UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONA TECH

DOCTORAL THESIS

Coupling shallow water models with three-dimensional models for the study of fluid-structure interaction problems using the particle finite element method

Author:
Miguel MASÓ

Supervisors:
Dr. Eugenio OÑATE
Dr. Ignasi DE-POUPLANA

A thesis submitted in fulfillment of the requirements for the degree of Civil Engineering Ph. D.

in the

International Centre for Numerical Methods in Engineering
Department of Civil and Environmental Engineering

July 1, 2022

Abstract

This thesis investigates numerical methods for the simulation of surface water flows, focusing on the interaction between the large scale and the local scale and its application to natural hazards. Several families of numerical methods for the approximation of large scale phenomena and the coupling with the local scale have been analyzed.

The general motion of a fluid mass is governed by the Navier-Stokes equations, which can accurately solve the local scale phenomena. However, the same level of accuracy is not required by the large scale solution of the water-related events. In this context, the shallow water equations are defined. In contrast to the extensive use of the Finite Element Method for solving the Navier-Stokes equations, the shallow-water equations are usually solved with the Finite Volume Method. Thus, an effort have been done to solve both equations in an unified framework.

The first part of this thesis is devoted to study stabilized formulations of Finite Element Method for the different forms of the shallow water equations. Stabilized formulations arise from the need to mitigate the various instabilities inherent in numerical approximations. The first source of instability is the incompatibility of the equal interpolation of the variables. The second source of instability is the presence of shocks due to the change of regime or hydraulic jumps. Finally, Gibbs oscillations may appear on the moving shoreline and monotonic properties of the physical system are lost by the numerical approximation.

The second part of the thesis is committed to the coupling strategies of the numerical methods for the Navier-Stokes and the shallow water equations. The case of a coupling from the local scale to the large scale is analyzed. This type of coupling corresponds to the generation of cascading natural hazard. The proposed strategy combines a Lagrangian Navier Stokes multi-fluid solver with an Eulerian method based on the Boussinesq equations, an extension of the shallow water equations.

Finally, the proposed technique is applied to the numerical simulation of landslide-generated impulse waves. The Particle Finite Element Method has been used to model the landslide runout, its impact against the water body and the consequent wave generation. The results of this fully-resolved analysis are stored at selected interfaces and then used as input for the modelling of waves propagation on the far-field. This one-way coupling scheme drastically reduces the computational cost of the analyses while maintaining high accuracy in reproducing the key phenomena of cascading natural hazards.

Resumen

En esta tesis se investigan métodos numéricos para la simulación de flujos de aguas superficiales, haciendo énfasis en la interacción entre las distintas escalas y su aplicación a desastres naturales. Se han analizado diversas familias de métodos numéricos para aproximar los fenómenos a gran escala y su acoplamiento con la escala local.

El movimiento general de una masa de fluido se rige por las ecuaciones de Navier-Stokes, que pueden resolver con precisión los fenómenos a escala local. Sin embargo, la solución numérica a gran escala de dichos fenómenos, no requiere el mismo nivel de precisión. En este ámbito, se definen las acuaciones de agua poco profundas. En contraste con el amplio uso del Método de los Elementos Finitos para aproximar las ecuaciones de Navier-Stokes, las ecuaciones de aguas poco profundas se suelen resolver con el Método de los Volúmenes Finitos. Por ello, se ha realizado un esfuerzo para resolver ambas ecuaciones en un marco unificado.

La primera parte de esta tesis está dedicada a estudiar formulaciones estabilizadas para el Método de los Elementos Finitos aplicado a las diferentes formas de las ecuaciones de aguas someras. Las formulaciones estabilizadas surgen de la necesidad de mitigar las diferentes inestabilidades inherentes a las aproximaciones numéricas. La primera fuente de inestabilidad es la incompatibilidad debida a la interpolación de las variables. La segunda fuente de inestabilidad es la presencia de discontinuidades debidos al cambio de régimen o a los saltos hidráulicos. Por último, pueden aparecer oscilaciones de Gibbs en la línea de costa en movimiento, dado que las propiedades monótonas del sistema físico se pierden por la aproximación numérica.

La segunda parte de la tesis está dedicada a las estrategias de acoplamiento de los métodos numéricos para las ecuaciones de Navier-Stokes y de aguas poco profundas. Se ha analizado el caso de acoplamiento desde la escala local a la escala global. Este tipo de acoplamiento corresponde a la generación de desastres naturales en cascada. La estrategia propuesta combina un solver Lagrangiano de Navier Stokes para multi-fluidos con un método Euleriano basado en las ecuaciones de Boussinesq, una extensión de las ecuaciones de aguas someras.

Finalmente, la técnica propuesta se ha aplicado a la simulación numérica de olas generadas por deslizamientos. El deslizamiento de ladera, su impacto contra la masa de agua y la consiguiente generación de olas se ha modelado con el Método de Elementos Finitios de Partículas. Los resultados de este análisis detallado se almacenan en las interfaces seleccionadas que, luego, se utilizan como punto de entrada para modelar la propagación de olas en el campo lejano. Este esquema de acoplamiento unidireccional reduce drásticamente el coste computacional, a la vez que se mantiene una alta precisión en la simulación de los fenómenos clave de desastres naturales.

Acknowledgements

Over the years developing this thesis there have been many people around me who have helped me to achieve this goal. I am indebted to all of them and it would be impossible to mention all of them.

I want to thank to Jaume Armenogu, who encouraged me to begin a Doctorate, and also to thank Eugenio Oñate for giving me the opportunity to work on this thesis and for his advices as supervisor. I also want to thank Ignasi de-Pouplana for being my co-supervisor.

From CIMNE, tanks to Sergio Idelsohn and Riccardo Rossi for all the clarifying and fruitful lessons that they have given to me. Thanks to Alessando and Alejandro for their collaboration during the last part of the thesis. I also want to mention my colleagues who finished their doctorate in the previous years and I have been helped by their knowledge and advice, Vicente, Pablo, Jordi, Guillermo, Rubén. Let me also mention my colleagues from CIMNE. From CIMAT, thanks to Prof. Botello, Jorge López and all the friends I met in Guanajuato.

In the same line, thanks to all my colleagues, friends and so many other people that I may not mention explicitly. Thanks to my family for their constant support and encouragement.

Finally, I want to acknowledge the doctoral scholarship received from Universitat Politècnica de Catalunya and Banco Santander (53 FPI-UPC 2018) and the big support that CIMNE gave to this research.

Contents

List of Figures

List of Tables

List of Abbreviations

ALE	Arbitrary Lagrangian Eulerian
ASGS	Algebraic Sub-Grid Scale
BDF	Backward Differentiation Formula
CG	Continuous Galerkin
DG	Discontinuous Galerkin
DAB	Double Absorbing Layer
FC	Flux Correction
FD	Finite Difference method
FEM	Finite Element Method
FFS	Far-Field Solver
FIC	Finite-Increment Calculus
FSI	Fluid-Structure Interaction
FV	Finite Volume method
GJV	Gradient Jump viscosity
LGW	Landslide Generated Wave
LTS	Local Time Step
MPM	Material Point Method
NFS	Near-Field Solver
NS	Navier-Stokes
PFEM	Particle Finite Element Method
PFEM2	Particle Finite Element Method 2-nd generation
PML	Perfectly Matched Layer
RV	Residual Viscosity
SPH	Smooth Particle Hydrodynamics
SUPG	Stream-Upwind Petrov-Galerkin
SW	Shallow Water
TVD	Total Variational Diminishing
VMS	Virtual Multi-Scale
VOF	Volume Of Fluid

Chapter 1

Introduction

1.1 Overview

The continued increase of computing resources available for scientific and industrial research has motivated the simulation of complex flows. Several attempts have been done in order to fully simulate water-related hazards, from the generation up to their interaction with structures.

Usually, water-related hazards involve floods, which are complex natural phenomena with a great assortment of causes and a huge casuistic depending on the land characteristics or the climatic conditions. Preventing floods is still a great challenge due to uncertainty of climatic conditions and complexity of modeling land, usually defined by a great number of variables and parameters.

Furthermore, it is possible to identify several scales, local and global, with the local scale requiring a high fidelity simulation. The presence of such different levels of resolution makes the global simulation of these phenomena still unaffordable. Firstly, the event generation can be global or local. If it is local, a *Fluid-Structure Interaction* (FSI) simulation may be needed. Then the propagation of the event usually allows to assume several simplifications, corresponding to the large scale field. Finally, the interaction with structures is strictly a local scale problem.

Landslide-Generated Waves (LGW) are an example of water-related natural hazards. These events are originated when a landslide impacts a water reservoir and produces large-amplitude waves. LGW events can have devastating effects on the coastal areas of water basins, such as lakes, fjords and artificial reservoirs. Accurate modeling and prediction of LGWs are of key importance to reduce their catastrophic effects. Both experimental and numerical studies have greatly contributed to the enhancement of the forecasting capabilities against these natural hazards. In that case, both the generation of the impulse wave and the interaction with structures require a local-scale resolution.

The direct simulation of all the scenario with a local scale resolution is computationally highly demanding and it can become unaffordable as the size of the scenario increases. This fact leads to the consideration of alternative strategies combining the numerical simulation of the large and local scale in a staggered frame. Figure 1.1 shows a conceptual identification of the different scales and the possible approaches. The computational cost is proportional to the area enclosed by the space-time domain and proportional to the efficiency of each solver. In Figure 1.1 the local-scale resolution is identified with a *Near Field Solver* (NFS) and the large-scale resolution is identified with a *Far Field Solver* (FFS).

The staggered approach, apart from being less computationally intense, allows to decouple the uncertainty related to the generation of the event from the FSI simulation. In addition, the staggered approach allows to concentrate the computational

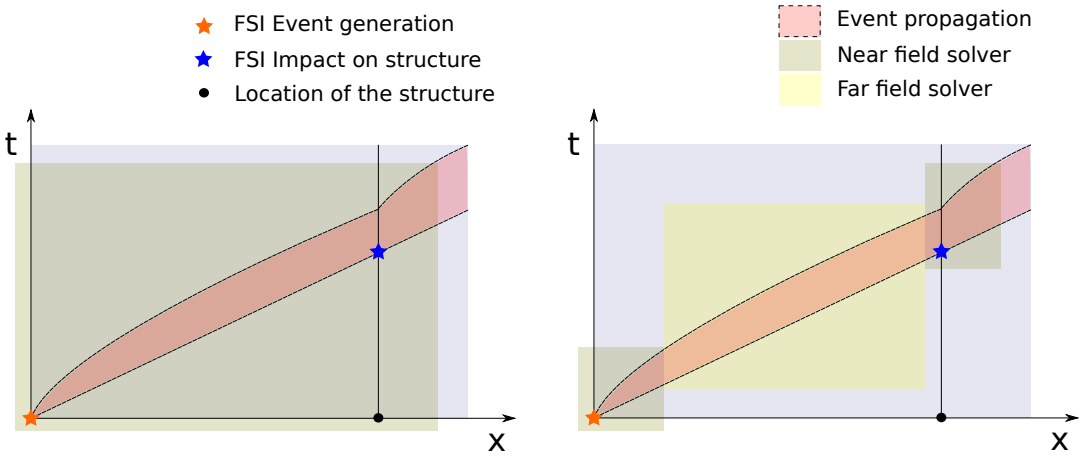


FIGURE 1.1: Propagation path of the event and possible computational approaches. Left: Brute force approach. Right: Staggered approach.

resources only where they are needed, thus leading to a more efficient level of accuracy.

The main aspects that have been explored in this thesis are the numerical approximation of the FFS and the coupling strategy of the proposed FFS with an existing NFS. Special attention have been paid to the convergence and stability properties of the FFS aimed to solve the large scale.

1.2 Objectives and outline

The work developed within the framework of this thesis can be enclosed in the global objective of investigating Finite Element formulations applied to large-scale water-related hazards. In particular, the analyzed methods have to be capable of reproducing local-scale effects around structures of interest and accurately model the large-scale propagation. This requirement lead to the analysis of the Finite Element Method for solving the Navier-Stokes equations on local scales and solving the shallow water equations on large scales.

The Navier-Stokes equations are typically solved using the finite element method, while the shallow water equations are often solved using the finite volume method. Since we are interested in solving the global simulation within the same framework, a previous investigation on stabilized finite elements for the shallow water equations is presented.

This thesis is structured in 6 chapters, including the present introduction and a bibliographic revision. The bibliographic revision is specially extensive for the shallow water equations. Chapter 2 presents a review of the governing equations which will be solved in the latter chapters.

Chapter 3 is dedicated to the Finite Element Method applied to the shallow water equations. Stabilized formulations and quasi-monotonic formulations are presented. Part of the developments of the numerical methods for the shallow water equations are moved to appendices A and B, where the particle methods and mesh refinement are explored.

An extension of the shallow water equations to wave propagation on intermediate and deep water is presented in chapter 4. These equations are known as the Boussinesq equations. The strategies developed in the previous chapter are applied

to the Boussinesq equations, however, additional requirements must be fulfilled in order to ensure stability and accuracy. This chapter is of crucial importance in order to analyze impulse waves in the context of water-related hazards.

Chapter 5 is devoted to the coupling strategies between the shallow water approximations –specially the Boussinesq equations– and the fully resolved methods. An example of coupling for landslide long wave generation is provided.

Finally, the thesis is closed with some conclusions and possible further research lines in chapter 6.

1.3 State of the art

In this section a bibliographic revision of the methods employed to solve free surface flows problems is presented. The general motion of a fluid is described by the Navier-Stokes equations. When free-surface water flows are considered, there are two implications, the incompressibility and homogeneous continuum medium enclosed by the interface. After presenting the most common numerical methods used to approximate the incompressible Navier-Stokes equations for free surface flows, the bibliographic review will be focused on the shallow water equations. The shallow water equations are derived from the Navier-Stokes equations when some simplifications are assumed. Finally, the bibliographic revision is closed with the coupling strategies for both models.

1.3.1 Numerical methods for the Navier-Stokes equations

Nowadays, the accurate approximation of the fluid flow equations aims at representing the flow up to the smallest scales. In that sense, the effort is focused either on obtaining higher precision numerical schemes or on refining the discretization. The first approach is cheaper in contrast to the refinement approach, but the refinement approach has been proven to be very versatile. In the case of complex geometries, higher order approximations are equivalent to the *Finite Volume methods* (FV) or *Finite Element Methods* (FEM).

Nevertheless, when the Galerkin discretization is used within the frame of the FEM, an unstable behavior might be obtained. These instabilities are related to the non symmetric convection operator and to interpolation [21, 34]. The stabilized methods like the *Streamline Upwind Petrov Galerkin* (SUPG) [22, 84] can be framed in the *Variational Multi-Scale* (VMS) concept [82]. Latter, other stabilizations were presented, such as *Finite-Increment Calculus* (FIC) [141] or *Galerkin Least Squares* (GLS) [83]. In fact, the latter stabilizations belong to the same family of the previous ones, which consist in adding extra terms based on the residual of the balance equations. Since these stabilization techniques are consistent, they allow using higher order approximations.

However, the terms introduced by stabilizations in order to keep consistency notably increase the complexity of the equations, can couple unknowns, and even increase the non-symmetry of the system. In order to overcome this issues, projection methods only introduce the terms required for stability purpose. The key of these methods is the choice of the projector. A global L^2 projector is used in the orthogonal sub scales method [38]. Other methods avoiding the global projection are named local projection stabilization [17, 123] or nodal projection stabilization [8].

Apart from the stabilization technique for the Navier-Stokes equations, the numerical approximation must be able to deal with the interface discontinuity. Concerning the coordinate frame where the governing equations are solved, the solution methods can be classified in Eulerian and Lagrangian formulations. Classical numerical methods to solve CFD problems typically use the Eulerian formulation with a level set function [31] with enriched shape functions [27]. Lagrangian formulations take advantage on the possibility of the FEM to discretize complex geometries. In that case, the FEM is combined with a mesh moving and remeshing strategy. It can be applied in an *Arbitrary Lagrangian-Eulerian formulation* (ALE) [55] or in a pure Lagrangian formulation. This is the case of the *Particle Finite Element Method* (PFEM) [88, 139]

Regardless of the framework where the governing equations are solved, the formulation needs to be stabilized in order to fulfill the *inf-sup* condition [21]. In this work, the PFEM formulation stabilized with FIC will be used. It has been successfully applied to solve free surface flows [52] and FSI analysis [140] or in combination with other numerical models [143].

1.3.2 Numerical methods for the shallow water models

When large scale free surface flows are considered, some simplifications can be assumed, such as small vertical scale and hydrostatic pressure distribution. These assumptions allow to express the Navier-Stokes equations in a depth averaged balance, which are the shallow water equations. Depending on the assumptions, slightly different equations are obtained. While the Saint-Venant equations are well suited for convective flows, the Boussinesq equations describe the oscillatory phenomena of waves. Anyway, both systems of equations are hyperbolic and present an analogy with the Euler conservation laws or compressible Navier-Stokes equations. Hence, the numerical methods commonly used for the compressible Navier-Stokes equations can be applied to the shallow water equations.

The difference of the shallow water equations with respect to the Navier-Stokes equations is the fact that the domain is restricted to the wet area. The wet area is described by the positive water depth, where the depth integration is defined. The shoreline moves according to the dynamics and is characterized by a null water depth, a region where the equations are singular and the system is no longer hyperbolic. This property will need specific numerical considerations.

Löhner [111] made a review of possible approximation methods to solve fluid dynamics. Being a discretization $u^h = N^i \hat{u}_i$ $(i = 1, 2, \ldots, m)$ approximating the solution u, the weighted residual is defined as

$$\int_\Omega W^i r(u^h) d\Omega = 0$$

This description allows to wrap the most common numerical methods in the same frame. Depending on the choice of the basis functions N and the trial function W, the weighted residual yields several numerical methods. Table 1.1 summarizes the most common choices of trial functions.

The most commonly used method by the shallow water community is FV because of its stability properties when dealing with the moving shoreline [2, 105]. More recently, the *Discontinuous Galerkin* technique has been applied to the shallow water equations [6, 95, 107]. Like FV, the DG method has the advantage of computing the fluxes at the element boundary, allowing the method to naturally consider

	N^i	W^i
Finite differences (FD)	polynomial	$\delta(x_i)$
Finite volumes (FV)	polynomial	1 if $x \in \Omega_{el}$
Galerkin finite elements (FEM)	polynomial	N^i
Discontinuous Galerkin (DG)	polynomial	N^i if $x \in \Omega_{el}$

TABLE 1.1: Possible choices of trial and test functions N^i and W^i

the moving shoreline. However, the introduction of high order DG schemes is not straightforward.

In this research the FEM for the shallow water equations will be explored. Apart from its ability to solve the oscillatory problem of the Boussinesq equations, it will allow a unified formulation for the coupling strategy. When the FEM is considered, several sources of instabilities arise. Firstly, the *inf-sup* condition if the same space of interpolation is used for all the variables. Secondly, analogously to compressible flows, shocks might appear in the regime variations, from sub-critical to super-critical. Lastly, Gibbs oscillations appear at the shoreline, since the partially wet elements are not able to preserve the water positivity.

Several authors reported the presence of instabilities on the numerical approximation of the shallow water equations on the wet domain. Ad-hoc techniques were proposed in order to achieve stability, such as a DG for one variable [68], different order of interpolation [106], division in sub-triangles [78], non conforming interpolations [74] or addition of extra diffusion [51]. A similar situation happened with the Boussinesq equations, see [158, 161] as an example. Heniche related the instabilities to the *inf-sup* condition in [78], but until Codina in [37] the FEM stabilization was not applied to the shallow water equations.

The stabilized FEM are known to ensure global stabilization, however, monotonic properties are not inherited by the numerical scheme. This issue is related to the dry-wet interface, since the wet domain is defined by a positive water depth. Usually, the front tracking is related to the techniques used for the shock capturing, since the shoreline is also a discontinuity. Monotonic properties of the numerical approximation are related to the Godunov barrier theorem [66] which states that a scheme of order higher than 1 is not oscillatory free. The *Flux Corrected Transport* FCT algorithm [114] uses the Godunov theorem to construct a non linear scheme which combines a low order non-oscillatory scheme with a high order one. The process of combining the two schemes is called limiting. An application of it to the shallow water equations can be found in [137].

Other methods inspired by the flux limiting are the edge based schemes [113]. The edge based structure is an efficient way to assemble the system matrix which resort to a FV approximation of convective terms. If it is assumed that the fluxes of the variables are constant along the edges, a discontinuity will occur at the edge midpoint. Then, one can replace the Galerkin flux by a Riemann flux and obtain a *Total Variational Diminishing* scheme (TVD).

Finally, non-linear stabilizations are a more consistent way to add local stabilizations around shocks and the moving shoreline [36]. Badia and Hierro presented a monotonicity preserving stabilized finite element for hyperbolic equations [9].

1.3.3 Large-local scale approach and coupling strategies

The simulation of a large scenario where local effects are of key importance can be tackled in different ways. In many practical situations, the behavior of a fluid can

be modelled by the SW equations, even though the assumptions of small vertical acceleration are not perfectly fulfilled. However, the accurate modelling of the event triggering the impulse wave or the interaction with structure can be out of the scope of reduced models.

On the one hand, physical models are particularly helpful to identify the key parameters of the flow characteristics. The authors in [64, 126, 135] experimentally characterized the parameters for the definition of the impulse wave generated by a landslide. [96, 97] performed an extensive characterization of the wave run-up and interaction with a nearshore structure. Nevertheless, laboratory tests are mainly devoted to determine near-field conditions under specific simplifications, whereas the global scenario is a combination of a variety of situations.

On the other hand, numerical methods have the potential to predict both near- and far-field waves characteristics. Due to its computational requirements when large domains and long time durations are considered, numerical methods can be classified into three main groups.

Firstly, the entire domain can be simulated using a reduced model, typically based on SW. For the case of LGW, the impact can be imposed as an equivalent boundary condition [7, 159]. In the case of the interactions with bodies, some large scale applications have been modelled using the SW equations for FSI problems [20, 65]. However, the strong simplifications allow to have only an approximate idea of the global scenario.

Secondly, the global scenario can be simulated in a unique coupled model. Some holistic attempts have been made both for the wave generation and the impact against structures (see [47, 61, 170] for example). However, the computational cost for a fully resolved method is still unaffordable for large-scale events.

Lastly, the partitioned approach splits the global scenario into several simulations that interact with each other at their interface. A first inspiration of the strong coupling of different order models can be found in [59]. There are also some recent strong couplings between the SW and Navier-Stokes equations [146]. However, the weak coupling simplification preserves the computational advantages of the partitioned approach and it still ensures an accurate modeling of the key phenomena of an wave generation scenario or a FSI scenario.

One of the first applications of the weakly-coupled partitioned method for LGWs was presented in [77]. In this work, a simplified 3D model was used for the landslide-water impact and a shallow water model was applied for the far-field wave propagation. Other examples can be found in [110] where a flooding scenario was studied by coupling a 3D compressible Eulerian solver with a Boussinesq model.

In this work, we propose and validate a novel partitioned model for LGW scenarios. In this new strategy, a Lagrangian finite element method, namely the Particle Finite Element Method (PFEM) [46, 88, 139], is used as the NFS and a Boussinesq model is used as FFS. Particular attention is devoted here to analyze the effect of the near-field boundary conditions on the far-field propagating wave.

1.4 Research dissemination

Some of the developments if this thesis have been published in the format of articles in peer reviewed journals. Since the research has advanced gradually, the articles are related to a chapter, but there are some differences, which can be big. The chapters are more extensive than the articles and some parts of the articles are omitted to avoid repetitions. There is also not the same sequence between the publication date

and the chapter order. On the other hand, since the notation is introduced gradually, it has been unified in the thesis and may be slightly different from the articles and this document.

Chapter 3 M. Masó, I. de Pouplana, and E. Oñate. "A FIC-FEM procedure for the shallow water equations over partially wet domains". In: *Computer Methods in Applied Mechanics and Engineering* 389 (Feb. 2022), p. 114362. DOI: 10.1016/j.cma.2021.114362

Chapter 5 M. Masó, A. Franci, I. de Pouplana, A. Cornejo, and E. Oñate. "A Lagrangian-Eulerian procedure for the coupled solution of the Navier-Stokes and shallow water equations for landslide-generated waves". In: *Advanced Modeling and Simulation in Engineering Sciences* (Mar. 2022). DOI: 10.21203/rs.3.rs-1457837/v1 (preprint)

Appendix A A. Puigferrat, M. Maso, I. De Pouplana, and G. Casas. "Semi-Lagrangian formulation for the advection-diffusion-absorption equation". In: *Computer methods in applied mechanics and engineering* 380 (July 2021), 113807:1–113807:26. DOI: 10.1016/j.cma.2021.113807

In addition, the content of the chapters has been also disseminated in the form of oral presentations in scientific conferences and congresses:

Chapter 5 M. Masó, I. de Pouplana, A. Franci, A. Cornejo, and E. Oñate. "A one-way coupled Lagrangian-Eulerian procedure for the solution of landslide-generated waves". In: *WCCM*. 2022 (Abstract accepted)

Chapter 5 M. Masó, I. de Pouplana, A. Franci, A. Cornejo, and E. Oñate. "Un procedimiento Lagrangiano-Euleriano para la solución acoplada de las ecuaciones de Navier-Stokes y aguas poco profundas para olas generadas por deslizamientos de laderas". In: *WCCM*. 2022 (Abstract accepted)

Chapter 5 A. Franci, M. Cremonesi, U. Perego, G. Crosta, M. Masó, I. D. Pouplana, and E. Oñate. "Numerical Modelling of Landslide-Generated Waves". In: *ECCOMAS*. 2022

Appendix A M. Masó, I. de Pouplana, and E. Oñate. "Coupling shallow water models with three-dimensional models for the study of fluid-structure interaction problems using the particle finite element method". In: *VI International Congress on Particle-based Methods. Fundamentals and Applications*. Barcelona: CIMNE, 2019

Chapter 2

Governing equations for free surface flows

Water related natural hazards usually involve free surface flows. This property allows to assume some simplifications, specially at the large scale. For this reason, the general case describing the fluid motion is presented first, the Navier-Stokes equations. Then, some simplifications for the free surface flows are made, yielding the shallow water equations. Different assumptions for the depth integrated models will provide different sets of equations and its range of applicability as well as physical properties will be explained.

2.1 Navier-Stokes equations

The motion of a fluid is described by the Navier-Stokes equations. The case of a free surface flow is described as well by those equations and the free surface is located where the densities or the fluid properties are discontinuous. In general, an air-water interface is assumed but some natural phenomena may include a more complex configuration, such as debris flow-air-water interfaces. All this continuous media can be considered isotermal and incompressible and thus, the standard formulation of the Navier-Stokes equations can be used.

The problem consists in the incompressible Navier-Stokes equations in a time interval $(0, t_f)$ and in a spatial domain $\Omega \in \mathbb{R}^{n_d}$, being n_d the number of space dimensions, 3 unless otherwise stated. Let t be a certain time instant in the temporal domain $(0, t_f)$ and $\mathbf{x}$ a given point in the spatial domain Ω. The balance equations for the momentum and mass are written in the following form:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \frac{1}{\rho} \nabla p + \frac{1}{\rho} \nabla \cdot \boldsymbol{\tau} = \mathbf{f} \tag{2.1a}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{2.1b}$$

With the appropriate initial and boundary conditions. Let Γ be the boundary of the domain Ω and $\mathbf{n}$ the unit outward normal on Γ. Dirichlet and Neumann boundary conditions are considered, Γ_D and Γ_N respectively, such that $\Gamma_D \cup \Gamma_N = \Gamma$. ρ is the fluid density, p is the pressure, $\mathbf{u}$ is the velocity, $\boldsymbol{\tau}$ is the viscous stresses tensor and $\mathbf{f}$ is the body forces vector.

2.2 Shallow water equations

The flow of water in shallow layers occurs in a wide range of situations, such as coastal dynamics and hydraulics. In such cases, free surface flows in relatively thin

layers compared to the characteristic horizontal length and thus, the horizontal velocities are of primary importance. Under those circumstances, the problem can be reasonably approximated by the horizontal plane.

The shallow water equations are the result of integrating vertically the Navier-Stokes equations, assuming incompressibility, small vertical velocity and negligible vertical acceleration [1, 171]. The assumptions over the vertical velocity and its acceleration are equivalent to hydrostatic pressure, in fact, the vertical component of the momentum equation (2.1a) is reduced to

$$\frac{1}{\rho}\frac{\partial p}{\partial x_3} + g = 0 \tag{2.2}$$

After substitution of the hydrostatic pressure assumption 2.2 into the mass conservation 2.1b, the governing equations are integrated from the bottom z to the free surface η. The problem is closed by adding two kinematic boundary conditions at the bottom and at the free surface:

$$u_3(z) = \frac{Dz}{Dt}\,, \quad u_3(\eta) = \frac{D\eta}{Dt} \tag{2.3}$$

The governing equations are expressed in terms of a new set of primary variables: the water depth and the horizontal flow rate. The water depth is defined by the integration limits, $h = z + \eta$ and the averaged horizontal flow rate $\mathbf{q}$ is defined by the following integrated value,

$$\mathbf{q} = \int_z^\eta \mathbf{u}\, dx_3 \tag{2.4}$$

In order to avoid introducing extra notation, in a shallow water context, $\mathbf{u}$ refers to the averaged horizontal velocities. In that case, the expression 2.4 can be reduced to a compact form as $\mathbf{q} = h\mathbf{u}$.

Here we find that the resulting equations are written in the same form as the Euler conservation equations. In spite of the equations presenting some similarities to the compressible flow, the shallow water equations are describing a purely incompressible fluid: the variable water depth is playing the role of the variable pressure in compressible fluids. The equations read

$$\frac{\partial \boldsymbol{\phi}}{\partial t} + \frac{\partial \mathbf{F}_i}{\partial x_i} + \frac{\partial \mathbf{G}_i}{\partial x_i} + \mathbf{Q} = \mathbf{0} \qquad \text{for } i = 1, 2 \tag{2.5}$$

with

$$\boldsymbol{\phi} = \begin{Bmatrix} hu_1 \\ hu_2 \\ h \end{Bmatrix} \tag{2.6a}$$

$$\mathbf{F}_i = \begin{Bmatrix} hu_1u_i + \delta_{1i}\frac{1}{2}g(h^2 - z^2) \\ hu_2u_i + \delta_{2i}\frac{1}{2}g(h^2 - z^2) \\ hu_i \end{Bmatrix} \tag{2.6b}$$

$$\mathbf{G}_i = \begin{Bmatrix} -(h/\rho)\bar{\tau}_{1i} \\ -(h/\rho)\bar{\tau}_{2i} \\ 0 \end{Bmatrix} \tag{2.6c}$$

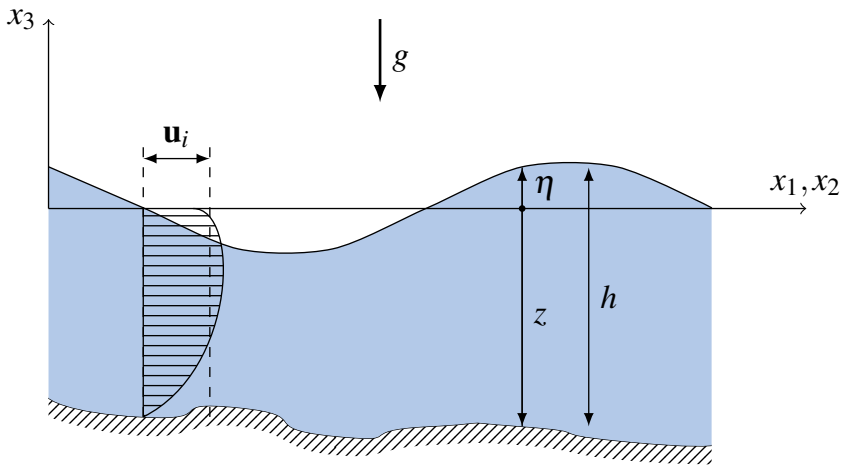


FIGURE 2.1: Diagram and notation for the balance equations (2.5) and (2.6)

$$\mathbf{Q} = \begin{Bmatrix} -g(h-z)\dfrac{\partial z}{\partial x_1} + \dfrac{h}{\rho}\dfrac{\partial p_a}{\partial x_1} - \dfrac{1}{\rho}\tau^s_{31} + \dfrac{1}{\rho}\tau^b_{31} \\ -g(h-z)\dfrac{\partial z}{\partial x_2} + \dfrac{h}{\rho}\dfrac{\partial p_a}{\partial x_2} - \dfrac{1}{\rho}\tau^s_{32} + \dfrac{1}{\rho}\tau^b_{32} \\ r \end{Bmatrix} \tag{2.6d}$$

where $\boldsymbol{\phi}$ is the vector of conserved variables, $\mathbf{F}_i$ is the vector of convective fluxes, $\mathbf{G}_i$ is the vector of viscous fluxes and $\mathbf{Q}$ is the vector source terms. In Figure 2.1 there is a representation of the variables and the notation. The coordinates are denoted with the index notation x_i. Since this formulation is defined in a two dimensional space ($n_d = 2$), in the following we will consider $i = 1, 2$. δ_{ij} is the Kronecker delta. The topography is expressed with the variable z and the free surface elevation is expressed in terms of the topography and the total depth, $\eta = z + h$. $\bar{\tau}_{ij}$ are the averaged horizontal stresses, and τ^b_{3i} and τ^s_{3i} denote the bottom and surface friction stresses respectively. Finally, r is the rain source term and p_a is the atmospheric pressure.

Usually, the bottom friction τ^b_{3i} from (2.6) is modelled with a semi-empirical formula, such as the Chezy or the Manning formula. The Manning formula generalized for two dimensions is as follows:

$$\frac{\tau^b_{3i}}{\rho} = -gn^2 \frac{|\mathbf{q}|\mathbf{q}}{h^{7/3}} \tag{2.7}$$

where n is the Manning roughness coefficient. It defines the resistance to flow by the roughness of the bottom or other macroscopic factors and it is determined empirically. In practice, the Manning coefficient varies from 0.009 for a very smooth bed (concrete) to 0.05 for a rough bed (rocks) [33].

The averaged horizontal stresses are calculated from the combination of the molecular stresses and the Reynolds stresses as follows

$$\frac{\bar{\tau}_{ij}}{\rho} = (\nu + \nu_t)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_k}{\partial x_k}\right) \tag{2.8}$$

where ν is the kinematic viscosity and ν_t is the turbulent kinematic viscosity. When any model of turbulence is considered, the turbulent stresses can be considered as

included in the bottom friction with the Manning formula [30]. In this work, the turbulent stresses will be neglected.

Returning again to the equations structure, the Euler conservation laws define two eigenvalues fo the one-dimensional case: $\lambda = u \pm c$. Where u is the modulus of the velocity and $c = \sqrt{gh}$ is the phase speed or wave speed. In the case of positive eigenvalues, the flux is supercritical, this means that the information travels in one direction, the velocity is higher than the phase speed. When the eigenvalues are of different sign, the flow is subcritical. If one of the eigenvalues is zero, the flow becomes critical and is unstable, so it will derive to a stable solution, sub or supercritical.

For the two dimensional case, does not exist a unique decomposition and this is known as amplitude dispersion. This means there is not an unique direction of propagation. The projection to the velocity direction -the main direction- will provide the eigenvalues $\lambda_1 = u - c$, $\lambda_2 = u$ and $\lambda_3 = u + c$.

2.2.1 Boundary conditions

The problem is closed with appropriate boundary conditions: an initial condition

$$\boldsymbol{\phi}(t = t_0) = \boldsymbol{\phi}_0 \tag{2.9}$$

where $\boldsymbol{\phi}_0$ are the initial water height and specific discharge. And boundary conditions at Γ, being Γ the boundary of the domain Ω. The boundary Γ is split into three subdomains, Γ_I, Γ_O and Γ_S: inflow, outflow and solid.

Inflow boundary the flow rate is known

$$\mathbf{q} = \mathbf{q}_{in} \quad \text{in} \quad \Gamma_{in}$$

If the inflow is supercritical, the water depth is also specified

$$\left.\begin{array}{l} \mathbf{q} = \mathbf{q}_{in} \\ h = h_{in} \end{array}\right\} \quad \text{in} \quad \Gamma_{in}$$

Outflow boundary The water depth is known

$$h = h_{out} \quad \text{in} \quad \Gamma_{out}$$

if the outflow is supercritical, no conditions have to be imposed.

Solid boundary slip or no slip condition can be imposed

$$\mathbf{q} \cdot \mathbf{n} = 0 \quad \text{or} \quad \mathbf{q} = \mathbf{0} \quad \text{in} \quad \Gamma_{solid}$$

2.2.2 Linearization

Usually, in the numerical study of the conservation equations, them are expressed in a quasi-linear form. The balance equation (2.5) can be linearized as follows

$$\frac{\partial \boldsymbol{\phi}}{\partial t} + \mathbf{A}_i \frac{\partial \boldsymbol{\phi}}{\partial x_i} - \frac{\partial}{\partial x_i}\left(\mathbf{K}_{ij} \frac{\partial \boldsymbol{\phi}}{\partial x_j}\right) + \mathbf{S}\boldsymbol{\phi} + \mathbf{b}_i \frac{\partial z}{\partial x_i} = 0 \tag{2.10}$$

where the matrices $\mathbf{A}_i$ and $\mathbf{K}_{ij}$ are the linearization matrices of the convective fluxes and the diffusive fluxes respectively. The convective matrices $\mathbf{A}_i$ are obtained after applying the chain rule to the vector of fluxes $\mathbf{F}_i$,

$$\frac{\partial \mathbf{F}_i}{\partial x_i} = \frac{\partial \mathbf{F}_i}{\partial \boldsymbol{\phi}} \frac{\partial \boldsymbol{\phi}}{\partial x_i} \tag{2.11a}$$

$$\mathbf{A}_i = \frac{\partial \mathbf{F}_i}{\partial \boldsymbol{\phi}} \tag{2.11b}$$

$$\mathbf{A}_1 = \begin{bmatrix} 2u_1 & 0 & -u_1^2 + gh \\ u_2 & u_1 & -u_1 u_2 \\ 1 & 0 & 0 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} u_2 & u_1 & -u_1 u_2 \\ 0 & 2u_2 & -u_2^2 + gh \\ 0 & 1 & 0 \end{bmatrix} \tag{2.11c}$$

As seen before, the equation (2.5) is characterized by three eigenvalues. Those eigenvalues are defined by the matrices $\mathbf{A}_i$. For the one dimensional case, there is a unique definition of the eigenvalues, $\lambda_{1,2} = u \pm c$. In two dimensions, given the unit vector $\mathbf{e}$, the eigenvalues of the matrix $e_i \mathbf{A}_i$ are $\lambda_{1,3} = \mathbf{e} \cdot \mathbf{u} \pm c$ and $\lambda_2 = \mathbf{e} \cdot \mathbf{u}$. The eigenvalues are real and always different ($\lambda_1 < \lambda_2 < \lambda_3$), this property is called strictly hyperbolicity [149]. The eigenvalues are velocities, namely the ones of surface waves on the fluid. Note that in the dry zones, where $h = 0$, the eigenvalues coincide and the system is no longer hyperbolic. This introduces difficulties at theoretical and numerical level.

The vectors $\mathbf{b}_i$ are the result of the linearization of the topography using the same procedure taken for $\mathbf{A}_i$. They are obtained by the linearization of the fluxes $\mathbf{F}_i$ respect to the topography coordinate z. Rearranging terms with the independent vector $\mathbf{Q}$ yields

$$\mathbf{b}_i = \begin{bmatrix} \delta_{i1} c^2 \\ \delta_{i2} c^2 \\ 0 \end{bmatrix} \tag{2.12}$$

Analogously, the viscous fluxes $\mathbf{G}_i$ are rewritten in a more convenient manner as $\mathbf{G}_i = \mathbf{K}_{ij} \partial \boldsymbol{\phi} / \partial x_j$. The fourth order tensor $\mathbf{K}_{ij}$ is obtained making use of equation (2.8). It is an auxiliary variable to write the linearized tensor in Voigt's notation. This tensor will be defined latter, in the numerical model section.

The bottom friction term acting on the source term vector is linearized using a reaction matrix $\mathbf{S}$

$$\mathbf{S} = \begin{bmatrix} \frac{g n^2 |\mathbf{u}|}{h^{4/3}} & 0 & 0 \\ 0 & \frac{g n^2 |\mathbf{u}|}{h^{4/3}} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{2.13}$$

In the following sections, the rain, the atmospheric pressure and the wind friction will be neglected.

2.3 Shallow water equations with primitive variables

The presented, conservative, form of the shallow water equations –Saint-Venant equations– is generally applicable. However, many variations are present in the literature. The most common simplification is to express those equations in terms of the reduced or primitive variables –velocity instead of flow rate–. Another alternative is to use relative variables, taking the free surface instead of the total water depth.

The primitive variables simplification reduces the non-linearity of the equations, but its range of applicability is reduced. Particularly, the momentum would not be conserved in a change of regime, like an hydraulic jump. This fact is related to the evaluation of the convective fluxes, which depend on the flow rate gradient. While the evaluation of the flow rate gradient using conservatives variables does not present any problem, this operation is ill-conditioned when primitive variables are used. In other words, in a change of regime there is a discontinuity in both velocity and water depth, and the computation of the flow rate gradient involves the division of two gradients tending to $\pm\infty$.

In spite of this accuracy limitation near shocks, the non linearity reduction makes the primitive variables interesting from the numerical point of view, since less iteration will be needed to achieve convergence. Furthermore, good results are obtained for flows at low Froude numbers, such as estuaries, tidal currents or waves propagation.

2.3.1 Equations and linearization

The non linearity of the SW equations can be reduced if they are expressed in terms of the velocity. The primitive SW equations can be obtained by replacing the mass balance equation into the momentum balance and expanding the derivatives:

$$\frac{\partial \boldsymbol{\psi}}{\partial t} + \frac{\partial \mathbf{F}_i}{\partial x_i} + \mathbf{Q} = \mathbf{0} \tag{2.14a}$$

$$\boldsymbol{\psi} = \left\{ \begin{array}{c} \mathbf{u} \\ h \end{array} \right\} \tag{2.14b}$$

$$\mathbf{F}_i = \left\{ \begin{array}{c} u_1 u_i + \delta_{1i} g(h - z_b) \\ u_2 u_i + \delta_{2i} g(h - x_b) \\ u_i h \end{array} \right\} \tag{2.14c}$$

$$\mathbf{Q} = \left\{ \begin{array}{c} gS_1 \\ gS_2 \\ 0 \end{array} \right\} \tag{2.14d}$$

The same linearization procedure can be applied if the variables ψ are smooth enough. The following quasi-linear formulation will be obtained after applying the chain rule,

$$\frac{\partial \boldsymbol{\psi}}{\partial t} + \mathbf{A}_i \frac{\partial \boldsymbol{\psi}}{\partial x_i} + \mathbf{S}\boldsymbol{\psi} + \mathbf{b}_i \frac{\partial z}{\partial x_i} = 0 \tag{2.15}$$

where the tangent matrices $\mathbf{A}_i$ have been defined according to the differentiation of the convective fluxes with respect to the unknowns, $\partial \mathbf{F}_i / \partial \boldsymbol{\psi}$. Analogously, the same procedure is applied to the topography terms and to the bottom friction. The expression of the tangent matrices is

$$\mathbf{A}_1 = \begin{bmatrix} u_1 & 0 & g \\ 0 & u_2 & 0 \\ h & 0 & u_1 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} u_1 & 0 & 0 \\ 0 & u_2 & g \\ 0 & h & u_2 \end{bmatrix} \tag{2.16}$$

The same boundary conditions (2.2.1) are applied to the shallow water equations written in primitive variables. The only difference is the substitution of the flow rate by the velocity and that the flow rate is then expressed in terms of the velocity and the hater depth.

2.4 Boussinesq modified equations

The presented SW equations are suited to solve convective flows as well as free surface waves. Both phenomena are present in the hyperbolic equations. However, as long as the water depth increases, the oscillatory phenomenon or amplitude dispersion increases its relative importance. A new mechanism not included in the SW equations appears in a wave dominated problem, the frequency dispersion [157]. It is thereby necessary to quantify the effect of the new mechanism.

According to the classification of Peregrine [144], the dimensionless numbers of non-linearity and dispersion relate the wave amplitude η, the water depth H and the characteristic wavelength λ as:

$$\epsilon = \frac{\eta}{H}, \quad \mu = \frac{H}{\lambda} \tag{2.17}$$

Both parameters allow to link the concepts of amplitude and frequency dispersion. These concepts define how the wave propagates. Is well known that a wave propagates at speed $c = \sqrt{gH}$, but due to the convective term, this speed also depends on the wave amplitude, then introducing a non-linearity. Thus, considering the non-linearity, the wave speed increases as $c = \sqrt{g(H+\eta)}$. This phenomenon is known as amplitude dispersion and a first consequence is that every wave will end breaking, since the wave crest propagates faster than the wave bosom. The importance of the amplitude dispersion is related to the non-linearity ratio ϵ.

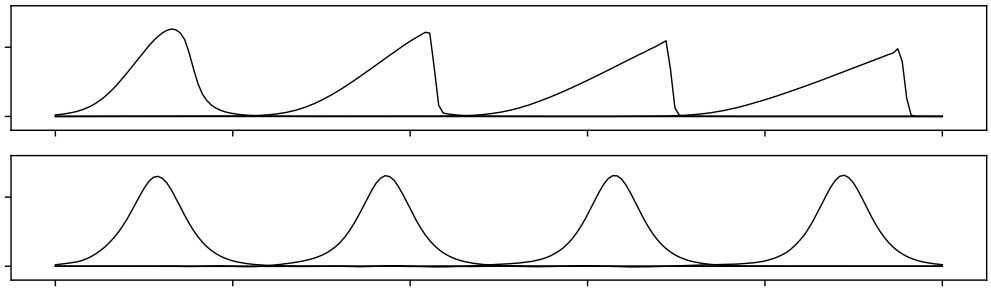


FIGURE 2.2: Boussinesq equations. Snapshots of a wave propagation.
Top: Amplitude dispersion. Bottom: Frequency dispersion

In practice, this phenomenon does not happen. From linear wave theory we know that the celerity depends not only on the water depth, but also on the wavenumber $k = 2\pi/\lambda$. This mechanism is known as frequency dispersion, which is missing on the Saint-Venant equations. Figure 2.2 shows the two possible types of wave propagation, with apmplitude dispersion or frequency dispersion. The introduction of some extra terms leads to the Boussinesq equations, which model the frequency dispersion. Once the frequency dispersion is included, classical soliton waves can be obtained. In a soliton wave, breaking never occurs during the propagation, since the non linear terms are in equilibrium with the dispersive terms.

2.4.1 Derivation of the modified Boussinesq equations

There are different ways to derive the Boussinesq equations with slightly different results. Nwogu presented a general framework in [136]. A three dimensional wave field with free surface elevation $\eta(x_1, x_2, t)$ and water depth at rest $H(x_1, x_2)$ is considered. The fluid is governed by the Navier-Stokes equations and the shallow water

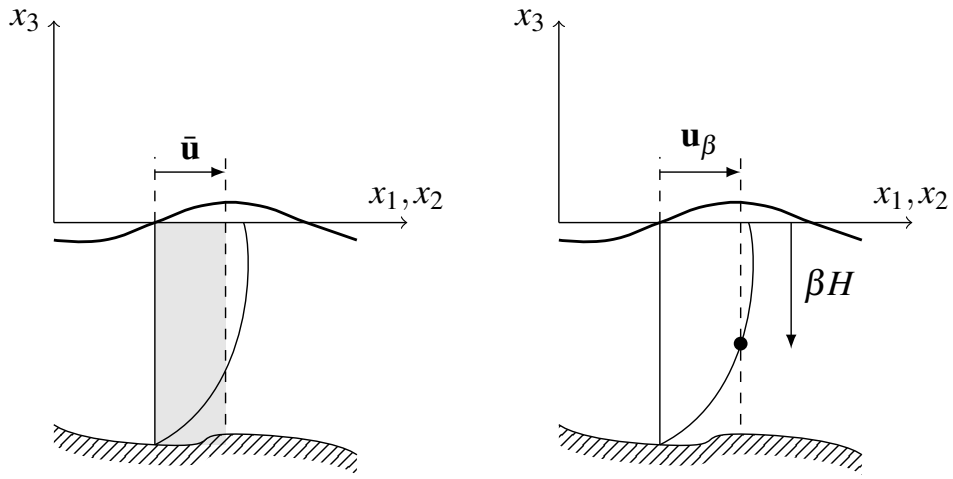


FIGURE 2.3: The modified Boussinesq equations consider the velocity at an arbitrary depth βH instead of the mean velocity

assumptions are modified. The fluid is assumed to be incompressible and the flow is assumed to be irrotational. As in the shallow water equations, the vertical velocity is considered to be small, but the vertical acceleration is not negligible. The nonlinearity and dispersion ratios (2.17) are assumed to be small. The last difference between the shallow water equations and the Boussinesq consist in considering the horizontal velocity at a specified water depth instead of the mean value (see Fig. 2.3).

Finally, the fluid at the free surface has to satisfy a dynamic and kinematic boundary conditions and a kinematic boundary condition at the bottom.

$$\begin{aligned} p &= 0\,, && \text{at } x_3 = \eta \\ u_3 &= \frac{\partial \eta}{\partial t} + u_1 \frac{\partial \eta}{\partial x_1} + u_2 \frac{\partial \eta}{\partial x_2}\,, && \text{at } x_3 = \eta \\ u_3 &= -u_1 \frac{\partial H}{\partial x_1} - u_2 \frac{\partial H}{\partial x_2}\,, && \text{at } x_3 = -H \end{aligned} \tag{2.18}$$

Then, the continuity and momentum equations are integrated from the bottom to the free surface and applying the boundary conditions (2.18). Since the average horizontal velocity has been substituted by the velocity at a certain depth, the vertical profile of the velocities must be known. The key of the Boussinesq equations consist in finding an assumption which preserves the the effect of the frequency dispersion. The horizontal velocities $\mathbf{u} = (u_1, u_2)$ are expanded in Taylor series from the seabed ($x_3 = -H$),

$$\begin{aligned} \mathbf{u}(x_1, x_2, x_3, t) = \mathbf{u}(x_1, x_2, -H, t) + (z+H)\mathbf{u}_3(x_1, x_2, -H, t) \\ + 1/2(z+H)^2 \mathbf{u}_{3,3}(x_1, x_2, -H, t) + \dots \end{aligned} \tag{2.19}$$

where the 3 subscript denotes differentiation with respect to x_3. Finally, the equations are evaluated at an arbitrary depth $x_3 = \beta H$ and the set of Boussinesq-type equations are:

$$\frac{\partial \mathbf{u}_\beta}{\partial t} + \nabla \eta + (\mathbf{u}_\beta \cdot \nabla)\mathbf{u}_\beta + \mathbf{J}_u = \mathbf{0} \tag{2.20a}$$

$$\frac{\partial \eta}{\partial t} + \nabla \cdot \left((H + \eta) \mathbf{u}_\beta \right) + \nabla \cdot \mathbf{J}_\eta = 0 \tag{2.20b}$$

where the auxiliary fields $\mathbf{J}_\eta$ and $\mathbf{J}_u$ introduce the dispersive mechanism and are defined according to the following expressions

$$\mathbf{J}_\eta = C_1 H^3 \nabla\nabla \cdot \mathbf{u}_\beta + C_3 H^2 \nabla\nabla \cdot (H \mathbf{u}_\beta) \tag{2.21a}$$

$$\mathbf{J}_u = C_2 H^2 \nabla\nabla \cdot \frac{\partial \mathbf{u}_\beta}{\partial t} + C_4 H \nabla\nabla \cdot \frac{\partial (H \mathbf{u}_\beta)}{\partial t} \tag{2.21b}$$

and the C_i constants depend on the choice of β

$$C_1 = \frac{1}{2}\left(\beta^2 - \frac{1}{3}\right), C_2 = \frac{\beta^2}{2}, C_3 = \beta + \frac{1}{2}, C_4 = \beta \tag{2.22}$$

2.4.2 Dispersion properties and range of applicability

These equations present as a free parameter β the relative elevation at which the velocity is evaluated. Its value ranges from -1 at the seabed, to 0 at the free surface. Since the equations are an approximation of the fully dispersive and nonlinear problem, the parameter β is chosen to minimize the errors introduced by the approximation. In fact, the original Boussinesq equations do not present any dispersive term in the mass balance equation (2.20b) and correspond to a specific choice of β.

The parameter β is fixed to -0.531 in [136]. This value has been obtained by reducing the equations (2.20) to one dimension and flat bottom. Then, a trial function of small amplitude periodic wave with wavenumber k and frequency ω of the type

$$\eta = a_0 \exp(i(kx - \omega t)), \quad u_\beta = u_0 \exp(i(kx - \omega t)) \tag{2.23}$$

is substituted into (2.20b) and (2.20a). After some algebraic manipulation the following expression for the phase speed is obtained:

$$c^2 = \frac{\omega^2}{k^2} = gh \left(\frac{1 - \left(\frac{1}{2}\beta^2 + \beta + \frac{1}{3}\right)(kh)^2}{1 - \left(\frac{1}{2}\beta^2 + \beta\right)(kh)^2} \right) \tag{2.24}$$

The relation between the frequency and the wavelength is also known as *dispersion relation*. On the other hand, the dispersion relation given by the Linear wave theory or Airy theory is given by

$$c^2 = gh \frac{\tanh kh}{kh} \tag{2.25}$$

Finally, the value of β has been chosen to minimize the error at the range of applicability. Some other classical values of beta were obtained in [117, 128] and are shown in Figure 2.4. Figure 2.4 shows the sensitivity of the phase speed c depending on the parameter β with respect to the phase speed obtained with Linear theory c_{Airy}. The shallow water equations, which drop the dispersive terms of equations (2.20) are only valid for the shallow water regime ($kh < 0.3$). Fixing the free parameter $\beta = -0.531$ extends the range of applicability of the Boussinesq equations to intermediate depths ($kh < 3$).

According to [157] the nonlinearity and dispersion parameters ϵ and μ can be used for an alternative classification:

$\epsilon \ll \mu$ This configuration correspond to the case where frequency dispersion dominates the problem and linear or Airy theory must be used.

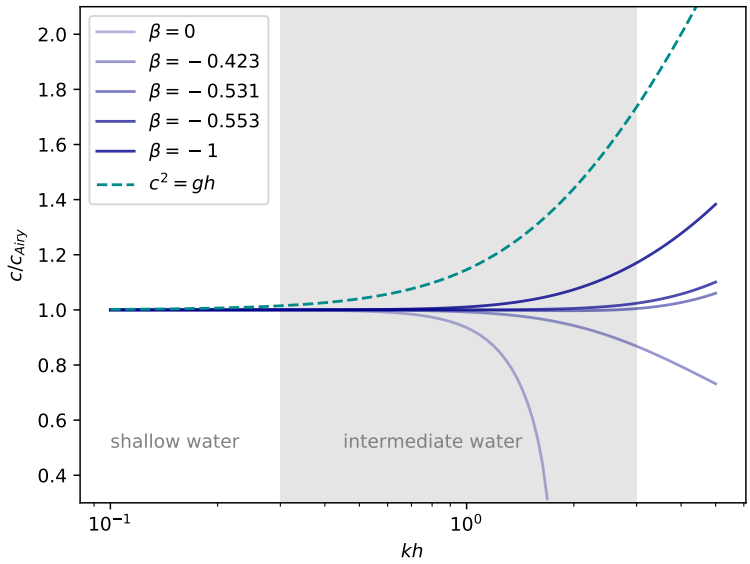


FIGURE 2.4: Comparison of normalized phase speeds of the Boussinesq modified equations for different values of β

$\epsilon \sim \mu$ In this case frequency and amplitude dispersion are of the same magnitude and the Boussinesq approximation can be used.

$\epsilon \gg \mu$ In this situation the amplitude dispersion dominates the problem and the wave will eventually break. This can be simulated using the Saint-Venant or shallow water equations. Since the vertical velocity is assumed negligible, the pressure distribution is assumed to be constant.

2.4.3 Boundary conditions

The boundary conditions presented in section 2.2.1 are applicable to the Boussinesq equations. However, since the oscillatory behavior is usually prevalent over the convective phenomenon, the boundary conditions are slightly different, both from the physical and the numerical point of view. The three types of boundary conditions considered for the Boussinesq problems, inflow Γ_I, reflecting Γ_R and absorbing Γ_A boundaries have a direct equivalence respectively with the inflow Γ_I, solid Γ_S and outflow Γ_O boundaries stated in section 2.2.1. Those subdomains are such that $\Gamma_I \cup \Gamma_R \cup \Gamma_A = \partial\Omega$, being $\partial\Omega$ the boundary of the domain.

Reflecting boundary No fluid should pass through an impermeable wall. This implies imposing the normal component of the velocity to be zero.

$$\bar{\mathbf{u}} \cdot \mathbf{n} = 0 \quad \text{on } \Gamma_R$$

Following Woo and Liu [161], the above relation must be rewritten in terms of $\mathbf{u}_\beta$ and the velocities are related as

$$\bar{\mathbf{u}} = \mathbf{u}_\beta + H^{-1}\mathbf{J}_\eta$$

Hence, the complete formulation of a reflective boundary is

$$\bar{\mathbf{u}}_\beta \cdot \mathbf{n} = 0\,, \quad \mathbf{J}_\eta \cdot \mathbf{n} = 0 \quad \text{on } \Gamma_R \tag{2.26}$$

Inflow boundary Both free surface and velocity are known at Γ_I. Typically it is used to impose a wave generator. Since the wave amplitude is known, the horizontal velocity can be obtained using linear wave theory.

$$\bar{\mathbf{u}} \cdot \mathbf{n} = u' \quad \text{or} \quad \eta = \eta' \quad \text{on } \Gamma_I$$

Since the Boussinesq equations are used to solve oscillatory problems, typically in a subcritical regime, only one variable should be imposed. In the case of a supercritical regime, both variables must be imposed in order to define a well-posed problem. Whereas the velocity is imposed, the same procedure with the dispersive term than in Γ_R should be considered. The corresponding $\mathbf{J}_\eta$ must be computed. Imposing $\mathbf{J}_\eta \cdot \mathbf{n} = 0$ is another possibility and it corresponds to consider $\bar{\mathbf{u}} = \mathbf{u}_\beta$.

Absorbing boundary Absorbing boundaries are known as the approximation of open boundaries. An outgoing wave should not return to the computational domain through Γ_A. A practical implementation of absorbing boundaries are sponge layers [92, 160] and will be explained in section 4.2

Chapter 3

Finite Element Methods for the shallow water equations

Due to the complexity of the geometry and the source terms, it is not possible to find an analytical solution for the SW equations. This explains the need to design strategies to find numerical solutions. In this chapter some numerical strategies with the FEM in an eulerian framework are explored. The Eulerian frameworks are robust and very efficient for continuous problems. However, discontinuities and front tracking may require especial attention. Since flooding or moving shoreline are part of the physical phenomena of interest, the identification of the dry domain is a challenging problem for the Eulerian FEM. Moreover, monotonic properties are especially interesting when partially wet domains are considered because the water depth is a positive magnitude but, in general, numerical methods do not verify this property.

The SW equations have traditionally been modelled using finite volumes (FV) because of its advantages of stability and monotonicity. Given its geometric flexibility and its natural way to introduce high order schemes, the FEM has been applied too [132, 133, 171]. Halfway between FV and FEM, there is the discontinuous Galerkin (DG) technique [6, 95, 107]. DG method has the advantages of the geometrical flexibility of the FEM and the stability of FV, but the introduction of high order DG schemes is not straightforward. Since the FEM can exhibit spurious oscillations, different strategies such as stabilization, monotonic schemes or different order of polynomial interpolation can be explored [81, 137, 171].

In this chapter, the general procedure for FEM is presented. Then, some stabilization techniques are explored: the *Finite Increment Calculus* (FIC) stabilization, the *Flux Correction* (FC) technique and the *Gradient jump viscosity* (GJV) methods. Several examples are provided to show the capabilities and limitations of each method.

3.1 Galerkin weak formulation

We consider a balance equation in the space domain Ω and the time interval $[0, T]$. Let $\boldsymbol{\phi}$ and W be vector functions of the domain $\Omega \times [0, T]$ and $\mathbf{r}$ the residual of the balance equations expressed in terms of the unknown $\boldsymbol{\phi}$. The space of functions that are square-integrable in Ω is denoted as $L^2(\Omega)$. And the space of functions whose derivatives up to order $m \geq 0$ belong to $L^2(\Omega)$ is denoted by $H^m(\Omega)$. The space $H_0^1(\Omega)$ consists on a subspace of $H^1(\Omega)$ vanishing on $\partial\Omega$.

Using this notation, the space of functions for the continuous problem are $V := H^1(\Omega)$ and for the test functions $V_0 := H_0^1(\Omega)$. The variational form of the balance

equations can be expressed as find $\boldsymbol{\phi} \in V$ such that $\forall W \in V_0$

$$\int_\Omega W \cdot \mathbf{r}(\boldsymbol{\phi})\, d\Omega = 0 \tag{3.1}$$

The standard Galerkin approximation 3.1 is straightforward. Let $\mathcal{P}_h$ be a partition of the domain Ω. The diameter of an element domain $e \in \mathcal{P}_h$ is denoted by l_e. From the polynomial spaces defined by the finite elements, we can construct the subspaces $V_h \in V$ and $V_{h,0} \in V_0$. The Galerkin approximation of the balance equations can be expressed as find $\boldsymbol{\phi}_h \in V_h$ such that $\forall W_h \in V_{h,0}$

$$\int_\Omega W_h \cdot \mathbf{r}(\boldsymbol{\phi}_h)\, d\Omega = 0 \tag{3.2}$$

Unfortunately, this problem is not straightforward to solve using finite elements, as its discrete version is not numerically stable. In fact, to ensure that the problem of finding $\boldsymbol{\phi}_h$ has a stable solution, the space must verify the inf-sub condition [36].

3.2 Stabilized formulations for the shallow water equations

Several families of stabilization methods can be found in the literature, usually applied to the convection-diffusion equations and the Navier-Stokes equations. The most relevant are SUPG [22], ASGS[35], GLS [83] and FIC[138, 141]. Due to the hyperbolic character of the SW equations, a particular stabilization method for compressible flow or the Euler equations need to be developed. The FIC approach is based on the incremental solution of a modified system of non-local governing equations accounting for higher order terms obtained by applying the balance laws in domains of finite size. The FIC-based stabilization has been applied in conjunction with the FEM to convection-diffusion and incompressible flows, and solid mechanics [58, 141, 142]. In those cases, where the convective term has an important role, a first order FIC term is enough to provide stability to the system. However, the SW equations are governed by the convective term and the wave equation in a mixed formulation [37]. In consequence, the common derivation of the FIC-based stabilization is not enough to provide stability in all the range of applicability of the SW equations. A generalization of this method is proposed in order to provide a global stability for the SW equations.

Once global stability is achieved, local instabilities may appear near discontinuities, which are inherent to the supercritical flows. A local shock capturing technique was initially proposed by Hughes [85] and a review of shock capturing techniques can be found in Codina [36]. Other possibilities of the FIC-based formulations are explored to provide a shock capturing stabilization [43].

Additionally, the dry domain requires an accurate modeling because the hyperbolic equations require positive water depth in all the domain. Several authors have proposed different methods to solve the shallow water equations with moving shoreline. Leclerc et al. [106] proposed an Eulerian method. Later, Heniche et al. [78] modified the method allowing the free surface to plunge under the topography. Other authors developed a rough-porous layer [11, 28] or a modified depth integration [51]. These approaches introduce new physical parameters in the balance equations. An Eulerian approach based on the work of [106] and [78] is presented in this thesis.

3.2.1 FIC stabilization

Let us start considering the quasi-linear balance equations written in residual form as a vector

$$\mathbf{r} := \frac{\partial \boldsymbol{\phi}}{\partial t} + \mathbf{A}_i \frac{\partial \boldsymbol{\phi}}{\partial x_i} - \frac{\partial}{\partial x_i}\left(\mathbf{K}_{ij}\frac{\partial \boldsymbol{\phi}}{\partial x_j}\right) + \mathbf{S}\boldsymbol{\phi} + \mathbf{b}_i \frac{\partial z}{\partial x_i} \qquad i,k \in \{1, n_d\} \tag{3.3}$$

where $n_d = 2$ is the number of dimensions. The size of the vector $\mathbf{r}$ is equal to the number of balance equations, $n_b = 3$.

In the one dimensional case ($n_d = 1$) and scalar balance ($n_b = 1$), the FIC-based stabilization is based on a modified non local version of the governing equations [141], where the fluxes are expanded using a Taylor approximation. The modified residual reads

$$r - \frac{1}{2} l^e \frac{\partial r}{\partial x} = 0 \tag{3.4}$$

The stabilization parameter l^e is usually taken the element length. However, in 2D and 3D, or when the number of balance equations n_b is different than n_d, the choice of the l^e parameter is non trivial. Several approaches can be found in the literature. In [141] l^e is chosen as a vector, but in later publications such as [142] a generalized formulation for different values of n_d and n_b was presented. For the stabilization of the Navier-Stokes equations different projections of the element size over the velocity and over the velocity gradient have been proposed [43]. Here we will use index notation for the residual vector $\mathbf{r}$ in order to distinguish the indices that goes to n_d or to n_b. To sum up, the different forms of the FIC stabilization procedure can be written as

$$r_k - \frac{1}{2} l^e_i \frac{\partial r_k}{\partial x_i} = 0 \qquad i \in \{1, n_d\}\ ,\ k \in \{1, n_b\} \tag{3.5a}$$

$$r_k - \frac{1}{2} l^e_u \frac{u_i}{\|\mathbf{u}\|} \frac{\partial r_k}{\partial x_i} = 0 \qquad i,k \in \{1, n_d\} \tag{3.5b}$$

$$r_k - \frac{1}{2} l^e_{g_i} \frac{\partial \mathbf{u}/\partial x_i}{\|\nabla u_i\|} \frac{\partial r_k}{\partial x_i} = 0 \qquad i,k \in \{1, n_d\} \tag{3.5c}$$

We propose a stabilization term which is oriented along the characteristics of the hyperbolic equations, as

$$r_k - \frac{1}{2} l^e \frac{\mathbf{A}_i}{\lambda} \frac{\partial r_k}{\partial x_i} = 0 \qquad i \in \{1, n_d\}\ ,\ k \in \{1, n_b\} \tag{3.6}$$

For consistency the linearization matrix $\mathbf{A}_i$ is normalized with the maximum eigenvalue $\lambda = |\mathbf{u}| + c$. This stabilization is analogue to the virtual multi-scale stabilization proposed in [39]. The linearization matrix $\mathbf{A}_i$ provides a weighting procedure between the stabilization of the convective and the mixed wave equation terms. In practice the element size is multiplied by an algorithmic constant in order to control the amount of diffusion added by the stabilization and it will be studied in the examples of Section 3.5. Recovering the vector notation for the residual, the FIC-balance reads

$$\mathbf{r} - \beta l^e \frac{\mathbf{A}_i}{\lambda} \frac{\partial \mathbf{r}}{\partial x_i} \qquad i \in \{1, n_d\} \tag{3.7}$$

The FIC formulation is the result of introducing the residual of the shallow water equations (3.3) into the expression in Eq (3.7). The variational expression of the

equation is obtained by multiplying the equation by a test function ω_k and integrating over the domain Ω. This gives

$$\int_\Omega \left(\omega_k \mathbf{r} - \omega_k \beta l^e \frac{\mathbf{A}_i}{\lambda} \frac{\partial \mathbf{r}}{\partial x_i} \right) d\Omega = 0 \tag{3.8}$$

The second term of Equation (3.8) is integrated by parts. Note that the element length l^e, the linearization matrix $\mathbf{A}_i$ and its eigenvalue λ are defined constant inside the element. Hence, the boundary integral which appears after integration by parts should be understood as the boundary of all the elements

$$\int_\Omega \omega_k \mathbf{r} d\Omega + \int_\Omega \beta l^e \frac{\mathbf{A}_i}{\lambda} \frac{\partial \omega_k}{\partial x_i} \mathbf{r} d\Omega - \sum_e \int_{\Gamma_e} \beta l^e \frac{\mathbf{A}_i}{\lambda} \omega_k n_k \mathbf{r} d\Gamma = 0 \tag{3.9}$$

In this work we neglect the boundary integrals assuming that the residual $\mathbf{r}$ is null at the boundary of the elements. At this point we introduce the balance Equation (3.3) and integrate by parts the diffusive term. Derivatives of order higher tan two will be neglected since we are using linear triangles. The result is

$$\begin{aligned} \int_\Omega \left(\omega_k \frac{\partial \boldsymbol{\phi}}{\partial t} + \omega_k \mathbf{A}_i \frac{\partial \boldsymbol{\phi}}{\partial x_i} + \frac{\partial \omega_k}{\partial x_i} \mathbf{K}_{ij} \frac{\partial \boldsymbol{\phi}}{\partial x_j} + \omega_k \mathbf{S} \boldsymbol{\phi} + \omega_k \mathbf{b}_i \frac{\partial z}{\partial x_i} \right) d\Omega \\ + \int_\Omega \frac{\beta l^e}{\lambda} \left(\frac{\partial \omega_k}{\partial x_j} \mathbf{A}_j \frac{\partial \boldsymbol{\phi}}{\partial t} + \frac{\partial \omega_k}{\partial x_j} \mathbf{A}_j \mathbf{A}_i \frac{\partial \boldsymbol{\phi}}{\partial x_i} + \frac{\partial^2 \omega_k}{\partial x_j^2} \mathbf{A}_j \mathbf{K}_{ij} \frac{\partial \boldsymbol{\phi}}{\partial x_i} \right. \\ \left. + \frac{\partial \omega_k}{\partial x_j} \mathbf{A}_j \left(\mathbf{S} \boldsymbol{\phi} + \mathbf{b}_i \frac{\partial z}{\partial x_i} \right) \right) d\Omega = 0 \end{aligned} \tag{3.10}$$

Equation (3.10) is the stabilized variational form for the shallow water equations, similar to the expression obtained by SUPG. Note that the parameter $\beta l^e / \lambda$ is analogous to the characteristic time τ of the classical SUPG or GLS techniques [43].

3.2.1.1 Shock capturing stabilization

In this section we explore other possibilities of the characteristic length definition in order to obtain a shock capturing stabilization. Here, the mass balance and the momentum balance are considered separately and the characteristic length is projected onto the gradient of the unknown

$$\text{Momentum balance:} \qquad r_i^q - \frac{l^e}{2\|\nabla q_i\|} \frac{\partial q_i}{\partial x_j} \frac{\partial r_i^q}{\partial x_j} = 0 \tag{3.11a}$$

$$\text{Mass balance:} \qquad r^h - \frac{l^e}{2\|\nabla h\|} \frac{\partial h}{\partial x_j} \frac{\partial r^h}{\partial x_j} = 0 \tag{3.11b}$$

Multiplying the momentum balance Equation (3.11a) by a proper test function ω_k, integrating over the domain in the same way as in Equation (3.8), one obtains the following variational form:

$$\int_\Omega \omega_k r_i^q d\Omega - \int_\Omega \omega_k \frac{l^e}{2\|\nabla q_i\|} \frac{\partial q_i}{\partial x_j} \frac{\partial r_i^q}{\partial x_j} d\Omega = 0 \tag{3.12}$$

After integration of Equation (3.12) by parts and rearranging terms we obtain

$$\int_\Omega \omega_k r_i^q d\Omega + \int_\Omega \frac{\partial \omega_k}{\partial x_j} \frac{l^e r_i^q}{2\|\nabla q_i\|} \frac{\partial q_i}{\partial x_j} d\Omega + \int_\Omega \omega_k \frac{\partial}{\partial x_j} \left(\frac{l^e}{2\|\nabla q_i\|} \frac{\partial q_i}{\partial x_j} \right) r_i^q d\Omega - \int_\Omega \frac{\partial}{\partial x_j} \left(\omega_k \frac{l^e}{2\|\nabla q_i\|} \frac{\partial q_i}{\partial x_j} r_i^q \right) d\Omega = 0 \quad (3.13)$$

Since we will use linear triangles, the last two terms of Equation (3.13) are dropped because they involve derivatives of the characteristic length and can be transformed into a boundary integral.

The same procedure is applied to the mass balance equation (3.11b). As a result we obtain the following system of equations for both unknowns

$$\text{Momentum balance:} \quad \int_\Omega \omega_k r_i^q d\Omega + \int_\Omega \frac{\partial \omega_k}{\partial x_j} \frac{l^e r_i^q}{2\|\nabla q_i\|} \frac{\partial q_i}{\partial x_j} d\Omega = 0 \quad (3.14a)$$

$$\text{Mass balance:} \quad \int_\Omega \omega_k r^h d\Omega + \int_\Omega \frac{\partial \omega_k}{\partial x_j} \frac{l^e r^h}{2\|\nabla h_i\|} \frac{\partial q_i}{\partial x_j} d\Omega = 0 \quad (3.14b)$$

The above expressions (3.14) are equivalent to a classical shock capturing method, in which the artificial diffusivity k_{art} and artificial viscosity ν_{art} can be identified as

$$\nu_{art} = \frac{1}{2} \alpha l_e \frac{|r_i^q|}{\|\nabla u_i\|} \quad (3.15a)$$

$$k_{art} = \frac{1}{2} \alpha l_e \frac{|r^h|}{\|\nabla h\|} \quad (3.15b)$$

where α is an algorithmic constant.

Such approach can be refined by introducing the stabilization along the streamlines. This way, k_{art} and ν_{art} need to be added only in the crosswind direction. The diffusive term is added to the mass balance with the following orthogonal tensor

$$\mathbf{D}_{art} = k_{art} \left(\mathbf{I} - \frac{1}{|\mathbf{u}|^2} \mathbf{u} \otimes \mathbf{u} \right) \quad (3.16)$$

The viscosity is introduced into the momentum balance with a fourth order tensor in the crosswind direction. Using Voigt's notation,

$$\mathbf{C}_{art} = \nu_{art} \mathbf{I}_4 \mathbf{J} \quad (3.17)$$

with

$$\mathbf{J} = \begin{bmatrix} 1 - \frac{q_1 q_1}{\mathbf{q}\mathbf{q}} & -\frac{q_1 q_2}{\mathbf{q}\mathbf{q}} & 0 \\ -\frac{q_1 q_2}{\mathbf{q}\mathbf{q}} & 1 - \frac{q_2 q_2}{\mathbf{q}\mathbf{q}} & 0 \\ 0 & 0 & 1 - \frac{q_1 q_2}{\mathbf{q}\mathbf{q}} \end{bmatrix} \quad (3.18)$$

where $\mathbf{I}_4$ is the fourth order identity tensor for the stresses, which is derived from Equation (2.8) and will be defined in Section 3.2.2.

3.2.2 Finite element formulation

It is conventional to use a higher order of interpolation for the momentum or velocity than for the water depth or free surface in order to develop stable finite element formulations [13, 78, 81]. In this work we restrict ourselves to linear triangles for

both $\mathbf{q}$ and h unknowns, since the FIC-FEM procedure is intrinsically stable. For that reason, all terms including spatial derivatives of order higher than two will be neglected. Bilinear quadrilaterals and higher order elements with the same number of degrees of freedom for all the variables will be also stable.

3.2.2.1 Spatial discretization

Now the variational problem is expressed by its discrete counterpart to obtain the algebraic formulation. The problem variables are interpolated with the basis functions of the finite elements space as

$$\phi_i = \sum_a^{n_\Omega} N_a(\mathbf{x})\phi_{ai} \qquad i \in \{1, n_b\} \tag{3.19}$$

where n_Ω represents the total number of nodes in Ω_h and ϕ_i are the problem variables defined in (2.6). Note that the shape functions are the same for all the variables, h and q_i. Here we introduce the notation $\boldsymbol{\phi}_h$ for the vectors of nodal unknowns -momentum and water height- on the finite element domain. Following the standard Galerkin discretization, the shape functions N_a are used to interpolate the test functions ω_k and the unknowns. The continuous equation (3.10) is combined with equation (3.14) and can be expressed as the following algebraic system of equations

$$[\mathbf{M} + \mathbf{M}_F]\dot{\boldsymbol{\phi}}_h + [\mathbf{G} + \mathbf{G}_F + \mathbf{L} + \mathbf{L}_{SC} + \mathbf{R} + \mathbf{R}_F]\boldsymbol{\phi}_h = \mathbf{T} + \mathbf{T}_F \tag{3.20}$$

where the dot ($\dot{}$) means temporal derivative. The matrices in Eq. (3.20) without subscript are related to the original problem (3.3); the matrices with subscript F correspond to the terms added by the FIC procedure to ensure stability, and those with the subscript SC are the terms added by the shock capturing technique. Using a, b to denote the nodes, i, j to denote the space dimension index and k, l to denote the balance equation number, the matrices in Eq. (3.20) are defined as

$$\begin{aligned}
\mathbf{M}^{ab} &= \int_{\Omega_e} N_a \mathbf{I} N_b d\Omega & \mathbf{G}^{ab} &= \int_{\Omega_e} N_a \mathbf{A}_i \frac{\partial N_b}{\partial x_i} d\Omega \\
\mathbf{L}^{ab} &= \int_{\Omega_e} \mathbf{B}_a \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{D} \end{bmatrix} \mathbf{B}_b^T d\Omega & \mathbf{R}^{ab} &= \int_{\Omega_e} N_a \mathbf{S} N_b d\Omega \\
\mathbf{T}^{ab} &= \int_{\Omega_e} N_a \mathbf{b}_i \frac{\partial z}{\partial x_i} d\Omega + \int_{\Gamma_e} N_a \mathbf{t}_b d\Gamma
\end{aligned} \tag{3.21}$$

where the diffusive matrix $\mathbf{L}^{ab}$ is defined using the derivatives matrix $\mathbf{B}_a$ and the isotropic tensors $\mathbf{C}$ and $\mathbf{D}$ of viscosity and diffusivity. Note that the diffusivity is zero, but the matrix structure will be reused for the stabilization. The viscosity tensor in Voigt's notation is constructed using the linearization matrices $\mathbf{K}_{ij}$. The matrix and the tensors are given by

$$\mathbf{B}_a = \begin{bmatrix} \frac{\partial N_a}{\partial x_1} & 0 & \frac{\partial N_a}{\partial x_2} & 0 & 0 \\ 0 & \frac{\partial N_a}{\partial x_2} & \frac{\partial N_a}{\partial x_1} & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial N_a}{\partial x_1} & \frac{\partial N_a}{\partial x_2} \end{bmatrix} \tag{3.22a}$$

$$\mathbf{C} = \nu \mathbf{I}_4 \,, \quad \mathbf{D} = k\mathbf{I}_2 \,, \quad \mathbf{I}_4 = \frac{1}{3}\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \,, \quad \mathbf{I}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{3.22b}$$

The stabilization and shock capturing terms from Equation (3.20) result in analogous matrices with higher derivatives order, the boundary integral is neglected, i. e.,

$$\begin{aligned}
\mathbf{M}_F^{ab} &= \int_{\Omega_e} \frac{\beta l^e}{2} \frac{\partial N_a}{\partial x_i} \mathbf{A}_i N_b d\Omega & \mathbf{G}_F^{ab} &= \int_{\Omega_e} \frac{\beta l^e}{2} \frac{\partial N_a}{\partial x_i} \mathbf{A}_i \mathbf{A}_j \frac{\partial N_b}{\partial x_j} d\Omega \\
\mathbf{L}_{SC}^{ab} &= \int_{\Omega_e} \mathbf{B}_a \begin{bmatrix} \mathbf{C}_{\text{art}} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_{\text{art}} \end{bmatrix} \mathbf{B}_b^T d\Omega & \mathbf{R}_F^{ab} &= \int_{\Omega_e} \frac{\beta l^e}{2} \frac{\partial N_a}{\partial x_i} \mathbf{A}_i \mathbf{S} N_b d\Omega \\
\mathbf{T}_F^{ab} &= \int_{\Omega_e} \frac{\beta l^e}{2} \frac{\partial N_a}{\partial x_i} \mathbf{A}_i \mathbf{b}_j \frac{\partial z}{\partial x_j} d\Omega
\end{aligned} \tag{3.23}$$

3.2.3 Temporal integration

The resulting expression from the spatial discretization (3.20) can be written in the following compact form

$$\tilde{\mathbf{M}} \dot{\boldsymbol{\phi}}_h + \tilde{\mathbf{K}} \boldsymbol{\phi}_h = \tilde{\mathbf{f}} \tag{3.24}$$

where the symbol ($\tilde{}$) denotes the assembly of the system matrices and vectors for all the elements. We have integrated this equation introducing a time discretization using the well known BDF2 implicit scheme [19, 48]. The system of equations in a discrete time domain yields

$$\begin{aligned}
\tilde{\mathbf{M}} \dot{\boldsymbol{\phi}}_h^{n+1} + \tilde{\mathbf{K}}^{n+1} \boldsymbol{\phi}_h^{n+1} &= \tilde{\mathbf{f}}^{n+1} \\
\dot{\boldsymbol{\phi}}_h^{n+1} &= \beta_0 \boldsymbol{\phi}_h^{n+1} + \beta_1 \boldsymbol{\phi}_h^{n} + \beta_2 \boldsymbol{\phi}_h^{n-1}
\end{aligned} \tag{3.25}$$

We will consider a variable time step to compute the BDF coefficients using the notation $t^{n+1} = t^n + \Delta t^n$:

$$\begin{aligned}
\beta_0 &= \tau(\rho^2 + 2\rho) \\
\beta_1 &= -\tau(\rho^2 + 2\rho + 1) \\
\beta_2 &= \tau
\end{aligned} \tag{3.26}$$

with

$$\begin{aligned}
\tau &= \frac{1}{\Delta t^n (\rho^2 + \rho)} \\
\rho &= \frac{\Delta t^{n-1}}{\Delta t^n}
\end{aligned} \tag{3.27}$$

The solution of this implicit system requires an iterative procedure. We have used the Newton-Raphson method, by which the problem unknowns are computed in an incremental way as $\boldsymbol{\phi}_h^{n+1,i+1} = \boldsymbol{\phi}_h^{n+1,i} + \delta \boldsymbol{\phi}_h^i$, where the superscript i denotes the non linear iteration. This notation allows us to rewrite the system of equations (3.25) defining a left hand side matrix multiplied by the increment $\delta \boldsymbol{\phi}_h^i$ and a right hand side vector which depends on the previous non linear iteration as

$$[\beta_0 \tilde{\mathbf{M}} + \tilde{\mathbf{K}}^{n+1,i}] \delta \boldsymbol{\phi}_h^i = \tilde{\mathbf{f}}^{n+1,i} - \tilde{\mathbf{K}}^{n+1,i} \boldsymbol{\phi}_h^{n+1,i} - \tilde{\mathbf{M}} \dot{\boldsymbol{\phi}}_h^{n+1,i} \tag{3.28}$$

The first non linear iteration $\boldsymbol{\phi}_h^{n+1,0}$ is initialized using a prediction given from the BDF formula at the last time step:

$$\boldsymbol{\phi}_h^{n+1,0} = \boldsymbol{\phi}_h^n + \Delta t^n \dot{\boldsymbol{\phi}}_h^n \tag{3.29}$$

3.2.4 Dry domain model

When small or quasi zero water depths are involved in simulations, some instabilities may arise. In addition, the solution of the time integration scheme requires the inverse of a matrix which is singular in the dry regions. In this section we review the challenges associated to such a problem, and the way we have circumvented them.

Recovery of the velocity field The evaluation of the characteristic matrices $\mathbf{A}_i$ involves the velocities, which are recovered given the primary variables $\boldsymbol{\phi}$ from the previous iteration. Since the computation of the velocity is the result of dividing the discharge by the water height, this operation is ill-conditioned in the dry regions. In this research, the velocity field is computed in a two step procedure. First of all, the inverse of the water depth is computed at each element following the next expression, initially proposed in [99]:

$$\hat{h}^{-1} := \frac{\sqrt{2}\max(h,0)}{\sqrt{h^4 + \max(h^4, \varepsilon^4)}} \tag{3.30}$$

where ε is a threshold which depends on the element size; usually $\varepsilon = 0.1 l_e$ is chosen. Figure 3.1 shows a dimensionless representation of equation (3.30). The second step in the velocity computation is a diffusive projection on the nodes:

$$\mathbf{M}_L \mathbf{u} = \hat{h}_k^{-1} \mathbf{M}(\mathbf{q}) \tag{3.31}$$

where $\mathbf{M}$ is the consistent mass matrix and $\mathbf{M}_L$ is the lumped mass matrix. This projection will introduce some artificial diffusion in the velocity field near the dry-wet interface reducing the possible maxima extrema.

The expression(3.30) tends to zero in dry or partially dry regions, while the analytical expression of the height inverse is recovered when $h > \varepsilon$.

Avoiding the singularity of the system matrix Since all the elements are included in the computational domain, the last issue to overcome with small water depths are the numerical difficulties stated in Section 2.2.2. When there is a null water depth, the theoretical flow rate and velocity are zero. In that point the matrices $\mathbf{A}_i$ are not invertible and the eigenvalues are all equal to zero. In that case, the hyperbolicity property of the system is lost. In practice, due to spurious oscillations, the flow rate and the velocity may be different from zero. The idea is to freeze the flow and to allow to invert the system matrix adding a diagonal of non zero terms to the momentum equation, i. e.,

$$\mathbf{G} := \mathbf{G} + \xi \,\mathrm{diag}(1,1,0) \tag{3.32}$$

The selection of the areas where there is a dry domain is controlled with the wet fraction function and ξ is defined as

$$\xi = k(1-w) \tag{3.33}$$

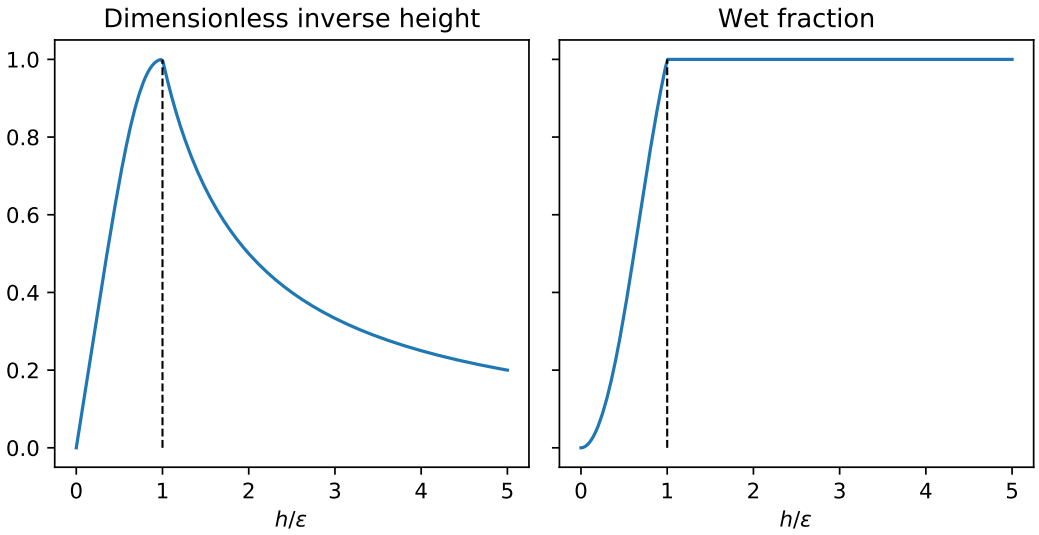


FIGURE 3.1: Dimensionless functions to compute the inverse height and the wet fraction.

In our numerical experiments we have chosen $k = 10^3$. The addition of a diagonal matrix remembers the artificial Manning friction proposed in [78]. This term plays the role of freezing the flow in dry areas.

Mass conservation properties The stabilized method proposed is not monotonic and the dry domain model is acting to ensure stability, but it does not provide monotonicity. We note that all the modifications have been done at the momentum balance level. This means that mass is conserved globally by the weak formulation, but the mass sign preservation is not guaranteed.

Both unknowns, water depth and flow rate, are continuous at the dry-wet interface, but its derivatives are discontinuous. Even though the shock capturing scheme can not avoid this kind of oscillations, it will mitigate them, and the order of accuracy will be lost due to the introduction of the non-linear artificial diffusion [9].

3.2.5 Validation

In this section the linear stabilization is tested. Conceptual examples are presented only to show the consistency of the linear stabilization and the lack of stability if it is not considered. More complex examples which involve shock capturing at some part of the domain are included at the end of the chapter.

3.2.5.1 Patch test

Following Zienkiewicz [172], the patch test has been used as a first verification of convergence. These tests have been developed imposing stationary solutions and obtaining the topography from the primary unknowns. The spatial domain Ω is a single element e. Since the solution is stationary, the temporal domain is null and the test consist on the verification of zero accelerations. Then, if the the solution belongs to the basis functions space, the test will pass analytically. Otherwise, the test will pass asymptotically when the element is refined by subdivision of the domain (h-refinement). In that case, even if the element is not passing the test, the patch test is also useful since is checking the correctness of the implementation.

Several exact solutions have been applied to an element with size of $1m$. For the stabilized formulations a stabilization factor $\beta = 0.01$ is used. The flux corrected solution depends directly on the stabilized formulation. If the solution belongs to the FE space, the obtained accelerations are less than 10^{-16}, which is the round-off of machine precision.

Water at rest In this case, the free surface gradient and the velocity are zero. Some solutions can be built with that conditions, such as flat and non-flat topography, and bottom friction. In all the cases the accelerations are zero.

Slope in equilibrium This family of solutions verify constant water depth and constant velocity. The gravity terms (coming from the slope) are in equilibrium with the bottom friction terms. Several combinations are obtained with different directions of the slope and different Froude numbers, subcritical and supercritical.

Backwater analysis Finally, that family of analytical solutions, presents a constant flow rate where the gravity terms are in equilibrium with the bottom friction. But in that case, the primary unknowns do not belong to the FE space, since, either the velocity or the water depth are not linear. This test is passing asymptotically.

3.2.5.2 Wave in a channel with a backward step

The aim of this example is to show that the Galerkin formulation applied to the shallow water equations is unstable and how the present stabilization method can overcome this issue. A calibration of the stabilization parameter is performed to optimize the effect of the stabilization terms in the obtained solution. We study the propagation of a wave in a channel with a backward step (Figure 3.2) where all the boundaries are slip. The channel depth is 1m. An initial perturbation in the free surface at the left wall generates a wave which travels from 0 to 6s. The initial perturbation reads

$$\eta(t=0) = 0.05\cos(\pi x) \quad \text{if} \quad x < 1, \quad \eta = 0 \quad \text{otherwise} \tag{3.34}$$

The wave is reflected at the right wall and then faces the step in the opposite direction. Figure 3.3 shows the propagation of the wave along the channel. The problem is discretized with a mesh fine enough to test the artificial diffusion added by the stabilization (Figure 3.2). The average element size is $0.06m$ and near the corner the mesh is refined to $0.02m$. The time step is set automatically to keep a maximum Courant number equal to 1.0 at every step. The problem is run three times with different algorithmic constants $\beta = 0.001$, 0.01 and 0.1. In this example, the shock capturing term is disabled.

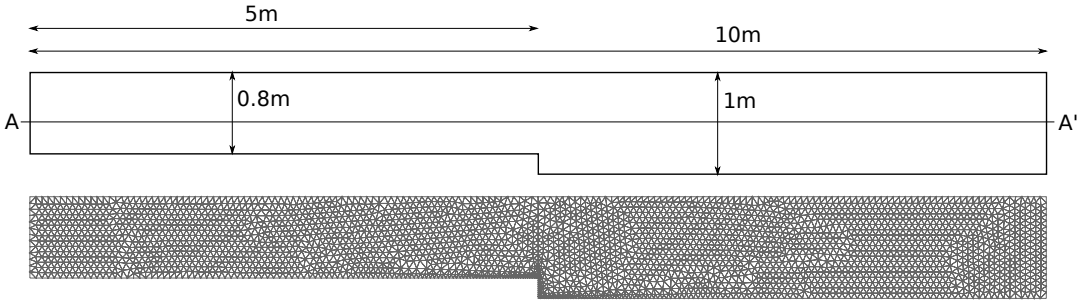
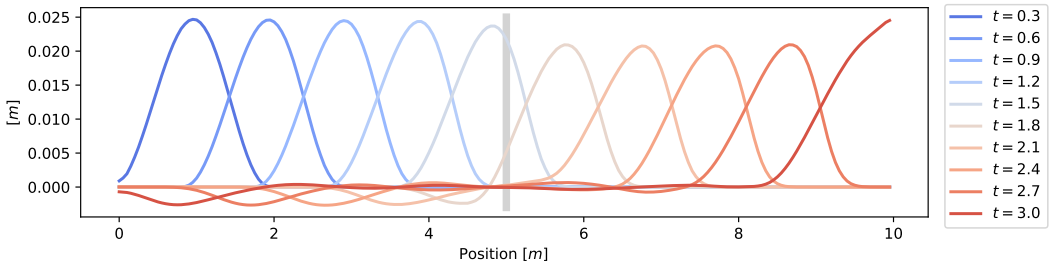
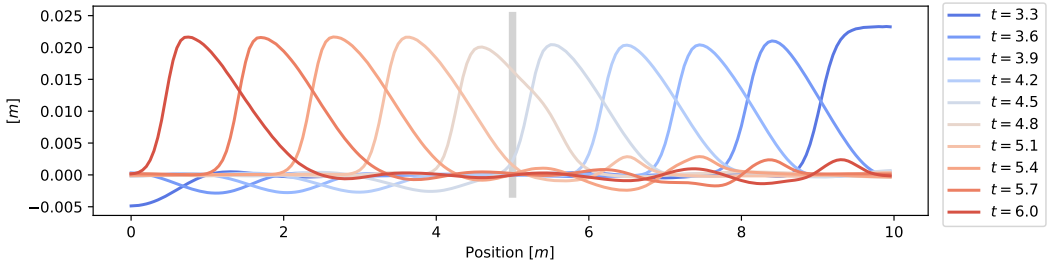


FIGURE 3.2: Channel with a backward step. Domain and mesh used in the simulation. All the boundary conditions are slip. The average element size is $0.06m$. Near the obstacle the mesh size is $0.02m$. There are 3.125 nodes and 5.826 elements.



(A) Time from 0 to $3s$



(B) Time from 3 to $6s$

FIGURE 3.3: Channel with a backward step. Timestamps of the free surface along the cut AA' from Figure 3.2. (A) The initial perturbation is propagating to the right. (B) Propagation of the reflected wave from right to left.

The best results are achieved with the intermediate value and it has been fixed for the rest of the examples in this paper. Figures 3.4 and 3.5 show that the lower value of β is not enough to provide stability, while the higher value is over diffusive.

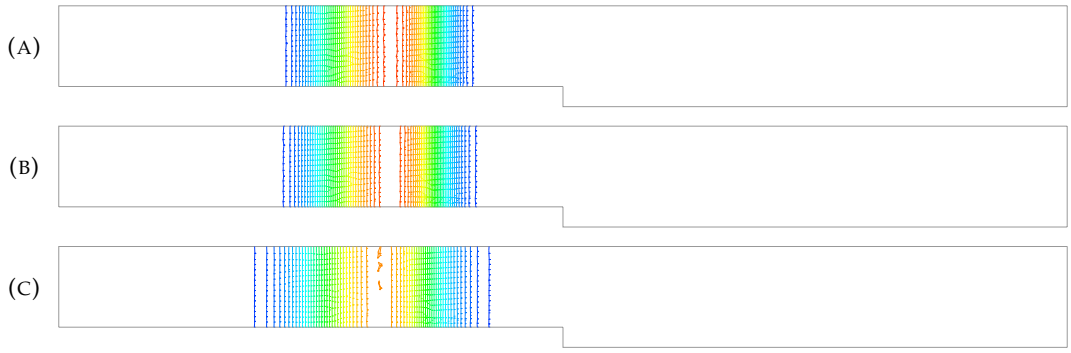


FIGURE 3.4: Channel with a backward step. Contour plots of the free surface elevation at time $t = 1s$ for different stabilization factors. (A) $\beta = 0.001$, (B) $\beta = 0.01$, (C) $\beta = 0.1$

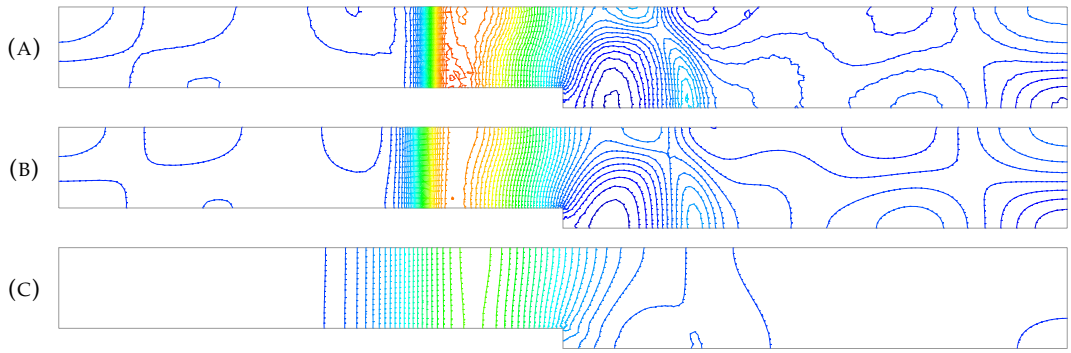


FIGURE 3.5: Channel with a backward step. Contour plots of the free surface elevation at time $t = 5s$ for different stabilization factors. (A) $\beta = 0.001$, (B) $\beta = 0.01$, (C) $\beta = 0.1$

3.3 The flux corrected algorithm for the shallow water equations

As seen in Section 3.2, the numerical approximation of hyperbolic systems -convection dominant- exhibit instabilities. Those instabilities are inherent to schemes of order greater than 1 (Godunov barrier theorem [66]). With the aim of suppressing the spurious oscillations and to extend to stable schemes to higher orders, the flux corrected (FC) algorithms were proposed. This new family of methods seeks for non oscillatory solutions and also for monotonicity of the numerical solution. At first, the FC algorithms were proposed for structured meshes and explicit time integration schemes [15, 16]. Generally, they were based for FD algorithms, but rapidly were extended also to Galerkin discretizations and implicit schemes [114, 129]. Even though, the FC transport has not been extended to many applications in the FEM community.

In this section the concept of FC is extended to the SW equations. The linearized equations presented in Chapter 2 can be understood as a generalization of the scalar transport equations, where the convection operator is extended with a matrix and the scalar unknown is replaced by a vector. This extension is proposed as an alternative to Lagrangian methods (Appendix A), where the shoreline discontinuity is naturally captured. At the same time, it seeks for a better accuracy than the stabilized Eulerian methods (Section 3.2): the introduction of the *shock capturing* leads to

a lower convergence order, making them first order. The main purpose to explore the FC algorithm is to make the numerical solution inherit the monotonicity of the physical solution. The water column is a positive property, whereas null water depth means the dry sub-domain. Tracking the zero isoline properly is of crucial importance to accurately model the wetting and drying processes -floods-. Unfortunately, the zero isoline presents a discontinuity in the unknowns and the hyperbolic property is lost, leading to numerical oscillations and consequently, losing the monotonic property.

The FC algorithm is based on the modification of the resulting matrices from the discrete weak variational. The formulation presented in Section 3.2.2 will be modified in order to enforce it to verify monotonic properties at the discrete level. The subjacent correction acts in such way that all the modifications are conservatives, namely, there is not gain or loss of fluid mass. The original scheme is recovered where the solution is smooth and does no require modifications.

In order to achieve that properties, the low order scheme is constructed through a lumping procedure of the mass matrix and the addition of enough diffusion in order to eliminate the negative entries of the off-diagonal terms. In that way, the construction of the low and high order schemes is performed elementally. On the other hand, the computation of the fluxes and its limiters is computed nodally. Finally, the global assembly is element-wise, as far as the nodes belong to the elements. The fact of recovering the high order scheme with consistent mass matrix is specially interesting to accurately simulate transient problems.

3.3.1 Equivalence with pure convection problems

In a first instance, the FC algorithms where developed for the transport equations as a two stage predictor-corrector algorithm. The correction of the fluxes had a direct relation with the physical fluxes and were evaluated at the edges of the mesh, acting directly to the entries of the global matrix.

Since the pure convection problem is a particular hyperbolic system of conservation law, this algorithm was extended to more complex systems, like the Euler equations. Here we will show the equivalence of the inviscid SW equations with pure convection problems. The starting point are the linearized SW equations in conservative form (2.10), which are analogue to the hyperbolic euler equations:

$$\frac{\partial \boldsymbol{\phi}}{\partial t} + \boldsymbol{A}_i \frac{\partial \boldsymbol{\phi}}{\partial x_i} = 0 \tag{3.35}$$

The matrix $\boldsymbol{A}_i$ is diagonalizable and can be decomposed as $\boldsymbol{A}_i = \boldsymbol{T}_i \boldsymbol{\Lambda}_i \boldsymbol{T}_i^{-1}$. Kuzmin [102] proposes this transformation for the one dimensional case. In high order dimensions, we can define a vector of transformed unknowns $\boldsymbol{\Phi}_i = \boldsymbol{T}_i^{-1} \boldsymbol{\phi}$ which yields

$$\frac{\partial \boldsymbol{\Phi}_j}{\partial t} + \boldsymbol{\Lambda}_i \frac{\partial \boldsymbol{\Phi}_j}{\partial x_i} = 0 \tag{3.36}$$

Since $\boldsymbol{\Lambda}$ is a diagonal matrix, the above expression consists on a set of decoupled convection equations.

$$\frac{\partial \Phi_{jk}}{\partial t} + \Lambda_{ikk} \frac{\partial \Phi_{jk}}{\partial x_i} = 0, \quad k = 1 : n_b, \quad i, j = 1 : n_d \tag{3.37}$$

In practice, instead of applying the FC concept to the diagonal system, the algorithm will be extended to the original system os equations. The main difficulty

which appears when the system is decomposed is that there isn't a unique direction of propagation for each uncoupled unknowns. This is known as the dispersive behavior of the shallow water equations.

3.3.2 Flux-corrected algorithm

The FC algorithm inherits the concept of *limiting* introduced by Boris and Book [16] and was developed by Löhner [114] and Kuzmin [100]. They also extended the concept of FC to the euler equations [102, 114]. In more recent developments, the FC technique was applied to the shallow water equations by Ortiz [137]. However, Ortiz presents an algorithm with a fully decoupled high and low order schemes. In this section, the FC scheme is applied to the shallow water equations preserving the same discretization for both high and low order schemes.

3.3.2.1 High and low order schemes

Usually, the high order scheme is a FEM solution, such as a Taylor-Galerkin [112] or Characteristic Based Split (CBS) [137]. In this work, the high order solution uses the spatial discretization presented in Section 3.2.2 with linear stabilization and without shock capturing. The time integration scheme is the backward differentiation formula from Section 3.2.3.

$$\mathbf{M}_C\dot{\phi}^{n+1} + \mathbf{K}\phi^{n+1} = \mathbf{q} \tag{3.38}$$

$$\dot{\phi}^{n+1} = \beta_1\phi^{n+1} + \beta_2\phi^n + \beta_3\phi^{n-1} \tag{3.39}$$

where ϕ is the vector of the nodal unknowns and $\dot{(\,)}$ denotes the time derivative, $\mathbf{M}_C$ is the consistent mass matrix, $\mathbf{K}$ is the system matrix and $\mathbf{q}$ is the source term. This scheme will be taken as reference for the high order scheme but, as any stabilized and not monotonicity preserving formulation, is not oscillation-free, specially near the wet-dry interface.

The low order solution is obtained from the high order scheme with a mass lumping procedure and adding scalar diffusion to the high order scheme:

$$\mathbf{M}_L\dot{\phi}^{n+1} + (\mathbf{K} + \mathbf{D})\,\phi^{n+1} = \mathbf{q} \tag{3.40}$$

$$\mathbf{D} = c_\tau\,(\mathbf{M}_L - \mathbf{M}_C) \tag{3.41}$$

The difference between the two systems of equations is the amount of artificial diffusion which ensures monotonicity and can be formally written as

$$\mathbf{P}(\phi^{n+1}, \phi^n) = (\mathbf{M}_L - \mathbf{M}_C)\dot{\phi}^{n+1} - \mathbf{D}\phi^{n+1} \tag{3.42}$$

Following Kuzmin [101], one can observe that the anti-diffusion $\mathbf{P}$ can be decomposed into a sum of internodal fluxes f_{ij} where the contribution to node i from the node j is the opposite from i to j, that is $f_{ij} = -f_{ji}$. This edge-based approach shows that the quantity added to a node, is subtracted from another node, and the anti-diffusion is conservative. For convenience we will keep the finite element structure. The high order scheme is recovered if the unlimited correction $\mathbf{P}$ is added to the low order scheme.

3.3.2.2 Flux correction limiter

The essentials of the FC method is to switch between the low and high order solutions in an adaptive fashion to satisfy the positivity constraint. The Zalesak limiter is employed to define the amount of anti-diffusion added to the low order solution which at the same time guarantee the positivity constraint for a given variable. The main steps explained above are depicted in Figure 3.6.

The anti-diffusive fluxes are decomposed in those that increase the value and those that decrease it, the increment $\mathbf{P}_i$ is decomposed in a sum of positive and negative contributions

$$P_i = P_i^+ + P_i^- \ , \qquad P_i^\pm = \sum_{j \neq i} \underset{\min}{\max}\{0, j_{ij}\} \tag{3.43}$$

While the maximum/minimum admissible increments depends on the solution values at the neighboring nodes that share an element with node i

$$Q_i^\pm = \underset{\min}{\max} \Delta\phi_{ij}^\pm \ , \qquad \text{where} \quad \Delta\phi_{ij}^\pm = \underset{\min}{\max}\{0, \tilde{\phi}_j - \tilde{\phi}_i\} \tag{3.44}$$

where the $\tilde{(\ )}$ symbol denotes a predictor for the solution at time t^{n+1}. In order to prevent the formation of spurious oscillations, the positive/negative anti-diffusive flux should be limited by the following ratio

$$R_i^\pm = \begin{cases} \min\{1, Q_i^\pm / P_i^\pm\} & \text{if} P_i^\pm \neq 0 \\ 1 & \text{if} P_i^\pm = 0 \end{cases} \tag{3.45}$$

It is important to note that, in order to preserve mass conservation property, the same limiter should be applied to all the element contributions. So, the most restrictive ratio is chosen:

$$c_e = \min\{R_i^\pm\} \ , \quad \forall \text{node}_i \in \text{element}_e \tag{3.46}$$

3.3.2.3 Iterative solution

Since the linearization matrix A depends on the solution at time t^{n+1}, the system matrix K needs to be updated at the end of each iteration and the low order algebraic system of equations (3.40) is rewritten in residual based form

$$\left(\beta_0 \mathbf{M}_L + \mathbf{K}^{(m)} + c_\tau \mathbf{D}\right) \Delta\phi^{(m+1)} = \mathbf{q} - \left(\mathbf{K}^{(m)} + c_\tau \mathbf{D}\right) \phi^{(m)} - \mathbf{M}_L \dot{\phi}^{(m)} \tag{3.47}$$

where $\phi^{(0)} = \phi^n$. After adding the limited flux correction and substitutions, the system (3.47) yields

$$\begin{aligned} \left(\beta_0 (c\mathbf{M}_c + (1-c)\mathbf{M}_L) + \mathbf{K}^{(m)} + c_\tau (1-c)\mathbf{D}\right) \Delta\phi^{(m+1)} = \\ \mathbf{q} - \left(\mathbf{K}^{(m)} + c_\tau (1-c)\mathbf{D}\right) \phi^{(m)} - (c\mathbf{M}_c + (1-c)\mathbf{M}_L)\, \dot{\phi}^{(m)} \end{aligned} \tag{3.48}$$

NOTE: the displacement or residual criteria does not converge if the c_i limiter vary along the iterations (m). There are some alternatives to overcome this issue:

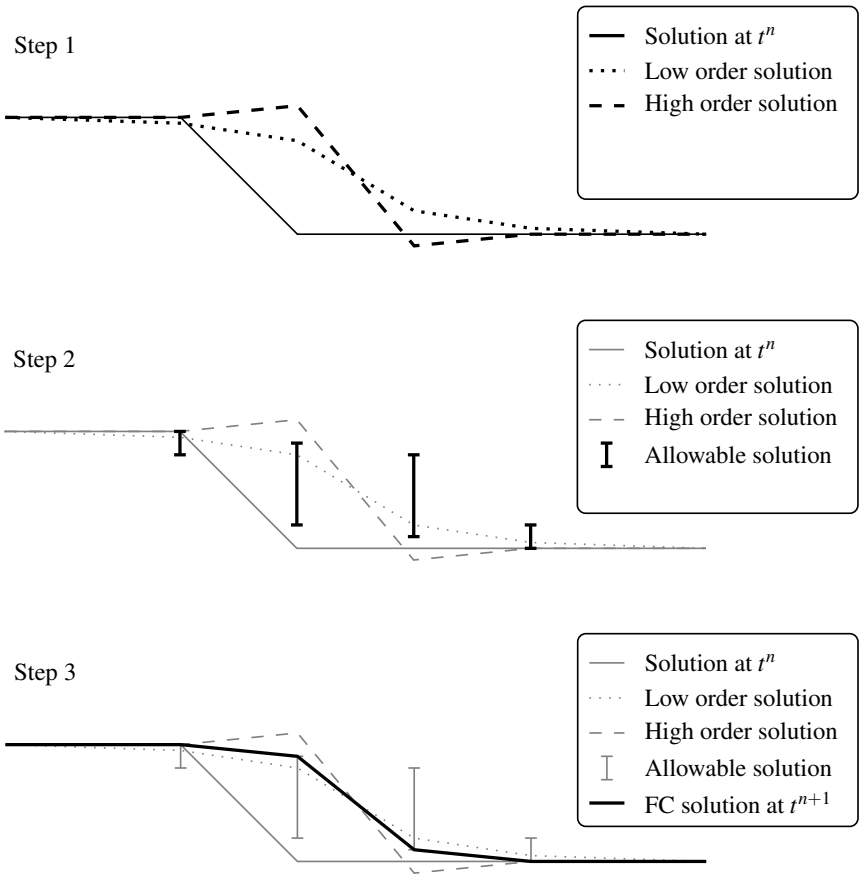


FIGURE 3.6: Main steps of the FC algorithm for the computation of the solution at time step t^{n+1}.

- to use a constant c_i, obtained from the first iteration. This solution may not be optimal and possibly does not preserve monotonicity;
- to use a semi-implicit time integration scheme which defines the predictor $\tilde{\phi}$ see equation (3.44);
- to define an incremental limiting, this seems to be the most accurate scheme, even though it is more diffusive than the presented one.

3.3.2.4 Limiting a set of equations and variable transformation

The presented limiting is based on the monotonicity requirements for scalar convection, however, the coupling of the shallow water equations require a synchronized limiting for the correction factors. It is commonly accepted that for coupled systems of equations the flux limiter taken the most restrictive among the set of variables $c_i^+ = \min\{c_{ik}^+\}$ and $c_i^- = \min\{c_{ik}^-\}$. There are the following strategies to compute the set of limiters $c_{ik}^{\pm}$:

- flux limiting in terms of the independent variables $h, \mathbf{q}$;
- flux limiting based on several indicators as the total energy, water pressure or maximum eigenvalue;
- transformation of the variables into characteristic variables in order to obtain a set of decoupled system of equations where the principles of the flux correction for pure convection can be applied.

In omission of source terms, equation (3.3) can be multiplied by the $\mathbf{T}_i$ base of eigenvectors and transformed into a set of decoupled pure convection problems

$$\frac{\partial \Phi_{jk}}{\partial t} + \Lambda_{ikk}\frac{\partial \Phi_{jk}}{\partial x_i} = 0, \quad k = 1 : \dim + 1, \quad i, j = 1 : \dim \tag{3.49}$$

This transformation should be computed locally at each node and depends on the current solution at time t^{n+1}. However, for the two dimensional case, the set of variables ϕ is transformed in a two dimensional vector of variables Φ_i, hence the scalar transport correction techniques should be extended to vectorial transport.

The difficulties presented by the variables transformation lead to the consideration if it is worth computing them rather that directly using the independent variables or an indicator. The limiting based on the independent variables was the choice presented in [114] and it is the approach followed in this study.

In section 3.5 the accuracy of the flux corrected algorithm will be analyzed and compared against the residual-based formulation. In the following section an alternative to the flux corrected algorithm is explored.

3.4 Quasi monotonicity preserving formulations

The scope of this section is to recover the stabilized formulation from Section 3.2 and apply some of the monotonic properties from Section 3.3. Authors as Burman and Badia demonstrated that, under certain situations, stabilized monotonic solutions can be obtained (see [9, 25, 26] for example). Following these authors, the stabilizations can be classified into residual and projections-based. In the first category, we can find some stabilization methods, such as SUPG, GLS, VMS or FIC, based on the

proposals of Hughes et al [22, 82, 83]. This is the option followed in Section (3.2). In the second group, Codina proposed the Orthogonal Sub Scales (OSS) method [38]. The inconvenient of this method is the introduction of a global projection to solve the sub-scales. As an alternative, methods with local projection stabilization (LPS) appeared, see ([8, 9, 17, 24, 123])). The main difference among this methods and the residual-based is tha they introduce only the terms which guarantee stability.

Nevertheless, the stabilizations are not a sufficient condition to ensure oscillation-free solutions near discontinuities as they are not monotonicity preserving. The monotonicity can be verified through the well-known shock-capturing methods. Those family of methods present a good global convergence and introduce non-linear viscosity around discontinuities [94]. Once again, there are two families for the design of non-linear viscosity: the residual-based methods and the gradient jump methods.

Taking into consideration both the accuracy and convergence, the best option seems to be the combination of linear stabilization and residual-based shock capturing. It presents a good global convergence and, only where it is needed, the order of convergence is sacrificed with the introduction of artificial viscosity. Nevertheless, this choice of stabilization leads to a scheme which does not guarantee the monotonic properties of the system.

Burman proved that, under certain conditions, monotonicity can be achieved using a nonlinear viscosity based on the variation of the gradient [26]. However, it is important to consider that an arbitrary combination of methods can lead to the loss of the monotonic properties offered by one of the two methods [9].

3.4.1 Shock capturing

Traditionally, the most used method to eliminate the spurious oscillations near discontinuities is the local artificial viscosity. Considering a discretization of finite elements $\langle \boldsymbol{\omega}, \mathbf{r} \rangle$ where $\langle \cdot, \cdot \rangle$ is the discrete counterpart of the variational principle, a non linear stabilization can be expressed as

$$\langle \boldsymbol{\omega}, \mathbf{r} \rangle + s(\boldsymbol{\omega}, \boldsymbol{\phi}) = 0 \tag{3.50}$$

where

$$s(\boldsymbol{\omega}, \boldsymbol{\phi}) = \langle \epsilon(\boldsymbol{\phi}) \nabla \boldsymbol{\omega}, \nabla \boldsymbol{\phi} \rangle \tag{3.51}$$

and the non linear viscosity $\epsilon(\boldsymbol{\phi})$ is sufficiently large near the discontinuities in order to mitigate oscillations. In the framework of the FEM, the design of $\epsilon(\boldsymbol{\phi})$ can be residual-based (RV), gradient jump based (GJV) or entropy based (EV) [71]. Usually, RV have been combined with linear stabilization and EV have been applied alone. The argument that justify the choice of using EV without stabilization is that linear stabilization acts as an hyper-viscosity term eliminating DMP property.

GJV methods introduce the concept of jump $[[v]]_a$ and average $\{\{v\}\}_a$ of a property v in a point a belonging to the boundary of an element. Being $\mathbf{n}$ the normal vector to the boundary, there are the following definitions

$$[[v]]_a = \lim_{\epsilon \to 0} (v(\mathbf{a} + \epsilon \mathbf{n}) \cdot \mathbf{n} - v(\mathbf{a} + \epsilon \mathbf{n}) \cdot \mathbf{n}) \tag{3.52a}$$

$$\{\{v\}\}_a = \frac{1}{2} \lim_{\epsilon \to 0} (v(\mathbf{a} + \epsilon \mathbf{n}) \cdot \mathbf{n} + v(\mathbf{a} + \epsilon \mathbf{n}) \cdot \mathbf{n}) \tag{3.52b}$$

Burman proposed in [26] a shock capturing with monotonic properties for the Burgers equation in 1D. Badia extended that method to multiple dimensions for

the scalar transport in [9]. Here, the proposed formulation for scalar transport is applied to the SW equations, using the water depth as the transported property. The formulation proposed in [9] is defined by the following artificial viscosity for each element e:

$$\epsilon(h)_e = c_{gjv} l_e \|\mathbf{A}\|_{L_e^{\inf}} \max_{b \in e} \left[\frac{\int_b |\nabla h \cdot \mathbf{n}| d\sigma}{\int_b \|\nabla h\| d\sigma} \left(\frac{\int_b |[[\nabla h]]| d\sigma}{\int_b \{\{|\nabla h \cdot \mathbf{n}|\}\} d\sigma} \right)^q \right] \tag{3.53}$$

The parameter c_{gjv} is an algorithmic constant. The original expression depends on some mesh shape parameters and the angles of the element. In practice $c_{gjv} = \frac{1}{2}$ is chosen, this expression coincides with the minimum value to guarantee monotonicity in 1D. Nevertheless, the monotonicity is lost in 2D. It is not surprising to lose monotonicity, since that property is even lost for the pure diffusion equation. Even though, it exhibits an optimal behavior.

3.5 Examples

In this section we present four different examples. Three of them are oriented to verify a single aspect of the procedure explained in this research, the global stabilization, the shock capturing technique and the dry-wet interface. The last one is devoted to test all the capabilities of the formulation in a practical case for which experimental data is available.

3.5.1 Oscillation in a parabolic basin

This is a classical benchmark oriented to test the accuracy of the location of the moving boundary. The topography follows a parabolic profile and the water body oscillates on it. The initial configuration corresponds to zero velocity but the free surface is in a non horizontal plane. Given the parabolic profile of the topography, exists an analytical solution for the oscillatory problem and the free surface elevation remains planar. Details about the derivation of the analytical solution can be found in the compilation made by Delestre et al. in [53].

The spatial domain Ω is defined in the interval $[0, L] \times [0, 1]m$ where $L = 10m$ and all the boundaries are reflective ($\mathbf{u} \cdot \mathbf{n} = 0$). The topography is given by the following expression

$$z(x, y) = h_0 \left(\frac{1}{a^2} \left(x - \frac{L}{2} \right)^2 - 1 \right) \tag{3.54}$$

The primitive variables are defined by

$$h(x, y) = \begin{cases} -h_0 \left(\left(\frac{1}{a} \left(x - \frac{L}{2} \right) + \frac{1}{2a} \cos(2Bt) \right)^2 - 1 \right) & \text{if } x_1(t) < x < x_2(t) \\ 0 & \text{otherwise} \end{cases} \tag{3.55a}$$

$$\mathbf{u}(x, y) = \begin{cases} (B, 0) \sin(2Bt) & \text{if } x_1(t) < x < x_2(t) \\ (0, 0) & \text{otherwise} \end{cases} \tag{3.55b}$$

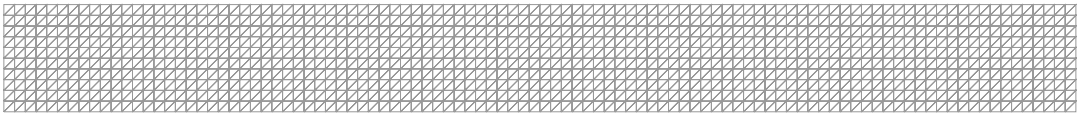


FIGURE 3.7: Parabolic basin. One of the meshes used in the analysis. The element size is $0.1m$.

where $B = \sqrt{2gh_0}/2a$, and x_1, x_2 are time dependent functions which define the location of the dry-wet interface:

$$\begin{aligned} x_1(t) &= -\frac{1}{2}\cos(2Bt) - a + \frac{L}{2} \\ x_2(t) &= -\frac{1}{2}\cos(2Bt) + a + \frac{L}{2} \end{aligned} \tag{3.56}$$

In that example the selected parameters are $h_0 = 1m$ and $a = 1m$.

The domain Ω is discretized using several meshes in order to perform a convergence analysis. The meshes employed are listed in Table 3.1. Figure 3.7 shows one of the intermediate mesh. Once the simulation begins, the water starts to oscillate on the parabolic basin and the velocity field is constant on the spatial domain, while follows a periodic function respect to the time.

A cut along the mesh is depicted in Figure 3.8 showing the free surface, the velocity and the location of the shoreline. Since the shoreline is moving, several problems may arise, such as the stability of the wetting or drying element or capturing the discontinuity of the velocity. Fortunately, the velocity is not a a degree of freedom and this will help to avoid spurious oscillations in the vicinity of the shoreline.

n_{nodes}	Δx	Δt	CFL	$L_2(e_{rel})$
205	0.25	0.008	0.5	0.24
1,111	0.1	0.003	0.5	0.064
4,221	0.05	0.002	0.5	0.023
11,356	0.03	0.001	0.5	0.013
101,101	0.01	0.0003	0.5	0.0049

TABLE 3.1: Parabolic basin. Convergence error for the water height.

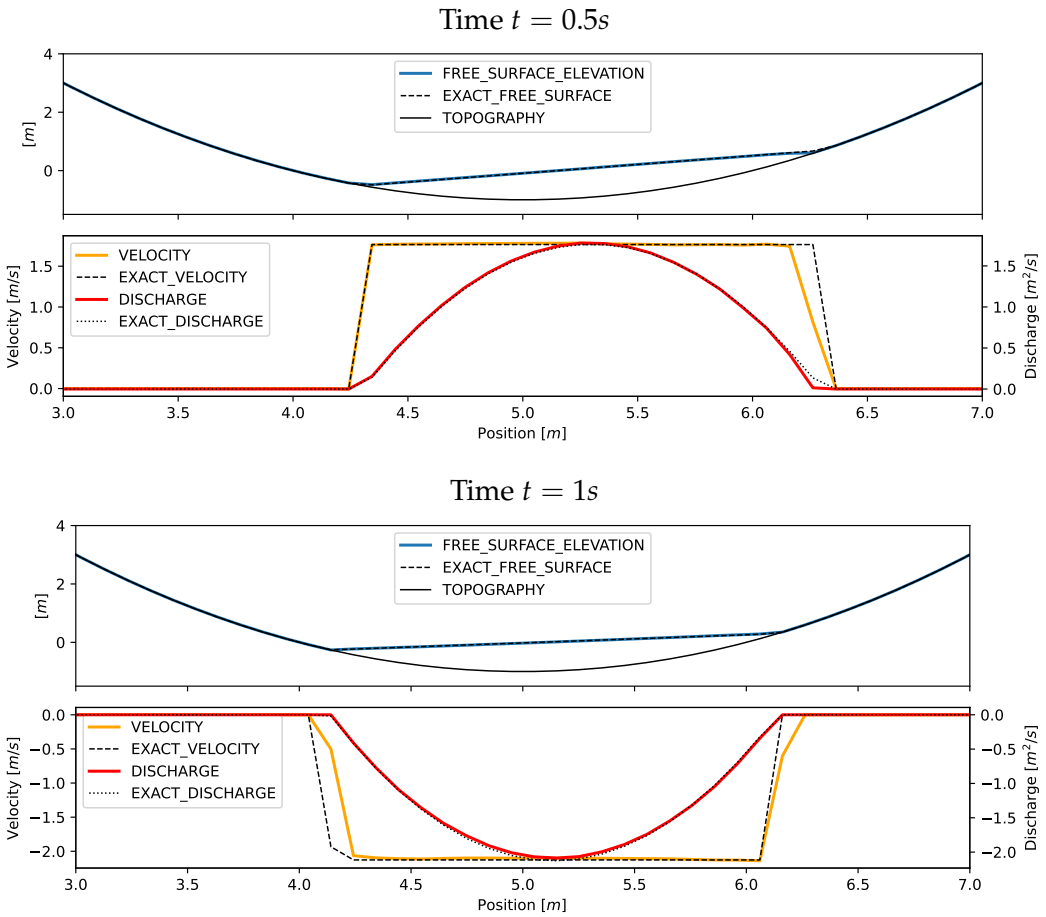


FIGURE 3.8: Parabolic basin. Cuts along the mesh of size $0.03m$ at different times. There are 333 nodes on the cut.

A global representation of the solution at time $t = 1s$ is presented in Figure 3.9. In that case the finest mesh ($\Delta x = 0.01m$) has been used, for visualization purpose the discretization is not shown. As expected, there is no variation on the results in the transversal section.

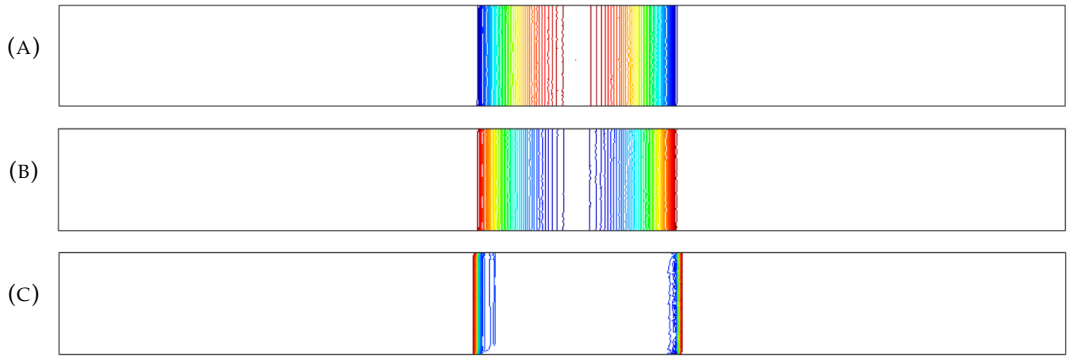


FIGURE 3.9: Parabolic basin. Contour lines with the fine mesh of size $0.01m$ at time $t = 1s$. (A) Water height, (B) x-discharge and (C) x-velocity. There is no legend for simplicity.

Combination of linear with non linear stabilization The main problem of linear stabilizations is the loss of the monotonic properties which may be provided by

nonlinear stabilizations. For that reason, in [71] is proposed to use a nonlinear stabilization alone. This is the case of EV. However, the use of nonlinear stabilization alone, may lead to unwanted results. The most common undesired results are such as terracing near the boundaries [102, 114] and mesh-dependent diffusivity [9].

It is known that the combination of GJV with SUPG or FIC is not monotonicity preserving. However, the monotonic properties can be improved if it is compared against a standard RV method. The convergence of the presented methods is compared in Figure 3.10.

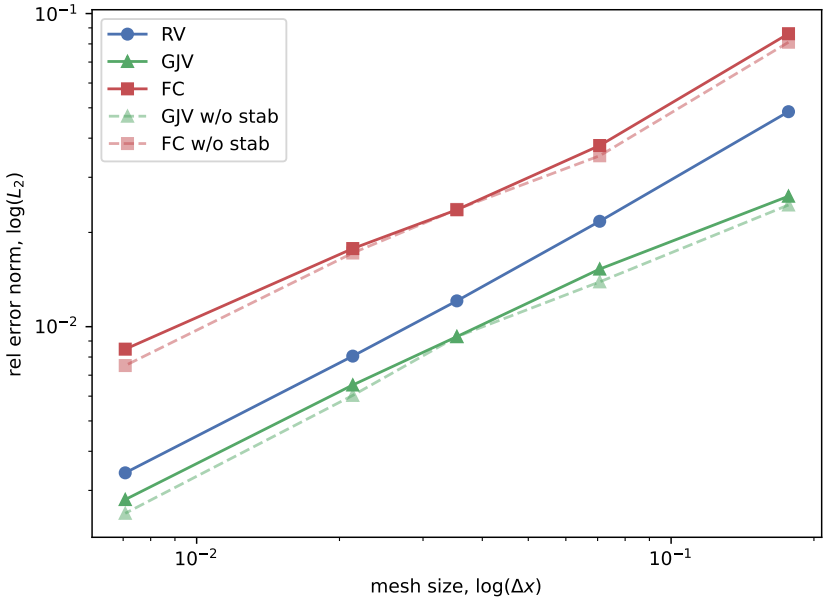


FIGURE 3.10: Parabolic basin. Convergence graph.

A mass conservation test is performed with the mesh of size $0.05m$ in a simulation of $5s$. The results are presented in Figure 3.11. The integration of the mass is performed over all the domain and over the wet domain. The wet domain is identified with the wet fraction, requiring that it is equal to 1. Since the presented schemes are not mass sign preserving, the wet mass can not be equal to the total mass and a small fraction is lost from the wet domain. The loss depends on the element size and presents an oscillatory behavior inherent of the method.

Regarding the different non linear stabilizations, it is clear that none of the presented methods is able to ensure monotonicity. Nevertheless, the gradient jump viscosity method reduces the magnitude of the oscillations. Another interesting feature is that the maximum oscillation is bounded and does not increase during the natural oscillations of the sloshing. The flux-corrected algorithm presents the maximum spurious oscillations.

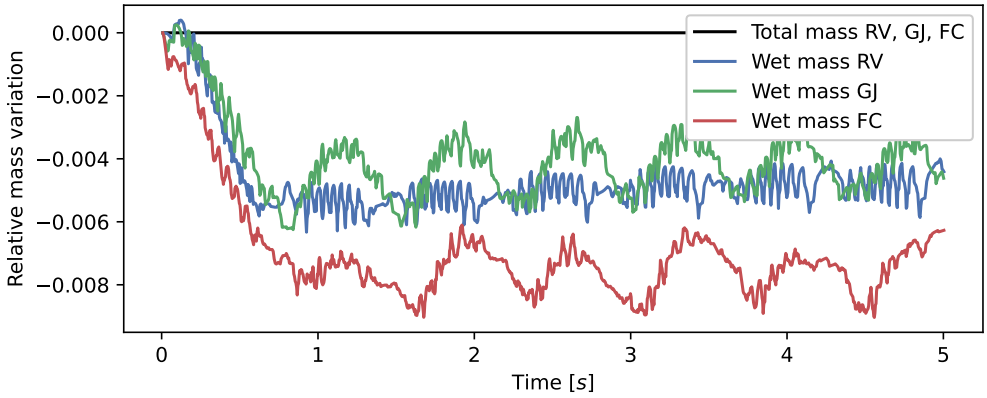


FIGURE 3.11: Parabolic basin. Mass conservation error. The element size used in this simulation is $0.05m$.

3.5.2 Short channel with smooth transition and shock

The second example consists on a benchmark based on the Mac Donald's type solutions [115]. The analytical solution can be found in the compilation which was referenced in the previous example [53]. This test presents a channel with a steady state solution. There is a subcritical inlet and a transcritical flow is produced. The outlet is also subcritical and then a shock is generated at $2/3$ of the channel. The aim of this example is to evaluate the shock capturing technique presented and the correct location of the hydraulic jump, which depends on the bottom friction law.

Here we will consider the 1D shallow water equations without diffusion and only with Manning bottom friction as source term. A steady state solution satisfies $\frac{\partial q}{\partial x} = 0$ and Equation (2.5) reduces to

$$\frac{\partial z}{\partial x} = \left(\frac{v^2}{gh} - 1\right)\frac{\partial h}{\partial x} - n^2\frac{|v|v}{h^{4/3}} \tag{3.57}$$

This relation allows to integrate the topography given an analytical expression for the water height. Another approach in hydraulics is to consider a given discharge and topography and integrate the water height using Equation (3.57). Following both approaches exact solutions can be obtained. Since this expression involves the bottom friction, we can verify if the friction term is correctly coded in order to satisfy the steady state.

For this benchmark we have considered the domain defined by the spatial domain $[0, 100] \times [0, 5]$ which is a channel of $100m$ length and $5m$ width (Figure 3.12), and the following boundary conditions:

$$\begin{aligned} q_x &= 2\ \mathrm{m/s} \quad \text{in} \quad \Gamma_{upstream} \\ h &= h_{ex}(100) \quad \text{in} \quad \Gamma_{downstream} \\ q_y &= 0 \quad \text{in} \quad \Gamma_{walls} \end{aligned} \tag{3.58}$$

The problem is initialized with the following values:

$$\begin{aligned} h(x,0) &= \max(h_{ex}(100) - z(x), h_{ex}(0)) \\ \mathbf{q}(x,0) &= \mathbf{0} \end{aligned} \tag{3.59}$$

n_{nodes}	Δx	Δt	CFL	$L_2(e_{rel})$
204	2.0	0.005	0.016	0.177
606	1.0	0.005	0.031	0.138
2211	0.5	0.005	0.062	0.088
13026	0.2	0.005	0.15	0.041

TABLE 3.2: Short channel. Error for the x-discharge.

The Manning coefficient is 0.0328 $\mathrm{m}^{-1/3}\mathrm{s}$ and the water height $h_{ex}(x)$ is a piecewise function defined in [53]. The discontinuity of the water height function is located at $x = 200/3m$ and defines the hydraulic jump. The expression of the stationary exact water depth is:

$$h_{ex} = \begin{cases} \left(\frac{4}{g}\right)^{\frac{1}{3}} \left(\frac{4}{3} - \frac{x}{L}\right) - \frac{9x}{10L}\left(\frac{x}{L} - \frac{2}{3}\right), \text{ for } x < \frac{2L}{3} \\ \left(\frac{4}{g}\right)^{\frac{1}{3}} \left(a_1\left(\frac{x}{L} - \frac{2}{3}\right)^4 + a_1\left(\frac{x}{L} - \frac{2}{3}\right)^3 - a_2\left(\frac{x}{L} - \frac{2}{3}\right)^2 \right. \\ \qquad \left. + a_3\left(\frac{x}{L} - \frac{2}{3}\right) + a_4\right), \text{ for } x \geq \frac{2L}{3} \end{cases} \tag{3.60}$$

where $a_1 = 0.674202m$, $a_2 = 21.7112m$, $a_3 = 14.492m$, $a_4 = 1.4305m$ and $L = 100m$. The topography is obtained by a numerical integration using the fourth order Runge Kutta method.

As in the previous example, several meshes are employed and a convergence analysis is performed (Figure 3.13). The shock capturing parameter is $\alpha = 1.0$ and we will study the accuracy of the hydraulic jump. Given the initial conditions, the hydraulic jump is generated between the first 50 and 80s. The overall error is computed at time $t = 200s$, in order to ensure the stationary state is achieved. The table from Figure 3.13 shows the error of the x-discharge over all the domain using the L_2 norm.

Results from different meshes are compared in Figures 3.14a and 3.14b. The oscillations are reduced with the finer mesh (Figure 3.14b), but there is a peak on the discharge at the location of the shock. This peak is initiated because the momentum balance includes the gradient of the total water depth, and the analytical gradient is a Dirac delta function.

FIGURE 3.12: Short channel. Geometry of the channel. The vertical line shows the position of the hydraulic jump.

3.5.3 Experimental dam break flow against an isolated building

The last example consists on the reproduction of the experiment carried out by Soares [155]. A dam break flow with a building downstream is simulated. The problem definition is depicted in Figure 3.15. The channel is 3.4m wide and the end of the dam is located at $x = 0$. As initial conditions, the water depth is set to 0.4m in the reservoir, while the channel is dry. The Manning coefficient is $0.01sm^{-1/3}$ over all the domain. At the beginning of the simulation, the gate of the dam is removed and the water is allowed to flow around the building.

There are some gauges (Table 3.3 and Figure 3.15) where the water height and velocity are recorded. Experimental data is used to validate the numerical method.

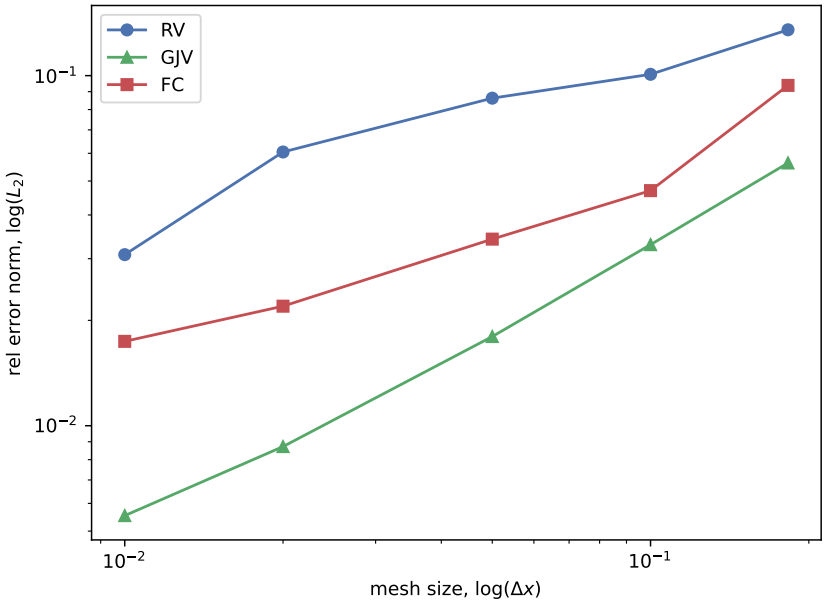


FIGURE 3.13: Short channel. Convergence graph for the x-discharge.

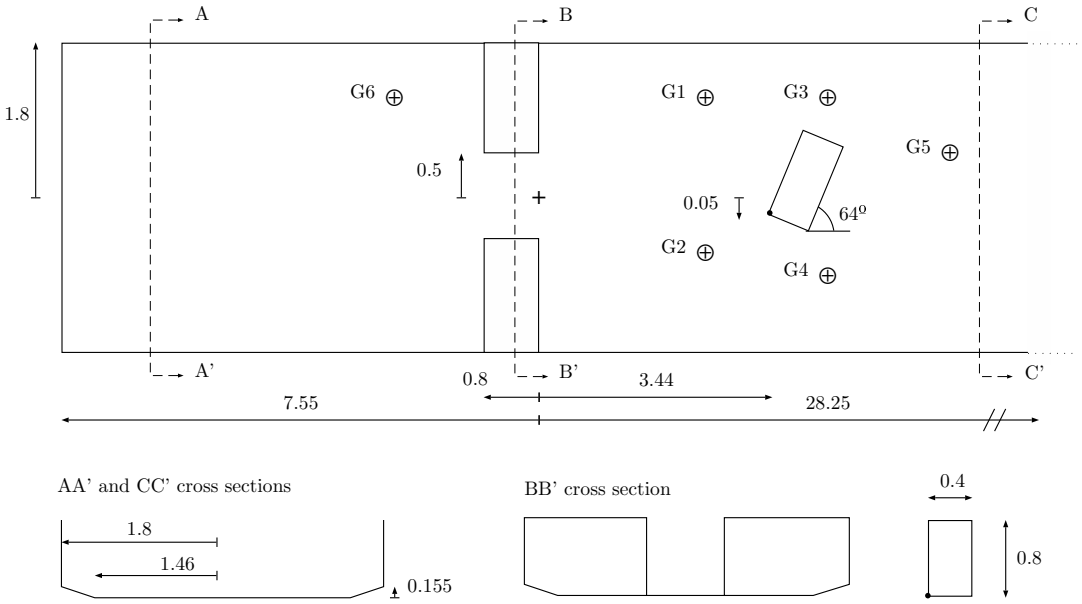
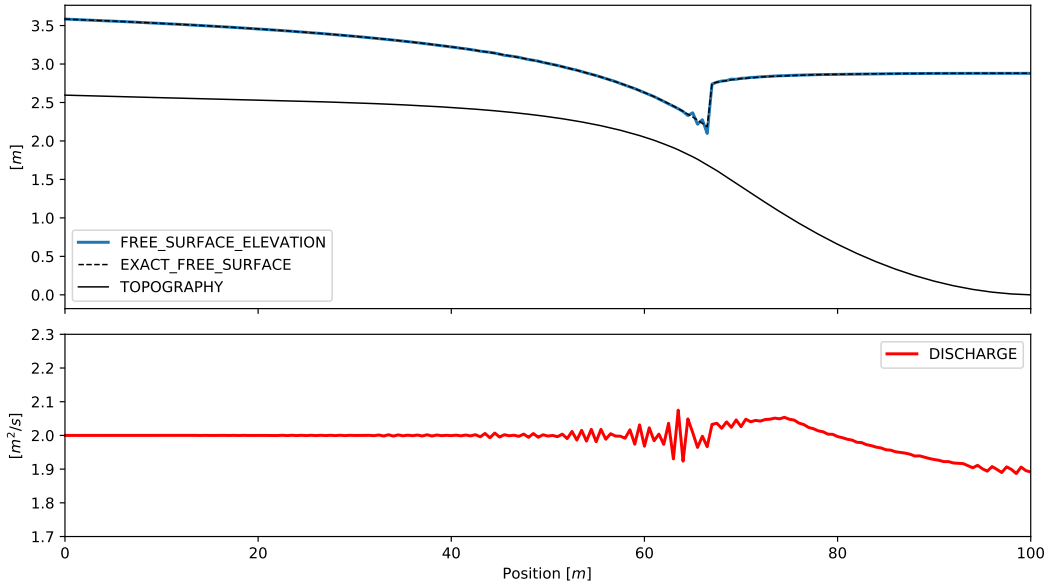


FIGURE 3.15: Experimental dam break flow. Definition of the isolated building benchmark. The dimensions are in m.

The domain is discretized with a mesh with an average element size of $0.05m$. There are 115.000 elements and the time step is computed to keep a courant number of 1.0. Figure 3.16 displays two details of the mesh, near the dam and around the building.

(A) Mesh size $0.5m$



(B) Mesh size $0.2m$

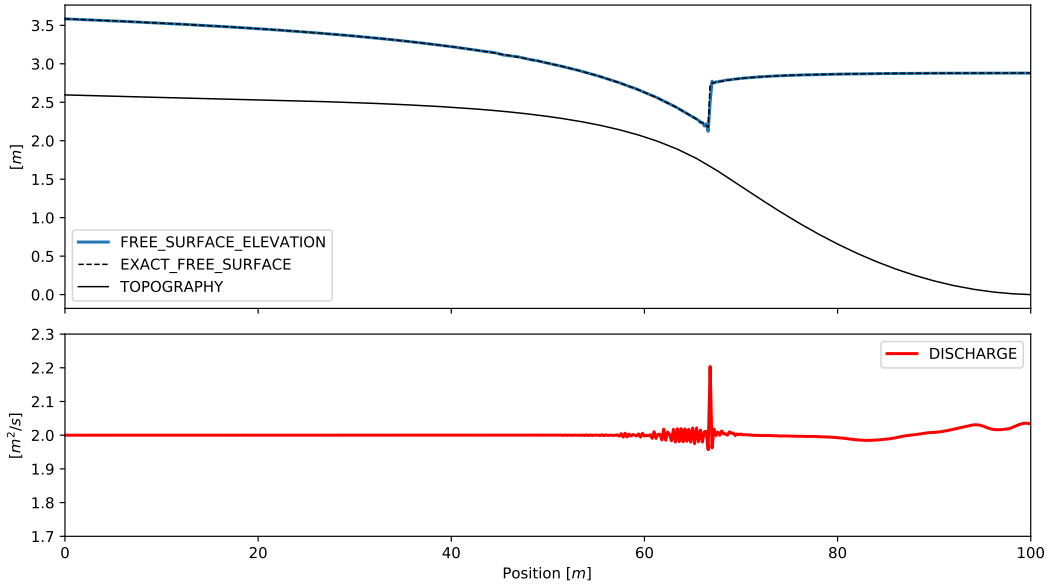


FIGURE 3.14: Short channel. Graph along the cut defined by the center of the channel.

Gauge number	X	Y
1	2.65	1.15
2	2.65	-0.60
3	4.00	1.15
4	4.00	-0.80
5	5.20	0.30
6	-1.87	1.10

TABLE 3.3: Experimental dam break flow. Positions of the gauges, units in m.

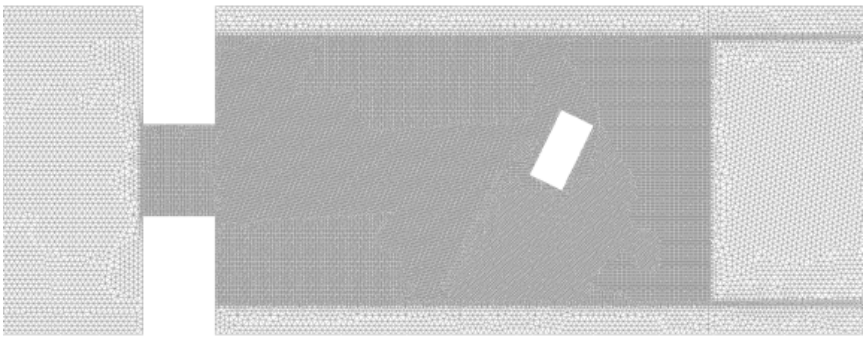


FIGURE 3.16: Experimental dam break flow. Detail of the mesh near the dam and around the building. The coarse elements have an average size of $0.06m$ and the refined area has an average element size of $0.02m$. There are 160.000 elements.

To get a general idea of the flow, Figure 3.17 shows several results of the water depth after the gate release. Figure 3.18 shows the evolution of the water depth at the gauges. An initial delay is observed in the propagation of the front in the gauges 1 to 5.

Gauge 1 is located upstream of the building and close to the left wall of the channel. In gauge 1 there are the main discrepancies between the numerical and experimental results. The vicinity of the wall is responsible for the rapid variations in the water level. After the arrival of the front wave, this is reflected in the wall and a second raise of the water level is observed. About $t = 6s$, an oblique hydraulic jump is formed and registered in gauge 1. The two first shocks are well captured by the numerical results, but the oblique hydraulic jump is registered latter, at $t = 10s$, and there is a general overestimation of the ones for the water depth values.

The main hydraulic jump in gauge 2 formed by the reflection against the building is registered at $t = 15s$, but in the numerical simulation is it formed very rapidly, presenting a discontinuity in time.

Gauge 3 is located at the left hand side of the building, where multiple waves are reflected and practically always it is in subcritical regime. There is a good correlation between numerical and experimental results. Gauge 4 is at the opposite side of the building but the superposition of the reflected waves is more clearly identified. The main discrepancies in the results are concentrated in the first seconds, where the flow is more dynamic.

Gauge 6 is located at the reservoir and registers the superposition of smooth waves during the emptying of the tank.

As stated in [155] there are some difficulties in the recording of the velocity and its validity is discussed. Here we compare only the most representative gauges. Gauge 2 is not fully submerged and the validity of the measurements is good after $t = 14.5s$. It illustrates the change from supercritical flow to subcritical (Figure 3.19). The considerations are similar to the water depth study.

The experimental measurements in gauge 4 (Figure 3.20) show a change in the velocity direction around $t = 15s$ due to the rise of water level. The numerical results do not capture this change but the mean and the stationary values are correctly simulated.

In gauge 5 the formation of eddies behind the building can be appreciated from $t = 20$ (Figure 3.21). In that case, the experimental results have difficulties to capture the eddies. Results will probably improve by extending the refined region of the mesh. The only method able to reproduce the eddies is the RV.

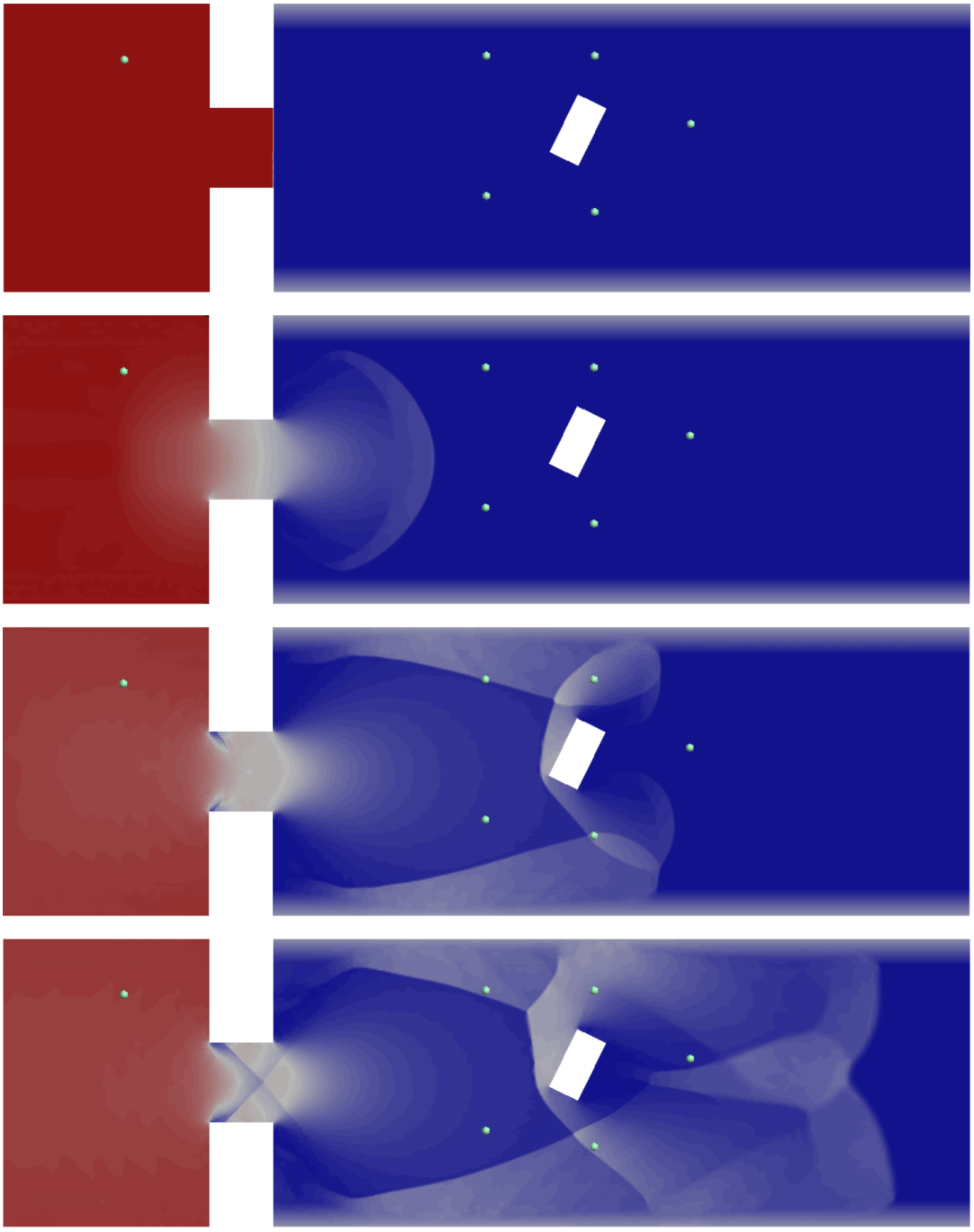


FIGURE 3.17: Experimental dam break flow. Results of the benchmark at times 0, 1 and 3 seconds

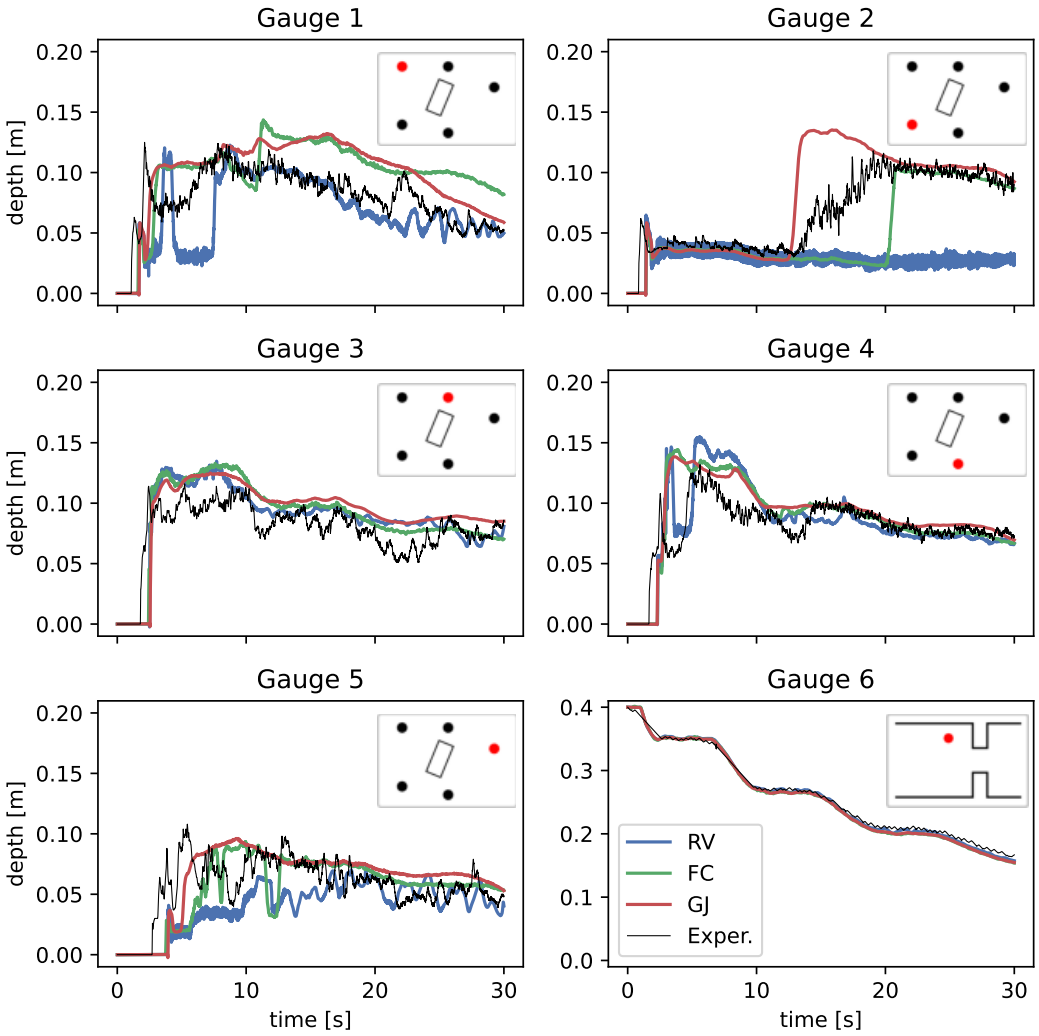


FIGURE 3.18: Experimental dam break flow. Comparison between the obtained water depth with the reference data.

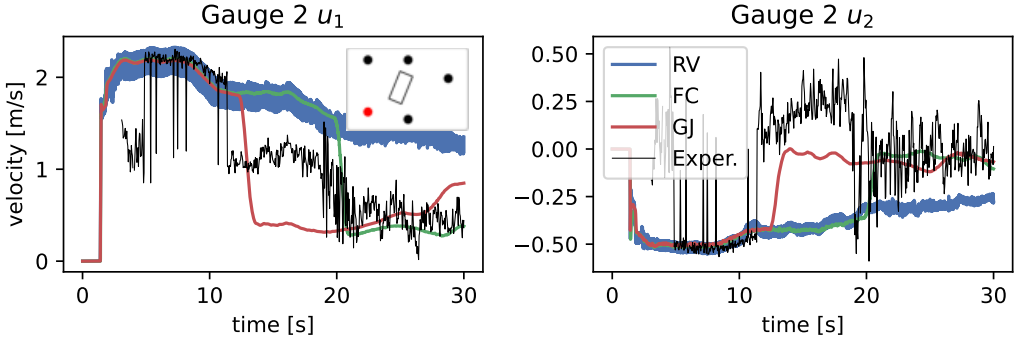


FIGURE 3.19: Experimental dam break flow. Comparison of velocity at gauge 2.

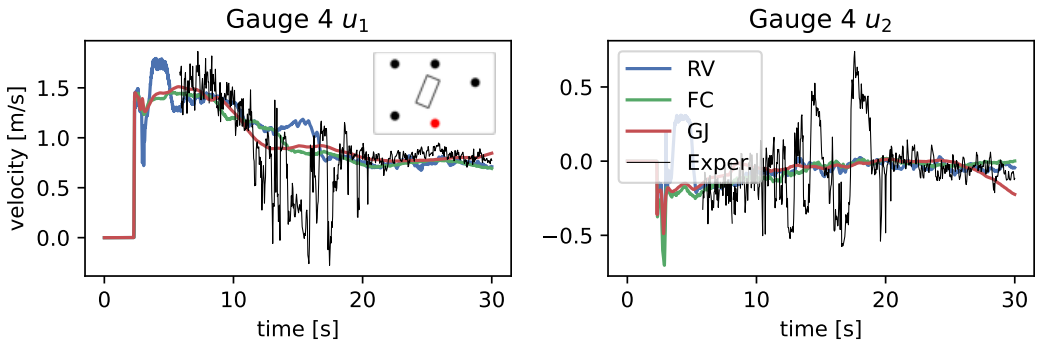


FIGURE 3.20: Experimental dam break flow. Comparison of velocity at gauge 4.

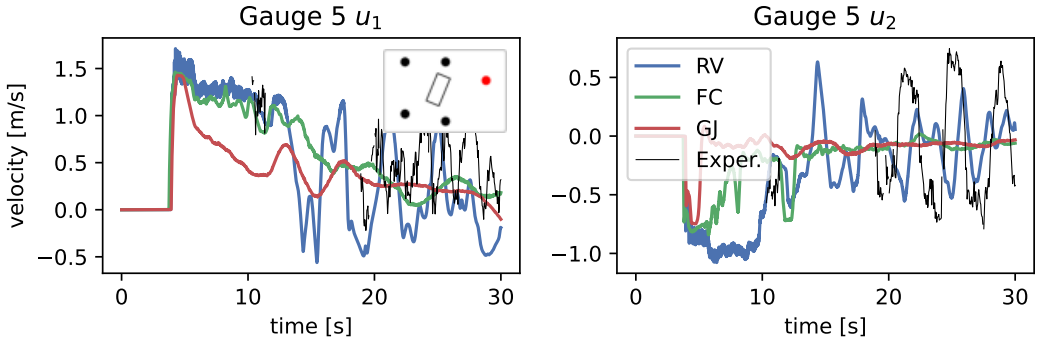


FIGURE 3.21: Experimental dam break flow. Comparison of velocity at gauge 5.

In fluid-structure interaction studies, pressure is needed to compute to forces over the structure. The shallow water assumptions drop the dynamic pressures retaining only the hydrostatic ones. In the present case, the pressures are recovered by integrating the hydrostatic pressures given the water depth around the building. However, there is no experimental data about the forces applied to the building and thus, the importance of the dynamic pressures can not be evaluated.

Regarding the performance of the FIC-FEM formulation, it captures the main aspects of the flow, but not the details, since there are some regions on that experiment which violate the shallow water approximations.

3.6 Concluding remarks

In this chapter the Finite Element approximation for the shallow water equations has been presented. The interpolation of the solution to the shallow water equations using Finite Elements induces some difficulties which have been addressed.

The first problem is the instabilities related to the *inf-sup* condition when the same interpolation is used for all the unknowns. The FIC-FEM formulation has been extended to the SW equations to overcome these instabilities. The main contribution is the projection of the characteristic length onto the linearization matrices $\mathbf{A}_i$. This procedure present an algorithmic constant and has been fixed to $\beta = 0.01$.

The second and third issues are closely related. One is the rise of local instabilities around hydraulic jumps and the last is the Gibbs oscillations at the shoreline. Both phenomena are solved introducing enough diffusion near the problematic regions, this procedure is known as non-linear stabilization. Some traditional methods aiming to provide local stability have been extended to the SW equations.

The *Residual Viscosity* (RV) method is firstly presented. It is the most consistent method among the three non-linear stabilizations presented in this chapter. Since the added viscosity depends on the residual, it introduces a non linearity and the order of convergence is lost in the regions where the non-linear stabilization is acting. Nevertheless, the global order of convergence is kept and good solutions are obtained. An algorithmic constant controls the amount of artificial viscosity introduced and it has been fixed to $\alpha = 1.0$.

The second non-linear stabilization presented in this chapter is the *Flux Corrected* (FC) algorithm, which combines two solutions. It could be used without linear stabilization, however, it is helpful to use a stabilized high order solution. Otherwise, phenomena such as terracing or excessive diffusivity, may appear. There are some difficulties with the definition of the convergence criterion. Even if the incremental displacements of the iterative solution are used as a measure of the convergence, the desired convergence is slowly achieved. The required number of iterations is higher than those required for a RV shock capturing. A practical option is to limit the iterations up to a fixed number. However, this approach will lack an accurate control of the obtained precision for transient problems. For example, the flow in front an obstacle is highly dynamic and it might require more iterations than the prefixed number.

Furthermore, the extension of the FC principles to the SW equations is not monotonicity preserving. Apart from the linear stabilization, the main responsible of the lack of monotonicity is the flux limiter. The flux limiters were initially designed for pure convection problems and when those are extended to the SW equations some limitations arise. Mainly, there is no a unique decomposition in pure convection problems. Then, the naive option has been chosen in this chapter: to choose the

water depth a indicator. This option lose the monotonicity but the solution is reasonably good without increasing the computational cost.

The computational cost of FC is high. Though the algorithm has to solve a system with the same number of degrees of freedom than a linear stabilized problem (3.2), the introduction of the corrections is non linear. Regarding the global solution and its robustness, good results are obtained. However, a better order of convergence is expected.

Finally, the *Gradient Jump Viscosity* (GJV) belongs to the same family than RV, since it it based on the introduction of viscosity. Though the extension of this method to more than one dimension is not monotonic, that is the less oscillatory method presented in this chapter. Its main drawbacks are the amount of additional viscosity and its lack of consistency with the source terms. Over-diffusive results have been observed in the last example, the dam break against an isolated building 3.5.3. Especially, the added diffusion inhibits the formation of shedding vortices behind the building. The lack of consistency is observed on the way the viscosity is added. Since the viscosity is not based on the residual but on the gradient jump, the presence of a high gradient on the topography will be translated on a region of additional diffusivity, even if the solution does not present a discontinuity.

To summarize, all the three methods have presented good results. There are slightly differences depending on the problem, if it is transient or stationary, if it presents a moving shoreline, if there are high topography gradients, etc. The preferred option is the RV method since it is consistent with the equations and it is derived from the linear stabilization. This method will be used in the following chapters.

Chapter 4

Finite Element Method for the Boussinesq modified equations

In this chapter, a FEM approximation for the Boussinesq modified equations is presented. Having a numerical approximation for the Boussinesq modified equations is mandatory when waves are studied on a two dimensional scenario and a complex bathymetry is considered.

Several numerical models appeared in the bibliography to solve the Boussinesq equations, under the possible variations for considering the dispersive terms. Abbot pioneered the FD schemes in [1, 2] applied to the original Boussinesq model. Later, Wei and Kirby solved the modified equations presented by Nwogu [136] using a FD scheme in [160]. The main advantage of the FD is the ease of treatment higher differences, even though the main difficulty consists on representing complex domains. Smallman was a pioneer applying the FD to harbours [154].

A decade later, some FEM were developed to solve Boussinesq equations. In that case, some special treatment needs to be applied to the high order derivatives. Langtangen [103] used quadratic triangles while most of the authors use an intermediate variable. Li and Liu [108] used bilinear quadrilaterals and used a projection to approximate the gradient. Walkley [158] and Woo [161, 162] used a projection to interpolate the second order derivatives filed. Surprisingly, the authors reported high frequency oscillations and proposed ad-hoc solutions, but until Codina [37, 39] none of them associated the oscillations to the incompatibility of interpolation.

The method has been solved using DG [57] and FV [18, 156]. As reported by Stansby [156], since the FV is non-oscillatory, it needs special techniques to accurately approximate the oscillatory behavior of the Boussinesq equations.

This chapter includes the developments presented in chapter 3 and presents the new techniques associated to the oscillatory and dispersive problem. Som examples are included in order to test the accuracy and to show the capabilities of the presented algorithms. Finally, the chapter is closed with the concluding remarks.

4.1 Stabilized formulation for the Boussinesq modified equations

As stated in section 2, the Boussinesq equations and the Saint Venant equations are of the same family. Apart from the choice of different primary variables, the main difference consists on the inclusion of the dispersive terms. However, part of the structure of the equations remains unmodified.

The weak principle, the linear stabilization and the spatial discretization presented in section 3 will be directly applied for the Boussinesq modified equations. The main differences arise from the dispersion terms. First of all, the dispersion

terms include spatial derivatives of order higher than two. Thus, special techniques for considering the third order derivatives will be included in the FEM procedure.

Secondly, the time discretization will be modified using a semi-implicit fourth-order scheme. The new time scheme allows to deal with the oscillatory behavior of the Boussinesq equations. Additionally, the matrix structure will suffer some modifications related to the high order derivatives and the time integration scheme.

Regarding the shock capturing, usually it is not needed since the dispersion effect prevents from breaking and the formation of steep gradients. However, in the vicinity of the shoreline or very shallow domains, the amplitude dispersion effect could dominate over the frequency dispersion, making very recommended the inclusion of a shock capturing technique. Furthermore, if the shoreline is included, the shock capturing is required, since it provides stabilization for the moving front.

On another note, a section regarding the numerical treatment of absorbing boundary conditions is included. The inclusion of the frequency dispersion and the need of shortening the computational domain, lead to the open boundary conditions. On an open boundary conditions, waves can exit the computational domain and reflections must be avoided. The numerical approximation for the open boundary conditions are called absorbing boundary conditions.

4.1.1 Weak formulation and linear stabilization

The Boussinesq equations are solved using the FEM. The background of chapter 3 allows to interpolate the space domain with a Galerkin discretization of linear triangles or bilinear quadrilaterals. A finite difference scheme with constant time step is used to integrate the equations in time.

The variational form of the Boussinesq modified equations (2.20) presented in section 2.4 reads

$$\int_\Omega \left(U\frac{\partial \mathbf{u}_\beta}{\partial t} + U\nabla\eta + U(\mathbf{u}_\beta\cdot\nabla)\mathbf{u}_\beta + U\mathbf{J}_u \right) d\Omega = \mathbf{0} \tag{4.1a}$$

$$\int_\Omega \left(E\frac{\partial \eta}{\partial t} + E\nabla\cdot\big((H+\eta)\mathbf{u}_\beta\big) + E\nabla\cdot\mathbf{J}_\eta \right) d\Omega = 0 \tag{4.1b}$$

where the auxiliary fields $\mathbf{J}_\eta$ and $\mathbf{J}_u$ are defined in (2.21) and involve second order derivatives of $\mathbf{u}_\beta$ and $\partial\mathbf{u}_\beta/\partial t$. E and U are the corresponding test functions for η and $\mathbf{u}_\beta$.

In spite of the staggered formulation for the mass and momentum conservation, it is still a mixed wave formulation and the linear stabilization obtained by the FIC procedure will be applied. The additional term associated to the FIC residual could be projected onto a more general linearization matrix, including also the effects of the dispersive terms. However, the dispersive terms include different derivatives order. Since the current FIC stabilization only considers first order derivatives, the new residual term will be projected onto the fluxes matrices $\mathbf{A}_i$:

$$\mathbf{A}_1 = \begin{bmatrix} u_{\beta1} & 0 & g \\ 0 & u_{\beta2} & 0 \\ H+\eta & 0 & u_{\beta1} \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} u_{\beta1} & 0 & 0 \\ 0 & u_{\beta2} & g \\ 0 & H+\eta & u_{\beta2} \end{bmatrix} \tag{4.2}$$

where each term is obtained by linearization of the Boussinesq modified equations. The derivation of the full variational stabilized formulation is straightforward and for the sake of simplicity it is not shown. Once the equations are stabilized, a numerical procedure to deal with the dispersive fields $\mathbf{J_u}$ and $\mathbf{J}_\eta$ needs to be defined.

4.1.2 High order derivatives and dispersive fields

Most of the authors (see [39, 108] as an example) use a projection for the intermediate variables. However, $\mathbf{J_u}$ only includes one time derivative and two spatial derivatives. Hence, it belongs to the finite element space and in can be integrated by parts.

Following [158], the third order spatial derivatives corresponding to the mass balance dispersion are modelled using $\mathbf{J}_\eta$ as an intermediate variable. There are several ways of computing the intermediate field $\mathbf{J}_\eta$. The first possibility is to compute it in a monolithic way, adding $\mathbf{J}_\eta$ to the degrees of freedom. A more interesting option is to compute it in a staggered way using a projection. In that case, the number of degrees of freedom is not increased and thus the time consuming performance of the scheme is unaltered.

Finally, the projection for the auxiliary field $\mathbf{J}_\eta$ can be replaced by a gradient recovery technique. In that section this option is analyzed, since the gradient recovery provides more regularity, at the same time, it is very efficient.

The gradient recovery technique was initially proposed by Zienkiewicz and Zhu in [173]. Later, Zhang and Naga proposed some improvements for the super convergent gradient recovery in [169] at boundaries. Finally, some improvements have been proposed in [3, 163] for adaptive meshes, but the last improvements are not going to be considered in this research. The basis of the gradient recovery consists on fitting a $k+1$ polynomial degree where the finite elements space is of degree k.

4.1.2.1 Derivatives recovery algorithm

The main idea of the derivatives recovery is to introduce an operator $G_h : S_h \to S_h^{n_d}$, where S_h is a polynomial finite element space of degree k over a triangulation T_h and n_d is the space dimension. After determining the values of G_h at all nodes, given a solution u_h , $G_h u_h \in \mathbb{S}_h^{n_d}$ is obtained on the whole domain. The triangulation can be replaced by a quadrilateral discretization. In both cases, only vertices nodes are considered, nor edge or internal nodes.

For a vertex $\mathbf{x}_i$ and being l_i^e the size of the largest edge attached to $\mathbf{x}_i$, a ball is defined around it:

$$B_{l_i^e}(\mathbf{x}_i) = \{\mathbf{x} \in T_h : |\mathbf{x} - \mathbf{x}_i| \leq l_i^e\} \tag{4.3}$$

and all the vertices inside the ball are selected. If the number of nodes n_n inside the ball $B_{l_i^e}(\mathbf{x}_i)$ is less than $m = (k+n_d)(k+n_d+1)/2$, then the ball is extended to two times the edge length, $B_{2l_i^e}$. This process of increasing the radius is repeated until there are enough nodes in the ball. The nodes inside a ball are denoted as $\mathbf{x}_{ij}$ and the polynomial p_{k+1} is fitted using local coordinates with $\mathbf{x}_i$ at the origin and the scaling parameter $l = l_i^e$,

$$p_{k+1} = \mathbf{P}^T \mathbf{a} = \hat{\mathbf{P}}^T \hat{\mathbf{a}} \tag{4.4}$$

where

$$\mathbf{P}^T = (1, x, y, x^2, ..., x^{k+1}, x^k y, ..., y^{k+1}), \quad \hat{\mathbf{P}}^T = (1, \xi, \eta, \xi^2, ..., \xi^{k+1}, \xi^k \eta, ..., \eta^{k+1})$$

$$\mathbf{a}^T = (a_1, a_2, ..., a_m), \quad \hat{\mathbf{a}}^T = (a_1, l a_2, ..., l^{k+1} a_m),$$

And the coefficient vector $\mathbf{a}$ is determined by the linear system

$$A^T A \hat{\mathbf{a}} = A^T \mathbf{b}_h \tag{4.6}$$

where $\mathbf{b}_h^T = (u_h(\mathbf{x}_{i1}), u_h(\mathbf{x}_{i2}), ..., u_h(\mathbf{x}_{in_n}))$ and

$$A = \begin{pmatrix} 1 & \xi_1 & \eta_1 & ... & \eta_1^{k+1} \\ 1 & \xi_2 & \eta_2 & ... & \eta_2^{k+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \xi_{n_n} & \eta_{n_n} & ... & \eta_{n_n}^{k+1} \end{pmatrix}$$

A condition for (4.6) having solution is that $Rank(A) = m$, which is practically always satisfied when $n_n > m$ and the mesh has a reasonable quality. Finally, it is possible to define the gradient at the point $\mathbf{x}_i$ as

$$G_h u_h(\mathbf{x}_i) = \nabla p_{k+1}(0, 0, \mathbf{x}_i) \tag{4.7}$$

Similarly, the gradient of a divergence can be obtained when the polynomial is applied to a vector field. This procedure can be applied up to derivatives of order $k+1$, which for linear elements are second-order derivatives. Finally, the gradient of the divergence reads

$$L_h u_h(\mathbf{x}_i) = \nabla \nabla \cdot p_{k+1}(0, 0, \mathbf{x}_i) \tag{4.8}$$

Since the values of the nodal unknowns u_h are not known a priory, the vector of coefficients $\mathbf{a}$ will be expressed as a linear combination of $u_h(\mathbf{x}_{ij})$. Depending on which derivatives of $\mathbf{P}$ are seek, the linear combination will be some selected rows of $(A^T A)^{-1} A^T$.

4.1.2.2 Discrete counterpart

Finally, the combination of the discrete variational principle of equations (4.1) with the gradient recovery technique yields

$$\int_\Omega \left(U_h \frac{\partial \mathbf{u}_{\beta h}}{\partial t} + U_h \nabla \eta_h + U_h (\mathbf{u}_{\beta h} \cdot \nabla) \mathbf{u}_{\beta h} + \right.$$
$$\left. \nabla U_h \left(C_2 H^2 \nabla \cdot \frac{\partial \mathbf{u}_{\beta h}}{\partial t} + C_4 H \nabla \cdot \left(H \frac{\partial \mathbf{u}_{\beta h}}{\partial t} \right) \right) \right) d\Omega = \mathbf{0} \tag{4.9a}$$

$$\int_\Omega \left(E_h \frac{\partial \eta_h}{\partial t} + E_h \nabla \cdot \left((H + \eta_h) \mathbf{u}_{\beta h} \right) + \right.$$
$$\left. E_h \nabla \cdot \left(C_1 H^3 L_h \mathbf{u}_{\beta h} + C_3 H^2 L_h (H \mathbf{u}_{\beta h}) \right) \right) d\Omega = 0 \tag{4.9b}$$

Definition (4.9) completes the spatial discretization of the problem. It is important to recall that the stabilization terms are omitted but should be introduced in the algebraic formulation.

Moreover, in order to extend the present formulation to the shoreline, three modifications are made. Firstly, the terms not including time derivatives are multiplied by the et fraction w. Secondly, a residual-based viscosity stabilization is added, acting as a shock capturing where waves break and steep gradients are formed. Lastly, an artificial damping in the dry domain is added to the momentum balance in order to freeze the flow in dry areas.

4.1.3 Time discretization

The discrete formulation (4.9) is expressed in matrix form as

$$(M + K)\dot{x} = F(x) \tag{4.10}$$

where x is the vector of nodal unknowns, M is a mass matrix, K correspond to the second derivatives associated to $\mathbf{J_u}$, and F is a non-linear vector which is a function of $\mathbf{u}_\beta$, η and $\mathbf{J}_\eta$.

Therefore, the system of ordinary equations needs to be integrated in time taking account of its non-linearity. The commonly used fourth order Adams-Moulton scheme is employed for the time integration of (4.10), see [39, 160, 161] as an example. Let us consider a partition of the time interval $[0, T]$ into time steps of, for simplicity, equal size δt. We denote with superscript n the approximation of a function at time $t^n = n\delta t$. To obtain a solution at the time step t^{n+1}, the iterative fixed point method is employed to deal with the non-linearity of the system. Given a guess $x^{n+1,i-1}$ for $x^{n+1,i}$ at iteration i, the increment δx^i is computed from

$$\begin{aligned} 24(M + K)\delta x^i = 24(M + K)(x^n - x^{n+1,i-1}) \\ + \delta t(9F^{n+1} + 19F^n - 5F^{n-1} + F^{n-2}) + O(\delta t^4) \end{aligned} \tag{4.11}$$

The time integration is closed using the explicit third order Adams-Bashforth scheme to predict the initial guess $x^{n+1,0}$ for the non-linear iterations

$$12(M + K)x^{n+1,0} = 12(M + K)x^n + \delta t(23F^n - 16F^{n-1} + 5F^{n-2}) + O(\delta t^3) \tag{4.12}$$

The system of (4.11) and (4.12) is solved using a direct solver. Given that usually, the problems solved by the Boussinesq equations are small compared with fully resolved model, the use of a direct solver will not be a problem in terms of time consumption. Convergence is achieved when δx^i is sufficiently small. Then, the solution x^{n+1} is set as $x^{n+1,i}$ and the procedure is repeated for the next time step.

4.2 Absorbing boundary conditions

Frequently, the need to shorten the numerical domain arises. This can be achieved by the imposition of open boundaries, also known as radiant boundaries. The open boundaries allow the exit of the waves as well as the consistency of the system of equations in order to ensure the existence and uniqueness of the solution. Additionally, the numerical tool for the boundary has to be compatible with the numerical approach for the inner domain.

Generally, the radiant boundary condition denotes the analytical formulation for open boundaries and the term absorbing boundary is related to the numerical approximation of the radiation condition [134]. An equilibrium between the precision offered by the radiant boundary and the numerical cost required by the absorbing boundary is seek. From one side, the boundary conditions must define a well posed problem and the spurious oscillations in the open boundary shall be as small as possible. From the other side, the computational cost of the boundary conditions should be small compared to the computational cost of the inner domain.

Considering the unidirectional wave propagation and being η the free surface elevation, a wave function will verify a radiant boundary condition where

$$\frac{\partial \eta}{\partial t} + c\cos(\theta)\frac{\partial \eta}{\partial x} = 0 \tag{4.13}$$

where θ is tha incidence angle of the wave with respect to the boundary.

Some authors proposed local approximations highly diffusives [56]. Latter, Bayliss explored the radiation boundary conditions for the Helmholtz equations in [12]. Similarly, Collino [40] extended the formulation using higher order of derivatives than 4.13. However, the generalization for the Euler equations -analogously the SW equations- in 2 or 3D is not trivial, since the incidence angle θ is not known *a priori* and because of the dispersive behavior of the equations [160].

The dispersive behavior is associated to the fact that there is not an unique celerity characterizing the system: in the case of the SW or the Boussinesq equations there are three waves superposed propagating at different speeds. For example, in section 3.3 a decomposition in characteristics was presented through diagonalization of the tangent matrices $\mathbf{A}_i$.

$$\frac{\partial \mathbf{\Phi}_j}{\partial t} + \mathbf{\Lambda}_i \frac{\partial \mathbf{\Phi}_j}{\partial x_i} = 0 \tag{4.14}$$

in that minimal wave equation $\mathbf{\Lambda}$ is a diagonal matrix such that $\boldsymbol{A}_i = \boldsymbol{T}_i \mathbf{\Lambda}_i \boldsymbol{T}_i^{-1}$. This problem can be understood a the superposition of three waves, each one of them propagating at speed $u - c$, u y $u + c$, namely the eigenvalues λ_i. The sign of λ_i determines if the wave is incoming or outgoing and the regime of the problem: subcritical or supercritical.

Besides having to impose the radiant boundary condition for the three waves, this approximation is limited, since in 2 or 3D does not exist a genuine characteristic-based formulation. A more detailed development can be found in [109].

Most of the proposals to approximate the open boundaries consist on relaxing the condition 4.13 adding a dissipation term K and extending the domain a distance d, losing the local definition. See [29, 92, 134] as an example. Then, the radiation condition is rewritten as

$$\frac{\partial \eta}{\partial t} + c\cos(\theta)\frac{\partial \eta}{\partial x} = -K\eta \tag{4.15}$$

This approximation can be used in combination with local absorbing boundary conditions [160].

Another of the explored possibilities is the Perfectly Matched Layer (PML) [14], widely used in the literature. It presents the inconvenient of special treatment of corners. Later, this methodology wa used in combination with the high order absorbing boundaries, leading to the Double Absorbing Boundary (DAB), proposed by Hagstrom and Rabinovich in [73, 148]. This technology places two parallel boundaries separated by a short distance. Its advantage is that they do not need to incorporate the derivatives in the normal direction neither to apply any special treatment to the corners.

Finally, and given to the simplicity of the formulation, there is the possibility of including only a sponge layer. It can be found in the formulations of Israeli and Carmigniani in [29, 92]. A Newtonian dissipative term analogous to the bottom friction S_f is added as

$$S_a = -\gamma(\mathbf{x})\mathbf{u} \tag{4.16}$$

The dissipative term γ varies from 0 at a given distance to the discrete boundary to

the maximum value γ_{max} at the discrete boundary. This variation follows an exponential law [145] of order n. The parameters of the sponge layer d_0 and the maximum value γ_{max} need to be specified for each case, in terms of the wavelength. A possible expression for γ is

$$\gamma(\mathbf{x}) = \gamma_{\max}\mathcal{H}(d_0 - d(\mathbf{x}))\frac{e^{\left(\frac{d_0-d(\mathbf{x})}{d_0}\right)^n} - 1}{e-1} \tag{4.17}$$

where $\mathcal{H}$ is the Heaviside function, d is the distance from a point to the computational boundary and the exponent n is a constant which is taken as 3.

4.3 Examples

The examples included in this section are devoted to test the accuracy of the presented formulation. In the first example, the propagation of a solitary wave along a uniform chanel is analyzed and compared against an analytical solution. In the second example, the accuracy of the absorbing boundary conditions is studied, and a sensitivity analysis of the damping parameters is performed. More complex examples can be found in chapter 5

4.3.1 Solitary wave propagation

Analytical solutions for the Boussinesq equations can be found under specific situations, such as solitary or regular waves and uniform topography. If the two dimensional case is considered, symmetry is required. In that case, we select a solitary wave propagating along a channel with constant depth. The analytical solution found bw Wei in [160] is employed to generate the solitary wave and to check if it propagates correctly. The numerical wave is compared against the analytical wave at several points in the channel.

4.3.1.1 Analytical solution

Wei obtained the analytical solution for a solitary wave by rewriting the equations (2.20) in dimensionless form, substituting the velocity by its potential (φ such that $u = \partial\varphi/\partial x$) and assuming an hyperbolic solution for the velocity, $\varphi' = A\,\mathrm{sech}^2(B\xi)$. Terms of order $O(\epsilon^2, \epsilon\mu^2)$ are truncated. Finally, after recovering the dimensional form, the analytical solution reads

$$u = A_0\,\mathrm{sech}^2\phi \tag{4.18}$$

$$\eta = A_1\,\mathrm{sech}^2\phi + A_2\,\mathrm{sech}^4\phi \tag{4.19}$$

where

$$\begin{aligned} A_0 &= \frac{C^2-1}{C}\sqrt{gh} \\ A_1 &= \frac{C^2-1}{3(C_2+C_4+1/3-C^2(C_2+C_4))}H \\ A_2 &= -\frac{(C^2-1)^2}{2C^2}\frac{C_2+C_4+1/3+2C^2(C_2+C_4)}{C_2+C_4+1/3-C^2(C_2+C_4)}H \\ \phi &= k(x-ct) \end{aligned} \tag{4.20}$$

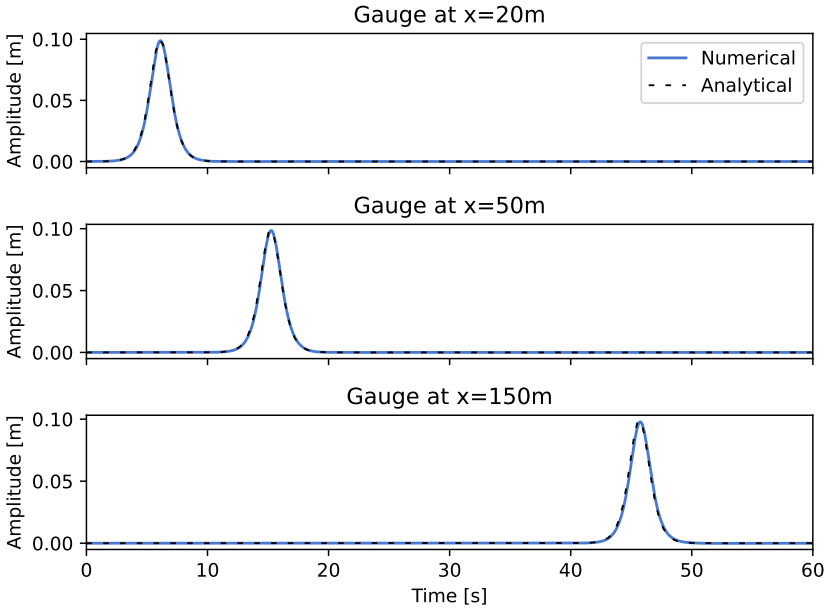


FIGURE 4.1: Solitary wave propagation. Time series at fixed positions.

The constants C_2 and C_4 are part of the Boussinesq equations and are defined in (2.22). ϕ is the phase function, c is the phase speed and k is the wavenumber. The dimensionless phase speed C is computed as the root of the polynomial

$$2(C_2+C_4)C^6-\left(3C_2+3C_4+1/3+2\frac{\eta_{\max}}{H}(C_2+C_4)\right)C^4 + 2\frac{\eta_{\max}}{H}(C_2+C_4+1/3)C^2+C_2+C_4+1/3=0$$

Finally,

$$c=C\sqrt{gh} \quad \text{and} \quad k=\frac{1}{H}\sqrt{\frac{C^2-1}{4(C_2+C_4+1/3-C^2(C_2+C_4))}}$$

The solitary wave is characterized by two physical parameters, the maximum amplitude $\eta_{\max}$ and the water depth of the channel H.

4.3.1.2 Numerical results

A rectangular domain $\Omega=[0,100m]\times[0,1m]$ is considered. The time interval considered is $t\in[0,20s]$. The full signal $-\eta$ and $u-$ given by the analytical solution is imposed at the left boundary Γ_{left} during all the temporal domain. Reflecting boundary conditions are considered at the rest of boundaries, $\Gamma_R=\Gamma\cap\Gamma_{\text{left}}$. The water depth is $H=1m$ and several amplitudes as considered, $\eta_{\max}=0.1$, 0.3 and $0.3m$.

Both triangular and quadrilateral discretizations of $l^e=0.3m$ are considered. The time step is constant during all the simulation and is computed in order to keep a $CFL\leq 0.5$. The resultant time step is $\Delta t\approx 0.02s$.

The propagation of the smallest solitary wave is recorded at Figure 4.1. Three time series are shown, near the generation of the solitary wave ($20m$), after a short

period of propagation ($50m$) and after a long distance ($150m$). For that wave specifications, $\epsilon = 0.1$ and $\mu = 0.04$ there is no appreciable variation in the wave amplitude nor in the phase speed from theory.

The amplitude variations of the three solitary waves is analyzed in detail in Figure 4.2. For each wave, two time series are compared against the analytical solution. The shapes of the solitary waves are compared centered at the phased and using dimensionless coordinates. The important characteristic of the numerical method is that the wave travels a long distance without being affected by numerical dissipation. The main difference with respect to the analytical solution is that the predicted phase speed is somewhat small. This difference is bigger as the wave amplitude increases. It would be interesting to compare the analytical solution against experimental results.

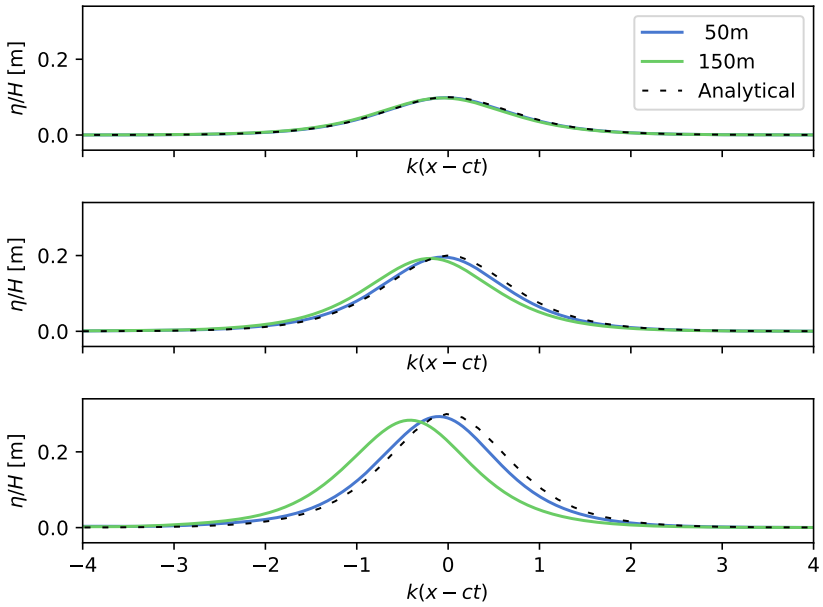


FIGURE 4.2: Solitary wave propagation. Shape modification after propagation. Several amplitudes are compared.

The obtained results are comparable to those exposed in [160]. There is no appreciable damping on the wave amplitude and the phase speed is underestimated for high wave amplitudes.

4.3.2 Absorbing boundary

In this example the accuracy of absorbing boundary conditions is analyzed. Let $\Omega = [0, 100m] \times [0, 1m]$ be the spatial domain $[0, 100s]$ the temporal domain. This correspond to a $1m$ width and $100m$ long channel, the considered water depth is $1m$. At the lef end of domain waves are generated during the interval $[0, 30s]$ and at the right end of the channel there is an absorbing boundary condition. Fig. 4.3 shows the propagation of the train of waves and the absorption at the right end of the channel.

The train of waves is monochromatic with $0.075m$ amplitude and $2s$ period. The corresponding wavelength is $\sim 5m$. In order to avoid the generation of higher frequencies, the signal has been imposed gradually during the first two periods and decreased gradually during the last two periods.

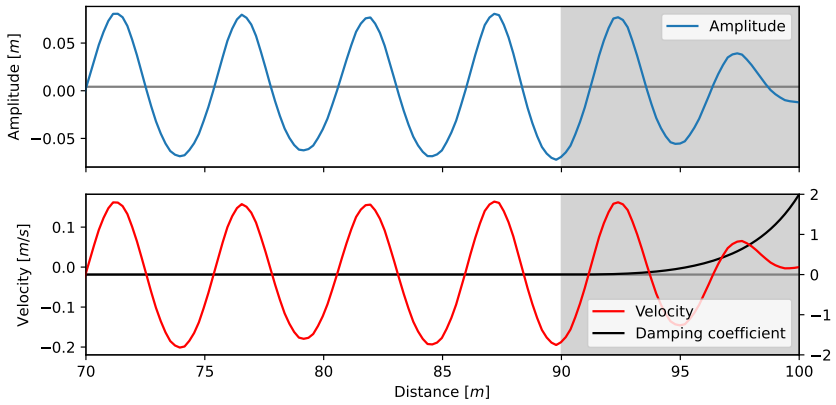


FIGURE 4.3: Absorbing boundary condition. Propagation and absorption of a train of waves. The shadow region shows the width of the sponge layer.

Regarding the absorbing boundary, studies carried out by Carmigniani in [29] show that some combinations of the maximum absorption, the layer width and the exponential degree can lead to undesired reflections. For example, an excessive absorption can generate reflections at the beginning of the sponge layer.

An exponent $n = 3$ for expression 4.17 is chosen, it considerably minimizes the reflection [29]. A sensitivity analysis for the parameters d_0 and $\gamma_{\max}$ is performed. It is useful to express the maximum distance and absorption in terms of the wavelength and the frequency.

$$d_0 = D\lambda\,, \quad \gamma_{max} = B\omega$$

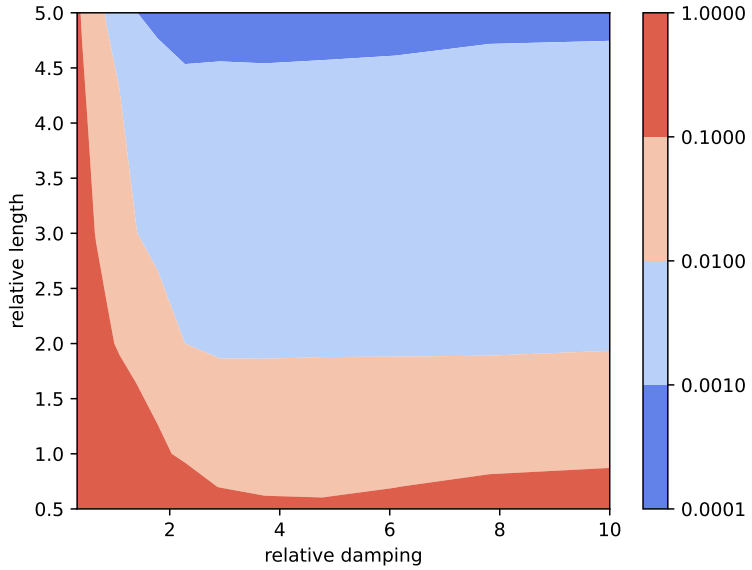


FIGURE 4.4: Absorbing boundary condition. Computed reflection coefficients.

Several numerical experiments have been carried out with a set of relative dampings and relative lengths. A total of 150 combinations have been simulated in order

to cover the region $B \in [1, 10] \times D \in [0.5, 5]$. The computed reflection coefficients are summarized in Fig. 4.4.

The reflected amplitude is monotonically decreasing as the sponge width increases. Nevertheless, the reflection exhibits a local minimum with respect to the absorption coefficient. This makes highly recommended to adjust the parameters of the sponge layer according to the maximum wavelength. In practice, the width of the sponge layer is used to be $d_0 \approx 2\lambda$ and the absorption coefficient $\gamma \approx 4\,\omega$.

4.4 Concluding remarks

In this chapter, a finite element approximation of the Boussinesq modified equations has been presented. From the numerical point of view, the model has been approximated by the FIC-based stabilization and a semi-implicit predictor-corrector scheme has been used to advance in time. High order derivatives are approximated using an intermediate projection, computed with the superconvergent gradient recovery technique. The numerical approximation of the radiation boundary condition is known as absorbing boundary and it has been modelled using a sponge layer.

The accuracy of the method, has been tested for linear triangles and bilinear quadrilaterals. The FEM formulation has been tested with the simulation of a solitary wave and comparing it against an analytical solution. The effectivity of the absorbing boundary has been analyzed with a train of regular waves in a channel.

The proposed method has proven to be accurate and the introduction of numerical dissipation is small. The results are comparable to the ones proposed in the literature [39, 136, 158, 160]. The absorbing boundary conditions have been improved.

Chapter 5

Coupling the Navier-Stokes Equations to the Shallow Water Equations

5.1 Introduction

The aim of this chapter is to explore the possibilities of coupling the presented shallow water models with the Navier-Stokes equations. In this research, the developments of chapters 3 and 4 are combined with a NS solver, and have been presented in [118]. Having all the formulations within the same framework -FEM and implementation- ease the strategy. The first part of the staggered approach presented in chapter 1, an action triggering an event is modelled with the Navier-Stokes equations and the information is transferred to the SW domain. This first coupling is applied to impulse waves generated by landslides.

Landslide Generated Waves (LGW) are caused by a landslide impacting a water reservoir. LGW events can have devastating effects on the coastal areas of water basins, such as lakes, fjords and artificial reservoirs. The fjord district of western Norway is one of the zones of the world most affected by this major natural hazard [79]. Historical records over the last 400 years show that Norway has experienced at least two major LGW events every century [76]. Only in the first half of the last century, the catastrophic events of Loen (in 1905 and 1936) [69] and Tafjord (in 1934) [80] caused the death of 174 people. The LGW events of Lituya Bay, Alaska, in 1958 [125] and Vajont, Italy, in 1963 [153] are among the most well-known cases of this cascading natural hazard. A wider overview of LGW historical events can be found in [150]. Furthermore, this situation is made even more critical by the effects of global warming, which is clearly leading to an increment in number and intensity of natural disasters [75].

In this new strategy, the Particle Finite Element Method (PFEM) [46, 88, 139], is used as the NFS and a the Boussinesq model presented in chapter 4 is used as the FFS. Several previous works have shown the accuracy of the PFEM to model landslides [165–167], also in cascading events [44, 45, 151, 168]. In this work, we use the PFEM approach that has been successfully applied to LGW scenarios in [127] and in [61, 62], where 3D simulations of the Vajont disaster were presented. This work aims at being a proof of concept of this new coupled strategy for real LGW scenarios. For this reason, a deep validation of the method is presented by analyzing the performance and accuracy of the new partitioned technique in targeted tests, using reference solutions obtained with other numerical methods, experimental tests and analytical solutions. In partitioned methods, the momentum transfer between the Navier-Stokes and the Boussinesq models must be accurate in order to obtain a

faithful representation of the LGW scenario. Thus, particular attention is devoted here to analyze the effect of the near-field boundary conditions on the far-field propagating wave. Convergence and sensitivity analyses are carried out in order to verify the accuracy and robustness of the proposed method.

In this chapter, after explaining the staggered strategy, it is analyzed in detail. Three examples are provided in order to validate the proposed partitioned strategy and to show its potential to large-scale LGW events.

5.2 Coupling strategy

The LGW problem is here simulated using a weakly coupled (one-way) method which makes interact the PFEM solver presented in Section 1.3, and the SW solver described in Chapters 3 and 4.

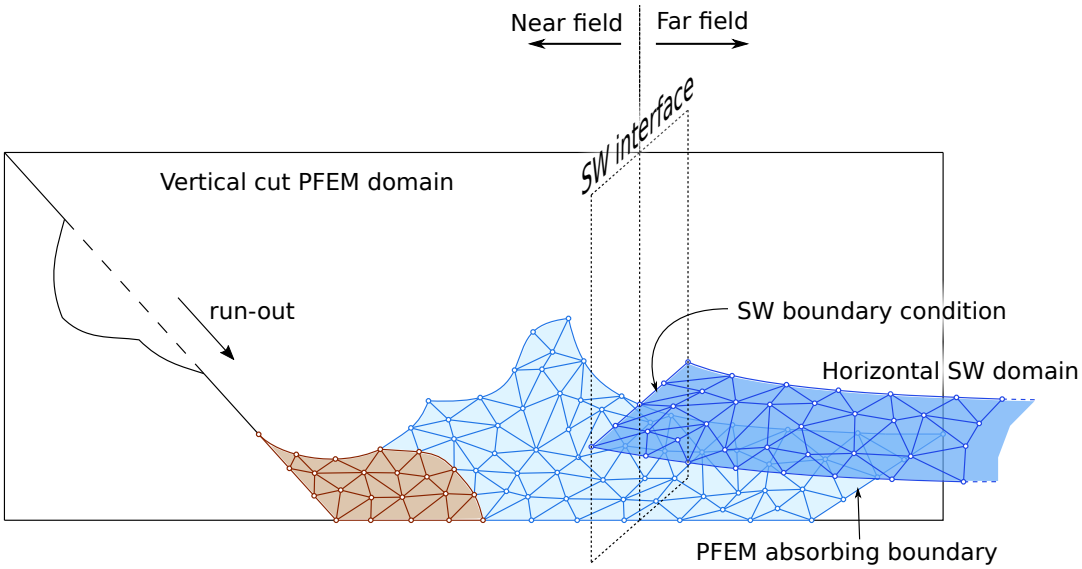
We note that problem (2.20) defines a phase speed c which in the SW limit is computed as $\sqrt{gh}$. Being u the modulus of the horizontal velocity $\mathbf{u}_\beta$, the information travels at velocities $u + c$ and $u - c$. If the flow is subcritical, which is the case analyzed in this work, then $u < c$ and the information travels both upstream and downstream.

Considering the bidirectional characteristics of the equations, a first possibility would be to consider two domains adjacent to each other and define a strong (two-way) coupling (see for example [146]). However, such a strongly coupled approach is computationally expensive, as it requires running parallelly the PFEM and the SW solvers.

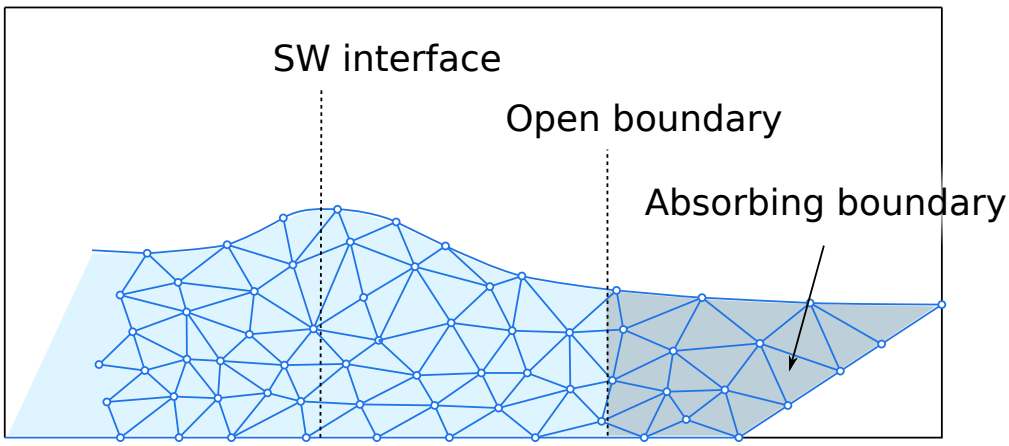
Moreover, the accuracy increment given by a two-way method over a one-way strategy on the far-field wave propagation can be considered negligible. For these reasons, here, we adopt a decoupled space-time one-way strategy in which the PFEM solution is stored at the SW interface and, in a second stage, it is imposed as a boundary condition for the SW simulation. In order to avoid perturbations of the results at the interface, the computational domain of the PFEM is extended beyond the position of the SW interface using non-reflecting boundaries.

Taking advantage of such a one-way coupling strategy, the PFEM and SW simulations can be executed independently leading to a very versatile tool for LGWs with significant saves of computing time. Since the PFEM is a Lagrangian strategy, a search algorithm is constructed at every time step in order to find all the elements cut by the SW interface. Then, the PFEM calculations beyond the interface are not relevant. This fact is the key to the computational savings, since the computational domain can be shortened by means of an open boundary. However, the numerical approximation of open boundaries –the absorbing boundaries– introduces some reflections. In this work, the absorbing boundary is modelled by extending the domain after the open boundary with a gentle slope. The computational domain ends when the slope reaches the mean water level, at this point, the impulse waves leave the computational domain.

In a later stage, the characteristic variables computed at the interface are imposed to the SW domain through an inflow boundary condition. We recall the subcritical characteristics of the analyzed flows, hence, one variable is required to be imposed in order to define a well-posed problem: the wave amplitude or the horizontal velocity. We choose to impose the velocity, since it is more representative of the momentum exchange from the PFEM and the SW computation. It has proven to be accurate, even when the Boussinesq assumptions are not perfectly fulfilled. A general picture of the coupling strategy is illustrated in Fig. 5.1.



(A) Scheme of a 2D-PFEM-SW coupled strategy.



(B) Detail of the open boundary.

FIGURE 5.1: Schematic view of the *near-field solver* and the *far-field solver* for the coupled solution of LGW.

Even though the Boussinesq equations are expressed in terms of the velocity evaluated at a certain depth, $\mathbf{u}_\beta$, this magnitude is a measure of the depth-averaged velocity $\bar{\mathbf{u}}$. In other words, it can be understood as a numerical quadrature of one integration point. When the waves are regular, the choice of one magnitude or another is not relevant, but when wave breaking is present, the depth-averaged velocity is more representative of the momentum exchange.

We remark that the average vertical velocity of the fluid corresponds to the time derivative of the free surface elevation. This variable does not correspond to a boundary condition for the studied cases.

Finally, there is an additional condition associated to Γ_I (see Chapter 4): the dispersive field $\mathbf{J}_\eta$ relates $\bar{\mathbf{u}}$ and $\mathbf{u}_\beta$. The assumption of equal velocities is equivalent to imposing $\nabla\nabla\cdot\mathbf{u}=\mathbf{0}$.

5.3 Examples

In this section, three different cases are presented. These examples are selected to validate the partitioned strategy and to show its potential for practical applications. The first numerical example is aimed at reproducing a unidirectional wave generated in a laboratory channel. For this test, we carry out a detailed validation of the coupled method paying special attention to the transmission of boundary conditions between the near- and the far-field solvers. The simplicity of this test allows us to compare our results with both experimental measures and analytical solutions, and also with the numerical solution obtained with a full PFEM model. In the second example, we apply the partitioned method to a more representative example of LGW problems. In this test, we reproduce numerically the water wave generated experimentally by the impact of a second mass of water sliding at high velocity over a steep slope. The last test aims at showing the applicability of the method to real-world LGW problems. For this purpose, we considered a realistic configuration of a LGW event occurring in an alpine lake. Our numerical solution is compared to another LGW solver presented in the literature.

5.3.1 Solitary wave in a channel

In this test, we reproduce the laboratory experiment carried out at a large wave flume of the Coastal Research Center in Hannover. A solitary wave is generated by a piston-type maker and travels 180m until reaching the final inclined slope. A schematic view of the wave flume is depicted in Fig 5.2. More details about the experiment can be found in [96–98].

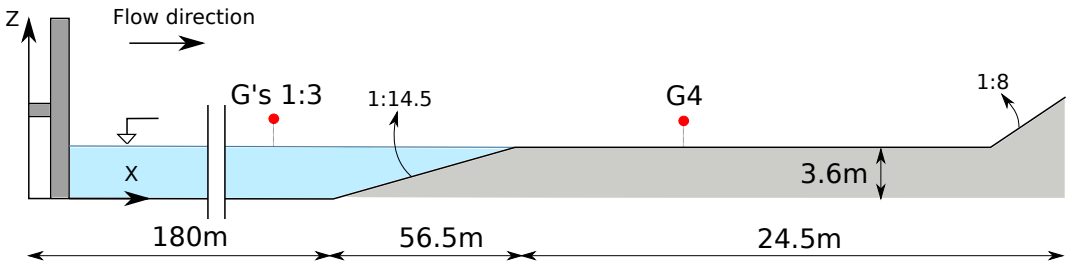


FIGURE 5.2: Solitary wave example. Schematic side view of the experimental flume studied. Units in m. Approximate position of the different wave gauges are also depicted.

Fig. 5.2 shows the horizontal stroke of the paddle along time. The wave height has been monitored at different positions of the flume, including the on-shore zone. In this work, we will compare our numerical solution to the experimental measures obtained at the four wave gauges whose coordinates are given in Table 5.1. The selected gauges are placed at key positions of the channel and allow us to monitor wave generation (G1), propagation (G2), shoaling (G3) and flooding (G4).

5.3.1.1 Physical considerations

The aforementioned specifications generate a solitary wave of $0.6m$ amplitude and $65m$ wavelength. The wave generation, propagation and breaking were analyzed using the PFEM approach reported by Oñate et al. [143]. Given the properties of such a solitary wave, it can be simulated using the Boussinesq approximation and thus reducing drastically the computational demand. This experiment is very interesting

Gauge	Position [m]
G1	60.0
G2	170.0
G3	223.5
G4	239.7

TABLE 5.1: Solitary wave example. Position of the different gauges in the flume.

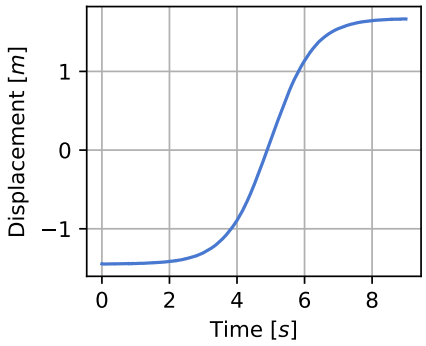


TABLE 5.2: Solitary wave example. Paddle position according to time. Data provided in Krautwald et al. [96–98].

for two reasons. Firstly, we can perform a verification test of both formulations and compare the numerical results against experimental data. Secondly, the simplicity of the geometry allows us to obtain analytical solutions for the Boussinesq equations. The analytical solution is a wave equation of the type

$$u = A_0 \text{sech}^2 \phi$$
$$\eta = A_1 \text{sech}^2 \phi + A_2 \text{sech}^4 \phi$$

where $\phi = kx - \omega t$. Details of the parameters A_0, A_1 and A_2 and the relation between the wavelength, period and amplitude can be found in [160].

The generation of solitary waves has motivated several discussions and a review can be found in [72]. The kinematic description of the piston wave maker is the origin of the discussion, since it cannot represent the exact solution of a solitary wave due to construction limitations. Some expressions for the motion of the piston can be obtained by integrating the analytical solution of the wave and truncating it on a finite space and time domain, corresponding to the features of the piston. Then, the experimental solitary wave is generated with a tail of secondary oscillations.

The Lagrangian formulation of PFEM perfectly tracks the movement of the paddle and thus the numerical simulation reproduces the experimental results with high fidelity. On the other hand, since the Boussinesq equations are implemented in an Eulerian frame, this boundary condition is difficult to impose. An easier alternative is to apply the analytical solution as a boundary condition.

Fig. 5.3 shows the comparison between the solitary wave propagation obtained with the PFEM, the experimental results, and the Boussinesq and analytical solutions. The Boussinesq simulation shows no secondary oscillations because the solitary wave has been imposed perfectly. The PFEM analysis matches the experimental data and the SW analysis matches the analytical solution. The analytical solution overestimates the phase speed and this mismatch will be reflected in the following analyses.

We remark that the difference in the phase speed of the wave is not originated by the coupling strategy, but by the Boussinesq approximation. The accuracy of the approximation depends on the non linearity ratio $\epsilon = \eta/H$ and dispersion ratio $\mu = H/\lambda$. A more detailed study can be found in [164], particularly when $\epsilon < 0.4$.

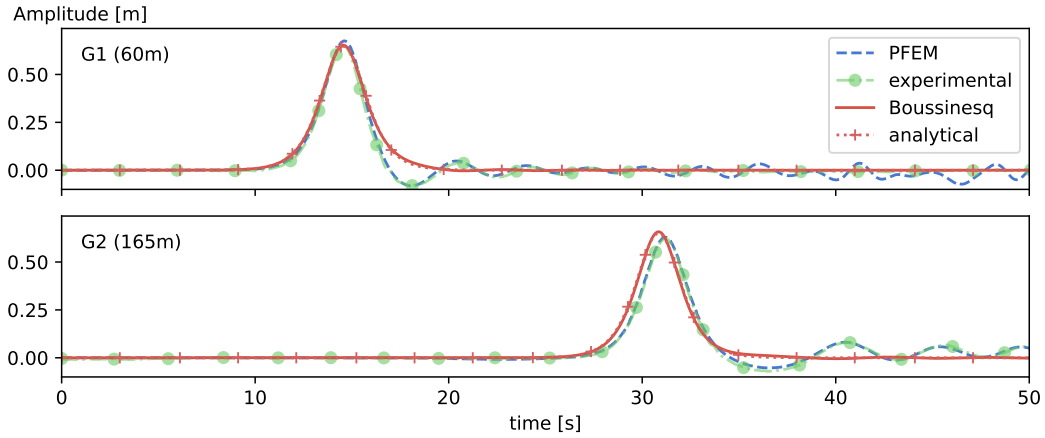


FIGURE 5.3: Solitary wave example. Time evolution of the free surface at two gauges.

5.3.1.2 Numerical results of the coupled strategy

A global representation of the wave propagation is found in Fig. 5.4. In this simulation, the first $10m$ are simulated using the 2D PFEM and the rest of the channel is simulated using the SW solver. Additionally, the full channel has been simulated with the PFEM to provide a reference solution for the coupled method and to analyze better its performance. Concerning the space and time discretizations used in the two solvers, the PFEM domain has a mesh of mean size $\Delta x = 0.3m$ and the time step increment $\Delta t = 0.001s$ is used, while the SW domain is discretized with $\Delta x = 0.8m$ and $\Delta t = 0.025s$.

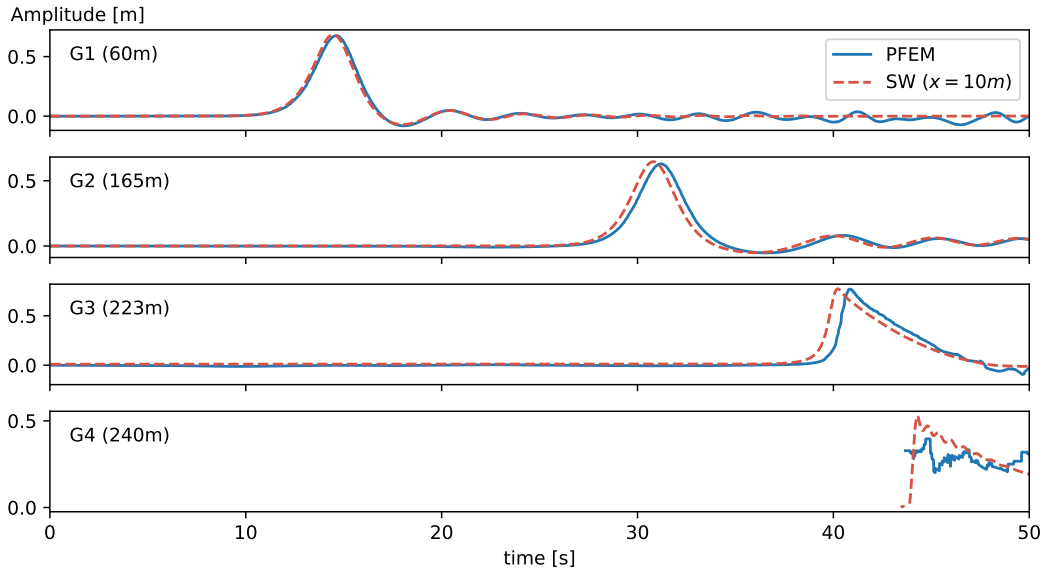


FIGURE 5.4: Solitary wave example. Time series obtained with the interface at $x = 10m$.

The results of gauges G2 and G3 show a small gap between the predicted wave by the two solvers. The Boussinesq approximation is triggering this gap, originated by an overestimation of the phase speed. This difference is consistent with wave

theory and the current wave specifications. Note that the same gap can be observed in Fig. 5.3. The run-up (G4) is out of the SW theory assumptions, but still relevant results are obtained.

The magnitude of the computational time saving of the coupled method versus the full PFEM solution is about 95%. These savings will be analyzed in more detail in the next paragraphs. The savings depend on the spatial and temporal domain chosen for the NFS, that have to be carefully designed in order not to introduce additional errors.

5.3.1.3 Sensitivity to the interface position

The FFS sensitivity has been tested with some SW interface positions at $x_1 = 10$, 20, 30 and $40m$. One would expect to obtain a more accurate response as the interface is placed further away from the paddle. Nevertheless, since in this example the wave is very regular, the observed influence of the interface position on the results is not significant. The solutions obtained with all the interfaces can be considered already converged (Table 5.3). These results were expected due to the regularity of the wave. For this reason, a similar study is also performed in Section 5.3.2, where the SW interfaces are placed into a more chaotic fluid flow.

Interface position			
$10m$	$20m$	$30m$	$40m$
2.48%	3.01%	3.12%	2.84%

TABLE 5.3: Solitary wave example. Wave amplitude errors computed at gauge 3 ($x = 170m$) for different positions of the SW interface. Reference solution: full PFEM simulation.

5.3.1.4 Sensitivity to the temporal domain

Part of the saving in computational time comes from reducing the duration of the PFEM simulation up to the minimum time needed. Once the initial impulse has generated the wave and it has been transferred to the SW domain, the PFEM computations do not provide relevant information. From that time on, the initial boundary condition, which corresponds to water at rest, is imposed at the SW domain.

This transition in the BC has to be carefully treated in order to avoid unphysical oscillations. A good duration for the transition is half of the period of the current wave.

In this test, we evaluate the effect of feeding the FFS with NFS solutions limited in time. In particular, we considered four PFEM analyses of duration 10, 20, 30, and $40s$.

Fig. 5.5 shows the time evolution of the wave amplitude at the first gauge. In the graph, we also added dots representing the time instant when one analysis starts to diverge from the rest. It is clearly observed that the four solutions have an identical behavior in the first part of the graph. In particular, even with just $10s$ of the PFEM simulation, the main wave is well reproduced. Beyond this time, the curves diverge progressively. As expected, a time interval of around 10 seconds separates the consecutive diverging points.

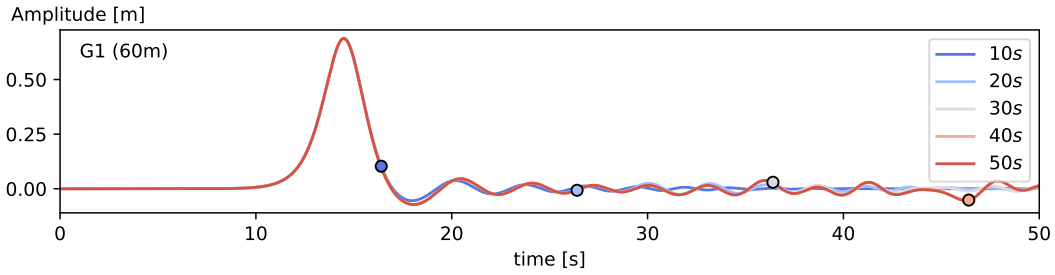


FIGURE 5.5: Solitary wave example. Set of analysis where the interface is active only in a part of the time domain. The marker shows when the solution tends to the resting condition.

5.3.1.5 Sensitivity to PFEM domain length

Besides the reduction of the time duration of the analyses, the optimization of the size of the PFEM computing domain can drastically reduce the computational cost of the simulations without affecting the accuracy of the results. For this reason, we analyze here the effect of considering partial PFEM domains of 10, 20 and 30m length plus an extension acting as absorbing boundary condition, as shown in Fig. 5.6. The study is carried out for both 2D and 3D PFEM domains. The errors introduced by

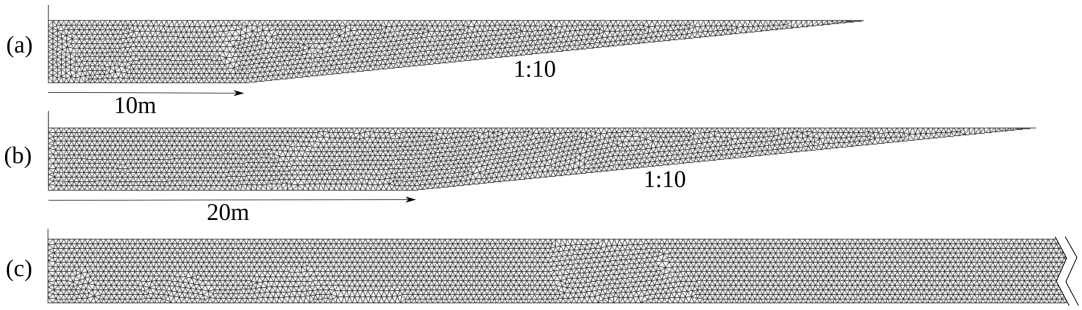


FIGURE 5.6: Solitary wave example. (a) The mesh with interface at 10m, 2 700 elements. (b) The mesh with interface at 20m, 3 800 elements. (c) Detail of the full mesh of the channel, 20 000 elements. The slope has a dissipative effect and is acting as an absorbing boundary.

the effect of shortening the PFEM domain are listed in Table 5.4.

PFEM domain length	PFEM 2D SW interface position			PFEM 3D SW interface position		
	10m	20m	30m	10m	20m	30m
30m	-0.652%	-0.984%	-6.37%	0.228%	-3.0%	-7.16%
20m	-0.635%	-5.97%	-	-0.438%	-5.62%	-
10m	-5.21%	-	-	-5.24%	-	-

TABLE 5.4: Solitary wave example. Errors of the wave amplitude computed at gauge 3 ($x = 170m$) with different configurations. Reference solution: coupled solution obtained with the full PFEM domain, as shown in Fig. 5.6c.

It is important to note that the vicinity of the absorbing boundary condition of the PFEM may affect the accuracy of the interface. The small errors obtained when the interface is far enough from the absorbing boundary show that the presented

methodology allows to effectively reduce the PFEM domain without virtually affecting the quality of the solution. This is particularly noticeable in the 2D case.

The 3D case presents a similar behavior, but higher errors are observed in the $30m$ domain length. However, these errors are more attributable to the capabilities of the NFS for reproducing the fluid-solid interaction at the lateral walls (see [143] for more details) than to the coupling strategy. A finer discretization in the PFEM mesh would reduce this bias.

5.3.2 Wave generated by a water landslide

In the second example, we simulate the experiment carried out at the Queen's University landslide flume presented in [23]. In this laboratory test, a mass of water is released from an elevated reservoir and, after flowing downhill over a 30° slope, it impacts at high velocity the water at rest placed on a $33.8m$-long channel. In the reference work [23], 41 experiments were presented covering a wide range of source volumes and reservoir depths. In [127], a comparison of experimental and numerical results obtained for three different water depths in the channel is presented. In this research, we select the largest volume case ($0.45m^3$) and water depth ($0.60m$). Fig. 5.7 shows the geometry of the experimental setup considered in this work.

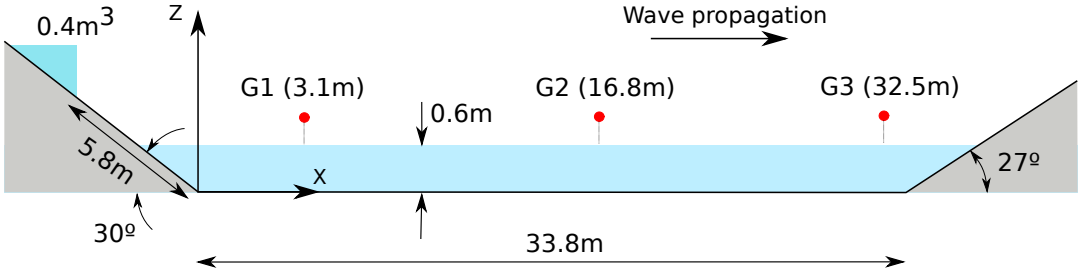
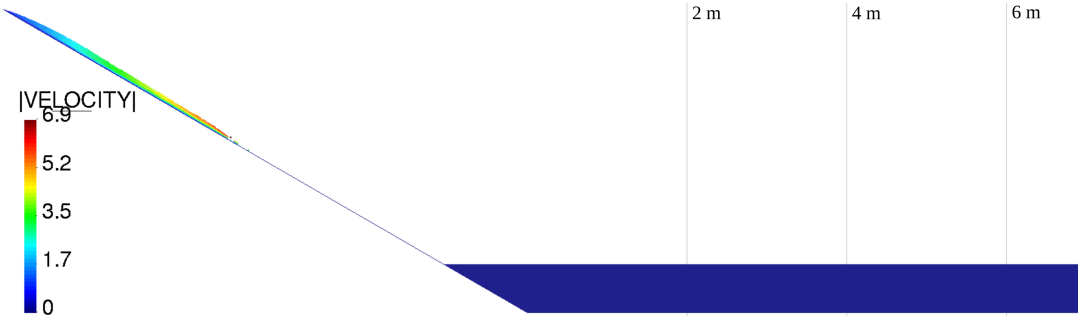


FIGURE 5.7: Landslide wave problem. Setup of the LGW flume for the experimental and numerical analyses.

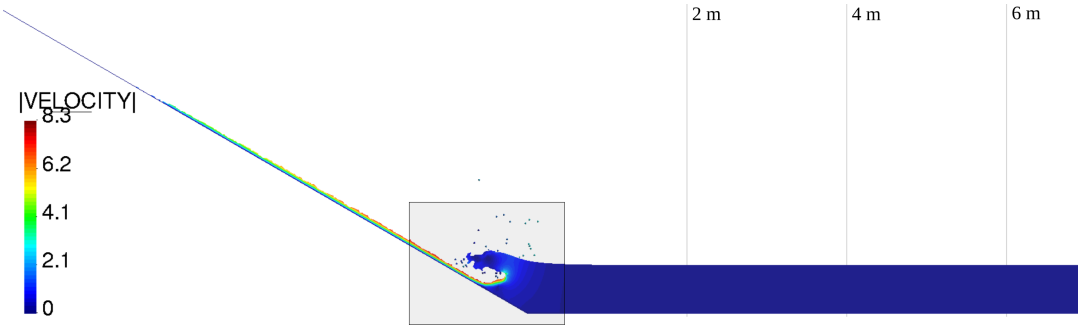
We remark that considering a water landslide does not affect the relevance of the test in the field of LGWs. In fact, the phenomena produced by the water runout and impact are totally representative of a realistic LGW scenario with a fast mobilized material. Furthermore, the use of water as sliding material removes the uncertainty related to the rheological properties of the slide and allows repeatability of the test.

The PFEM is used to simulate the water runout, the impact against the water at rest and the consequent wave formation (Fig. 5.8). Remarkably, the front of the water landslide reaches the end of the slope with a thin layer of less than $10cm$ and it impacts the water in the channel at a speed of about $10m/s$. Thus, in order to capture accurately the phenomena at the impact zone, a fine mesh and time discretizations are necessary. For this reason, a mesh size of $\Delta x = 1.5cm$ and a time step increment of $\Delta t = 5 \cdot 10^{-4}s$ are used in the PFEM simulations. On the other hand, a much coarser mesh and time discretizations can be used to model the wave propagation along the channel with the SW solver. In particular, in the FFS a time step of $\Delta t = 0.025s$ and a mesh size of $\Delta x = 0.3m$ have been used. We remark that the possibility of using much different and yet adequate space and time parameters in the FFS and NFS solvers is one of the main advantages of this partitioned method and one of the reasons for its high computational efficiency.

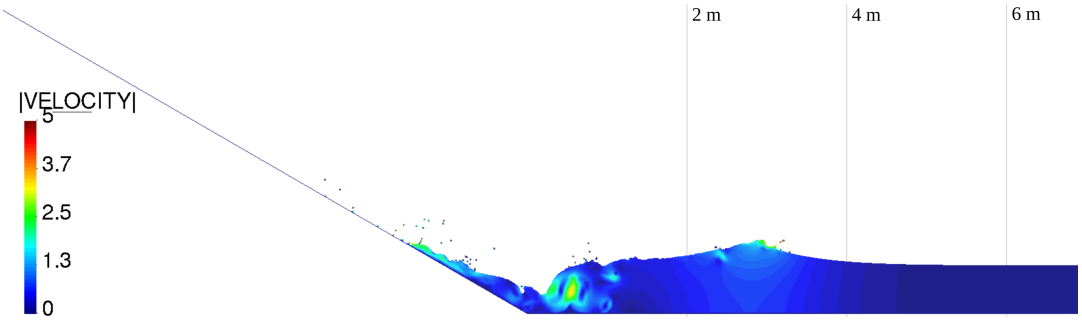
(A) Runout ($t = 0.7s$)



(B) Impact ($t = 1.5s$)



(C) Wave formation ($t = 2.8s$)



(D) Impact zone modeled with the PFEM. Detail of Figure (b) adding the solving mesh.

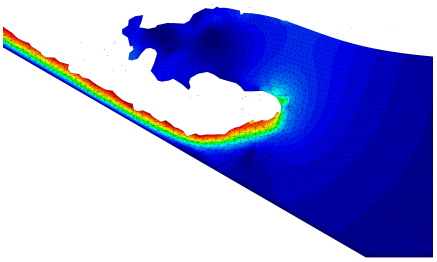


FIGURE 5.8: Landslide wave problem. Near-field results with the PFEM solution of Navier-Stokes problem. The thin vertical lines show the SW interfaces positions.

5.3.2.1 Numerical results

This LGW scenario has been solved using a time and space reduced PFEM domain in combination with three SW interfaces. The PFEM spatial domain includes the runout, the first $7m$ of the flume and an absorbing boundary condition, while the temporal domain includes only the first $5s$. The SW interfaces are positioned at 2, 4 and $6m$. Fig. 5.9 presents the results obtained at the gauges and a representation of the wave propagation.

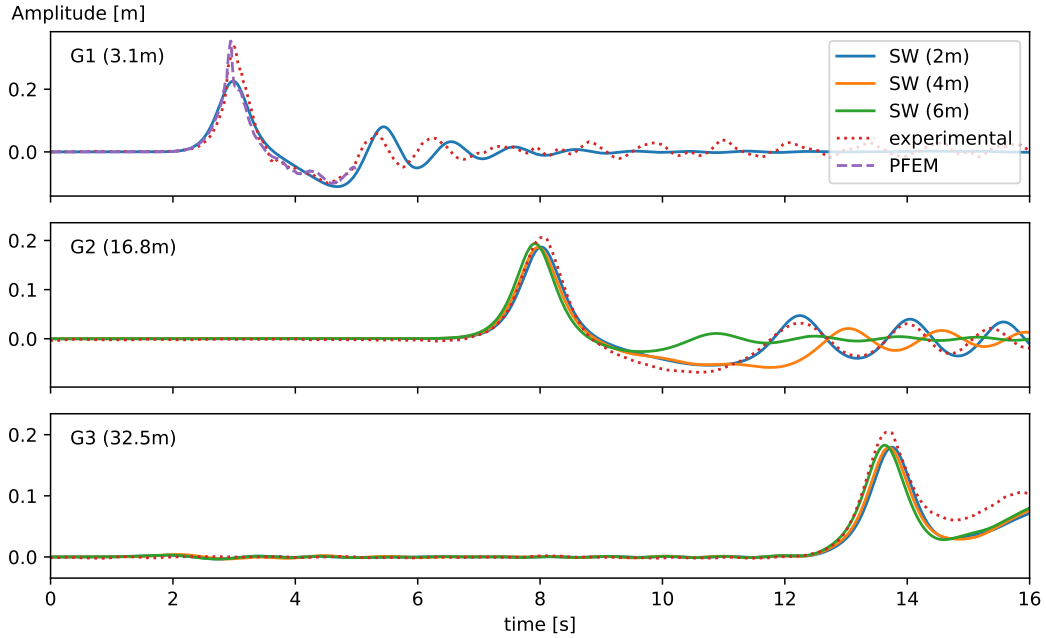


FIGURE 5.9: Landslide wave problem. Time series of the wave amplitude at the different recording points.

In the top image, it can be observed the vicinity of the first gauge and the first SW interface to the wave generation zone. Indeed, gauge G1 can only record the PFEM solution and the SW solution obtained by the first interface. It is clear that the imposed boundary condition does not satisfy the Boussinesq assumptions and the interpolated wave does not fit the profile of a breaking wave. However, although the wave interpolated by the FFS at the first stages is not equivalent in terms of wave height, the stored momentum is the correct one. This can be observed at gauges G2 and G3, where the experimental wave has adopted the solution of a solitary wave and matches the profile of the FFS.

The results obtained at gauges G2 and G3, placed at the middle and the end of the channel, respectively, show that all the three SW interface positions reproduce well the main wave obtained experimentally. This is particularly remarkable considering that the SW interface placed at $x = 2m$ is completely inside the impact zone (Fig. 5.8). These results show that, as long the momentum is well transferred from the NFS to FFS, the wave propagation process in the far-field can be accurately reproduced even considering the SW interface in a zone where the wave is not completely generated. We also remark that this can be done safely in this test, since water has been considered for the sliding material. In case of considering a different landslide material, either the interface is placed further the zone of material deposition of the landslide, or the interface boundary conditions have to take into account the presence of different materials in the computation of the overall momentum.

Gauge G2 also records a considerable time interval after the first wave, this allows us to analyze also the secondary waves. In this case, we note some discrepancies between the results obtained by three SW interface positions. In particular, the first solution that diverges from the experimental one (and from the two other numerical solutions) is that obtained by the farthest interface position ($6m$). This result is totally consistent with the time domain truncation explained in Example 5.3.1 and Fig. 5.5. As the interface position is further from the impact zone, the signal arrives later. Given the phase speed is about $2.5m/s$, the time difference between each interface is around $0.8s$.

As a concluding remark for this example, the computational cost of the full simulation of the LGW has been estimated proportionally to the time needed by the signal to arrive at the end of the channel and proportionally to the number of elements required to discretize the full domain. The resources consumed by the FFS can be neglected since they are two orders of magnitude smaller. According to these considerations, the overall time saving given by the proposed partitioned strategy is 95%.

5.3.3 Landslide in a representative alpine lake

In [63], different metrics of real alpine lakes were used to define the configuration of theoretical mountain basins of different sizes and shapes. These geometries were used in [63] to study LGW scenarios with a finite volume solver and to obtain correlations between the lake configuration and the landslide-generated waves. Here, we analyze one of the lakes considered in [63] to test the proposed coupled strategy in a 3D complex setup.

Fig. 5.10 shows the side and top views of the geometry of the lake. The case study is a circular lake with a diameter of 1500 m. The landslide has a prismatic shape of 20 m thick, 208 m long and 120 m wide. Following [63], a bulk material density of 1620 kg/m^3 is used for the landslide material and an initial velocity of 20 m/s has been prescribed to the sliding body.

Preliminary NFS analyses of the LGW scenario showed that the landslide material reaches a deposition distance of around $350m$. This information is useful to place the SW interface at a position that is not trespassed by the sliding material. For this reason, the interface of the FFS has been placed at $400m$ from the center of coordinates, which is the center of the run-out impact.

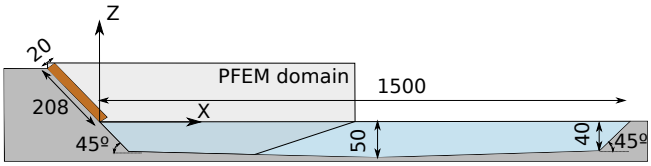
5.3.3.1 Numerical results

Fig. 5.11 shows a global view of the simulated LGW and a superposition of the NFS and FFS results.

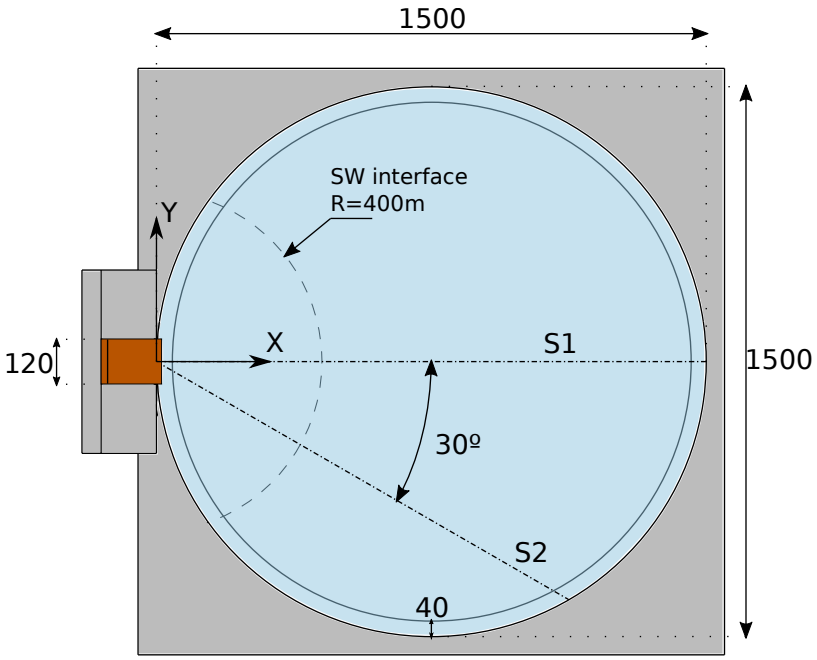
In order to assess the quality of the obtained solution, in Fig. 5.12 we compare the envelope of the maximum wave height measured along sections $S1$ and $S2$ with the reference solution given in [63].

Globally, the results obtained with the proposed method agree well with the reference numerical solution, both in the near and far fields. Although with some differences in terms of magnitude, both methods are also able to reproduce the amplification of the wave near the shoreline. This phenomenon is produced by the combined effect of shoaling and the wave reflection given by the steep bottom surface.

We also highlight that the results of the FFS are in good agreement with wave propagation theory. In an unconstrained plane, the wave amplitude is inversely



(A) Side view.



(B) Top view.

FIGURE 5.10: Landslide in a representative lake. Side and top view of the geometry. Dimensions in m.

proportional to the distance from the origin. Section S1 is closer to the unconstrained decay, while section S2 shows a smaller decay since it is closer to the boundary.

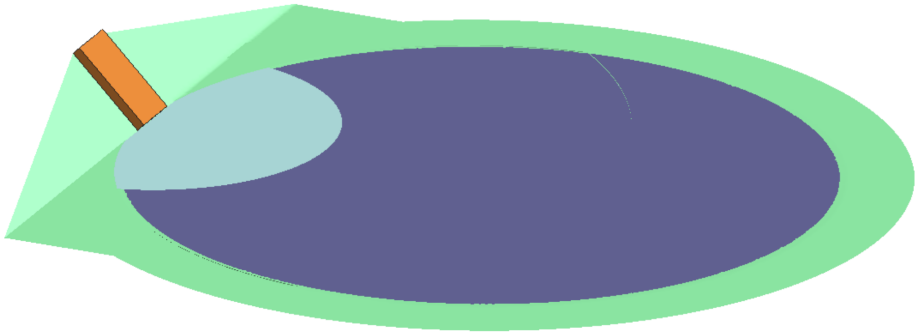
Finally, it is worth commenting on the peaks in amplitude exhibited by the FFS solution close to the SW interface. As mentioned before, the imposed signal coming from the PFEM simulation is still not fulfilling the Boussinesq theory. On the other hand, the generation of stable waves by Dirichlet boundary conditions requires some traveling distance to be modulated by the fluid system [164]. For this reason, the wave amplitude results obtained close to the SW interface with the FFS should be disregarded. We emphasize again that, on the other hand, the overall momentum computed in that zone is still correct.

In any case, the presented partitioned approach would be really interesting for an exercise like the lakes classification in [63]. Indeed, a single landslide calculated with the NFS could be used to simulate different representative lakes with the FFS. Also, in a more detailed study it would allow concentrating the computational resources in the analysis of the run-out and wave generation, thus enhancing the overall accuracy of the partitioned scheme.

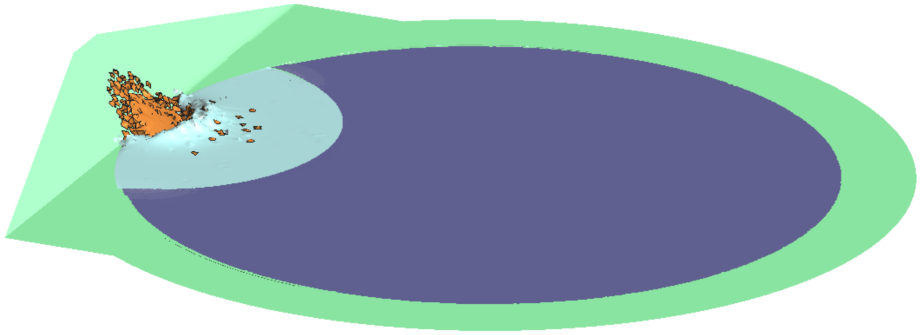
5.4 Concluding remarks

In all the examples presented, the results obtained with the new partitioned method had shown a very good agreement with the reference solutions, both in 2D and

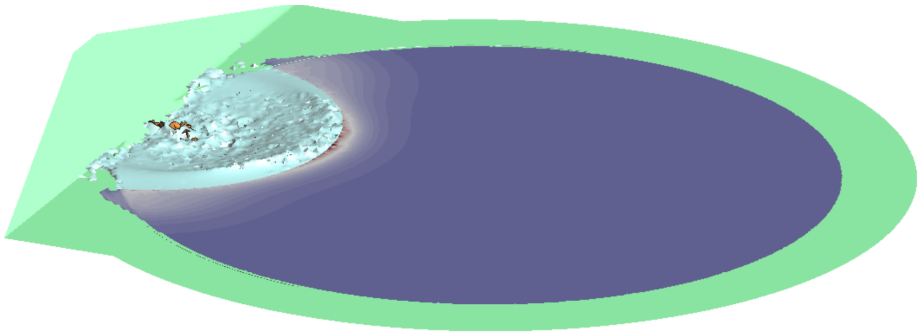
(A) Initial configuration ($t = 0s$)



(B) Run-out and impact ($t = 5s$)



(C) Wave generation ($t = 10s$)



(D) Wave propagation ($t = 50s$)

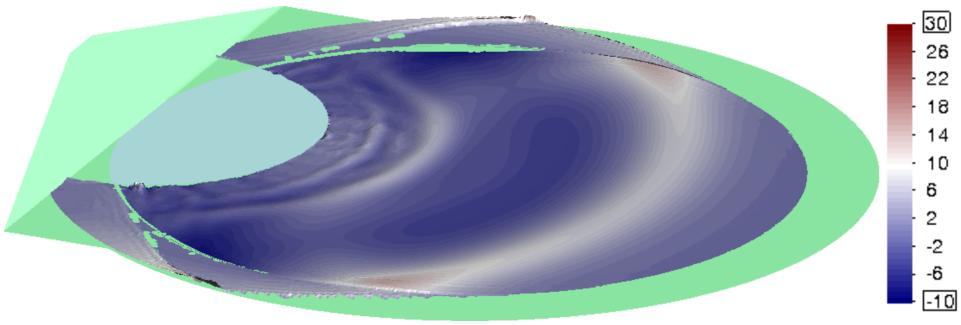


FIGURE 5.11: Landslide in a representative lake. Global representation of the LGW. The NFS domain is plotted until the SW interface and only the geometry is shown. For the FFS, results for the free surface elevation are depicted.

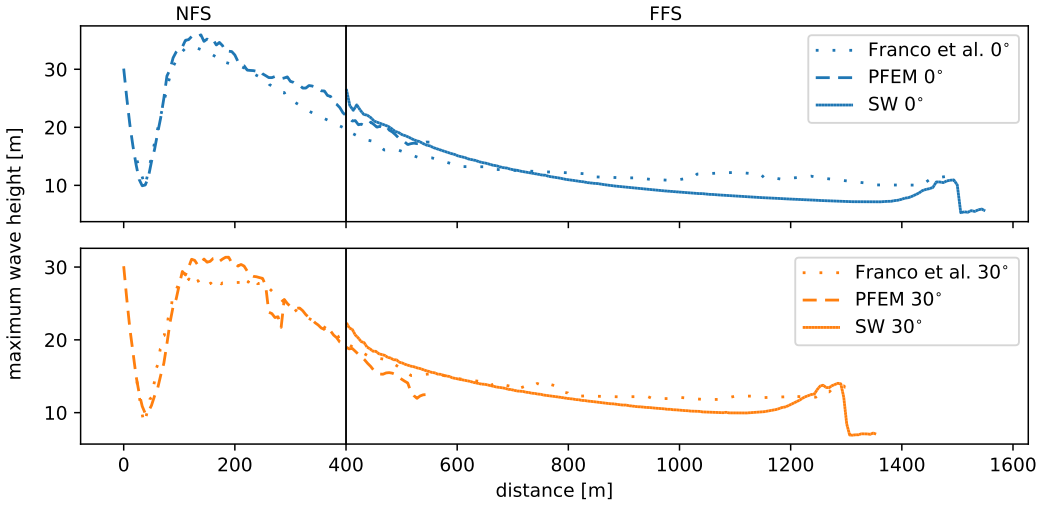


FIGURE 5.12: Landslide in a representative lake. Envelope of the free-water-surface elevation along sections S1 and S2.

3D problems. Remarkably, we have been able to compare our numerical results with analytical solutions, fully-resolved numerical simulations of LGW events, other coupled methods presented in the literature, and experimental observations.

Placing the SW interface as close as possible to the impact zone gives the major advantage of reducing the NFS domain and, consequently, the overall computational cost of the analysis. For this reason, we have compared the FFS results obtained considering different positions of the SW interface for the same NFS solution. This study showed that, as long the momentum of the NFS is well transferred to the FFS, the SW interface can be also placed very close to the impact zone, even if the wave is not already formed. More specifically, the SW interface can be placed at one wavelength from the impact zone. In fact, although locally the FFS results may give spurious amplitudes since the input wave is not fulfilling the Boussinesq theory, the stored momentum is correct and the far-field wave propagation is reproduced accurately. We remark that this can be easily done in case of having the same density between the sliding material and the water in the reservoir, such in the water landslide scenario analyzed in Section 5.3.2. In a more general case, the interface should be placed further than the deposition zone of the landslide, or the SW interface should take into account the variation of material densities on depth.

We have also verified the effect of reducing the size of the PFEM domain by using absorbing boundary conditions. For this purpose, a gentle final slope with an inclination of 1:10 was placed at the end of the PFEM domain. We showed that, as long as the SW interface is not placed too close to the absorbing boundary, the PFEM domain can be safely truncated without affecting the global results. To be precise, the gentle slope should begin at least one half wavelength after the SW interface.

Finally, we also studied the effect of reducing the time duration of the NFS analyses. We have shown that, if the main interest of the simulation of the LGW scenario is to reproduce the main wave propagation, the PFEM analysis can be safely stopped after it has modeled the impact of the landslide on the water and the first wave formation. Indeed, this time truncation of the NFS will only affect the secondary waves propagation. We also showed that, knowing the NFS duration and the wave propagation speed, it is possible to have a quite accurate estimation of the reliability of the secondary waves results.

All these specific studies will allow us the define the most computational efficient NFS-FFS scheme for practical LGW simulations. Although the overall computational cost depends inevitably on the geometry and the proportions among the near and far fields, in the examples here presented, we could estimate a 90% of time saving versus a fully-resolved simulation of the same LGW scenario.

Among the possible enhancements of the proposed method, we consider it of primary interest to investigate more efficient strategies for the NFS absorbing boundaries and to develop a reverse one-way coupled algorithm where the FFS transfers the information to the NFS. This FFS-NFS model would allow us simulating with high accuracy the effect of tsunami waves produced by landslides (or by some other source, *i.e.*, an earthquake) on the shoreline and the civil constructions placed therein.

Chapter 6

Conclusions

6.1 Achievements

The main objective of this thesis is the investigation of Finite Element formulations applied to large-scale water-related hazards. In the first part of the thesis, a revision of reduced models for the large scale has been done.

In chapter 3 The FIC-FEM procedure has been extended to the shallow water equations. Unlike the FIC-based stabilizations for incompressible flows, the present procedure is applied to the coupled mass and momentum balance at the same time using the linearization matrix $\mathbf{A}_i$. This procedure can be seen as the classical FIC-stabilization for convection diffusion problems, taking the velocity as linearization term. The same procedure can be applied to develop stabilized formulations for compressible flows.

The proposed extension of the FIC-procedure to the shallow water equations uses the linearization matrix $\mathbf{A}_i$ for the flux terms to project the characteristic length. However, an alternative framework can be explored with the ASGS [37, 82] formulation, which includes the linearization matrices of the viscous terms and reaction terms. Since the shallow water equations are dominated by the convective matrix $\mathbf{A}_i$, and thus are strictly hyperbolic, the present stabilization is enough to provide stability, as shown in Section 3.5.

The stabilization provides two algorithmic constants, one for the global stabilization and other one for the shock capturing term. From our numerical experiments, we have chosen $\beta = 0.01$ for the stabilization and $\alpha = 1.0$ for the shock capturing.

Regarding the accuracy of the shock capturing and the dry domain model, one must notice that this method is not monotonic. Therefore, like in many other stabilized methods, the order of convergence is dropped around discontinuities such as hydraulic jumps and the shoreline. However, the spurious oscillations, especially the oscillations related to the moving shoreline, are bounded and the method is globally mass preserving. The method -Residual Based- has been compared against the Flux Corrected algorithm and the Gradient Jump Viscosity. The Residual Based method has provided better performance for the global situations, steady or transient state, complex topography and reduced artificial viscosity.

The present FIC-FEM procedure has produced accurate results for the examples considered. In the first example, the artificial diffusion is evaluated and it has been proved to be small and practically inappreciable. The shock capturing term allows to solve supercritical problems with discontinuities and the present procedure is also able to deal with partially wet domains. Finally, a numerical simulation of a dam break flow against an isolated building is performed. The limitations of the model essentially come from the shallow water equations hypothesis. In fact, that last example presents local regions where the dynamic pressure is not negligible. It

is not an obstacle to simulate the main aspects of the flow and the numerical results are in good agreement with the experimental data.

In chapter 4 the presented stabilization technique has been applied to the Boussinesq equations. Special attention has been paid to higher order derivatives with linear finite elements. Additionally, a numerical approximation to open boundaries has been implemented. The Boussinesq model is able to accurately capture the dispersion effects, which are of crucial importance to analyze the propagation of impulse waves in the context of Landslide Generated Waves (LGW).

Finally, chapter 5 presented a novel partitioned strategy for solving landslide-generated wave (LGW) problems. The coupled method makes a near-field solver (NFS) interact with a far-field one (FFS). The NFS reproduces the landslide runout and the impact zone by solving the Navier Stokes equations with the Lagrangian Particle Finite Element Method (PFEM). On the other hand, the FFS uses as input the NFS results stored at a certain interface to model the wave propagation with the Eulerian Finite Element Method (FEM) based on Boussinesq equations, derived in the previous chapter. In order to substantially improve the computational performance of the method and, thus, to allow for the simulation of large-scale problems, we adopt a one-way coupling scheme, meaning that the NFS solution is insensitive to the FFS one. This partitioned method also allows us to freely decouple the time and space discretizations of the two solvers, giving a further advantage in terms of accuracy, efficiency and flexibility.

The coupling strategy has been designed in order to minimize the computational cost, keeping the same accuracy than a fully resolved model. Specifically, the influence of the position of the interface, the temporal domain of the coupling and the open boundary had been analyzed. The coupled strategy has proven to be accurate and promising results have been obtained. It has been compared against literature results and has been applied to a LGW in a representative lake.

All the presented formulations had been implemented in KratosMultiphysics [49, 50], an open source framework of numerical methods written in C++. The main contributions of this work can be summarized in the following list:

- Derivation of a stable FIC-FEM formulation for the SW equations.
- Comparison of three non-linear stabilizations for the mitigation of oscillations around shocks and the moving shoreline.
- Extension of the FIC-FEM formulation from the SW equations to the Boussinesq equations.
- Design of a partitioned strategy to couple the solution obtained from the NS equations to the SW or Boussinesq equations.
- Application of the partitioned strategy to LGW events.

6.2 Further research

The development of the thesis opens several subjects that can be tackled in the future. Regarding the reduced models for the large scale, some issues can be addressed. The appendix A show that the convergence at the shoreline can be improved. In fact, the stabilized eulerian framework has a first order of convergence, while the lagrangian framework converges at second order. Nevertheless, the eulerian framework is more robust and faster and, in global terms, it is the preferred

option. However, a selective strategy can be developed with the advantages of both frames, for example, an arbitrary lagrangian-eulerian scheme or a cutting technology for partially wet elements. Cutting the elements at the shoreline would help to mitigate the Gibbs oscillations since the real free surface does not belong to the finite element space given its discontinuous basis.

A second issue to overcome is the shock capturing for the supercritical regime before the shocks. The non-linear stabilization with the residual viscosity approach exhibits some upstream oscillations, which are related to the numerical model and thus, nonphysical. The other non-linear stabilizations are more dissipative and avoid these oscillations, however, the global accuracy is lost due to the excessive diffusion.

Regarding the coupled procedure, the main pending task is the second part of the coupling, from the shallow water equations to the Navier-Stokes equations. Some advances have been made in this regard, but are not fully mature to be included in this thesis as a chapter.

Also, the optimization of the coupling interface in chapter 5 has shown that the open boundary is extended over a large domain, thus increasing considerably the computational cost. The development of an open boundary in a Lagrangian frame using the PFEM has been little studied by the scientific community. It would help to achieve higher computational savings up to more than 95%. Preliminary analyses suggest that the imposition of an inert paddle moving with the waves can be accurate. However, the generalization to the 3D case with arbitrary directions of propagation is not straightforward.

Finally, a more scientific contribution of the coupled strategy is the extension of the shallow water model with a two way coupled solver. The aim of the two way coupled model is to extend the range of applicability of the reduced model to post breaking waves.

Appendix A

Particle Finite Element Methods for the shallow water equations

In chapter 3 the SW equations have been analyzed considering the coupled convective and oscillatory mechanisms in an Eulerian framework. However, in some regions of the domain, the solution of the equations can be convection dominated. Specifically, whenever there is a movement of the shoreline –like run-up or flooding–, the problem is convection dominated. This mechanism suggests the use of Lagrangian strategies, which have been successfully applied to convection diffusion and Navier-Stokes problems.

In this appendix, the developments of the Particle Finite Element Method (PFEM) are applied to the SW equations. The basis of the PFEM family of methods consists on a splitting operator, solving two stage fashion the convective operator and the rest of the equations. In the case of the SW equations, there is a stage for convection and other stage for wave mechanism. The main advantage of the PFEM lies in using the variational principle of FEM. Hence, the formulations presented in chapter 3 are applicable with small modifications. The novelty of the PFEM consists on solving the convection with a particle method.

In the family of PFEM there are two main groups, the moving mesh and the fixed mesh. The mesh moving algorithm was firstly presented to the scientific community [87, 88] and has been widely applied to a high number of situations [104, 140, 151]. In this method, the particles traditionally coincide with the nodes. After the convection stage –and eventually, remeshing–, the mesh inherits the displacements of the convection. Those displacements are part of the material derivative involved in the variational principle of the equations.

Lately, a second generation of the PFEM was presented [90]. It uses a fixed mesh and is known as PFEM-2. The main idea consists on decoupling the particles from the nodes, leading to a duality of spaces: the FEM discretization and the particles discretization. Its main drawback is the introduction of projections between the two spaces but the cost of the projection is expected to be compensated by the use of a fixed mesh strategy without remeshing [91, 147].

A.1 Introduction

The Lagrangian formulation starts by the definition of the material derivative. Let φ be a scalar or vector property, the material derivative is obtained by applying the

chain rule:

$$\frac{D}{Dt}\varphi(\mathbf{x},t) = \frac{\partial\varphi}{\partial t} + u\cdot\nabla\varphi \tag{A.1a}$$

$$\frac{\partial\mathbf{x}}{\partial t} = \mathbf{u} \tag{A.1b}$$

The Lagrangian procedure consist on solving separately the equations (A.1a) and (A.1b). To apply this staggered procedure, a generic conservation balance is considered,

$$\frac{\partial\varphi}{\partial t} + \nabla\mathbf{F} = 0 \tag{A.2}$$

where $\mathbf{F}$ is the fluxes vector in an infinitesimal volume of control. The balance equation is split into the convective and non convective fluxes, $\mathcal{L}_1$ and $\mathcal{L}_2$ respectively.

$$\frac{\partial\varphi}{\partial t} + \mathcal{L}_1\varphi + \mathcal{L}_2\varphi = 0 \tag{A.3}$$

After introducing equation (A.1) into (A.3), the following expression is obtained,

$$\frac{D\varphi}{Dt} + \mathcal{L}_2\varphi = 0 \tag{A.4}$$

Finally, the numerical strategy for solving a balance equation (A.2) in a Lagrangian framework consists on applying a splitting, generally, the first order Godunov splitting [105] or the second order Strang splitting [116]. The order of accuracy of the splitting operator is related to the sequence and the mode how equations (A.1b) and (A.4) are temporally integrated.

A.2 Mesh moving methods

The PFEM method has been widely applied to solve the incompressible Navier-Stokes equations, especially with free surface problems [104], multi-fluids [124] and fluid-structure interaction [140]. The moving mesh allows to fit the sub-domains with the discretization and the boundaries are tracked in a natural way. When the PFEM is applied to the SW equations, the mesh moving will be solving the water domain in the horizontal plane. That is, the moving shoreline in the SW equations is playing the role of the free surface in the NS equations.

Once the discretization describes and follows the fluid motion –η and $\mathbf{u}$–, there appears the need to introduce another discretization to define the fixed topography and its variations –z–, since it does not move with the fluid. The duality of discretizations introduce the need of a mapping between the two meshes. The topography data need to be mapped to the computational mesh for solving the equations. And the primal variables need to be mapped to the topographical mesh for visualization purpose. See figure A.1, where the fluid domain Ω_w si inside the computational domain Ω. The topographic domain Ω_T coincides with the computational domain Ω. This approach presents some analogies with embedded formulations, except on the fact that the discontinuity is on the fluid –computational domain– instead of the topography –geometric domain–.

Going back to the definition of the SW flow, the set of particles moving in the Lagrangian frame, convect the intrinsic properties (density, water depth, velocity, flow rate, etc.). The equations follow an *updated lagrangian* formulation, that is, the variables are assumed to be known at time t but unknown for the time $t + \Delta t$. Given

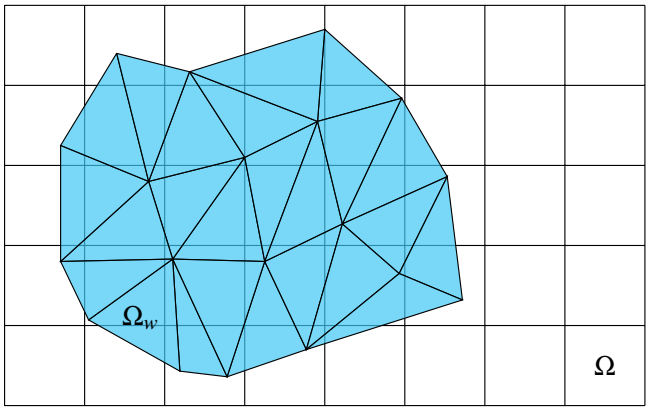


FIGURE A.1: Spatial discretization for the mesh moving PFEM algorithm.

that the variational principle is used to solve the equations in the continuum media, those is identified with the fluid domain, excluding the dry part. Once the convection is solved and the shoreline is updated, the system of equations is solved again.

Some modifications to this procedure may be considered. For example, an excessive deformation of the elements, without inverting them, may require a remeshing step. In that case, the identification of the shoreline coincides with the previous time step, but the interior domain will be refined. With a proper mesh quality control, the remeshing step can be applied only to those steps which really need it.

Additionally, an iterative scheme can be introduced, since the solution of the nwe system of equations at time step $t + \Delta t$ modifies the integration of the convective term. Usually this outer iteration loop is skipped by an explicit approximation of the convective term. The full path of the solution procedure is resumed in figure A.2.

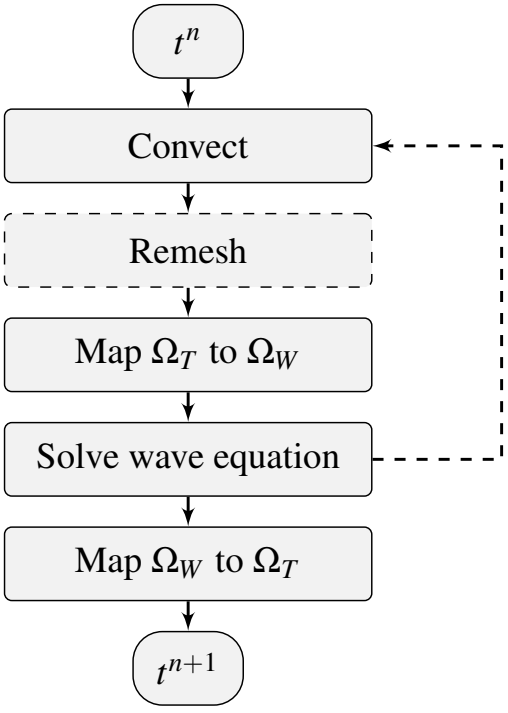


FIGURE A.2: PFEM algorithm for the SW equations.

A.2.1 Governing equations

Given that the nodes of the SW discretization are particles, the intrinsic properties are in a Lagrangian frame, thus, both mass and momentum balances follow a Lagrangian formulation. This fact reduces the non-linearity of the system of equations to be solved, making them easier to solve. On the other hand, in the Lagrangian framework the convection can be evaluated in terms of the primitive variables. We will take advantage of this characteristic to solve the primitive SW, which are more linear than the conservative ones. The accuracy of the solutions obtained with the proposed method will be compared against the methods presented in the previous chapter.

Let the conservative vector of unknowns $\boldsymbol{\phi}$ be expressed in terms of the primitive set of variables $\boldsymbol{\psi}$. we recover the SW equations in terms of the primitive variables:

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{u} \cdot \nabla \mathbf{u} + g\nabla(h - z_b) + g\mathbf{S}_f = \mathbf{0} \tag{A.5a}$$

$$\frac{\partial h}{\partial t} = \mathbf{u} \cdot \nabla h + h\nabla \mathbf{u} = 0 \tag{A.5b}$$

The introduction of the material derivative definition yields

$$\frac{D\mathbf{u}}{Dt} = g\nabla(h - z_b) + g\mathbf{S}_f = \mathbf{0} \tag{A.6a}$$

$$\frac{Dh}{Dt} = h\nabla \mathbf{u} = 0 \tag{A.6b}$$

Analogously to the previous sections, if the solution ψ is sufficiently smooth, it will also verify the quasi-linear form

$$\frac{D\boldsymbol{\psi}}{Dt} + \mathbf{A}_i \frac{\partial \boldsymbol{\psi}}{\partial x_i} + \mathbf{S}\boldsymbol{\psi} + \mathbf{T} = 0 \tag{A.7}$$

Where the matrix $\mathbf{S}$ and the vector $\mathbf{T}$ are the same than those defined in section 2. The tangent matrices $\mathbf{A}_i$ have a new expression for this case, according to the change of variables and to the material derivative. The null diagonal corresponds to the non-convective look of the equations.

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0 & g \\ 0 & 0 & 0 \\ h & 0 & 0 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & g \\ 0 & h & 0 \end{bmatrix} \tag{A.8}$$

The eigenvalues of $\mathbf{A}_i$ for the one-dimensional case are $\lambda = \pm c$, being $c = \sqrt{gh}$ the surface waves speed. These eigenvalues differ from $u + c$ and $u - c$ since we are in a Lagrangian frame, moving at velocity u. For the two-dimensional case, the eigenvalues for each direction are obtained projecting $\mathbf{A}_i$ onto a unit vector and are always $-c$, 0 and c. The system is still hyperbolic and the positivity of h is required.

A.2.2 Variational principle

Despite the non-convective look of (A.6), they need stabilization because of the incompatibility of the interpolation [37]. The fic-based stabilization method from [141] and from the previous sections will be reused. As stated before, this formulation presents a significative simplification with respect to the conservative equations in

an Eulerian frame. The residual of the equations is defined as

$$\mathbf{r} := \frac{D\boldsymbol{\psi}}{Dt} + \mathbf{A}_i \frac{\partial \boldsymbol{\psi}}{\partial x_i} + \mathbf{S}\boldsymbol{\psi} + \mathbf{T} = 0 \quad i = 1,2 \tag{A.9}$$

The number of dimensions is $n_d = 2$ and the number of balance equations is $n_b = 3$. The FIC-balance using a first order expansion of Taylor series has the following expression:

$$r_j - \frac{1}{2} l^e \frac{\mathbf{A}_i}{\lambda_{max}} \frac{\partial r_j}{\partial x_i} = 0 \qquad i \in \{1, n_d\}\ ,\ j \in \{1, n_b\} \tag{A.10}$$

To introduce stability in the desired direction, the characteristic element length l^e has been projected onto the normalized characteristics of the equation, $\mathbf{A}_i / \lambda_{max}$. In practice, the term $\frac{1}{2}$ is replaced by an algorithmic constant β in order to control the amount of extra diffusion. This parameter will be analyzed latter.

The resulting SW FIC-based stabilization is obtained from (A.7) and (A.10). The variational principle is obtained by multiplying the FIC balance by a test function ω_k and integrating over the domain Ω_w.

$$\int_{\Omega_w} \left(\omega_k \mathbf{r} + \omega_k \beta l^e \frac{\mathbf{A}_i}{\lambda} \frac{\partial \mathbf{r}}{\partial x_i} \right) d\Omega = 0 \tag{A.11}$$

The second term of Equation (A.11) is integrated by parts. Note that the geometries are linear and the element length l^e and the linearization matrix $\mathbf{A}_i$ are defined constant inside the element. Hence, the boundary integral which appears after integration by parts should be understood as the boundary of all the elements

$$\int_{\Omega_w} \omega_k \mathbf{r} d\Omega - \int_{\Omega_w} \beta l^e \frac{\mathbf{A}_i}{\lambda} \frac{\partial \omega_k}{\partial x_i} \mathbf{r} d\Omega + \sum_e \int_{\Gamma_e} \beta l^e \frac{\mathbf{A}_i}{\lambda} \omega_k n_k \mathbf{r} d\Gamma = 0 \tag{A.12}$$

In this work we neglect the boundary integrals assuming that the residual $\mathbf{r}$ is null at the boundary of the elements. At this point we introduce the balance Equation (A.6) and integrate by parts again. The result is

$$\begin{aligned} \int_{\Omega_w} \left(\omega_k \frac{\partial \boldsymbol{\psi}}{\partial t} + \omega_k \mathbf{A}_i \frac{\partial \boldsymbol{\psi}}{\partial x_i} + \frac{\partial \omega_k}{\partial x_j} \mathbf{K}_{jk} \frac{\partial \boldsymbol{\psi}}{\partial x_i} + \mathbf{S}\boldsymbol{\psi} + \mathbf{F} \right) d\Omega \\ - \int_{\Omega_w} \frac{\beta l^e}{\lambda} \left(\frac{\partial \omega_k}{\partial x_j} \mathbf{A}_j \frac{\partial \boldsymbol{\psi}}{\partial t} + \frac{\partial \omega_k}{\partial x_j} \mathbf{A}_j \mathbf{A}_i \frac{\partial \boldsymbol{\psi}}{\partial x_i} + \frac{\partial^2 \omega_k}{\partial x_j^2} \mathbf{A}_j \mathbf{K}_{jk} \frac{\partial \boldsymbol{\psi}}{\partial x_i} \right. \\ \left. + \frac{\partial \omega_k}{\partial x_j} \mathbf{A}_j (\mathbf{S}\boldsymbol{\psi} + \mathbf{F}) \right) d\Omega = 0 \end{aligned} \tag{A.13}$$

Equation (A.13) is the stabilized variational form for the shallow water equations, similar to the expression obtained by SUPG. Note that the parameter $\beta l^e / \lambda$ is analogous to the characteristic time τ of the classical SUPG or GLS techniques [43].

A.2.3 Convective operator

Apart from solving Eq. (A.6), Eq. (A.1a) needs to be integrated in time. Given that (A.1a) does not involve a spatial derivatives, there is no need to use a variational principle and can be evaluated nodally. In other words, the trajectory of the particles is decoupled from the wave.

In order to evaluate the trajectory of the particles, Eq. (A.1a) is rewritten in integral form

$$\mathbf{x}(t^{n+1}) = \mathbf{x}(t^n) + \int_{t^n}^{t^{n+1}} \mathbf{u}(t)dt \tag{A.14a}$$

$$\mathbf{u}(t^{n+1}) = \mathbf{u}(t^n) + \int_{t^n}^{t^{n+1}} \mathbf{a}(t)dt \tag{A.14b}$$

As far as the time variable is evaluated at discrete intervals $t = t^1, \dots t^n$, the continuous value of t is not known. Therefore, a generic finite difference can be used to evaluate the position continuously in time:

$$\mathbf{x}^{n+1} = \mathbf{x}^n + (1-\theta)(\Delta t \mathbf{u}^n + \frac{1}{2}\Delta t^2 \mathbf{a}^n) + \theta(\Delta t \mathbf{u}^{n+1} + \frac{1}{2}\Delta t^2 \mathbf{a}^{n+1}) \tag{A.15}$$

The fact of using the velocity interpolated is related to the coupled nature of equations (A.1a) y (A.6). The choose of an implicit or explicit finite differencing depends on the splitting and the iterative scheme.

A.2.4 Limitations of the method

The main advantage of this method consists on the possibility of solving convective problems with the primitive equations, specially when there is a moving shoreline. However, the major limitation is faced on the inner discontinuities, since the semi-implicit convection operator may invert elements. This problem is not solved with remeshing because the nodal values will resort on a wrong interpolation of the variables. Figure A.3 shows a graphical representation of the strong conditions $CFL < 1$ in order to prevent the inversion of elements.

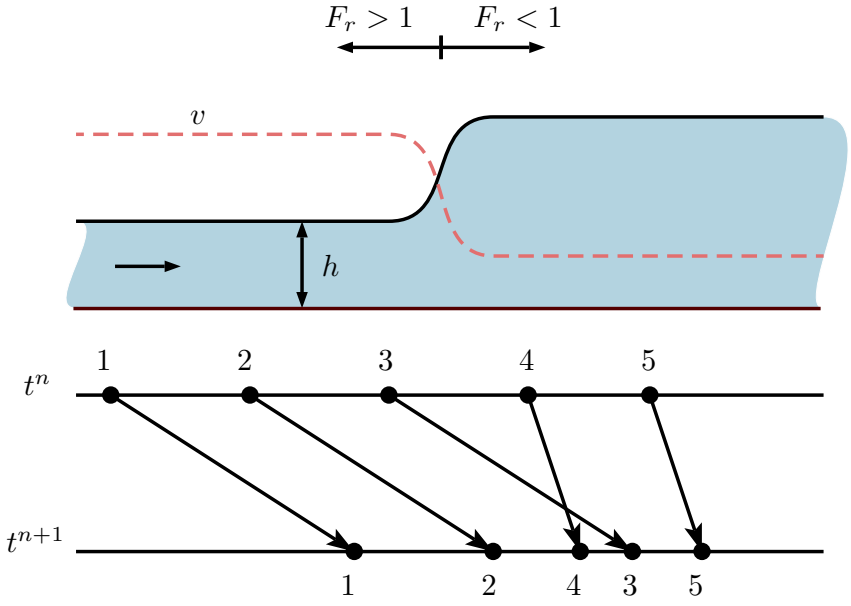


FIGURE A.3: Incompatibility of the mesh moving algorithm for solving problems with shocks.

While the primitive variables are not optimal to solve shocks, the mesh moving algorithm presents another difficulty. Both methods present restrictions, but the nature of that difficulties are different. The restriction of the primitive variables is

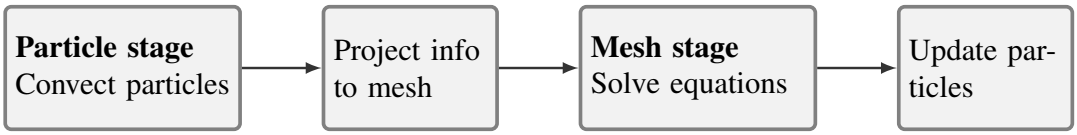


FIGURE A.4: PFEM-2 flowchart

related to the conservation of the momentum at discrete level. The restriction of the mesh moving algorithm is associated to the finite differencing convective operator.

A.3 Fixed mesh methods

In this section a Lagrangian procedure with fixed mesh is presented. This procedure is an extension of PFEM-2 [90] to the SW equations. In contrast to the mesh moving algorithm, the particles does not coincide with the nodes, but they move over the FE mesh. This procedure has the advantage of avoiding the remeshing step, but the cost of a projection from the particles to the mesh.

Let $\mathcal{P}$ be a set of particles randomly seed over the SW domain. The density of the particles is such that the average number of particles per element is greater than one. This particles move over the domain transporting the intrinsic variables of the fluid, namely density, water depth, flow rate, etc. This convection stage is finalized with a projection of the variables of the particles to the FE mesh. The solution of the step is completed with the FE counterpart, solved in the fixed mesh following a Lagrangian framework. Finally, updating the particles is a trivial operation since the FE interpolation allows to evaluate the unknowns at an arbitrary point, namely, the particles. Figure A.4 summarizes the main stages of the PFEM-2 algorithm.

In the same way as the mesh moving algorithm, this scheme may undergo certain variations. Strictly speaking, the mass and momentum balance are a coupled system of equations and the Lagrangian split does not modify that property. This will be analyzed in the following sections.

A.3.1 Governing equations

In contrast to the moving mesh algorithm, the nodes receive information from the particles without experimenting mesh displacement. This fact allows to arbitrarily choose a Lagrangian or Eulerian framework, even for only one of the balance equations. This property of the PFEM-2 algorithm allows more flexibility. The choose of the equations as well as the framework can optimize the computational cost and the approximation of the physical properties. For example, Heniche pointed out in [78] that the residual of the momentum balance does not distinguish the sign of the water depth if conservative variables are employed, $\mathbf{r_q}(h) = \mathbf{r_q}(-h)$. This property suggests the use of conservative variables in combination with a mixed framework.

$$\frac{D\mathbf{q}}{Dt} + \mathbf{q}\nabla\cdot\mathbf{u} + gh\nabla(h-z) + S_f = \mathbf{0} \tag{A.16a}$$

$$\frac{\partial h}{\partial t} + \nabla\cdot\mathbf{q} = 0 \tag{A.16b}$$

The first term involving spatial derivatives in the momentum balance comes from applying the chain rule to the fluxes vector and subtracting the convective term. The remaining terms corresponds to the compressibility of the SW equations

–it is important to remember the analogy between the SW equations and the compressible NS equations–.

Several possibilities are presented to deal with this term. The first one consists on recovering the Eulerian quasi-linear form and then, apply the material derivative. The second one imply a projection to compute the divergence of the velocity. Lastly, an updated Lagrangian formulation can be used to compute the divergence term.

The first option is preferred since it is more consistent with the previous sections and does not introduce extra steps. The linearization matrices follow the next expression:

$$\mathbf{A}_1 = \begin{bmatrix} u_1 & 0 & -u_1^2 + gh \\ u_2 & 0 & -u_1 u_2 \\ 1 & 0 & 0 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} 0 & u_1 & -u_1 u_2 \\ 0 & u_2 & -u_2^2 + gh \\ 0 & 1 & 0 \end{bmatrix} \tag{A.17}$$

A.3.2 PFEM-2 algorithm

The outlined algorithm is explained in more detail in the section. The more important parts are the computations carried out by each discrete space. The communication between them and the integration in time are strictly related.

A.3.2.1 Convection

Let us assume that the particles move as material points and that each one stores the point concentration of the property $\phi_p = \phi(\mathbf{x}_p)$. Since the variables are not known for any arbitrary time t, but only for the discrete time steps $1,\ 2 \dots n,\ n+1 \dots$, the advection of a particle can be approximated using a θ-family discretization as:

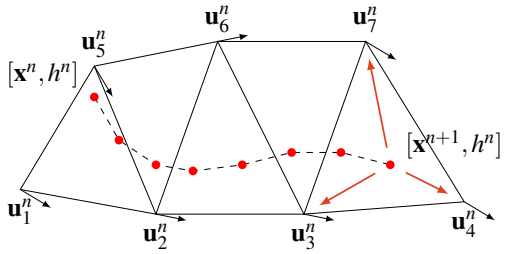
$$\mathbf{x}_p^{n+1} = \mathbf{x}_n^p + (1-\theta) \int_{t_n}^{t_{n+1}} \mathbf{v}_n(\mathbf{x}_p^t) dt + \theta \int_{t_n}^{t_{n+1}} \mathbf{v}_{n+1}(\mathbf{x}_p^t) dt \tag{A.18}$$

If the velocity field is known, the system becomes explicit and the problem is reduced to moving the particles along the streamlines. The problem is solved using an explicit forward integration ($\theta = 0$) with a proper sub-step [90]. An illustration is given in figure A.5a. This method, also known as XIVAS [86][89], was initially applied to a variable velocity field. After the particles are moved, the ones that leave the domain are removed.

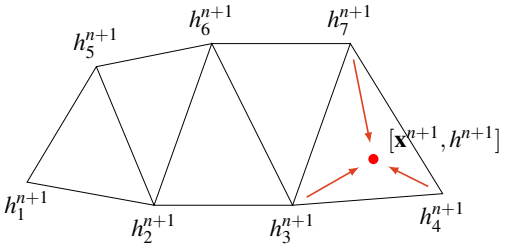
In this work the computational domain is initially seeded with fifteen particles per element. This number does not remain constant during the simulation because particles can freely enter and exit finite elements as they move through the domain. Thereby, in order to properly perform the advection stage, every time step the domain is reseed with particles so as to ensure that a minimum number is present within each finite element. This number of particles was chosen to be four. Particles can also be eliminated from each finite element in order to limit the computational cost. In this case, the maximum number of particles allowed per finite element is sixteen. These particle thresholds were chosen as they have proven to give accurate results in previous works [91].

A.3.2.2 Projection

When solving the advective stage in Equation (A.16), the particles concentration at $\mathbf{x}_p^{n+1}$ is the same as at the onset of the time step ($\mathbf{x}_p^n$). This is equivalent to saying that the advective step assumes $\frac{D\phi}{Dt} = 0$. This modification in the field described



(A) Explicit advection stage using particles (in red). Next, the particles' information is transferred to the mesh nodes.



(B) After the computation of the wave stage, the FE nodes contribute to the particles.

FIGURE A.5: Illustration of the two main steps of the PFEM2 framework.

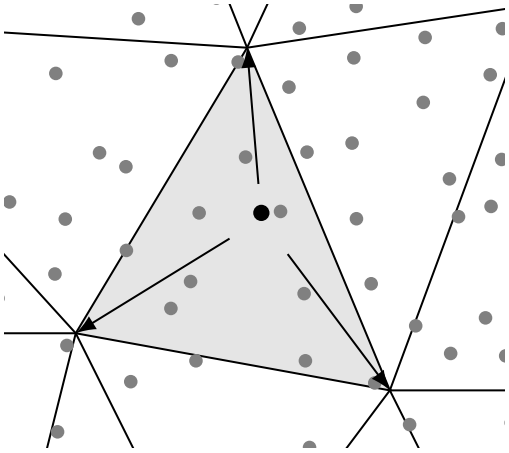
by the particles needs to be transferred onto the finite element space. As usual in particle-based techniques, such as PFEM, a projection procedure is used to transfer the information from the particles to the finite elements in the underlying mesh. In our work we use

$$\phi^* = \mathcal{L}(\phi_p) \tag{A.19}$$

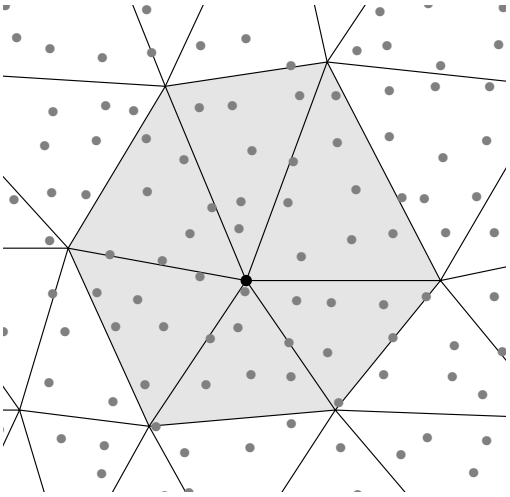
where $\mathcal{L}$ is the projection operator from the particles to the finite element space and ϕ^* is the result of the advection at the time step $n+1$. In this case, a first order explicit projection has been used and all the particles in the elements surrounding a node contribute to that node, i.e.

$$\phi_i^* = \frac{\Sigma_e \Sigma_{p_e} w_p \phi_p}{\Sigma_e \Sigma_{p_e} w_p} \quad \text{with} \quad w_p = N_{ei}(\mathbf{x}_p) \tag{A.20}$$

where the index i runs over all the mesh nodes, where e runs over the elements sharing node i and where p_e runs over the particles contained in element e.



(A) Each particle contribute to the nodes from its element.



(B) Kernel of elements and particles contributing to a node.

FIGURE A.6: Illustration of the projection stage of the PFEM2 framework.

A.3.2.3 Wave equation stage

Once the convective problem is solved explicitly in the particles and the results are transferred to the mesh nodes, the lagrangian residual (Equation (A.16)) is solved in a fixed mesh with an Eulerian FIC-FEM technique. The spatial discretization and the time integration scheme follows the procedure explained in Section 3.2. However, the time integration scheme follows a first-order BDF or Backward Euler. The time derivatives are computed according to the next expressions

$$\dot{\mathbf{q}} = \frac{\mathbf{q}^{n+1} - \mathbf{q}^*}{\Delta t} \tag{A.21}$$

$$\dot{h} = \frac{h^{n+1} - h^n}{\Delta t} \tag{A.22}$$

Equation (A.21) includes the contribution of the advection computed with the particles through the variables *. It is important to note that the water depth is still using a partial derivative.

Another minor modification with respect to the Eulerian framework, is the use of the convected values as initial guess for the iterative strategy:

$$\mathbf{q}^{n+1,0} = \mathbf{q}^* \tag{A.23}$$

$$h^{n+1,0} = h^* \tag{A.24}$$

Finally, it is worth to remark the need of stabilization, since the discretization does not fulfill the compatibility of interpolation. Though, several authors report instabilities but does not associate them to the *inf-sup* condition. The stability is introduced following the procedure explained in section 3.2.

A.3.2.4 Particles update

The last step of the PFEM2 algorithm is to add the contribution of the solution of Equation (A.16) to the particles. To avoid the accumulation of projection errors and additional diffusion, the information from the particles is updated using an incremental scheme. This step only involves the evaluation of the unknown at each particle position in the finite element mesh as:

$$\phi_p^{n+1} = \phi_p^n + \phi(\mathbf{x}_p^{n+1}) - \phi(\mathbf{x}_p^*) \tag{A.25}$$

A.4 Examples

In this section an example is presented as a proof of concept of the Lagrangian methods. Its advantages and drawbacks compared to the Eulerian framework are discussed. Here, the example presented in section 3.5.1 is taken up again, since the oscillation benchmark is a good option to test the accuracy of the shoreline tracking.

As a reminder, this benchmark has been extracted from [53] and has an analytical solution. The problem consists on a 1D parabolic basin containing a mass of water. The initial condition is zero velocity and the free surface describing an inclined plane. After the initial time $t = 0$, the mass of water begins to slice over the parabolic basin and describes an oscillatory movement. The topography is described in equation (3.54) and the analytical solution follows equation (3.55). The

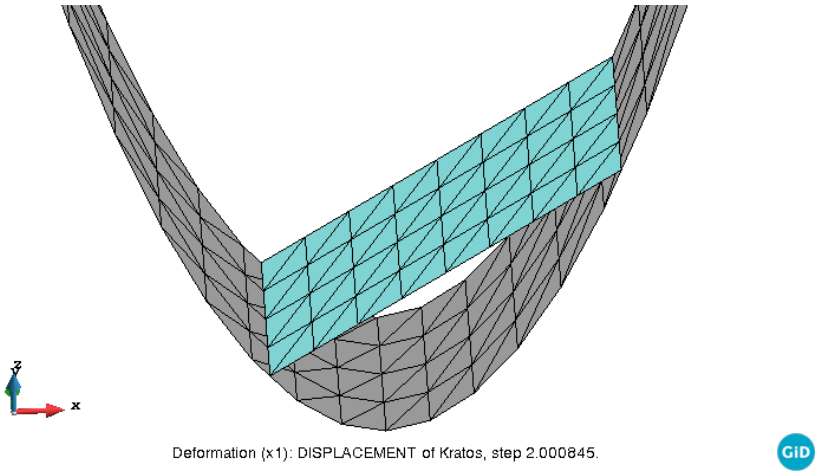


FIGURE A.7: Detail of the discretizations used to solve the oscillation in a parabolic basin with the mesh moving algorithm (PFEM).

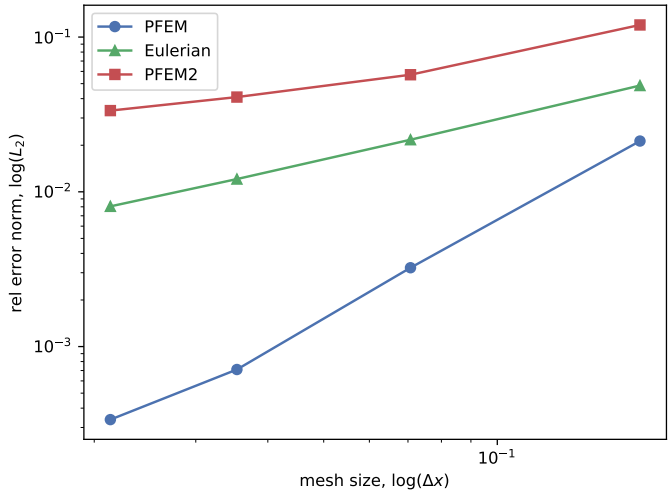


FIGURE A.8: Comparison of the convergence for the presented methods.

same parameters than the ones in section 3.5.1 are used in this example:

$$h_0 = 1 \quad , \quad a = 1$$

The spatial domain is $\Omega = [0, 10] \times [0, 1] m^2$, where the parabola is aligned with the larges dimension. Figure A.7 shows the discretization of the problem using the PFEM algorithm. For visualization purpose, the meshes are deformed vertically with the topography and the free surface respectively.

A convergence analysis has been carried out with the proposed method and the results are summarized in Figure A.8. As expected, the order of convergence of the PFEM method is not affected by the moving shoreline, keeping the second order of the FE formulation.

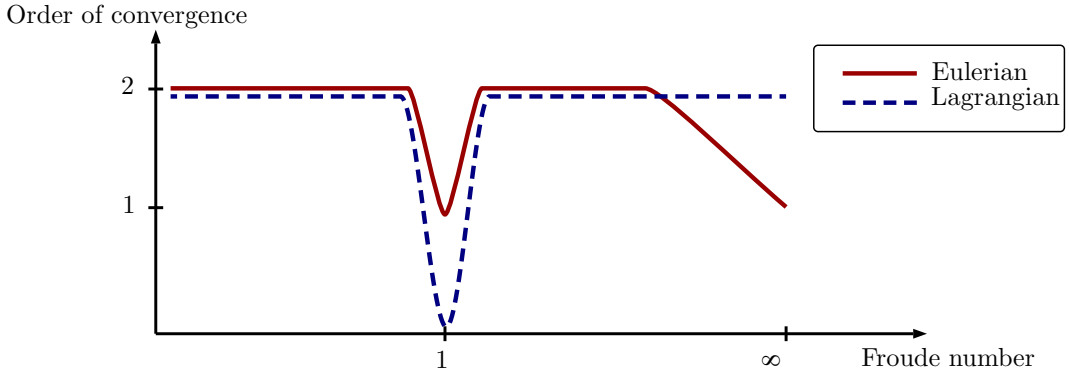


FIGURE A.9: Qualitative representation of the order of convergence in terms of the Froude number.

A.5 Concluding remarks

Lagrangian methods exhibit some computational benefits but the main advantage is its natural tracking of the shoreline. Especially the mesh moving algorithm, does not need additional treatment since it is not affected by the dry domain. The main drawback of the Lagrangian methods is related to the compressible behavior of the SW equations. On the on side, the velocity field is discontinuous around shocks and the staggered character of the time strategy is not suited to this phenomena. Furthermore, the explicit calculation of the convection faces a strong restriction on the CFL number. On the other side, the Lagrangian formulation does not achieve the desired linearization when compressible formulations are considered. Figure A.9 shows a schematic of the order of convergence in terms of the Froude number. When the Froude number is equal to 1, there is a discontinuity. The moving shoreline is characterized by a very elevated Froude number, since the shoreline involves a quasi zero water depth.

A global overview of the Lagrangian and Eulerian methods identifies the Eulerian framework as the robust one. Nevertheless, the Lagrangian framework is still interesting for the shoreline tracking. A promising combination would be an arbitrary Lagrangian-Eulerian formulation using the Eulerian framework where shocks and oscillatory behavior is dominant, and using a Lagrangian framework where the shoreline is moving.

Appendix B

Hierarchical mesh refinement with local time step

Keeping the time step under a certain value relative to the element size in transitory problems is of key importance to achieve good results. Otherwise, an excessively large time step could resort on an over-diffusive solution. Furthermore, the element size is governed by the physics, it has to be small enough to capture the modes of interest. It is a usual practice to refine the mesh near the region of interest or where the solution is changing rapidly. consequently, the local reduction of the mesh size is imposing a global reduction of the time step.

This section seeks for a strategy with *Local Time Step* (LTS). The main idea consists on adding subdomains characterized by smaller mesh sizes. Thus, a hierarchic mesh refinement is defined with a characteristic mesh size and its corresponding time step. The hierarchical refinement allows to use both non-conforming discretization at space and time level. The only requirement is having a natural number of divisions in order to perform a communication at the coarse level. Figure B.1 shows a hierarchical spatial refinement.

This framework eases the refinement and coarsening procedures. The coarsening process is specially simple, since it consists just on removing elements from the lowest level without having to rebuild the connectivities. On the other hand, a procedure must be defined for the hanging nodes at the boundary and the hanging time steps.

There have been previous advances in LTS. An early proposal can be found in [32], where a LTS was proposed for waves propagation using the Maxwell's equations. The main interest of the LTS is the reduction of computational resources and to avoid the numerical diffusion caused by small time steps on coarse regions of the mesh. The cost of introducing a LTS with local mesh refinement is an instability at

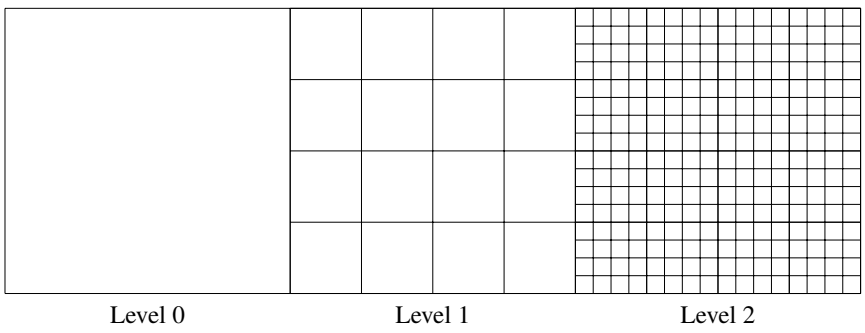


FIGURE B.1: Two refinement levels. Each refinement has two levels of sub divisions.

the coarse-fine interface. Collino analyzed the LTS for the hyperbolic 1D equations in [41, 42] and overcame this instability analyzing the conservation of the discrete energy through refinement levels. Usually the LTS has been linked to explicit time steps.

DG have been successfully applied to overcome the stability constraint of explicit LTS. For example, in [54] the non-conforming properties of DG are exploited to ensure stability. In that case, the continuity is enforced by the so-called numerical fluxes arising from the non conforming discretization of DG. On the other hand, CG is still a good solution, in [5] stability is ensured by the classical technique of overlapping one coarse element with the fine mesh. A similar example can be found in [70].

Finally, the most recent studies move towards massively parallel implementation. In [10] a new library is presented. In this appendix, the implementation is designed in parallel processing, but without memory parallelization. On the other hand, attention is devoted to the algorithm, which is fully decoupled from the time integration, allowing for implicit or explicit schemes. In the future, this procedure could be easily extended to shared memory parallelization.

B.1 Algorithm

As stated in [5, 41], the time step is driven by the finest mesh and the stability condition arising from the coarse-fine interface is overcome with a partial overlap. In the present case, the choice is to have a hierarchical structure of refined meshes fully overlapped, see Figure B.2. Apart from the advantages in parallel implementation, it allows to fully decouple the LTS from the time integration.

Having several meshes overlapped has an extra cost, since the coarse mesh has to be computed with the coarse time step at the refined region. However, its computational cost is insignificant in comparison with the resolution of the fine level. This step is considered as a predictor and is necessary for applying the boundary conditions at the fine level.

Hence, the stability relies on the boundary conditions applied to the fine level. The boundary conditions stated in chapter 2 does not necessary link all the variables, thus, some of the unknowns might not be continuous across the coarse-fine interface. This is the cost to pay for stability.

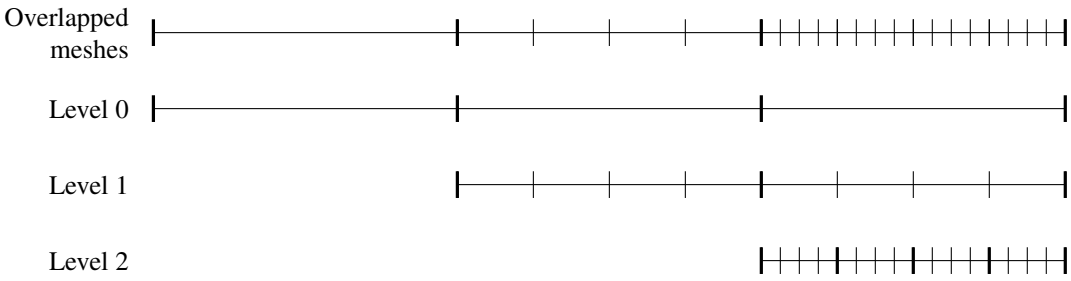


FIGURE B.2: Unfolding of the overlapped hierarchical refinement for a 1D mesh.

Once defined the spatial refinement, the hanging nodes are included in the algebraic system using *Multi-Point Constraints* (MPC). The values of a hanging node are computed by an average from the father nodes. This operation is performed recursively at all the divisions within the same hierarchic level.

Finally, after the prediction step of a coarse level, its sub-level is advanced with a smaller sub-step. Then, the interface is updated with the values from the prediction and is proceed to compute the solution of the sub-level. This procedure can be executed recursively. Once all the sub-steps have reached the end of the main step, the coarse prediction is updated with the values from the fine level. These steps are resumed in Figure B.3.

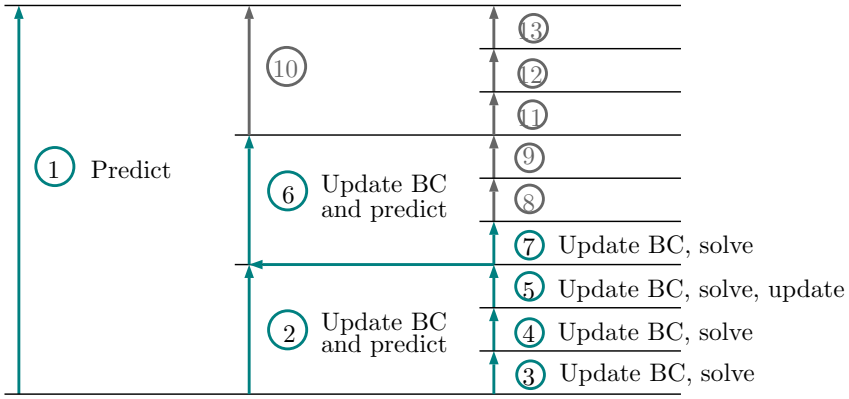


FIGURE B.3: Steps for solving a coarse time step with hierarchical refinement.

It is important to remark that this procedure does not impose any restriction on having a discontinuity of refinement level higher than one. Regardless of the jump of refinement level, the presented procedure is applied as well.

B.2 Data structure

The hierarchical refinement has been designed in order to be compatible with structured and unstructured meshes. Hence, the data structure is based on a finite element mesh, where nodes are defined by its coordinates and elements are defined by its connectivities.

The refinement begins when some elements are selected to refine according to a criterion which is to be defined. Then, those elements are copied to another mesh container, identified by the consecutive refinement level. The nodes are also copied. It is important to remark that the copied elements are only an auxiliary tool to define the connectivities. The mesh container identified with the refined level might contain elements already refined (Figure B.4a). During the copying process, the destination nodes save a pointer to the origin node and vice-versa.

Once the coarse connectivities are transferred to the refined mesh, an iterative process begins to refine the elements and conditions until the desired refinement is achieved (Figure B.4b). Every entity is split by dividing the edges into two. In the case of quadrilaterals, a node is added in the middle of the face, in the case of hexahedra, an extra node is added in the center of the volume. This operation requires some especial attention, because the edges and faces are shared by two elements. For this purpose, an auxiliary variable is defined in order to register the created entities. This variable is a map where the keys are the connectivities of the main edge and the mapped value is the middle node. The iterative process is controlled by the number of divisions executed at every element, each element has a variable storing

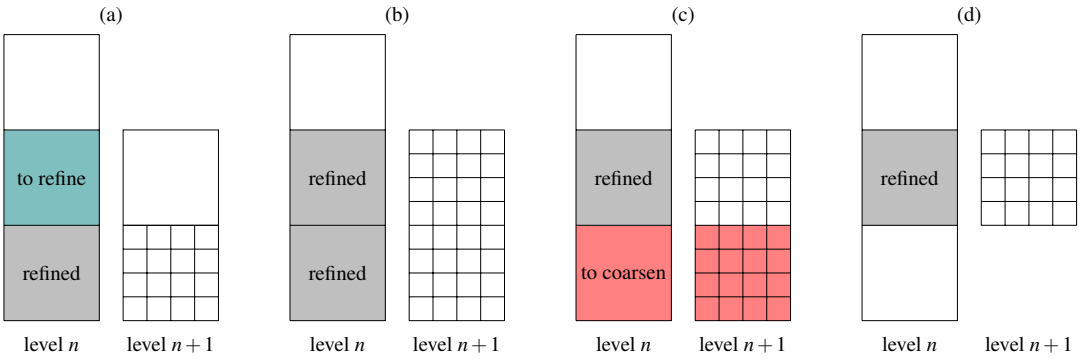


FIGURE B.4: Main steps for the refinement and coarsening process.

the number of divisions performed. After executing a division, this value is incremented by one. This process is repeated until the specified number of divisions are reached.

The coarsening process is straightforward. It begins when the refinement criterion marks some elements at the coarse level to coarsen. This information is transferred to the refined elements using an auxiliary variable that allows to map from a refined element to its parent element. Hence, each refined element asks to the parent element if it is to be coarsened (Figure B.4c).

The coarsening process is done by erasing all the elements to be coarsened from the refined level (Figure B.4d). The process is finished by cleaning the hanging nodes and the auxiliary variables. To sum up, the following variables are used by the refinement procedure:

LEVEL Scope: Mesh. The current refinement level.

REFINED Scope: Elements, conditions. Flag indicating if the current entity has a nested refinement.

TO_REFINE Scope: Elements, conditions. Flag indicating if it must be refined.

TO_COARSEN Scope: Elements, conditions. Flag indicating if it must be coarsened.

FATHER_NODES Scope: Nodes. If the node is overlapped with a coarse node, there is only one father node and it points to the coarse node. If the current node is refined, it points to the nodes in the refined mesh.

FATHER_NODES_WEIGHTS Scope: Nodes. The averaging of the nodal values. Has the same size than FATHER_NODES and the sum of the wights is 1.

SLAVE_NODE Scope: Nodes. It points to a cloned node in the refined mesh.

FATHER_ELEMENT Scope: Elements, conditions. It points to the father entity at he coarse mesh.

NUMBER_OF_DIVISIONS Scope: Elements, conditions. It counts how many times the entity has been divided.

NUMBER_OF_DIVISIONS Scope: Mesh. It signifies how many times the entities had to be divided.

NODES_MAP Scope: Mesh. A map pointing to a refined node from the fathers nodes.

This can be easily implemented in parallel with the standard reduction techniques, except the NODES_MAP. Adding new items to a map in parallel requires some extra architecture to ensure the thread safety. An alternative would be to split this map and reduce its scope to the nodes. This definition has some extra operations to write, read and remove values, but is worth for its scalability.

B.3 Refinement criterion

Finally, the refinement criterion has to be defined. There are two main criteria, the dynamic and the static one. The dynamic depends on the solution after the prediction step. In that case, the residual of the equations is evaluated –it is just the local right hand side when a residual-based system is used– and compared against a fixed value. When the local residual is too high, the element must be refined if there it has not been already refined. Whenever the local residual is sufficiently small, the element must be coarsened when it has a nested refinement.

The static criterion does not depend on the solution and the mesh is not modified during the analysis. This criterion serves to enhance the quality of the solution in a region. It is important to note that, by its hierarchical definition, this refinement cannot be used to have a better definition of the boundaries. This algorithm is specially suited for the dynamic criterion.

B.4 Examples

This section includes two basic examples. In the first example is shown how the mesh refinement can help to obtain better solutions. The second example is devoted to test its capabilities in a more complex test with a dynamic meshing.

B.4.1 Static refinement

In this example the propagation os a pulse is analyzed. Let $\Omega = [0, 10] \times [0, 1] m^2$ be the spatial domain. The initial water depth is $1m$ over all the domain, except near the right boundary, where there is an increment of $1m$. The initial velocity is null and for $t > 0s$ a wave is generated and propagated. Reflective conditions are considered over all the boundary, $\mathbf{u} \cdot \mathbf{n} = 0$ on Γ.

The domain is meshed with a structured triangular mesh of size $0.1m$. In this test a one-level refinement with two division is considered for $y > 0.5$. This criterion is static and kept constant during all the simulation. An overview of the mesh and a detail of the refinement is shown in Figure B.5.

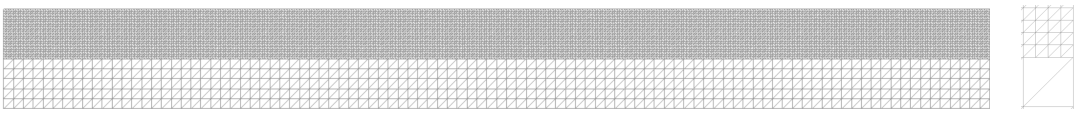


FIGURE B.5: Mesh used for the static refinement test and closeup of the refinement interface.

In Figure B.6 can be appreciated how the solutions obtained at the different regions are very different. In order to magnify this difference, the coarse and the fine domains are decoupled. Firstly, the boundary condition is not updated at the interface from the coarse prediction. Secondly, the coarse solution is not updated with the fine results.

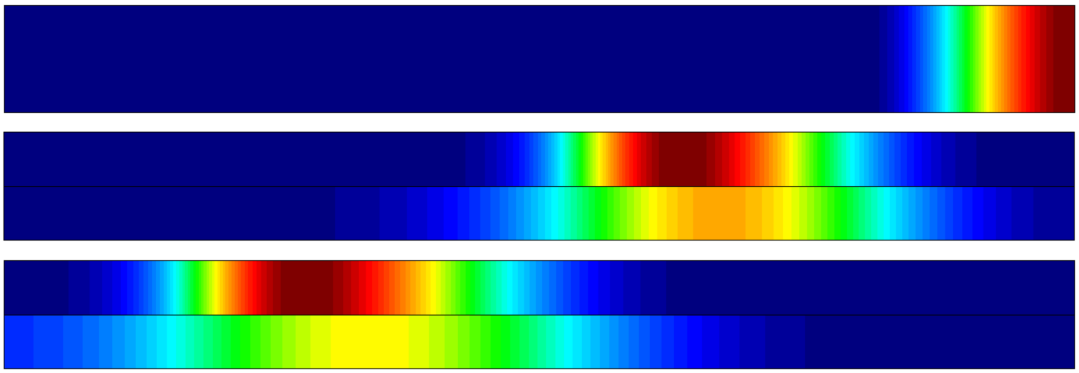


FIGURE B.6: Propagation of a pulse with static refinement. Results at time $t = 0, 1, 2s$, The different resolution levels are uncoupled.

When the two domains are coupled, the information is transferred between the two domains and the solution is modified. In Figure B.7 it is possible to appreciate how the wave front is is modified at each sub-domain and how the solution on one sub-domain is modified by the other. For simplicity, the legend is not shown and the contour plot represents only the relative values.

In this test two basic features of the hierarchic refinement are shown. First of all, an accurate discretization is required to obtain an accurate solution without dissipative effects (Figure B.6). Secondly, the robustness of the data exchange at the interface (Figure B.7).

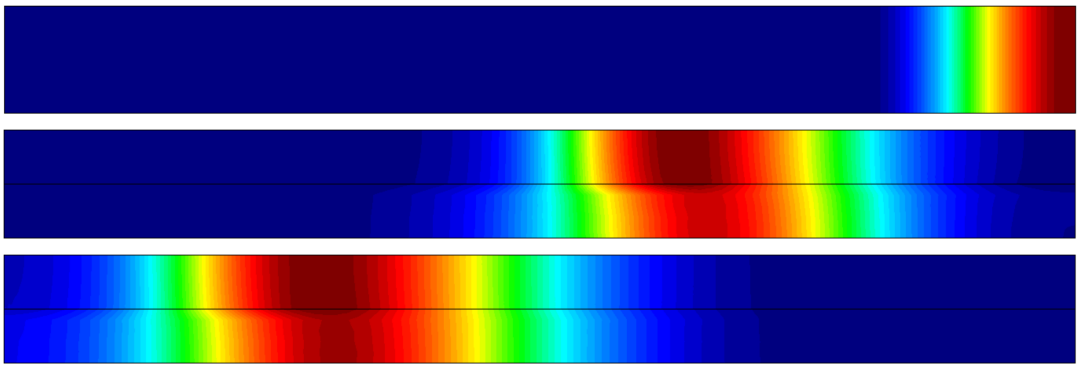


FIGURE B.7: Propagation of a pulse with static refinement. Results at time $t = 0, 1, 2s$. The different resolution levels are coupled.

B.4.2 Dynamic refinement

The second examples combines the hierarchical refinement with a residual-based criterion. Hence, the resolution level depends on the solution, this fact allows to test the refinement and coarsening processes.

A rectangular domain $\Omega = [0, 20] \times [0, 10]m^2$ is considered. The mean water depth is $0.1m$ and the solution is initialized with an increment of $0.1m$ at the coordinate $\mathbf{x}_0 = (4, 3)m$. Reflective conditions are considered over all the boundary, $\mathbf{u} \cdot \mathbf{n} = 0$ on Γ. After $t = 0$ a circular wave is generated, generating reflections at the boundaries which interact with each other.

The domain is initially discretized with an unstructured mesh of triangles of size $0.05m$. A refinement level is added when the local residual is greater than a threshold. When the wave front arrives to a material point, the refinement level is increased and after the wave front has passed, the refinement level is decreased. Two snapshots of the solution at time $t = 10$ and $20s$ are shown in Figure B.8.

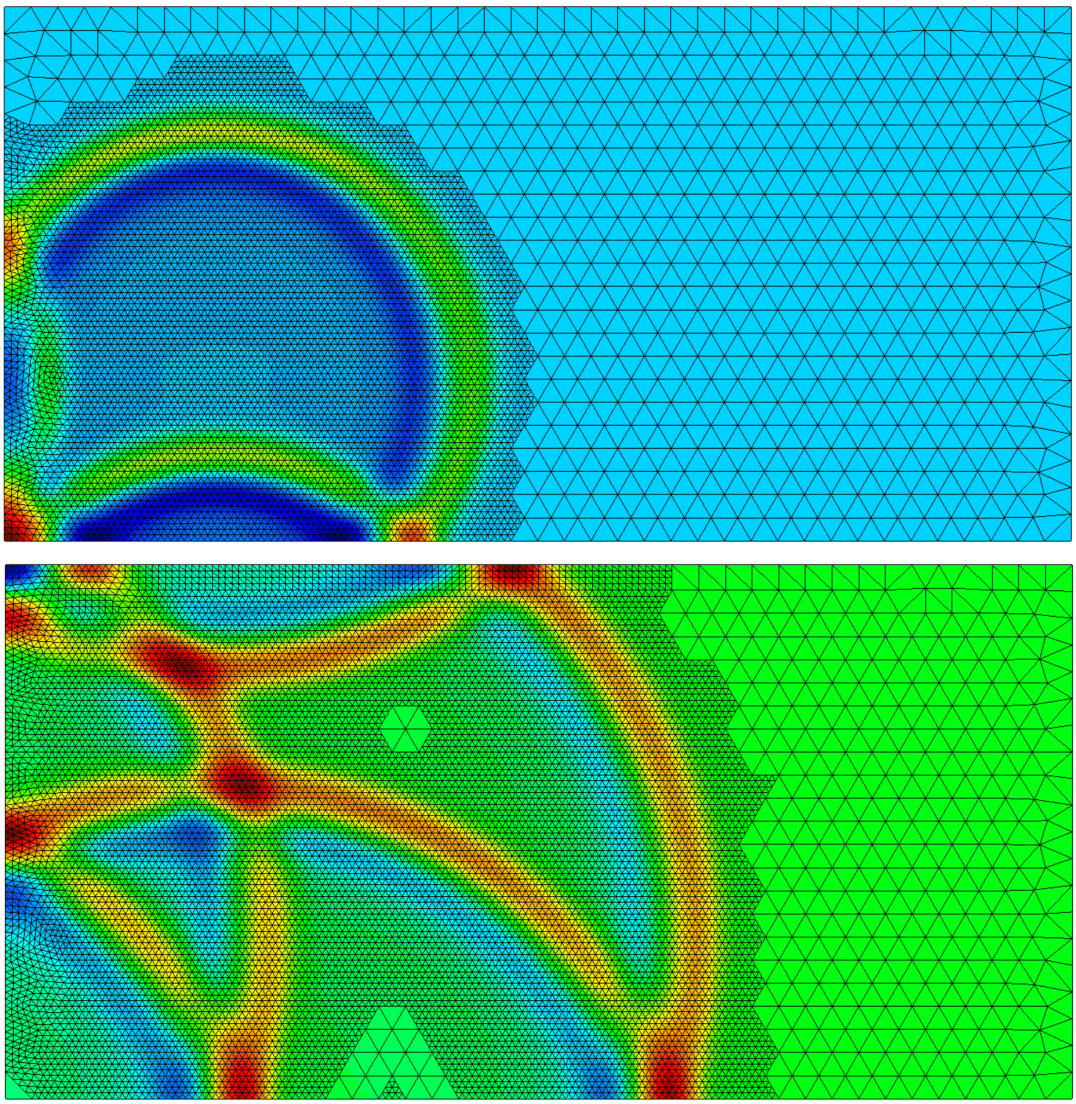


FIGURE B.8: Propagation of a wave with a dynamic refinement. Results at time $t = 10$ and $20s$. The different resolution levels are coupled.

B.5 Conclusions

The basic structure of a hierarchical refinement with LTS has been presented in this appendix. The design of the local refinement with LTS is decoupled from the time integration scheme, this is achieved with an overlap of the fine domain with the coarse one. This fact allows to use arbitrarily an explicit or implicit time step without compromising the stability of the method near the fine interface.

Despite duplicating the coarse degrees of freedom at the refined sub-domain, the computational requirements are not increased. The computation of the duplicated

degrees of freedom represents 1/64 over the cost of the fine solution, when two divisions are made. The duplicated degrees of freedom are part of a prediction step which is used to specify the boundary conditions at the interface during the sub stepping of the lower level. Moreover, this design extremely simplifies the refinement and coarsening procedures.

This type of refinement is specially interesting when obtaining an accurate solution is linked to a fine discretization in a small part of the domain. The hierarchical mesh refinement can be applied recursively in order to obtain several levels of accuracy depending on the quality of the solution.

Appendix C

Search algorithm

Usually, Lagrangian frameworks involve the computation of the streamlines or projections. The computation of the streamlines is related to the material derivative and is part of the differentiation. The projections are related to the communication between the Lagrangian and Eulerian domains, and them belong to the numerical environment. Those task require the search of material points (streamlines) or nodes (projections) over the spatial discretization. This happens both in the fixed mesh PFEM-2 algorithm or in the mesh moving PFEM algorithm. As counterpart to the simplicity in the system computation provided by the Lagrangian algorithms, a search algorithm is required. In this appendix, the search algorithm used and implemented in the presented work is explained.

The cost of the search of intersections or point vicinity to objects is quadratic, all the objects have to be tested against all. However, this cost can be drastically reduced if the domain is subdivided in cells or bins, containing a few objects. The *divide and conquer* strategy is based on the fact that the sum of the search of a few objects is smaller than the search of all the objects ($\sum \text{few} \cdot \text{few} \ll \sum \text{all} \cdot \text{all}$). The best method for dividing the objects in bins depends on the target geometries. Several trees can be defined using Axis-Aligned Bounding Boxes (AABB) [152] or Object-Oriented (OBB) [67]. Figure C.1 shows ans schematic of some tree partitions.

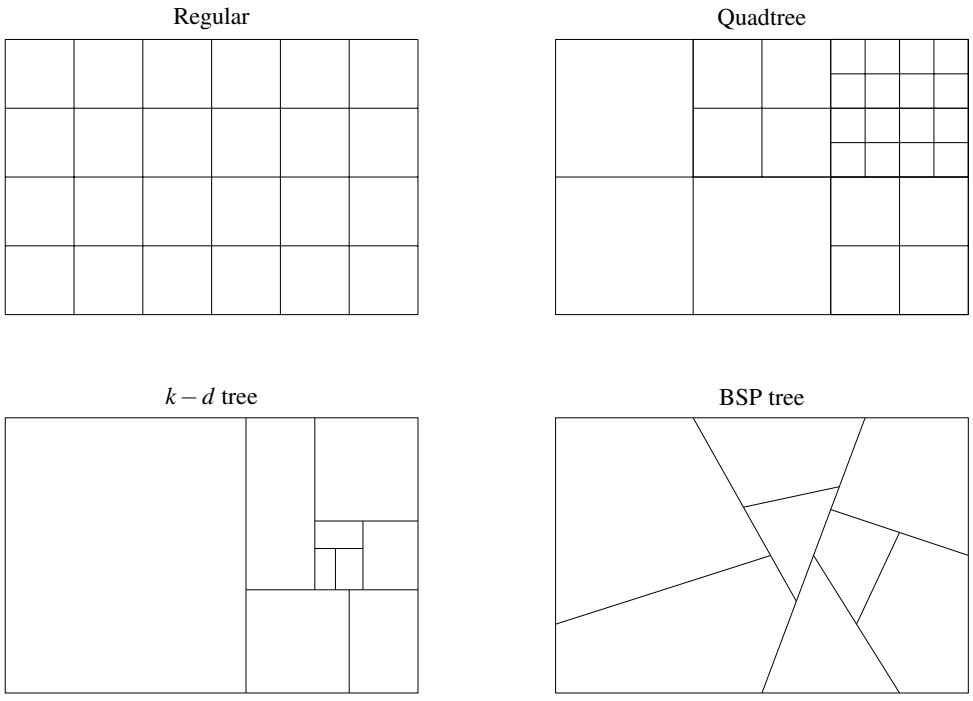


FIGURE C.1: Some common tree structures.

Since the computational domains are dense and we are not interested in the surface, but inside the body, an AABB structure will be employed (Fig. C.2). Firstly, the main domain is partitioned using a dynamic bins structure, a detailed explanation can be found in [152]. In brief, the dynamic bins are a search structure where the cell size is higher than the characteristic element size, each cell contains a set of elements. Once the search structure is initialized, the location of a point is performed in two steps. The location on the cells, and the location on the candidate elements.

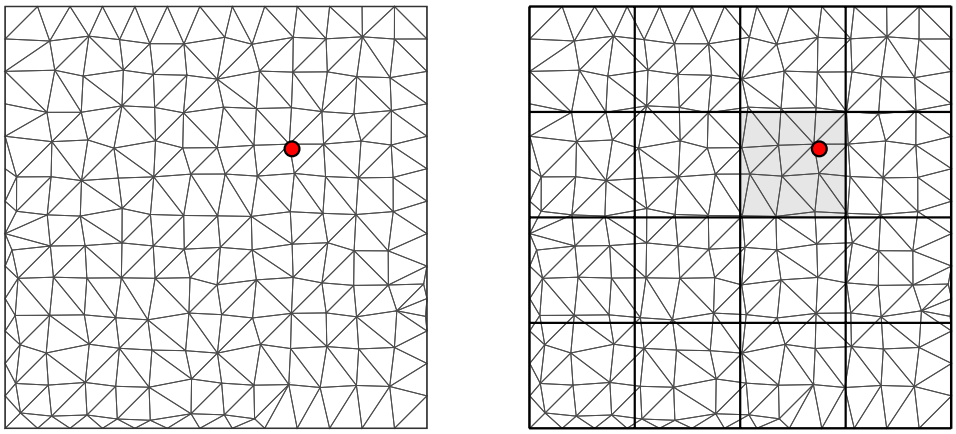


FIGURE C.2: Left: Mesh and entity to locate in the mesh. Right: Mesh and entity with the search structure (AABB)

Building the search structure has an initial cost, but is recovered during the search, specially when the mesh becomes larger. The definition of the cells is straightforward, while the main cost of the search structure building is the definition of the intersected elements against each cell. The element-cell intersections have been computed using the method described by Möller [4], described below.

Once the bins structure is finalized, the search process is quite simple. Besides the fast process of locating a node, the location of multiple nodes can be executed in parallel. Both properties make this algorithm very efficient.

Hereafter, the most representative algorithms are described. Other direct intersection algorithms such as ray-triangle [93, 131] or triangle-triangle [130] are used in this work, but for sake of brevity are not described here.

C.1 Triangle vs aligned box intersection

The full explanation of the algorithm can be found in [4]. Since this procedure is a generic tool for numerical methods, it has been applied for the shallow water, convection diffusion and Navier-Stokes equations. The generic 3D version is explained here, as well a the 2D particularization.

The derivation of this algorithm is based on the *Separating Axis Theorem* [67]. This theorem consists on looking fo a separating axis between the two objects. The algorithm looks for a separating axis along different directions. If a separating axis is found, the algorithm stops since there is not intersection. If all the tests pass, there is an intersection. Figure C.3 shows the 13 possible directions where a separating axis can be found. To sum up, the directions are the three cartesian basis, $\mathbf{e}_i$, the normal vector $\mathbf{n}$ and the nine combinations of the triangle edges against the cartesian basis, $\mathbf{f}_j \times \mathbf{e}_i$.

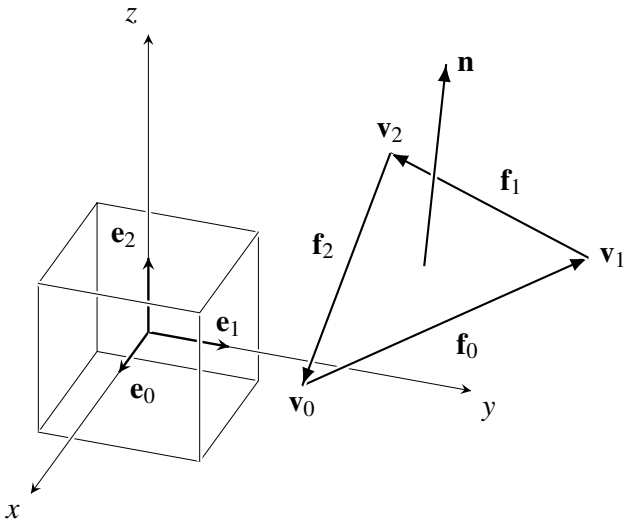


FIGURE C.3: Notation for the triangle-AABB intersection test.

According to Figure C.3, the first operation consists on a translation in order to get the AABB in the origin of coordinates. The tests to evaluate are:

Cartesian axis (3 tests) The first set of tests involves the comparison of the AABB against the triangle AABB on the three directions $\mathbf{e}_i$. In pseudocode, the search of a separating axis is:

```
for i = 0:2
    min, max = min_max(triangle_points)
    if (min > box_half_size or max < -box_half_size)
        return true
return false
```

Normal vector (1 test) The second set of tests consists on the AABB-plane comparison. The plane containing the triangle is defined with the unit vector $\mathbf{n}$ and the distance to origin d. The test is performed in the same quadrant of $\mathbf{n}$ and is analogous to the previous one.

```
for i = 0:2
    if (normal[i] > 0)
        box_half_size_proj += box_half_size[i] * normal[i]
    else
        box_half_size_proj -= box_half_size[i] * normal[i]
if (distance > box_half_size_proj)
    return true
return false
```

Edges (9 tests) A test is performed for every axis $\mathbf{a}_{ij} = \mathbf{n}_i \times \mathbf{f}_j$. Each tests involves a projection of the triangle and the AABB. Fortunately, the expansion of the projections allow to make some simplifications. For the first projection we

have:

$$p_0 = \mathbf{a}_{00} \cdot \mathbf{v}_0 = (0, -f_{0z}, f_{0,y}) \cdot \mathbf{v}_0 = v_{0z}v_{1y} - v_{0y}v_{1z}$$
$$p_1 = \mathbf{a}_{00} \cdot \mathbf{v}_1 = (0, -f_{0z}, f_{0,y}) \cdot \mathbf{v}_1 = v_{0z}v_{1y} - v_{0y}v_{1z} = p_0$$
$$p_2 = \mathbf{a}_{00} \cdot \mathbf{v}_2 = (0, -f_{0z}, f_{0,y}) \cdot \mathbf{v}_2 = (v_{1y} - v_{0y})v_{2z} - (v_{1z} - v_{0z})v_{2y}$$

The fact that $p_0 = p_1$ makes easier the finding of the maximum and minimum of p_0, p_1 y p_2. Those are compared against the *radius* of the box, which is the projection of the corner onto $\mathbf{a}_{00}$:

$$r = h_x|a_{00x}| + h_y|a_{00y}| + h_z|a_{00z}|$$

Finally, finding a separating axis is:

```
if (min(p0, p2) > r or max(p0, p2) < -r)
    return true
```

If the 13 tests return false, it means there is not a separating axis, thus, there is not intersection. Note that if one test returns true, the algorithm stops since there would not be a possible intersection. In practice, the bullet 3 is evaluated at first, then bullet 1 and finally, bullet 2.

C.1.1 2D particularization

The two dimensional case is a simplification of the previous algorithm and some test can be omitted. Only five tests are needed:

Cartesian axis (2 tests) A separating axis is sought over x and y.

Normal vector (none) Is omitted.

Edges (3 tests) Only the projections of the edges $\mathbf{f}_j$ against the edge $\mathbf{e}_z$ are relevant.

C.2 Quadrilateral vs aligned box intersection

The above algorithm is easily extrapolated to quadrilaterals. On the one hand, there is an extra edge test in the third bullet. On the other hand, looking for the minimum and maximum coordinate in the first bullet, involves more conditionals.

However, in practice, each quadrilateral is subdivided in two triangles and two comparisons are made. Even the number of tests is greater, this strategy allows to reduce code duplication and to make easier the code maintaining. This approach is done since we are not interested on the code optimization, but on the evaluation of the FEM.

C.3 Point intersection

Determining if a point is inside a linear triangle or bilinear quadrilateral is a straightforward task. This procedure involves determining the local coordinates and verifying if all of them are between 0 and 1 for triangles, or between -1 and 1 for quadrilaterals. An equivalent verification is expressed in terms of the shape functions N, this involves less conditionals than the previous case:

```
function is_inside
    for i = 1 : n_nodes
        if N(i) < 0
            return false
    return true
```

Bibliography

[1] M. B. Abbott. *Computational hydraulics : elements of the theory of free surface flows / M.B. Abbott*. Pitman Pub. ; Fearon-Pitman Publishers London : Belmont, Calif, 1979, xviii, 324 p. : ISBN: 0273011405.

[2] M. B. Abbott, H. M. Petersen, and O. Skovgaard. "On the numerical modelling of short waves in shallow water". In: *Journal of Hydraulic Research* 16.3 (1978), pp. 173–204. DOI: 10.1080/00221687809499616.

[3] M. Ahmed, D. Singh, S. Alqadhi, and M. Alrefae. "Improvement of the Zienkiewicz-Zhu Error Recovery Technique Using a Patch Configuration". In: *Applied Sciences* 11 (Sept. 2021), p. 8120. DOI: 10.3390/app11178120.

[4] T. Akenine-moller. "Fast 3D Triangle-Box Overlap Testing". In: *Journal of Graphics Tools* 6 (May 2004). DOI: 10.1145/1198555.1198747.

[5] M. Almquist and M. Mehlin. "MultiLevel Local Time-Stepping Methods of Runge–Kutta-type for Wave Equations". In: *SIAM Journal on Scientific Computing* 39 (Jan. 2017), A2020–A2048. DOI: 10.1137/16M1084407.

[6] V. Ambati and O. Bokhove. "Space-time discontinuous Galerkin finite element method for shallow water flows". In: *Journal of Computational and Applied Mathematics* 204.2 (2007). Special Issue: The Seventh International Conference on Mathematical and Numerical Aspects of Waves (WAVES'05), pp. 452 –462. ISSN: 0377-0427. DOI: 10.1016/j.cam.2006.01.047.

[7] B. Ataie-Ashtiani and S. Malek-Mohammadi. "Mapping impulsive waves due to sub-aerial landslides into a dam reservoir: a case study of Shafa-Roud Dam". In: *Dam Engineering* 18.4 (2008), p. 243.

[8] S. Badia. "On stabilized finite element methods based on the Scott-Zhang projector. Circumventing the inf-sup condition for the Stokes problem". In: *Computer Methods in Applied Mechanics and Engineering* 247-248 (2012), pp. 65–72. ISSN: 0045-7825. DOI: 10.1016/j.cma.2012.07.020.

[9] S. Badia and A. Hierro. "On monotonicity-preserving stabilized finite element approximations of transport problems". In: *SIAM J. Sci. Comput.* 36.6 (2014). DOI: 10.1137/130927206.

[10] J. Baiges and C. Bayona. "Refficientlib: An Efficient Load-Rebalanced Adaptive Mesh Refinement Algorithm for High-Performance Computational Physics Meshes". In: *SIAM Journal on Scientific Computing* 39 (Jan. 2016). DOI: 10.1137/15M105330X.

[11] M. L. C. Barros, P. C. C. Rosman, J. C. F. Telles, and J. P. S. Azevedo. "A simple wetting and drying method for shallow water flow with application in the Vitória Bay estuary, Brazil". In: *Water Resources Management VI*. Vol. 145. 2011, pp. 215 –225. ISBN: 978-1-84564-514-4. DOI: 10.2495/WRM110181.

[12] A. Bayliss, M. Gunzburger, and E. Turkel. "Boundary Conditions for the Numerical Solution of Elliptic Equations in Exterior Regions". In: *Siam Journal on Applied Mathematics - SIAMAM* 42 (Apr. 1982). DOI: 10.1137/0142032.

[13] M. Bercovier and O. Pironneau. "Error Estimates for Finite Element Method Solution of the Stokes Problem in the Primitive Variables." In: *Numerische Mathematik* 33 (1979), pp. 211–224.

[14] J.-P. Berenger. "A perfectly matched layer for the absorption of electromagnetic waves". In: *Journal of Computational Physics* 114.2 (1994), pp. 185–200. ISSN: 0021-9991. DOI: 10.1006/jcph.1994.1159.

[15] D. Book, J. Boris, and K. Hain. "Flux-corrected transport II: Generalizations of the method". In: *Journal of Computational Physics* 18.3 (1975), pp. 248–283. ISSN: 0021-9991. DOI: 10.1016/0021-9991(75)90002-9.

[16] J. P. Boris and D. L. Book. "Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works". In: *Journal of Computational Physics* 11.1 (1973), pp. 38–69. ISSN: 0021-9991. DOI: 10.1016/0021-9991(73)90147-2.

[17] M. Braack and E. Burman. "Local Projection Stabilization for the Oseen Problem and its Interpretation as a Variational Multiscale Method". In: *SIAM Journal on Numerical Analysis* 43.6 (2006), pp. 2544–2566. DOI: 10.1137/050631227.

[18] S. Bradford and B. Sanders. "Finite-volume models for unidirectional, nonlinear, dispersive waves". In: *Journal of Waterway Port Coastal and Ocean Engineering* 128 (July 2002), pp. 173–182. DOI: 10.1061/(ASCE)0733-950X(2002)128:4(173(.

[19] R. Brayton, F. Gustavson, and G. Hachtel. "A new efficient algorithm for solving differential-algebraic systems using implicit backward differentiation formulas". In: *Proceedings of the IEEE* 60.1 (1972), pp. 98–108. DOI: 10.1109/PROC.1972.8562.

[20] D. Bresch, D. Lannes, and G. Métivier. "Waves interacting with a partially immersed obstacle in the Boussinesq regime". In: *Analysis & PDE* 14.4 (2021), pp. 1085–1124. DOI: 10.2140/apde.2021.14.1085.

[21] F. Brezzi and M. Fortin. "Mixed and Hybrid Finite Element Method". In: *Springer Series In Computational Mathematics; Vol. 15* (Jan. 1991), p. 350. DOI: 10.1007/978-1-4612-3172-1.

[22] A. Brooks and T. Hughes. "Streamline Upwind/Petrov-Galerkin formulations for convective dominated flows with particular emphasis on the incompressible Navier-Stokes equations". In: *Computational Methods Applied to Mechanical Engineering* 32.1 (1982), pp. 199–259. DOI: 10.1016/0045-7825(82)90071-8.

[23] G. Bullard, R. Mulligan, A. Carreira, and W. Take. "Experimental analysis of tsunamis generated by the impact of landslides with high mobility". In: *Coastal Engineering* 152 (2019), p. 103538. ISSN: 0378-3839. DOI: 10.1016/j.coastaleng.2019.103538.

[24] E. Burman. "A monotonicity preserving, nonlinear, finite element upwind method for the transport equation". In: *Applied Mathematics Letters* 49 (May 2015). DOI: 10.1016/j.aml.2015.05.005.

[25] E. Burman. "Consistent SUPG-method for transient transport problems: Stability and convergence". In: *Computer Methods in Applied Mechanics and Engineering* 199.17 (2010), pp. 1114–1123. ISSN: 0045-7825. DOI: 10.1016/j.cma.2009.11.023.

[26] E. Burman. "On nonlinear artificial viscosity, discrete maximum principle and hyperbolic conservation laws". In: *BIT Numerical Mathematics* 47 (Dec. 2007), pp. 715–733. DOI: 10.1007/s10543-007-0147-7.

[27] E. Burman, S. Claus, P. Hansbo, M. G. Larson, and A. Massing. "CutFEM: Discretizing geometry and partial differential equations". In: *International Journal for Numerical Methods in Engineering* 104.7 (2015), pp. 472–501. DOI: 10.1002/nme.4823.

[28] A. S. Candy. "An implicit wetting and drying approach for non-hydrostatic baroclinic flows in high aspect ratio domains". In: *Advances in Water Resources* 102 (2017), pp. 188 –205. ISSN: 0309-1708. DOI: 10.1016/j.advwatres.2017.02.004.

[29] R. A. Carmigniani and D. Violeau. "Optimal sponge layer for water waves numerical models". In: *Ocean Engineering* 163 (2018), pp. 169–182. ISSN: 0029-8018. DOI: 10.1016/j.oceaneng.2018.05.068.

[30] E. B. Castellet. "Modelación del flujo en lámina libre sobre cauces naturales. Análisis integrado con esquemas en volúmenes finitos en una y dos mensiones". PhD thesis. Universitat Politécnica de Catalunya, July 2005.

[31] S. Chen, B. Merriman, S. Osher, and P. Smereka. "A Simple Level Set Method for Solving Stefan Problems". In: *Journal of Computational Physics* 135.1 (1997), pp. 8–29. ISSN: 0021-9991. DOI: 10.1006/jcph.1997.5721.

[32] M. Chevalier, R. Luebbers, and V. Cable. "FDTD local grid with material traverse". In: *IEEE Transactions on Antennas and Propagation* 45.3 (1997), pp. 411–421. DOI: 10.1109/8.558656.

[33] V. T. Chow, D. R. Maidment, and L. W. May. *Applied hydrology*. eng. McGraw-Hill series in water resources and environmental engineering. New York: McGraw-Hill Book Company, 1988. ISBN: 0070108102.

[34] R. Codina. "Analysis of a stabilized finite element approximation of the Oseen equations using orthogonal subscales". In: *Applied Numerical Mathematics* 58.3 (2008), pp. 264–283. ISSN: 0168-9274. DOI: 10.1016/j.apnum.2006.11.011.

[35] R. Codina. "Comparison of some finite element methods for solving the diffusion-convection-reaction equation". In: *Computer Methods in Applied Mechanics and Engineering* 156.1 (1998), pp. 185 –210. ISSN: 0045-7825. DOI: 10.1016/S0045-7825(97)00206-5.

[36] R. Codina. "Finite Element Approximation of the Convection-Diffusion Equation: Subgrid-Scale Spaces, Local Instabilities and Anisotropic Space-Time Discretizations". In: *BAIL 2010 - Boundary and Interior Layers, Computational and Asymptotic Methods*. Ed. by C. Clavero, J. L. Gracia, and F. J. Lisbona. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 85–97. ISBN: 978-3-642-19665-2. DOI: 10.1007/978-3-642-19665-2_10.

[37] R. Codina. "Finite element approximation of the hyperbolic wave equation in mixed form". In: *Computer Methods in Applied Mechanics and Engineering* 197.13 (2008), pp. 1305 –1322. ISSN: 0045-7825. DOI: 10.1016/j.cma.2007.11.006.

[38] R. Codina. "Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods". In: *Computer Methods in Applied Mechanics and Engineering* 190 (Dec. 2000), pp. 1579–1599. DOI: 10.1016/S0045-7825(00)00254-1.

[39] R. Codina, J. M. González-Ondina, G. Díaz-Hernández, and J. Principe. "Finite element approximation of the modified Boussinesq equations using a stabilized formulation". In: *International Journal for Numerical Methods in Fluids* 57.9 (2008), pp. 1249–1268. DOI: 10.1002/fld.1718.

[40] F. Collino. "High Order Absorbing Boundary Conditions for Wave Propagation Models.Straight Line Boundary and Corner Cases". In: *2nd Int. Conf. on Mathematical & Numerical Aspects of Wave Propagation*. Ed. by R. K. et al. SIAM, Delaware, 1993, pp. 161–171.

[41] F. Collino, T. Fouquet, and P. Joly. "A Conservative Space-Time Mesh Refinement Method for the 1-D Wave Equation. Part I: Construction". In: *Numer. Math.* 95.2 (Aug. 2003), pp. 197–221. ISSN: 0029-599X. DOI: 10.1007/s00211-002-0446-5.

[42] F. Collino, T. Fouquet, and P. Joly. "A Conservative Space-Time Mesh Refinement Method for the 1-D Wave Equation. Part II: Analysis". In: *Numer. Math.* 95.2 (Aug. 2003), pp. 223–251. ISSN: 0029-599X. DOI: 10.1007/s00211-002-0447-4.

[43] J. Cotela-Dalmau, R. Rossi, and E. Oñate. "A FIC-based stabilized finite element formulation for turbulent flows". In: *Computer Methods in Applied Mechanics and Engineering* 315 (Nov. 2016). DOI: 10.1016/j.cma.2016.11.020.

[44] M. Cremonesi, F. Ferri, and U. Perego. "A basal slip model for Lagrangian finite element simulations of 3D landslides". In: *International Journal for Numerical and Analytical Methods in Geomechanics* 41 (June 2016), pp. 30–53. DOI: 10.1002/nag.2544.

[45] M. Cremonesi, A. Frangi, and U. Perego. "A Lagrangian finite element approach for the simulation of water-waves induced by landslides". In: *Computer and Structures* 89 (June 2011), pp. 1086–1093. DOI: 10.1016/j.compstruc.2010.12.005.

[46] M. Cremonesi, A. Franci, S. Idelsohn, and E. Oñate. "A state of the art review of the particle finite element method (PFEM)". In: *Archives of Computational Methods in Engineering* 27.5 (2020), pp. 1709–1735. DOI: 10.1007/s11831-020-09468-4.

[47] G. Crosta, S. Imposimato, and D Roddeman. "Landslide spreading, impulse water waves and modelling of the Vajont rockslide". In: *Rock Mechanics and Rock Engineering* 49(6) (2016), pp. 2413–2436. DOI: 10.1007/s00603-015-0769-z.

[48] C. F. Curtiss and J. O. Hirschfelder. "Integration of Stiff Equations". In: *Proceedings of the National Academy of Sciences of the United States of America* 38(3) (1952), pp. 235–43. DOI: 10.1073/pnas.38.3.235.

[49] P. Dadvand, R. Rossi, M. Gil, X. Martorell, J. Cotela, E. Juanpere, S. Idelsohn, and E. Oñate. "Migration of a generic multi-physics framework to HPC environments". In: *Computers and Fluids* 80 (2013), pp. 301–309. DOI: 10.1016/j.compfluid.2012.02.004.

[50] P. Dadvand, R. Rossi, and E. Oñate. "An object-oriented environment for developing finite element codes for multi-disciplinary applications". In: *Archives of Computational Methods in Engineering* 17(3) (2010), pp. 253–297. DOI: 10.1007/s11831-010-9045-2.

[51] A. Defina. "Two-dimensional shallow flow equations for partially dry areas". In: *Water Resources Research - WATER RESOUR RES* 36 (Nov. 2000). DOI: 10.1029/2000WR900167.

[52] F. Del Pin, S. Idelsohn, E. Oñate, and R. Aubry. "The ALE/Lagrangian particle finite element method: a new approach to computation of free-surface flows and fluid-object interactions". In: *Computers and Fluids* 36.1.1 (2007), pp. 27–38. DOI: 10.1016/j.compfluid.2005.06.008.

[53] O. Delestre, C. Lucas, P.-A. Ksinant, F. Darboux, C. Laguerre, T.-N.-T. Vo, F. James, and S. Cordier. "SWASHES: a compilation of shallow water analytic solutions for hydraulic and environmental studies". In: *International Journal for Numerical Methods in Fluids* 72 (May 2013), pp. 269–300. DOI: 10.1002/fld.3741. eprint: 1110.0288.

[54] J. Diaz and M. J. Grote. "Energy Conserving Explicit Local Time Stepping for Second-Order Wave Equations". In: *SIAM Journal on Scientific Computing* 31.3 (2009), pp. 1985–2014. DOI: 10.1137/070709414.

[55] J. Donea, A. Huerta, J.-P. Ponthot, and A. Rodríguez-Ferran. "Arbitrary Lagrangian-Eulerian Methods". In: *Encyclopedia of Computational Mechanics*. John Wiley & Sons, Ltd, 2004. Chap. 14. ISBN: 9780470091357. DOI: 10.1002/0470091355.ecm009.

[56] B. Engquist and A. Majda. "Absorbing Boundary Conditions for the Numerical Simulation of Waves". In: *Mathematics of Computation* 31.139 (1977), pp. 629–651. ISSN: 00255718, 10886842.

[57] C. Eskilsson and S. J. Sherwin. "A Discontinuous Spectral Element Model for Boussinesq-Type Equations". In: *Journal of Scientific Computing* 12 (2002), pp. 143–152. DOI: 10.1023/A:1015144429543.

[58] O. Eugenio. "Finite increment calculus (FIC): a framework for deriving enhanced computational methods in mechanics". In: *Advanced Modeling and Simulation in Engineering Sciences* 3.1 (Apr. 2016), p. 14. ISSN: 2213-7467. DOI: 10.1186/s40323-016-0065-9.

[59] L. Formaggia, J. Gerbeau, F. Nobile, and A. Quarteroni. "On the coupling of 3D and 1D Navier-Stokes equations for flow problems in compliant vessels". In: *Computer Methods in Applied Mechanics and Engineering* 191.6 (2001). Minisymposium on Methods for Flow Simulation and Modeling, pp. 561–582. ISSN: 0045-7825. DOI: 10.1016/S0045-7825(01)00302-4.

[60] A. Franci, M. Cremonesi, U. Perego, G. Crosta, M. Masó, I. D. Pouplana, and E. Oñate. "Numerical Modelling of Landslide-Generated Waves". In: *ECCOMAS*. 2022.

[61] A. Franci, M. Cremonesi, U. Perego, G. Crosta, and E. Oñate. "3D Simulation of Vajont disaster. Part 1: Numerical formulation and validation". In: *Engineering Geology* 279 (2020), p. 105854. DOI: 10.1016/j.enggeo.2020.105854.

[62] A. Franci, M. Cremonesi, U. Perego, E. Oñate, and G. Crosta. "3D Simulation of Vajont disaster. Part 2: Multi-failure scenarios". In: *Engineering Geology* 279 (2020), p. 105856. DOI: 10.1016/j.enggeo.2020.105856.

[63] A. Franco, B. Schneider-Muntau, N. J. Roberts, J. J. Clague, and B. Gems. "Geometry-Based Preliminary Quantification of Landslide-Induced Impulse Wave Attenuation in Mountain Lakes". In: *Applied Sciences* 11.24 (2021). ISSN: 2076-3417. DOI: 10.3390/app112411614.

[64] H. M. Fritz, W. H. Hager, and H.-E. Minor. "Near field characteristics of landslide generated impulse waves". In: *Journal of waterway, port, coastal, and ocean engineering* 130.6 (2004), pp. 287–302. DOI: 10.1061/(ASCE)0733-950X(2004)130:6(287).

[65] M. Geveler, D. Ribbrock, D. Göddeke, and S. Turek. "Lattice-Boltzmann Simulation of the Shallow-Water Equations with Fluid-Structure Interaction on Multi- and Manycore Processors". In: vol. 6310. Oct. 2010, pp. 92–104. ISBN: 978-3-642-16232-9. DOI: 10.1007/978-3-642-16233-6_11.

[66] S. K. Godunov. "A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics". In: *Matematicheskii Sbornik* 89.3 (1959), pp. 271–306.

[67] S. Gottschalk, M. C. Lin, and D. Manocha. "OBBTree: A Hierarchical Structure for Rapid Interference Detection". In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '96. New York, NY, USA: Association for Computing Machinery, 1996, pp. 171–180. ISBN: 0897917464. DOI: 10.1145/237170.237244.

[68] O. Gourgue, R. Comblen, J. Lambrechts, T. Kärnä, V. Legat, and E. Deleersnijder. "A flux-limiting wetting-drying method for finite-element shallow-water models, with application to the Scheldt Estuary". In: *Advances in Water Resources* 32.12 (2009), pp. 1726–1739. ISSN: 0309-1708. DOI: 10.1016/j.advwatres.2009.09.005.

[69] E. Grimstad and S. Nesdal. "The Loen rockslides—a historical review". In: *Publikasjon-Norges Geotekniske Institutt* 182 (1991), pp. 1–6.

[70] M. J. Grote, S. Michel, and S. A. Sauter. "Stabilized leapfrog based local time-stepping method for the wave equation". In: *Math. Comp.* 90 (June 2021), pp. 2603–2643. DOI: 10.1090/mcom/3650.

[71] J.-L. Guermond, R. Pasquetti, and B. Popov. "Entropy viscosity method for nonlinear conservation laws". In: *Journal of Computational Physics* 230.11 (2011), pp. 4248–4267. DOI: 10.1016/j.jcp.2010.11.043.

[72] K. Guizien and B. Eric. "Accuracy of solitary wave generation by a piston wave maker". In: *Journal of Hydraulic Research* May 2002 (Feb. 2010), pp. 321–331. DOI: 10.1080/00221680209499946.

[73] T. Hagstrom, D. Givoli, D. Rabinovich, and J. Bielak. "The Double Absorbing Boundary Method". In: *J. Comput. Phys.* 259.C (Feb. 2014), pp. 220–241. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2013.11.025.

[74] E. Hanert, D. Y. L. Roux, V. Legat, and E. Deleersnijder. "An efficient Eulerian finite element method for the shallow water equations". In: *Ocean Modelling* 10.1-2 (2005). The Second International Workshop on Unstructured Mesh Numerical Modelling of Coastal, Shelf and Ocean Flows, pp. 115 –136. ISSN: 1463-5003. DOI: 10.1016/j.ocemod.2004.06.006.

[75] U. Haque et al. "The human cost of global warming: Deadly landslides and their triggers (1995-2014)". In: *Science of the Total Environment* 673 (2019).

[76] C. Harbitz, S. Glimsdal, F. Løvholt, V. Kveldsvik, G. Pedersen, and A. Jensen. "Rockslide tsunamis in complex fjords: From an unstable rock slope at Åkerneset to tsunami risk in western Norway". In: *Coastal Engineering* 88 (2014), pp. 101–122. DOI: 10.1016/j.coastaleng.2014.02.003.

[77] P. Heinrich, A. Mangeney, S. Guibourg, R. Roche, G. Boudon, and J.-L. Cheminée. "Simulation of water waves generated by a potential debris avalanche in Montserrat, Lesser Antilles". In: *Geophysical Research Letters* 25.19 (1998), pp. 3697–3700. DOI: 10.1016/S1464-1895(00)00115-0.

[78] M. Heniche, Y. Secretan, P. Boudreau, and M. Leclerc. "A two-dimensional finite element drying-wetting shallow water model for rivers and estuaries". In: *Advances in Water Resources* 23.4 (2000), pp. 359 –372. ISSN: 0309-1708. DOI: 10.1016/S0309-1708(99)00031-7.

[79] R. L. Hermanns, T. Oppikofer, N. J. Roberts, and G. Sandøy. "Catalogue of historical displacement waves and landslide-triggered tsunamis in Norway". In: *Engineering Geology for Society and Territory–Volume 4*. Springer, 2014, pp. 63–66. DOI: 10.1007/978-3-319-08660-6_13.

[80] B. Higman et al. "The 2015 landslide and tsunami in Taan Fiord, Alaska". In: *Scientific reports* 8.1 (2018), pp. 1–12. DOI: 10.1038/s41598-018-30475-w.

[81] P. Hood and C. Taylor. *Navier Stokes equations using mixed interpolation. Finite-element Methods in Flow Problems*. Huntsville Press, 1974.

[82] T. J. Hughes. "Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods". In: *Computer Methods in Applied Mechanics and Engineering* 127.1 (1995), pp. 387–401. ISSN: 0045-7825. DOI: 10.1016/0045-7825(95)00844-9.

[83] T. J. Hughes, L. P. Franca, and G. M. Hulbert. "A new finite element formulation for computational fluid dynamics: VIII. The Galerkin least-squares method for advective-diffusive equations". In: *Computer Methods in Applied Mechanics and Engineering* 73.2 (1989), pp. 173–189. ISSN: 0045-7825.

[84] T. J. Hughes and M. Mallet. "A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective-diffusive systems". In: *Computer Methods in Applied Mechanics and Engineering* 58.3 (1986), pp. 305–328. ISSN: 0045-7825. DOI: 10.1016/0045-7825(86)90152-0.

[85] T. J. Hughes and M. Mallet. "A new finite element formulation for computational fluid dynamics: IV. A discontinuity-capturing operator for multidimensional advective-diffusive systems". In: *Computer Methods in Applied Mechanics and Engineering* 58.3 (1986), pp. 329–336. ISSN: 0045-7825. DOI: 10.1016/0045-7825(86)90153-2.

[86] S. Idelsohn, N. Nigro, J. Gimenez, R. Rossi, and J. Marti. "A fast and accurate method to solve the incompressible Navier-Stokes equations." In: *Engineering Computations: Int. J. for Computer-Aided Engineering* 30.2 (2013), pp. 197–222. DOI: 10.1108/02644401311304854.

[87] S. Idelsohn, E. Oñate, N. Calvo, and F. Del Pin. "The meshless finite element method". In: *International Journal for Numerical Methods in Engineering* 58.6 (2003), pp. 893–912. DOI: 10.1002/nme.798.

[88] S. Idelsohn, E. Oñate, and F. D. Pin. "The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves". In: *International Journal for Numerical Methods in Engineering* 61.7 (2004), pp. 964–989. DOI: 10.1002/nme.1096.

[89] S. R. Idelsohn, J. Marti, P. Becker, and E. Oñate. "Analysis of multifluid flows with large time steps using the particle finite element method". In: *International Journal for Numerical Methods in Fluids* 75.9 (2014), pp. 621–644. DOI: 10.1002/fld.3908.

[90] S. Idelsohn, N. Nigro, A. Limache, and E. Oñate. "Large time-step explicit integration method for solving problems with dominant convection Comput". In: *Computer Methods in Applied Mechanics and Engineering* s 217-220 (Apr. 2012), pp. 168–185. DOI: 10.1016/j.cma.2011.12.008.

[91] S. Idelsohn, E. Oñate, N. Nigro, P. Becker, and J. Gimenez. "Lagrangian versus Eulerian integration errors". In: *Computer Methods in Applied Mechanics and Engineering* 293 (May 2015). DOI: 10.1016/j.cma.2015.04.003.

[92] M. Israeli and S. A. Orszag. "Approximation of radiation boundary conditions". In: *Journal of Computational Physics* 41.1 (1981), pp. 115–135. ISSN: 0021-9991. DOI: 10.1016/0021-9991(81)90082-6.

[93] J. J. Jiménez, R. J. Segura, and F. R. Feito. "A robust segment / triangle intersection algorithm for interference tests. Efficiency study". In: *Computational Geometry* 43.5 (2010), pp. 474–492. ISSN: 0925-7721. DOI: 10.1016/j.comgeo.2009.10.001.

[94] C. Johnson, A. Szepessy, and P. Hansbo. "On the convergence of shock-capturing streamline diffusion finite element methods for hyperbolic conservation laws". In: *Mathematics of Computation* 54.189 (Jan. 1990), pp. 107–129. DOI: 10.2307/2008684.

[95] A. A. Khan and W. Lai. *Modeling Shallow Water Flows Using the Discontinuous Galerkin Method*. 1st ed. CRC Press, 2014. ISBN: 978-1-4822-2602-7,978-1-4822-2601-0.

[96] C. Krautwald, H. von Häfen, P. Niebuhr, K. Vögele, D. Schürenkamp, M. Sieder, and N. Goseberg. "Large-scale physical modeling of broken solitary waves impacting elevated coastal structures". In: *Coastal Engineering Journal* (Jan. 2022), pp. 1–21. DOI: 10.1080/21664250.2021.2023380.

[97] C. Krautwald, J. Stolle, J. Hitzegrad, P. Niebuhr, N. Goseberg, I. Nistor, and M. Sieder. "Large-scale physical modelling of a broken solitary wave impact on rigid and non-rigid box-like structures". In: *Coastal Engineering Proceedings* 19.36v (Dec. 2020), structures.19. DOI: 10.9753/icce.v36v.structures.19.

[98] C. Krautwald et al. "Engineering Lessons from September 28, 2018 Indonesian Tsunami: Scouring Mechanisms and Effects on Infrastructure". In: *Journal of Waterway, Port, Coastal and Ocean Engineering* 147.2 (Mar. 2021). Publisher Copyright: © 2020 American Society of Civil Engineers. ISSN: 0733-950X. DOI: 10.1061/(ASCE)WW.1943-5460.0000620.

[99] A. Kurganov and G. Petrova. "A Second-Order Well-Balanced Positivity Preserving Central-Upwind Scheme for the Saint-Venant System". In: *Communications in mathematical sciences* 5 (Mar. 2007). DOI: 10.4310/CMS.2007.v5.n1.a6.

[100] D. Kuzmin. "Positive finite element schemes based on the flux-corrected transport procedure". In: *Computational Fluid and Solid Mechanics, Elsevier* (2001), pp. 887–888. DOI: 10.1016/B978-008043944-0/50795-2.

[101] D. Kuzmin and M. Möller. "Algebraic Flux Correction I. Scalar Conservation Laws". In: *Flux-Corrected Transport: Principles, Algorithms, and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. Chap. 6, pp. 155–206. ISBN: 978-3-540-27206-9. DOI: 10.1007/3-540-27206-2_6.

[102] D. Kuzmin and M. Möller. "Algebraic Flux Correction II. Compressible Euler Equations". In: *Flux-Corrected Transport: Principles, Algorithms, and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. Chap. 7, pp. 207–250. ISBN: 978-3-540-27206-9. DOI: 10.1007/3-540-27206-2_7.

[103] H. P. Langtangen and G. Pedersen. "Computational models for weakly dispersive nonlinear water waves". In: *Computer Methods in Applied Mechanics and Engineering* 160.3 (1998), pp. 337–358. ISSN: 0045-7825. DOI: 10.1016/S0045-7825(98)00293-X.

[104] A. Larese, R. Rossi, E. Oñate, and S. Idelsohn. "Validation of the particle finite element method (PFEM) for simulation of free surface flows". In: *Engineering Computations* 25.4 (May 2008), pp. 385–425. DOI: 10.1108/02644400810874976.

[105] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002. DOI: 10.1017/CBO9780511791253.

[106] M. Leclerc, J.-F. Bellemare, G. Dumas, and G. Dhatt. "A finite element model of estuarian and river flows with moving boundaries". In: *Advances in Water Resources* 13.4 (1990), pp. 158–168. ISSN: 0309-1708.

[107] H. Lee. "Implicit discontinuous Galerkin scheme for shallow water equations". In: *Journal of Mechanical Science and Technology* 33 (July 2019). DOI: 10.1007/s12206-019-0625-2.

[108] Y. Li, S. Liu, Y. Yu, and G. Lai. "Numerical modeling of Boussinesq equations by finite element method". In: *Coastal Engineering* 37.2 (Jan. 1999), pp. 97–122. ISSN: 0378-3839. DOI: 10.1016/S0378-3839(99)00014-9.

[109] I. Lie. "Well-posed transparent boundary conditions for the shallow water equations". In: *Applied Numerical Mathematics* 38.4 (2001), pp. 445–474. ISSN: 0168-9274. DOI: 10.1016/S0168-9274(01)00045-9.

[110] F. Løvholt, G. Pedersen, and G. Gisler. "Oceanic propagation of a potential tsunami from the La Palma Island". In: *Journal of Geophysical Research: Oceans* 113.C9 (2008). DOI: 10.1029/2007JC004603.

[111] R. Löhner. *Applied Computational Fluid Dynamics Techniques. An introdutcion based on Finite Element Methods*. John Wiley & Sons, Ltd, 2008. ISBN: 9780470989746. DOI: 10.1002/9780470989746.

[112] R. Löhner. "Discretization in Time". In: *Applied Computational Fluid Dynamics Techniques*. John Wiley & Sons, Ltd, 2008. Chap. 6, pp. 133–136. ISBN: 9780470989746. DOI: 10.1002/9780470989746.ch6.

[113] R. Löhner. "Edge-Based Compressible Flow Solvers". In: *Applied Computational Fluid Dynamics Techniques*. John Wiley & Sons, Ltd, 2008. Chap. 10, pp. 187–200. ISBN: 9780470989746. DOI: 10.1002/9780470989746.ch10.

[114] R. Löhner. "Flux-Corrected Transport Schemes". In: *Applied Computational Fluid Dynamics Techniques*. John Wiley & Sons, Ltd, 2008. Chap. 9, pp. 175–185. ISBN: 9780470989746. DOI: 10.1002/9780470989746.ch9.

[115] I. MacDonald, M. J. Baines, N. K. Nichols, and P. G. Samuels. "Analytic Benchmark Solutions for Open-Channel Flows". In: *Journal of Hydraulic Engineering* 123.11 (1997), pp. 1041–1045. DOI: 10.1061/(ASCE)0733-9429(1997)123:11(1041).

[116] S. MacNamara and G. Strang. "Operator Splitting". In: *Splitting Methods in Communication, Imaging, Science, and Engineering*. Ed. by R. Glowinski, S. J. Osher, and W. Yin. Cham: Springer International Publishing, 2016, pp. 95–114. ISBN: 978-3-319-41589-5. DOI: 10.1007/978-3-319-41589-5_3.

[117] P. A. Madsen, R. Murray, and O. R. Sørensen. "A new form of the Boussinesq equations with improved linear dispersion characteristics". In: *Coastal Engineering* 15.4 (1991), pp. 371–388. ISSN: 0378-3839. DOI: 10.1016/0378-3839(91)90017-B.

[118] M. Masó, A. Franci, I. de Pouplana, A. Cornejo, and E. Oñate. "A Lagrangian-Eulerian procedure for the coupled solution of the Navier-Stokes and shallow water equations for landslide-generated waves". In: *Advanced Modeling and Simulation in Engineering Sciences* (Mar. 2022). DOI: 10.21203/rs.3.rs-1457837/v1.

[119] M. Masó, I. de Pouplana, A. Franci, A. Cornejo, and E. Oñate. "A one-way coupled Lagrangian-Eulerian procedure for the solution of landslide-generated waves". In: *WCCM*. 2022.

[120] M. Masó, I. de Pouplana, A. Franci, A. Cornejo, and E. Oñate. "Un procedimiento Lagrangiano-Euleriano para la solución acoplada de las ecuaciones de Navier-Stokes y aguas poco profundas para olas generadas por deslizamientos de laderas". In: *WCCM*. 2022.

[121] M. Masó, I. de Pouplana, and E. Oñate. "A FIC-FEM procedure for the shallow water equations over partially wet domains". In: *Computer Methods in Applied Mechanics and Engineering* 389 (Feb. 2022), p. 114362. DOI: 10.1016/j.cma.2021.114362.

[122] M. Masó, I. de Pouplana, and E. Oñate. "Coupling shallow water models with three-dimensional models for the study of fluid-structure interaction problems using the particle finite element method". In: *VI International Congress on Particle-based Methods. Fundamentals and Applications*. Barcelona: CIMNE, 2019.

[123] G. Matthies, P. Skrzypacz, and L. Tobiska. "A unified convergence analysis for local projection stabilisationsapplied to the Oseen problem". In: *Mathematical Modelling and Numerical Analysis* 41 (July 2007), pp. 713–742. DOI: 10.1051/m2an:2007038.

[124] M. de Mier Torrecilla. "Numerical simulation of multi-fluid flows with the Particle Finite Element Method". PhD thesis. Barcelona: Universitat Politècnica de Catalunya. Departament de Resistència de Materials i Estructures a l'Enginyeria, June 2010.

[125] D. Miller. "Giant waves in Lituya Bay Alaska". In: *USGS Prof. Paper* 354-C (1960), pp. 51–83. DOI: 10.3133/pp354C.

[126] R. P. Mulligan and W. A. Take. "On the transfer of momentum from a granular landslide to a water wave". In: *Coastal Engineering* 125 (2017), pp. 16–22. ISSN: 0378-3839. DOI: 10.1016/j.coastaleng.2017.04.001.

[127] R. Mulligan, A. Franci, M. Celigueta, and W. Take. "Simulations of Landslide Wave Generation and Propagation Using the Particle Finite Element Method". In: *Journal of Geophysical Research: Oceans* 125 (June 2020), e2019JC015873. DOI: 10.1029/2019JC015873.

[128] R. Murray. "Short Wave Modelling Using New Equations of Boussinesq Type". In: *In: Ninth Australasian Conference on Coastal and Ocean Engineering, Preprints of Papers.* Institution of Engineers, Australia, 1989, pp. 332–338.

[129] M. Möller, D. Kuzmin, and S. Turek. "Implicit Flux-Corrected Transport Algorithm for Finite Element Simulation of the Compressible Euler Equations". In: *Conjugate Gradient Algorithms and Finite Element Methods*. Ed. by M. Křížek, P. Neittaanmäki, S. Korotov, and R. Glowinski. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 325–354. ISBN: 978-3-642-18560-1. DOI: 10.1007/978-3-642-18560-1_20.

[130] T. Möller. "A Fast Triangle-Triangle Intersection Test". In: *Journal of Graphics Tools* 2.2 (1997), pp. 25–30. DOI: 10.1080/10867651.1997.10487472.

[131] T. Möller and B. Trumbore. "Fast, Minimum Storage Ray-Triangle Intersection". In: *Journal of Graphics Tools* 2.1 (1997), pp. 21–28. DOI: 10.1080/10867651.1997.10487468.

[132] I. M. Navon. "A review of finite-element methods for solving the shallow-water equations". In: *Computer Modelling in Ocean Engineering* (1988), pp. 273 –278.

[133] I. Navon. "Finite-element simulation of the shallow-water equations model on a limited-area domain". In: *Applied Mathematical Modelling* 3.5 (1979), pp. 337 –348. ISSN: 0307-904X. DOI: 10.1016/S0307-904X(79)80040-2.

[134] I. M. Navon, B. Neta, and M. Hussaini. "A Perfectly Matched Layer Approach to the Linearized Shallow Water Equations Models". In: *Monthly Weather Review* 132 (2004), pp. 1369–1378.

[135] E. Noda. "Water waves generated by landslides". In: *Journal of the Waterways, Harbors and Coastal Engineering Division* 96.4 (1970), pp. 835–855. DOI: 10.1061/AWHCAR.0000045.

[136] O. Nwogu. "Alternative Form of Boussinesq Equations for Nearshore Wave Propagation". In: *Journal of Waterway, Port, Coastal, and Ocean Engineering* 119.6 (1993), pp. 618–638. DOI: 10.1061/(ASCE)0733-950X(1993)119:6(618).

[137] P. Ortiz. "Non-oscillatory continuous FEM for transport and shallow water flows". In: *Computer Methods in Applied Mechanics and Engineering* s 223-224 (June 2012), pp. 55–69. DOI: 10.1016/j.cma.2012.02.022.

[138] E. Oñate, S. Idelsohn, O. C. Zienkiewicz, and R. L. Taylor. "A finite point method in computational mechanics. Applications to convective transport and fluid flow". In: *International Journal for Numerical Methods in Engineering* 39.22 (1996), pp. 3839–3866. ISSN: 1097-0207. DOI: 10.1002/(SICI)1097-0207(19961130)39:22<3839::AID-NME27>3.0.CO;2-R.

[139] E. Oñate, S. Idelsohn, F. Del Pin, and R. Aubry. "The particle finite element method. An overview". In: *International Journal of Computational Methods* 2 (2004), pp. 267–307. DOI: 10.1142/S0219876204000204.

[140] E. Oñate, S. Idelsohn, C. M.A., and R. Rossi. "Advances in the particle finite element method for the analysis of fluid-multibody interaction and bed erosion in free surface flows". In: *Computer Methods in Applied Mechanics and Engineering* 197.19-20 (2008), pp. 1777–1800. DOI: 10.1016/j.cma.2007.06.005.

[141] E. Oñate. "Derivation of stabilized equations for numerical solution of advective-diffusive transport and fluid flow problems". In: *Computer Methods in Applied Mechanics and Engineering* 151.1 (1998). Containing papers presented at the Symposium on Advances in Computational Mechanics, pp. 233 –265. ISSN: 0045-7825. DOI: 10.1016/S0045-7825(97)00119-9.

[142] E. Oñate. "Possibilities of finite calculus in computational mechanics". In: *International Journal for Numerical Methods in Engineering* 60.1 (2004), pp. 255–281. DOI: 10.1002/nme.961.

[143] E. Oñate, A. Cornejo, F. Zárate, K. Kashiyama, and A. Franci. "Combination of the finite element method and particle-based methods for predicting the failure of reinforced concrete structures under extreme water forces". In: *Engineering Structures* 251 (2022), p. 113510. ISSN: 0141-0296. DOI: 10.1016/j.engstruct.2021.113510.

[144] D. H. Peregrine. "Long waves on a beach". In: *Journal of Fluid Mechanics* 27.4 (1967), pp. 815–827. DOI: 10.1017/S0022112067002605.

[145] R. Perić and M. Abdel-Maksoud. "Analytical prediction of reflection coefficients for wave absorbing layers in flow simulations of regular free-surface waves". In: *Ocean Engineering* 147 (2018), pp. 132–147. ISSN: 0029-8018. DOI: 10.1016/j.oceaneng.2017.10.009.

[146] W. Pringle, N. Yoneyama, and N. Mori. "Two-way coupled long wave - RANS model: Solitary wave transformation and breaking on a plane beach". In: *Coastal Engineering* 114 (Aug. 2016), pp. 99–118. DOI: 10.1016/j.coastaleng.2016.04.011.

[147] A. Puigferrat, M. Maso, I. De Pouplana, and G. Casas. "Semi-Lagrangian formulation for the advection-diffusion-absorption equation". In: *Computer methods in applied mechanics and engineering* 380 (July 2021), 113807:1–113807:26. DOI: 10.1016/j.cma.2021.113807.

[148] D. Rabinovich, D. Givoli, J. Bielak, and T. Hagstrom. "The double absorbing boundary method for elastodynamics in homogeneous and layered media". In: *Advanced Modeling and Simulation in Engineering Sciences* 2 (Dec. 2015). DOI: 10.1186/s40323-015-0026-8.

[149] P.-A. Raviart and E. Godlewski. *Numerical approximation of hyperbolic systems of conservation laws*. eng. Vol. 118. Applied Mathematical Sciences. New York, NY: Springer, 1996. ISBN: 9780387945293.

[150] N. J. Roberts, R. McKillop, R. L. Hermanns, J. J. Clague, and T. Oppikofer. "Preliminary global catalogue of displacement waves from subaerial landslides". In: *Landslide science for a safer geoenvironment*. Springer, 2014, pp. 687–692. DOI: 10.1007/978-3-319-04996-0_104.

[151] F. Salazar, E. Oñate, and R. Morán. "Numerical Modelling of lanslide flows on reservoirs with the Particle Finite Element Method (PFEM)". In: *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería* 28(2) (2012), pp. 112–123.

[152] H. Samet. "The Quadtree and Related Hierarchical Data Structures". In: *ACM Comput. Surv.* 16.2 (June 1984), pp. 187–260. ISSN: 0360-0300. DOI: 10.1145/356924.356930.

[153] E. Semenza. *Sintesi degli studi geologici sulla frana del Vaiont dal 1959 al 1964*. Vol. 16. Memorie del Museo Tridentino di Scienze Naturali, A XXIX-XXX, 1965, pp. 1–51.

[154] J. Smallman and A. Cooper. "A mathematical model for set down in harbours". In: *Coastal Engineering* 13.3 (1989), pp. 247–261. ISSN: 0378-3839. DOI: 10.1016/0378-3839(89)90051-3.

[155] S. Soares-Frazão and Y. Zech. "Experimental study of dam-break flow against an isolated obstacle". In: *Journal of Hydraulic Research* 45.sup1 (2007), pp. 27–36. DOI: 10.1080/00221686.2007.9521830.

[156] P. K. Stansby. "Solitary wave run up and overtopping by a semi-implicit finite-volume shallow-water Boussinesq model". In: *Journal of Hydraulic Research* 41.6 (2003), pp. 639–647. DOI: 10.1080/00221680309506896.

[157] F. Ursell. "The long-wave paradox in the theory of gravity waves". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 49.4 (1953), pp. 685–694. DOI: 10.1017/S0305004100028887.

[158] M. Walkley and M. Berzins. "A finite element method for the two-dimensional extended Boussinesq equations". In: *International Journal for Numerical Methods in Fluids* 29 (Jan. 2002), p. 143. DOI: 10.1002/fld.349.

[159] C. F. Waythomas and P. Watts. "Numerical simulation of tsunami generation by pryoclastic flow at Aniakchak Volcano, Alaska". In: *Geophysical research letters* 30.14 (2003).

[160] G. Wei and J. T. Kirby. "Time-Dependent Numerical Code for Extended Boussinesq Equations". In: *Journal of Waterway, Port, Coastal, and Ocean Engineering* 121.5 (1995), pp. 251–261. DOI: 10.1061/(ASCE)0733-950X(1995)121:5(251).

[161] S.-B. Woo and P. L.-F. Liu. "Finite-Element Model for Modified Boussinesq Equations. I: Model Development". In: *Journal of Waterway, Port, Coastal, and Ocean Engineering* 130.1 (2004), pp. 1–16. DOI: 10.1061/(ASCE)0733-950X(2004)130:1(1).

[162] S.-B. Woo and P. L.-F. Liu. "Finite-Element Model for Modified Boussinesq Equations. II: Applications to Nonlinear Harbor Oscillations". In: *Journal of Waterway, Port, Coastal, and Ocean Engineering* 130.1 (2004), pp. 17–28. DOI: 10.1061/(ASCE)0733-950X(2004)130:1(17).

[163] H. Wu and Z. Zhang. "Can We Have Superconvergent Gradient Recovery Under Adaptive Meshes?" In: *SIAM Journal on Numerical Analysis* 45.4 (2007), pp. 1701–1722. DOI: 10.1137/060661430.

[164] Y.-T. Wu and S.-C. Hsiao. "Generation of stable and accurate solitary waves in a viscous numerical wave tank". In: *Ocean Engineering* 167 (Nov. 2018), pp. 102–113. DOI: 10.1016/j.oceaneng.2018.08.043.

[165] W. Zhang, Z.-h. Zhong, C. Peng, W.-h. Yuan, and W. Wu. "GPU-accelerated smoothed particle finite element method for large deformation analysis in geomechanics". In: *Computers and Geotechnics* 129 (2021), p. 103856. ISSN: 0266-352X. DOI: 10.1016/j.compgeo.2020.103856.

[166] X. Zhang, K. Krabbenhoft, D. Pedroso, A. Lyamin, D. Sheng, M. V. da Silva, and D. Wang. "Particle Finite element analysis of large deformation and granular flow problems". In: *Computer and Geotechnics* 54 (2013), pp. 133–142. DOI: 10.1016/j.compgeo.2013.07.001.

[167] X. Zhang, K. Krabbenhoft, D. Sheng, and W. Li. "Numerical simulation of a flow-like landslide using the particle finite element method". In: *Computational Mechanics* 55 (2015), pp. 167–177. DOI: 10.1007/s00466-014-1088-z.

[168] X. Zhang, E. Oñate, S. Galindo, J. Bleyer, and K. Krabbenhoft. "A unified Lagrangian formulation for solid and fluid dynamics and its possibility for modelling submarine landslides and their consequences". In: *Computer Methods in Applied Mechanics and Engineering* 343 (2019), pp. 314–338. DOI: 10.1016/j.cma.2018.07.043.

[169] Z. Zhang and A. Naga. "A New Finite Element Gradient Recovery Method: Superconvergence Property". In: *SIAM J. Scientific Computing* 26 (Jan. 2005), pp. 1192–1213. DOI: 10.1137/S1064827503402837.

[170] M. Zhu, I. Elkhetali, and M. H. Scott. "Validation of OpenSees for Tsunami Loading on Bridge Superstructures". In: *Journal of Bridge Engineering* 23.4 (2018), p. 04018015. DOI: 10.1061/(ASCE)BE.1943-5592.0001221.

[171] O. C. Zienkiewicz and R. L. Taylor. *Finite Element Method: Fluid dynamics*. 5th. Vol. 3. Butterworth-Heinemann, 2000. ISBN: 978 0 7506 5050 2,0 7506 5050 8.

[172] O. C. Zienkiewicz and R. L. Taylor. *Finite Element Method: The basis*. 5th. Vol. 1. Butterworth-Heinemann, 2000. ISBN: 978 0 7506 6320 5,0 7506 6320 0.

[173] O. Zienkiewicz and J. Zhu. "The superconvergent patch recovery (SPR) and adaptive finite element refinement". In: *Computer Methods in Applied Mechanics and Engineering* 101.1 (1992), pp. 207–224. ISSN: 0045-7825. DOI: 10.1016/0045-7825(92)90023-D.