Available online at www.sciencedirect.com

ScienceDirect

Comput. Methods Appl. Mech. Engrg. xxx (2007) xxx–xxx

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

Advances in the particle finite element method for the analysis of fluid–multibody interaction and bed erosion in free surface flows

Eugenio Oñate *, Sergio R. Idelsohn [1], Miguel A. Celigueta, Riccardo Rossi

International Center for Numerical Methods in Engineering (CIMNE), Technical University of Catalonia, Campus Norte UPC, 08034 Barcelona, Spain

Received 16 November 2006; received in revised form 1 June 2007; accepted 4 June 2007

Abstract

We present some advances in the formulation of the particle finite element method (PFEM) for solving complex fluid–structure interaction problems with free surface waves. In particular, we present extensions of the PFEM for the analysis of the interaction between a collection of bodies in water allowing for frictional contact conditions at the fluid–solid and solid–solid interfaces via mesh generation. An algorithm to treat bed erosion in free surface flows is also presented. Examples of application of the PFEM to solve a number of fluid–multibody interaction problems involving splashing of waves, large motions of floating and submerged bodies and bed erosion situations are given.

Keywords: Lagrangian formulation; Fluid–structure interaction; Particle finite element method; Bed erosion; Free surface flows

1. Introduction

The analysis of 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 which interact among themselves is of big relevance in many areas of engineering. Examples are common in ship hydrodynamics, off-shore and harbour structures, spill-ways in dams, free surface channel flows, environmental flows, liquid containers, stirring reactors, mould filling processes, etc.

Typical difficulties of fluid–multibody interaction analysis in free surface flows using the FEM with both the Eulerian and ALE formulation include the treatment of the convective terms and the incompressibility constraint in the fluid equations, the modelling and tracking of the free surface in the fluid, the transfer of information between the fluid and the moving solid domains via the contact interfaces, the modeling of wave splashing, the possibility to deal with large motions of the bodies within the fluid domain, the efficient updating of the finite element meshes for both the structure and the fluid, etc. For a comprehensive list of references in FEM for fluid flow problems see [5,34] and the references there included. A survey of recent works in fluid–structure interaction analysis can be found in [16,25,32].

Most of the above problems disappear if a *Lagrangian description* is used to formulate the governing equations of both the solid and the fluid domains. 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 material points (hereforth called "particles"). Hence, the motion of the mesh discretizing the total domain (including both the fluid and solid parts) is followed during the transient solution.

The authors have successfully developed in previous works a particular class of Lagrangian formulation for solving problems involving complex interaction between fluids and solids. The method, called the *particle finite element method* (PFEM), treats the mesh nodes in the fluid

* Corresponding author. Tel.: +34 932057016; fax: +34 934016517.
E-mail addresses: onate@cimne.upc.edu, majesus@cimne.upc.edu (E. Oñate).
URL: http://www.cimne.com (E. Oñate).

[1] ICREA Research Professor at CIMNE.

0045-7825/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2007.06.005

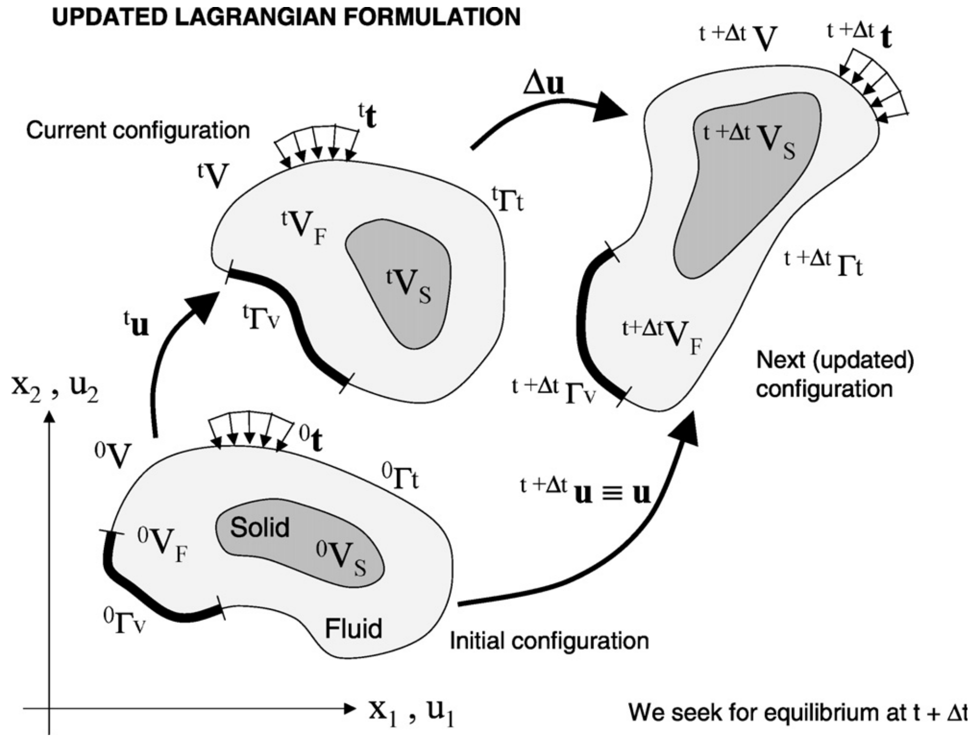


Fig. 1. Updated Lagrangian description for a continuum containing a fluid and a solid domain.

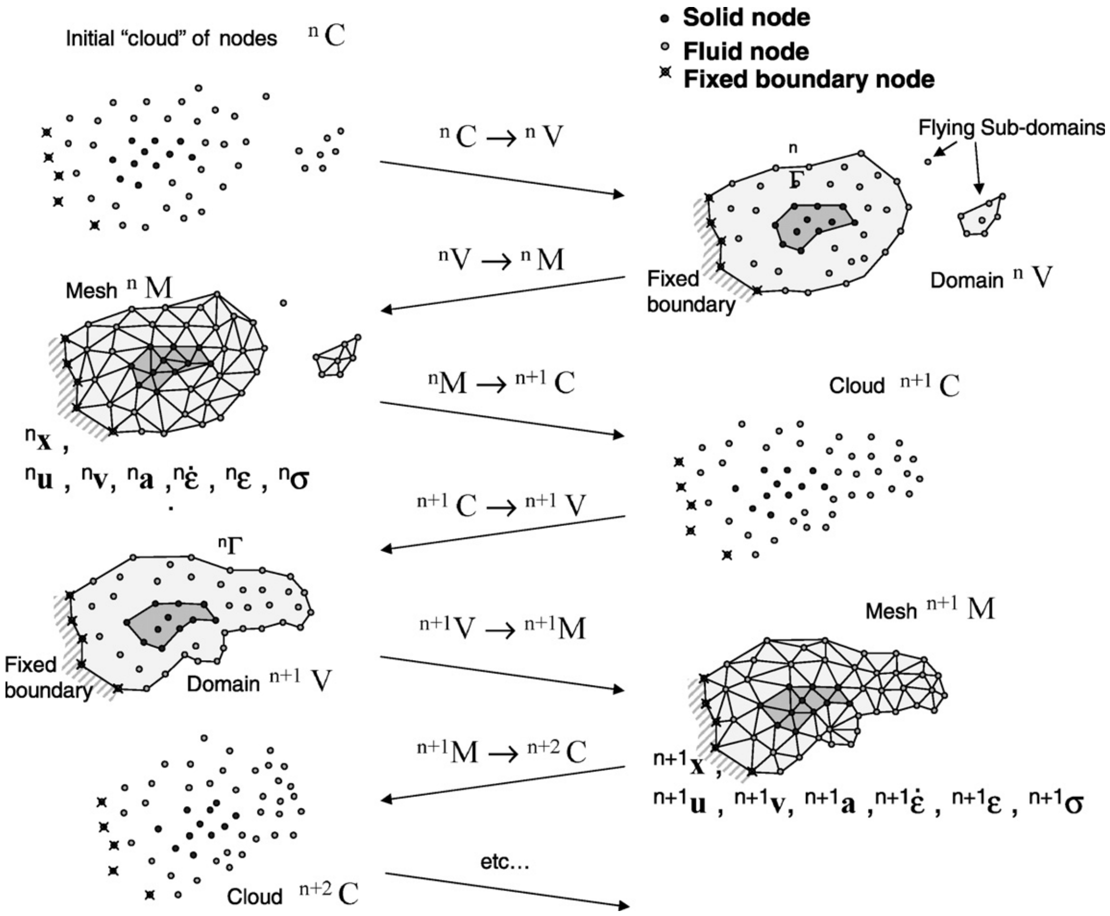

Initial “cloud” of nodes ^{n}C
● Solid node
○ Fluid node
✱ Fixed boundary node
$^{n}C \rightarrow {}^{n}V$
$^{n}\Gamma$
Flying Sub-domains
$^{n}V \rightarrow {}^{n}M$
Fixed boundary
Domain ^{n}V
Mesh ^{n}M
$^{n}M \rightarrow {}^{n+1}C$
Cloud ^{n+1}C
$^{n}\mathbf{x}$,
$^{n}\mathbf{u}$, $^{n}\mathbf{v}$, $^{n}\mathbf{a}$, $^{n}\dot{\varepsilon}$, $^{n}\varepsilon$, $^{n}\sigma$
$^{n+1}C \rightarrow {}^{n+1}V$
$^{n}\Gamma$
Mesh ^{n+1}M
$^{n+1}V \rightarrow {}^{n+1}M$
Fixed boundary
Domain ^{n+1}V
$^{n+1}M \rightarrow {}^{n+2}C$
$^{n+1}\mathbf{x}$,
$^{n+1}\mathbf{u}$, $^{n+1}\mathbf{v}$, $^{n+1}\mathbf{a}$, $^{n+1}\dot{\varepsilon}$, $^{n+1}\varepsilon$, $^{n+1}\sigma$
etc...
Cloud ^{n+2}C

Fig. 2. Sequence of steps to update a “cloud” of nodes from time n ($t = t_n$) to time $n + 2$ ($t = t_n + 2\Delta t$).

Please cite this article in press as: E. Oñate et al., Advances in the particle finite element method for the analysis of ..., Comput. Methods Appl. Mech. Engrg. (2007), doi:10.1016/j.cma.2007.06.005

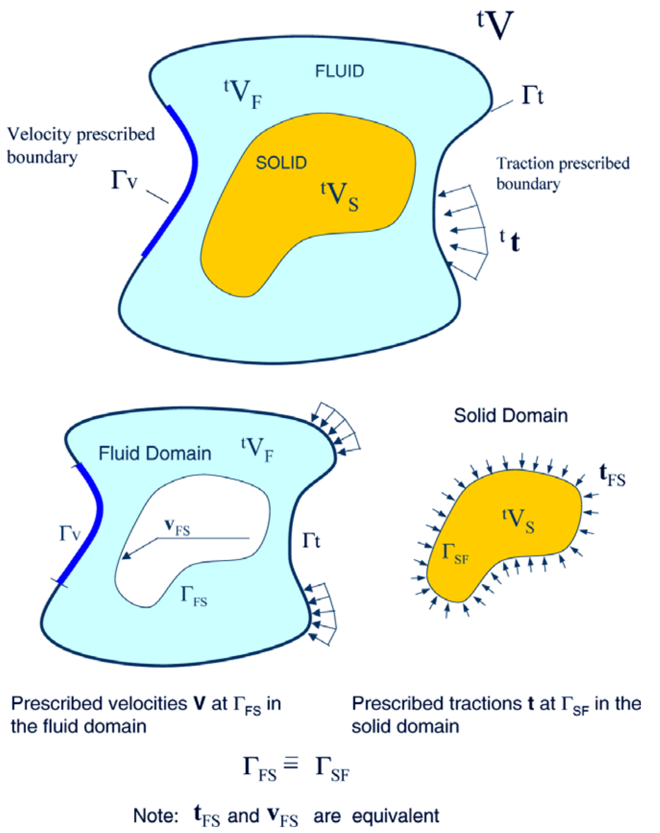

${}^{t}V$
FLUID
${}^{t}V_F$
Γt
Velocity prescribed boundary
Γv
SOLID
${}^{t}V_S$
Traction prescribed boundary
${}^{t}\mathbf{t}$
Solid Domain
Fluid Domain
${}^{t}V_F$
$\mathbf{t}_{FS}$
Γv
$\mathbf{v}_{FS}$
Γt
${}^{t}V_S$
Γ_{SF}
Γ_{FS}
Prescribed velocities **V** at Γ_{FS} in the fluid domain
Prescribed tractions **t** at Γ_{SF} in the solid domain
$\Gamma_{FS} \equiv \Gamma_{SF}$
Note: $\mathbf{t}_{FS}$ and $\mathbf{v}_{FS}$ are equivalent

Fig. 3. Split of the analysis domain V into fluid and solid subdomains. Equality of surface tractions and kinematic variables at the common interface.

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 using a stabilized FEM based in the Finite Calculus (FIC) approach. An 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. We use a mesh regeneration procedure blending elements of different shapes using an extended Delaunay tesselation with special shape functions [9,11]. The theory and applications of the PFEM are reported in [2,4,9,10,12,13,24–26,28,29].

The aim of this paper is to describe two recent advances of the PFEM: (a) the analysis of the interaction between a collection of bodies which are floating and/or submerged in the fluid, and (b) the modeling of bed erosion in open channel flows. Both problems are of great relevance in many areas of civil, marine and naval engineering, among others. It is shown in the paper that the PFEM provides a general analysis methodology for treat such a complex problems in a simple and efficient manner.

The layout of the paper is the following. In the next section the key ideas of the PFEM are outlined. Next the basic equations for an incompressible flow using a Lagrangian description and the FIC formulation are presented. Then a fractional step scheme for the transient solution is briefly described. Details of the treatment of the coupled FSI problem are given. The methods for mesh generation and for identification of the free surface nodes are outlined. The procedure for treating at mesh generation level the contact conditions at fluid–wall interfaces and the frictional contact interaction between moving solids is explained. A methodology for modeling bed erosion due to fluid forces is described. Finally, the efficiency of the PFEM is shown in its application to a number of problems

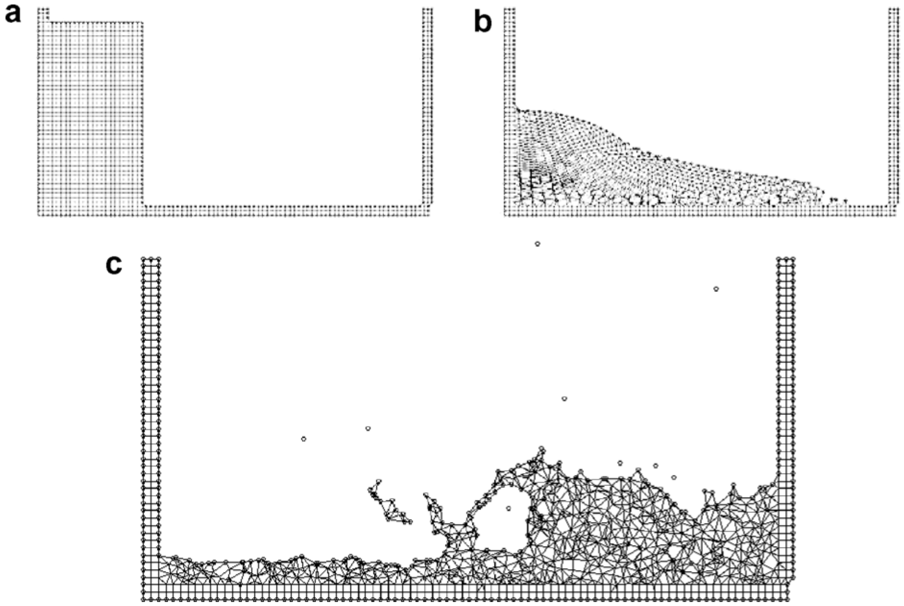


Fig. 4. Breakage of a water column. (a) Discretization of the fluid domain and the solid walls. Boundary nodes are marked with circles. (b) and (c) Mesh in the fluid domain at two different times.

involving large flow motions, surface waves, moving bodies in water and bed erosion.

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 both the fluid and the solid domains are modelled using an *updated Lagrangian formulation*. That is, all variables in the fluid and solid domains are assumed to be known in the *current configuration* at time t. The new set of variables in both domains are sought for in the *next or updated configuration* at time $t + \Delta t$ (Fig. 1). 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. Recall that the nodes discretizing the fluid and solid domains are treated as *material particles* which motion is tracked during the transient solution. This is useful to model the separation of fluid particles from the main fluid domain in a splashing wave, or soil particles in a bed erosion problem, and to follow their subsequent motion as individual particles with a known density, an initial acceleration and velocity and subject to gravity forces. The mass of a given domain is obtained by integrating the density at the different material points over the domain.

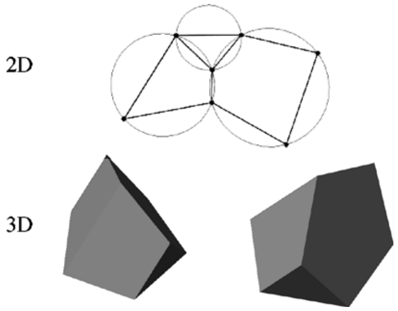


Fig. 5. Generation of non standard meshes combining different polygons (in 2D) and polyhedra (in 3D) using the extended Delaunay technique.

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.

2.1. Basic steps of the PFEM

For clarity purposes we will define the *collection or cloud of nodes* (C) pertaining to the fluid and solid domains, the *volume* (V) defining the analysis domain for the fluid and the solid and the *mesh* (M) discretizing both domains.

A typical solution with the PFEM involves the following steps:

1. The starting point at each time step is the cloud of points in the fluid and solid domains. For instance ${}^{n}C$ denotes the cloud at time $t = t_n$ (Fig. 2).
2. Identify the boundaries for both the fluid and solid domains defining the analysis domain ${}^{n}V$ in the fluid and the solid. This is an essential step as some boundaries (such as the free surface in fluids) may be severely

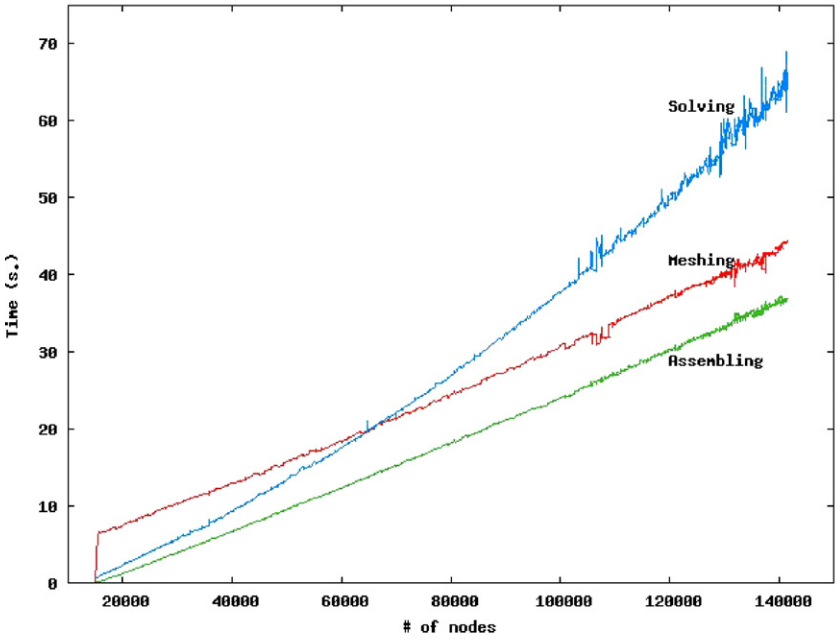


Fig. 6. 3D flow problem solved with the PFEM. CPU time for meshing, assembling and solving the system of equations at each time step in terms of the number of nodes.

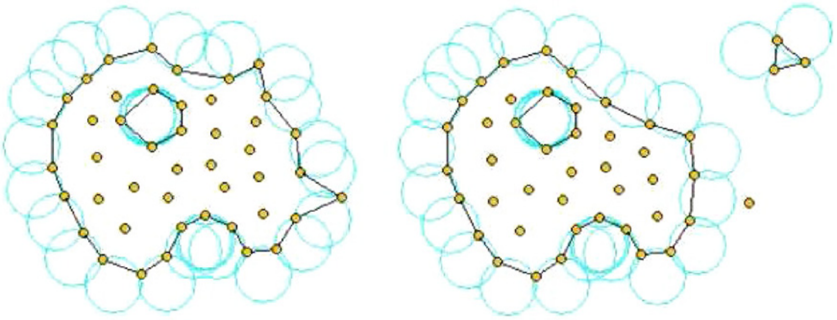


Fig. 7. Identification of individual particles (or a group of particles) starting from a given collection of nodes.

distorted during the solution, including separation and re-entering of nodes. The Alpha Shape method [6] is used for the boundary definition (Section 5).

3. Discretize the fluid and solid domains with a finite element mesh ${}^{n}M$. In our work we use an innovative mesh generation scheme based on the extended Delaunay tesselation (Section 4) [9,10,12].
4. Solve the coupled Lagrangian equations of motion for the fluid and the solid domains. Compute the relevant state variables in both domains at the next (updated) configuration for $t+\Delta t$: velocities, pressure and viscous stresses in the fluid and displacements, stresses and strains in the solid.
5. Move the mesh nodes to a new position ${}^{n+1}C$ where $n+1$ denotes the time $t_n+\Delta t$, in terms of the time increment size. This step is typically a consequence of the solution process of step 4.
6. Go back to step 1 and repeat the solution process for the next time step to obtain ${}^{n+2}C$. The process is shown in Fig. 2.

2.2. Overview of the coupled FSI algorithm

Fig. 3 shows a typical domain V with external boundaries $\Gamma_{\rm V}$ and $\Gamma_{\rm t}$ where the velocity and the surface tractions are prescribed, respectively. The domain V is formed by fluid ($V_{\rm F}$) and solid ($V_{\rm S}$) subdomains (i.e. $V=V_{\rm F}\cup V_{\rm S}$). Both subdomains interact at a common boundary $\Gamma_{\rm FS}$ where the surface tractions and the kinematic variables (displacements, velocities and accelerations) are the same for both subdomains. Note that both set of variables (the surface tractions and the kinematic variables) are equivalent in the equilibrium configuration.

Let us define ${}^{t}S$ and ${}^{t}F$ the set of variables defining the kinematics and the stress–strain fields at the solid and fluid domains at time t, respectively, i.e.

$$ {}^{t}S := [{}^{t}\mathbf{x}_{\rm s}, {}^{t}\mathbf{u}_{\rm s}, {}^{t}\mathbf{v}_{\rm s}, {}^{t}\mathbf{a}_{\rm s}, {}^{t}\boldsymbol{\varepsilon}_{\rm s}, {}^{t}\boldsymbol{\sigma}_{\rm s}, \ldots]^{\rm T}, \quad (1)$$

$$ {}^{t}F := [{}^{t}\mathbf{x}_{\rm F}, {}^{t}\mathbf{u}_{\rm F}, {}^{t}\mathbf{v}_{\rm F}, {}^{t}\mathbf{a}_{\rm F}, {}^{t}\dot{\boldsymbol{\varepsilon}}_{\rm F}, {}^{t}\boldsymbol{\sigma}_{\rm F}, \ldots]^{\rm T}, \quad (2)$$

where $\mathbf{x}$ is the nodal coordinate vector, $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{a}$ are the vector of displacements, velocities and accelerations, respectively, $\boldsymbol{\varepsilon}$, $\dot{\boldsymbol{\varepsilon}}$ and $\boldsymbol{\sigma}$ are the strain vector, the strain rate (or rate of deformation) vectors and the Cauchy stress vector, respectively and F and S denote the variables in the fluid and solid domains, respectively. In the discretized problem, a bar over these variables will denote nodal values.

The coupled fluid–structure interaction (FSI) problem of Fig. 3 is solved, in this work, using the following *strongly coupled staggered scheme*:

1. We assume that the variables in the solid and fluid domains at time t (${}^{t}S$ and ${}^{t}F$) are known.
2. Solve for the variables at the solid domain at time $t+\Delta t$ (${}^{t+\Delta t}S$) under prescribed surface tractions at the fluid–solid boundary $\Gamma_{\rm FS}$. The boundary conditions at the part of the external boundary intersecting the domain are the standard ones in solid mechanics.
3. Solve for the variables at the fluid domain at time $t+\Delta t$ (${}^{t+\Delta t}F$) under prescribed surface tractions at the external boundary $\Gamma_{\rm t}$ and prescribed velocities at the external and internal boundaries $\Gamma_{\rm V}$ and $\Gamma_{\rm FS}$, respectively.Iterate between 1 and 2 until convergence.

The variables at the solid domain ${}^{t+\Delta t}S$ are found via the integration of the equations of dynamic motion in the solid written as

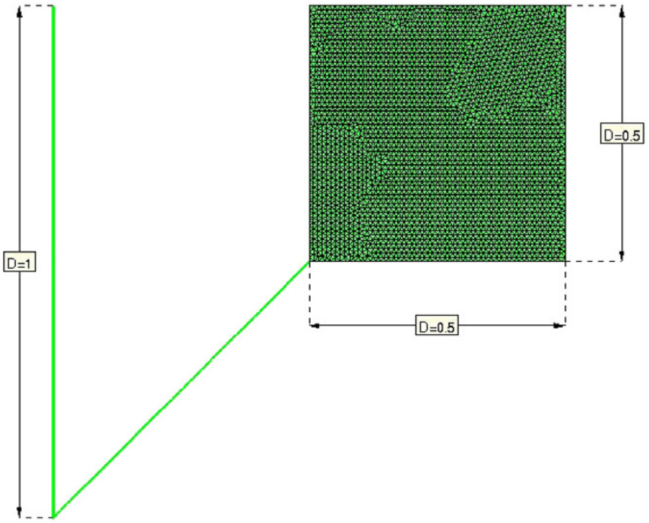
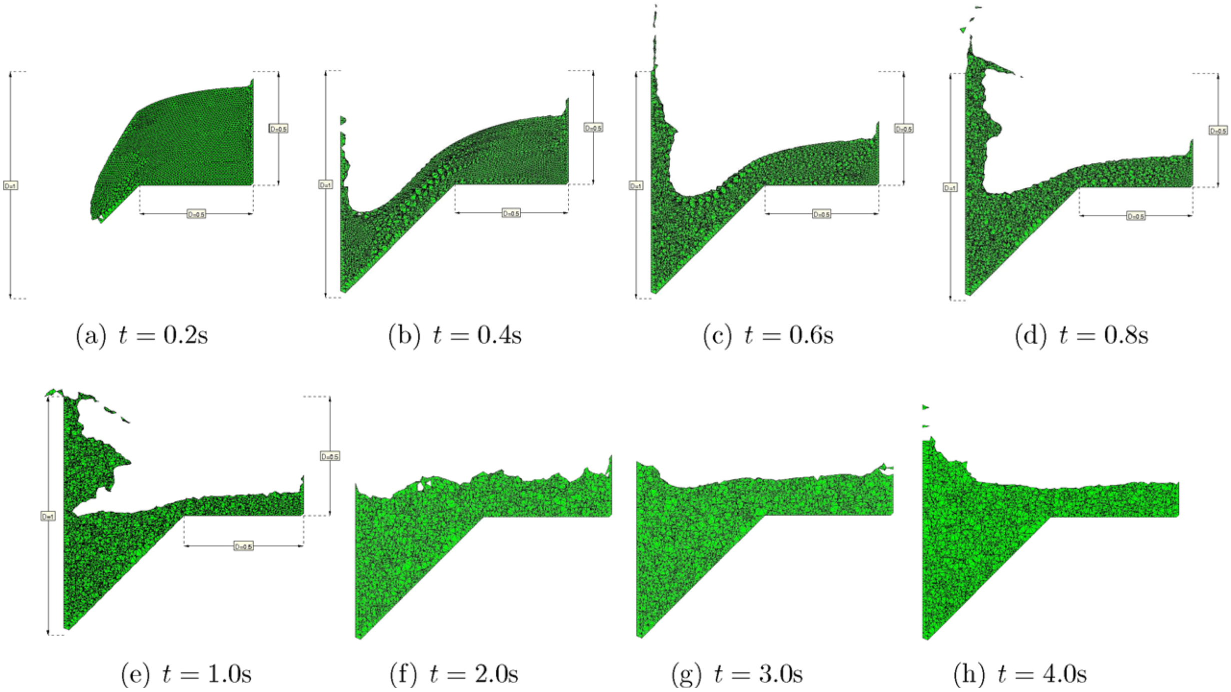


Fig. 8. Fluid domain following into a recipient. Initial position. Fine mesh of 3105 nodes (element size of 0.01 m).



(a) $t = 0.2$s (b) $t = 0.4$s (c) $t = 0.6$s (d) $t = 0.8$s

(e) $t = 1.0$s (f) $t = 2.0$s (g) $t = 3.0$s (h) $t = 4.0$s

Fig. 9. Positions of the fluid domain at different time steps.

$$\mathbf{M}_s\mathbf{a}_s + \mathbf{g}_s - \mathbf{f}_s = \mathbf{0}, \tag{3}$$

where $\mathbf{M}_s$, $\mathbf{g}_s$ and $\mathbf{f}_s$ denote the mass matrix, the internal node force vector and the external nodal force vector at the solid domain. The time integration of Eq. (3) is performed using a standard Newmark method. An incremental iterative scheme is implemented within each time step to account for non linear geometrical and material effects [35]..

The FEM solution of the variables in the (incompressible) fluid domain implies solving the momentum and incompressibility equations. As mentioned above this is not such as simple problem as the incompressibility condition limits the choice of the FE approximations for the velocity and pressure to overcome the well-known *div*-stability condition [5,34]. In our work we use a stabilized mixed FEM based on the Finite Calculus (FIC) approach which allows for a linear approximation for the velocity and pressure variables. Details of the FEM/FIC formulation are given in the next section.

Fig. 4 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 [12,26]. Fig. 4a shows the initial grid of four node rectangles discretizing the fluid domain and the solid walls. Fig. 4b and c show the mesh for the solution at two later times.

3. FIC/FEM formulation for a Lagrangian incompressible fluid

The standard infinitesimal equations for a viscous incompressible fluid can be written in a Lagrangian frame as [17,34].

Momentum

$$r_{m_i} = 0 \quad \text{in } V_{\mathrm{F}}. \tag{4}$$

Mass balance

$$r_{\mathrm{d}} = 0 \quad \text{in } V_{\mathrm{F}}, \tag{5}$$

where

$$r_{m_i} = \rho\frac{\partial v_i}{\partial t} + \frac{\partial \sigma_{ij}}{\partial x_j} - b_i, \quad \sigma_{ji} = \sigma_{ij}, \tag{6}$$

$$r_{\mathrm{d}} = \frac{\partial v_i}{\partial x_i} \quad i,j = 1, n_{\mathrm{d}}. \tag{7}$$

Above n_{d} is the number of space dimensions, v_i is the velocity along the ith global axis ($v_i = \frac{\partial u_i}{\partial t}$, where u_i is the ith displacement), ρ is the (constant) density of the fluid, b_i are the body forces, σ_{ij} are the total stresses given by

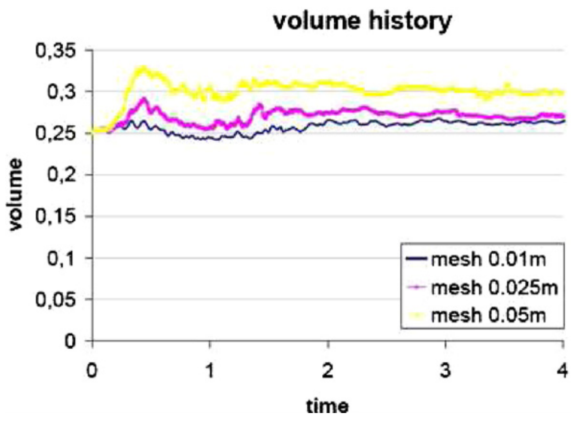


Fig. 10. Total volume change as a function of time for different meshes.

$\sigma_{ij} = s_{ij} - \delta_{ij}p$, 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

$$s_{ij} = 2\mu\left(\dot{\varepsilon}_{ij} - \delta_{ij}\frac{1}{3}\frac{\partial v_k}{\partial x_k}\right), \tag{8}$$

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

$$\dot{\varepsilon}_{ij} = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right). \tag{9}$$

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 [17–19,21].

Momentum

$$r_{m_i} - \underline{\frac{1}{2}h_j\frac{\partial r_{m_i}}{\partial x_j}} = 0 \quad \text{in } V_{\mathrm{F}}. \tag{10}$$

Mass balance

$$r_{\mathrm{d}} - \underline{\frac{1}{2}h_j\frac{\partial r_{\mathrm{d}}}{\partial x_j}} = 0 \quad \text{in } V_{\mathrm{F}}. \tag{11}$$

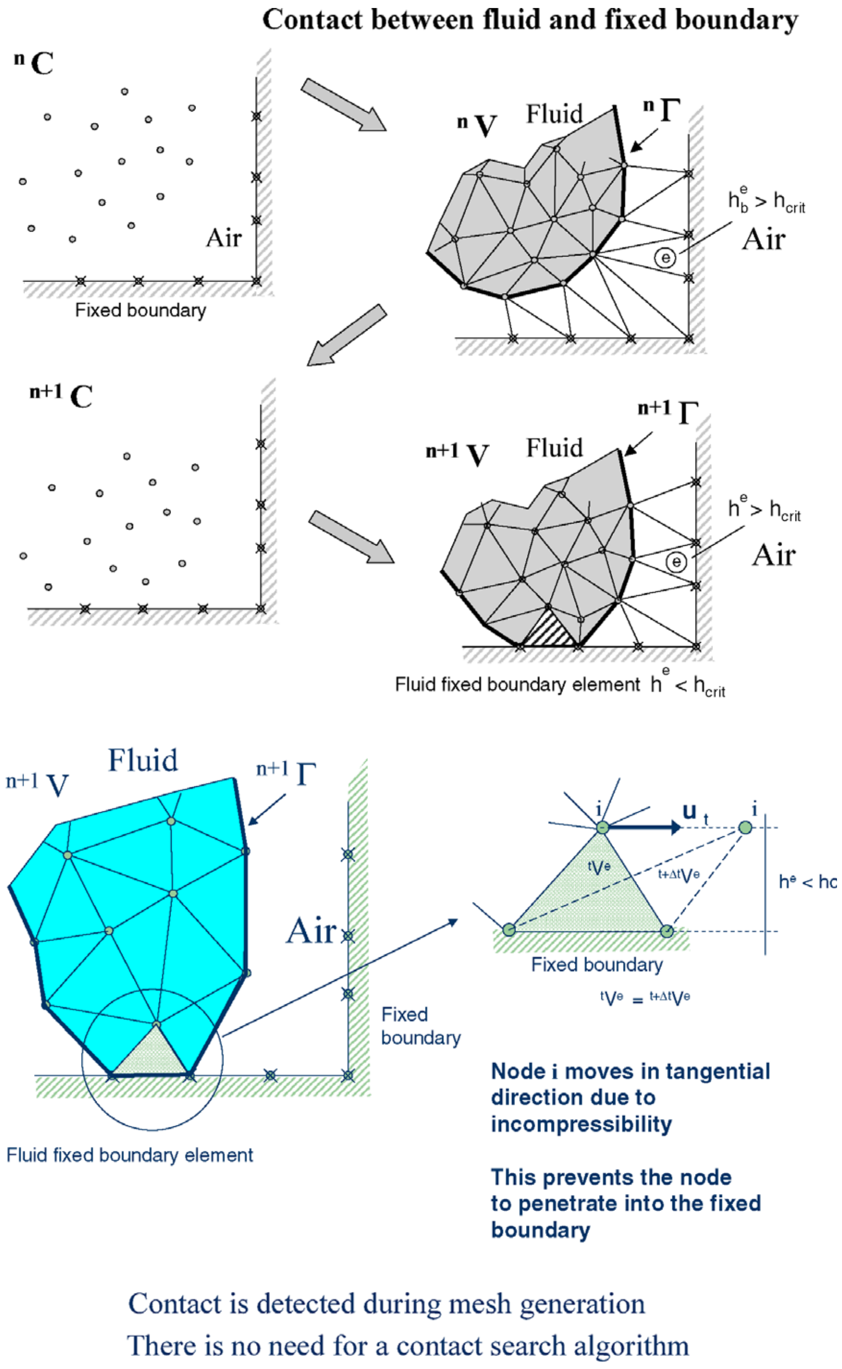

Contact between fluid and fixed boundary
ⁿC
Air
Fixed boundary
ⁿV
Fluid
ⁿΓ
$h_b^e > h_{crit}$
Air
e
n+1 C
n+1 V
Fluid
n+1 Γ
$h^e > h_{crit}$
Air
e
Fluid fixed boundary element $h^e < h_{crit}$
n+1 V
Fluid
n+1 Γ
Air
Fixed boundary
Fluid fixed boundary element
i
u_t
i
$^tV^e$
$^{t+\Delta t}V^e$
$h^e < hc$
Fixed boundary
$^tV^e = {}^{t+\Delta t}V^e$
Node i moves in tangential direction due to incompressibility
This prevents the node to penetrate into the fixed boundary
Contact is detected during mesh generation
There is no need for a contact search algorithm

Fig. 11. Automatic treatment of contact conditions at the fluid–wall interface.

The problem definition is completed with the following boundary conditions:

$$n_j\sigma_{ij} - t_i + \underline{\frac{1}{2}h_j n_j r_{m_i}} = 0 \quad \text{on } \Gamma_t, \tag{12}$$

$$v_j - v_j^p = 0 \quad \text{on } \Gamma_v \tag{13}$$

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. (12) and (13) 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. Recall that Γ_v includes both the external boundary and the internal boundary Γ_{F_S} (Fig. 3).

The h_i's in above equations are *characteristic lengths* of the domain where balance of momentum and mass is enforced. In Eq. (12) these lengths define the domain where equilibrium of boundary tractions is established. We note that at the discretized level, the h_i's become of the order of a typical element or grid dimension [17–19].

Eqs. (10)–(13) 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 [2,8–10,12,24]. Application of the FIC formulation to finite element and meshless analysis of fluid flow problems can be found in [7,18–21,23,25,27].

3.1. Transformation of the mass balance equation. Integral governing equations

The underlined term in Eq. (11) can be expressed in terms of the momentum equations. The new expression for the mass balance equation is [18,26]

$$r_d - \sum_{i=1}^{n_d} \tau_i \frac{\partial r_{m_i}}{\partial x_i} = 0, \tag{14}$$

with

$$\tau_i = \frac{3h_i^2}{8\mu}. \tag{15}$$

In our work we have taken the characteristic distances h_i to be constant within each element and equal to a typical element dimension computed as $h^e = [V^e]^m$ where V^e is the element domain and $m = 1/2$ for 2D problems and $m = 1/3$ for 3D problems.

At this stage it is no longer necessary to retain the stabilization terms in the momentum equations and the traction boundary conditions (Eqs. (10) and (12)). These terms are critical in Eulerian formulations to stabilize the numerical solution for high values of the convective terms [3,18,21,27,28].

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

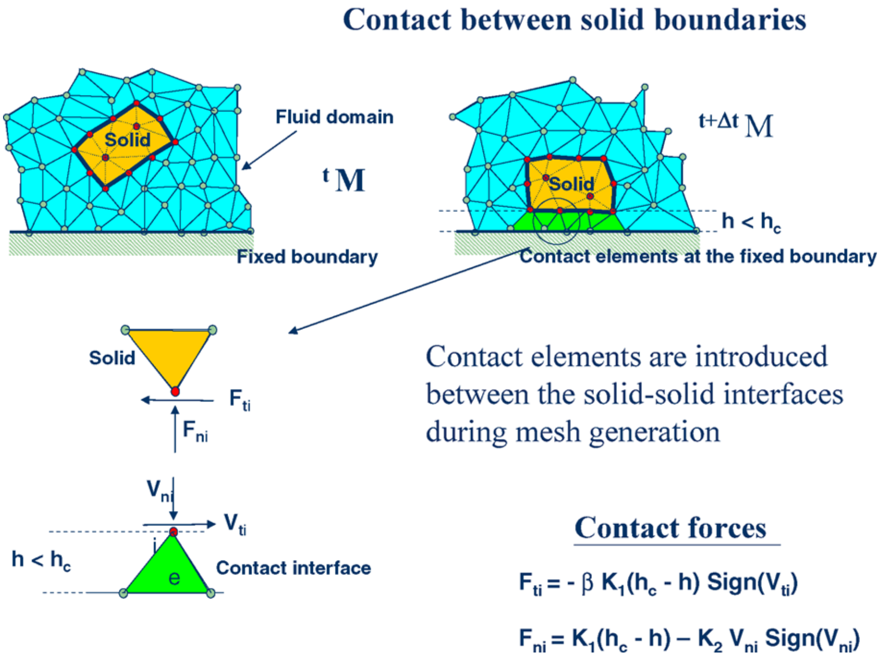

Contact between solid boundaries
Solid
Fluid domain
^{t}M
Fixed boundary
t+Δt M
Solid
h < h$_c$
Contact elements at the fixed boundary
Solid
F$_{ti}$
F$_{ni}$
V$_{ni}$
V$_{ti}$
i
h < h$_c$
e
Contact interface
Contact elements are introduced between the solid-solid interfaces during mesh generation
Contact forces
$F_{ti} = -\beta K_1(h_c - h)\,\mathrm{Sign}(V_{ti})$
$F_{ni} = K_1(h_c - h) - K_2 V_{ni}\,\mathrm{Sign}(V_{ni})$

Fig. 12. Contact conditions at a solid–solid interface.

Please cite this article in press as: E. Oñate et al., Advances in the particle finite element method for the analysis of ..., Comput. Methods Appl. Mech. Engrg. (2007), doi:10.1016/j.cma.2007.06.005

$$\int_{V_{\mathrm{F}}} \delta v_i r_{m_i}\, \mathrm{d}V + \int_{\Gamma_{\mathrm{t}}} \delta v_i (n_j \sigma_{ij} - t_i)\, \mathrm{d}\Gamma = 0, \tag{16}$$

$$\int_{V_{\mathrm{F}}} q \left[r_{\mathrm{d}} - \sum_{i=1}^{n_{\mathrm{d}}} \tau_i \frac{\partial r_{m_i}}{\partial x_i} \right] \mathrm{d}V = 0, \tag{17}$$

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

The r_{m_i} term in Eq. (17) and the deviatoric stresses and the pressure terms within r_{m_i} in Eq. (16) are integrated by parts to give

$$\int_{V_{\mathrm{F}}} \left[\delta v_i \rho \frac{\partial v_i}{\partial t} + \delta \dot{\varepsilon}_{ij} (s_{ij} - \delta_{ij} p) \right] \mathrm{d}V - \int_{V_{\mathrm{F}}} \delta v_i b_i\, \mathrm{d}\Omega - \int_{\Gamma_{\mathrm{t}}} \delta v_i t_i\, \mathrm{d}\Gamma = 0, \tag{18}$$

$$\int_{V_{\mathrm{F}}} q \frac{\partial v_i}{\partial x_i} \mathrm{d}V + \int_{V_{\mathrm{F}}} \left[\sum_{i=1}^{n_{\mathrm{d}}} \tau_i \frac{\partial q}{\partial x_i} r_{m_i} \right] \mathrm{d}V = 0. \tag{19}$$

In Eq. (18) $\delta \dot{\varepsilon}_{ij}$ are virtual strain rates. Note that the boundary term resulting from the integration by parts of r_{m_i} in Eq. (19) has been neglected in this work. Retaining this term has been recently found to be advantageous for enhancing the satisfaction of the incompressibility condition in FEM predictor–corrector schemes for incompressible fluid flow analysis [30].

3.2. Pressure gradient projection

The computation of the residual terms in Eq. (19) is simplified if we introduce the pressure gradient projections π_i, defined as

$$\pi_i = r_{m_i} - \frac{\partial p}{\partial x_i}. \tag{20}$$

We express r_{m_i} in Eq. (19) in terms of the π_i which then become additional variables. The system of integral equations is now 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:

$$\int_{V_{\mathrm{F}}} \left[\delta v_i \rho \frac{\partial v_i}{\partial t} + \delta \dot{\varepsilon}_{ij} (s_{ij} - \delta_{ij} p) \right] \mathrm{d}V - \int_{V_{\mathrm{F}}} \delta v_i b_i\, \mathrm{d}V - \int_{\Gamma_{\mathrm{t}}} \delta v_i t_i\, \mathrm{d}\Gamma = 0, \tag{21}$$

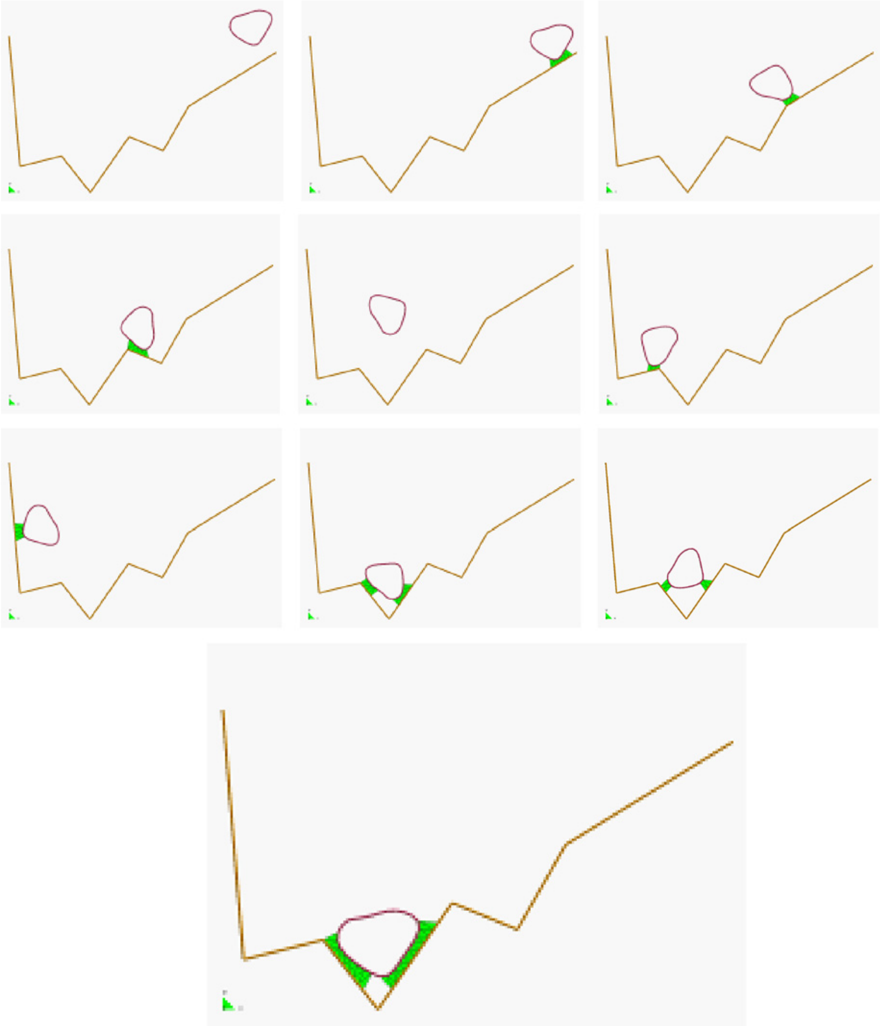


Fig. 13. Bumping of a ball within a container. The layer of contact elements is shown at each contact instant.

Please cite this article in press as: E. Oñate et al., Advances in the particle finite element method for the analysis of ..., Comput. Methods Appl. Mech. Engrg. (2007), doi:10.1016/j.cma.2007.06.005

$$\int_{V_\mathrm{F}} q\frac{\partial v_i}{\partial x_i}\,\mathrm{d}V + \int_{V_\mathrm{F}}\sum_{i=1}^{n_\mathrm{d}}\tau_i\frac{\partial q}{\partial x_i}\left(\frac{\partial p}{\partial x_i}+\pi_i\right)\mathrm{d}V = 0, \tag{22}$$

$$\int_{V_\mathrm{F}} \delta\pi_i\tau_i\left(\frac{\partial p}{\partial x_i}+\pi_i\right)\mathrm{d}V = 0 \quad \text{no sum in } i, \tag{23}$$

with $i,j,k = 1,n_\mathrm{d}$. In Eq. (23) $\delta\pi_i$ are appropriate weighting functions and the τ_i weights are introduced for symmetry reasons.

3.3. 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

$$v_i = \sum_{j=1}^{n} N_j\bar{v}_i^j, \quad p = \sum_{j=1}^{n} N_j\bar{p}^j, \quad \pi_i = \sum_{j=1}^{n} N_j\bar{\pi}_i^j, \tag{24}$$

where $\overline{(\cdot)}^j$ denotes nodal variables and N_j are the shape functions [34]. More details of the mesh discretization process and the choice of shape functions are given in Section 4.

Substituting the approximations (24) into Eqs. (21)–(23) and choosing a Galerkin form with $\delta v_i = q = \delta\pi_i = N_i$ leads to the following system of discretized equations:

$$\mathbf{M}\dot{\bar{\mathbf{v}}} + \mathbf{K}\bar{\mathbf{v}} - \mathbf{G}\bar{\mathbf{p}} - \mathbf{f} = \mathbf{0}, \tag{25a}$$

$$\mathbf{G}^\mathrm{T}\bar{\mathbf{v}} + \mathbf{L}\bar{\mathbf{p}} + \mathbf{Q}\bar{\boldsymbol{\pi}} = \mathbf{0}, \tag{25b}$$

$$\mathbf{Q}^\mathrm{T}\bar{\mathbf{p}} + \hat{\mathbf{M}}\bar{\boldsymbol{\pi}} = \mathbf{0}. \tag{25c}$$

The form of the element matrices and vectors in Eqs. (25) can be found in [28].

3.4. Fractional step algorithm for the fluid variables

The starting point of the iterative algorithm are the variables at time n in the fluid domain (${}^{n}F$). The sought variables are the variables at time $n+1$ (${}^{n+1}F$). For the sake of clarity we will skip the upper left index $n+1$ for all variables, i.e.

$${}^{n+1}\bar{\mathbf{x}} \equiv \bar{\mathbf{x}}; \quad {}^{n+1}\bar{\mathbf{p}} \equiv \bar{\mathbf{p}}; \quad {}^{n+1}\bar{\boldsymbol{\pi}} \equiv \bar{\boldsymbol{\pi}}; \quad {}^{n+1}\bar{\mathbf{x}} \equiv \bar{\mathbf{x}}; \quad \text{etc.} \tag{26}$$

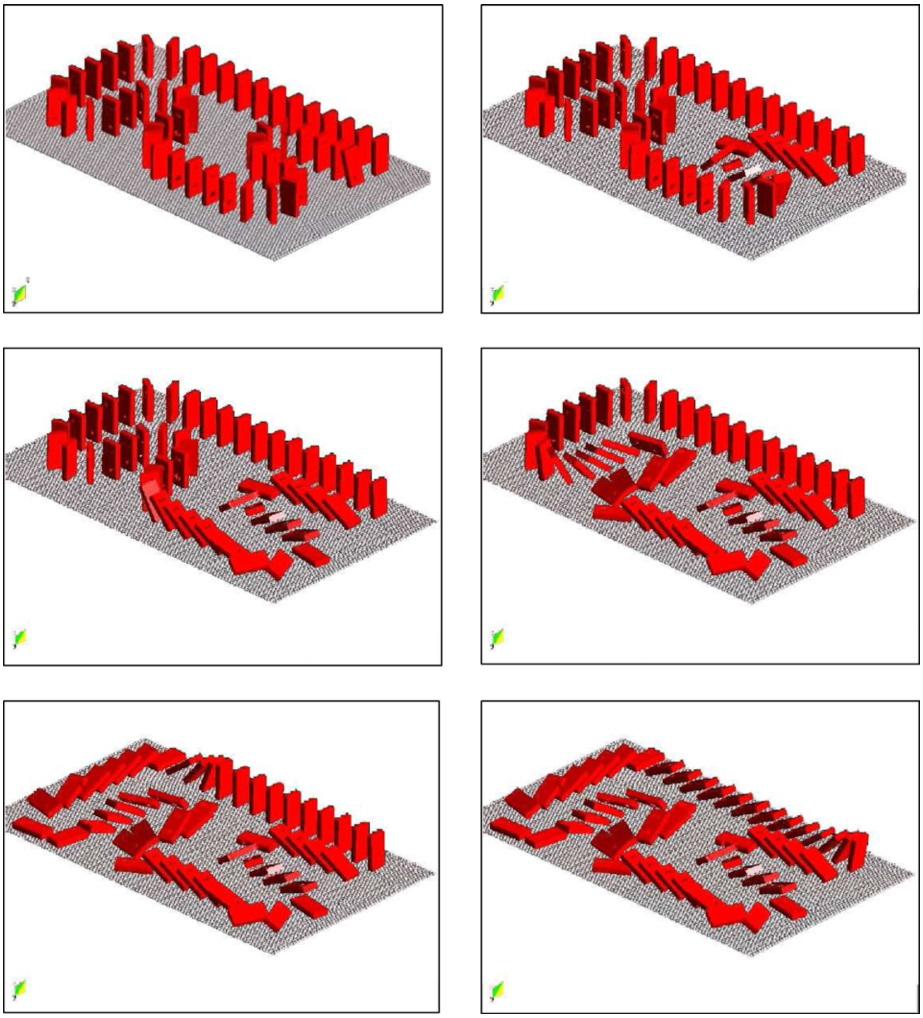


Fig. 14. Failure of a domino set. The distance between the domino chips shows the contact element layer.

A simple iterative algorithm is obtained by splitting the pressure from the momentum equations as follows

$$\bar{\mathbf{v}}^* = {}^n\bar{\mathbf{v}} - \Delta t\mathbf{M}^{-1}[\mathbf{K}\bar{\mathbf{v}}^j - \mathbf{G}^n\mathbf{p} - \mathbf{f}], \tag{27}$$

$$\bar{\mathbf{v}}^{j+1} = \bar{\mathbf{v}}^* + \Delta t\mathbf{M}^{-1}\mathbf{G}\delta\bar{\mathbf{p}}, \tag{28}$$

where $\delta\bar{\mathbf{p}}$ denotes the pressure increment. In above equations and in the following the left upper index n refers to values in the current configuration ${}^nV$, whereas the right upper index j denotes the iteration number within each time step.

The value of $\bar{\mathbf{v}}^{j+1}$ from Eq. (29) is substituted now into Eq. (25b) to give

$$\mathbf{G}^{\mathrm{T}}\bar{\mathbf{v}}^* + \Delta t\mathbf{S}\delta\bar{\mathbf{p}} + \mathbf{L}\bar{\mathbf{p}}^{j+1} + \mathbf{Q}\bar{\boldsymbol{\pi}}^j = \mathbf{0}, \tag{29a}$$

where

$$\mathbf{S} = \mathbf{G}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{G}. \tag{29b}$$

Typically matrix $\mathbf{S}$ is computed using a diagonal matrix $\mathbf{M} = \mathbf{M}_{\mathrm{d}}$, where the subscript 'd' denotes hereonward a diagonal matrix. We note that the form of $\mathbf{S}$ in Eq. (29b) avoids the need for prescribing the pressure at the boundary nodes.

An alternative is to approximate $\mathbf{S}$ 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. These problems are overcome how-

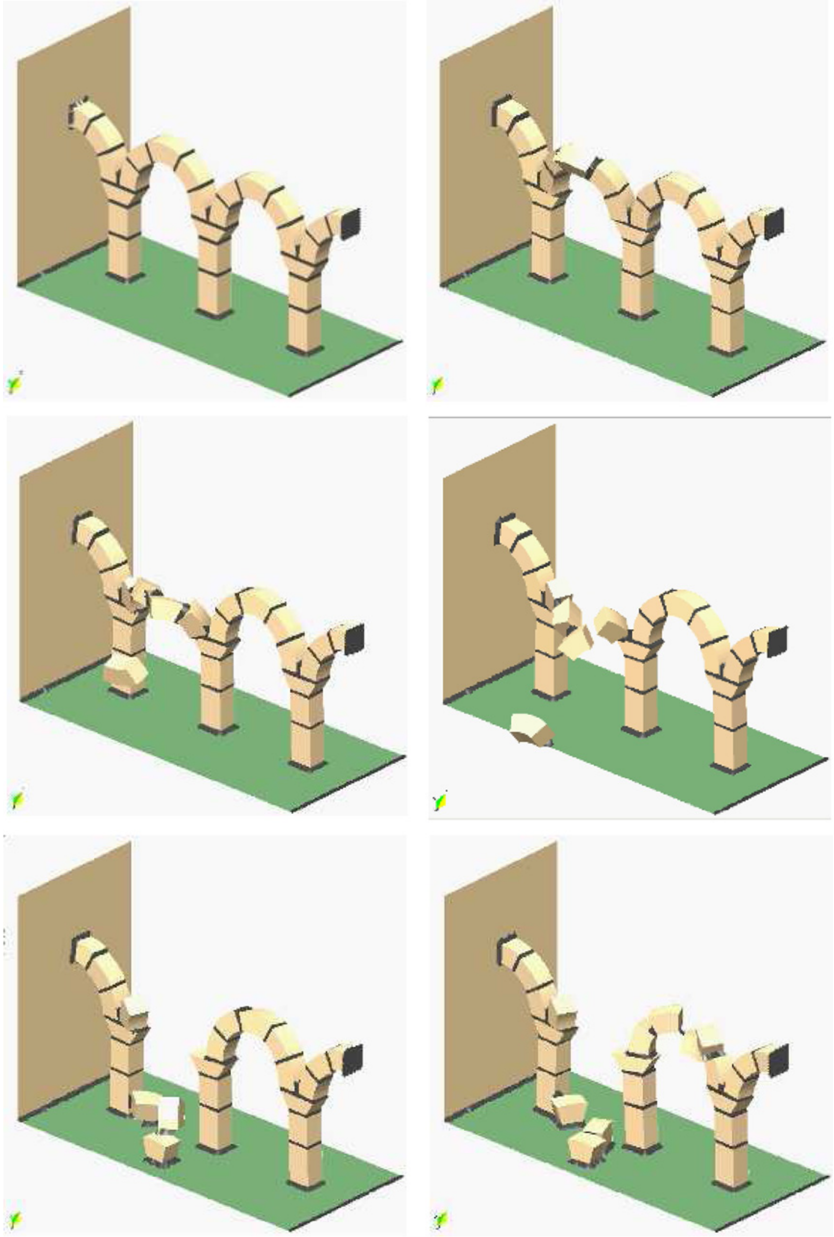


Fig. 15. Failure of an arch formed by stone blocks under seismic loading.

ever by retaining the residual term r_{m_i} in the boundary integral resulting from the integration by parts of Eq. (17) [30]. In this work however the form of matrix $\mathbf{S}$ given by Eq. (29a) has been used.

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

Step 1. Compute $\mathbf{v}^*$ from Eq. (27) with $\mathbf{M} = \mathbf{M}_{\mathrm{d}}$. For the first iteration $(\bar{\mathbf{v}}^1, \bar{\mathbf{p}}^1, \bar{\boldsymbol{\pi}}^1, \bar{\mathbf{x}}^1) \equiv ({}^n\bar{\mathbf{v}}, {}^n\bar{\mathbf{p}}, {}^n\bar{\boldsymbol{\pi}}, {}^n\bar{\mathbf{x}})$.

Step 2. Compute $\delta\bar{\mathbf{p}}$ and $\mathbf{p}^{j+1}$ from Eq. (29a) as

$$\delta\bar{\mathbf{p}} = -(\mathbf{L} + \Delta t\mathbf{S})^{-1}[\mathbf{G}^{\mathrm{T}}\bar{\mathbf{v}}^* + \mathbf{Q}\bar{\boldsymbol{\pi}}^j + \mathbf{L}\bar{\mathbf{p}}^j]. \tag{30a}$$

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

$$\bar{\mathbf{p}}^{j+1} = \bar{\mathbf{p}}^j + \delta\bar{\mathbf{p}}^j. \tag{30b}$$

Step 3. Compute $\bar{\mathbf{v}}^{j+1}$ from Eq. (28) with $\mathbf{M} = \mathbf{M}_{\mathrm{d}}$.

Step 4. Compute $\bar{\boldsymbol{\pi}}^{j+1}$ from Eq. (25c) as

$$\bar{\boldsymbol{\pi}}^{j+1} = -\hat{\mathbf{M}}_{\mathrm{d}}^{-1}\mathbf{Q}^{\mathrm{T}}\bar{\mathbf{p}}^{j+1}. \tag{31}$$

Step 5. Update the coordinates of the mesh nodes as

$$\mathbf{x}_i^{j+1} = {}^n\mathbf{x}_i + \bar{\mathbf{v}}_i^{j+1}\Delta t. \tag{32}$$

Step 6. Check the convergence of the velocity and pressure fields. If convergence is achieved move to the next time step, otherwise return to step 1 for the next iteration with $j \leftarrow j + 1$.

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

$$\Delta t = \min(\Delta t_i) \quad \text{with } \Delta t_i = \frac{h_i^{\min}}{|\mathbf{v}|}, \tag{33}$$

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 Eq. (25) are computed at the updated configuration ${}^{n+1}V_{\mathrm{F}}$. This means that the integration domain changes for each iteration and, hence, all the terms involving space derivatives must be updated at each iteration.

The boundary conditions are applied as follows. No condition is applied for the computation of the fractional velocities $\mathbf{v}^*$ in Eq. (27). The prescribed velocities at the boundary are applied when solving for $\bar{\mathbf{v}}^{j+1}$ in step 3. As mentioned earlier there is no need for prescribing the pressure at the boundary if the form of Eq. (29b) is chosen for $\mathbf{S}$.

Box 1 shows a summary of the staggered scheme used for the solution for the variables in the solid and fluid domain at the updated configuration $({}^{n+1}F, {}^{n+1}S)$.

```
LOOP OVER TIME STEPS $n = 1,\ldots n_{\mathrm{time}}$
                $^{n}S,\ ^{n}F$
  LOOP OVER STAGGERED SOLUTION $j = 1,\ldots n_{\mathrm{stag}}$
      Solve for solid variables (prescribed tractions at $^{n+1}\Gamma_{FS}$)
    LOOP OVER ITERATIONS $i = 1,\ldots n_{\mathrm{iter}}$
      Solve for $^{n+1}S_j^i$
      Integrate Eq.(3) using a Newmark scheme
      Check convergence. Yes: solve for fluid variables
    NO: Next iteration $i \leftarrow i+1$

      Solve for fluid variables (prescribed velocities at $^{n+1}\Gamma_{FS}$)
    LOOP OVER ITERATIONS $i = 1,\ldots n_{\mathrm{iter}}$
      Solve for $^{n+1}F_j^i$ using the scheme of Section 3.4
      Check convergence. Yes: go to C
    Next iteration $i \leftarrow i+1$

    C Check convergence of surface tractions at $^{n+1}\Gamma_{FS}$
      Yes: Next time step
    Next staggered solution $j \leftarrow j+1,\ i \leftarrow i+1$

  Next time step $^{n+1}S \leftarrow {}^{n+1}S_j^i,\ {}^{n+1}F \leftarrow {}^{n+1}F_j^i$
```

Box 1. Staggered scheme for the FSI problem (see also Fig. 3).

4. Generation of a new mesh

One of the key points for the success of the PFEM is the fast regeneration of a mesh at every time step on the basis of the position of the nodes in the space domain. Indeed, any fast meshing algorithm can be used for this purpose. In our work the mesh is generated at each time step using the so-called extended Delaunay tesselation (EDT) presented in [9,11,12]. The EDT allows one to generate non standard meshes combining elements of arbitrary polyhedrical 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 (Fig. 5). The C^0 continuous shape functions of the elements can be simply obtained using the so called meshless finite element interpolation (MFEM). In our work the simpler linear C^0 interpolation has been chosen. Details of the mesh generation procedure and the derivation of the linear MFEM shape functions can be found in [9,11,12].

Fig. 6 shows the evolution of the CPU time required for generating the mesh, for solving the system of equations and for assembling such a system in terms of the number of nodes. the numbers correspond to the solution of a 3D flow in an open channel with the PFEM. The figure shows the CPU time in seconds for each time step of the algorithm of Section 3.4. It is clearly seen that the CPU time required for meshing grows linearly with the number of nodes, as expected. Note also that the CPU time for solving the equations exceeds that required for meshing as the

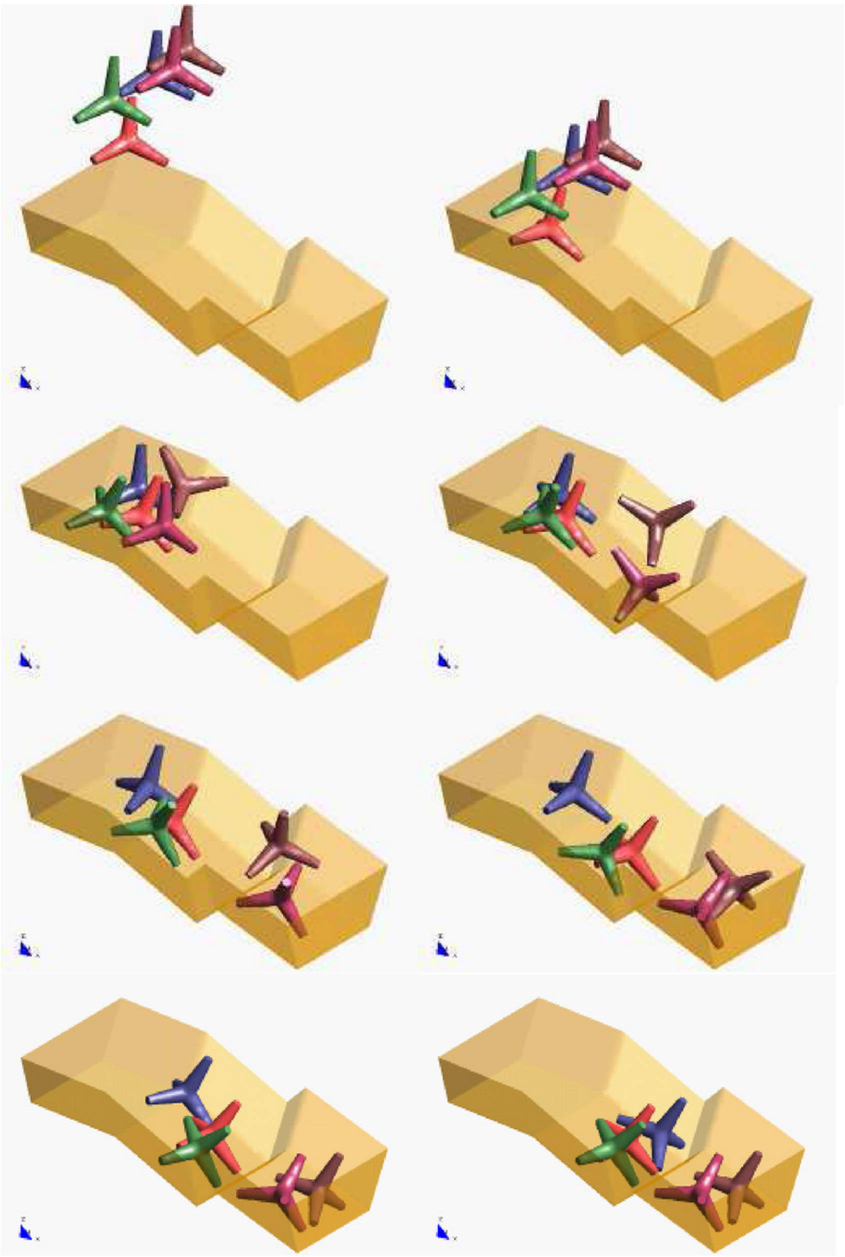


Fig. 16. Motion of five tetrapods on an inclined plane.

number of nodes increases. This situation has been found in all the problems solved with the PFEM. As a general rule for large 3D problems meshing consumes around 30% of the total CPU time for each time step, while the solution of the equations and the assembling of the system consume approximately 40% and 20% of the CPU time for each time step, respectively. These figures prove that the generation of the mesh has an acceptable cost in the PFEM solution. An improvement of the mesh generation process will in any case help to reducing the computational cost.

5. Identification of boundary surfaces

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

In our work we use an extended Delaunay partition for recognizing 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 crite-

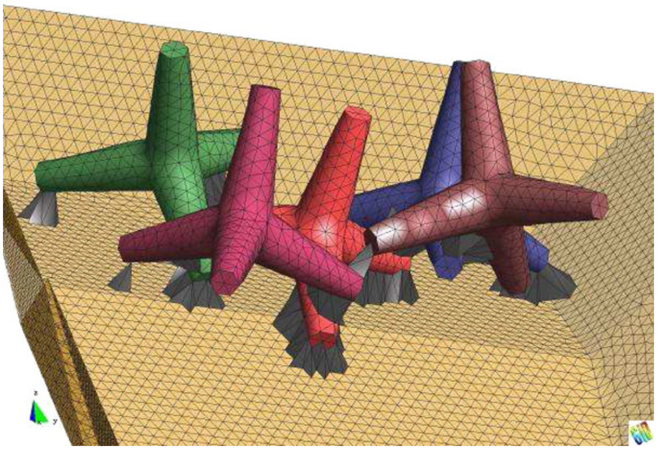


Fig. 17. Detail of five tetrapods on an inclined plane. The layer of elements modeling the frictional contact conditions is shown.

rion 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. Values of α ranging between 1.3 and 1.5 have been found to be optimal in all examples analyzed. This criterion is coincident with the Alpha Shape concept [6]. Fig. 7 shows an 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 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. We recall that each particle is a material point characterized by the density of the solid or fluid domain to which it belongs. The mass which is lost when a boundary element is eliminated due to departure of a node (a particle) from the main analysis domain is again regained when the "flying" node falls down and a new boundary element is created by the Alpha Shape algorithm (Figs. 2 and 7).

The boundary recognition method above described is also useful for detecting contact conditions between the fluid domain and a fixed boundary, as well as between different solids interacting with each other. The contact detection procedure is detailed in Section 6.

In order to show the quality of the boundary recognition approach, the following simple example has been performed. A square fluid domain of 0.25 m^2 is at a stationary position within a recipient (Fig. 8). Then, as time evolves, the fluid falls down into the lower part of the recipient due to gravity effects. At the end of the process the total volume of the fluid within the recipient must be the same as that of the initial square domain. It must be noted that during the different time steps, the fluid has completely different free surfaces including waves, breaking waves and fluid fragmentation zones.

The meshes used have average element sizes of 0.05 m, 0.025 m and 0.01 m each which correspond to a total initial number of particles of 161, 552 and 3105 each. A value of $\alpha = 1.4$ for the Alpha-Shape method was used for the three

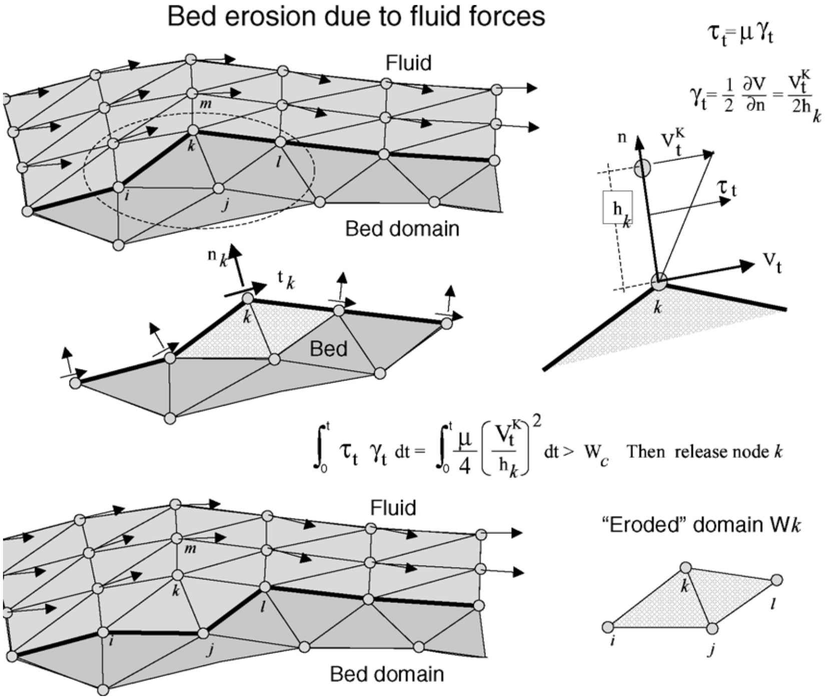

Bed erosion due to fluid forces
$\tau_t = \mu \gamma_t$
Fluid
$\gamma_t = \frac{1}{2}\frac{\partial V}{\partial n} = \frac{V_t^K}{2h_k}$
m
k
l
i
j
Bed domain
n
V_t^K
τ_t
h_k
V_t
k
n_k
t_k
k
Bed
$\int_0^t \tau_t\ \gamma_t\ dt = \int_0^t \frac{\mu}{4}\left(\frac{V_t^K}{h_k}\right)^2 dt > W_c$ Then release node k
Fluid
"Eroded" domain Wk
m
k
l
i
j
Bed domain
k
i
j
l

Fig. 18. Modeling of bed erosion by dragging of bed material.

Please cite this article in press as: E. Oñate et al., Advances in the particle finite element method for the analysis of ..., Comput. Methods Appl. Mech. Engrg. (2007), doi:10.1016/j.cma.2007.06.005

analyses. Fig. 8 shows the initial position of the fluid domain and one of the three meshes used for the analysis. Fig. 9 shows the fluid domain at different time steps.

This simple example is interesting to show the quality of the boundary identification procedure. Another aim is to evaluate the volume variation from the incompressibility point of view, as well as the preservation of the total volume of the fluid due to possible errors in the boundary recognition using the Alpha-Shape method. Fig. 10 shows the total fluid volume during the different time steps for the three different meshes. The change of volume is insignificant for the fine mesh and becomes larger but acceptable for the coarse meshes.

It must be noted that the main difference between the PFEM and the classical FEM is just the remeshing technique and the evaluation of the boundary position at each time step. The rest of the steps in the computation are coincident with those of the classical FEM. This simple example shows that in spite of the permanent remeshing and the evaluation of the boundary position via the Alpha-Shape method, the total fluid mass is preserved. We note however, that a good selection of the α parameter is essential for the good behaviour of the boundary recognition process. Examples showing the accuracy of the PFEM for fixed boundary problems can be found in [2].

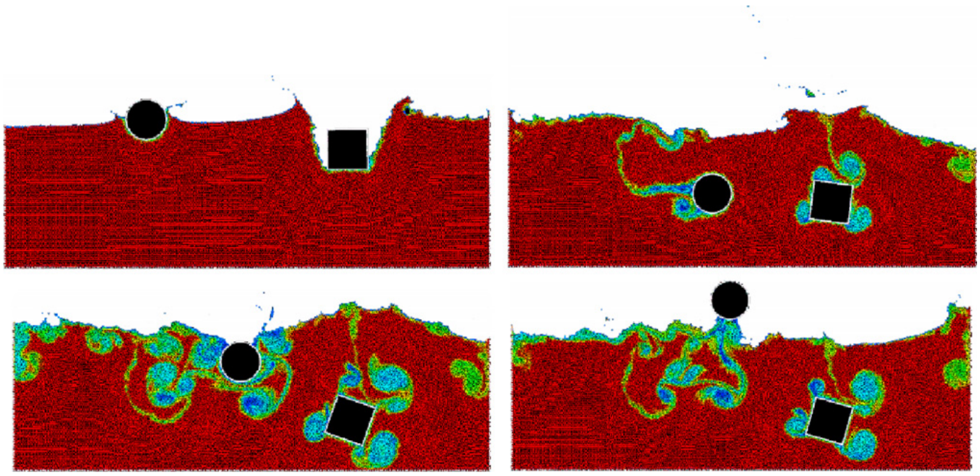


Fig. 19. 2D simulation of the penetration and evolution of a cube and a cylinder in a water container. The colours denote the different sizes of the elements at several times. (For interpretation of the references in colour in this figure legend, the reader is referred to the web version of this article.)

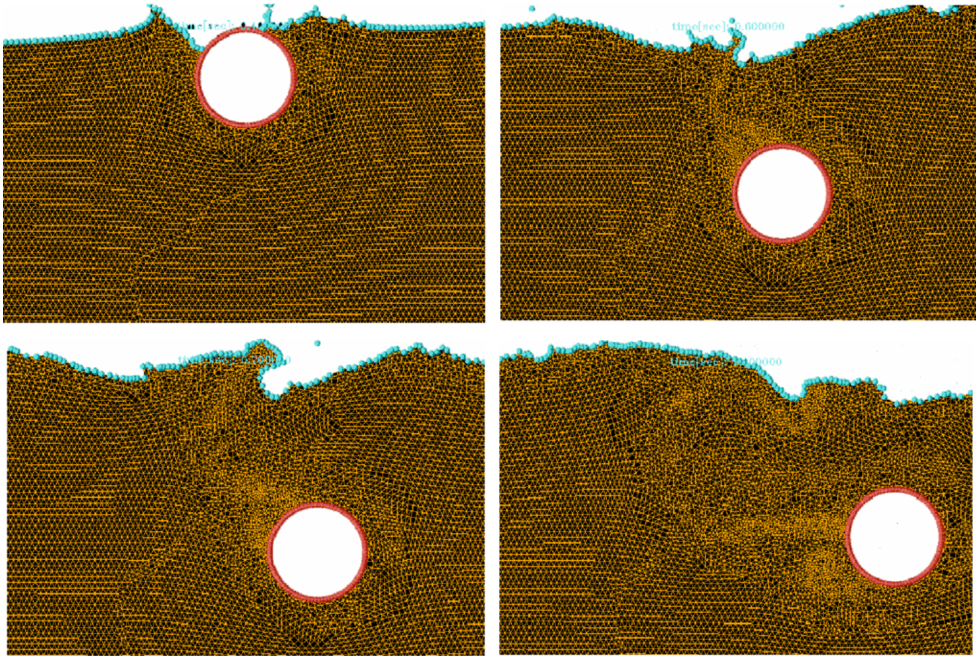


Fig. 20. Detail of element sizes during the motion of a rigid cylinder within a water container.

Please cite this article in press as: E. Oñate et al., Advances in the particle finite element method for the analysis of ..., Comput. Methods Appl. Mech. Engrg. (2007), doi:10.1016/j.cma.2007.06.005

6. Treatment of contact conditions in the PFEM

6.1. Contact between the fluid and a fixed boundary

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 common boundary Γ_{FS}, as mentioned above.

The condition of prescribed velocities at the fixed 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 the fluid particles and the fixed boundaries is accounted for by the incompressibility condition which *naturally prevents the fluid nodes to penetrate into the solid boundaries* (Fig. 11). This simple way to treat the fluid–wall contact at mesh generation level is a distinct and attractive feature of the PFEM formulation.

6.2. Contact between solid–solid interfaces

The contact between two solid interfaces is simply treated by introducing a layer of *contact elements* between the two interacting solid interfaces. This layer is *automatically created during the mesh generation step* by prescribing a minimum distance (h_c) between two solid boundaries. If the distance exceeds the minimum value (h_c) then the generated elements are treated as fluid elements. Otherwise the elements are treated as contact elements where a relationship between the tangential and normal forces and the corresponding displacement is introduced so as to model elastic and frictional contact effects in the normal and tangential directions, respectively (Fig. 12).

This algorithm has proven to be very effective and it allows to identifying and modeling complex frictional contact conditions between two or more interacting bodies moving in water in an extremely simple manner. Of course the accuracy of this contact model depends on the critical distance above mentioned.

This contact algorithm can also be used effectively to model frictional contact conditions between rigid or elastic solids in standard structural mechanics applications. Figs. 13–16 show examples of application of the contact algorithm to the bumping of a ball falling in a container, the failure of a domino set, the failure of an arch formed by a collection of stone blocks under a seismic loading and the motion of five tetrapods as they fall and slip over an

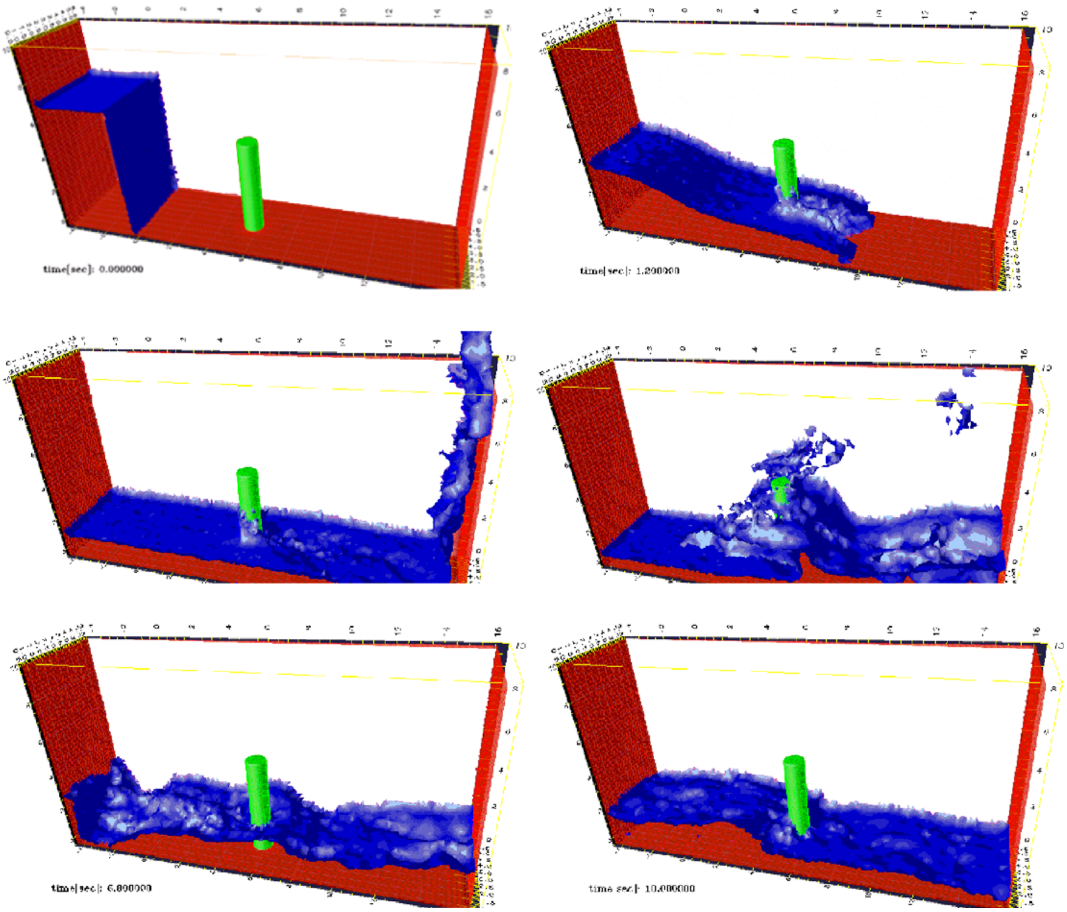


Fig. 21. Evolution of a water column within a prismatic container including a vertical cylinder.

inclined plane, respectively. The images in Figs. 13 and 17 show explicitly the layer of contact elements which controls the accuracy of the contact algorithm.

7. Modeling of bed erosion

Prediction of bed erosion and sediment transport in open channel flows are important tasks in many areas of river and environmental engineering. Bed erosion can lead to instabilities of the river basin slopes. It can also undermine the foundation of bridge piles thereby favouring structural failure. Modeling of bed erosion is also relevant for predicting the evolution of surface material dragged in earth dams in overspill situations. Bed erosion is one of the main causes of environmental damage in floods.

Bed erosion models are traditionally based on a relationship between the rate of erosion and the shear stress level [14,33]. The effect of water velocity on soil erosion was studied in [31]. In a recent work we have proposed an extension of the PFEM to model bed erosion [29]. The erosion model is based on the frictional work at the bed surface originated by the shear stresses in the fluid. The resulting erosion model resembles Archard law typically used for modeling abrasive wear in surfaces under frictional contact conditions [1,22].

The algorithm for modeling the erosion of soil/rock particles at the fluid bed is the following:

1. Compute at every point of the bed surface the resultant tangential stress τ induced by the fluid motion. In 3D problems $\tau = (\tau_s^2 + \tau_t)^2$ where τ_s and τ_t are the tangential stresses in the plane defined by the normal direction $\mathbf{n}$ at the bed node. The value of τ for 2D problems can be estimated as follows:

$$\tau_t = \mu\gamma_t, \tag{34a}$$

with

$$\gamma_t = \frac{1}{2}\frac{\partial v_t}{\partial n} = \frac{v_t^k}{2h_k}, \tag{34b}$$

where v_t^k is the modulus of the tangential velocity at the node k and h_k is a prescribed distance along the normal of the bed node k. Typically h_k is of the order of magnitude of the smallest fluid element adjacent to node k (Fig. 18).

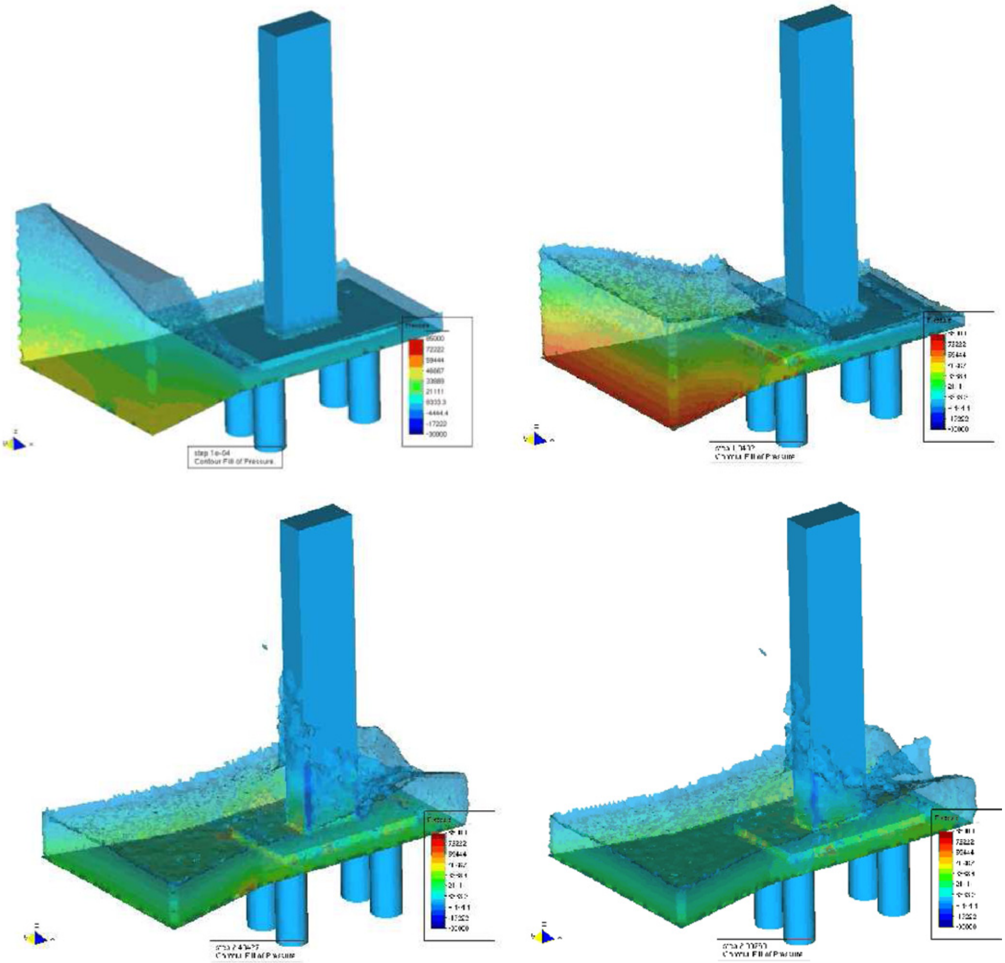


Fig. 22. Impact of a wave on a prismatic column on a slab sustained by four pillars.

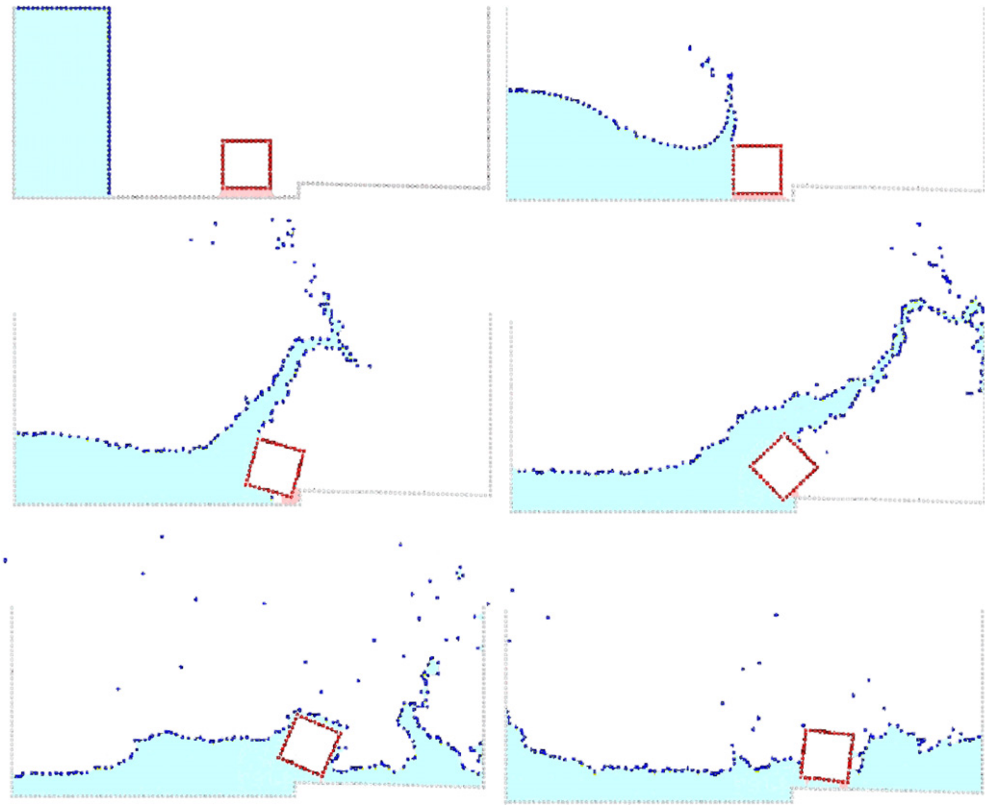


Fig. 23. Dragging of a cubic object by a water stream.

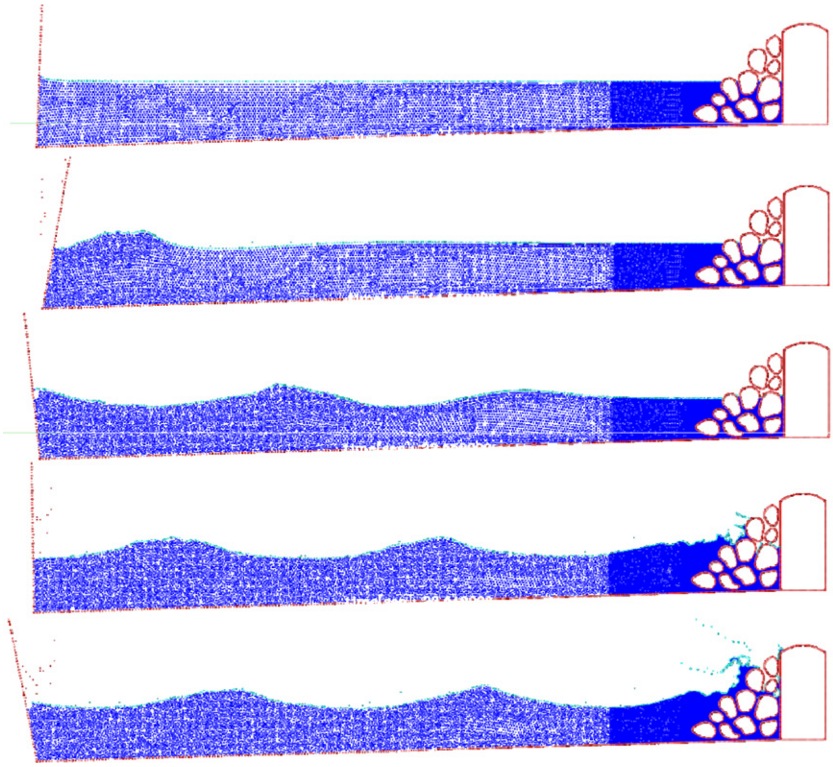


Fig. 24. Generation and impact of a wave on a collection of rocks in a breakwater.

2. Compute the frictional work originated by the tangential stresses at the bed surface as

$$W_{\mathrm{f}} = \int_0^t \tau_{\mathrm{t}}\gamma_{\mathrm{t}}\,\mathrm{d}t = \int_0^t \frac{\mu}{4}\left(\frac{v_{\mathrm{t}}^k}{h_k}\right)^2 \mathrm{d}t. \qquad (35)$$

Eq. (35) is integrated in time using a simple scheme as

$$^{n}W_{\mathrm{f}} = {}^{n-1}W_{\mathrm{f}} + \tau_{\mathrm{t}}\gamma_{\mathrm{t}}\Delta t. \qquad (36)$$

3. The onset of erosion at a bed point occurs when $^{n}W_{\mathrm{f}}$ exceeds a critical threshold value W_{c} defined empirically according to the specific properties of the bed material.

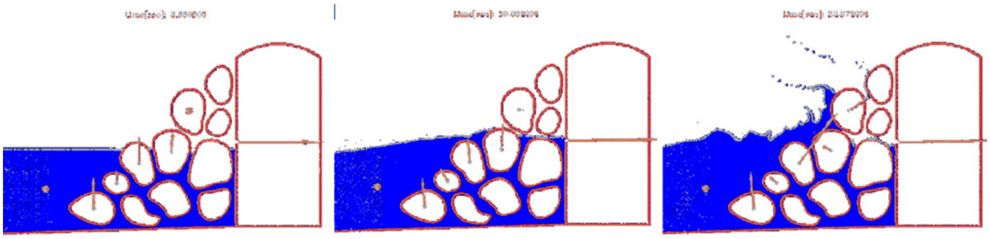


Fig. 25. Detail of the impact of a wave on a breakwater. The arrows indicate the water force on the rocks at different instants.

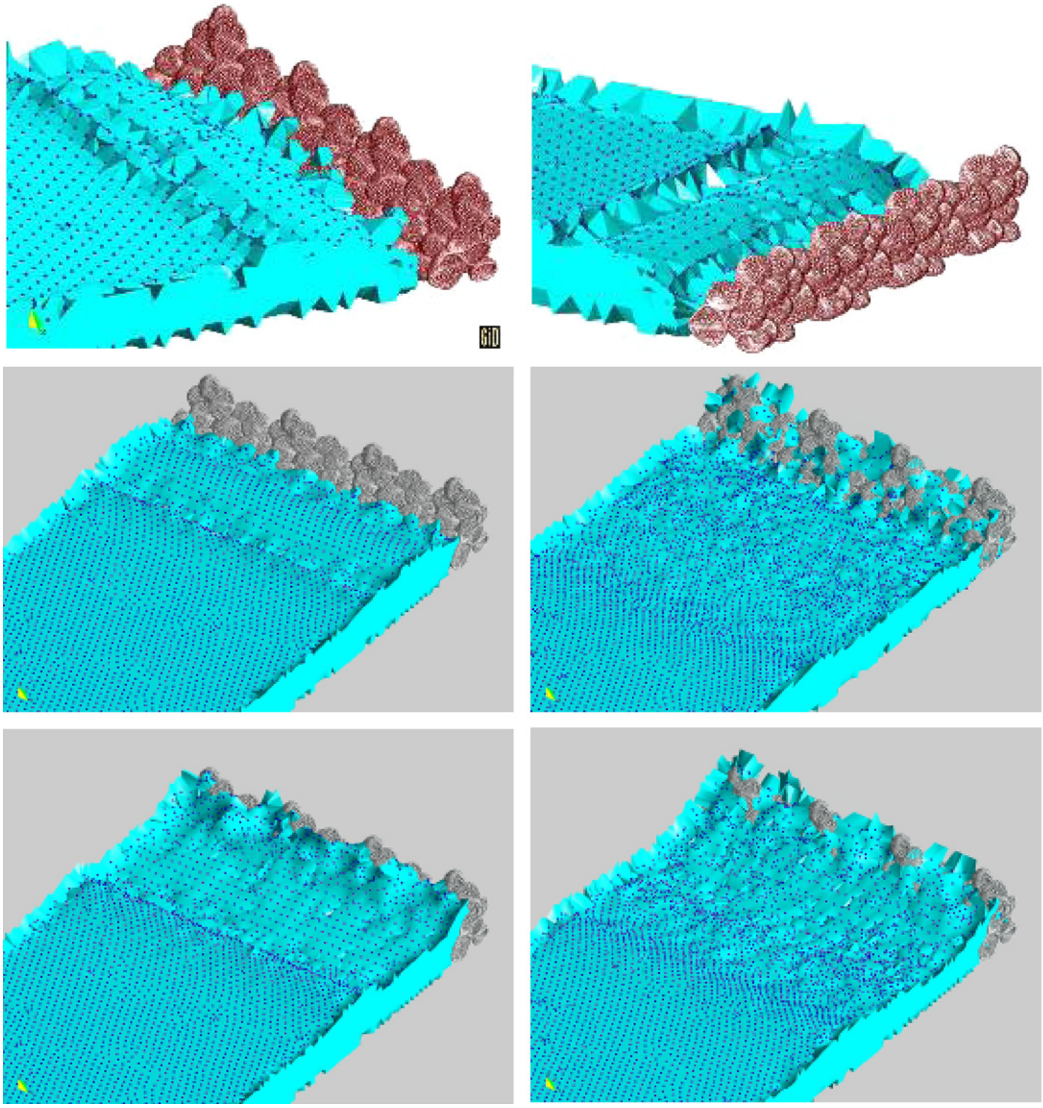


Fig. 26. 3D simulation of the impact of a wave on a collection of rocks in a breakwater.

4. If ${}^{n}W_{\mathrm{f}} > W_{\mathrm{c}}$ at a bed node, then the node is detached from the bed region and it is allowed to move with the fluid flow, i.e. it becomes a fluid node. As a consequence, the mass of the patch of bed elements surrounding the bed node vanishes in the bed domain and it is transferred to the new fluid node. This mass is subsequently

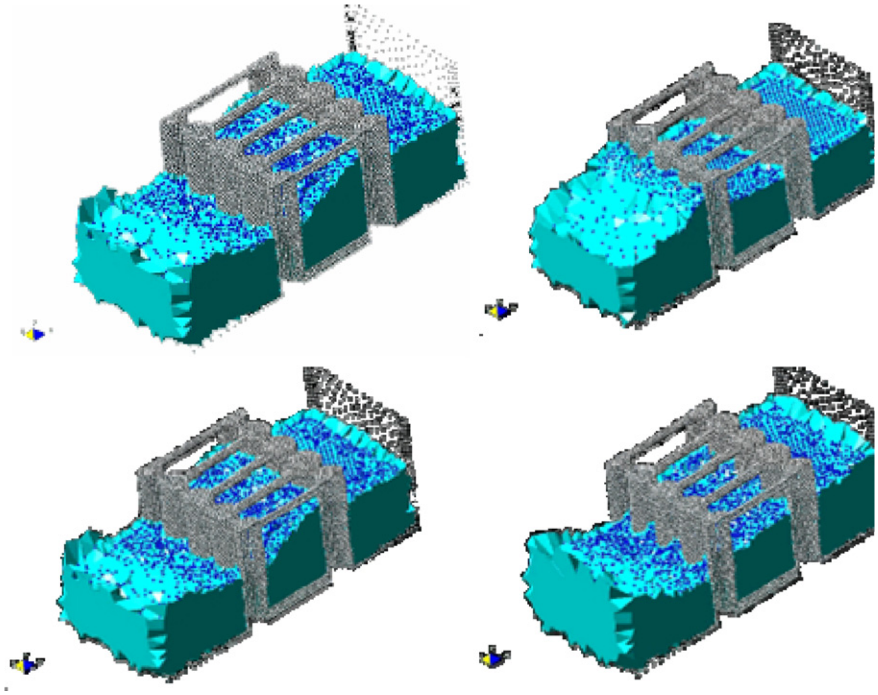


Fig. 27. Interaction of a wave with a vertical pier formed by reinforced concrete cylinders.

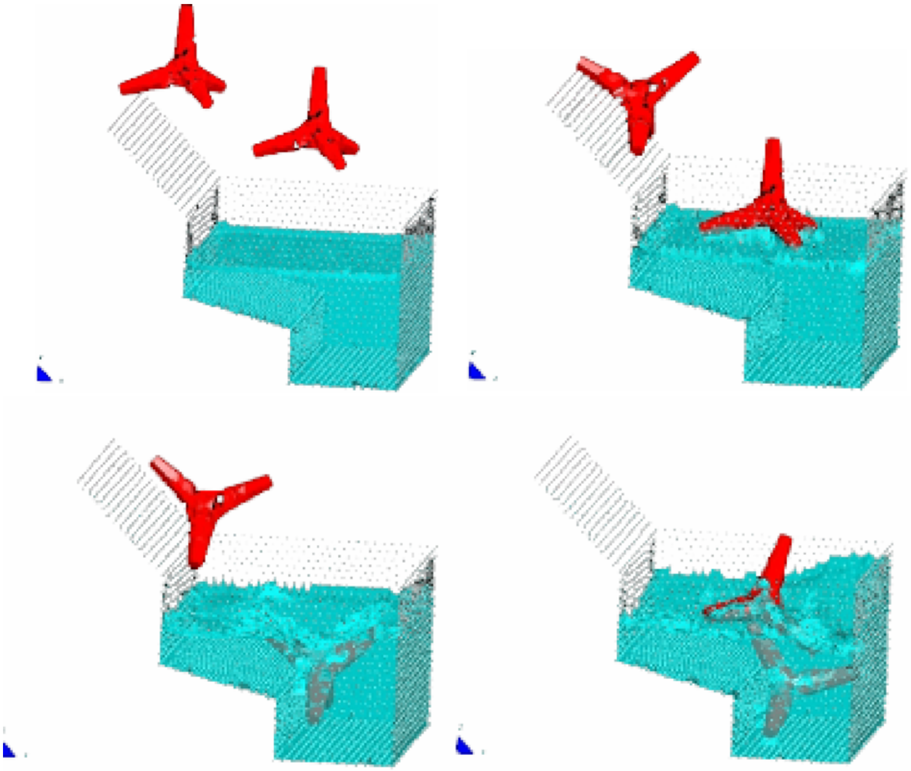


Fig. 28. Motion of two tetrapods falling in a water container.

transported with the fluid. Conservation of mass of the bed particles within the fluid is guaranteed by changing the density of the new fluid node so that the mass of the suspended sediment traveling with the fluid equals the mass originally assigned to the bed node. Recall that the mass assigned to a node is computed by multiplying the node density by the tributary domain of the node.

5. Sediment deposition can be modeled by an inverse process to that described in the previous step. Hence, a suspended node adjacent to the bed surface with a velocity below a threshold value is assigned to the bed surface. This automatically leads to the generation of new bed elements adjacent to the boundary of the bed region. The original mass of the bed region is recovered by adjusting the density of the newly generated bed elements.

Fig. 18 shows an schematic view of the bed erosion algorithm proposed.

8. FSI examples

The examples chosen show the applicability of the PFEM to solve problems involving large motions of the free surface, fluid–multibody interactions and bed erosion.

8.1. Rigid objects falling 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.

Fig. 19 shows the penetration and evolution of a cube and a cylinder of rigid shape in a container with water. The colours denote the different sizes of the elements at several times. In order to increase the accuracy of the FSI problem smaller size elements have been generated in the vicinity of the moving bodies during their motion (Fig. 20).

8.2. Impact of water streams on rigid structures

Fig. 21 shows an example of a wave breaking within a prismatic container including a vertical cylinder. Fig. 22 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.

8.3. Dragging of objects by water streams

Fig. 23 shows the effect of a wave impacting on a rigid cube representing a vehicle. This situation is typical in

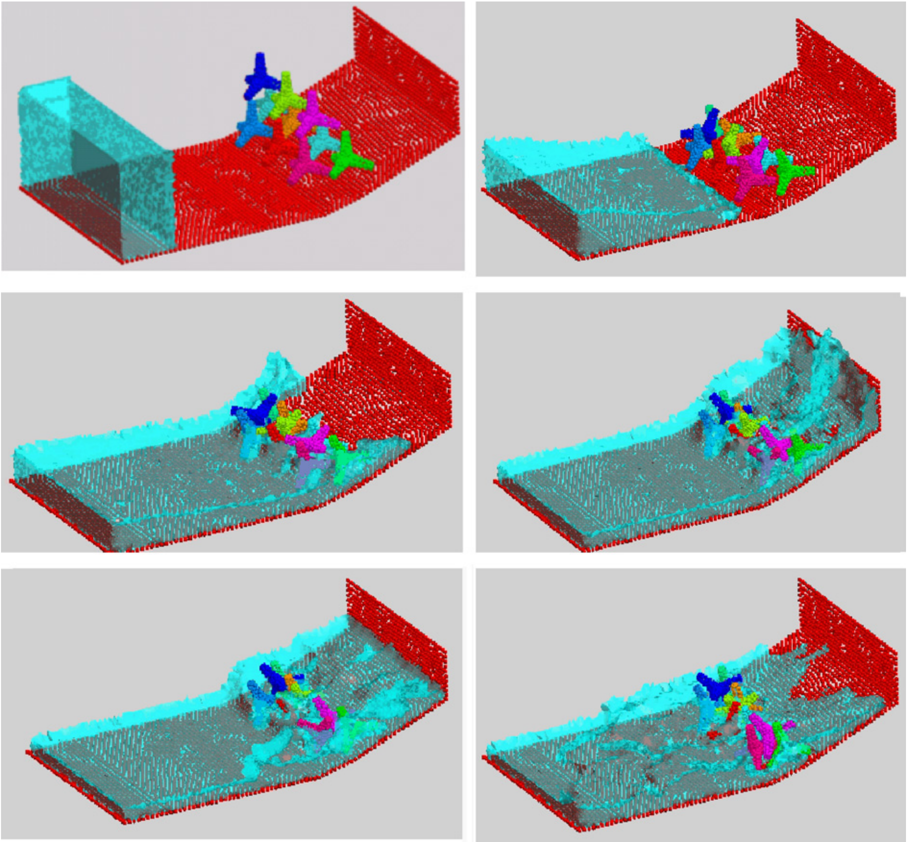


Fig. 29. Motion of 10 tetrapods on a slope under an incident wave.

Please cite this article in press as: E. Oñate et al., Advances in the particle finite element method for the analysis of ..., Comput. Methods Appl. Mech. Engrg. (2007), doi:10.1016/j.cma.2007.06.005

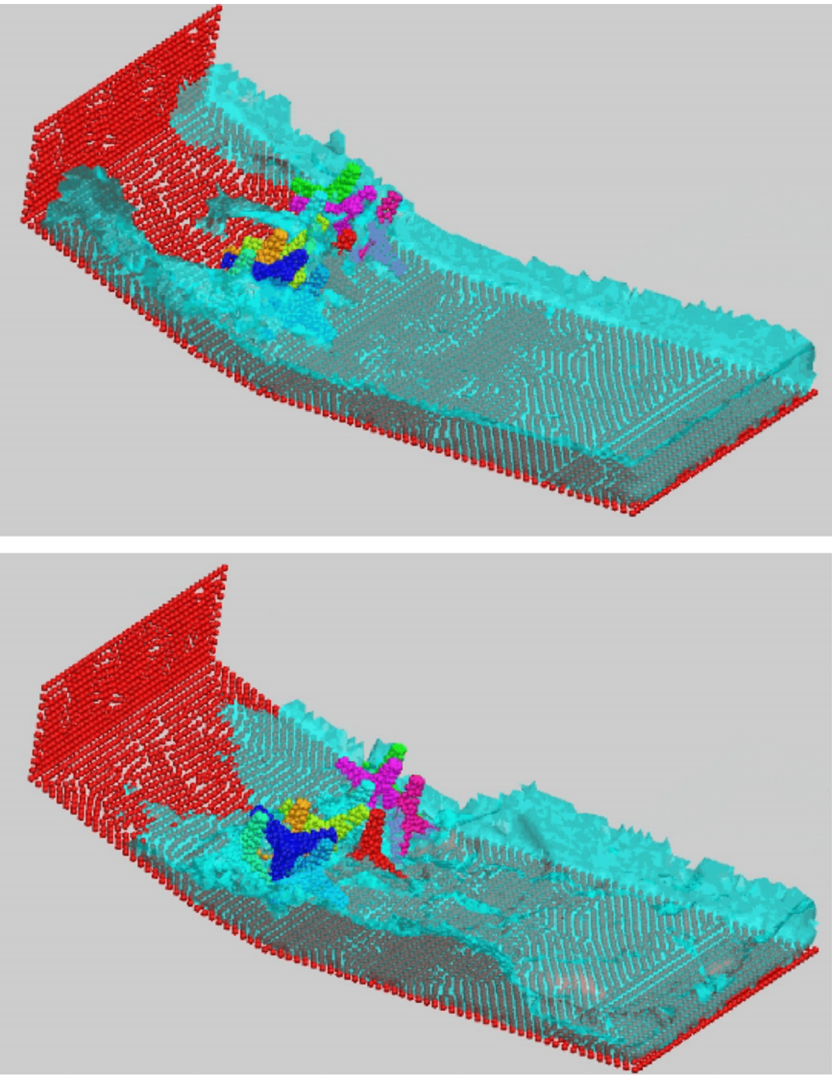


Fig. 30. Detail of the motion of 10 tetrapods on a slope under an incident wave. The figure shows the complex interactions between the water particles and the tetrapods.

flooding and Tsunami situations. Note the layer of contact elements modeling the frictional contact conditions between the cube and the bottom surface.

8.4. Impact of sea waves on breakwaters and piers

Fig. 24 shows the 2D simulation of the impact of a wave generated in an experimental flume on a collection of rigid rocks representing a breakwater. Details of the water–rock interaction are shown in Fig. 25.

Fig. 26 shows a 3D analysis of a similar problem. Fig. 27 shows the 3D simulation of the interaction of a wave with a vertical pier formed by a collection of reinforced concrete cylinders.

The examples shown in Figs. 28 and 29 evidence the potential of the PFEM to solve 3D problems involving complex interactions between water and moving solid objects. Fig. 28 shows the simulation of the falling of two tetrapods in a water container. Fig. 29 shows the motion of a collection of ten tetrapods placed in a slope under an incident wave.

Fig. 30 shows a detail of the complex three-dimensional interactions between the water particles and the tetrapods and between the tetrapods themselves, which can be easily modeled with the PFEM.

8.5. Erosion of a 3D earth dam due to an overspill stream

We present finally a simple, schematic, but very illustrative example showing the potential of the PFEM to model bed erosion in free surface flows.

The example represents the erosion of an earth dam under a water stream running over the dam top. A schematic geometry of the dam has been chosen to simplify the computations. Sediment deposition is not considered in the solution. The images of Fig. 31 show the progressive erosion of the dam until the whole dam is dragged out by the fluid flow.

Please cite this article in press as: E. Oñate et al., Advances in the particle finite element method for the analysis of ..., Comput. Methods Appl. Mech. Engrg. (2007), doi:10.1016/j.cma.2007.06.005

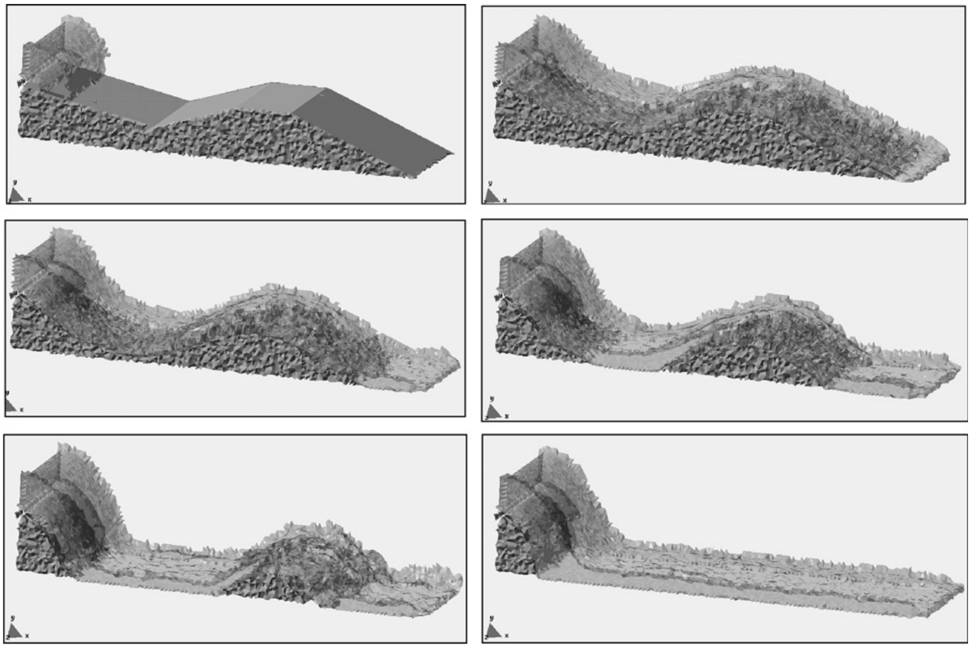


Fig. 31. Erosion of a 3D earth dam due to an overspill stream.

Other applications of the PFEM to bed erosion problems can be found in [29].

9. Conclusions

The particle finite element method (PFEM) is ideal to treat problems involving fluids with free surfaces and submerged or floating structures and bodies within a unified Lagrangian finite element framework. Problems such as fluid–structure interaction, large motion of fluid or solid particles, surface waves, water splashing, separation of water drops, frictional contact situations between fluid–solid and solid–solid interfaces, bed erosion, 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 an updated Lagrangian formulation and 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 tesselation, the identification of the boundary nodes using an Alpha-Shape type technique and the simple algorithm to treat frictional contact conditions at fluid–solid and solid–solid interfaces via mesh generation. The examples presented have shown the great potential of the PFEM for solving a wide class of practical FSI problems in engineering. Examples of validation of the PFEM results with data from experimental tests are reported in [15].

Acknowledgements

Thanks are given to Mrs. M. de Mier for many useful suggestions. This research was partially supported by project SEDUREC of the Consolider Programme of the Ministerio de Educación y Ciencia of Spain.

References

[1] J.F. Archard, Contact and rubbing of flat surfaces, J. Appl. Phys. 24 (8) (1953) 981–988.

[2] R. Aubry, S.R. Idelsohn, E. Oñate, Particle finite element method in fluid mechanics including thermal convection–diffusion, Comput. Struct. 83 (17–18) (2005) 1459–1475.

[3] R. Codina, O.C. Zienkiewicz, CBS versus GLS stabilization of the incompressible Navier–Stokes equations and the role of the time step as stabilization parameter, Commun. Numer. Methods Engrg. 18 (2) (2002) 99–112.

[4] F. Del Pin, S.R. Idelsohn, E. Oñate, R. Aubry, The ALE/Lagrangian particle finite element method: a new approach to computation of free-surface flows and fluid–object interactions, Comput. Fluids 36 (2007) 27–38.

[5] J. Donea, A. Huerta, Finite Element Method for Flow Problems, John Wiley, 2003.

[6] H. Edelsbrunner, E.P. Mucke, Three dimensional alpha shapes, ACM Trans. Graphics 13 (1999) 43–72.

[7] J. García, E. Oñate, An unstructured finite element solver for ship hydrodynamic problems, J. Appl. Mech. 70 (2003) 18–26.

[8] S.R. Idelsohn, E. Oñate, F. Del Pin, N. Calvo, Lagrangian formulation: the only way to solve some free-surface fluid mechanics problems, in: H.A. Mang, F.G. Rammerstorfer, J. Eberhardsteiner (Eds.), Fifth World Congress on Computational Mechanics, July 7–12, Vienna, Austria, 2002.

[9] S.R. Idelsohn, E. Oñate, N. Calvo, F. Del Pin, The meshless finite element method, Int. J. Numer. Methods Engrg. 58 (6) (2003) 893–912.

[10] S.R. Idelsohn, E. Oñate, F. Del Pin, A Lagrangian meshless finite element method applied to fluid–structure interaction problems, Comput. Struct. 81 (2003) 655–671.

[11] S.R. Idelsohn, N. Calvo, E. Oñate, Polyhedrization of an arbitrary point set, Comput. Meth. Appl. Mech. Engrg. 192 (22–24) (2003) 2649–2668.

[12] S.R. Idelsohn, E. Oñate, F. Del Pin, The particle finite element method: a powerful tool to solve incompressible flows with free-

surfaces and breaking waves, Int. J. Numer. Methods Engrg. 61 (2004) 964–989.

[13] S.R. Idelsohn, E. Oñate, F. Del Pin, N. Calvo, Fluid–structure interaction using the particle finite element method, Comput. Meth. Appl. Mech. Engrg. 195 (2006) 2100–2113.

[14] A. Kovacs, G. Parker, A new vectorial bedload formulation and its application to the time evolution of straight river channels, J. Fluid Mech. 267 (1994) 153–183.

[15] L. Larese, R. Rossi, E. Oñate, S.R. Idelsohn, Validation of the Particle Finite Element Method (PFEM) for simulation of free surface flows, Engrg. Comput., submitted for publication.

[16] R. Ohayon, Fluid–structure interaction problem, in: E. Stein, R. de Borst, T.J.R. Hugues (Eds.), Encyclopedia of Computational Mechanics, vol. 2, John Wiley, 2004, pp. 683–694.

[17] E. Oñate, Derivation of stabilized equations for advective–diffusive transport and fluid flow problems, Comput. Meth. Appl. Mech. Engrg. 151 (1998) 233–267.

[18] E. Oñate, A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation, Comput. Meth. Appl. Mech. Engrg. 182 (1–2) (2000) 355–370.

[19] E. Oñate, Possibilities of finite calculus in computational mechanics, Int. J. Numer. Methods Engrg. 60 (1) (2004) 255–281.

[20] E. Oñate, S.R. Idelsohn, A mesh free finite point method for advective–diffusive transport and fluid flow problems, Comput. Mech. 21 (1998) 283–292.

[21] E. Oñate, J. García, A finite element method for fluid–structure interaction with surface waves using a finite calculus formulation, Comput. Meth. Appl. Mech. Engrg. 191 (2001) 635–660.

[22] E. Oñate, J. Rojek, Combination of discrete element and finite element method for dynamic analysis of geomechanic problems, Comput. Meth. Appl. Mech. Engrg. 193 (2004) 3087–3128.

[23] E. Oñate, C. Sacco, S.R. Idelsohn, A finite point method for incompressible flow problems, Comput. Visual. Sci. 2 (2000) 67–75.

[24] E. Oñate, S.R. Idelsohn, F. Del Pin, Lagrangian formulation for incompressible fluids using finite calculus and the finite element method, in: Y. Kuznetsov, P. Neittanmaki, O. Pironneau (Eds.), Numerical Methods for Scientific Computing Variational Problems and Applications, CIMNE, Barcelona, 2003.

[25] E. Oñate, J. García, S.R. Idelsohn, Ship hydrodynamics, in: E. Stein, R. de Borst, T.J.R. Hughes (Eds.), Encyclopedia of Computational Mechanics, vol. 3, John Wiley, 2004, pp. 579–610.

[26] E. Oñate, S.R. Idelsohn, F. Del Pin, R. Aubry, The particle finite element method. An overview, Int. J. Comput. Methods 1 (2) (2004) 267–307.

[27] E. Oñate, A. Valls, J. García, FIC/FEM formulation with matrix stabilizing terms for incompressible flows at low and high Reynold's numbers, Comput. Mech. 38 (4–5) (2006) 440–455.

[28] E. Oñate, J. García, S.R. Idelsohn, F. Del Pin, FIC formulations for finite element analysis of incompressible flows. Eulerian, ALE and Lagrangian approaches, Comput. Meth. Appl. Mech. Engrg. 195 (23–24) (2006) 3001–3037.

[29] E. Oñate, M.A. Celigueta, S.R. Idelsohn, Modeling bed erosion in free surface flows by the Particle Finite Element Method, Acta Geotechnia 1 (4) (2006) 237–252.

[30] E. Oñate, S.R. Idelsohn, R. Rossi, Enhanced FIC–FEM formulation for incompressible flows, Research Report, CIMNE Barcelona, March, 2007.

[31] D.B. Parker, T.G. Michel, J.L. Smith, Compaction and water velocity effects on soil erosion in shallow flow, J. Irrigation Drainage Engrg. 121 (1995) 170–178.

[32] T.E. Tezduyar, Finite element method for fluid dynamics with moving boundaries and interface, in: E. Stein, R. de Borst, T.J.R. Hugues (Eds.), Encyclopedia of Computational Mechanics, vol. 3, John Wiley, 2004, pp. 545–578.

[33] C.F. Wan, R. Fell, Investigation of erosion of soils in embankment dams, J. Geotechn. Geoenviron. Engrg. 130 (2004) 373–380.

[34] O.C. Zienkiewicz, R.L. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, Elsevier, 2006.

[35] O.C. Zienkiewicz, R.L. Taylor, The Finite Element Method for Solid and Structural Mechanics, Elsevier, 2005.

Please cite this article in press as: E. Oñate et al., Advances in the particle finite element method for the analysis of ..., Comput. Methods Appl. Mech. Engrg. (2007), doi:10.1016/j.cma.2007.06.005