UNIVERSIDADE DE VIGO
DEPARTAMENTO DE FÍSICA APLICADA

PhD Thesis

Application of the
Smoothed Particle Hydrodynamics
model SPHysics
to free-surface hydrodynamics

Alejandro Jacobo Cabrera Crespo

June 2008

.

Informe del director

D. RAMÓN GÓMEZ GESTEIRA, Profesor Titular de Universidad del Departamento de Física Aplicada de la Universidad de Vigo

CERTIFICA que la presente memoria *Development of the Smoothed Particle Hydrodynamics model SPHysics*, resume el trabajo de investigación realizado, bajo su dirección, por DON ALEJANDRO JACOBO CABRERA CRESPO y constituye su Tesis Doctoral para optar al Grado de Doctor en Ciencias Físicas.

Y para que conste y en cumplimiento de la legislación vigente, firma el presente certificado.

Fdo. Dr. Ramón Gómez Gesteira

.

Acknowledgements / Agradecimientos

En primer lugar tengo que darle las gracias a Moncho Gómez-Gesteira, por su confianza en mí, su paciencia, todo el tiempo que me ha dedicado, todo lo que he viajado y porque todo lo que he conseguido hasta ahora se lo debo a él.

A Maite de Castro Rodríguez, Inés Álvarez y Nieves Lorenzo por toda la ayuda prestada siempre que la he necesitado y por reírse con mis bromas.

A mis actuales compañeros de laboratorio: Isabel y Alex, por hacer mucho más ameno y divertido el trabajo diario y porque son algo más que mis compañeros. Y por supuesto a todos los que fueron compañeros míos durante mi periodo de doctorando.

I would like to thank my SPH friends; Ben (Dr. Benedict Rogers) and Muthu (almost Dr. Muthukumar Narayanaswamy). Because of our job, I knew two brilliant researchers and two brilliant persons. And, for sure, I am grateful for his help in everything I needed.

I also want to thank Prof. Robert A. Dalrymple for the visiting period I could spend working at the Johns Hopkins University of Baltimore, MD, USA and for all the useful discussions and corrections in the articles we write together.

A mi familia, que por supuesto es lo más importante que tengo, así que me gustaría mencionarlos a todos; a mis abuelos de Bexe y Xinzo, tíos, padrinos y primos y especialmente a mi madre y a mi hermano, porque sin ellos no hubiese llegado a donde he llegado. Este trabajo está dedicado a ellos.

A toda la gente que he conocido en la Universidad; Alberto, Ana, Diego, Eli, Elena, Eva, Feliciano, Fernando, Iria, Javi, Laura, María Montero, María Rodas, Maruxa, Montse, Noemi, Paloma, Paula, Servando y otros muchos.

A mis amigos de toda la vida; Álvaro, Antonio, David, Foski, Jacobo, Miguel, Noel, Pedro, Hipo, Iván, Victor y Alex, porque no sólo están a mi lado cuando hay que ir de fiesta. Les dedicaría un párrafo a cada uno, pero que se conformen así.

.

Abstract

Smoothed Particle Hydrodynamics, SPH, is a purely Lagrangian method developed during the seventies as an attempt to model continuum physics avoiding the limitations of finite difference methods.

It has been used in a wide variety of astrophysical applications and hydrodynamical problems. In coastal engineering, the problems are associated with propagating waves across the nearshore region, through the breaker line, and up the beach face. This area is difficult to model due to the moving boundary at the shoreline, wave breaking, and the variation in water depth from at least intermediate water depth to extremely shallow water. The SPH method is capable of dealing with problems with free surface, deformable boundary, moving interface, especially wave propagation and solid simulation.

The SPH model, as one of the oldest Meshfree Particle Methods (MPM), is quickly approaching its mature stage. With the continuing improvements and modifications, the accuracy, stability and adaptivity of the model have reached an acceptable level for practical engineering applications.

The principle advantages of SPH arise directly from its Lagrangian nature. A Lagrangian approach can tackle difficulties related with lack of symmetry or large voids much more efficiently than Eulerian methods can do. There are no constraints imposed either on the geometry of the system or in how far it may evolve from the initial conditions. Since there is no mesh to distort, the method can handle large deformations in a pure Lagrangian frame. Thus, material interfaces are followed naturally, and complex constitutive behavior can be implemented simply and accurately.

The foundation of SPH is interpolation theory. The conservation laws of continuum fluid dynamics, in the form of partial differential equations, are transformed into their particle forms by integral equations through the use of an interpolation function usually known as kernel. Computationally, information is known only at discrete points, so that the integrals are evaluated as sums over neighboring points.

This PhD is focusing on a code version of SPH called SPHysics. The report describes the development of the method and its application to water waves.

SPHysics is developed jointly by researchers at the Johns Hopkins University (U.S.), the University of Vigo (Spain), the University of Manchester (U.K.) and the University of Rome La Sapienza (Italy). Furthermore, it is an open code, so other researchers can contribute to the SPHysics project, by sending improvements.

All applications presented in this manuscript were developed using different versions of SPHysics with their own limitations. The model has been validated in two and three dimensional version using different experimental data.

The role of boundary conditions, kernel renormalization, floating bodies, multiphase modeling, parallel versions and coupling to generation propagation models are still an open field of research.

Contents

Chapter 1

Introduction

Background knowledge of numerical simulation is introduced below. The basic ideas of grid-based numerical methods, meshfree methods and particle methods are briefed (following [Liu and Liu, 2003]). Some general features, advantages and applications of smoothed particle hydrodynamics are described. Finally, the open-source code SPHysics is addressed.

1.1 Background

In order to understand and then build a physical or engineering system, it must be attempted a very sophisticated process of modeling, simulation, visualization, analysis, designing, prototyping, testing, fabrication and construction. The process is very often repetitive in nature, so some of the processes are repeated based on the results obtained at the present stage to obtain optimal behavior for the system under construction. This report deals with physical, mathematical and computational modeling and numerical simulation.

1.1.1 Numerical simulation

The key idea is translating aspects of a physical problem into a numerical model. Thus, numerical simulation can be used instead of performing expensive and large experiments. The role of numerical simulation is becoming more important in engineering since it is a suitable tool to solve complex problems.

1.1.2   Grid-based methods

In describing the physical governing equations, the Lagrangian description and the Eulerian description can be used.

The Lagrangian description is typically represented by the finite element method (FEM) ([Zienkiewicz and Taylor, 2000]; [Liu and Quek, 2003]). The grid or mesh is fixed to the material along the simulation. The track of the material is known at grid points. Time-history data (such as position, mass, momentum, energy, etc.) are computed at each point. The mesh is deformed with the material. It is easy to track a moving boundary and interface. However large deformations are difficult to handle. Thus, Lagrangian methods are well suited to solve computational solid mechanics (CSM) problems, where the deformation is less large as that in the fluid flows.

The Eulerian description is typically represented by the finite volume method (FVM) ([Versteeg and Malalasekera, 1995]; [Leveque, 2002]). The grid or mesh is fixed in the space, but not to the material, which moves across the grid cells. As opposed to the formulation above, time-history is difficult to obtain at a point attached on material; moving boundaries and interfaces are difficult to track and irregular geometries are difficult to model accurately. However large deformations are easy to handle because the mesh remain unchanged preventing problems typically associated to Lagrangian methods like the lack of accuracy or the quasi- unlimited decrease in time step fixed by the smallest element. Eulerian methods are usually employed to study explosion and high velocity impacts.

A comparison between Lagrangian and Eulerian methods can be seen in table 1.1 (Table 1.2 in [Liu and Liu, 2003]).

	Lagrangian methods	Eulerian methods
Grid	Attached on the moving material	Fixed in the space
Track	Movement of any point on materials	Mass, momentum and energy flux across grid nodes and mesh cell boundary
Time history	Easy to obtain time-history data at a point attached on materials	Difficult to obtain time-history data at a point attached on materials
Moving boundary and interface	Easy to track	Difficult to track
Irregular geometry	Easy to model	Difficult to model with good accuracy
Large deformation	Difficult to handle	Easy to handle

Table 1.1: Comparisons of Lagrangian and Eulerian methods.

1.1.3 Meshfree methods

Grid-based numerical methods present difficulties in some aspects, which limit their applications in many complex problems. One of the main limitations is the grid generation, which is not always a straightforward process and can constitute an expensive task, both in terms of computational time and mathematical complexity. Meshfree methods facilitate the simulation of problems that require the ability to treat large deformations, advanced materials, complex geometry, nonlinear material behavior, discontinuities and singularities. Meshfree methods are used for solid mechanics as well as for fluid dynamics. They share some common features, but they are different in the methods of approximation, see table 1.2 (Table 1.3 in [Liu and Liu, 2003]).

Methods	Methods of approximation
Smoothed particle hydrodynamics SPH	Integral representation
Finite point method	Finite difference representation
Diffuse element method (DEM)	Moving least square (MLS) approximation Galerkin method
Element free Garlekin (EFG) method	MLS approximation Garlekin method
Reproduced kernel particle method (RKPM)	Integral representation Garlekin method
HP-cloud method	MLS approximation Partition of unity
Free mesh method	Garlekin method
Meshless local Petrov-Garlekin (MLPG) method	MLS approximation Petrov-Garlekin method
Point interpolation method (PIM)	Point interpolation (Radial and Polynomial basis), Garlekin method, Petrov-Garlekin method
Meshfree weak-strong form (MWS)	MLS, PIM, radial PIM (RPIM), Collocation plus Petrov-Garlekin

Table 1.2: Some typical meshfree methods in chronological order.

1.1.4 Meshfree particle methods

Meshfree particle methods (MPM) treat the system as a set of particles, which represents a physical object or a parcel of the domain. For Computational Fluid Dynamics (CFD) problems, variables such as mass, momentum, energy, position, etc. are computed at each particle. Some examples of these methods are shown in table 1.3 (Table 1.4 in [Liu and Liu, 2003]).

Methods
Molecular dynamics (MD)
Monte Carlo
Direct simulation Monte Carlo (DSMC)
Dissipative particle dynamics (DPD)
Lattice gas Cellular Automata (CA)
Lattice Boltztmann equation (LBE)
Particle-in-Cell (PIC)
Marker-and-Cell (MAC)
Fluid-in-Cell (FLIC)
Moving Particle Semi-implicit (MPS)
Discrete element method (DEM)
Vortex methods
Smoothed Particle Hydrodynamics (SPH)

Table 1.3: Some typical particle methods.

1.2 Smoothed Particle Hydrodynamics

Smoothed Particle Hydrodynamics is a meshfree, Lagrangian, particle method for modeling fluid flows. SPH integrates the hydrodynamic equations of motion on each particle in the Lagrangian formalism. Relevant physical quantities are computed for each particle as an interpolation of the values of the nearest neighboring particles, and then particles move according to those values. The conservation laws of continuum fluid dynamics, in the form of partial differential equations, are transformed into their particle forms by integral equations through the use of an interpolation function that gives the kernel estimate of the field variables at a point. Computationally, information is known only at discrete points (the particles), so that the integrals are evaluated as sums over neighboring particles.

1.2.1 Advantages and disadvantages of SPH

The Lagrangian nature of SPH provides the method with some advantages when compared to the usual limitations in Eulerian methods.
- The density number of particles increases in regions where the fluid is present, in such a way that the computational effort is mainly concentrated in those regions. So time is not wasted calculating empty areas.
- There are no constraints imposed either on the geometry of the system or in how far it may evolve from the initial conditions, such that the initial conditions can be easily programmed without need of complicated gridding algorithms as used in finite element methods.
- Including other physical process in the code is straightforward.

However, the method also presents some intrinsic limitations.
- Boundary condition implementation is a hard task and fluid particles penetration into boundaries must be avoided.
- The interpolation method used in SPH is very simple and it will be strongly affected by particle disorder. SPH gives reasonable results for the first order gradients (although [Bonet and Lok, 1999] recommend gradient corrections), but they can be worse for higher order derivatives. Sometimes, it is necessary to use special techniques when second derivatives are included.
- The method is typically slower computationally when compared to other modern grid-based methods, as the time step is based on a sound speed in the fluid, although new research has been developed during the last few years to overcome these limitations.
- A kernel with spherical symmetry is often used. Distribution of particle neighbors must be approximately isotropic for the interpolation form in order to work properly, which is not fulfiled in astrophysical processes involving the formation of sheets or disks.

1.2.2 Applications of SPH

SPH was developed as an attempt to model continuum physics, avoiding the limitations of finite difference methods. It was first applied thirty years ago to solve astrophysical problems ([Lucy, 1977]; [Gingold and Monaghan, 1977]), since the collective movement of those particles is similar to the movement of a fluid so it can be modeled by the governing equations of the classical Newtonian hydrodynamics. The numerical method has been shown to be robust and applicable to a wide variety of fields:

ASTROPHYSICS:
- Stellar collisions: [Faber and Rasio, 2000]; [Faber and Manor, 2001];
[Benz, 1988]; [Benz, 1990]; [Monaghan, 1992]; [Frederic and James, 1999]
- Moon formation and impacts problems: [Benz, 1989]
- Cloud fragmentations and collisions: [Durisen et al., 1986]
- Cosmology : [Evrad, 1988]; [Shapiro et al., 1996]
- Supernovas explosion: [Nagasawa et al., 1988]; [Herant and Benz, 1991]
- Collapse and formation of galaxies: [Monaghan and Lattanzio, 1991];
[Berczik, 2000]; [Berczik and Kolesnik, 1993]; [Berczik and Kolesnik, 1998]
- Evolution of the universe: [Monaghan, 1990]

MAGNETO-HYDRODYNAMICS:
- Magnetic collapse of gas clouds: [Habe, 1989]
- Alfvenic waves propagation: [Phillips and Monaghan, 1985]
- Development of expansive wave in a magnetic cloud:
[Stellingwerf and Peterkin, 1990]

SOLID MECHANICS:
- Impact problems: [Johnson et al., 1996]; [Libersky and Petscheck, 1991];
[Libersky and Petscheck, 1993]
- Fractures simulation: [Benz and Asphaug, 1993]
- Impacts of solids simulation: [Benz and Asphaug, 1994]
- Study of brittle solids: [Benz and Asphaug, 1995]
- Metal forming: [Bonet and Kulasegaram, 2000]

FLUID DYNAMICS:
- Multi-phase flows: [Monaghan and Kocharyan, 1995]
- Heat conduction: [Chen et al., 1999]
- Underwater explosions: [Swegle and Attaway, 1995]
- Gravity currents: [Monaghan, 1996]; [Monaghan et al., 1999]
- Free-surface flows: [Monaghan, 1994]; [Monaghan and Kos, 1999];
[Monaghan and Kos, 2000]

RECENT APPLICATIONS IN CFD

LONG WAVES:
- [Panizzo and Dalrymple, 2004] study that underwater landslide generates waves.
- [Panizzo et al., 2006] and [Rogers and Dalrymple, 2007] show that SPH model allows studying a sliding mass impacting a body of water.
- [Del Guzzo and Panizzo, 2007] use the nonlinear shallow water equations for other long wave solutions.

SEDIMENT TRANSPORT AND SUSPENSION:
- [Zou and Dalrymple, 2006] and [Zou, 2007] have examined suspended sediment under waves using the Lagrangian form of convection-diffusion equation.

TWO-PHASE SPH:
- [Colagrossi and Landrini, 2003] examine a rising bubble in a fluid.
- [Cuomo et al., 2006] also examine two-phase flows.
- [Hu and Adams, 2007] determine a surface force between phases using an incompressible multi-phase SPH model.

INCOMPRESSIBLE SPH:
- [Lo and Shao, 2002] and [Ellero et al., 2007] solve an elliptic Poisson equation for pressure, which provides a considerable computational load.
- [Shao et al., 2006] use an incompressible SPH for run-up.

BREAKING WAVES AND WAVE IMPACT STUDIES ON OFFSHORE STRUCTURES:
- [Gómez-Gesteira and Dalrymple, 2004] the impact of a single wave generated by a dam break with a tall structure is modeled with a three-dimensional version of SPH.
- [Shao and Gotoh, 2004] analyze the interaction between waves and a floating curtain wall attached to the bottom.
- [Gómez-Gesteira et al., 2005] analyze green water overtopping of horizontal decks.
- [Lee et al., 2006] study the run-up of Boussinesq and SPH waves on a coastal structure.
- [Dalrymple and Rogers, 2006] examine the coherent turbulence under breaking waves.
- [Crespo et al., 2007c] compare SPH results to laboratory experiments from [Janosi et al., 2004].
- [Crespo et al., 2007a] the role of protecting barriers (dikes) to mitigate the force and moment exerted by large waves on the structure is analyzed.

1.2.3 SPHysics

SPHysics is a Smoothed Particle Hydrodynamics (SPH) code inspired by the formulation of [Monaghan, 1992]. It is a joint collaboration between several researchers at the Johns Hopkins University (U.S.), the University of Vigo (Spain), the University of Manchester (U.K.) and the University of Rome La Sapienza (Italy). The first version of SPHysics was released on August 2007 and it is available for public use at http://wiki.manchester.ac.uk/sphysics (see Figure 1.1). A users manual and several testcases are provided.

SPHysics code can simulate various phenomena including wave breaking, dam breaks, sliding objects, wave impact on a structure, etc. The model has been put into modular form and a variety of features are avaliable to choose different compiling options (see table 1.4).

Dimensions	2D
	3D
Time scheme	Predictor-Corrector algorithm
	Verlet algorithm
Time step	Constant
	Variable
Kernel functions	Gaussian
	Quadratic
	Cubic Spline
	Wendland kernel
Kernel corrections	Kernel gradient correction
Solid boundary conditions	Dynamic boundaries
	Repulsive forces
	Periodic open boundaries
Viscosity treatment	Artificial Viscosity
	Laminar Viscosity
	Sub-Particle Scale (SPS) Turbulence Model
Density filter	Shepard filter
	MLS filter
Different types of moving objects	Moving Gate
	Wavemaker
	Sliding Wedge
Operating systems and compilers	Windows: Compaq Visual Fortran
	Linux: GPL gfortran and Intel fortran compiler
	Mac: GPL gfortran

Table 1.4: Compiling options available in SPHysics.

One of the purpose of this code is encourage other researchers to try SPH more easily and to contribute to the SPHysics project. In this way they can send improvements in the code back to the developers or can report on any error found in the code.

See APPENDIX I where all details about installation and source code of SPHysics are described in detail.

Figure 1.1: SPHysics website.
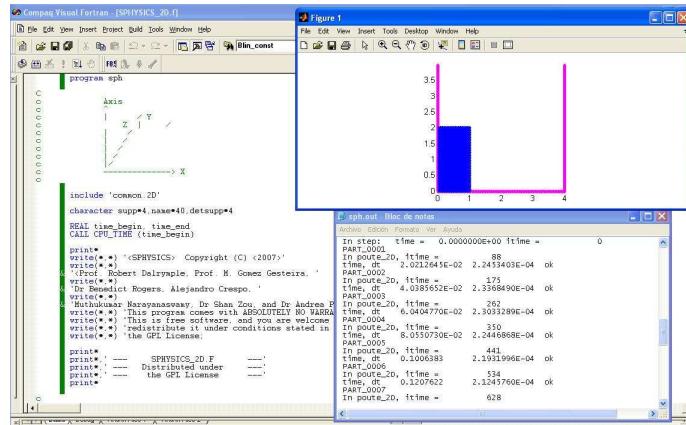
Figure 1.2: A typical screenshot running SPHysics on Windows.

1.3 Thesis outline

The thesis provides a description of the SPHysics method and different CFD applications. It is organized in a total of eight chapters that are briefed as follows:

Chapter 1 introduces background knowledge of numerical simulation. The features of the grid-based numerical methods and the basic ideas of meshfree and particle methods are briefed following [Liu and Liu, 2003]. Some general features, advantages and applications of smoothed particle hydrodynamics are described. SPHysics code associated with this thesis is introduced.

Chapter 2 provides fundamentals and basic concepts of the SPH method such as integral interpolants, smoothing kernels and the momentum conservation, mass conservation and energy conservation in terms of discrete notation.

Chapter 3 describes the implementation of SPHysics. Some numerical aspects such as choice of the kernel, numerical filters, choice of time step, computational efficiency, initial conditions and solid boundary treatment are discussed. Some tests and applications are simulated to study the role of boundary conditions and the formulation of viscosity terms. SPHysics theory and implementation chapters are based on the SPHysics users manual written by A.J.C. Crespo, M. Gómez-Gesteira, B.D. Rogers, R.A. Dalrymple and M. Narayanaswamy.

Chapter 4 describes one of the first applications in 2D, water overtopping. The experiment and the model calibration are addressed to analyze wave impact on a deck and jet formation under extreme waves. This chapter is based on the article [Gómez-Gesteira et al., 2005] whose main responsible is M. Gómez-Gesteira.

Chapter 5 shows the study of the dam break problem and the effect of standing water in front of the dam with the SPH model. The method will be shown to fit accurately experimental results. In addition, the model captures most of the features of a dam break over a wet bed, in particular it will allow analyzing the mixing and dissipation associated to the interaction between the dam break and the still water placed near bed. This chapter is based on the article [Crespo et al., 2007c] whose main responsible is A.J.C. Crespo.

Chapter 6 presents the interaction between large waves and structures by means of a three dimensional SPH method. Different scenarios for impact reduction are evaluated. In particular, a single dike was considered as the mitigation mechanism. Both the distance between protected structure and the dike and the dike height have proven to play a key role in the degree of protection. This chapter is based on the article [Crespo et al., 2007a] whose main responsible is A.J.C. Crespo.

Chapter 7 draws together conclusions and ongoing research.

Appendix A presents all details about installation and source code of SPHysics. See the SPHysics guide (section 3).

Chapter 2

SPH Theory

SPH method is a meshfree, stable and Lagrangian solver for free-surface hydrodynamics problems. The detailed formulation and these features of SPH will be addressed in this and following chapters and they will be demonstrated in some working examples in the later chapters. All the concepts, strategies and essential formulations discussed in this chapter are very useful in the development of the SPHysics code.

The main features of SPH method are described in detail in the following papers [Monaghan, 1982]; [Monaghan, 1992]; [Benz, 1990]; [Liu and Liu, 2003]; [Liu, 2003]; [Monaghan, 2005].

2.1   Integral interpolants

SPH is based on integral interpolants. The fundamental principle is to approximate any function $A(\mathbf{r})$ by (kernel approximation):

$$A\left(\mathbf{r}\right) = \int_{\Omega} A\left(\mathbf{r}'\right) W\left(\mathbf{r} - \mathbf{r}', h\right) d\mathbf{r}' \qquad (2.1)$$

where $\mathbf{r}$ is the vector position; W is the weighting function or kernel; h is called smoothing length and it controls the influence domain Ω (see Figure 2.1). Typically, value of h must be higher than initial particle separation.

The approximation 2.1, in discrete notation, leads to the following approximation of the function at a particle a, (particle approximation):

$$A\left(\mathbf{r}\right) = \sum_{b} m_b \frac{A_b}{\rho_b} W_{ab} \qquad (2.2)$$

where the summation is over all the particles within the region of compact support of the kernel function. The mass and density are denoted by m_b and ρ_b respectively and $W_{ab} = W\left(\mathbf{r}_a - \mathbf{r}_b, h\right)$ is the weight function or kernel.

One of the advantages of the SPH kernel approach is that the derivative of a function is calculated analytically, as compared to a method like finite differences, where the derivatives are calculated from neighboring points using the spacing between them. For the irregularly spaced SPH particles, this would be extremely complicated. The derivatives of this interpolation can be obtained by ordinary differentiation, neither a finite difference method is needed nor a mesh is used.

$$\nabla A\left(\mathbf{r}\right) = \sum_{b} m_b \frac{A_b}{\rho_b} \nabla W_{ab} \qquad (2.3)$$

thus, this equation is derived by performing the integral in Eq. 2.1 for a functional derivative, and applying integration by parts.

2.2 The smoothing kernel

The performance of an SPH model depends on the choice of the weighting functions. They should satisfy several conditions such as positivity, compact support, and normalization. Also, W_{ab} must be monotonically decreasing with increasing distance from particle a and behave like a delta function as the smoothing length, h, tends to zero:

$$\text{Positivity: } W\left(\mathbf{r} - \mathbf{r}', h\right) \geq 0 \quad \text{inside the domain } \Omega \tag{2.4}$$

$$\text{Compact support: } W\left(\mathbf{r} - \mathbf{r}', h\right) = 0 \quad \text{out of the domain } \Omega \tag{2.5}$$

$$\text{Normalization: } \int_{\Omega} W\left(\mathbf{r} - \mathbf{r}', h\right) d\mathbf{r}' = 1 \tag{2.6}$$

$$\text{Delta function behavior: } \lim_{h \to 0} W\left(\mathbf{r} - \mathbf{r}', h\right) d\mathbf{r}' = \delta\left(\mathbf{r} - \mathbf{r}'\right) \tag{2.7}$$

$$\text{Monotonically decreasing behavior of } W\left(\mathbf{r} - \mathbf{r}', h\right) \tag{2.8}$$
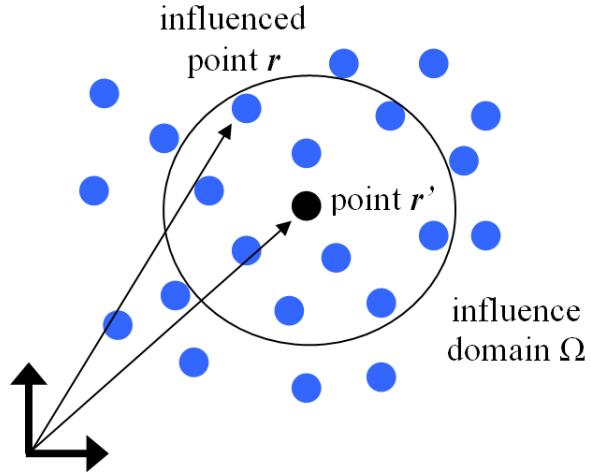
Figure 2.1: Sketch of the influence domain

Kernels depend on the smoothing length, h, and the non-dimensional distance between particles given by $q = r/h$, r being the distance between particles a and b. The parameter h controls the size of the area around particle a where contribution from the rest of the particles cannot be neglected.

2.3   Governing equations

The basic governing equations of fluid dynamics are based on the following three fundamental physical conservation laws:
 a) conservation of mass
 b) conservation of momentum
 c) conservation of energy

The SPH equations of motion are derived based on these governing equations in Lagrangian form.

2.3.1   Momentum equation

The momentum conservation equation in a continuum field is:

$$\frac{D\mathbf{v}}{Dt} = -\frac{1}{\rho}\nabla P + \mathbf{g} + \mathbf{\Theta} \qquad (2.9)$$

where $\mathbf{v}$ is velocity, P and ρ are pressure and density; $\mathbf{g} = (0, 0, -9.81)ms^{-2}$ is the gravitational acceleration; $\mathbf{\Theta}$ refers to the diffusion terms.

Different approaches, based on various existing formulations of the diffusive terms, can be considered in the SPH method to describe the momentum equation. Three different options for diffusion can be used in SPHysics: (i) artificial viscosity, (ii) laminar viscosity and (iii) turbulence modeling (laminar viscosity+ Sub-Particle Scale (SPS) Turbulence):

Artificial viscosity

The artificial viscosity proposed by [Monaghan, 1992] has been very often used due to its simplicity. In SPH notation, Eq. 2.9 can be written as

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} + \Pi_{ab} \right) \nabla_a W_{ab} + \mathbf{g} \qquad (2.10)$$

The pressure gradient term in symmetrical form is expressed in SPH notation as

$$-\frac{1}{\rho}\nabla P = -\sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} \right) \nabla_a W_{ab} \qquad (2.11)$$

where P and ρ are pressure and density.

$\prod_{ab}$ is the viscosity term:

$$\Pi_{ab} = \begin{cases} \frac{-\alpha \overline{c_{ab}} \mu_{ab}}{\overline{\rho_{ab}}} & \text{if } \mathbf{v}_{ab} \cdot \mathbf{r}_{ab} < 0, \\ 0 & \text{otherwise} \end{cases} \quad (2.12)$$

with:

$$\mu_{ab} = \frac{h \mathbf{v}_{ab} \cdot \mathbf{r}_{ab}}{\mathbf{r}_{ab}^2 + \eta^2}$$

where $\overline{\rho_{ab}} = \frac{1}{2}(\rho_a + \rho_b)$, $\overline{c_{ab}} = \frac{1}{2}(c_a + c_b)$; $\eta^2 = 0.01h^2$; α is a free parameter that can be changed according to each problem.

Artificial viscosity is used because it is robust for the applications presented in the following chapters. Other options for modeling viscosity which more closely resemble the full governing equations are available such as laminar viscosity and SPS turbulence model.

Laminar viscosity

The momentum conservation equation with laminar viscous stresses is given by

$$\frac{D\mathbf{v}}{Dt} = -\frac{1}{\rho}\nabla P + \mathbf{g} + \upsilon_0 \nabla^2 \mathbf{v} \quad (2.13)$$

where the laminar stress term simplifies ([Morris et al., 1997]; [Lo and Shao, 2002]) to:

$$\left(\nu_0 \nabla^2 \mathbf{v}\right)_a = \sum_b m_b \left(\frac{4\nu_0 \mathbf{r}_{ab} \nabla_a W_{ab}}{(\rho_a + \rho_b)|\mathbf{r}_{ab}|^2}\right) \mathbf{v}_{ab} \quad (2.14)$$

where ν_0 is the kinetic viscosity of laminar flow $(0.893 \cdot 10^{-6} m^2/s)$.

So, in SPH notation, Eq. 2.13 can be written as:

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2}\right)\nabla_a W_{ab} + \mathbf{g} + \sum_b m_b \left(\frac{4\nu_0 \mathbf{r}_{ab}\nabla_a W_{ab}}{(\rho_a + \rho_b)|\mathbf{r}_{ab}|^2}\right)\mathbf{v}_{ab} \quad (2.15)$$

Laminar viscosity and Sub-Particle Scale (SPS) Turbulence

In order to adequately represent the viscosity of the fluid and the turbulent motions, [Rogers and Dalrymple, 2004] and [Dalrymple and Rogers, 2006] used a large eddy simulation (LES) model ([Christensen, 2006]). This sub-grid scaling method, which uses a Smagorinsky ([Smagorinsky, 1963]) eddy viscosity term, is similar to that used by [Lo and Shao, 2002], except is a turbulence model for compressible flow.

[Gotoh et al., 2001] also use sub-particle scaling with the Moving Particle Semi-implicit (MPS) technique. However, for a compressible fluid, the procedure is slightly different and involves Favre averaging, which is a density-weighted time averaging scheme.

The momentum conservation equation is:

$$\frac{D\mathbf{v}}{Dt} = -\frac{1}{\rho}\nabla P + \mathbf{g} + \upsilon_{\mathbf{0}}\nabla^{\mathbf{2}}\mathbf{v} + \frac{1}{\rho}\nabla\overline{\tau} \tag{2.16}$$

where the laminar term can be treated following Eq. 2.14 and $\overline{\tau}$ represents the SPS stress tensor.

The eddy viscosity assumption (Boussinesq's hypothesis) is often used to model the SPS stress tensor using Favre-averaging (for a compressible fluid):

$$\frac{\tau_{ij}}{\rho} = 2\nu_t S_{ij} - \frac{2}{3}k\delta_{ij} - \frac{2}{3}C_I\Delta^2\delta_{ij}|S_{ij}|^2 \tag{2.17}$$

where τ_{ij} is the sub-particle stress tensor, $\nu_t = [min(C_S,\Delta l)]^2 \cdot |S|$ the turbulence eddy viscosity, k the SPS turbulence kinetic energy, C_S the Smagorinsky constant (0.12), $C_I = 0.0066$, Δl the particle-particle spacing, $|S| = (2S_{ij}S_{ij})^{1/2}$, S_{ij} the element of SPS strain tensor.

So, following [Dalrymple and Rogers, 2006], Eq. 2.16 can be written in SPH notation as

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b\left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} + \frac{\tau_a}{\rho_a^2} + \frac{\tau_b}{\rho_b^2}\right)\nabla_a W_{ab} + \sum_b m_b\left(\frac{4\nu_0\mathbf{r}_{ab}\nabla_a W_{ab}}{(\rho_a+\rho_b)|\mathbf{r}_{ab}|^2}\right)\mathbf{v}_{ab} + \mathbf{g} \tag{2.18}$$

2.3.2   Continuity equation

The fluid in the standard SPH formalism is treated as compressible, which allows the use of an equation of state to determine fluid pressure, rather than solving another differential equation. However, the compressibility is adjusted to slow the speed of sound so that the time step in the model (based on the speed of sound) is reasonable.

Changes in the fluid density were calculated by means of

$$\frac{d\rho_a}{dt} = \sum_b m_b\mathbf{v}_{ab}\nabla_a W_{ab} \tag{2.19}$$

instead of using a weighted summation of mass terms ([Monaghan, 1992]), since it is known to result in an artificial density decrease near fluid interfaces.

2.3.3 Equation of state

As mentioned above, the fluid in the standard SPH formalism is treated as weakly compressible, which facilitates the use of an equation of state to determine fluid pressure, which is much faster than solving an equation such as the Poissons equation.

Following [Monaghan, 1994] and [Batchelor, 1974], the relationship between pressure and density is assumed to follow the next expression, known as Tait's equation of state. It can be seen that a small oscillation in density may result in a large variation in pressure.

$$P = B\left[\left(\frac{\rho}{\rho_0}\right)^\gamma - 1\right] \tag{2.20}$$

The parameter B is a constant related to the bulk modulus of elasticity of the fluid; $\rho_0 = 1000.0 Kg/m^3$ is the reference density, usually taken as the density of the fluid at the free surface, γ is the polytrophic constant, usually between 1 and 7, the minus one term in the equation of state is to obtain zero pressure at a surface.

This compressible fluid permits a speed of sound, c, which is given by square root of the derivative of this equation of state with respect to density:

$$c^2(\rho) = \frac{\partial P}{\partial \rho} = \frac{B\gamma}{\rho_0}\left(\frac{\rho}{\rho_0}\right)^{\gamma-1} = \frac{B\gamma}{\rho_0^\gamma}\rho^{\gamma-1} \tag{2.21}$$

$$c_0^2 = c^2(\rho_0) = \frac{\partial P}{\partial \rho}|_{\rho=\rho_0} = \frac{B\gamma}{\rho_0} \tag{2.22}$$

where c_0 is the speed of sound at the reference density (at the surface of the fluid); the constant B is shown to be equal to $B = c_0^2\rho_0/\gamma$, it sets a limit for the maximum change in the density.

The choice of B is going to play a key role since it determines the speed of sound. Using a value corresponding to the real value of the speed of sound in water, a very small time step must be chosen for numerical modeling, based on the Courant-Fredrich-Levy condition (section 3.6). Monaghan showed that the speed of sound could be artificially slowed significantly for fluids without affecting the fluid motion, however [Monaghan, 1994] suggests that the minimum sound speed should be about ten times greater than the maximum expected flow speeds.

2.3.4 Moving the particles

Particles are moved using the XSPH variant ([Monaghan, 1989])

$$\frac{d\mathbf{r}_a}{dt} = \mathbf{v}_a + \varepsilon \sum_b \frac{m_b}{\overline{\rho}_{ab}} \mathbf{v}_{ba} W_{ab} \qquad (2.23)$$

where $\overline{\rho}_{ab} = \frac{1}{2}\left(\rho_a + \rho_b\right)$ and ε is a constant, whose values ranges between zero and unity, $\varepsilon = 0.5$ is often used.

This method is a correction for the velocity of a particle a. This velocity is recalculated taking into account the velocity of that particle and the average velocity of all particles that are interacting with particle a. Because of the compact support of the kernel, only the closest neighborhood will be included.

This correction lets particles to be more organized and, for high fluid velocities, helps to avoid particle penetration.

2.3.5 Energy conservation

During the simulation, kinetic, potential and thermal energy are calculated. The thermal energy associated to each particle using artificial viscosity is calculated using the expression given by [Monaghan, 1992]

$$\frac{de_a}{dt} = \frac{1}{2} \sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} + \Pi_{ab} \right) \mathbf{v}_{ab} \nabla_a W_{ab} \qquad (2.24)$$

The total energy of the system is calculated as the sum of kinetic, potential and thermal energy. Energies represented in Figure 2.2 correspond to the collapse of a water column.
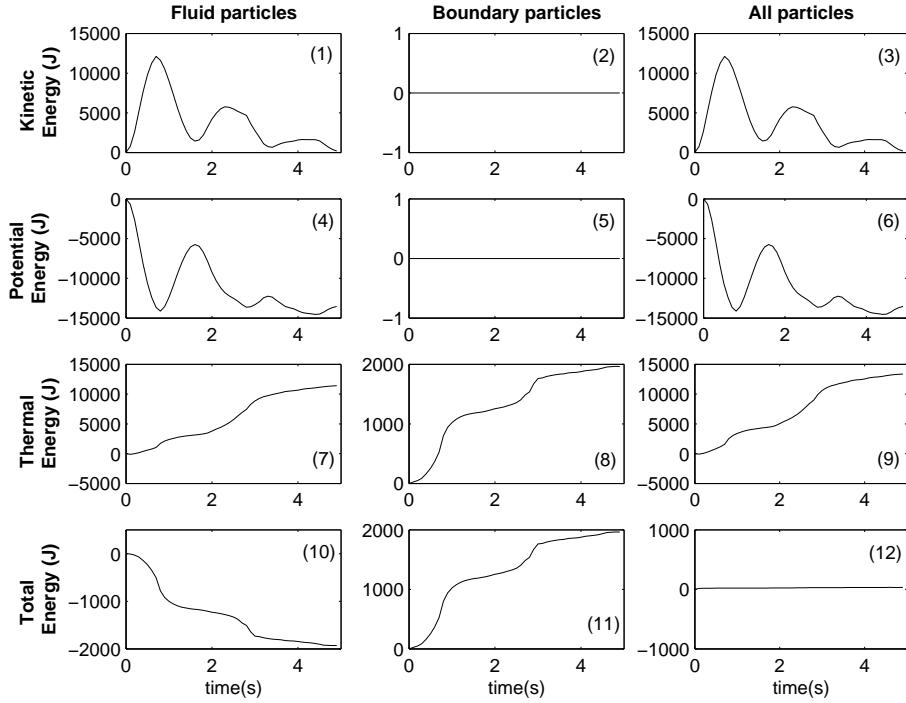
Figure 2.2: Time variation of the system energy during dam-break evolution.

Analyzing the figure in detail, it is observed how kinetic energy for stationary boundaries is always zero (2). The initial potential energy is set to zero (5). Thermal energy is calculated following equation 2.24. Frame 12 in Figure 2.2 represents the total energy of the system, the total energy of the fluid particles plus total energy of boundaries. According to [Monaghan, 1992], energy is conserved in the limit of 0.5% in 400 time steps. In this simulation, energy increases 0.3% in 500 time steps, so energy variation lies inside the limits proposed by Monaghan.

Chapter 3

SPHysics Implementation

Although SPH equations have been described in the previous chapter, some numerical techniques are used to improve the numerical accuracy and to reduce the computational time in SPHysics code. Also different choices of initial and boundary conditions are described in detail in these sections.

3.1   Kernel choice

Several kernel approximations are described in [Monaghan, 1992]; [Liu, 2003]; [Liu and Liu, 2003]; [Monaghan, 2005].

In the particular case of SPHysics, any of the following kernels can be used:

1. Gaussian.

2. Quadratic.

3. Cubic Spline.

4. Quintic.

1) Gaussian:

$$W\left(\mathbf{r}, h\right) = \alpha_D \cdot \exp(-q^2) \qquad 0 \leq q \leq 2 \qquad\qquad (3.1)$$

where $q = r/h$, r being the distance between particles a and b and α_D (the dimensional factor) is $1/(\pi h^2)$ in 2D and $1/(\pi^{3/2} h^3)$ in 3D.
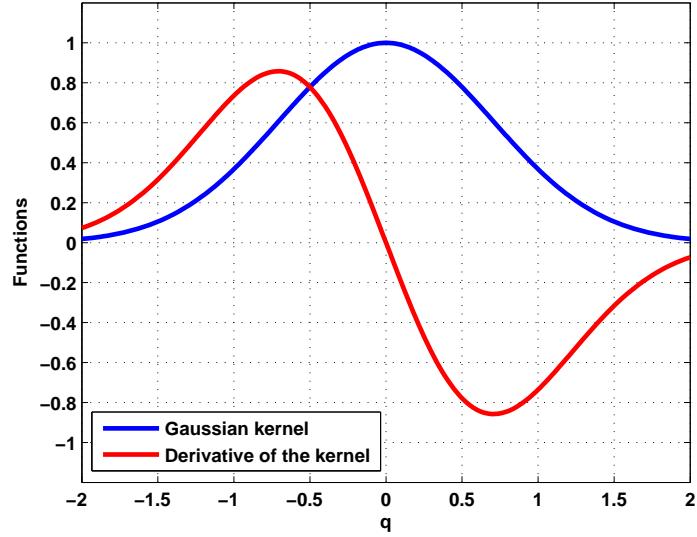
Figure 3.1: Gaussian kernel and its derivative divided by the dimensional factor α_D

Figure 3.1 shows the values of the Gaussian kernel and its derivative. The values of the functions are divided by the dimensional factor.

2) Quadratic:

$$W\left(\mathbf{r}, h\right) = \alpha_D \left[\frac{3}{16}q^2 - \frac{3}{4}q + \frac{3}{4}\right] \qquad 0 \leq q \leq 2 \qquad (3.2)$$

where α_D is $2/(\pi h^2)$ in 2D and $5/(4\pi h^3)$ in 3D.
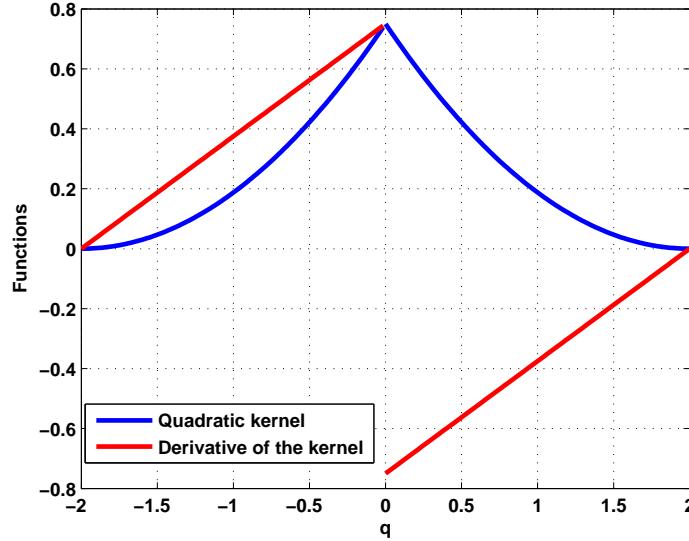
Figure 3.2: Quadratic kernel and its derivative divided by the dimensional factor α_D

[Johnson et al., 1996] used this smoothing function to simulate the high velocity impact problem. This function prevents particle clustering in compression problems (no tensile correction is needed, see section 3.2). The derivative of this kernel always increases as the particles move closer, and always decreases as they move apart.

3) Cubic spline:

$$W\left(\mathbf{r}, h\right) = \alpha_D \begin{cases} 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3 & 0 \leq q \leq 1 \\ \frac{1}{4}(2 - q)^3 & 1 \leq q \leq 2 \\ 0 & q \geq 2 \end{cases} \tag{3.3}$$

where α_D is $10/(7\pi h^2)$ in 2D and $1/(\pi h^3)$ in 3D.
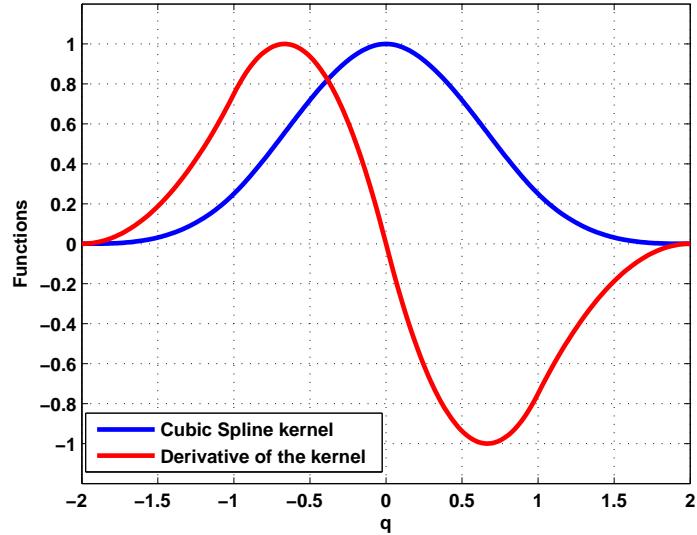
Figure 3.3: Cubic Spline kernel and its derivative divided by the dimensional factor α_D

The cubic spline kernel (introduced by [Monaghan and Lattanzio, 1985]) has been, so far, the most widely used smoothing function in the SPH literature since it resembles a Gaussian function while having a narrower compact support. Thus, one of the advantages of using this kernel instead of a Gaussian kernel is that it has a compact support (it is equal to zero for $q > 2$) and numerical computations are reduced.

4) Quintic ([Wendland, 1995]):

$$W\left(\mathbf{r}, h\right) = \alpha_D \left(1 - \frac{q}{2}\right)^4 (2q + 1) \qquad 0 \le q \le 2 \tag{3.4}$$

where α_D is $7/(4\pi h^2)$ in 2D and $7/(8\pi h^3)$ in 3D.
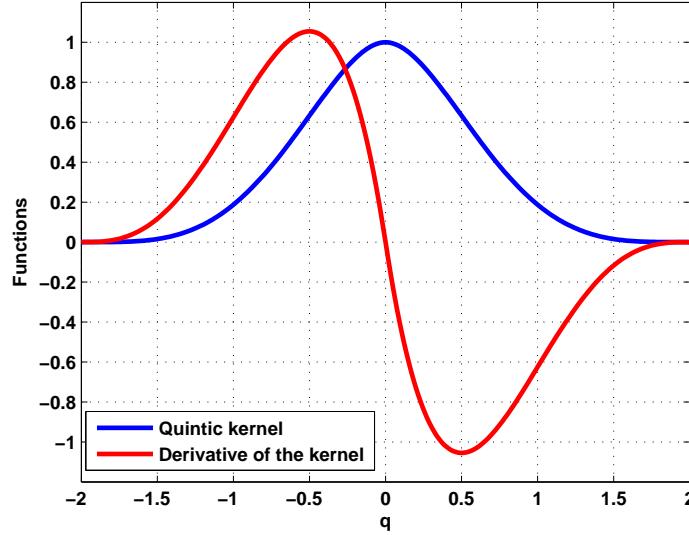
Figure 3.4: Quintic kernel and its derivative divided by the dimensional factor α_D

Following [Panizzo et al., 2007], results show that the best compromise between accuracy and time computation cost is reached by the use of the Wendland kernel. In general, the higher the order of the kernels, the greater the accuracy of the SPH scheme.

3.2   Tensile correction

[Swegle et al., 1995] studied stability criteria for SPH equations, and it was observed that a condition for unstable growth is:

$$\sum W^{''}(r,h) \cdot T > 0 \tag{3.5}$$

where $W^{''}$ is the second derivative of the kernel and the stress T is negative in compression and positive in tension.
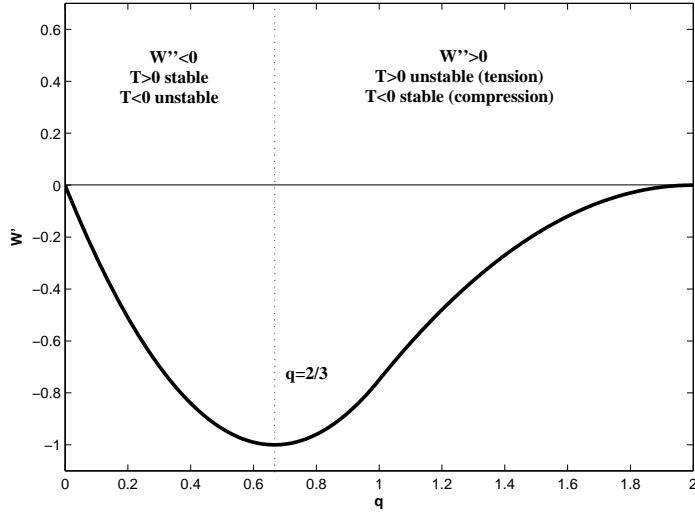
Figure 3.5: Stability regimes for the cubic spline kernel.

Figure 3.5 (see Figure 3.1 in [Swegle et al., 1995]) shows the stability regimes for the cubic spline kernel. If the second derivative is positive ($W^{''} > 0$), the method is unstable in tension ($T > 0$) and stable in compression ($T < 0$). If the second derivative is negative, it is unstable in compression and stable in tension.

The tensile instability results in a clustering of SPH particles. The clustering is clear in materials with a equation of state which can give rise to negative pressures. The clumping of the SPH particles is unphysical because it will be prevented in a real solid by the repulsive forces between the atoms. In this section it is shown how the instability in the case of fluids can be removed by using an artificial pressure, following [Monaghan, 2000].

So, this artificial pressure is added to momentum equation

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} + \Pi_{ab} + R f_{ab}^n \right) \nabla_a W_{ab} + \mathbf{g} \qquad (3.6)$$

The added term, $R f_{ab}^n$ is called tensile correction term.

Considering a kernel, the repulsive force must increase as the separation between two particles decreases. In order to remove the numerical instability, this repulsive force is written in terms of the kernel. A suitable function which increases as the separation decreases is

$$f_{ab} = \frac{W(q)}{W(\Delta p)} \qquad (3.7)$$

where Δp is the average particle spacing divided by the smoothing length h.
using $q = \Delta p \Rightarrow W(q) = W(\Delta p) \Rightarrow f_{ab} = 1$
using $q > \Delta p \Rightarrow W(q) < W(\Delta p) \Rightarrow f_{ab} \to 0$
using $q < \Delta p \Rightarrow W(q) > W(\Delta p) \Rightarrow f_{ab} >> 1$ (area of interest to be corrected)

For example, considering the cubic spline kernel, kernel derivative has a minimum for $q = 2/3$ (see figure 3.5) so the second derivative of the kernel is equal to zero for this value of q. Analyzing in detail:

$$W(r,h) = \frac{1}{3} \left(1 - \frac{3}{2} q^2 + \frac{3}{4} q^3 \right) \qquad 0 \le q \le 1 \qquad (3.8)$$

$$\frac{dF}{dq} = 0 \Rightarrow W''(r,h) = 0 \Rightarrow \frac{3}{\pi h^3} \left(-1 + \frac{6}{4} q \right) = 0 \Rightarrow q = \frac{2}{3} \Rightarrow \Delta p = \frac{1}{1.5} \quad (3.9)$$

[Monaghan, 2000] suggests assuming the values: $n = 4$ and $\Delta p = 1/1.3$.

The factor R can be determined by relating it to the pressure. So that:

$$R = R_a + R_b \qquad where \begin{cases} R_a = 0.006 \frac{P_a}{\rho_a^2} & \text{if } P_a > 0, \\ R_a = 0.6 \frac{P_a}{\rho_a^2} & \text{if } P_a < 0 \end{cases} \qquad (3.10)$$

the same for R_b.

Finally:

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} + \Pi_{ab} + c \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} \right) f_{ab}^n \right) \nabla_a W_{ab} + \mathbf{g} \qquad (3.11)$$

So the pressure term is multiplied by $P_a(1 + c f_{ab}^n)$.

Depending on the kernel choice, the second derivative of the kernel goes to zero for a value of Δp (particle spacing divided by h). These values appear in the next table:

Kernel	Values for tensile correction
Gaussian	$\sqrt{2}/2$
Quadratic	no correction is needed
Cubic Spline	2/3
Wendland	0.5

Table 3.1: Values of Δp used for tensile correction of different kernels.

3.3   Density reinitialization

While the dynamics from SPH simulations are generally realistic, the pressure field of the particles exhibits large pressure oscillations. Efforts to overcome this problem have concentrated on several approaches including correcting the kernel (see section 3.4) and developing an incompressible solver. One of the most straightforward and computationally least expensive is to perform a filter over the density of the particles and the re-assign a density to each particle ([Colagrossi and Landrini, 2003]). There are two orders of correction, zeroth order and first order.

3.3.1   Zeroth order: Shepard filter

It is clear that, if the particle at hand is close to a boundary layer or to the water surface, the kernel function W suffers the lack of particles, and a corrected kernel function $\tilde{W}$ has to be considered. So, the density of the particles can be re-initialized. The Shepard filter ([Panizzo, 2004]) is a quick and simple correction to the density field, and the following procedure is applied every M time steps (M on the order of 30 time steps).

$$\bar{\rho}_a = \sum_b \rho_b \tilde{W}_{ab} \frac{m_b}{\rho_b} = \sum_b m_b \tilde{W}_{ab} \qquad (3.12)$$

where the kernel has been corrected using a zeroth-order correction:

$$\tilde{W}_{ab} = \frac{W_{ab}}{\sum\limits_b W_{ab} \frac{m_b}{\rho_b}} \qquad (3.13)$$

3.3.2   First order: Moving Least Squares (MLS)

The Moving Least Squares (MLS) approach was developed by [Dilts, 1999] and successfully applied by [Colagrossi and Landrini, 2003] and [Panizzo, 2004]. This is a first-order correction so that the variation of a linear density field can be exactly reproduced:

$$\bar{\rho}_a = \sum_b \rho_b W_{ab}^{MLS} \frac{m_b}{\rho_b} = \sum_b m_b W_{ab}^{MLS} \qquad (3.14)$$

The corrected kernel is evaluated as follows:

$$W_{ab}^{MLS} = W_b^{MLS}(\mathbf{r}_a) = \beta(\mathbf{r}_a) \cdot (\mathbf{r}_a - \mathbf{r}_b) W_{ab} \qquad (3.15)$$

so that in 2-D

$$W_{ab}^{MLS} = \left[\beta_0\left(\mathbf{r}_a\right) + \beta_{1x}\left(\mathbf{r}_a\right)\left(x_a - x_b\right) + \beta_{1z}\left(\mathbf{r}_a\right)\left(z_a - z_b\right) \right] W_{ab} \qquad (3.16)$$

where the correction vector β is given by

$$\beta\left(\mathbf{r}_a\right) = \begin{pmatrix} \beta_0 \\ \beta_{1x} \\ \beta_{1z} \end{pmatrix} = \mathbf{A}^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad (3.17)$$

where

$$\mathbf{A} = \sum_b W_b\left(\mathbf{r}_a\right) \tilde{\mathbf{A}} \frac{m_b}{\rho_b} \qquad (3.18)$$

with the matrix $\tilde{\mathbf{A}}$ being given by

$$\tilde{\mathbf{A}} = \begin{bmatrix} 1 & (x_a - x_b) & (z_a - z_b) \\ (x_a - x_b) & (x_a - x_b)^2 & (z_a - z_b)(x_a - x_b) \\ (z_a - z_b) & (z_a - z_b)(x_a - x_b) & (z_a - z_b)^2 \end{bmatrix} \qquad (3.19)$$

Similar to the Shepard filter, this is applied every 30 time steps or similar. The equations are similar in 3-D but just include the y-direction.

Reasonable results have been obtained using the different filters. The density field for the simple case of free-surface dam-break flow is presented in Fig. 3.6 showing a comparison of using artificial viscosity (AV), and then applying the Shepard and MLS density filter. These images represent the impact of the water front against the vertical wall at the right end of the tank. Without a density filter (a), high-frequency density oscillations are observed, however more regular density distribution can be obtained using a density filter (b, c).
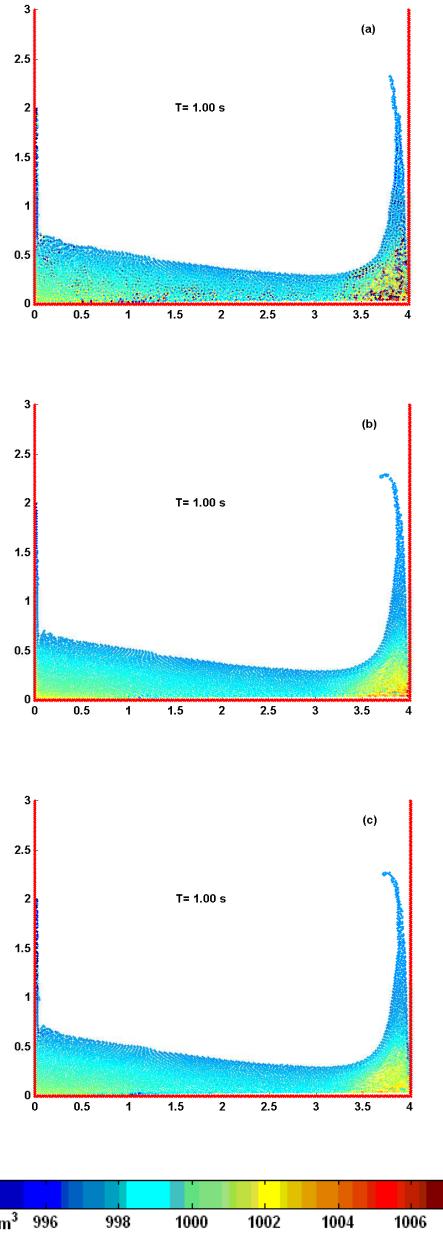
Figure 3.6: Density field for simple 2-D Dam break showing effect of density filters. (a) corresponds to AV, (b) corresponds to AV with Shepard filter and (c) corresponds to AV with MLS filter.

3.4 Kernel renormalization

A periodic correction of the kernel function W is necessary in SPH hydraulics computations, where a finite domain and a free surface are often part of the computational domain. Particles near boundaries or the free surface have a kernel smoothing function truncated due to the absence of neighboring particles. The condition of consistency of W (Eq. 2.6) fails. However it is still possible to handle these situation by opportunely correcting the kernel function W itself or its gradient. In the followings the most commonly used techniques to avoid errors from a corrupted interpolating function will be introduced in detail, making reference to the works of [Randles and Libersky, 1996], [Bonet and Lok, 1999] and [Bonet and Kulasegaram, 2000].

a) Kernel correction. The correction involves the kernel function. A kernel correction technique was proposed by [Li and Liu, 1996] and [Liu et al., 1997] with the aim to correctly interpolate polynomial functions. Another kernel correction technique, introduced by [Bonet and Lok, 1999], which involves the modification of the kernel gradient by a correction matrix, is also used.

b) Kernel Gradient correction. The correction involves the gradient of the kernel function. In order to ensure that the gradient of a velocity field is correctly evaluated, the gradient correction proposed by [Bonet and Lok, 1999] can be implemented.

c) Mixed kernel and gradient correction. A third possible correction technique is obtained by combining the constant kernel correction with the gradient correction.

3.4.1 Kernel gradient correction

The correction modifies the kernel gradient by introducing a correction matrix $\mathbf{L}$:

$$\tilde{\nabla} W_b(\mathbf{r}_a) = \mathbf{L}_a \nabla W_b(\mathbf{r}_a) \tag{3.20}$$

The velocity gradient is consequently estimated as

$$\nabla \mathbf{v}_a = \sum_{b=1}^{N} \frac{m_b}{\rho_b} \left(\mathbf{v}_b - \mathbf{v}_a \right) \otimes \tilde{\nabla} W_b(\mathbf{r}_a) = \sum_{b=1}^{N} \frac{m_b}{\rho_b} \left(\mathbf{v}_b - \mathbf{v}_a \right) \otimes \mathbf{L}_a \nabla W_b(\mathbf{r}_a) \tag{3.21}$$

The corrected kernel gradient must satisfy the condition

$$\sum_{b=1}^{N} \frac{m_b}{\rho_b} \left(\mathbf{r}_b - \mathbf{r}_a \right) \otimes \nabla W_b(\mathbf{r}_a) = \mathbf{I} \tag{3.22}$$

Thus,

$$\sum_{b=1}^{N} \frac{m_b}{\rho_b} \left(\mathbf{r}_b - \mathbf{r}_a\right) \otimes \tilde{\nabla} W_b(\mathbf{r}_a) = \left(\sum_{b=1}^{N} \frac{m_b}{\rho_b} \left(\mathbf{r}_b - \mathbf{r}_a\right) \otimes \nabla W_b(\mathbf{r}_a)\right) \mathbf{L}_a^T = \mathbf{I} \quad (3.23)$$

from which $\mathbf{L}$ is evaluated as

$$\mathbf{L}_a = \left(\sum_{b=1}^{N} \frac{m_b}{\rho_b} \nabla W_b(\mathbf{r}_a) \otimes \left(\mathbf{r}_b - \mathbf{r}_a\right)\right)^{-1} \tag{3.24}$$

If this correction technique is used, the gradient of a velocity field is correctly evaluated ([Bonet and Lok, 1999]).

Density reinitialization and kernel renormalization were not used along the applications presented in chapters 4, 5 and 6 because: (i) they are a late development, (ii) it is slightly more expensive computationally, (iii) they will not make a huge difference to the presented results because SPH gets the dynamics right

3.5   Time stepping

There are several ways to develop the solution of the SPH equations in time. It is advisable to use at least a second-order accurate scheme in time. Two numerical schemes are implemented in SPHysics: (i) the Predictor-Corrector algorithm described by [Monaghan, 1989]; (ii) the Verlet algorithm ([Verlet, 1967]). Other different time integration schemes have been employed in SPH and in other Lagrangian numerical models such as the Two Step Velocity Verlet algorithm ([Monaghan, 2006]) and the Beeman ([Beeman, 1976]) algorithms.

Consider the momentum (2.9), density (2.19), position (2.23) and density of energy (2.24) equations in the following form

$$\frac{d\mathbf{v}_a}{dt} = \mathbf{F}_a \tag{3.25}$$

$$\frac{d\rho_a}{dt} = D_a \tag{3.26}$$

$$\frac{d\mathbf{r}_a}{dt} = \mathbf{V}_a \tag{3.27}$$

$$\frac{de_a}{dt} = E_a \tag{3.28}$$

where $\mathbf{V}_a$ represents the velocity contribution from particle a and from neighboring particles (XSPH correction).

3.5.1 Predictor-corrector scheme

This scheme predicts the evolution in time as,

$$\mathbf{v}_a^{n+1/2} = \mathbf{v}_a^n + \frac{\Delta t}{2}\mathbf{F}_a^n \tag{3.29}$$

$$\rho_a^{n+1/2} = \rho_a^n + \frac{\Delta t}{2}D_a^n \tag{3.30}$$

$$\mathbf{r}_a^{n+1/2} = \mathbf{r}_a^n + \frac{\Delta t}{2}\mathbf{V}_a^n \tag{3.31}$$

$$e_a^{n+1/2} = e_a^n + \frac{\Delta t}{2}E_a^n \tag{3.32}$$

calculating $P_a^{n+1/2} = f\left(\rho_a^{n+1/2}\right)$ according to Eq. 2.20.

These values are then corrected using forces at the half step

$$\mathbf{v}_a^{n+1/2} = \mathbf{v}_a^n + \frac{\Delta t}{2}\mathbf{F}_a^{n+1/2} \tag{3.33}$$

$$\rho_a^{n+1/2} = \rho_a^n + \frac{\Delta t}{2}D_a^{n+1/2} \tag{3.34}$$

$$\mathbf{r}_a^{n+1/2} = \mathbf{r}_a^n + \frac{\Delta t}{2}\mathbf{V}_a^{n+1/2} \tag{3.35}$$

$$e_a^{n+1/2} = e_a^n + \frac{\Delta t}{2}E_a^{n+1/2} \tag{3.36}$$

Finally, the values are calculated at the end of the time step following:

$$\mathbf{v}_a^{n+1} = 2\mathbf{v}_a^{n+1/2} - \mathbf{v}_a^n \tag{3.37}$$

$$\rho_a^{n+1} = 2\rho_a^{n+1/2} - \rho_a^n \tag{3.38}$$

$$\mathbf{r}_a^{n+1} = 2\mathbf{r}_a^{n+1/2} - \mathbf{r}_a^n \tag{3.39}$$

$$e_a^{n+1} = 2e_a^{n+1/2} - e_a^n \tag{3.40}$$

Finally, the pressure is calculated from density using $P_a^{n+1} = f\left(\rho_a^{n+1}\right)$.

[Monaghan, 1989] uses this development to show that the SPH method conserves both linear and angular momenta. In practice, they used the midpoint value of the previous time step instead of computing the value at instant n, which saves time and creates only a small error. The overall scheme is second order.

3.5.2 Verlet scheme

The Verlet algorithm ([Verlet, 1967]) is probably the most commonly used time integration scheme in molecular dynamics. The basic idea is to write two third-order Taylor expansions, one forward and one backward in time.

This time stepping algorithm, to discretize Equations 3.25-3.28, is split into two parts: In general, variables are calculated according to

$$\mathbf{v}_a^{n+1} = \mathbf{v}_a^{n-1} + 2\Delta t \mathbf{F}_a^n \tag{3.41}$$

$$\rho_a^{n+1} = \rho_a^{n-1} + 2\Delta t D_a^n \tag{3.42}$$

$$\mathbf{r}_a^{n+1} = \mathbf{r}_a^n + \Delta t \mathbf{V}_a^n + 0.5\Delta t^2 \mathbf{F}_a^n \tag{3.43}$$

$$e_a^{n+1} = e_a^{n-1} + 2\Delta t E_a^n \tag{3.44}$$

Once every M time steps (M on the order of 50 time steps), variables are calculated according to

$$\mathbf{v}_a^{n+1} = \mathbf{v}_a^n + \Delta t \mathbf{F}_a^n \tag{3.45}$$

$$\rho_a^{n+1} = \rho_a^n + \Delta t D_a^n \tag{3.46}$$

$$\mathbf{r}_a^{n+1} = \mathbf{r}_a^n + \Delta t \mathbf{V}_a^n + 0.5\Delta t^2 \mathbf{F}_a^n \tag{3.47}$$

$$e_a^{n+1} = e_a^n + \Delta t E_a^n \tag{3.48}$$

This is to stop the time integration diverging since the equations are no longer coupled.

3.6 Variable time step

Time-step control is dependant on the forcing terms, the Courant-Fredrich-Levy condition and the viscous diffusion term ([Monaghan, 1989]). A variable time step Δt is calculated according to [Monaghan and Kos, 1999]:

$$\Delta t = 0.3 \cdot min\left(\Delta t_f, \Delta t_{cv}\right) \tag{3.49}$$

$$\Delta t_f = min_a \sqrt{h/|f_a|} \tag{3.50}$$

$$\Delta t_{cv} = min_a \frac{h}{c_s + max_b \left|\frac{h\mathbf{v}_{ab}\cdot\mathbf{r}_{ab}}{\mathbf{r}_{ab}^2}\right|} \tag{3.51}$$

here Δt_f is based on the force per unit mass $|f_a|$, and combines the Courant condition and the viscous time-step controls.
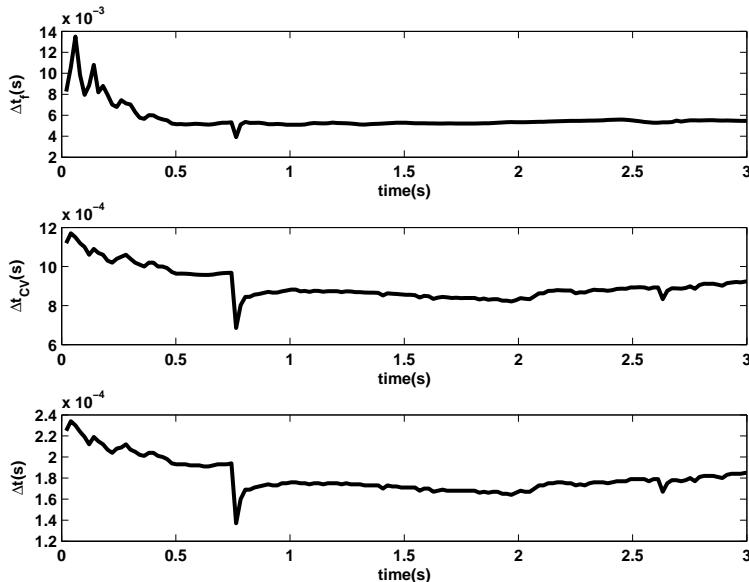
Figure 3.7: Time-step control in the collapse of a water column.

In Figure 3.7, it is observed how the new time step is controlled by the term that combines the CFL condition and the viscous diffusion term.

3.7   Computational efficiency: link list

In the code the computational domain is divided in square cells of side $2h$ (see Figure 3.8) following [Monaghan and Lattanzio, 1985]. Thus, for a particle located inside a cell, only the interactions with the particles of neighboring cells need to be considered. In this way the number of calculations per time step and, therefore, the computational time diminish considerably, from N^2 operations to $N \cdot logN$, N being the number of particles.
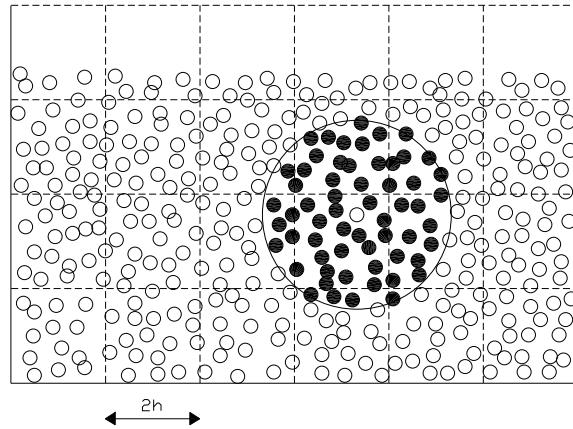
Figure 3.8: Set of neighboring particles in 2D. The possible neighbors of a fluid particle are in the adjacent cells but it only interacts with particles marked by black dots.

The SPH code in 2D sweeps through the grid along the x-direction, for each z-level. Around each cell, the E, N, NW NE neighboring cells are checked to minimize repeating the particle interactions. Thus, for example, when the center cell is $i = 5$ and $k = 3$ (see scheme in Figure 3.9), the target cells are (5,4), (4,4), (6,4) and (6,3). The rest of the cells were previously considered through the sweeping (e.g. the interaction between cell (5,3) and (5,2) was previously accounted when (5,2) was considered to be the center cell).
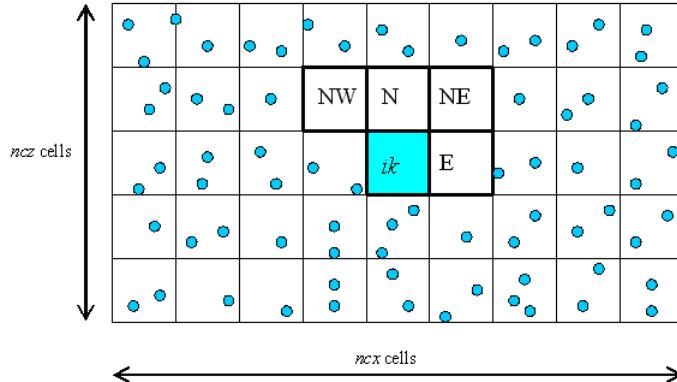
Figure 3.9: Sweeping through grid cells in 2D. Starting from the lower left corner, particles inside the center cell ik interact with adjacent cells only in E, N, NW and NE directions. The interactions with the rest of the cells W, S, SW SE directions were previously computed using reverse interactions.

A similar protocol is used in 3D calculations (Figure 3.10).
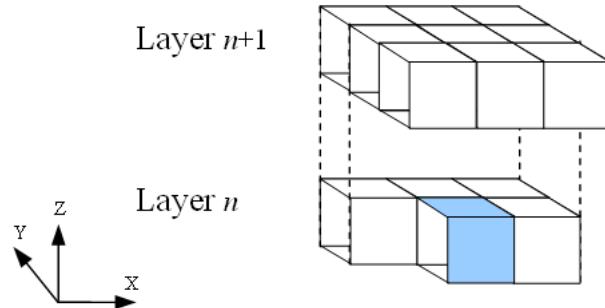
Figure 3.10: Sweeping through grid cells in 3D. Only 13 out of 26 possible neighboring cells are considered when centered on a particular ijk cell. The rest were previously considered when centered on adjacent cells using reverse interactions.

Two link lists are considered. The first one tracks the boundary particles and it is partially upgraded every time step. This is due to the fact that the only boundary particles that change their position in time are the ones that describe moving objects such as gates and wavemakers. The second link list corresponds to fluid particles and it is completely updated every time step.

3.8   Initial conditions

For all the cases considered in this work the initial velocity of the fluid particles is considered zero. The particles are assigned an initial density ρ_0 based on hydrostatic pressure. So, density of a particle a (located at depth z_a) must be calculated taking in account the water column height:

$$\rho\left(x, z\right) = \rho_0 \left(1 + \frac{\rho_0 g (H(x) - z)}{B}\right)^{1/\gamma} \tag{3.52}$$

where $H(x)$ is the initial water depth at position x and z is the vertical distance from the bottom. Pressure is calculating using equation of state (2.20) following the density value.

Fluid particles were initially placed in a fixed position. Two different initial configurations can be choice:

A) Cartesian grid:
In 2 dimensions, particles are located at nodes of a square grid, a single example can be a square grid as used in [Monaghan and Kos, 1999]. In 3 dimensions, it is corresponded to a simple cubic grid. Nodes of the grid are located at $\overrightarrow{R} = ldx\,\overrightarrow{i} + mdy\,\overrightarrow{j} + ndz\,\overrightarrow{k}$, where l,m and n are integers, $\overrightarrow{i}$, $\overrightarrow{j}$ and $\overrightarrow{k}$ are unitary vectors in X, Y and Z directions and dx, dy, dz are the spacing between particles in each direction. This configuration can be seen in Figure 3.11. The dimensions of the box are $1m$ x $1m$ and $dx = dz = 0.03m$, the number of particles is 937, 201 of them are boundaries.
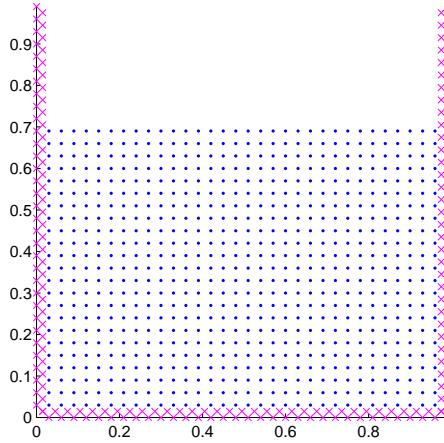
Figure 3.11: Sketch of the initial positions for a Cartesian grid. Crosses represent boundary particles and points represent fluid particles.

B) Staggered grid ([Monaghan and Lattanzio, 1991]):
In 2 dimensions, particles are located at nodes of a square grid with a node in the center of the square. In 3 dimensions particles are located at nodes of body centerd cubic grid (BCC). This is commonly used because of a bigger amount of neighboring particles would be generated for each particle using this kind of grid. Thus, nodes of the grid are located at $\overrightarrow{R} = ldx\,\overrightarrow{i} + mdy\,\overrightarrow{j} + ndz\,\overrightarrow{k}$ with a two- point basis (0,0,0) and (dx/2, dy/2, dz/2) referred to the corner defined by $\overrightarrow{R}$. l,m and n are integers and $\overrightarrow{i}$, $\overrightarrow{j}$ and $\overrightarrow{k}$ are unitary vectors in X, Y and Z directions. The obtained configuration can be seen in Figure 3.12. In this case, the number of particles is 1619 (201 are boundaries).
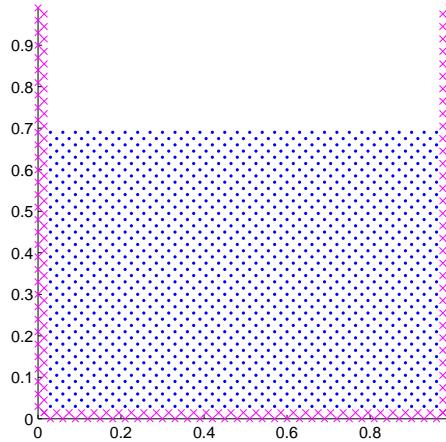
Figure 3.12: Sketch of the initial positions for a staggered grid. Crosses represent boundary particles and points represent fluid particles.

3.9   Boundary conditions

The boundary conditions do not appear in a natural way in the SPH formalism. When a particle approaches a solid boundary, in the summations (Eq. 2.2) only the particles located inside the system intervene without any interaction from the outside. This contribution can generate unrealistic effects, due to the different nature of the variables to solve, since some, like the velocity, fall to zero when they approach the boundaries, while others, such as the density, not. The different solutions to avoid boundary problems consist on the creation of several virtual particles that characterize the system limits. Basically, three different types of particles can be distinguished:

Ghost particles. [Libersky and Petscheck, 1991];[Randles and Libersky, 1996] considered boundary particles whose properties, included their position, vary each time step. When a real particle is close to a boundary (at a distance shorter than the kernel smoothing length) then a virtual (ghost) particle is generated outside of the system, constituting the specular image of the incident one. Both particles have the same density and pressure, but opposite velocity. Thus, the number of boundary particles varies in each time step, which complicates its implementation in the code. This method was also used by [Colagrossi and Landrini, 2003].

Repulsive particles. These type of boundaries are due to [Monaghan, 1994]. In this case the particles that constitute the frontier exert central forces on the fluid particles, in analogy with the forces among molecules. Thus, for a boundary particle and a fluid particle separated a distance r the force for unit of mass has the form given by the Lennard-Jones potential. In a similar way, other authors ([Peskin, 1977]) express this force assuming the existence of forces in the boundaries, which can be described by a delta function. This method was refined in [Monaghan and Kos, 1999] by means of an interpolation process, minimizing the inter-spacing effect of the boundary particles on the repulsion force of the wall. A new revision was done in [Monaghan et al., 2003].

Dynamic particles. These particles verify the same equations of continuity and of state as the fluid particles, but their position remains unchanged or is externally imposed. An interesting advantage of these particles is their computational simplicity, since they can be calculated inside the same loops as fluid particles with a considerable saving of computational time. These particles were first presented in [Dalrymple and Knio, 2001]) and used in further studies on the interaction between waves and coastal structures ([Gómez-Gesteira et al., 2005]; [Gómez-Gesteira and Dalrymple, 2004]; [Crespo et al., 2007a]).

Three boundary conditions have been implemented in SPHysics: (i) Periodic open boundary conditions; (ii) Repulsive boundary conditions and (iii) Dynamic Boundary conditions:

3.9.1 Dynamic boundary conditions

The aim of this section is the study of the role of the so called Dynamic Boundary Particles (DBPs from now on). These boundary conditions are analyzing in detail because they are the most common used ones in SPHysics applications (following chapters). These particles share the same properties as the fluid particles. They follow the same equations of state, continuity and the energy equation. However, they are not allowed to move and they remain fixed in position (fixed boundaries) or move according to some externally imposed function (moving objects like gates, wavemakers $\cdots$).

Repulsion mechanism of DBPs

The boundaries exert a force to the fluid particles when approaching. In order to analyze the fluid particles movement due to boundary particles, a schematic system composed by two particles, a boundary particle and a fluid one, was considered. The equation of state can be obtained from the first term of the Taylor expansion of Eq. 2.20, assuming that the speed of sound, c, is constant.

$$P_a = c^2(\rho_a - \rho_{a0}) \qquad P_b = c^2(\rho_b - \rho_{b0}) \tag{3.53}$$

with a the moving particle and b the boundary one.

Considering the radial coordinate joining the center of both particles to coincide with an axis (Z), the equation of motion for the particle a (the fluid one) becomes

$$\frac{dw_a}{dt} = -m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} \right) \frac{\partial}{\partial z_a} W_{ab} \tag{3.54}$$

in absence of viscosity ($\Pi_{ab} = 0$) and gravity, being w_a the velocity of the particle a in Z- direction.

Using Eq. 3.53 we obtain

$$\frac{dw_a}{dt} = -m_b c^2 \left(\frac{(\rho_b - \rho_0)}{\rho_b^2} + \frac{(\rho_a - \rho_0)}{\rho_a^2} \right) \frac{\partial}{\partial z_a} W_{ab} \tag{3.55}$$

The continuity equation can be written following [Monaghan, 1996]

$$\rho_a = \sum_b m_b W_{ab} \tag{3.56}$$

$$\rho_a = m_b W_{ab} + m_a W_{aa} \qquad \rho_b = m_a W_{ab} + m_b W_{bb} \tag{3.57}$$

for the two particles under scope.

Being $W_0 = W_{aa} = W_{bb} = W_{r_{ab}=0}$. Assuming the same mass for both particles ($m_a = m_b = m$), the densities are calculated following Eq. (3.57).

$$\rho_a = \rho_b = \rho = m\left(W_{ab} + W_o\right) \qquad \rho_0 = mW_0 \qquad (3.58)$$

Thus Eq. 3.55 becomes

$$\frac{dw_a}{dt} = -2c^2 \frac{W_{ab}}{\left(W_{ab} + W_0\right)^2} \frac{\partial}{\partial z_a} W_{ab} \qquad (3.59)$$

Considering the particular case of a Gaussian kernel

$$W_{ab} = \frac{2\pi}{h} e^{-z_{ab}^2/h^2} \qquad \nabla W_{ab} = -\frac{2z_{ab}}{h^2} W_{ab} \qquad W_0 = \frac{2\pi}{h} \qquad (3.60)$$

Eq. 3.59 becomes

$$\frac{dw_a}{dt} = \frac{4c^2}{h^2} z_{ab} \left(\frac{1}{1 + e^{z_{ab}^2/h^2}}\right)^2 \qquad (3.61)$$

Thus, the direction of the force exerted on particle a by a boundary particle b, depends on the sign of z_{ab}. When a approaches b from the above (below) z_{ab} becomes positive (negative) and, consequently, particle a is pushed up (down). Note that force tends to zero when z_{ab} tends to zero. This result is usually attained when using kernels whose first derivative goes to zero at z_{ab}. This effect can be prevented in numerical simulations using the tensile correction proposed by Monaghan (2000).

In general, the forces exerted on the moving particle can be summarized as

$$\frac{dw_a}{dt} = -\left(2c^2 \frac{W_{ab}}{\left(W_{ab} + W_0\right)^2} + m\Pi_{ab}\right) \frac{\partial}{\partial z_a} W_{ab} - g \qquad (3.62)$$

where the viscosity (Π_{ab}) and gravity (g) terms have been added. Note that Eq. 3.62 does not depend on a particular kernel definition. Actually, a Gaussian kernel was considered in Eq. 3.61 for mathematical simplicity, although a cubic spline kernel will be considered in further numerical simulations.

Test problem 1: Particle movement inside a box

A simple test corresponding to the movement of a single particle inside a box was considered to depict the main features of the interaction between moving and boundary particles. In spite of the schematic nature of the test, it proves that the particle can be kept inside the box due to the repulsive force without losses in the mechanical energy of the system. Artificial viscosity is considered (Eq. 2.10).

Different tests were carried out with numerical model to study the evolution of a single particle inside a box $(0.5 \times 0.5m)$. The boundary particles were placed in two rows forming a staggered grid as shown in Figure 3.13.
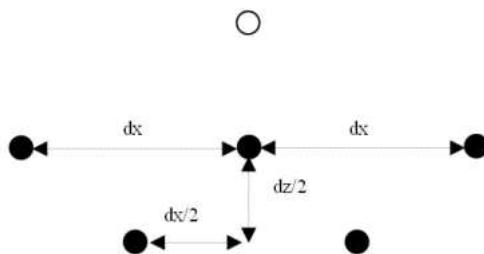
Figure 3.13: Sketch of the interaction between a fluid particle (empty circle) and a set of boundary particles (full circles). The boundary particles are placed in a staggered manner.

The separation between the boundary particles is $dx = dz = h/1.3$ and $h = 2.097 \cdot 10^{-2} m$. In the Z axes the distance will be measured from the boundary particles. The first experiment was the fall of a particle from $(X_0, Z_0) = (0.25, 0.3)$ m without initial velocity and zero viscosity ($\alpha = 0$ in Eq. 2.12). The particle was initially far from the boundaries, in such a way that gravity was the only initial force on the particle. This particle does not feel the interaction of the boundary particles until it approaches the bottom of the box. It is important to note that the boundary particle is situated exactly at the same X position as the moving particle, but at $Z = 0$.

Figure 3.14 shows the repulsion mechanism. The incoming particle, a, increases the density locally (Fig. 3.14a) according to Eq. 2.19, which results in an increase in pressure following Eq. 2.20 (Fig. 3.14b) and in an increase of the pressure term (P/ρ^2) in Eq. 2.10. The normalized pressure term, $NPT_Z = \left(P/\rho^2\right)_Z / \left(P/\rho^2\right)_R$, is represented in Fig. 3.14c, where Z refers to the distance from the incoming particle to the wall and R to the return point of the incoming particle. Note how the fluid particle suffers the effect of the boundary when the distance particle boundary is shorter than $2h$.
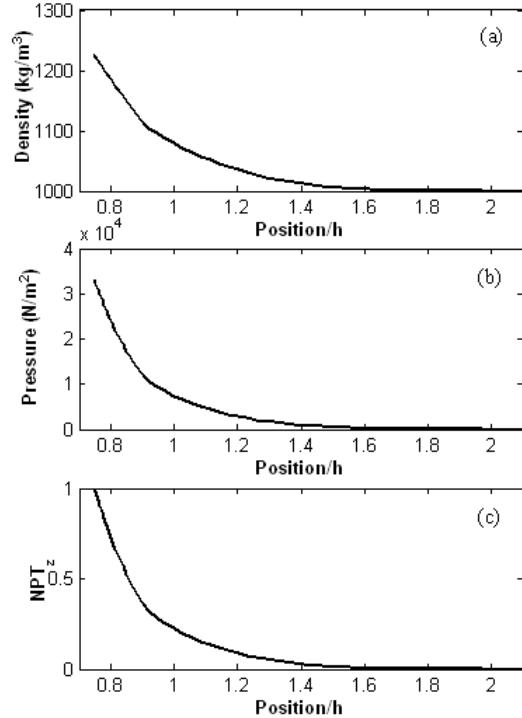
Figure 3.14: Variation of density (a), pressure (b) and normalized pressure term (c) for a moving particle approaching a solid boundary. Calculations were run without viscosity.

Figure 3.15 represents the movement of the particle using the SPH method (circles) in good agreement with the theoretical results (line) obtained from Eq. 3.62. The position and velocity are observed to be periodic. The particle trajectory in phase space follows a cycle. The collision is observed to be elastic. During most of the time, from 0 to A and from B to 0, the particle is under gravitational forces. Only from A to B the particle is under the force exerted from the boundary, verifying $V_z(2h^-) = -V_z(2h^+)$, where the superscripts $-$ and $+$ refer to before and after the collision. Thus, the particle apparently conserves the mechanical energy, showing a closed trajectory and bouncing in a elastic manner.
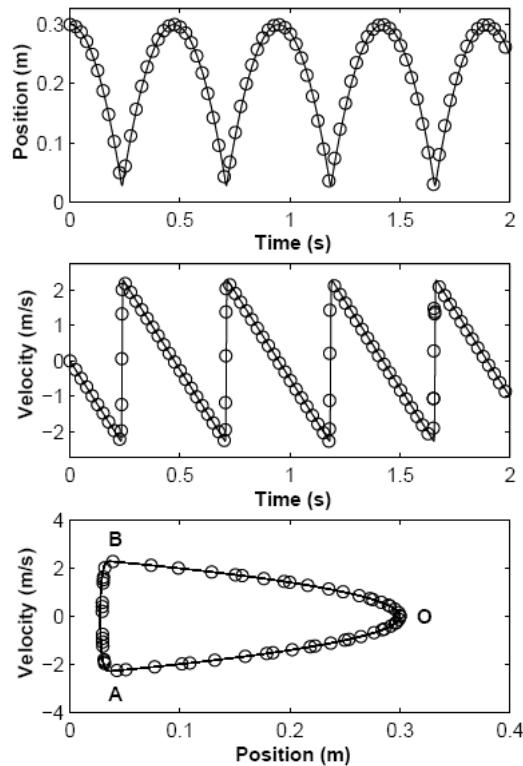
Figure 3.15: Single particle collision with a boundary in the absence of viscosity ($\alpha = 0$).

One of the main advantages of DBPs is the fact that boundary particles are considered a part of the system, in such a way that their energy can be calculated every time step to check energy evolution. To analyze energy conservation during the calculation the thermal energy associated to each particle is calculated using the expression given by [Monaghan, 1994] (Eq. 2.24).

The energy corresponding to the first collision between the moving particle and the boundary (Figure 3.15) is depicted in Figure 3.16. The energy evolution of fluid particles is represented in the left panel. Potential Energy (green line) decreases continuously until particle bounces. Kinetic Energy (red line) increases from the beginning of the experiment and decreases sharply when the particle approaches to the tank bottom to increase in the same way after collision with the boundary. This rapid process corresponds to the inversion of velocity observed at the moment of the collision ($t = 0.24 - 0.25s$). The thermal energy of the fluid particle (blue line) increases at the moment of the collision although does not balance the decrease in kinetic energy. The remaining energy corresponds to changes in the thermal energy of the boundary. The total energy of fluid and boundary particles is represented in the right panel. First of all, the potential energy of the boundary particles was set to zero for the sake of clarity. Note that the potential energy of boundaries remains unchanged during the calculation and it can be considered as an offset. The energy of the moving particle (red line) decreases during the collision in an amount similar to the boundary energy increase (green line). This boundary energy is totally thermal since the boundary particles remain unchanged during the calculation. Instantaneous changes in the total energy (blue line), with a maximum increase around 1.5% of the total energy, are only observed during the collision. However, changes are balanced in such a way that the total energy of the system is exactly the same before and after the collision ($E = 0.9196J$).
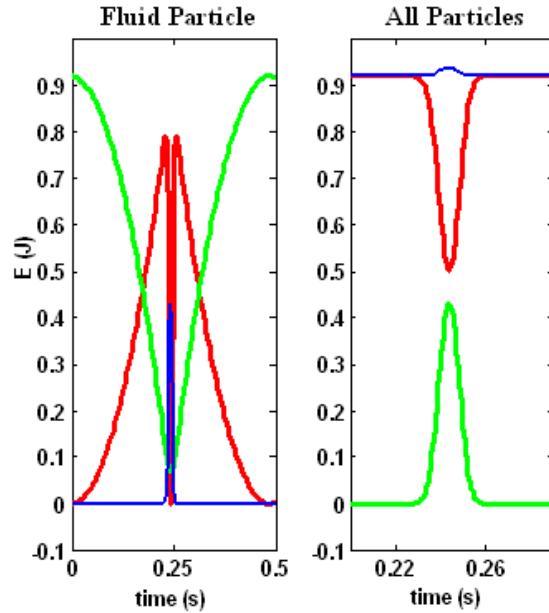
Figure 3.16: Energy of the fluid particle and all particles.

To analyze the role of the viscosity on the fluid-boundary collision the same test depicted in Fig. 3.15 was carried out with $\alpha = 0.05$ (Fig. 3.17). The line represents the theoretical prediction given by Eq. 3.62 and the circles represent the numerical results. Fig. 3.17 shows how the maximum height reached after each collision, decreases in time. This decrease can also be observed for velocity. The phase diagram shows an open trajectory due to the loss of mechanical energy of the fluid particle when approaching the boundary. Actually, one can observe this in a single collision $|V_z(2h^-)| > |V_z(2h^+)|$.
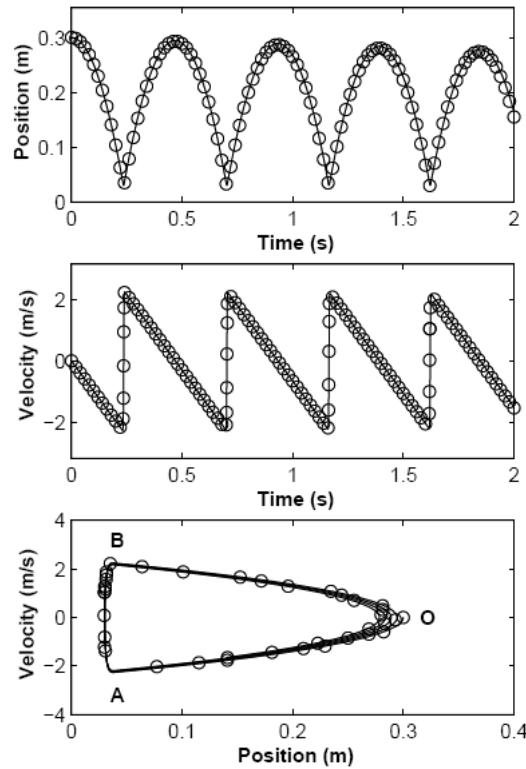
Figure 3.17: Single particle collision with a boundary in a viscous medium ($\alpha = 0.05$).

DBPs do not prevent wall penetration, which can be attained when the fluid particle approaches the boundary fast enough. The return point can be defined as the minimum distance from the incoming particle to the boundary divided by the smoothing length (Z_{min}/h). The following numerical experiment was considered to analyze the dependence of the return point on the incident particle velocity. Once again, an inviscid medium ($\alpha = 0.0$) was considered. In addition, gravity was turn off in the model to assure a constant velocity, which was initially imposed to the incoming particle. Figure 3.18 shows the return point decrease when increasing the velocity of the incoming particle.
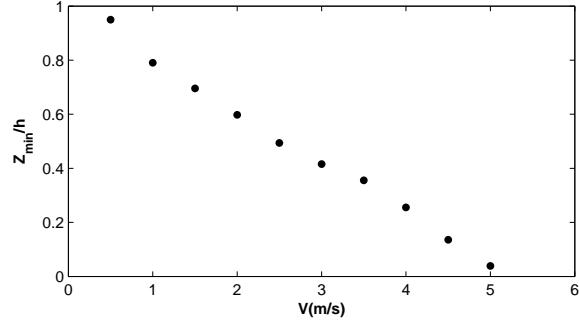
Figure 3.18: Return point for different incoming velocities of the moving particle.

The inter-spacing between boundary particles can affect the repulsion force exerted by the wall. Actually, the repulsive particle method was refined in [Monaghan and Kos, 1999], by means of an interpolation process to minimize this effect. The dependence of the return point on interspacing should be checked in the DBPs method since there is not a specific mechanism to interpolate the exerted force. The calculation parameters previously described for Figure 3.18 were used in this case with an incoming velocity $v_z = 0.5ms^{-1}$. The base configuration corresponds to the one shown in Figure 3.13, where the falling particle has the same X coordinate as the boundary particle. This configuration can be changed in X direction ($\Delta x \in [-dx/2, dx/2]$). The normalized return point was calculated using $Z_n = z/z_0 * 100$, where z is the return point for a certain Δx and z_0 the return point for $\Delta x = 0$ (Figure 3.19). The return point is observed to decrease with $|\Delta x|$ although variations from the case $\Delta x = 0$ are always lower than 0.1%. Obviously, the behavior is symmetric and only depends on $|\Delta x|$, not on the sign of the displacement from the base configuration.
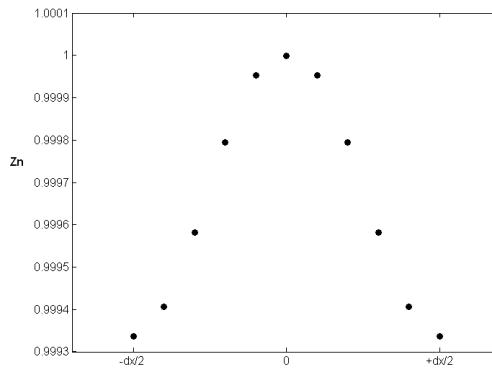
Figure 3.19: Dependence of the return point on the fall position.

Test problem 2: Collapse of a water column

Once the main properties of DBPs have been described in the previous over-simplified test case, DBPs will be used in a more realistic test. It consists in the collapse due to the gravity of a $2m$ high 2D water column in a tank. A complete description of the experiment is given by [Koshizuka and Oka, 1996] and a brief setup can observed in Figure 3.20. The same setup was used by [Violeau and Issa, 2006] to check the accuracy of their SPH code. The tank is $4m$ long, the initial volume of water is $1m$ long and $2m$ high. The number of boundary particles is 4,000 and the number of fluid particles is 40,000. A smoothing length, $h = 0.012m$ and a viscosity term, $\alpha = 0.5$, were considered.
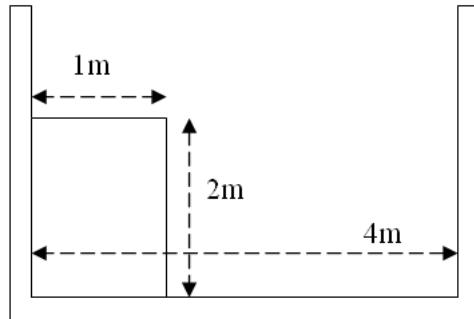
Figure 3.20: Initial configuration of the water column and the tank experiment.

This laboratory test case will allow checking different properties of DBPs, namely, the fluid movement parallel to the left wall and bottom and the fluid collision against the right wall.

On the one hand, the movement of the fluid inside the box depends on the interaction between the fluid and the boundary apart from the geometrical constraints of the initial water parcel. Thus, a proper boundary treatment will generate a realistic water height decrease near the left wall and an accurate water velocity near the dam toe. On the other hand, the boundaries must prevent fluid escape through the right wall, which suffers the most energetic water collision in the experiment.
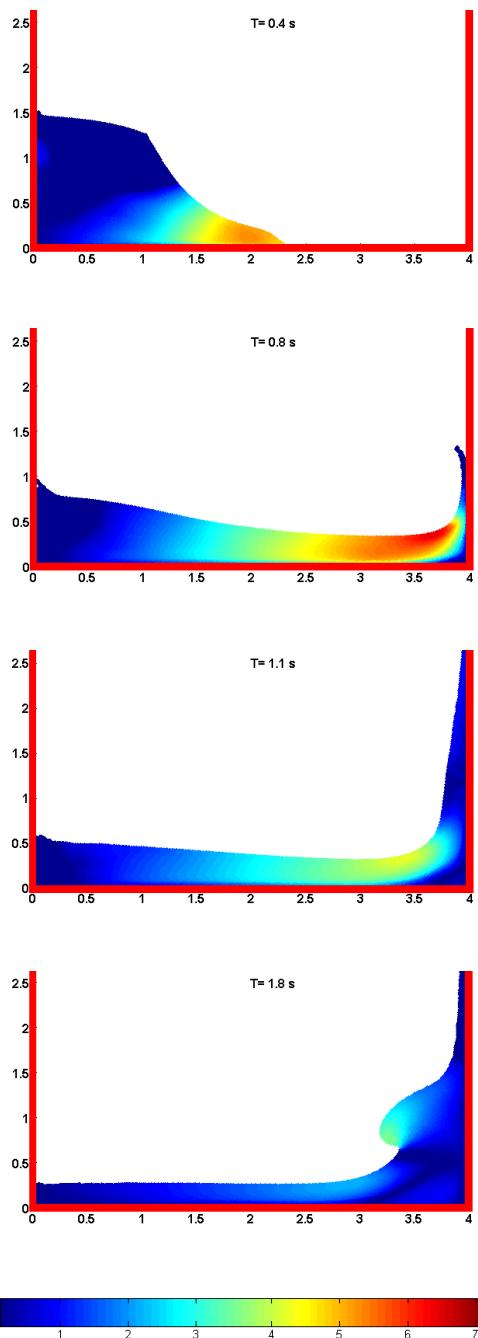
Figure 3.21: Collapse of a water column in a tank simulated with SPH model plotting the particle velocities.

In the Figure 3.21, velocity magnitude ($v = \sqrt{v_x^2 + v_z^2}$) is depicted at different instants of dam evolution. The colorbar is common to all snapshots. Distances are in meters and velocities in meters per second. Each particle is represented by a color corresponding to its instantaneous velocity. At $T = 0.4s$ the maximum dam break velocities are observed near the toe. The toe velocity evolution will be compared with experimental data in next figure. At $T = 0.8s$ the wave front has collided with the right wall. In $T = 1.1s$ water climbed onto the right wall. At $T = 1.8s$ water starts to fall over. The water height decrease near the left wall, which can be observed during the whole dam break, will also be compared to experimental data in Fig. 3.22.
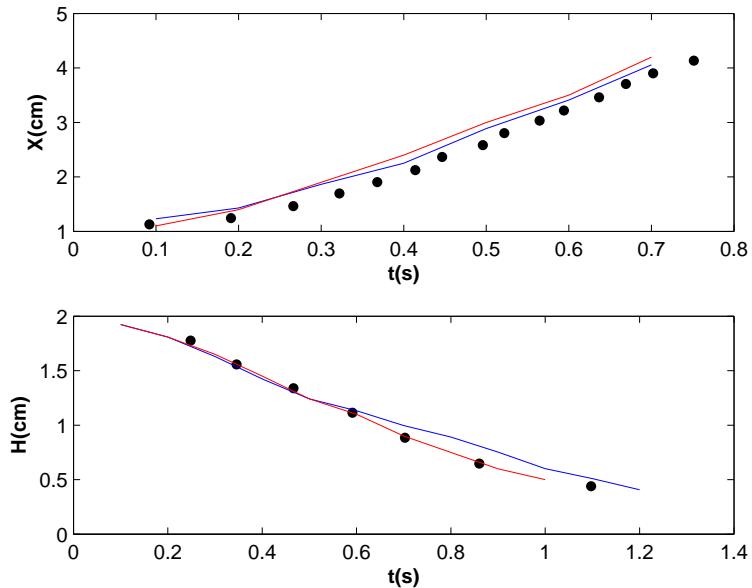
Figure 3.22: Collapse of a water column in a tank simulated with SPH model (blue solid line) comparing with experimental data (circles) and [Violeau and Issa, 2006] results (red solid line).

An accurate water height (H) decreases near the left wall and dam toe advance (X) proves the proper behavior of boundary conditions. Fig. 3.22 shows how H and X fit data provided by [Koshizuka and Oka, 1996] experiment in an accurate way. Comparing both SPH results, water height H calculated by [Violeau and Issa, 2006] fits slightly better the experimental data, while our SPH results about the dam toe advance X are closer to experimental data than [Violeau and Issa, 2006] results are. Small diferences can be due to the different viscosity treatment used in both methods since they used a $\kappa - \epsilon$ model.

Summary of DBPs

Dynamic boundary particles (DBPs) have been considered to study the movement of fluid particles inside a container in the framework of SPHysics method. These boundaries are constituted by fixed particles placed in a staggered grid manner and follow the same equations of state and continuity as the fluid particles. From the computational point of view, the treatment of the system is considerably simplified, since no special considerations are necessary for the boundary particles. In the looping over the particles they are simply marked with an index.

The validity of the method has been checked in an oversimplified geometry where a single particle impinges a boundary. The moving particle is observed to bounce due to the local increase of pressure terms in momentum equation. Thus, the boundaries retain the main features of the physical process: (a) they only exert a normal force on the fluid particles when approaching at a certain distance ($r < 2h$); (b) the exerted force is almost independent of the particular position where the incident particle collides with the boundary; (c) the mechanical energy of the incident particle is conserved in absence of viscosity ($\alpha = 0$).

The validity of the approach has also been checked in a dam break experiment. There, DBPs prevent fluid to leave the container and guarantee a proper water movement close to the walls.

Finally, DBPs can also be applied to mimic obstacles inside the computational domain and solid boundaries whose movement is externally imposed. DBPs have been used to generate wave mitigating dikes ([Crespo et al., 2007c]), sliding doors ([Crespo et al., 2007a]) and wavemakers ([Crespo et al., 2008])).

3.10 Checking limits

In SPH, fluid particles can leave the computational domain in different ways, both physically and non-physically (as described later). Once the particle is outside the domain, it is continuously accelerated under the effect of gravity. These particles must be identified and removed from the run to avoid spurious effects. The treatment of these particles is called *checking limits* in SPHysics.

3.10.1 Fixing the limits

Limits of the computational domain are fixed at the beginning of the run depending on the initial position of the particles. In each direction: $\Lambda_k^{max} = max(\Lambda_k(i, t = 0)) + h$ and $\Lambda_k^{min} = min(\Lambda_k(i, t = 0)) - h$, where Λ_k refers to the direction (X, Y or Z) and $i \in [1, N]$ refers to all particles. These limits fix the number of cells of dimensions $2h$ x $2h$ x $2h$ (in 3D) used to cover the computational domain. Limits in X, Y and Z^- directions remain unchanged during the run. The limit in Z^+ is allowed to vary in time, since fluid can splash and surpass the initial upper limit of the container. All limits are checked at every time step.

3.10.2 Changing the limits in Z^+

When a fluid particle surpasses the upper limit in the vertical Z direction, the computational domain is extended and new cells are created (see Fig. 3.23). The number of boundary particles inside these new cells is immediately set to zero. Fluid particles can then occupy these cells depending on their position. The number of cells in the vertical is thus dynamically modified depending on the position of the highest fluid particle. Furthermore, the number of boxes decreases when the particles fall down and it will be equal to the initial amount (last frame of Fig. 3.23). This generates savings in execution time, since calculations in empty cells are prevented.
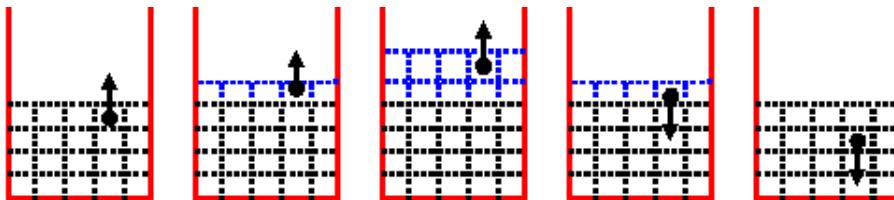
Figure 3.23: Evolution of new cells in Z direction depending on the fluid particles movement.

3.10.3   Limits in X, Y or Z^- directions

A fluid particle can surpass the initial limits in X, Y or Z^- direction due to several reasons. Dynamic boundary particles are not completely impermeable. Hence a single particle, accelerated by collision in the proximity of a boundary, can possibly penetrate the boundary. On the other hand, the fluid can collide with the container overtopping the lateral walls. Once the fluid leaves the container, fluid particles are continuously accelerated by gravity away from the domain of interest, giving rise to very small time steps according to Eq. 3.49 and slowing down the calculations.

The position of particles is checked every time step, in such a way that when a particle is found outside the container, the particle is replaced at a previously defined position outside the container and marked with a *flag*. Thus, although the particle is not eliminated from the list (the number of particles, N, remains constant) the particle is not allowed to move with time.

Chapter 4

Green Water Overtopping

Wave overtopping on the decks of offshore platforms and ships can cause severe damage risks due to the high forces generated by the wave. This phenomenon is analyzed in the framework of the Smoothed Particle Hydrodynamics (SPH) method. The presence of a horizontal deck fixed above the free surface modifies strongly the wave kinematics. In particular, the flow is split into two, showing a different behavior above and below the deck. Numerical results generated by the SPH method are compared to laboratory experiments.

Figure 4.1: Damage in an oil platform (http://discardedlies.com/).

The impacts between a wave and an oil platform can cause important structure damages (Figure 4.1).

4.1 Introduction

Wave overtopping, particularly green water overtopping (unbroken waves passing over the deck), is a major cause of damage to ships and can result in ship loss. This overtopping, particularly that due to freak or rogue waves, leads to significant unexpected forces on the superstructure of the ship or offshore platform (Buchner, 1996a, 1996b). In Europe, a significant amount of effort has gone into determining the susceptibility of Floating Production, Storage and Offloading/Floating Storage Units (FPSO/FSU) to greenwater incidents ([Buchner and Cozijn, 1997]; [Buchner, 1996a]; [Buchner, 1996b] and [Buchner and van Ballegoyen, 1997a]; [Buchner and van Ballegoyen, 1997b]; [Buchner and van Ballegoyen, 1997c]; [Buchner and van Ballegoyen, 1997d]; [Wang et al., 1998]; [Faltinsen and Greco, 2001]; [Health and Executive, 2001]; [Baarholm, 2001]; [Greco, 2001]).

The modeling of greenwater overtopping is difficult in that the fluid has to flow over an object. There are some models available, for example, UNDA, which is a nonlinear potential flow model that was used by [Trulsen et al., 2002] to examine the greenwater flow over a 3D floating object. A disadvantage of UNDA is the lack of wave breaking and vorticity generation.

The aim of this section is to examine the impact of a single wave on a flat horizontal deck to study the overtopping. [Cox and Ortega, 2002], in a laboratory study, noticed that both the velocities on the top of the deck and those under the deck are not the same as the orbital motions that would have been present in the absence of the deck. The SPH model captures most of the features of an overtopping event not only qualitatively but quantitatively when compared to laboratory experiments ([Cox and Ortega, 2002]). In addition, the model permits the study of other phenomena like the formation of a jet close to the rear of the deck.

4.2 The experiment

Here we use laboratory experiments by [Cox and Ortega, 2002] to motivate an SPH study of waves overtopping a flat deck, representing the fixed deck of an offshore structure. To simplify the overtopping process and measurement techniques, the experiment was conducted in a narrow wave flume at Texas AM University, restricting the study to two dimensions. The experimental setup is shown schematically in Figure 4.2, where x is the horizontal coordinate positive in the direction of wave propagation with $x = 0m$ at the wavemaker, and z is the vertical coordinate positive upward. The glass-walled flume was $36m$ long by $0.95m$ wide by $1.5m$ high, and was equipped with a programmable wavemaker. The model deck consisted of a fixed Plexiglas plate rigidly mounted to a steel frame and suspended from the top of the flume horizontally. The plate was $93.5cm$ wide by $61cm$ long by $1.15cm$ thick. The plate bottom was placed

5.25*cm* above the surface water and 8*m* away from the wavemaker. The water depth in the flume was 65.0*cm*. The flume was sufficiently long to prevent interactions resulting from reflection from the far wall.
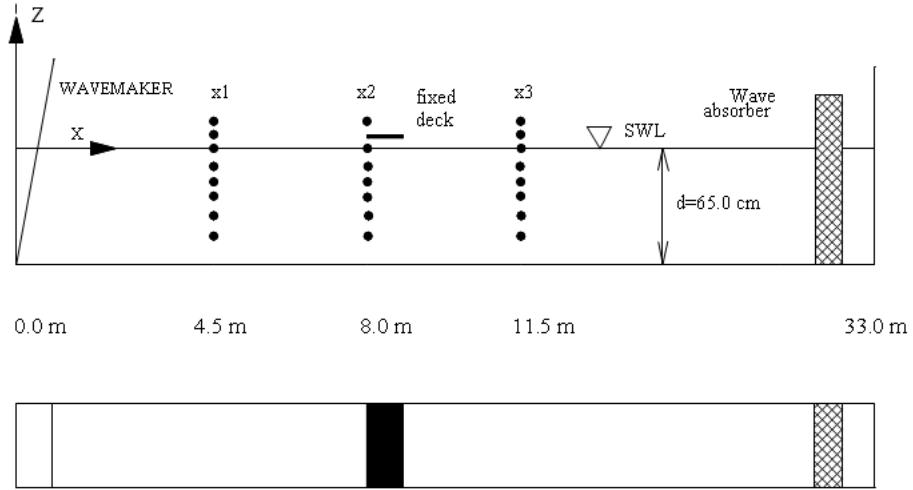
Figure 4.2: Elevation and plan view of experimental setup of Cox and Ortega.

The wavemaker was driven by a drive signal of two cycles of a $T = 1.0s$ sinusoidal wave followed by two and half cycles of a $T = 1.5s$ sinusoidal wave with larger amplitude. This signal was designed to allow the preceding waves to pass under the deck without reflection that would have interfered with the overtopping wave.

The free surface elevation was measured using a surface- piercing wave gage, beginning at $x = 4.5m$ from the wavemaker and continuing in increments of $0.5m$ to $x = 11.5m$. Velocities were measured using a laser- Doppler velocimeter (LDV) at a transect coinciding with the leading edge of the deck (8*m* from the wavemaker at rest). At this transect, the velocities were measured at 20 elevations ranging from $0.51m$ to $0.10m$.

4.3   Model calibration

The conditions used to simulate the interaction between a single wave and
a deck to reproduce the experiments described in section 4.2 are summarized in
next subsections.

4.3.1   Model inputs

Smoothing length

A smoothing length, $h_F = 4.55cm$, is associated to the interaction between
fluid particles and a smaller smoothing length, $h_B = 0.90cm$, to the interac-
tion between boundary particles. The interaction between fluid and boundary
particles is calculated using an average length, $h_{BF} = 0.5(h_F + h_B)$.

Initial conditions

Fluid particles were initially placed in a staggered grid with particle spacing
$dx = dz = 3.50$ cm and zero initial velocity as shown in Figure 4.3.
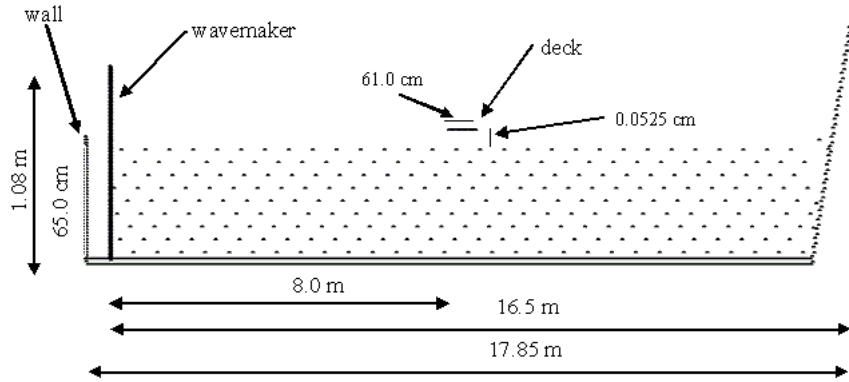
Figure 4.3: Initial SPH configuration of fluid and boundary particles to mimic
the experimental setup shown in Fig. 4.2.

The initial conditions were designed to fit the experimental conditions. The
computational system consists of a wavemaker at one end of the tank and a
sloping wall at the other end. The computational tank was $17.3m$ long and
$1.08m$ high. Despite the computational tank being shorter than the experimen-
tal one for computational saving reasons, no wave reflection was observed in the
region of interest, close to the deck. The computational deck ($61cm$ long), was
placed at the same position as the experimental one relative to the wavemaker
($x = 8m$) and $5.25cm$ above the free surface at rest. Using this initial config-
uration, the total number of particles in the numerical experiment was 18,387
(including 1,301 boundaries particles).

Boundaries

Due to the particular geometry of the numerical experiment three different boundary conditions were considered: fixed particles, deck particles and wavemaker particles. All of them are treated as DBPs ([Dalrymple and Knio, 2001]; [Crespo et al., 2007b]), since they follow the continuity equation and the equation of state, but they do not follow the momentum equation, in such a way that their position and velocity are externally fixed.

Fixed Particles (including bottom, sloping wall and fixed walls) are placed in two rows forming a staggered grid with $dx = dz = 3.50cm$ and zero initial velocity. Their position remained unchanged during the numerical experiment ($\mathbf{V}^{fixed}(t) = 0$ and $\mathbf{r}^{fixed}(t) = \mathbf{r}^{fixed}(0)$).

Deck Particles are initially placed in a single row with $dx = 3.50cm$. Their position remained unchanged during the numerical experiment.

It is a well known fact that boundary conditions are an artificial tool to mimic the effect of real boundaries and while they reproduce a physical behavior similar to the real ones, they can generate spurious effects. That is the reason why most models tend to place the boundaries as far as possible from the region of interest. In particular, the presence of a deck inside the computational domain can give rise to anomalously high density gradients near the deck. To minimize this effect, the calculated density of deck particles was filtered, in such a way that strong deviations from the reference density (ρ_0) are smoothed out using the expression:

$$\rho_a = \rho_{calculated} + \frac{dt}{T_{relaxation}} \left(\rho_{imposed} - \rho_{calculated} \right) \qquad (4.1)$$

where $\rho_{calculated}$ was calculated using the continuity equation, $\rho_{imposed}$ is the reference density (ρ_0), being dt the time step and $T_{relaxation} = dt/0.3$, the time scale in which $\rho_{calculated}$ tends to the imposed value $\rho_{imposed}$. The pressure is recalculated using the new density and the equation of state (2.20).

Wavemaker Particles are initially placed in two parallel rows with a spacing of $dx = 1.75cm$ and $dz = 1.17cm$. Velocities and positions are externally imposed to reproduce the movement of the experimental wavemaker described in [Cox and Ortega, 2002]. As described above, the drive signal of Cox and Ortega is composed by several cycles with different frequencies and amplitudes that can give rise to the appearance of instabilities in the transition between different cycles. In particular, sharp changes in velocity can generate very high instantaneous accelerations and forces. A smoothing function was considered to prevent the occurrence of these forces. Thus, if the wavemaker moves with amplitude A_i and frequency f_i in the interval $t \in [t_i, t_{i+1}]$, and with amplitude A_{i+1} and frequency f_{i+1} in the interval $t \in [t_{i+1}, t_{i+2}]$ then, for any t between $(t_i + t_{i+1})/2$ and $(t_{i+1} + t_{i+2})/2$, the piston- like movement of the wavemaker in x direction is determined by the function

$$x_p(t) = smf_1(t)A_i sin\left(f_i\left(t - t_i\right)\right) + smf_2(t)A_{i+1} sin\left(f_{i+1}\left(t - t_{i+1}\right)\right) \quad (4.2)$$

$$v_p(t) = smf_1(t)A_i f_i cos\left(f_i\left(t - t_i\right)\right) + smf_2(t)A_{i+1}f_{i+1} cos\left(f_{i+1}\left(t - t_{i+1}\right)\right)$$
$$(4.3)$$

where smf_1 and smf_2 are smoothing functions

$$smf_1(t) = 0.5\left(-tanh\left(\left(t - t_{i+1}\right)\upsilon\right) + 1\right) \quad (4.4)$$

$$smf_2(t) = 0.5\left(tanh\left(\left(t - t_{i+1}\right)\upsilon\right) + 1\right) \quad (4.5)$$

with $\upsilon = max(f_i, f_{i+1})$. Thus, when $(t_{i+1} - t)\upsilon \gg 0$, $smf_2(t)$ is almost 0 and $smf_1(t)$ close to 1; and when $(t - t_{i+1})\upsilon \gg 0$, $smf_2(t)$ is close to 1 and $smf_1(t)$ close to 0. Only near the transition $((t - t_{i+1})\upsilon \gg 0)$ do both functions have a similar weight.

Figure 4.4(a) shows the wavemaker position and 4.4(b) the wavemaker velocity in X direction corresponding to the wavemaker movement used in the numerical experiment. No discontinuities in velocity are observed in the transition between different cycles.
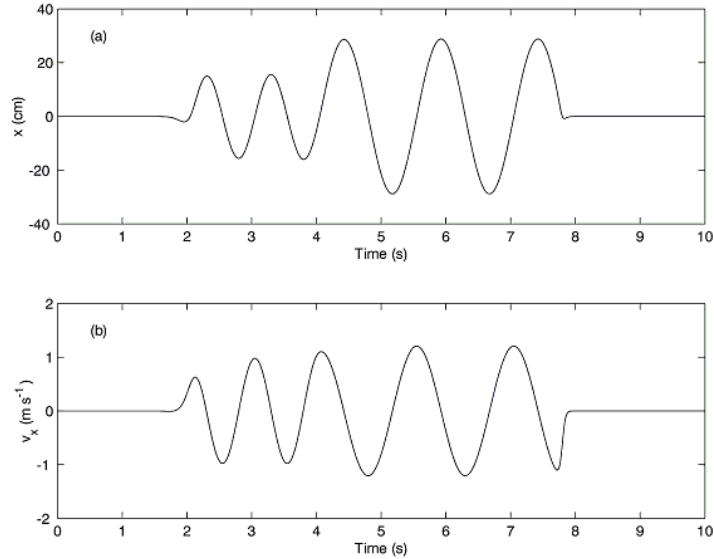
Figure 4.4: Wavemaker signal for transient wave generation. (a) Horizontal displacement; (b) Horizontal velocity.

Time stepping

A single predictor-corrector scheme ([Monaghan, 1989]) was used in our numerical simulations with a constant time step $= 4 \cdot 10^{-4}s$. This time step has proven to be small enough to fulfill the Courant condition and to control the stability of force and viscous terms ([Monaghan, 1992]).

Visualizing data

To define the free-surface position, a virtual wave gage is placed at a certain x position and at a certain z position much higher than the hypothetical free surface. The accumulated mass exerted by the fluid particles at position b is calculated from the mass of neighbor particles using the kernel definition:

$$m_b = \sum_a m_a \left(\frac{W_{ab} m_a}{\rho_a} \right) \tag{4.6}$$

When the mass obtained is bigger than some reference mass, m_{ref}, then the free surface is considered to be at z_b; if not, the z position of the test particle is reduced by $\Delta z = h/50$, and the procedure is repeated until arriving at the free surface. Thus the accuracy of the free surface estimation is $O(h/50)$. In the laboratory, velocities were measured at $x = 8.0m$, which corresponds to the leading edge of the deck. Different z locations starting from the bottom of the tank and with $\Delta z = 0.02$ cm were considered.

The velocity at any location a was calculated by averaging the velocities of the nearby fluid particles (b):

$$V_a = \frac{\sum_b V_b W_{ab}}{W_{ab}} \tag{4.7}$$

4.3.2 Results

To compare our results to the experimental ones ([Cox and Ortega, 2002]) numerical experiments were carried out with and without the deck. The numerical free-surface position in absence of deck, calculated using Eq. 4.6, is compared to the experimental one in Fig. 4.5.

The eight frames correspond to different positions from $x = 7.0m$ to $x = 10.5m$, measured from the wavemaker. Again $x = 8.0m$ coincides with the leading edge of the deck. The numerical signal is observed to be in good agreement with the experimental one, both in phase and amplitude, although there are several slight discrepancies between the experimental and numerical profiles. The first numerical wave, the one with the maximum at $T = 5.0s$ in first frame, is a bit delayed with respect to the experimental one. This is due to the different response of the numerical wavemaker at the beginning of the movement, where fast accelerations give rise to some water splash. In addition, the height of the

highest wave is slightly smaller in numerical experiments. This is probably due
to the fact that the numerical signal was filtered using a low pass filter, in such
a way that sharp peaks are considerably smoothed out.
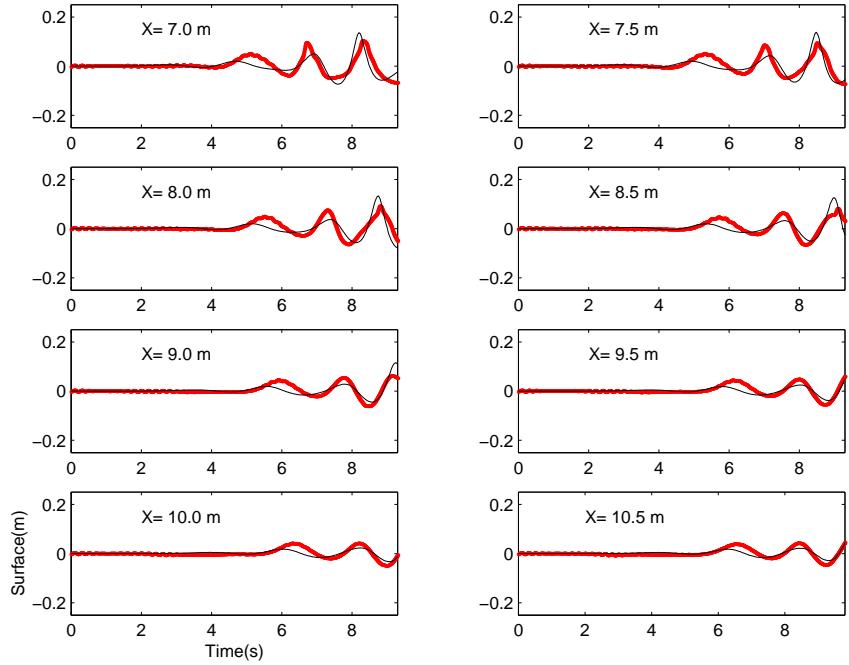
Figure 4.5: Free surface measurements in absence of deck. Comparison between
numerical signal (heavy solid line) and experimental signal (light solid line).

Figure 4.6 shows the vertical variation of the horizontal velocity correspond-
ing to the experiment by [Cox and Ortega, 2002] (these frames correspond to
six of the twelve snapshots shown in Fig. 5 in their paper). Data without the
deck are plotted using ($\bullet$) and with deck using ($\circ$).

There is an important effect of the deck on the velocity magnitude. The
horizontal velocity (v_x) in absence of the deck is bigger at the free surface and
decreases monotonically with depth. With the deck in place, the horizontal
velocity variation is no longer monotonic. At $t = 10.22s$ the wave has not
arrived at the deck and the profile is similar in both cases. At $t = 10.46s$ the
flow separation is observed in the case with deck. This separation reaches the
highest values at $t = 10.54s$, when the horizontal velocity is at a maximum of
nearly $0.7ms^{-1}$ and is fairly uniform with depth above the deck. Below the
deck the velocities are lower than velocities measured without the deck. Above
the deck, the horizontal velocity is observed to decrease at $t = 10.58s$ and it has
reversed at $t = 10.70s$. Below the deck, the horizontal velocities peak at $10.70s$

with a maximum velocity greater than that over the top. Finally, at $t = 10.78s$ the overtopping event has passed and the phase discrepancy between the top and bottom flows is resolved.
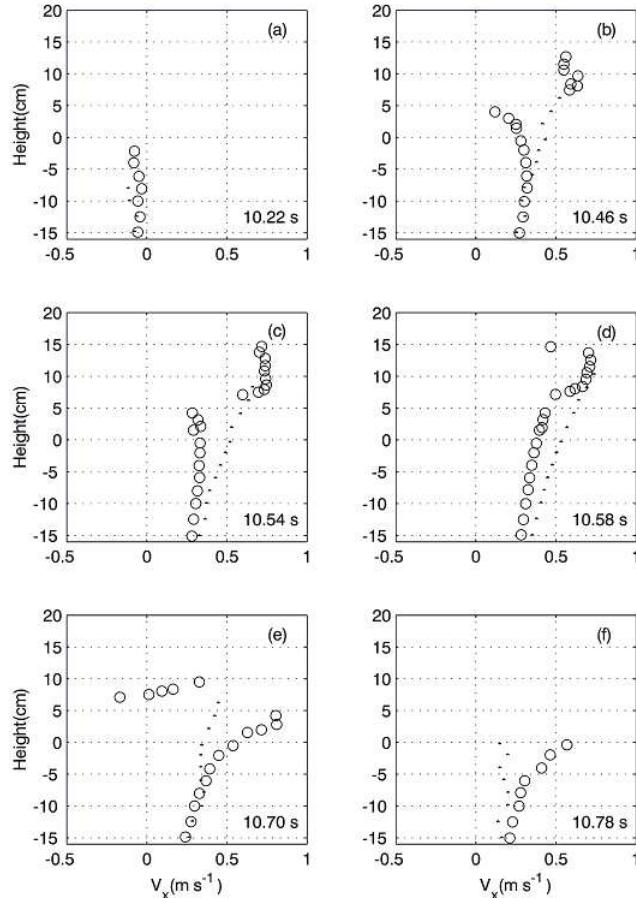
Figure 4.6: Vertical variation of experimental horizontal velocity. Data without deck (•) and with deck (○). Velocities were measured at the leading edge of the deck ($x = 8.0m$).

Figure 4.7 shows the numerical horizontal velocities, reproducing the experimental results shown in Fig. 4.6. Once again, the cases with (○) and without deck (•) are examined in six frames to represent the overtopping event. At $t = 10.19s$ the wave has not arrived at the deck and the profile is similar in both cases. At $t = 10.45s$ the flow separation is observed in the case with deck. This separation reaches the highest values at $t = 10.54s$, when the horizontal velocity is at a maximum, reaching values close to $0.7ms^{-1}$. Above the deck,

the horizontal velocity is observed to decrease at $t = 10.60s$ and it has reversed at $t = 10.70s$. Below the deck, the horizontal velocities peak at $10.70s$. Finally, at $t = 10.78s$ the overtopping event has passed and the phase discrepancy is resolved.
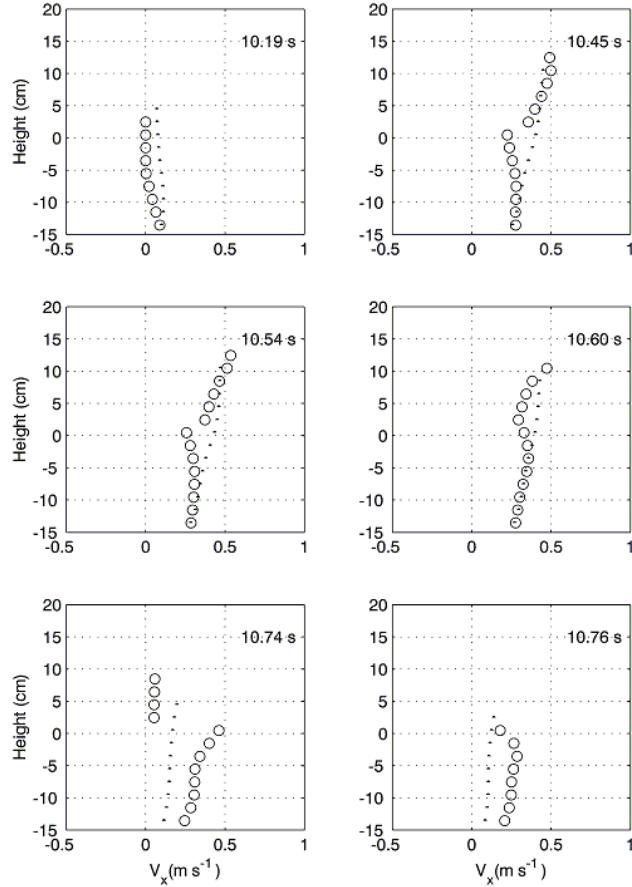
Figure 4.7: Vertical variation of numerical horizontal velocity. Data without deck ($\bullet$) and with deck ($\circ$). Velocities were measured at the leading edge of the deck ($x = 8.0m$).

Despite both figures being quite similar, some differences should be noted. Comparing Figs. (4.6) and (4.7) one can see the maximum velocities attained are slightly smaller in the numerical case. This is mainly due to the velocity calculation following kernel definition (Eq. 4.7), which averages the velocity in the neighborhood of the measuring point and tends to diminish the magnitude of higher values.

4.4 Jet formation under extreme waves

A numerical experiment similar to the one previously described was considered to study a higher wave hitting the structure. In this case, the flow is again split as it passes the deck as described in last subsections, but a strong jet is observed when the divided flows rejoin at the rear of the deck.

4.4.1 Model inputs

Most of the information described in subsection 4.3.1 such as the smoothing length, the treatment of the boundaries, the model initialization and the visualization procedure are the same. In this case, the only quantitative changes were the wavemaker movement and the tank geometry. The piston was considered to follow a periodic movement with $T = 2s$ and amplitude $A = 0.25m$. The tank was considered to be $9m$ long and the distance between the wavemaker at rest and the leading edge of the deck was $2m$. The deck height over the free surface at rest and the deck length were the same as described in Fig. 4.3. In this case, the total number of particles was 9,283 and the number of boundary particles 819.

4.4.2 Results

The piston movement described in last subsection gives rise to a wave with amplitude close to $0.35m$. Figure 4.8 shows the wave in the region close to the deck The left column represents the particle position at different phases during the overtopping process and the right column the corresponding particle velocities. The bottom row corresponds to the pressure measured at the rear of the deck, which was calculated by averaging the instantaneous pressure at the two last particles of the deck (note that the deck is constituted by 20 particles, so the averaged region represents 10% of the deck).
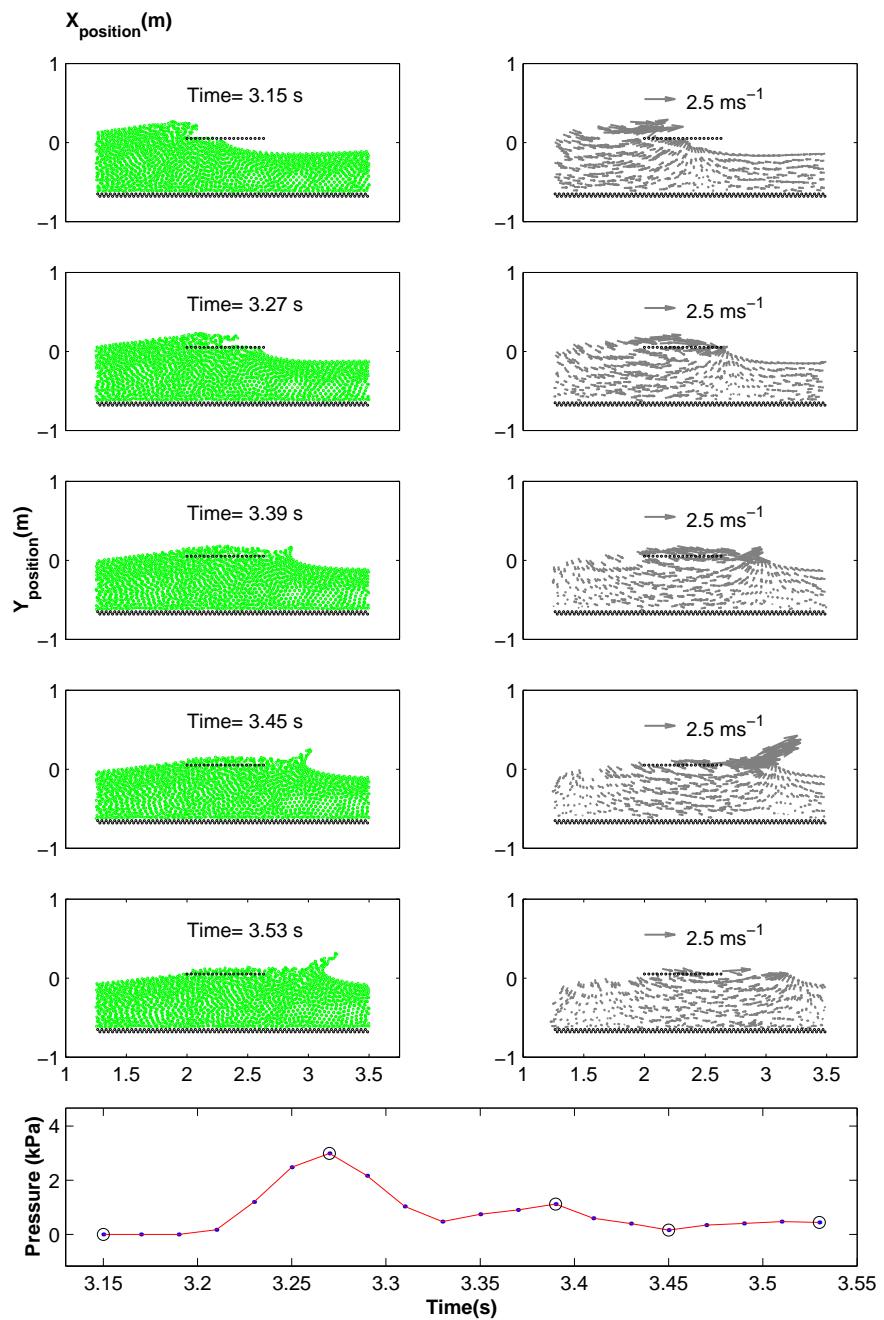
Figure 4.8: Overtopping and jet formation.

Those time instants represented in the previous frames are marked with a circle. At $T = 3.15s$ the wave starts propagating over the deck after passing the leading edge and the flow is accelerated in the horizontal direction close to the wave crest. At $T = 3.27s$ the flow is split into two, and the wave propagating under the deck is hitting the rear of the deck. This corresponds to the first (highest) peak in pressure representation. At $T = 3.39s$ the wave over the deck is arriving at the rear of the deck, which corresponds to the second peak in pressure. The starting of the jet can be observed both in particle and in velocity representation. At $T = 3.45s$ the part of the wave propagating over the deck has collided with the part of flow that propagated under the deck. After that collision the jet formation is reinforced close to the rear of the deck. This can be seen in velocity representation, where high vertical velocities are observed close to the rear of the deck. Finally, the wave generated by the jet is observed to propagate away from the deck ($T = 3.53s$), with a propagating velocity mainly directed in horizontal direction.

The observed velocity on the top of the deck can be compared to a dam break. Theoretically, the velocity at the toe of the dam break is $V = 2\sqrt{gH_0} = 2.88 \pm 0.07ms^{-1}$, where H_0 is the wave height over the deck ($H_0 = 0.211 \pm 0.006m$). The numerical velocity calculated following the leading part of the wave over the deck ($V_n = 2.99 \pm 0.06ms^{-1}$) is in good agreement with the theoretical one.

To observe the jet formation after the deck the maximum height attained by the wave at different x positions around the deck is plotted in Figure 4.9. The dark line close to bottom of the figure corresponds to the deck position.
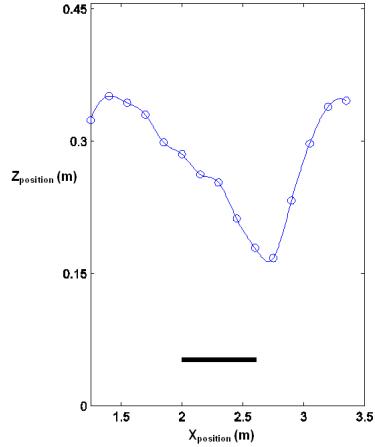
Figure 4.9: Maximum wave height attained near the deck.

The front reaches $0.35m$ before the deck, decreases when passing over the deck and increases suddenly after the deck. The line with crosses is associated with blocking the flow over the deck. The wave height is observed to reach values close to $0.35m$ just after the deck. This height decreases rapidly when the wave propagates over the deck, reaching a minimum value, $0.15m$ close to the rear of the deck. The jet formation just behind the deck gives rise to the lifting of the free surface reaching a height close to $0.35m$. This increase is much steeper than the previous decrease observed over the deck.

The jet is mainly generated by the collision of the fluid overwashing the deck colliding with the water from under the deck moving upwards after being released from the deck constraint. The importance of the fluid moving upwards after the deck was shown by a numerical experiment, where the same forcing conditions (piston amplitude and frequency) were considered and an obstacle was placed on top of the deck to prevent water from passing over the deck. Even in this case the presence of a jet can be observed as shown in Figure 4.8, although its amplitude is 33% lower than in the case with water passing over the deck. In summary, one can conclude that the jet is caused by two phenomena. A major part is the moving of fluid upwards from below the deck and the second part is the upper jet smacking into the water like a breaking wave.
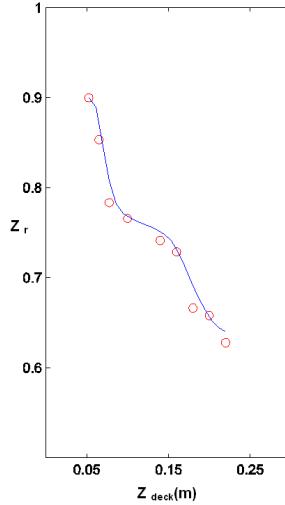
Figure 4.10: Relative height of the jet (Z_r) decrease in terms of the distance between the free surface at rest and the deck. The same piston displacement, $A = 0.25$ m and $t = 2s$, was considered in all simulations.

Jet formation is observed to be strongly dependent on the deck elevation above the mean free surface. Fig. 4.10 shows how the relative amplitude ($Z_r = Z_{jet}/Z_{wave}$) decreases whith the distance between the deck and the free surface.

4.5   Summary

The wave profiles generated by the method are in good qualitative and quantitative agreement with the experimental ones, both in phase and amplitude; and, in addition it has reproduced successfully the main features observed when a wave hits a horizontal platform. Thus, the initial continuous flow, flow separation when hitting the structure and further flow restoration match accurately the ones observed in experiments.

In addition, the appearance of a jet close to the deck rear has been analyzed under extreme conditions. The incoming wave amplitude is observed to decrease progressively over the deck and then to increase suddenly after passing the rear of the deck. The relative amplitude of the jet has been shown to decrease when increasing the distance between the free surface at rest and the deck.

Chapter 5

Dam Break Behavior

Dam break evolution over dry and wet beds is analyzed within the framework of the SPHysics model. The model is shown to fit accurately both experimental dam break profiles and the measured velocities. In addition, the model allows studying different propagation regimes during the dam break evolution. In particular, different dissipation mechanisms were identified: bottom friction and wave breaking.
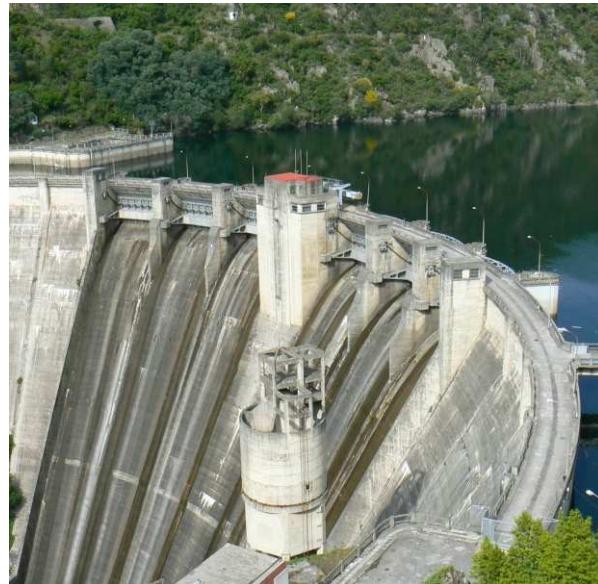
Figure 5.1: Picture taken by Benedict Rogers in "San Esteban" dam (Ourense).

5.1   Introduction

Typical dam break experiments show a rapidly moving tongue of water generated by the instantaneous release of a given volume of water confined in a rectangular channel. [Ritter, 1892] introduced a theoretical description of the two dimensional dam break problem for an inviscid fluid on a dry bed by solving the non-linear shallow water equations. The solutions provide a parabolic water surface profile that is concave downward. The front travels downstream with a celerity $V = 2\sqrt{gd_0}$, where g is acceleration due to gravity and d_0 is the initial quiescent water depth behind the dam. Although the theoretical approach assumes no boundary friction, experiments show good agreement with the theory, except for the leading edge of the wave as bottom friction affects the leading tip significantly. Actually, for a horizontal dry channel, the wave front celerity was observed to depend on time. The problem becomes much more complex when the dam break wave propagates over pre-existing still water (with an initial depth $d > 0$) ([Henderson, 1966]; [Montes, 1998]). In this case, the wave front celerity can be fitted in terms of d and d_0. [Leal et al., 2006] examine the influence of erodible beds. In spite of previous studies, the dynamics of dam breaks are far from being completely understood, especially when the wave front advances over a wet bed. Apart from the theoretical interest of this configuration, it can also contribute to the understanding of tsunamigenic waves when reaching the shoreline as mentioned [Chanson et al., 2003]. Models based on Smoothed Particle Hydrodynamics (SPH) are an option to address dam break evolution. Dam break over dry bed has been previously treated using SPH techniques ([Monaghan, 1994]; [Gómez-Gesteira and Dalrymple, 2004]; [Violeau and Issa, 2006] and [Crespo et al., 2007c]) showing reasonable accuracy with experiments. However, as far as we know, the method has only been previously applied to wet beds in [Gómez-Gesteira and Dalrymple, 2004], although they considered a very thin water layer close to bed.

The aim of this section is the study of the dam break problem and the effect of standing water in front of the dam with the SPH model. The method will be shown to fit accurately experimental results. In addition, the model captures most of the feature of a dam break over a wet bed; in particular allows analyzing the mixing and dissipation associated to the interaction between the dam break and the still water placed near bed.

5.2   The experiment

Here we use laboratory experiments by [Janosi et al., 2004] to validate an SPH model of dam break evolution over a wet bed. The channel, beginning at $x = 38cm$, is $955cm$ long and $15cm$ wide. The bottom and side walls of the channel were constructed with glass; the second part, comprising the lock and lock gate, is $38cm$ long and made from Plexiglas. The initial fill height of the lock (d_0) for our comparisons is taken as $0.15m$. The initial water depth in the

channel downstream of the lock was varied depending on the experiment (Figure 5.2). The experiments were recorded by two CCD cameras: a fixed fast-shutter camera (Sensicam, PCO Imaging), which provided the side or plan views and another small portable camcorder (Sony DCRPC115E) was fixed on a trolley that was moved along the tank, following the front. The position of the water front as a function of time was determined from digitized pictures. The gate separating the lock from the rest of the tank was removed from above at an approximate constant velocity ($V_{gate} = 1.5ms^{-1}$). The schematic arrangement of their experimental tank is shown in Fig. 5.3.
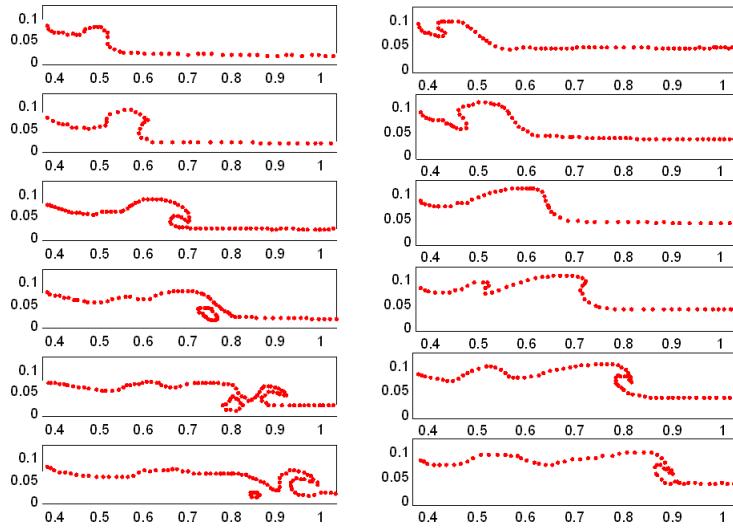
Figure 5.2: Experimental profiles digitized from Figure 2 in the paper of [Janosi et al., 2004]. The fluid depth in front of the lock is $d = 18mm$ in the left frames and $d = 38mm$ in the right ones.
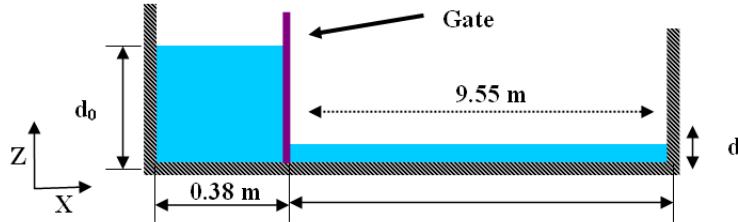
Figure 5.3: Schematic arrangement of the dam-break experiments.

5.3   2D Model validation

5.3.1   Model inputs

Initial conditions

Fluid particles were initially placed on a staggered grid with zero initial velocity. Nodes of the grid are located at $\overrightarrow{R} = ldx\,\overrightarrow{i} + mdz\,\overrightarrow{k}$ with a two- point basis (0,0) and (dx/2, dz/2) referred to the corner defined by $\overrightarrow{R}$. l and m are integers, $\overrightarrow{i}$, $\overrightarrow{k}$ are unitary vectors in X, Z directions and $dx = dz = 0.005m$. A smoothing length, $h = 0.006m$, was considered in all simulations.

Boundaries

Due to the particular geometry of the numerical experiment, two different boundary conditions were considered: fixed particles and gate particles. All of them are treated as DBPs since they follow the continuity equation and the equation of state, but they do not follow the momentum equation and the XSPH variant, in such a way that their position and velocity are externally fixed. Fixed particles (including bottom and fixed walls) are placed in two rows forming a staggered grid with $dx = dz = 0.005m$ and zero initial velocity. Their positions and velocities remained unchanged during the numerical experiment.

Gate particles are initially placed in a single row with a finer inter-particle spacing (dx/2, dz/2) to prevent particle penetration. Their velocities and positions are externally imposed to mimic the experimental movement of the gate according to $(V_x(t) = 0.0ms^{-1}; V_z(t) = 1.5ms^{-1})$. The movement of this gate will play a key role when fitting numerical results to experimental ones, since the gate velocity is on the same order of magnitude as the wavefront celerity $V \cong 2\sqrt{gd_0} \cong 2ms^{-1}$, in such a way that the experimental and numerical conditions are far from the usual dam break approach, where the gate between the lock and the channel is not considered.

Numerical parameters

The parameter B in the equation of state ([Batchelor, 1974]) was chosen to guarantee that the speed of sound is a factor 10 larger than the velocities in the model. This can be achieved by taking $B = 100gH_{ref}\rho_0/\gamma$, where H_{ref} is the maximum water height in the tank ($0.15m$ in the numerical experiments). The viscosity term given by [Monaghan, 1992] was calculated using $\alpha = 0.08$ and $\beta = 0$. In addition, fluid particles were moved using XSPH ([Monaghan, 1989]) with $\varepsilon = 0.5$.

The numerical tank was $9m$ long and $0.16m$ deep. The number of boundary particles was 3,879. The number of fluid particles depends on the experiment (specifically, the thickness of the water layer (d) in the channel before the gate is lifted). It ranges from 4,484 for dam break movement on a dry bed to 30,884 with a water layer comparable to the initial dam break height ($d = 0.078m$).

Time stepping

The Verlet algorithm ([Verlet, 1967]), was used in our numerical simulations. A variable time step δt was calculated according to [Monaghan and Kos, 1999] (See section 3.6).

5.3.2 Wave profiles

Experimental wave profiles ([Janosi et al., 2004]) were digitized for comparison with SPH results. The dimensions of the digitized snapshots are $0.38m \leq X \leq 1.04m$ and $0.0m \leq Z \leq 0.13m$. Distances were measured from the left-lower corner of the tank.

Two cases (Figure 5.4 and Figure 5.5) with different d values (0.018 and $0.038m$) were considered to compare numerical results and experiments. Experimental values are represented by red dots and SPH values by the blue surface. The model is observed to reproduce the experimental profiles in both cases. For $d = 0.018m$ (left column) the water initially placed behind the gate pushes the still water (first and second snapshots), generating the "mushroom" jet mentioned by [Janosi et al., 2004] and first reported by [Stansby et al., 1998] which suffers successive breakings. Bubble capture is reproduced by the model. A similar accuracy in numerical results is observed for $d = 0.038m$ (right column). Only a wave breaking is observed in this case.

Apart from this visual comparison, the observed difference between numerical and experimental results can be quantified considering two statistical parameters, comprised of experimental and numerical values:

$$A = \sqrt{\sum_{i=1}^{N}(Val_i^{num})^2 / \sum_{i=1}^{N}(Val_i^{exp})^2} \tag{5.1}$$

$$P = \sqrt{\sum_{i=1}^{N}(Val_i^{num} - Val_i^{exp})^2 / \sum_{i=1}^{N}(Val_i^{exp})^2} \tag{5.2}$$
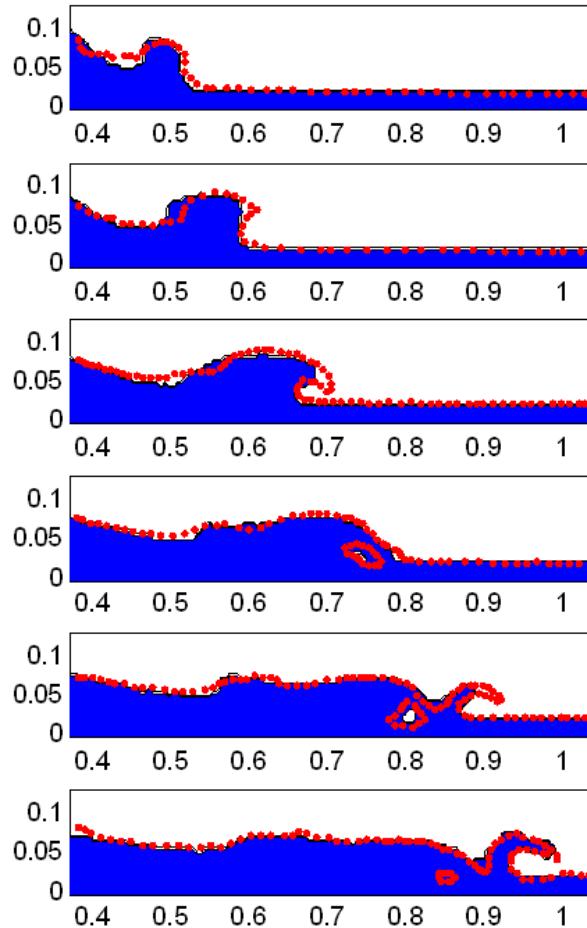
Figure 5.4: Comparison between experimental and numerical profiles of dam-break evolution over a wet bed ($d = 0.018m$). Experimental values are represented by red dots and numerical ones by the blue surface.

A perfect agreement between experiment and numerical model should result in $A \rightarrow 1$ (Eq. 5.1) and $P \rightarrow 0$ (Eq. 5.2). The good results obtained for $d = 0.018m$ (Figure 5.4): $A = 1.014$, $P = 0.076$ show the accuracy of the method.
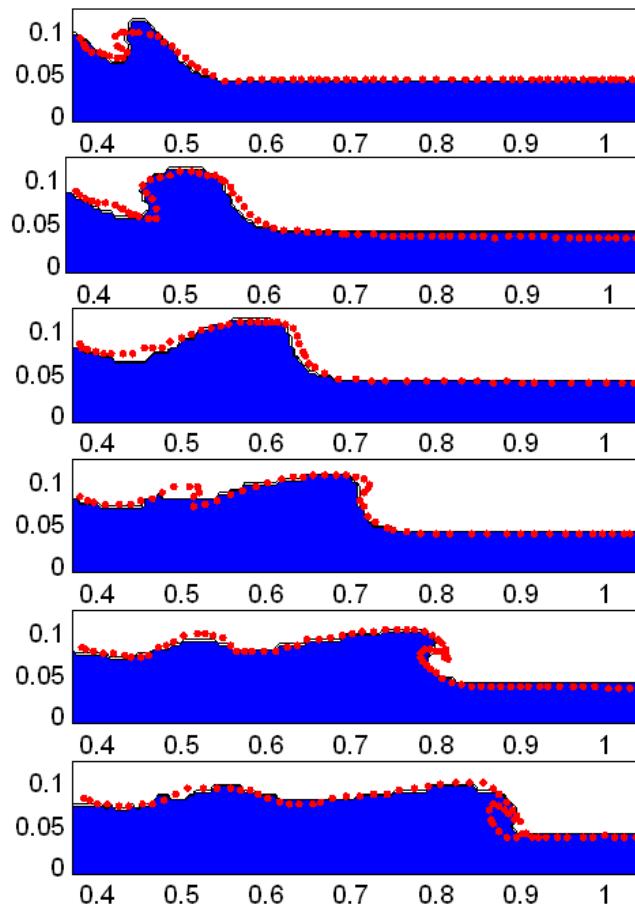
Figure 5.5: Comparison between experimental and numerical profiles of dam-break evolution over a wet bed ($d = 0.038m$). Experimental values are represented by red dots and numerical ones by the blue surface.

For fluid depth $d = 0.038m$ (Figure 5.5): $A = 1.012$, $P = 0.058$ also show the accuracy of the model.

5.3.3   Wave front velocity

The experimental and numerical velocities were averaged in space along the first 3 meters of the tank (Figure 5.6). Numerically, the position of the leading edge was calculated every $0.06s$ and velocity was obtained by linear fitting. Both velocities and distances are depicted in a dimensionless form. Velocity (V_N) is normalized with $c = \sqrt{gd_0}$ and d/d_0 is the ratio between the depth of the fluid layer near bed and the initial dam height. The normalized velocity is observed to decrease with d. The agreement between experimental measurements (light dots) and numerical results (dark squares) is excellent in most of the cases. Note that SPH velocity for dry bed is higher than observed in experiments, since experiments were not performed on a real dry bed, due to the impossibility of drying completely the tank.
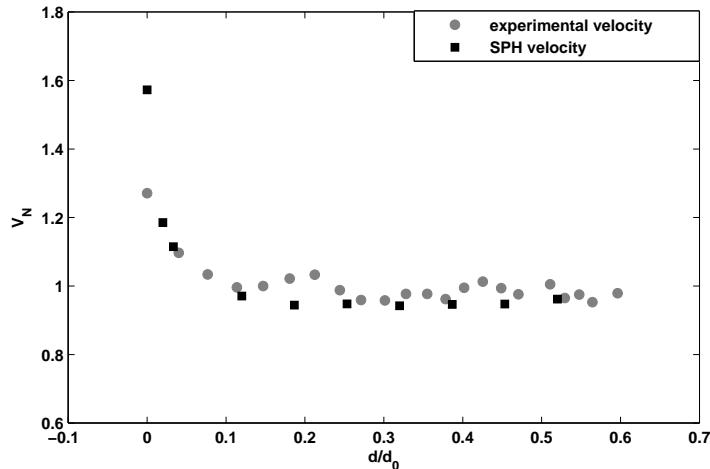
Figure 5.6: Comparison between experimental (light circles) and numerical (dark squares) dam-break velocity. The velocity was averaged in space during the first $3m$ in both cases.

In the figure 5.7, different instants of the dam evolution are depicted; gate movement, wave formation, wave breakings. Color represents pressure values.
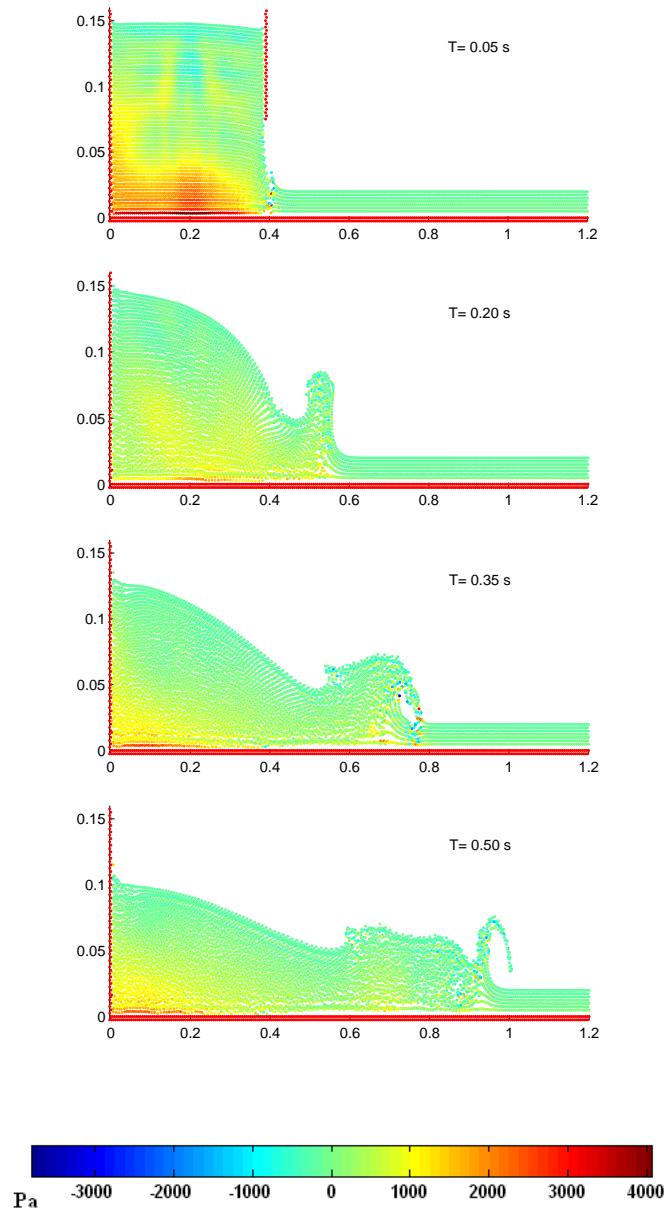
Figure 5.7: Different instants of the dam evolution for the case $d = 0.018m$. Pressure field is represented.

5.4   Dam-break analysis

Now that SPH has been shown to provide accurate results on dam break propagation, it can be used to analyze the dynamics of that propagation:

5.4.1   Mixing process

Dam-break propagation on a wet bed is strongly dependent on the interaction between both fluids. From now on, we will refer to the water initially placed behind the gate as lock water and to the still water initially placed beyond the gate as tank water.

As SPH is a Lagrangian method, the trajectories of all the fluid particles are known. In the [Janosi et al., 2004] experiments, clear water was released into a channel filled with dyed water. SPH is able to reproduce accurately the interface between both liquids as is shown in figure 5.8 corresponding to $d = 0.015m$.

The numerical interface between lock water (light color) and tank water (dark color) coincides with the experimental interface (black line) calculated from [Janosi et al., 2004] experiments (see Figure 14 in their manuscript).

Of course, the model does not pretend to reproduce the dispersion observed near the interface in the experiments. The existence of an area of partially dyed water - light colors in Figure 14 of [Janosi et al., 2004] - comes from the mixing between dyed (dark colors) and clear water (transparent). This mixing cannot be obtained in numerical simulations, where colors represent the origin of particles. Thus, the presence of a few clear particles in the dyed area (or vice versa) is masked by the massive presence of dyed particles in that area.
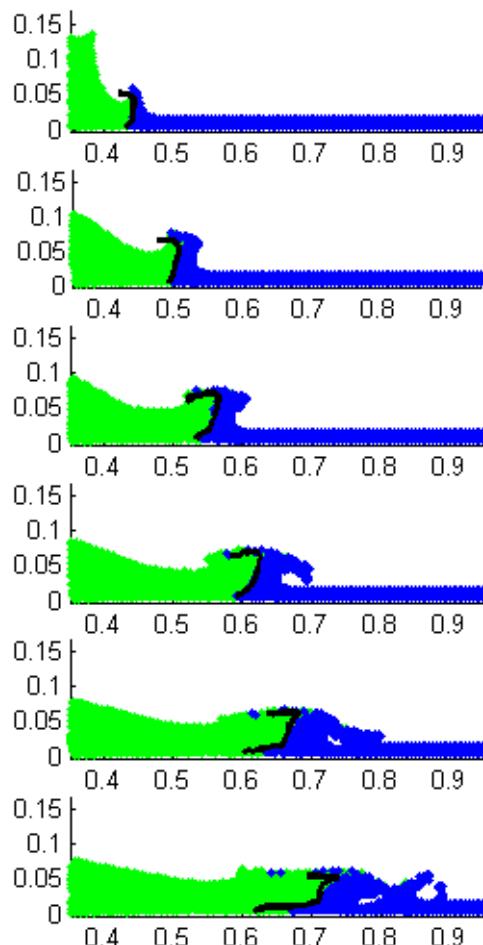
Figure 5.8: Experimental and numerical interface between lock and tank water ($d = 0.015m$). Green color corresponds to lock water, blue color to tank water and the black line to the experimental interface.

Figure 5.9 represents different instants during the dam-break propagation for different fluid depths (d). The dimensions of the snapshots are $0.0m \leq X \leq 1.6m$ and $0.0m \leq Z \leq 0.16m$. The observed behavior depends on the thickness of the water layer near bed. For $d = 0.018m$, lock water lifts the tank water near the contact point, which results in successive wave breakings as previously shown in Figure 5.4. A similar behavior with only a breaking was observed for $d = 0.038m$ and $d = 0.058m$. Nevertheless, the generation of a wave-train was observed for $d = 0.078m$. In spite of different transient behaviors observed for each configuration, the overall behavior is similar in all cases: lock water mainly pushes tank water without significant mixing. Only for $d = 0.018m$ there is some mixing close to the lock water tip due to the successive breakings previously as shown by [Janosi et al., 2004]. In addition, the higher the fluid level in the channel, d, the smaller the displacement of the lock water (see different columns in Fig. 5.9) as pointed by Janosi et al. 2004.
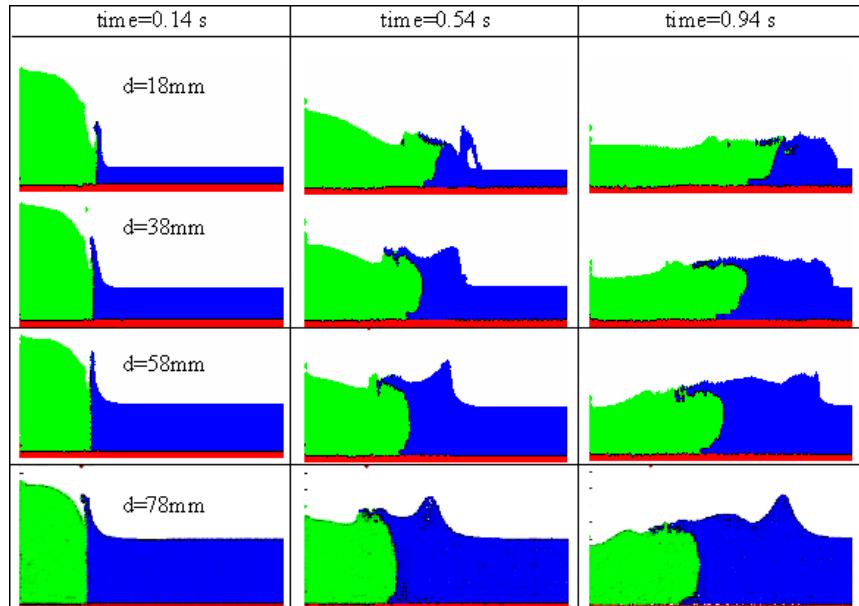
Figure 5.9: Different instants of dam evolution for different fluid depths. Green color corresponds to lock water and blue color to tank water.

5.4.2   Dam break evolution

The advance of a dam-break on a wet bed is far from being a stochastically process, especially at the beginning of the movement, where the interaction between both fluids gives rise to different behaviors depending on the depth ratio d/d_0 as previously shown. Thus, different propagation regimes can be observed depending on the zone: the observed horizontal velocity is considerably faster along the first 3 meters (dark squares in Figure 5.10) than along the first 6 meters (light circles in Figure 5.10). In addition, the fastest propagation corresponds along the first 3 meters to the dry bed and decreases with d. Note that the initial interaction between both fluids results in strong vertical displacements and vorticity generation, which tend to slow down horizontal propagation. The opposite behavior is observed when averaging in space along the first 6 meters. The slowest propagation in X- direction corresponds to the dry bed in this case. Movement is mainly controlled by bottom friction during the second 3 meters, which is enhanced when reducing the standing water depth.
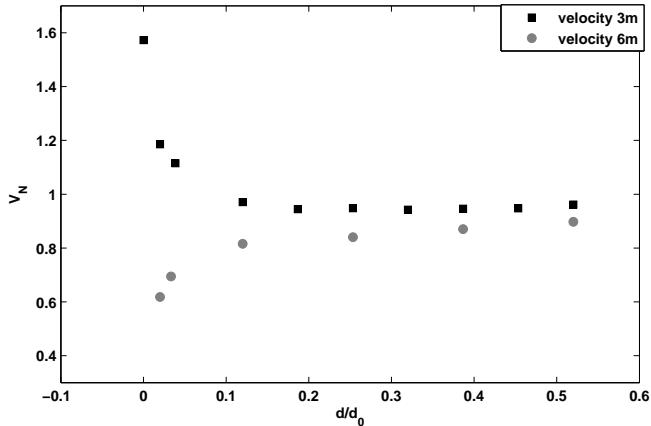
Figure 5.10: Spatial average of the velocity along the first 3 meters (dark squares) and the first 6 meters (light circles) for different fluid depths (d).

We should note that the average velocity in space is almost independent to the water level in the channel if $d/d_0 > 0.1$, this result is consistent with theoretical predictions ([Klemp et al., 1997]).

Changes in horizontal velocity are dramatic when lock water starts propagating (along the first 3 meters). Thus, the mean values shown in Figure 5.10 are far from instantaneous velocities, which are strongly dependent on the interaction between both fluids. Figure 5.11 shows the instantaneous horizontal velocity measured during the first 3 meters. First of all, the velocity corresponding to the dry bed (black solid line) is considerable faster than in the rest of cases. Velocity is observed to increase slightly during the first $1.25m$ and decreases from then on. The velocities corresponding to wet beds are considerably slower and present marked oscillations. These oscillations are especially marked for shallow water layers ($d = 18mm$ and $d = 38mm$) where local minima in velocity correspond to wave breakings shown in Figure 5.4. Thus, for example, the strong velocity decrease observed from $0.8m$ to $1.2m$ corresponds to the first wave breaking and eddy generation.
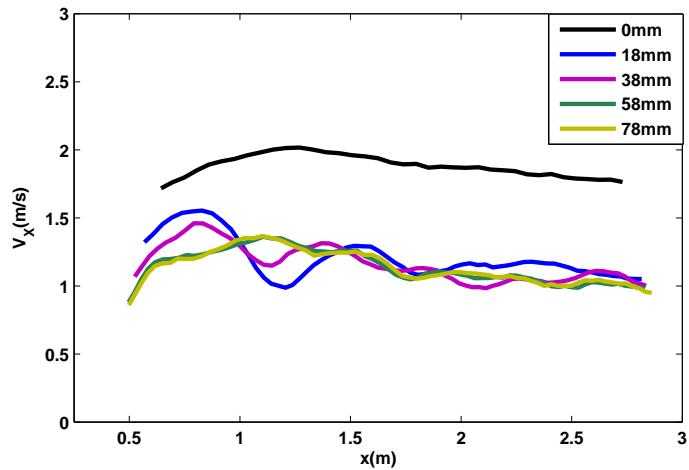
Figure 5.11: Velocity evolution with distance for different d values.

5.4.3   Energy dissipation

The transition between both propagation regimes can be studied in terms of the dissipated energy. This energy can be defined as:

$$\Delta E(t) = \left(\sum_{b=1}^{N} E_i^{kinetic}(0) + \sum_{b=1}^{N} E_i^{potential}(0) \right) - \left(\sum_{b=1}^{N} E_i^{kin}(t) + \sum_{b=1}^{N} E_i^{pot}(t) \right)$$
$$(5.3)$$

where N is the number of fluid particles. Note that this definition (in terms of an increment) allows comparing different experiments, where the amount of fluid and, hence, the number of fluid particles is different.

Figure 5.12 shows the energy dissipated in different experiments over a wet bed ($d = 0.018m$; $0.038m$; $0.058m$; $0.078m$) and over a dry bed ($d = 0m$). Energy dissipation is observed to increase in time in all cases. In addition, when considering experiments over a wet bed, the energy dissipation measured at each particular instant is observed to be higher for small d values, decreasing monotonically with d. The behavior observed over a dry bed is completely different, energy dissipation measured at the beginning of the movement is lower than in the rest of the cases, since the interaction between both fluids constitutes the dominant dissipation mechanism. However, bottom friction becomes in time the main dissipation mechanism, in such a way that energy dissipation becomes higher over a dry bed than over wet beds. Vorticity will be analyzed for different fluid depths to better understand this effect.
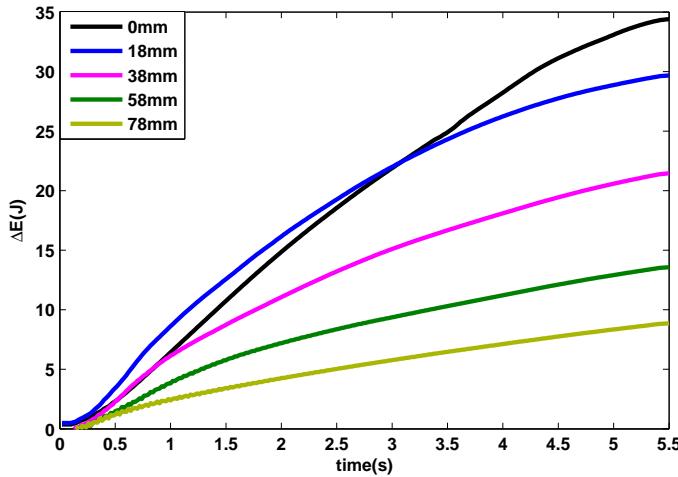
Figure 5.12: Energy dissipation for different d values.

5.4.4   Vorticity

SPH is used to examine complicated flow fields in breaking waves. SPH has an advantage over other computational methods to examine this problem as it allows studying vorticity and splashing of the fluid. The source of vorticity becomes important when studying wave propagation and breaking wave turbulence.

Figure 5.13: A plunging breaker on a beach creates a tremendous amount of vorticity (http://www.corbis.com).

As SPH is a Lagrangian method, the trajectories of each particle in the system are known at every instant and the vorticity of particle a is estimated following [Monaghan, 1992]:

$$\omega_a = (\nabla \times \mathbf{v})_a = \sum_b m_b \frac{\mathbf{v_a} - \mathbf{v_b}}{\rho_a} \nabla W_{ab} \tag{5.4}$$

The next figures show the Y-component of vorticity (perpendicular to X and Z plane, see Fig. 5.3). Positive vorticity values (red and yellow colors) correspond to clock wise rotation and negative ones (dark blue colors) correspond to counterclockwise rotation. The dimensions of the snapshots are $0.0m \leq X \leq 1.6m$ and $0.0m \leq Z \leq 0.16m$.
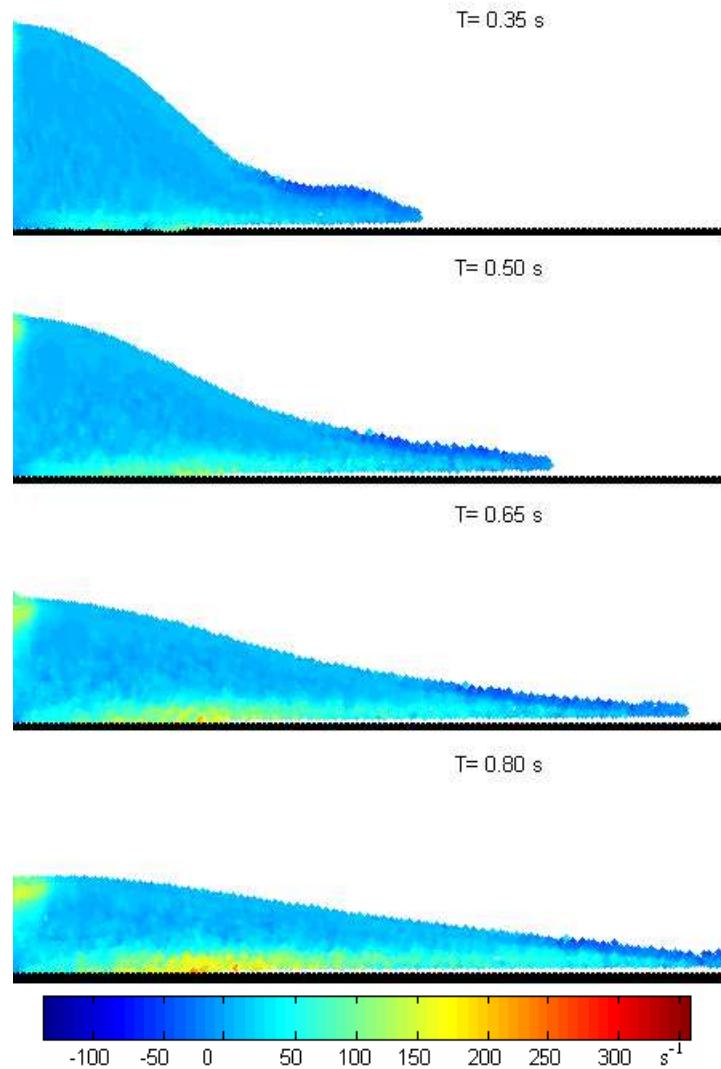
Figure 5.14: Vorticity plot for $d = 0m$.

Vorticity was calculated in dam-break evolution for dry bed (Fig. 5.14). The highest negative values appear near the tip of the dam and positive values appear near bed due to bottom friction.
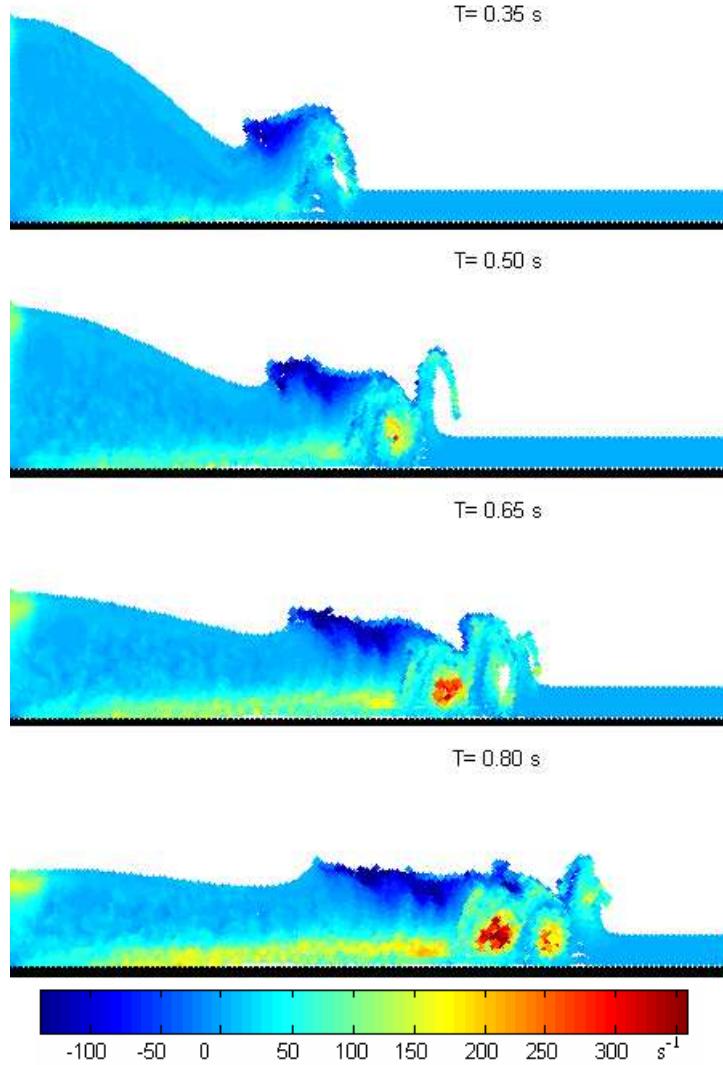
Figure 5.15: Vorticity plot for $d = 0.018m$ ($d/d0 = 0.12$).

Analyzing different instants for $d = 0.018m$ (Fig. 5.15), the wave formation is observed in this case. The water initially placed behind the gate pushes the initially still water (first snapshot), generating the "mushroom" jet mentioned by [Janosi et al., 2004]. Negative vorticity appears on the left side of this "mushroom" (dark blue colors) due to counter clock- wise water rotation. $T = 0.35s$ shows first breaking, that then generates the first positive eddy (red and yellow colors in $T = 0.50s$). $T = 0.65s$ shows the second breaking that is going to generate a second positive eddy (red and yellow colors in $T = 0.80s$).
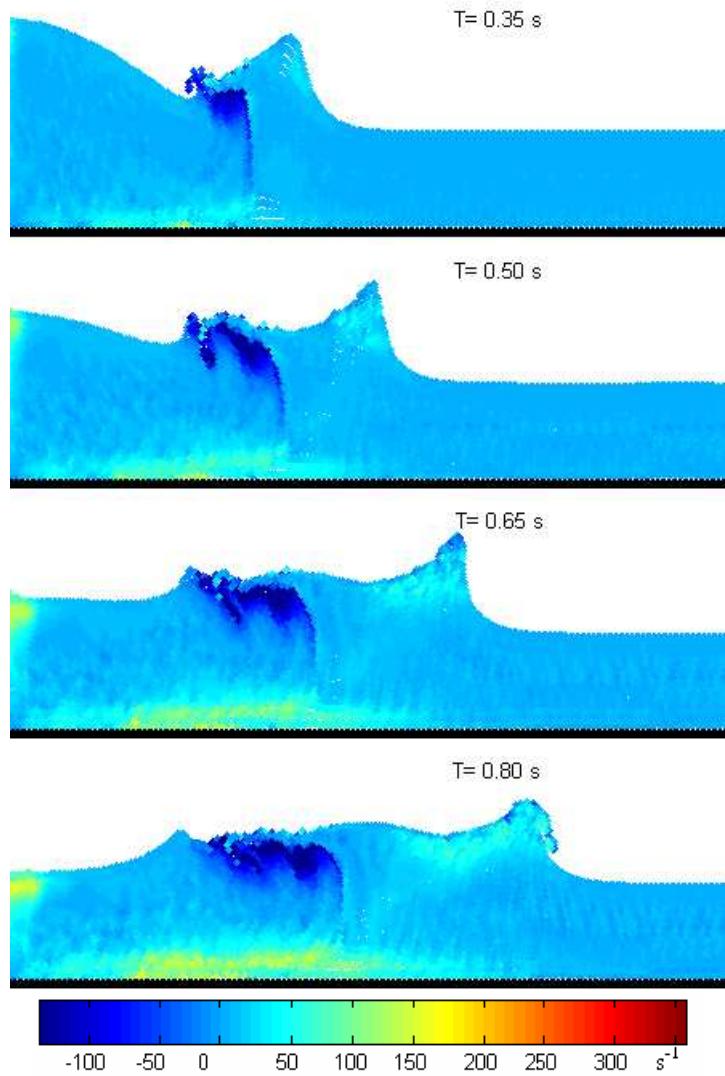
Figure 5.16: Vorticity plot for $d = 0.058m$ $(d/d0 = 0.39)$.

Vorticity for $d = 0.058m$ is also calculated (Fig. 5.16). Positive values appear mainly near bed due to close to bottom friction. There is no wave breaking so there are no positive eddies. Water initially placed behind the gate pushes the still water initially placed beyond the gate without significant mixing. The interface of two water masses is clear and it coincides with the negative vorticity (dark blue colors).

5.5   Summary

Experimental profiles and horizontal velocities are properly reproduced by the model.

The mixing process observed in experiments over wet bed between lock and tank water is almost negligible. Actually, the basic propagation mechanism is due to the pressure exerted by lock water on tank water. Some mixing is observed in experiments with a shallow water layer, where the successive wave breakings result in eddy generation. Two regimes are defined in dam evolution. Initial propagation (dam release) in horizontal direction is faster than observed for longer distances where velocity is mainly reduced by bottom friction. The difference between both regimes is higher for dry beds and shallow water layers.

Energy dissipation was observed to be responsible of both regimes. Energy dissipation for wet beds is higher at the beginning of the experiments, since breaking constitutes the main dissipation mechanism. However, bottom friction becomes in time the main dissipation mechanism, which is especially important on dry beds.

Vorticity is shown to depend on the fluid depth (d). Thus, when the dam-break propagates over a dry bed, positive vorticity is mainly observed near bed due to bottom friction. Low negative values are only observed at the leading tip. Vorticity over a wet bed depends on water height: eddy formation with positive vorticity is observed for $d << d_0$ due to wave breaking. Negative vorticity is also observed in this case due to the so called "mushroom" jet. The breaking process is stopped when increasing d, in such a way that negative vorticity is only observed at the interface between both fluids.

Chapter 6

Wave Structure Interaction

The interaction between a large wave and a coastal structure is studied with the 3D SPHysics model. The role of protecting barriers (dikes and seawalls) to mitigate the force and moment exerted on the structure is analyzed in terms of the dike height and the distance from the dike to the structure. The existence of different propagation modes (different ways for the water to surpass the protection barrier) has been identified. In general, the flow is split into two parts, one overtopping the barrier and the other one flowing around it. The interaction between both parts of the fluid is shown to be responsible for the force and moment exerted on the coastal structure.

Figure 6.1: "The great wave of Kanagawa" by Hokusai.

6.1   Introduction

A tsunami is a long period ocean wave (15 to 60 minutes) that is usually formed by an underwater seismic disturbance, a subaerial or submarine landslide, or a volcanic eruption. These waves travel at high speeds across the ocean with a small wave height (usually less than $1m$). Existing propagation models, based on linear or Boussinesq wave theories, provide very good predictive capabilities at sea. However, these waves can steepen to great heights when entering shallow water, becoming very nonlinear, and may break and produce great damage at the coast. Three dimensional tsunami models, needed to provide flow information when the waves impinge on coastal structures, cannot be applied in real time at present, since the travel time of the tsunami from its inception to its arrival at coastal areas is too short compared to the running time of those. Thus, the study of different tsunami scenarios on representative coastal regions constitutes a practical solution. The knowledge of the potential devastation of vulnerable regions can then constitute a powerful tool for urban designers and decision-makers.

Several computational models have been developed to analyze the effect of the tsunamis at the coastline (mainly forces acting on a structure). COBRAS (Cornell Breaking waves and Structures model) ([Hsu et al., 2001]) is an example of a 2D computational model. The code TRUCHAS (http://truchas.lanl.gov) is an example of a 3D finite volume model that can be used as a tool for tsunami-structure interaction (e.g., [Liu et al., 2005]). This model uses a two-step projection method to solve the Navier-Stokes equations and the Volume of Fluid Method (VOF) is applied to treat the free surface.

Models based on Smoothed Particle Hydrodynamics (SPH) are an option to address tsunami coastal effects, particularly tsunami forces on a structure. Despite the use of the SPH method in a great variety of problems, little has been done relative to three-dimensional problems, mainly due to the high computational cost of the method resulting from the need for many particles and a very small time step. The interaction between a single wave and an isolated structure is presented in [Gómez-Gesteira and Dalrymple, 2004], as a first approach to 3D calculations, showing the resulting velocities and the force exerted by the wave on the structure in good agreement with experimental data.

The aim of this section is the study of the interaction between large waves and structures by means of a three dimensional SPH method. Different scenarios for impact reduction are evaluated. In particular, a single dike was considered as the mitigation mechanism. Both the distance between protected structure and the dike and the dike height have proven to play a key role in the degree of protection.

6.2 3D Model validation

Although SPH has proved to provide accurate results in different numerical experiments dealing with interaction between large wave coastal structures ([Gómez-Gesteira and Dalrymple, 2004]; [Gómez-Gesteira et al., 2005]), the model will be validated here with different experiments with geometries similar to the one used in the mitigation experiments described in next section. In particular, the accuracy of SPH to describe wave profiles, wave velocities and forces exerted by waves on structures will be analyzed.

Model results were compared to experimental data provided by Yeh and Petroff at the University of Washington. This experiment was a dam break problem confined within a rectangular box $160cm$ long, $61cm$ wide and $75cm$ high. The volume of water initially contained behind a thin gate at one end of the box was $40cm$ long x $61cm$ wide x $30cm$ high. A tall structure, which was $12cm$ x $12cm$ x $75cm$ in size, was placed $50cm$ downstream of the gate and $24cm$ from the nearest sidewall of the tank. In this experiment an initial layer of water (approximately $1cm$ deep) existed on the bottom of the tank, due to the difficulty to completely drain the tank downstream of the gate before to the dam break. The configuration can be seen in Fig. 6.2.
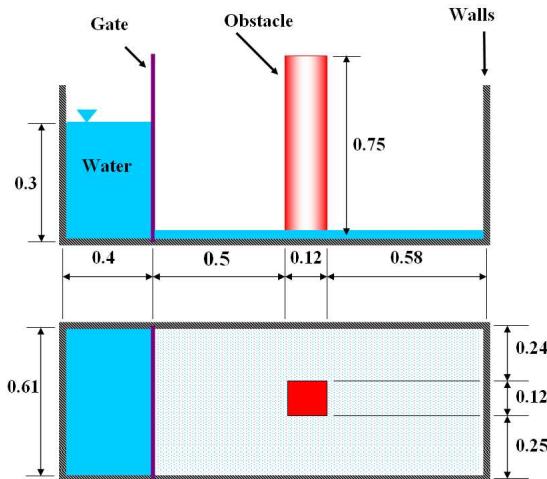
Figure 6.2: Initial configuration of the numerical experiment corresponding to interaction wave-structure.

This data set had been previously used by Raad at Southern Methodist University (http://engr.smu.edu/waves/solid.html) to validate his three dimensional EulerianLagrangian marker and microcell method ([Chen et al., 1997]). These experimental data were also used by [Gómez-Gesteira and Dalrymple, 2004]

within the framework of SPH calculations.

Experimental measurements included the time series of the force exerted on the structure and the time series of the fluid velocity in the dam break direction. Net forces were measured on the structure and velocities at a single point $14.6cm$ upstream of the center of the structure and $2.6cm$ off the floor of the tank, i.e., at $75.4, 31, 2.6cm$, referred to absolute tank coordinates. This experimental setup, included the moving gate which was lifted at a constant velocity ($V_z = 2.0ms^{-1}$) following [Arnason, 2005], was reproduced in the SPH numerical experiments. Fluid particles were initially placed on a staggered grid ($dx = dy = dz = 0.0225m$) with zero initial velocity. A smoothing length, $h = 0.033m$, was considered, being the total number of particles $np = 34000$.

Experimental data present some discontinuities and are not equally spaced, so the experimental series was treated prior to validation. Data were equally spaced ($\Delta t = 0.015s$) and discontinuities were filled by means of a cubic spline. Numerical data were smoothed by means of a running average using the nearest neighbors. This protocol reduces the presence of spikes in time series and enables the comparison between numerical simulations and experiments. The series were considered in the interval to better capture the collision between the dam break and the front of the structure. Figure 6.3 shows the good agreement between numerical (solid line) and experimental (dark circles) forces. The SPH model is able to reproduce both the position and the amplitude of the force peak generated by the collision between the incoming wave and the structure.
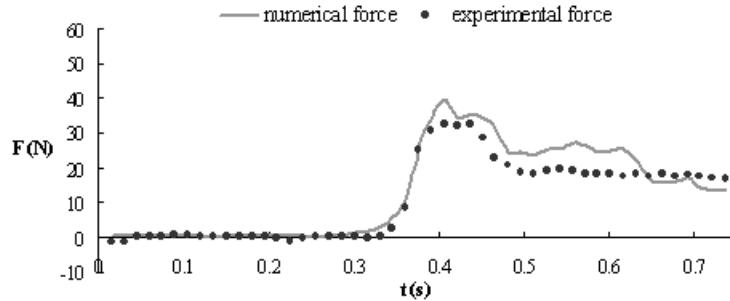
Figure 6.3: Comparison between numerical (solid line) and experimental (dark circles) forces exerted by the incoming wave on the structure.

Along the next frames (Figure 6.4), different instants of the interaction can be observed: $T = 0.01s$ corresponds to initial state of the simulation, $T = 0.20s$ corresponds to the wave advance, the impact between the wave and the structure is plotted at $T = 0.40s$ and the wave is hitting the back wall at $T = 0.60$.
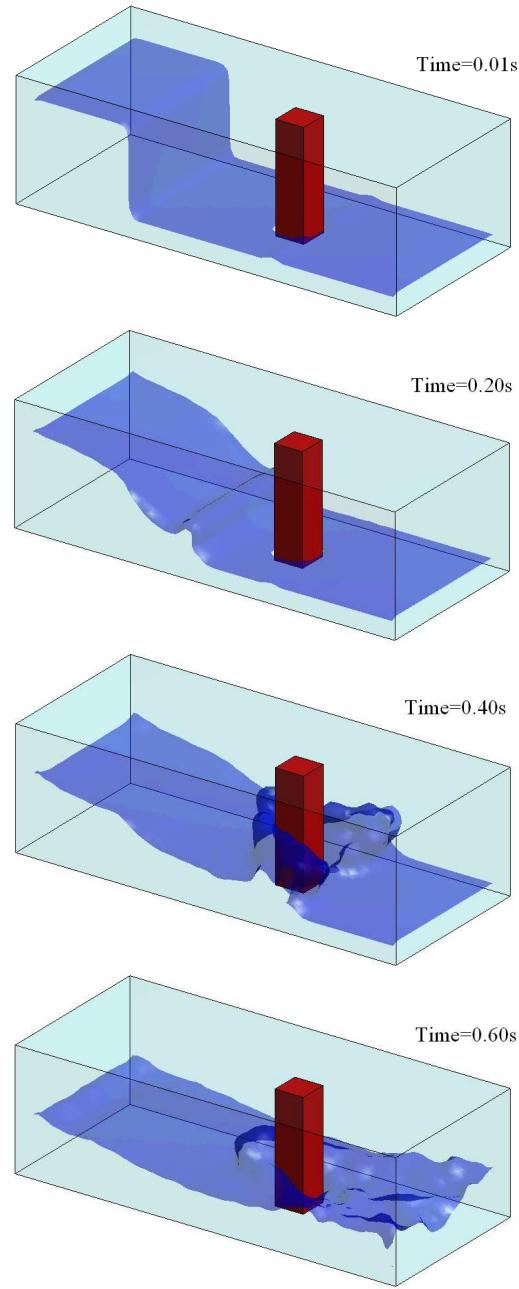
Figure 6.4: Different instants of wave-structure impact.

Apart from this visual comparison, the calculated root-mean-square (RMS) of the difference between experimental series is $4.1N$, which is satisfactory compared to the mean force exerted on the structure ($11.0N$). Note, that the RMS of the difference between consecutive realizations of this experiment can even reach values close to $6N$, depending on the stiffness of the structure as described in [Arnason, 2005].

6.3   Wave mitigation by a dike

A dike was used to mitigate the impact of the large wave on the structure. A sketch of the numerical setup can be observed in Figure 6.5. The lateral dimension of the box, the dimensions of the structure and the volume of water ($H_w = 0.3m$) initially placed behind the gate coincide with the ones given in previous section to validate the model and in [Gómez-Gesteira and Dalrymple, 2004]. In the present application, the distance from the gate to the dike was kept constant ($d_0 = 0.3m$) in all simulations, in such a way that the wave arriving at the protection barrier was the same. Two free parameters were used in this study: The dike height (H_D) and the distance from the dike to the structure (d). The width ($0.12m$) and thickness ($0.03m$) of the dike were kept constant in all experiments. In particular, the width coincides with the structure width ($Y_2 - Y_1$). Finally, the distance from the structure to the back wall was also constant and big enough ($0.63m$) to diminish the effect of reflection on the back wall on the area near the structure.
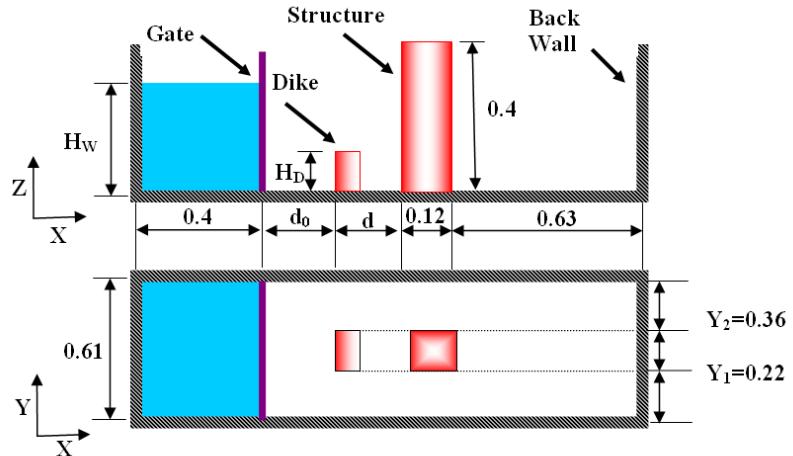
Figure 6.5: Initial configuration of the numerical experiment corresponding to wave mitigation by a dike.

Two variables will be analyzed in the following numerical experiments, the maximum force and moment exerted on the structure. The force will be calculated following the protocol described in [Gómez-Gesteira and Dalrymple, 2004] and the moment will be calculated as the summation of the force exerted at each structure point times the vertical coordinate of the point. The normalized force and moment will be considered instead of the measured values, since the main goal of the study will be to analyze the degree of protection generated by the dike. These normalized values (F_n and M_n) will be the ratio between the force (moment) exerted on the structure with and without protecting barriers ($V_n(d) = V_{with}(d)/V_{without}(d)$, where $V(d)$ is the force or the moment at distance d). From now on we will we refer to these variables as force and moment instead of maximum normalized force and maximum normalized moment.

6.3.1 Model inputs

Boundaries

Walls in the tank were built with two parallel layers of fixed boundary particles placed in a staggered manner ([Gómez-Gesteira and Dalrymple, 2004]).

Structures and dikes

Structures and dikes are implemented in a way similar to boundary particle. An empty square shaped structure was considered in the numerical experiments, being the walls built by means of a single layer. On the other hand, dikes were considered to be solid and built in the same manner as the tank walls.

Initial Conditions

Fluid particles were initially placed on a staggered grid with zero initial velocity. This grid can be seen as a cubic lattice whose nodes are located at with a two- point basis $(0, 0, 0)$ and $(dx/2, dy/2, dz/2)$ referred to the corner defined by $\overrightarrow{R} = ldx\,\overrightarrow{i} + mdy\,\overrightarrow{j} + ndz\,\overrightarrow{k}$. l, m and n are integers and $\overrightarrow{i}$, $\overrightarrow{j}$ and $\overrightarrow{k}$ are unitary vectors in X, Y and Z directions. The particles are assigned an initial density ρ_0 that needed to be adjusted to give the correct hydrostatic pressure when the pressure is calculated from the equation of state (Eq. 2.20).

Time stepping

The Verlet algorithm ([Verlet, 1967]), was used in our numerical simulations. A variable time step δt was calculated according to [Monaghan, 1992].

Numerical parameters

The number of particles ($n_p \in [2700, 30000]$) will also depend on H_D and d. The variable time step will depend on the particular features of every single experiment, in general $dt \in \left[10^{-5}, 10^{-4}\right] s$.

6.3.2   Collision without protection

Force and moment exerted on the structure without a protection barrier were studied in order to characterize the system. Figure 6.6 shows the decrease (in percentage) of both variables with the distance ($D = d_0 + d$, see figure 6.5) from the dam break to the structure. This decrease (about 20% in force and 25% in moment from the considered distances) shows a linear trend with a correlation coefficient of $R^2 = 0.91$ for the force and $R^2 = 0.83$ for the moment.
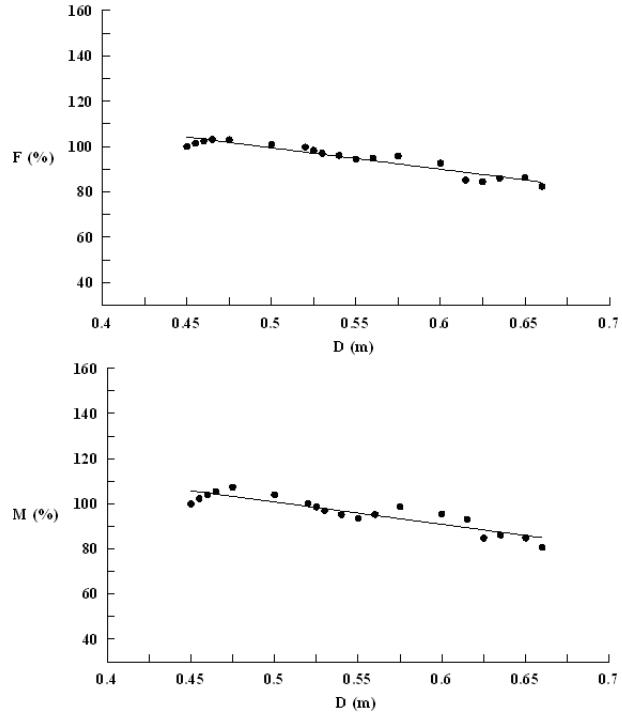
Figure 6.6: Dependence of the force and moment calculated without protection barrier on the distance between the dam break and the structure.

The percentage of force was calculated dividing the force at any distance by the force at the shortest distance considered in the numerical experiments. The percentage of moment ($M(\%)$) was calculated in a similar way. Both variables are observed to decrease linearly with distance.

The variation in force and moment is mainly due to the finite size of the reservoir and to the non- stationary nature of the front when hitting the structure. Note that distance from the gate to the structure varies from 0.45 to 0.65m, which is comparable to the extent of the reservoir, both in X (0.40m) and in Z direction (0.30m). Thus, the theoretical description given by [Ritter, 1892] is

not completely valid as shown in [Janosi et al., 2004].

6.3.3   Protection by a tall dike

The first case to be analyzed corresponds to a tall dike. This dike is considered to be tall, since the dike height, $H_D = 0.125m$, is about 74% of the water average height initially placed in the area between the front wall ($X = 0$) and the dike, $H_n = H_w \left(0.4/\left(0.4 + d_0\right)\right) \cong 0.17m$.
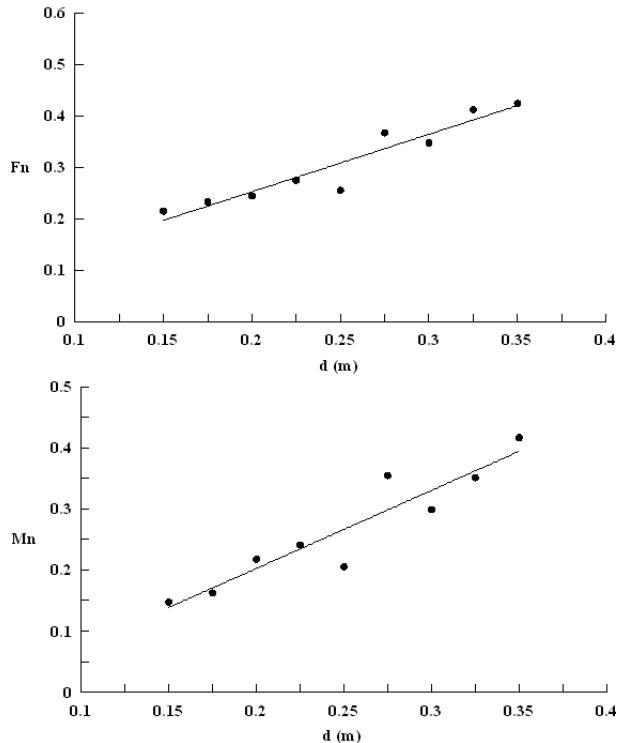
Figure 6.7: Dependence of the normalized force (F_n) and normalized moment (M_n) on the distance dike-structure for a tall dike ($H_D = 0.125m$).

Figure 6.7 shows the normalized force and moment exerted on the structure for different distances (d) from the dike to the structure. In all cases, the dike generates an effective protection, which ranges from 60 to 80% in force and from 60 to 85% in moment. In addition, the protection is monotonically decreasing with distance because force and moment increase linearly with d ($R^2 = 0.90$ and $R^2 = 0.87$ respectively).

This dike is high enough to prevent water for surpassing it, in such a way that the water mass is forced to surround the obstacle, splitting the flow. Thus, the intensity of the collision with the structure placed behind the protection barrier depends on the distance barrier- structure. Actually, flow reconstruction is more effective when increasing this distance, given rise to an increase on the force and moment exerted on the structure.

Finally, the times corresponding to the maxima in force and moment exerted on the structure are observed to coincide for the same distance d. This is due to single origin of the water hitting the structure since the whole water body is forced to surround the obstacle as we mentioned above.

Different instants of the wave-structure impact mitigated by a tall dike are represented in figure 6.8: $T = 0.01s$ corresponds to initial state of the simulation, $T = 0.20s$ corresponds to the wave arrival at dike position, the impact between the wave and the dike is plotted for $T = 0.40s$ and at $T = 0.60$ it is observed how the dike is tall enough to prevent a jet formation.
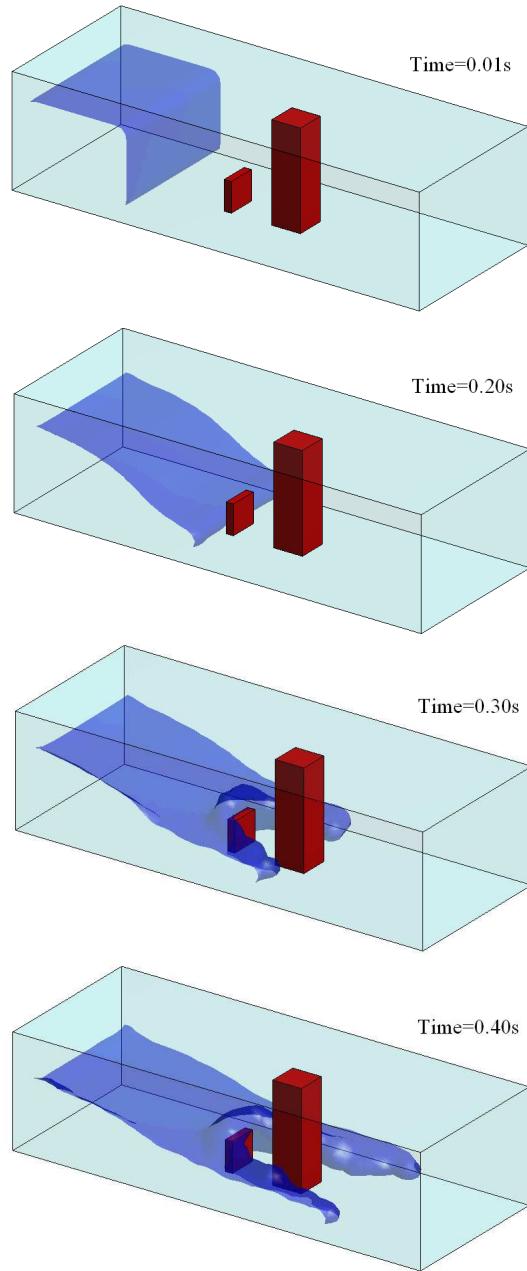
Figure 6.8: Different instants of wave-structure impact mitigated by a tall dike.

6.3.4   Protection by a short dike

The second case to be analyzed corresponds to a short dike. This dike is considered to be short, since the dike height, $H_D = 0.05m$, is only about 30% of the average water height that would fit in the area contained between the front wall ($X = 0$) and the dike, $H_n = H_w \left(0.4/\left(0.4 + d_0\right)\right) \cong 0.17m$. Figure 6.9 shows the normalized force and moment exerted on the structure for different distances from the dike to the structure. Note that the dike generates a protection lower than in previous case. This protection ranges from 25 to 40% in force and from 5 to 40% in moment.
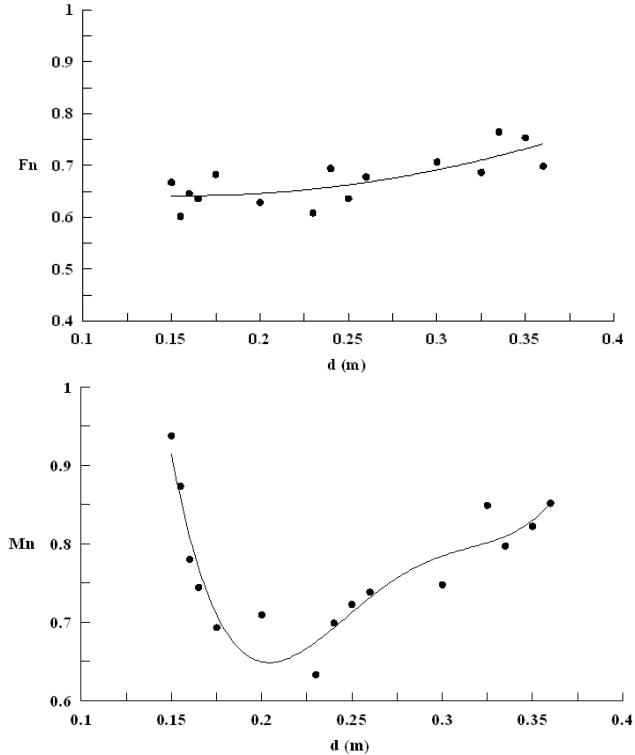
Figure 6.9: Dependence of the normalized force (F_n) and normalized moment (M_n) on the distance dike-structure for a tall dike ($H_D = 0.05m$).

Contrary to the previous case, the dependence of force and moment on the distance shows different patterns. Thus, the force (Fig. 6.9a) increases with the distance as observed in previous case, although in a smoother way. Once again, the increase in the effectiveness of flow reconstruction with the distance is responsible of force increase. The variation in moment with wall location (Fig. 6.9b) is more complicated. Thus, for short distances ($d = 0.15 - 0.23m$) from the structure, the moment decreases markedly with the distance, increasing in

a smoother way for long distances. Actually, the dike behaves like a ramp for short distances projecting the impact point towards the upper part of the structure. This behavior was postulated to have occurred in the Thailand during the December 26, 2004 tsunami, where a sloped seawall ramped the tsunami run-up jet into the upper story of a building ([Dalrymple and Kriebel, 2005]; see Figure 6.10).

Figure 6.10: Picture taken in Patong Beach in Thailand showing the run-up ramped into second floor.

The moment decrease with distance can be understood in terms of the jet profile shown in Figure 6.11. This jet height, which was calculated in absence of the large structure, is shown to decrease with the distance to the dike. Finally, the moment increase observed for long distances is due to the effectiveness of flow reconstruction as previously mentioned for tall dikes. No direct impact between wave and structure is present for these distances, which is consistent with the jet fall shown in Figure 6.11.
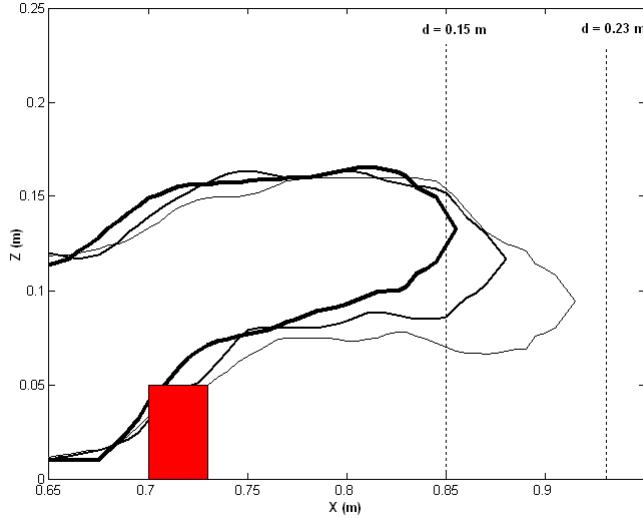
Figure 6.11: Jet profile corresponding to different instants during dike over-topping. The vertical lines correspond to the region with a sharp decrease in moment ($H_D = 0.05m$).

The flow splitting characterized by water overtopping and surrounding the dike is illustrated in Figure 6.12 where the successive collisions both in lateral and top view are depicted for a short distance, $d = 0.15m$. The Lagrangian nature of SPH allows following the movement of single particles, identifying their origin, in such a way that the evolution of the particles hitting the structure can be traced back. The particles represented by green circles correspond to those particles whose initial Y position coincides with the Y- extent of the dike and the structure (central band). The rest of particles are represented by blue dots (lateral bands).
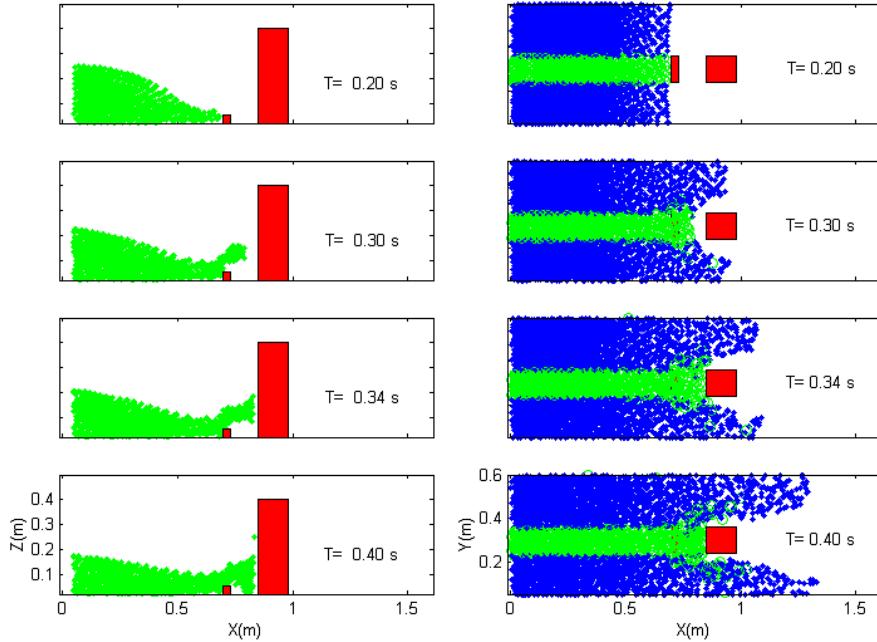
Figure 6.12: Lateral and top view of the water collision with the dike and the structure ($H_D = 0.05$ m and $d = 0.15m$).

Considering the lateral view of the central band (left panels in Fig. 6.12), $T = 0.20s$ corresponds to the water arrival at the dike, at $T = 0.30s$ the water jet is flying over the dike and approaching the structure, which is hit at $T = 0.34s$ generating the maximum in moment, finally, at $T = 0.40s$, the amount of water colliding with the structure has increased generating the maximum in force. In addition, the top view (right panels) in Figure 11 can be used to complement the information giving by the lateral view. Thus, the green particles overtop the dike, $T = 0.30s$, and collide with the structure, $T = 0.34 - 0.40s$, while the dark ones are scattered by the dike, $T = 0.30s$, and only a small percentage hit the structure. The different collisions of water and structure showed in Figure 6.12 show the existence of a delay between the instants of maximum force and moment on the structure. This delay is only present for short distances as shown in Fig. 6.13. Thus, for short distances the jet impinges directly on the structure giving rise to a maximum in moment, while the water continues surrounding the dike and accumulating in the vicinity of the structure generating a maximum in force delayed in time. However for long distances, both instants coincide since there is not direct collision between the jet and the structure.
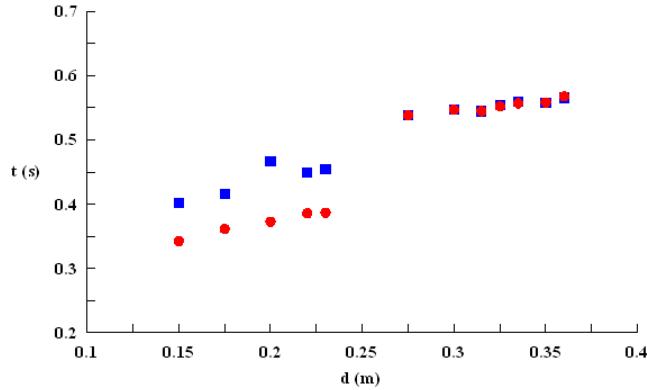
Figure 6.13: Dependence of the instants of maximum force and moment on the distance from the dike to the structure. The instants of maximum force (blue squares) are delayed with respect to the instants of maximum moment (red circles) for short distances.

The existence of these two different regimes can also be observed by analyzing the volume of water hitting the structure at the times of maximum force and moment (Figure 6.14). The volume was calculated starting from the number of particles placed in the rectangle determined $X_1 = 0.4+d+d_0-2hm$, $Y_1 = 0.24m$, and $X_2 = 0.4 + d + d0$, $Y_2 = 0.36m$ (see Figure 6.5). Note that $2h$ corresponds to twice the smoothing length of the method, which determines the distance at which the interaction between two particles becomes zero. V_C is the volume of water constituted by particles satisfying the previous condition and initially placed at the central band (green particles in Fig. 6.12) and V_L the volume of water constituted by particles satisfying the same previous condition but initially placed at the lateral bands (blue particles in Fig. 6.12). Both in force and moment, V_C is considerably bigger than V_L, since most of the particles are able to overtop the short dike. In addition, V_L tends to increase with d. Note that water from lateral bands is initially scattered by the dike, although it tends to re-enter the central zone. This process becomes more important when the distance between the dike and the structure to be protected increases giving rise to the increase in force and moment shown in Fig. 6.9. The evolution of V_C with d is more irregular and follows a different pattern in force and moment. The moment evolution is characterized by the two different regimes, for short distances, the dike generates a jet that is projected toward the upper part of the structure, generating an important moment with a small amount of water. The impact point decreases when increasing the distance from the dike to the structure, so decreasing the exerted moment. In fact, for long distances there is not direct collision between the jet and the structure and the timing of maximum moment coincides with the maximum force generated by a bigger number of particles after surpassing the dike. As we mentioned above, the maximum force is not generated by the jet, and it is characterized by a large amount of water.
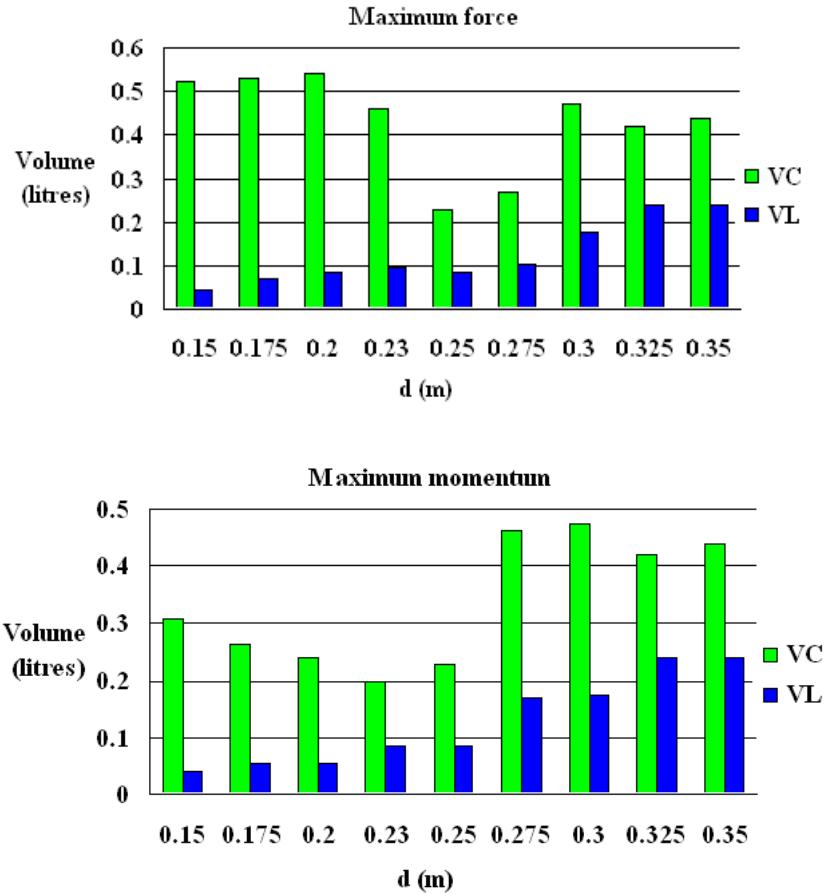
Figure 6.14: Volume of water classified according to their origin at the instants of maximum force and moment ($H_D = 0.05m$). V_C corresponds to the water initially placed at the central band ($Y \in [0.24, 0.36]$, see Figure 6.5). V_L corresponds to the water initially placed at the lateral bands.

6.3.5 Protection by a dike of intermediate height

This dike height is considered to be intermediate because it represents a case between tall and short dikes previously analyzed. The dike height, $H_D = 0.0875m$, is about 51% of the water average height initially placed in the area between the front wall and the dike, $H_n = H_w \left(0.4 / \left(0.4 + d_0 \right) \right) \cong 0.17m$. In this case water is able to overtop the dike but does not generate a jet as previously described for short dikes.

Figure 6.15 shows the normalized force and moment exerted on the structure for different distances from the dike to the structure. In all cases, the dike generates an effective protection much higher than in case of the short dike, but lower than for tall dike. This protection ranges from 25 to 65% in force and from 25 to 70% in moment.
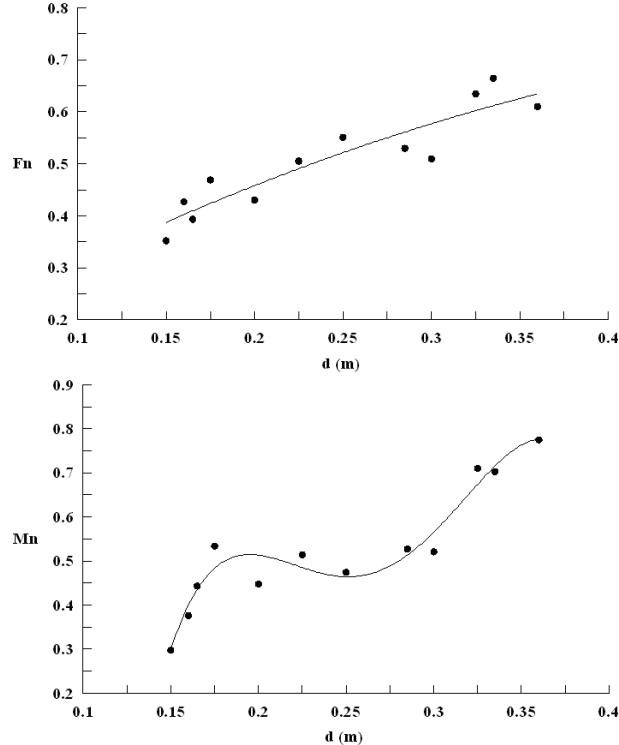
Figure 6.15: Dependence of the normalized force (F_n) and normalized moment (M_n) on the distance dike-structure for a tall dike ($H_D = 0.0875m$).

Once again, the force and moment patterns are different: Thus, the force tends to increase with distance with flow reconstruction being the mechanism responsible for this increase as previously mentioned for short and tall dikes. Three regimes can be observed for moment evolution. Moment tends to increase with distance for short distances, reaching a plateau for intermediate distances and increasing again with distance for long distances. The existence of these regimes can be explained in terms of the interaction between the water overtopping and surrounding the barrier. Thus, the moment increase observed for long distances is due to the effectiveness of flow reconstruction as previously mentioned for tall and short dikes.
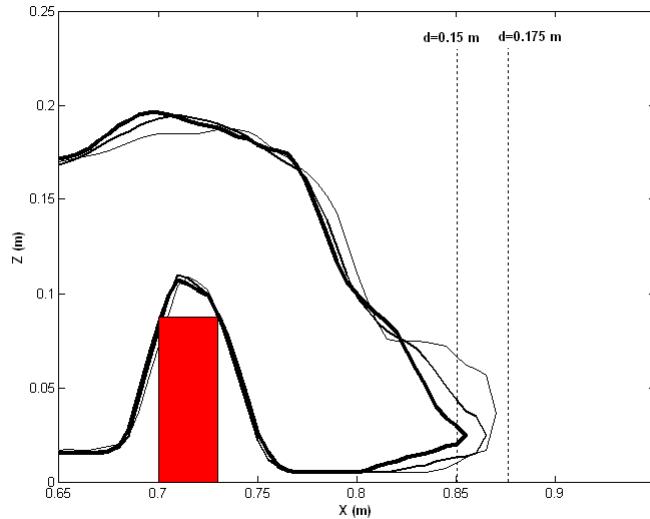
Figure 6.16: Water profile corresponding to different instants during dike over-topping. The vertical lines correspond to the region with a sharp increase in moment ($H_D = 0.0875m$).

The behavior observed for short distances is mainly due to the effect of wa-ter overtopping the barrier. As we see in previous cases, water from lateral bands is scattered by the barrier and has little effect on the collision with the structure. Nevertheless, water from the central band overtops the barrier and collides directly with the structure. In spite of no jet formation and hence no direct collision water- structure, there is considerable water splash after hitting the ground as shown in Figure 6.16. The splashed water is projected in such a way that the impact point on the structure increases when increasing the dis-tance dike- structure, at least for short distances ($d = 0.15 - 0.175m$).

This profile can be compared to the previously depicted in Figure 6.11 to observe the differences between both overtopping modes. Once again, the profile was calculated in absence of the structure to show the extent of the overtopping generated by the dike without the influence of additional structures. Finally, the moment is observed to be almost constant for intermediate distances since the splash water falls again before reaching the structure at the same time as water coming from the lateral bands starts entering the central zone. In addition, the origin of the different particles hitting the structure (Fig. 6.17) was studied following the same protocol as in Figure 6.14. In both cases (force and moment) V_C is observed to decrease with distance and V_L to increase. This effect is due to the separation between the dike and the structure, thus the higher the distance the higher the probability of the initially scattered particles to hit

the structure. In fact, for short distances, the amount of water overtopping the dike, represented by V_C, is the dominant factor on the exerted impact, being dominant the amount of water surrounding the dike, V_L, for long distances. For intermediate distances both contribute in a similar amount.
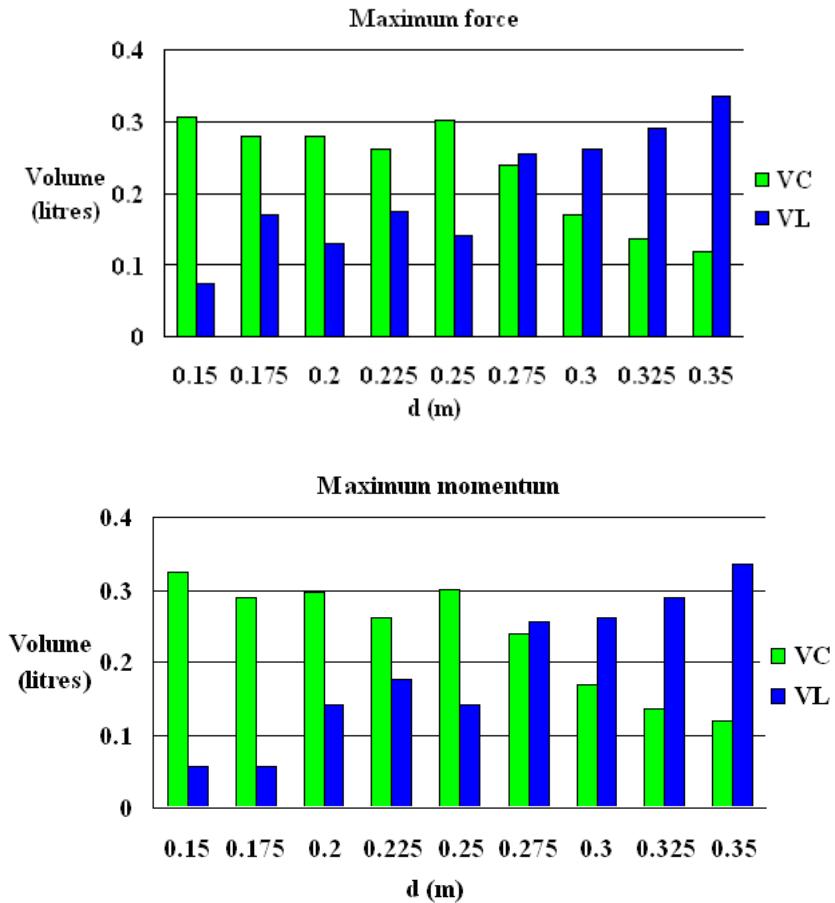
Figure 6.17: Volume of water classified according to their origin at the instants of maximum force and moment ($H_D = 0.0875m$).

6.3.6   Summary

The 3D version of the SPH model has proven to be a suitable tool to reproduce phenomena related to wave collision with a structure. In particular, the presence of dikes to mitigate the effect of large waves on coastal structures was considered. Thus, considering the same incoming wave, a dam break, the distance from the dike to the protected structure and the dike height have been shown to be key parameters to control the mitigation process. Some additional parameters, e.g. the barrier width or the initial water height, were not considered in the present study.

Two variables were investigated, the force and the moment exerted on the structure. In addition to force, moment should be considered since the protection barrier can behave as a ramp under certain circumstances.

In general, the force (Fig. 6.18) increases when decreasing the dike height and when increasing the distance dike- structure, this effect is due to the flow reconstruction, which is more effective when increasing the distance.
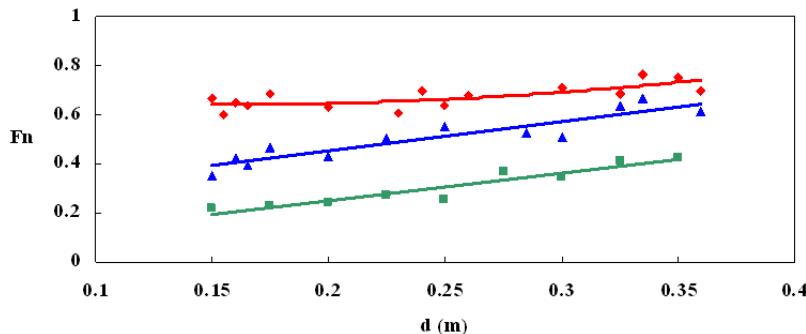
Figure 6.18: Dependence of the normalized force (F_n) for different distances dike-structure and different dike heights normalized H_D; $H_D = 0.05m$ (red line); $H_D = 0.0875m$ (blue line); $H_D = 0.125m$ (green line).

As for the moment (Fig. 6.19), the protection is also more effective when increasing the dike height. Nevertheless, the moment dependence on the distance is far from being linear in most of the cases. This nonlinearity is due to the interaction of the water overtopping the dike and surrounding it. Only for tall dikes, where the overtopping effect is negligible there is a linear increase of moment with distance. Different regimes are observed in the rest of the cases, due to the formation of an overtopping jet when considering short dikes that strikes the structure at a much higher elevation than when there is no dike or to the existence of splashed water when considering dikes of intermediate height.
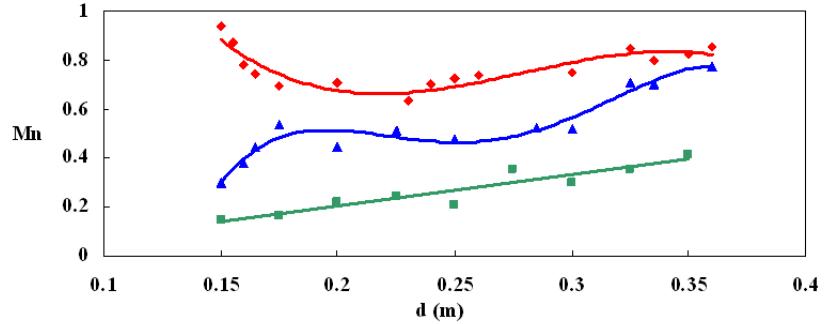
Figure 6.19: Dependence of the normalized moment (M_n) for different distances dike-structure and different dike heights normalized H_D; $H_D = 0.05m$ (red line); $H_D = 0.0875m$ (blue line); $H_D = 0.125m$ (green line).

In summary, the Lagrangian nature of the method permits the study of discontinuities in the flow without constraints due to the presence of a grid. In particular, the formation of a jet over a dike and its further interaction with the fluid surrounding the dike can be treated in a natural way. In addition, the Lagrangian nature of the model approach allows tracing back the origin of water at a certain place inside the medium, which is especially interesting in the area near the protected structure, since it permits identifying the weight of the different propagation modes (dike overtopping or dike surrounding)

6.4 Effect of dike slope and opening

In the previous section, impact mitigation by dikes was evaluated. The distance between the structure and the dike and dike height have been proven to play a key role in the degree of protection ([Crespo et al., 2006]). Now dike slope and the presence of an opening will be analyzed.

6.4.1 Test problem 1: Effect of dike slope

The first situation in order to study is the effect of wall slope on wave mitigation. The sketch of the numerical setup can be observed in Figure 6.20.
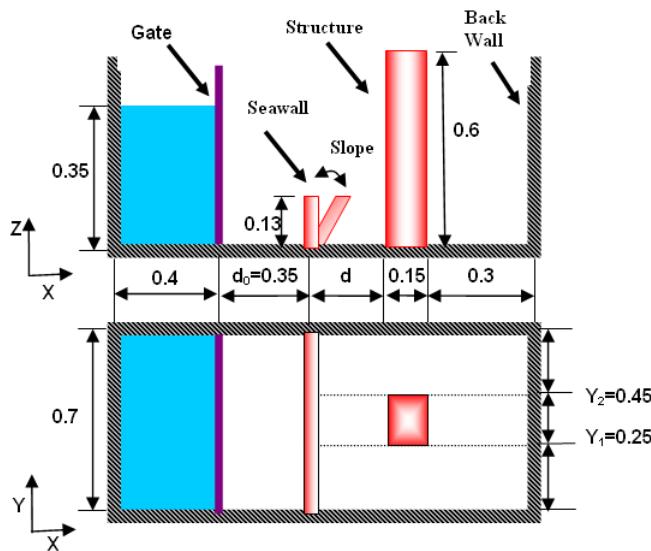
Figure 6.20: Initial configuration of the wave collision with the structure using different sloped walls.

The distance from the gate to the dike was kept constant ($d_0 = 0.35m$) in all simulations, in such a way that the conditioning wave, the wave arriving at the protection barrier, was the same. Wall covers all lateral extent of the tank to avoid flow reconstruction effects around it. Two free parameters were used in this study. The first one is d: different distances from wall to structure (0.25 or $0.35m$). The second parameter is the slope of the seawall. Zero slope corresponds to a vertical wall and positive (negative) slopes to walls tilted landward (seaward).

Data obtained for straight dike and sloped dikes were compared, since the main goal of the study will be to analyze the degree of protection generated by wall. In the Figure 6.21, different instants of the water propagation using the sloped walls are depicted; the water advance to the barrier, the jet formation and the impact between the jet and the structure. The jet formation will be different depending on which slope was used to protect the structure.
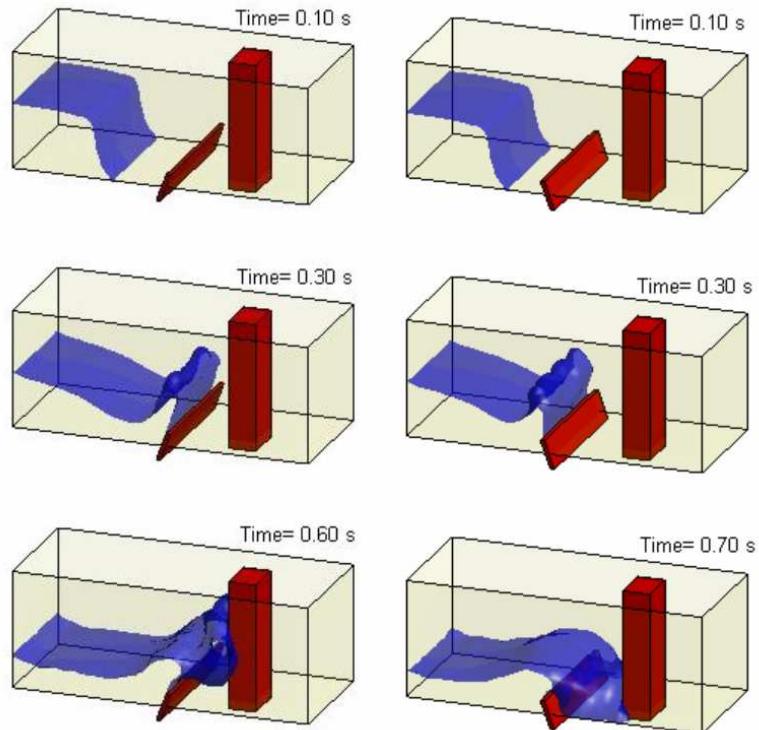
Figure 6.21: Different instants of the wave collision with a structure using a wall sloped landward (left) or seaward (right).

The Figure 6.22a shows how the highest impact points on structure increases (decreases) when the wall is tilted landward (seaward). This occurs for distances comparable to the wall height. It implies that moment increases when walls oriented in the flow direction are used to mitigate the impact and decreases with walls oriented against flow direction. Maximum moment depicted in the Figure 6.22b is normalized using the moment measured with a vertical barrier.
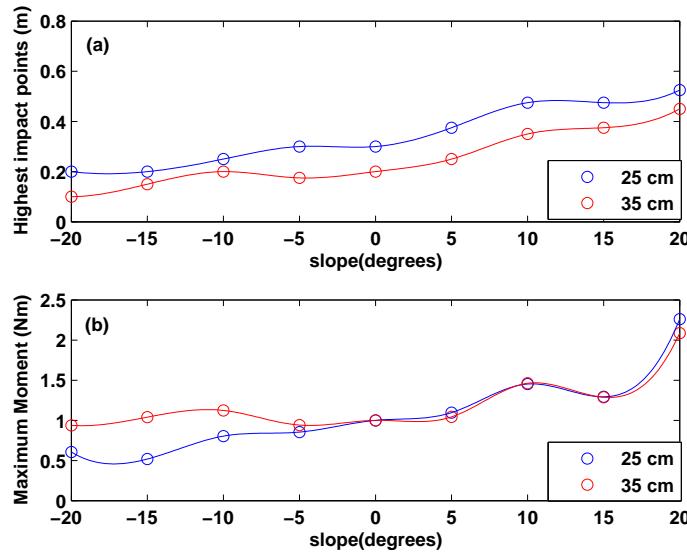
Figure 6.22: Dependence of highest impact points (a) and moment (b) with different wall slopes.

In summary, walls tilted seaward provide a more effective protection on coastal structures.

6.4.2 Test problem 2: Effect of dike opening

The second situation under study is the effect of an opening in the wall of a pedestrian seawalk (Figure 6.23).

The sketch of the numerical setup can be observed in the Figure 6.24. Also, the distance from the gate to the dike was kept constant so the wave arriving at the protection barrier was the same.

Figure 6.23: The picture was taken from [Dalrymple and Kriebel, 2005] and it shows a pedestrian seawall opening correlated with damage. The most important damage appears in front of the opening.
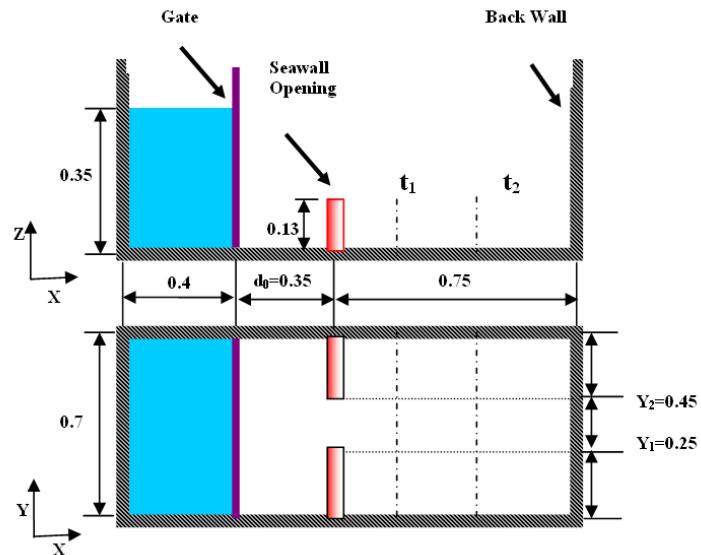
Figure 6.24: Configuration of the experiment corresponding to wave collision with an open wall.

Different instants of the dam evolution are represented in Figure 6.25; the water advance to the wall, the jet formation and the flow reconstruction after the wall. It is observed how using an open wall, water not only surpasses the wall but it also flows through the opening.
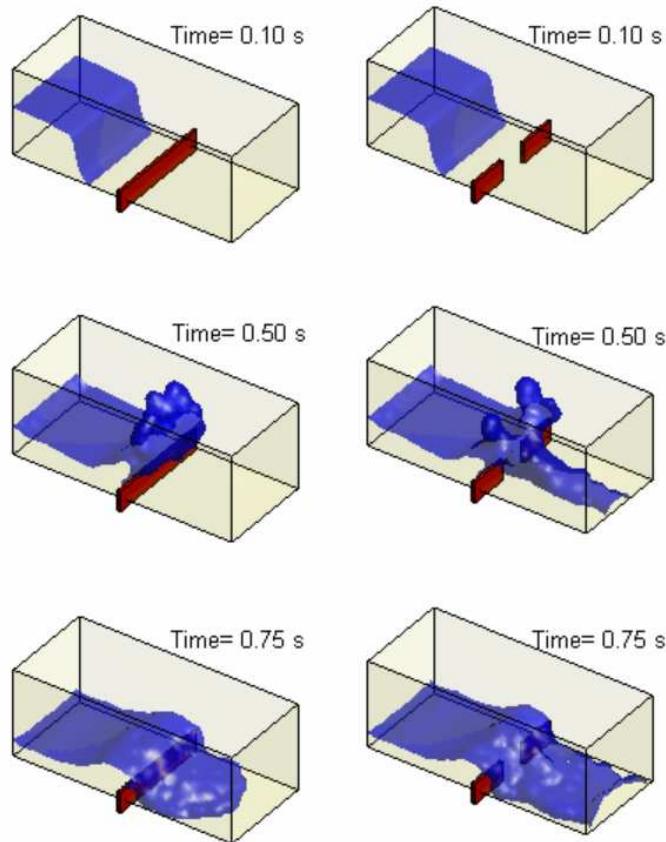
Figure 6.25: Different instants of the wave collision with a continuous wall (left frames) an open wall (right frames).

Dam velocities obtained for continuous and open wall were compared to analyze the degree of damage generated by wall opening. Velocity along lateral extent was measured at different transects in X direction (t_1, t_2 in Figure 6.24). Velocities for the close transect t_1 (located at $0.2m$ from the wall) and for a far transect t_2 (located at $0.5m$ from the wall) are represented in Figure 6.26.
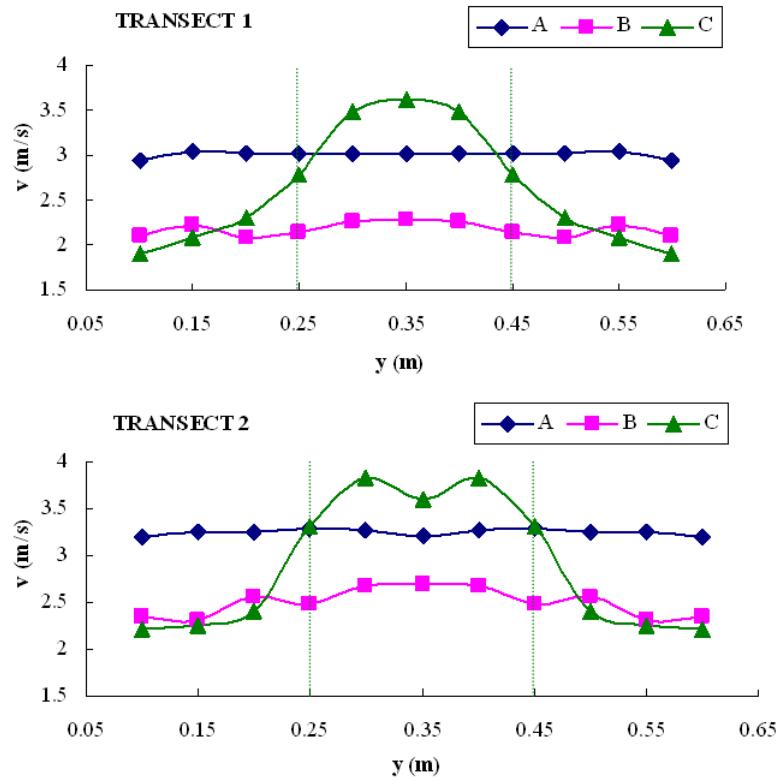
Figure 6.26: Measured velocity along the lateral extent. A: without wall. B: continuous wall. C: open wall.

The presence of the wall provides a considerable velocity decreasing along its extent (compare A and B). Nevertheless, the presence of openings in the wall gives rise to flow velocity variation, in such a way that the observed velocities are even faster than without protection.

6.4.3 Summary

The 3D version of the SPHysics model has proven to be a suitable tool to reproduce phenomena related to wave collision with a structure. In particular, the presence of seawalls to mitigate the effect of large waves on coastal structures was considered.

Impact points and moment are higher (lower) with landward (seaward) slope (Figure 6.22). So walls tilted seaward provided a higher degree of protection. The presence of openings in protection walls tends to accelerate the flow generating velocities higher than measured without protection.

Chapter 7

Concluding Remarks and Future Work

7.1 Conclusions

- A Smoothed Particle Hydrodynamics model called SPHysics has been described. The model has been jointly developed by researchers at the Johns Hopkins University (US), the University of Vigo (Spain), the University of Manchester (UK) and the University of Rome La Sapienza (Italy). The model has been validated in two and three dimensional version using different experimental data and the examples presented here show that SPHysics method is robust enough to simulate a variety of wave problems.

- The 2D version of the SPHysics model has proven to be a suitable tool to analyze green water overtopping. The wave profiles generated by the method are in good qualitative and quantitative agreement with the experimental ones, both in phase and amplitude; and, in addition it has reproduced successfully the main features observed when a wave hits a horizontal platform. Thus, the initial continuous flow, flow separation when hitting the structure and further flow restoration match accurately the ones observed in experiments. In addition, the appearance of a jet close to the deck rear has been analyzed under extreme conditions. The incoming wave amplitude is observed to decrease progressively over the deck and then to increase suddenly after passing the rear of the deck. The relative amplitude of the jet has been shown to decrease when increasing the distance between the free surface at rest and the deck.

- The 2D version of the SPHysics model has proven to be a suitable tool to reproduce a dam break evolution over dry and wet beds. Experimental profiles and horizontal velocities are properly reproduced by the model. The mixing process observed in experiments over wet bed between lock

and tank water is almost negligible. Actually, the basic propagation mechanism is due to the pressure exerted by lock water on tank water. Some mixing is observed in experiments with a shallow water layer, where the successive wave breakings result in eddy generation. Two regimes are defined in dam evolution. Initial propagation (dam release) in horizontal direction is faster than observed for longer distances where velocity is mainly reduced by bottom friction. The difference between both regimes is higher for dry beds and shallow water layers. Energy dissipation was observed to be responsible of both regimes. Energy dissipation for wet beds is higher at the beginning of the experiments, since breaking constitutes the main dissipation mechanism. However, bottom friction becomes in time the main dissipation mechanism, which is especially important on dry beds.

- The 3D version of the SPHysics model has proven to be a suitable tool to reproduce phenomena related to wave collision with a structure. In particular, the presence of dikes to mitigate the effect of large waves on coastal structures was considered. Thus, considering the same incoming wave, a dam break, the distance from the dike to the protected structure and the dike height have been shown to be key parameters to control the mitigation process. Some additional parameters, e.g. the barrier width or the initial water height, were not considered in the present study. Two variables were investigated, the force and the moment exerted on the structure. In addition to force, moment should be considered since the protection barrier can behave as a ramp under certain circumstances. In general, the force increases when decreasing the dike height and when increasing the distance dike- structure, this effect is due to the flow reconstruction, which is more effective when increasing the distance. As for the moment, the protection is also more effective when increasing the dike height. Nevertheless, the moment dependence on the distance is far from being linear in most of the cases. This nonlinearity is due to the interaction of the water overtopping the dike and surrounding it. Only for tall dikes, where the overtopping effect is negligible there is a linear increase of moment with distance. Different regimes are observed in the rest of the cases, due to the formation of an overtopping jet when considering short dikes that strikes the structure at a much higher elevation than when there is no dike or to the existence of splashed water when considering dikes of intermediate height. The Lagrangian nature of the method permits the study of discontinuities in the flow without constraints due to the presence of a grid. In particular, the formation of a jet over a dike and its further interaction with the fluid surrounding the dike can be treated in a natural way. In addition, the Lagrangian nature of the model approach allows tracing back the origin of water at a certain place inside the medium, which is especially interesting in the area near the protected structure, since it permits identifying the weight of the different propagation modes (dike overtopping or dike surrounding). In addition, the presence of seawalls to mitigate the effect

of large waves on coastal structures was considered. Impact points are higher (lower) with landward (seaward) slope. This implies that moment is higher (lower) with landward (seaward) slope. So a wall tilted seaward provided a higher degree of protection. The presence of openings in protection walls tends to accelerate the flow generating velocities higher than measured without protection.

7.2 Future Developments in SPHysics

Multiphase modeling. More work needs to be done with two-phase modeling. [Cuomo and Dalrymple, 2008] study the dynamics of water wave impacts including the compressibility of air-water mixture. Air entrapment and entrainment play a fundamental role in the dynamics of water wave impacts, so the compressibility of air-water mixture must be addressed more in detail using a SPH technique.

Parallel version. Simulating water wave mechanics using SPH requires the use of a parallel code to run the large number of particles needed in 3-D. A first preliminary version has been developed by Dr. Benedict Rogers at the University of Manchester ([Rogers et al., 2007]). The code is parallelized using Message Passing Interface (MPI) formalism, which is the set of functions that allow different processors to talk to each other.

Turbulence and viscosity treatment. Different treatments of viscosity have been described in the text. Closure models similar to the $\kappa - \epsilon$ model developed by [Violeau and Issa, 2006] should be implemented in the future.

More accurate algorithms. New SPH schemes need to be developed to increase the accuracy of the results decreasing computational time. Beeman ([Beeman, 1976]) and Two Step Velocity Verlet ([Monaghan, 2006]) algorithms will be considered.

Kernel renormalization. Kernel corrections are needed to avoid errors from a corrupted interpolating function. Kernel gradient correction was described in section 3.4.1. In the future a new correction will be implemented in SPHysics, the mixed kernel and gradient correction, which is achieved by combining the constant kernel correction with the gradient correction ([Bonet and Lok, 1999]).

Coupling Funwave-SPHysics. One of the important advances to be made for SPH is the development of a hybrid model that couples the computationally intensive, but highly resolved, SPH code to a more efficient, but less descriptive, model in the offshore region. For example Boussinesq models are reasonable efficient at propagating waves in open ocean; however, they can not treat the details of breaking near the coast.

The coupling approach proposed by [Nie et al., 2004] will be used. The development of a hybrid method is in progress starting from two existing wave propagation models. The model couples the finite difference Boussinesq FUNWAVE ([Wei et al., 1995]) to SPHysics model. FUNWAVE is employed for wave propagation in the offshore region and SPHysics to handle wave breaking, run-up and overtopping in the domain close to coastal structures. Hybrid model will be used to provide significantly improved predictions of wave heights, velocities, breaking points and overtopping information.

Floating bodies and breakable structures.

Preprocessing code. In order to generate the initial geometry introducing real bathymetries and complex geometries, a user friendly interface will be developed.

Postprocessing code. The processing to analyze and visualize the results must be improved. Visualization codes (Paraview) and other codes (C++) are going to be developed to show animations with more quality.

Appendix A

SPHysics Code

A.1 Installation

Two versions of SPHysics are available in this release:

- SPHysics_2D. The computational domain is considered to be 2D, where x corresponds to the horizontal direction and z to the vertical direction.

- SPHysics_3D. The computational domain is fully 3D. x and y are the horizontal directions and z the vertical direction.

SPHysics is distributed in a compressed file (gz or zip). The directory tree shown in Figure A.1 can be observed after uncompressing the package.

The following directories can be observed in the figure both in 2D and 3D:

source contains the FORTRAN codes. This directory contains two subdirectories:
- SPHysicsgen: contains the FORTRAN codes to create the initial conditions of the run.
- SPHysics: contains the FORTRAN source codes of SPH.

execs contains all executable codes.

run_directory is the directory created to run the model. The different subdirectories Case1, $\cdots$, CaseN placed in this directory correspond to the different working cases to be created by the user. Input and output files are written in these directories

post_processing this directory contains MATLAB codes to visualize results.
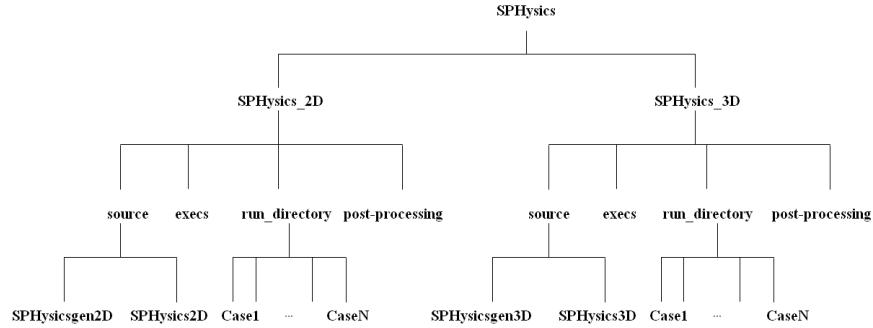
Figure A.1: Directory tree.

A.2   Program outline

Both the 2D and 3D version consist in two programs, which are run separately and in the following order:

2D Code:

- SPHysicsgen_2D: Creates the initial conditions and files for a given case.

- SPHysics_2D: Runs the selected case with the initial conditions created by SPHysicsgen_2D code.

3D Code:

- SPHysicsgen_3D: Creates the initial conditions and files for a given case.

- SPHysics_3D: Runs the selected case with the initial conditions created by SPHysicsgen_3D code.

In general, 2D or 3D appended to the source file name means that the source is suited for 2D or 3D calculations.

In the remainder of this appendix, SPHysicsgen and SPHysics, when used, refer to both the aforementioned 2D and 3D programs for convenience. For example, SPHysicsgen will refer to both SPHysicsgen_2D and SPHysicsgen_3D.

A.2.1   SPHYSICSGEN

All subroutines are included in two source files (SPHysicsgen_2D.f or SPHysicsgen_3D.f), depending on the nature two or three- dimensional of the calculation. Each source uses a different common file, where most of the variables are stored. The common files are common.gen2D (in 2D) and common.gen3D (in 3D). Both

versions (2D and 3D) can be compiled by the user with any FORTRAN compiler and the resulting executable file is placed in subdirectory *execs*.

SPHysicsgen plays a dual role: (i) Creating the MAKEFILE to compile SPHysics; and (ii) Creating the output files that will be the input files to be read by SPHysics. These files contain information about the geometry of the domain, the distribution of particles and the different running options.

Compiling options

The compilation of SPHysics code depends on the nature of the problem under consideration and on the particular features of the run. Thus, the user can choose the options that are better suited to any particular problem and only those options will be included in the executable versions of SPHysics. This protocol speeds up calculations since the model is not forced to make time consuming logical decisions.

Both in 2D and 3D the following compiling options can be considered (See table 1.4):

1. Kind of kernel: 1=Gaussian; 2=Quadratic; 3=Cubic Spline; 5=Quintic.

2. Kernel corrections: 0=no correction; 1=kernel gradient correction.

3. Time stepping: 1=Predictor-Corrector algorithm; 2=Verlet algorithm.

4. Density filter: 0=no filter; 1=Shepard; 2=MLS.

5. Viscosity treatment: 1=Artificial viscosity; 2=Laminar viscosity; 3=Laminar viscosity +SPS.

6. Equation of state: 1=Weakly Compressible Fluid (Tait equation); 2=Ideal Gas Equation; 3=Incompressible Fluid (Poisson equation).

7. Boundary conditions: 1=Repulsive BC; 2=Dynamic BC.

8. Choice of compilers: 1=gfortran; 2=ifort; 3=CVF.

Output files

As we mentioned above, different output files are created by SPHysicsgen. These files can be used either by the SPHysics executable as input files or by MATLAB codes to visualize results (different MATLAB codes are provided in *post_processing* subdirectory.

<u>SPHysics.mak</u>

Compiling file created by the executable SPHysicsgen. It depends on the running options defined by input file. It was prepared for COMPAQ VISUAL

FORTRAN, IFORT and GFORTRAN although it can be adapted to other compilers.

INDAT

Created by SPHysicsgen. Read by SPHysics code at GETDATA. The file contains the following variables:

i_kernel : Kind of kernel: 1=Gaussian; 2=Quadratic; 3=Cubic Spline; 5=Quintic.

i_algorithm : Kind of algorithm: 1=Predictor- Corrector algorithm; 2=Verlet algorithm.

i_densityFilter : Use of a density filter: 0=no filter; 1=Shepard; 2=MLS.

i_viscos : Viscosity definition: 1=Artificial; 2=Laminar; 3=Laminar + SPS.

IBC : Boundary conditions: 1=Monaghan repulsive forces; 2=DBPs.

i_periodicOBs(1) : Periodic Lateral boundaries in x direction? (1=yes)

i_periodicOBs(2) : Periodic Lateral boundaries in y direction? (1=yes)

i_periodicOBs(3) : Periodic Lateral boundaries in z direction? (1=yes)

lattice : Lattice: (1) SC; (2) BCC

i_EoS : Equation of State: (1) Tait equation; (2) Ideal Gas; (3) Poisson equation.

h_SWL : Still water level (m).

B : Parameter in Equation of State.

gamma : Parameter in Equation of State (Default value 7).

Coef : Coeffficient to calculate the smoothing length (h) in terms of dx,dy,dz, since $h = coefficient \cdot sqrt(dx \cdot dx + dy \cdot dy + dz \cdot dz)$

eps : Epsilon parameter in XSPH approach (Default value 0.5).

rho0 : Reference density (Default value $1000 kg \cdot m^{-3}$).

viscos_val : Viscosity parameter, it corresponds to α if $i\_viscos = 1$ and to ν (kinematical viscosity) if $i\_viscos = 2$ or 3.

visc_wall : Wall viscosity value for Repulsive Force BC.

vlx : medium extent in X direction.

vly : medium extent in Y direction. It is set to zero when $IDIM = 2$.

vlz : medium extent in Z direction.

dx : Initial interparticle spacing in x direction.

dy : Initial interparticle spacing in y direction.

dz : Initial interparticle spacing in z direction.

h : Smoothing length.

np : Total number of particles.

nb : Number of boundary particles.

nbf : Number of fixed boundary particles. Note that boundary particles can be fixed or move according to some external dependence (e.g. gates, wavemakers).

ivar_dt : Variable time step calculated when $ivar\_dt = 1$.

dt : Initial time step. It is kept throughout the run when $ivar\_dt = 0$.

tmax : RUN duration (in seconds)

out : Recording time step (in seconds). The position, velocity, density, pressure and mass of every particle is recorded in PART file every out seconds.

trec_ini : Initial recording time.

dtrec_det : Detailed recording step.

t_sta_det : Start time in detailed recording.

t_end_det : End Time in detailed recording.

i_restartRun : (0) Start a new RUN; (1) Restart an old RUN.

IPART

Created by SPHysicsgen. Read by SPHysics code at GETDATA. The file contains the following variables recorded at time=0:

xp(i) Position in x direction of particle i.

yp(i) Position in y direction of particle i.

zp(i) Position in z direction of particle i.

up(i) Velocity in x direction of particle i.

vp(i) Velocity in y direction of particle i.

wp(i) Velocity in z direction of particle i.

rhop(i) Density of particle i.

p(i) Pressure at particle i.

pm(i) Mass of particle i.

vortx(i),vorty(i),vortz(i) correspond to vorticity in x, y and z constant planes

MATLABIN

Created by SPHysicsgen. To be used by MATLAB codes for graphical representation. The file contains the following variables:

np : Total number of particles.

vlx : medium extent in X direction.

vly : medium extent in Y direction. It is set to zero when $IDIM = 2$.

vlz : medium extent in Z direction.

out : Recording time step (in seconds). The position, velocity, density, pressure and mass of every particle is recorded in PART file every out seconds.

nb : Number of boundary particles.

NORMALS

Created by SPHysicsgen. To be used by SPHysics code when $IBC = 1$. It contents the normal and tangent vectors to each boundary particle. The file contains the following variables:

xnb(i), ynb(i), znb(i) : Components of the unitary vector normal to the boundary at point i..

xtb(i), ytb(i), ztb(i) : Components of the unitary vector tangential to the boundary at that point.

xsb(i), ysb(i), zsb(i) : Components of the unitary vector tangential to the boundary at point i and perpendicular to the previous one.

iBP_Pointer_Info(i,1) : Absolute index BP

iBP_Pointer_Info(i,2) : Rank of BP (default=0, reserved for MPI)

iBP_Pointer_Info(i,3) : Absolute index of $i - 1$ neighbour BP

iBP_Pointer_Info(i,4) : Absolute index of $i + 1$ neighbour BP

iBP_Pointer_Info(i,5) : Absolute index of $j - 1$ neighbour BP

iBP_Pointer_Info(i,6) : Absolute index of $j + 1$ neighbour BP

BP_xyz_Data(i,1), BP_xyz_Data(i,2), BP_xyz_Data(i,3) : xp(BP), yp(BP), zp(BP) needed for the future release of a MPI version of the code.

OBSTACLE

Created by SPHysicsgen. To be used by MATLAB codes for graphical representation. The file contains the following variables:

iopt_obst : Conditional variable (1=obstacle exists; 0=it does not exist). The last one is always zero.

XXmin : Minimum value of the obstacle in x direction.

XXmax : Minimum value of the obstacle in x direction.

YYmin : Minimum value of the obstacle in y direction.

YYmax : Minimum value of the obstacle in y direction.

ZZmin : Minimum value of the obstacle in z direction.

ZZmax : Minimum value of the obstacle in z direction.

Slope : Obstacle slope in x direction.

WAVEMAKER

Created by SPHysicsgen. To be used by SPHysics code. Parameters fix the wavemaker extent and movement. It will only move in x direction. The file contains the following variables:

iopt_wavemaker : Conditional variable (1=Wavemaker exists; 0=it does nor exist).

i_paddleType : Enter Paddle-Type (1=Piston, 2=Piston-flap)

nwavemaker_ini : First wavemaker particle.

nwavemaker_end : Last wavemaker particle.

X_PaddleCentre : Wavemaker Centre position in X coordinates

X_PaddleStart : X_PaddleStart $=0.5 \cdot stroke$

paddle_SWL : Enter paddle Still Water Level (SWL)

flap_length : Enter piston-flap flap_length

stroke : Wavemaker Stroke $=2 \cdot Amplitude$

twavemaker : Initial time of wavemaker

Nfreq : Number of frequencies

A_wavemaker : Amplitude of wavemaker movement.

Period : Period of wavemaker movement.

phase(n) : Phase of wavemaker movement.

twinitial(n) : Start of wavemaker movement (seconds).

    <u>GATE</u>

Created by SPHysicsgen. To be used by SPHysics code. Parameters fix the gate extent and movement. The file contains the following variables:

iopt_gate : Conditional variable (1=gate exists; 0=it does nor exist).

ngate_ini : First gate particle

ngate_end : Last gate particle

VXgate,VYgate,VZgate : Gate velocity in coordinates

tgate : Start of gate movement (seconds).

Subroutines

All subroutines in SPHysicsgen are inside a single source file SPHysicsgen_2D.f or SPHysicsgen_3D.f

SPHysicsgen Main program.

Depending on the subroutine, different container geometries can be used.

BOX Subroutine to build a box in 2D or 3D.

BEACH Subroutine to build a beach in 2D or 3D. The beach consists in a flat area followed by a tilted region. The tilted area always has a slope in x-direction and a possible slope in y- direction.

Each subroutine calls new subroutines to generate the walls of the container and the different obstacles placed inside it.

BOUNDARIES_LEFT Subroutine to generate the left boundary of the container both in 2D and 3D.

BOUNDARIES_RIGHT Subroutine to generate the right boundary of the container both in 2D and 3D.

BOUNDARIES_BOTTOM Subroutine to generate the bottom boundary of the container both in 2D and 3D.

BOUNDARIES_FRONT Subroutine to generate the front of the container in 3D.

BOUNDARIES_BACK Subroutine to generate the back of the container in 3D.

WALL Subroutine to generate a wall with an arbitrary slope in x- direction inside the container.

WALL_HOLE Subroutine to generate a wall with a round shaped hole inside the container (Only in 3D version).

WALL_SLOT Subroutine to generate a wall with a slot inside the container (Only in 3D version).

OBSTACLE Subroutine to generate an obstacle inside the container.

WAVEMAKER Subroutine to generate a piston that can move in x- direction.

GATE Subroutine to generate gate that can move in any direction.

EXTERNAL_GEOMETRY This subroutine, which only works in 2D, reads the container and the initial fluid distribution from a file previously generated. The MATLAB software to generate the pre-processing will be provided in next release.

Apart from previous subroutines, which control the shape and dimensions of the container, other subroutines are responsible of the fluid properties inside that container.

FLUID_PARTICLES Subroutine to choose between different initial distributions of the fluid.

DROP Subroutine used to generate a round shaped area (2D or 3D) as initial position. The velocity of the particles inside the region can be fixed by the user (all particles share the same velocity).

SET Subroutine used to generate a set of particle as initial condition. The number of particle and the initial position and velocity of each particle can be decided by the user. This configuration is particularly useful when checking changes in the code since it permit runs with a small number of moving particles.

FILL PART Subroutine used to generate a cubic area as initial position (2D or 3D). Different cubes can placed at different position inside the computational domain.

WAVE Subroutine used to generate a wave (2D or 3D) advancing in x- direction as initial position.

POS_VELOC Subroutine used to determine the initial position and velocity of particles.

PRESSURE Subroutine used to determine the initial pressure of particles.

P_BOUNDARIES Subroutine to assign density equal to the reference density to the boundary particles and gage pressure equal to zero.

CORRECT_P_BOUNDARIES Subroutine to correct pressure at boundaries. It considers the density to be equal to the reference density plus a hydrostatic correction. Pressure is then calculated according to Batchelor equation.

PERIODICITYCHECK Subroutine to determine the limits in periodic boundary conditions. These BC are only available in 3D and in y- direction.

NORMALS_CALC_2D/_3D Subroutines to calculate the normals to be used in repulsive boundary conditions.

NORMALS_FILEWRITE_2D/_3D Subroutines to write the normals to be used in repulsive boundary conditions.

TOCOMPILE_IFORT Subroutine to create the MAKEFILE, SPHysics.mak, used to compile SPHysics using a IFORT compiler. The source files to be included in SPHysics.mak depend on the particular conditions of the run fixed by the input files.

TOCOMPILE_GFORTRAN Subroutine to create the MAKEFILE, used to compile SPHysics using a GFORTRAN compiler. The source files to be included in the MAKEFILE depend on the particular conditions of the run fixed by the input files.

TOCOMPILE_CVF Subroutine to create the MAKEFILE necessary to compile SPHysics using a Compaq Visual Fortran compiler. The source files to be included in the MAKEFILE depend on the particular conditions of the run fixed by the input files.

A.2.2   SPHYSICS

SPHysics nature depends on the compiling option determined by SPHysicsgen

Input files

The input files correspond to the output files generated by SPHysicsgen.

Output files

PART_klmn

Created by SPHysics at POUTE_2D.f or POUTE_3D.f with a periodicity in seconds fixed by the input file used to run SPHysicsgen. The structure of

PART_klmn is the same as that of IPART previously described. The indices k, m, n and l can take any integer value from 0 to 9, in such a way that the maximum number of images is 9999. Each PART_klmn file is opened, recorded and closed in each call to POUTE_3D.f or POUTE_2D.f subroutines.

SCAL

Created by SPHysics at POUTE_2D/_3D.f with the same periodicity as PART_klmn. The following variables are recorded:

itime : Number of iterations since the beginning of the run.

time : Time instant (in seconds).

np : Total number of particles.

nb : Number of boundary particles.

nbf : Number of fixed boundary particles. Note that boundary particles can be fixed or move according to some external dependence (e.g. gates, wavemakers).

h : Smoothing length.

DT

Created by SPHysics at POUTE_2D/_3D.f. The following variables are recorded:

time : Time instant (in seconds)

dt1 : Time step based on the force per unit mass.

dt2 : Time step combining the Courant and the viscous conditions.

dtnew : Time step corresponding to next step using $dt1$ and $dt2$.

DETPART_klmn

Created by SPHysics at POUTE_2D/_3D.f. The same as PART_klmn but with a shorter periodicity during a certain interval of the run.

EPART

Created by SPHysics at POUTE_2D/_3D.f at the end of the run. This file contains the same information as IPART but corresponding to the end of the run.

ESCAL

Created by SPHysics at POUTE_2D/_3D.f at the end of the run. It contains the same information as SCAL.

ENERGY

Created by SPHysics at ENERGY_2D/_3D.f. The file contains the following variables recorded with the same periodicity as PART_kmnl.

time : Time instant (in seconds)

Eki_p : Kinetical energy summation (for fluid particles)

Epo_p : Potential energy summation (for fluid particles)

TE_p : Thermal energy summation (for fluid particles)

Eki_b : Kinetical energy summation (for boundary particles)

Epo_b : Potential energy summation (for boundary particles)

TE_b : Thermal energy summation (for boundary particles)

NOTE: Boundary particle energies only make sense when using Dynamic Boundary Conditions.

RESTART

Created by SPHysics at SPHYSICS_2D/_3D.f. The following variables are recorded:

itime : Number of iterations since the beginning of the run.

time : Time instant (in seconds).

ngrab : Recording instant.

NOTE: This option is especially useful after a power cut or when the estimated running time was underestimated and an additional computational time should be considered.

Subroutines

All subroutines in SPHysicsgen are placed in the same source file, however SPHysics ones are placed in different source files. A short description of each possible subroutine follows.

SPHysics Main program containing the main loop.

GETDATA Subroutine called from SPHysics at the beginning of the run. It provides data about the run (scales, kernel parameters, steps, use of gates and/or wavemakers).

ENERGY Subroutine called from SPH to record information about energy (kinematical, potential and thermal). This subroutine is called at the beginning and end of the run and also every out seconds (variable provided by INDAT file). It creates the file ENERGY described in previous section.

INI_DIVIDE Subroutine called from SPH at the beginning of the run (just for fixed boundary particles) and from subroutine STEP during the run (every time step for moving objects and fluid particles). It initializes the link list.

DIVIDE Subroutine called from SPHysics at the beginning of the run and from subroutine STEP during the run (every time step). The first time (when called from SPHysics) creates the link list corresponding to the fixed boundary particles. The rest of the calls the subroutine allocates the fluid particles and the moving boundary particles into the link list.

KEEP_LIST Subroutine called from SPHysics at the beginning of the run just after calling DIVIDE. It keeps the list of fixed boundary particles, which is never recalculated again.

CHECK_LIMITS Subroutine called from SPHysics every time step. The subroutines detect the position of particles outside the computational domain and relocate them (see section 3.10).

POUTE Subroutine called from SPHysics to record information about particles (position, velocity, density, pressure and mass). This subroutine is called at the beginning and end of the run and also every out seconds. It creates the SCAL, PART, ESCAL and EPART files previously described.

STEP Subroutine called from SPHysics. It basically manages the marching procedure, depending on the computational algorithm (Predictor- Corrector or Verlet) (see section 3.5).

CORRECT This subroutine is called by STEP every time step. It basically accounts for the body forces and XSPH correction (and SPS terms are calculated if $i\_visos = 3$).

RECOVER_LIST This subroutine is called from STEP every time step. It recovers the list corresponding to the fixed boundary particles created by KEEP_LIST.

VARIABLE_TIME_STEP This subroutine is called from STEP every time step. It calculates the time step considering maximum inter-particle forces, the speed of sound and the viscosity (see section 3.6).

AC This subroutine is called from STEP every time step. It controls the boundary particles movement (gates and wavemakers) and calls the subroutines SELF and CELIJ.

SELF This subroutine is called from AC every time step. It controls the interaction between particles inside the same "cell" determined by the link list.

CELIJ This subroutine is called from AC every time step. It controls the interaction between particles inside adjacent "cells" determined by the link list.

KERNEL This subroutine is called from SELF and CELIJ every time step. It calculates the particle-particle interaction according to kernel definition (1=gaussian, 2=quadratic; 3=cubic; 5=wendland) and dimensionality of the problem (2D or 3D).

VISCOSITY This subroutine is called from SELF and CELIJ every time step. It calculates viscosity terms depending on the chosen option ((1) Artificial (2) Laminar (3) Laminar +SPS) and dimensionality of the problem (2D or 3D).

MONAGHANBC This subroutine is called from CELIJ and SELF. It accounts for Monaghans repulsive force between fluid and boundary particles.

MOVINGOBJECTS This subroutine is called from STEP.

MOVINGPADDLE This subroutine is called from MOVINGOBJECTS.

MOVINGWEDGE This subroutine is called from MOVINGOBJECTS.

DENSITYFILTER_SHEPARD Subroutine called from SPHysics every 30 time steps. It uses a Shepard filter when selected in initial conditions.

DENSITYFILTER_MLS Subroutine called from SPHysics every 30 time steps. It uses a MLS filter when selected in initial conditions.

AC_SHEPARD This subroutine is called from DENSITYFILTER_MLS. It calls the subroutines PRE_SELF and PRE_CELIJ.

PRE_SELF This subroutine is called from AC_MLS.

PRE_CELIJ This subroutine is called from AC_MLS.

LU_DECOMPOSITION It is called from DENSITYFILTER_MLS. It constructs the LU-decomposition matrix

EOS_IDEALGAS (It uses the equation of Ideal Gases to solve the pressure.

EOS_POISSON It uses the equation of Poisson to solve the pressure.

EOS_TAIT It uses the equation of Tait to solve the pressure.

List of Figures

145

List of Tables

Bibliography

[Arnason, 2005] Arnason, H. (2005). *Interaction between an Incident Bore and a Free-Standing Coastal Structure.* PhD thesis, University of Washington.

[Baarholm, 2001] Baarholm, R. J. (2001). *Theoretical and Experimental Studies of Wave Impact underneath Decks of Offshore Platforms.*

[Batchelor, 1974] Batchelor, G. K. (1974). *Introduction to fluid dynamics.* Cambridge University Press.

[Beeman, 1976] Beeman, D. (1976). Some multistep methods for use in molecular dynamics calculations. *Journal of Computational Physics*, 20:130–139.

[Benz, 1988] Benz, W. (1988). Applications of Smoothed Particle Hydrodynamics (SPH) to Astrophysical Problems. *Computer Physics Communications*, 48:130–139.

[Benz, 1989] Benz, W. (1989). Smooth Particle Hydrodynamics: A review. Numerical Modeling of Stellar Pulsation: Problems and Prospects. In *Proceedings of NATO Workshop, Les Arcs, France.*

[Benz, 1990] Benz, W. (1990). *Smoothed Particle Hydrodynamics: a review.* Numerical Modelling of Nonlinear Stellar Pulsations: Problems and Prospects, Kluwer Acad. Boston Publ., pages 269-288.

[Benz and Asphaug, 1993] Benz, W. and Asphaug, E. (1993). Explicit 3D continuum fracture modeling with Smoothed Particle Hydrodynamics. In *Proceedings of Twenty- fourth Lunar and Planetary Science Conference*, pages 99–100.

[Benz and Asphaug, 1994] Benz, W. and Asphaug, E. (1994). Impact simulations with fracture. *I. Methods and tests. Icarius*, 107:98–116.

[Benz and Asphaug, 1995] Benz, W. and Asphaug, E. (1995). Simulations of brittle solids using Smoothed Particle Hydrodynamics. *Computational Physics Communications*, 87:253–265.

[Berczik, 2000] Berczik, P. (2000). Modeling the star formation in galaxies using the chemodynamical SPH code. *Astronomy and Astrophysics.*, 360:76–84.

[Berczik and Kolesnik, 1993] Berczik, P. and Kolesnik, I. G. (1993). Smoothed particle hydrodynamics and its applications to astrophysical problems. *Kinematics and Physics of Celestial Bodies*, 9:1–11.

[Berczik and Kolesnik, 1998] Berczik, P. and Kolesnik, I. G. (1998). Gas dynamical model for the triaxial protogalaxy collapse. *Astronomy and Astrophysical transactions*, 16:163–185.

[Bonet and Kulasegaram, 2000] Bonet, J. and Kulasegaram, S. (2000). Corrections and stabilization of Smooth Particle Hydrodynamics methods with applications in metal forming simulations. *International Journal for Numerical Methods in Engineering*, 47:1189–1214.

[Bonet and Lok, 1999] Bonet, J. and Lok, T.-S. L. (1999). Variational and momentum preservation aspects of Smoothed Particle Hydrodynamics formulations. *Computat. Methods Appl. Mech. Engineering.*, 180:97–115.

[Buchner, 1996a] Buchner, B. (1996a). The impact of green water on FPSO design. In *Proceedings of the Offshore Technology Conference*, pages 45–57.

[Buchner, 1996b] Buchner, B. (1996b). The influence of bow shape of FPSOs on drift forces and green water. In *Proceedings of the Offshore Technology Conference*, pages 389–400.

[Buchner and Cozijn, 1997] Buchner, B. and Cozijn, J. L. (1997). An investigation into the numerical simulation of green water. *MARIN, February 1997.*

[Buchner and van Ballegoyen, 1997a] Buchner, B. and van Ballegoyen, G. (1997a). Joint Industry Project: FPSO Green Water Loading. *MARIN, May 1997.*, A2: Technical Report.

[Buchner and van Ballegoyen, 1997b] Buchner, B. and van Ballegoyen, G. (1997b). Joint Industry Project: FPSO Green Water Loading. *MARIN, May 1997.*, A3: Scale effect tests.

[Buchner and van Ballegoyen, 1997c] Buchner, B. and van Ballegoyen, G. (1997c). Joint Industry Project: FPSO Green Water Loading. *MARIN, May 1997*, C9.

[Buchner and van Ballegoyen, 1997d] Buchner, B. and van Ballegoyen, G. (1997d). Joint Industry Project: FPSO Green Water Loading. *MARIN, December 1997.*, A1: Discussion Report.

[Chanson et al., 2003] Chanson, H., Aoki, S., and Maruyama, M. (2003). An experimental study of tsunami runup on dry and wet horizontal coastlines. *Science of Tsunami Hazards*, 20(5):278–293.

[Chen et al., 1999] Chen, J. K., Beraun, J. E., and Carney, T. C. (1999). A corrective Smoothed Particle Method for boundary value problems in heat conduction. *Computer Methods in Applied Mechanics and Engineering*, 46:231–252.

[Chen et al., 1997] Chen, S., Johnson, D. B., Raad, P. E., and Fadda, D. (1997). The surface marker and microcell method. *Int. J. Numer. Methods Fluids*, 25:749–778.

[Christensen, 2006] Christensen, E. D. (2006). Large eddy simulation of spilling and plunging breakers. *Coastal Engineering*, 53:463–485.

[Colagrossi and Landrini, 2003] Colagrossi, A. and Landrini, M. (2003). Numerical simulation of interfacial flows by Smoothed Particle Hydrodynamics. *Journal of Computational Physics*, 191:448–475.

[Cox and Ortega, 2002] Cox, D. T. and Ortega, J. A. (2002). Laboratory observations of green water overtopping a fixed deck. *Ocean Engineering*, 29:1827–1840.

[Crespo et al., 2006] Crespo, A. J. C., Gómez-Gesteira, M., and Dalrymple, R. A. (2006). Use Of SPH3D Method To Analyze The Effect Of Dike Slope And Opening On Tsunami Mitigation. In *Proc. 1st International Conference on the Application of Physical Modelling to Port and Coastal Protection COASTLAB06*, pages 331–339.

[Crespo et al., 2007a] Crespo, A. J. C., Gómez-Gesteira, M., and Dalrymple, R. A. (2007a). 3D SPH Simulation of large waves mitigation with a dike. *Journal of Hydraulic Research*, 45,(5):631–642.

[Crespo et al., 2007b] Crespo, A. J. C., Gómez-Gesteira, M., and Dalrymple, R. A. (2007b). Boundary Conditions Generated by Dynamic Particles in SPH Methods. *CMC: Computers, Materials, Continua*, 5(3):173–184.

[Crespo et al., 2007c] Crespo, A. J. C., Gómez-Gesteira, M., and Dalrymple, R. A. (2007c). Modeling dam break behavior over a wet bed by an SPH technique. *J. Waterway, Port, Coastal and Ocean Engineering, IN PRESS.*

[Crespo et al., 2008] Crespo, A. J. C., Gómez-Gesteira, M., and Dalrymple, R. A. (2008). Hybridation of generation propagation models and SPH model to study extreme wave events in Galician Coast. *Journal of Marine Systems*, 72(1-4):135–144.

[Cuomo and Dalrymple, 2008] Cuomo, G. and Dalrymple, R. A. (2008). A compressible two-phase SPH-LES model for interfacial flows with fragmentation. *Submitted to Coastal Engineering.*

[Cuomo et al., 2006] Cuomo, G., Panizzo, A., and Dalrymple, R. A. (2006). SPH-LES two phase simulation of wave breaking and wave-structure interaction. In *Proc. 30th International Conference on Coastal Engineering*, pages 274–286.

[Dalrymple and Knio, 2001] Dalrymple, R. A. and Knio, O. (2001). SPH Modelling of Water Waves. *Proc. Coastal Dynamics, Lund*, pages 779–787.

[Dalrymple and Kriebel, 2005] Dalrymple, R. A. and Kriebel, D. L. (2005). Lessons in Engineering from the Tsunami in Thailand. *The Bridge, National Academy of Engineering*, 35(2).

[Dalrymple and Rogers, 2006] Dalrymple, R. A. and Rogers, B. (2006). Numerical modeling of water waves with the SPH method. *Coastal Engineering*, 53:141–147.

[Del Guzzo and Panizzo, 2007] Del Guzzo, A. and Panizzo, A. (2007). Application of SPH-NSWE to simulate landslide generated waves. In *SPHERIC, Second International Workshop*, pages 161–164.

[Dilts, 1999] Dilts, G. A. (1999). Moving-LeastSquares-Particle Hydrodynamics I. Consistency and stability. *Int. J. Numer. Meth. Engineering.*, 44:1115–1155.

[Durisen et al., 1986] Durisen, R. H., Gingold, R. A., and Boss, A. P. (1986). Dynamic Fission Instabilities in Rapidly Rotating n=3/2 Polytropes: A Comparison of Results from Finite-difference and Smoothed Particle Hydrodynamics Codes. *Astron. J.*, 305:281–308.

[Ellero et al., 2007] Ellero, M., Serrano, M., and Espanol, P. (2007). Incomprensible Smoothed Particle Hydrodynamics. *Journal of Computational Physics*, 226:1731–1752.

[Evrad, 1988] Evrad, A. E. (1988). Beyond N-body: 3D cosmological gas dynamics. *Mon. Not. R. Astr.*, 235:911–934.

[Faber and Rasio, 2000] Faber, J. and Rasio, F. A. (2000). Post Newtonian SPH Calculations of Binary Neutron Stars Coalescence. Method and First Results. *Physical Review D*, 62:1–23.

[Faber and Manor, 2001] Faber, J. A. and Manor, J. B. (2001). Post Newtonian SPH Calculations of Binary Neutron Star Coalescence. II. Mass- ratio, equation of state and spin. *Physical Review D*, 63:1–16.

[Faltinsen and Greco, 2001] Faltinsen, O. and Greco, M.and Landrini, M. (2001). *Green water loading on a FPSO.*

[Frederic and James, 1999] Frederic, A. R. and James, C. L. (1999). Smoothed Particle Hydrodynamics calculations of stellar interactions. *Journal of Computational and Applied Mathematics*, 109:213–230.

[Gingold and Monaghan, 1977] Gingold, R. A. and Monaghan, J. J. (1977). Smoothed Particle Hydrodynamics: theory and application to non-spherical stars. *Mon. Not. R. Astr. Soc.*, 181:375–389.

[Gómez-Gesteira et al., 2005] Gómez-Gesteira, M., Cerqueiro, D., Crespo, C., and Dalrymple, R. A. (2005). Green water overtopping analyzed with a SPH model. *Ocean Engineering*, 32:223–238.

[Gómez-Gesteira and Dalrymple, 2004] Gómez-Gesteira, M. and Dalrymple, R. A. (2004). Using a 3D SPH Method for Wave Impact on a Tall Structure. *J. Waterway, Port, Coastal and Ocean Engineering.*, 130(2):63–69.

[Gotoh et al., 2001] Gotoh, H., Shibihara, T., and Hayashi, M. (2001). Subparticle-scale model for the mps method-lagrangian flow model for hydraulic engineering. *Computational Fluid Dynamics Journal*, 9(4):339–347.

[Greco, 2001] Greco, M. (2001). *A Two-Dimensional Study of Green-Water Loading.*

[Habe, 1989] Habe, A. (1989). *Status Rep. Super Computing Japan.* Ed. T. Nakamura, M. Nagasawa. National Lab. High Energy Phys.

[Health and Executive, 2001] Health and Executive, S. (2001). *Analysis of green water susceptibility of FPSO/FSUs on UKCS.* HSE Books, Sudbury.

[Henderson, 1966] Henderson, F. M. (1966). *Open channel flow.* Macmillan Company, New York, USA.

[Herant and Benz, 1991] Herant, M. and Benz, W. (1991). Hydrodynamical instabilities and mixing in SN 1987A - Two-dimensional simulations of the first 3 months. *Astrophysical Journal.*, 370:81–84.

[Hsu et al., 2001] Hsu, T.-J., Sakakiyama, T., and Liu, P. L.-F. (2001). Numerical modeling of tsunami wave forces and overtopping on coastal structures. In *Proceedings ITS 2001.*

[Hu and Adams, 2007] Hu, X. and Adams, N. (2007). An incompressible multiphase SPH method. *Journal of Computational Physics*, 227:264–278.

[Janosi et al., 2004] Janosi, I. M., Jan, D., Szabo, K. G., and Tel, T. (2004). Turbulent drag reduction in dam-break flows. *Experiments in Fluids*, 37:219–229.

[Johnson et al., 1996] Johnson, G. R., Stryk, R. A., and R., B. S. (1996). SPH for high velocity impact computations. *Comput. Methods Appl. Mech. Engineering.*, 139:347–373.

[Klemp et al., 1997] Klemp, J. B., Rotunno, R., and Skamarock, W. C. (1997). On the propagation of internal bores. *J. Fluid Mech.*, 331:81–106.

[Koshizuka and Oka, 1996] Koshizuka, S. and Oka, Y. (1996). Moving-particle semi-implicit method for fragmentation of compressible fuid. *Nuclear Science Engineering.*, 123:421–434.

[Leal et al., 2006] Leal, J. G. A. B., Ferreira, R. M. L., and Cardoso, A. H. (2006). Dam-Break Wave-Front Celerity. *Journal of Hydraulic Engineering*, 132(1):69–76.

[Lee et al., 2006] Lee, E.-S., Violeau, D., Benoit, M., Issa, R., Laurence, D., and Stansby, P. (2006). Prediction of wave overtopping on coastal structures by using extended Boussinesq and SPH models. In *Proc. 30th International Conference on Coastal Engineering*, pages 4727–4740.

[Leveque, 2002] Leveque, R. (2002). *Finite Volume Methods for Hyperbolic Problems.* Cambridge University Press.

[Li and Liu, 1996] Li, S. and Liu, W. (1996). Moving least square Kernel Galerkin method (II) Fourier analysis. *Comput. Methods Appl. Mech. Engineering.*, 139:159.

[Libersky and Petscheck, 1991] Libersky, L. D. and Petscheck, A. G. (1991). Smoothed Particle Hydrodynamics with strength oif materials. In Trease, H, F. J. and Crowley, W. S.-V., editors, *Proceedings of the Next Free Lagrange Conference*, volume 395, pages 248–257.

[Libersky and Petscheck, 1993] Libersky, L. D. and Petscheck, A. G. (1993). High strain Lagrangian hydrodynamics- a three- dimensional SPH code for dynamic material response. *J. Comput. Phys.*, 109:67–75.

[Liu, 2003] Liu, G. R. (2003). *Mesh Free methods: Moving beyond the finite element method.* CRC Press.

[Liu and Liu, 2003] Liu, G. R. and Liu, M. B. (2003). *Smoothed Particle Hydrodynamics: a meshfree particle method.* World Scientific.

[Liu and Quek, 2003] Liu, G. R. and Quek (2003). *The finite element method: a practical course.* Butterworth Herinemann.

[Liu et al., 2005] Liu, P., Wu, T. R., Raichlen, F.and Synolakis, C. E., and Borrero, J. C. (2005). Runup and rundown generated by three-dimensional sliding masses. *J. Fluid Mechanics*, 536(1):107–144.

[Liu et al., 1997] Liu, W., Li, S., and Belytschko, T. (1997). Moving least square Kernel Galerkin method (I) methodology and convergence. *Comput. Methods Appl. Mech. Engineering.*, 143:113.

[Lo and Shao, 2002] Lo, E. and Shao, S. (2002). Simulation of near-shore solitary wave mechanics by an incompressible SPH method. *Applied Ocean Research*, 24:275–286.

[Lucy, 1977] Lucy, L. (1977). A numerical approach to testing the fission hypothesis. *Journal Astronomical.*, 82:1013–1924.

[Monaghan, 1982] Monaghan, J. J. (1982). Why particle methods work. *Siam J. Sci. Stat. Comput.*, 3:422–433.

[Monaghan, 1989] Monaghan, J. J. (1989). On the Problem of Penetration in Particle Methods. *Journal Computational Physics*, 82:1–15.

[Monaghan, 1990] Monaghan, J. J. (1990). Modeling the universe. In *Proceedings of the Astronomical Society of Australia*, volume 18, pages 233–237.

[Monaghan, 1992] Monaghan, J. J. (1992). Smoothed Particle Hydrodynamics. *Annual Rev. Astron. Appl.*, 30:543–574.

[Monaghan, 1994] Monaghan, J. J. (1994). Simulating free surface flows with SPH. *Journal Computational Physics*, 110:399– 406.

[Monaghan, 1996] Monaghan, J. J. (1996). Gravity currents and solitary waves. *Physica D.*, 98:523–533.

[Monaghan, 2000] Monaghan, J. J. (2000). SPH without Tensile Instability. *Journal Computational Physics*, 159:290–311.

[Monaghan, 2005] Monaghan, J. J. (2005). Smoothed Particle Hydrodynamics. *Reports on Progress in Physics*, 68:1703–1759.

[Monaghan, 2006] Monaghan, J. J. (2006). Time stepping algorithms for SPH. Technical report, Monash University.

[Monaghan et al., 1999] Monaghan, J. J., Cas, R. F., Kos, A., and Hallworth, M. (1999). Gravity currents descending a ramp in a stratified tank. *Journal Fluid Mechanics*, 379:39–70.

[Monaghan and Kocharyan, 1995] Monaghan, J. J. and Kocharyan, A. (1995). SPH simulation of multi-phase flow. *Computer Physics Communication*, 87:225–235.

[Monaghan and Kos, 1999] Monaghan, J. J. and Kos, A. (1999). Solitary Waves on a Cretan Beach. *J. Waterway, Port, Coastal and Ocean Engineering.*, 125:145–154.

[Monaghan and Kos, 2000] Monaghan, J. J. and Kos, A. (2000). Scott Russells Wave Generator. *Physics of Fluids*, 12:622–630.

[Monaghan et al., 2003] Monaghan, J. J., Kos, A., and Issa, N. (2003). Fluid motion generated by impact. *J. Waterway, Port, Coastal and Ocean Engineering*, 129(6):250–259.

[Monaghan and Lattanzio, 1985] Monaghan, J. J. and Lattanzio, J. C. (1985). A refined method for astrophysical problems. *Astron. Astrophys*, 149:135–143.

[Monaghan and Lattanzio, 1991] Monaghan, J. J. and Lattanzio, J. C. (1991). A simulation of the collapse and fragmentation of cooling molecular clouds. *Astrophysical Journal.*, 375:177–189.

[Montes, 1998] Montes, J. S. (1998). *Hydraulics of open channel flow.* ASCE Press, New- York.

[Morris et al., 1997] Morris, J., Fox, P., and Zhu, Y. (1997). Modeling low Reynolds number incompressible flows using SPH. *Journal of Computational Physics*, 136:214–226.

[Nagasawa et al., 1988] Nagasawa, M., Nakamura, T., and Miyama, S. M. (1988). Three-dimensional hydrodynamical simulations of type II supernova - Mixing and fragmentation of ejecta Publ. *Astron. Soc. Jpn.*, 40:691–708.

[Nie et al., 2004] Nie, X. B., Chen, S. Y., E, W. N., and Robbins, M. O. (2004). A Continuum and Molecular Dynamics Hybrid Method for Micro- and Nano-Fluid Flow. *Journal of Fluid Mechanics*, 500:55–64.

[Panizzo, 2004] Panizzo, A. (2004). *Physical and Numerical Modelling of Sub-aerial Landslide Generated Waves*. PhD thesis, Universita degli Studi di L'Aquila.

[Panizzo et al., 2007] Panizzo, A., Capone, T., and Dalrymple, R. A. (2007). Accuracy of kernel derivatives and numerical schemes in SPH. *Submitted to Journal of Computational Physics*.

[Panizzo et al., 2006] Panizzo, A., Cuomo, G., and Dalrymple, R. A. (2006). 3D-SPH simulation of landslide generated waves. In *Proc. 30th International Conference on Coastal Engineering*.

[Panizzo and Dalrymple, 2004] Panizzo, A. and Dalrymple, R. A. (2004). SPH modelling of underwater landslide generated waves. In *Proc. 29th International Conference on Coastal Engineering*, pages 1147–1159.

[Peskin, 1977] Peskin, C. S. (1977). Numerical analysis of blood flow in the heart. *Journal Computational Physics*, 25:220– 252.

[Phillips and Monaghan, 1985] Phillips, G. J. and Monaghan, J. J. (1985). A Numerical Method for Three-dimensional simulations of Collapsing, Isothermal, Magnetic Gas Clouds. *Mon. Not. R. Astr. Soc.*, 216:883–895.

[Randles and Libersky, 1996] Randles, P. and Libersky, L. (1996). Smoothed Particle Hydrodynamics some recent improvements and applications. *Comput. Methods Appl. Mech. Engineering.*, 138:375– 408.

[Ritter, 1892] Ritter, A. (1892). Die Fortpflanzung der Wasserwellen. *Zeitschrift des Vereines Deutscher Ingenieure.*, 36:947–954.

[Rogers and Dalrymple, 2004] Rogers, B. and Dalrymple, R. A. (2004). SPH modeling of breaking waves. In *Proc. 29th International Conference on Coastal Engineering*, pages 415–427.

[Rogers and Dalrymple, 2007] Rogers, B. and Dalrymple, R. A. (2007). SPH modeling of tsunami waves. In *Advanced Numerical Models for Simulating Tsunami Waves and Runup*.

[Rogers et al., 2007] Rogers, B., Dalrymple, R. A., Stansby, P., and Laurence, D. (2007). Development of a Paralell SPH code for free-surface wave hydrodynamics. In *SPHERIC, Second International Workshop*, pages 111–114.

[Shao and Gotoh, 2004] Shao, S. and Gotoh, H. (2004). Simulating coupled motion of progressive wave and floating curtain wall by SPH-LES model. *Coastal Engineering Journal*, 46(2):171–202.

[Shao et al., 2006] Shao, S., Ji, C.-M., Graham, D., Reeve, D., James, P., and Chadwick, A. (2006). Simulation of wave overtopping by an incompressible SPH model. *Coastlab Engineering*, 53:723–735.

[Shapiro et al., 1996] Shapiro, P. R., Martel, H., Villumsen, J. V., and Owen, J. (1996). Adaptive Smoothed Particle Hydrodynamics, with Application to Cosmology: Methodology. *Astrophysical Journal Supplement*, 103:269–330.

[Smagorinsky, 1963] Smagorinsky, J. (1963). General circulation experiments with the primitive equations: I. the basic experiment. *Monthly Weather Review*, 91:99–164.

[Stansby et al., 1998] Stansby, P. K., Chegini, A., and Barnes, T. C. (1998). The initial stages of dambreakflow. *J. Fluid Mech.*, 374:407–424.

[Stellingwerf and Peterkin, 1990] Stellingwerf, R. F. and Peterkin, R. E. (1990). Smooth particle magnetohydrodynamics. Technical report, Albuquerque: Mission Res. Corp.

[Swegle and Attaway, 1995] Swegle, J. W. and Attaway, S. (1995). On the feasibility of using Smoothed Particle Hydrodynamics for underwater explotion calculation. *Comput- Mech.*, 17:151–168.

[Swegle et al., 1995] Swegle, J. W., Hicks, D. L., and Attaway, S. W. (1995). Smoothed Particle Hydrodynamics Stability Analysis. *Journal of Computational Physics*, 116:123–134.

[Trulsen et al., 2002] Trulsen, K., Spjelkavik, B., and Mehlum, E. (2002). Green water computed with a spline-based collocation method for potential flow. *Int. J. Appl. Mech. Engineering.*, 7(1):107–123.

[Verlet, 1967] Verlet, L. (1967). Computer Experiments on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Phys. Rev.*, 159:98–103.

[Versteeg and Malalasekera, 1995] Versteeg, H. and Malalasekera, W. (1995). *An Introduction to Computational Fluid Dynamics, The Finite Volume Method.* Longman Scientific Technical.

[Violeau and Issa, 2006] Violeau, D. and Issa, R. (2006). Numerical Modelling of Complex Turbelent Free-Surface Flows with the SPH Method: an Overview. *Int. J. Numer.l Meth. Fluids*, 53:277–304.

[Wang et al., 1998] Wang, Z., Jensen, J. J., and Xia, J. (1998). On the Effect of Green Water on Deck on the Wave Bending Moment. In *Proceedings of the Seventh International Symposium on Practical Design of Ships and Mobile Units, The Hague.*

[Wei et al., 1995] Wei, G., Kirby, J. T., Grilli, S. T., and Subramanya, R. (1995). A Fully Nonlinear Boussinesq Model for Surface Waves. *Journal of Fluid Mechanics*, 294:71–92.

[Wendland, 1995] Wendland, H. (1995). *Computational aspects of radial basis function approximation.* Elsevier.

[Zienkiewicz and Taylor, 2000] Zienkiewicz, O. C. and Taylor, R. L. (2000). *The Finite Element.* Butterworth Herinemann, 5th edition.

[Zou, 2007] Zou, S. (2007). *Coastal Sediment Transport Simulation by Smoothed Particle Hydrodynamics.* PhD thesis, Civil Engineering, Johns Hopkins University.

[Zou and Dalrymple, 2006] Zou, S. and Dalrymple, R. A. (2006). Sediment suspension simulation under oscillatory flow with SPH-SPS method. In *Proc. 30th International Conference on Coastal Engineering.*