
Advances in the particle finite element method for fluid-structure interaction problems

Eugenio Oñate, Sergio R. Idelsohn, Miguel A. Celigueta and Riccardo Rossi

International Center for Numerical Methods in Engineering (CIMNE)
Universidad Politecnica de Cataluna
Campus Norte UPC, 08034 Barcelona, Spain
onate@cimne.upc.edu

We present a general formulation for analysis of fluid-structure interaction problems using the particle finite element method (PFEM). The key feature of the PFEM is the use of a Lagrangian description to model the motion of nodes (particles) in both the fluid and the structure domains. Nodes are thus viewed as particles which can freely move and even separate from the main analysis domain representing, for instance, the effect of water drops. A mesh connects the nodes defining the discretized domain where the governing equations, expressed in an integral form, are solved as in the standard FEM. The necessary stabilization for dealing with the incompressibility of the fluid is introduced via the finite calculus (FIC) method. A fractional step scheme for the transient coupled fluid-structure solution is described. Examples of application of the PFEM to solve a number of fluid-structure interaction problems involving large motions of the free surface and splashing of waves are presented.

1 Introduction

There is an increasing interest in the development of robust and efficient numerical methods for analysis of engineering problems involving the interaction of fluids and structures accounting for large motions of the fluid free surface and the existence of fully or partially submerged bodies. Examples of this kind are common in ship hydrodynamics, off-shore structures, spill-ways in dams, free surface channel flows, liquid containers, stirring reactors, mould filling processes, etc.

The movement of solids in fluids is usually analyzed with the finite element method (FEM) [Zienkiewicz *et al.* (2006)] using the so called arbitrary Lagrangian-Eulerian (ALE) formulation [Donea and Huerta (2003)]. In the ALE approach the movement of the fluid particles is decoupled from that of the mesh nodes. Hence the relative velocity between mesh nodes and particles is used as the convective velocity in the momentum equations.

Typical difficulties of FSI analysis using the FEM with both the Eulerian and ALE formulation include the treatment of the convective terms and the incompress-

ibility constraint in the fluid equations, the modelling and tracking of the free surface in the fluid, the transfer of information between the fluid and solid domains via the contact interfaces, the modelling of wave splashing, the possibility to deal with large rigid body motions of the structure within the fluid domain, the efficient updating of the finite element meshes for both the structure and the fluid, etc.

Most of these problems disappear if a *Lagrangian description* is used to formulate the governing equations of both the solid and the fluid domain. In the Lagrangian formulation the motion of the individual particles are followed and, consequently, nodes in a finite element mesh can be viewed as moving “particles”. Hence, the motion of the mesh discretizing the total domain (including both the fluid and solid parts) is followed during the transient solution.

In this paper we present an overview of a particular class of Lagrangian formulation developed by the authors to solve problems involving the interaction between fluids and solids in a unified manner. The method, called the *particle finite element method* (PFEM), treats the mesh nodes in the fluid and solid domains as particles which can freely move and even separate from the main fluid domain representing, for instance, the effect of water drops. A finite element mesh connects the nodes defining the discretized domain where the governing equations are solved in the standard FEM fashion. The PFEM is the natural evolution of recent work of the authors for the solution of FSI problems using Lagrangian finite element and meshless methods [Aubry *et al.* (2005); Idelsohn *et al.* (2003a; 2003b; 2004); Oñate *et al.* (2003; 2004a,b)].

An obvious advantage of the Lagrangian formulation is that the convective terms disappear from the fluid equations. The difficulty is however transferred to the problem of adequately (and efficiently) moving the mesh nodes. Indeed for large mesh motions remeshing may be a frequent necessity along the time solution. We use an innovative mesh regeneration procedure blending elements of different shapes using an extended Delaunay tessellation [Idelsohn *et al.* (2003a; 2003c)]. Furthermore, this special polyhedral finite element needs special shape functions. In this paper, meshless finite element (MFEM) shape functions have been used [Idelsohn *et al.* (2003a)].

The need to properly treat the incompressibility condition in the fluid still remains in the Lagrangian formulation. The use of standard finite element interpolations may lead to a volumetric locking defect unless some precautions are taken [Donea and Huerta (2003), Zienkiewicz *et al.* (2006)]. In our work the stabilization via a finite calculus (FIC) procedure has been chosen [Oñate (2000)]. Recent applications of the FIC method for incompressible flow analysis using linear triangles and tetrahedra are reported in [García and Oñate (2003); Oñate (2004); Oñate *et al.* (2000; 2004a,b); Oñate and García (2001); Oñate and Idelsohn (1998)].

The layout of the paper is the following. In the next section the basic ideas of the PFEM are outlined. Next the basic equation for an incompressible flow using a Lagrangian description and the FIC formulation are presented. Then a fractional step scheme for the transient solution via standard finite element procedures is described. Details of the treatment of the coupled FSI problem are given. The procedures for mesh generation and for identification of the free surface nodes are briefly outlined.

Finally, the efficiency of the *particle finite element method* (PFEM) is shown in its application to a number of FSI problems involving large flow motions, surface waves, moving bodies. etc.

2 The basis of the Particle Finite Element Method

Let us consider a domain containing both fluid and solid subdomains. The moving fluid particles interact with the solid boundaries thereby inducing the deformation of the solid which in turn affects the flow motion and, therefore, the problem is fully coupled.

In the PFEM approach presented here, both the fluid and the solid domains are modelled using an *updated Lagrangian formulation*. The finite element method (FEM) is used to solve the continuum equations in both domains. Hence a mesh discretizing these domains must be generated in order to solve the governing equations for both the fluid and solid problems in the standard FEM fashion. We note once more that the nodes discretizing the fluid and solid domains can be viewed as material particles which motion is tracked during the transient solution.

The Lagrangian formulation allows us to track the motion of each single fluid particle (a node). This is useful to model the separation of water particles from the main fluid domain and to follow their subsequent motion as individual particles with an initial velocity and subject to gravity forces.

The quality of the numerical solution depends on the discretization chosen as in the standard FEM. Adaptive mesh refinement techniques can be used to improve the solution in zones where large motions of the fluid or the structure occur.

A typical solution with the PFEM involves the following steps.

1. Discretize the fluid and solid domains with a finite element mesh. In our work we use an innovative mesh generation scheme based on the extended Delaunay tessellation (Section 7) [Idelsohn *et al.* (2003a; 2003b; 2004)].
2. Identify the external boundaries for both the fluid and solid domains. This is an essential step as some boundaries (such as the free surface in fluids) may be severely distorted during the solution process including separation and re-entering of nodes. The Alpha Shape method [Edelsbrunner and Mucke (1999)] is used for the boundary definition (see Section 8).
3. Solve the coupled Lagrangian equations of motion for the fluid and the solid domains. Compute the relevant state variables in both domains at each time step: velocities, pressure and viscous stresses in the fluid and displacements, stresses and strains in the solid.
4. Move the mesh nodes to a new position in terms of the time increment size. This step is typically a consequence of the solution process of step 3.
5. Generate a new mesh if needed. The mesh regeneration process can take place after a prescribed number of time steps or when the actual mesh has suffered severe distortions due to the Lagrangian motion.
6. Go back to step 2 and repeat the solution process for the next time step.

Figure 1 shows a typical example of a PFEM solution in 2D. The pictures correspond to the analysis of the problem of breakage of a water column [Oñate *et al.* (2004); Idelsohn *et al.* (2004)]. Figure 1a shows the initial grid of four node rectangles discretizing the fluid domain and the solid walls. Boundary nodes identified with the Alpha-Shape method have been marked with a circle. Figures 1b and 1c show the mesh for the solution at two later times.

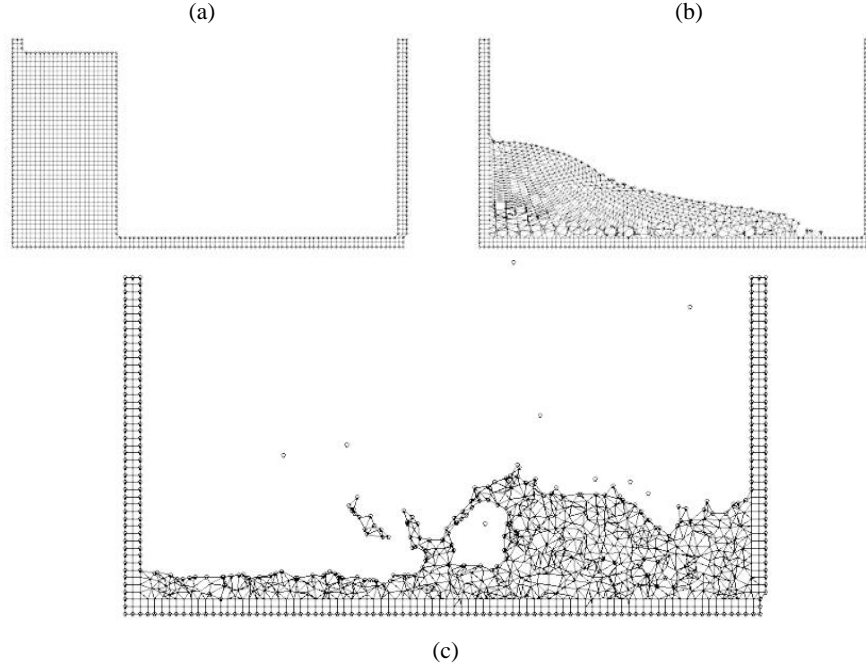


Fig. 1. Breakage of a water column. (a) Discretization of the fluid domain and the solid walls. Boundary nodes are marked with circles. (b) and (c) Mesh in the fluid and solid domains at two different times.

3 Lagrangian Equations for an Incompressible Fluid. FIC Formulation

The standard infinitesimal equations for a viscous incompressible fluid can be written in a Lagrangian frame as [Oñate (1998); Zienkiewicz *et al.* (2006)].

Momentum

$$r_{m_i} = 0 \quad \text{in } \Omega \quad (1)$$

Mass balance

$$r_d = 0 \quad \text{in } \Omega \quad (2)$$

where

$$r_{m_i} = \rho \frac{\partial v_i}{\partial t} + \frac{\partial \sigma_{ij}}{\partial x_j} - b_i \quad , \quad \sigma_{ji} = \sigma_{ij} \quad (3)$$

$$r_d = \frac{\partial v_i}{\partial x_i} \quad i, j = 1, n_d \quad (4)$$

Above n_d is the number of space dimensions, v_i is the velocity along the i th global axis ($v_i = \frac{\partial u_i}{\partial t}$, where u_i is the i th displacement), ρ is the (constant) density of the fluid, b_i are the body forces, σ_{ij} are the total stresses given by $\sigma_{ij} = s_{ij} - \delta_{ij}p$, p is the absolute pressure (defined positive in compression) and s_{ij} are the viscous deviatoric stresses related to the viscosity μ by the standard expression

$$s_{ij} = 2\mu \left(\dot{\varepsilon}_{ij} - \delta_{ij} \frac{1}{3} \frac{\partial v_k}{\partial x_k} \right) \quad (5)$$

where δ_{ij} is the Kronecker delta and the strain rates $\dot{\varepsilon}_{ij}$ are

$$\dot{\varepsilon}_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \quad (6)$$

In the above all variables are defined at the current time t (current configuration).

In our work we will solve a *modified set of governing* equations derived using a finite calculus (FIC) formulation. The FIC governing equations are [Oñate (1998; 2000; 2004); Oñate *et al.* (2001)].

Momentum

$$r_{m_i} - \frac{1}{2} h_j \frac{\partial r_{m_i}}{\partial x_j} = 0 \quad \text{in } \Omega \quad (7)$$

Mass balance

$$r_d - \frac{1}{2} h_j \frac{\partial r_d}{\partial x_j} = 0 \quad \text{in } \Omega \quad (8)$$

The problem definition is completed with the following boundary conditions

$$n_j \sigma_{ij} - t_i + \frac{1}{2} h_j n_j r_{m_i} = 0 \quad \text{on } \Gamma_t \quad (9)$$

$$v_j - v_j^p = 0 \quad \text{on } \Gamma_v \quad (10)$$

and the initial condition is $v_j = v_j^0$ for $t = t_0$. The standard summation convention for repeated indexes is assumed unless otherwise specified.

In Eqs.(7) and (8) t_i and v_j^p are surface tractions and prescribed velocities on the boundaries Γ_t and Γ_v , respectively, n_j are the components of the unit normal vector to the boundary.

The h'_i s in above equations are *characteristic lengths* of the domain where balance of momentum and mass is enforced. In Eq.(9) these lengths define the domain

where equilibrium of boundary tractions is established. Details of the derivation of Eqs.(7)–(10) can be found in [Oñate (1998; 2000; 2004)].

Eqs.(7)–(10) are the starting point for deriving stabilized finite element methods to solve the incompressible Navier-Stokes equations in a Lagrangian frame of reference using equal order interpolation for the velocity and pressure variables [Idelsohn *et al.* (2002; 2003a; 2003b; 2004); Oñate *et al.* (2003); Aubry *et al.* (2005)]. Application of the FIC formulation to finite element and meshless analysis of fluid flow problems can be found in [García and Oñate (2003); Oñate (2000; 2004); Oñate *et al.* (2000; 2004a); Oñate and García (2001); Oñate and Idelsohn (1988)].

3.1 Transformation of the Mass Balance Equation. Integral Governing Equations

The underlined term in Eq.(8) can be expressed in terms of the momentum equations. The new expression for the mass balance equation is [Oñate (2000); Oñate *et al.* (2004b)]

$$r_d - \sum_{i=1}^{n_d} \tau_i \frac{\partial r_{m_i}}{\partial x_i} = 0 \quad (11)$$

with

$$\tau_i = \frac{3h_i^2}{8\mu} \quad (12)$$

At this stage it is no longer necessary to retain the stabilization terms in the momentum equations. These terms are critical in Eulerian formulations to stabilize the numerical solution for high values of the convective terms. In the Lagrangian formulation the convective terms disappear from the momentum equations and the FIC terms in these equations are just useful to derive the form of the mass balance equation given by Eq.(11) and can be disregarded there onwards. Consistently, the stabilization terms are also neglected in the Neuman boundary conditions (eqs.(9)).

The weighted residual expression of the final form of the momentum and mass balance equations can be written as

$$\int_{\Omega} \delta v_i r_{m_i} d\Omega + \int_{\Gamma_t} \delta v_i (n_j \sigma_{ij} - t_i) d\Gamma = 0 \quad (13)$$

$$\int_{\Omega} q \left[r_d - \sum_{i=1}^{n_d} \tau_i \frac{\partial r_{m_i}}{\partial x_i} \right] d\Omega = 0 \quad (14)$$

where δv_i and q are arbitrary weighting functions equivalent to virtual velocity and virtual pressure fields.

The r_{m_i} term in Eq.(14) and the deviatoric stresses and the pressure terms within r_{m_i} in Eq.(13) are integrated by parts to give

$$\int_{\Omega} \left[\delta v_i \rho \frac{\partial v_i}{\partial t} + \delta \varepsilon_{ij} (s_{ij} - \delta_{ij} p) \right] d\Omega - \int_{\Omega} \delta v_i b_i d\Omega - \int_{\Gamma_t} \delta v_i t_i d\Gamma = 0 \quad (15)$$

$$\int_{\Omega} q \frac{\partial v_i}{\partial x_i} d\Omega + \int_{\Omega} \left[\sum_{i=1}^{n_d} \tau_i \frac{\partial q}{\partial x_i} r_{m_i} \right] d\Omega = 0 \quad (16)$$

In Eq.(15) $\delta \dot{\varepsilon}_{ij}$ are virtual strain rates. Note that the boundary term resulting from the integration by parts of r_{m_i} in Eq.(16) has been neglected as the influence of this term in the numerical solution has been found to be negligible.

3.2 Pressure Gradient Projection

The computation of the residual terms in Eq.(16) can be simplified if we introduce the pressure gradient projections π_i , defined as

$$\pi_i = r_{m_i} - \frac{\partial p}{\partial x_i} \quad (17)$$

We express now r_{m_i} in Eq.(17) in terms of the π_i which then become additional variables. The system of integral equations is therefore augmented in the necessary number of equations by imposing that the residual r_{m_i} vanishes within the analysis domain (in an average sense). This gives the final system of governing equation as:

$$\int_{\Omega} \left[\delta v_i \rho \frac{\partial v_i}{\partial t} + \delta \dot{\varepsilon}_{ij} (s_{ij} - \delta_{ij} p) \right] d\Omega - \int_{\Omega} \delta v_i b_i d\Omega - \int_{\Gamma_t} \delta v_i t_i d\Gamma = 0 \quad (18)$$

$$\int_{\Omega} q \frac{\partial v_i}{\partial x_i} d\Omega + \int_{\Omega} \sum_{i=1}^{n_d} \tau_i \frac{\partial q}{\partial x_i} \left(\frac{\partial p}{\partial x_i} + \pi_i \right) d\Omega = 0 \quad (19)$$

$$\int_{\Omega} \delta \pi_i \tau_i \left(\frac{\partial p}{\partial x_i} + \pi_i \right) d\Omega = 0 \quad \text{no sum in } i \quad (20)$$

with $i, j, k = 1, n_d$. In Eqs.(20) $\delta \pi_i$ are appropriate weighting functions and the τ_i weights are introduced for symmetry reasons.

4 Finite Element Discretization

We choose equal order C^0 continuous interpolations of the velocities, the pressure and the pressure gradient projections π_i over each element with n nodes. The interpolations are written as

$$v_i = \sum_{j=1}^n N_j \bar{v}_i^j, \quad p = \sum_{j=1}^n N_j \bar{p}^j, \quad \pi_i = \sum_{j=1}^n N_j \bar{\pi}_i^j \quad (21)$$

where $(\bar{\cdot})^j$ denotes nodal variables and N_j are the shape functions [Zienkiewicz *et al.* (2006)]. More details of the mesh discretization process and the choice of shape functions are given in Section 7.

Substituting the approximations (21) into Eqs.(19–20) and choosing a Galerkin form with $\delta v_i = q = \delta \pi_i = N_i$ leads to the following system of discretized equations

$$\mathbf{M}\dot{\bar{\mathbf{v}}} + \mathbf{K}\bar{\mathbf{v}} - \mathbf{G}\bar{\mathbf{p}} - \mathbf{f} = \mathbf{0} \quad (22a)$$

$$\mathbf{G}^T \bar{\mathbf{v}} + \mathbf{L}\bar{\mathbf{p}} + \mathbf{Q}\bar{\boldsymbol{\pi}} = \mathbf{0} \quad (22b)$$

$$\mathbf{Q}^T \bar{\mathbf{p}} + \hat{\mathbf{M}}\bar{\boldsymbol{\pi}} = \mathbf{0} \quad (22c)$$

The matrices and vectors in Eqs.(22) are assembled from the element contributions given by (for 2D problems)

$$\begin{aligned} \mathbf{M}_{ij} &= \int_{\Omega^e} \rho \mathbf{N}_i \mathbf{N}_j d\Omega \quad , \quad \mathbf{K}_{ij} = \int_{\Omega^e} \mathbf{B}_i^T \mathbf{D} \mathbf{B}_j d\Omega \\ \mathbf{D} &= \mu \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad , \quad \mathbf{B}_i = \begin{bmatrix} \frac{\partial N_i}{\partial x_1} & 0 \\ 0 & \frac{\partial N_i}{\partial x_2} \\ \frac{\partial N_i}{\partial x_2} & \frac{\partial N_i}{\partial x_1} \end{bmatrix} \\ L_{ij} &= \int_{\Omega^e} \tau_k \frac{\partial N_i}{\partial x_k} \frac{\partial N_j}{\partial x_k} d\Omega \quad , \quad \mathbf{Q} = [\mathbf{Q}^1, \mathbf{Q}^2] \quad , \quad Q_{ij}^k = \int_{\Omega^e} \tau_k \frac{\partial N_i}{\partial x_k} N_j d\Omega \\ \hat{\mathbf{M}} &= \begin{bmatrix} \hat{\mathbf{M}}^1 & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{M}}^2 \end{bmatrix} \quad , \quad \hat{\mathbf{M}}_{ij}^k = \int_{\Omega^e} \tau_k N_i N_j d\Omega \quad , \quad \mathbf{G}_{ij} = \int_{\Omega^e} \mathbf{B}_i^T \mathbf{m} N_j d\Omega \\ \mathbf{f}_i &= \int_{\Omega^e} N_i \mathbf{b} d\Omega + \int_{\Gamma_i^e} N_i \mathbf{t} d\Gamma \quad , \quad \mathbf{b} = [b_1, b_2]^T \quad , \quad \mathbf{t} = [t_1, t_2]^T \end{aligned} \quad (23)$$

with $i, j = 1, n$ and $k, l = 1, 2$.

In above \mathbf{B} is the strain rate matrix and $\mathbf{m} = [1, 1, 0]^T$ for 2D problems.

5 Fractional Step Method for Fluid-Structure Interaction Analysis

A simple and effective iterative algorithm can be obtained by splitting the pressure from the momentum equations as follows

$$\bar{\mathbf{v}}^* = \bar{\mathbf{v}}^n - \Delta t \mathbf{M}^{-1} [\mathbf{K} \bar{\mathbf{v}}^{n+1, j-1} - \mathbf{G} \bar{\mathbf{p}}^n - \mathbf{f}^{n+1}] \quad (24)$$

$$\bar{\mathbf{v}}^{n+1, j} = \bar{\mathbf{v}}^* + \Delta t \mathbf{M}^{-1} \mathbf{G} \delta \bar{\mathbf{p}} \quad (25)$$

where $\delta \bar{\mathbf{p}}$ denotes a pressure increment. In above equations and in the following superindex n refers to the time step whereas superindex j denotes an iteration number within each time step.

The value of $\bar{\mathbf{v}}^{n+1, j}$ from Eq.(28b) is substituted now into Eq.(22b) to give

$$\mathbf{G}^T \bar{\mathbf{v}}^* + \Delta t \mathbf{S} \delta \bar{\mathbf{p}} + \mathbf{L} \bar{\mathbf{p}}^{n+1, j} + \mathbf{Q} \bar{\boldsymbol{\pi}}^{n+1, j-1} = \mathbf{0} \quad (26a)$$

where

$$\mathbf{S} = \mathbf{G}^T \mathbf{M}^{-1} \mathbf{G} \quad (26b)$$

Typically matrix \mathbf{S} is computed using a diagonal matrix $\mathbf{M} = \mathbf{M}_d$, where the subscript d denotes hereonward a diagonal matrix.

An alternative is to approximate matrix \mathbf{S} by a Laplacian matrix. This reduces considerably the bandwidth of \mathbf{S} . The disadvantage is that the pressure increment must be then prescribed on the free surface and this reduces the accuracy in the satisfaction of the incompressibility condition in these regions (see Remark 1).

A semi-implicit algorithm can be derived as follows. For each iteration:

Step 1 Compute \mathbf{v}^* from Eq.(24) with $\mathbf{M} = \mathbf{M}_d$. For the first iteration \mathbf{p}^1 is taken as the hydrostatic pressure.

Step 2 Compute $\delta \bar{\mathbf{p}}$ and \mathbf{p}^{n+1} from Eq.(26a) as

$$\delta \bar{\mathbf{p}} = -(\mathbf{L} + \Delta t \mathbf{S})^{-1} [\mathbf{G}^T \bar{\mathbf{v}}^* + \mathbf{Q} \bar{\boldsymbol{\pi}}^{n+1,j-1} + \mathbf{L} \bar{\mathbf{p}}^n] \quad (27a)$$

The pressure $\bar{\mathbf{p}}^{n+1,j}$ is computed as follows

$$\bar{\mathbf{p}}^{n+1,j} = \bar{\mathbf{p}}^n + \delta \bar{\mathbf{p}} \quad (27b)$$

Step 3 Compute $\bar{\mathbf{v}}^{n+1,j}$ from Eq.(25) with $\mathbf{M} = \mathbf{M}_d$

Step 4 Compute $\bar{\boldsymbol{\pi}}^{n+1,j}$ from Eq.(22c) as

$$\bar{\boldsymbol{\pi}}^{n+1,j} = -\hat{\mathbf{M}}_d^{-1} \mathbf{Q}^T \bar{\mathbf{p}}^{n+1,j} \quad (28)$$

Step 5 Solve for the motion of the structure due to the fluid flow forces.

This implies solving the dynamic equations of motion for the structure written as

$$\mathbf{M}_s \ddot{\mathbf{d}} + \mathbf{K}_s \mathbf{d} = \mathbf{f}_{\text{ext}} \quad (29)$$

where \mathbf{d} and $\ddot{\mathbf{d}}$ are respectively the displacement and acceleration vectors of the nodes discretizing the structure, \mathbf{M}_s and \mathbf{K}_s are the mass and stiffness matrices of the structure and \mathbf{f}_{ext} is the vector of external nodal forces accounting for the fluid flow forces induced by the pressure and the viscous stresses. Clearly the main driving forces for the motion of the structure is the fluid pressure which acts as normal surface traction on the structure. Indeed Eq.(29) can be augmented with an appropriate damping term. The form of all the relevant matrices and vectors can be found in standard books on FEM for structural analysis [Zienkiewicz and Taylor (2005)].

Solution of Eq.(29) in time can be performed using implicit or fully explicit time integration algorithms. In both cases the values of the nodal displacements, velocities and accelerations of the structure at t^{n+1} are found for the j th iteration.

Step 6 Update the mesh nodes in a Lagrangian manner. From the definition of the velocity $v_i = \frac{\partial u_i}{\partial t}$ it is deduced.

$$\mathbf{x}_i^{n+1,j} = \mathbf{x}_i^n + \bar{\mathbf{v}}_i^{n+1,j} \Delta t \quad (30)$$

Step 7 Check the convergence of the velocity and pressure fields in the fluid and the displacements strains and stresses in the structure. If convergence is achieved move to the next time step, otherwise return to step 1 for the next iteration with $j + 1 \rightarrow j$.

Despite the motion of the nodes within the iterative process, in general there is no need to regenerate the mesh at each iteration. A new mesh is typically generated after a prescribed number of converged time steps, or when the nodal displacements induce significant geometrical distortions in some elements. *In the examples presented the mesh in the fluid domain has been regenerated at the beginning of each time step.*

Note that solution of steps 1, 3 and 4 does not require the solution of a system of equations as a diagonal form is chosen for \mathbf{M} and $\hat{\mathbf{M}}$.

In the examples presented in the paper the time increment size has been chosen as

$$\Delta t = \min(\Delta t_i) \quad \text{with} \quad \Delta t_i = \frac{h_i^{\min}}{|\mathbf{v}|} \quad (31)$$

where h_i^{\min} is the distance between node i and the closest node in the mesh.

Although not explicitly mentioned all matrices and vectors in Eqs.(27)–(31) are computed at the configuration $\Omega^{n+1,j}$. This means that the integration domain changes for each iteration and, hence, all the terms involving space derivatives must be updated at each iteration. An alternative is to refer the integrations domain at each time step to Ω^n . The jacobian matrix is needed in this case to transform the space derivatives and the differential of volume from $\Omega^{n+1,j}$ to Ω^n at each iteration.

Remark 1. The boundary conditions are applied as follows. No condition is applied in the computation of the fractional velocities \mathbf{v}^* in Eq.(25a). The prescribed velocities at the boundary are applied when solving for $\bar{\mathbf{v}}^{n+1,j}$ in step 3.

The form of \mathbf{S} in Eq.(26b) avoids the need to prescribing the pressure at the boundary nodes. It is however recommended to fix the pressure at a point in the analysis domain so as to ensure the correct definition of the pressure field for all problems. An alternative procedure is to approximate \mathbf{S} by a Laplacian matrix [Oñate *et al.* (2004), Idelsohn *et al.* (2004)]. The pressure increment on the free surface must be prescribed in this case to the value $2\mu \frac{\partial \Delta \bar{v}_n}{\partial n}$ where $\Delta \bar{v}_n = \mathbf{n}^T \Delta \bar{\mathbf{v}}$ is the velocity increment along the normal direction to the boundary and $\Delta \bar{\mathbf{v}} = \bar{\mathbf{v}}^{n+1,j} - \bar{\mathbf{v}}^n$. We have found however that the form of \mathbf{S} given by Eq.(26b) allows to satisfy better the incompressibility condition in the vicinity of the free surfaces, thereby leading to smaller volume losses in transient problems involving large motions of the free surface.

6 Treatment of Between Contact Fluid and Solid Interfaces

The motion of the solid is governed by the action of the fluid flow forces induced by the pressure and the viscous stresses acting at the solid boundary, as mentioned above.

The condition of prescribed velocities at the solid boundaries in the PFEM are applied in strong form to the boundary nodes. These nodes might belong to fixed external boundaries or to moving boundaries linked to the interacting solids. Contact between water particles and the solid boundaries is accounted for by the incompressibility condition which *naturally prevents the penetration of the water nodes into the solid boundaries*. This simple way to treat the water-wall contact is another attractive feature of the PFEM formulation.

7 Generation of a New Mesh

One of the key points for the success of the Lagrangian flow formulation described here is the fast regeneration of a mesh at every time step on the basis of the position of the nodes in the space domain. In our work the mesh is generated using the so called extended Delaunay tessellation (EDT) presented in [Idelsohn *et al.* (2003a; 2003c; 2004)]. The EDT allows one to generate non standard meshes combining elements of arbitrary polyhedral shapes (triangles, quadrilaterals and other polygons in 2D and tetrahedra, hexahedra and arbitrary polyhedra in 3D) in a computing time of order n , where n is the total number of nodes in the mesh (Figure 2). The C^0 continuous shape functions of the elements can be simply obtained using the so called meshless finite element interpolation (MFEM). Details of the mesh generation procedure and the derivation of the MFEM shape functions can be found in [Idelsohn *et al.* (2003a; 2003c; 2004)].

Once the new mesh has been generated the numerical solution is found at each time step using the fractional step algorithm described in the previous section.

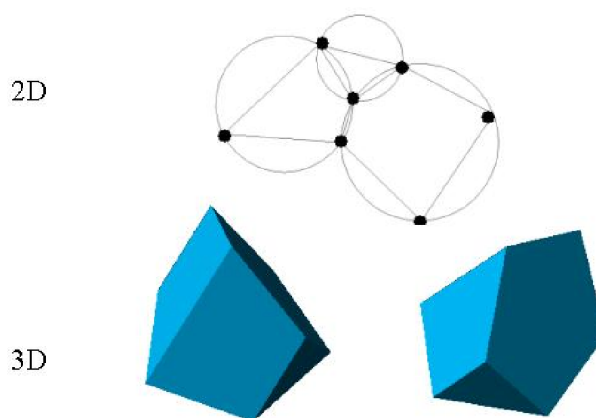


Fig. 2. Generation of non standard meshes combining different polygons (in 2D) and polyhedra (in 3D) using the extended Delaunay technique.

8 Identification of Boundary Surfaces

One of the main tasks in the PFEM is the correct definition of the boundary domain. Sometimes, boundary nodes are explicitly identified differently from internal nodes. In other cases, the total set of nodes is the only information available and the algorithm must recognize the boundary nodes.

The extended Delaunay partition makes it easier to recognize boundary nodes. Considering that the nodes follow a variable $h(x)$ distribution, where $h(x)$ is typically the minimum distance between two nodes, the following criterion has been used. *All nodes on an empty sphere with a radius greater than αh , are considered as boundary nodes.* In practice α is a parameter close to, but greater than one. This criterion is coincident with the Alpha Shape concept [Edelsbrunner and Mücke (1999)].

Figure 3 shows example of the boundary recognition using the Alpha Shape technique.

Once a decision has been made concerning which nodes are on the boundaries, the boundary surface is defined by all the polyhedral surfaces (or polygons in 2D) having all their nodes on the boundary and belonging to just one polyhedron.

The correct boundary surface is important to define the normal external to the surface. Furthermore, in weak forms (Galerkin) such as those used here a correct evaluation of the volume domain is also important. In the criterion proposed above, the error in the boundary surface definition is proportional to h which is an acceptable error.

The method described also allows one to identify isolated fluid particles outside the main fluid domain. These particles are treated as part of the external boundary where the pressure is fixed to the atmospheric value (Figure 3) .

A practical application of the method for identifying free surface particles is shown in Figure 4. The example corresponds to the analysis of the motion of a fluid within an oscillating ellipsoidal container.

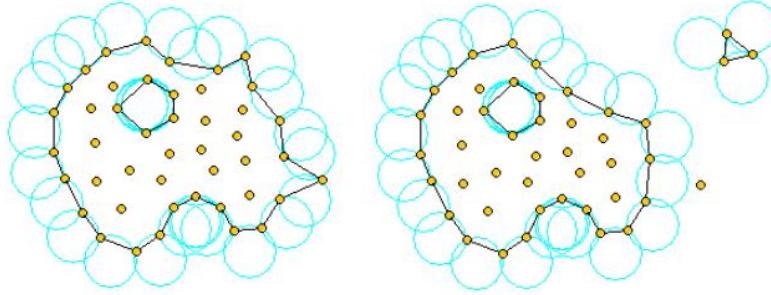


Fig. 3. Identification of individual particles (or a group of particles) starting from a given collection of nodes.

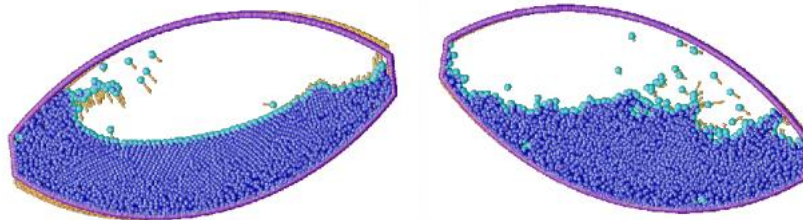


Fig. 4. Motion of a liquid within an oscillating container. Position of the liquid particles at two different times.

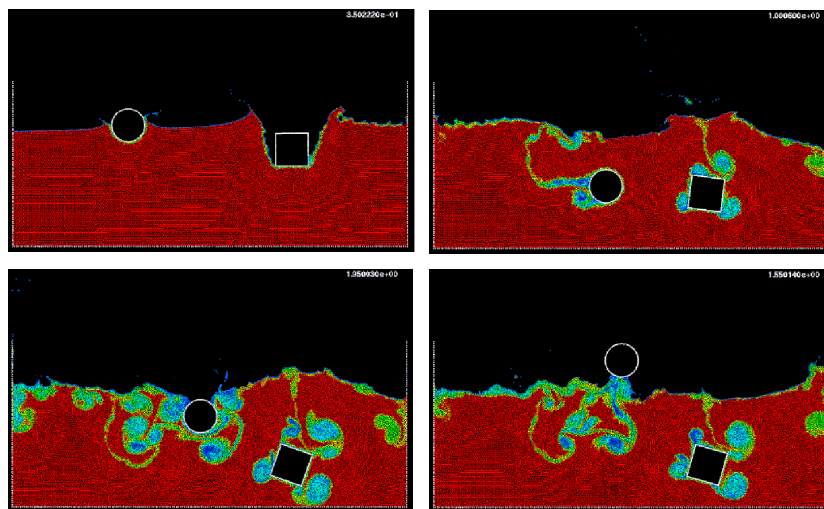


Fig. 5. 2D simulation of the penetration and evolution of a cube and a cylinder in a water container. The colours denote the different sizes of the elements at several times.

9 Examples

The examples chosen show the applicability of the PFEM to solve problems involving large fluid motions and FSI situations.

9.1 Rigid objects filling into water

The analysis of the motion of submerged or floating objects in water is of great interest in many areas of harbour and coastal engineering and naval architecture among others.

Figure 5 shows the penetration and evolution of a cube and a cylinder in a container with water. The colours denote the different sizes of the elements at several times. Figure 6 shows that the mesh generation algorithm ensures a smaller size of elements in the vicinity of the moving bodies during their motion.

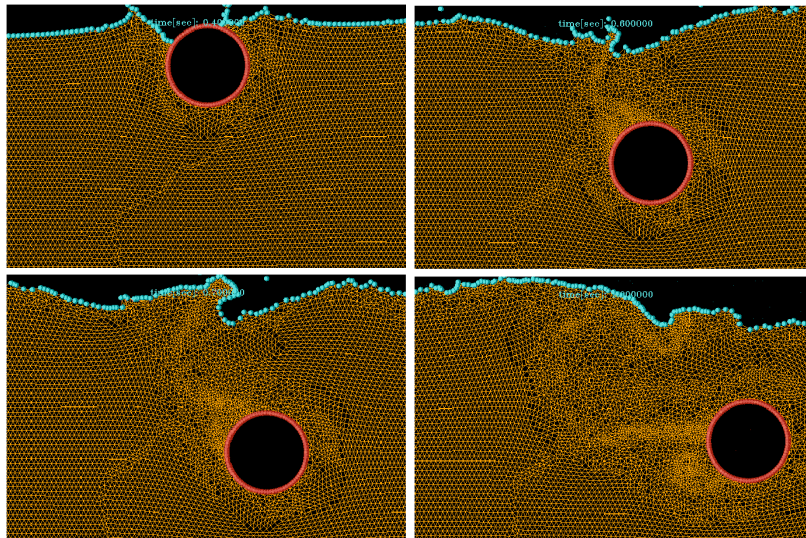


Fig. 6. Detail of element sizes during the motion of a rigid cylinder within a water container.

9.2 Impact of water streams on rigid structures

Figure 7 shows the simulation of the propagation of a wave generated in a test canal impacting on a vertical wall. Figure 8 shows a comparison of the experimental and numerical values of the water pressure on the vertical wall. The overall agreement is noticeable.

Figure 9 shows an example of a wave breaking within a prismatic container including a vertical cylinder. Finally Figure 10 shows the impact of a wave on a vertical column sustained by four pillars. The objective of this example was to model the impact of a water stream on a bridge pier accounting for the foundation effects.

9.3 Dragging of objects by water streams

Figure 11 shows the effect of a wave impacting on a cube representing a vehicle. This situation is typical in flooding and Tsunami situations.

9.4 Impact of sea waves on breakwater and piers

Figure 12 shows the simulation of the impact of a wave generated in an experimental flume on a collection of water motion in the vicinity of the rocks represent a breakwater. Details of the water-rock interaction are shown in Figure 13.

Figure 14 shows a 3D analysis of a similar problem. Figure 15 shows the 3D simulation of the interaction of a wave with a vertical pier formed by a collection of reinforced concrete cylinders.

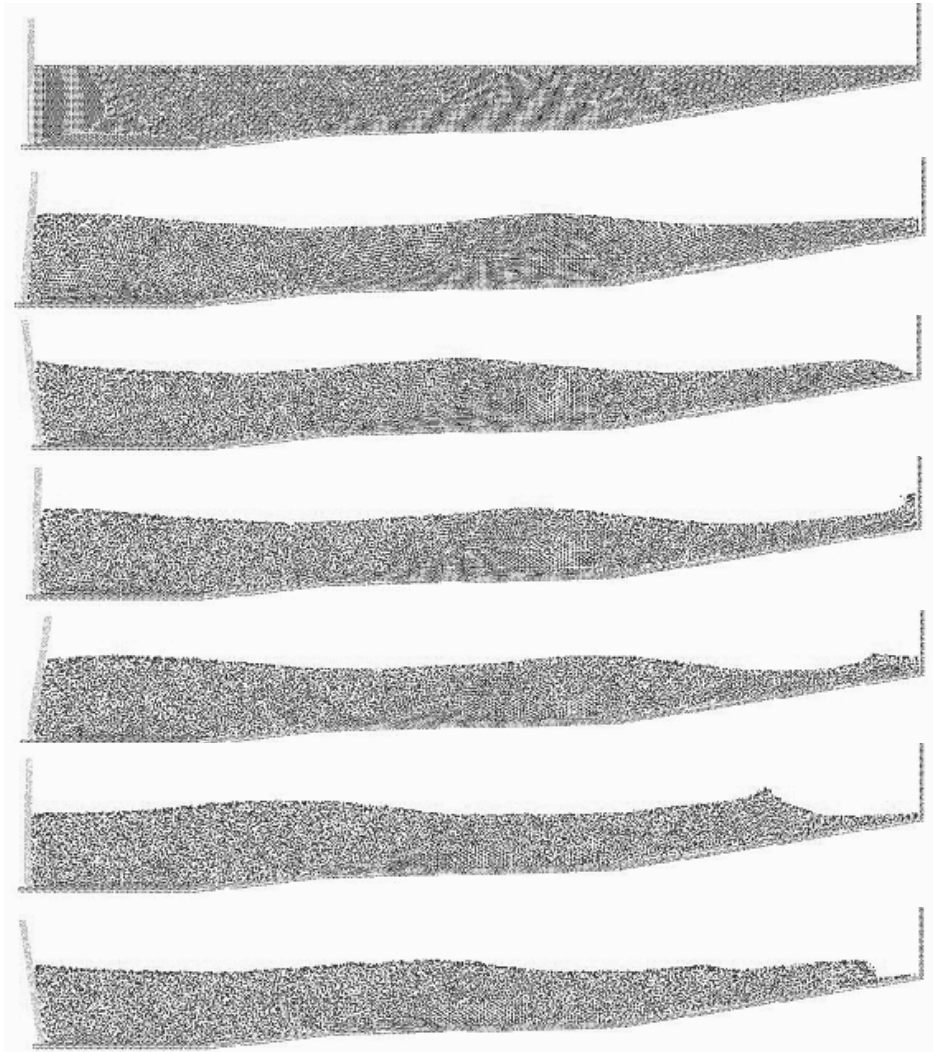


Fig. 7. Propagation of a wave generated in a test canal impacting with a vertical wall.

The last examples shown in Figures 16 and 17 evidence the potential of the PFEM to solve 3D problems involving complex interactions between water and solid objects. Figure 16 shows the simulation of the falling of two tetrapods in a water container. Finally, Figure 17 shows the motion of a collection of ten tetrapods placed in a slope under an incident wave.

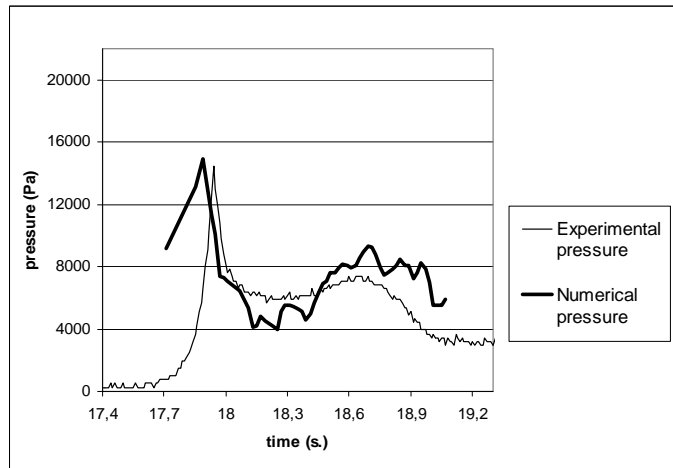


Fig. 8. Comparison of experimental and numerical water pressures at the vertical wall for the example of Figure 7.

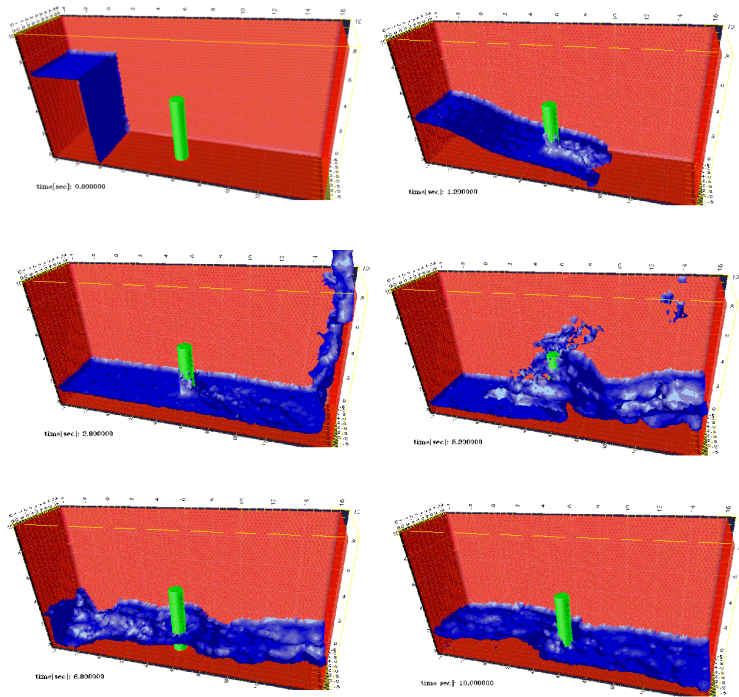


Fig. 9. Evolution of a water column within a prismatic container including a vertical cylinder.

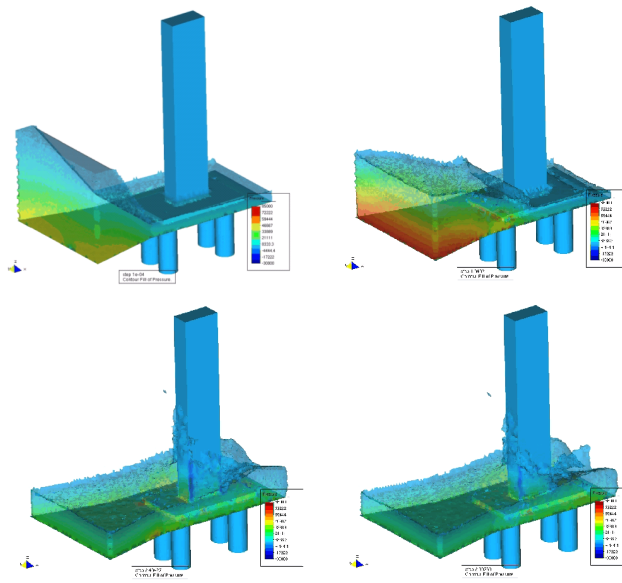


Fig. 10. Impact of a wave on a prismatic column on a slab sustained by four pillars.

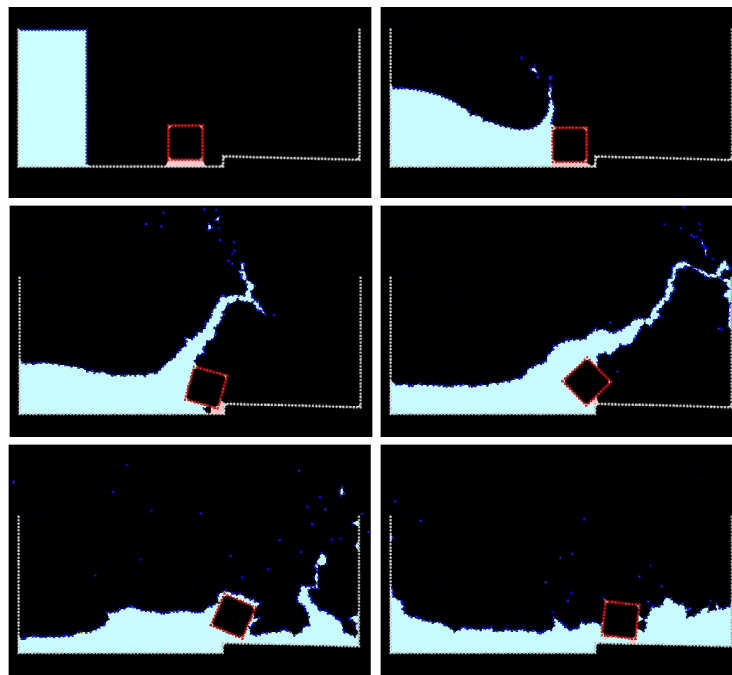


Fig. 11. Dragging of a cubic object by a water stream.

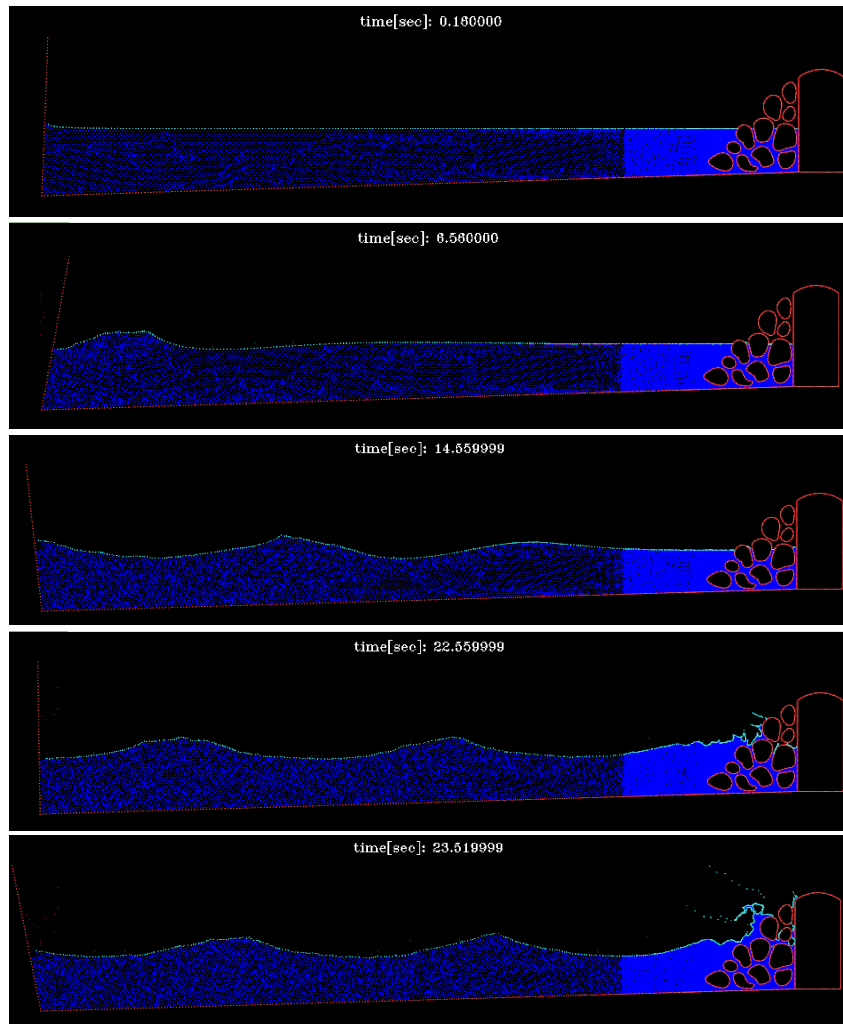


Fig. 12. Generation and impact of a wave on a collection of rocks in a breakwater.

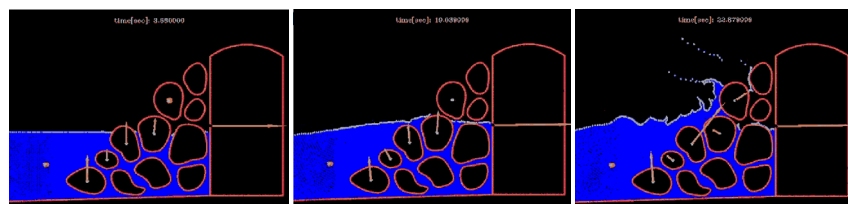


Fig. 13. Detail of the impact of a wave on a breakwater. The arrows indicate the water force on the rocks at different instants.

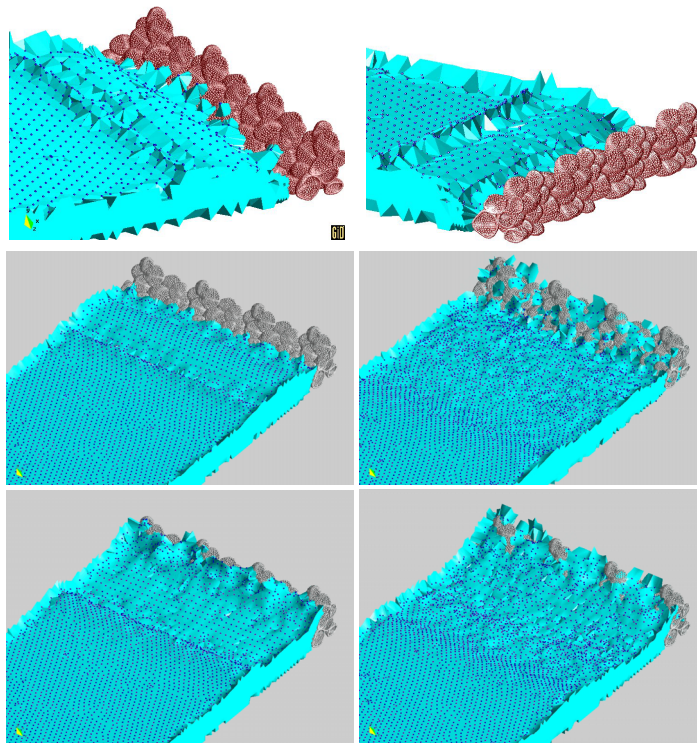


Fig. 14. 3D simulation of the impact of a wave on a collection of rocks in a breakwater.

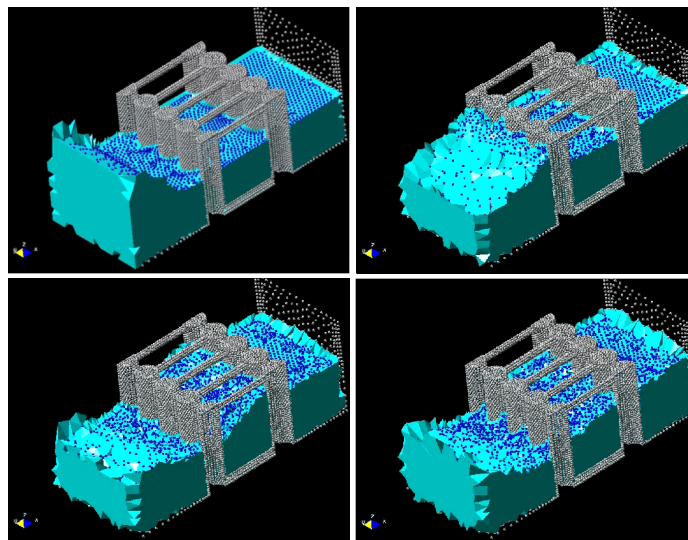


Fig. 15. Interaction of a wave with a vertical pier formed by reinforced concrete cylinders.

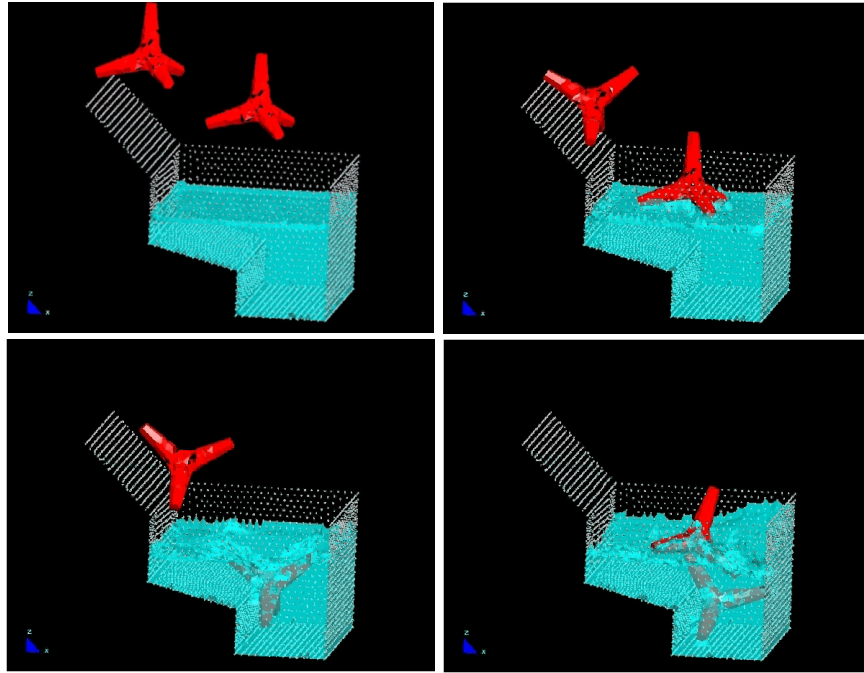


Fig. 16. Motion of two tetrapods falling in a water container.

10 Conclusions

The particle finite element method (PFEM) is ideal to treat problems involving fluids with free surface and submerged or floating structures within a unified Lagrangian finite element framework. Problems such as the analysis of fluid-structure interaction, large motion of fluid or solid particles, surface waves, water splashing, separation of water drops, etc. can be easily solved with the PFEM. The success of the method lies in the accurate and efficient solution of the equations of an incompressible fluid and of solid dynamics using a stabilized finite element method allowing the use of low order elements with equal order interpolation for all the variables. Other essential solution ingredients are the efficient regeneration of the finite element mesh using an extended Delaunay tessellation, the meshless finite element interpolation (MFEM) and the identification of the boundary nodes using an Alpha Shape type technique. The examples presented have shown the potential of the PFEM for solving a wide class of practical FSI problems.

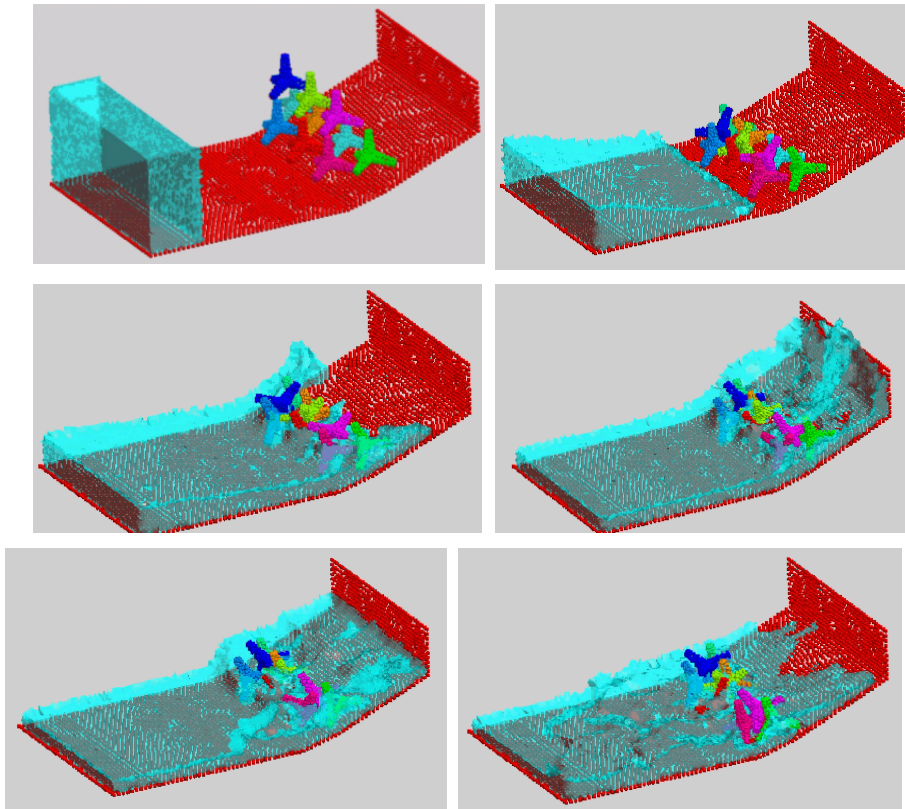


Fig. 17. Motion of ten tetrapods on a slope under an incident wave.

Acknowledgements

Thanks are given to Dr. F. Del Pin, Dr. N. Calvo and Ms. M. de Mier for many useful suggestions.

References

1. Aubry R, Idelsohn SR, Oñate E (2005) Particle finite element method in fluid mechanics including thermal convection-diffusion. *Computer & Structures* 83(17-18):1459–1475
2. Codina R, Zienkiewicz OC (2002) CBS versus GLS stabilization of the incompressible Navier-Stokes equations and the role of the time step as stabilization parameter. *Communications in Numerical Methods in Engineering* 18(2):99–112
3. Donea J, Huerta A (2003) *Finite element method for flow problems*. J. Wiley
4. Edelsbrunner H, Mücke EP (1999) Three dimensional alpha shapes. *ACM Trans. Graphics* 13:43–72

5. García J, Oñate E (2003) An unstructured finite element solver for ship hydrodynamic problems. *J. Appl. Mech.* 70:18–26 January
6. Idelsohn SR, Oñate E, Del Pin F, Calvo N (2002) Lagrangian formulation: the only way to solve some free-surface fluid mechanics problems. *Fifth World Congress on Computational Mechanics*, Mang HA, Rammerstorfer FG, Eberhardsteiner J (eds), July 7–12, Vienna, Austria
7. Idelsohn SR, Oñate E, Calvo N, Del Pin F (2003a) The meshless finite element method. *Int. J. Num. Meth. Engng.* 58(6):893–912
8. Idelsohn SR, Oñate E, Del Pin F (2003b) A lagrangian meshless finite element method applied to fluid-structure interaction problems. *Computer and Structures* 81:655–671
9. Idelsohn SR, Calvo N, Oñate E (2003c) Polyhedrization of an arbitrary point set. *Comput. Method Appl. Mech. Engng.* 192(22-24):2649–2668
10. Idelsohn SR, Oñate E, Del Pin F (2004) The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves. *Int. J. Num. Meth. Engng.* 61:964–989
11. Oñate E (1998) Derivation of stabilized equations for advective-diffusive transport and fluid flow problems. *Comput. Meth. Appl. Mech. Engng.* 151:233–267
12. Oñate E, (2000) A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation. *Comp. Meth. Appl. Mech. Engng.* 182(1–2):355–370
13. Oñate E (2004) Possibilities of finite calculus in computational mechanics. *Int. J. Num. Meth. Engng.* 60(1):255–281
14. Oñate E, Idelsohn SR (1998) A mesh free finite point method for advective-diffusive transport and fluid flow problems. *Computational Mechanics* 21:283–292
15. Oñate E, García J (2001) A finite element method for fluid-structure interaction with surface waves using a finite calculus formulation. *Comput. Meth. Appl. Mech. Engng.* 191:635–660
16. Oñate E, Sacco C, Idelsohn SR (2000) A finite point method for incompressible flow problems. *Comput. Visual. in Science* 2:67–75
17. Oñate E, Idelsohn SR, Del Pin F (2003) Lagrangian formulation for incompressible fluids using finite calculus and the finite element method. *Numerical Methods for Scientific Computing Variational Problems and Applications*, Y Kuznetsov, P Neittanmaki, O Pironneau (Eds.), CIMNE, Barcelona
18. Oñate E, García J, Idelsohn SR (2004a) Ship hydrodynamics. *Encyclopedia of Computational Mechanics*, E Stein, R de Borst, T.J.R. Hughes (Eds), J. Wiley.
19. Oñate E, Idelsohn SR, Del Pin F, Aubry R (2004b) The particle finite element method. An overview. *Int. J. Comput. Methods* 1(2):267–307
20. Zienkiewicz OC, Taylor RL, Nithiarasu P (2006) *The finite element method for fluid dynamics*, Elsevier
21. Zienkiewicz OC, Taylor RL (2005) *The finite element method for solid and structural mechanics*. Elsevier