Comp. Part. Mech.
DOI 10.1007/s40571-016-0135-2

An assessment of the potential of PFEM-2 for solving long real-time industrial applications

Juan M. Gimenez[1,2] · Damián E. Ramajo[1,2] · Santiago Márquez Damián[1,3] · Norberto M. Nigro[1,2] · Sergio R. Idelsohn[4,5]

Received: 29 March 2016 / Revised: 29 August 2016 / Accepted: 17 September 2016

Abstract The latest generation of the particle finite element method (PFEM-2) is a numerical method based on the Lagrangian formulation of the equations, which presents advantages in terms of robustness and efficiency over classical Eulerian methodologies when certain kind of flows are simulated, especially those where convection plays an important role. These situations are often encountered in real engineering problems, where very complex geometries and operating conditions require very large and long computations. The advantages that the parallelism introduced in the computational fluid dynamics making affordable computations with very fine spatial discretizations are well known. However, it is not possible to have the time parallelized, despite the effort that is being dedicated to use space–time formulations. In this sense, PFEM-2 adds a valuable feature in that its strong stability with little loss of accuracy provides an interesting way of satisfying the real-life computation needs. After having already demonstrated in previous publications its ability to achieve academic-based solutions with a good compromise between accuracy and efficiency, in this work, the method is revisited and employed to solve several nonacademic problems of technological interest, which fall into that category. Simulations concerning oil–water separation, waste-water treatment, metallurgical foundries, and safety assessment are presented. These cases are selected due to their particular requirements of long simulation times and or intensive interface treatment. Thus, large time-steps may be employed with PFEM-2 without compromising the accuracy and robustness of the simulation, as occurs with Eulerian alternatives, showing the potentiality of the methodology for solving not only academic tests but also real engineering problems.

Keywords Particle methods · PFEM-2 · Large time-steps · Multiphase flows

1 Introduction

Hardware has been evolved considerably over the past few decades, increasing the computing performance and allowing better facilities for data entry and postprocessing of results. However, there have been no substantial improvements concerning the efficiency on the numerical methods. Computer simulation of incompressible fluid flows has been mainly based on the Eulerian formulation of the fluid mechanics equations on fixed domains [1]. The drawback of these strategies is quickly encountered in most practical engineering problems: very fine meshes and very small time-steps are needed to reach acceptable results. This handicap exceeds most of the time the capacity and speed of current powerful computers.

Starting from the work of Monaghan [2], another research branch that emerged was the development of methods based on the Lagrangian formulation [3]. Researchers found that the Lagrangian approach presents advantages over Eulerian one while solving problems in which deformations are not

✉ Juan M. Gimenez
jmarcelogimenez@gmail.com

[1] Centro de Investigación de Métodos Computacionales (CIMEC) - UNL/CONICET, Santa Fe, Argentina

[2] Facultad de Ingeniería y Ciencias Hídricas, Universidad Nacional del Litoral, Santa Fe, Argentina

[3] Departamento de Ingeniería Mecánica, Universidad Tecnológica Nacional, Facultad Regional Santa Fe, Santa Fe, Argentina

[4] International Center for Numerical Methods in Engineering (CIMNE), Barcelona, Spain

[5] Institució Catalana de Recerca i Estudis Avançats (ICREA), Barcelona, Spain

Published online: 07 October 2016

negligible. In addition, the explicit formulation of Lagrangian approach is in better position regarding the computational efficiency when employed in parallel computing. A better option is the use of the so-called hybrid methods, where the Lagrangian perspective is supported by an Eulerian reference grid or mesh, where the forces are computed with less error than meshless alternatives. In this context, several methodologies such as particle-in-cell (PIC) [4], marker-and-cell (MAC) [5], material point method (MPM) [6], and finally particle finite element method (PFEM) [7] have emerged. Despite the better capabilities regarding free-surfaces management, fluid–solid interaction, and explicit computing, the hybrid methods present their drawbacks and require expensive processing unnecessary in pure Eulerian frameworks. The cost of the extra tasks is mainly the construction of a new nondistorted mesh at each time-step, which besides being expensive, also limits the time-step leading to expensive simulations.

With the aim of reducing the computational requirements of the current methodologies, the works of Idelsohn et. al. [8,9] introduce improvements to PFEM, such as the novel X-IVS integration strategy and the employment of a fixed mesh, leading to PFEM-2. This latter method has demonstrated its ability to employ larger time-steps on convective dominant problems reducing the computing time required to solve a wide range of simulations [12,13]. The strategy was also extended and validated for solving multiphase flows [14,15], fluid–structure problems [16], and those with surface tension dominance [10,17], always preserving the mentioned capabilities.

As demonstrated in the work of Idelsohn et. al. [11], for convection-dominated cases, the usage of a Lagrangian frame with particles as PFEM-2 produces results with the same accuracy as classical Eulerian alternatives even while using coarser meshes and larger time-steps. Moreover, while employing the same numerical setting in cases with interface evolution, it is not possible to obtain any useful solution (simulations blow-up) with Eulerian methods without reducing the time-step. In this context, this work reviews the PFEM-2 methodology and applies it to simulate several problems with high impact on industrial needs. In order to make use of the goodness of the method, every simulation employs large time-steps. This choice leads to high CFL numbers, exceeding those normally employed by Eulerian formulation-based simulations, due to stability reasons. This valuable capability of enlarging the time-step provides the PFEM-2 with the ability of making affordable very long real-life phenomena often found in industrial applications. Details about the PFEM-2 code employed to run the tests were previously presented in [12,13] where parallel efficiency up to 32 processors was analyzed. Although new features were included to solve current tests, the main core of the code was not modified, and therefore, no discussion about implementation details is included here considering that conclusions are almost the same as those reported before.

Having thus demonstrated its ability in academic situations, the next step is to prove how well PFEM-2 behaves while we go for solving engineering problems. In industrial applications, it is common to find equipment that needs a long time to carry out physical–chemical processes making the computational fluid dynamics (CFD) to encounter severe difficulties in terms of feasibility. A problem which involves long-time running operation is the residence time distribution computation, normally employed by chemical engineers to design their plant's flow-sheet, aiming at the optimization of the operation. Keeping in mind the expansion of the CFD in the engineer's life and the need for finding the optimal configuration, the main question to be answered in the near future is what degree of maturity is required for the CFD to afford these challenges. In this work, two typical examples, like a contact tank employed in a waste-water treatment plant and a water–oil separation tank normally used in enhanced oil recovery (EOR) in petroleum extraction, are presented. This set of cases requires very long real-time simulations, which probably disqualifies any standard Eulerian formulation. Without the possibility of parallelizing the time evolution, one of the few possible alternatives is using PFEM-2 allowing for increasing the time-steps, thereby reducing noticeably the computing time. However, the last is possible to be achieved only due to the Lagrangian approach on which PFEM is based. On the other hand, the chaotic nature of the flow inside the tanks requires turbulence modeling at particle level. In this context, this work presents another technique associated with the extension of the X-IVS methodology in combination with a strategy like random-walk to simulate the randomness of particle paths and compare with the Eulerian alternative of computing the turbulence diffusion implicitly over the mesh. Even though preliminary results have not shown any advantage in favor of this technique over the previous methodology, further work could be carried out based on this idea in order to improve it.

Remaining test cases are focused on showing the advantages of PFEM-2 in managing two-phase simulations, mainly when a proper computing of the interface evolution dominates the quality of the solution. Interface modeling incorporates an advection equation to the system, which is solved naturally, i.e., rapidly and accurately, using the Lagrangian frame. This approach overcomes the restrictions of Eulerian alternatives, such as the need of high-resolution schemes, stabilization terms, flux-limiters, and in most of cases, reduction in the number of time-steps. In the first place, an application of PFEM-2 to mold filling, one of the most important processes in metallurgical foundries, is presented. Those cases are solved with the two-phase version of PFEM-2, presented previously in [10,17], where the accuracy and efficiency were assessed for academic problems, and here they were compared with experimental references coming from the literature. Finally, the application of the method to the problem of a catastrophic failure of a storage tank is reported. In this simulation, qualitative (snapshots of the collapse and visual description) and quantitative results (mass bound overtopping and forces over walls) are shown.

The final section of this is devoted to conclusions based on the results presented in this work, where the capabilities of the method are highlighted, and future research lines are suggested and commented upon.

2 Numerical modeling

For solving the problems planned in this paper, a multiphase, isothermal, and incompressible Newtonian flow model is presented. This formulation, like a typical volume of fluid (VOF), considers a mixture of separated phases in a domain Ω, where the fluid-intensive properties depend on an indicator function λ used to distinguish between the two phases, i.e., $\rho = \rho(\lambda)$ and $\mu = \mu(\lambda)$ for density and dynamic viscosity, respectively. Then, the interface capturing model is composed by the Navier–Stokes equation constrained by the mass-conservation equation and the pure advective transport equation for only one of the phases with an equation for the interface movement, all of them employing the Lagrangian reference frame as

$$\begin{cases} \rho \dfrac{D\mathbf{V}}{Dt} = \nabla \cdot \left[\mu \left(\nabla \mathbf{V} + \nabla \mathbf{V}^T \right) \right] - \nabla p + \rho \mathbf{g} + \mathbf{F}_\Gamma \\ \nabla \cdot \mathbf{V} = 0 \\ \dfrac{D\lambda}{Dt} = 0 \\ \dfrac{D\mathbf{x}}{Dt} = \mathbf{V} \end{cases} \quad \text{in } \Omega, \tag{1}$$

where $\mathbf{V}$ and p are the unknowns, velocity and pressure, respectively, and, as the Lagrangian frame is employed, a trajectory equation for particle positions $\mathbf{x}$ is also included. The first equation is the so-called *momentum-conservation* equation. It relates the total variation of the momentum with the forces applied, with $\mathbf{g}$ being the gravity acceleration and $\mathbf{F}_\Gamma = \sigma \kappa \delta_\Gamma \mathbf{n}_\Gamma$ being the force over the interface between the phases due to surface tension, where δ_Γ is the Dirac delta function that localizes the surface tension force on the interface. The set of Eq. (1) has the advantage of avoiding the constant splitting of the domain into two sub-domains. This is why this method is called the interface capturing.

The numerical method employed to solve the equation system is PFEM-2. This methodology is based on a hybrid spatial discretization which includes a cloud of Lagrangian particles and a fixed background Eulerian mesh. In this work, just a brief description of the theory of the numerical method is presented. Further details can be found in several previous works of the authors in the literature [9,10,15,17].

In the context of incompressible flows, a point, known as particle and denoted by p, has associated physical variables like velocity $\mathbf{V}_\mathrm{p}$, temperature T_p; and mathematical variables like λ_p which indicates the kind of phase of the particle. In PFEM-2, massless particles are employed (they do not have associated size), allowing to use variable densities of particles through the domain. This is useful for refining the approximation in some regions of interest. Also, the particles carry the original data which is never replaced but only updated according to the computing obtained by the mesh, which plays a secondary role, thus avoiding the excessive numerical diffusion.

Temporal discretization of equations is based on the fractional-step-splitting method (FSM) [18]. However, as explained in depth in [10], the momentum predictor is also split into three stages: one Lagrangian step where the convection is solved just moving the particles; a second step where the convected states over particles are projected to the mesh nodes; and the last one which solves the remaining terms (diffusive, reaction, body forces) over the fixed mesh using standard finite elements. The Lagrangian step integrates the particle trajectories using the X-IVS algorithm presented in [8] and extended to two-phase flows in [15,16], while different strategies for the projection step are analyzed in [10,23]. It must be also remarked that the last step working over the mesh is substantially easier to solve problems such as nonlinearities, stabilization, nonsymmetric matrices which disappear without the convective term in the equation being removed by the Lagrangian formulation. Finally, the time splitting is completed by solving the remaining two steps in the traditional FSM, i.e., the pressure calculation and the velocity correction. Readers interested in additional features of the method can find a formulation for the treatment of surface tension forces in [17] or fluid–structure interaction in [35], and a high-performing implementation along with a parallel efficiency analysis in [12,13].

2.1 Algorithm synopsis

It is assumed that all fluid variables are known at time t^n for both the particles and the mesh nodes. Subindexes $()_j$ y $()_\mathrm{p}$ represent a generic mesh node j and a generic particle p, respectively. Let N be the finite element linear basis functions. According to this notation, the steps are presented in Algorithm 1, where $\mathbf{x}$ is a spatial coordinate, $\delta \mathbf{V} = \mathbf{V} - \widehat{\widehat{\mathbf{V}}}$ is the velocity correction computed on the mesh, where $\widehat{\widehat{\mathbf{V}}}$ is an initial value of the velocity considering only convection, and $\widehat{\mathbf{V}}$ is the predicted velocity considering both convection and diffusion. The numerical parameter θ_p can be 0 or 1 depending on the pressure restart choice affecting the approximation order of the pressure, and the other parameter, θ_μ, can be 0 or 1 depending on the necessity or not of an accurate diffusion calculation when large Fourier numbers are employed. Also, the external forces are compacted in a vector $\mathbf{F}$ such as $\mathbf{F} = \int_\Omega N_j\,(\rho\mathbf{g} + \mathbf{F}_\Gamma)\;d\Omega$, and $\mathbf{M}$, $\mathbf{K}$, and $\mathbf{B}$ are the standard mass, stiffness and gradient (or mixed) matrices, respectively, of any FEM assembling. Finally, τ is a stabilization parameter for the pressure according to the work of Codina [34], while $\boldsymbol{\pi}^n$ is the recovery of the pressure gradient on mesh nodes enforcing its continuity.

Algorithm 1 - Time-Step PFEM-2 for two-phase incompressible fluids.

```
1. Convective Stage over particles:
     $\begin{cases} \mathbf{x}_\mathrm{p}^{n+1} = \mathbf{x}_\mathrm{p}^{n} + \int_{n}^{n+1} \mathbf{V}^n(\mathbf{x}_\mathrm{p}^{\tau})\;d\tau \\ \widehat{\widehat{\mathbf{V}}}_\mathrm{p}^{n+1} = \mathbf{V}_\mathrm{p}^{n} \\ \lambda_\mathrm{p}^{n+1} = \lambda_\mathrm{p}^{n} \end{cases}$
2. Projection Stage from particles to nodes:
     $\mathbf{M}_{ij}\widehat{\widehat{\mathbf{V}}}_j^{n+1} = \mathbf{M}_{ip}\widehat{\widehat{\mathbf{V}}}_\mathrm{p}^{n+1}$
     $\mathbf{M}_{ij}^{L}\lambda_j^{n+1} = \mathbf{M}_{ip}\lambda_\mathrm{p}^{n+1}$
3. Momentum Stage over mesh:
     $\left(\mathbf{M}\left(\frac{\rho}{\Delta t}\right) + \mathbf{K}(\mu)\right)\widehat{\mathbf{V}}^{n+1} = \mathbf{M}\left(\frac{\rho}{\Delta t}\right)\widehat{\widehat{\mathbf{V}}}^{n+1} - \theta_\mathrm{p}\mathbf{G}p^n + \mathbf{F}^{n+1}$
4. Pressure-Calculation Stage over mesh:
     $\mathbf{K}\left(\frac{\Delta t}{\rho} + \tau\right)p^{n+1} = \mathbf{B}\widehat{\mathbf{V}}^{n+1} + \mathbf{K}\left(\frac{\Delta t}{\rho}\right)\theta_\mathrm{p}p^n + \theta_\mathrm{p}\mathbf{B}(\tau)\boldsymbol{\pi}^n$
5. Correction Stage over mesh and particles:
     $\mathbf{M}(\rho) + \mathbf{V}^{n+1} = \mathbf{M}(\rho)\widehat{\mathbf{V}}^{n+1} - \Delta t\mathbf{G}\left(p^{n+1} - \theta_\mathrm{p}p^n\right) + \theta_\mu\mathbf{K}\left(\frac{\mu}{\rho}\right)\left(\mathbf{V}^{n+1} - \widehat{\mathbf{V}}^{n+1}\right)$
     $\rho_\mathrm{p}\mathbf{V}_\mathrm{p}^{n+1} = \rho_\mathrm{p}\widehat{\widehat{\mathbf{V}}}_\mathrm{p}^{n+1} + \sum_j \delta\mathbf{V}_j^{n+1} N_j(\mathbf{x}_p^{n+1})$
```

3 Residence time distributions

3.1 Background

In the oil and gas industry, the water–oil separators are an essential part of the superficial installations downstream of the wells. Particularly, in secondary oil recovery, the extraction requires large amount of water that needs to be reinjected into the wells in order to attend environmental issues. The produced water is reinjected to help to extract crude by drag and buoyancy forces. The produced water must be thoroughly cleaned of the remaining oil in order to avoid possible obstructions making the well to be put out of service. This cleaning process is carried out in part with a separation equipment called skimmer, which basically consists of a big tank with internals where the water–oil mixture enters and the crude is separated by buoyancy and collected on the free-surface, while the water is removed from the bottom.

A parameter used to characterize the mixing and flow within tanks and to compare the behaviors of real skimmers with their ideal models is the residence time distribution (RTD). It is a probability distribution function that describes the time a fluid element could remain inside the tank, for its purification. The RTD is useful, not only for troubleshooting in the existing skimmers allowing for their optimization, but also in estimating the yield of a given separation and improving the future tank designs. The standard function $E(t)$, normally employed for RTD, has the units of time^{-1} and is defined such that

$$\int_0^\infty E(t)\,\mathrm{d}t = 1 \qquad (2)$$

In the absence of dispersion and for a constant volumetric flow ($Q = Q_0$), the theoretical residence time $\hat{t}$ is defined as

$$\hat{t} = \frac{V}{Q}, \qquad (3)$$

with V the tank volume, must be equal to the mean residence time, t_m. As is the case with other variables described by distribution functions, the mean value of the variable is equal to the first moment of the RTD function, $E(t)$. Thus, the mean residence time is

$$t_\mathrm{m} = \int_0^\infty t\;E(t)\,\mathrm{d}t \qquad (4)$$

Tracer injection from the inlet is the strategy which allows for experimental or numerical measurements of the RTD function. The kind of material and amount of the tracer is selected such that it modifies neither the fluid's physical properties nor its hydrodynamic conditions. There are two options mainly used for the tracer inlet distribution: a pulse or a step. In the first case, a small volume of tracer, which approximates to the Dirac delta function, is introduced. The injection time t_inj must be much smaller than the theoretical residence time, i.e., $t_\mathrm{inj} \ll \hat{t}$. Then, considering $C(t)$, the tracer concentration at the outlet at any time t, the RTD function can be obtained as

$$E(t) = \frac{C(t)}{\int_0^\infty C(t)\,\mathrm{d}t} \qquad (5)$$

In an experiment with a step function, the tracer concentration at the inlet changes suddenly at a given time t from 0 to C_0. The tracer concentration is measured at the outlet and normalized in order to obtain the cumulative RTD function $F(t)$ having values between 0 and 1:

$$F(t) = \frac{C(t)}{C_0} \tag{6}$$

The responses to the pulse and to the step are related as

$$\begin{cases} F(t) = \int_0^t E(t)\,\mathrm{d}t \\ E(t) = \frac{\mathrm{d}F(t)}{\mathrm{d}t} \end{cases} \tag{7}$$

The efficiency of a skimmer tank is measured as the capacity of separating water and oil. Because the equipment works only by buoyancy, the more time the injected mixture resides in the tank, the more separation is obtained. Thus, it is expected that the skimmer works as a plug–flow reactor (PFR) where no dispersion of $E(t)$ is obtained. However, because of geometric or economic constraints, real tanks have channeling (bypassing or short circuiting) and dead zones which tend to modify the desired ideal behavior by reducing the residence times of some injected fluid portions. Moreover, the presence of flow turbulence leads to nonideal behaviors. In this context, numerical experiments are widely used to optimize geometric designs and operational conditions without the need for a large set of experimental tests. Finding $E(t)$ and/or $F(t)$ for several tank prototypes using numerical experiments requires very long time-consuming simulations (for standard skimmers, several hours of real time). As aforementioned, PFEM-2 is able to employ large time-steps that are useful to solve in a faster way this type of problems without compromising stability. In addition, because the flow behavior inside a skimmer tank has regions dominated by inertia and a proper tracer transport mainly depends on the convective term, it has been demonstrated that the method will introduce less numerical errors than other Eulerian alternatives.

3.2 Turbulent dispersion

The concentration C of the tracer follows an advective–diffusive equation, i.e.,

$$\frac{DC}{Dt} = \nabla \cdot (\alpha_\mathrm{T} \nabla C) \tag{8}$$

Most of real skimmers present turbulent flow regime, i.e., the experimental RTD can be described by the plug–flow with axial dispersion model. In this context, the simulated dispersion coefficient α_T can be described as the sum of three contributions, two of them are physical: molecular (α_m) and turbulent dispersions (α_t), and the third term is the numerical diffusion (α_n):

$$\alpha_\mathrm{T} = \alpha_\mathrm{m} + \alpha_\mathrm{t} + \alpha_\mathrm{n} \tag{9}$$

The tracer employed must guarantee negligible molecular diffusion. In general, it is known that molecular diffusion has little effect on ensemble mean concentrations in high Reynolds number flows [19]. However, experiments show that the turbulent contribution α_t is not negligible for flow regimes being analyzed in this section, requiring the numerical computation of the diffusion to accurately predict the behavior of the tracer. Regarding the third term, it is well known that Lagrangian integration does not introduce excessive numerical diffusion, while Eulerian strategies introduce more or less diffusion depending on the discretization strategy selected and the mesh employed.

With the PFEM-2 method, the advection (i.e., left-hand side of Eq. 8) is automatically done through the movement of the particles. Each particle carries its own character or concentration value C_p, marking with $C_\mathrm{p} = 1$ for those that belong to the tracer, leaving the others with $C_\mathrm{p} = 0$. The diffusive part of the equation can be solved in PFEM-2 either explicitly or implicitly. The former, which computes the diffusion over particles during its streamline integration (X-IVAS method [8]), cannot be employed with time-steps larger than $\alpha_\mathrm{T}\Delta t/\Delta x^2 > 1$. On the other hand, the latter is unconditionally stable and consists of projecting the concentration field from particles to nodes, solving a Laplacian equation on the mesh, and interpolating the incremental correction again over the particles. When only the tracer is solved, this strategy adds three extra stages per time-step, increasing the computational work, mainly caused by the solution of the equation system for diffusion. It can be noted that both approaches lose the particle character, allowing it to take values $0 \leq C_\mathrm{p} \leq 1$. Readers interested in further information can refer to a previous work of the authors where they discuss in depth both alternatives [23].

In this work, an alternative approach, which can be solved explicitly over particles without any stability issue and without modifying the particle character, is presented. It is based on the traditional idea of modeling the motion of single particles in a turbulent flow assuming that the turbulent motions of the particles are similar to the Brownian motions of particles or molecules, i.e., a stochastic strategy. Therefore, the movement of the particles due to the turbulent part of the flow field is modeled with a random-walk scheme. The tracer position $\mathbf{x}(t)$ in a continuous random-walk model (CRW) can be described by the nonlinear Langevin equation:

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{A}(\mathbf{x}, t) + \overline{\overline{B}}(\mathbf{x}, t)\boldsymbol{\xi}(t) \tag{10}$$

where the vector $\mathbf{A}(\mathbf{x}, t)$ represents the variation of $\mathbf{x}(t)$ due to the mean velocity field (i.e., deterministic part), while the second term is a stochastic term which depends on a tensor $\overline{\overline{B}}(\mathbf{x}, t)$ that characterizes the random motion and a vector $\boldsymbol{\xi}(t)$ with random numbers normalized between 0 and 1. Following the assumptions presented by Fabbroni [21] and considering isotropic turbulent diffusivity, the Eq. 10 can be applied to each particle leading to

$$\Delta \mathbf{x}_{\mathrm{p}} = \mathbf{V}_j(\mathbf{x}_{\mathrm{p}})\Delta t + \sqrt{2\alpha_{\mathrm{t}}\Delta t}\ \mathbf{w} \tag{11}$$

where $\mathbf{w}$ is a random Gaussian variable with zero mean and standard deviation of 1. An additional term that allows incorporating physical anisotropy to the model was introduced in recent works [24]. The key idea is to move the particles from high turbulent zones to lower ones in order to represent the diffusion in a more realistic way.

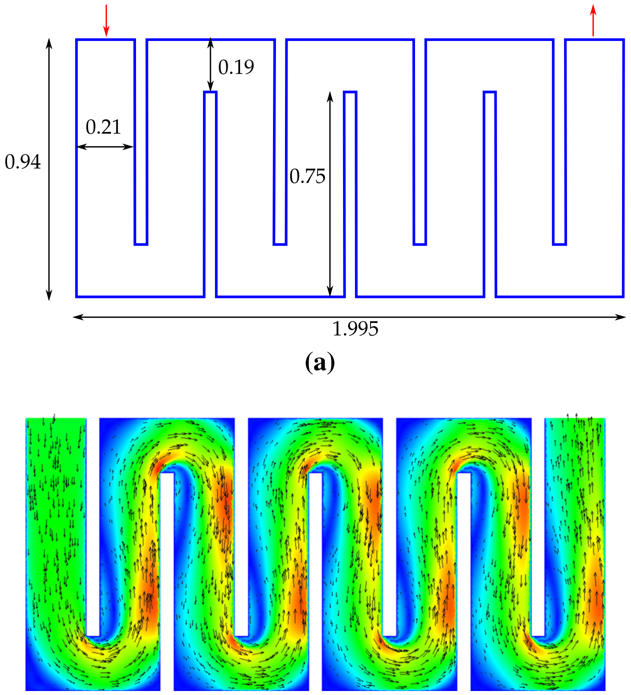
$$\Delta \mathbf{x}_{\mathrm{p}} = \mathbf{V}_j(\mathbf{x}_{\mathrm{p}})\Delta t - \nabla\alpha_{\mathrm{t}}(\mathbf{x}_{\mathrm{p}})\Delta t + \sqrt{2\alpha_{\mathrm{t}}\Delta t}\ \mathbf{w} \tag{12}$$

There are other alternatives in order to model the turbulence over particles, the most used being called Discrete Random Walk (DRW). This model, which is also known as Eddy Interaction Model (EIM) [22], is based on reconstructing the instantaneous field from the local mean values of velocity and turbulent intensity. The discrete time intervals between velocity updates are dictated by the *eddy lifetime* which varies spatially according to the turbulent fields. Due to the explicit and parallel integration employed by PFEM-2, this approach is far from being efficient when time-steps larger than eddy lifetime are used. Finally, in this work, the CRW, based on the approach assumed by Eq. 12, is selected for modeling turbulent dispersion.

3.3 Contact tank

Although the purpose is rather different, the optimization of chlorine contact tanks follows similar design strategies as presented at the beginning of this section for the case of skimmer tanks. The target task of a chlorine contact tank, based on the volume displacement criterion, is to achieve enough detention time to successfully conclude the chlorination process. This criterion relies upon the assumption that plug–flow conditions are held along the tank. However, the existence and arrangement of baffles in the tank, and the inlet and outlet configurations, can result in a much more complex flow pattern, where turbulent mixing, dead zones, and short-circuiting exist. Therefore, a RTD analysis is usually employed to evaluate and optimize the design.

Following the experimental work of Shiono [26], an example is presented to show the performance of the CRW compared with the calculation of turbulent diffusion over the mesh. The case consists on a 2D model of a contact tank with seven baffles and eight compartments, as presented in Fig. 1. The discharge entering the tank is 1.17 (L s^{-1}), resulting in a mean cross-sectional velocity of 0.0104 (m s^{-1}). This flow configuration leads to a $\hat{t} \approx 750$ (s). The geometry is discretized with an unstructured mesh composed of triangles. Elements have an almost uniform size of $\Delta x \approx 0.01$ (m).



(a)

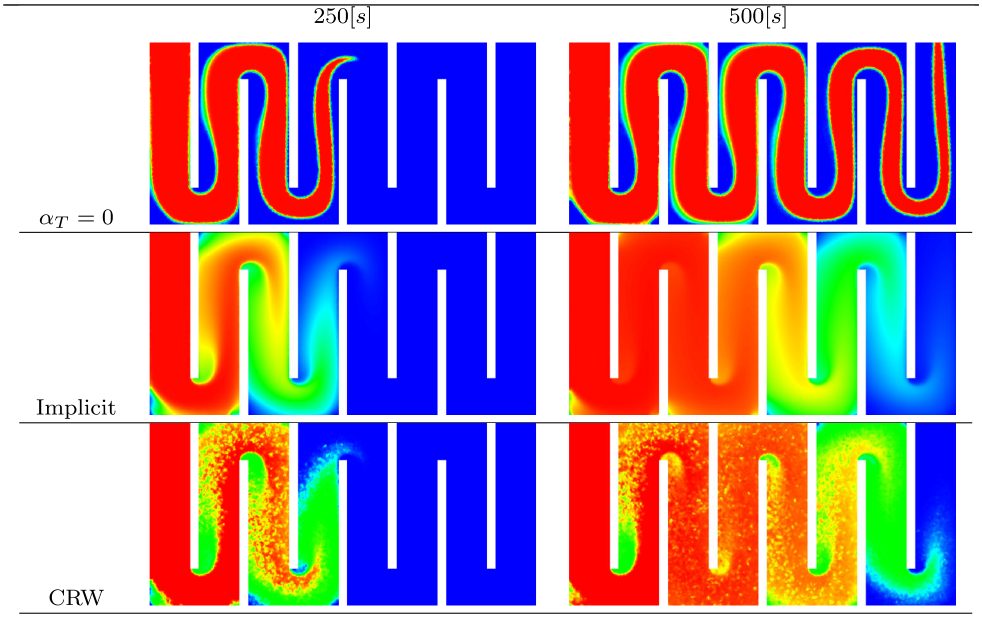
(b)

Fig. 1 Contact tank case. Geometric configuration and steady-state incompressible flow field calculated with $k - \epsilon$. Geometry dimensions are expressed in meters. Magnitude of velocity is scaled from 0 (m s^{-1}) (*blue*) to 0.22 (m s^{-1}) (*red*). **a** Case configuration. **b** Steady flow

In order to obtain the time-averaged flow quantities, a first simulation is done. Turbulence is modeled employing the $k - \epsilon$ model. This flow, shown in Fig. 1b, is in good agreement with the other published works [25,27]. The turbulent kinematic viscosity ν_{t} can be obtained as $\nu_{\mathrm{t}} = C_\mu k^2/\epsilon$, with $C_\mu = 0.09$ being a constant parameter. Then, the turbulent dissipation is found as $\alpha_{\mathrm{t}} = \nu_{\mathrm{t}}/\sigma_{\mathrm{t}}$, where σ_{t} is named turbulent Schmidt number. The value of σ_{t} is adjusted from experiments. Previous works have verified that a turbulent Schmidt number of 0.44 would give a good agreement between experimental and simulated RTDs.

Next step is computing the tracer transport. In order to solve Eq. (8), the steady-state velocity and turbulent fields are employed. The simulation is carried out using the two numerical strategies available, which vary according to the turbulent dispersion approach employed, i.e., an *implicit* calculation isotropic over the mesh, or a stochastic simulation with *CRW*. Table 1 presents tabulated snapshots for the tracer evolution for both strategies also comparing with a pure advective solution.

Table 1 Screen-shots for tracer distribution at $t = 250$ (s) (left) and $t = 500$ (s) (right) simulated with several PFEM-2 variants



Scale goes from 0 (blue) to 1 (red)

Without turbulent diffusion treatment, the solutions obtained are nonphysical leading to unrealistic RTDs ($F(t)$ never reaches the unity). On the other hand, the inclusion of dispersion modeling let to the entire volume will be filled by tracer. However, the transient evolution from the initial state (without tracer) to the final state is simulated differently with implicit or CRW strategies. The random-walk solution shows greater axial diffusion while the implicit one presents larger diffusion at front of tracer step.

Figure 2 compares among the cumulative RTDs calculated using PFEM-2 and a numerical reference calculated with the CFD software OpenFOAM®2.4.0 which implements the finite volume method (FVM). Finer-mesh ($\Delta x = 0.005$ (m)) and high-order methods for convection and other interpolations were employed in order to obtain a reference solution. The reference cumulative RTD obtained is in agreement with other numerical results reported in the literature [27]. Comparing the reference with PFEM-2 simulations, a better agreement with the Implicit option can be seen. The stochastic approach tends to transport a lower dissipated step leading to a cumulative RTD being too sharp.

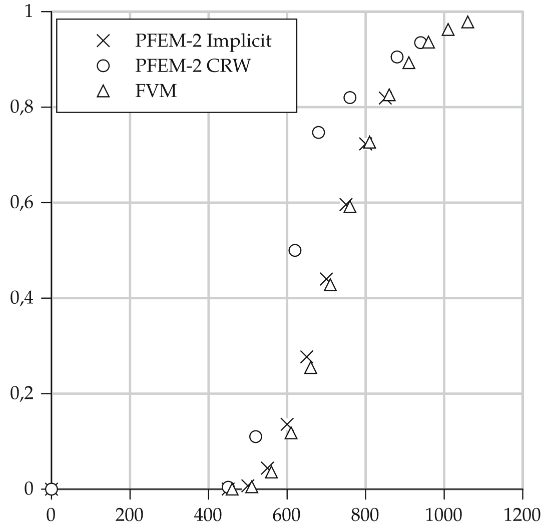


Fig. 2 Cumulative residence time distribution for contact tank case. Comparison between both options for turbulent dispersion in PFEM-2 (implicit and random-walk) and an Eulerian reference solution

Regarding to computing times, if only the transport equation for the tracer is solved, only then the random-walk approach becomes cheaper. The need of projection, solution of an equation system, and correction over particles makes the implicit strategy approximately 40 % more expensive than the stochastic strategy. Although this 2D case was solved considering a steady-state flow, most of tracer transports must be simulated together with the unsteady flow. In such cases, PFEM-2 already requires the three mentioned steps for the velocity and pressure calculation; therefore, the inclusion of the tracer treatment does not add significant extra work. Considering the above-mentioned facts, the implicit treatment of the turbulent dispersion is chosen as a modeling strategy for the next example.

3.4 Skimmer tank

The skimmer selected for the numerical experiments is a cylindrical tank with aspect ratio between the height H and

Table 2 Skimmer tank's main features and operative conditions

Feature	Symbol	Value
Height	H	17.89 (m)
Diameter	D	11.93 (m)
Volume	V	2000 (m^3)
Inlet Flow	Q	1/9 ($\mathrm{m}^3\mathrm{s}^{-1}$)
Theoretical mean residence time	$\hat{t}$	5 (hs)
Fluid viscosity	μ	0.001 ($\mathrm{Kg\,s}^{-1}\,\mathrm{m}^{-1}$)
Fluid density	ρ	1000 ($\mathrm{Kg\,m}^{-3}$)
Tracer diffusivity	α	0 ($\mathrm{m}^2\,\mathrm{s}^{-1}$)

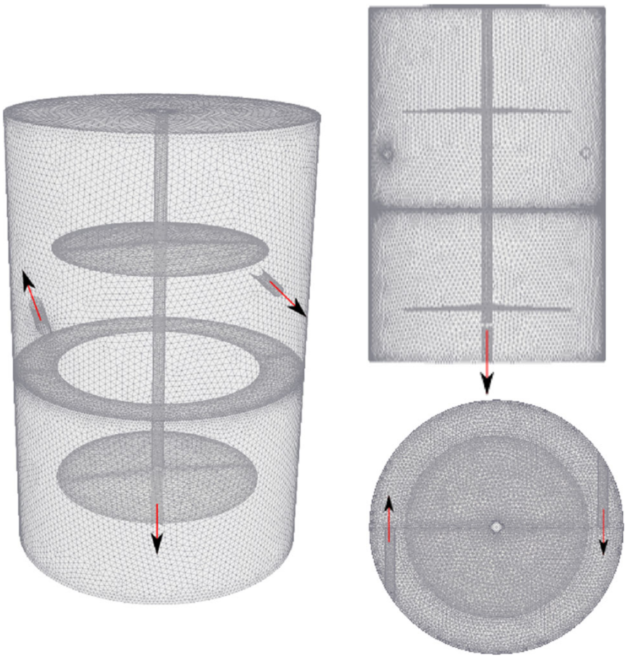


Fig. 3 Skimmer tetrahedral mesh employed in the numerical experiments

diameter D of $H/D = 1.5$. It comprises three plates. Two of them are held by the central column which is also used for inlet and outlet, and the third is supported by the cylindrical tank wall named ring plate. Every member has a three-degree inclination. The fluid enters through two pipes located below the upper plate that injects the load tangentially to the tank walls in order to produce a rotational movement concentric with the vertical axis of the tank. Table 2 presents the main characteristics of the tank and the operative condition. The mesh employed can be observed in Fig. 3, which consists of 306,768 nodes corresponding to 1,683,575 tetrahedral elements.

As the tracer injection is performed when the flow is established, a start-up simulation is carried out started with a hydrostatic initial condition along the tank, i.e., filled with standing water without tracer. Concerning the boundary conditions, a fixed and uniform velocity is imposed at the inlet to satisfy the required flow rate, the pressure is fixed to a reference value ($p = 0$) at the outlet, the free-surface is assumed as slip, and the remaining boundaries are considered as walls.

In order to model the turbulent behavior of the flow, the static Smagorinsky large eddy simulation (LES) turbulence model [28] is employed, which captures the effects unresolved by the scale of the mesh. This approach models the turbulent viscosity as

$$\mu_\mathrm{t} = C_\mathrm{s}^2 \rho \Delta^2 |\mathbf{S}| \tag{13}$$

being $\mathbf{S} = \nabla\mathbf{V} + \nabla\mathbf{V}^T$ and $C_\mathrm{s} = 0.2$ the Smagorinsky constant. More references about the turbulence modeling in PFEM-2 can be found in [29] and [12].

The start-up simulation was performed employing a $\Delta t = 2$ (s) which leads to a maximum $\mathrm{CFL}_{\max} \approx 20$, being $\mathrm{CFL} = \dfrac{|\mathbf{V}|\Delta t}{\Delta x}$. The final time T_f is chosen when the integral of the kinetic energy reaches a converged value, as shown by Fig. 4a. At this moment, no significant changes are observed at the main flow, apart from small fluctuations caused by its own unsteadiness. In Fig. 4b, the magnitude of the velocity field is presented. The tank plates are not shown for clarity, but three horizontal and one vertical slices are included to visualize the obtained solution. With a mean inlet velocity of about 0.24 ($\mathrm{m\,s}^{-1}$), it is possible to see the flow surrounding the tank going upward and downward close to the walls. Using a limited scale adjusted between 0 and 0.01 (Fig. 4b), it is also possible to identify a toroidal flow pattern.

Considering the solution presented in Fig. 4 as the initial condition, a new simulation is performed, but now by injecting a fully concentrated tracer at the inlet and solving an extra transport equation for it (Eq. 8). The entering particles through the inlet are marked with $C_\mathrm{p} = 1$, leaving the others with $C_\mathrm{p} = 0$. Each particle carries its own concentration value C_p, which is projected to the mesh in order to calculate the turbulent dispersion implicitly. The advective–diffusive tracer equation is solved together with the incompressible flow equations, remarking that the former does not produce any modification into the latter (passive scalar).

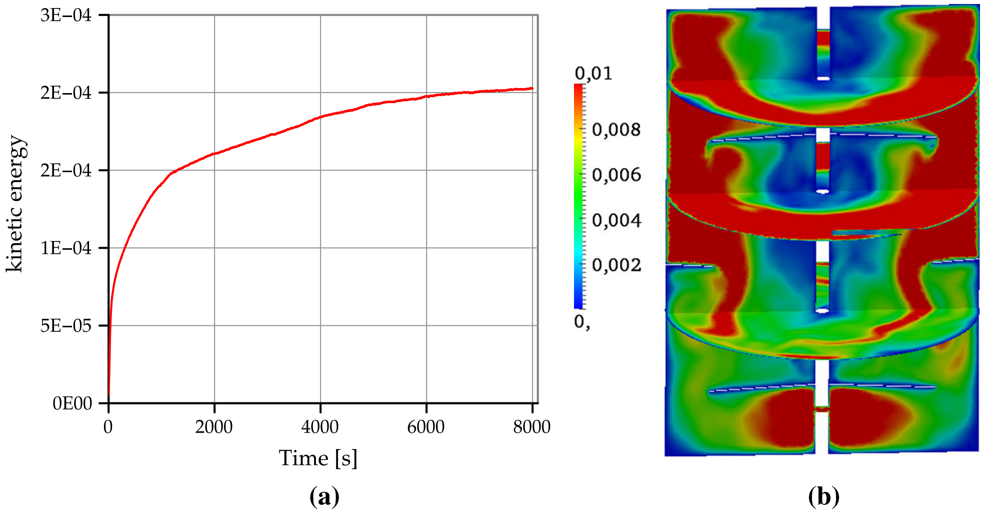


Fig. 4 **a** Integral of the kinetic energy. **b** Velocity magnitude of the final solution in the start-up simulation sampled at slices

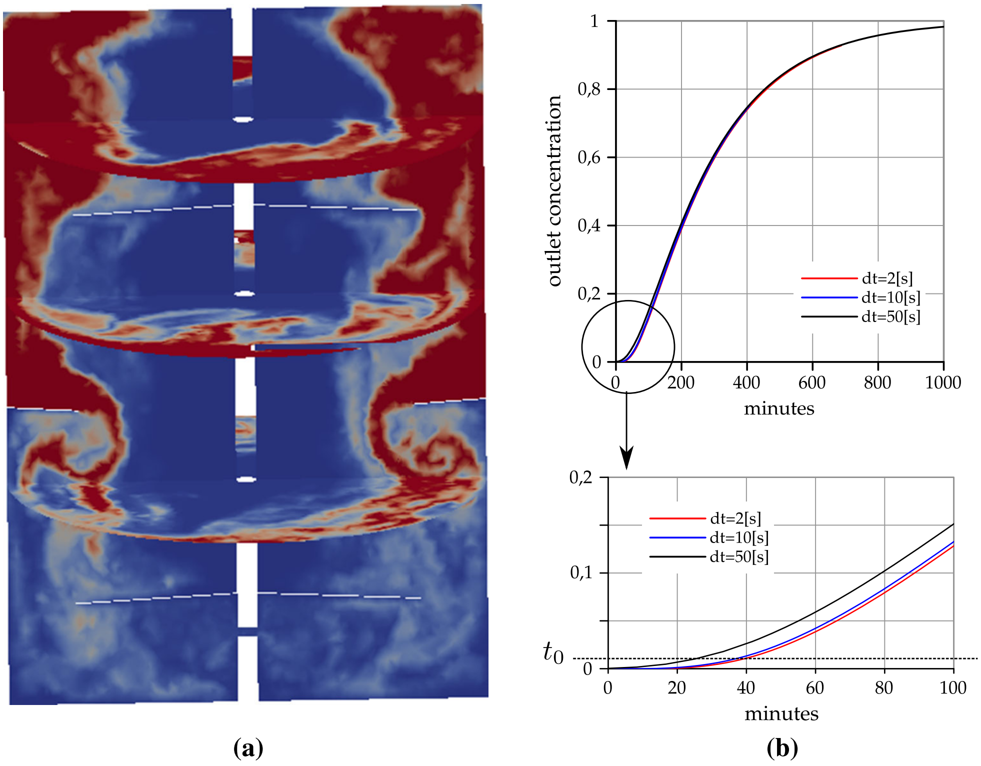


Fig. 5 *Left* snapshot with the concentration field at $T = 80$ min. *Right* cumulative RTD using different Δt. **a** Concentration saturated from $C = 0$ (*blue*) to $C = 0.25$ (*red*). **b** Accumulated RTD. (Color figure online)

A qualitative analysis of the numerical experiment with step-type injection is presented in Fig. 5a. In the figure, a capture of the tracer concentration employing a scale limited to $0 < C < 0.25$ after 80 min is shown. The tracer is concentrated mostly above the ring plate because the main flow is ascending, but some tracer overflows that plate and gradually fills the lower tank section, which allows us to obtain concentration at the outlet. On the other hand, a quantitative analysis is presented in the Fig. 5b. The temporal evolution of the concentration measured at the outlet corresponds directly to the function $F(t)$ (Eq. 6). The same tests were simulated employing different time-steps $\Delta t = 2$, 10, and 50 which leads to $\text{CFL}_{\text{mean}} \approx 4$ and $\text{CFL}_{\text{max}} \approx 500$ in the last case. The three cases show similar overall solutions.

A relevant result in this type of analysis is the fraction of tracer that has been in the tank for less time than $\hat{t}$, i.e.,

$F(\hat{t}-\epsilon)$ with ϵ small enough. An ideal PFR would obtain $F_{\mathrm{PFR}}(\hat{t}-\epsilon) = 0$, but considering current simulations of the real tank, the obtained value is $F_{\Delta t=2}(\hat{t}-\epsilon) \approx 0.598$ (taken from the simulation with smallest time-step). This value should be compared with the obtained by simulating other skimmer designs. This allows to determine the relative quality of the tank, characterizing the troubleshooting of this particular design. The more relevant problem is the reduction of the effective volume mainly because of the dead zone around the axial axis of the tank where the tracer rarely enters. Simulations with larger time-step predict similar values for $F(\hat{t}-\epsilon)$, the solution for $\Delta t = 10$ being very similar to the reference, $F_{\Delta t=10}(\hat{t}-\epsilon) \approx 0.6$, but slightly different from the solution while employing the largest time-step ($F_{\Delta t=50}(\hat{t}-\epsilon) \approx 0.608$).

In this work, the first arrival time t_0 is defined as

$$t_0 = \min(t) \ \ \forall t \mid F(t) > 0.01, \tag{14}$$

which gives information of the shortest path between the inlet and the outlet (i.e., short-circuiting) in the tank. In this case, solutions with $\Delta t = 2$ and 10 (s) present similar results ($t_0 \approx 38$ min), while the simulation with largest time-step underestimate the arrival time at 25 min.

Regarding to computational times, as mentioned above, the benefits of employing large time-steps are very much appreciated for solving this type of problems. In this particular case, and considering at least 500 min of simulation time required, if the numerical scheme is restricted to $\mathrm{CFL}_{\max} < 1$, then more than 3×10^6 iterations are needed. A moderate $\Delta t = 2$ (s), which can be used with other implicit Eulerian alternatives, was selected as a PFEM-2 reference solution requiring one order less of iterations, although it required more than two days of CPU running time for the twelve processes in parallel. By employing a larger time-step $\Delta t = 10$ (s), PFEM-2 achieves a similar solution, but spending only 12 h. Moreover, PFEM-2 has the chance of obtaining a useful result in only 3 h, employing the largest time-step, thanks to its high robustness; however, while using Eulerian alternatives, that is not possible, i.e., simulation blows-up.

Remark This type of problems where the Reynolds number is not large enough require a finer mesh, but the process that takes very long time makes this methodology very useful. Remember that it is very straightforward to parallelize the spatial coordinates, but it is not so obvious as to how to parallelize the time scale.

4 Mold filling

The casting's mold-filling process is a widely employed technique in material processing engineering. The procedure involves a liquid–gas two-phase flow where a proper study of the interaction between the molten metal and the gas in the complex geometries used as molds is required to optimize the process. In this context, numerical simulations can be used to properly predict the gas-entrapment defects and better understand the complex motions of the gas phase and the liquid phase.

Most of the current mold-filling simulation researches limit to single-phase models where only the molten metal fluid flow is considered, while the role of gas dynamics is completely or largely neglected. Based on such models, gas-entrapment defects are only qualitatively estimated because the coupled effects of gas velocity and pressure, i.e., the back pressure is neglected or inaccurately estimated. Employing two-phase model is a better strategy, but it is still under development due to the complexity for treating the large density and viscosity ratios of the phases involved. A promising work was done by Pang et.al. [31] where they employed the SOLA particle level set method (SOLA-PLSM) to simulate the process. However, the results are not qualitatively correct due to the poor prediction of gas-entrapped regions. Two three-dimensional benchmark tests are reproduced in this Section with the two-phase solver of PFEM-2. Large time-steps such as $\mathrm{CFL}_{\max} \approx 15$ are employed at the interface.

4.1 High-speed water filling into an S-shaped channel

An experiment performed by Schmidt and Klein [30] on a cold chamber die-casting machine is simulated. It consists of a S-shaped channel, initially filled with air, where liquid water is injected at very high velocity (8.7 $\mathrm{m\,s^{-1}}$). A 2D sketch of the geometry is presented in Fig. 6a, with the thickness of the third dimension being 8 (mm). The outlet is explicitly modeled in order to allow the gas to escape. It is very small, and consequently it can be predictable that there will be a large amount of gas-entrapment phenomena inside the mold during water filling.

This process was simulated by Pang et.al. [31] using an algorithm based on the Particle Level Set Method (PSLM) and was compared with the experimental data. Current work presents the results obtained by employing PFEM-2 with two different spatial discretizations: the first one with a coarser mesh which allows using time-steps one order larger and the second with a mesh refinement similar to Pang's. Table 3 reveals the discretization parameters employed for both simulations. The target is to know if our numerical strategy can predict filling evolution and gas entrapping even when we employ cheaper computational resources and spending less time.

Figures 7, 8 and 9 present snapshots at different simulation times comparing experimental and numerical references extracted from the work of Pang et. al. Due to initialization differences, a relative time t^* is employed, setting $t^* = 0$ at

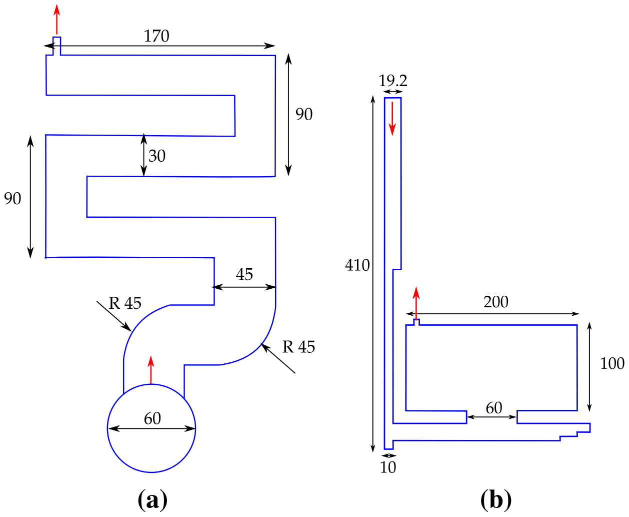


Fig. 6 Mold-filling tests. Sizes are expressed in millimeters. A valve has been included in both geometries to allow the gas to escape. **a** S-shaped test. **b** Campbell's benchmark

Table 3 Tests solved and numerical parameters employed

Name	Δx	Δt	Turbulence model
Test 1	2.5 (mm)	0.25 (ms)	LES
Test 2	0.075 (mm)	0.05 (ms)	LES

the moment of the first snapshots. Subsequent snapshot times are referred to the initial one.

The general filling evolution is well reproduced by both PFEM-2 simulations. As shown by Fig. 7, at $t^* = 0$ (ms), all solutions are in agreement. They obtain a sharp and well-delimited water front. At $t^* = 17.88$ (ms), two entrapped gas bubbles appear in the cavity that are captured by each simulation. However, PFEM-2 results present better agreement regarding the impact of the free-surface against the left wall, where the main water jet is split in two streams. However, at the moment of the impact with the right wall, i.e., $t^* = 32.19$ (ms), (Fig. 9), the overall interface shape of each simulation is still similar to the experiment. PFEM-2 shows some improvements due to the possibility of capturing the backward creeping flow at the outer wall on the first curve. In this last snapshot, the lack of mesh refinement in Test 1 (specially in the front) is responsible for the production of a mixture between air and water with a nonphysical behavior. However, this approximation with a cheaper and less time-consuming simulation (it spends only 10 % of the computing time compared with Test 2) is good enough to represent the overall behavior of the mold filling process.

4.2 Campbell benchmark test

The work of Campbell et. al. [32] presents a well-characterized casting experiment that is a useful benchmark for the modeling community. This case is included in order to strengthen the conclusion that the S-shaped channel brings about the agreement of PFEM-2 solutions with experiments, even when employing large CFL numbers. The model is a plate with dimensional sizes: 200 × 100 × 10 (mm) in X, Y, and Z directions, respectively. The mold cavity is made of resin-bonded sand. The pouring liquid is pure aluminum. The filling process is recorded using the in situ X-ray imaging technology. Since the metal temperature is 720 °C, the density of the aluminum liquid is 2385 $\mathrm{kg\,m^{-3}}$ and the kinematic viscosity is set to be $1.3 \times 10^{-6}\ \mathrm{m^2\,s^{-1}}$. The physical parameters for gas phase is air at room temperature. The average injection velocity, estimated from the gating system and numerical experiences, is 0.4 $\mathrm{m\,s^{-1}}$.

A slice of the simplified geometry of the mold cavity is presented in Fig. 6b. The grid step is uniformly set to be 2.5 (mm), obtaining a mesh with 40 K nodes conforming to 200 K tetrahedra. Regarding the time-step, it is fixed as $\Delta t = 0.01$ (s). Therefore, the simulation reaches a $\mathrm{CFL_{max}} \approx 20$ at the moment the interface enters into the rectangular cavity. With these parameters, simulation takes approximately 35 min to reach the $T_\mathrm{f} = 2$ (s) of simulation time for running on a desktop computer with an Intel i7-2600K processor and 16 Gb RAM. These computing requirements make the simulation with PFEM-2 affordable for any engineering and design office.

Figure 10 compares a snapshot series of three experiments taken from the reference and the simulated interfaces between the molten aluminum and the gas phase. As mentioned before, an average velocity was employed at the inlet because the real pouring condition is not available. Therefore, the first numerical snapshot is taken at the most favorable time, i.e., when qualitatively there is more agreement, and subsequent snapshots are taken relative to this initial time. Under this strategy, comparisons present good overall agreement. Interface does not look as sharp as in experiments, but the filling volume by time is coincident in achieving a solution that can be considered as an accurate and cheap first approach. One can conclude that, even while employing very large time-steps when the interface evolution is computed, the current two-phase flow method can be used to properly predict the evolution of the actual mold-filling process.

5 Failure of a storage vessel

Several industries employ tanks for bulk storage of hazard liquids. Normally, these tanks are surrounded by a wall or earth embankment in order to provide a secondary containment for any spillage from the tank. Although the capacity of the bounded area is large enough to provide full containment of the more likely spills, it is not able to contain the surge of liquid that would follow a catastrophic failure of the

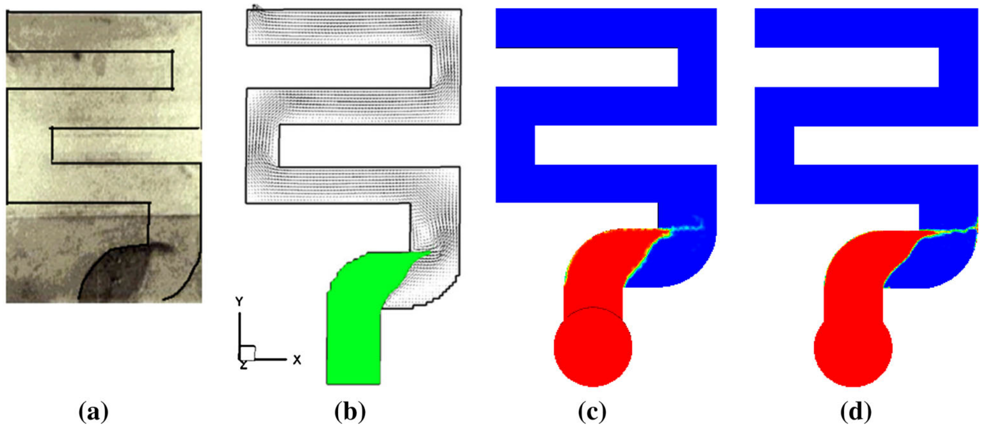


Fig. 7 Interface shape of S-shaped channel filling at $t^* = 0$ (ms). Liquid water is shown in *green* (**b**) and *red* (**c**, **d**). **a** Experimental. **b** Pang et. al. **c** Test 1. **d** Test 2. (Color figure online)

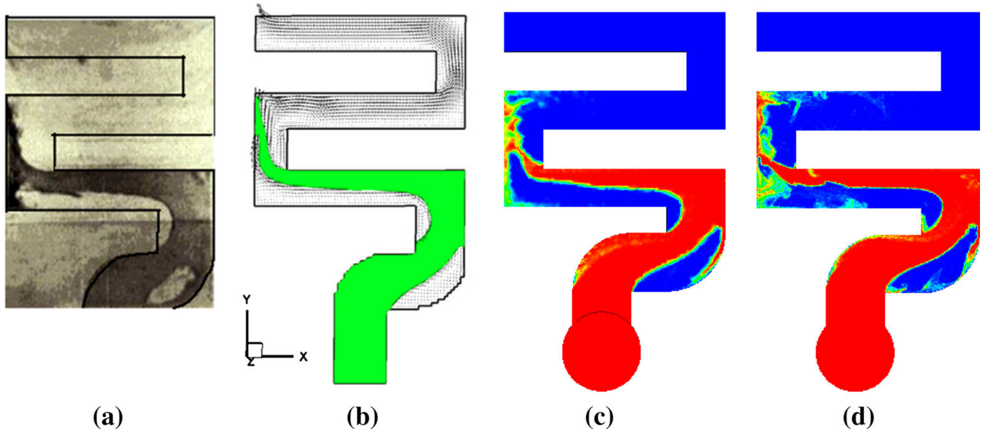


Fig. 8 Interface shape of S-shaped channel filling at $t^* = 17.88$ (ms). Liquid water is shown in *green* (**b**) and *red* (**c**, **d**). **a** Experimental. **b** Pang et. al. **c** Test 1. **d** Test 2. (Color figure online)

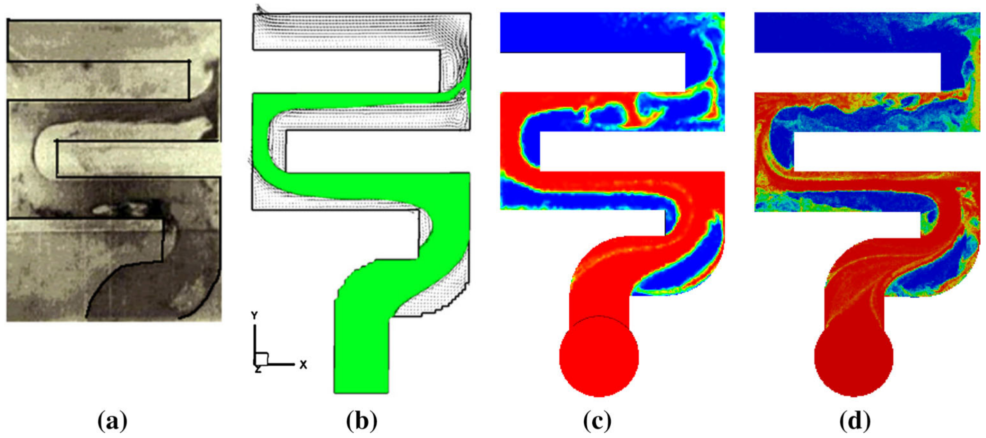
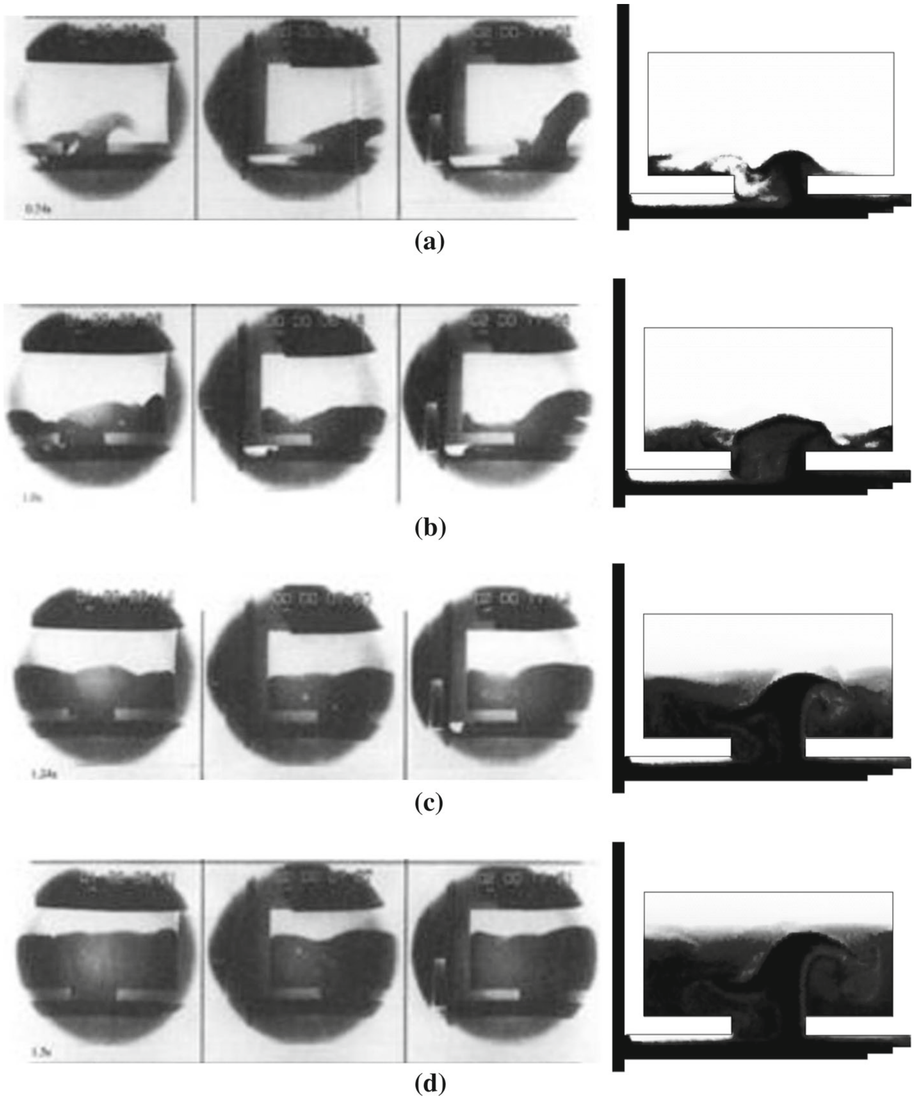


Fig. 9 Interface shape of S-shaped channel filling at $t^* = 32.19$ (ms). Liquid water is shown in *green* (**b**) and *red* (**c**, **d**). **a** Experimental. **b** Pang et. al. **c** Test 1. **d** Test 2. (Color figure online)

tank. Even if the surge does not destroy the bound wall, the flood wave is likely to overtop it. While catastrophic failure of bulk storage tanks is rare, the consequences for site personnel, any local community, and the environment can be severe. Such accidents have occurred in the USA, in Greece, and in Lithuania, and more recently in Argentina among others.

Fig. 10 Progressive filling of the mold. Snapshots from three experiments of [32] (*left side*) are compared with PFEM-2 solution at different times (*right column*). Simulation results are colored from black (aluminum) to *white* (air) in order to compare with the X-ray coloring of the experiments. **a** $t = 0.74$ (s). **b** $t = 1.0$ (s). **c** $t = 1.24$ (s). **d** $t = 1.5$ (s)



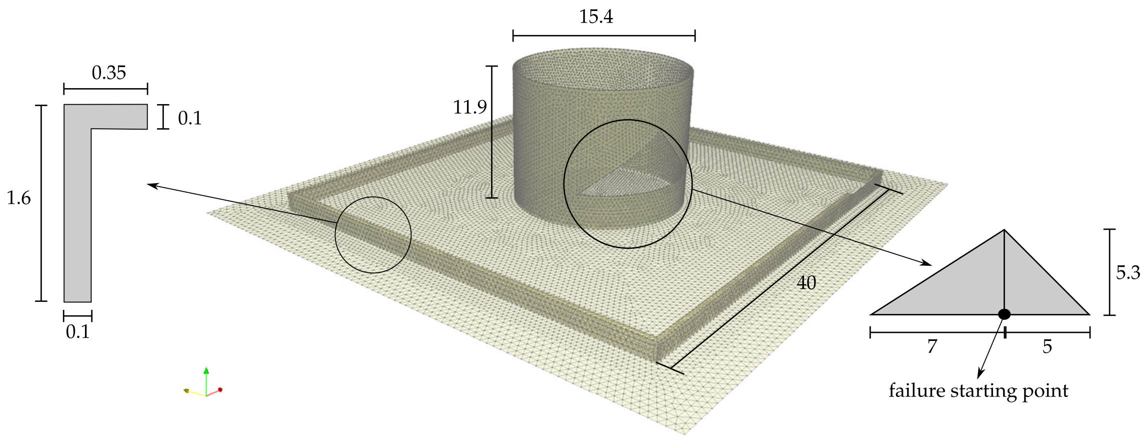


Fig. 11 Some surface patches of the 3D mesh employed for the tank collapse test

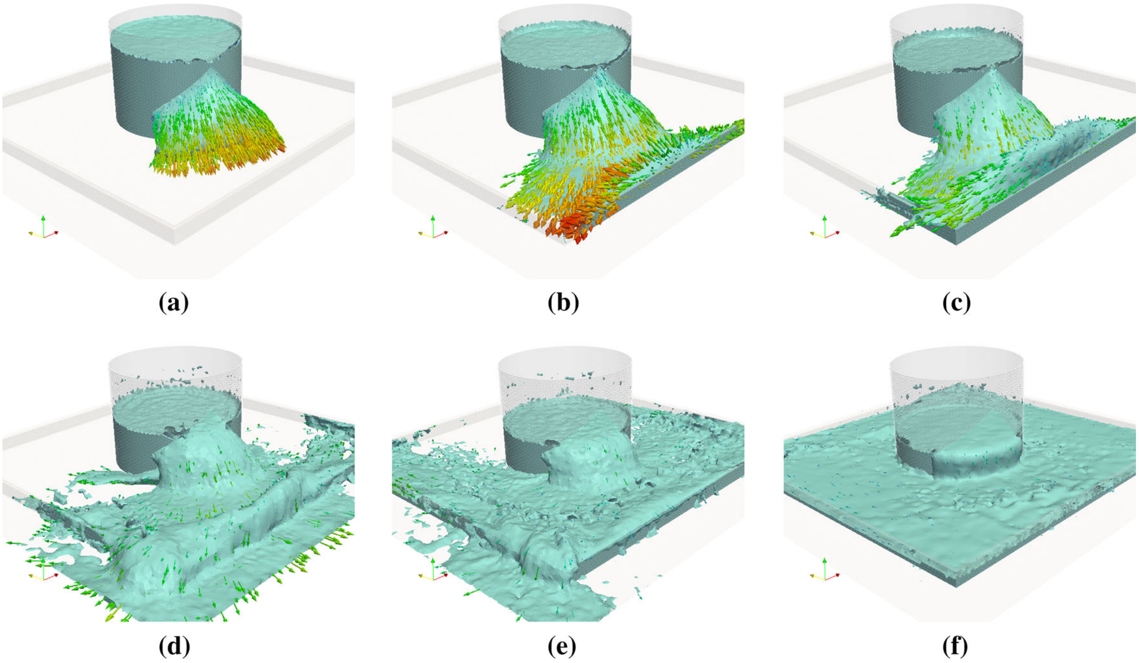


Fig. 12 Evolution of the collapse observed from the first impacted corner. Isocontour of $\lambda = 0$. Vectors depict velocity field, being scaled and colored according to velocity magnitude. **a** 1 s. **b** 2 s. **c** 3 s. **d** 6 s. **e** 10 s. **f** 20 s

A relevant experimental study was done by Atherton [33] which performed simulations of catastrophic failure of a storage tank, covering a comprehensive range of tank and bound arrangements, and to measure both the dynamic pressures that are exerted on the bounding walls and the quantity of liquid that overtops it. However, the cited work supposes a instantaneous disappearance of the tank wall during the failure, and the models lack of breakwaters at the top of the bound wall which are usually utilized to diminish the amount of liquid that overtops the secondary container. To overcome these missing features, numerical simulations appear as a cheap and accurate complement to experimental measurements. Computational models can be employed to simulate different failure types and test the efficiencies of several options of breakwaters (measured in minimizing the amount of overtopped liquid), among other possibilities. Due to the catastrophic nature of the failure events, the flow regime is dominated by inertia. As mentioned in previous works [10,11], this type of situations are suitable to be solved by employing a numerical method like PFEM-2.

The study case presented in this section corresponds to the catastrophic collapse of a tank containing a water–oil mixture. The target is the fluid-dynamic evaluation of the mixture during the surge and the collision with the bounding walls. Neither details related to failure causes nor structural aspects of failure propagation are objectives of this simulation.

The PFEM-2 two-phase model is employed, considering an isothermal and turbulent flow. The fluid initially fills the 90 % of the tank and is considered as a mixture of 3/4 part of water and 1/4 part of oil, with a density of 925 ($\mathrm{Kg\,m^{-3}}$) and dynamic viscosity of 0.034225 ($\mathrm{Kg\,m^{-1}\,s^{-1}}$). The rest of the domain is considered filled with air at 15 °C. Surface tension is neglected.

Figure 11 shows the geometric features of the tank and the surrounding walls. Also, this model includes breakwaters over the walls shape of which is presented in the said figure. A relevant characteristic of the simulation is the failure model adopted, which was based on fractomechanical studies done upon the real tank failure. It assumes that the failure started at some point, and it is propagated erratically toward the top, following a path described by the heat-affected zone of the welding toward both sides. Considering an instantaneous propagation of the failure, simulation uses a fixed triangular opening of the tank wall from the initial time. Its dimensions and shape, depicted in Fig. 11, pertain to a model supplied by the mechanical department of the company interested in analyzing this accident.

The used mesh consists of 312,251 nodes conforming to 1,724,389 tetrahedral. The average size of the elements is $\Delta x = 0.3$ (m), and the time-step is set constant with a value of $\Delta t = 0.05$ (s), leading to a $\mathrm{CFL_{max}} = 10$ which occurs at the moment of the splashing. An Intel(R) Core(TM) i7-3930K CPU @ 3.20GHz computer was employed for the parallel simulation (6 cores). It has taken around 5 h to complete 20 s of simulation time, which is enough to drain the tank.

Figure 12 presents several snapshots of the free-surface ($\lambda = 0$) observed from one corner. It can be seen clearly that the first surge overtops the wall in spite of the use of

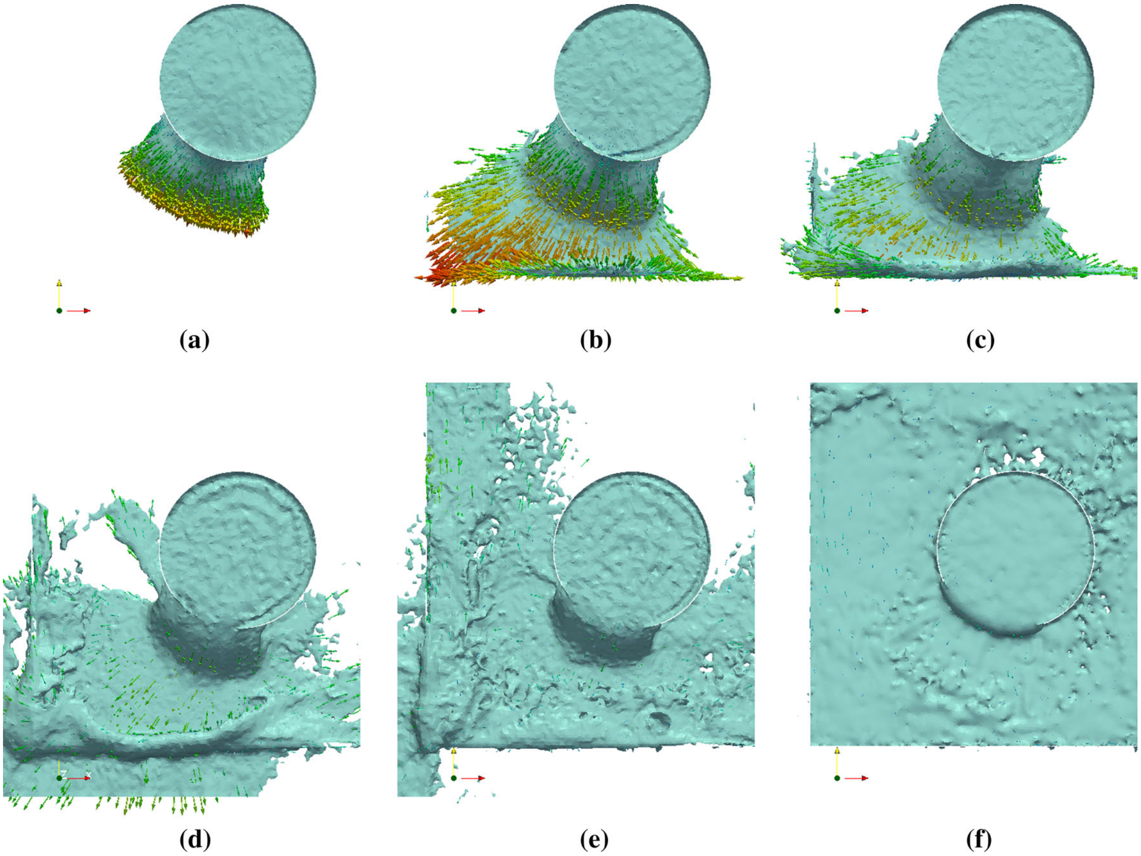


Fig. 13 Evolution of the collapse observed from the *top*. Isocontour of $\lambda = 0$. Vectors depict velocity field, being scaled and colored according to velocity magnitude. **a** 1 s. **b** 2 s. **c** 3 s. **d** 6 s. **e** 10 s. **f** 20 s

breakwaters (see Fig. 12d), leading to a calculated loss of around 20 % of the initial inventory. Secondary splashes are well contained by the structure. Maximum forces exerted by the fluid are also detected at the moment of the first surge impact the wall. The horizontal component detected at the first-impacted wall is 2000 kgf while the breakwater receives around 6000 kgf of vertical force. These values justify consequences experienced after the accident according to the company report.

Figure 13 presents a sequence of snapshots observed from top. This point of view demonstrates the need of using breakwaters to minimize liquid losses and also allows to examine how the secondary container bounded by walls is quickly filled by the liquid.

6 Conclusions

From previous works, PFEM-2 has demonstrated to be a promising option for engineers and scientists when a good compromise between efficiency and accuracy for CFD problems is required. This hybrid Lagrangian–Eulerian strategy found advantages when advection plays a main role: convection-dominant problems, or in general when the transport of a quantity (tracer, reactive, or passive species; temperature, momentum, free-surface, and so on) has an important advective component. This is justified because there is no need for employing very fine mesh like those found when using an Eulerian formulation to avoid excessive numerical diffusion and also when there is a strong need of enlarging the time-step to afford very long real-time simulations requiring normally in the order of hours of real time. This kind of situations is typically found in most of the industrial applications, and in the current work, several representative problems have been simulated with PFEM-2 obtaining good agreement with experimental references even while employing large time-steps. Among the more auspicious situations, the calculations of the RTDs in tanks appear to be very attractive, because they largely exploit the advantages of PFEM: its ability to resolve a long-time interval without the need of having a very fine mesh. The enlargement of the time-step and coarsening of the mesh are two advantages of PFEM against any Eulerian-based solver which shows instabilities with high Courant numbers and inaccuracies due to excessive numerical dissipation when using coarse mesh.

Different strategies for modeling turbulent dispersion are also presented. If the flow is very unsteady, other techniques such as particle tracking cannot be employed ignoring the necessity of coupling the flow computing. In this context, the utility of PFEM-2 to manage both situations (steady and unsteady flows) is another strong point that is presented for the case of steady flow in a contact tank and troubleshooting detection in unsteady flow in a skimmer tank.

For the case of two-phase flows, the advantages of PFEM-2 are extended due to its capability for using time-steps larger than $\mathrm{CFL}_{\max} \gg 1$ at the interface. Such kind of flows appears in real-time industry, and in this work, some representative cases were successfully simulated.

Two mold-filling process were computed where, despite using coarser meshes and larger time-steps than those used with other numerical alternatives, PFEM-2 has demonstrated that its solution reproduces the experiment even better than its competitors. Finally, the last test, concerned with the collapse of a container, allows to show again the capabilities of PFEM-2 to deal with other interesting engineering problems, like industrial accidents. Useful results such as forces over contentions and inventory losses inside them were obtained justifying the consequences experienced after the accident by the company interested in this analysis.

In conclusion, this work aims to show the feasibility of application of the method PFEM-2 to cases of industrial interest. After several publications where its capabilities were demonstrated in terms of accuracy, robustness and efficiency, here a series of interesting examples of real-life applications are presented, showing that the method is not only an attractive numerical method for research, but also a valuable tool for design engineers at the moment of taking decisions, in particular, in such cases that require long real-time simulations.

Acknowledgments The authors wish to offer their thanks to the CONICET, the Universidad Nacional del Litoral, and the ANPCyT for their financial supports through Grants PIP-2012 GI 11220110100331, CAI+D 2011 501 201101 00435 LI, and PICT-2013 0830.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Donea J, Huerta A (2003) Finite element method for flow problems. Wiley, Chichester
2. Gingold RA, Monaghan JJ (1977) Smoothed particle hydrodynamics, theory and application to non-spherical stars. R Astron Soc 181:375–389
3. Monaghan JJ (1988) An introduction to SPH. Comput Phys Commun 48:89–96
4. Harlow FH (1955) A machine calculation method for hydrodynamic problems. Los Alamos Scientific Laboratory Report LAMS-1956
5. Harlow FH, Welch J (1965) Numerical calculation of time dependent viscous incompressible flow of fluid with free surface. Phys Fluids 8(12):2182–2189
6. Wieckowsky Z (2004) The material point method in large strain engineering problems. Comput Methods Appl Mech Eng 193(39):4417–4438
7. 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 Numer Methods 61:964–989
8. Idelsohn SR, Nigro NM, Limache A, Oñate E (2012) Large time-step explicit integration method for solving problems with dominant convection. Comput Methods Appl Mech Eng 217–220:168–185
9. Idelsohn SR, Nigro NM, Gimenez JM, Rossi R, Marti J (2013) A fast and accurate method to solve the incompressible Navier–Stokes equations. Eng Comput 30(2):197–222
10. Gimenez JM (2015) Enlarging time-steps for solving one and two phase flows using the particle finite element method. Ph.D. Thesis, Universidad Nacional del Litoral, Santa Fe, Argentina
11. Idelsohn S, Oñate E, Nigro N, Becker P, Gimenez JM (2015) Lagrangian versus Eulerian integration errors. Comput Methods Appl Mech Eng 293:191–206
12. Gimenez JM (2014) Implementacin del mtodo PFEM sobre arquitecturas paralelas, Facultad de Ingeniería y Ciencias Hídricas, Centro de Investigación de Métodos Computacionales, Universidad Nacional del Litoral
13. Gimenez JM, Nigro NM, Idelsohn SR (2014) Evaluating the performance of the particle finite element method in parallel architectures. J Comput Part Mech 1(1):103–116
14. Idelsohn SR, Marti J, Becker P, Oñate E (2014) Analysis of multifluid flows with large time steps using the particle finite element method. Int J Numer Methods Fluids 75(9):621–644
15. Gimenez JM, Gonzlez LM (2015) An extended validation of the last generation of particle finite element method for free surface flows. J Comput Phys 284:186–205
16. Becker P, Idelsohn SR, Oñate E (2014) A unified monolithic approach for multi-fluid flows and fluid-structure interaction using the particle finite element method with fixed mesh. Comput Mech 55(6):1091–1104
17. Gimenez JM, Nigro N, Oñate E, Idelsohn S (2016) Surface tension problems solved with the particle finite element method using large time-steps. Comput Fluids
18. Chorin A (1968) Numerical solution of the Navier–Stokes equations. Math Comput 22:745–762
19. Thomson DJ (1988) Random walk models of turbulent dispersion. Ph.D. Thesis, Department of Mathematics and Statistics, Brunel University
20. Le Moullec Y, Potier O, Gentric C (2008) Flow field and residence time distribution simulation of a cross-flow gasliquid wastewater treatment reactor using CFD. Chem Eng Sci 63:2436–2449
21. Fabbroni N (2009) Numerical simulations of passive tracers dispersion in the sea. Ph.D. Thesis, Alma Mater Studiorum Università di Bologna
22. Graham DI, James PW (1996) Turbulent dispersion of particles using eddy interaction models. Int J Multiph Flow 22:157175
23. Gimenez JM, Nigro NM, Idelsohn SR (2012) Improvements to solve diffusion-dominant problems with PFEM-2. Mecánica Comput 31:137–155
24. Hryb D, Cardozo M, Ferro S, Goldschmit M (2009) Particle transport in turbulent flow using both Lagrangian and Eulerian formulations. Int Commun Heat Mass Transf 36(5):451–457

25. Gualtieri C (2006) Numerical simulation of flow and tracer transport in a disinfection contact tank. In: Conference: third biennial meeting: international congress on environmental modelling and software (iEMSs 2006)
26. Shiono K, Teixeira EC (2000) Turbulent characteristics in a baffled contact tank. J Hydraul Res 38(6):403–416
27. Wilson J, Venayagamoorthy S (2010) Evaluation of hydraulic efficiency of disinfection systems based on residence time distribution curves. Environ Sci Technol 44(24):9377–9382
28. Smagorisnky J (1963) General circulation experiments with the primitive equations. Mon Weather Rev 91:99–164
29. Norberto N, Gimenez JM, Idelsohn S (2014) Recent advances in the particle finite element method towards more complex fluid flow applications. Numer Simul Coupled Probl Eng 33:267–318
30. Schmid M, Klein F (1995) Fluid flow in die-cavities: experiments and numerical simulation. In: The 18th international die casting congress and exposition, Indianapolis, Indiana, USA, p 9397
31. Pang S, Chen L, Zhang M, Yin Y, Chen T, Zhou J, Liao D (2010) Numerical simulation two phase flows of casting filling process using SOLA particle level set method. Appl Math Model 34:4106–4122
32. Sirrell B, Holliday M, Campbell J (1996) Benchmark testing the flow and solidification modeling of AI castings. JOM 48(3):20–23
33. Atherton W (2005) An experimental investigation of bund wall overtopping and dynamic pressures on the bund wall following catastrophic failure of a storage vessel. Research Report, Liverpool John Moores University, England
34. Codina R (2001) Pressure stability in fractional step finite element methods for incompressible flows. J Comput Phys 170:112140
35. Becker P, Idelsohn S, Oñate E (2015) A unified monolithic approach for multi-fluid flows and fluid-structure interaction using the particle finite element method with fixed mesh. Comput Mech 55(6):10911104