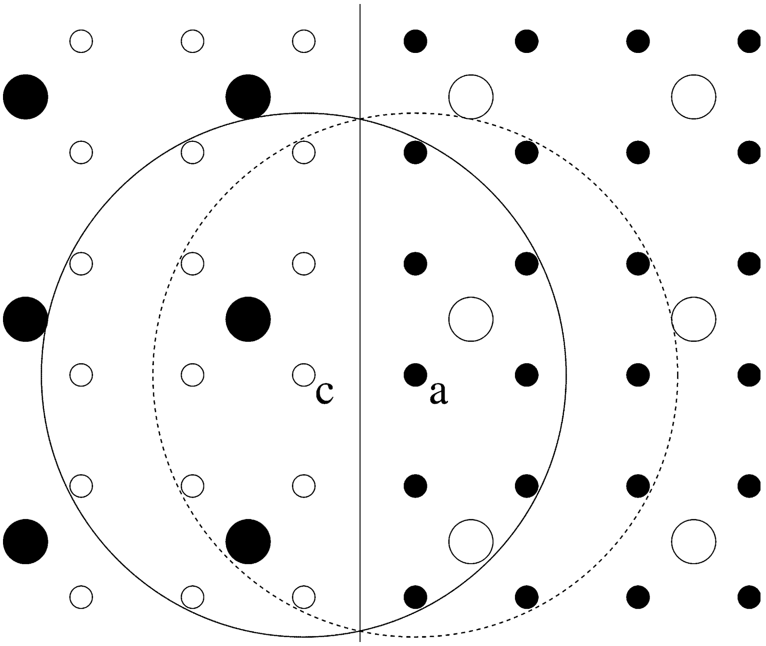

Fig. 1. Filled and open circles indicate regular SPH particles and auxiliary particles, respectively. Particles of the latter type are passively updated through interpolation.

particles. The dashed line indicates the interaction range of particle *a*. If the summations were restricted to regular particles with identical *h*, particle *a* would erroneously interpret the step in *h* as a free surface. When the auxiliary particles are included in the summations, all regular particles, including particle *a*, can be updated through standard SPH summations. When the attributes of the regular particles have been updated, the attributes of the auxiliary particles are updated through interpolation between the regular particles found within interaction range. In Fig. 1, the interaction range of auxiliary particle *c* is indicated by the dotted circle. It should be noted that with the current approach the auxiliary particles are passive nodes providing interpolated field data.

The inability of SPH to accurately interpolate functions to higher order for a set of irregularly spaced particles has been addressed on more general terms by e.g. Dilts [7] and Bonet and Kulasegaram [1]. A more statistical approach to the well-known problem of local polynomial fitting can be found in [9]. In this work we restrict ourselves to achieving auxiliary particle interpolation of up to second-order accuracy. Let δ and n denote the number of spatial dimensions and the order of accuracy, respectively. Then we define the polynomial vectors $\boldsymbol{p}_{\delta 0}$ to $\boldsymbol{p}_{32}$ as functions of the 3D positional vector $\boldsymbol{r} = [x, y, z]$:

$$\boldsymbol{p}_{\delta 0} = [1], \tag{8}$$

$$\boldsymbol{p}_{11} = [1, x], \tag{9}$$

$$\boldsymbol{p}_{12} = [\boldsymbol{p}_{11}, x^2], \tag{10}$$

$$\boldsymbol{p}_{21} = [1, x, y], \tag{11}$$

$$\boldsymbol{p}_{22} = [\boldsymbol{p}_{21}, x^2, xy, y^2], \tag{12}$$

$$\boldsymbol{p}_{31} = [1, x, y, z], \tag{13}$$

$$\boldsymbol{p}_{32} = [\boldsymbol{p}_{31}, x^2, xy, xz, y^2, yz, z^2]. \tag{14}$$

The number of elements of vector $\boldsymbol{p}_{\delta n}$ is given as

$$m(\delta, n) = \begin{cases} 1 & \text{if } n = 0, \\ \frac{(\delta+1)(\delta+2)^{n-1}}{n} & \text{if } n \in \{1, 2\}. \end{cases} \tag{15}$$

If $\boldsymbol{a}_c^f$ is a constant vector of $m(\delta, n)$ elements, then the function $\hat{f}_c(\boldsymbol{r}) = \boldsymbol{a}_c^f \cdot \boldsymbol{p}_{\delta n}(\boldsymbol{r})$ will be a polynomial of order n in δ variables.

For each auxiliary particle c and each function f to be interpolated, the vector $\boldsymbol{a}_c^f$ will be determined uniquely so that

$$H_c = \sum_b \frac{m_b}{\rho_b} [f_b - \hat{f}_c(\boldsymbol{r}_{bc})]^2 \mathscr{W}_{cb} \tag{16}$$

is minimized with respect to $\boldsymbol{a}_c^f$, where $\boldsymbol{r}_{bc} = \boldsymbol{r}_b - \boldsymbol{r}_c$. This is achieved by solving the set of linear equations produced by differentiating H_c with respect to $\boldsymbol{a}_c^f$. Following this standard approach gives

$$\boldsymbol{a}_c^f = \mathscr{P}_c^{-1} \boldsymbol{b}_c^f, \tag{17}$$

where

$$\mathscr{P}_c = \sum_b \frac{m_b}{\rho_b} \boldsymbol{p}_{\delta n}(\boldsymbol{r}_{bc}) \boldsymbol{p}_{\delta n}(\boldsymbol{r}_{bc}) \mathscr{W}_{cb} \tag{18}$$

and

$$\boldsymbol{b}_c^f = \sum_b \frac{m_b}{\rho_b} f_b \boldsymbol{p}_{\delta n}(\boldsymbol{r}_{bc}) \mathscr{W}_{cb}. \tag{19}$$

If $\mathscr{P}_c$ is singular, the order of interpolation must be reduced by 1 before performing the computations once more. Since all positional vectors in Eqs. (18) and (19) are given in terms of $\boldsymbol{r}_{bc}$, the interpolated value $\hat{f}_c$ of function f for auxiliary particle c is found simply as a_0, the first component of vector $\boldsymbol{a}_c^f$. This implies that only the first row of $\mathscr{P}_c^{-1}$ is required when performing the subsequent interpolations.

When using higher-order interpolation, or when using linear, one-sided interpolation, the interpolated value $\hat{f}_c$ might in some cases be outside the range defined by the values of f at the neighbouring, regular particles that were included in the interpolation. This does not usually represent a problem. However, if the physical model puts restrictions on the allowed range of f, special care should be taken. If $\hat{f}_c$ also exceeds the allowed range of f, the value of $\hat{f}_c$ is set to either the minimal or maximal value of f found among the neighbouring, regular particles, depending on whether $\hat{f}_c$ is smaller or larger than the allowed range. In the hydrodynamic model used in this work, the thermal pressure is restricted to being non-negative. If, as in the blast wave test in Section 4.5, a large discontinuous drop in the pressure is present, the second-order interpolation might in some cases result in negative interpolation values for the pressure. Following the general approach just outlined, the pressure for the auxiliary particle in question is then reset to the minimal pressure found among the neighbouring regular particles.

3.1. Conservation of linear momentum

In SPH, exact momentum conservation is achieved by using a symmetric formulation of the momentum equation (see Eq. (2)). When calculating the interaction across a step in h, this symmetry is not formally maintained and the momentum is therefore not automatically conserved. This lack of symmetry applies to both the two techniques proposed to handle the interaction near steps in h. However, the boundary conditions dictate the evolution of the total linear momentum to be given as

$$\sum_b \frac{\mathrm{d}\boldsymbol{p}_b}{\mathrm{d}t} = \oint_S \boldsymbol{F} \cdot \mathbf{d}\boldsymbol{S}, \tag{20}$$

where $\boldsymbol{p}_b$ in this context is the linear momentum of particle b, $\boldsymbol{F}$ is the force field, and the integration is performed over the closed boundary S.

In order to maintain a correct evolution of the total linear momentum of the system, the linear momentum for each regular particle a can be set to $\mathrm{d}\tilde{\boldsymbol{p}}_a/\mathrm{d}t$, where

$$\frac{\mathrm{d}\tilde{\boldsymbol{p}}_a}{\mathrm{d}t} = \frac{\mathrm{d}\boldsymbol{p}_a}{\mathrm{d}t} + \omega_a \left(\oint_S \boldsymbol{F} \cdot \mathbf{d}\boldsymbol{S} - \sum_b \frac{\mathrm{d}\boldsymbol{p}_b}{\mathrm{d}t} \right). \tag{21}$$

The factor ω_a determines the portion of the total momentum correction assigned to a given particle a. A necessary requirement is therefore that $\sum_a \omega_a = 1$. It is also reasonable to assume that ω_a should be proportional to m_a, and to the norm of v_a in high Mach number flows. Based on these observations, the following form of ω_a is chosen:

$$\omega_a = \frac{\sigma_a m_a(|v_a| + C_{s,a})}{\sum_b \sigma_b m_b(|v_b| + C_{s,b})}, \tag{22}$$

where $C_{s,a}$ is the sound speed at the position of particle a. Assuming that the boundary conditions are handled correctly, any deviation from the condition given by Eq. (20) must be caused by the correctional calculations performed near steps in h. Based on this observation, σ_a is chosen to be 0 if particle a is sufficiently far from any steps in h to only interact with other regular particles. If however, particle a interacts with any auxiliary particles, $\sigma_a = 1$. It is important to note that this correction will only secure global, not local, conservation of momentum.

3.2. Implementation

When implementing SPH based methods such as RSPH, there are four main tasks that need to be handled. First, one must determine how to initialize the particles. In the case of RSPH, a similar, additional task arise when redefining the particle distribution at later points in time. How this can be done with RSPH is described in detail elsewhere [2]. The auxiliary particles can be initialized in the same way as the regular particles, the only difference being that in the former case particles are only placed in limited sub-regions sufficiently close to steps in h. Secondly, a list of interacting pairs of particles needs to be updated for each step in time. This task is often referred to as the nearest neighbour problem. When simulating self-gravitating systems, the search for nearest neighbours is typically performed using a hierarchical tree method [12]. For other applications, grid based search algorithms where the grid size equals the interaction range of the particles are more common [28]. Information on which particles are found in which grid cells can then be easily found and stored in linked lists. Since RSPH allows h for a given particle to be equal to one of a finite number, quantized values, a multi grid based search algorithm has been developed to solve the nearest neighbour problem.

The process of finding the regular particles that a given auxiliary particle c will base its interpolation on, is in principle no different from the nearest neighbour problem. For this reason, we can apply the same algorithm for this task. However, to ensure stable and efficient interpolations, the number of neighbouring particles used by c in the interpolation, denoted N_c, should not differ too much from the standard neighbouring particle number N_0. N_0 is defined as the number of particles found within the interaction range when h is uniform and the particles are placed in a regular lattice configuration with standard inter-particle separation. To make sure that $N_c \sim N_0$, an additional smoothing length, denoted $\tilde{h}$, is used. For each regular particle b, $\tilde{h}_b = h_b$ at all times. In contrast, for each auxiliary particle c is $\tilde{h}_c$ time-dependent and only equal to h_c right after redefining the particle distribution. The method used for updating $\tilde{h}_c$ is essentially the same as that used for determining the smoothing length in the standard SPH code [12]. The process of finding the N_c neighbouring particles, updating $\tilde{h}_c$, and computing $\mathscr{P}_c^{-1}$ is summarized in the following pseudo code:

```
for each auxiliary particle $c$ do
    $b := c$
    for each regular particle $b :=$ SearchLinkedList($b$) do
        if $|\boldsymbol{r}_{bc}| < \tilde{h}_b + \tilde{h}_c$ then
            Add $b$ to the list of interpolation points for $c$.
            $N_c := N_c + 1$
        endif
    endfor
    $\tilde{h}_c := \tilde{h}_c\Big((1-\chi)(\frac{N_0}{N_c})^{1/\delta} + \chi\Big)$
    Compute $\mathscr{P}_c$ and $\mathscr{P}_c^{-1}$ based on the $N_c$ interpolation points.
    Store the 1st row of $\mathscr{P}_c^{-1}$.
endfor
```

Parameter χ, $0 \leqslant \chi \leqslant 1$, determines how sensitive $\tilde{h}_c$ is to changes in N_c. In the current work $\chi = 0.25$.

The third main task is to handle boundaries. This is a complex issue that will not be treated extensively here. In the current work, we have restricted ourselves to treating periodic and reflective boundaries, both of which have been implemented through the use of so-called mirror particles or ghost particles [2,15,26]. While both regular and auxiliary particles are internal particles in the sense that they at all times should be inside the computational domain, mirror particles are restricted to being positioned outside the computational domain. These particles are replaced at the beginning of each time step and, as the name indicates, are mirror images of internal particles. This means that the state of a given mirror particle only depends on the state of the corresponding internal particle. Mirror particles that are images of regular and auxiliary particles will be referred to as regular and auxiliary mirror particles, respectively. The mirror particles contribute to the dynamics of the regular particles in the same way as the other particle types do. When updating auxiliary particles through interpolation, auxiliary mirror particles are ignored.

Finally, a time integration scheme must be chosen. In the current work, a leap-frog algorithm is used [12]. However, the internal energy (and if present, magnetic field) is updated using a predictor-corrector scheme [5]. In order to step a given function f correctly forward in time, the different types of particles are updated in the following order: regular particles, regular mirror particles, auxiliary particles, and finally auxiliary mirror particles. The exception to this rule is the position. All internal particles update the position based on their respective velocities (see Eq. (1)). All mirror particles are replaced by new mirror particles without updating the position of the original ones. In this way, both the standard SPH summations and API can be used near boundaries in the same way as in the interior of the computational domain.

4. Test results

It is important for the robustness and applicability of the RSPH method that the smoothing length can be varied in space as freely as possible. A major concern is therefore whether spatial variations in h itself can generate major artifacts, such as unphysical instabilities or reflections. In this section, 2D test results using a nonuniform h-profile with CS or API are compared to results obtained using a uniform h-profile. The nonuniform h-profiles chosen in the first three tests are all static and they only apply two smoothing length levels. They clearly do not represent optimized numerical descriptions but have instead been chosen so as to accentuate any unphysical effects caused by an imperfect treatment of a spatially varying h. For the same reason, the particle distribution is not redefined during the simulation. Except where otherwise explicitly stated, exact conservation of total linear momentum is enforced for both CS and API as discussed in Section 3.1. In the latter case, second-order interpolation is assumed.

In the last test, RSPH simulations are performed where either CS or API is used with both particle regularization and h-profile optimization. This means that the h-profile is allowed to vary both in time and space. Exact conservation of linear momentum as described in Section 3.1 is in this case not enforced. The results from these simulations are then compared with results from a simulation using a standard SPH code as described in Section 2.1 and a high-resolution RSPH simulation with particle regularization but with uniform h. The test describes the interaction of two cylindrical blast waves and is meant to illustrate the capabilities of RSPH relative to standard SPH when no artificial restrictions are put on the h-profile as is the case for the first three tests.

In general, tests using a nonuniform h-profile will be referred to as mixed-resolution tests. The corresponding tests using a uniform h-profile, where h equals the minimum or maximum smoothing length level, will be referred to as the corresponding high- and low-resolution tests, respectively. The accuracy obtained in the mixed-resolution results should ideally be close to the accuracy obtained in the corresponding high-resolution results in regions where h equals the minimum h. In regions where h equals the maximum h, the accuracy should be no less than that obtained in the low-resolution results. All simulations discussed in this section deal with a $\gamma = 1.4$ perfect gas and the results are all given in dimensionless units.

4.1. Error estimate Q

Like any other numerical method, SPH with uniform or variable h does not give an exact solution to any physical problem. Depending on the resolution and characteristics of the problem, the numerical errors can be substantial. In the current work, we are interested in focusing on numerical errors introduced when h is allowed to vary in steps of 2 using either CS or API. To clearly illustrate the difference between the two methods, it is important that errors inherent in SPH (when no step in h is present) are ignored. In this case, we have 4 different solutions of each problem: Two solutions with uniform resolution (high and low resolution) and two solutions with mixed resolution (using either CS or API). The uniform-resolution results are regarded as the reference SPH solutions of the problem for the given resolutions. The initial particle distribution in the mixed-resolution simulations can be viewed as a linear combination of the high- and low-resolution particle distributions. So for each particle belonging to a mixed-resolution distribution, a corresponding particle is found in either the high- or low-resolution distributions. If the steps in h do not introduce extra numerical errors, this correspondence should hold also at later times. The quality of the handling of steps in h for a given problem should be indicated by the difference between the solutions found for the mixed-resolution particles and their corresponding uniform-resolution particles. This can then be compared to the difference between the high- and low-resolution results.

The error estimate scheme chosen in the current work attempts to utilize the one-to-one relation initially found between the mixed-resolution particles and the corresponding uniform-resolution particles. At the same time, it applies a simple interpolation strategy for situations where, for a given point in time, such a one-to-one relation cannot be established. The latter case applies when comparing the high- and low-resolution results. The error estimate is found by calculating a time-dependent parameter of the form:

$$Q_f(t) = \sqrt{\frac{\sum_a m_a[f_a(t) - \tilde{f}_a(t)]^2}{\sum_a m_a f_a^2(t)}}. \tag{23}$$

When the sum is taken over all particles a belonging to a given mixed-resolution particle distribution, the parameter will be referred to as $Q_{M,f}$. When the sum is taken over all particles a belonging to a corresponding low-resolution particle distribution, the parameter will be referred to as $Q_{L,f}$. The estimated value $\tilde{f}_a$ is obtained from function f by

$$\tilde{f}_a = \left(\sum_{b\in\tilde{N}_a} \frac{m_b}{\rho_b} f_b \widehat{\mathscr{W}}(\boldsymbol{r}_{ab}, h_b)\right) \Bigg/ \left(\sum_{b\in\tilde{N}_a} \frac{m_b}{\rho_b} \widehat{\mathscr{W}}(\boldsymbol{r}_{ab}, h_b)\right). \tag{24}$$

The estimate $\tilde{f}_a$ is computed by taking summations over the particle distribution marked $\tilde{N}_a$. If particle a belongs to the mixed-resolution distribution and h_a equals the minimum or maximum h, $\tilde{N}_a$ equals the corresponding high- or low-resolution distribution, respectively. If particle a belongs to the low-resolution distribution, $\tilde{N}_a$ equals the corresponding high-resolution distribution. In order for $\tilde{f}_a$ to be strongly dominated by the particles found closest to particle a, a narrow weight function is used:

$$\widehat{\mathscr{W}}(\boldsymbol{r}, h) = \begin{cases} \frac{\epsilon^2 h^2}{r^2+\epsilon^2 h^2} & \text{if } 0 \leqslant r < h, \\ 0 & \text{if } r \geqslant h, \end{cases} \tag{25}$$

where ϵ is chosen to be 0.02. The weight function has not been normalized since the calculation of $\tilde{f}_a$ includes an explicit division by the unity function to avoid large discretization errors.

To help interpret temporal variations in $Q_{M,f}$ and $Q_{L,f}$, consider the case of simple harmonic signals: We assume that at a given point in time $f_a = f_1 \sin(\phi_a)$ and $\tilde{f}_a = f_1 \sin(\phi_a + \Delta\phi)$ for all particles a belonging to the same particle distribution. An approximation to Q_f, only dependent on the phase difference $\Delta\phi$, can be found by taking the continuous limit of Eq. (23),

$$Q_f \approx \sqrt{\frac{\int_0^{2\pi}[\sin\phi - \sin(\phi + \Delta\phi)]^2\,\mathrm{d}\phi}{\int_0^{2\pi}\sin^2\phi\,\mathrm{d}\phi}} = \sqrt{2(1-\cos\Delta\phi)}. \tag{26}$$

As $\Delta\phi$ varies from 0 to π, Q_f will vary from 0 to 2. If $\Delta\phi < 1$ then $Q_f \approx \Delta\phi$. If f and $\tilde{f}$ represent waves with different phase velocities and amplitudes, the time variation of $Q_f(t)$ should indicate whether errors in phase or amplitude are dominant. From linear stability analysis [21] it is known that the numerically obtained phase velocity of a simulated wave will depend on the resolution. As one would expect, this dependence is at its strongest in the low-resolution parameter range and negligible in the parameter range of sufficiently high resolution.

4.2. Sound waves

In the following set of tests, the stability properties of linear sound waves are investigated. The period and the wave length of the simulated wave are denoted by P and λ, respectively. The mean particle spacing in the low-resolution runs is denoted $\Delta_{\max}$ and equals $\lambda/6$, $\lambda/24$, and $\lambda/96$. In the mixed-resolution runs the value of h switches periodically in space between two levels, with a spatial periodicity equal to 2λ between each step. The length of the simulated area parallel to the propagating wave (x-direction) equals 4λ, while the corresponding width perpendicular to the propagating wave (y-direction), is equal to $10h_{\max}$, where $h_{\max}$ is the smoothing length used in the low-resolution runs. Periodic boundary conditions are applied in both directions.

The particle distribution is assumed to have only small perturbations to an otherwise square lattice equilibrium configuration with 1.2 particles per h. According to the stability analysis in [21], isothermal linear sound waves will then be stable in the case of uniform h. This result also applies for $\gamma>1$ [5]. However, simulations using uniform h show that a slowly growing nonlinear component can be detected even for very small initial wave amplitudes [5]. The growth rate of these instabilities increases with increasing wave amplitude and with number of particles per wave length. The instabilities can effectively be eliminated by applying small amounts of artificial viscosity. Since increased artificial viscosity also leads to more effective damping of the wave, no artificial viscosity was applied in the following tests. It is therefore reasonable to expect these instabilities, not predicted by the linear theory but observed in the uniform h case, to be present also when steps in h are introduced.

When CS is used to handle the variable smoothing length profile in the mixed-resolution tests, the simulated wave is dominated by numerical noise within 1.5, 0.4, and 0.1 periods for $\lambda/\Delta_{\max} = 6$, 24, and 96, respectively. Soon, this noise causes the numerical description to break down completely. The reason for this rapid destruction of the wave is that the level of noise inherent to CS is substantial compared to the wave amplitude. In particular, the inability to fully separate the dynamics parallel and perpendicular to the propagation vector is crucial.

When API is used with variable smoothing length, the dynamics of the simulated wave is better captured. In Fig. 2, the velocity profile in the x-direction, δv_x, divided by the initial amplitude, v_1, is plotted at $t/P = 40$ for the case $\lambda/\Delta_{\max} = 6$, with the solid, dashed, and dotted lines representing the high-, low-, and mixed-resolution results. The corresponding error estimates $Q_{L,vx}(t)$ (solid line) and $Q_{M,vx}(t)$ (dotted line) are plotted in Fig. 3. The two solutions produced with a uniform h-profile both maintain the wave amplitude well despite the small number of particles per wave length. From this snapshot it might seem that the difference in phase velocity between the two solutions are negligible. But from the solid line in Fig. 3 one can see that the phase difference has a period of roughly 13 wave periods and is close to zero after 40 periods. This is in accordance with linear stability analysis [5].

The dotted lines in Figs. 2 and 3 show that other wave modes have been introduced in the mixed-resolution results, leading to amplitude reduction and a generally more complex signal. The distortion is

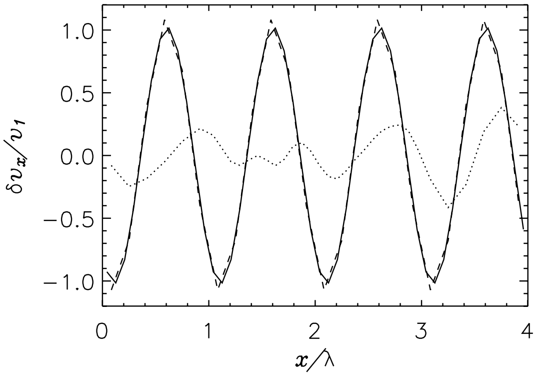

Fig. 2. Solutions of $\delta v_x/v_1$ in the linear wave test at $t/P = 40$ for the case $\lambda/\Delta_{\max} = 6$, solved using the high- (solid line), low- (dashed line), and mixed-resolution (dotted line) particle distributions.

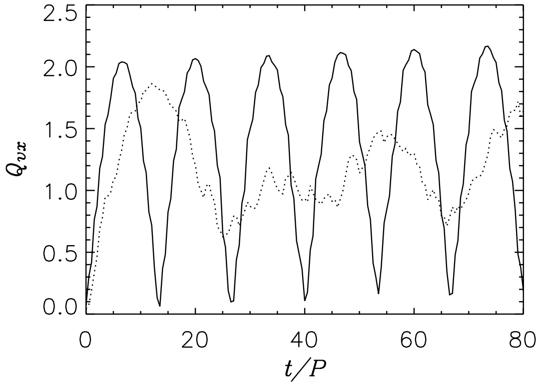

Fig. 3. Error estimates $Q_{L,vx}$ (solid line) and $Q_{M,vx}$ (dotted line) as functions of time for the case $\lambda/\Delta_{\max} = 6$.

probably mostly due to the fact that the phase speed varies initially in space as a result of the varying resolution. Observations of other wave modes becoming considerably less important when the second-order interpolation is replaced by the zeroth-order interpolation, strengthen the assumption that the generation of other wave modes is not in this case an indication of the need for a higher-order API when $\lambda/\Delta_{\max}$ is relatively small. It should also be noted that although the wave will be distorted in roughly 30 periods when using the second-order interpolation, the mixed-resolution description seems to remain relatively stable after this point in time as well.

It is expected from linear theory that for $\lambda/\Delta_{\max} = 24$, the difference in phase speed in the low- and high-resolution runs should be greatly reduced. This might not be clear when studying Fig. 4 where $\delta v_x/v_1$ is again plotted at $t/P = 40$ with the solid, dashed, and dotted lines representing the high-, low-, and mixed-resolution results. However, the corresponding error estimates $Q_{L,vx}(t)$ (solid line) and $Q_{M,vx}(t)$ (dotted line) shown in Fig. 5 confirm the theoretical predictions. From studying Fig. 4, it is also apparent that especially the high-resolution result exhibits nonlinear effects in that the norm of the maximum and minimum values of the velocity are not the same. The dotted line shows that the wave is still at this point well represented by the mixed-resolution description, but the nonlinear effects are less apparent than in the two cases of uniform h. This can probably be attributed to the fact that higher harmonics are not as well captured by the second-order API. By comparing the dotted and solid lines in Fig. 5 and using the approximation given in Eq. (26),

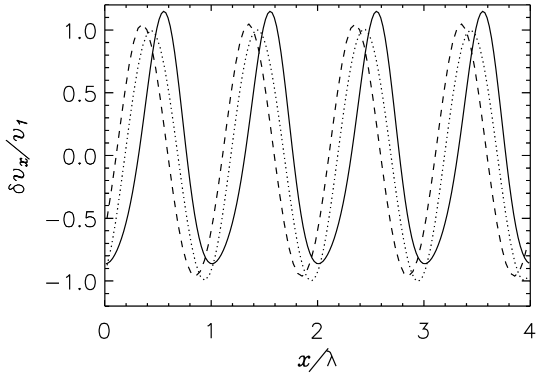

Fig. 4. Solutions of $\delta v_x/v_1$ in the linear wave test at $t/P = 40$ for the case $\lambda/\Delta_{\max} = 24$, solved using the high- (solid line), low- (dashed line), and mixed-resolution (dotted line) particle distributions.

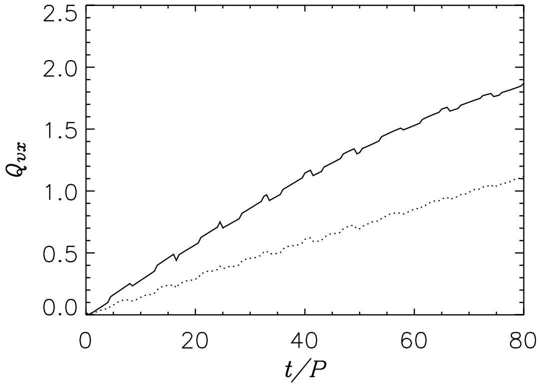

Fig. 5. Error estimates $Q_{L,vx}$ (solid line) and $Q_{M,vx}$ (dotted line) as functions of time for the case $\lambda/\Delta_{\max} = 24$.

the phase speed in the mixed-resolution case can be estimated to be close to the mean phase speed for the corresponding high- and low-resolution cases.

Finally, we look at the case when $\lambda/\Delta_{\max} = 96$. In this case the difference in phase speeds obtained using the high-, low-, and mixed-resolution description is negligible after 40 periods, as illustrated in Fig. 6. The solid and dotted lines in Fig. 7 illustrating the error estimates $Q_{L,vx}$ and $Q_{M,vx}$, respectively, confirm the approximate relation $Q_M \approx Q_L/2$ up until roughly 60 periods. At later times, high frequency nonlinearities become important. The growth rate of these nonlinearities increases with increasing resolution. Due to the instabilities the system will become noise dominated. It is important to note that the occurrence of this instability is not dependent on whether a uniform or a non-uniform *h*-profile has been used [5]. However, once the simulated system has become noise dominated, the numerical description will, in the absence of numerical viscosity, be stabilized only when a uniform *h*-profile is used. In the case of a non-uniform *h*-profile, the description of a non-viscous, noise dominated system will not be stable, eventually leading to a break-down of the numerical description.

Other mixed-resolution tests of linear waves using API are performed as well. Still enforcing conservation of total linear momentum, the results are not particularly sensitive to the order of the API. As mentioned earlier when $\lambda/\Delta_{\max} = 6$, the zeroth-order interpolation gives in fact a slightly better description than the first and second-order interpolation. When $\lambda/\Delta_{\max} = 24$ and $\lambda/\Delta_{\max} = 96$, an increase in interpolation

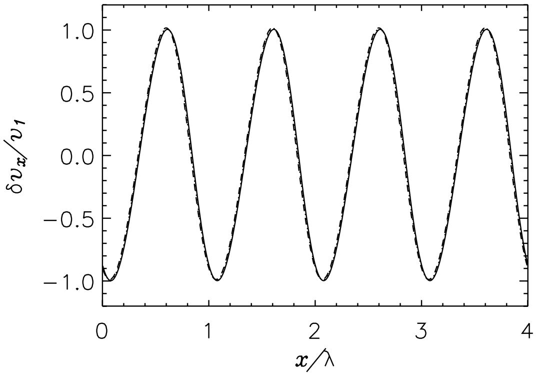

Fig. 6. Solutions of $\delta v_x/v_1$ in the linear wave test at $t/P = 40$ for the case $\lambda/\Delta_{\max} = 96$, solved using the high- (solid line), low- (dashed line), and mixed-resolution (dotted line) particle distributions.

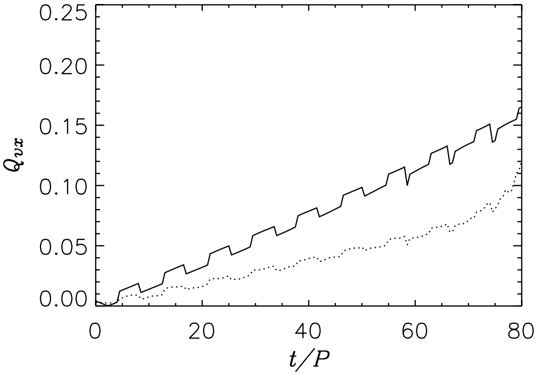

Fig. 7. Error estimates $Q_{L,vx}$ (solid line) and $Q_{M,vx}$ (dotted line) as functions of time for the case $\lambda/\Delta_{\max} = 96$.

order gives a slight improvement in the description of the wave. However, when conservation of total linear momentum is not enforced as described in Section 3.1, using zeroth or first-order interpolation will cause the particles gradually to start drifting. This effect seems to be negligible when second-order interpolation is used.

4.3. Plane shock

The next test illustrates how well the CS and API techniques handle the sub-optimal situation of plane shocks propagating across a step in h. Since the option of redefining the particle distribution at temporal intervals is not used in this test, a weak shock test is chosen with an initial particle spacing of $h/1.5$. The simulation area is 2 length units long with reflective boundary conditions in the x-direction and 0.3 length units wide with periodic boundary conditions in the y-direction. The gas is initially at rest with jump conditions in pressure and density specified as

$$P(x,y) = \begin{cases} 1.0 & \text{if } 0 \leqslant x \leqslant 1, \\ 0.5 & \text{if } 1 < x \leqslant 2, \end{cases} \tag{27}$$

and

$$\rho(x,y) = \begin{cases} 1.0 & \text{if } 0 \leqslant x \leqslant 1, \\ 0.9 & \text{if } 1 < x \leqslant 2. \end{cases} \tag{28}$$

These conditions cause a plane, Mach 1.2 shock to be generated and to propagate to the right, a contact discontinuity to be formed, and a rarefaction wave to propagate to the left.

Four different simulations are performed. All tests use artificial viscosity with the viscosity parameters α and β both set to 1 [19]. First, high- and low-resolution simulations are performed where h equals $h_{\min} = 0.015$ and $h_{\max} = 0.03$, respectively. Then, two additional simulations are performed where the initial h-profile is given by

$$h(x,y) = \begin{cases} h_{\min} & \text{if } 0.8 < x < 1.2, \\ h_{\max} & \text{otherwise.} \end{cases} \tag{29}$$

One of the mixed-resolution tests uses the CS technique to handle the steps in h, while the other test uses API. In both cases, the evolution of total linear momentum is corrected so as to be in accordance with the boundary conditions.

In Figs. 8(a)–(d) the density and velocity in the x-direction, respectively, are plotted at times $t = 0.15$ ((a) and (c)) and $t = 0.5$ ((b) and (d)) for the high-resolution case (solid line), the mixed-resolution case using CS (dotted line), and the mixed-resolution case using second-order API (dashed line). At time $t = 0.15$ the shock propagating to the right and the rarefaction wave propagating to the left have not yet reached the steps in the resolution. The three solutions are therefore seen in Figs. 8(a) and (c) to be more or less identical. Due to particle motion, the steps in h for the mixed-resolution tests are at time $t = 0.5$ found at $x \approx 0.88$ and $x \approx 1.28$. A comparison of the solid and dashed lines in Figs. 8(b) and (d) shows the mixed-resolution results using API to fit well with the high-resolution results in the interval (0.88, 1.28), and more specifically, that the contact discontinuity located at $x \approx 1.1$ is equally well resolved in the two simulations. Outside this interval, the local h in the mixed-resolution simulation is a factor of 2 higher than in the high-resolution simulation. Jumps in the resulting density and velocity profiles located in these intervals are therefore not as sharply defined.

The dotted lines in Figs. 8(b) and (d) reveal that when CS is used to handle the steps in h, the mixed-resolution results show substantial deviations from the high-resolution results. These deviations come from

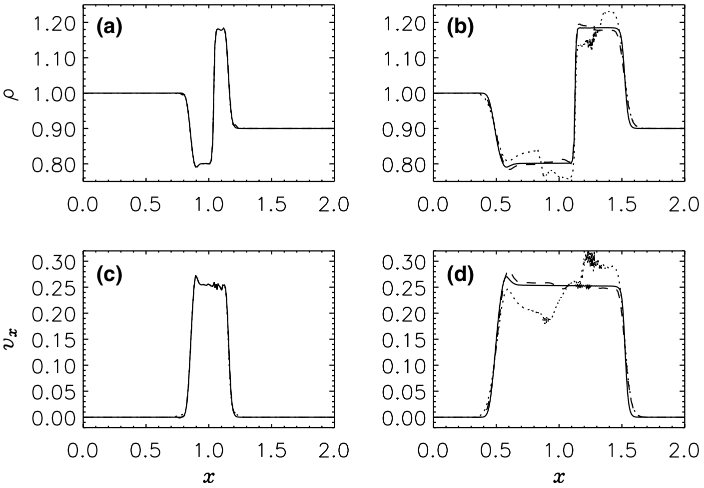

Fig. 8. Solutions of ρ (plot a and c) and v_x (plot b and d) for the plane shock test at $t = 0.15$ and $t = 0.5$ projected onto the x-axis in the cases of high resolution (solid line), mixed resolution with API (dashed line), and mixed resolution with CS (dotted line).

the fact that the dynamics parallel and perpendicular to the shock front are not sufficiently well separated near steps in h when CS is used. This is confirmed by looking at the velocity in the y-direction (not shown). Contradicting theory, disturbances in v_y with an amplitude of roughly 0.02 are observed near the steps in h. In comparison, the magnitude of the disturbances in v_y for the other three simulations of this test is only slightly larger than round-off errors and therefore negligible. In Fig. 9 the error estimate Q_ρ is plotted as a function of time for the low-resolution simulation (solid line), and the mixed-resolution simulations using CS (dotted line) and API (dashed line), respectively. This plot shows that a density profile closer to that obtained in the high-resolution case is achieved with the mixed resolution using API. On the other hand, using CS, the errors near the steps in h will after some time outweigh the higher resolution that a partly reduced h should theoretically imply.

4.4. Cylindrical shock

The following test describes a cylindrical shock generated by a circular, high-density and high-pressure region centred at $(x_0, y_0) = (1, 1)$ with initial radius 0.25. The simulation area is a square with sides 2 length units. The gas is initially at rest and the jump conditions in pressure and density is similar to that used in the plane shock test:

$$P(r) = \begin{cases} 1.0 & \text{if } 0 \leqslant r \leqslant 0.25, \\ 0.5 & \text{otherwise,} \end{cases} \tag{30}$$

and

$$\rho(r) = \begin{cases} 1.0 & \text{if } 0 \leqslant r \leqslant 0.25, \\ 0.9 & \text{otherwise,} \end{cases} \tag{31}$$

where $r = \sqrt{(x - x_0)^2 + (y - y_0)^2}$. The initial density profile is indicated by the grey-scale coding of Fig. 10(a). The initial particle spacing is chosen to be $h/2$ and once again the viscosity parameters α and β are set to 1.

Once again four different simulations are performed. The high- and low-resolution simulations with a uniform smoothing length have h equal to $h_{\min} = 0.02$ and $h_{\max} = 0.04$, respectively. The initial h-profile used in the last two simulations is indicated by the closed contour line in Fig. 10(a). Inside the contour line h equals $h_{\min}$ while outside h equals $h_{\max}$. One of the mixed-resolution simulations uses CS while the other uses API. In Figs. 10(b)–(d), the norm of the velocity is plotted at time $t = 0.5$ for the simulations using the

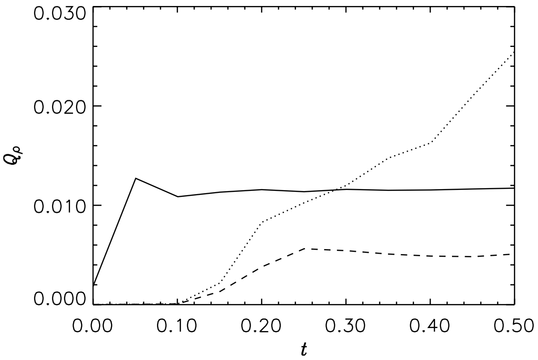

Fig. 9. Error estimates $Q_{L,\rho}$ (solid line) and $Q_{M,\rho}$ for the plane shock test using API (dashed line) and CS (dotted line) as functions of time.

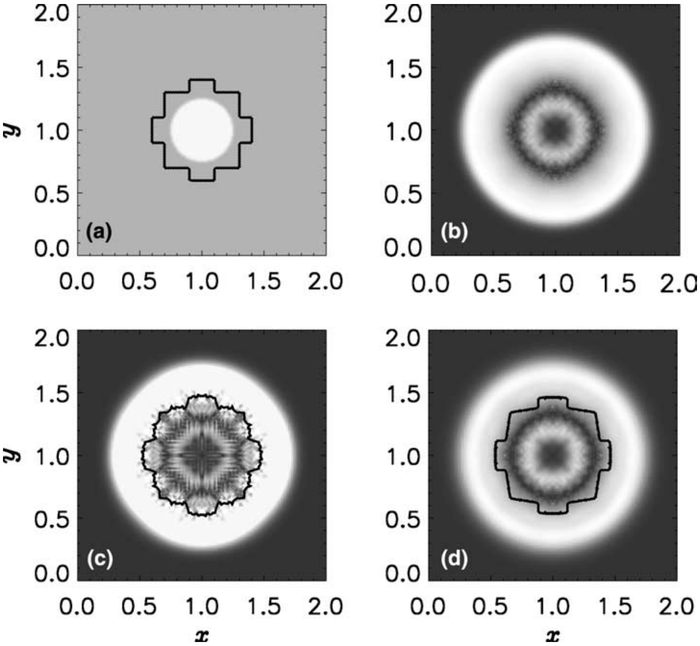

Fig. 10. Solutions of ρ at $t = 0$ (plot a) and v for the cylindrical shock test at $t = 0.5$ in the cases of high resolution (plot b), mixed resolution with CS (plot c), and mixed resolution with API (plot d). Superimposed on plot a, c, and d are contour lines of the h-profile.

high resolution, the mixed resolution with CS, and the mixed resolution with API, respectively. The bright areas indicate high-velocity regions. The contour lines in Figs. 10(c) and (d) separate the regions of high and low h in the mixed-resolution case. It is evident from comparing Figs. 10(b) and (d) that the mixed-resolution simulation with API to a large extent reproduces the results of the high-resolution simulation. The difference in maximum and minimum velocity and density in the two solutions are roughly 2% or less. In comparison, Fig. 10(c) shows that the mixed-resolution simulation with CS creates unphysically high velocities on the inside of the interface between regions of different values of h. The maximum velocity found in this case is roughly 18% larger than the maximum velocity found in the high-resolution results. A similar error is found for the minimum density.

In Fig. 11 the error estimate Q_ρ is plotted as a function of time for the low-resolution simulation (solid line), and the mixed-resolution simulations using CS (dotted line) and API (dashed line), respectively. Even more clearly than in the plane shock test, the error estimate indicates that the mixed-resolution results benefit greatly from using API instead of CS to handle steps in h. Moreover, the difference in the quality of the two mixed-resolution solutions is evident already from the beginning of the simulations. In this 2D problem, it is clear that the use of CS makes it more difficult to represent curved features across steps in h.

4.5. Two interacting, cylindrical blast waves

In the simulations presented so far, restrictions have been put on the functionality of RSPH. This has been done in order to clearly demonstrate differences in the accuracy achieved near steps in h when using either CS or API. With these restrictions in mind, the results from the tests presented so far are not particularly well suited for a direct comparison with results obtained with other numerical methods. To illustrate how the RSPH method with full functionality using either CS or API compares to the standard SPH code described in Section 2.1, results from 2D simulations of two interacting, cylindrical blast waves are

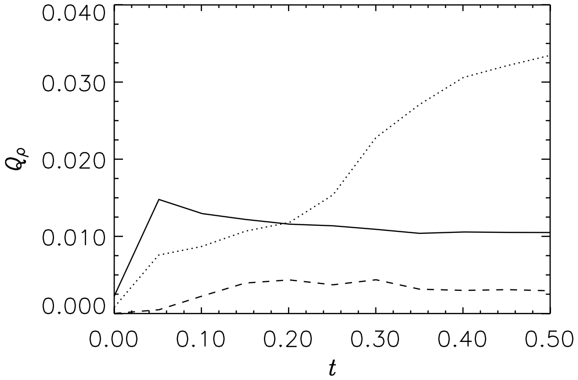

Fig. 11. Error estimates $Q_{L,\rho}$ (solid line) and $Q_{M,\rho}$ for the cylindrical shock test using API (dashed line) and CS (dotted line) as functions of time.

presented. For reference purposes, a high-resolution version of RSPH using a uniform *h*-profile (therefore neither CS or API is used) is also run. The three RSPH simulations all have 1.5 particles per *h* and 32 time steps between each regularization. The number of particles in the reference simulation with uniform *h* was 640,000, while the time-averaged particle number in the CS and API simulation were 35,900 and 49,200, respectively. In the case of API, this number includes both regular and auxiliary particles. In the case of CS, summation cells are required in addition to particles. Both total memory use and CPU-time are therefore roughly the same for the CS and API simulations. For the standard SPH test, the optimal number of interactions involving one particle, $\mathcal{N}_s$, was set to approximately 28.3 in 2D, which is equivalent to having 1.5 particles per *h*. The total number of particles were set to 50,000.

The simulation domain is a square with reflective sides of unity length. The initial condition is characterized by two quarter-cylindrical, bounded regions with radius of 0.4, here referred to as regions R_0 and R_2, centred at $\boldsymbol{r}_0 = (0, 0)$ and $\boldsymbol{r}_2 = (1, 1)$, respectively. The remaining part of the simulation domain is referred to as region R_1. The gas is initial at rest, but the following jump conditions are set for the pressure and the density:

$$P(\boldsymbol{r}) = \begin{cases} 10^4 & \text{if } \boldsymbol{r} \in R_0, \\ 10^3 & \text{if } \boldsymbol{r} \in R_2, \\ 10^{-2} & \text{otherwise,} \end{cases} \tag{32}$$

and

$$\rho(\boldsymbol{r}) = \begin{cases} 10 & \text{if } \boldsymbol{r} \in R_0 \cup R_2, \\ 1 & \text{otherwise.} \end{cases} \tag{33}$$

At time $t = 0$, the imagined membranes separating the three regions are removed and the high-pressure gas will start to expand into the intermediate gas. The Mach numbers of the left- and right-hand blast waves are approximately 370 and 110, respectively. At about time $t = 0.011$ the two blast waves collide, causing the density to peak at roughly 20, and at about $t = 0.014$ the first shock will hit the walls. Beyond this point in the time a complex combination of shock interactions will take place.

In Fig. 12 the solutions for the density at times $t = 0.015$ and $t = 0.025$ are plotted for the four different methods to be compared. As the density varies from roughly 0.5 up to more than 25, the colouring changes from dark green, through red and purple, to white. The top, leftmost plot of the high-resolution solution shows that at $t = 0.015$ the two blast waves have started to interact. A weak, almost cylindrical shock front, here referred to as the main weak shock (MWS), is found at a radial distance of roughly 0.55 from $\boldsymbol{r}_2$.

Closer to $\boldsymbol{r}_0$, the dynamics is completely dominated by the left-hand blast wave. At smaller radii from $\boldsymbol{r}_2$, the density profile is dominated by an elongated, high-density region, here referred to as the main compression region (MCR), with two almost plane-parallel boundaries perpendicular to the diagonal between $\boldsymbol{r}_0$ and $\boldsymbol{r}_2$. The boundary located closer to $\boldsymbol{r}_0$ is the main contact discontinuity (MCD), while the other is the main stagnation line (MSL). Closer to $\boldsymbol{r}_2$ the right-hand blast wave dominates the dynamics. Notice also the two kinks found on both the MCD and the MSL near the intersection of the two. These kinks are effects of the MCD and the MSL propagating through regions of different sound speed. Inside the MCR, a narrow strip of even higher density is found, denoted the central compression region (CCR). Close to the top and right-hand walls, roughly triangular shaped low-density regions (TLRs), mark the remnants of the low-temperature state initially found in R_1. At the same time, the density has started to increase in the areas where the blast waves have hit the walls.

The high-resolution result illustrated in the bottom, leftmost plot, show that at $t = 0.025$ the effects of shocks reflecting off the walls have become more important. Roughly half the simulation domain is at this point in time left with an almost uniformly low density and will therefore be referred to as the main rarefaction region (MRR). Near the top and right-hand boundaries, however, small-scale fluctuations with traces of vortex formation have appeared. The MCR has propagated closer to $\boldsymbol{r}_2$ and has in the process become shorter and thicker due to boundary effects. Regular reflections of the stagnation line MSL began around $t = 0.018$, and at $t = 0.025$ these reflections have evolved into Mach reflections with accompanying triple points. The line emanating from the triple points separating the MCR and the boundary regions are then the reflected stagnation lines (RSLs). These lines extend to the weak shock front marking the transition from the MRR to the boundary-reflected shocks (BRSs).

Comparing the three solutions using a variable h with the reference solution, one can easily see that for both snapshots, the standard SPH solution shows a significantly higher level of noise than the other solutions. The shape of the MCD and the MSL has clearly been distorted at $t = 0.015$. The TLRs starting where the MCD and the MSL intersect are not recognizable. The same applies to the narrow CCR. At $t = 0.025$, the SPH solution does a somewhat better job at reproducing the features seen in the reference solution, such as the triple points attached to the MSL, but still the noise level is substantial, and the symmetry in the problem has been broken. The RSLs are clearly visible only in the transition from the MRR to the BRSs. In contrast, RSPH using either CS or API both reproduce to a large extent the features seen in the reference solution. In particular, both solutions have produced the MCD and the MSL with kinks clearly intact. The cylindrical MWS, the narrow CCR, and the TLRs are all visible features at $t = 0.015$. At $t = 0.025$, both the CS and API solutions still maintain the symmetry in the problem and exhibit no problem with shock waves being distorted. The small-scale fluctuations near the top and right-hand boundaries have been slightly better captured in the API solution than in the CS solution. Other than that, it is difficult to see any difference in performance between CS and API based on these 2D plots.

To be able to compare the four solutions in more detail, we have in Fig. 13 plotted the density at $t = 0.015$ along the symmetry line as a function of the radial distance from the origin. The figure has been created by plotting all particles found at a distance of less than h from the symmetry line. The solid and dotted lines indicate the reference and standard SPH solutions, respectively, while the triangles and circles indicate the CS and API solutions, respectively. We have only plotted a relatively small interval containing the dominant interactions between the two blast waves. We see that the MWS, the MCD, the CCR and the MSL are located at approximate radial distances of 0.84, 0.90, 0.94, and 0.97, respectively. The reference solution reaches a peak density level in the CCR of about 15. Once again, one can see that the standard SPH code fails to produce more than a poor reproduction of the density profile found in the reference solution. This applies in particular to the interval covering the MCR. The density jumps associated with the MWS, the MCD, and the MSL can all be detected in the standard SPH solution, but the location of the MSL jump is shifted in space compared to that of the high-resolution solution. The CS solution reproduces all the density jumps seen in the reference solution and the density peak in the CCR. However, the peak

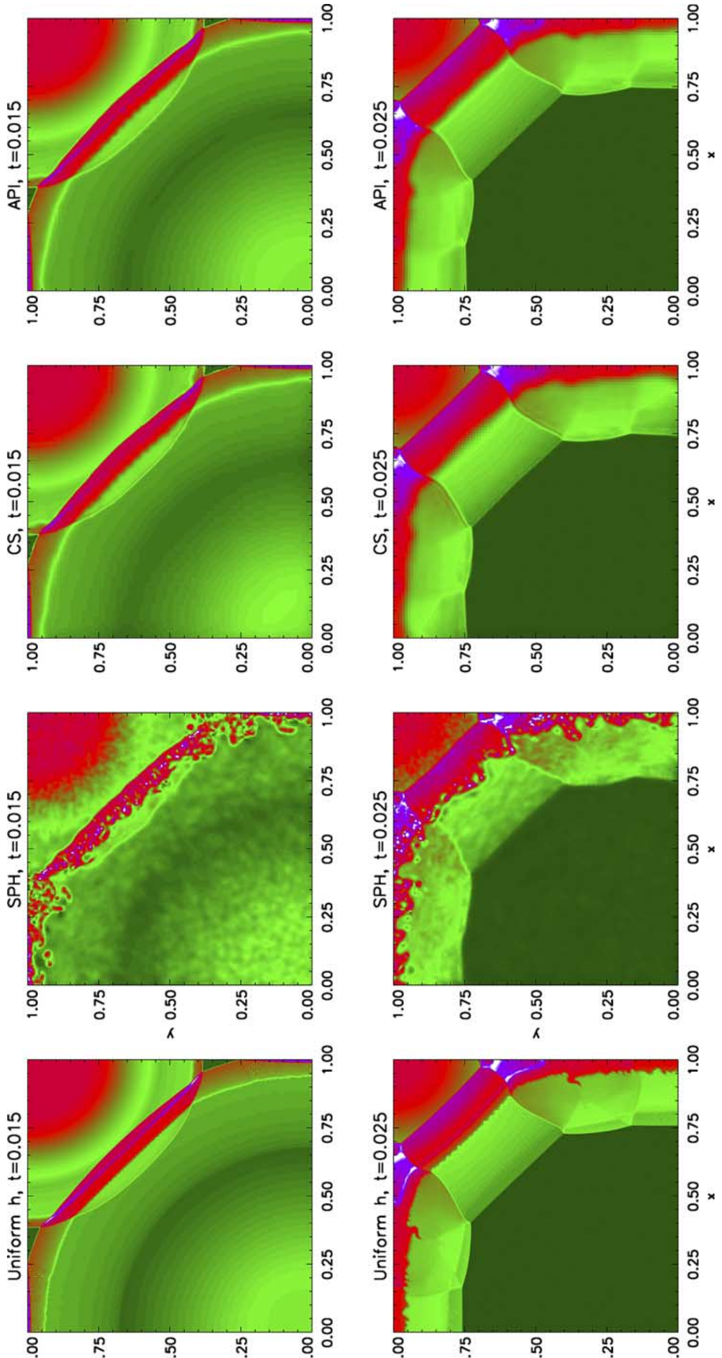

Fig. 12. Solutions of ρ at $t = 0.015$ and $t = 0.025$ of the two interacting, cylindrical blast waves using standard SPH, RSPH with API, RSPH with CS, and RSPH with uniformly high resolution.

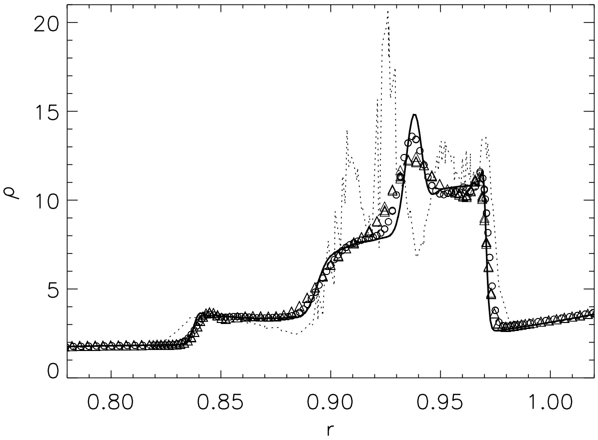

Fig. 13. Particles found closer than a distance h from the diagonal $x = y$ is plotted at $t = 0.015$. The solid and dotted lines indicate the reference RSPH solution and the standard SPH solution, respectively, while the triangles and circles indicate the RSPH solution with CS and API, respectively.

value is lower than in the reference solution, about 12.4. In addition, the slope leading up to the density peak is not steep enough for the jump associated with the MCD to be more than barely recognizable. The API solution, on the other hand, shows a density peak in the CCR of about 13.6. The density jump associated with the MCD is also more clearly defined than in the CS solution due to a steeper slope leading up to the CCR.

5. Discussion

In this paper, we propose a new technique used in RSPH for calculating the dynamics of particles found within interaction range of a step in h, referred to as auxiliary particle interpolation (API). Through the first three tests described in Section 4, API is compared to cell summation (CS), the technique originally used for handling steps in h. These are all severe tests of how well a piecewise constant h is handled. In particular, the smoothing length is allowed to vary in regions where the dynamics play an important part in determining the evolution of the simulated systems. They illustrate the limitations and errors connected with introducing steps in h when either CS or API is used. All three tests clearly show that the quality of the results improve significantly when CS is replaced by API. One might argue that the h-profiles used in these tests are far from being optimal and that the errors occurring in the test results using CS could be avoided by optimizing the h-profile. Although this in principle is true, in large scale simulations one is often forced to choose sub-optimal representations when taking computational costs into account. Having a robust and accurate way of handling steps in the resolution is therefore of great importance. For many applications where little or no artificial viscosity can be allowed, it is important that the amplitude of noise generated at a step in h is kept at a minimum. The linear wave tests presented in Section 4.2 are examples of applications where this is the case.

The mixed-resolution results presented in Sections 4.2–4.4 make use of the linear momentum correction introduced in Section 3.1. However, the importance of this correction varies considerably. For the case of CS, the additional errors introduced by not conserving the total linear momentum is relatively small compared to other error terms. For the case of API, the importance of the correction seems to decrease when choosing a higher interpolation order. Using zeroth or first-order API, the long-term stability of the linear waves described in Section 4.2 is dependent on the momentum correction. The importance of correcting the total linear momentum is less pronounced in the shock tests. In the case of second-order API, the effect of applying the linear momentum correction seems to be negligible for all the tests presented in Section 4. As seen in the Section 4.5, both CS and API can function well without the linear momentum correction when the *h*-profile is optimized according the problem at hand.

In addition to the improved results obtained when using API instead of CS, API benefits from being considerably simpler to implement. CS requires the equations of motion to be reformulated and summation cells, also referred to as template particles, to be generated at each time step [3]. No reformulation of the equations of motion is required when using API. The two methods are comparable when it comes to computational costs and memory usage. Using API requires solving a linear problem for each auxiliary particle at each step in time. The efficiency of the API scheme will therefore depend on the ratio of auxiliary particles to regular particles. It is, however, important to bear in mind that the number of auxiliary particles will typically be substantially smaller than the number of regular particles. For the shock type simulations described in Sections 4.3–4.5, the auxiliary particles constituted about 15–20% of the total number of internal particles. The efficiency of the API scheme was more or less the same for all tests, roughly 60%. This number indicate the ratio between the reduction in CPU time and the reduction in the time-averaged particle number when changing from an RSPH scheme with uniform *h* to an API scheme with corresponding minimum *h*. In other words, if the time-averaged particle number is reduced by a factor of 10 when using a variable *h*-profile instead of a uniform *h*-profile, the resulting reduction in CPU time should be roughly by a factor of 6.

The last test included in Section 4 describes the interaction of two cylindrical blast waves. The test is meant to illustrate the difference in performance between a standard SPH with variable *h* [12] and RSPH using either CS and API. The former code uses equal-mass particles and couple the smoothing length directly to the local particle number density. This approach gives the best overall resolution that the given particle distribution at any point in time can produce. However, it does not provide any means for reshaping the particle distribution. This lack of control over the *h*-profile is particularly critical when studying problems where important physical processes are taking place in areas of relatively low density, such as in the test described in Section 4.5. The RSPH code provide a substantial increase in accuracy compared to the standard SPH code for the blast wave test. The difference between the CS and API solutions is more subtle in this test than in the previous three test cases. This might come as a surprise considering that the last test would normally be regarded to be much harder than the previous tests. The explanation for the small difference between CS and API in this case is that the full functionality of RSPH is taken advantage of, including both particle regularization and *h*-profile optimization. This means that, unlike what was the case for the three previous tests, shocks or other dominating dynamical features are to very little extent allowed to propagate across steps in *h*. Whenever this is the case, the CS technique has previously been shown to work well [3]. This is confirmed in the blast wave test. Still, the API results show the new technique to be marginally better than CS even in this case.

As a Lagrangian method, SPH has demonstrated unique properties in dealing with a number of applications within astrophysics, geophysics, and industrial processes. Unfortunately, the method has been known to lack some of the accuracy and efficiency of state-of-the-art grid based methods. Regularized smoothed particle hydrodynamics (RSPH) was developed with the intention of constructing a SPH based method that could retain most of the unique and favourable properties of SPH, but which at the same time would offer increased accuracy, flexibility, and efficiency compared to standard SPH. Unlike template particles used in CS, auxiliary particles are generated in the same way as the regular particles. As noted in Section 3, the auxiliary particles are only replaced if or when the entire particle distribution is redefined. This might put an upper limit on how long a multi-resolution simulation can be accurately calculated in time before regularization must be performed. This could be especially important to consider for problems exhibiting large degrees of particle mixing. Note, however, that all simulations described in Sections 4.2–4.4 were performed without regularization, which should indicate that API is quite robust in this respect as well. By introducing the technique of API, the hope is therefore that we are one step closer to achieving a robust and accurate alternative to standard SPH. So far, the work has focused on various applications of compressible hydrodynamics or magnetohydrodynamics. It is our view that RSPH in the future can become of special interest as a numerical tool when studying fluid dynamical problems involving free surfaces or moving interfaces such as found in [10]. Work is also in progress to extend the method to incompressible flow.

References

[1] J. Bonet, S. Kulasegaram, Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations, Int. J. Numer. Methods Eng. 47 (2000) 1189.

[2] S. Børve, A numerical study of ionospheric and magnetospheric phenomena using particle simulation techniques, Ph.D. Thesis, University of Oslo, 2001.

[3] S. Børve, M. Omang, J. Trulsen, Regularized smoothed particle hydrodynamics: a new approach to simulating magnetohydrodynamic shocks, Astrophys. J. 561 (2001) 82.

[4] S. Børve, M. Omang, J. Trulsen, SPH simulations of MHD shocks using a piecewise constant smoothing length profile, in: Griebel, Schweitzer (Eds.), Proc. Int. Workshop Meshfree Meth. Partial Differential Equations, Bonn, Germany, 2001, Lecture Notes in Sci. Eng. Springer-Verlag, Berlin, 2002, p. 51.

[5] S. Børve, M. Omang, J. Trulsen, 2D MHD smoothed particle hydrodynamics stability analysis, Astrophys. J. Suppl. Ser. 153 (2004) 447.

[6] P.W. Cleary, J.J. Monaghan, Conduction modelling using smoothed particle hydrodynamics, J. Comput. Phys. 148 (1999) 227.

[7] G.A. Dilts, Moving-least-squares-particle hydrodynamics – I. Consistency and stability, Int. J. Numer. Methods Eng. 44 (1999) 1115.

[8] A. Evrard, Mon. Not. R. Astron. Soc. 235 (1988) 911.

[9] J. Fan, I. Gijbels, Local Modelling and Its Applications, Chapman & Hall, London, 1996, p. 57.

[10] J.-F. Gerbeau, T. Lelièvre, C. Le Bris, Simulations of MHD flows with moving interfaces, J. Comput. Phys. 184 (2003) 163.

[11] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: theory and applications to non-spherical stars, Mon. Not. R. Astron. Soc. 181 (1977) 375.

[12] L. Hernquist, N. Katz, TREESPH: a unification of SPH with the hierarchical tree method, Astrophys. J. Suppl. Ser. 70 (1989) 419.

[13] S. Inutsuka, Reformulation of smoothed particle hydrodynamics with Riemann solver, J. Comput. Phys. 179 (2002) 238.

[14] O. Kessel-Deynet, A. Burkert, Ionizing radiation in smoothed particle hydrodynamics, Mon. Not. R. Astron. Soc. 315 (2000) 713.

[15] L.D. Libersky, A.G. Petschek, Smoothed particle hydrodynamics with strength of materials, in: Trease, Fritts, Crowley (Eds.), Proc. Next Free-Lagrange Method, Jackson Hole, USA, June 1990, Lecture Notes in Phys., Springer-Verlag, Berlin, 1991, vol. 395, p. 248.

[16] W.K. Liu, S. Jun, Y.F. Zhang, Reproducing kernel particle method, Int. J. Numer. Methods Eng. 20 (1995) 1081.

[17] L.B. Lucy, A numerical approach to the testing of the fission hypothesis, Astron. J. 82 (1977) 1013.

[18] J.J. Monaghan, On the problem of penetration in particle methods, J. Comput. Phys. 82 (1989) 1.

[19] J.J. Monaghan, Smoothed particle hydrodynamics, Annu. Rev. Astron. Astrophys. 30 (1992) 543.

[20] J.J. Monaghan, Simulating free surface flows with SPH, J. Comput. Phys. 110 (1994) 399.

[21] J.P. Morris, Analysis of smoothed particle hydrodynamics with applications, Ph.D. thesis, Monash University, July 1996, p. 56.

[22] M. Nagasawa, K. Kuwahara, Smoothed particle simulations of the pyroclastic flow, Int. J. Mod. Phys. 7 (1993) 1980.

[23] R.P. Nelson, J.C.B. Papaloizou, Variable smoothing length and energy conservation in smoothed particle hydrodynamics, Mon. Not. R. Astron. Soc. 270 (1994) 1.

[24] M. Omang, S. Børve, J. Trulsen, Numerical simulation of shock–vortex interactions using regularized smoothed particle hydrodynamics, Comput. Fluid Dynamics J. 12 (2003) 258.

[25] J.M. Owen, J.V. Villumsen, P.R. Shapiro, H. Martel, Adaptive smoothed particle hydrodynamics: Methodology. II, Astrophys. J. Suppl. Ser. 116 (1998) 155.

[26] P.W. Randles, L.D. Libersky, Smoothed particle hydrodynamics: some improvements and applications, Comput. Methods Appl. Mech. Eng. 139 (1996) 375.

[27] P.W. Randles, L.D. Libersky, Normalized SPH with stress points, Int. J. Numer. Methods Eng. 48 (2000) 1445.

[28] J.C. Simpson, Numerical techniques for three-dimensional smoothed particle hydrodynamics simulations: applications to accretion disks, Astrophys. J. 448 (1995) 822.

[29] R.F. Stellingwerf, C.A. Wingate, Impact modelling with SPH, Soc. Astron. Italiana 65 (1994) 1117.