

The ALE/Lagrangian Particle Finite Element Method: A new approach to computation of free-surface flows and fluid–object interactions

Facundo Del Pin ^a, Sergio Idelsohn ^{a,b,*}, Eugenio Oñate ^a, Romain Aubry ^a

^a *International Center for Numerical Methods in Engineering (CIMNE), Universidad Politécnica de Cataluña, Campus Norte UPC, 08034 Barcelona, Spain*

^b *International Center for Computational Methods in Engineering (CIMEC), Universidad Nacional del Litoral, Güemes 3450, 3000 Santa Fé, Argentina*

Received 1 June 2005; accepted 22 June 2005

Available online 9 December 2005

Abstract

The Particle Finite Element Method (PFEM) is a well established numerical method [Aubry R, Idelsohn SR, Oñate E, Particle finite element method in fluid mechanics including thermal convection–diffusion, *Comput Struct* 83 (2004) 1459–75; Idelsohn S, Oñate E, Del Pin F, A Lagrangian meshless finite element method applied to fluid–structure interaction problems, *Comput Struct* 81 (2003) 655–71; Idelsohn SR, Oñate E, Del Pin F, The particle finite element method a powerful tool to solve incompressible flows with free-surfaces and breaking waves, *Int J Num Methods Eng* 61 (2004) 964–84] where critical parts of the continuum are discretized into particles. The nodes treated as particles transport their momentum and physical properties in a Lagrangian way while the rest of the nodes may move in an Arbitrary Lagrangian–Eulerian (ALE) frame. In order to solve the governing equations that represent the continuum, the particles are connected by means of a Delaunay Triangulation [Idelsohn SR, Oñate E, Calvo N, Del Pin F, The meshless finite element method, *Int J Numer Methods Eng* 58–4 (2003)]. The resulting partition is a mesh where the Finite Element Method is applied to solve the equations of motion. The application of a fully Lagrangian formulation on the particles provides a natural and simple way to track free surfaces as well as to compute contacts in an accurate and robust fashion. Furthermore, the usage of an ALE formulation allows large mesh deformation with larger time steps than the full Lagrangian scheme.

© 2005 Elsevier Ltd. All rights reserved.

1. Introduction

In the last years a large amount of energy has been invested to accurately solve fluid mechanics problems with free surfaces, large deformation of parts of the fluid domain, fluid fragmentation and interaction with structures. Different methods have been developed to track interfaces or free surfaces. Among some of them we

can mention the Marker Particles Method [5,6], the High Functions Method [7], the Line Segments Method [8] and the Volume of Fluid Method (VOF) [9,10]. Although these methods have been used successfully in a variety of problems, their range of application is limited either because they are not able to manage over folding regions or because too large a mesh has to be built to cover the whole domain where the fluid may move. The VOF method is worthless unless an algorithm can be devised for accurately computing the evolution of the function that defines the interfaces. Another branch of methods included in the interface-tracking techniques are the deformable-spatial-domain/stabilized space–time

* Corresponding author. Tel.: +54 342 455 6673; fax: +54 342 455 0944.

E-mail addresses: fdelpin@cimne.upc.es (F.D. Pin), sergio@ceride.gov.ar (S. Idelsohn).

(DSD/SST) method [11,12]. A more recent version of this method increase the accuracy with the enhanced-discretization interface-capturing technique (EDICT) [13].

A different approach to solve interfaces would be to have a partition that moves and adapts to the deformations according to the equations of motion. This kind of approach is obtained when the equations of motion are written in an Arbitrary Lagrangian–Eulerian (ALE) way. From a physical point of view a full Lagrangian formulation provides a more natural way to solve interfaces when large deformations occur. Nevertheless, the remaining part of the domain may be solved with an ALE formulation.

In what follows summation convention over repeated indices will be used to describe the equations. Lagrangian formulations take place when the transport equation is written in a material fashion, avoiding the convective terms. By the chain rule of differentiation the following relation is found:

$$\frac{DQ}{Dt} = \frac{\partial Q}{\partial t} + \frac{\partial Q}{\partial x_i} \frac{dx_i}{dt}, \quad (1)$$

where Q is the property to transport and x_i is the spatial coordinate. In Eq. (1) the property Q may be transported either by directly moving the volume of control or by leaving the volume of control fixed and convecting Q by means of convective terms or by a mix of both.

One of the first methods to successfully apply this approach in fluid mechanics was the Smooth Particle Hydrodynamics (SPH) Method [14–17]. The remarkable simplicity of SPH and its accurate way to solve free surfaces inspired other researchers to develop numerical methods based on a Lagrangian description. For instance it may be mentioned the Moving Particle Semi-Implicit Method (MPS) [18] and a method proposed by Ramaswamy and Kawahara [19] where a Finite Element approximation is applied over a deformable mesh with fixed topology.

In the present paper the governing equations for an incompressible flow are approximated using a particle formulation for the free surfaces, fluid–solid contacts and any interface, while an ALE formulation are employed in the rest of the domain. The ALE treatment of internal nodes avoids the mesh deformation due to vortexes or large velocity gradients. Thus, a larger time step can be used without the mesh distortion that would be observed in the full Lagrangian case.

The numerical method chosen for the discrete problem is the Particle Finite Element Method (PFEM) [1–3]. In this method the FEM is used to write the weak form of the equations of motion. The continuum domain is partitioned according to the Extended Delaunay Tessellation (EDT) [4] providing an acceptable support in two and three dimensions where the weak form is integrated. The alpha-shape [20] technique is applied to solve the free surfaces and the contacts. In

a particle formulation all the physical properties are convected together with the nodes of the partition. Thus fluid–fluid or solid–fluid interfaces are naturally captured.

The outline of this work is as follows: Section 2 introduces the ALE equations of motion, Section 3 describes the time integration scheme, Section 4 provides the FEM discretization, Section 5 briefly explains the coupled problem. In Section 6 the mesh moving algorithm is explained. In Section 7 examples of applications and validation problems are presented. Finally in Section 8 a brief conclusion is given.

2. ALE equations of motion

2.1. Fluid mechanics equations

Conservation of momentum and mass for incompressible Newtonian fluids in the Eulerian conventional form are represented by the Navier–Stokes equations and the continuity equation, as follows:

$$\rho \left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) = \frac{\partial \sigma_{ij}}{\partial x_j} + \rho f_i \quad \text{in } \Omega, \quad (2)$$

$$\frac{\partial u_i}{\partial x_i} = 0 \quad \text{in } \Omega, \quad (3)$$

where $u_i = u_i(x_i, t)$ are the components of the velocity field in a spatial coordinate system x_i , ρ is the fluid density, f_i is the acceleration due to external volumetric forces and t is time. The total stress tensor σ_{ij} is given by

$$\sigma_{ij} = -p\delta_{ij} + \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_l}{\partial x_l} \delta_{ij} \right), \quad (4)$$

where p is the pressure and μ is the coefficient of viscosity. For nearly incompressible flows we obtain:

$$\frac{\partial \sigma_{ij}}{\partial x_j} = -\frac{\partial p}{\partial x_j} \delta_{ij} + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right). \quad (5)$$

The above expression simplifies the viscous term, making Eq. (2) applicable only for incompressible flows.

In the case of a moving reference frame (ALE) and χ being the coordinate with respect to the moving frame, three cases may be mentioned:

1. the moving reference frame moves with the particle: $\chi = X$, thus the full Lagrangian formulation is found;
2. the moving reference frame is fixed in space $\chi = x$ and the full Eulerian formulation is achieved;
3. the moving reference frame moves with an arbitrary velocity and then, the ALE formulation [21] is retrieved.

Expanding the left hand side of Eq. (1) the three cases may be written as follows:

$$\frac{Du_i}{Dt} = \frac{du_i}{dt} + \frac{\partial u_i}{\partial x_j}(u_j - v_j), \quad (6)$$

where Du_i/Dt means the total derivative of the velocity; $\partial u_i/\partial t$ means the partial derivative of the velocity; du_i/dt is the local derivative of the velocity with respect to a moving frame and v_j is the velocity of the moving frame.

Rewriting the equations of motion for an ALE formulation:

$$\begin{aligned} \rho \left(\frac{du_i}{dt} + (u_j - v_j) \frac{\partial u_i}{\partial x_j} \right) \\ = -\frac{\partial p}{\partial x_j} \delta_{ij} + \mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right) + \rho f_i \quad \text{in } \Omega, \end{aligned} \quad (7)$$

$$\frac{\partial u_i}{\partial x_i} = 0 \quad \text{in } \Omega. \quad (8)$$

In this formulation, the coordinates x_i appear as dependent variables while the independent variables are the ALE coordinates χ_i . Both frames of reference are related by the deformation mapping Φ such that:

$$x_i = \Phi_i(\chi_i, t). \quad (9)$$

The derivatives of Φ with respect to the ALE coordinates are defined by the matrix:

$$F_{iJ} = \frac{\partial x_i}{\partial \chi_j} = \frac{\partial \Phi_i(\chi_i, t)}{\partial \chi_j}, \quad (10)$$

where the matrix F are the deformation gradients and the index J indicates that the derivatives are computed with respect to the ALE coordinates.

The ALE description of the equations of motion given by Eq. (7) is not complete as the space derivatives are still written on the reference configuration. To overcome this last step of the formulation we have to take into consideration the dependence between x_i and χ_i . So by the chain rule of derivation and by means of Eq. (10) we may write:

$$\frac{\partial q}{\partial x_i} = \frac{\partial q}{\partial \chi_i} \frac{\partial \chi_i}{\partial x_j} = \frac{\partial q}{\partial \chi_i} F_{iJ}^{-1} = F_{Ji}^{-1} \frac{\partial q}{\partial \chi_i}, \quad (11)$$

for any given scalar quantity q . This expression gives a fundamental relation for the ALE formulation as it links the reference configuration with the deformed configuration. In Eq. (7) the gradients of pressure will be rewritten in terms of Eq. (11) simply by replacing q by p .

Eq. (7) includes another spatial derivative that is associated with the viscous terms. In this case the classical Piola transform of a first order tensor applied to the divergence operator results in

$$\frac{\partial r_j}{\partial x_j} = \frac{1}{J} \frac{\partial}{\partial \chi_j} (J F_{Jj}^{-1} r_j), \quad (12)$$

where $J = \det F$ is the Jacobian of the deformation. The viscous term may be expressed in the deformed configuration as

$$\mu \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} \right) = \mu \frac{1}{J} \frac{\partial}{\partial \chi_j} \left(J F_{Ji}^{-1} F_{Jj}^{-1} \frac{\partial u_i}{\partial \chi_j} \right). \quad (13)$$

Replacing the new expressions found for the derivatives in terms of the deformed configuration and reordering the indices we may rewrite Eq. (7) to get the final ALE equations of motion:

$$\begin{aligned} \rho \left(\frac{\partial u_i}{\partial t} \Big|_{\chi_i} + (u_j - v_j) F_{Jj}^{-1} \frac{\partial u_i}{\partial \chi_j} \right) \\ = -F_{Ji}^{-1} \frac{\partial p}{\partial \chi_i} \delta_{ij} + \mu \frac{1}{J} \frac{\partial}{\partial \chi_j} \left(J F_{Ji}^{-1} F_{Jj}^{-1} \frac{\partial u_i}{\partial \chi_j} \right) + \rho f_i, \end{aligned} \quad (14)$$

$$F_{Ji}^{-1} \frac{\partial u_i}{\partial \chi_i} = 0. \quad (15)$$

The last equation is strictly a representation of an incompressible Newtonian fluid flow where the frame of reference moves in an arbitrary way. This is called the ALE frame of reference.

2.2. Solid mechanics equations

In this work the structure is considered as a rigid body. The motion consists in the superposition of the translation produced by the resultant force F_i and the rotation produced by the couple T_i . The body must satisfy:

$$m \frac{Du_i}{Dt} = F_i, \quad (16)$$

$$I_i \frac{D\Omega_i}{Dt} + \epsilon_{ijk} e_i \Omega_j (I_k \Omega_k) = T_i, \quad (17)$$

where m is the mass of the body, u is the linear velocity, I are the principal moments of inertia, Ω is the angular velocity with respect to body fixed axes, ϵ is the permutation symbol, e are orthogonal unit basis vectors and $M_i = I_i \Omega_i$ as the body fixed axes coincide with the center of mass of the body.

This set of non linear partial differential equations are linearized and integrated in time by means of the Newmark algorithm.

3. Time integration scheme

3.1. Splitting of the fluid mechanics equations

For the time integration of the Navier–Stokes equation a projection method was adopted [22–24]. These methods stand out from classical monolithic schemes as pressure and velocity are uncoupled. Thus four linear systems of equations are obtained, namely three for the momentum equation and one to solve the incompressibility constrain.

From Eq. (7) the time derivative may be approximated with forward differences as

$$\left. \frac{du_i}{dt} \right|_{x_i} \approx \frac{u_i^{n+1} - u_i^n}{\delta t}. \quad (18)$$

An auxiliary velocity u_i^* will be employed to split the momentum equation, such that:

$$u_i^* = u_i^n - \gamma \frac{\delta t}{\rho} \frac{\partial p^n}{\partial x_j} \delta_{ij} + (u_j^{n+\theta} - v_j) \frac{\partial u_i^{n+\theta}}{\partial x_j} + \frac{\mu \delta t}{\rho} \frac{\partial}{\partial x_j} \left(\frac{\partial u_i^{n+\theta}}{\partial x_j} \right) + \delta t f_i, \quad (19)$$

$$u_i^{n+1} = u_i^* - \frac{\delta t}{\rho} \frac{\partial}{\partial x_j} (p^{n+1} - \gamma p^n) \delta_{ij}, \quad (20)$$

where $\theta = 0$ implies a explicit forward Euler scheme, $\theta = 1$ is a implicit backward Euler scheme and $\theta = 0.5$ is the Crank–Nicholson second order scheme. In the present analysis $\theta = 1$ will be adopted.

From Eqs. (8) and (20) a Poisson equation of pressure is obtained:

$$\delta t \frac{\partial}{\partial x_i} \frac{1}{\rho} \frac{\partial}{\partial x_j} (p^{n+1} - \gamma p^n) \delta_{ij} = \frac{\partial}{\partial x_i} u_i^*. \quad (21)$$

The variable $\gamma = 1$ implies a second order projection scheme and $\gamma = 0$ a first order scheme.

4. Spatial discretization by the Finite Element Method

One of the key to solve a fluid mechanics problem using a Lagrangian formulation is to generate efficiently the shape functions to approximate the spatial unknown. In the Finite Element context, this means to generate permanently, at each time step, a new mesh.

In this work the interpolation function used by the Meshless Finite Element Method (MFEM) [4] has been implemented. This function uses the Voronoï diagram of the cloud of points to construct the interpolant. The Extended Delaunay Tessellation (EDT) [4] is applied to connect the neighboring particles. The EDT provides polyhedral elements that are sliver-free in 3-D, thus avoiding the instabilities of the Delaunay tessellation due to distorted tetrahedral. The EDT provides a way to generate meshes at each time step very efficiently in a computing time which is largely smaller than the computing time needed to solve the linearized system of equation.

The unknown functions are approximated using an equal order interpolation for all variables in the final configuration:

$$u_i = N_l(\chi, t) U_{il}, \quad (22)$$

$$p_i = N_l(\chi, t) P_{il}, \quad (23)$$

where N_l are the shape functions provided by the MFEM. The matrix version of Eqs. (19)–(21) is as follows:

$$Mu_i^* + \delta t K u_i^{n+\theta} + S(u_i^{n+\theta})(u_j^{n+\theta} - v_j^{n+\theta}) = Mu_i^n - \gamma \delta t G p^n + \delta t F, \quad (24)$$

$$\delta t L p^{n+1} = Bu_i^* + \gamma \delta t L p^n - \tilde{U}, \quad (25)$$

$$\bar{M} u_i^{n+1} = \bar{M} u_i^* - \delta t G (p^{n+1} - \gamma p^n), \quad (26)$$

where the matrices are defined by

$$M = \int_{\Omega} N_q N_l d\Omega, \quad L = \int_{\Omega} \left(\frac{\partial N_q}{\partial x_i} \frac{1}{\rho} \frac{\partial N_l}{\partial x_i} \right) d\Omega,$$

$$B = \int_{\Omega} \left(\frac{\partial N_q}{\partial x_i} N_l \right) d\Omega, \quad G = \int_{\Omega} \left(N_q \frac{1}{\rho} \frac{\partial N_l}{\partial x_i} \right) d\Omega,$$

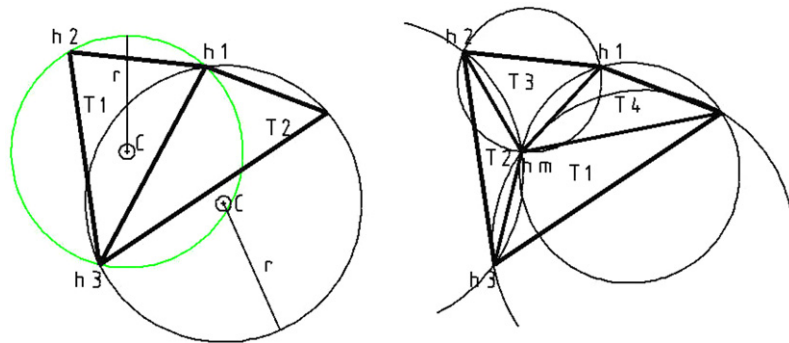


Fig. 1. Adding a node in the triangulation.

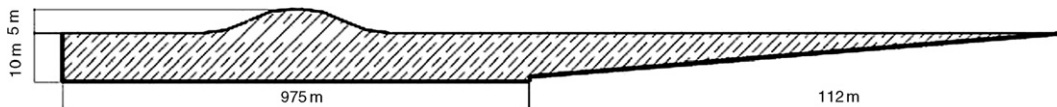


Fig. 2. Wave breaking on a 1/14 slope beach.

$$K = \int_{\Omega} \left(\frac{\partial N_q}{\partial x_i} \mu \frac{\partial N_l}{\partial x_i} \right) d\Omega, \quad S = \int_{\Omega} \left(N_q u_i^{n+\theta} \frac{\partial N_l}{\partial x_i} \right) d\Omega,$$

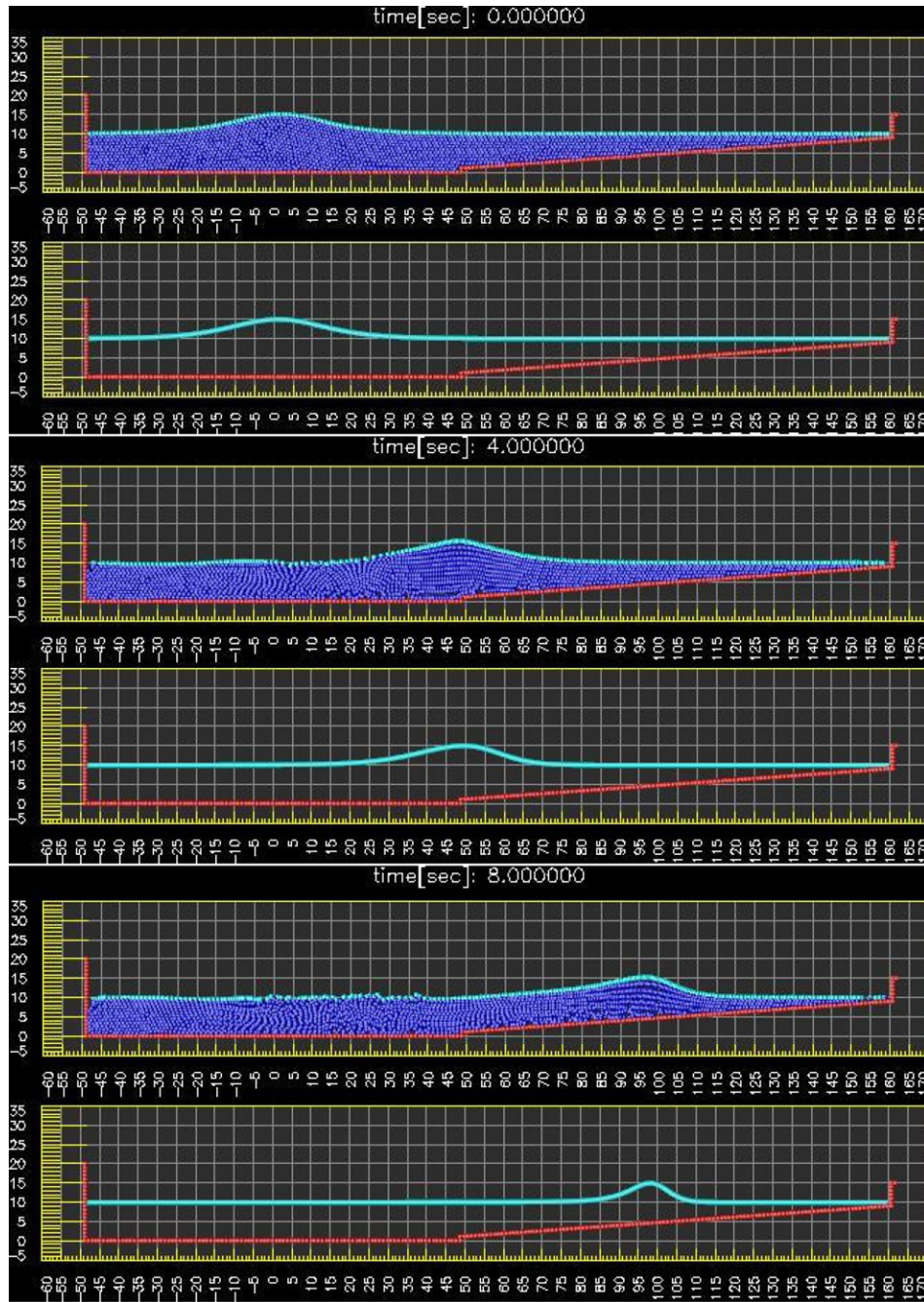
$$F = \int_{\Omega} N_q f_i d\Omega, \quad \tilde{U} = \int_{\Gamma} N_q u_i^{n+1} n_i d\Gamma.$$

As an equal order approximation is used for the velocity and pressure variables, the Finite Calculus (FIC) [27,26,28] method has been used to overcome the inf-sup condition.

The indices (q, l) denote the nodes of an element. The matrix \bar{M} in Eq. (26) is the lumped mass matrix M . Standard stabilization terms [25,26] have been added to Eq. (24) to stabilize the convection terms when $Re \gg 1$.

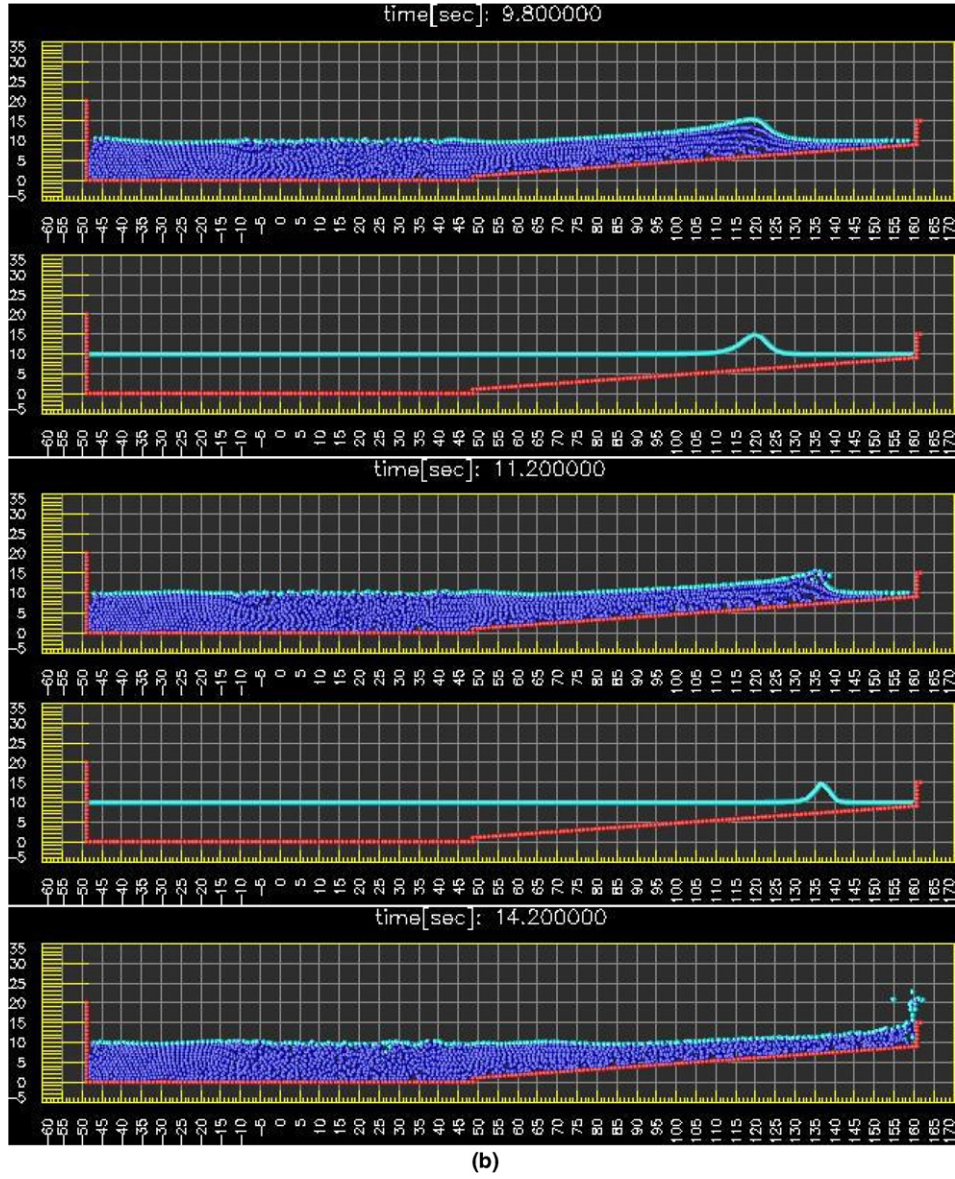
5. Coupling the fluid and solid equations

On the coupling boundary the fluid velocity and the solid velocity must satisfy:



(a)

Fig. 3. (a) Comparison between numerical solutions and analytical solution for the wave breaking problem. (b) Continuation from (a).



(b)

Fig. 3 (continued)

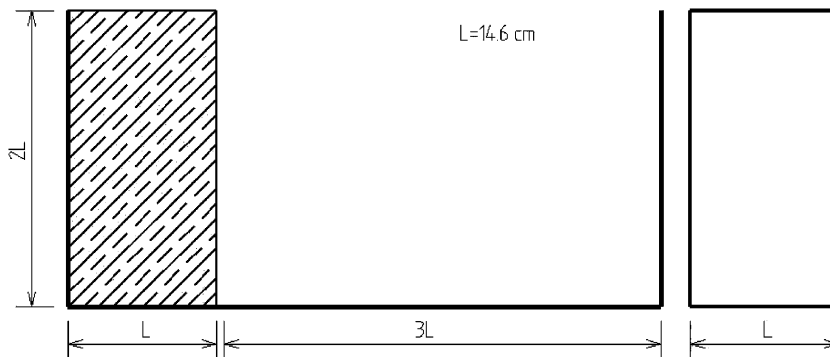


Fig. 4. 2-d sketch of the water collapse problem (side view and front view).

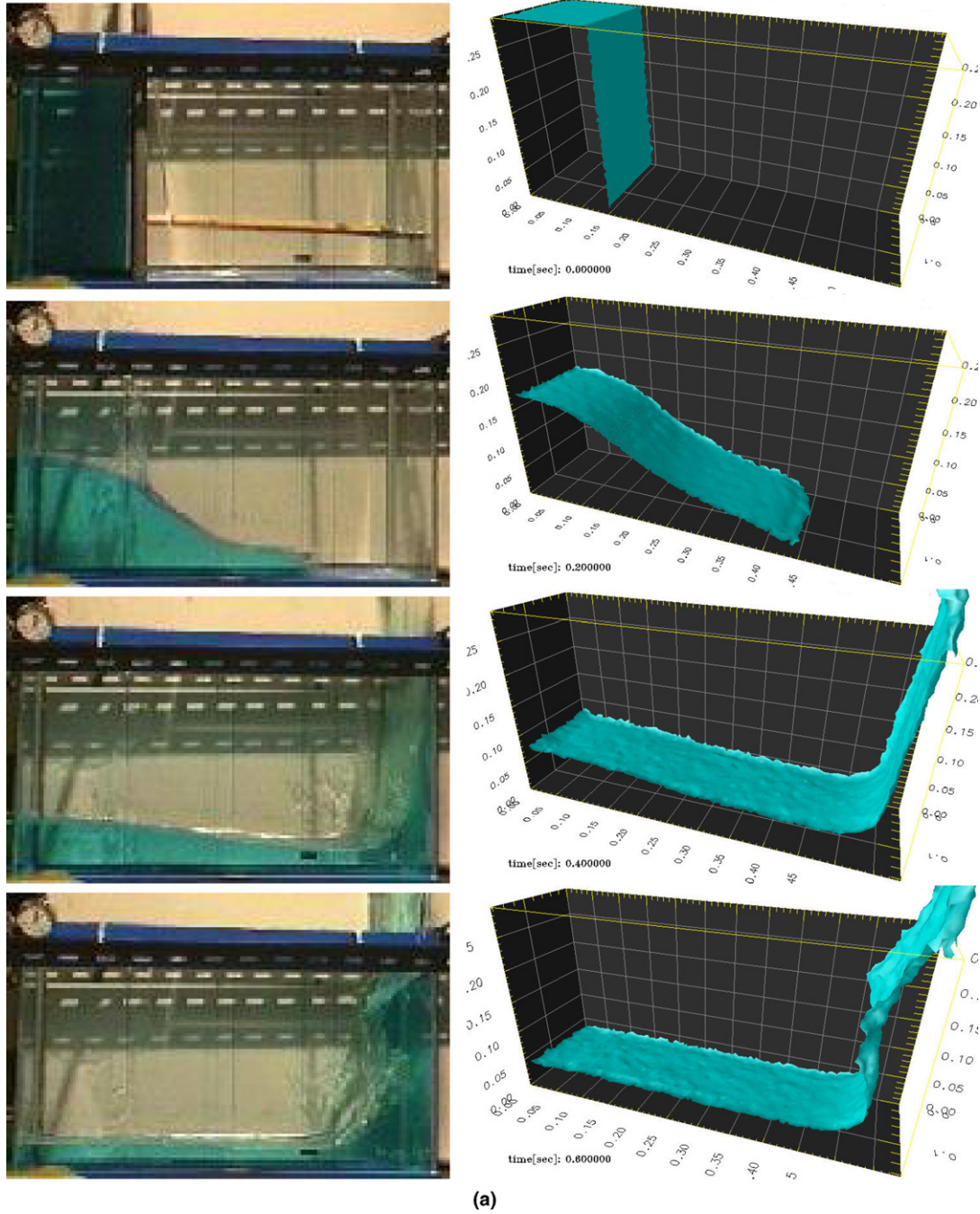


Fig. 5. (a) Comparison between the numerical solution and the experimental solution for six different time steps. Courtesy of Koshizuka and Oka [18]. (b) Continuation from (a).

$$u_f|_{\sigma_1} = u_s|_{\sigma_2}. \tag{27}$$

Thus, two subsystems need to be solved, namely the fluid system:

$$F(u_f^{n+1}, p^{n+1}, x_f^{n+1}, u_s^{n+1}, x_s^{n+1}) = 0, \tag{28}$$

and the solid subsystem:

$$S(u_s^{n+1}, x_s^{n+1}, u_f^{n+1}, p^{n+1}, x_f^{n+1}) = 0. \tag{29}$$

In the equations above only the variables to be solved at time step $n + 1$ are shown. To linearized both subsystems a Gauss–Seidel process has been chosen [29]. In this way, the iterative procedure means to solve first one of the subsystems and use the output to solve the next subsystem. The final expression for the computation is as follows:

$$(u_{f,k+1}^{n+1}, p_{k+1}^{n+1}, x_{f,k+1}^{n+1}) = F(u_{f,k}^{n+1}, p_k^{n+1}, x_{f,k}^{n+1}, u_s^{n+1}, x_s^{n+1}), \tag{30}$$

$$(u_{s,k+1}^{n+1}, x_{s,k+1}^{n+1}) = S(u_{s,k}^{n+1}, x_{s,k}^{n+1}, u_f^{n+1}, p^{n+1}, x_f^{n+1}), \tag{31}$$

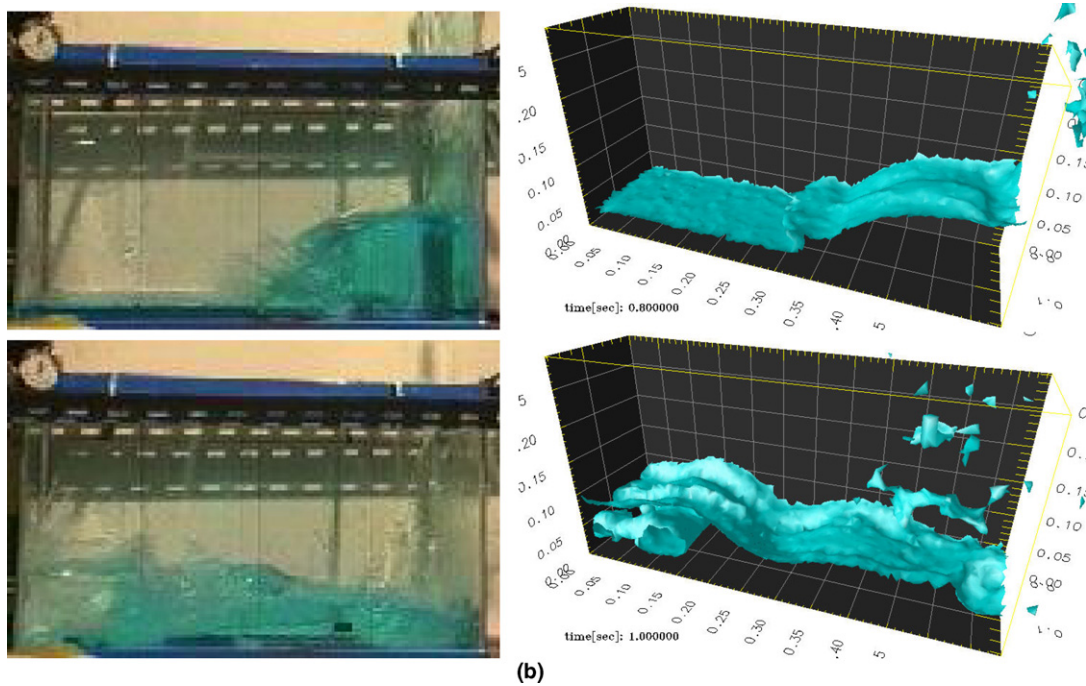


Fig. 5 (continued)

where the subscript k is the iteration counter. Convergence occurs when the difference between velocities of successive iteration steps is less than the acceptable error. A consistent artificial compressibility [30] is added to the elements of the boundaries to allow the compressible movement of the solid. When convergence is reached the incompressibility constrain is recovered.

6. Mesh update

6.1. Mesh movement

The most appealing feature of the ALE formulation is the fact that for a highly deformable domain the distortion of the Finite Element mesh may be minimized. For this reason the election of a robust algorithm to move the mesh is essential. Among all the different options found in the literature [31–37], the Laplacian operator [38,39] has been chosen.

The basic idea consist in solving a Laplace equation for each direction of the mesh velocity, this is

$$\frac{\partial^2 v_i}{\partial x_j^2} = 0 \quad \text{in } \Omega, \quad (32)$$

$$v_i = \bar{u}_i \quad \text{in } \Gamma, \quad (33)$$

where \bar{u}_i is the prescribed Lagrangian velocity on the boundary or interfaces. Thus, three linear systems of

equations need to be solved at each time step. Enough accuracy for the mesh movement is obtained after three iterations of the conjugate gradient solver.

It is worth mentioning that although the Laplacian provides an acceptable velocity field for the mesh, the final decision whether a node is treated as a material particle or not remains as a user option. This is a key feature of the formulation that allows to maintain the key particle features of the method.

6.2. Mesh restoration

For problems where large boundary deformation occurs a constant restoration of the mesh is needed in the deformed region. This is the case for breaking waves or problems where large displacements of solids is observed. The mesh is thus rebuilt by means of the EDT. This algorithm will reconnect the nodes that are already part of the domain, thus no projection of a new mesh over an old mesh needs to be done.

6.3. Adding and removing nodes

In areas where large gradients are observed the mesh should be refined to control the error over the whole domain or to keep an homogeneous distance between nodes in order to preserve a large admissible time step. To do this a point is added in the EDT as depicted in Fig. 1. In the figure a node is added in the center of the Delaunay circle of the element when:

$$r > \beta_1 \bar{h} \quad \text{and} \quad c \subset T, \quad (34)$$

where r is the radii of the circle, β_1 is an arbitrary parameter, typically $\beta = 0.6$, c is the center of the circle, T is the Delaunay triangle and \bar{h} is the average size of the element. The Delaunay tessellation is locally rebuilt every time a new node is added. The fields are interpolated on the new nodes directly by using the finite element shape functions.

A point is not inserted in the tessellation if it lies closer than $\beta_2 \bar{h}$ from the nearest node, with $\beta = 0.5$.

7. Validation and examples of applications

7.1. Wave breaking

The solution that is presented in this section describes the movement of a wave along a channel whose slope changes abruptly from zero to 1/14. The wave ends up breaking due to shoaling. The geometry of the problem with the original scale is shown in Fig. 2. This test problem was first introduced in [40] as validation of a Lagrangian formulation. It showed to be a good example that enhances the advantages of representing free surfaces when the equations are viewed from a Lagrangian point of reference.

The initial conditions are taken from the potential theory solution of a solitary wave of finite amplitude propagating without change of shape [41]. The quasi-analytical solution that will be used for comparison is called Laitone’s solution. The different parameters that describe the state of the solution at each time step are given by

$$u = \sqrt{gd} \frac{H}{d} \operatorname{sech}^2 \left[\sqrt{\frac{3H}{4d^3}} (x - ct) \right], \quad (35)$$

$$v = \sqrt{3gd} \left(\frac{H}{d} \right)^{3/2} \frac{y}{d} \operatorname{sech}^2 \left[\sqrt{\frac{3H}{4d^3}} (x - ct) \right] \tanh \left[\sqrt{\frac{3H}{4d^3}} (x - ct) \right], \quad (36)$$

$$\eta = d + H \operatorname{sech}^2 \left[\sqrt{\frac{3H}{4d^3}} (x - ct) \right], \quad (37)$$

$$p = \rho g (\eta - y), \quad (38)$$

$$c = \sqrt{gd \left(1 + \frac{H}{d} \right)}. \quad (39)$$

In the equations written above, H is the height of the wave with respect to the unperturbed surface, L is the length of the flat part of the channel, d is the depth from the bottom to the unperturbed surface, g is the acceleration of gravity, ρ is the density of water, η is the elevation of the free surface.

Laitone’s solution holds for infinitely long channels. However, in [42] a simplification is done in order to be able to compare the numerical results to the analytical solution. Indeed, it has been observed that if the domain of simulation is truncated at a distance $L/2$ of the wave crest where the fluid is essentially still, both numerical and analytical solution are comparable. If we assume that the fluid is still when the surface elevation satisfies $\eta = 0.01H$, the result for L is

$$\frac{L}{d} = 6.9 \left(\frac{d}{H} \right)^{1/2}. \quad (40)$$

As shown in Fig. 2, $L = 97.5$ m will be the region with flat slope, $d = 10$ m is the still water depth and $H = 5$ m is the wave height. The slope of the shoaling bottom is set to 1/14.

The results for different time steps are displayed in Fig. 3a and b. The figures are displayed in pairs, showing on top the numerical solution and underneath the analytical solution.

The numerical solution is in complete agreement with the analytical solution in the range of values where the analytical solution is valid.

8. Water column collapse

The problem was solved by Koshizuka and Oka as validation of the Moving Particle Semi-Implicit method presented in [18]. In the paper they included not only the numerical results but also a laboratory experiment that became a classical example to test the validation of the Lagrangian formulation in fluid flows.

The experiment consists of a water column initially at rest supported on the left by a removable board. When the board is slid up the collapse starts. A 2-d sketch of the problem is shown in Fig. 4 where we can appreciate the dimensions of the problem. Viscosity has been neglected in this example. The total number of nodal points used to solve the 3-d simulation was 14838.

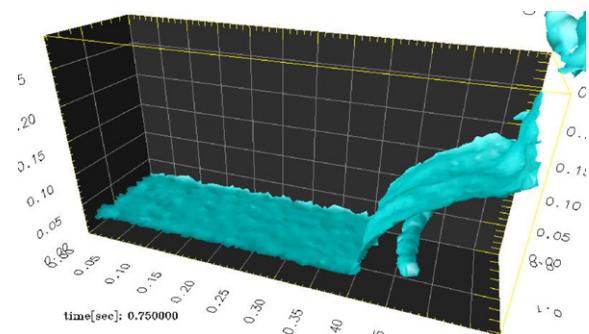


Fig. 6. Internal surface captured by the alpha-shape algorithm.

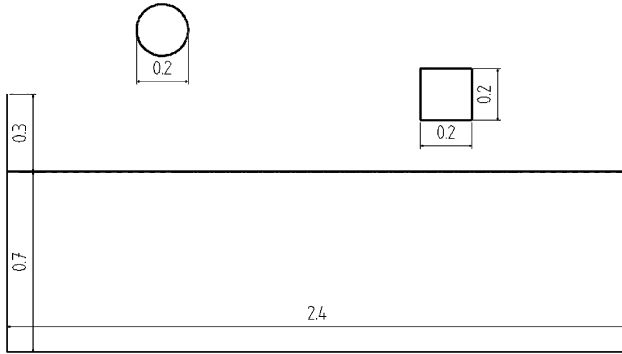


Fig. 7. Schematic representation of the two rigid bodies falling in the tank.

The comparison between the experiment and the numerical solution is shown in Fig. 5a and b. Both the free surface as well as the temporal development of the problem are in good agreement with the experimental data.

Fig. 6 shows the development of the fluid free surface at $t = 0.75$ s. At this stage an internal surface forms a bubble inside the fluid. The alpha-shape technique can capture this surface and appropriate boundary conditions are applied.

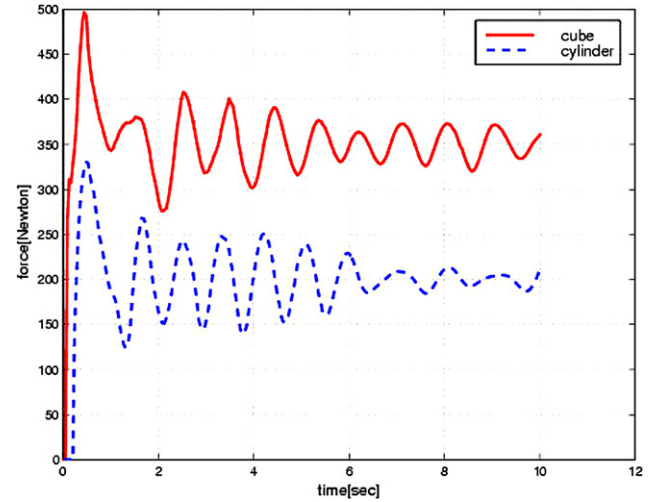


Fig. 8. The evolution of the total force acting on the two rigid bodies falling in the tank.

8.1. Rigid bodies falling in water

The last example is a fluid–object interaction problem, where the structure behaves as a rigid body. Fig. 7 shows an schematic representation of the problem. Two bodies with different shapes fall in the water

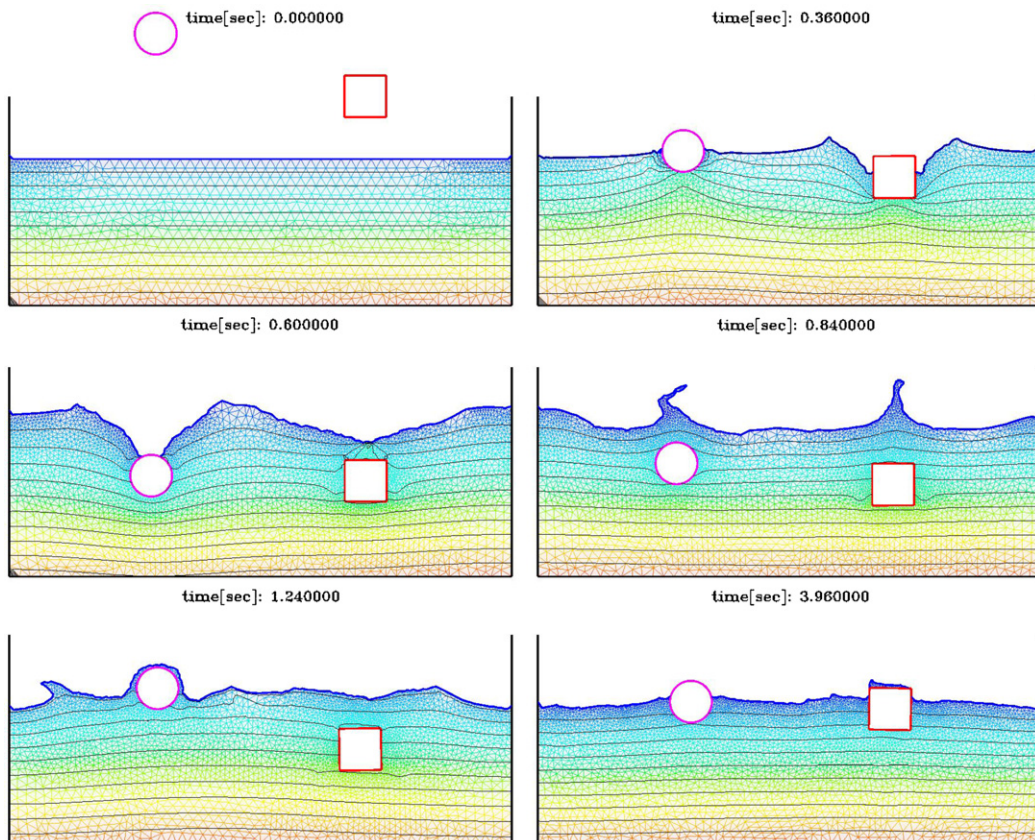


Fig. 9. Motion of the two falling bodies at different times. Also show the mesh and the pressure contour lines.

container from different heights. The cylinder has a mass $m = 20$ kg and the cube has a mass of $m = 35$ kg. For this problem the free surface as well as the rigid bodies move in a fully Lagrangian way, while the rest of the fluid is ALE. Fig. 9, shows states at different time steps. In Fig. 8 the forces of the fluid over the structures are shown.

9. Conclusions

The Particle Finite Element Method (PFEM) combining Lagrangian and ALE formulations has shown to be an excellent option in problems involving fluid–object interaction, large deformations, free surfaces and fluid segmentation (Fig. 9).

The potential of the method relies on the fast generation of the EDT partition and the accuracy of the Finite Element Method. The use of the alpha-shape technique to compute the boundaries greatly simplifies the contacts between fluid and structures. Even in the case where constant re-meshing is needed the combined Lagrangian–ALE formulation still has advantages over the full Lagrangian formulation as no distortion due to vortexes is observed in the mesh. As an implicit algorithm is chosen the time step in the ALE case is greater than for the full Lagrangian case as it depends upon the mesh velocity instead of the material velocities. This provides time steps over 50% larger than in the Lagrangian case.

References

- [1] Aubry R, Idelsohn SR, Oñate E. Particle finite element method in fluid mechanics including thermal convection–diffusion. *Comput Struct* 2004;83:1459–75.
- [2] Idelsohn S, Oñate E, Del Pin F. A Lagrangian meshless finite element method applied to fluid–structure interaction problems. *Comput Struct* 2003;81:655–71.
- [3] Idelsohn SR, Oñate E, Del Pin F. The particle finite element method a powerful tool to solve incompressible flows with free-surfaces and breaking waves. *Int J Num Methods Eng* 2004;61:964–84.
- [4] Idelsohn SR, Oñate E, Calvo N, Del Pin F. The meshless finite element method. *Int J Num Methods Eng* 2003;58(4).
- [5] Harlow FH, Welch JE. *Phys Fluids* 1965;8:2182.
- [6] Welch JE, Harlow FH, Shannon JP, Daly BJ. The MAC method: a computing technique for solving viscous, incompressible, transient fluid flow problems involving free surfaces. Los Alamos Scientific Laboratory Report, LA-3425, 1966.
- [7] Hirt CW, Nichols BD, Romero NC. Sola—a numerical solution algorithm for transient fluid flows, Los Alamos Scientific Laboratory Report, LA-5852, 1975.
- [8] Nichols BD, Hirt CW. *J Comput Phys* 1971;8:434.
- [9] Nichols BD, Hirt CW. In: First international conference of numeric ship hydrodynamics, Gaithersburg, October 1975.
- [10] Hirt CW, Nichols BD. Volume of fluid (vof) method for the dynamics of free boundaries. *J Comput Phys* 1979;39(November): 201–25.
- [11] Tezduyar T, Behr M, Liou J. A new strategy for finite element formulations involving moving boundaries and interfaces the deforming-spatial-domain/space–time procedure: I. The concept and the preliminary numerical tests. *Comput Methods Appl Mech Eng* 1992;94:339–51.
- [12] Tezduyar T, Behr M, Mittal S, Liou J. A new strategy for finite element formulations involving moving boundaries and interfaces the deforming-spatial-domain/space–time procedure: II. Computation of free-surface flows two-liquid flows and flows with drifting cylinders. *Comput Methods Appl Mech Eng* 1992;94:353–71.
- [13] Tezduyar T, Aliabadi S, Behr M. Enhanced-discretization interface-capturing technique (edict) for computation of unsteady flows with interfaces. *Comput Methods Appl Mech Eng* 1998;155:235–48.
- [14] Lucy LB. A numerical approach to the testing of the fission hypothesis. *Astron J* 1977;82:1013–24.
- [15] Gingold RA, Monaghan JJ. Kernel estimates as a basis for general particle methods in hydrodynamics. *J Comput Phys* 1981;46(March):429–53.
- [16] Monaghan JJ. Why particle methods work. *J Sci Statist Comput* 1982;4:422.
- [17] Liu WK, Jun S, Zhang YF. Reproducing kernel particle methods. *Int J Numer Methods Fluids* 1995;20:1081–106.
- [18] Koshizuka S, Oka Y. Moving particle semi-implicit method for fragmentation of incompressible fluid. *Nucl Eng Sci* 1996;123:421–34.
- [19] Ramaswamy B, Kawahara M. Lagrangian finite element analysis applied to viscous free surface fluid flow. *Int J Numer Methods Fluids* 1987;7:953–84.
- [20] Edelsbrunner H, Kirkpatrick DG, Seidel R. On the shape of a set of points in the plane. *IEEE Trans Inform Theory* 1983;IT–29(4):551–9.
- [21] Hirt CW, Amsden AA, Cook JL. An arbitrary Lagrangian–Eulerian computing method for all flow speeds. *J Comput Phys* 1974;14:227–53.
- [22] Chorin AJ. Numerical solution of the Navier–Stokes equations. *Math Comput* 1968;22:745–62.
- [23] Chorin AJ. Numerical study of slightly viscous flow. *J Fluid Mech* 1973;57:785.
- [24] Temam R. Sur l’approximation de la solution des équations de Navier–Stokes par la méthode des pas fractionnaires II. *Arch Rat Mech Anal* 1969;33:377–85.
- [25] Codina R, Soto O. Approximation of the incompressible Navier–Stokes equations using orthogonal-subscale stabilization and pressure segregation on anisotropic finite element meshes. *Comput Methods Appl Mech Eng* 2004;193:1403–19.
- [26] Oñate E. A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation. *Comput Methods Appl Mech Eng* 2000;182(1–2):355–70.
- [27] Oñate E. Derivation of stabilized equations for advective diffusive transport and fluid flow problems. *Comput Methods Appl Mech Eng* 1998;151(1–2):233–67.
- [28] Oñate E. Possibilities of finite calculus in computational mechanics. *Int J Num Methods Eng* 2004;60(1):255–81.
- [29] Codina R. Block-iterative algorithms for nonlinear coupled problems. In: Bugeda G, Papadrakakis M, editors. *Advanced computational methods in structural mechanics*. CIMNE; 1996.
- [30] Raback P, Ruokolainen J, Lyly M, Järvinen E. Fluid–structure interaction boundary conditions by artificial compressibility. In: *ECCOMAS computational fluid dynamics conference*, Swansea, Wales, September 2001.
- [31] Batina M. Unsteady Euler airfoil solutions using unstructured dynamic meshes. *AIAA J* 1990;28(8):1381–8.
- [32] Behr M, Tezduyar T. Finite element solution strategies for large-scale flow simulations. *Comput Methods Appl Mech Eng* 1994;113:3–24.

- [33] Tezduyar T, Behr M, Mittal S, Johnson AA. Computation of unsteady incompressible flows with the stabilized finite element methods—space–time formulations, iterative strategies and massively parallel implementations. *New Methods Transient Anal* 1992;246:7–24.
- [34] Johnson A, Tezduyar T. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Comput Methods Appl Mech Eng* 1994;119:73–9.
- [35] Guler I, Behr M, Tezduyar T. Parallel finite element computation of free-surface flows. *Comput Mech* 1999;23:117–23.
- [36] Behr M, Abraham F. Free-surface flow simulations in the presence of inclined walls. *Comput Methods Appl Mech Eng* 2002;191:5467–83.
- [37] Tezduyar T, Aliabadi S, Behr M, Johnson A, Kalro V, Litke M. Flow simulation and high performance computing. *Comput Mech* 1996;18:397–412.
- [38] Löhner R, Yang C. Improved ALE mesh velocities for moving bodies. *Commun Numer Methods Eng* 1996;12:599–608.
- [39] Masud A, Hughes T. A space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations for moving domain problems. *Comput Methods Appl Mech Eng* 1997;146:91–126.
- [40] Radovitzky R, Ortiz M. Lagrangian finite element analysis of Newtonian fluid flow. *Int J Numer Methods Eng* 1998;43:607–19.
- [41] Grilli ST, Subramanya R. Numerical modeling of wave breaking introduced by fixed or moving boundaries. *Comput Mech* 1996;17:374–91.
- [42] Ramaswamy B, Kawahara M. Numerical simulation of unsteady viscous free surface flow. *J Comput Phys* 1990;90:396–430.