Development of a computational instrument using a lagrangian particle method for physics teaching in the areas of fluid dynamics and transport phenomena

Carlos Alberto Dutra Fraga Filho^{*1}

¹Federal Institute of Education, Science and Technology of Espírito Santo, Vitória, ES, Brazil

Received on December 2, 2016. Revised on January 16, 2017. Accepted on February 18, 2017.

Particle methods are an alternative to the traditional mesh Eulerian methods for solution of fluid dynamics and transport phenomena problems. The Lagrangian Meshless Smoothed Particle Hydrodynamics (SPH) method presents advantages compared to mesh methods as a better visualisation of the spatio-temporal evolution of the flow (and the fluid properties), and a lower computational cost in the study of complex geometries with topological changes, or free surfaces. A computer code has been written for serial computation using FORTRAN Programming Language. Three classical physical problems have been simulated: diffusion in a flat plate, still fluid within an immobile reservoir and dam breaking. In code validation, numerical results obtained showed a good agreement with the analytical and experimental results reported in the literature. The numerical code developed and presented in this work is a valuable tool for the teaching of Physics.

Keywords: Physics teaching, Fluid dynamics, Transport phenomena, Particle method, Computer code.

1. Introduction

The Eulerian view is historically the most commonly used in modelling of physical problems involving fluids and phenomena transport. This modelling requires the use of numerical methods employing grids or meshes, such as finite differences, finite volumes and finite elements, for the solution of partial differential equations (PDEs) or integral formulations [1]. The greatest difficulties in using meshes are in ensuring consistency. Employing meshes can lead to a number of difficulties when dealing with fluid flow problems with free surface, mobile interfaces, complex geometries and topological changes. If there are these features, generating a mesh, a prerequisite for the numerical simulation of quality, becomes a difficult process, both time-consuming and expensive.

Lagrangian modelling is based on the hypothesis that the problem domain can be divided into a finite number of particles that do not interact with one another (one-particle model). Each particle center of mass receives the coordinates that define its position in space and the physical properties of each Lagrangian element are found at each instance in time. The Lagrangian particle model is meshless and has been an alternative applied in research, pointing to the use of more effective computational methods for solving complex problems, providing

stable and accurate numerical solutions for PDEs or integral equations.

Some of the advantages of the use of Lagrangian modelling regarding the Eulerian are the simplicity and lower computational cost in complex geometries, not needing the use of meshes, non-production of numerical oscillations, conservation of mass at the Lagrangian element, and finally the graphical visualisation of the results, that allows a better understanding of the spatio-temporal evolution of the fluid and its properties.

Until a few years ago, Lagrangian particle modelling was not used in the simulation of problems in fluid mechanics and transport phenomena. In 2011, Pritchard ([2]) reported that describing the movements of individual fluid particles would be impractical, advising the use of the control volume analysis in modelling for fluids. For a long time, Lagrangian modelling was put aside in favour of Eulerian methods due mainly to the processing time: in general, higher than that required by mesh methods.

Currently, however, the progress of the processing techniques, the rapid spread of multi-core processors, and parallelisation, made the use of particle methods possible. Graphics processing units (GPUs), using Compute Unified Device Architecture (CUDA), developed by NVIDIA, allows the use of more than one million particles in the discretisation of the fluid domain [3].

Most graduate and undergraduate students still study physical laws according to the Eulerian view, basically

^{*}Endereço de correspondência: cadff1@gmail.com.

due to the academic domain of numerical methods that use meshes to solve the differential equations used in problem modelling. However, nowadays, the meshless particle methods have become an alternative for the solution of the physical problems, with a promising future, due mainly to the advance of the processing techniques. In this context, the learning of the Lagrangian modelling and numerical particle methods is a current need in the courses of Physics. In addition, there is an increase in researches that aim at the development of Lagrangian mathematical models and new computational tools to solve physical problems.

This paper aims to present the employment of a Lagrangian particle method (Smoothed Particle Hydrodynamics) in the implementation of a computer code for solving certain physical problems. Thus, a valuable new tool for the Physics teaching will be presented to the students and researchers.

The remainder of this article is organised as follows. Section 2 brings a briefly presentation of the SPH Lagrangian method fundamentals and numerical aspects. The computer code developed is presented in Section 3. The validation of the computer code (with numerical simulations performed and results analysis) is shown in Section 4. In Section 5 conclusions are presented.

2. SPH Lagrangian Particle Method Fundamentals

SPH is based on the valid mathematical identity for a scalar function $f(\mathbf{X})$, defined and continuous over the whole domain, as show the equation (1):

$$f(\mathbf{X}) = \int f(\mathbf{X}') (X' - X) dX' \quad (1)$$

where $f(\mathbf{X})$ is a scalar function defined at the fixed point $\mathbf{X} = (x, y)$, $\delta(X' - X)$ is the Dirac Delta function:

$$\delta(X' - X) = \begin{cases} 1, & \text{if } X = X', \\ 0, & \text{if } X \neq X'. \end{cases}$$

By replacing the Dirac Delta function with the smoothing function, W , the approximation to the function at position $\mathbf{X}$ is obtained (equation (2)):

$$f(\mathbf{X}) = \int f(\mathbf{X}') W(\mathbf{X}' - \mathbf{X}, h) dX' \quad (2)$$

where h is the support radius, $dX' = dx'dy'$ is the infinitesimal area element, $\mathbf{X}$ and $\mathbf{X}'$ are the positions of a fixed and a variable point at domain, respectively.

The essence of the SPH method is in the discretisation of the continuum domain (Ω), in a finite number of particles and in these Lagrangian elements to obtain the values of the physical properties through from weighted

interpolations, from the properties of the particles in the vicinity. Only the neighbouring particles, which are within the influence domain (at a maximum distance kh of the fixed particle considered, k being a scale factor that depends on the kernel employed), contribute to its behaviour. Figure 1 shows the arrangement of particles within the influence domain.

Searching for the neighbouring particles can be performed directly or using grids, which leads to a lower number of mathematical operations and reduces the computational cost ([4, 5]). Figure 2 shows the direct search (in which all pairs of particles "ab" at the domain will have the distances calculated and compared to the kh value) and the use of a grid, which reduces the number of searching operations (only particles within the shaded region, with three, nine or 27 cells, in 1-D, 2-D or 3-D cases, respectively, will be verified, whether they are or not neighbours of a reference particle).

Different kernels can be used, and so that they are considered suitable for interpolation, it is necessary that

Figure 1: Graphical representation of the influence domain. (a) The reference (fixed) particle, a , has neighbouring particles (b) within the influence domain. (b) The kernel (W) ensures a greater contribution from the nearest neighbouring particles for the value of the physical property in the reference particle.

Figure 2: Neighbouring particles search. (a) Directly. (b) Using a grid (Adapted from [4]).

each one follows certain properties: smoothness, positivity, symmetry, convergence, decay, compact support, and normalisation within the domain of influence [4]. Figure 3 shows the kernel used by Lucy in his pioneer SPH work [6], and the cubic spline kernel.

SPH approaches physical properties (such as density), gradients (pressure gradients), divergence (velocity divergence) and Laplacians (temperature and velocity Laplacians) related to fluid flow, with a second-order error.

Table I presents the kernels employed in this work.

2.1. SPH Approximations for Differential Equations in Fluid Mechanics and Transport Phenomena

Modelling of fluid flows and energy transport is performed by the conservation (mass and energy) and momentum balance equations. The evolution of the field densities, velocities and energy is defined by equations (3)–(5). Table II shows the conservation and momentum balance equations, according to the Lagrangian frame of refer-

Table 1: Interpolation Kernels employed in the SPH simulations

KERNEL	$W(r, h) \text{ (m}^{-2}\text{)}$
1	$\frac{15}{7\pi h^2} \begin{cases} \left(\frac{2}{3} - q^2 + \frac{1}{2}q^3\right), 0 \leq q \leq h \\ \left(\frac{1}{6}(2 - q^3)\right), h < q \leq 2h \\ 0, \text{ in other case.} \end{cases}$
2	$\frac{5}{\pi h^2} \begin{cases} (1 + 3q)(1 - q)^3, 0 \leq q \leq h \\ 0, \text{ in other case.} \end{cases}$
3	$\frac{15}{7\pi h^2} \begin{cases} \left(\frac{2}{3} - \frac{9}{8}q^2 + \frac{19}{24}q^3 - \frac{5}{32}q^4\right), 0 \leq q \leq 2h \\ 0, \text{ in other case.} \end{cases}$
4	$\frac{7}{478\pi h^2} \begin{cases} (3 - q)^5 - 6(2 - q)^5 + 15(1 - q)^5, 0 \leq q \leq h \\ (3 - q)^5 - 6(2 - q)^5, h < q \leq 2h \\ (3 - q)^5, 2h < q \leq 3h \\ 0, \text{ in other case.} \end{cases}$

¹Cubic spline kernel [4]; ²Quartic kernel [6]; ³New quartic kernel [1]; ⁴Quintic Spline kernel [7]; $q = \left(\frac{|r - r_b|}{h}\right)$ and h is the support radius.

Table 2: Differential Equations and SPH Approximations

Differential Equation (Continuum domain)	SPH Approximation (Domain discretised by particles)
Mass conservation:	
$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}$	$\frac{d\rho_a}{dt} = \sum_{b=1}^n m_b \mathbf{v}_{ab} \cdot \nabla W(\mathbf{X}_a - \mathbf{X}_b, h) \quad (3)$
Momentum balance:	
$\frac{d\mathbf{v}}{dt} = -\frac{\nabla P}{\rho} + \nu \nabla^2 \mathbf{v} + \mathbf{g}$	$\begin{aligned} \frac{d\mathbf{v}_a}{dt} = & \sum_{b=1}^n m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} \right) \nabla W(\mathbf{X}_a - \mathbf{X}_b, h) \\ & + 2\nu_a \sum_{b=1}^n \frac{m_b}{\rho_b} \mathbf{v}_{ab} \frac{(\mathbf{X}_a - \mathbf{X}_b)}{ \mathbf{X}_a - \mathbf{X}_b ^2} \cdot \nabla W(\mathbf{X}_a - \mathbf{X}_b, h) + \mathbf{g} \end{aligned} \quad (4)$
Energy conservation:	
$\frac{de}{dt} = \frac{1}{\rho} (-P \nabla \cdot \mathbf{v} + \varepsilon_v + \nabla \cdot \mathbf{q} + q_H)$	$\frac{de_a}{dt} = \frac{1}{\rho} \left[-P_a \sum_{b=1}^n m_b \mathbf{v}_{ab} \cdot \nabla W(\mathbf{X}_a - \mathbf{X}_b, h) + \varepsilon_v + \nabla \cdot \mathbf{q} + q_H \right] \quad (5)$

where ρ is the fluid density, t is the time, ∇ is the mathematical vector operator nabla, $\mathbf{v}$ is the fluid velocity, W is the kernel, $\mathbf{X}$ is the spatial position, $\mathbf{g}$ is the acceleration due to gravity, ν is the kinematic fluid viscosity, P is the absolute pressure, e is the specific internal energy, ε_v is the energy dissipation per unit volume, $\mathbf{q}$ is the conduction heat flux, q_H is the heat generated by other sources per unit volume, $\mathbf{v}_{ab} = \mathbf{v}_a - \mathbf{v}_b$, a and b are subscripts that refer to fixed and neighbouring particles, respectively.

Figure 3: Kernels and their first-order derivatives. (a) Lucy's quartic kernel and (b) Cubic spline kernel. Adapted from [4].

ence, and their approximations by SPH to a viscous and incompressible fluid.

A complete presentation of the SPH method can be found in [4]. Fluid mechanics physical laws (mathematical Lagrangian modelling), computational implementations and results achieved until this moment (summarized in the this paper) can be found in [8].

2.2. Errors in the SPH approximations

The approximations to a function and its derivatives taken by the SPH method have errors. This subject will be discussed below.

Applying the expansion of the Taylor series for the function $f(\mathbf{X})$, which is differentiable in the domain around the fixed position $\mathbf{X}$, we obtain equation (6):

$$\begin{aligned} \langle f(X) \rangle &= \int f(X') W(X' - X, h) dX' + \\ &f'(X) \int (X - X') W(X' - X, h) dX' \\ &+ Re(h^2) \end{aligned} \quad (6)$$

where $f'(\mathbf{X})$ is the derivative of the function and $Re(h^2)$ is the residue of the second order.

Due to the kernel symmetry, equation (6) results in:

$$\langle f(X) \rangle = \int f(X') W(X' - X, h) dX' + Re(h^2) \quad (7)$$

From the analysis of equation (7), it is seen that the approximation to the function value has a second-order error. Similarly, the approximations for derivatives of the first and second orders also have errors.

2.3. Artificial Viscosity

The transformation of kinetic energy into heat takes place in problems involving mainly shock waves and needs to be measured. That energy transformation can be represented as a form of viscous dissipation, called artificial viscosity. Its application in the simulations aims to avoid numerical instabilities and the interpenetration between particles. The formulation used in the modelling of artificial viscosity is shown in equation (8), as presented in [4].

$$\pi_{ab} = \begin{cases} \frac{-\alpha_{\pi} \chi_{ab} c_{ab} + \beta_{\pi} \chi_{ab}^2}{\rho_{ab}}, & (\mathbf{v}_a - \mathbf{v}_b) \cdot (\mathbf{X}_a - \mathbf{X}_b) < 0, \\ 0, & (\mathbf{v}_a - \mathbf{v}_b) \cdot (\mathbf{X}_a - \mathbf{X}_b) \geq 0. \end{cases} \quad (8)$$

$$\begin{aligned} \chi_{ab} &= \frac{h_{ab} (\mathbf{v}_a - \mathbf{v}_b) \cdot (\mathbf{X}_a - \mathbf{X}_b)}{|\mathbf{X}_a - \mathbf{X}_b|^2 + 0.01 \varphi^2}, \\ c_{ab} &= \frac{c_a + c_b}{2}, \quad \rho_{ab} = \frac{\rho_a + \rho_b}{2}, \\ h_{ab} &= \frac{h_a + h_b}{2}. \end{aligned}$$

where π_{ab} is the artificial viscosity, α_{π} and β_{π} are coefficients used in the calculation of artificial viscosity, c_a and c_b are the velocities of sound in the fixed and neighbouring particles, respectively, h_a and h_b are the support radii of the fixed and neighbouring particles, respectively, φ^2 is a factor that prevents numerical differences when two particles approach one another.

The linear and quadratic coefficients are parameters controlling the strength of the artificial viscosity. The viscosity associated with α_{π} produces a bulk viscosity, while the second term associated with β_{π} , intended to suppress the particle interpenetration at high Mach number. The best choice for their values depends on the problem being addressed. The factor φ^2 was set to $0.01 h_{ab}^2$.

The term related to artificial viscosity (π_{ab}) is added to the terms of pressure in the SPH approximations for the momentum balance and energy conservation equations [8].

Figure 4: Domain of influence in a polar coordinate system. (a) Complete and (b) incomplete. It is possible to see that the kernel is not defined in a whole domain.

2.4. Particle Inconsistency

Particles near the boundaries do not have complete domains of influence, which causes truncation of the kernel and consequent lower accuracy in the interpolations [1]. This phenomenon is known as particle inconsistency. Figure 4 shows the occurrence of the truncation of the kernel in the coordinate polar system.

In the border regions, as shown in Fig. 5 (b), the kernel consistency is not guaranteed. There is not a full domain of influence and the kernel is truncated. A correction method based on the Taylor series called the Corrected Smoothed Particle Method (CSPM) is used in the correction of the inconsistency [9].

3. Computer Code

Figure 5 shows the flowchart of the computer code implemented using FORTRAN programming language. The algorithm has been written for serial computation.

Below follows a brief presentation of the code and its routines.

1. Initial conditions: the initial positions, velocities, densities, temperatures, support radius and other fluid particles' physical properties are set to the beginning of the simulation.
2. Boundary conditions definition: the technique to be employed for the boundary treatment is defined.
3. Searching for the neighbouring particles: particles within the domain of influence of a reference particle may vary over time, and the search must be performed at each numerical iteration.
4. Pressure calculus: in this routine, updating of the pressure field acting on the particles is performed.
5. Kernel definition: smoothing function used in interpolations is chosen.
6. Mass conservation equation solution: the particles' density calculus for each particle is conducted from solution of equation (3).
7. Obtaining of surface forces: approximations to surface forces acting on the particles are obtained (the first two terms of equation (4), pressure and viscous forces, respectively).
8. External forces application: the last term on the right side of equation (4) is the sum of all external forces. In the class of external forces (third term on the right side of the equation (4)) is the gravity. Other forces can also be considered such as the repulsive forces exerted by the particles on the contours of fluid particles (Lennard-Jones forces) and free surface forces, if they exist in the studied problem.
9. Energy conservation equation solution: SPH approximations are employed for solution of equation (5).
10. Momentum balance equation solution: particles' accelerations field is obtained from solution of equation (4).
11. Temporal integration: prediction of the particles' properties to the next iteration. The Courant-Friedrichs-Lewy (CFL) stability criterion was applied in the time-step setting to ensure convergence of results [10].
12. Output files: these files are obtained at the end of each numerical iteration and, from their data, graphical representations of the fluid's physical properties are generated.
13. Accuracy: this routine verifies whether the desirable accuracy has been achieved or if a new iteration will have to be executed. In the latter case, we must return to step 2 (boundary conditions definition) and follow the remaining steps until a new iteration

Figure 5: Algorithm implemented with all routines.

verification of the accuracy of numerical results be performed.

In steps 6 and 7, the correction of the particles' physical properties (density and pressure gradient) located near the boundaries can be made using the Corrected Smoothed Particle Method (CSPM), [9].

In the present work, the simulations have been performed using a computer with 8GB of RAM memory, Intel Core i5-2450M 2nd generation processor, and Linux operational system.

4. Code Validation

The numerical code has been used for solution of some physical problems. Before the beginning of simulations, an examination of the physical phenomenon was carried out to select the routines to be used (Fig. 5 shows all available routines). From the numerical results analysis, made through comparisons with analytical and results reported in the literature, it has been possible to validate the computational code for application in studied cases. In this section, the set of problems, their results, and a discussion of their results are presented and discussed.

4.1. Heat Diffusion in a Homogeneous Flat Plate

The equation that governs the energy transfer in the absence of pressure fields and energy dissipation is presented below (equation (9)):

$$\rho \frac{de}{dt} = \nabla \cdot \mathbf{q} + q_H \quad (9)$$

where e is the specific energy, $\mathbf{q}$ is the heat flux, and q_H is the heat source.

The heat flux is obtained by Fourier's law. For an isotropic and homogeneous material, thermal conductivity is constant, and this law is given by equation (10):

$$\mathbf{q} = k \nabla T \quad (10)$$

where k is the thermal conductivity and T is the temperature.

In the absence of heat source, in a thermal steady state, the energy conservation equation is written as follows (equation (11)):

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (11)$$

The calculation of the SPH Laplacian of temperature for each iteration was done explicitly, in a polar coordinate system, by solving equation (12), [11]:

$$\left(\frac{\partial^2 T_a}{\partial x^2} + \frac{\partial^2 T_a}{\partial y^2} \right)^m = 2 \sum_{b=1}^n \frac{m_b}{\rho_b} [T(X_a)^m - T(X_b)^m] \times \frac{\partial W(r, h)}{\partial r} \frac{1}{r_{ab}} \quad (12)$$

where m is the current time step, b is the subscript indicating each neighbouring particle of the fixed particle, which is denoted by the subscript a , n is the number of particles within the domain of influence, $r_{ab} = |\mathbf{X}_a - \mathbf{X}_b|$ is the distance between two particles, and r is the radial direction.

The technique of the separation of variables was employed to obtain the solution by series for the temperature distribution in an isotropic, homogeneous and flat plane, at the stationary state, as shows equation (13), [12].

$$T(x, y) = \sum_{N=1}^{\infty} A_N \sin(N\pi x) \sinh[N\pi(y-1)] \quad (13)$$

where $A_N = \frac{2T_S}{N\pi} \frac{[(-1)^N - 1]}{\sinh(N\pi)}$, $T_S = T(x, 0)$ is the temperature at the lower boundary and N is the number of terms employed in the series.

In this work the first ninety terms of the series were used.

The dimensions of the plate were 1.0 m x 1.0 m. The whole domain had temperature T_0 equal to 0°C, at the initial time; that is, the initial temperature was uniform. The prescribed boundary conditions (Dirichlet boundary conditions) have been imposed: 0°C at right, left and lower sides and 100°C at upper side of the plate.

The particles have been placed in the domain separated by a lateral distance $dx = 1.0/n_p$ (n_p is the number of particles on each side of the plate) and their positions did not change with time. The particles temperatures inside the domain have been initialized with 0°C and the thermal diffusivity was defined as 1.0 m²/s, which was held constant for all simulations. The influence domain (kh) was defined as 2.5 dx , which was invariable during simulations. The time step was 1.0 x 10⁻⁵ s. The forecast of temperatures was performed by Euler integration method. In the defining of the boundaries, virtual particles were fixed at the contours at a ratio of two virtual particles for each real particle and the temperatures of the sides of the plate were attributed to them. Virtual particles' properties were not subject to interpolations to predict their temperatures (Dirichlet boundary condition).

Simulations were carried out starting from the transient until the steady state was reached. Different combinations of numbers of particles (50, 60, 70, 80, and 90 per side of the flat plate) / kernels have been used.

Figure 6 shows the initial distribution of 50 x 50 particles in the domain (and the prescribed temperatures at the boundaries, attributed to the virtual particles) and the temperature distribution when the steady state was reached ($t = 0.431$ s).

For internal regions of the domain, the approximation of the Laplacian of the temperature by the SPH method gave results that were in agreement with the analytical solution.

The largest relative errors found in the steady state (ΔT_p), obtained from the analysis of the differences be-

Figure 6: (a) Initial distribution of 50×50 particles in the domain and the prescribed temperatures at the boundaries (attributed to the virtual particles). (b) Temperature distribution when the steady state has been reached (a cubic spline kernel has been employed).

tween the analytical solution and the SPH numerical results, occurred in a few positions of the lower corners of the flat plate.

Figure 7 shows these larger relative errors for different combinations of particle numbers and kernels.

Smaller differences in temperatures in regions close to the boundaries were observed when a kernel of higher degree was utilized (quintic spline), except for the distribution with 3600 particles. The complete analysis of simulations' results is in [11, 13].

From the results analysis, we can conclude that to achieve better results it is necessary to perform a correction of the temperature values in regions near the contours (due to the kernel truncation problem).

Figure 8 shows the CPU time for different combinations number of particles/kernel employed in simulations.

Figure 9 presents the algorithm implemented in a flowchart scheme (wherein the enabled routines in the computer program is shown).

4.2. Incompressible, uniform and isothermal fluid at rest within an immobile reservoir

This is one of the first problems that must be understood and solved in Fluid Mechanics. This hydrostatic problem

Figure 7: Behaviour of the higher relative error.

Figure 8: CPU times for the different combinations number of particle/ kernel employed in simulations.

consists of a tank that is at rest, open to the atmosphere, and filled with a liquid. The origin of the frame of reference, with a positive vertical orientation of pointing up, is fixed at the bottom of the reservoir, as shown in Fig. 10.

The solution for a Newtonian, incompressible and isothermal fluid at rest within an immobile reservoir is well known. The application of the macroscopic view and the continuum theory (where the molecular motion of the fluid are not considered) allows the use of partial differential equations in the problem mathematical modelling. The integration of the momentum balance equation in the vertical direction leads to equation (14):

$$(P - P_0) = \rho g(H - y) \quad (14)$$

The physical laws of mass conservation and momentum balance (equations (3) and (4)), and the concept of the modified pressure, presented in [14], have been employed in the problem solving. The modified pressure concept is presented mathematically by equation (15):

$$P_{mod} = (P - P_0) - \rho g(H - y) \quad (15)$$

Figure 9: Algorithm implemented for diffusion in a homogeneous flat plane.

Figure 10: Open reservoir containing a Newtonian, uniform, incompressible, and isothermal fluid.

where P_{mod} is the modified pressure, P_0 is the reference pressure (atmospheric pressure), y is the ordinate of fluid (with the referential frame fixed at bottom) and H is the ordinate of the fluid surface.

Taking into account that $(P - P_0)$ is the pressure exerted by the fluid column, the modified pressure is equal to zero, equation (16):

$$P_{mod} = 0 \quad (16)$$

where g is the gravity magnitude.

The use of the modified pressure changes the form of the momentum balance equation (equation (4)), as shown in equation (17).

$$\frac{d\mathbf{v}}{dt} = -\frac{\nabla P_{mod}}{\rho} + \nu \nabla^2 \mathbf{v} \quad (17)$$

An SPH approximation for the modified momentum balance equation is:

$$\begin{aligned} \frac{d\mathbf{v}_a}{dt} = & \frac{1}{\rho_a} \sum_{b=1}^n m_b (P_{mod(b)} - P_{mod(a)}) \\ & \times \nabla W(\mathbf{X}_a - \mathbf{X}_b, h) \\ & + 2\nu_a \sum_{b=1}^n \frac{m_b}{\rho_b} \mathbf{v}_{ab} \frac{(\mathbf{X}_a - \mathbf{X}_b)}{|\mathbf{X}_a - \mathbf{X}_b|^2} \\ & \times \nabla W(\mathbf{X}_a - \mathbf{X}_b, h) \end{aligned} \quad (18)$$

where $P_{mod(a)}$ and $P_{mod(b)}$ are the modified pressures on the fixed particle and the neighbouring particle, respectively.

The initial distribution of the particles in the domain has been performed on a regular disposition with 50 particles per side of the tank with dimensions of 1.00×1.00 m, or 2500 water particles in total, at 20° C and at sea level ($\rho = 1.00 \times 10^3$ kg/m³, $\nu = 1.00 \times 10^{-6}$ m²/s). The number of particles within the influence domain was twenty-one. The boundary treatment has used virtual particles. Three rows and three columns of this particle type have been added to the regular distribution of fluid particles, totalling 636 virtual particles (whose physical properties were equal to the properties of the fluid particles). Figure 11 shows the initial particle distribution at domain and the distribution of hydrostatic pressure on the fluid particles. The modified pressures for all particles (fluid and virtual), at rest, were initialized to zero. Cubic and quintic spline kernels have been used in simulations. The time step employed was 1.00×10^{-4} s.

After the solution of equation (18) and updating from the particle positions and velocities by the numerical temporal integration, for each numerical iteration, those remained unchanged, and the hydrostatic equilibrium has been maintained. The modified pressures field were restored with 0 at the start of each new iteration, since the particles were at rest (and free surface has not changed). After starting the simulation, at the time instance 5.00

Figure 11: Initial distribution of fluid particles (red) and virtual particles (blue). The latter are in an expanded region of the domain.

s (50,000 numerical iterations), the coincidence of the positions of the centers of mass at that moment and their initial positions was verified (regardless of the kernel used) in agreement with the analytical solution (Fig.12).

In previous literature studies, standard SPH method has been employed and results have been obtained using the momentum balance equation in its traditional form, presented in equation (4) ([15-18]). When the standard SPH was applied to the obtaining of approximations for the pressure and viscosity forces, terms which were present in the momentum balance equation and which were added to the gravitational force, made small mistakes appear. Despite of the small orders of magnitude from these errors, they have spread and increased during the numerical simulation. The particles did not remain in their initial positions, and their movements did not cease with time progress, which is not physically correct, considering the continuum hypothesis applied for fluids (which does not take into account the molecular motion). The particle and the continuum approaches, employed in SPH studies, can agree since the microscopic fluctuations, which are absent in continuum mechanics, can be ignored [19].

In this research, by using the modified pressure concept, the errors from the approximations provided by the SPH method for the physical laws (written mathematically in the form of partial differential equations) were eliminated. In consequence the continuum theory was obeyed and the particles oscillations haven't appeared.

The processing times were 6 h 37 min 6 s and 6 h 57 min 23 s, when the cubic and quintic spline kernels were used, respectively.

The flowchart, in Fig. 13, shows the enabled routines in the computer program used for solution of the hydrostatic problem.

4.3. Dam Breaking

This is a classic problem of fluid dynamics. Dam breaking studies are of great importance to the prevention of

Figure 12: Positions of the particles within the reservoir, time invariant.

Figure 13: Flowchart of the numerical code for the reservoir problem containing an incompressible fluid and open to the atmosphere.

accidents, which can cause serious environmental consequences and also damage to inhabitants located in the vicinity.

The fluid is considered incompressible, uniform and isothermal; the Navier-Stokes equations, mass conservation and momentum balance (equations (3) and (4)), must be solved.

Numerical simulations have been performed for two different heights of dammed water (H), and the results have been compared to the results of experiments available in literature [20]. The 2-D computational domains simulated were: (a) a square area with height and width of 0.114 m, discretized by 1296 fluid particles, with lateral distances between their centres of mass of 3.26 mm; (b) a rectangular area with a height of 0.228 m and width of 0.114 m, discretized by 2556 fluid particles, with lateral distances between their centres of mass of 3.26 mm. The experimental apparatus is schematically shown in Fig. 14.

Figure 14: Experimental apparatus used in the dam breaking experiments [20].

Figure 15 presents the 2-D computational domains simulated and the initial distribution of the particles.

The absolute pressure (P) on the particles presents terms related to the hydrostatic and dynamic pressures. The estimation of the dynamic pressure field with the use of the Tait equation, suggested in [14], is presented in equation (19):

$$P_{dyn} = B \left[\left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right] \quad (19)$$

where P_{dyn} is the dynamic pressure, B is the term associated with fluctuations in the fluid density, ρ_0 is the fluid density at rest, and γ is the polytropic constant, usually 7.

The term B is obtained from equation (20):

$$B = \frac{c_0^2 \rho_0}{\gamma} \quad (20)$$

where c_0 is the magnitude of sound velocity in the fluid.

For the Tait equation to be applied in predicting the dynamic pressure, it must be ensured that the Mach number is not greater than 0.10. In simulations, the term B had the value 0.85×10^5 Pa.

The free surface particles were marked and the constant pressure condition (zero) was applied first and renewed in every numerical iteration. The time step employed was 1.00×10^{-5} s. The simulations were run for a physical time of 0.30 s (30,000 numerical iterations). For both

geometries simulated, the term related to the artificial viscosity has been added to the terms of pressure in the SPH approximations for the momentum balance equation (to avoid numerical instabilities and interpenetration between particles). The formulation used in the modelling of artificial viscosity is presented in [4, 8]. The coefficients α_π and β_π (present in that formulation) received the values 0.20 and 0.30; and 0.00, respectively. The laminar flow modelling has been employed. The correction of the particles' physical properties (density and pressure gradient) located near the boundaries have been made using the Corrected Smoothed Particle Method (CSPM), [9].

In the boundary treatment technique, the geometric boundary conditions have used. The collisions of the fluid particles against the solid walls (considered as well-defined planes) have been studied.

SPH is a model of one particle; that is, collisions between particles are not considered. An algorithm for the study of collisions and trajectories of the particles that collide against the solid walls of the domain has been implemented. After carrying out the temporal integration and obtaining the position of the centre of mass of each fluid particle by the SPH method (in the situation in which there are no walls delimiting the domain), the collision algorithm, based on mathematical and geometry fundamentals, brings the particles back into the domain. This is based on the fact that the fluid particles could not escape the domain due to the presence of the predefined geometrical planes. The degree of elasticity of the collisions was measured by a coefficient of restitution of kinetic energy (CR). In this work, collisions have been considered perfectly elastic (CR = 1.00), [8].

After the solution of equation (4), temporal integration for the position and velocity has been solved using the Euler method.

Figures 16 to 18 show the experimental and numerical results for both simulated domains.

Table III shows the percentage differences between the abscissas of the wave fronts obtained by SPH and experimentally, calculated using the equation (21):

$$\Delta x_p = \frac{(x_{exp} - x_{SPH})}{x_{exp}} \times 100 \quad (21)$$

where x_{exp} and x_{SPH} are the abscissas of the wave fronts obtained by experiments and SPH, respectively, and Δx_p is the percentage difference between the abscissas of the wave fronts.

From the Table III analysis, it can be seen that the greatest differences between the wave fronts occurred in the initial stages of dam breaking. With the time progress, the differences decreased, as shown at time 0.20 s. The smaller differences between wave fronts (1.03% in the first and 2.33% in the second geometry) have occurred when it was used the coefficient α_π with the value 0.30. The numerical results showed a good agreement with experimental results reported in the literature [20].

Figure 15: The 2-D computational domains simulated and the initial distribution of particles for the dam breaking.

Figure 16: Experimental and numerical results for the first simulated geometry, with the implementation of the artificial viscosity ($\alpha_\pi = 0.20$).

Figure 17: Experimental and numerical results for the first simulated geometry, with the implementation of the artificial viscosity ($\alpha_\pi = 0.30$).

Table 3: Percentage differences between abscissas of wave fronts (comparison between experimental and numerical results)

First Domain Simulated					Second Domain Simulated				
α_π	t (s)	x_{exp} (m)	x_{SPH} (m)	Δx_p (%)	α_π	t (s)	x_{exp} (m)	x_{SPH} (m)	Δx_p (%)
0.20	0.10	19.00	18.07	4.89	0.20	0.10	22.00	19.00	13.64
	0.20	29.00	30.28	-4.41		0.20	36.00	37.00	-2.78
0.30	0.10	19.00	17.58	7.47	0.30	0.10	22.00	18.07	17.86
	0.20	29.00	28.70	1.03		0.20	36.00	35.16	2.33

Figure 18: Experimental and numerical results for the second simulated geometry, with the implementation of the artificial viscosity. In (a), $\alpha_\pi = 0.20$ and in (b), $\alpha_\pi = 0.30$.

Figure 19: Flowchart of the numerical code implemented for dam breaking.

In the first geometry simulations, CPU processing times were 2 h 21 min 24 s ($\alpha_\pi = 0.20$) and 2 h 22 min 40 s ($\alpha_\pi = 0.30$). In the second geometry, CPU times were 4 h 49 min 8 s ($\alpha_\pi = 0.20$) and 4 h 49 min 31 s ($\alpha_\pi = 0.30$).

In Figure 19, a flowchart shows the numerical code implemented for solution of dam break problem.

5. Conclusion

Due to the advances in computer processing techniques, there has been an increase in the applications of the particle methods for solution of problems in fluid mechanics and transport phenomena areas. SPH is a particle method that has advantages compared to traditional

methods using meshes, e.g. better visualisation of the spatio-temporal evolution of the flow, and a lower computational cost in the study of complex geometries with topological changes or free surfaces beyond the control of undesirable numerical diffusion. From the graphical analysis of the results, by viewing the properties of particles (which may be fields of positions, velocities, accelerations, pressures, among others) a better understanding of the physical phenomenon that occurs is reached.

The learning of the Lagrangian modelling and numerical particle methods is a current need in the Physics courses. The recent progress of processing techniques, the rapid spread of multi-core processors, and parallelisation, make possible the use of particle methods for physical problems solving. In addition, there is an increase in re-

searches aimed at the development of Lagrangian models applicable to physical problems and the need to development and use new computational tools.

The computational code presented, and validated for the study of certain physical problems, is a valuable tool and points to the future, as an instrument for the application of particle modelling in Physics teaching.

References

- [1] M.B. Liu and G.R. Liu, Archives of Computational Methods in Engineering **17**, 25 (2010).
- [2] P.J. Pritchard, *Fox and McDonald's Introduction to Fluid Mechanics* (John Wiley & Sons, Inc., New Jersey, 2011), 8th ed.
- [3] A.J.C. Crespo, J.M. Dominguez, A. Barreiro, M. Gómez-Gesteira and B.D. Rogers, PLoS ONE **6**, e20685 (2011), available at <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0020685>.
- [4] M.B. Liu and G.R. Liu, *Smoothed Particle Hydrodynamic, A Meshfree Particle Method* (World Scientific Publishing Co. Pte. Ltd., Singapore, 2003).
- [5] M.G. Gesteira, B.D. Rogers, R.A. Dalrymple, A.J.C. Crespo and M. Narayanaswamy, *User Guide for SPHysics Code* (Manchester, 2010), available at [https://wiki.manchester.ac.uk/sphysics/index.php?title=\\$Special:UserLogin{%&}returnto=SPHYSICS+2D+Download+v2.2](https://wiki.manchester.ac.uk/sphysics/index.php?title=$Special:UserLogin{%&}returnto=SPHYSICS+2D+Download+v2.2), accessed in February 21, 2017.
- [6] L.B. Lucy, Astronomical Journal **82**, 1013 (1977).
- [7] J.P. Morris, P.J. Fox and Y. Zhu, Journal of Computational Physics **136**, 214 (1997).
- [8] C.A.D. Fraga Filho, *Estudo da Fase Gravitacional-inercial do Espalhamento de Óleo em Mar Calmo empregando o Método Lagrangiano de Partículas Smoothed Particle Hydrodynamics*. PhD Thesis, Federal University of Espírito Santo, Brazil, 2014, available at http://cfd.mace.manchester.ac.uk/sph/SPH_PhDs/2014/Carlos_Alberto_DUTRA_FRAGA_FILHO_PhD_Thesis_2014.pdf, accessed in February 21, 2017.
- [9] J.K. Chen, J.E. Beraun and T. C. Carney, International Journal for Numerical Methods in Engineering **46**, 231 (1999).
- [10] R. Courant, K. Friedrichs and H. Lewy, IBM Journal **11**, 215 (1967).
- [11] C.A.D. Fraga Filho and J.T.A. Chacaltana, International Review on Modelling and Simulations **7**, 994 (2014).
- [12] J.C. Tanenhill, D.A. Anderson and R.H. Pletcher, *Computational Fluid Mechanics and Heat Transfer* (Taylor and Francis, London, 1997), 2nd ed.
- [13] C.A.D. Fraga Filho, D.F. Pezzin and J.T.A. Chacaltana, in: *Proceedings of the Thirteenth International Conference on Simulation and Experiments in Heat and Mass Transfer (Heat Transfer XIII)*, edited by B. Sundén and C.A. Brebbia (WIT Press, Southampton, 2014), p. 15.
- [14] G.K. Batchelor, *An Introduction to Fluid Dynamics* (Cambridge University Press, Cambridge, 2000), 3rd ed.
- [15] A. Vorobyev, *A Smoothed Particle Hydrodynamics Method for the Simulation of Centralized Sloshing Experiments* (KIT Scientific Publishing, Wissenschaftsverlag, 2013).
- [16] L. Goffin, S. Erpicum, B.J. Dewals, M. Piroton and P. Archambeau, in: *Proceedings of the 3rd SimHYDRO Conference*, edited by P. Gourbesville, J.A. Cunge and G. Caignaert (Springer, Nice, 2014), p. 175.
- [17] S.P. Korzilius, A.C.H. Kruisbrink, W.H.A. Schilders, M.J.H. Anthonissen and T. Yue, *CASA-Report 14-15* (Technische Universiteit Eindhoven, Eindhoven, 2014).
- [18] G. Fourtakas, R. Vacondio and B. Rogers, International Journal Numerical Methods in Fluids **78**, 475 (2015).
- [19] W.G. Hoover, *Smooth Particle Applied Mechanics (The State of the Art)*, *Advanced Series in Nonlinear Dynamics (25)* (World Scientific Publishing Co. Pte. Ltd., Singapore, 2006).
- [20] M.A. Cruchaga, D.J. Celentano and T.E. Tezduyar, Computational Mechanics **39**, 453 (2007).