

Numerical Comparison of the Particle Finite Element Method Against an Eulerian Formulation

Juan M. Gimenez, Pedro Morin, Norberto Nigro, and Sergio Idelsohn

Abstract The main goal of this paper is to validate experimentally the principal conclusions previously published in [17]. Two manufactured test cases were considered with their respective analytic solutions. First, a scalar transport equation is considered written in such a way that several parameters are included to stress the limiting situation where the Eulerian and the Lagrangian approaches behave better. The results show conditions to be fulfilled in order to choose between both formulations, according to the problem parameters. A brief discussion about the projection needed for PFEM-2 method is included, specially due to its impact on the error convergence rate. Lately, an extension to Navier-Stokes equations is introduced using also a manufactured case to verify again the same conclusions. This paper intends to establish the first steps towards a mathematical error analysis for the particle finite element method which supports the preliminary theoretical and experimental results presented here.

J.M. Gimenez • N. Nigro

Centro de Investigación de Métodos Computacionales, CIMEC-UNL/CONICET, Predio CONICET Santa Fe, Colectora Ruta Nac 168, Km 472, Paraje El Pozo, 3000 Santa Fe, Argentina
e-mail: jmarcelogimenez@gmail.com
<http://www.cimec.org.ar>

Facultad de Ingeniería y Ciencias Hídricas - Universidad Nacional del Litoral, Ciudad Universitaria, Paraje “El Pozo”, Santa Fe, Argentina
<http://www.fich.unl.edu.ar>

P. Morin

Instituto de Matemática del Litoral, IMAL-UNL/CONICET, Predio CONICET Santa Fe, Colectora Ruta Nac 168, Km 472, Paraje El Pozo, 3000 Santa Fe, Argentina
<http://www.imal.santafe-conicet.gov.ar>

S. Idelsohn (✉)

International Center for Numerical Methods in Engineering. CIMNE Edificio C1, Campus Norte UPC, C/ Gran Capitán S/N, 08034 Barcelona, Spain
<http://www.cimne.com>

ICREA, Institució Catalana de Recerca i Estudis Avançats (ICREA), Barcelona, Spain
e-mail: sergio@cimne.upc.edu

1 Introduction

Over the last decades, computer simulation of incompressible fluid flow has been mainly based on the Eulerian formulation of the fluid mechanics equations on fixed domains [3]. During this period, hardware has evolved considerably increasing the speed performance of computations and allowing better facilities for data entry and the display of results. However in these decades there have been no substantial improvements on the numerical methods used concerning the efficiency of the algorithm. In most practical engineering problems, very fine mesh and very small time-steps are needed to reach acceptable results. This handicap exceeds most time the efficiency of current powerful computers.

More recently, particle-based methods in which each particle is followed in a Lagrangian manner have been used for fluid flow problems. Monaghan [21] proposed the first ideas for the treatment of astrophysical hydrodynamic problems with the so-called Smoothed Particle Hydrodynamics Method (SPH), which was later generalized to fluid mechanics problems [6, 7, 21]. Koshizuka and coworkers [18, 19] developed a similar method to SPH, named Moving Particle Simulation (MPS). SPH and MPS belong to the family of the so-called meshless methods, as well as the Finite Point Method [24–26]. Lately, the meshless ideas were generalized to take into account the finite element type approximations in order to obtain more accurate solutions [9]. This method was called the Meshless Finite Element Method (MFEM) and uses the extended Delaunay tessellation [8] to build the mesh in a computing time, which is linear with the number of nodal points. A natural evolution of the last work was the Particle Finite Element Method (PFEM) [10]. The PFEM combines the particle precept with the Finite Element Method (FEM) shape functions using a background finite element mesh. This mesh may be quickly rebuilt at each time-step (PFEM with moving mesh) or may be a fixed mesh (PFEM with fixed mesh). In the last case, the results from the Lagrangian particles are projected on a fixed mesh at each time-step. The idea of combining fixed meshes with moving particles is not new. It was introduced for convection-diffusion problems in [22] and was used in the so-called Particle in Cell method (PIC) [2] and later in its extension called the Material Point Method (MPM) [29]. All these methods use a Finite Element (FE) background mesh. Despite that both the PFEM and the MPM use a fixed FE mesh and a set of Lagrangian particles, there are important differences in the way the particles are employed: thus, while in the MPM all computations are performed on the mesh, in the PFEM the aim is to calculate as much as possible on the particles, leaving small corrections to be performed on the mesh. However, the most important difference is that in the PFEM the particles do not represent a fixed amount of mass, but rather material points that transport only intrinsic properties of the fluid. This allows to use a variable number of particles and therefore simplifying refinement.

The PFEM has been successfully used to solve the Navier-Stokes equations [1, 20] and fluid–structure interaction problems [11, 12, 28] as well as solid

mechanics problems [23]. The advantages of the PFEM concerning the tracking of internal interfaces have also been explored and used to solve fluid mechanics problems including multi-fluid flows [13].

The possibility to use the PFEM to solve non-linear problems with large time-steps in order to obtain an accurate and fast solution was successfully explored by the authors for the solution of the homogeneous incompressible Navier-Stokes equations [5, 14, 15] and multi-fluid problems [4, 16]. This new strategy was named PFEM-2.

In [17] the first trial of an error analysis for Lagrangian based methods like PFEM and its comparison with an Eulerian formulation was presented in order to demonstrate why the former is more accurate than the latter in certain particular cases when large time-step and/or coarse meshes were used. There the authors claim that nowadays, the best way to improve the efficiency of the algorithms in order to take advantages of the increasing computer power is using this particle-based method.

The goal of this paper is to demonstrate numerically the validity of that first error analysis attempt for scalar transport problems and also for homogeneous fluid flow problems. During this analysis the projection error arises as one of the main limitations to reach that goal and a new proposal of projection algorithm is presented. This projection shows to have several collateral benefits like having good mathematical properties as the existence of reciprocity in the operation of going back and forth between mesh and particles and also smoothing properties in the computation of fluid flow forces.

The layout of the paper is the following. After a review of the main results presented in [17] a numerical validation of these results for scalar transport problems is shown highlighting the conditions under which a Lagrangian approximation is preferred, in particular when using large time-steps. Next, the new projection scheme is presented putting in evidence the possibility and the requirement of reaching a second order approximation in space and time. Finally, an extension to fluid flow problems is presented with a manufactured example that serves to demonstrate the conclusions written in [17]. Some conclusions and future trends are highlighted at the end.

2 Error Analysis Applied to a Scalar Transport Equation

Many problems in engineering may be mathematically expressed by transport equations written in Lagrangian or Eulerian reference frames. For such mathematical models there is a strong division between transporting scalar fields or transporting vector or tensor fields. Also the linear and non-linear approaches become another important feature normally included in the modeling. Moreover, advection and diffusion effects present very different behaviors from the physical, mathematical, and numerical point of view. With such a complexity it is very difficult to find analytic solutions to be used as reference for formal error analysis. In order to

circumvent such a drawback in this paper manufactured test cases are developed to be used as good candidates for getting exact solutions to our test problems.

Several control parameters for the exact solution are used to place the problem in various conditions from some beneficial to Lagrangian schemes to other more beneficial to an Eulerian one.

To begin, in this section a scalar transport equation is solved. The differential equation associated to solve this problem may be written in an Eulerian (fixed) frame as

$$\frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T = Q(\mathbf{x}) \quad (1)$$

where $T = T(\mathbf{x}, t) = T^t(\mathbf{x})$ is the scalar unknown, \mathbf{v} is the velocity field, and $Q(\mathbf{x})$ includes the source term $q(\mathbf{x})$, the diffusion term $\nabla \cdot (k \nabla T)$, and the linearized version of a reactive source term cT , etc., being

$$Q(\mathbf{x}) = q(\mathbf{x}) + \nabla \cdot (k \nabla T) - cT + \dots \quad (2)$$

To write Eq. 1 in the Lagrangian (mobile) frame, it is necessary to use the material derivative D/Dt , which condenses the temporal derivative and the convective term into a unique term. However, as it is well known for Lagrangian formulations not only the respective field should be computed, but also the particle trajectories, obtaining

$$\begin{cases} \frac{DT}{Dt} = Q(\mathbf{x}_p) \\ \frac{D\mathbf{x}_p}{Dt} = \mathbf{v} \end{cases} \quad (3)$$

where $T = T(\mathbf{x}_p^t, t) = T^t(\mathbf{x}_p^t)$ is the scalar unknown. Here the superscript t means the time dependency and the subscript p represents the particle itself.

Eqns. (1) and (3) are integrated in time in an Eulerian frame using a linear method named θ -method:

$$T^{n+1}(\mathbf{x}) = T^n(\mathbf{x}) + \int_n^{n+1} (Q^t - \mathbf{v}^t \nabla T^t) dt \approx T^n(\mathbf{x}) + (Q - \mathbf{v} \nabla T)^{n+\theta} \Delta t \quad (4)$$

where $f^{n+\theta} = (1-\theta)f^n + \theta f^{n+1}$ is the linear in time interpolation for any function f . The time integration error of a function f^t for $\theta = 1/2$ is proportional to the second derivative of the function and Δt^2 , i.e.,

$$\int_n^{n+1} f^t dt \approx f^{n+\frac{1}{2}} \Delta t \pm \epsilon \quad \text{with } \epsilon = \mathcal{O}\left(\frac{\partial^2 f}{\partial t^2} \Delta t^2\right) \Delta t. \quad (5)$$

In the case of time integration in the Lagrangian frame, there are several options to evaluate that integral [17]. An option is to perform the integration with the moving particles following the streamlines (taking into account intermediate positions), splitting the integration into an explicit part and an implicit part, or decoupling the trajectory and temperature integrations employing two different values for θ , i.e.,

$$\begin{cases} T^{n+1}(\mathbf{x}_p^{n+1}) \approx T^n(\mathbf{x}_p^n) + (1 - \theta_1) \int_n^{n+1} Q^n(\mathbf{x}_p^t) dt + \theta_1 Q^{n+1}(\mathbf{x}_p^{n+1}) \Delta t \\ \mathbf{x}_p^{n+1} \approx \mathbf{x}_p^n + (1 - \theta_2) \int_n^{n+1} \mathbf{v}^n(\mathbf{x}_p^t) dt + \theta_2 \mathbf{v}^{n+1}(\mathbf{x}_p^{n+1}) \Delta t \end{cases} \quad (6)$$

The case of performing the unknown integration following streamlines with $\theta_1 = 1/2$, and the trajectory, also following streamlines, but with $\theta_2 = 0$, has been named X-IVAS+implicit correction [15]. On the other hand, if the choice for the unknown integration is $\theta_1 = 1$ the method is named X-IVS, which was presented in [14]. The latter integration will be used in all the examples presented here when using a Lagrangian frame. However, in the present study for the error analysis, the standard θ -method will be considered:

$$\begin{cases} T^{n+1}(\mathbf{x}_p^{n+1}) \approx T^n(\mathbf{x}_p^n) + Q^{n+\theta}(\mathbf{x}_p^{n+\theta}) \Delta t \\ \mathbf{x}_p^{n+1} \approx \mathbf{x}_p^n + \mathbf{v}^{n+\theta}(\mathbf{x}_p^{n+\theta}) \Delta t \end{cases} \quad (7)$$

This assumption is based on the fact that the X-IVAS integration improves the results and decreases the integration errors. The evaluation of this difference is out of the scope of this work.

2.1 Eulerian Errors

As presented in Equation (5), the linear θ -method introduces a minimum error for $\theta = 1/2$ which is proportional to the second derivative of the integrated function and the square of the time-step, i.e.,

$$T^{n+1}(\mathbf{x}) = T^n(\mathbf{x}) + \int_n^{n+1} (Q^t - \mathbf{v}^t \nabla T^t) dt \approx T^n(\mathbf{x}) + \left[(Q - \mathbf{v} \nabla T)^{n+\frac{1}{2}} \right] \Delta t \pm \epsilon_t \quad (8)$$

with the time integration error

$$\epsilon_t = \mathcal{O}\left\langle \frac{\partial^2}{\partial t^2} [Q - \mathbf{v}\nabla T] \Delta t^2 \right\rangle \Delta t = \mathcal{O}\langle [Q - \mathbf{v}\nabla T] \Delta t^2 \rangle \Delta t. \quad (9)$$

On the other hand, the FEM approximation of the functions and the space derivatives introduces spatial errors. As it was mentioned, this analysis only considers linear finite element approximations of the unknown, therefore the spatial errors are proportional to the second derivative of the functions and the square of the mesh-size, i.e.,

$$T^{n+1}(\mathbf{x}) = T^n(\mathbf{x}) + \int_n^{n+1} (Q^t - \mathbf{v}^t \nabla T^t \pm \epsilon_x) dt \approx T^n(\mathbf{x}) + \left[(Q - \mathbf{v}\nabla T)^{n+\frac{1}{2}} \Delta t \pm \epsilon_x \right] \Delta t \pm \epsilon_t \quad (10)$$

with the spatial discretization error

$$\epsilon_x = \mathcal{O}\left\langle \frac{\partial^2}{\partial \mathbf{x}^2} [Q - \mathbf{v}\nabla T] \Delta x^2 \right\rangle \Delta t. \quad (11)$$

Finally, avoiding higher order terms, the unknown function after a time-step using the Eulerian framework is

$$T^{n+1}(\mathbf{x}) = T^n(\mathbf{x}) + (Q - \mathbf{v}\nabla T)^{n+\frac{1}{2}} \Delta t \pm \epsilon_x + \epsilon_t = T^n(\mathbf{x}) + (Q - \mathbf{v}\nabla T)^{n+\frac{1}{2}} \Delta t \pm \epsilon^E \quad (12)$$

with $\epsilon^E = \epsilon_x + \epsilon_t$.

2.2 Lagrangian Errors

In this section first a general analysis taking into account the whole problem involved in a Lagrangian formulation, i.e., solving not only for the scalar field, but also for the particle trajectories, is done. Afterwards a specific analysis of projection errors for PFEM-2 scheme due to the mapping of the field between particles and the mesh is presented considering the importance of this projection stage in the global error of the method.

2.2.1 General Analysis

In the case of the Lagrangian frame, the errors in the numerical evaluation of the unknown function and the particle position are

$$\begin{cases} T^{n+1}(\mathbf{x}_p^{n+1}) \approx T^n(\mathbf{x}_p^n) + Q^{n+\frac{1}{2}}(\mathbf{x}_p^{n+\frac{1}{2}})\Delta t \pm \mathcal{O}\langle Q'' \Delta x^2 \rangle \Delta t \pm \mathcal{O}\langle Q'' \Delta t^2 \rangle \Delta t \\ \mathbf{x}_p^{n+1} \approx \mathbf{x}_p^n + \mathbf{v}^{n+\frac{1}{2}}(\mathbf{x}_p^{n+\frac{1}{2}}) \Delta t \pm \mathcal{O}\langle \mathbf{v}'' \Delta x^2 \rangle \Delta t \pm \mathcal{O}\langle \mathbf{v}'' \Delta t^2 \rangle \Delta t \end{cases} \quad (13)$$

The error in the evaluation of the particle position $\epsilon_x^p = \mathcal{O}\langle \mathbf{v}'' \Delta x^2 \rangle \Delta t \pm \mathcal{O}\langle \mathbf{v}'' \Delta t^2 \rangle \Delta t$ introduces also an error in the evaluation of the unknown function. Performing a series expansion around \mathbf{x}_p^{n+1} , i.e.,

$$T^{n+1}(\mathbf{x}_p^{n+1} + \epsilon_x^p) = T^{n+1}(\mathbf{x}_p^{n+1}) + \nabla T \epsilon_x^p + \mathcal{O}\langle T'' (\epsilon_x^p)^2 \rangle \quad (14)$$

and replacing in the first equation of (13), it is

$$T^{n+1}(\mathbf{x}_p^{n+1}) \approx T^n(\mathbf{x}_p^n) + Q^{n+\frac{1}{2}}(\mathbf{x}_p^{n+\frac{1}{2}})\Delta t \pm \mathcal{O}\langle Q'' \Delta x^2 \rangle \Delta t \pm \mathcal{O}\langle Q'' \Delta t^2 \rangle \Delta t \pm \nabla T \epsilon_x^p \quad (15)$$

the expression for the Lagrangian error ϵ^L can be found

$$\epsilon^L = \pm \mathcal{O}\langle (Q'' \pm \mathbf{v}'' \nabla T) \Delta x^2 \rangle \Delta t \pm \mathcal{O}\langle (Q'' \pm \mathbf{v}'' \nabla T) \Delta t^2 \rangle \Delta t \quad (16)$$

Comparing the Eulerian error expression in 12 with the Lagrangian one in 16, the main differences are in the following terms:

$$\underbrace{[\mathbf{v} \nabla T]'' \Delta x^2 \pm [\mathbf{v} \nabla T]'' \Delta t^2}_{\text{Eulerian}} \neq \underbrace{\mathbf{v}'' \nabla T \Delta x^2 \pm \mathbf{v}'' \nabla T \Delta t^2}_{\text{Lagrangian}} \quad (17)$$

The difference presented in Equation (17) leads to a big advantage of the Lagrangian framework against its Eulerian counterpart for some problems. For instance, in the standard convection-diffusion problem of a non-constant unknown, where the convective field is known and has a constant or nearly constant velocity, Equation (17) reads

$$\underbrace{[\mathbf{v} \nabla T]'' \Delta x^2 \pm [\mathbf{v} \nabla T]'' \Delta t^2}_{\text{Eulerian}} \neq 0 \quad ; \quad \underbrace{\mathbf{v}'' \nabla T \Delta x^2 \pm \mathbf{v}'' \nabla T \Delta t^2}_{\text{Lagrangian}} = 0 \quad (18)$$

that is, the Lagrangian integration does not have any error because $\mathbf{v}'' = \mathbf{v}'' = 0$, but the Eulerian framework presents an error due to $[\mathbf{v} \nabla T]'' \pm [\mathbf{v} \nabla T]'' = \mathbf{v} [\nabla T]'' \pm \mathbf{v} [\nabla T]'' \neq 0$.

2.2.2 Projection Errors

In order to complete the error analysis for the particular Lagrangian approximation employed by PFEM-2, the projection errors should be taken into account. In [15] PFEM-2 had been introduced as a hybrid method using particles and mesh in order to exploit the advantages of Lagrangian and Eulerian methods. This duality

between mesh and particles requires that data should be sent back and forth along the whole computation several times. If this projection is not designed properly a huge source of inaccuracies is introduced and the error degrades a lot. Moreover, the projection is normally responsible of getting noisy values represented by the mesh, like forces in a fluid–structure interaction problems. In [17] Idelsohn et al. conclude that these errors depend on the distance between particles which is denoted by h . Therefore, the importance to use a great amount of particles with $h \ll \Delta x$ becomes obvious for decreasing projection errors. However this assumption relies on an ideal projection strategy which consists in generating a mesh with the particles positions guaranteeing that the projection error depends only on particle mesh-size h .

Even being an ideal strategy, the triangulation of the particle positions at each time-step may be unaffordable because of the computational cost involved when particles tend to grow in number.

Typical strategies employed by PFEM-2 [4] to project a given field ϕ between nodes using subindices (j) and particles using subindices (p) are of the following form:

$$\phi_j = \Pi_{p \rightarrow j} \phi_p = \frac{\sum_P \phi_p W_j(\mathbf{x}_p)}{\sum_P W_j(\mathbf{x}_p)} \quad (19)$$

where the function W_j , associated with the node j , can be either the typical kernel functions used in particle methods such as SPH[7] or the linear shape functions raised to a power $\alpha > 0$ (it is $W_j(\mathbf{x}) = N_j(\mathbf{x})^\alpha$), while \mathbf{x}_p is the position of the particle p with state ϕ_p and P is the number of particles in a region around the node j .

This projection algorithm does not preserve the global second order error introducing a degradation of the solution accuracy proportional to the number of projections done. This last variable depends on the time-step with a smaller impact when the time step is large and a bigger one when the time-step tends to reduce. This fact is observed experimentally in the next section.

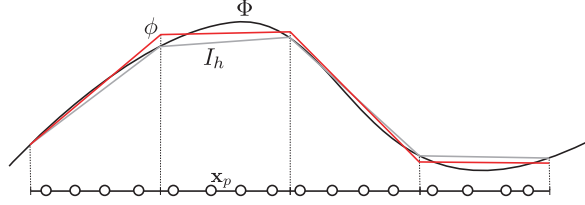
On the other hand this family of projections algorithms is in general not commutative, i.e., the projection does not satisfy the inverse property of operators:

$$\Pi_{p \rightarrow j} \left(\Pi_{p \rightarrow j} \right)^{-1} \neq \mathbf{I} \quad (20)$$

This last statement strongly endangers the solution accuracy and may also be responsible for excessive diffusion of the primal variables or for noisy secondary (dual) fields computed on the mesh, like forces, heat fluxes, etc.

In order to circumvent this drawback in this paper a new projection is presented overcoming most of the above cited weakness. Here only its definition and the main idea behind this development are presented. The mathematical demonstration of its incidence on the error analysis is still in elaboration.

Fig. 1 Graphical example of the target function Φ and its two possible FEM approximations ϕ and I_h



The idea is based on using least squares approximations to find the nodal values from particles states.

More precisely, the idea is to find ϕ_j that realizes the following minimum:

$$\min \left[\frac{1}{2} \sum_p \left(\phi_p - \sum_j \phi_j N_j(\mathbf{x}_p) \right)^2 \right] \quad (21)$$

Solving the Equation (21) leads to an equation system of J unknowns $\mathbf{M}\phi = \mathbf{f}$, where J is the number of nodes, $\mathbf{M}_{ij} = \sum_p N_i(\mathbf{x}_p)N_j(\mathbf{x}_p)$ is a consistent mass matrix, and $\mathbf{f}_i = \sum_p N_i(\mathbf{x}_p)\phi_p$.

Figure 1 shows a graphic representation of the approximation of a target function Φ where the particles look like quadrature points where the FEM solution is evaluated. The function ϕ is obtained minimizing (21), while the function I_h is the Lagrange interpolator which takes the values of the function Φ at nodal positions. Because an approximation as I_h , which belongs to the same discretization space than ϕ , has an error proportional with the square of the mesh-size, i.e., $\epsilon \approx C\Delta x^2$, and ϕ has the lowest error among the functions of this space, then it follows that ϕ has also an error proportional with the square of the mesh.

Finally the projection error is introduced at each projection step independently of the time-step size. Then, in a total period of time $(t_f - t_0)$ the Lagrangian error can be extended to

$$\begin{aligned} \epsilon^L = & \pm \mathcal{O} \langle (Q'' \pm \mathbf{v}'' \nabla T) \Delta x^2 \rangle (t_f - t_0) \pm \mathcal{O} \langle (Q'' \pm \mathbf{v}'' \nabla T) \Delta t^2 \rangle (t_f - t_0) \pm \\ & \pm \mathcal{O} \langle T'' \Delta x^2 \rangle \frac{(t_f - t_0)}{\Delta t} \end{aligned} \quad (22)$$

2.3 Validation Test

The differences presented in Equation (17) should be analyzed in depth. In the Eulerian case the error formula includes the spatial and temporal derivative of the product between the velocity and the scalar gradient, while in the Lagrangian formula they only affect the velocity. It has important theoretical consequences, which can be enumerated:

1. Eulerian frames are better for diffusion dominant problems. In these cases, the errors between the Lagrangian and Eulerian approaches are of the same order, but in the Lagrangian frames the projection errors must be added.
2. Lagrangian frames are better for convective dominant problems when the convective flow is constant or nearly constant in time $\mathbf{v}'' \approx \mathbf{v}''' \approx 0$. The remaining cases, when the convective flow presents high variations the Lagrangian or the Eulerian frame will be better or worse depending on the projection errors.
3. Eulerian frames are better for stationary problems. In these cases, $[\nabla T]'' = 0$ and there are no projection errors.

The aim of this section is to verify these facts experimentally, with numerical simulations employing both frames. The modification of some parameters in these tests will increase or decrease the terms which introduce errors in each formula, being the objective to match the experimental and theoretical results. In order to do this a particular 2D problem was tailored with an analytical solution to compare the results.

The proposed domain is $[-1, -1] \times [1, 1]$. The velocity field $\mathbf{v} = u\hat{\mathbf{i}} + v\hat{\mathbf{j}}$ has variations both in time and space: a rigid rotation with a periodic temporal variation of its angular velocity which is modulated in space by a parabola. The proposed velocity field is

$$\begin{cases} u(x, y, t) = -\omega_1 y(1 - x^2)(1 - y^2)(1 + C \sin(\omega_2 t)) \\ v(x, y, t) = \omega_1 x(1 - x^2)(1 - y^2)(1 + C \sin(\omega_2 t)) \end{cases} \quad (23)$$

where ω_1 is the mean angular velocity of the rotation field, ω_2 is the frequency at which the field increases or decreases its rotation. The parameter C allows to control the amplitude of the variation of the rotation direction, such as if $|C| > 1$ the rotation is inverted for certain time, if $C = 1$ the movement vanishes at some instant, and if $|C| < 1$ the field does not change its rotation direction. A source term q was included such that the unknown function becomes

$$T(x, y, t) = \sin(\omega_3 t) \sin(\pi x) \sin(\pi y) \quad (24)$$

which is identically zero at boundaries and has four hills at points $[\pm 0.5, \pm 0.5]$. A snapshot of the proposed scalar field is presented in Figure 2. An initial value of $T = 0$ and Dirichlet boundary conditions $T = 0$ were used. The diffusivity k and the oscillation frequency ω_3 can take independent values allowing to analyze different situations.

The reference grid used has 50×50 nodes conforming 4802 triangles, which gives a mesh-size $\Delta x = H = 0.04$. In the Eulerian case, Crank-Nicholson as time discretization scheme was employed. For Lagrangian simulations four particles per element were evenly seeded. The particle grid size h is defined as $h = \sqrt{2A/N_p}$, where A is the area of the element containing N_p particles. With this definition, the area of the element is divided by the number of particles contained.

Fig. 2 Snapshot for the manufactured scalar field solution

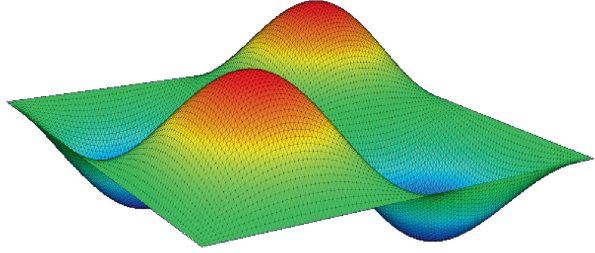


Table 1 Parameters for the manufactured 2D scalar transport case

Case	C	ω_1	ω_2	ω_3	k
1	1	2π	0	2π	0
2	1	2π	2π	2π	0
3	1	2π	2π	3π	0
4	1/2	2π	2π	2π	0

Four cases were designed to cover a wide range of problem types. Table 1 presents the parameters employed in each case. Two time-steps are employed in order to simulate with small $CFL_{max} \approx 1$ and large $CFL_{max} \approx 10$, respectively, being $CFL = |\mathbf{v}| \Delta t / \Delta x$ the Courant-Friedrich-Levy number.

In the following cases the root-mean-square (RMS) as an error measure is employed, defined as

$$RMS(t) = \sqrt{\frac{1}{N} \sum_{j=1}^N \left(T_j^{ex}(t) - T_j^{ap}(t) \right)^2}$$

where N is the number of nodes on the mesh, T^{ex} is the analytic reference solution, and T^{ap} is the numerical solution.

The results are presented in Figure 3. In the first case, which has a steady velocity field ($\mathbf{v}'' = 0$) and no diffusion, as anticipated in previous section, the Lagrangian framework has better results than the Eulerian one, being specially remarkable when large Courant numbers are employed. In the mentioned case, the PFEM-2 error does not depend on the time-step size showing a periodic variation due the spatial error which is proportional to the temporal harmonic function ∇T . On the other hand, FEM shows large errors when the time-step is increased as it is expected.

In the second case, the velocity field is unsteady having a harmonic variation in its amplitude. Here $\mathbf{v}'' \neq 0$, then the Lagrangian solution shows similar problems to the Eulerian one. However, if the oscillation frequency of the solution is larger than the amplitude variation of the velocity ($\omega_3 > \omega_2$), as happens in the third Case, the temporal error term in the Eulerian case ($[\mathbf{v} \nabla T]'' \Delta t^2$) increases, while the Lagrangian one ($\mathbf{v} [\nabla T]''$) remains almost equal. In this way, the second and third case also prove experimentally the theoretical error formula presented in Equation (17).

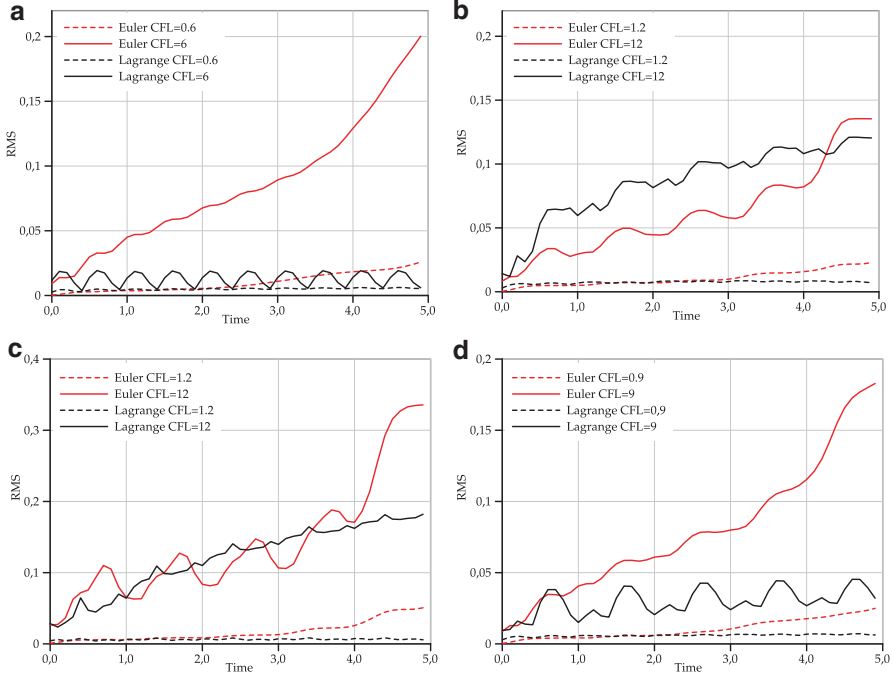


Fig. 3 Case Four hills without diffusion. RMS error evolution for Eulerian and Lagrangian simulations with steady (Figure 3(a)) and unsteady (Figures 3(b), 3(c), and 3(d)) velocity fields. The Lagrangian error is presented with black lines and the Eulerian with red lines

The second and the third cases use a velocity field which vanishes every $T = 1/\omega_2$, this gives to the Eulerian formulation some advantages due the spatial error term $[\mathbf{v} \nabla T]'' \Delta x^2$ also vanishes. That advantage is, in some cases, recovered by the Lagrangian approach when the time-step is increased. The fourth case overcomes this fact using another value for the constant C which avoids that $\mathbf{v}(t) = 0$ for any t . Figure 3(d) shows that the error differences are more visible, proving again the theoretical error formula.

As a preliminary conclusion, the results presented in Figure 3 confirm the proposed error formula in [17], showing that the Lagrangian error using large time-steps is affected by the unsteadiness of both, the velocity field and the source term. As a footnote based on experimental facts, it can be mentioned that an X-IVAS calculation of the unsteady source is mandatory to obtain accurate results, which is not possible with the time-step selected (it only samples seven points inside the source's period). Regarding to Eulerian simulations, as expected, they show larger errors when the CFL grows. Also, this error increases when high $\nabla T''$ are employed.

On the other hand, Figure 4 presents a convergence analysis for Case A. Three mesh refinements were used $\Delta x = 2H$, H , and $H/2$, modifying the Δt in order to keep the CFL_{max} in a constant value. The RMS error presented for each case is the

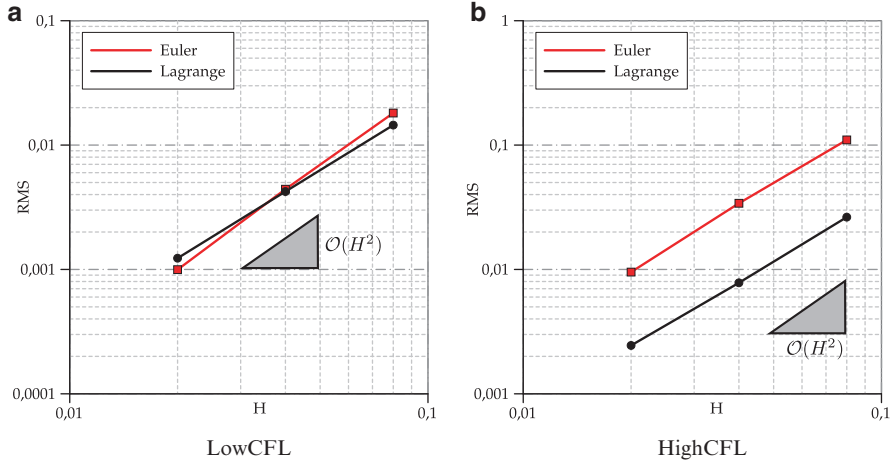


Fig. 4 Convergence of the RMS Error. Case A. $CFL_{max} = 0.6$ in 4(a) and $CFL_{max} = 6$ in 4(b)

average value of RMS from $T = 0$ to $T = 2[s]$ of simulation time. As presented in Figure 3(a), using $CFL_{max} \approx 0.6$ the Lagrangian and Eulerian errors are almost the same. However, when the dimensionless number is increased to $CFL_{max} \approx 6$, the Lagrangian solution obtains better accuracy than Eulerian even using both time-step and mesh-size twice larger. This fact represents a notable advantage of the Lagrangian approach over the Eulerian in this type of problems, showing that PFEM-2 simulation can obtain the same precision than FEM even solving a problem eight times smaller.

Regarding to convergence orders, both Lagrangian and Eulerian solutions show concordance with theoretical formula, presenting a second order convergence in every case. In the Lagrangian approach this behavior is reached if the projection operator fulfills this requirement.

3 An Extension to Viscous Incompressible Fluid Flow Simulations

In this case the model to be solved is represented by the Navier-Stokes equations added with the mass conservation that imposes a constraint on the velocity field to be divergence free. This condition is enforced by the pressure acting as a Lagrange multiplier. Inspecting the momentum equation and comparing this with the scalar transport equation just analyzed, here the same unknown variable acts as the velocity field that drives the convection term. This fact produces a non-linear term that is responsible for the chaotic nature of the model that pretends to emulate the physical effects produced by the turbulence.

3.1 General Comments

In this case the error equation looks like the same [17], but the conclusions are fundamentally different.

Generalizing Equation (17), the difference between the errors in the Eulerian and Lagrangian frames for convective dominant problems (high Reynolds number) is

$$\underbrace{[\mathbf{v}\nabla\mathbf{v}]'' \Delta x^2 \pm [\mathbf{v}\nabla\mathbf{v}]'' \Delta t^2}_{\text{Eulerian}} \neq \underbrace{\mathbf{v}''\nabla\mathbf{v}\Delta x^2 \pm \mathbf{v}''\nabla\mathbf{v}\Delta t^2}_{\text{Lagrangian}} \quad (25)$$

In this case, it is not anymore possible to separate the case in which the convective field is constant from the case in which the gradient of the transported field ($\nabla\mathbf{v}$) is not constant. Nevertheless, some particular cases may be analyzed in order to draw some conclusions:

- Eulerian frames are better for low Reynolds number. In these cases, the errors between the Lagrangian and the Eulerian approaches are of the same order, but on Lagrangian frames the projection errors must be added.
- Lagrangian frames are better for convective dominant problems when the velocity has a smooth variation in time but the gradient of the velocity has high spatial variations. This case is very common in fluid mechanics problems, such as in shock waves. The remaining cases are better or worse depending on the projection errors.
- Lagrangian frames are better for multi-fluid flows. This is because Eulerian frames need to solve a level set equation to know the position of the interface. The level set equation [27] is a convection equation that requires small time-steps to yield accurate results due to the considerations concluded in the previous section, i.e.,

$$\mathbf{v}''\nabla\mathbf{v} < 1 \quad \text{and} \quad [\mathbf{v}\nabla\mathbf{v}]'' \approx \mathbf{v}[\nabla\mathbf{v}]'' \gg 1$$

3.2 Validation Test

Following the same criteria as the validation test in Section 2.3, a manufactured case is employed where a solution is proposed and the external force \mathbf{f} must be adapted in order to satisfy the differential equation.

The case is adapted from the book of Donea & Huerta [3] where the authors solved a stationary Stokes flow. In this work, the problem includes also a convective term, leading to recalculation of the external force needed to satisfy the Navier-Stokes equations. A two-dimensional problem in the square domain $[0, 0] \times [1, 1]$ is considered, which possesses a closed-form analytical solution. The problem consists in determining the velocity field \mathbf{v} and the pressure p such that

$$\left\{ \begin{array}{ll} \frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} - \nabla \cdot (\nu \nabla \mathbf{v}) + \nabla p = \mathbf{b} & \text{in } \Omega \\ \nabla \cdot \mathbf{v} = 0 & \text{in } \Omega \\ \mathbf{v} = 0 & \text{in } \Gamma \end{array} \right. \quad (26)$$

where the fluid viscosity ν can be changed in order to simulate for different Reynolds numbers, being $Re = |\mathbf{v}|L/\nu$ with $L = 1$. In order to solve the previous equation system, an analytical solution is proposed:

$$\left\{ \begin{array}{l} u(x, y, t) = x^2(1-x)^2(2y-6y^2+4y^3)(1+0.5\sin(\omega t)) \\ v(x, y, t) = -y^2(1-y)^2(2x-6x^2+4x^3)(1+0.5\sin(\omega t)) \\ p(x, y, t) = x(1-x) \end{array} \right. \quad (27)$$

which allows to find the expression for \mathbf{b} . Varying ν and ω is possible to modify the Reynolds number and the unsteadiness of the solution, respectively. Cases were solved using a 50×50 Cartesian mesh split into triangles and setting the time-step such as $CFL_{max} \approx 10$. Figure 5 shows the shape of the proposed manufactured flow for two different times.

Table 2 shows the configuration and the RMS errors at $T = 1000[s]$ of the simulated cases.

Figure 6 presents graphically the evolution of the *RMS* for Cases 1 and 2. In the first case, a low Reynolds number was selected, therefore the Eulerian solution obtains better results although simulating with large CFL. This is an equivalent

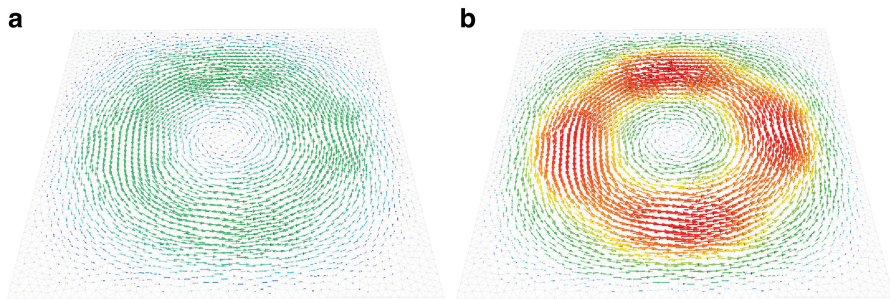


Fig. 5 Manufactured velocity field solution of the Navier-Stokes equations. Snapshots for two different times. Arrows indicating direction of the velocity field are colored by its magnitude

Table 2 Configuration and $RMS \times 10^3$ errors. Every case was run with $CFL_{max} \approx 10$

Case	ω	Re	Eulerian			Lagrangian		
			$RMS \mathbf{v}_x$	$RMS \mathbf{v}_y$	$RMS p$	$RMS \mathbf{v}_x$	$RMS \mathbf{v}_y$	$RMS p$
1	$\pi/5000$	10	0.75	0.73	0.036	2	1.9	0.058
2	$\pi/5000$	1000	0.012	0.011	0.014	0.0071	0.0062	0.013
3	$\pi/50$	10	0.8	0.82	0.026	0.83	0.85	0.023
4	$\pi/50$	1000	$\gg 1$	$\gg 1$	$\gg 1$	1.3	1.5	0.06

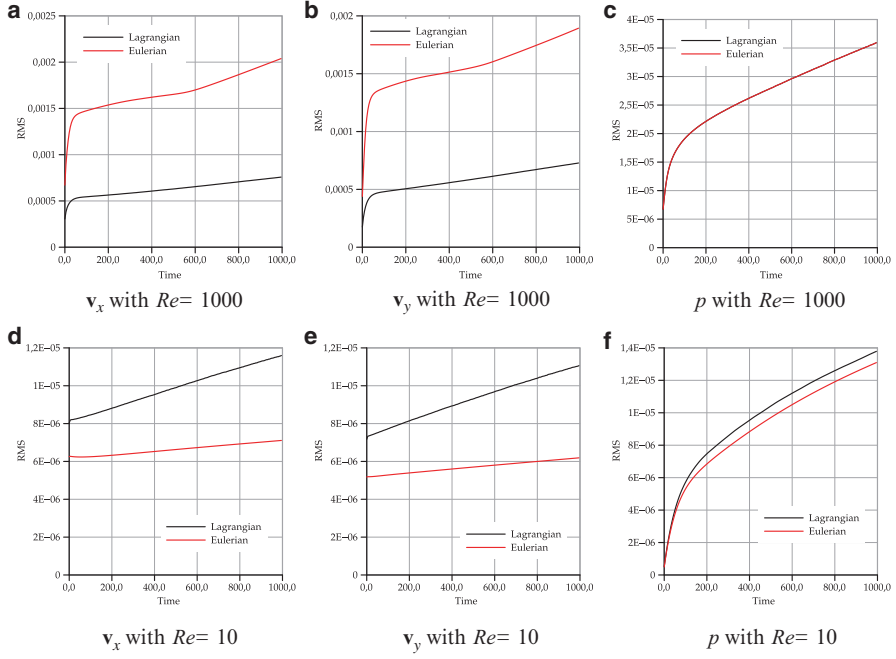


Fig. 6 RMS measured for a Navier-Stokes analytical solution using Lagrangian and Eulerian schemes. Cases with $\omega = \pi/5000$

problem to that diffusion dominated scalar transport problem. However, when the Reynolds number is increased, the Lagrangian framework recovers its advantage. In the third and fourth cases, the unsteadiness of the solution is increased, leading to an increment of the temporal derivatives. In the Lagrangian approach the error enlargement is lower than the Eulerian case, mainly because $\mathbf{v}''\nabla\mathbf{v} < [\mathbf{v}\nabla\mathbf{v}]''$. This fact is of extreme importance in the fourth case where the Eulerian solution diverges, concluding that beyond the increasing of the Reynolds number, the growth of the unsteadiness also affects the Eulerian simulations. On the other hand, Lagrangian framework is not adequate in the case of low Reynolds number, which is also consistent with the theoretical formula.

4 Conclusions

From the theoretical results published in [17] several conclusions were achieved allowing to decide in which situation an Eulerian framework is preferred against a Lagrangian one, and vice versa. In this paper the focus was on the experimental validation of those statements with the main goal of beginning to write a formal error analysis based on mathematical tools.

After confirming the validity of those conclusions through the results presented here not only for scalar problems, but also for vector systems like Navier-Stokes, the next step is establishing the main baseline for writing some a priori error analysis including not only the spatial and temporal approximations, but also the projection error for the particle finite element method.

As a conclusion, when advective dominant flows were analyzed, the PFEM-2 methodology has shown several advantages over pure Eulerian strategies. Beyond the possibility of enlarging time-steps, the Lagrangian method also allows to get as accurate solutions as Eulerian ones even using coarser meshes. Last means a significant saving of computational cost which is an invaluable feature in order to solve the challenging problems of next decade

Acknowledgements This work was partially supported by the European Research Council under the Advanced Grant: ERC-2009-AdG Real Time Computational Mechanics Techniques for Multi-Fluid Problems. Norberto Nigro and Juan Gimenez want to thank to CONICET, Universidad Nacional del Litoral and ANPCyT for their financial support (grants PICT 0830 (2013), PIP-2012 GI 11220110100331 and CAI+D 2011 501 201101 00435 LI).

References

1. Aubry, R., Idelsohn, S., Oñate, E.: Particle finite element method in fluid mechanics including thermal convection-diffusion. *Comput. Struct.* **83**(17–18), 1459–1475 (2005)
2. Brackbill, J., Ruppel, H.: Flip: a method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. *J. Comput. Phys.* **65**(2), 314–343 (1986)
3. Donea, J., Huerta, A.: *Finite Element Method for Flow Problems*. Wiley, Chichester (1983)
4. Gimenez, J., González, L.: An extended validation of the last generation of particle finite element method for free surface flows. *J. Comput. Phys.* **284**, 186–205 (2015)
5. Gimenez, J., Nigro, N., Idelsohn, S.: Evaluating the performance of the particle finite element method in parallel architectures. *J. Comput. Part. Mech.* **1**, 103–116 (2014)
6. Gingold, R., Monaghan, J.: Kernel estimates as a basis for general particle methods in hydrodynamics. *J. Comput. Phys.* **46**(3), 429–453 (1982)
7. Gingold, R.A., Monaghan, J.J.: Smoothed particle hydrodynamics, theory and application to non-spherical stars. *R. Astron. Soc.* **181**, 375–389 (1977)
8. Idelsohn, S., Calvo, N., Oñate, E.: Polyhedrization of an arbitrary 3d point set. *Comput. Methods Appl. Mech. Eng.* **192**, 2649–2667 (2003)
9. Idelsohn, S., Oñate, E., Calvo, N., Del Pin, F.: The meshless finite element method. *Int. J. Numer. Methods Eng.* **58**(6), 893–912 (2003)
10. Idelsohn, S., Oñate, E., Del Pin, F.: The particle finite element method a powerful tool to solve incompressible flows with free-surfaces and breaking waves. *Int. J. Numer. Methods* **61**, 964–989 (2004)
11. Idelsohn, S., Oñate, E., Pin, F.D., Calvo, N.: Fluid-structure interaction using the particle finite element method. *Comput. Methods Appl. Mech. Eng.* **195**, 2100–2113 (2006)
12. Idelsohn, S., Marti, J., Limache, A., Oñate, E.: Unified Lagrangian formulation for elastic solids and incompressible fluids: application to fluid-structure interaction problems via the {PFEM}. *Comput. Methods Appl. Mech. Eng.* **197**(19–20), 1762–1776 (2008). *Computational Methods in Fluid-Structure Interaction*
13. Idelsohn, S., Mier-Torrecilla, M., Oñate, E.: Multi-fluid flows with the particle finite element method. *Comput. Methods Appl. Mech. Eng.* **198**, 2750–2767 (2009)

14. Idelsohn, S., Nigro, N., Limache, A., Oñate, E.: Large time-step explicit integration method for solving problems with dominant convection. *Comput. Methods Appl. Mech. Eng.* **217–220**, 168–185 (2012)
15. Idelsohn, S., Nigro, N., Gimenez, J., Rossi, R., Marti, J.: A fast and accurate method to solve the incompressible Navier–Stokes equations. *Eng. Comput.* **30**(2), 197–222 (2013)
16. Idelsohn, S., Marti, J., Becker, P., Oñate, E.: Analysis of multifluid flows with large time steps using the particle finite element method. *Int. J. Numer. Methods Fluids* **75**(9), 621–644 (2014)
17. Idelsohn, S., Oñate, E., Nigro, N., Becker, P., Gimenez, J.: Lagrangian versus Eulerian integration errors. *Comput. Methods Appl. Mech. Eng.* **293**, 191–206 (2015)
18. Koshizuka, S., Oka, Y.: Moving-particle semi-implicit method for fragmentation of incompressible fluid. *Nucl. Sci. Eng.* **123**(3), 421–434 (1996). Cited By 646
19. Koshizuka, S., Tamako, H., Oka, Y.: A particle method for incompressible viscous flow with fluid fragmentation. *Comput. Fluid Mech. J.* **113**, 134–147 (1995)
20. Laresse, A., Rossi, R., Oñate, E., Idelsohn, S.: Validation of the particle finite element method (PFEM) for simulation of the free-surface flows. *Eng. Comput.* **25**(4), 385–425 (2008)
21. Monaghan, J.: An introduction to SPH. *Comput. Phys. Commun.* **48**, 89–96 (1988)
22. Neuman, S.P.: Adaptive Eulerian-Lagrangian finite element method for advection-dispersion. *Int. J. Numer. Methods Eng.* **20**(2), 321–337 (1984)
23. Oliver, J., Cante, J., Weyler, R., González, C., Hernandez, J.: Particle finite element methods in solid mechanics problems. In: *Computational Plasticity*, pp. 87–103. Springer, Berlin (2007)
24. Oñate, E., Idelsohn, S., Zienkiewicz, O., Taylor, R.: A finite point method in computational mechanics. applications to convective transport and fluid flow. *Int. J. Numer. Methods Eng.* **39**(22), 3839–3866 (1996)
25. Oñate, E., Idelsohn, S., Zienkiewicz, O., Taylor, R., Sacco, C.: A stabilized finite point method for analysis of fluid mechanics problems. *Comput. Methods Appl. Mech. Eng.* **139**(1–4), 315–346 (1996)
26. Oñate, E., Sacco, C., Idelsohn, S.: A finite point method for incompressible flow problems. *Comput. Vis. Sci.* **3**(1–2), 67–75 (2000)
27. Osher, S., Fedkiw, R.: Level set methods: an overview and some recent results. *J. Comput. Phys.* **169**, 463–502 (2001)
28. Pin, F.D., Idelsohn, S., Oñate, E., Aubry, R.: The ALE/Lagrangian particle finite element method: a new approach to computation of free-surface flows and fluid-object interactions. *Comput. Fluids* **36**(1), 27–38 (2007). *Challenges and Advances in Flow Simulation and Modeling*
29. Wieckowski, Z.: The material point method in large strain engineering problems. *Comput. Methods Appl. Mech. Eng.* **193**(39–41), 4417–4438 (2004). *The Arbitrary Lagrangian-Eulerian Formulation*

Advances in Computational Fluid-Structure Interaction
and Flow Simulation

New Methods and Challenging Computations

Bazilevs, Y.; Takizawa, K. (Eds.)

2016, XII, 500 p. 193 illus., 158 illus. in color.,

Hardcover

ISBN: 978-3-319-40825-5

A product of Birkhäuser Basel