# Particle initialization through a novel packing algorithm

B. Bouscasse, M. Antuono, S. Marrone
CNR-INSEAN (The Italian Ship Model Basin)
Rome, Italy

A. Colagrossi
CNR-INSEAN (The Italian Ship Model Basin), Rome, Italy
CESOS (Centre of Excellence for Ship and Ocean Structures)
NTNU, Trondheim, Norway

***Abstract*—A novel particle packing algorithm has been derived basing on a simplified standard SPH scheme and is used to initialize the particle distribution for generic SPH solvers. Using some intrinsic features of the SPH schemes, the proposed algorithm leads to a final particle distribution that is very stable and is not affected by a further resettlement during the early stages of the evolution. Moreover, the computational costs maintain very low since the final configuration is attained very quickly. Finally, a strong point of the packing algorithm is that it can be easily derived using whatever SPH scheme.**

## I. Introduction

The matter of how initialize the particle positions in the SPH schemes plays a relevant role. Indeed, particles which are not initially set in "equilibrium" positions may resettle giving rise to spurious motions which can strongly affect the fluid evolution. Hereinafter we refer to an equilibrium configuration as the set of particle positions which, under static conditions, does not lead to particle resettlement.

Unfortunately, apart from few cases characterized by simple geometries, the equilibrium configuration is not known "a priori". This is a major issue since the generation of spurious currents/vorticity is particularly strong in presence of complex solid boundary profiles (i.e. corners, bended bodies, etc.).

In the SPH literature some ad-hoc procedures have been proposed to reduce particle resettlement. The simplest one consists in starting the SPH simulation using a high numerical viscosity and leaving a long enough time to make particle self-resettle in equilibrium positions (see, for example, Monaghan [1]). The actual SPH simulation (that is, the simulation with the correct viscosity and the desired initial conditions) starts after particles have reached an equilibrium configuration. Unfortunately, the attainment of a stable configuration can require a very long evolution, this leading to a large increase of computational costs. Moreover, the high viscosity used for particle initialization does not exclude that a further resettlement occurs when the actual simulation is started with smaller values of the viscosity.

In the SPH framework, the first attempt to define a proper algorithm for particle initialization is due to Oger et al. [2] who adapted the Bubble method described in Shimada [3] to SPH solvers. This algorithm is based on the use of Van der Waals-like forces to place particles throughout the fluid domain. This method proves to be quite fast, applies to general geometries and provides a regular particle distribution. One of the weak points is that the particle positions obtained through the Bubble algorithm may be not perfectly compatible with the SPH static solution leading to a further resettlement.

Then, the key point to build a robust packing algorithm relies on the capability of providing a regular particle distribution which is compatible with the SPH scheme, that is, that satisfies the static conditions when the SPH scheme is used. To this purpose a novel packing algorithm has been derived taking advantage of some intrinsic features of the SPH schemes. Thanks to this, the proposed method allows the attainment of a regular particle distribution compatible with the static solution. Further, it can be easily derived starting from whatever SPH solvers and applies to weakly-compressible or incompressible SPH schemes as well.

The paper is organized as follows: Section §II gives an insight of the constitutive features which are used to build the packing algorithm. Section §III describes the proposed algorithm and highlights some interesting aspects about its Lagrangian structure. Finally, Section IV provides a broad range of numerical test cases which prove the packing algorithm to be fast, robust and reliable also for complex geometrical configurations.

## II. Some intrinsic features of the SPH

In the present paper we adopt the standard SPH scheme:

$$\begin{cases} \dfrac{D\rho_i}{Dt} = -\rho_i \displaystyle\sum_j (\boldsymbol{u}_j - \boldsymbol{u}_i)\cdot \nabla_i W_{ij}\, V_j \\ \dfrac{D\boldsymbol{u}_i}{Dt} = \boldsymbol{g} - \dfrac{1}{\rho_i}\displaystyle\sum_j (p_j + p_i)\nabla_i W_{ij}\, V_j + \boldsymbol{T}_i^{(v)}, \\ p_i = F(\rho_i), \\ \dfrac{D\boldsymbol{r}_i}{Dt} = \boldsymbol{u}_i, \end{cases} \tag{II.1}$$

As usual, the subscripts indicate the quantities associated with the $i$-th and $j$-th fluid particle. In the specific, $V_i$ is the particle volume, $\rho_i = m_i/V_i$ and $m_i$ is the particle mass. The term $\boldsymbol{T}_i^{(v)}$ indicates an artificial viscous force per unit of mass. This term is generally implemented in the SPH schemes for stability reasons (see, for example, [4]). In the present work, we use the artificial viscous term proposed by Monaghan and Gingold

[7] but a different choice does not affect the results shown in the following.

For the analysis which follows, it is convenient to introduce the following variables:

$$\Gamma_i = \sum_j W_{ij} V_j \qquad \nabla\Gamma_i = \sum_j \nabla_i W_{ij} V_j \,. \tag{II.2}$$

Variables $\Gamma_i$ and $\nabla\Gamma_i$ give a "measure" of the unevenness in the particle distribution. In fact, if the particle distribution is perfectly uniform, $\Gamma_i = 1$ and $\nabla\Gamma_i = 0$ otherwise $\Gamma_i < 1$ and $\nabla\Gamma_i \neq 0$.

Further, the use of $\Gamma_i$ and $\nabla\Gamma_i$ helps understand the convergence of the discrete differential operators. In fact, two different kinds of errors are made when the exact differential formulas are substituted with the discrete smoothed formulas. One kind is due to the interpolation procedure (errors proportional to the smoothing length, $h$) while the other is caused by the approximation of continuous integrals with finite summations (see, for example, [11]). In the latter case the error decreases as the number of particles inside the kernel domain increases. Then, if the mean number of particles in the kernel domain is large enough, this error can be assumed to be smaller than $O(h)$. Denoting by $\langle\nabla p\rangle$ the SPH pressure gradient and using the standard convergence results for the SPH differential operators [9], we can write:

$$\begin{aligned}
\langle\nabla p\rangle_i &= \sum_j (p_j + p_i)\,\nabla_i W_{ij} V_j = \\
&= \sum_j (p_j - p_i)\,\nabla_i W_{ij} V_j + 2\,p_i \sum_j \nabla_i W_{ij} V_j = \\
&= \Gamma_i\,(\nabla p)_i + 2\,p_i\,\nabla\Gamma_i + O(h)\,.
\end{aligned} \tag{II.3}$$

Here, both $\Gamma_i$ and $\nabla\Gamma_i$ are responsible for a deviation from the exact pressure operator. As shown in the following example, this is the main cause of the particle resettlement.

Let us assume we want to start a SPH simulation with hydrostatic conditions. Then, we assign $\boldsymbol{u}_i = 0$, $\rho_i = \rho(\boldsymbol{r}_i)$ at $t = t_0$ and try to find the hydrostatic pressure. The continuity equation is satisfied exactly while, as a consequence of (II.3), the momentum equation gives:

$$\Gamma_i\,(\nabla p)_i + 2\,p_i\,\nabla\Gamma_i - \rho_i\,\nabla\Phi = O(h)\,, \tag{II.4}$$

where $\Phi$ is the gravitational potential. This expression clearly shows that, unless $\Gamma_i = 1$ and $\nabla\Gamma_i = 0$, it is not possible to attain any hydrostatic solution. In this context, $\Gamma_i$ plays only a minor role since it just causes an increase/decrease of the intensity of the correct pressure gradient. On the contrary, $\nabla\Gamma_i$ is responsible for an unbalance in both the intensity and the direction of the SPH differential operator. Further, $\nabla\Gamma_i$ diverges like $1/h$ when the particle distribution is strongly irregular. Then, the only way to get a good initialization of the SPH scheme is to reduce the magnitude of $\nabla\Gamma_i$ as much as possible. This is the principal idea at the basis of the Particle Packing Algorithm described in the next section.

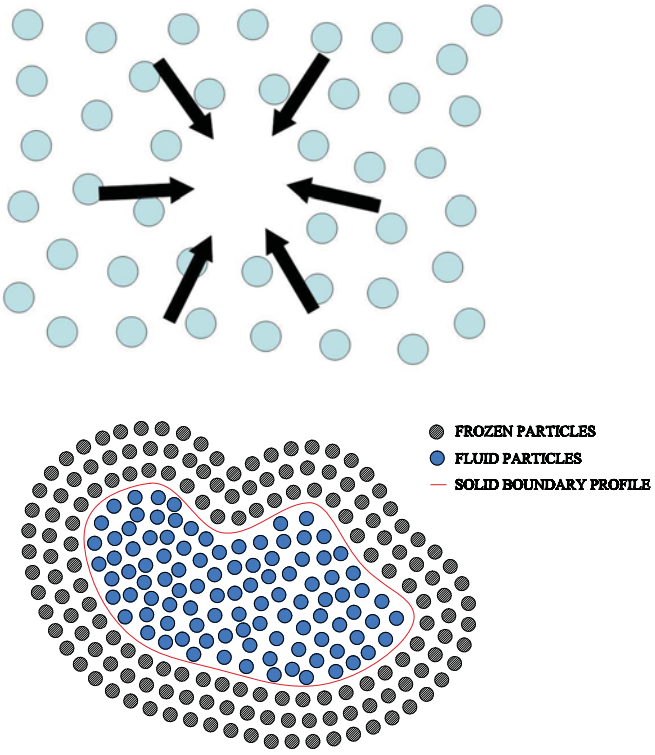


Fig. 1. Top panel: sketch of the vector $\boldsymbol{w} = -\nabla\Gamma$ in the neighborhood of a spatial anisotropy. Bottom panel: sketch of the solid and fluid particles in the packing algorithm framework.

## III. Particle Packing Algorithm

The Particle Packing Algorithm is built on a simple idea: to use the SPH features highlighted in the previous section to initialize the particle distribution and minimize $\|\nabla\Gamma\|$. This is made by observing that the vector $\boldsymbol{w} = -\nabla\Gamma$ always points in the direction of the maximum lack of "mass" and maximum anisotropy (see the top panel of figure 1). Now, let assume to use it to move particles during the initialization. If the fluid domain is bounded and particles are not allowed to escape form the boundaries, $\boldsymbol{w}$ tends to fill all the asymmetries in the particle distribution and, at the same time, it reduces as a consequence of the more regular distribution of particles themselves. Then, the final distribution would be the most regular possible and $\|\boldsymbol{w}\|$ (that is, $\|\nabla\Gamma\|$) would be minimized as requested.

The first step to build the Particle Packing Algorithm is to close the domain boundaries. As a consequence, this implies that the free surface has to be treated as a solid boundary. The domain boundary has to be modeled through fixed solid particles, that is, particles with zero velocity and fixed positions. This approach can be regarded as a special use of the frozen particles (for details see [12]) or as a straightforward application of the fixed ghost particle technique proposed by Marrone et al. [8]. A sketch of this procedure is displayed in the bottom panel of figure 1. Note that particles do not need any specific rule to be positioned inside the fluid domain nor

inside the solid bodies. The second step consists in assuming the density, the pressure and the volumes constant all over the fluid domain. We indicate them through symbols $\rho_0, p_0$ and $V_0$ respectively. Since volumes are constant and the packing algorithm has to converge towards a static solution, we neglect the continuity equation. Conversely, the momentum equation of system (II.1) becomes:

$$\begin{cases} \dfrac{D\boldsymbol{u}_i}{Dt} = -\beta \nabla\Gamma_i + \boldsymbol{T}_i^{(v)} \\ \dfrac{D\boldsymbol{r}_i}{Dt} = \boldsymbol{u}_i \end{cases} \tag{III.5}$$

where $\beta = 2\,p_0/\rho_0$ and $\Gamma_i = \sum_j W_{ij} V_0$. The viscous force can be chosen independently from the adopted SPH scheme since $\boldsymbol{T}_i^{(v)}$ is just used to ensure the convergence of the Particle Packing Algorithm. In the specific, we choose:

$$\boldsymbol{T}_i^{(v)} = -\nu\,\boldsymbol{u}_i \qquad \text{with} \qquad \nu = \alpha\,\frac{\sqrt{\beta}}{V_0^{1/d}} \tag{III.6}$$

where $d$ is the spatial dimension and $\alpha$ is a free dimensionless parameter. By numerical simulations we found that a good choice for $\alpha$ ranges between $1 \cdot 10^{-3}$ and $5 \cdot 10^{-3}$. Then, the Particle Packing system becomes:

$$\begin{cases} \dfrac{D\boldsymbol{u}_i}{Dt} = -\beta \nabla\Gamma_i - \nu\,\boldsymbol{u}_i \\ \dfrac{D\boldsymbol{r}_i}{Dt} = \boldsymbol{u}_i\,. \end{cases} \tag{III.7}$$

The initial conditions for the Particle Packing Algorithm are obtained by setting all the particle velocities to zero and $V_0 = V_{tot}/N_{part}$ where $V_{tot}$ is the total fluid volume and $N_{part}$ is the total number of particles. The time-step adopted for the present algorithm is:

$$\Delta t = CFL\,\frac{V_0^{1/d}}{\sqrt{\beta}}\,, \tag{III.8}$$

where $CFL = 1$. Because of its structure, system (III.7) tends to converge as much as possible towards a steady state characterized by $\boldsymbol{u}_i = 0$ and $\nabla\Gamma_i = 0$. When the fluid system is sufficiently close to this state, the particle positions are used to initialize the SPH simulations. Since the spatial distribution is very regular, the particle volumes can be assumed to be identical. Then, the volume used for the initialization of the SPH is $V^* = V_0$. The initial particle pressure, $p_i^*$, is assigned by using the analytical expression for the hydrostatic pressure and the particle positions. Then, inverting the state equation, the initial density $\rho_i^*$ is computed and, finally, the particle mass is obtained through $m_i^* = \rho_i^*/V^*$. During the SPH simulations, the particle masses are kept constant while the densities and the volumes are updated using the continuity equation and the relation $V_i = m_i/\rho_i$.

## IV. Applications

In the present section we show some applications of the particle packing algorithm. We first deal with the initialization of hydrostatic conditions in complex geometrical configurations, then we show some dynamical problems, that is, problems in which the fluid evolves after the particle packing initialization. In all the simulations, the standard SPH scheme has been used (see [4] for more details) and solid profiles have been modeled through the fixed ghost particles described in Marrone et al. [8]. In any case, the qualitative results obtained in the following also hold true for those SPH scheme that implement frozen particles.

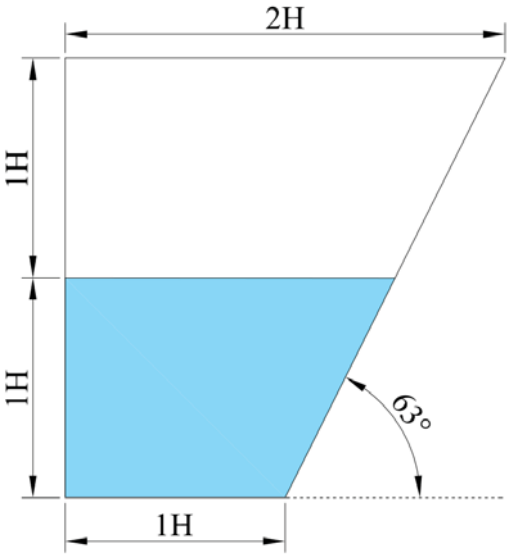


Fig. 2. sketch of the trapezoidal tank.

### A. Hydrostatic problems

We first consider a trapezoidal tank like that drawn in figure 2 ($H$ is the filling height) and study the influence of the particle initialization on the capability of the SPH of simulating the hydrostatic solution. As stated in Section III, the first step is to "close" the fluid domain. This means that the free surface has to be substituted by a solid boundary and modeled accordingly. This procedure is displayed in figure 3 where the fluid domain has been initialized through a Cartesian grid (left panel) and using the packing algorithm (right panel).

In the former case, the use of a Cartesian grid leads to the generation of large spatial anisotropies along the inclined plane. Here, $\|\nabla\Gamma\| = O(1)$ and, therefore, an intense particle resettlement is expected during the early stages of the fluid evolution. Conversely, the particle packing algorithm eliminates the spatial anisotropies and drastically reduces the magnitude of $\|\nabla\Gamma\|$ (whose order of magnitude is about $10^{-13}H$). Incidentally, we note that it is possible to derive SPH schemes which intrinsically correct the spatial anisotropies close to the solid profiles through the use of special boundary conditions (see, for example, [13]). However, these schemes

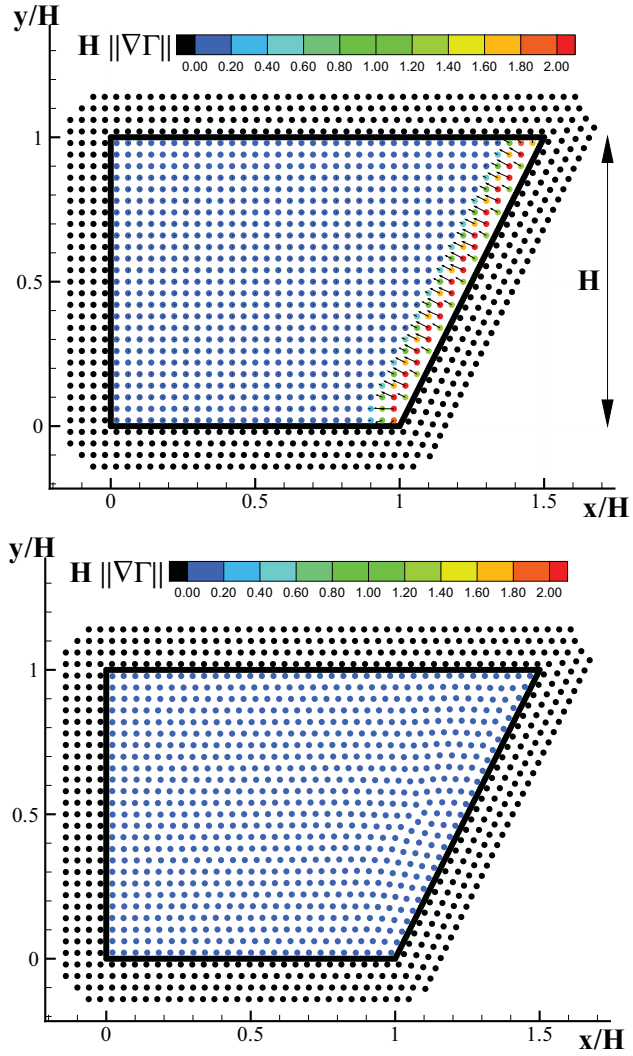


Fig. 3. trapezoidal tank ($H/dx$ = 25). Initialization using a Cartesian grid (top) and through the particle packing algorithm (bottom).

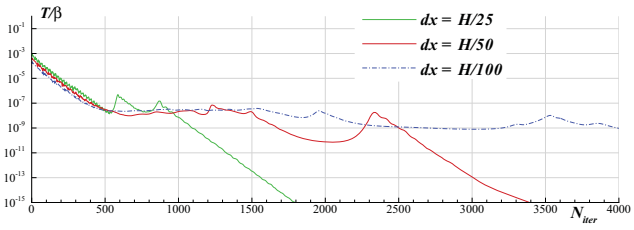


Fig. 4. trapezoidal tank. Evolution of the specific kinetic energy during the initialization through the particle packing algorithm.

are generally more complex than the standard SPH model and leads to higher computational costs.

It is also interesting to give a brief insight on the computational costs of the proposed algorithm and on its dependence on the spatial resolution. In figure 4 the specific Kinetic energy of the packing scheme (that is, $\mathscr{T}$) is displayed for three different spatial resolutions versus the number of iterations. This heuristically shows that after 2, 500 iterations, the particle are practically motionless (the order of magnitude of the specific kinetic energy is about $10^{-8}\beta$). This means that the equilibrium configuration has been attained and that the packing algorithm can be stopped. Obviously, the number of iterations required for the attainment of equilibria may vary according to the specific problem at hand, to the choice of the kernel function (i.e., Gaussian, cubic spline, quintic spline etc.) and to the adopted viscosity but generally range between 2, 000 – 2, 500 iterations.

When the particle initialization is complete, the hydrostatic solution is assigned to the fluid domain (that is, hydrostatic pressure field and zero initial velocity) and we start the simulation through the standard SPH scheme [4].

As shown in the left panel of figure 5, the initialization through the Cartesian grid, because of the high values of $\|\nabla\Gamma\|$ near the sloping plane, leads to the generation of high spurious currents which, on the contrary, are completely absent when the simulations is initialized through the particle packing algorithm (right panel of the same figure).

A global measure of the particle resettlement phenomenon is easily obtained by inspecting the kinetic energy evolution during the simulation of the hydrostatic solution. As shown in figure 6, the kinetic energy of the SPH simulation after the use of packing algorithm is at least two orders of magnitude smaller than the simulation started on the Cartesian grid. In the former case the dimensionless kinetic energy is of order $10^{-7}$ – $10^{-8}$ proving that the fluid particles are practically motionless. On the contrary, the simulation started on a Cartesian grid shows a particle motion which still persists at $t = 100\sqrt{H/g}$.

As already mentioned in the Introduction, an alternative solution to reduce particle resettlement is to start the SPH simulation using a high numerical viscosity and leave a long enough time to make particle self-resettle in equilibrium positions. The actual numerical simulation starts after the equilibrium configuration is attained. Here, we show that such a procedure (that is, the initialization using the SPH scheme itself) only leads to minor improvements. Following Monaghan [1], we use the standard SPH scheme with a linear viscous term identical to that adopted in the Packing algorithm. In this case, the dimensionless viscosity has the same order of magnitude of that used for the Packing simulation, that is, $O(10^{-3})$. The evolution during the particle initialization is displayed in figure 7. Note that particles are still moving after 10, 000 iterations (the dimensionless kinetic energy is of order $10^{-5}$) while, for the same configuration, the proposed packing algorithm reaches an equilibrium after about 2, 500 iterations (see figure 4). Moreover, when the actual simulation starts, particles are subjected to a further resettlement (see figure 8). In this case the global motion is weaker than that observed starting on a Cartesian grid but is considerably stronger with respect to the simulation which starts after the particle packing initialization.

As a second example, we consider a complex geometry characterized by bended profiles with different curvatures and

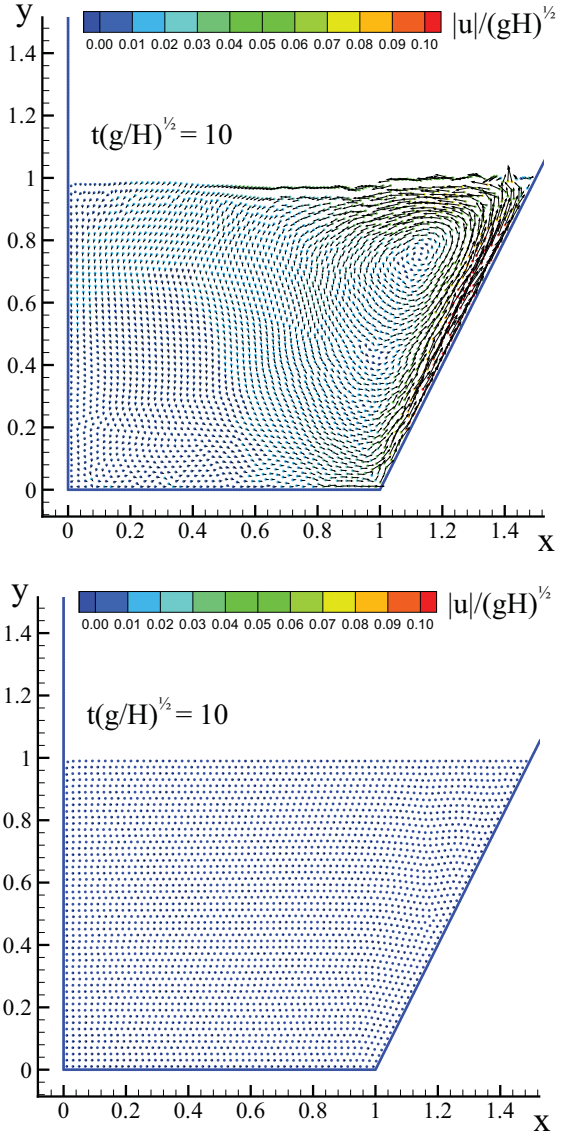


Fig. 5. hydrostatic solution for the trapezoidal tank ($H/dx$ = 50). Evolution using a Cartesian grid (top) and after the initialization through the particle packing algorithm (bottom).

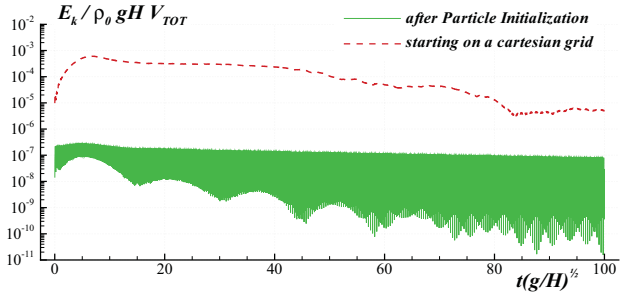


Fig. 6. hydorstatic solution for the trapezoidal tank ($H/dx$ = 50). Time history of the kinetic energy.

by acute and obtuse solid angles (see figure 9). Because of these features, the particle initialization of such a geometry represents a very difficult problem.

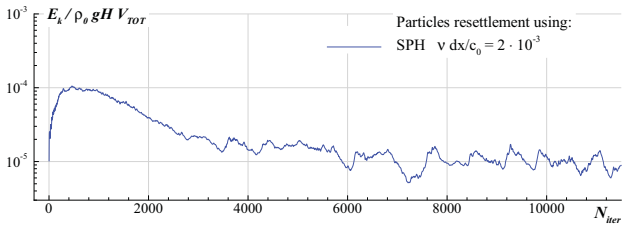


Fig. 7. kinetic energy evolution during the initialization with the standard SPH scheme.

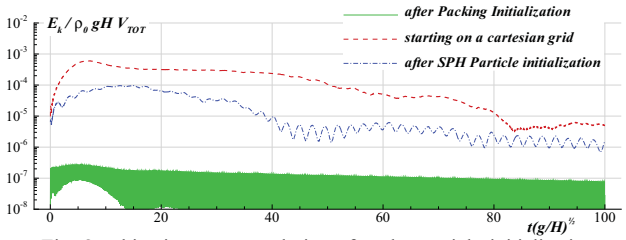


Fig. 8. kinetic energy evolution after the particle initialization.

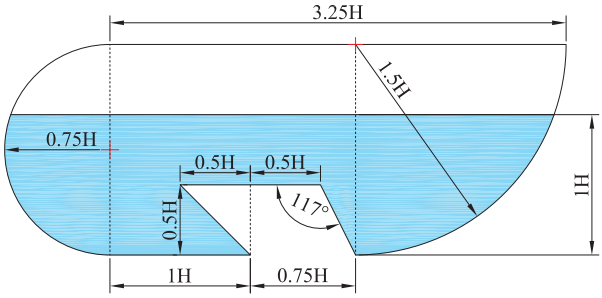


Fig. 9. sketch of the complex tank geometry.

The top panel of figure 10 displays the fluid evolution under hydrostatic conditions after the initialization on a Cartesian grid. In this case, the generation of spurious currents and vorticity near corners and bended profiles is very strong and persists for long times. On the contrary, the use of the particle packing algorithm eliminates such an undesirable behavior and gives a uniform particle distribution which keeps stable for long times (middle panel of figure 10) . The evolution of the kinetic energy confirms the findings above proving that, after the use of the proposed algorithm, particles are almost motionless (bottom panel of figure 10).

## *B. Dynamical problems*

Let us consider a fluid domain Ω which at the initial time is a two-dimensional ball of radius $R$, subjected to the velocity field:

$$\begin{cases} u_0(x,y) = A_0\, x \\ v_0(x,y) = -A_0\, y \end{cases} \quad \Rightarrow \quad \nabla \boldsymbol{u}_0 = \begin{pmatrix} A_0 & 0 \\ 0 & -A_0 \end{pmatrix}. \quad \text{(IV.9)}$$

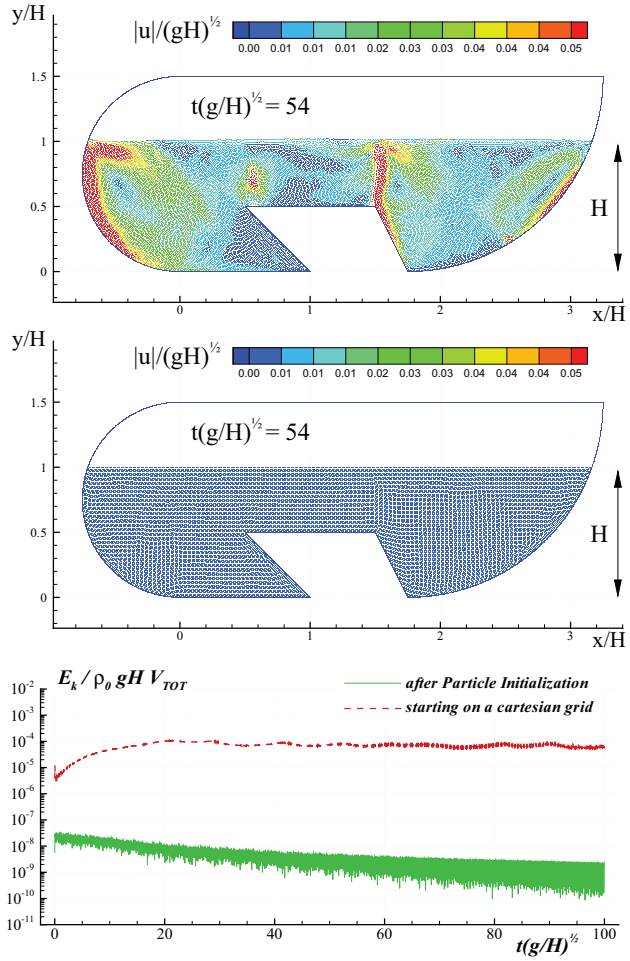


Fig. 10. hydrostatic solution for a complex tank geometry. Top: evolution after initialization on a Cartesian grid. Middle: evolution after initialization through packing algorithm. Bottom: time history of the kinetic energy.

The initial pressure field is derived using the Poisson equation [14] and reads:

$$p_0(x,y) = \frac{\rho_0 A_0^2}{2}\left[R^2 - (x^2+y^2)\right]. \qquad \text{(IV.10)}$$

Assuming the flow to be inviscid, Ω preserves an elliptical form during the evolution and this form can be derived analytically (see [1] and [14] for details). This domain is initialized using a Cartesian grid (top panel of figure 11) and the proposed particle packing algorithm (bottom panel of the same figure). Since the SPH is a Lagrangian scheme, particles move along stream lines. Consequently, when particles are initially set on a Cartesian grid, the flow evolution given by (IV.9) leads particles to clump along straight lines (see top panel of figure 12). This partially prevents the SPH solution to match with the analytical solution for the domain boundary (dashed lines in figure 12). On the contrary, the flow evolution after the use of the packing algorithm displays a more uniform particle distribution and, consequently, leads to a better agreement with the analytical solution (bottom panel of figure 12).

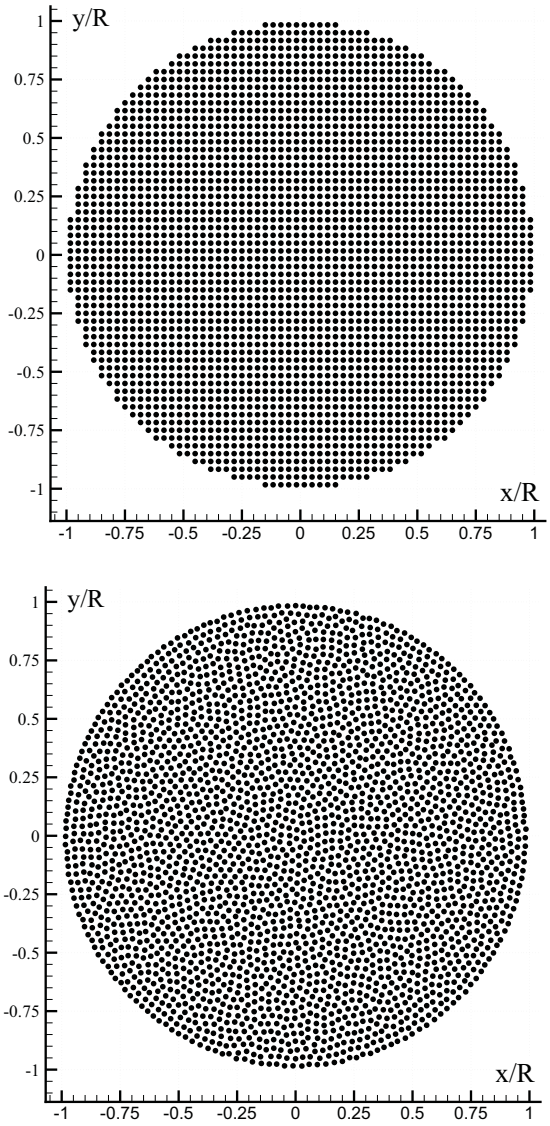


Fig. 11. evolution of an elliptical drop. Initialization using a Cartesian grid (top) and the particle packing algorithm (bottom).

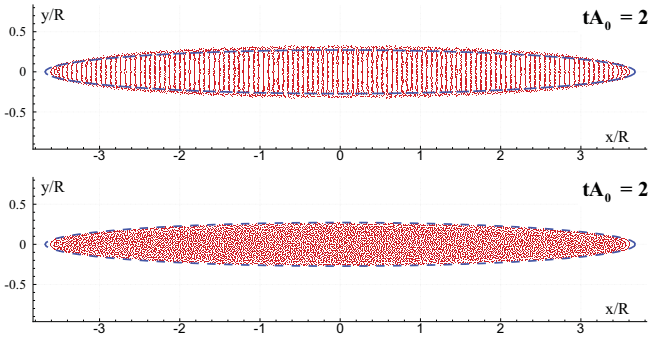


Fig. 12. evolution of an initially circular patch of fluid using a Cartesain grid (top) and the particle packing algorithm (bottom). Dashed lines indicate the analytical solution for the domain boundary.

As a second example, we consider a ship hull section

floating in hydrostatic conditions. Under such a hypothesis, the hull section should maintain motionless. However, because of the particle resettlement, an unphysical deviation of the ship hull from the initial position may be observed. Similarly to the test cases studied in the previous sections, we initialize the fluid domain using a Cartesian grid (top panel of figure 13) and the packing algorithm (bottom panel of the same figure). Figure 14 display the related SPH simulations. In this case, the spurious currents that generate in the neighborhood of the hull because of the Cartesian grid force the ship hull to move. Figure 15 shows the motion of the mass center (top panel) and the roll motion (bottom panel) of the ship hull. Because of the reduction of the spurious currents, the packing algorithm drastically reduces the unphysical ship motion ensuring the attainment of the correct hydrostatic solution.

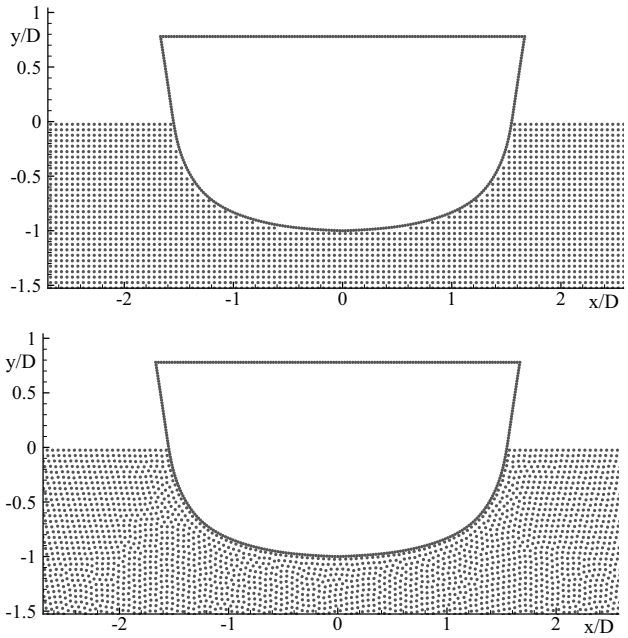


Fig. 13. freely floating of a ship hull section. Initialization on a Cartesian grid (top panel) and with the particle packing algorithm (bottom panel).

## Conclusions

Using some intrinsic features of the SPH scheme, a novel packing algorithm has been derived for the particle initialization. The proposed algorithm has been validated again several tests cases proving to be robust, fast and reliable also for complex geometrical configurations. As shown for the evolution of the elliptical drop, the particle distribution obtained through the packing algorithm may even avoid the formation of those filamentous structures that are caused by the Lagrangian nature of the SPH.

## Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n. 225967 "NextMuSE". This work was also partially supported by the Centre of Excellence for Ship and Ocean Structures of NTNU Trondheim (Norway) within the "Violent Water-Vessel Interactions and Related".

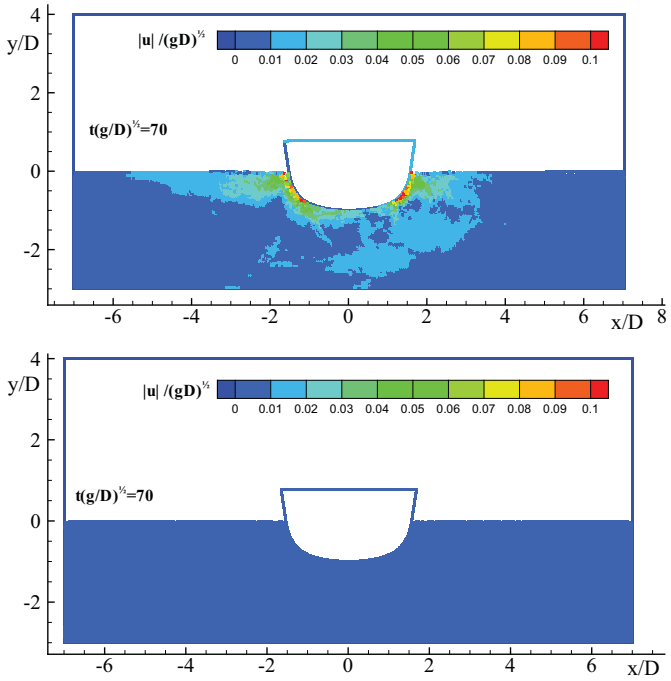


Fig. 14. freely floating of a ship hull section after initialization on a Cartesian grid (top panel) and with the particle packing algorithm (bottom).

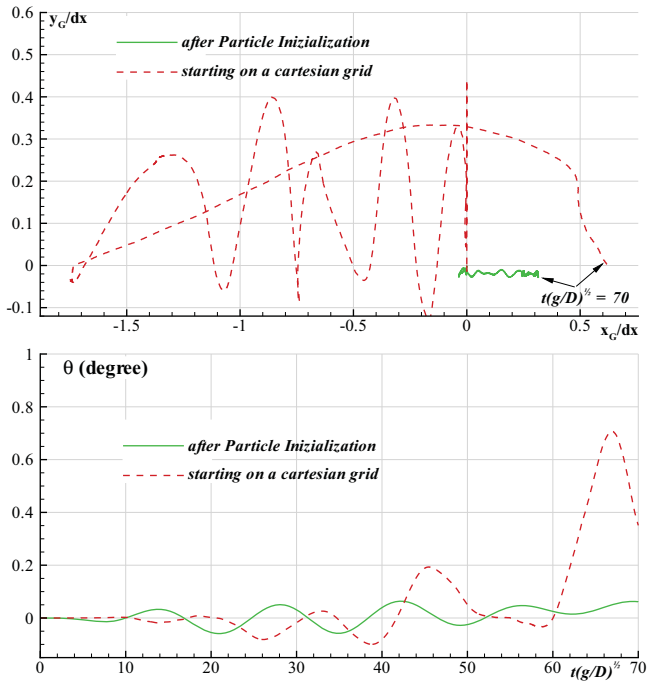


Fig. 15. freely floating of a ship hull. Top panel: motion of the mass center. Bottom panel: roll motion.

## References


[1] J.J. Monaghan, *Simulating free surface flows with SPH*, Journal of Computational Physics 110 (1994) 399–406

[2] G. Oger, C. Leroy, E. Jacquin, D. Le Touzé, B. Alessandrini, *Specific pre/post treatments for 3-D SPH applications through massive HPC simulations*, Proccedings 4th Spheric Workshop on SPH, Nantes, May 26-29 (2009), France

[3] K. Shimada, *Physically-based mesh generation; automated triangularization of surface and volumes via Bubble packing*, Phd thesis, MIT, (1993)

[4] J.J. Monaghan, *Smoothed particle hydrodynamics*, Rep. Progr. Phys. 68 (2005) 1703-1759.

[5] S.D. Shao, *Incompressible SPH simulation of wave breaking and overtopping with turbulence modeling*, Int. J. Num. Methods in Fluids 50(5) (2006) 597-621

[6] P.A. Madsen, H.A. Shäffer , *A discussion of artificial compressibility*, Coast. Eng. 53 (2006) 9398

[7] J.J. Monaghan, R.A. Gingold, *Shock Simulation by the Particle Method SPH*, J. Comp. Phys., 52 (1983) 374–389

[8] S. Marrone, M. Antuono, A. Colagrossi, G. Colicchio, D. Le Touzé, G. Graziani, *δ-SPH model for simulating violent impact flows*, Comput. Methods Appl. Mech. Engrg. 200 (2011) 1526-1542

[9] A. Colagrossi, M. Antuono, D. Le Touzé, *Theoretical considerations on the free surface role in the SPH model*, Phys. Rev. E 79 (5) (2009) 113 (056701).

[10] J. Bonet, T.S.L. Lok, *Variational and momentum preservation aspects of Smoothed Particle Hydrodynamics formulations*, Comput. Methods Appl. Mech. Eng. 180 (1999) 97-115

[11] N. J. Quinlan, M. Basa, M. Lastiwka, *Truncation error in mesh-free particle methods*, Int. J. Numer. Meth. Engng. 66 (2006) 2064-2085

[12] M. Liu, G. Liu, *Smoothed Particle Hydrodynamics (SPH): an Overview and Recent Developments*, Arch. Comput. Methods Eng. 17(1) (2010) 25-76

[13] M. Ferrand, D. Laurence, B. D. Rogers, D. Violeau, *Improved time scheme integration approach for dealing with semi analytical boundary conditions with SPARTACUS2D*, 5th International Spheric Workshop, 22-25 June, 2010, Manchester

[14] A. Colagrossi, *A Meshless Lagrangian Method for Free-Surface and Interface Flows with Fragmentation*, PhD Thesis, Università “La Sapienza”, Roma