

WAVE GENERATION, PROPAGATION AND ABSORPTION IN CFD SIMULATIONS OF FREE SURFACE FLOWS

Bülent Düz

WAVE GENERATION, PROPAGATION AND ABSORPTION IN CFD SIMULATIONS OF FREE SURFACE FLOWS



Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 21 september 2015 om 15:00 uur

door

Bülent Düz

Ingenieur Werktuigbouwkunde
Boğaziçi University, Turkije
geboren te İspir, Turkije

Dit proefschrift is goedgekeurd door de

promotors: Prof. dr. ir. R.H.M. Huijsmans and Prof. dr. A.E.P. Veldman

Samenstelling promotiecommissie:

Rector Magnificus

Prof. dr. ir. R.H.M. Huijsmans

Prof. dr. A.E.P. Veldman

voorzitter

Technische Universiteit Delft

Rijksuniversiteit Groningen

Onafhankelijke leden:

Prof. dr. M. Visonneau

Prof. dr. ir. G.S. Stelling

Prof. dr. ir. E.W.C. van Groesen

Prof. dr. ir. H.W.M. Hoeijmakers

Dr. ir. P.R. Wellens

Prof. dr. ir. M. Stive

École Centrale de Nantes, Frankrijk

Technische Universiteit Delft

Universiteit Twente

Universiteit Twente

Deltares

Technische Universiteit Delft, reservelid

The research is supported by the Dutch Technology Foundation STW, applied science division of NWO and the technology programme of the Ministry of Economic Affairs in The Netherlands (contracts GWI.6433 and 10475).

ISBN 978-94-6169-748-6

Copyright © 2015 by Bülent Düz.

Cover painting: "The Surge" by Bissan Rafe Düz, Copyright © 2015,
www.nohra-studio.com



to my parents, İbrahim and Güllü Düz

CONTENTS

1	Introduction	1
1.1	Numerical computation	2
1.2	Experimentation	3
1.3	The story and the scope of this research.	4
1.4	Outline	6
	References	7
2	Mathematical and numerical modeling	9
2.1	Mathematical model	9
2.1.1	Boundary conditions.	12
2.1.2	Modeling moving bodies.	13
2.2	Numerical model	14
2.2.1	Computational domain and grid.	14
2.2.2	Object and fluid configuration	15
2.2.3	Cell labeling	17
2.2.4	Spatial discretization of the governing equations	17
2.2.5	Temporal discretization and solution technique	24
2.2.6	Stability	25
	References	25
3	Absorbing Boundary Condition (ABC)	29
3.1	Introduction	29
3.2	Motivation	35
3.3	Wave generation	37
3.3.1	Regular waves	37
3.3.2	Irregular waves.	37
3.4	Design of the ABC.	40
3.4.1	Statement of the problem	40
3.4.2	ABC-1 (Dispersive ABC)	41
3.4.3	ABC-2 (Dispersive Directional ABC)	42
3.4.4	Reflection coefficients of the absorbing boundary conditions	43
3.4.5	Stability	45
3.4.6	From ABC to GABC (Generating and Absorbing BC)	47
3.4.7	Numerical discretization of the ABC-1 and ABC-2	48
3.5	Results and discussions	53
3.5.1	Dispersive properties of the ABCs	53
3.5.2	Directional properties of the ABCs	55
3.5.3	Short-crested wave test	56
3.5.4	Circular concentric waves radiating from a localized source	61

3.6	Concluding remarks	65
	References	68
4	Volume of fluid (VOF) method	73
4.1	Introduction	73
4.2	Interface reconstruction	77
4.2.1	Parker and Youngs' method	78
4.2.2	Least-square method by Rider and Kothe	80
4.2.3	Mixed Youngs-centered (MYC) method	81
4.2.4	Efficient least squares interface reconstruction algorithm (ELVIRA)	83
4.2.5	Computation of the plane constant	85
4.3	Interface advection	86
4.4	Numerical results	92
4.4.1	Rotation of a slotted disk - 2D	93
4.4.2	Single vortex test - 2D	95
4.4.3	Rotation of a slotted sphere - 3D	99
4.4.4	Deformation field - 3D	101
4.4.5	Shearing flow - 3D	104
4.4.6	Application example: Propagating Rienecker-Fenton waves	106
4.5	Concluding remarks	110
	References	112
5	Validation	119
5.1	Semi-submersible test	119
5.1.1	Experimental modeling	120
5.1.2	Numerical simulation	122
5.1.3	Results	124
5.2	Breaking dam test	131
5.2.1	Experimental modeling	131
5.2.2	Numerical simulation	139
5.2.3	Results	140
	References	144
6	Summary and conclusions	145
6.1	Free surface modeling	145
6.2	Generation and absorption of free surface water waves	147
6.3	Epilogue	148
	References	149
A	Formulae for plane constant and cut volume	151
A.1	Forward Problem	153
A.2	Inverse Problem	154
	References	156
B	Computation of flux volume	157
	Summary	161
	References	162

Samenvatting	163
Acknowledgements	165
Curriculum Vitæ	167



1

INTRODUCTION

THE oceans occupy more than 70 percent of the Earth's surface area and have a dominant influence on the global climate and on its plant, animal and human life. From scientists and engineers' point of view, the oceans are sometimes interesting not only because they provide food, water and means of transportation, but especially because of one of the broadest physical phenomena which has been extensively studied in various fields of science: waves. Although waves can occur in all types amongst a variety depending on the forces acting on the water, one aspect exists in all the cases and attracts our attention: waves exert loads and stresses on numerous kinds of man-made structures in the oceans. As these structures spend their lifetime in the oceans, their design and operation require a thorough understanding of the physical character and behavior of the ocean environment and the atmosphere above it. Throughout history many human experiences have shown that the journey towards this goal is by no means complete. When the environmental conditions are harsh, such as during hurricanes and tsunamis, major catastrophic failures arise threatening human safety and causing economic loss.

In the last couple of decades, man's utilization of the oceans as a resource has stimulated developments in many areas, and ocean hydrodynamics has seen an explosion of interest. This interest led to an improved and more realistic understanding of the physical characteristics of fluid flows about offshore structures, their mathematical formulation and experimental exploration. Mathematical and numerical methods have enjoyed a portion of this interest, and advances in these fields have allowed for the nearly exact solutions of some wave-loading situations. At the same time, a part of this attention has also been focused on controlled laboratory experiments, which allow for the understanding of the separate effects of the governing and influencing parameters. Overall, advances along these paths supported a better physical understanding of the mechanics of the environment, aerodynamic and hydrodynamic loadings on structures, their consequent static and dynamic response, the material behavior of the structure and a multitude of other aspects. However, there is still much to accomplish. In order to continue carrying out activities in the oceans in a safe, cost-effective and environment-

friendly manner, the design, construction, installation, operation and maintenance of offshore structures need further investigations and improvements.

1.1. NUMERICAL COMPUTATION

Real fluid flows in nature are sometimes too complex to be addressed solely by analytical theories or experimentation. These flows are often highly nonlinear, turbulent and three-dimensional, involve complex geometries, and manifest coupled behavior, which suggests that the interaction between two or more disciplines of science should be taken into account, such as fluid-structure interactions and chemically reacting flows. In some situations, limitations of theory, experimentation or both become overwhelmingly challenging, and the need for numerical computation arises. In other cases, the problem at hand is so complex involving many intricate yet essential details, it becomes equally difficult to simulate with a numerical computation tool.

Throughout the last couple of decades computational fluid dynamics (CFD) has gained more and more popularity and wide acceptance among scientific and engineering communities in the design and evaluation of the performance of products and processes. Its unique advantages over traditional engineering methods allow researchers to quickly and inexpensively explore different design options or to investigate how a system can be constructed and operated more efficiently. The interest in CFD is predicted to go nowhere but up especially with availability of affordable high-performance computing hardware, further reduction of computational costs through more efficient use of computing resources such as smart grid generation techniques and parallelization, and more robust, accurate and efficient numerical algorithms. Although CFD has been enjoying such popularity, it has never been considered as a utility to replace other methods, but rather a valuable complement to them. A CFD code is only as good as the mathematical model it employs to describe a physical phenomenon that actually takes place. In that regard, advances in mathematical models support advances in CFD. Moreover, it is essential to remember that a model (such as a code and conceptual modeling assumptions) should be verified and the predicted results should be validated. Through verification we make sure that the equations are solved right by checking convergence with mesh and time step refinement and making sure that numerical errors are sufficiently small. Through validation we make sure that the right equations are solved by comparing the predicted results with measured data and/or by comparing with benchmark analytical and/or numerical solutions [1]. These concepts are clear indications of the close relation between theory, numerics and experimentation.

Numerical computation of wave propagation and wave-structure interaction is a complex endeavor. Since the physical mechanisms that describe these phenomena cover a wide range of topics in ocean hydrodynamics, simulation tools that numerically model these phenomena can understandably become rather compound programs. For a long time and even today, potential theory has been a popular approach to develop basic mathematical models, and numerical tools benefited from these models to a great extent. If the phenomenon in question is, however, local and nonlinear such as wave breaking, then this theory becomes inadequate and the need for a more advanced approach arises. Solving the full Navier-Stokes equations using a numerical model is always a possibility, but an expensive one even today. This sometimes prompts researchers

to combine both approaches in a model that employs a domain decomposition type of strategy. In this case, the Navier-Stokes equations are solved in a compact domain surrounding the area of interest, while potential theory is utilized in the rest of the domain.

1.2. EXPERIMENTATION

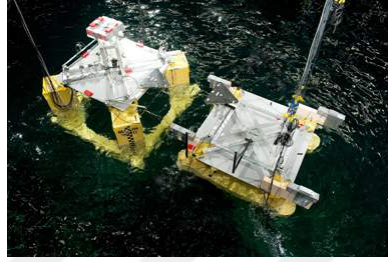
Experiments on model scales are a common practice in the development and validation of large scale offshore vessels and structures in oil, gas, military and marine industries. It is easy to understand the reason behind this since working with a scaled model is generally easier, faster, safer and cheaper compared to full scale tests. Moreover, measurements on model scale provide real and accurate data for model validation and can also be used for diagnostic purposes to determine the cause of a particular problem existing in real life. Simply put, the main objective of a model test is to obtain reliable results that can be scaled up and correlated with analytical design tools. In a typical model test of a structure, a variety of phenomena may be investigated, such as wave impact pressure on the structure, wave run-up around the structure and the dynamic response and motions of the structure. The structure in question may be kept at a fixed location in a test section of the basin (see Fig. 1.1a for an example), or may be allowed to float freely according to water flow (see Fig. 1.1b for an example), or may be towed along the basin using a towing mechanism such as a crane rail system (see Fig. 1.1c for an example).

Waves are produced using a wave generator installed at one or more than one end of the basin. Most modern wave basins are equipped with one of two types of wave generator: flap or piston. Flap paddles are used to generate deep water waves. The motion at the free end of the flap is the largest while the motion at the hinge of the paddle is zero. This resembles the orbital motion of the fluid particles in deep water where the particles move in circles with the radius decreasing exponentially with respect to the distance to the free surface, see Fig. 1.2a for an illustration of this type of wave generators. The motion of the particles at the bottom is negligible which is analogous to the zero motion of the flap at the hinge. Piston type wave makers are, however, used to produce shallow water waves. Here the water particles move in ellipses, which become flatter towards the bottom. Since the horizontal particle motion is almost constant at all depths, piston type wave maker proves to be a convenient utility to generate such a motion as shown in Fig. 1.2b. Sometimes it may be necessary to incorporate both piston-derived and flap-derived components to generate more complex seas and to be more effective than either method alone over a wide range of depths. In addition to waves, modern experimental facilities provide a number of features at the same time, such as wave, wind and current generation, which allows researchers to investigate the interactions between structures and rather complex environmental conditions.

Although model scale testing of a new concept is extremely worthwhile, it has various limitations and challenges. To carry out such a test, a specialized experimental facility is often needed, which is expensive and sometimes not available. To give an example, for model testing of a structure in ultra deep water (a water depth that is equal to or greater than 1,500 meters), an extremely deep water facility is required. Even if such a facility is found, aggressive scaling becomes inevitable which has a number of issues in its backpack. One of the issues lies in scaling itself, which sometimes causes some distortion in the important parameters of a model. This situation naturally brings



(a) A semi-submersible model whose position is fixed at the test section of the wave basin.



(b) A Tension Leg Wellhead Platform (TLWP) and a semi-submersible floating during a test.



(c) A ship being towed using a crane rail system along the basin.

Figure 1.1: Three examples where the motion of the model is controlled in three different ways (courtesy of the Maritime Research Institute in the Netherlands - MARIN).

up another issue related to the effects of the Reynolds number. The Reynolds number for a model is sometimes orders of magnitude smaller than that for a structure. This necessitates proper investigation of the scaling effects and some corrective measures in the model tests and scaled data [2]. In addition to this, there are other limitations and difficulties in model testing, and a comprehensive discussion on this topic is outside the scope of this introduction. For such a discussion, the reader is referred to the related literature which is abounding, to say the least.

1.3. THE STORY AND THE SCOPE OF THIS RESEARCH

This work presents a four-year doctorate study on the CFD program ComFLOW. The story of ComFLOW started in the late nineties when the Maritime Research Institute in the Netherlands (MARIN) learned of the micro-gravity applications at the University of

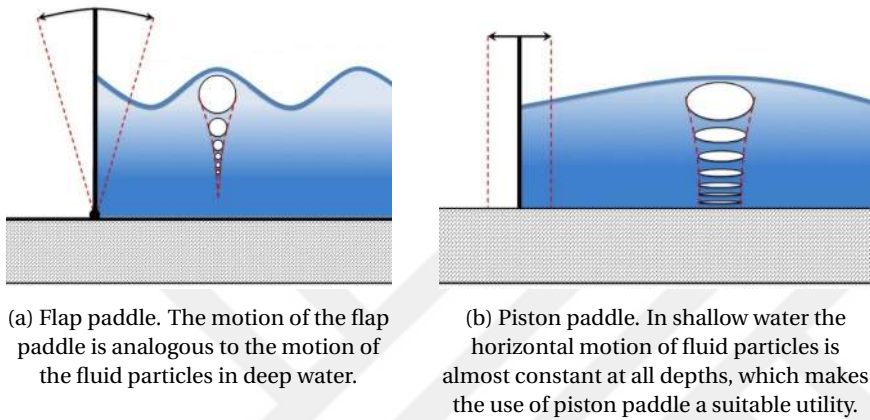


Figure 1.2: Two types of wave makers that are often installed in experimental facilities.

Groningen (RUG) with their dynamic free-surface motion. The idea came up to test this approach for violent wave motion at sea. The cooperation between RUG and MARIN resulted in an MSc project in 1998, which was later continued as a MARIN-funded PhD project. In 2000 the EU-funded SAFE-FLOW JIP (Joint Industry Project) on hydrodynamic wave loading was started with the cooperation from a world-wide consortium of offshore-related companies. The project was aimed at identifying major physical phenomena related to extreme wave events and to draw conclusions regarding potentially successful (numerical) methods of analyses. As a follow-up, in 2004 the ComFLOW-2 JIP was born; it was partly funded by the Dutch Technology Foundation (STW). In this project the functionality of the simulation method was extended. The interaction between water and air has been included in the ComFLOW program: two-phase flow. A new type of generating and absorbing boundary conditions (GABC) has been developed for the inflow and outflow boundaries. These are fully transparent conditions allowing waves to enter and leave the domain over the same boundary at the same time. In addition, an interactive coupling between fluid forces, floater motions and mooring systems has been implemented. After successfully finishing this project, the ComFLOW-2 JIP partners started the ComFLOW-3 JIP with the following main objective: *"To further improve, develop and validate the ComFLOW program for complex free-surface flows in the offshore industry and make it useable for advanced engineering applications by improved functionality and speed-up of the algorithms."* One post-doctorate and three doctorate researchers were employed to work on four different sets of tasks. The post-doctorate researcher along with two of the doctorate researchers were based at RUG, while the last doctorate researcher, the author of this thesis, started his studies at the Delft University of Technology (TUDelft).

In the progress meetings towards the end of the ComFLOW-2 JIP a number of topics were discussed and identified where major improvements were highly desirable. The current work focuses on two of these topics, and a separate chapter is dedicated to each one. The first topic was energy dissipation in wave propagation. The wave dissipation can be caused by physical and/or numerical phenomena, such as the free surface treat-

ment, physical viscosity and artificial viscosity due to discretization of the momentum equations. Among these the free surface treatment is an important factor, and its effect on wave dissipation was studied extensively. The Volume-of-Fluid (VOF) method, which is responsible for displacement of the free surface, was improved to a great extent, and a significant reduction in wave dissipation is obtained. The second topic of this work was to improve the functionality and the accuracy of the GABC. At the end of the ComFLOW-2 project the method was working well for long-crested waves propagating perpendicular to the boundary. After this work, the GABC is further improved to absorb oblique long-crested waves as well as short-crested waves. This is achieved through the implementation of a higher-order boundary condition which account for both dispersive and directional effects of free surface water waves.

1.4. OUTLINE

The thesis is comprised of six chapters. Chapter 2 presents an overview of the mathematical and numerical model that was implemented in the CFD code. This includes spatial and temporal discretization of the governing equations, various boundary conditions and modeling moving bodies. In addition more particular information regarding how objects, fluids and the grid are configured in the computational domain is presented. Stability of the overall numerical method is also investigated. All this information is given in a concise manner for the sake of completeness since the implemented techniques are well-known and well-documented. Chapter 3 examines the problem of absorbing boundary conditions (ABCs), which is the other main focus of this work. After presenting a summary of the related works on this topic, the chapter develops an absorbing boundary condition which possesses much desired characteristics such as being transparent allowing waves to enter and leave the computational domain at the same time, and accounting for dispersive and directional effects of the free surface water waves. These strong suites of the boundary condition are established through the numerical tests collected at the end of the chapter. Chapter 4 offers a lengthy analysis on the volume-of-fluid (VOF) method starting with an overview of the works in this field throughout the last couple of decades. As this is one of the main subjects of the thesis, the VOF concept is examined in a detailed fashion containing reviews of the available VOF methods and choosing the most suitable one for implementation in ComFLOW. This chapter concludes with many numerical examples in order to demonstrate the features of the implemented method. Chapter 5 presents results from two experiments that we have found useful for demonstrating the strengths and weaknesses of the CFD simulation tool, and are also closely related to the practical situations in nature. The first test is the problem of wave impacts on a semi-submersible platform in deep water. Simulation results are compared to the measurements in terms of free surface elevation and wave impact pressure at a number of significant positions. The second experiment is the breaking dam, which resembles green water flow on the deck of a ship. The rather violent flow in this test is especially useful to show the accuracy and the robustness of the overall numerical code as well as the VOF method. Finally, Chapter 6 provides conclusions and ideas for future research based on the results and discussions from this research.

REFERENCES

- [1] P. Roache, *Quantification of uncertainty in computational fluid dynamics*, Annu. Rev. Fluid. Mech. **29**, 123 (1997).
- [2] S. Chakrabarti, *Physical model testing of floating offshore structures*, in *The Dynamic Positioning Conference, 13-14 October 1998, Houston (USA)* (1998).

2

MATHEMATICAL AND NUMERICAL MODELING

2.1. MATHEMATICAL MODEL

CONSERVATION laws of physics can be used to describe fluid flow in a given domain if the fluid is treated as a homogeneous continuum. These laws are conservation of mass, conservation of momentum based on Newton's second law, and conservation of energy stemming from the first law of thermodynamics. Derivation of these laws is well-established and well-documented, as they have been known for hundreds of years. For the sake of brevity, only the final forms of these laws will be stated here.

Conservation of mass

In an arbitrary and fixed control volume Ω with boundary Γ , conservation of mass can be stated in an integral form as the following

$$\int_{\Omega} \frac{\partial \rho}{\partial t} d\Omega = - \int_{\Gamma} \rho \mathbf{u} \cdot \mathbf{n} d\Gamma \quad (2.1)$$

where ρ is the fluid density, $\mathbf{u} = (u, v, w)^T$ is the flow velocity, and $\mathbf{n}$ is the normal vector pointing outward along the boundary of the control volume. This equation is also widely known as the continuity equation. The term on the left hand side of (2.1) denotes the time rate of increase of mass in Ω whereas the term on the right hand side denotes the net inflow of mass through the boundary of the control volume. As the normal is defined as pointing out, the sign in front of the surface integral is negative, which actually denotes an inflow.

Using Gauss' divergence theorem, the above surface integral can be written as a volume integral. Afterwards, a differential form of (2.1) can be obtained. Essentially different forms of a conservation law are equivalent to each other, but they become more relevant when the equations are numerically discretized while developing a numerical method to solve them. Here, we will not derive nor discuss various forms of the conservation laws, and express them only in the integral form. This form is more convenient to

the manner we discretize the governing equations: the finite volume formulation. For more information on various forms, see [1].

Conservation of momentum

Based on another fundamental physical principle, Newton's second law, conservation of momentum for fluid flow can be written as

$$\int_{\Omega} \frac{\partial \rho \mathbf{u}}{\partial t} d\Omega = - \int_{\Gamma} \rho \mathbf{u} \mathbf{u} \cdot \mathbf{n} d\Gamma + \int_{\Gamma} \sigma \cdot \mathbf{n} d\Gamma + \int_{\Omega} \rho \mathbf{f} d\Omega \quad (2.2)$$

where σ represents the total stress tensor and $\mathbf{f}$ represents the external body forces acting on the fluid such as gravity, centrifugal, Coriolis and electromagnetic forces. In Eq. (2.2), the term on the left hand side indicates the time rate of increase of momentum in Ω . This increase can be due to three reasons given on the right hand side, respectively: The net inflow of momentum through the boundary Γ , contribution from the surface forces acting on the surface of the control volume, and contribution from the body forces which act directly on the fluid in Ω . The total stress tensor σ can be written as

$$\sigma = -p\mathbf{I} + \tau \quad (2.3)$$

where p is the pressure, $\mathbf{I}$ is the unit tensor, and τ is the viscous stress tensor. Using the definition of the rate of deformation tensor $\mathbf{D}$,

$$\mathbf{D} = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T), \quad (2.4)$$

the viscous stress tensor τ may be expressed by

$$\tau = \lambda (\nabla \cdot \mathbf{u}) \mathbf{I} + 2\mu \mathbf{D} \quad (2.5)$$

where μ is the first (dynamic) viscosity, and λ is the second viscosity coefficient. In the above relations $\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)^T$ is the gradient operator. The commonly-used expression for λ was introduced by Stokes as a hypothesis

$$\lambda = -\frac{2}{3}\mu. \quad (2.6)$$

Although the Stokes' hypothesis is used in many fluid problems, it has not been verified nor refuted with measurements [2].

Conservation of energy

Exploiting the first law of thermodynamics, the conservation law of energy can be stated as

$$\int_{\Omega} \frac{\partial \rho E}{\partial t} d\Omega = - \int_{\Gamma} E \rho \mathbf{u} \cdot \mathbf{n} d\Gamma + \int_{\Gamma} (\sigma \cdot \mathbf{u}) \cdot \mathbf{n} d\Gamma + \int_{\Omega} \rho \mathbf{f} \cdot \mathbf{u} d\Omega - \int_{\Gamma} \mathbf{q}' \cdot \mathbf{n} d\Gamma + \int_{\Omega} s d\Omega \quad (2.7)$$

where E is the total energy of the fluid, which is the internal energy e plus the kinetic energy $E = e + \mathbf{u} \cdot \mathbf{u}/2$, $\mathbf{q}'$ denotes the heat flux, and s denotes a volumetric source of heat which can originate from, for example, chemical reactions. In Eq. (2.7) the term on the

left hand side indicates the time rate of change of the total energy within the fluid. According to the first law of thermodynamics, this must be balanced by the contributions given on the right hand side, respectively: the net inflow of the total energy through the boundary Γ , the rate of work done on the fluid due to surface forces, the rate of work done on the fluid due to body forces, the net heat flux through the boundary, and the net volumetric source of heat within the fluid. This equation is written in a general form for a moving fluid. Additionally, we also need constitutive relations to close the equation. For example, the heat flux $\mathbf{q}'$ can be related to temperature gradients through Fourier's law.

The above conservation laws for mass, momentum and energy closed by a number of thermodynamic relations are extremely difficult to model as a whole system of equations in a fluid flow. Usually, these equations are simplified through a number of assumptions that are inherent to the fluid flow considered in a study. In what follows we will explain the assumptions made in the present work.

Assumptions

- The numerical model explained here is based on single-phase flows, therefore there is only one fluid in the domain: water. Water is assumed to be a homogeneous, incompressible material. Thus, $\rho = \text{constant}$ everywhere all the time. This consequently implies that $\partial\rho/\partial t = 0$ and $\nabla\rho = 0$. (Having noted that, it is also possible to perform two-phase flow simulations in ComFLOW. This feature has been available after the PhD study by [Wemmenhove](#) [3]. In that work, water is modeled as an incompressible viscous fluid, and a gas phase, such as the air, is modeled as a compressible viscous fluid. For comprehensive information on numerical modeling of two-phase flows, see the aforementioned study.)
- As the external body force only gravity is considered, thus $\mathbf{f} = \mathbf{g} = (g_x, g_y, g_z)^T$.
- Temperature variation and any subsequent results of temperature variation are ignored.

When these assumptions are taken into account, the conservation laws for mass and momentum degenerate to, respectively,

$$\int_{\Gamma} \mathbf{u} \cdot \mathbf{n} \, d\Gamma = 0, \quad (2.8)$$

$$\int_{\Omega} \frac{\partial \mathbf{u}}{\partial t} \, d\Omega + \int_{\Gamma} \mathbf{u} \mathbf{u} \cdot \mathbf{n} \, d\Gamma = -\frac{1}{\rho} \int_{\Gamma} p \, \mathbf{n} \, d\Gamma + \nu \int_{\Gamma} \nabla \mathbf{u} \cdot \mathbf{n} \, d\Gamma + \int_{\Omega} \mathbf{f} \, d\Omega, \quad (2.9)$$

where ν is the kinematic viscosity, $\nu = \mu/\rho$. As the incompressibility condition effectively replaces the conservation law of energy, we are left with the two conservation laws to model in our numerical method. Although Eqs. (2.8) and (2.9) are much simpler compared to the initial forms given before, a meticulous study is required to discretize each term in them. This will be the main subject of Section 2.2. Now, we will explain the boundary conditions imposed on various boundaries that may be encountered in a domain.

2.1.1. BOUNDARY CONDITIONS

Application of physical and initial boundary conditions to solve the governing equations in a computational domain is a critical part of a numerical method. These conditions allow us to obtain particular solutions from the governing equations. Therefore, employing a proper set of boundary conditions for a specific fluid flow makes all the difference for making the solution unique, stable and accurate. This issue, in fact, is the subject of many studies in CFD. Later in this work, we will discuss and expand more on this matter. Here, a set of boundary conditions is presented which will be commonly-used in the numerical experiments throughout this work. These conditions are classified into subcategories according to the boundary where they are applied.

Solid boundary:

At fixed solid boundaries such as a structure, wall or bottom of a sea,

$$\mathbf{u} = 0 \quad (2.10)$$

is applied, which is also known as the Dirichlet condition. This condition states that the solid boundary is impenetrable, and the fluid does not slip along the boundary. Thus, both normal and tangential components of the velocity vector become zero. Sometimes, this condition is also referred to as the no-slip no-penetration condition. If the solid boundary in question is moving with the velocity $\mathbf{u}_b$, then the above condition is replaced by $\mathbf{u} = \mathbf{u}_b$.

Free surface:

In most of the numerical experiments encountered in this research, there exists a separation layer between water and air. This is called the free surface or interface. Since air is modeled as void, we prefer the term free surface rather than interface. At the free surface, the normal and tangential forces are balanced. This is imposed through the following conditions

$$\mu \left(\frac{\partial u_n}{\partial t} + \frac{\partial u_t}{\partial n} \right) = 0, \quad (2.11)$$

$$-p + 2\mu \frac{\partial u_n}{\partial n} = -p_0 + \sigma \kappa, \quad (2.12)$$

where u_n and u_t correspond to the normal and tangential component of the velocity, respectively, p_0 is the atmospheric pressure, and κ is the total curvature of the free surface.

Furthermore, evolution of the free surface in time with respect to the underlying velocity field must be modeled as well. This is performed using the volume-of-fluid (VOF) method, which is explained extensively in Chapter 4.

Inflow and outflow boundaries:

As the names suggest, inflow and outflow boundaries allow the fluid to move into and out of the computational domain. The conditions applied at these boundaries are very critical such that comprehensive studies have been dedicated to them. Below, we will briefly explain how these boundaries are treated in this work.

At the inflow boundary we usually have the information we need. If, for example, we are sending a wave into the domain, we can easily obtain velocity and pressure values at the inflow boundary corresponding to the considered wave model. The outflow boundary, however, causes various bottlenecks. Some of these bottlenecks and possible remedies are discussed by Sani and Gresho [4]. Corresponding to the problem at hand, the outflow condition can become very simple.

In our numerical experiments, the difficulty associated with the implementation of the outflow condition increases substantially as the condition is expected to satisfy rather stringent requirements. Well-posedness, accuracy on the discrete level, stability, ease of implementation and efficiency can be listed under this category. This is the subject of much current CFD research, and a separate chapter in this work is dedicated to this matter.

2.1.2. MODELING MOVING BODIES

In order to introduce moving bodies into the computational domain, we need to reexamine and reformulate the governing equations. Consider an arbitrary control volume Ω with boundary $\partial\Omega$ that includes an object (gray color) moving with the velocity $\mathbf{u}_b$ and a fluid (blue color) such as shown in Fig. 2.1. The part of the domain occupied by the

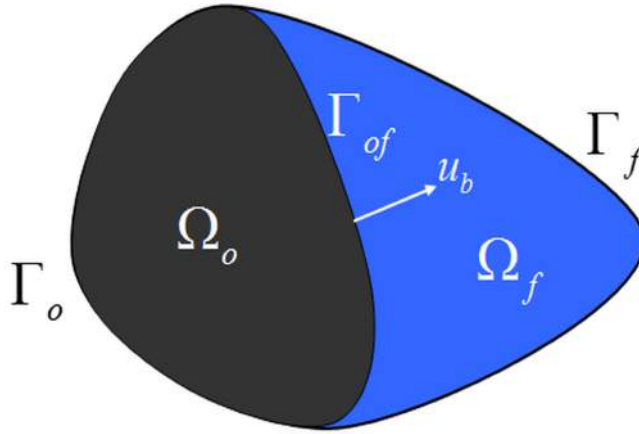


Figure 2.1: An object, illustrated in gray color, moving with the velocity $\mathbf{u}_b$ in an arbitrary control volume which also contains a fluid shown with blue color. Ω_o represents the part occupied by the object, and Ω_f by the fluid. The boundary of the control volume is divided into two parts: the object part Γ_o and the fluid part Γ_f . The boundary between the object and the fluid is denoted by Γ_{of} .

object is given by Ω_o and the remaining part is filled with the fluid and given by Ω_f . Similarly, the boundary of the control volume is also divided into two parts: the object part defined by $\Gamma_o = \partial\Omega_o \cap \partial\Omega$, and the fluid part defined by $\Gamma_f = \partial\Omega_f \cap \partial\Omega$. The boundary between the object and the fluid is then defined by $\Gamma_{of} = \partial\Omega_o \cap \partial\Omega_f$.

When the governing equations (2.8) and (2.9) are considered with this configuration,

it is possible to write them, respectively, as

$$\int_{\partial\Omega_f} \mathbf{u} \cdot \mathbf{n} d\partial\Omega_f = \int_{\Gamma_f} \mathbf{u} \cdot \mathbf{n}_f d\Gamma_f + \int_{\Gamma_{of}} \mathbf{u}_b \cdot \mathbf{n}_f d\Gamma_{of}, \quad (2.13)$$

$$\int_{\Omega_f} \frac{\partial \mathbf{u}}{\partial t} d\Omega_f + \int_{\partial\Omega_f} \mathbf{u} \mathbf{u} \cdot \mathbf{n}_f d\partial\Omega_f = -\frac{1}{\rho} \int_{\partial\Omega_f} p \mathbf{n}_f d\partial\Omega_f + \nu \int_{\partial\Omega_f} \nabla \mathbf{u} \cdot \mathbf{n}_f d\partial\Omega_f + \int_{\Omega_f} \mathbf{f} d\Omega_f \quad (2.14)$$

where $\mathbf{n}_f$ denotes the normal vector of the fluid part in the control volume.

Apart from deriving the governing equations, modeling fluid flow in combination with moving body requires a concerted effort when developing various algorithms of a CFD program. This rather sensitive subject is beyond the scope of this thesis. For more information, see [5, 6].

2.2. NUMERICAL MODEL

Before starting the numerical treatment of the terms in the governing equations, we lay out the computational framework for the spatial discretization employed in the numerical method. For simplicity, this will be done in separate categories.

2.2.1. COMPUTATIONAL DOMAIN AND GRID

The computational domain Ω is rectangular, and covered with a Cartesian grid. The grid is fixed, structured, and allowed to mildly stretch such that it can be finely spaced in the zone of interest. Far away from this part, the grid becomes progressively coarser. This is where the major disadvantage of this procedure lies. If one desires a closely spaced grid

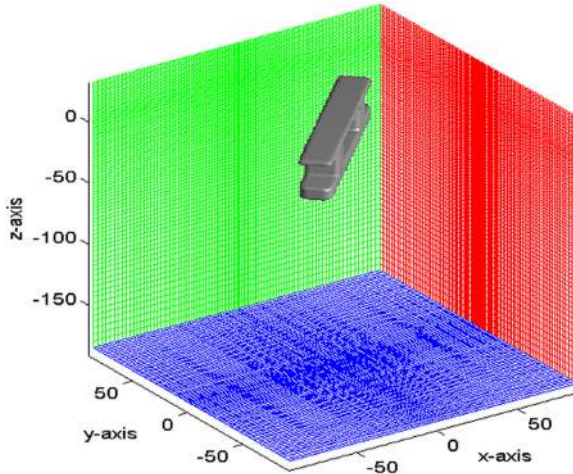


Figure 2.2: A 3D stretched grid in a domain with a semi-submersible.

around a particular location such as the structure illustrated in Fig. 2.2, the grid does not become stretched only around that location but also along the entire coordinate

directions. This is nicely shown, for example, on the plane with red color where refinement is done along the entire z -direction. This procedure increases computational cost unnecessarily.

In order to have a computationally less expensive mechanism for grid generation, a local grid refinement procedure is implemented in the program. This allows us to have a locally refined structured grid only in pre-specified zones of the computational domain. For more information, the reader is referred to [7, 8].

The grid also covers any object in the computational domain, and even if the object is moving grid generation is performed only once before starting the numerical computation. As a result, the grid generation procedure is fast, and the subsequent bookkeeping is straightforward. However, implementation of the boundary conditions in cut-cells that contain both the fluid and the object is difficult.

The flow variables $\mathbf{u} = (u, v, w)$ and p are positioned in a grid cell using the staggered grid arrangement. This is illustrated in Fig. 2.3. The velocity components are located at

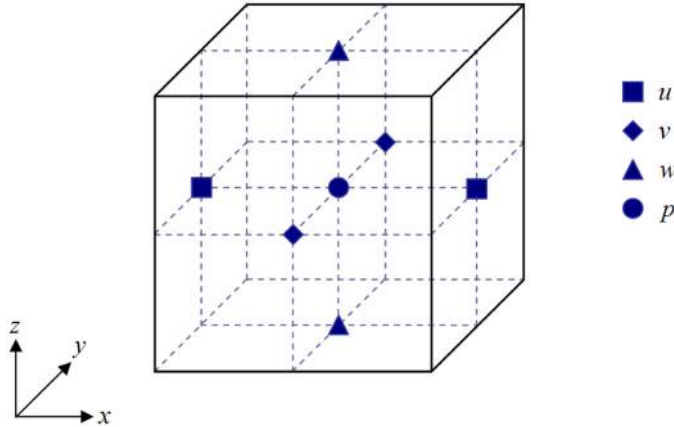


Figure 2.3: Positions of the velocity components and pressure in a grid cell based on the staggered grid arrangement.

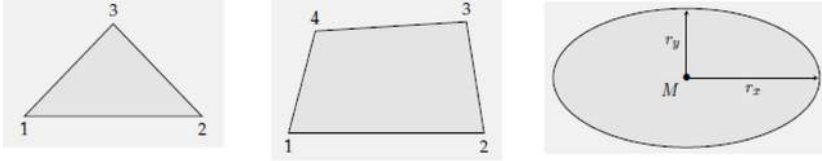
the face centers of the grid cell while the pressure is located at the center of the grid cell.

2.2.2. OBJECT AND FLUID CONFIGURATION

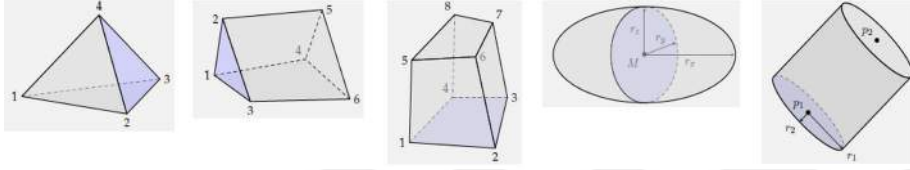
Generation of objects in the computational domain can be done through the use of geometrical entities based on finite elements. In 2D and 3D, a number of various entities are available to generate arbitrary and complex object configurations. These are shown in Fig. 2.4.

Similarly, the 2D and 3D geometrical entities can also be used to specify the initial fluid configuration. As a result, the computational domain contains the object and fluid configurations both of which are covered with a structured Cartesian grid. Afterwards, the introduction of various scalar fields is necessary because geometrical information regarding the initial object and fluid configurations is required.

To explain the framework clearly, a typical cut-cell is shown in Fig. 2.5. A part of the



(a) From left to right: triangle, quadrilateral, ellipse.



(b) From left to right: tetrahedron, wedge, brick, ellipsoid and cylinder.

Figure 2.4: Various 2D and 3D geometrical entities.

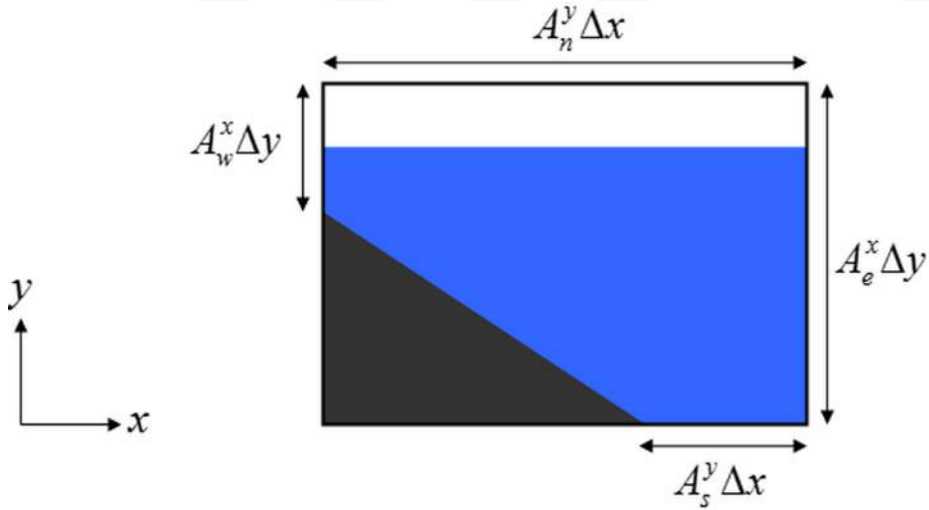


Figure 2.5: A cut-cell containing both the fluid (blue color) and the object (gray color). Face apertures (A^x and A^y) define the proportion of the cell faces open to flow.

cut-cell is filled with water (blue color) and another part is occupied by the object (gray color), and what remains is simply void (white color). We have three sets of scalar fields that represent the fluid-object configuration in each grid cell. The volume aperture F^b denotes the proportion of the cell volume which is open to flow (Although the cut-cell in the figure is shown in 2D, we prefer the term 'volume' rather than 'area'). This indicates that the proportion of the cell volume occupied by the object is $(1 - F^b)$. The proportion of the cell volume filled with the fluid is designated by F^s , which is generally called the VOF function. Since a cell cannot contain more fluid than it allows, we have $0 \leq F^s \leq$

$F^b \leq 1$. Finally, we have the edge apertures A^x , A^y (and A^z in 3D) which define the portion of the cell faces open to flow. Through the use of the three scalar fields, we have the required geometrical information, which will be used when the governing equations are discretized.

2.2.3. CELL LABELING

Based on the object and the fluid configurations, the grid cells are labeled in order to simplify treating cells of similar characteristics. This is performed by employing four labels which help distinguish between cell groups: Empty, Surface, Boundary and Fluid cells.

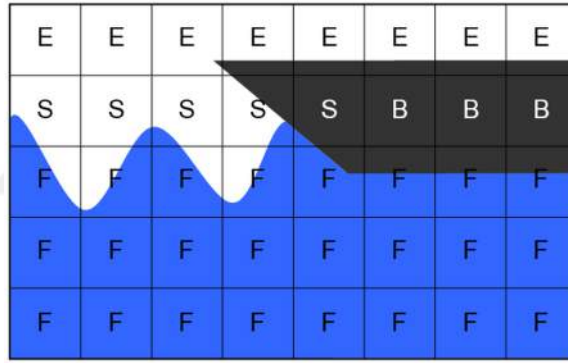


Figure 2.6: Four labels distinguishing cells of similar characteristics: Empty, Surface, Boundary and Fluid cells.

The grid cells that contain no fluid at all are labeled as Empty cells while the ones that are completely occupied by the structure ($F^b = 0$) are labeled as the Boundary cells. The cells bordering the Empty cells are labeled as the Surface cells and the remaining cells become the Fluid cells. An example of the labeling system is illustrated in Fig. 2.6, where the domain includes an object shown with gray color, a fluid shown with blue color and some void shown with white color. The labeling process plays a critical role in different parts of the numerical method, such as implementing the boundary conditions and solving the governing equations. This process is performed at each time level to update the labels throughout the domain.

2.2.4. SPATIAL DISCRETIZATION OF THE GOVERNING EQUATIONS

As mentioned before the finite volume formulation is employed to spatially discretize the governing equations inside the computational domain. Each term in the governing equations will be separately treated based on the numerical framework explained in the previous sections. In what follows, discretization will be performed only in 2D for the sake of simplicity, as the extension to 3D is straightforward.

Using matrix-vector notation, the equations of motion can be written as

$$\mathbf{D}\mathbf{u}_h = 0 \quad (2.15)$$

$$\Omega \frac{d\mathbf{u}_h}{dt} + \mathbf{C}(\mathbf{u}_h)\mathbf{u}_h + \mathbf{V}\mathbf{u}_h + \mathbf{G}\mathbf{p}_h = 0 \quad (2.16)$$

where $\mathbf{D}$ and $\mathbf{G}$ denote, respectively, the discrete divergence and gradient operators, Ω is a diagonal matrix containing the sizes of the control volumes, $\mathbf{u}_h$ contains the discrete velocities, $\mathbf{p}_h$ contains the discrete pressures, and $\mathbf{C}(\mathbf{u}_h)$ and $\mathbf{V}$ represent the convective and viscous diffusive coefficient operators, respectively. This is actually where a specific discretization technique, symmetry preserving discretization [9], comes into play: the coefficient matrices $\mathbf{D}$, $\mathbf{G}$, $\mathbf{C}(\mathbf{u}_h)$ and $\mathbf{V}$ are formed in such a way that crucial properties of the corresponding differential operators are preserved. This entails that $\mathbf{C}(\mathbf{u}_h)$ is skew-symmetric as the corresponding differential operator $(\mathbf{u} \cdot \nabla)$, $\mathbf{V}$ is symmetric negative-definite as $(\nabla \cdot \nabla)$, and $\mathbf{G}^* = -\mathbf{D}$ holds as $\nabla^* = -(\nabla \cdot)$. When the evolution of the discrete energy of the discrete governing equations is analyzed, [Verstappen and Veldman](#) [9] state that it decreases in time when the symmetry preserving discretization is employed. This consequently results in a stable system, and allows us to determine a solution to the above system on any arbitrary grid. For more information, see [9–11]. An explanation on how each discrete operator is formed will be provided below.

THE CONTINUITY EQUATION

The continuity equation (2.13) is applied in cell centers, and discretized using the velocity components defined on each face of the cell, see Fig. 2.7. A part of the grid cell is occupied by the object (gray color) and the remaining part is filled with the fluid (blue color). As shown in the figure the velocity components are positioned in the centers of

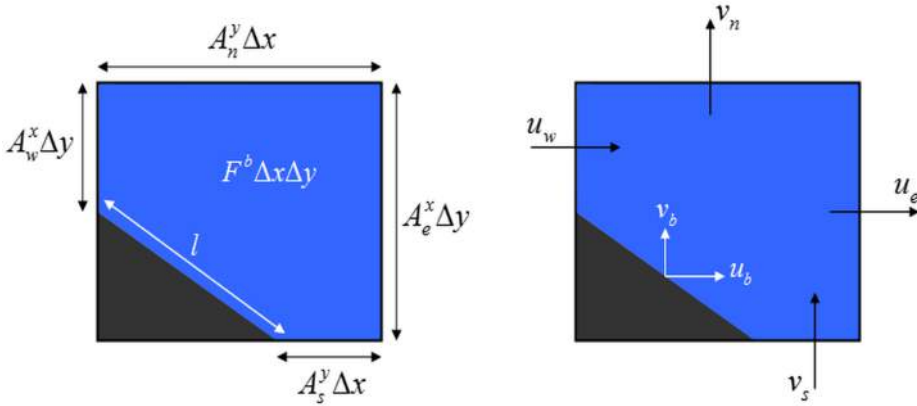


Figure 2.7: A cut-cell in which the continuity equation is applied.

the open parts of the cell faces. Furthermore, the object in the cell is also moving with the velocity $\mathbf{u}_b = (u_b, v_b)$ which is constant over l . Discretization of the continuity equation in this cell yields the following expression

$$\begin{aligned} u_e A_e^x \Delta y + v_n A_n^y \Delta x - u_w A_w^x \Delta y - v_s A_s^y \Delta x + \\ u_b (A_e^x - A_w^x) \Delta y + v_b (A_n^y - A_s^y) \Delta x = 0. \end{aligned} \quad (2.17)$$

THE MOMENTUM EQUATION

The spatial discretization of the momentum equation will be explained term by term in 2D for only the x -direction. The same methodology can easily be applied for the other coordinate directions and also in 3D.

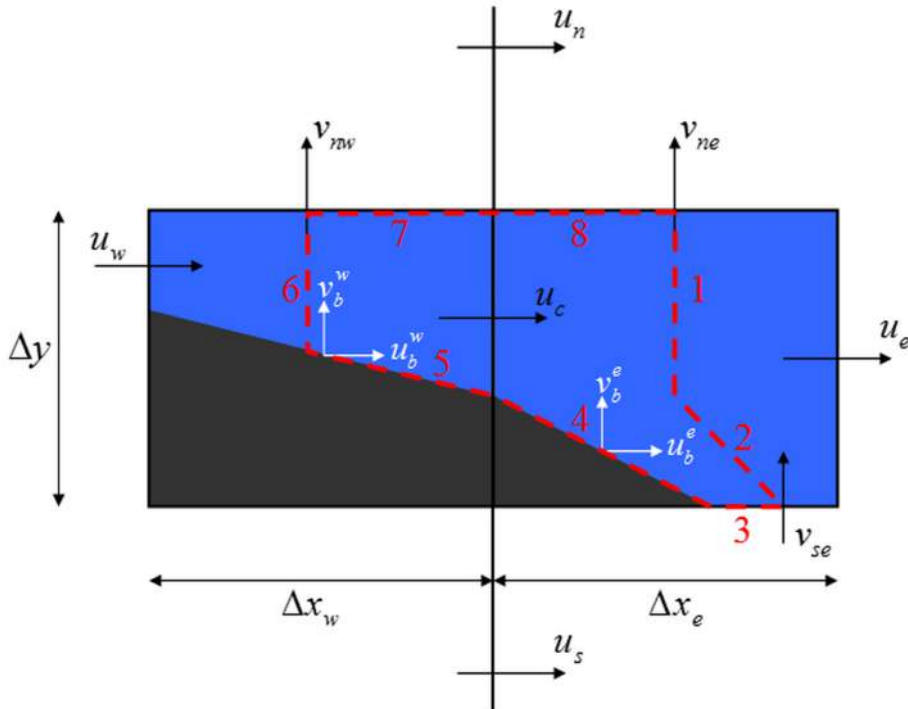


Figure 2.8: Notation used in the discretization of the convection term. Control volume enclosed by the red dashed line.

Time derivative:

Spatial discretization of the time derivative in Eq. (2.14) is performed by employing the midpoint rule,

$$\int_{\Omega_f} \frac{\partial u}{\partial t} d\Omega_f = \frac{\partial u_c}{\partial t} \Omega_c \quad (2.18)$$

where Ω_c denotes the size of the control volume enclosed by the red dashed line belonging to the horizontal velocity u_c , see Fig. 2.8 for the illustration. Taking F_e^b and F_w^b as the volume apertures of the eastern and western cell, respectively, Ω_c is computed using the following expression

$$\Omega_c = F_c^b \Delta x_c \Delta y = \left(\frac{F_w^b \Delta x_w + F_e^b \Delta x_e}{2} \right) \Delta y. \quad (2.19)$$

Convection term:

The convection term in the x -direction momentum equation reads

$$\int_{\partial\Omega_f} u \mathbf{u} \cdot \mathbf{n}_f d\partial\Omega_f.$$

The convection term actually consists of two scalars: u denoting the velocity component in the x -direction and $\mathbf{u} \cdot \mathbf{n}_f d\partial\Omega_f$ denoting the mass flux through the entire boundary of the control volume. As demonstrated in Fig. 2.8, in order to simplify discretizing this term, the boundary of the control volume illustrated by the red dashed line is divided into eight segments. This allows us to evaluate the integral over each of the segments separately,

$$\int_{\partial\Omega_f} u \mathbf{u} \cdot \mathbf{n}_f d\partial\Omega_f = \sum u_i m_i \quad (2.20)$$

where u_i and m_i indicate the velocity component in the x -direction at the i th segment and the mass flux through the i th segment, respectively. Below, we will treat u_i and m_i separately over each segment.

Recognizing the velocity of the object in the eastern cell $\mathbf{u}_b^e = (u_b^e, v_b^e)$ and in the western cell $\mathbf{u}_b^w = (u_b^w, v_b^w)$, the mass flux through each segment is calculated as follows

$$\begin{aligned} m_1 &= \frac{1}{2} (u_e + u_c) A_c^x \Delta y, \\ m_2 &= \frac{1}{2} (u_e + u_b^e) (A_e^x - A_c^x) \Delta y + \frac{1}{2} v_b^e (A_{ne}^y - A_{se}^y) \Delta x_e, \\ m_3 &= -\frac{1}{2} v_{se} A_{se}^y \Delta x_e, \\ m_4 &= -u_b^e (A_e^x - A_c^x) \Delta y + v_b^e (A_{ne}^y - A_{se}^y) \Delta x_e, \\ m_5 &= -\frac{1}{2} u_b^w (A_c^x - A_w^x) \Delta y + \frac{1}{2} v_b^w (A_{nw}^y - A_{sw}^y) \Delta x_w, \\ m_6 &= -\frac{1}{4} (u_c + u_w) (A_c^x + A_w^x) \Delta y, \\ m_7 &= \frac{1}{2} v_{nw} A_{nw}^y \Delta x_w, \\ m_8 &= \frac{1}{2} v_{ne} A_{ne}^y \Delta x_e. \end{aligned}$$

Note the minus signs of the mass fluxes through the segments 3, 4, 5 and 6.

Determining u_i at the i th segment of the boundary is, however, more critical. The simplest choice seems to be taking the average of the velocities on each side of the i th segment. For example, for u_1 we take $u_1 = (u_e + u_c)/2$, and the other velocities can be found in a similar fashion. In this case, the coefficient matrix for convection becomes skew-symmetric; a characteristic also possessed by the continuous operator for convection [9]. Several upwind-based schemes are discussed by Loots [12]. Also, Gerrits [13]

presents a discussion on this matter. Consequently, substituting u_i computed with a chosen scheme (in this study this choice is made among the first- and second-order up-wind and central schemes) and m_i into Eq. (2.20) completes discretizing the convective term of the x -momentum equation.

Diffusion term:

The diffusion term in the x -direction momentum equation reads

$$\int_{\partial\Omega_f} \nabla \mathbf{u} \cdot \mathbf{n}_f d\partial\Omega_f$$

where the factor v is omitted.

The diffusion term was discretized in two different ways in the studies by Gerrits [13] and Kleefsman *et al.* [14]. Gerrits [13] writes the integrand as $\partial u / \partial \mathbf{n}$ and obtains the following

$$\int_{\partial\Omega_f} \nabla u \cdot \mathbf{n}_f d\partial\Omega_f = \sum \frac{u_i - u_c}{|\mathbf{n}_i|} A_i \quad (2.21)$$

where u_i is, $|\mathbf{n}_i|$ is the distance between u_i and u_c , and A_i is the length (area in 3D) of the i th segment. Further, $|\mathbf{n}_i|$ is approximated by

$$|\mathbf{n}_i| = \frac{V_i}{A_i} \quad (2.22)$$

where V_i is the area (volume in 3D) of the i th segment. For u_i , A_i and V_i , Gerrits [13] makes use of simple but reasonable ideas, and the resulting coefficient matrix for diffusion becomes symmetric. This approach is also used by Dröge and Verstappen [10]. More information can also be found in the PhD thesis by Dröge [15].

Kleefsman *et al.* [14] state that the above method may introduce instabilities stemming from the division by the volume V_i which can be arbitrarily small in cut-cells. To circumvent this issue, the object is treated in a staircase manner, allowing to treat a cut-cell as uncut. For this purpose, Kleefsman *et al.* [14] employ the idea introduced by Verstappen and Veldman [9], and write the diffusion term as a volume integral,

$$\int_{\partial\Omega_f} \nabla u \cdot \mathbf{n}_f d\partial\Omega_f = \int_{\Omega} \nabla \cdot \nabla u d\Omega. \quad (2.23)$$

The resulting second-order derivative is calculated using the midpoint rule which was previously used when discretizing the time derivative. The inner-derivative is applied at the faces of the control volume 1, 2, 3 and 4, and the outer derivative is applied at the center of the control volume, see Fig. 2.9 for notation. This results in

$$\nabla \cdot \nabla u = \frac{\left(\frac{\partial u}{\partial x}\right)_1 - \left(\frac{\partial u}{\partial x}\right)_3}{\Delta x_c} + \frac{\left(\frac{\partial u}{\partial y}\right)_4 - \left(\frac{\partial u}{\partial y}\right)_2}{\Delta y_c} \quad (2.24)$$

where $\Delta x_c = (\Delta x_e + \Delta x_w)/2$, $\Delta y_c = (\Delta y_n + \Delta y_s)/2$, and the derivative at the face 1 can be obtained by

$$\left(\frac{\partial u}{\partial x}\right)_1 = \frac{u_e - u_c}{\Delta x_e}.$$

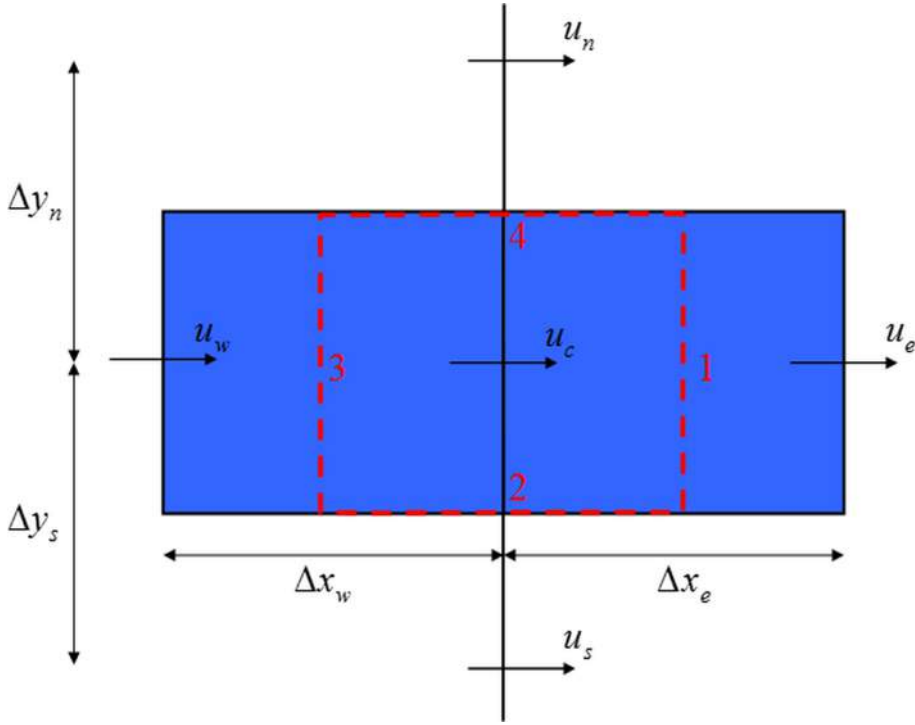


Figure 2.9: Notation used in the discretization of the diffusion term.

Other derivatives can be computed in a similar fashion.

Although this approach compromises accuracy yielding a first-order discretization, it produces a symmetric negative-definite coefficient matrix for diffusion just as the continuous operator [9]. Recently the discretization of the diffusion term has been improved further in the vicinity of walls [16–18]. We will not explain these improvements here as it is outside the scope of this short introduction.

Pressure term:

Discretization of the pressure term in the x -direction momentum equation is more straightforward compared to the convection and diffusion terms since the pressure is assumed to have a constant value in a grid cell. This lets us assign the same pressure value to all boundary segments in a grid cell. Recalling the pressure term,

$$\int_{\partial\Omega_f} p n_x d\partial\Omega_f$$

in which n_x is the x -component of the outward pointing normal along the boundary of the control volume, and the factor $-1/\rho$ is ignored. Analogous with the treatments of the convection and diffusion terms, this term is also evaluated over each segment of the

boundary as follows

$$\int_{\partial\Omega_f} p n_x d\partial\Omega_f = \sum p_i \int_i n_{x_i} d\partial\Omega_f. \quad (2.25)$$

Considering the notation exhibited in Fig. 2.10, we observe that $p = p_e$ holds for the

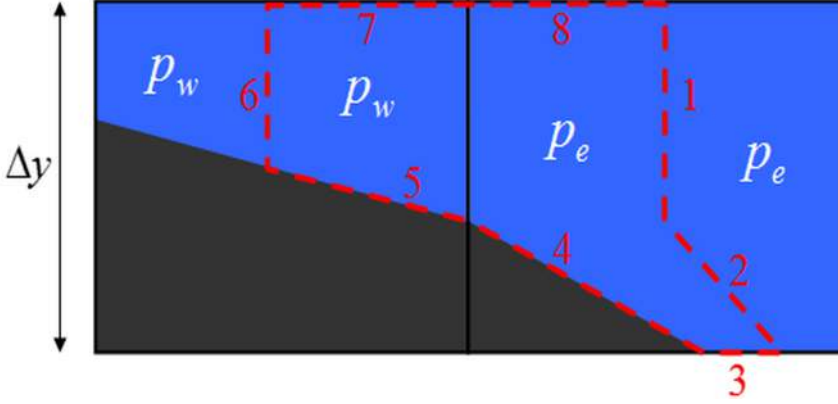


Figure 2.10: Notation used in the discretization of the pressure term.

segments 1, 2, 3, 4 and 8 while $p = p_w$ for 5, 6, and 7. However, the segments 3, 7 and 8 do not contribute to the integral as $n_x = 0$ along them. Furthermore, the segments 2 and 4 cancel each other because they have the same pressure value while the x -component of their normals are equal but have the opposite signs. Finally, we are left with only the segments 1, 5 and 6, and their contributions are calculated as

$$\int_{\partial\Omega_f} p n_x d\partial\Omega_f = p_e A_c^x \Delta y - p_w \left(\frac{A_c^x - A_w^x}{2} \right) \Delta y - p_w \left(\frac{A_c^x + A_w^x}{2} \right) \Delta y. \quad (2.26)$$

Rearranging the right hand side yields

$$\int_{\partial\Omega_f} p n_x d\partial\Omega_f = (p_e - p_w) A_c^x \Delta y \quad (2.27)$$

which depends on the open part of the central edge in the control volume but does not depend on the exact location of the object. Looking at Eqs. (2.27) and (2.17) we notice a nice property: the analytical property $\nabla = -(\nabla \cdot)^T$ is also preserved by the discrete divergence and gradient operators.

Body force term:

Recalling the body force term in the momentum equation,

$$\int_{\Omega_f} \mathbf{f} d\Omega_f,$$

we remind that the only external force is gravity, thus $\mathbf{f} = \mathbf{g} = (g_x, g_y, g_z)^T$. Here we look for a discretization which will match to that of the pressure term since the change in

pressure must be related to the gravity term in the absence of all other terms. If, for example, we take $\mathbf{g} = (0, 0, -g)^T$, then we need to discretize the gravity term only for the momentum equation in the z -direction. Using Gauss' divergence theorem the volume integral is written as a boundary integral, and the following expression is obtained

$$\int_{\partial\Omega_f} -gz n_z d\partial\Omega_f.$$

When this boundary integral for the hydrostatic pressure potential is computed, we reach

$$\int_{\partial\Omega_f} -gz n_z d\partial\Omega_f = -A_c^z \Delta x g (z_n - z_s) \quad (2.28)$$

in which z_n and z_s stand for the center locations along the z -direction of the northern and southern cells, respectively.

2.2.5. TEMPORAL DISCRETIZATION AND SOLUTION TECHNIQUE

Here, discretization of the equations of motion (2.13) and (2.14) in time is briefly explained. This process is realized using either the first-order forward Euler or the second-order Adams-Bashforth methods. For a simple time derivative term $d\phi/dt = f(\phi)$, forward Euler and Adams-Bashforth take the following forms, respectively

$$\phi^{n+1} = \phi^n + \Delta t f(\phi^n), \quad (2.29)$$

$$\phi^{n+1} = \phi^n + \Delta t f\left(\frac{3}{2}\phi^n - \frac{1}{2}\phi^{n-1}\right). \quad (2.30)$$

Both schemes have been implemented by [Wemmenhove \[3\]](#) in ComFLOW. Here, we consider the forward Euler integration. As we will see later in this study, the forward Euler scheme is more compatible with the time integration method employed in the evolution of the VOF function.

In order to explain the solution technique in a clear manner, we write the equations of motion in a more schematic form

$$\text{div } \mathbf{u}^{n+1} = 0, \quad (2.31)$$

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \frac{1}{\rho} \text{grad } p^{n+1} = \mathbf{R}^n \quad (2.32)$$

where n and $n+1$ indicate the old and new time level, respectively, Δt is the time step, and $\mathbf{R}$ contains all convective, diffusive and body forces

$$\mathbf{R}^n = -(\mathbf{u}^n \cdot \text{grad}) \mathbf{u}^n + \nu \text{div grad } \mathbf{u}^n + \mathbf{F}^n \quad (2.33)$$

in which ν is the kinematic viscosity. Discretizing the continuity equation (2.31) at the new time level ensures a divergence-free velocity field at this time level. Rearranging the terms in Eq. (2.32) yields

$$\mathbf{u}^{n+1} = \tilde{\mathbf{u}}^n - \frac{\Delta t}{\rho} \text{grad } p^{n+1} \quad (2.34)$$

where

$$\tilde{\mathbf{u}}^n = \mathbf{u}^n + \Delta t \mathbf{R}^n. \quad (2.35)$$

The term $\tilde{\mathbf{u}}^n$ is often referred to as an auxiliary velocity, and calculated first in the solution process. After substituting (2.34) into (2.31), we have

$$\text{div grad } p^{n+1} = \frac{\rho}{\Delta t} \text{div } \tilde{\mathbf{u}}^n. \quad (2.36)$$

Equation (2.36) is the Poisson equation for the pressure. Typically, this equation is solved using the SOR (Successive Over Relaxation) method such as developed by [Botta and El-lenbroek](#) [19]. Once the pressure field is obtained, the velocity field at the new time level is calculated via Eq. (2.34).

2.2.6. STABILITY

[Verstappen and Veldman](#) [9] and [Fekken](#) [5] discuss the stability of the overall method extensively. Regarding the explicit time integration and convection, the stability criterion is the CFL condition [20] which reads in one spatial dimension

$$\frac{|u| \Delta t}{h} \leq 1$$

in which h is the size of the uncut cell. This condition also holds in the case of cut-cells with fixed objects [5]. When the objects are moving, arbitrarily small control volumes may cause stability issues. In order to overcome this problem, additional measures are employed when the velocity field at the new time level is calculated. These measures counteract the effect of very small control volumes without stipulating very small time step. We will not go into the details of this topic here, and refer the interested reader to [5] for a more comprehensive discussion.

From the diffusive term, an additional stability criterion is considered, which takes the following form in one spatial dimension

$$\frac{2\nu \Delta t}{h^2} \leq 1.$$

When the diffusive term is discretized in a staircase manner where cut-cells are assumed to be uncut, this criterion is valid for all cells. If other approaches are used for the discretization of the diffusive term, we may need to adopt a different criterion, see [16–18] for more information.

REFERENCES

- [1] J. Anderson, *Computational fluid dynamics* (McGraw-Hill, 1995).
- [2] M. G. el Hak, *Stokes' hypothesis for a Newtonian, isotropic fluid*, J. Fluids Eng. **117**, 3 (1995).
- [3] R. Wemmenhove, *Numerical simulation of two-phase flow in offshore environments*, Ph.D. thesis, University of Groningen (2008).

- [4] R. Sani and P. Gresho, *Résumé and remarks on the open boundary condition minisymposium*, Int. J. Numer. Meth. Fluids **18**, 983 (1994).
- [5] G. Fekken, *Numerical simulation of free-surface flow with moving rigid bodies*, Ph.D. thesis, University of Groningen (2004).
- [6] K. Kleefsman, *Water impact loading on offshore structures*, Ph.D. thesis, University of Groningen (2005).
- [7] P. van der Plas, H. van der Heiden, R. Luppés, and A. Veldman, *Local grid refinement for free-surface flow simulations in offshore applications*, in *Proceedings of the 5th Int. Conf. on Computational Methods in Marine Engineering MARINE*, 29-31 May 2013, Hamburg (Germany) (2013).
- [8] P. van der Plas, A. Veldman, H. van der Heiden, and R. Luppés, *Adaptive grid refinement for free-surface flow simulations in offshore applications*, in *Proceedings of the 34th Int. Conf. on Ocean, Offshore and Arctic Eng. OMAE*, 31 May-5 June 2015, St. John's (Canada), paper OMAE2015-42029 (2015).
- [9] R. Verstappen and A. Veldman, *Symmetry-preserving discretization of turbulent flow*, J. Comput. Phys. **187**, 343 (2003).
- [10] M. Dröge and R. Verstappen, *A new symmetry-preserving cartesian-grid method for computing flow past arbitrarily shaped objects*, Int. J. Numer. Meth. Fluids **47**, 979 (2005).
- [11] A. Veldman, *Computational Fluid Dynamics* (Lecture notes in Applied Mathematics, 2010).
- [12] G. Loots, *Fluid-structure interaction in hemodynamics*, Ph.D. thesis, University of Groningen (2003).
- [13] J. Gerrits, *Dynamics of liquid-filled spacecraft*, Ph.D. thesis, University of Groningen (2001).
- [14] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, and B. Buchner, *A Volume-of-Fluid based simulation method for wave impact problems*, J. Comput. Phys. **206**, 363 (2005).
- [15] M. Dröge, *Cartesian grid methods for turbulent flow simulation in complex geometries*, Ph.D. thesis, University of Groningen (2007).
- [16] P. van der Plas, H. van der Heiden, A. Veldman, R. Luppés, and R. Verstappen, *Efficiently simulating viscous flow effects by means of regularization turbulence modeling and local grid refinement*, in *Proceedings of the 6th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS)*, 10-14 September 2012, Vienna (Austria) (2012).

- [17] H. van der Heiden, P. van der Plas, A. Veldman, R. Luppens, and R. Verstappen, *Efficient computation and modeling of viscous flow effects in ComFLOW*, in *Proceedings of the 32nd Int. Conf. on Ocean, Offshore and Arctic Eng. OMAE*, 9-14 June 2013, Nantes (France), paper OMAE2013-11000 (2013).
- [18] H. van der Heiden, A. Veldman, R. Luppens, P. van der Plas, J. Helder, and T. Bunnik, *Turbulence modeling for free-surface flow simulations in offshore applications*, in *Proceedings of the 34th Int. Conf. on Ocean, Offshore and Arctic Eng. OMAE*, 31 May-5 June 2015, St. John's (Canada), paper OMAE2015-41578 (2015).
- [19] E. Botta and M. Ellenbroek, *A modified SOR method for the Poisson equation in unsteady free-surface flow calculations*, *J. Comput. Phys.* **60**, 119 (1985).
- [20] R. Courant, K. Friedrichs, and H. Lewy, *Über die partiellen Differenzgleichungen der mathematischen Physik*, *Mathematische Annalen* **100**, 32 (1928).

3

ABSORBING BOUNDARY CONDITION (ABC)

3.1. INTRODUCTION

SINCE the 19th century when first wave theories were proposed, understanding the motion and behavior of the waves in nature has been a very popular subject among researchers from various fields of science. Nevertheless, even today when we have highly capable numerical methods and computational power at our disposal, particular aspects of numerical modeling of water wave propagation remain a formidable challenge. Typically the phenomena of interest are local but embedded in a vast spatial domain; for example, the interaction between free surface waves and numerous kinds of man-made structures in the ocean. For efficient computational modeling, this vast spatial domain around the region of interest is truncated via artificial boundaries, which suggests that a compact computational domain around the structure and a residual infinite domain are introduced. At this point, one of the most elusive and difficult topics surfaces when we try to answer this question: What is the boundary condition to be imposed on these artificial boundaries in such a way that the solution in the compact domain coincides with the solution in the original domain?

In the literature the boundary conditions applied on the artificial boundaries are called by various names, such as non-reflecting, absorbing, open, transparent and radiating boundary conditions. Throughout the rest of this thesis we will use the term Absorbing Boundary Condition (ABC). Several types can be listed under the wide variety of ABCs: Nonlocal, semi-local or local operators, numerical dissipation zones and Dirichlet-to-Neumann (DtN) map based conditions.

Each type of ABC has significant amount of work behind it since a large number of researchers from different fields extended earlier works in various directions over the last few decades. This has resulted in a substantially broad literature. Fortunately, there are a number of good reviews. For a review regarding only high-order local ABCs, see [Givoli \[1\]](#). [Givoli \[2\]](#) and [Tsynkov \[3\]](#) present lengthy overviews of local and non-local

ABCs along with other artificial boundary conditions. Also, Hagstrom [4] offers a survey of exact ABCs. Colonius [5] discusses the use of ABCs and numerical dissipation zones suitable especially for compressible turbulent shear flows. For a review of ABCs and perfectly matched layers (PML) for wave propagation problems, the reader is referred to Appelö [6]. Several techniques to absorb free surface waves are briefly discussed by Romate [7].

For convenience we will present a short introduction to the first hierarchy of local absorbing boundary conditions developed by Engquist and Majda [8]. This work can be considered as one of the cornerstones in the field of absorbing boundary conditions, and is essentially related to the ABC that we will design later in this chapter. We start the discussion by considering the two-dimensional wave equation

$$\frac{\partial^2 \phi}{\partial t^2} = c^2 \left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right) \quad (3.1)$$

where c is the propagation or phase speed. Solutions of this equation are plane waves that have the following form

$$\phi(x, y, t) = e^{i(k_x x + k_y y - \omega t)} \quad (3.2)$$

where ω is the frequency and (k_x, k_y) are the components of the wave number vector in the x and y -directions, respectively, $\mathbf{k} = (k_x, k_y)$. Substitution of Eq. (3.2) in Eq. (3.1) results in the following expression

$$\omega^2 = c^2 (k_x^2 + k_y^2) \quad (3.3)$$

which is called the dispersion relation. Now assume that we have an artificial numerical boundary at a location along the x direction and waves are propagating eastward. Correspondingly we solve Eq. (3.3) for k_x ,

$$k_x = \frac{\omega}{c} \sqrt{1 - \left(\frac{k_y c}{\omega} \right)^2}. \quad (3.4)$$

Because of the square root on the right-hand side, Eq. (3.4) will yield a pseudo-differential equation rather than a partial differential equation when transformed back into the physical domain. Unfortunately the pseudo-differential equation is nonlocal both in time and space. This suggests that the entire temporal history of the solution needs to be stored over the whole boundary where the ABC is applied, thus it is considerably nonviable to implement in practice. Therefore the square root on the right-hand side is replaced by a Taylor series representation in order to obtain a partial differential equation, which can be locally discretized in the physical domain. Considering only the first few terms of its Taylor series representation we can approximate the square root as follows

$$\sqrt{1-s} = 1 - \frac{s}{2} + O(s^2) \quad (3.5)$$

where $s = (k_y c / \omega)^2$ (see [9] for a discussion on various expansion techniques). If we take only the first term in this representation and ignore the rest, hence $\sqrt{1-s} \approx 1$, and

transform the resulting dispersion relation back into the physical domain, we obtain

$$-\omega + ck_x = 0 \Rightarrow \left(\frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right) \phi = 0. \quad (3.6)$$

Eq. (3.6) is the 1st-order Engquist-Majda boundary condition [8], or alternatively the Sommerfeld condition [10]. Similarly if we consider the first two terms in the Taylor series representation and redo the process, we reach

$$\omega^2 - ck_x \omega - \frac{1}{2} c^2 k_y^2 = 0 \Rightarrow \left(\frac{\partial^2}{\partial t^2} + c \frac{\partial^2}{\partial x \partial t} - \frac{1}{2} c^2 \frac{\partial^2}{\partial y^2} \right) \phi = 0 \quad (3.7)$$

which is the 2nd-order Engquist-Majda boundary condition. If we make use of the wave equation given in Eq. (3.1), we can reformulate Eq. (3.7) in the equivalent form:

$$\left(\frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right)^2 \phi = 0. \quad (3.8)$$

In order to assess the benefit of using Eq. (3.8) as opposed to Eq. (3.6), we need to take a look at the amount of spurious reflection generated by each scheme as a function of the angle of incidence. For this purpose we utilize the idea that at the artificial boundary, we can express the solution as the sum of outgoing and reflected waves, that is

$$\phi(x, y, t) = e^{i(k_x x + k_y y - \omega t)} + R e^{i(-k_x x + k_y y - \omega t)}, \quad (3.9)$$

where the first term represents the wave with amplitude equal to unity impinging on the boundary, and the second term represents the spuriously reflected wave with amplitude R . To evaluate R , we substitute Eq. (3.9) into the 1st and 2nd-order Engquist-Majda boundary conditions, and arrive at the following relations

$$\begin{aligned} |R_{E\&M-1}| &= \left| \frac{1 - \cos \theta}{1 + \cos \theta} \right|, \\ |R_{E\&M-2}| &= \left| \frac{1 - \cos \theta}{1 + \cos \theta} \right|^2. \end{aligned} \quad (3.10)$$

Here $R_{E\&M-1}$ and $R_{E\&M-2}$ denote the amount of spurious reflection corresponding to the 1st and 2nd-order Engquist-Majda boundary conditions, respectively, and θ is the angle of incidence measured in the clockwise or counter-clockwise direction from the positive x -direction, $|\theta| < \pi/2$. Observing Eqs. (3.10) it can be readily seen that both boundary conditions are nonreflecting when the waves are approaching in the normal direction to the boundary, $\theta = 0$. If the waves approach the boundary at nonnormal angles of incidence, then $|R_{E\&M-2}| < |R_{E\&M-1}|$ holds for $|\theta| < \pi/2$. When $|\theta| = \pi/2$, both boundary conditions provide full reflection. This result is of no consequence since it corresponds to the case where waves propagate parallel to the boundary, which is redundant with our discussion.

Higdon [11, 12] later showed that it is possible to modify the 1st-order Engquist-Majda boundary condition, and developed the following condition

$$\cos \alpha \frac{\partial \phi}{\partial t} + c \frac{\partial \phi}{\partial x} = 0, \quad (3.11)$$

with the corresponding reflection coefficient

$$|R_{H-1}| = \left| \frac{\cos \alpha - \cos \theta}{\cos \alpha + \cos \theta} \right|. \quad (3.12)$$

It can be readily seen that the 1st-order Higdon condition is more flexible than the 1st-order Enguist-Majda condition concerning the angle of incidence. As opposed to providing full absorption only at the normal incidence, Eq. (3.11) is nonreflecting for waves impinging on the boundary with an angle $\pm \alpha$. Higdon also presented the higher-order version of this boundary condition

$$\left[\prod_{p=1}^P \left(\cos \alpha_p \frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right) \right] \phi = 0, \quad (3.13)$$

which has the following reflection coefficient

$$|R_H| = \prod_{p=1}^P \left| \frac{\cos \alpha_p - \cos \theta}{\cos \alpha_p + \cos \theta} \right|. \quad (3.14)$$

It is easy to observe that his generalized scheme is perfectly absorbing for waves traveling towards the boundary at angles of incidence $\pm \alpha_1, \dots, \pm \alpha_P$ with the phase speed c . This rather intuitive approach allows us to adapt the choice of α to any situation at hand. It is also possible to see the connection between Eqs. (3.13) and (3.8): setting $P = 2$ and $\alpha_1 = \alpha_2 = 0$ in Eq. (3.13) yields Eq. (3.8). This suggests that Eq. (3.8) is a particular case of the general family of conditions stated by Eq. (3.13).

The reflection coefficient as a function of the angle of incidence for the 1st and 2nd-order Enguist-Majda and Higdon ABCs is illustrated in Fig. 3.1. Clearly the 2nd-order methods are superior to 1st-order ones. Among the four methods the 2nd-order Higdon is the most effective. The choice of $\alpha_1 = 0^\circ, \alpha_2 = 45^\circ$ appears to give the best overall performance for waves striking the boundary under the range of angles between $\theta = 0^\circ$ and $\theta = 50^\circ$.

Similar efforts have been made by other researchers to design high-order local absorbing boundary conditions. For example, Bayliss and Turkel [13] developed a high-order local boundary condition in spherical coordinates. On a spherical artificial boundary of radius R their condition takes the following form

$$\left[\prod_{p=1}^P \left(\frac{1}{c} \frac{\partial}{\partial t} + \frac{\partial}{\partial r} + \frac{2p-1}{R} \right) \right] \phi = 0. \quad (3.15)$$

Theoretically all the families of conditions presented above have an arbitrary accuracy that increases as the order of the conditions increases. However these high-order boundary conditions include high-order spatial and temporal derivatives, which eventually compromises the locality of these conditions. Furthermore numerical implementation starts to become a serious bottleneck since discretizing an arbitrary P th-order differential operator is unpractical. Indeed Grote and Sim [14] note that only up to 2nd-order conditions are most commonly used in practice. Nevertheless the above boundary

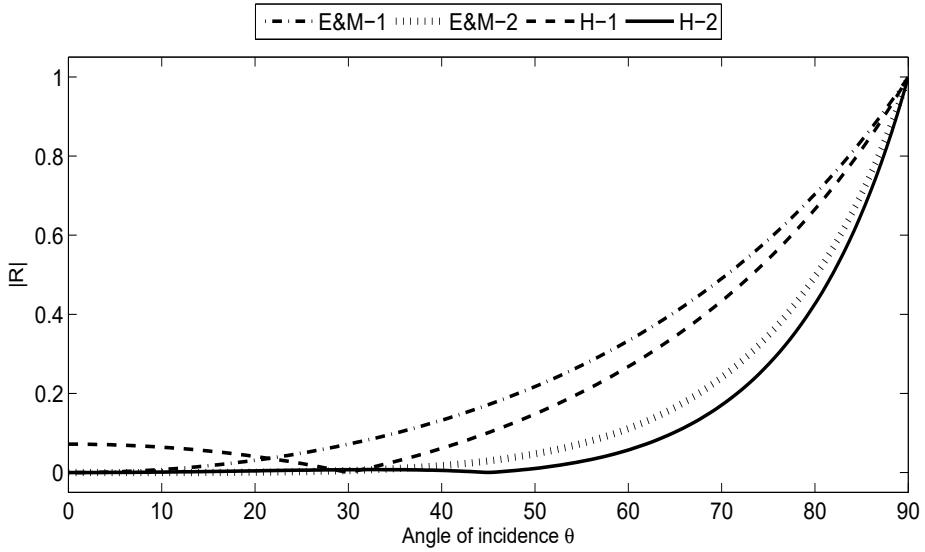


Figure 3.1: Reflection coefficient of four boundary conditions versus the angle of incidence. *E&M-1* and *E&M-2* stand for the 1st and 2nd-order Enguist-Majda ABC, and *H-1* and *H-2* the 1st and 2nd-order Higdon ABC. $\alpha_1 = 30^\circ$ is chosen for the first-order Higdon ABC, and $\alpha_1 = 0^\circ, \alpha_2 = 45^\circ$ for the second-order Higdon ABC.

conditions have been used in practical applications and satisfactory results have been achieved in many cases [15].

In order to circumvent the difficulty of having high-order derivatives associated with a high-order boundary condition, Collino [16] introduced the use of auxiliary variables. His boundary condition of order P takes the following form

$$\begin{aligned} \left(\frac{\partial}{\partial x} + \frac{1}{c} \frac{\partial}{\partial t} \right) \phi - \frac{1}{c} \sum_{p=1}^P \beta_p \frac{\partial}{\partial t} \psi_p &= 0, \\ \frac{1}{c^2} \frac{\partial^2}{\partial t^2} \psi_p - \frac{\partial^2}{\partial y^2} (\alpha_p \psi_p + \phi) &= 0, \end{aligned} \quad (3.16)$$

where $p = 1, \dots, P$ on the artificial boundary and

$$\begin{aligned} \alpha_p &= \cos^2 \left(\frac{p\pi}{2P+1} \right), \\ \beta_p &= \frac{2}{2P+1} \sin^2 \left(\frac{p\pi}{2P+1} \right). \end{aligned} \quad (3.17)$$

Here ψ_p are the auxiliary variables imposed on the artificial boundary. In total, for the P th-order Collino boundary condition, $P+1$ coupled equations are solved on the boundary for the $P+1$ functions $\phi, \psi_1, \dots, \psi_P$. It is important to note that this boundary does not involve any high derivatives beyond 2nd-order. Additionally derivatives of

the auxiliary variables ψ_p are taken only along the boundary (y -derivative) and in time. Givoli [15] states that this feature is fundamental as the auxiliary variables are defined only along the artificial boundary.

Many high-order local absorbing boundary conditions today utilize the idea of auxiliary variables as the basis in their design, e.g., [17], [18] and [19]. Thus Givoli [1] claims that Collino [16] can be regarded as the pioneer of the *practically* high-order local absorbing boundary conditions.

In addition to directional effects of the waves, dispersion has also been the focus of many researchers. Considering the highly nonlinear governing equations which are employed in many complex modeling applications, it is unlikely to have a priori knowledge regarding the phase speed c . As a result c is generally computed during the course of the numerical simulation. Orlanski [20] suggested the following idea to compute c

$$c = -\frac{\partial\phi/\partial t}{\partial\phi/\partial x}. \quad (3.18)$$

Clearly this expression is found from the boundary conditions (3.6) itself, and makes sense only in the discrete form. To discretize the right-hand side, he suggested using the information near the boundary at the previous time step via suitable finite difference approximations. An obvious problem is that we may encounter unphysical values from this computation. When the angle of incidence increases, i.e., the waves are rather oblique, the normal derivative approaches zero, which results in an infinitely large value for c . Furthermore if $c < 0$ is obtained, then this implies that information propagates back into the domain instead of outside. If these unphysical values are in turn used with the boundary condition, numerical instability will occur. Therefore, in practice, a restriction on c is imposed for stable solutions, for example when explicit leapfrog time differencing and upstream space differencing are used in discretizing the boundary condition, then c is bounded to satisfy the CFL condition, $0 \leq c \leq \Delta t/\Delta x$. Durran *et al.* [21] investigated the use of Orlanski's scheme in numerical simulations of shallow water waves, for which c has the well-defined value of $\sqrt{gh}$. They report that the scheme rarely yields a meaningful value for c because computed values of c often fall outside the stability limit, and have to be reset to either zero or $\Delta t/\Delta x$. In the spirit of Orlanski's work various schemes have been developed to approximate c in a more accurate and stable fashion. For a survey of those schemes the reader is referred to the work of Hedley and Yau [22] and Jensen [23].

Other attempts have also been made to incorporate the dispersive nature of the free surface waves into the construction of absorbing boundary conditions. The main trend among researchers seems to be using high-order local ABCs, e.g., [24] and [25] where auxiliary variables are utilized to eliminate high-order derivatives in the high-order ABCs, see [26] for a comparison between the two works for a dispersive one-dimensional medium.

The rest of the chapter is organized as follows. Section 3.2 includes the motivation for the present work. In Section 3.3 we give details of how regular and irregular waves are generated in the computational domain. Formulation of the ABCs is explained in a comprehensive manner in Section 3.4. This includes a discussion on the dispersion relation, theoretical reflection behavior, stability analysis and numerical discretization of

the ABCs. In Section 3.5 results of numerical experiments are presented along with comparisons and discussions in order to fully demonstrate the performance of the ABCs. We end the chapter with conclusions in Section 3.6.

3.2. MOTIVATION

After giving a brief introduction to the topic of local absorbing boundary conditions, we will state our motivation for the work presented in the remainder of this chapter. To do so, we will lay out some requirements, which are relevant for free surface water wave simulations, coastal and offshore applications. We will walk in the footsteps of Givoli [15], and present the criteria which a new ABC is expected to fulfill. We will give our comments together with his remarks written in italic form. In the mean time we will keep in mind the work by Romate [7] and Wellens [27] which discuss properties of different types of ABCs with respect to a number of criteria.

1. *Well-posedness: The problem in the domain including the implementation of the ABC on the artificial boundary is well-posed.* Well-posedness of an initial-boundary value problem (IBVP) needs to be addressed before discretization, and a number of classical references are available in that regard, e.g., [28], [29] and [30]. The starting point of our work will be the Higdon ABC given in Eq. (3.13), and the well-posedness of the IBVP with the Higdon ABC was studied by several researchers, e.g., [7, 11, 31].
2. *Accuracy on the continuous level: The amount of spurious reflection due to the ABC is small.* Theoretically speaking it is possible to reduce the amount of reflection by setting the order of an ABC as high as possible. But is that really necessary? In numerical simulations of relevant practical situations around five percent reflection is generally acceptable since this amount is also encountered in experimental basins and flumes. Therefore it may not be necessary to apply a high-order ABC for the sake of very small amount of reflection. Hence the 1st and 2nd-order Higdon ABCs will be the starting point in our derivation.
3. *Scheme compatibility: The ABC on the boundary is compatible with the numerical scheme used in the domain.* Overall scheme compatibility can be a strict limitation for an ABC. For example, some ABCs in the literature are applicable only in spherical coordinates, and/or in 2D and/or 3D, e.g., the Grote-Keller ABC [32, 33] is designed in spherical coordinates and is inherently 3D. For us it is of critical importance that the ABC is applicable both in 2D and 3D, and can be discretized on a rectangular Cartesian grid. In Section 3.4.7 we will present the ABC in discrete form, and explain how easily the ABC is incorporated into the framework of the overall numerical method.
4. *Stability: The ABC does not allow any unwanted modes which propagate into the computational domain.* Stability of an ABC, or any other boundary condition, is very critical since it provides a clear indication for a potentially successful or unsuccessful computation. Stability of the proposed ABC is discussed in Section 3.4.5. We will show that the ABC is stable as long as several conditions are respected. This suggests that the ABC we design is conditionally stable. However,

we will demonstrate that the range of applications where the ABC is stable is satisfactory for the considered practical situations.

5. *Accuracy on the discrete level: The error due to the numerical implementation of the discrete ABC is small.* Even though an ABC is accurate on the continuous level, it may not be so on the discrete level. Givoli [15] discusses this and reports that, for example, the Perfectly Matched Layer (PML) by Berenger [34] suffers from such an issue. In Section 3.5 we will present results from numerical tests, and show that the error produced by the use of the discrete ABC is at an acceptable level.
6. *Efficiency: The application of the ABC on the boundary does not increase computational cost significantly.* This is where the decision of using a local ABC rather than any other type of ABC becomes clearer. With local ABCs it is possible to locate the artificial boundary close to the area of interest, and hence reduce the amount of computational work to obtain a solution. This is not the case for some of the other ABCs, for example, with the damping layer where large zones of one or more wavelengths are usually added to the domain [35]. Such implementation requires considerable computational effort especially in 3D, which renders the method inefficient. The ABC we will present does not require a large computational effort. This will be shown in Section 3.4.7. This property also holds when the ABC is applied in 3D. The cost of using the ABC on the boundary is proportionate to the cost of solving the governing equations within the interior of the domain.
7. *Ease of implementation: The ABC is easy to apply, and resulting numerical code is maintainable.* Implementation of an ABC is generally not rudimentary. When several properties in this list are set out to be satisfied, the resulting ABC gradually becomes more complicated. Nevertheless we will demonstrate later in this chapter that considering the overall properties of the presented ABC, the difficulty of its numerical implementation is only commensurate.
8. *Generality: The ABC performs well in a wide variety of applications.* The ABC designed here is not a general ABC suitable for a large class of problems. The main focus in this work is to account for dispersive and directional effects of free surface waves in an arbitrary water depth in a 3D domain. Another goal is to design an *open* boundary condition allowing waves to travel in and out of the domain over the same boundary. This is especially needed when there is a structure in the domain. At the moment, it is not clear how to deliver this property with other type of ABCs due to the fact that an in-depth discussion on this topic has been made only by a small number of researchers, see Section 3.4.6 for more information. Observing the work performed with local ABCs we realized that these boundary conditions are more suitable in developing an *open* boundary condition.

The first four criteria in the list are somewhat associated with convergence of a numerical scheme. Naturally this can be demonstrated more readily once the ABC is applied. Givoli [15] states that most of the work in this field has focused on the combination of properties 5 and 6. Designing an ABC which satisfies all the criteria above remains a formidable challenge. Therefore, as many researchers, we made some compromises

when it comes to some properties, and tried to deliver an ABC which accomplishes the most needed goals in an efficient manner. This will be demonstrated throughout several sections in the rest of this chapter.

Before going into the details of the design of the boundary condition, we will now take a look at various wave models at our disposal. Since the absorbing boundary condition will be developed for free surface water waves, it is important to recall some of the properties of the available wave theories. The information that we present below will be concise as these theories are well-established and well-documented.

3.3. WAVE GENERATION

3.3.1. REGULAR WAVES

Linear or nonlinear regular wave models have been commonly used in engineering practice as they can sometimes provide sufficient theoretical information for a variety of physical situations: small amplitude or large amplitude (steep) waves in shallow or deep water. These models including Stokes I, II [36], Stokes V [37], and stream function theory [38] are currently available in ComFLOW. Knowing the limitations of each theory and applying the model accordingly is of critical importance for an accurate interpretation of the motion and behavior of the waves in nature. In this regard, the classical work of Méhauté [39] can help us realize the ranges of applicability of the various wave models. Stokes I theory, sometimes referred to as linear or Airy wave theory, can be used for small-amplitude waves. For large-amplitude waves high-order Stokes theories can be used. However, Stokes theory is not applicable for very shallow water. In this case the cnoidal wave theory, see [40] and [41], or stream function theory should be used [42]. For detailed analyses, the reader is kindly referred to the related papers, or other texts such as [43–45].

3.3.2. IRREGULAR WAVES

Unidirectional (long-crested) irregular waves may be assumed to consist of a finite number of monochromatic regular waves (i.e., Stokes I waves). In many practical situations, this rather simple linear superposition can accurately depict the variability of random ocean waves. Besides this approach, there are several nonlinear models for the interpretation of irregular waves. The simplest nonlinear model is the long-crested second-order model by Longuet-Higgins [46]. Other models include, for example, the second-order weakly nonlinear theory by Sharma and Dean [47] as an extension of the theory by Longuet-Higgins [46] since it also takes into account directional spreading, the hybrid wave model by Zhang *et al.* [48], and the Boussinesq model by Madsen *et al.* [49].

Based on the approach for linear superposition of regular wave components, the incident wave potential for a unidirectional irregular wave can be written as

$$\phi = \sum_{n=1}^N \frac{a_n g}{\omega_n} \frac{\cosh(k_n(z+h))}{\cosh k_n h} \sin(k_n x \cos \theta + k_n y \sin \theta - \omega_n t - \varepsilon_n) \quad (3.19)$$

where the subscript n indicates the values of the n -th wave component, N is the total number of wave components in the irregular wave, and a_n , ω_n , k_n , ε_n denote the wave

amplitude, the wave frequency, the wave number and the random phase lag of the n -th component, respectively. Also in Eq. (3.19), g , h , θ , t and (x, y) are the gravitational acceleration, the water depth, the angle of incidence (see Fig. 3.3), the time and the location in the domain where the potential is computed, respectively. Equation (3.19) requires the free surface elevation to be in the form of

$$\eta = \sum_{n=1}^N a_n \cos(k_n x \cos \theta + k_n y \sin \theta - \omega_n t - \varepsilon_n). \quad (3.20)$$

Here the wave number of each wave component k_n is related to the wave frequency of each wave component ω_n through the dispersion relation,

$$\omega_n^2 = k_n g \tanh k_n h. \quad (3.21)$$

In this case, the kinematic and dynamic properties of the wave should also be derived by the linear superposition:

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \sum_{n=1}^N \frac{a_n \omega_n}{\sinh k_n h} \begin{pmatrix} \cos \theta \cosh(k_n(z+h)) \cos(k_n x \cos \theta + k_n y \sin \theta - \omega_n t - \varepsilon_n) \\ \sin \theta \cosh(k_n(z+h)) \cos(k_n x \cos \theta + k_n y \sin \theta - \omega_n t - \varepsilon_n) \\ \sinh(k_n(z+h)) \sin(k_n x \cos \theta + k_n y \sin \theta - \omega_n t - \varepsilon_n) \end{pmatrix} \quad (3.22)$$

and

$$p = \sum_{n=1}^N \rho a_n g \frac{\cosh(k_n(z+h))}{\cosh k_n h} \cos(k_n x \cos \theta + k_n y \sin \theta - \omega_n t - \varepsilon_n) \quad (3.23)$$

where ρ denotes the fluid density. The above representation assumes that each component of the irregular wave is propagating under the same angle of incidence θ . Therefore, the wave crests will be, at least theoretically, infinitely long. In this case, randomness of the waves is introduced through the uniform distribution of the phase angles ε_n between 0 and 2π . The values of the phase angles are independent of each other and of the random component amplitudes a_n . Chakrabarti [50] states that the total number of wave components N should be sufficiently large (200-1000) to ensure randomness.

Directional (short-crested) random waves bear stronger resemblance to real sea states in nature than unidirectional random waves simply because sea waves are three-dimensional, see Fig. 3.2 for an example of each wave type. Looking out at the sea, we rarely observe waves traveling in a specific direction. On the contrary, the sea surface is rather chaotic with waves of different characteristics propagating in different directions. Disregarding this very dominant physical behavior may limit the accuracy of synthesizing random seas in reality. Furthermore, modeling wave directionality is of critical importance for the design of marine systems such as ships and offshore structures as the interaction effect between waves from various directions can become significant [51].

Analogous with the above discussion, directional irregular waves can also be expressed as a linear superposition of a finite number of monochromatic regular waves with different frequencies and propagation directions. Typically, the linear superposition can be performed by either the single or double summation model. These two models are described next.

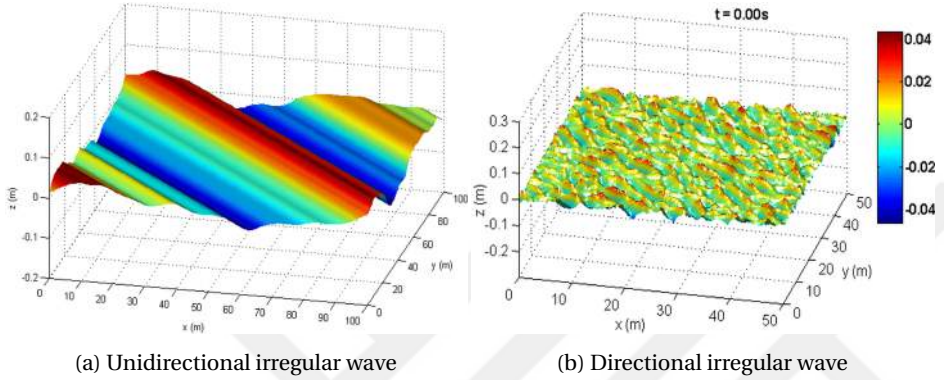


Figure 3.2: Free surfaces of two irregular waves: Unidirectional and directional.

Single summation model

In the single summation model first proposed by Jefferys [52], at each discrete frequency the wave component has its own propagation direction. When these components with different propagation directions and frequencies are superimposed, the expressions for the resulting confused sea become similar to those of the unidirectional random sea discussed above. The only difference is the additional dependence on the propagation direction. The general form of the wave profile for such a wave can be given as

$$\eta = \sum_{n=1}^N a_n \cos(k_n x \cos \theta_n + k_n y \sin \theta_n - \omega_n t - \varepsilon_n) \quad (3.24)$$

in which θ_n is the propagation direction of the n -th component. This mathematically simple addition results in a fundamental change in the simulation of random sea: lengths of wave crests are now limited. As a result, the generated sea with each wave component having different frequency and direction becomes more complex and realistic.

Double summation model

In the double summation model introduced by Miles and Funke [53], a finite number of propagation directions is assigned to each frequency of wave component. The resulting wave profile is hence written as

$$\eta = \sum_{n=1}^N \sum_{m=1}^M a_{nm} \cos(k_n x \cos \theta_m + k_n y \sin \theta_m - \omega_n t - \varepsilon_{nm}) \quad (3.25)$$

where the subscripts n and m denote the values of the n -th frequency and m -th direction, respectively, and N and M stand for the total number of wave components in frequency and direction, respectively. Compared to Eq. (3.24), this expression contains an inner summation which indicates that every wave component can have many propagation directions at each discrete frequency.

Miles *et al.* [54] state that the resulting wave field from the single summation model is spatially uniform for the durations that are generally used in experimental tests. On

the other hand, the double summation model generates wave fields that may form a partial standing wave pattern, which is caused by a widely-known phenomenon called *phase-locking*. Thus, the resulting wave field is no longer spatially homogeneous in the x - y plane, or may become homogeneous for very long record durations. Therefore, in the present work, the single summation model is employed. [Pascal and Bryden \[55\]](#) also briefly discuss this issue and prefer the single-summation model in order to avoid *phase-locking*. For a discussion on the relative merits of these models, see [\[52, 53, 56\]](#). For more information on both models, the reader is referred to the comprehensive study by [Hawkes et al. \[57\]](#).

3.4. DESIGN OF THE ABC

3.4.1. STATEMENT OF THE PROBLEM

In ComFLOW, the equations of motion are solved in a computational domain Ω (see Fig. 3.3) via imposing various types of boundary conditions such as free surface, wall, inflow and absorbing boundary conditions.

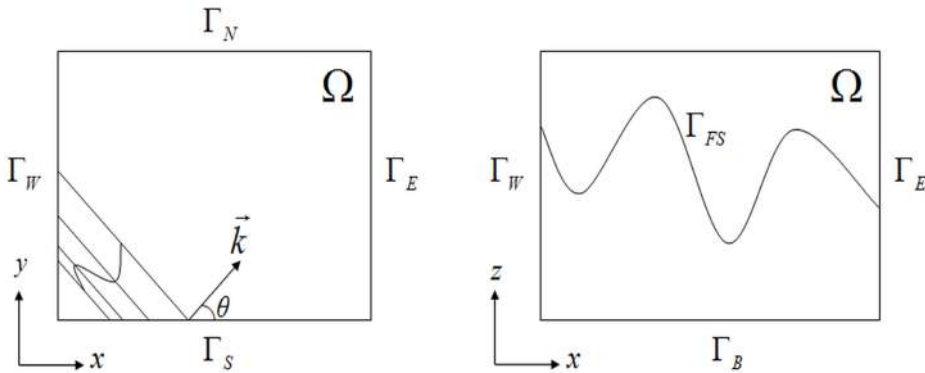


Figure 3.3: A computational domain with Γ_N and Γ_E as artificial boundaries.

Typically on the west and south boundaries Γ_W and Γ_S the incoming wave is prescribed. At every time level starting from $t = 0$, free surface elevations, values of the velocity components and pressure corresponding to the considered wave model are provided on Γ_W and Γ_S . At the bottom Γ_B we specify a no-slip no-penetration condition, which is simply the Dirichlet condition. At the free surface Γ_{FS} the continuity of normal and tangential stresses results in expressions for the velocity components and pressure. In-depth analysis of how these equations are implemented is beyond the scope of this chapter. For that purpose, the reader is referred to Chapter 2. We now introduce two artificial boundaries Γ_N and Γ_E , see Fig. 3.3. To complete the statement of the problem, we will implement an ABC on these artificial boundaries.

3.4.2. ABC-1 (DISPERSIVE ABC)

Consider the following boundary operator on Γ_E :

$$\left(\cos \alpha \frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right) \phi = 0. \quad (3.26)$$

Higdon [12] showed that (3.26) is perfectly absorbing if α is equal to the angle of incidence θ (see Fig. 3.3) for a wave described by the wave or velocity potential ϕ and traveling with phase speed c .

If we replace c in (3.26) by the dispersion relation, namely,

$$c = \sqrt{gh} \sqrt{\frac{\tanh(kh)}{kh}}, \quad (3.27)$$

we can rewrite (3.26) as

$$\left(\cos \alpha \frac{\partial}{\partial t} + \sqrt{gh} \sqrt{\frac{\tanh(kh)}{kh}} \frac{\partial}{\partial x} \right) \phi = 0. \quad (3.28)$$

The boundary condition (3.28) is perfectly absorbing for this single component, but recall that a wave is sometimes formed by superposition of a number of components. Each individual component of this wave has its own frequency, amplitude, wave number and phase. Therefore, the boundary condition (3.28) cannot annihilate all these wave components simply because it is evidently designed for only one of them.

At this point a question crosses one's mind: Is it possible to develop a boundary condition that allows reflection only to an acceptable threshold for all the components which all together form an irregular wave? One can deduce from the way this question is asked that we expect some amount of reflection for such a boundary condition, but it will be restricted within certain limits.

Now we introduce the following rational expression, which approximates the dispersion relation (3.27),

$$c_a \approx \sqrt{gh} \frac{a_0 + a_1(kh)^2}{1 + b_1(kh)^2}, \quad (3.29)$$

where a proper choice of coefficients a_0 , a_1 and b_1 leads to a close approximation for the largest possible range of kh values, see Fig. 3.4. The difference between the two curves gives an indication for the amount of reflection caused by the rational approximation. Although the accuracy of the approximation decreases for high wave numbers, eventual reflections at these wave numbers will be damped by other physical mechanisms.

Now a further improvement is introduced into the design of the boundary condition: by exploiting the approximately exponential behavior of the wave potential in the z -direction, the wave number k is computed *locally* from the potential itself. After straightforward algebraic manipulations, we can obtain the following relation

$$k^2 \phi = \frac{\partial^2}{\partial z^2} \phi. \quad (3.30)$$

By employing (3.30) there is no need to choose a value for k before the numerical simulation since it is calculated during the course of the simulation. Finally we substitute

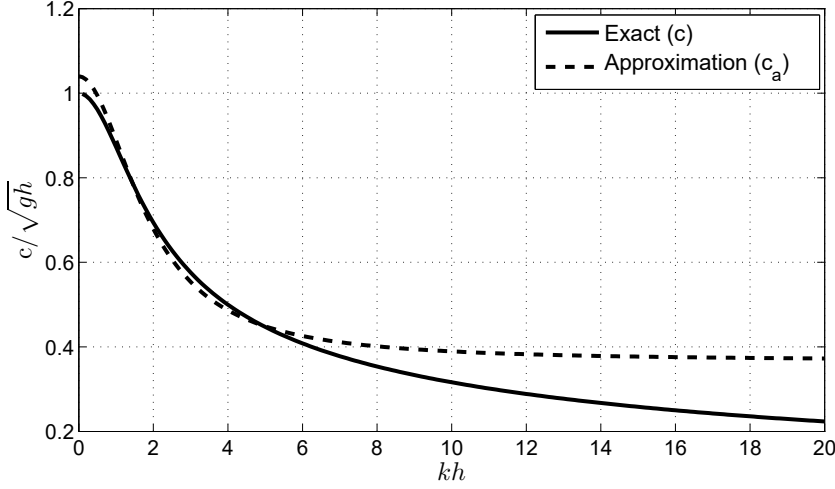


Figure 3.4: Approximation of the dispersion relation. For the coefficients in (3.29), $a_0 = 1.04$, $a_1 = 0.106$ and $b_1 = 0.289$ are used.

(3.30) and (3.29) in (3.26) to reach the final form of the absorbing boundary condition to be applied on Γ_E

$$\cos \alpha \left(1 + b_1 h^2 \frac{\partial^2}{\partial z^2} \right) \frac{\partial \phi}{\partial t} + \sqrt{gh} \left(a_0 + a_1 h^2 \frac{\partial^2}{\partial z^2} \right) \frac{\partial \phi}{\partial x} = 0. \quad (3.31)$$

Following the same method it is easy to write the ABC on Γ_N .

3.4.3. ABC-2 (DISPERSIVE DIRECTIONAL ABC)

Now, a further modification of the dispersive ABC will be discussed to account for both dispersive and directional effects of the waves. As it was already illustrated in Fig. 3.1, the 2nd-order Higdon ABC has superior performance over the 1st-order in terms of directional effects. Therefore, we will incorporate the improvements made in the previous section concerning dispersive effects into the 2nd-order Higdon ABC. We can write this ABC for Γ_E as the following:

$$\prod_{i=1}^2 \left(\cos \alpha_i \frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right) \phi = 0 \quad (3.32)$$

where $|\alpha_i| < \pi/2$ for all i . The condition (3.32) is satisfied exactly by any plane wave ϕ traveling out of the domain at angles of incidence α_1 and α_2 with the phase speed c . If we expand Eq. (3.32), we will have

$$\left(\cos \alpha_1 \frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right) \left(\cos \alpha_2 \frac{\partial \phi}{\partial t} + c \frac{\partial \phi}{\partial x} \right) = 0. \quad (3.33)$$

Considering the relations (3.29), (3.30) and (3.33), we realize that only one of the operators can include the approximation for the dispersion relation. Otherwise, the product

of two approximations would yield a fourth-order derivative in the z -direction, which will cause difficulties when discretized at the boundaries. Therefore, we substitute the relations (3.29) and (3.30) in one of the operators. The resulting expression for the ABC-2 is the following

$$\left(\cos \alpha_1 \frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right) \left(\left(1 + b_1 h^2 \frac{\partial^2}{\partial z^2} \right) \cos \alpha_2 \frac{\partial \phi}{\partial t} + \sqrt{gh} \left(a_0 + a_1 h^2 \frac{\partial^2}{\partial z^2} \right) \frac{\partial \phi}{\partial x} \right) = 0. \quad (3.34)$$

Analogous with the discussion for the ABC-1, this expression can also be easily written for the boundary operator to be applied on Γ_N . In what follows, we will demonstrate the effectiveness of the four boundary conditions, Eqs. (3.26), (3.31), (3.32) and (3.34), by displaying the amount of reflection as a function of both the angle of incidence θ and the dimensionless wave number kh .

3.4.4. REFLECTION COEFFICIENTS OF THE ABSORBING BOUNDARY CONDITIONS

Substituting the potential function (3.9) into the four boundary conditions, namely Eqs. (3.26), (3.31), (3.32) and (3.34), yields the following expressions for the reflection coefficient

$$R_{Higdon-1} = - \frac{c^{out} \cos \alpha - c \cos \theta}{c^{out} \cos \alpha + c \cos \theta}, \quad (3.35)$$

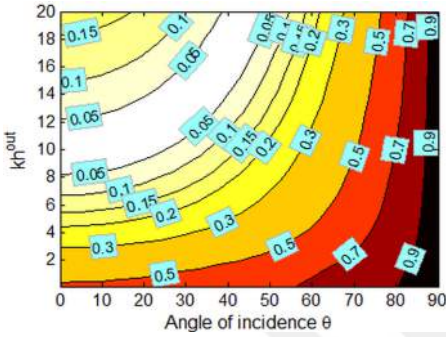
$$R_{Higdon-2} = - \prod_{i=1}^2 \left(\frac{c^{out} \cos \alpha_i - c \cos \theta}{c^{out} \cos \alpha_i + c \cos \theta} \right), \quad (3.36)$$

$$R_{ABC-1} = - \frac{(1 + b_1 (kh)^2) c^{out} \cos \alpha - \sqrt{gh} (a_0 + a_1 (kh)^2) \cos \theta}{(1 + b_1 (kh)^2) c^{out} \cos \alpha + \sqrt{gh} (a_0 + a_1 (kh)^2) \cos \theta}, \quad (3.37)$$

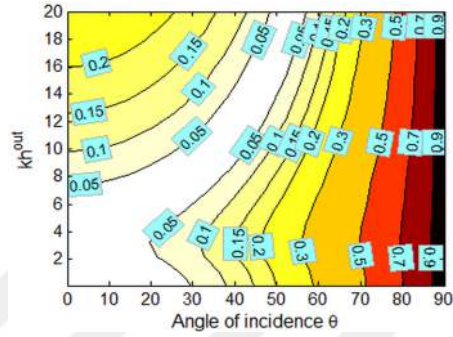
$$R_{ABC-2} = - \left[\frac{c^{out} \cos \alpha_1 - c \cos \theta}{c^{out} \cos \alpha_1 + c \cos \theta} \right] \left[\frac{c^{out} \cos \alpha_2 - \sqrt{gh} \frac{(a_0 + a_1 (kh)^2)}{(1 + b_1 (kh)^2)} \cos \theta}{c^{out} \cos \alpha_2 + \sqrt{gh} \frac{(a_0 + a_1 (kh)^2)}{(1 + b_1 (kh)^2)} \cos \theta} \right]. \quad (3.38)$$

Here c^{out} denotes the phase speed of the outgoing wave, whereas c , which approximates c^{out} , is a parameter in the operators. If c and c^{out} are equal to each other, then the reflection coefficient will depend only on wave direction. In this case if the wave direction θ is equal to the angle parameter in the operators α , then all the operators will be perfectly absorbing. This suggests that since both the effects of wave direction and phase speed contribute to the amount of reflection, we need to possess information *a priori* about both parameters for small reflection. In many practical situations, however, we unfortunately have information on neither of them at outflow boundaries.

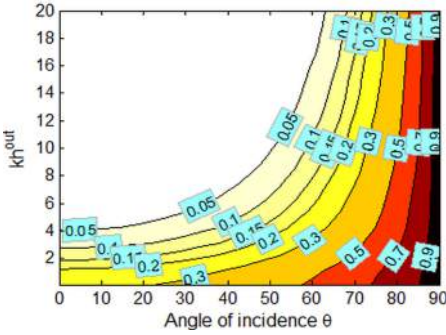
Figure 3.5 demonstrates the amount of reflection as a function of the angle of incidence θ and the dimensionless wave number kh to compare the effectiveness of the four boundary conditions; the 1st and 2nd-order Higdon operators, ABC-1 and ABC-2. For the coefficients in (3.29) we take $a_0 = 1.04$, $a_1 = 0.106$ and $b_1 = 0.289$ (see Section 3.4.5) while respecting the stability limits concerning these coefficients and providing a good approximation for the kh values from 0 to 20. We also take $c = 0.316 \sqrt{gh}$ in the boundary conditions in which this parameter appears. This choice corresponds to the



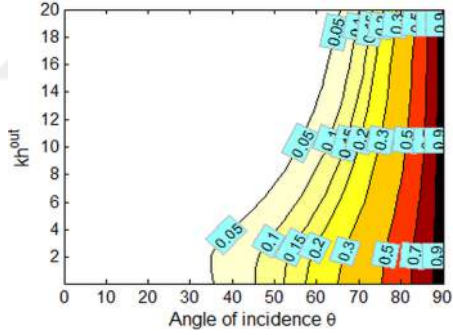
(a) First-order Higdon in Eq. (3.26).
Corresponding reflection coefficient is Eq. (3.35).



(b) ABC-1 in Eq. (3.31). Corresponding
reflection coefficient is Eq. (3.37).



(c) Second-order Higdon in Eq. (3.32).
Corresponding reflection coefficient is Eq. (3.36).



(d) ABC-2 in Eq. (3.34). Corresponding
reflection coefficient is Eq. (3.38).

Figure 3.5: Amount of reflection (in percent) as a function of the angle of incidence θ and kh when various boundary conditions are used. In the first-order boundary conditions $\alpha = 0$ is used, and in the second-order boundary conditions $\alpha_1 = \alpha_2 = 0$ is used. Also, $c = 0.316\sqrt{gh}$ is taken in all the boundary conditions. Reflection values are written on the contour lines.

kh value of 10, which is exactly in the middle of the considered kh range between 0 and 20. All the angle parameters in the boundary conditions are set to be zero as an arbitrary initial choice.

In Fig. 3.5 light-colored areas between isolines illustrate small reflection zones while as the color gets darker the amount of reflection increases since grazing incidence is approached where waves propagate parallel to the outflow boundary. More particularly, white-colored zones demonstrate the ranges where the reflection coefficient is below five percent, which is an acceptable threshold in practical wave simulations. Close examination of the results clearly shows the relative merits of the ABC-1 and ABC-2 over the first-order and second-order Higdon operators. The first-order Higdon operator generates reflection below five percent between kh values of 8 to 12 approximately,

which is expected since the pre-specified c value corresponds to the kh value of 10. However, this range is between 0 to 8 with the ABC-1. The reason for this behavior lies in the way the dispersion relation is approximated. The area between the two curves in Fig. 3.4 gives an indication of the amount of reflection caused by the approximation. From this plot, we clearly see that the dispersion relation is approximated well in the range $kh \in (0, 8]$. From 8 to 20, the reflection increases as the approximation becomes gradually poor, which is shown in Fig. 3.5b. In terms of directional effects, the performance of the ABC-1 is somewhat better than that of the first-order Higdon operator.

The second-order Higdon operator improves both the dispersive and directional behavior of the first-order operator. This is a result of the fact that the reflection coefficient of the second-order Higdon operator (3.36) is a product of two factors. Since each of these factors is smaller than 1, their product becomes even smaller, and areas indicating small reflections increase, see Fig. 3.5c. However, between kh values of 0 to 4, the second-order Higdon operator has a poor performance. This is where the approximation for the dispersion relation once again plays a role, and the ABC-2 consequently eliminates this poor performance.

The distinguishing characteristic of generating small reflection for a range of both kh values and angles of incidence reveals the reason of applying the approximation for the dispersion relation into the second-order Higdon operator: we have improved the dispersive behavior of the operator through the approximation for the dispersion relation while the operator itself has already favorable behavior in terms of directional effects. Thus, the ABC-2 becomes a versatile and accurate boundary condition for various sea states that possess both dispersive and directional features.

3.4.5. STABILITY

Stability of a numerical scheme which includes temporal and spatial discretization of the equations of motion is always a source of concern, and is usually satisfied through several criteria such as the well-known CFL condition. In addition to this issue, we also need to examine the stability of an absorbing boundary condition since implementation of such condition can also render a numerical simulation unstable.

Stability of an absorbing boundary condition stipulates that an unwanted mode growing in time is not allowed by the boundary condition. This can be verified by analyzing the behavior of the solution modes to the wave equation: propagating, evanescent and spurious modes. Both propagating and evanescent modes display periodic behavior in time. However, spurious modes display exponential behavior in time, which can lead to unstable simulations if they are allowed to grow. For stability, it is required that the absolute value of the reflection coefficient of this mode to always be less than 1. To satisfy this requirement, Wellens [27] investigated the behavior of this mode when combined with the first-order Engquist-Majda boundary condition, and proposed the following criteria for the parameters in (3.29)

$$\begin{aligned} \frac{a_0}{\pi^2} &< a_1 < \frac{4a_0}{\pi^2}, \\ a_1 &< b_1 < \frac{4}{\pi^2}. \end{aligned} \tag{3.39}$$

Following in the footsteps of this work, we will show that the criteria (3.39) can safely

be used for ABC-2 as well. We start the discussion by first obtaining the reflection coefficient of the ABC-2 for a spurious mode. Wellens [27] states that the dispersion relation of a spurious mode can be given as

$$c_{sm} = \sqrt{gh} \sqrt{\frac{\tan(kh)}{kh}}. \quad (3.40)$$

This expression can be approximated by the following relation

$$c_{asm} = \sqrt{gh} \frac{a_0 - a_1(kh)^2}{1 - b_1(kh)^2}. \quad (3.41)$$

Upon substitution of Eqs. (3.40) and (3.41) into the reflection coefficient of the ABC-2, we reach the following expression for the reflection coefficient of a spurious mode after some algebraic manipulations

$$R_{sm} = - \left[\frac{\sqrt{\frac{\tan(kh)}{kh}} \cos \alpha_1 - c \cos \theta}{\sqrt{\frac{\tan(kh)}{kh}} \cos \alpha_1 + c \cos \theta} \right] \left[\frac{\sqrt{\frac{\tan(kh)}{kh}} \cos \alpha_2 - \frac{(a_0 - a_1(kh)^2)}{(1 - b_1(kh)^2)} \cos \theta}{\sqrt{\frac{\tan(kh)}{kh}} \cos \alpha_2 + \frac{(a_0 - a_1(kh)^2)}{(1 - b_1(kh)^2)} \cos \theta} \right]. \quad (3.42)$$

The requirement for stability can then be stated as $|R_{sm}| < 1$. Analysis of this expression shows that the criteria (3.39) are valid for stability of the ABC-2. Without violating these inequalities, we can approximate the dispersion relation corresponding to the characteristics of the considered wave.

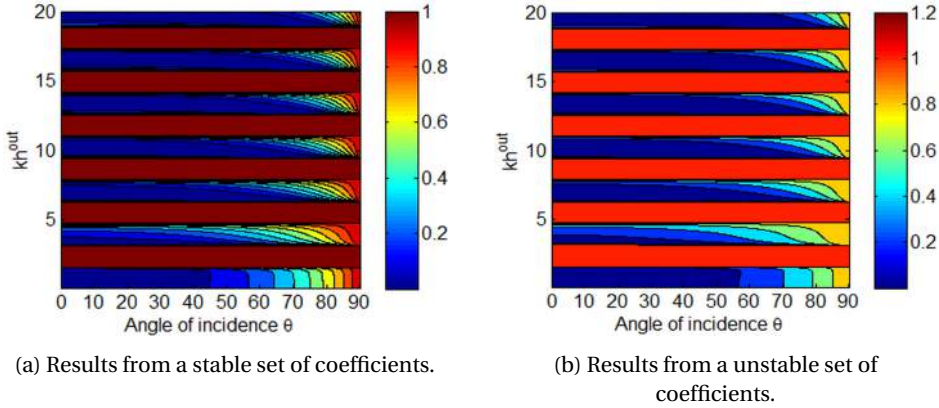


Figure 3.6: Amount of reflection (in percent) of a spurious mode as a function of the angle of incidence θ and kh when ABC-2 is used with two sets of coefficients: a stable set ($a_0 = 1.05$, $a_1 = 0.12$ and $b_1 = 0.31$), and an unstable set ($a_0 = 1.05$, $a_1 = 0.095$ and $b_1 = 0.31$). In both cases $\alpha_1 = \alpha_2 = 0$ and $c = 0.316\sqrt{gh}$ are used.

Figure 3.6 shows an example illustrating the amount of reflection of a spurious mode when the ABC-2 is used as a function of the angle of incidence θ and dimensionless wave number kh . Here we used two sets of coefficients: the first is a stable set ($a_0 = 1.05$,

$a_1 = 0.12$ and $b_1 = 0.31$), and the second one is an unstable set ($a_0 = 1.05$, $a_1 = 0.095$ and $b_1 = 0.31$) in which the first criterion in (3.39) is violated. We clearly see that the reflection coefficient becomes larger than 1.0 with the unstable set of coefficients, which indicates that we may observe an unphysical increase in energy in the vicinity of the outflow boundary where the ABC-2 is applied. With the stable set of coefficients, this possibility is altogether avoided.

3.4.6. FROM ABC TO GABC (GENERATING AND ABSORBING BC)

Sometimes it is necessary to radiate out an outgoing wave while allowing an incoming wave to propagate into the domain through the same boundary. A boundary treatment that provides such feature becomes a fully transparent boundary condition, which is also referred to as Open Boundary Condition (OBC) in the literature [58, 59]. In order to preserve consistency with the previous work in this field throughout the ComFLOW project, we will refer to it as Generating and Absorbing Boundary Condition (GABC). Carpenter [60] proposed such a method,

$$\left(\frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right) \phi = \left(\frac{\partial}{\partial t} + c \frac{\partial}{\partial x} \right) \phi^{in}. \quad (3.43)$$

Here ϕ^{in} is the external function known *a priori*, and x is the outward normal direction to the boundary where the condition is applied. Although the original paper was actually a 2-page note and did not include detailed analysis, this boundary condition has been used by a number of researchers including [59, 61–63]. van der Meer *et al.* [64] used the same idea in SKYLLA, a code developed at Delft Hydraulics (In 2008 Delft Hydraulics merged with GeoDelft establishing the new independent Dutch institute, Deltares, which participated in the ComFLOW project). A brief discussion on the method was also presented in [58, 65, 66]. The fact that Carpenter [60] used the same Sommerfeld operator on each side of the equation is simple and appealing. However, the Sommerfeld operator has strict limitations that effectively render the Carpenter scheme low order accurate. Mar-Or and Givoli [59] analyzed the Carpenter scheme for the one-dimensional linear shallow water equations. Later they improved the accuracy by replacing the Sommerfeld operator with the high-order Hagstrom-Warburton operator [19], which makes use of auxiliary functions, for a non-dispersive uniform medium [67]. Following that work they made a further extension to the method and applied it for a dispersive non-uniform medium [68].

Wellens [27] improved the Carpenter scheme in terms of dispersive effects by replacing the phase speed c with Eqs. (3.29) and (3.30), and called the boundary condition GABC. The GABC allows long-crested dispersive waves to propagate in and out through the same boundary as long as the range of accuracy of Eq. (3.29) is respected.

In this study we improve the GABC further by implementing Eqs. (3.29) and (3.30) into the Higdon operators instead of the Sommerfeld operator. Hence we will use the ABC-1 or ABC-2 in the Carpenter scheme, and design a boundary condition to account for short-crested dispersive waves traveling in and out of the computational domain over the same boundary. Since the same operator is used on each side of Eq. (3.43), we will consider only one side in the following section where the numerical discretization of the boundary conditions is explained.

3.4.7. NUMERICAL DISCRETIZATION OF THE ABC-1 AND ABC-2

Since the governing equations inside the computational domain are written in terms of the velocity components and pressure, the ABCs (3.31) and (3.34) must be interpreted using the same variables. As we have a staggered grid arrangement for the flow variables inside grid cells (see Fig. 3.7), the location of the outflow boundary must also be specified appropriately.

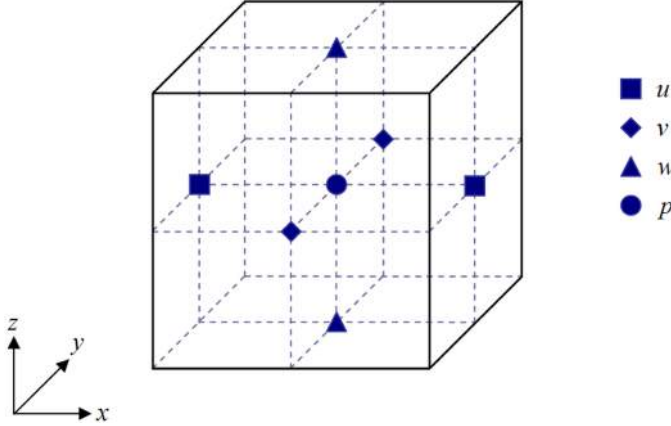


Figure 3.7: Locations of the velocity components and pressure in a grid cell based on the staggered grid arrangement.

All flow variables that we need here can be calculated by taking derivatives of the wave potential. Exploiting the potential theory and linearized Bernoulli equation, it is possible to obtain the following expressions for the derivatives of the wave potential with respect to time and space

$$\frac{\partial \phi}{\partial x} = u_b, \quad (3.44)$$

$$\frac{\partial \phi}{\partial t} = -\frac{p_b}{\rho} - g z_p. \quad (3.45)$$

In Eqs. (3.44) and (3.45), subscript b indicates that the quantity is defined at the boundary, subscript p indicates that the quantity is evaluated at the elevation of the pressure point, and for simplicity the fluid density is taken to be one. We see that both the velocity and pressure are defined at the same position in space in order to avoid any phase shift errors between flow variables. By the same logic, these variables are also defined at the same instant in time, which is the new time level t^{n+1} .

The outflow boundary is situated at the same position as u_l along the x -direction; therefore we can impose

$$u_{b,k} = u_{l,k}^{n+1} \quad (3.46)$$

for $k = 1, \dots, K$. In order to obtain p_b , however, we employ linear interpolation and introduce mirror cells adjacent to the outflow boundary (see Fig. 3.8), i.e.,

$$p_{b,k} = \frac{p_{l,k}^{n+1} + p_{l+1,k}^{n+1}}{2} \quad (3.47)$$

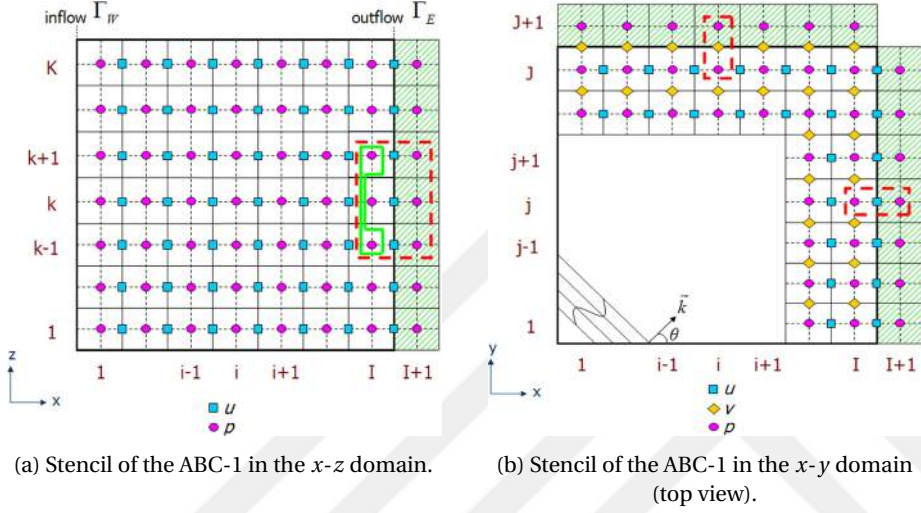


Figure 3.8: Stencil of the ABC-1 in space

for $k = 1, \dots, K$. The shaded areas around the computational domain illustrated in Fig. 3.8 contain the mirror cells.

In order to plug the ABC-1 into the pressure Poisson equation (2.36), which is solved inside the computational domain for the pressure at the new time step p^{n+1} , the velocity component at the new time step u^{n+1} should be eliminated. Utilizing the discrete form of the x-momentum equation (2.34), we have

$$u_{I,k}^{n+1} = \tilde{u}_{I,k}^n - \frac{\Delta t}{\Delta x_{p_{I+1}}} \left(p_{I+1,k}^{n+1} - p_{I,k}^{n+1} \right), \quad (3.48)$$

u^{n+1} is written in terms of the pressure p^{n+1} and the auxiliary velocity $\tilde{u}^n$ (2.35). As a result, the ABC-1 will have the same temporal character as the pressure Poisson equation. Consequently, we obtain the discrete form of the ABC-1 to be prescribed on Γ_E as follows

$$\begin{aligned} & \left[\frac{1}{2} \cos \alpha + a_0 \sqrt{gh} \frac{\Delta t}{\Delta x_{p_{I+1}}} + \left(\frac{1}{2} b_1 h^2 \cos \alpha + a_1 h^2 \sqrt{gh} \frac{\Delta t}{\Delta x_{p_{I+1}}} \right) \frac{\partial^2}{\partial z^2} \right] p_{I+1,k}^{n+1} \\ & + \left[\frac{1}{2} \cos \alpha - a_0 \sqrt{gh} \frac{\Delta t}{\Delta x_{p_{I+1}}} + \left(\frac{1}{2} b_1 h^2 \cos \alpha - a_1 h^2 \sqrt{gh} \frac{\Delta t}{\Delta x_{p_{I+1}}} \right) \frac{\partial^2}{\partial z^2} \right] p_{I,k}^{n+1} \\ & = \left(a_0 \sqrt{gh} + a_1 h^2 \sqrt{gh} \frac{\partial^2}{\partial z^2} \right) \tilde{u}_{I,k}^n - g z_{p_k} \cos \alpha \end{aligned} \quad (3.49)$$

for $k = 1, \dots, K$ where $\Delta x_{p_{I+1}} = x_{p_{I+1}} - x_{p_I}$. Equation (3.49) does not contain any flow variables from the neighboring cells along the y-direction. Therefore, for simplicity, we ignored the subscript j for the flow variables here and elsewhere. For the approximation

of the second-order derivative in the z -direction, the following relation designed for a stretched grid is employed,

$$\frac{\partial^2 p_k}{\partial z^2} = \frac{\Delta z_{p_k} p_{k+1} - (\Delta z_{p_{k+1}} + \Delta z_{p_k}) p_k + \Delta z_{p_{k+1}} p_{k-1}}{\frac{1}{2} \Delta z_{p_{k+1}} \Delta z_{p_k} (\Delta z_{p_{k+1}} + \Delta z_{p_k})} \quad (3.50)$$

where $\Delta z_{p_{k+1}} = z_{p_{k+1}} - z_{p_k}$, and Δz_{p_k} is calculated similarly. Veldman [69] shows that the approximation (3.50) is second-order accurate on smoothly stretched grids.

Following the same steps, we can easily derive the ABC-1 on Γ_N . The discrete ABC-1 on Γ_E and Γ_N is a set of equation for the pressure values in the mirror cells adjacent to the computational domain, see Fig. 3.8. The addition of these mirror cells to the domain actually represents the cost of implementing the ABC-1. In fact, the ABC-1 is solved for pressure values in a total number of $(I + J) \times K$ cells. The stencil for $p_{I+1,k}^{n+1}$ is plotted by a red dashed line in Fig. 3.8a. Observing Figs. 3.8a and 3.8b we realize that a typical stencil for a pressure point encompasses 9 flow variables, 6 of which reside in the computational domain whereas 3 can be associated with the treatment of the boundary condition.

Analogous with the discretization for the ABC-1, we follow the same steps to obtain the discrete form of the ABC-2. The derivatives of the pressure with respect to time and space are taken as

$$\frac{\partial p_b}{\partial t} = \frac{p_b^{n+1} - p_b^n}{\Delta t}, \quad (3.51)$$

$$\frac{\partial p_b}{\partial x} = \frac{(p_{I+1,k} - p_{I,k})^{n+1}}{\Delta x_{p_{I+1}}}. \quad (3.52)$$

Using the linear interpolation of the pressure values on either side of the boundary, we can have

$$\begin{aligned} p_b^{n+1} &= \frac{1}{2} (p_{I+1,k} + p_{I,k})^{n+1}, \\ p_b^n &= \frac{1}{2} (p_{I+1,k} + p_{I,k})^n. \end{aligned} \quad (3.53)$$

Substituting (3.53) into (3.51) results in the following expression for the final form of (3.51)

$$\frac{\partial p_b}{\partial t} = \frac{(p_{I+1,k} + p_{I,k})^{n+1} - (p_{I+1,k} + p_{I,k})^n}{2\Delta t}. \quad (3.54)$$

Similarly, the derivatives of the velocity component with respect to time and space are obtained by

$$\frac{\partial u_b}{\partial t} = \frac{u_{I,k}^{n+1} - u_{I,k}^n}{\Delta t}, \quad (3.55)$$

$$\frac{\partial u_b}{\partial x} = \frac{(u_{I,k} - u_{I-1,k})^{n+1}}{\Delta x_{u_I}} \quad (3.56)$$

where $\Delta x_{u_I} = x_{u_I} - x_{u_{I-1}}$. The form of the pressure Poisson equation (2.36) allows only the pressure to be at the new time level. Therefore, the velocity values at the new time

level in Eqs. (3.55) and (3.56) need to be eliminated. For this purpose, we resort to the discrete momentum equation

$$\begin{aligned} u_{I,k}^{n+1} &= \tilde{u}_{I,k}^n - \frac{\Delta t}{\Delta x_{p_{I+1}}} (p_{I+1,k}^{n+1} - p_{I,k}^{n+1}), \\ u_{I-1,k}^{n+1} &= \tilde{u}_{I-1,k}^n - \frac{\Delta t}{\Delta x_{p_I}} (p_{I,k}^{n+1} - p_{I-1,k}^{n+1}). \end{aligned} \quad (3.57)$$

Upon substitution of (3.57) into (3.55) and (3.56), we have

$$\frac{\partial u_b}{\partial t} = \frac{(\tilde{u}_{I,k} - u_{I,k})^n}{\Delta t} - \frac{(p_{I+1,k} - p_{I,k})^{n+1}}{\Delta x_{p_{I+1}}}, \quad (3.58)$$

$$\frac{\partial u_b}{\partial x} = \frac{(\tilde{u}_{I,k} - \tilde{u}_{I-1,k})^n}{\Delta x_{u_I}} - \frac{\Delta t}{\Delta x_{u_I}} \left(\frac{p_{I+1,k}^{n+1}}{\Delta x_{p_{I+1}}} - \frac{p_{I,k}^{n+1}}{\Delta x_{p_{I+1}}} - \frac{p_{I,k}^{n+1}}{\Delta x_{p_I}} + \frac{p_{I-1,k}^{n+1}}{\Delta x_{p_I}} \right). \quad (3.59)$$

If we substitute the above derivatives of pressure and velocity into (3.34), we obtain the final discrete form of the ABC-2 as follows

$$\begin{aligned} & \left[\chi + \varsigma \Delta t a_0 + \vartheta + \frac{\gamma a_0}{\Delta x_{p_{I+1}}} + \left(\chi b_1 + \varsigma \Delta t a_1 + \vartheta b_1 + \frac{\gamma a_1}{\Delta x_{p_{I+1}}} \right) \frac{h^2 \partial^2}{\partial z^2} \right] p_{I+1,k}^{n+1} \\ & + \left[\chi - \varsigma \Delta t a_0 - \vartheta - \gamma \xi a_0 + (\chi b_1 - \varsigma \Delta t a_1 - \vartheta b_1 - \gamma \xi a_1) \frac{h^2 \partial^2}{\partial z^2} \right] p_{I,k}^{n+1} \\ & + \left[\frac{\gamma a_0}{\Delta x_{p_I}} + \frac{\gamma a_1}{\Delta x_{p_I}} \frac{h^2 \partial^2}{\partial z^2} \right] p_{I-1,k}^{n+1} \\ & = \left[\chi + \chi b_1 \frac{h^2 \partial^2}{\partial z^2} \right] (p_{I+1,k} + p_{I,k})^n - \left[\varsigma a_0 \Delta x_{p_{I+1}} + \varsigma a_1 \Delta x_{p_{I+1}} \frac{h^2 \partial^2}{\partial z^2} \right] u_{I,k}^n \\ & + \left[\varsigma a_0 \Delta x_{p_{I+1}} + \frac{\gamma a_0}{\Delta t} + \left(\varsigma a_1 \Delta x_{p_{I+1}} + \frac{\gamma a_1}{\Delta t} \right) \frac{h^2 \partial^2}{\partial z^2} \right] \tilde{u}_{I,k}^n - \left[\frac{\gamma}{\Delta t} + \frac{\gamma a_1}{\Delta t} \frac{h^2 \partial^2}{\partial z^2} \right] \tilde{u}_{I-1,k}^n \end{aligned} \quad (3.60)$$

for $k = 1, \dots, K$ where

$$\chi = \frac{\cos \alpha_1 \cos \alpha_2}{2 \Delta t},$$

$$\vartheta = \frac{c \cos \alpha_2}{\Delta x_{p_{I+1}}},$$

$$\varsigma = \frac{\cos \alpha_1 \sqrt{g h}}{\Delta x_{p_{I+1}} \Delta t},$$

$$\gamma = \frac{c \Delta t \sqrt{g h}}{\Delta x_{u_I}},$$

$$\xi = \frac{1}{\Delta x_{p_I}} + \frac{1}{\Delta x_{p_{I+1}}}.$$

Analogous with the case for the ABC-1, Eq. (3.60) is in fact solved for the pressure values in the mirror cells adjacent to the computational domain. Figure 3.9 shows the stencil $p_{I+1,k}^{n+1}$ encompassing 15 solution variables keeping in mind that the second-order vertical derivative in the z -direction is approximated with the relation (3.50) using a three-point variable group.

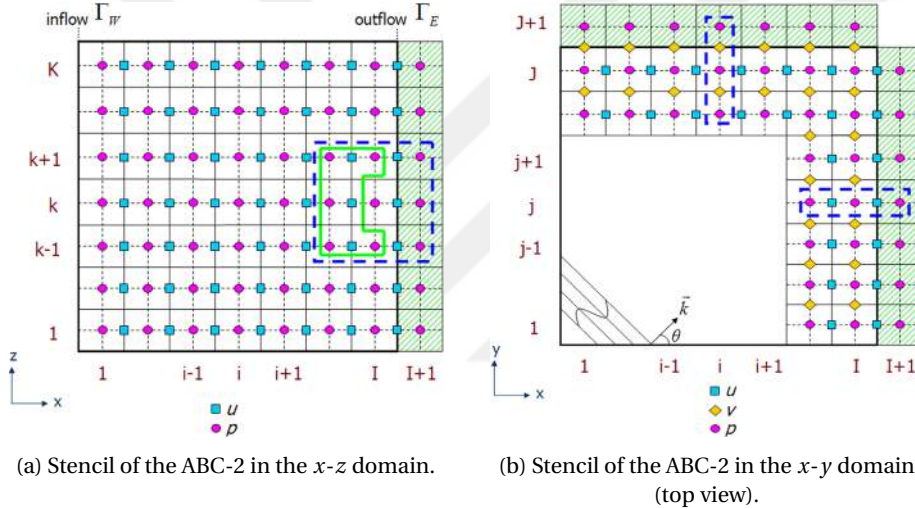


Figure 3.9: Stencil of the ABC-2 in space

Compared to that of the ABC-1, the stencil for the ABC-2 becomes larger and extends in the direction normal to the outflow boundary. However, the cost of implementing the ABC-2 is the same as the ABC-1: Eq. (3.60) is solved in the same total number of $(I+J) \times K$ cells. This is clearly advantageous compared to numerical dissipation zones such as damping layer, as the increase in the size of the computational domain is substantially small: one layer of mesh cells are added to the outflow boundaries of the computational domain in the case of the ABCs whereas large zones of several wavelengths are usually added to the domain when damping layer is employed [35]. Wei and Kirby [35] mention that the size of the damping layer is generally two or three times the wavelength. Considering previous numerical simulations of waves or wave-structure interactions where an absorbing boundary condition is applied in the domain [70, 71], the recommended size of the numerical damping zone is even larger than the actual computational domain itself. Wei and Kirby [35] also state that, for small reflection, numerical dissipation zones can be used in combination with absorbing boundary conditions, which increases computational cost even further.

The pressure values within the areas enclosed by the green lines in Figs. 3.8a and 3.9a do not exist in the standard stencil of the pressure Poisson equation. Hence, the large stencils of the ABCs are not suitable for a typical SOR solver [72] designed for the Poisson equation. To solve the pressure Poisson equation, therefore, we employed a Krylov subspace solver: BiCGSTAB. As preconditioner ILU(p) factorization is implemented, where the approximate LU-decomposition is chosen more accurate near the boundary.

3.5. RESULTS AND DISCUSSIONS

In this section we present results of the numerical experiments where the ABCs are applied at the outflow boundaries of the computational domain. In the first test, we will investigate the capability of the ABCs to account for only dispersive effects of the waves, and in the second test for only directional effects. In the third test, the ABCs will show a performance in terms of both, as we generate a short-crested wave in the computational domain. The final test will include an analysis with large-amplitude regular waves.

The results of the numerical simulations will be analyzed through error norms. For that purpose, we will use the following error measures,

$$e(i, j) = \eta_s(i, j) - \eta_r(i, j), \quad (3.61)$$

$$\|e\|_\infty = \max_{i=1,2,\dots,I} \max_{j=1,2,\dots,J} \{|\eta_s(i, j) - \eta_r(i, j)|\}, \quad (3.62)$$

where η is the free surface elevation. Here the subscript s indicates the solution in the domain of interest or 'small domain' and the subscript r indicates the reference solution. In the numerical experiments considered here, the reference solution is either obtained by solving the problem in a much larger domain with the same discretization in space and time or from analytical results. More information on the setup of the tests is given below.

The pointwise error $e(i, j)$ provides information at particular time instances throughout the simulation. In addition, it demonstrates the exact location of the error in the computational domain, which is not the case for the other error measure. The infinity norm $\|e\|_\infty$ captures the maximum value in the error vector, which is useful especially to check if a certain limit for the error is breached.

3.5.1. DISPERSIVE PROPERTIES OF THE ABCs

We apply the discrete ABCs to a problem in a two dimensional computational domain where a linear regular wave is traveling along the x -direction. The wave has a period of $T = 4s$ and a wave height of $H = 0.1m$. With this wave we choose four different water depths, which produces four different kh values: 5, 10, 15 and 20. The parameters in the ABCs are not tuned for a particular kh value; rather they are fixed so that the reflection coefficient is minimum for the range of kh values between 0 and 20. Therefore, this test becomes especially useful to investigate the dispersive character of the boundary conditions. Here we employ the values given in Fig. 3.5. Two uniform grid resolutions are considered: 0.5m and 0.25m.

The length of the long domain is arranged in such a way that the reflected waves can not pollute the solution in the small domain. This allows us to use the solution in the long domain as a reference solution. In order to ensure this requirement we use two pieces of information: the phase speed of the fastest propagating wave component $\sqrt{gh}$ and the duration of the simulation $t_{\max}$. The length of the long domain should be at least $\sqrt{gh}t_{\max}$ larger than the length of the small domain so that the reflected waves from the outflow boundaries of the long domain cannot reach the small domain.

Figure 3.10 shows the infinity norm $\|e\|_\infty$ (3.62) from the two ABCs on two grid resolutions. Errors are measured for a duration of $t_{\max} = 16s$ which corresponds to four wave periods. Both error measures are normalized by the wave height.

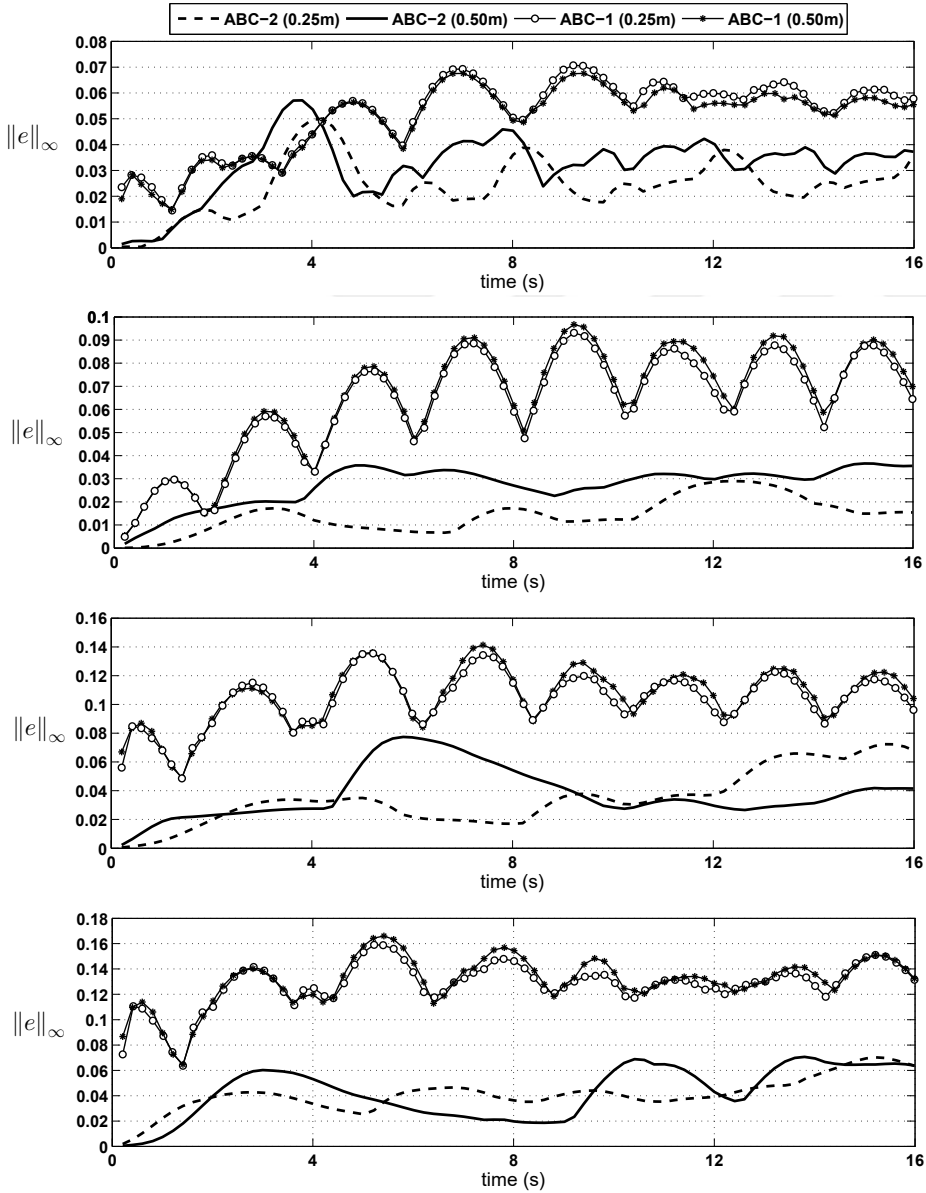


Figure 3.10: Infinity norms $\|e\|_\infty$ (3.62) from the two ABCs on two grid resolutions for waves with various kh values. From top to bottom results are shown for $kh = 5$, $kh = 10$, $kh = 15$ and $kh = 20$.

Results indicate that as kh values increase, the amount of reflection generated by ABC-1 increases from 7% for $kh = 5$ to nearly 16% for $kh = 20$. The amount of reflection due to the use of the ABC-1 is characterized by the approximation that we applied for the dispersion relation. As shown in Fig. 3.4 this approximation becomes poorer as kh increases, which corresponds to an increase in reflection. On the other hand, the largest value of $\|e\|_\infty$ changes between 4% and 8% when ABC-2 is used. This is slightly higher than the maximum theoretical reflection coefficient illustrated in Fig. 3.5d. The approximations that we applied for the temporal and spatial derivatives in ABC-2 introduce errors, which can be the cause of this difference. When the grid is refined, the results from ABC-1 change only slightly. With ABC-2 the change varies for different kh values: the infinity norm decreases 1% for $kh = 5$, 0.5% for $kh = 10$, 4% for $kh = 15$ and 2% for $kh = 20$. Examining the discrete forms of the ABCs reveals that ABC-2 is more sensitive to grid refinement as it contains more spatial and temporal derivative terms.

3.5.2. DIRECTIONAL PROPERTIES OF THE ABCS

In this numerical experiment, the ABCs are applied in a computational domain where a regular wave is propagating under an angle of incidence. To calculate the amount of reflection the simulations are performed in two computational domains: a large Ω_L and a small domain Ω_S . The size of the large domain Ω_L is greater than that of the small domain Ω_S only in the x - y plane, see Fig. 3.11 for the illustration of the problem. The lengths of Ω_S in the x - and y -directions are the same, $70m$. Similar to the previous test case, the size of Ω_L is defined in such a way that the reflected waves from the outflow boundaries, which are located at $x = 230m$ and $y = 230m$, do not reach Ω_S during the simulation. This procedure guarantees that in Ω_L , the solution in the part that has the size of Ω_S will not be perturbed by the reflected waves, and hence it can be used as a reference solution.

At $t = 0s$, a fully developed linear regular wave is generated in both domains. At each time level, the numerical solution in Ω_S is compared to the reference solution in Ω_L . In both domains all numerical parameters are the same. Therefore, we can attribute the difference between the two solutions only to the performance of the ABCs at the outflow boundaries of Ω_S .

In this test we consider a wave which has a period of $T = 5s$ and a wave height of $H = 0.1m$. The water depth is $h = 5.75m$. With this wave we choose four different angles of incidence: $\theta = 20^\circ$, $\theta = 40^\circ$, $\theta = 60^\circ$ and $\theta = 80^\circ$, all of which are measured in the counter-clockwise direction from the positive x -axis. Similar to the previous test the parameters of the ABCs will be kept constant for the four waves, thus the ABCs will not be tuned for a particular angle. The parameters are chosen in such a way that the reflection coefficient is minimum for the four angles of incidence. Thus we set $\alpha = 45^\circ$ for the ABC-1, and $\alpha_1 = 20^\circ$ and $\alpha_2 = 60^\circ$ for the ABC-2. The parameter c in the ABCs is chosen equal to the phase speed of the generated wave. In the simulations two uniform grid resolutions are considered: $0.5m$ and $0.25m$.

Figure 3.12 shows the infinity norm $\|e\|_\infty$ from the two ABCs on two grid resolutions. Errors are measured for a duration of $t_{\max} = 20s$ which corresponds to four wave periods. Both error measures are normalized by the wave height. Results indicate that ABC-1 produces the smallest reflection for the case with $\theta = 40^\circ$, i.e., the value of $\|e\|_\infty$ is around

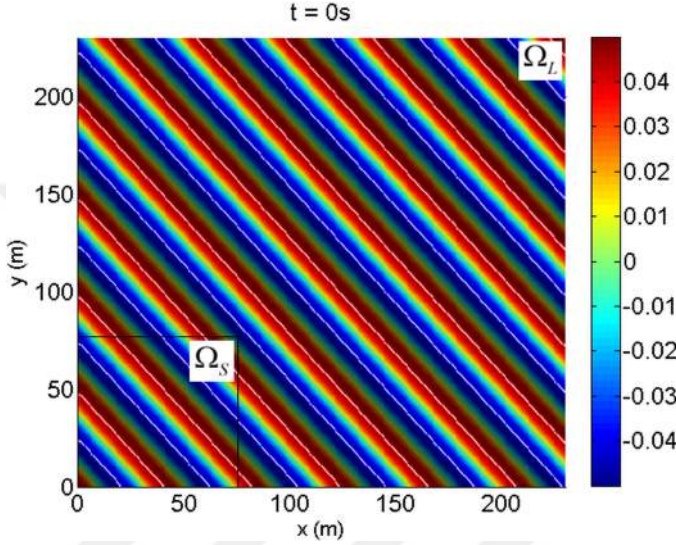


Figure 3.11: The setup for the tests where a regular wave is propagating under an angle of incidence. The solution in Ω_L is considered as the reference solution.

5% on the fine grid. This is expected since in this case θ is closest to $\alpha = 45^\circ$. As the angle of incidence differs more from α , reflection increases. This behavior was demonstrated in Fig. 3.1. $\|e\|_\infty$ has the values of 14% at $\theta = 60^\circ$, 23% at $\theta = 20^\circ$ and 33% at $\theta = 80^\circ$ from the ABC-1 on the fine grid. With the ABC-2 reflection is the smallest when θ is close to α_1 or α_2 . $\|e\|_\infty$ has the values of 8% at $\theta = 20^\circ$, 3% at $\theta = 60^\circ$, 9% at $\theta = 40^\circ$ and 23% at $\theta = 80^\circ$ from the ABC-2 on the fine grid.

Figures 3.13 and 3.14 demonstrate the pointwise error $\|e(i, j)\|$ (3.61), normalized with H and averaged over time, on the fine grid. Light-colored areas correspond to small amount of reflection; and as the color gets darker, the amount of reflection increases. We first observe that the reflected waves from the outflow boundaries travel back and perturb the solution in the computational domain. We also notice that large errors are local and generally close to the outflow boundaries, whereas we have relatively small errors in the major part of the domain. Analyzing Figs. 3.12 to 3.14 we conclude that the capability of the ABC-2 to account for directional effects is clearly superior to that of the ABC-1.

3.5.3. SHORT-CRESTED WAVE TEST

In this numerical experiment, the ABCs are applied in a computational domain where a directional irregular wave is generated. Similar to the previous cases the reference solution used in the error measures is computed by solving the problem in a vast domain Ω_L which is larger than the small domain Ω_S only in the x - y plane, see Fig. 3.15 for the illustration of the problem. Before, we showed the performances of the ABCs concerning the dispersive and directional effects of the waves separately. Here we combine the two effects and produce a test where the wave is composed of a large number of components

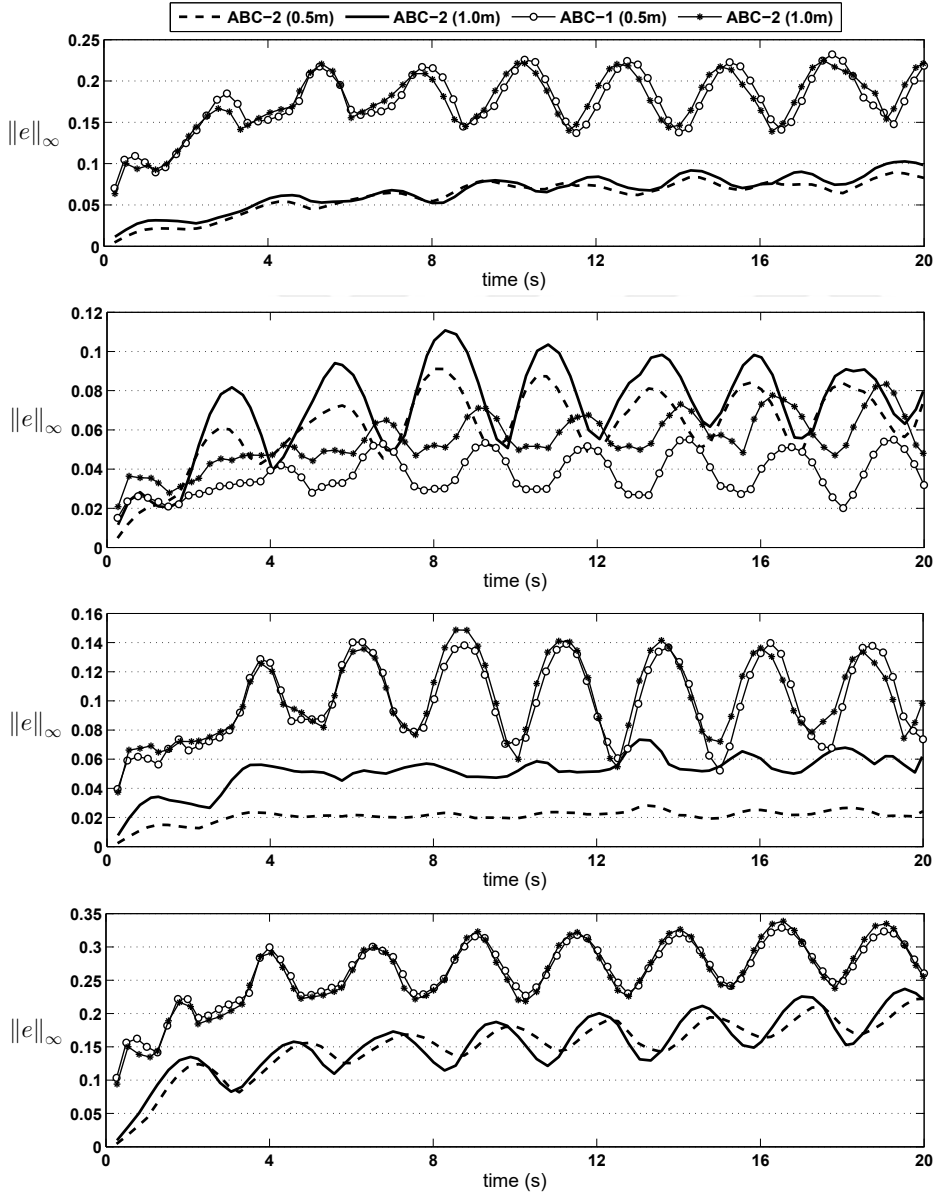


Figure 3.12: Infinity norms $\|e\|_\infty$ (3.62) from the two ABCs on two grid resolutions for waves with various angles of incidence. From top to bottom results are shown for $\theta = 20^\circ$, $\theta = 40^\circ$, $\theta = 60^\circ$ and $\theta = 80^\circ$.

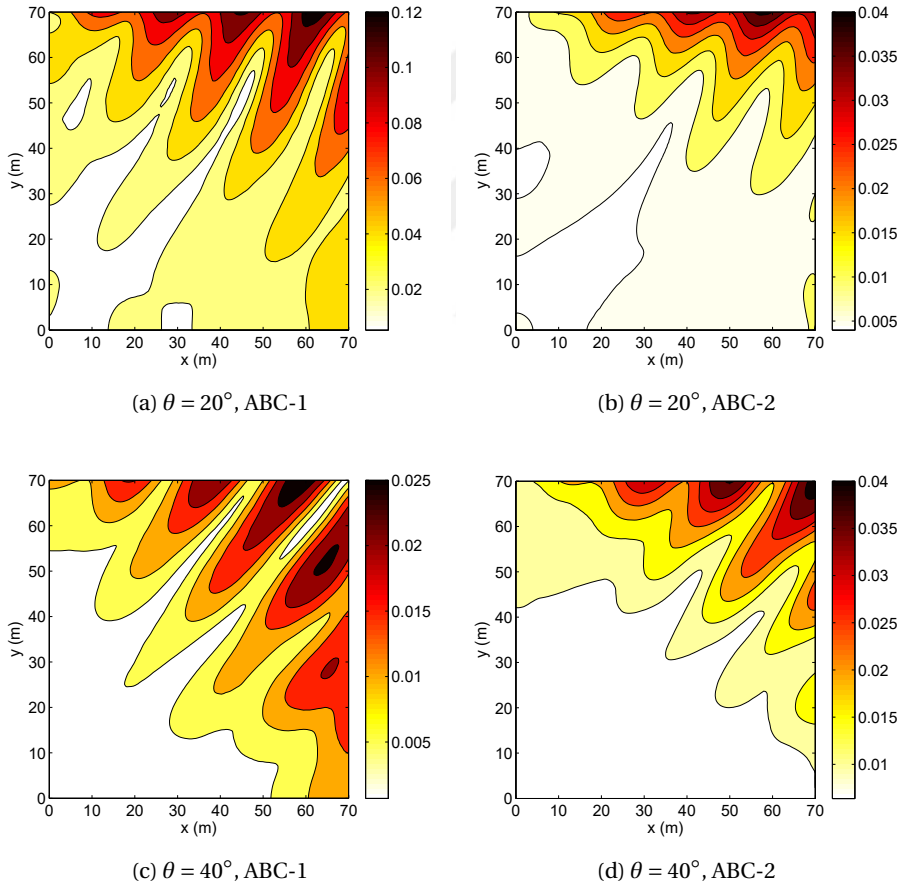


Figure 3.13: The pointwise error $|e(i, j)|$ (3.61) averaged over time for the tests on the fine grid with ABC-1 and ABC-2. The error has been scaled with the wave height $H = 0.1m$ (note the different color scale).

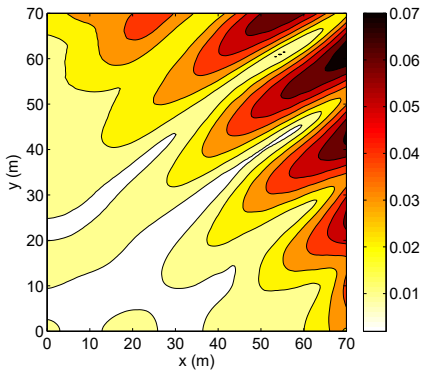
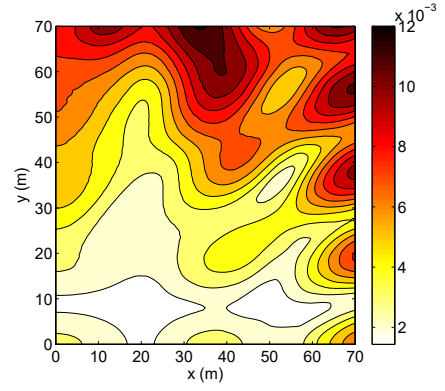
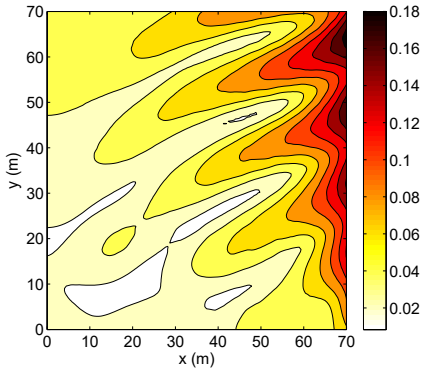
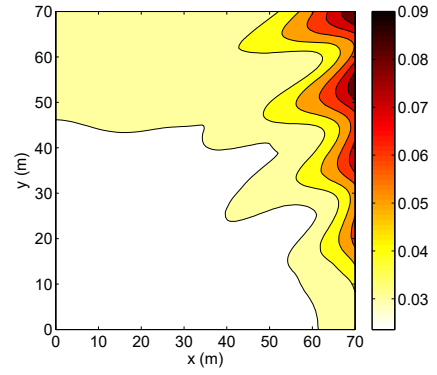
(a) $\theta = 60^\circ$, ABC-1(b) $\theta = 60^\circ$, ABC-2(c) $\theta = 80^\circ$, ABC-1(d) $\theta = 80^\circ$, ABC-2

Figure 3.14: Continued from Fig. 3.13.

each with its own propagation direction, which naturally resembles practical situations in a better way. In the previous numerical experiments we had the information regarding the phase speed and angle of incidence beforehand. In this case we do not know the phase speed or the angle of incidence of the component that is impinging on the outflow boundaries. Therefore we will choose the parameters in the ABCs in such a way that the reflection coefficient is minimum for the ranges of kh and θ of the components.

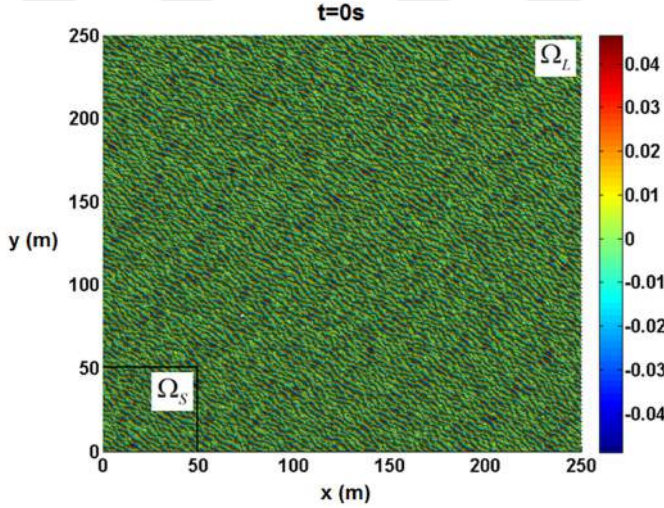


Figure 3.15: The setup for the short-crested wave simulation. The solution in Ω_L is considered as the reference solution.

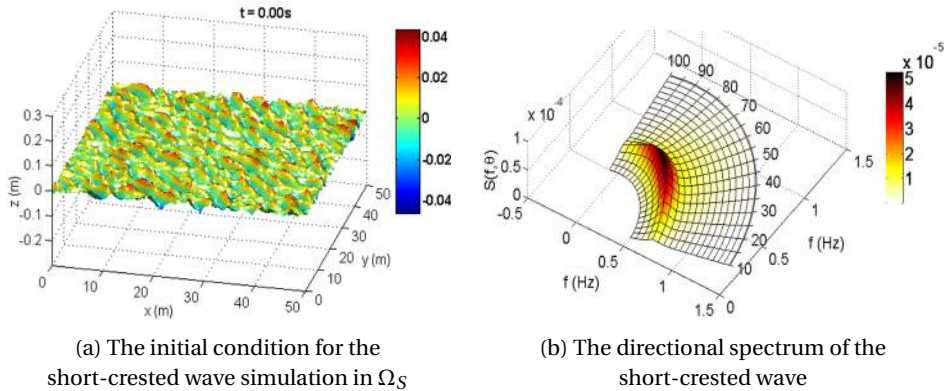


Figure 3.16: Properties of the short-crested wave

The free surface of the short-crested wave at $t = 0$ s is shown in Fig. 3.16a. This JONSWAP spectrum wave is composed of 537 Fourier components with $T_p = 9$ s and $H_s = 0.1$ m, and Fig. 3.16b illustrates how the energy of this wave spreads along frequency

and direction. Frequencies are defined between 0.5 and 1.4 Hz, and directions are defined between 10 and 100 degrees.

Numerical simulation is carried out by performing 4000 time-steps at $\Delta t = 0.009s$. The lengths of Ω_S in the x - and y -directions are the same, $50m$, and the water depth is $h = 2.875m$. By running the simulations for $t_{max} = 36.0s$ and taking into account the phase speed of the fastest propagating wave component $\sqrt{gh}$, we calculate the lengths of Ω_L in the x - and y -directions as $250m$. Two uniform grid resolutions are considered: $0.5m$ and $0.25m$.

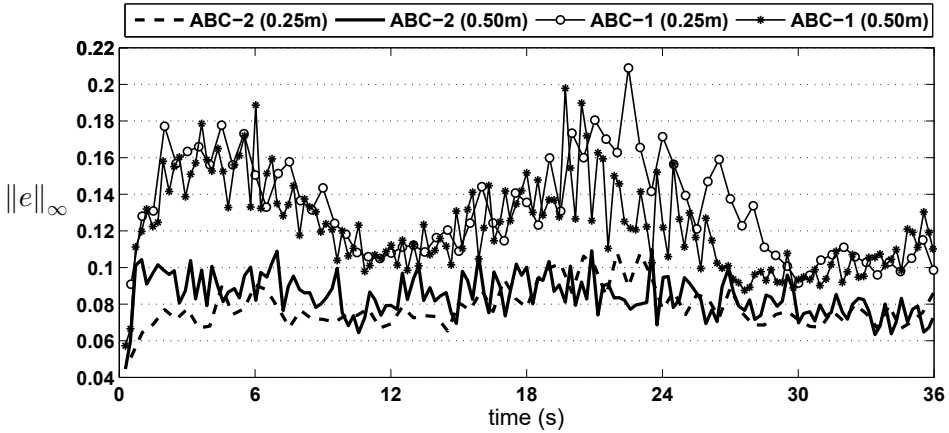


Figure 3.17: Error norm $\|e\|_\infty$ (3.62) as a function of time from the two ABCs on two grid resolutions for the short-crested wave test.

Figure 3.17 shows the infinity norm $\|e\|_\infty$ from the two ABCs on two grid resolutions. The largest value of $\|e\|_\infty$ for the ABC-1 is nearly 21.0% whereas it is only 11.0% with the ABC-2. $\|e\|_\infty$ demonstrates a highly erratic behavior with ABC-1, it reaches the value of 11.0% at $t = 12.0s$, changes to 21.0% at $t = 23.0s$, and decreases back to 9.0% at $t = 30.0s$. When it comes to ABC-2, however, it oscillates within a much more limited band of 7.0% to 11.0% during the entire simulation.

Figure 3.18 demonstrates the pointwise error $\|e(i, j)\|$ (3.61), normalized with H_s and averaged over time, on the fine grid. Analogous to the previous numerical tests, the reflected waves propagate back into the computational domain and spoil the solution. As the short-crested wave striking the outflow boundaries is composed of many components propagating in their own directions, the reflected wave is so as well. That is why the reflected waves in Fig. 3.18 are much more chaotic compared to the ones depicted in Figs. 3.13 and 3.14. It is imminent that the second-order ABC-2 is significantly better than its first-order counterpart ABC-1 (note the difference in color scale).

3.5.4. CIRCULAR CONCENTRIC WAVES RADIATING FROM A LOCALIZED SOURCE

In the last test an oscillating solid sphere with a prescribed motion is impacting on water surface in a tank. The sphere with a radius of $4m$ is initially located $4m$ above the free surface, see Fig. 3.19a for the initial condition of the problem. The sphere is allowed

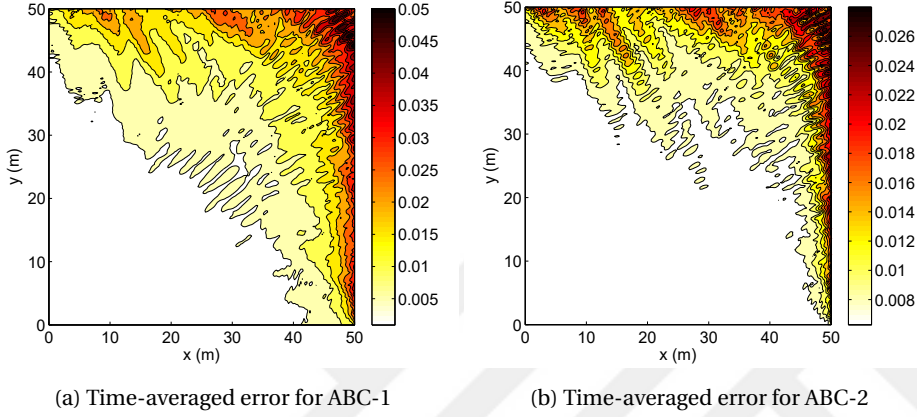


Figure 3.18: The pointwise error $|e(i, j)|$ (3.61) averaged over time for the short-crested wave test on the fine grid with ABC-1 and ABC-2. The error has been scaled with the critical wave height $H_s = 0.1m$ (note the different color scale).

to make only vertical motion along the z -direction. Sinusoidal motion of the sphere is prescribed by $z_s(t) = 2 + 2 \cos(2.4t)$. After the impact of the sphere on the water surface initially at rest, a series of circular concentric waves is produced radiating outward from the center in all directions. To absorb these waves ABC-1 and ABC-2 are used at the outflow boundaries and their performance is compared. Similar to the previous tests reference solution is obtained by repeating the simulation in a large domain, see Fig. 3.19b for the setup of the two domains. The length and the width of Ω_S is the same, $Lx_{\Omega_S} = Ly_{\Omega_S} = 50m$, and its depth is $Lz_{\Omega_S} = 10m$. Ω_L has the same depth but different length and width, $Lx_{\Omega_L} = Ly_{\Omega_L} = 400m$. The size of Ω_L is arranged in such a way that radiating circular waves do not reach the outflow boundaries of Ω_L throughout the simulation.

Two uniform grid resolutions of $0.25m$ and $0.5m$ are considered. Simulations are performed for $30s$. A large number of probes are placed in the domain to compare the free surface elevation records at various locations. Due to the symmetry of the problem we show results only from six probes, see Fig. 3.20 for the locations of the probes. Because of the prescribed motion of the sphere the generated circular wave is regular, and both the ABC-1 and ABC-2 are tuned to absorb this regular wave. However, since the wave is circular, it impinges on the outflow boundaries at different angles at different positions. To account for this directional effect, the angle coefficients in both boundary conditions are set to 45° .

Fig. 3.21 illustrates snapshots of the simulations at $t = 30s$ when ABC-1 and ABC-2 are used on the fine grid. With ABC-1 large amounts of reflection at different locations result in a chaotic free surface. With ABC-2, however, the free surface is considerably less disturbed.

Figures 3.22 and 3.23 show the free surface elevation history from the six probes on two grid resolutions. As the circular concentric wave travels out from the center, the am-

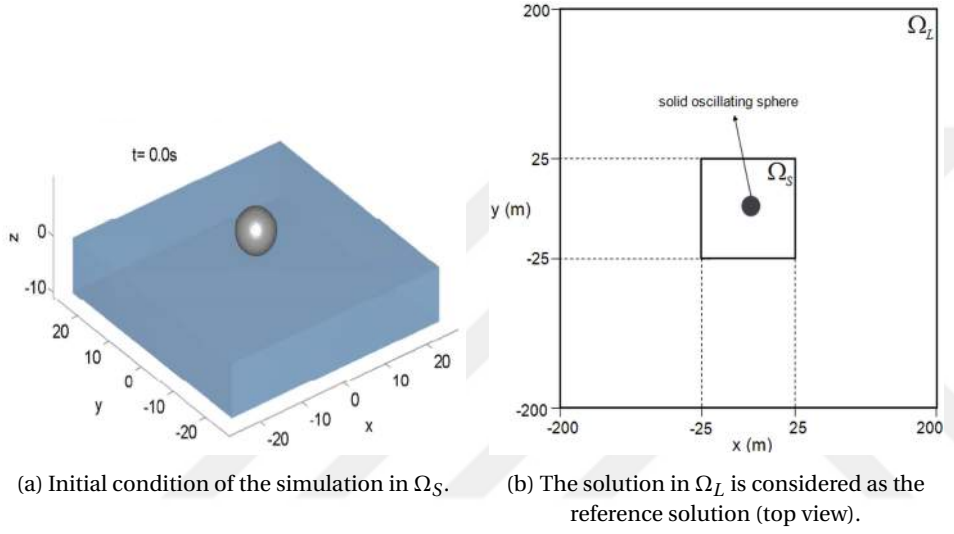


Figure 3.19: Setup for the test with the oscillating solid sphere.

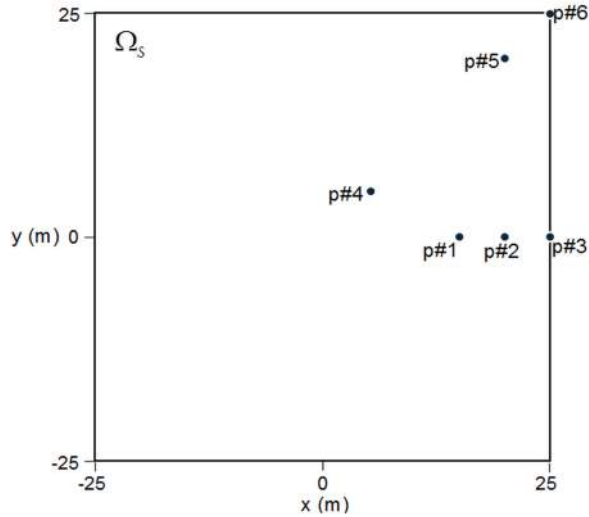
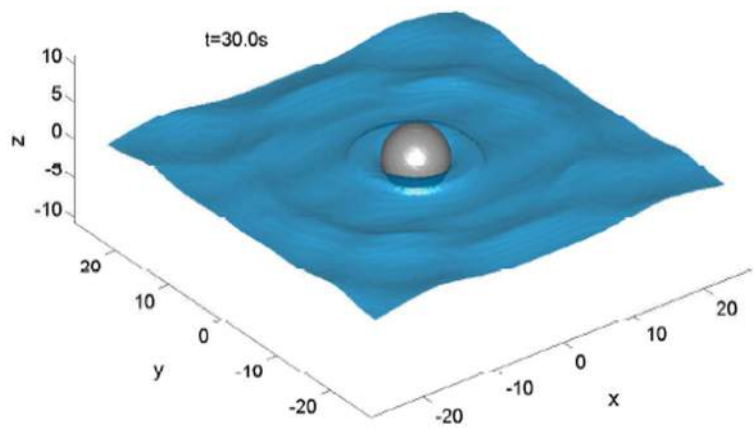
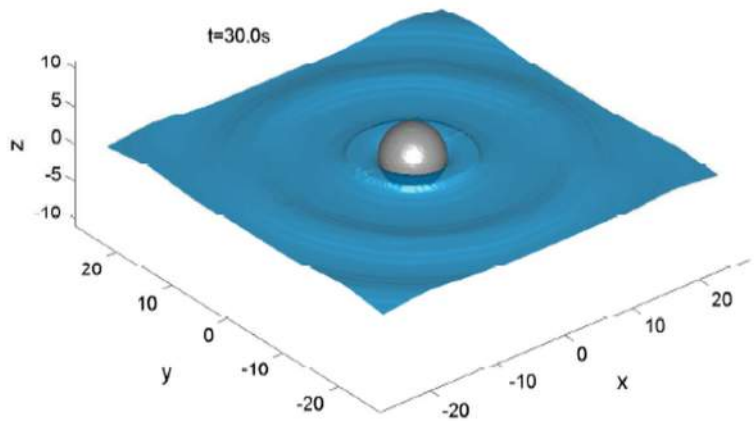


Figure 3.20: Six wave probes used for comparing the solutions in Ω_S and Ω_L . Positions are $p\#1 = (15m, 0m)$, $p\#2 = (20m, 0m)$, $p\#3 = (25m, 0m)$, $p\#4 = (5m, 5m)$, $p\#5 = (20m, 20m)$, $p\#6 = (25m, 25m)$.



(a) ABC-1



(b) ABC-2

Figure 3.21: Snapshots of the simulations at $t = 30.0s$ on the fine grid with uniform resolution of $0.25m$.

plitude of the wave decreases. This is due to the fact that the energy of the wave is spread over a larger area as the wave radiates from the center, which suggests that each particle of the wave gets less energy. This causes a decrease in the wave amplitude. Results demonstrate the superiority of the ABC-2 over ABC-1 clearly. Apart from the wave probe $p\#4$, the other five probes show a significant difference between the performance of the ABC-1 and ABC-2. The probe $p\#4$ is very close to the center of the domain, and therefore, it takes more time for the reflected waves to reach this probe. After $t = 25s$ this probe also starts to show some difference.

3.6. CONCLUDING REMARKS

This chapter presents the design and performance of two generating and absorbing boundary conditions. These boundary conditions are based on the first- and second-order Higdon operators [12]. The numerical implementation of the first- and second-order variant, ABC-1 and ABC-2 respectively, using a staggered grid arrangement is explained in detail. Various numerical examples are provided to demonstrate the performance of the ABC-1 and ABC-2. The reflection character is observed, and the results of the computations are discussed and compared. The numerical simulations showed that the second-order variant ABC-2 clearly outperforms the first-order variant ABC-1 in terms of both dispersive and directional effects of waves. Moreover, the increase in performance does not come at the expense of additional computational cost. It was shown that the ABC-2 is solved in the same total number of cells as ABC-1. Compared to that for the ABC-1, the stencil for the ABC-2 is larger and extends in the direction normal to the outflow boundary. Nonetheless, a Krylov subspace solver, BiCGSTAB, with ILU(p) factorization as preconditioner (which was chosen more accurate near the boundaries) was capable enough to solve the linear system within similar amounts of time with ABC-1 and ABC-2.

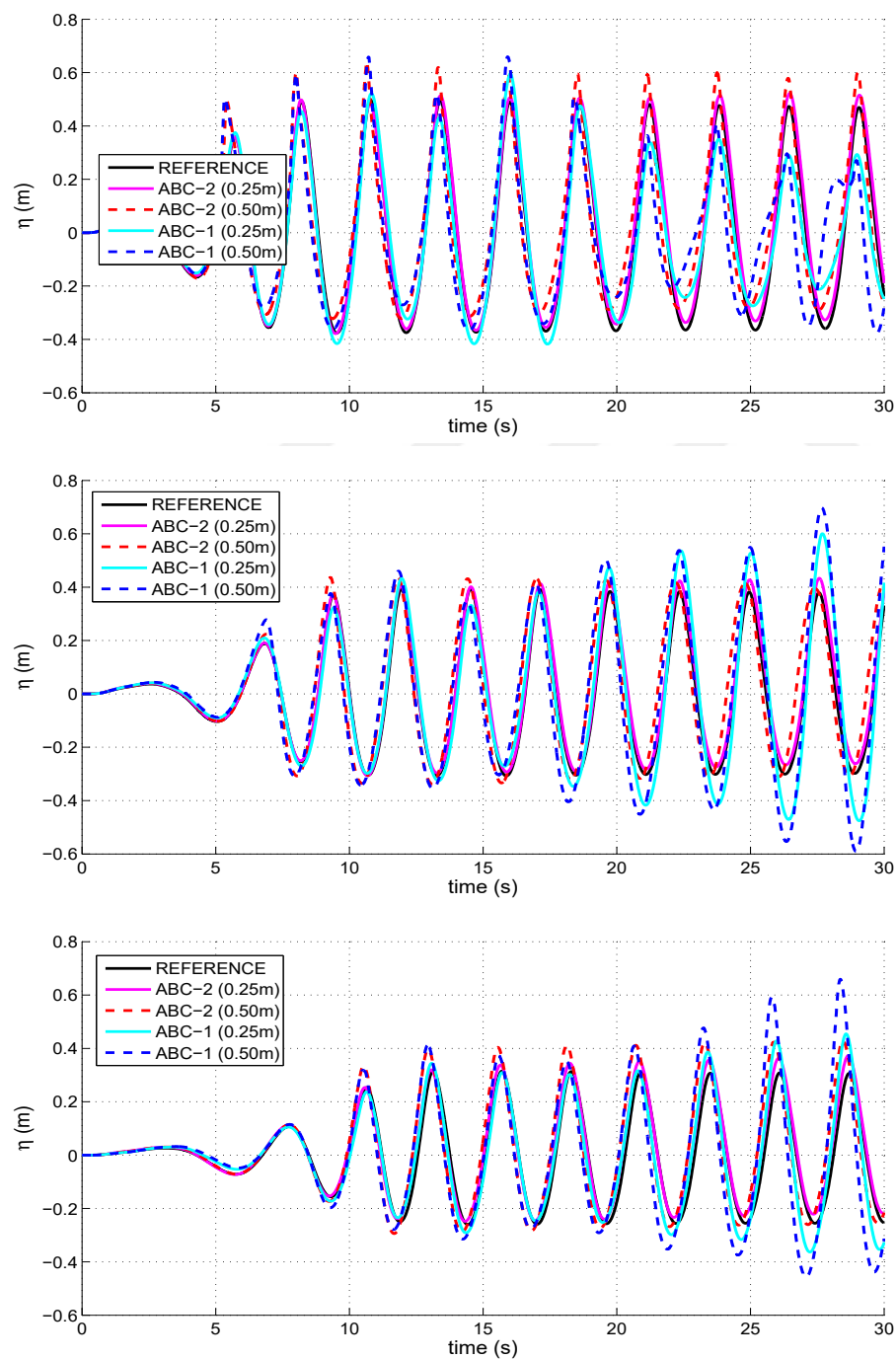


Figure 3.22: Free surface elevations as a function of time at various locations. From top to bottom results are shown at $p\#1$, $p\#2$ and $p\#3$.

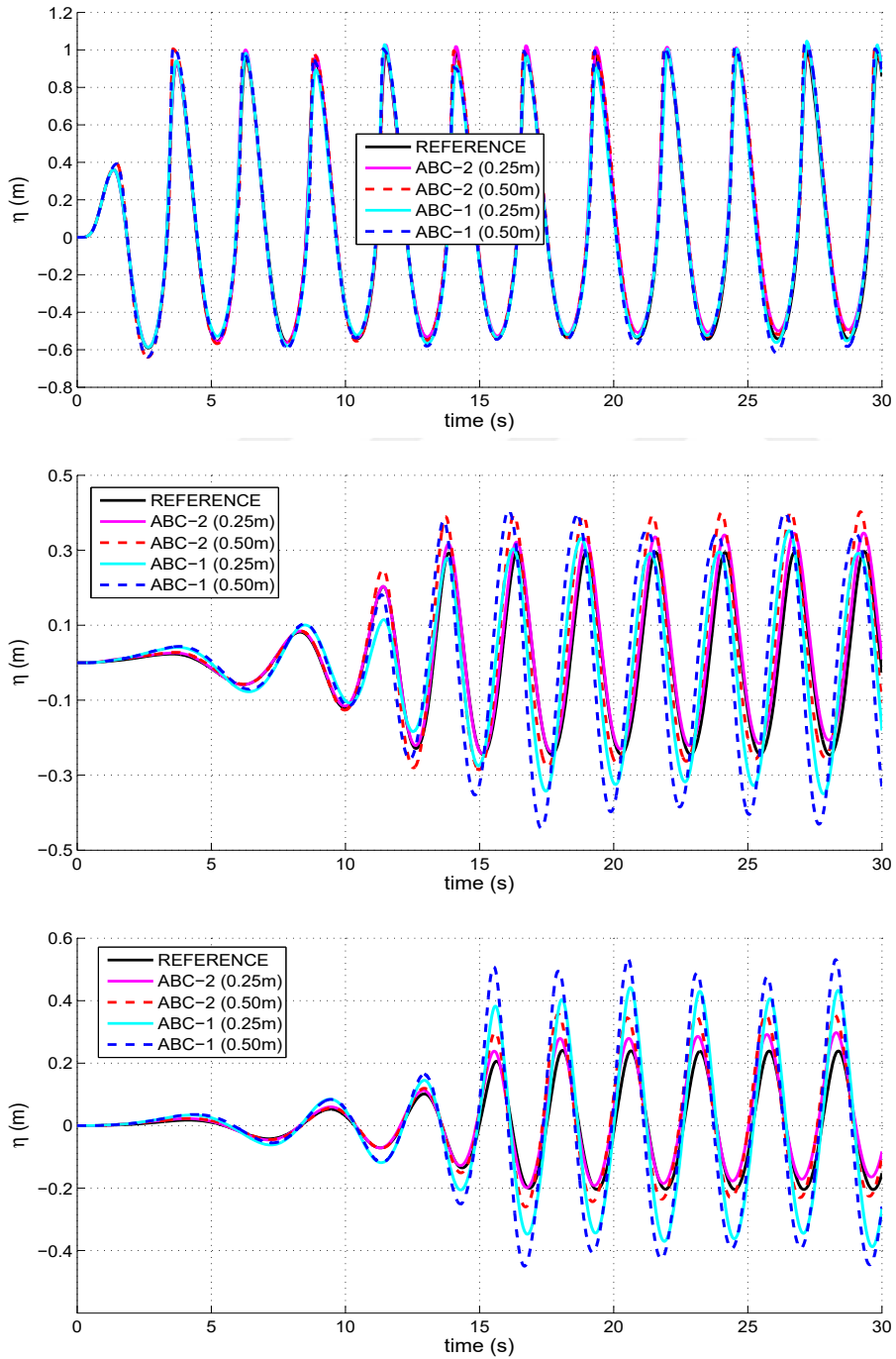


Figure 3.23: Free surface elevations as a function of time at various locations. From top to bottom results are shown at $p\#4$, $p\#5$ and $p\#6$.

REFERENCES

- [1] D. Givoli, *High-order local non-reflecting boundary conditions: a review*, Wave Motion **39**, 319 (2004).
- [2] D. Givoli, *Non-reflecting boundary conditions*, J. Comput. Phys. **94**, 1 (1991).
- [3] S. Tsynkov, *Numerical solution of problems on unbounded domains. A review*, Appl. Numer. Math. **27**, 465 (1998).
- [4] T. Hagstrom, *Radiation boundary conditions for the numerical simulation of waves*, Acta Numer. **8**, 47 (1999).
- [5] T. Colonius, *Modeling artificial boundary conditions for compressible flow*, Annu. Rev. Fluid Mech. **36**, 315 (2004).
- [6] D. Appelö, *Non-reflecting boundary conditions for wave propagation problems*, Ph.D. thesis, Royal Institute of Technology, Sweden (2003).
- [7] J. Romate, *Absorbing boundary conditions for free surface waves*, J. Comput. Phys. **99**, 135 (1992).
- [8] B. Engquist and A. Majda, *Absorbing boundary conditions for the numerical simulation of waves*, Math. Comput. **31**, 629 (1977).
- [9] R. Renaut and J. Parent, *Rational approximations to $1/\sqrt{1-s^2}$, one-way wave equations and absorbing boundary conditions*, J. Comput. Appl. Math **72**, 245 (1996).
- [10] A. Sommerfeld, *Partial differential equations in physics* (Academic Press, 1949).
- [11] R. Higdon, *Absorbing boundary conditions for difference approximations to the multi-dimensional wave equation*, Math. Comput. **47**, 437 (1986).
- [12] R. Higdon, *Numerical absorbing boundary conditions for the wave equation*, Math. Comput. **49**, 65 (1987).
- [13] A. Bayliss and E. Turkel, *Radiation boundary conditions for wave-like equations*, Comm. Pure Appl. Math. **33**, 707 (1980).
- [14] M. Grote and I. Sim, *On local nonreflecting boundary conditions for time dependent wave propagation*, Chin. Ann. Math. **30B**, 589 (2009).
- [15] D. Givoli, *Computational absorbing boundaries*, (in: S. Marburg and B. Nolte (Eds.), Computational Acoustics of noise propagation in fluids - Finite and boundary element methods, Springer-Verlag, Berlin Heidelberg, 2008) pp. 145–166.
- [16] F. Collino, *High order absorbing boundary conditions for wave propagation models. Straight line boundary and corner cases*, in *Proceedings of the 2nd International Conference on Mathematical and Numerical Aspects of Wave Propagation*, SIAM, Delaware (USA) (1993) pp. 161–171.

- [17] T. Hagstrom and S. Hariharan, *A formulation of asymptotic and exact boundary conditions using local operators*, Appl. Numer. Math. **27**, 403 (1998).
- [18] D. Givoli and B. Neta, *High-order non-reflecting boundary conditions for dispersive waves*, Wave Motion **37**, 257 (2003).
- [19] T. Hagstrom and T. Warburton, *A new auxiliary variable formulation of high-order local radiation boundary conditions: corner compatibility conditions and extensions to first-order systems*, Wave Motion **39**, 327 (2004).
- [20] I. Orlanski, *A simple boundary condition for unbounded hyperbolic flows*, J. Comput. Phys. **21**, 251 (1976).
- [21] D. Durran, M. Yang, D. Slinn, and R. Brown, *Toward more accurate wave-permeable boundary conditions*, Mon. Wea. Rev. **121**, 604 (1993).
- [22] M. Hedley and M. Yau, *Radiation boundary conditions in numerical modeling*, Mon. Wea. Rev. **116**, 1721 (1988).
- [23] T. Jensen, *Open boundary conditions in stratified ocean models*, J. Marine Syst. **16**, 297 (1998).
- [24] D. Givoli and B. Neta, *High-order non-reflecting boundary scheme for time-dependent waves*, J. Comput. Phys. **186**, 24 (2003).
- [25] T. Hagstrom, A. Mar-Or, and D. Givoli, *High-order local absorbing conditions for the wave equation: Extensions and improvements*, J. Comput. Phys. **227**, 3322 (2008).
- [26] G. Lancioni, *Numerical comparison of high-order absorbing boundary conditions and perfectly matched layers for a dispersive one-dimensional medium*, Comput. Methods Appl. Mech. Engrg. **209-212**, 74 (2012).
- [27] P. Wellens, *Wave simulation in truncated domains for offshore applications*, Ph.D. thesis, Delft University of Technology (2012).
- [28] H.-O. Kreiss, *Initial boundary value problems for hyperbolic systems*, Comm. Pure Appl. Math. **23**, 277 (1970).
- [29] L. Trefethen and L. Halpern, *Well-posedness of one-way wave equations and absorbing boundary conditions*, Math. Comp. **47**, 421 (1986).
- [30] R. Higdon, *Initial-boundary value problems for linear hyperbolic systems*, SIAM Rev. **28**, 177 (1986).
- [31] W. Mulder, *Experiments with Higdon's absorbing boundary conditions for a number of wave equations*, Comput. Geosci. **1**, 85 (1997).
- [32] M. Grote and J. Keller, *Exact nonreflecting boundary conditions for the time dependent wave equation*. SIAM J. Appl. Math. **55**, 280 (1995).

- [33] M. Grote and J. Keller, *Nonreflecting boundary conditions for time-dependent scattering*, J. Comput. Phys. **127**, 52 (1996).
- [34] J. Berenger, *A perfectly matched layer for the absorption of electromagnetic waves*, J. Comput. Phys. **114**, 185 (1994).
- [35] G. Wei and J. Kirby, *Time-dependent numerical code for extended Boussinesq equations*, J. Waterway, Port, Coastal, Ocean Eng. **121**, 251 (1995).
- [36] R. Dean and R. Dalrymple, *Water wave mechanics for engineers and scientists* (World Scientific Publishing, Singapore, 1991).
- [37] L. Skjelbreia and J. Hendrickson, *Fifth order gravity wave theory*, in *Proceedings of the 7th Conf. of Coastal Engineering, The Hague (The Netherlands)* (1960) pp. 184–196.
- [38] M. Rienecker and J. Fenton, *A Fourier approximation method for steady water-waves*, J. Fluid Mech. **104**, 119 (1981).
- [39] B. L. Méhauté, *Introduction to hydrodynamics and water waves* (Springer-Verlag, New York, 1976).
- [40] R. Wiegel, *A presentation of cnoidal wave theory for practical application*, J. Fluid Mech. **7**, 273 (1960).
- [41] G. Mallery and G. Clark, *Cnoidal wave theory for application to offshore structural design*, in *The 4th Annual Offshore Technology Conference, 1-3 May 1972, Texas (USA), paper OTC-1614-MS* (1972).
- [42] DNV-RP-C205, *Recommended Practice, Environmental conditions and environmental loads*, Det Norske Veritas (2010).
- [43] L. Holthuijsen, *Waves in oceanic and coastal waters* (Cambridge University Press, 2007).
- [44] A. Osborne, *Nonlinear ocean waves and the inverse scattering transform* (Academic Press, 2010).
- [45] A. Hermans, *Water waves and ship hydrodynamics* (Springer, 2011).
- [46] M. Longuet-Higgins, *The effect of non-linearities on statistical distributions in the theory of sea waves*, J. Fluid Mech. **17**, 459 (1963).
- [47] J. Sharma and R. Dean, *Second-order directional seas and associated wave forces*, SPE J. **21**, 129 (1981).
- [48] J. Zhang, L. Chen, M. Ye, and R. Randall, *Hybrid wave model for unidirectional irregular waves-part i. Theory and numerical scheme*, Appl. Ocean Res. **18**, 77 (1996).
- [49] P. Madsen, H. Bingham, and H. Liu, *A new Boussinesq method for fully nonlinear waves from shallow to deep water*, J. Fluid Mech. **462**, 1 (2002).

- [50] S. Chakrabarti, *Handbook of offshore engineering vol. 1* (Elsevier, 2005).
- [51] M. Renaud, F. Rezende, O. Waals, X.-B. Chen, and R. van Dijk, *Second-order wave loads on a LNG carrier in multi-directional waves*, in *Proceedings of the 27th Int. Conf. on Offshore Mech. and Arctic Eng. OMAE*, 16-19 June 2008, Estoril (Portugal), vol. 1 (2008) pp. 363–370.
- [52] E. Jefferys, *Directional seas should be ergodic*, Appl. Ocean Res. **9**, 186 (1987).
- [53] M. Miles and E. Funke, *A comparison of methods for synthesis of directional seas*, ASME OMAE J. **111**, 43 (1989).
- [54] M. Miles, M. Benoit, P. Frigaard, P. Hawkes, H. Schäffer, and C. Standsberg, *A comparison study of multidirectional waves generated in laboratory basins*, in *Proceedings of the 27th IAHR Congress*, 10-15 August 1997, San Francisco (USA) (1997).
- [55] R. Pascal and I. Bryden, *Directional spectrum methods for deterministic waves*, Ocean Eng. **38**, 1382 (2011).
- [56] T. Takayama and T. Hiraishi, *Reproducibility of directional random waves in laboratory wave simulation*, Report of the Port and Harbour Research Institute (1989).
- [57] P. Hawkes, J. Ewing, C. Harford, G. Klopman, C. Standsberg, M. Benoit, M. Briggs, P. Frigaard, T. Hiraishi, M. Miles, J. Santas, and H. Schäffer, *Comparative analyses of multidirectional wave basin data*, in *Proceedings of the 27th IAHR Congress*, 10-15 August 1997, San Francisco (USA) (1997).
- [58] E. Blayo and L. Debreu, *Revisiting open boundary conditions from the point of view of characteristic variables*, Ocean Model. **9**, 231 (2005).
- [59] A. Mar-Or and D. Givoli, *The global-regional model interaction problem: Analysis of Carpenter's scheme and related issues*, Int. J. Multiscale Comput. Eng. **4**, 617 (2006).
- [60] K. Carpenter, *Note on the paper 'Radiation conditions for the lateral boundaries of limited-area numerical models' by M.J. Miller, A.J. Thorpe*, Quart. J. R. Meteorol. Soc. **108**, 717 (1982).
- [61] J. Redelsperger and J. Lafore, *A three-dimensional simulation of a tropical squall line: Convective organization and thermodynamic vertical transport*, J. Atmos. Sci. **45**, 1334 (1988).
- [62] A. Perkins, L. Smedstad, D. Blake, G. Heburn, and A. Wallcraft, *A new nested boundary condition for a primitive equation ocean model*, J. Geophys. Res. **102**, 3483 (1997).
- [63] P. Bougeault, P. Mascart, and J. Chaboureaud, *The Meso-NH Atmospheric Simulation System: Scientific Documentation*, Tech. Rep. (Centre National de Recherches Météorologiques (Météo-France) and Laboratoire d'Aérodynamique (CNRS), 2013).

- [64] J. van der Meer, H. Petit, P. van den Bosch, G. Klopman, and R. Broekens, *Numerical simulation of wave motion on and in coastal structures*, in *Proceedings of the 23rd Int. Conf. on Coastal Engineering, Venice (Italy)*, vol. 2 (1992) pp. 1772–1784.
- [65] P. Troch and J. D. Rouck, *An active wave generating-absorbing boundary condition for VOF type numerical model*, *Coast. Eng.* **38**, 223 (1999).
- [66] D. Durran, *Numerical methods for Fluid Dynamics with applications to Geophysics* (Springer, 2010).
- [67] A. Mar-Or and D. Givoli, *High order global-regional model interaction: Extension of Carpenter's scheme*, *Int. J. Numer. Meth. Eng.* **77**, 50 (2009).
- [68] A. Mar-Or and D. Givoli, *High-order one-way model nesting in dispersive non-uniform media*, *J. Comput. Appl. Math* **234**, 1663 (2010).
- [69] A. Veldman, *Computational Fluid Dynamics* (Lecture notes in Applied Mathematics, 2010).
- [70] B. Duz, R. Huijsmans, P. Wellens, M. Borsboom, and A. Veldman, *Application of an absorbing boundary condition in a wave-structure interaction problem*, in *Proceedings of the 31st Int. Conf. on Ocean, Offshore and Arctic Eng. OMAE, 10-15 June 2012, Rio de Janeiro (Brazil)*, vol. 5 (2012) pp. 863–872.
- [71] B. Duz, R. Huijsmans, A. Veldman, M. Borsboom, and P. Wellens, *An absorbing boundary condition for regular and irregular wave simulations*, (in: L. Eça, E. Oñate, J. García-Espinosa, T. Kvamsdal and P. Bergan (Eds.), *Computational Methods in Applied Sciences*, vol. 29, Springer, 2013) pp. 31–45.
- [72] E. Botta and M. Ellenbroek, *A modified SOR method for the Poisson equation in unsteady free-surface flow calculations*, *J. Comput. Phys.* **60**, 119 (1985).

4

VOLUME OF FLUID (VOF) METHOD

4.1. INTRODUCTION

FREE surfaces or interfaces between immiscible fluids are broadly featured in many processes in modern industry as well as within the human body and in the environment we live in. In addition to various computational modeling challenges that the free surface flows present, the mere existence of a free surface alone poses some difficulties. On the one hand the solution region changes as the free surface evolves, and on the other hand, the motion of the free surface is in turn determined by the solution. Changes in the solution region include not only changes in size and shape, but in some cases, may also include the coalescence and break up of regions (i.e., the loss and gain of free surfaces). During these processes it is important to keep the free surface sharp and well-defined, and evolve it without smearing, dispersing or wrinkling. This is a critically significant task in order to achieve an accurate solution to the overall physical problem.

Numerical methods for modeling free surfaces have been a popular subject for several decades among researchers from various fields of science. This resulted in various types of strategies with different theoretical backgrounds that can robustly and efficiently represent evolving and topologically complex free surfaces. It is quite a challenging effort to cite all major developments due to the large volume of the existing literature. Below we give a brief survey of these strategies and refer interested readers to [1] and [2] for more extensive surveys.

In marker methods massless markers or tracers are used on a fixed mesh to track fluid volumes in the entire flow domain (volume markers) or to track exactly the location of the interface (surface markers). The positions of the markers are updated using the underlying velocity field in a Lagrangian fashion. One of the earliest works in this field goes back to [Harlow and Welch](#) [3] who used massless markers in the entire flow domain, the marker-and-cell method (MAC). In the original implementation there was only one fluid in the domain, and the example problem was the dam break. Later [Daly](#) [4] adopted the method by incorporating surface tension effects and examined the Rayleigh-Taylor instability. In the following works markers are used only on the interface. This resulted

in a significant reduction in computer time and storage, and also provided the explicit location of the interface. Over the years [Glimm *et al.* \[5\]](#), [Chen *et al.* \[6\]](#) and [Unverdi and Tryggvason \[7\]](#), among others, made important contributions to this idea.

In the level-set method introduced by [Osher and Sethian \[8\]](#) a continuous level-set function is used to track the motion of an interface. The interface is represented as the zero level set of this signed distance function. Inside one of the fluids the function takes a positive value, and in the other fluid it takes a negative value. The level-set function moves with the fluid, and therefore evolves according to a simple transport equation. Apart from solving this equation the level-set method does not require any additional complex procedures, and hence it is conceptually simple as well as easy to implement. However issues with mass conservation were reported in early implementations especially when the interface undergoes large deformations. [Sussman *et al.* \[9\]](#) and [Sussman and Fatemi \[10\]](#) proposed strategies to improve the mass conservation property of the level-set method, see also [Enright *et al.* \[11\]](#) for further discussions. In unsteady free surface flows [Iafraiti *et al.* \[12\]](#) showed results from flow over a jump and flow inside a tank using the level-set method.

[Yabe *et al.* \[13\]](#) proposed a powerful family of constrained interpolation profile (CIP) methods. The CIP method was initially presented by [Takewaki *et al.* \[14\]](#) and [Takewaki and Yabe \[15\]](#), and the abbreviation then stood for *cubic interpolated pseudo-particle*. Over the years the method has evolved, its name has changed but the abbreviation stayed the same. Nonetheless it has been successfully applied to various multi-phase flows; see [13]. The strength of this method stems from the strategy that it uses the primitive variables and their derivatives as a set of dependent variables. For the first group the conservation equations are used and for the second group the corresponding derivatives of these equations are used. The recent version of the method guarantees exact mass conservation, and results in a low dispersion error. The reader is referred to the aforementioned and other publications from these researchers for further discussions on the CIP method.

Before going into the details of the volume-of-fluid (VOF) method, which constitutes the backbone of the present work, it is important to note that there are other methods used for interface tracking/capturing in interfacial and free surface flows. Among these methods are phase-field [16] and [17], continuum advection [18], point-set [19], Lagrangian [20] and [21], and moment-of-fluid (MOF) [22].

Volume-of-fluid (VOF) methods have been successfully used in computational fluid dynamics (CFD) simulation of interfacial and free surface flows for several decades since the introduction by [Hirt and Nichols \[23\]](#). Typically, the VOF approach presents a model based on a scalar indicator function to transport the fluid from one cell to another on a fixed computational mesh using the underlying velocity field. This function is characterized by the volume fraction F occupying one of the fluids within each cell. If a cell is completely filled with one fluid, the volume fraction takes the value of 1, and 0 if only the second fluid is present. The values between these two limits indicate the presence of the interface or free surface. In the VOF approach, the volume fraction field is the only available and required information representing the interface profile. Therefore, if the explicit location of the interface is needed, special algorithms have to be applied to attain an approximate reconstruction of the interface by exploiting the volume fraction

distribution of the neighboring cells in a compact stencil.

In the VOF technique, the volume fraction field is propagated by solving a scalar transport equation. Discretization of the transport equation with an accurate numerical method is critical for not only the conservation of mass but also evolving the interface without smearing, dispersing or wrinkling it. This basically constitutes the principal drawback of the VOF approach, especially considering the fact that the discrete volume fraction field is not smoothly distributed at the interface (on the contrary it displays sharp discontinuous changes between 0 and 1). In this regard, conventional convective differencing schemes, such as upwinding, are unable to maintain a well-defined interface due to numerical diffusion, even if they do not violate the boundedness of the solution ($0 \leq F \leq 1$) through the sufficient boundedness criterion. In order to resolve the interface while modeling fluid flow behaviors such as large deformation, interface rupture and coalescence in a natural fashion, researchers have developed numerous techniques within the VOF function framework. They can be classified into three categories: donor-acceptor formulation, high resolution differencing schemes and line techniques (explicit geometrical reconstruction of the interface).

In the donor-acceptor formulation, the volume fraction values of the downwind and upwind of a flux boundary are used to estimate the amount of volume fraction transported through that boundary during a time step. These volume fraction values are used to predict the orientation of the interface when computing the flux volume. Hence, local interface reconstruction is not essentially needed. However, the inclusion of downwind information generally violates the boundedness criterion causing unphysical overshoots and undershoots. In order to ensure boundedness, several improvements have been incorporated into the donor-acceptor formulation, such as *controlled downwinding*. This idea established the structure for the derivation of the well-known VOF method by [Hirt and Nichols](#) [23]. To take a more in-depth look at the donor-acceptor formulation, see [2], [24], [25], [26], and [27].

High-resolution differencing schemes utilize the idea of implementing a higher order or blended differencing scheme to approximate the transport equation. These schemes outperform the first-order upwind scheme in terms of accuracy, and the second-order central difference scheme in terms of stability; see [28]. However, they fail to satisfy the boundedness criterion of the volume fraction values provoking overshoots and undershoots in regions where steep gradients of the flow variables appear along with the high local Peclet number. To surpass this shortcoming, many techniques have been proposed. For a discussion concerning high resolution schemes, see [27], [29], [30], [31], [32], and [33].

In line techniques, the interface is locally reconstructed and corresponding volume fluxes are computed to preserve the sharp profile of the interface. Based on the volume fraction values of the neighboring cells, the orientation and location of the interface in a cell can be calculated in a piecewise constant, piecewise linear or piecewise parabolic fashion. The resulting reconstructed interface is not necessarily continuous, but a rather discontinuous chain of discrete line segments. However, from piecewise constant to piecewise parabolic reconstruction, the discontinuities at the cell boundaries decrease substantially, see [34].

The Simple Line Interface Calculation (SLIC) by [Noh and Woodward](#) [35] was the cor-

nerstone of the geometric interface reconstruction techniques. As its name indicates, the reconstructed interface is a straight line parallel to one of the spatial directions. For the reconstruction, only the volume fraction values of the neighboring cells along a coordinate direction are taken into account in a 3×1 block of cells. Therefore, the interface has a different representation depending on the coordinate direction considered for the reconstruction. Chorin [36] further improved this reconstruction concept by evaluating the volume fraction information in a 3×3 block of cells, nevertheless his version also yields different fluid distributions for each sweep direction. In any case, the extension of SLIC to 3D is trivial. In addition to first-order accuracy, SLIC also results in the shedding of many isolated blobs of *flotsam* and *jetsam* by artificially breaking up the interface [1].

Among the three approaches, the piecewise linear reconstruction is nowadays the most popular approach, and the methods which fall into this category are usually referred to as Piecewise Linear Interface Calculation-or Construction (PLIC) methods. In this approach, the interface is reconstructed by oblique or piecewise linear line segments (or plane segments in 3D). As the reconstruction is performed in a multidimensional manner, the interface does not have a different representation depending on the sweep direction. After Youngs [37] and Lötstedt [38] designed first PLIC methods, researchers have made a significant effort developing methods to achieve second-order accuracy at reasonable computational cost. In Section 4.2, a more comprehensive survey of these methods will be given.

In Piecewise Parabolic Interface Calculation (PPIC) by Price [34], the interface is approximated by an arbitrarily rotated parabola. Hence, the interface is modeled in a more natural way especially in high curvature regions. Moreover, the local curvature of the interface is directly available, which is especially required for modeling the surface tension force acting on the interface. PPIC is inherently third-order accurate if the interface is sufficiently smooth. Unfortunately, this method is only available in 2D.

In the VOF context, to advect the volume fraction field in time, the following transport equation is solved,

$$\frac{\partial F}{\partial t} + \mathbf{u} \cdot \nabla F = 0, \quad (4.1)$$

where $\mathbf{u} = (u, v, w)$ denotes the fluid velocity vector, and $\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)$ is the gradient operator. Assuming a solenoidal velocity field (incompressible flow) modeled by $\nabla \cdot \mathbf{u} = 0$, Eq. (4.1) can alternatively take the form:

$$\frac{\partial F}{\partial t} + \nabla \cdot (\mathbf{u}F) = 0. \quad (4.2)$$

Equation (4.2) can be solved using either an unsplit advection or an operator split advection scheme. Although both strategies have been successfully applied in simulation of interfacial flows, direction split advection schemes are more common in standard VOF methods due to ease of implementation. Treating individual velocity components to compute 1D fluxes for a sequence of updates in each spatial direction is easier than treating velocity components acting in all directions to compute multidimensional fluxes using inherently difficult geometric tasks. However, operator splitting has a strict limitation: it is applicable only within structured mesh environments.

Several attempts have also been made to benefit from advantages of various strategies through coupling them in a hybrid method. Sussman and Puckett [39] developed

a coupled level-set/VOF method (CLSVOF) in order to exploit the favorable features of both the VOF and level-set methods. Here the contribution of the level-set method is to keep a fine description of the geometrical properties of the interface, while that of the VOF method is to minimize mass loss, see [Ménard *et al.* \[40\]](#) for a further discussion on this coupling. Recently [Jemison *et al.* \[41\]](#) proposed a hybrid level-set/moment-of-fluid method (CLSMOF) which uses information from the level set function, volume of fluid function, and reference centroid.

Since the CFD tool ComFLOW is currently in use for practical applications especially in 3D, the required VOF method must be computationally cheap, robust and easily applicable while retaining a reasonable accuracy for representing complex free surfaces. In each section below, after shortly explaining the corresponding subject, we will keep these criteria in mind while choosing or implementing a numerical technique. As one of the main application areas of ComFLOW is wave impact loading on offshore structures and coastal construction, accurate simulation of propagating free surface waves with reasonable computational resources is of critical importance. In that respect, we will pay particular attention to the performance of the methods in this problem. The layout of the rest of the chapter is as follows. In [Section 4.2](#), a brief introduction to piecewise linear interface reconstruction will be given. [Section 4.3](#) discusses interface advection along with geometrical flux computation. [Section 4.4](#) presents results from a series of test cases which includes advection tests, and an application example where VOF methods are used in ComFLOW to simulate two wave cases, a small-amplitude wave with low steepness and a relatively large-amplitude wave with high-steepness. We end the chapter with conclusions in [Section 4.5](#).

4.2. INTERFACE RECONSTRUCTION

The interface reconstruction is the first stage of a typical VOF method. In the PLIC approach, the interface in each cell is approximated by a line (or a plane in three dimensions). Within each cell, the approximated interface can be defined by the equation:

$$\mathbf{m} \cdot \mathbf{x} = m_x x + m_y y + m_z z = \alpha, \quad (4.3)$$

where $\mathbf{m}$ is the local surface normal, $\mathbf{x}$ is the position vector of a point on the interface and α is a constant which is related to the shortest distance from the origin of the cell. Essentially, the interface reconstruction involves two procedures: the determination of $\mathbf{m}$ and α . In this section, we will investigate the algorithms designed to obtain $\mathbf{m}$, and the determination of α is explained in [Section 4.2.5](#). For a given discrete volume fraction field, $\mathbf{m}$ in each cell is usually calculated using the data in a compact neighborhood of the cell considered. However, since the discrete volume fraction field is not smoothly distributed at the interface, computation of $\mathbf{m}$ with high accuracy can be complicated and expensive. Modeling the interface by simply adopting a piecewise linear representation does not always result in a second-order approximation since the accuracy of the reconstruction depends critically on the calculation of the normal. Furthermore, the size of the discontinuities at the cell boundaries is also sensitive to the calculation of the normal. As the grid resolution and the accuracy of the calculation of the normal increase, the discontinuities decrease. However, in general, truly second-order reconstruction

methods tend to require high computational effort especially for three-dimensional implementations. These methods make the situation even more insurmountable when they are used in combination with operator split advection techniques because the resulting algorithm will demand a number of reconstruction sweeps at each time step.

The reconstruction algorithms in the literature are typically based on two methodologies: using finite difference approximations for the gradient of the discrete volume fraction data ∇F and/or minimizing an error measure to obtain optimum interface configuration as a result of an iterative process. Methods by [Parker and Youngs \[42\]](#) and [Barth and Jespersen \[43\]](#), the least squares gradient (LSG) by [Rider and Kothe \[44\]](#) and the coupled Conservative Level-Contour and Cubic-Bezier-based Interface Reconstruction (CLC-CBIR) from [López *et al.* \[45\]](#) fall into the first category, while Least squares Volume-of-fluid Interface Reconstruction Algorithm (LVIRA) by [Pilliod and Puckett \[46\]](#), [Swartz \[47\]](#)'method, Centroid-Vertex Triangle-Normal Averaging (CVTNA) by [Liovic *et al.* \[48\]](#), Least-Squares Fit (LSF) by [Aulisa *et al.* \[49\]](#) and the Patterned Interface Reconstruction (PIR) by [Mosso *et al.* \[50\]](#) fall into the second category. For detailed information concerning the design of these methods, the interested reader is referred to the corresponding papers and the references therein. For an overview of interface reconstruction methods, see, e.g., [\[44\]](#), [\[46\]](#), [\[51\]](#), and [\[52\]](#).

Among many techniques that are available in the literature, we will consider four methods: Parker and Youngs' method (P&Y) [\[42\]](#), the least squares gradient (LSG) technique by [Rider and Kothe \[44\]](#), the Mixed Youngs-centered (MYC) implementation of [Aulisa *et al.* \[49\]](#), and the efficient least squares VOF interface reconstruction algorithm (ELVIRA) by [Pilliod and Puckett \[46\]](#). The ELVIRA scheme by [Pilliod and Puckett \[46\]](#) was originally proposed in 2D. For the 3D implementation, we will consider the approach by [Cervone *et al.* \[53\]](#). In general these methods have slightly lower accuracies compared to some of the contemporary PLIC schemes mentioned above. However, we will later demonstrate that the level of accuracy for some of these methods is satisfying for the problems we typically encounter. In addition, all the four schemes are mathematically simple, non-iterative and involve a small number of algorithmic tasks. Therefore, implementing these methods require a modest amount of programming work, and the result is achieved in an economical manner. In what follows, these methods are described.

4.2.1. PARKER AND YOUNGS' METHOD

Consider the mesh arrangement demonstrated in [Fig. 4.1](#). In this method, the volume fraction gradients are first calculated at the four corners of the central cell (i, j) with a finite difference formula using a nine-point stencil in two dimensions, and then the gradients at the corners are averaged to the center of the cell. At the corners of the control volume, the x -direction components are defined as

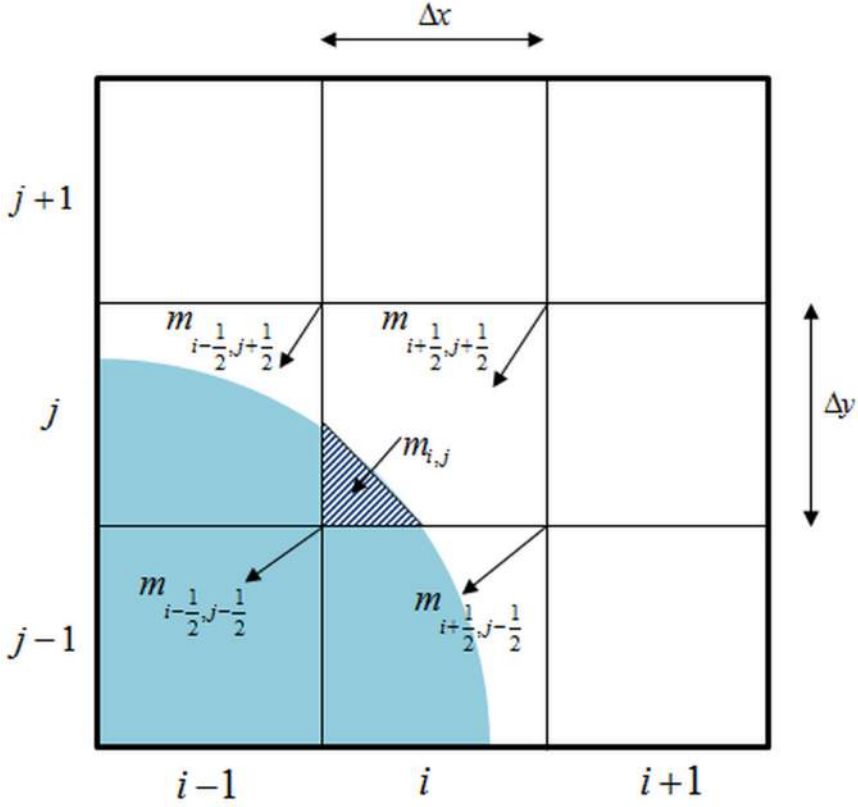


Figure 4.1: Piecewise linear reconstruction of the area in the central cell (i, j) in a 3×3 stencil. Normal vectors at the four corners of the central cell are computed using the gradients of the volume fraction values in the neighboring cells.

$$\begin{aligned}
 m_{i+\frac{1}{2}, j+\frac{1}{2}}^x &= \left[\frac{\partial F}{\partial x} \right]_{i+\frac{1}{2}, j+\frac{1}{2}} = \frac{1}{2\Delta x} [F_{i+1, j+1} - F_{i, j+1} + F_{i+1, j} - F_{i, j}] \\
 m_{i-\frac{1}{2}, j+\frac{1}{2}}^x &= \left[\frac{\partial F}{\partial x} \right]_{i-\frac{1}{2}, j+\frac{1}{2}} = \frac{1}{2\Delta x} [F_{i+1, j} - F_{i, j} + F_{i+1, j-1} - F_{i, j-1}] \\
 m_{i-\frac{1}{2}, j-\frac{1}{2}}^x &= \left[\frac{\partial F}{\partial x} \right]_{i-\frac{1}{2}, j-\frac{1}{2}} = \frac{1}{2\Delta x} [F_{i, j} - F_{i-1, j} + F_{i, j-1} - F_{i-1, j-1}] \\
 m_{i+\frac{1}{2}, j-\frac{1}{2}}^x &= \left[\frac{\partial F}{\partial x} \right]_{i+\frac{1}{2}, j-\frac{1}{2}} = \frac{1}{2\Delta x} [F_{i, j+1} - F_{i-1, j+1} + F_{i, j} - F_{i-1, j}].
 \end{aligned} \tag{4.4}$$

The required cell-centered x -direction component of the normal is then computed from the cell-corner values by averaging as

$$m_{i,j}^x = \frac{1}{4} \left[m_{i+\frac{1}{2},j+\frac{1}{2}}^x + m_{i-\frac{1}{2},j+\frac{1}{2}}^x + m_{i-\frac{1}{2},j-\frac{1}{2}}^x + m_{i+\frac{1}{2},j-\frac{1}{2}}^x \right]. \quad (4.5)$$

Similarly, the cell-centered y -direction component of the normal is evaluated according to

$$m_{i,j}^y = \frac{1}{4} \left[m_{i+\frac{1}{2},j+\frac{1}{2}}^y + m_{i-\frac{1}{2},j+\frac{1}{2}}^y + m_{i-\frac{1}{2},j-\frac{1}{2}}^y + m_{i+\frac{1}{2},j-\frac{1}{2}}^y \right] \quad (4.6)$$

in which

$$\begin{aligned} m_{i+\frac{1}{2},j+\frac{1}{2}}^y &= \left[\frac{\partial F}{\partial y} \right]_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{2\Delta y} [F_{i+1,j+1} - F_{i,j+1} + F_{i+1,j} - F_{i,j}] \\ m_{i-\frac{1}{2},j+\frac{1}{2}}^y &= \left[\frac{\partial F}{\partial y} \right]_{i-\frac{1}{2},j+\frac{1}{2}} = \frac{1}{2\Delta y} [F_{i+1,j} - F_{i,j} + F_{i+1,j-1} - F_{i,j-1}] \\ m_{i-\frac{1}{2},j-\frac{1}{2}}^y &= \left[\frac{\partial F}{\partial y} \right]_{i-\frac{1}{2},j-\frac{1}{2}} = \frac{1}{2\Delta y} [F_{i,j} - F_{i-1,j} + F_{i,j-1} - F_{i-1,j-1}] \\ m_{i+\frac{1}{2},j-\frac{1}{2}}^y &= \left[\frac{\partial F}{\partial y} \right]_{i+\frac{1}{2},j-\frac{1}{2}} = \frac{1}{2\Delta y} [F_{i,j+1} - F_{i-1,j+1} + F_{i,j} - F_{i-1,j}]. \end{aligned} \quad (4.7)$$

are the y -direction components at the corners of the control volume. The extension of this method to three dimensions is straightforward. In three dimensions, the gradients are calculated in a $3 \times 3 \times 3$ local cell cluster to estimate the cell-centered normal vector.

This method was investigated by, e.g., [46], [54], [55], [56], and was shown to be only first-order accurate. Nevertheless, for grids which have low to medium resolution, amplitudes of the errors produced by this method are comparable to second-order methods, see [46], [54] and [56].

4.2.2. LEAST-SQUARE METHOD BY RIDER AND KOTHE

Rider and Kothe [44] extended Youngs' finite difference approximations [57] for the volume fraction gradient to two- and three-dimensional arbitrary-connectivity (unstructured) meshes based on the work by Barth [58]. In this method, volume fraction Taylor series expansions are formed from each reference cell volume fraction F_i at point $\mathbf{x}_i$ to each cell neighbor F_s at point $\mathbf{x}_s$. The volume fraction gradient $\mathbf{m} = \nabla F_i$ is then calculated by solving a weighted least squares problem. In two dimensions, it can be written as

$$A^T A \mathbf{m} = A^T \mathbf{b} \quad (4.8)$$

where

$$\begin{aligned}
 A &= \begin{pmatrix} \omega_s (x_s - x_i) & \omega_s (y_s - y_i) \\ \vdots & \vdots \\ \omega_n (x_n - x_i) & \omega_n (y_n - y_i) \end{pmatrix} \\
 \mathbf{b} &= \begin{pmatrix} \omega_s (F_s - F_i) \\ \vdots \\ \omega_n (F_n - F_i) \end{pmatrix} \\
 \mathbf{m} &= \begin{pmatrix} m_i^x \\ m_i^y \\ m_i^z \end{pmatrix}
 \end{aligned} \tag{4.9}$$

in which

$$\omega_s = \frac{1}{|\mathbf{x}_s - \mathbf{x}_i|^2} \tag{4.10}$$

for a stencil involving n cells surrounding the cell where the gradient is being evaluated. The solution of this linear system (4.8) provides the normal components in the central cell (the linear system is 2×2 in 2D and 3×3 in 3D). The normal is then normalized by $\sqrt{(m_i^x)^2 + (m_i^y)^2}$. Similar to the Parker and Youngs' method, this method also uses 3×3 cell stencil in 2D and $3 \times 3 \times 3$ cell stencil in 3D.

Although it is powerful, attractive and independent of any particular mesh topology or dimensionality, [Rider and Kothe](#) [44] obtained first-order accuracy using the above least squares concept in a number of test cases. However, their results from static reconstruction tests indicate that the errors exhibited by this method on the coarsest grids are lower than those exhibited by genuinely second-order methods. They also mention that in regions of large curvatures (relative to mesh spacing) this method is generally quite robust.

4.2.3. MIXED YOUNGS-CENTERED (MYC) METHOD

This method exploits the strengths of both the Parker and Youngs' method and height function approach to compute the interface normal in a fast and accurate fashion. In three dimensions, this method chooses among a set of four different candidate planes, three of which are calculated with the height function approach and the last with the Parker and Youngs' method. Only after the choice for the normal vector is made, the plane constant is calculated and the reconstruction of the interface in that cell is completed.

In the height function approach, local interface heights are estimated by summing the volume fractions columnwise or rowwise in a block of cells. Components of the normal vector are then approximated with finite differences using these local interface heights. Let us consider the local $3 \times 3 \times 3$ block of cells illustrated in Fig. 4.2. The interface height in, for example, the z -direction can be estimated by

$$\bar{z}_{i,j} = \sum_{l=-1}^1 F_{i,j,k+l} \Delta z_{i,j,k+l}. \tag{4.11}$$

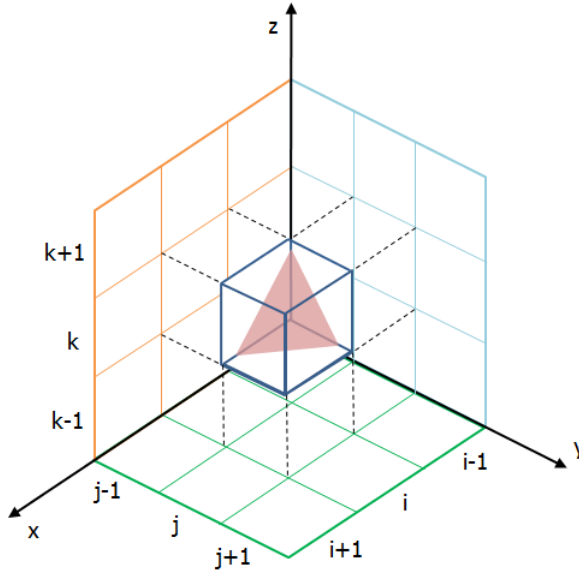


Figure 4.2: A $3 \times 3 \times 3$ cell cluster used in a typical reconstruction algorithm.

If the planar interface is defined by the linear equation $\text{sgn}(m_z)z = m_x x + m_y y + \alpha$ (the z -component of the normal has been set to 1, and the sign is calculated by central differencing of the F data summed along the top and bottom layers of the block of cells), a central differencing scheme may be used to approximate m_x and m_y

$$m_x = \frac{\bar{z}_{i+1,j} - \bar{z}_{i-1,j}}{x_{i+1} - x_{i-1}} \quad (4.12)$$

and

$$m_y = \frac{\bar{z}_{i,j+1} - \bar{z}_{i,j-1}}{y_{j+1} - y_{j-1}} \quad (4.13)$$

where x_{i+1} , x_{i-1} , y_{j+1} and y_{j-1} are located at the centers of the cells. As a result, we have the first candidate normal vector $(\text{sgn}(m_z), m_x, m_y)$. Similarly, the interface heights along the other directions are considered and two more candidate normal vectors are computed.

To make a choice between the four candidate normal vectors, two criteria are employed. First, the candidate vectors are normalized, for example the vector $(\text{sgn}(m_z), m_x, m_y)$ is normalized to (m_z^0, m_x', m_y') , with the sum of the absolute value of the three components equal to 1. Then, between the three components computed with the height function approach, the following criterion yields the optimum choice

$$|m^*| = \max(|m_x^0|, |m_y^0|, |m_z^0|). \quad (4.14)$$

Afterwards, to select between this chosen candidate and the normal computed with the Parker and Youngs' method, the following criterion is employed (let us assume that

$$|m^*| = |m_x^0| \quad \left| m^{final} \right| = \min \left(\left| m_x^0 \right|, \left| m_{x(youngs)}^0 \right| \right). \quad (4.15)$$

The MYC method was proposed by [Aulisa et al. \[49\]](#), and later used by [Popinet \[59\]](#) and [Wang et al. \[60\]](#). Results by [Aulisa et al. \[49\]](#) illustrate that this method performs similarly to their truly second-order reconstruction technique (LSF), and outperforms both the Youngs' and height function methods. In fact, LSF is an iterative scheme and actually uses the interface normal generated by MYC as an initial solution to the iterative process. In short, the MYC method is a good compromise between accuracy and computational cost.

4.2.4. EFFICIENT LEAST SQUARES INTERFACE RECONSTRUCTION ALGORITHM (ELVIRA)

Let us consider again the mesh arrangement in Fig. 4.1. Using the height function approach, local interface heights are estimated by summing the volume fraction values in each row and column of the stencil. With respect to the MYC method, not only the central differencing but also the backward and forward differencing of these local interface heights are considered to approximate the slope of the linear interface. Let us first consider the interface heights along the y -direction, and compute three candidates for the slope m_x of a line $\text{sgn}(m_y)y = m_x x + \alpha$ as; forward differencing

$$m_{xf} = \frac{\bar{y}_{i+1,j} - \bar{y}_{i,j}}{x_{i+1} - x_i}, \quad (4.16)$$

central differencing

$$m_{xc} = \frac{\bar{y}_{i+1,j} - \bar{y}_{i-1,j}}{x_{i+1} - x_{i-1}}, \quad (4.17)$$

backward differencing

$$m_{xb} = \frac{\bar{y}_{i,j} - \bar{y}_{i-1,j}}{x_i - x_{i-1}}. \quad (4.18)$$

Analogously, the interface heights along the x -direction can be utilized to calculate three candidates for the slope m_y of a line $\text{sgn}(m_x)x = m_y y + \alpha$ as; forward differencing

$$m_{yf} = \frac{\bar{x}_{i,j+1} - \bar{x}_{i,j}}{y_{j+1} - y_j}, \quad (4.19)$$

central differencing

$$m_{yc} = \frac{\bar{x}_{i,j+1} - \bar{x}_{i,j-1}}{y_{j+1} - y_{j-1}}, \quad (4.20)$$

backward differencing

$$m_{yb} = \frac{\bar{x}_{i,j} - \bar{x}_{i,j-1}}{y_j - y_{j-1}}. \quad (4.21)$$

For each one of the six candidates, the plane constant α is computed, and the interface is extended to the edges of the 3×3 block of cells. As a result, a temporary volume fraction distribution $\tilde{F}$ is generated in the neighboring eight cells for each candidate. The

selected slope then minimizes the discrete error E in L_2 defined by the following least-squares expression

$$E(\tilde{m}) = \left(\sum_{k,l=-1}^1 (\tilde{F}_{i+k,j+l}(\tilde{m}) - F_{i+k,j+l})^2 \right)^{1/2} \quad (4.22)$$

where $\tilde{m}$ is one of the six candidate slopes. The minimization procedure can be written as

$$\mathbf{m} = \min(E(m_{xf}), E(m_{xc}), E(m_{xb}), E(m_{yf}), E(m_{yc}), E(m_{yb})) \quad (4.23)$$

where $\mathbf{m}$ is the adopted slope. [Pilliod and Puckett \[46\]](#) mention that instead of the discrete L_2 error, discrete L_1 or L_∞ errors can also be chosen in the minimization.

The ELVIRA algorithm reconstructs linear interfaces exactly. However, the algorithm is originally designed in 2D. Later, it was extended to 3D by [Miller and Colella \[61\]](#). The 3D ELVIRA method relies on a $5 \times 5 \times 5$ stencil, and requires the use of 72 to 144 candidate normal vectors. The discrete error E in L_2 described above is calculated for each candidate normal vector on a $5 \times 5 \times 5$ stencil, which automatically entails the computation of the plane constants in each cell in the stencil. This approach is rather expensive, and the large stencil causes difficulties with resolving interface details smaller than the characteristic size of the stencil. [Cervone et al. \[53\]](#) propose a *reduced* version of this algorithm which relies only on a $3 \times 3 \times 3$ block of cells. Following their lead, let us write the plane equation as, for example, $\text{sgn}(m_z)z = m_x x + m_y y + \alpha$, and consider again the local discrete height function given by Eq. (4.11). Using only the backward, central and forward differencing, the following candidates are computed for m_x and m_y for the central cell of the structured mesh in Fig. 4.2

$$m_{xf} = \frac{\bar{z}_{i+1,j} - \bar{z}_{i,j}}{x_{i+1} - x_i}, \quad (4.24)$$

$$m_{xc} = \frac{\bar{z}_{i+1,j} - \bar{z}_{i-1,j}}{x_{i+1} - x_{i-1}}, \quad (4.25)$$

$$m_{xb} = \frac{\bar{z}_{i,j} - \bar{z}_{i-1,j}}{x_i - x_{i-1}}, \quad (4.26)$$

$$m_{yf} = \frac{\bar{z}_{i,j+1} - \bar{z}_{i,j}}{y_{j+1} - y_j}, \quad (4.27)$$

$$m_{yc} = \frac{\bar{z}_{i,j+1} - \bar{z}_{i,j-1}}{y_{j+1} - y_{j-1}}, \quad (4.28)$$

$$m_{yb} = \frac{\bar{z}_{i,j} - \bar{z}_{i,j-1}}{y_j - y_{j-1}}. \quad (4.29)$$

This indicates that using the local interface heights along the z -direction, we will have a total number of 9 candidate normal vectors. Applying the same approach to the local interface heights along the other directions yields a total 27 candidates. For the 28th candidate, [Cervone et al. \[53\]](#) use the Parker and Youngs' method for its high accuracy at low resolution. The discrete error E (4.22) is calculated for each candidate normal

vector on a $3 \times 3 \times 3$ cluster of cells, and the subsequent minimization procedure results in the adopted normal vector. This relatively cheap non-iterative method does not reconstruct all planar interfaces exactly, but is attractive especially in 3D considering the small number of candidate normal vectors and more compact stencil compared to the approach by Miller and Colella [61]. Therefore in this work, this approach is preferred, and referred to as simply ELVIRA for the sake of brevity.

4.2.5. COMPUTATION OF THE PLANE CONSTANT

Once the normal vector is known, the planar interface within the cell is located so that local volume conservation is satisfied. In other words, the resulting plane should pass through the cell in such a way that the truncated volume lying below the plane is equal to the exact material volume in that cell. In three dimensions, the intersection of a plane with a cube is an inscribed, irregular polygon with one of four basic shapes which have 3 to 6 intersection points, as shown in Fig. 4.3.

As Eq. (4.3) suggests, the location of the planar interface results from the computation of α . With the available knowledge of the normal vector $\mathbf{m}$ and the volume fraction F within the cell, α can be calculated either iteratively or analytically.

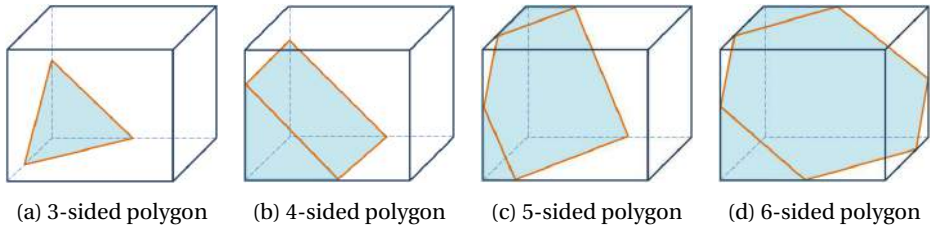


Figure 4.3: Possible candidates of irregular polyhedron within a cubic mesh cell.

Let us consider that α represents the plane constant in a reference cell and normal components in that cell are calculated using a reconstruction method. In this case, the cut cell volume below the plane can be represented as $V(\alpha)$ which can vary linearly, quadratically or cubically with α , depending upon the coordinate system and the shape of the n -sided polygon formed by the plane in the cell. Also, let us consider that the exact cell material volume is given as V^e . Then, the plane constant α is obtained when the generally nonlinear function

$$g(\alpha) = V(\alpha) - V^e \quad (4.30)$$

becomes zero. In other words, when the above volumes are equal (to within some tolerance), the interface location in that cell is fixed.

Several approaches are proposed to find the zero of the above function. These methods can be classified as analytical methods and iterative methods. Rider and Kothe [44] proposed the use of Brent's scheme as in [62], which includes a combination of bisection and inverse quadratic interpolation methods to find a near-optimal next guess for α . They also report that Newton's method does not guarantee convergence. An algorithm based on secant and bisection methods is given in [63].

Gueyffier *et al.* [54] introduce an analytical approach to evaluate α although not in detail. Extensive studies regarding analytical relations for the forward and inverse problems are performed in [64, 65] (forward problem refers to finding the volume fraction for a given α and inverse problem refers to finding α given the volume fraction). They report that analytical relations are computationally cheaper than a fast root-finding technique. In a recent study, Popinet [59] reports that the forward and inverse routines by Scardovelli and Zaleski [64] have been well tested and are free from inconsistencies which may occur due to round-off errors in limiting cases. In this work we will also use these routines, see Appendix A.

4.3. INTERFACE ADVECTION

After we determine the orientation and location of the planar interface, the volume fraction field is advected in time via Eq. (4.2). This is the second stage of a VOF method. Algorithms for the transport problem may generally be divided into two categories: unsplit schemes and operator split schemes. In operator splitting, the transport of the volume fraction field is realized by considering sequential updates along each direction with calculating one dimensional fluxes by treating only one velocity vector component in each update. This inherent feature makes operator splitting applicable only within structured mesh environments. After advancing the interface along each coordinate direction, intermediate volume fraction values are computed. As a result, three consecutive updates are required to transport the interface to the next discrete time level in three dimensions, $F^n \rightarrow F^* \rightarrow F^{**} \rightarrow F^{n+1}$, which also necessitates at least three interface reconstruction sweeps based on the corresponding intermediate volume fraction fields. Additionally, the sequence of updates in each direction must be changed in order to avoid, or at least minimize, asymmetries caused by the operator splitting; [44], [66]. For this purpose, six possible permutations of the sweep sequence in three directions need to be realized; [66], [67]. In unsplit advection, however, fluxes are calculated using the velocity vector components acting in all directions. Therefore, several neighboring cells may contribute to a multidimensional flux volume across a mesh cell face over a time step which is truncated by an arbitrarily-oriented planar interface. This procedure constitutes the fundamental bottleneck of unsplit advection since the resulting multidimensional flux volume may have a very complex geometry. Once multidimensional flux volumes through each mesh cell face are determined, the interface is propagated along all coordinate directions in a single update, $F^n \rightarrow F^{n+1}$, which entails only one interface reconstruction sweep. This clearly brings a significant advantage in terms of computational costs. Compared to unsplit advection, operator splitting is more common among researchers and well-documented.

Irrespective of the transport strategy, an advection algorithm should have certain fundamental properties. The total mass must be conserved in the discretized form of the advection equation without a posteriori numerical treatment. Constancy must be preserved; an initially uniform scalar field characterized by Eq. (4.2) should remain uniform in a solenoidal velocity field. Finally, an advection scheme should not generate unphysical undershoots or overshoots, i.e., the volume fraction field must always be bounded everywhere: $0 \leq F \leq 1$. This is referred to as *shape preservation*; [66]. Unfortunately, it is difficult to design an advection scheme that satisfies all criteria simul-

taneously. Sometimes, researchers prefer to develop algorithms that include numerical recipes to compensate shortcomings of proposed advection schemes. For example, when the third condition is not satisfied, excess values are clipped or dispensed over several neighboring cells, e.g., [van der Pijl *et al.* \[68\]](#) present a mass redistribution algorithm to account for overshoots and undershoots without violating mass conservation, [Rider and Kothe \[44\]](#) also search for and conservatively redistribute any volume fraction excess values ($F < 0$ and/or $F > 1$), and [Scardovelli and Zaleski \[56\]](#) only mention such a local redistribution procedure. When mass is redistributed, also momentum is likely to change. Hence, although mass conservation is repaired, momentum conservation is lost. Another approach to circumvent this difficulty is to combine the advection method with a flux correction scheme, e.g., [Lörsting and Fuchs \[69\]](#) introduce such a scheme that is embedded into the procedure for the computation of the flux volume. [Rudman \[32\]](#) also discusses this subject and presents an idea to overcome this problem which is based on the concept of allowing volume of a cell to change effectively during each one-dimensional sweep of advection. He mentions that even after this measure small round-off errors can accumulate and affect the boundedness of the volume fraction field later in the simulation. Furthermore, advection methods may sometimes induce mass (volume fraction) conservation problems. A popular approach to account for this behavior is, for example, to add a correction term to the last update of an operator split advection procedure. Such an advection method is used by, e.g., [\[39, 68, 70\]](#). However, this modification comes at the expense of the consequence that the volume fraction values are no longer guaranteed to be bounded. In the end, an additional correction sweep is required to surmount this consequence as both interface reconstruction and flux calculation depend on bounded volume fraction values. Further details of the discussion about the useful attributes which an advection scheme should retain can be found in [\[66\]](#), [\[71\]](#), [\[72\]](#), and [\[73\]](#).

Both strategies for advection have been investigated by many researchers. In this work, we restrict the discussion regarding advection methods to operator split schemes. For unsplit advection techniques, the reader is referred to the works in, e.g., [\[44\]](#), [\[46\]](#), [\[48\]](#), [\[52\]](#), [\[61\]](#), [\[74\]](#), [\[75\]](#), [\[76\]](#), [\[77\]](#), and [\[78\]](#).

[Puckett *et al.* \[79\]](#) propose an operator split and an unsplit advection method. The discrete form of their split method in 2D is

$$\begin{aligned} F^* &= F^n - \Delta t \frac{\partial u F^n}{\partial x} + \Delta t F^* \frac{\partial u}{\partial x}, \\ F^{n+1} &= F^* - \Delta t \frac{\partial v F^*}{\partial y} + \Delta t F^* \frac{\partial v}{\partial y}. \end{aligned} \tag{4.31}$$

where Δt is the time step, the superscripts n , $*$ and $n+1$ indicate the previous, intermediate and new time level, respectively. The above discretization contains the implicit time integration in the x -direction followed by the explicit time integration in the y -direction, and the fluxes in both steps are calculated in the Eulerian fashion. Summing these two steps yields

$$\frac{F^{n+1} - F^n}{\Delta t} + \left(\frac{\partial u F^n}{\partial x} + \frac{\partial v F^*}{\partial y} \right) = F^* \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right). \tag{4.32}$$

If the velocity field is discretely divergence free, then the right hand side of (4.32) vanishes, and the mass conservation is satisfied since the left hand side is written in conservation form. The stretching terms in (4.31) are required because the volume fraction field is advected along each direction in the possibly divergent one-dimensional velocity field. Without these terms, overshoots and/or undershoots unavoidably occur even after the first sweep of the advection. This method is employed also in, e.g., [39], [68], [70], [73], [80], and [81]. Unfortunately, they all note that this advection scheme is not strictly conservative as small overshoots and/or undershoots may still occur. An analogous 3D representation can be given as

$$\begin{aligned}
 F^* &= F^n - \Delta t \frac{\partial u F^n}{\partial x} + \Delta t F^* \frac{\partial u}{\partial x}, \\
 F^{**} &= F^* - \Delta t \frac{\partial v F^*}{\partial y} + \Delta t F^{**} \frac{\partial v}{\partial y}, \\
 F^{***} &= F^{**} - \Delta t \frac{\partial w F^{**}}{\partial z} + \Delta t F^{***} \frac{\partial w}{\partial z}, \\
 F^{n+1} &= F^{***} - \Delta t \left(F^* \frac{\partial u}{\partial x} + F^{**} \frac{\partial v}{\partial y} + F^{***} \frac{\partial w}{\partial z} \right).
 \end{aligned} \tag{4.33}$$

Rider and Kothe [44] and Kothe *et al.* [82] use the following advection scheme

$$\begin{aligned}
 F^* &= F^n - \Delta t \frac{\partial u F^n}{\partial x} + \Delta t F^n \frac{\partial u}{\partial x}, \\
 F^{n+1} &= F^* - \Delta t \frac{\partial v F^*}{\partial y} + \Delta t F^{n+1} \frac{\partial v}{\partial y}.
 \end{aligned} \tag{4.34}$$

The only difference between (4.34) and (4.31) is that the order of integration in (4.34) is reversed (first explicit, then implicit). As a result, the sum of both steps also degenerates to (4.32). However, as noted in [44], [56], [59], and [73], this scheme is not perfectly conservative either. Extension of this method to 3D is straightforward.

Gueyffier *et al.* [54] propose a Lagrangian scheme based on the works by Chorin [36] and Li [83]. In this method, the interface is first advanced along a coordinate direction using the underlying one-dimensional velocity field. If the one-dimensional dilatation term, for example $\partial u / \partial x$, is not zero, then the interface segment changes its location and orientation. Using the new interface segment and the velocity values on the two faces of the cell, one can calculate the flux volumes which have transported into or from the two adjacent cells, and what remains inside the cell. Once these volumes are determined, the volume fraction field is easily advected in the coordinate direction. The scheme can be extended to 3D in a simple fashion by applying the same concept to the other coordinate directions. Detailed explanation of the method is given in [54]. The Lagrangian method is investigated in, for example, [55], [56], [84], and [85]. Wang *et al.* [60, 86] improve the accuracy of the method in time (Gueyffier *et al.* [54] prove the first-order accuracy) by employing a second-order Runge-Kutta scheme. de Niem *et al.* [87] analyze the effects of cylinder coordinates on the Lagrangian advection.

Scardovelli and Zaleski [56] introduce the Eulerian Implicit-Lagrangian Explicit (EI-LE) method. While the discrete form of the EI-LE method in 2D is the same as (4.31), the flux calculation is different. For the implicit integration in the first step of the advection method (4.31) fluxes are calculated in the Eulerian manner, and for the explicit integration in the second step a Lagrangian flux treatment is employed as explained above. Scardovelli and Zaleski [56] show that this method conserves the global volume fraction exactly, and other operator split methods in the literature do not possess this attribute. On the other hand, this method does not naturally extend to 3D, although Aulisa *et al.* [49] propose a numerical workaround to this issue. The EI-LE scheme is also used by Cervone *et al.* [53] and Aulisa *et al.* [88].

Interestingly, the simplest operator split algorithm to approximate solutions of Eq. (4.2) is sometimes implemented by researchers. In 2D, this algorithm takes the form

$$\begin{aligned} F^* &= F^n - \Delta t \frac{\partial u F^n}{\partial x}, \\ F^{n+1} &= F^* - \Delta t \frac{\partial v F^*}{\partial y}. \end{aligned} \quad (4.35)$$

If the one-dimensional dilatation terms do not vanish, for example if $\partial u / \partial x$ and $\partial v / \partial y$ are not zero (they generally are not), then this scheme will induce overshoots and/or undershoots after each step of the advection. Nonetheless, it is considered in [37], [46], and [89].

Leonard *et al.* [66] introduce two operator split advection methods which, to the best knowledge of the author, have not been used in the context of interfacial or free surface flows. The schemes are Multidimensional Advective-Conservative Hybrid Operator (MACHO) and Conservative Operator Splitting for Multidimensions with Inherent Constancy (COSMIC). Leonard *et al.* [66] state that there are similarities between COSMIC and Extended Numerical Integration for Genuinely Multidimensional Advective Transport Insuring Conservation (ENIGMATIC) [90], the Flux-Integral Method (FIM) of [91], and the so-called "flux-form semi-Lagrangian" method in [71]. In 2D, the MACHO and COSMIC schemes take the following form:

- the 2D MACHO scheme:

$$\begin{aligned} F^* &= F^n - \Delta t \frac{\partial u F^n}{\partial x} + \Delta t F^n \frac{\partial u}{\partial x}, \\ F^{n+1} &= F^n - \Delta t \left(\frac{\partial u F^n}{\partial x} + \frac{\partial v F^*}{\partial y} \right), \end{aligned} \quad (4.36)$$

- the 2D COSMIC scheme:

$$\begin{aligned} F^X &= F^n - \Delta t \frac{\partial u F^n}{\partial x} + \Delta t F^n \frac{\partial u}{\partial x}, \\ F^Y &= F^n - \Delta t \frac{\partial v F^n}{\partial y} + \Delta t F^n \frac{\partial v}{\partial y}, \\ F^{n+1} &= F^n - \Delta t \left[\frac{\partial}{\partial x} \left(u \frac{F^n + F^Y}{2} \right) + \frac{\partial}{\partial y} \left(v \frac{F^n + F^X}{2} \right) \right]. \end{aligned} \quad (4.37)$$

The MACHO scheme is similar to the splitting given by (4.34). Similar to the previous advection methods, MACHO requires the direction of propagation to be alternated at each time step to reduce directional bias. This, however, is not required by COSMIC because of its inherent symmetric form. By considering the transverse contribution to each flux in (4.37), COSMIC maintains multidimensional stability. On the other hand, the COSMIC scheme demands one more reconstruction sweep per time step in 2D than any other operator split advection schemes mentioned in this work. In 3D, the increase in computational cost becomes more significant with COSMIC. MACHO and COSMIC can be written as the following in 3D:

- the 3D MACHO scheme:

$$\begin{aligned} F^* &= F^n - \Delta t \frac{\partial u F^n}{\partial x} + \Delta t F^n \frac{\partial u}{\partial x}, \\ F^{**} &= F^* - \Delta t \frac{\partial v F^*}{\partial y} + \Delta t F^* \frac{\partial v}{\partial y}, \\ F^{n+1} &= F^n - \Delta t \left(\frac{\partial u F^n}{\partial x} + \frac{\partial v F^*}{\partial y} + \frac{\partial w F^{**}}{\partial z} \right), \end{aligned} \quad (4.38)$$

- the 3D COSMIC scheme:

$$\begin{aligned} F^X &= F^n - \Delta t \frac{\partial u F^n}{\partial x} + \Delta t F^n \frac{\partial u}{\partial x}, \quad F^Y = F^n - \Delta t \frac{\partial v F^n}{\partial y} + \Delta t F^n \frac{\partial v}{\partial y}, \\ F^Z &= F^n - \Delta t \frac{\partial w F^n}{\partial z} + \Delta t F^n \frac{\partial w}{\partial z}, \\ F^{XY} &= F^Y - \Delta t \frac{\partial u F^Y}{\partial x} + \Delta t F^Y \frac{\partial u}{\partial x}, \quad F^{YX} = F^X - \Delta t \frac{\partial v F^X}{\partial y} + \Delta t F^X \frac{\partial v}{\partial y}, \\ F^{XZ} &= F^Z - \Delta t \frac{\partial u F^Z}{\partial x} + \Delta t F^Z \frac{\partial u}{\partial x}, \quad F^{ZX} = F^X - \Delta t \frac{\partial w F^X}{\partial z} + \Delta t F^X \frac{\partial w}{\partial z}, \\ F^{YZ} &= F^Z - \Delta t \frac{\partial v F^Z}{\partial y} + \Delta t F^Z \frac{\partial v}{\partial y}, \quad F^{ZY} = F^Y - \Delta t \frac{\partial w F^Y}{\partial z} + \Delta t F^Y \frac{\partial w}{\partial z}, \\ F^{n+1} &= F^n - \Delta t \left[\frac{\partial}{\partial x} \left(u \frac{2F^n + F^Y + F^Z + F^{YZ} + F^{ZY}}{6} \right) \right. \\ &\quad + \frac{\partial}{\partial y} \left(v \frac{2F^n + F^X + F^Z + F^{XZ} + F^{ZX}}{6} \right) \\ &\quad \left. + \frac{\partial}{\partial z} \left(w \frac{2F^n + F^X + F^Y + F^{XY} + F^{YX}}{6} \right) \right]. \end{aligned} \quad (4.39)$$

The COSMIC scheme thus entails computing and storing of the three basic one-dimensional updates (F^X , F^Y and F^Z) and the six cross-coupling updates (F^{XY} , F^{YX} , F^{XZ} , F^{ZX} , F^{YZ}

and F^{ZY}), all of which are substituted into the single-step explicit update written in conservation form. While MACHO and other operator split algorithms demand three interface reconstruction sweeps at each time step in 3D, COSMIC demands nine interface reconstruction sweeps. Obviously, the symmetry feature of the COSMIC scheme comes at a substantial price. Leonard *et al.* [66] note that MACHO and COSMIC are not strictly shape preserving under circumstances such as large time steps and deformational velocity fields. However, they add that shape preservation errors are quite small in most practical situations.

Geometric interpretation of the fluxes in MACHO and COSMIC is explained next. For simplicity, let us consider the first relation in Eq. (4.36) and discretize it in a grid cell such as illustrated in Fig. 4.4

$$F_{i,j}^* = F_{i,j}^n - \Delta t \left[\frac{(uF)_{i+\frac{1}{2},j}^n - (uF)_{i-\frac{1}{2},j}^n}{\Delta x_{i,j}} \right] + \Delta t F_{i,j}^n \left[\frac{u_{i+\frac{1}{2},j}^n - u_{i-\frac{1}{2},j}^n}{\Delta x_{i,j}} \right] \quad (4.40)$$

where $u_{i+\frac{1}{2},j}$ is the velocity component at the center of the right cell face. Suppose that $u_{i+\frac{1}{2},j}$ is positive and divide the cell into two parts, with areas $u_{i+\frac{1}{2},j} \Delta t \Delta y_{i,j}$ on the right and $(\Delta x_{i,j} - u_{i+\frac{1}{2},j} \Delta t) \Delta y_{i,j}$ on the left. The amount of fluid contained in $u_{i+\frac{1}{2},j} \Delta t \Delta y_{i,j}$

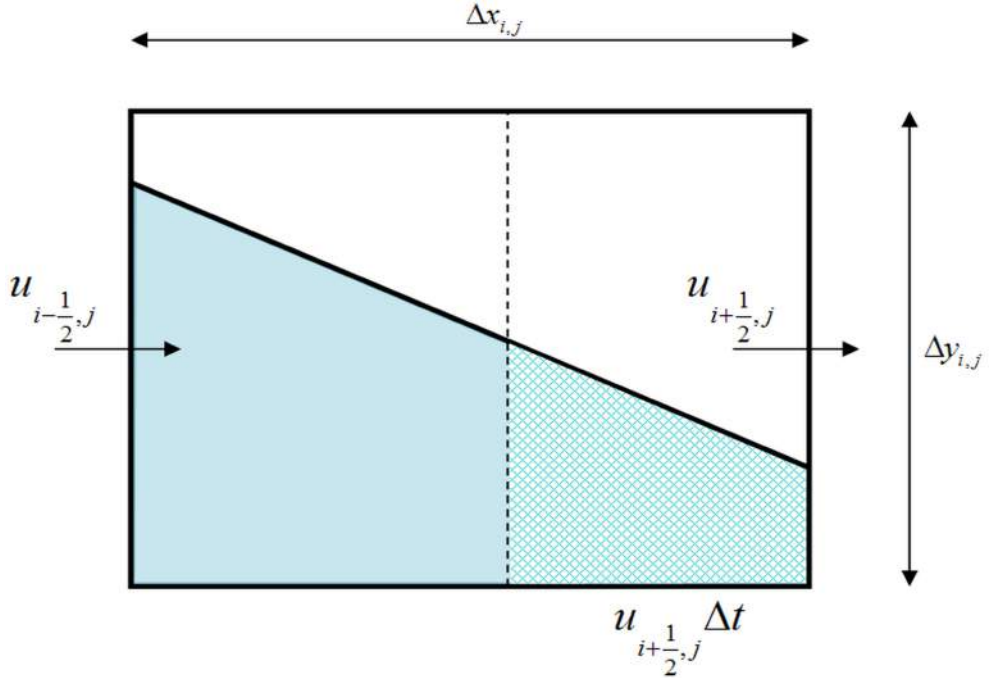


Figure 4.4: The amount of fluid illustrated as the cross-hatched area crosses the right cell edge in a splitting advection.

and illustrated as the cross-hatched area will be advected across the right cell face during

this time step. If $V_{i+\frac{1}{2},j}^n$ denotes the cross-hatched area corresponding to the initial F^n field, then the approximate volume fraction at the right cell face $F_{i+\frac{1}{2},j}^n$ can be written as

$$F_{i+\frac{1}{2},j}^n = \frac{V_{i+\frac{1}{2},j}^n}{u_{i+\frac{1}{2},j}^n \Delta t \Delta y_{i,j}}. \quad (4.41)$$

In a VOF method, the flux volume can be calculated by exploiting the location of the reconstructed interface. The limits for the flux volume can be stated as

$$V_{i+\frac{1}{2},j}^n = \min \left(\max \left(V_{i+\frac{1}{2},j}^n, 0 \right), u_{i+\frac{1}{2},j}^n \Delta t \Delta y_{i,j} \right). \quad (4.42)$$

After following the same process for the volume fraction at the left cell face and substituting the corresponding relations into Eq. (4.40), we obtain the following expression for the first step of the MACHO splitting

$$F_{i,j}^* = F_{i,j}^n - \left[\frac{V_{i+\frac{1}{2},j}^n - V_{i-\frac{1}{2},j}^n}{\Delta x_{i,j} \Delta y_{i,j}} \right] + \Delta t F_{i,j}^n \left[\frac{u_{i+\frac{1}{2},j}^n - u_{i-\frac{1}{2},j}^n}{\Delta x_{i,j}} \right]. \quad (4.43)$$

Now, consider the second step in the MACHO splitting in Eq. (4.36). Following the same procedure explained above, the discrete form of the second step is given as

$$F_{i,j}^{n+1} = F_{i,j}^* - \left[\frac{V_{i+\frac{1}{2},j}^n - V_{i-\frac{1}{2},j}^n}{\Delta x_{i,j} \Delta y_{i,j}} \right] - \left[\frac{V_{i,j+\frac{1}{2}}^* - V_{i,j-\frac{1}{2}}^*}{\Delta x_{i,j} \Delta y_{i,j}} \right] \quad (4.44)$$

where $V_{i,j+\frac{1}{2}}^*$ is the flux volume across the north cell face corresponding to the updated volume fraction field F^* , see Appendix B for more information on the computation of the flux volume.

4.4. NUMERICAL RESULTS

The order of accuracy of the reconstruction methods used in this work has been extensively studied by various researchers, see, e.g., [44], [46], [48], [49], [56], and [60]. Therefore, we will not present any static interface reconstruction tests and focus our attention on the kinematic simulations. In all advection tests presented here, velocity fields are defined in such a way that advected fluid bodies return to their initial shapes and locations at the end of the simulation time (often by multiplying the velocity with the Leveque cosine term; [92]). Therefore, a volume fraction distribution at the end of the simulation should be equivalent to the initial volume fraction distribution which can be considered as the exact solution. In order to compare the two distributions, the following error definitions are used

$$E = \sum_{i,j,k} |F_{i,j,k} - \tilde{F}_{i,j,k}| \Delta x_i \Delta y_j \Delta z_k, \quad (4.45)$$

$$E = \frac{\sum_{i,j,k} |F_{i,j,k} - \tilde{F}_{i,j,k}|}{\sum_{i,j,k} F_{i,j,k}}. \quad (4.46)$$

In both expressions, $F_{i,j,k}$ denotes the exact volume fraction distribution, and $\tilde{F}_{i,j,k}$ denotes the volume fraction distribution obtained by a pair of reconstruction/advection algorithms (simplification to 2D is straightforward). In order to have consistency with previous work, we will switch between the two formulae in the numerical tests.

In total, we present twelve alternative interface-reconstruction/advection combinations. Four interface reconstruction methods that we consider here are: Parker and Youngs' method (P&Y) [42], least squares gradient (LSG) method [44], mixed Youngs-centered (MYC) method [49] and Efficient least squares VOF interface reconstruction algorithm (ELVIRA) [46]. The ELVIRA scheme [46] was originally proposed in 2D, hence it will be used in 2D problems. For 3D problems we will consider the ELVIRA scheme in [53]. These interface reconstruction schemes will be combined with the three direction split advection schemes: MACHO and COSMIC [66], and EI-LE [56]. However, since the EI-LE scheme is inherently 2D, we will use only MACHO and COSMIC in 3D problems.

In addition to the above algorithms, we will show results obtained by using another VOF implementation in the CFD simulation tool ComFLOW. It employs the classical VOF technique introduced by Hirt and Nichols [23] and a local height function (LHF) to overcome the bottlenecks which originate from this VOF technique such as violation of mass conservation and spurious flotsam and jetsam. For a detailed description of the LHF, see [93]. The combination of this VOF method with the LHF will be referred to as H&N + LHF in the remainder of this chapter.

We will demonstrate the performance of the methods both qualitatively and quantitatively. For qualitative assessment, we will compare graphical results from numerical computations to the exact graphical solution. Figures will illustrate graphical results using $F = 0.5$ isosurfaces. For quantitative assessment, we will analyze the rate of convergence of the methods by using the error metrics given previously. The rate of convergence is computed using

$$O = \frac{\ln(E_{\Delta x}/E_{\Delta x/2})}{\ln(2)}, \quad (4.47)$$

where $E_{\Delta x/2}$ is the error obtained on a grid resolution of $\Delta x/2$, whereas $E_{\Delta x}$ is obtained on a grid resolution of Δx .

We will not focus our attention on the volume conservation since the methods are applied in such a way that volume is strictly conserved. In the numerical tests, we sometimes encountered mild overshoots and undershoots in volume fraction values after advecting the fluid configuration along a coordinate direction. Whenever and wherever this behavior is observed, the local height function is applied to ensure that these excess values are not thrown away.

4.4.1. ROTATION OF A SLOTTED DISK - 2D

The solid body rotation of the slotted disk problem, often referred to as Zalesak's test [29], has been commonly used by researchers such as [32], [41], [46], [56], and [60]. In this test, a slotted disk with a radius of 0.15, and a slot length of 0.05 and width of 0.25 is initially located at (0.5, 0.75) inside a unit sized box. The rotating flow is generated by a

constant vorticity velocity field which is given as:

$$\begin{aligned} u &= (\pi/3.14) (0.5 - y), \\ v &= (\pi/3.14) (x - 0.5). \end{aligned} \quad (4.48)$$

According to this configuration, the geometry makes a solid body rotation around the center of the unit sized box, and after 628 time units, it completes one revolution and is expected to return to its initial shape and position.

The velocity field in (4.48) has the one-dimensional incompressibility, $\partial u / \partial x = 0$ and $\partial v / \partial y = 0$. Therefore, there is no fluid shear introduced by the velocity field, and the interface topology should not change as a result of this advection. Furthermore, since the dilatation term in each one-dimensional update of an operator split advection method vanishes, the advection schemes discussed previously degenerate to similar expressions. Hence, this test becomes useful especially for assessing the convergence rate of reconstruction methods. As a result, only one advection method is considered in combination with several reconstruction algorithms.

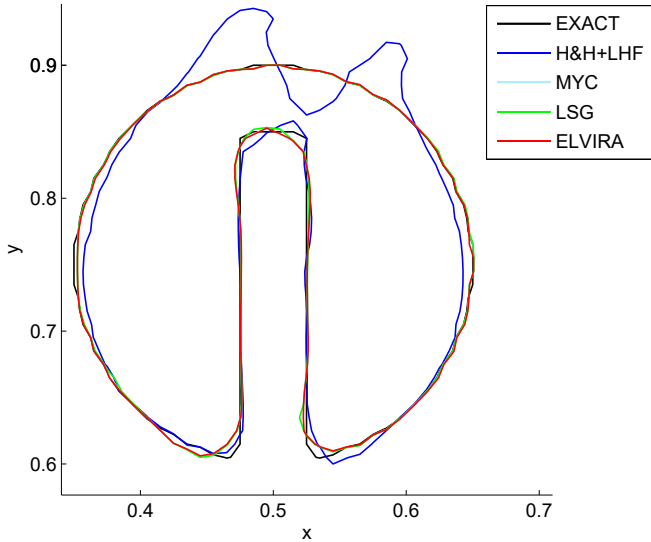


Figure 4.5: Solid body rotation of a slotted disk in 2D. Numerical results are obtained after one rotation on a 128^2 uniform grid with $CFL=0.5$. The PLIC methods are combined with the MACHO advection scheme.

The results after one rotation using four methods are illustrated in Fig. 4.5 in a close-up view. Graphical representation of the exact shape is also shown in black color. The results are obtained by combining the MACHO advection scheme with four reconstruction schemes; P&Y, LSG, MYC and ELVIRA. These methods are also compared to the Hirt and Nichols' VOF combined with the local height function (H&N + LHF). Here the grid size is 128×128 , and the CFL number is 0.5.

H&N + LHF performs quite poorly yielding significant distortion on the interface topology even on the smooth segments of the geometry. The results from the MACHO/PLIC schemes are considerably better. The profiles differ from the exact geometry only in regions with high curvature. We observe that in the corners of the slot, the MACHO/PLIC schemes fail to capture the sharp discontinuities, and smear them out over the neighboring cells. The difference between the PLIC schemes is hardly visible, which suggests that all the methods share similar characteristics in regions of sharp discontinuity.

Mesh	P&Y	MYC	LSG	ELVIRA	H&N+LHF
32 ²	1.64×10^{-2} <i>1.48</i>	1.66×10^{-2} <i>1.48</i>	1.68×10^{-2} <i>1.50</i>	1.65×10^{-2} <i>1.49</i>	8.26×10^{-2} <i>0.99</i>
64 ²	5.88×10^{-3} <i>1.07</i>	5.94×10^{-3} <i>1.10</i>	5.96×10^{-3} <i>1.11</i>	5.87×10^{-3} <i>1.12</i>	4.17×10^{-2} <i>1.05</i>
128 ²	2.81×10^{-3} <i>0.99</i>	2.78×10^{-3} <i>1.00</i>	2.77×10^{-3} <i>1.01</i>	2.71×10^{-3} <i>1.03</i>	2.01×10^{-2} <i>1.07</i>
256 ²	1.41×10^{-3}	1.39×10^{-3}	1.38×10^{-3}	1.33×10^{-3}	9.60×10^{-3}

Table 4.1: Nondimensional error (4.46) for the solid body rotation of a slotted disk problem. CFL is equal to 0.5. The PLIC methods are combined with the MACHO advection scheme. Rate of convergence (4.47) is written in italics between mesh entries.

Table 4.1 shows the nondimensional error (4.46) for the solid body rotation of a slotted disk problem. The first-order accuracy of the H&N + LHF method is well-established. Confirming the qualitative observation, the PLIC schemes produce comparable results in terms of both magnitude of nondimensional error and rate of convergence. This behavior was also observed in [46]. It appears that the errors from the four PLIC methods decrease at a rate which converges to 1.0 as the grid is refined.

4.4.2. SINGLE VORTEX TEST - 2D

After being introduced by Bell *et al.* [94], the single vortex test has been used by many researchers, e.g., [11], [32], [41], [44], [48], [56], and [95]. In the previous test, there was no fluid shear, therefore the geometry did not deform during its advection. In this test, we introduce shear into the velocity field in the form of a single vortex. A circle of radius 0.15 initially centered at point (0.5, 0.75) inside a unit sized box is subjected to the following non-uniform velocity field which imposes a single vortex in the domain,

$$\begin{aligned} u &= -\sin(2\pi y) \sin^2(\pi x) \cos\left(\frac{\pi t}{T}\right), \\ v &= \sin(2\pi x) \sin^2(\pi y) \cos\left(\frac{\pi t}{T}\right). \end{aligned} \tag{4.49}$$

The velocity field is multiplied by the so-called Leveque cosine term $\cos(\pi t/T)$ where T is the period at which the flow returns to its initial state [92]. This is merely a convenient

way to establish temporal accuracy. In this test, $T = 8$ is used. The circle stretches and spirals about the center of the domain reaching maximum deformation at $t = T/2 = 4$, and the Leveque cosine term reverses the velocity field returning the deformed body back to its initial state at $t = T = 8$. The velocity values are averaged at the cell faces so that the discrete incompressibility is satisfied. The CFL number computed using the maximum velocity in the domain is 1.

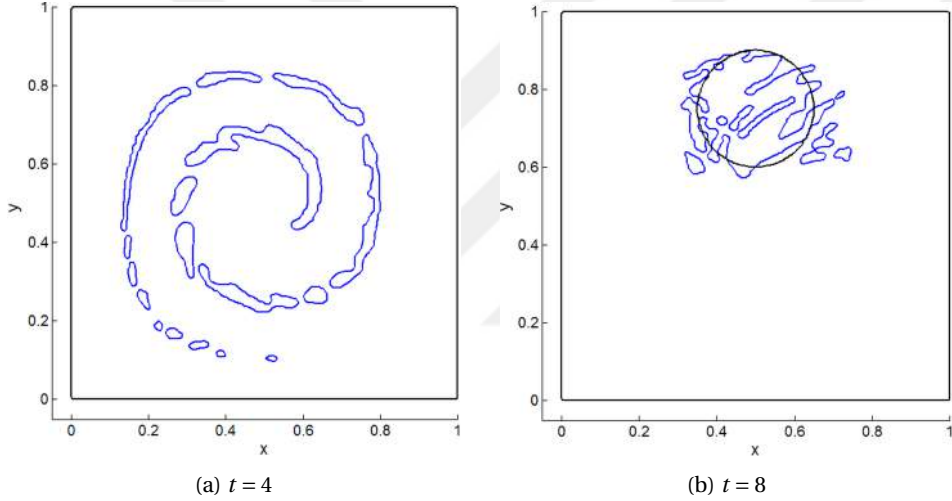


Figure 4.6: Profile at maximum deformation ($t = 4$), and after full flow reversal ($t = 8$) using the H&N+LHF method. The grid is 128×128 , and $CFL=1$.

Common use of the single vortex test among researchers is not surprising as it is a convenient yet challenging test to assess the ability of interface tracking methods in maintaining thin, elongated fluid filaments in a flow. In this test, these filaments are formed as a result of the vortex progressively stretching and wrapping the initial circular fluid body inward toward the vortex center. The duration of evolution from the initial configuration is controlled by the period T . Using large values for T indicates that the circle undergoes stretching for a long time, and the fluid filaments become thinner as more spirals are formed in the domain. Consequently, it becomes increasingly difficult for the interface tracking methods to bring the deformed body back to its initial state.

Figure 4.6 shows results from the H&N+LHF method on a 128×128 grid. At maximum deformation we observe a large amount of spurious fragmentation and coalescence, and at the end of the test the method completely fails to capture the initial shape.

Figure 4.7 illustrates profiles at maximum deformation at $t = 4$ using four reconstruction methods with the COSMIC advection scheme on a 128×128 grid at $CFL=1$. Results indicate that as the fluid filament becomes thinner, fragmentation occurs at the tail of the deformed geometry where the curvature is high and the interface is somewhat under-resolved. On the other hand, at the head of the geometry where the curvature is also high, we observe coalescence which results in a *blobby* structure. Figure 4.7 shows that the performances of the four PLIC methods are qualitatively similar. The amount

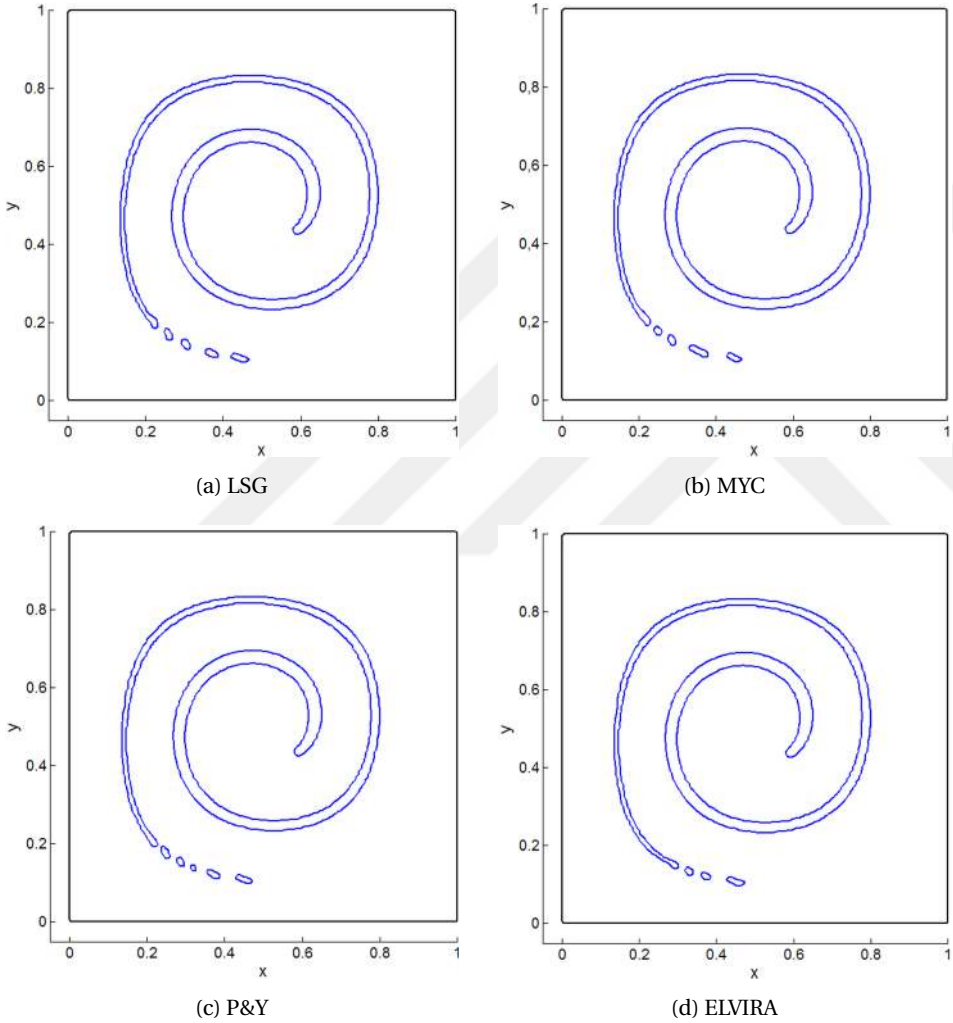


Figure 4.7: Profiles at maximum deformation ($t = 4$) using four reconstruction methods with the COSMIC advection scheme. The grid is 128×128 , and $CFL=1$.

of fragmentation is largest with the Parker and Youngs' method (P&Y), and smallest with the ELVIRA method.

When the deformed geometry is brought back to its initial shape and location at $t = 8$, the four reconstruction methods produce also somewhat similar results, see Fig. 4.8. The profiles indicate a slight phase shift compared to the initial circle.

Table 4.2 shows the geometrical error (4.45) from various VOF modules. To be able to compare our results with the literature, now we take $T = 2$. The data in the column with the header LSF/EI-LE are taken from [56] as benchmark results. Scardovelli and Zaleski [56] use their iterative linear Least-Square Fit (LSF) reconstruction scheme in combi-

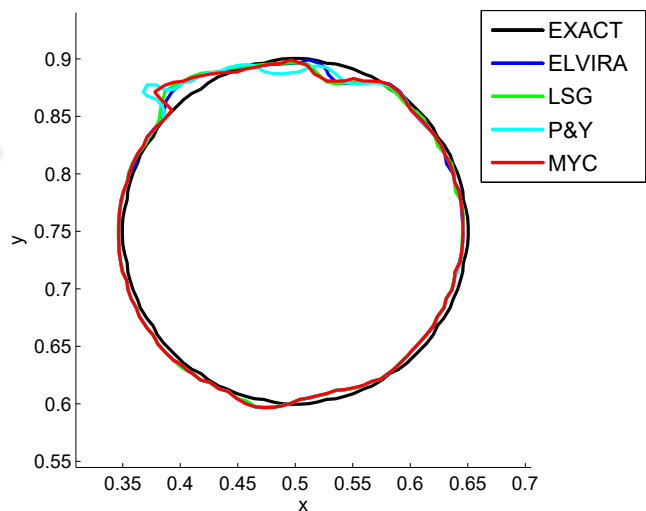


Figure 4.8: Profiles at $t = T = 8$ from four reconstruction methods with the COSMIC advection scheme. The grid is 128×128 , and $CFL=1$.

Mesh	LSG		ELVIRA		LSF	CVTNA	H&N+LHF
	COSMIC	EI-LE	COSMIC	EI-LE	EI-LE	PCFSC	
32^2	2.74×10^{-3} <i>1.97</i>	2.70×10^{-3} <i>1.96</i>	2.55×10^{-3} <i>1.97</i>	2.54×10^{-3} <i>1.97</i>	1.75×10^{-3} <i>1.91</i>	2.34×10^{-3} <i>2.12</i>	1.01×10^{-2} <i>0.94</i>
64^2	7.01×10^{-4} <i>1.84</i>	6.93×10^{-4} <i>1.87</i>	6.50×10^{-4} <i>2.11</i>	6.47×10^{-4} <i>2.16</i>	4.66×10^{-4} <i>2.19</i>	5.38×10^{-4} <i>2.03</i>	5.25×10^{-3} <i>1.09</i>
128^2	1.96×10^{-4}	1.89×10^{-4}	1.51×10^{-4}	1.45×10^{-4}	1.02×10^{-4}	1.31×10^{-4}	2.47×10^{-3}

Table 4.2: Error (4.45) for the 2D single vortex problem with $T = 2$. CFL is equal to 1. The results in the columns with the headers LSF/EI-LE and CVTNA/PCFSC are taken from [56] and [48], respectively. Rate of convergence (4.47) is written in italics between mesh entries.

nation with the EI-LE advection. In this work, LSF was applied only in 2D, but later Aulisa *et al.* [49] extended the method to the three-dimensional space. Another set of results is shown in the column with the header CVTNA/PCFSC taken from [48]. This work includes the implementation of the iterative Centroid-Vertex Triangle-Normal Averaging (CVTNA) for reconstruction, and the Piecewise-Constant Flux Surface Calculation (PCFSC) for unsplit advection. For the same advection method EI-LE, the LSG scheme performs the worst compared to ELVIRA and LSF in terms of both magnitude of geometrical error and rate of convergence. However, the difference is modest considering the fact that LSG is the cheapest and least complex among its counterparts. For the same reconstruction method, performances of the advection schemes COSMIC and EI-LE are

close. Although not shown in the table, this test was performed with MACHO as well, and the results were similar to those from COSMIC. The first-order accuracy of the H&N + LHF method is confirmed once again.

4.4.3. ROTATION OF A SLOTTED SPHERE - 3D

The solid body rotation of a slotted sphere test, also used in [11], is the extension of that explained in Section 4.4.1 to three dimensions. A slotted sphere of radius 0.15 and center (0.5, 0.75, 0.5) undergoes a rigid body rotation about the point (0.5, 0.5, 0.5) inside a unit sized cube. The slot is 0.05 wide and 0.25 deep. The rotating flow is formed by the following constant vorticity velocity field:

$$\begin{aligned} u &= (\pi/3.14) (0.5 - y), \\ v &= (\pi/3.14) (x - 0.5), \\ w &= 0. \end{aligned} \tag{4.50}$$

After 628 time units, the slotted sphere completes one revolution and is expected to return to its initial position without any deformation in surface topology. Figure 4.9 shows the results after one rotation on a $100 \times 100 \times 100$ uniform mesh with a CFL number equal to 0.5. Analogous with the two-dimensional problem, MACHO is used in combination with the four reconstruction methods: Parker and Youngs' method, LSG, MYC and ELVIRA. These methods are also compared to the Hirt and Nichols' VOF combined with the local height function (H&N + LHF). In the plots, geometries in blue color illustrate results from the numerical methods, while geometries in red color illustrate the exact solution.

Similar to the solid body rotation test in 2D, H&N + LHF performs quite poorly; the slotted sphere splits in half after one rotation. This and a large amount of coalescence results in a dramatically distorted geometry. Results from the MACHO/PLIC schemes are, however, significantly better. Smooth parts of the geometry are generally preserved after one rotation, whereas sharp corners of the slot are rounded. A typical feature of piecewise planar reconstruction is that it produces discontinuities at the cell boundaries especially in regions of high curvature. During advection these discontinuities are smeared out over the neighboring cells yielding smoother curvatures. Although this is a common property of the PLIC methods, the amount of smoothing may differ due to the design and accuracy of a reconstruction method.

Table 4.3 shows the nondimensional error (4.46). As was the case in the previous tests, the H&N + LHF method demonstrates a truly first-order behavior. Supporting the observations in the solid body rotation of Zalesak's disk in 2D, the PLIC methods tend to yield first-order accuracy as the grid is refined. The magnitudes of the nondimensional error from these methods are also similar.

Table 4.4 shows CPU timings to present a comparison of the cost of the reconstruction methods. These values are obtained by measuring the time spent only on the interface reconstruction process in a single time step. The results are presented in two different ways. In Table 4.4a, P&Y timing on the coarsest grid is used to normalize all the timings, whereas in Table 4.4b P&Y timing for a mesh size is used to normalize all

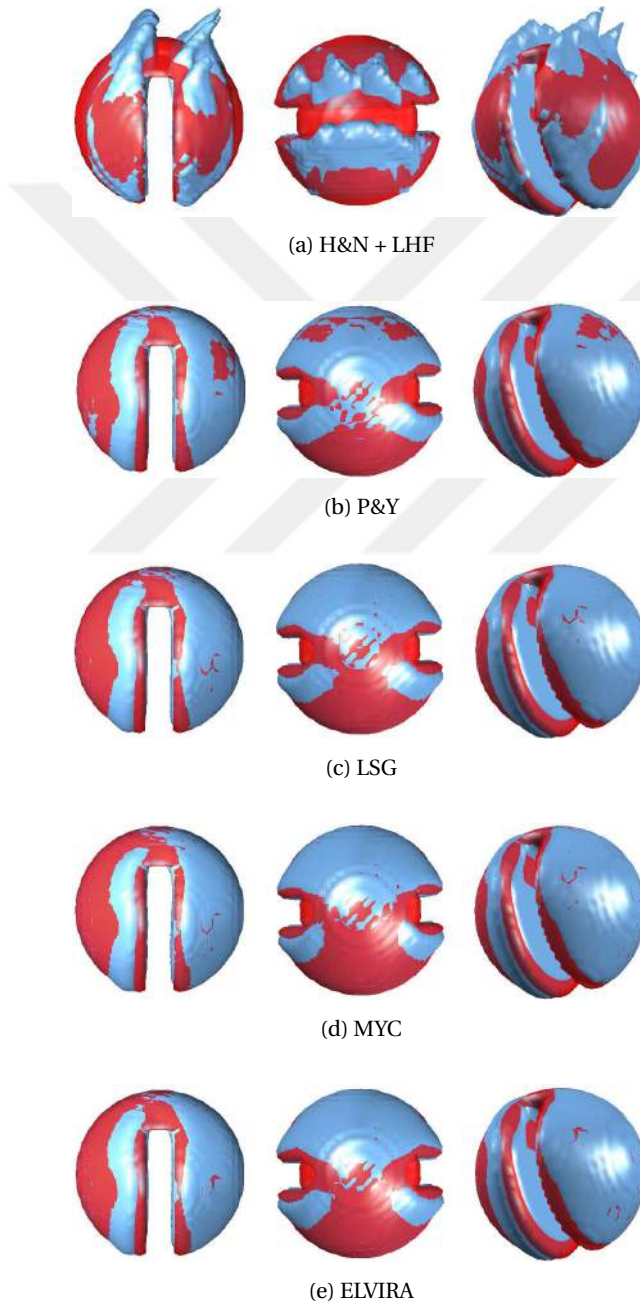


Figure 4.9: Solid body rotation of a slotted sphere after one revolution. Geometries in blue color indicate numerical results, and geometries in red color exact solution. Numerical results are obtained after one rotation on a 100^3 uniform grid at $CFL=0.5$. The PLIC methods are combined with the MACHO advection scheme.

Mesh	P&Y	MYC	LSG	ELVIRA	H&N+LHF
25^3	4.29×10^{-3} <i>1.48</i>	4.17×10^{-3} <i>1.48</i>	4.02×10^{-3} <i>1.41</i>	4.21×10^{-3} <i>1.57</i>	1.18×10^{-2} <i>1.25</i>
50^3	1.54×10^{-3} <i>1.24</i>	1.49×10^{-3} <i>1.21</i>	1.51×10^{-3} <i>1.25</i>	1.42×10^{-3} <i>1.22</i>	4.96×10^{-3} <i>1.11</i>
100^3	6.51×10^{-4}	6.46×10^{-4}	6.35×10^{-4}	6.11×10^{-4}	2.30×10^{-3}

Table 4.3: Nondimensional error (4.46) for the solid body rotation of a slotted sphere problem. CFL is equal to 0.5. The PLIC methods are combined with the MACHO advection scheme. Rate of convergence (4.47) is written in italics between mesh entries.

	Mesh	P&Y	MYC	LSG	ELVIRA
(a)	25^3	1.00	1.54	1.71	7.27
	50^3	4.23	7.15	7.95	29.23
	100^3	16.54	29.44	32.42	111.48
(b)	25^3	1.00	1.54	1.71	7.27
	50^3	1.00	1.69	1.88	6.91
	100^3	1.00	1.78	1.96	6.74

Table 4.4: Relative CPU timings from single-processor computations. (a) P&Y timing on the coarsest grid is taken as unity, and other timings are written relative to that. (b) P&Y timing for each mesh size is taken as unity, and other timings are written relatively.

the timings for that mesh size. The reason why P&Y timing is used for normalization is that P&Y is the most commonly used method for interface reconstruction in PLIC-VOF codes and can be considered as standard practice. We see that the computational efforts utilized by MYC and LSG are comparable to that by P&Y, while ELVIRA is the most expensive one. This result was expected since MYC and LSG, unlike many truly second-order counterparts, are non-iterative, and their designs include a moderate amount of mathematical operations using information in a compact $3 \times 3 \times 3$ cell cluster. Although ELVIRA is also non-iterative and employs the same stencil size, its design involves a relatively large amount of mathematical operations, which makes this scheme more time consuming than the others.

4.4.4. DEFORMATION FIELD - 3D

The 3D deformation problem was proposed in [92] as a truly challenging test, and later used in, e.g., [11], [41], [48], [60], and [96]. Following the problem setup described in [48], a sphere of radius 0.15 and center (0.35, 0.35, 0.35) is immersed in a 3D reversible deformation field inside a unit sized cube. The flow is formed by superposing deformation

on the $x - y$ plane with deformation on the $x - z$ plane. The velocity field is

$$\begin{aligned} u &= 2\sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) \cos(\pi t/T), \\ v &= -\sin(2\pi x) \sin^2(\pi y) \sin(2\pi z) \cos(\pi t/T), \\ w &= -\sin(2\pi x) \sin(2\pi y) \sin^2(\pi z) \cos(\pi t/T), \end{aligned} \quad (4.51)$$

where $T = 3$ is used. Due to this velocity field, the sphere undergoes severe and truly 3D deformation until it reaches maximum stretching at $t = 1.5$, then returns to its initial shape and position at $t = 3$ by means of the Leveque cosine term. The velocity field (4.51) is applied in a discretely divergence-free manner.

Figure 4.10 shows the results on a $128 \times 128 \times 128$ grid for CFL=0.5. From maximum deformation at $t = 1.5$ to the full reversal at $t = 3$, several snapshots are taken to display how the geometry evolves during the second half of the test. As expected, the H&N + LHF method has the worst performance, and superiority of the PLIC-VOF modules is evident. Even with the PLIC-VOF modules, holes in the thin, deformed fluid sheets are observed. The recovered shape in the end is roughly a sphere with some minor coalescence when the PLIC-VOF modules are used.

Mesh	LSG		ELVIRA		CVTNA	H&N+LHF
	MACHO	COSMIC	MACHO	COSMIC	PCFSC	
32^3	7.52×10^{-3} <i>1.67</i>	7.51×10^{-3} <i>1.68</i>	7.39×10^{-3} <i>1.68</i>	7.40×10^{-3} <i>1.72</i>	7.41×10^{-3} <i>1.90</i>	1.06×10^{-2} <i>1.05</i>
64^3	2.37×10^{-3} <i>1.81</i>	2.34×10^{-3} <i>1.82</i>	2.30×10^{-3} <i>1.82</i>	2.25×10^{-3} <i>1.79</i>	1.99×10^{-3} <i>2.69</i>	5.10×10^{-3} <i>1.08</i>
128^3	6.75×10^{-4}	6.64×10^{-4}	6.53×10^{-4}	6.50×10^{-4}	3.09×10^{-4}	2.40×10^{-3}

Table 4.5: Error (4.45) for the 3D deformation field test. CFL is equal to 0.5. The column with the header CVTNA/PCFSC shows data taken from [48] for comparison. Rate of convergence (4.47) is written in italics between mesh entries.

Table 4.5 shows the geometrical error (4.45). The data in the column with the header CVTNA/PCFSC are taken from [48] as a benchmark result. The H&N + LHF method shows a first-order behavior, while the results from the CVTNA/PCFSC combination tend toward second-order accuracy. The other modules have a rate of convergence between first and second-order. For the same reconstruction method, the performance of the advection schemes MACHO and COSMIC is similar. With MACHO, the sequence of advection in each direction is changed every time step in order to minimize asymmetries caused by directional splitting. It appears that this procedure makes MACHO quite comparable to the inherently-symmetric COSMIC. It should also be noted that computational effort due to the use of MACHO compared to COSMIC is substantially small. Considering the reconstruction schemes, ELVIRA shows a slightly better performance than LSG in terms of magnitude of error, though both schemes have similar order of accuracy.

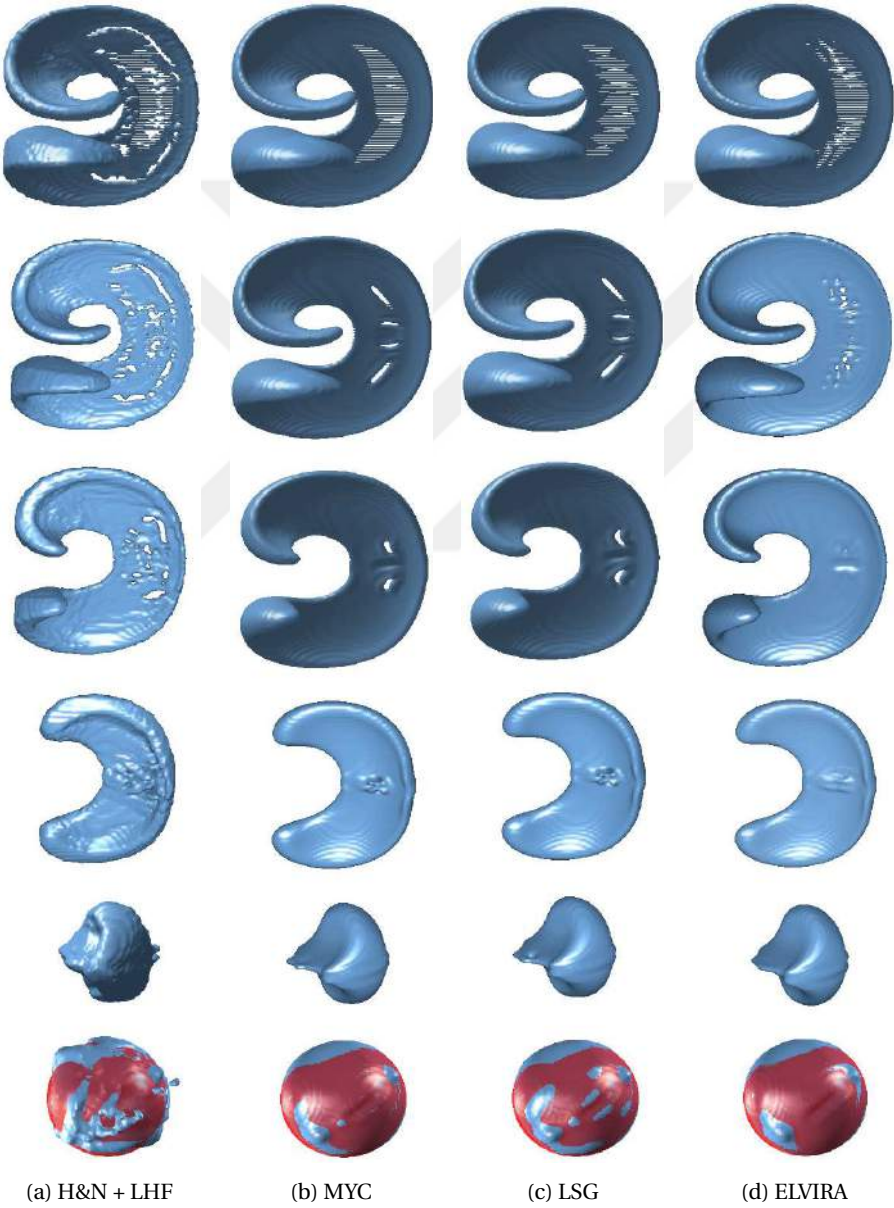


Figure 4.10: 3D reversible deformation field test. Snapshots are taken at several instances from $t = 1.5$ (plots at the top) to $t = 3$ (plots at the bottom). Results are obtained on a 128^3 uniform grid at $CFL = 0.5$. The PLIC methods are combined with the MACHO advection scheme. Geometries in blue color indicate numerical results, and spheres in red color in the plots at the bottom indicate the exact solution.

4.4.5. SHEARING FLOW - 3D

In the 3D shearing flow test, introduced by [Liovic et al. \[48\]](#) and later used by [López et al. \[45\]](#), a sphere of radius 0.15 and center (0.5, 0.75, 0.25) in a domain of $1 \times 1 \times 2$ is immersed in a velocity field which is a combination of the single vortex in the $x - y$ plane prescribed by Eq. (4.49) with laminar pipe flow in the z -direction. The velocity field is stated by

$$\begin{aligned} u &= \sin(2\pi y) \sin^2(\pi x) \cos\left(\frac{\pi t}{T}\right), \\ v &= -\sin(2\pi x) \sin^2(\pi y) \cos\left(\frac{\pi t}{T}\right), \\ w &= U_{\max} \left(1 - \frac{r}{R}\right)^2 \cos\left(\frac{\pi t}{T}\right) \end{aligned} \quad (4.52)$$

where $U_{\max} = 1.0$, $r = \sqrt{(x - x_0)^2 + (y - y_0)^2}$, $R = 0.5$, $x_0 = 0.5$ and $y_0 = 0.5$. Figure 4.11 shows the results which are obtained on a $64 \times 64 \times 128$ uniform grid with a CFL number equal to 1. The MYC, LSG, and ELVIRA schemes are used in combination with the MACHO advection scheme. In the plots, geometries in blue color illustrate results from the numerical methods, and spheres in red color at the bottom of the plots illustrate the initial geometry. Only the second half of the test is shown in several snapshots, from maximum deformation at $t = 3$ to full reversal of the flow at $t = 6$.

Analogous to the previous tests, H&N + LHF performs the worst amongst the VOF modules. With the other methods, we observe holes in the thin, deformed fluid sheets around the instant $t = 3$. Also, at the tail of the body, we observe spurious fragmentation. Comparing the PLIC schemes, ELVIRA shows the best qualitative performance around the instant of maximum deformation. We notice significant reduction in fragmentation at the tail, and less holes in the thin fluid sheets. However, the results from the PLIC-MACHO modules at the end of the test are almost indistinguishable.

Mesh	LSG		ELVIRA		CLC-CBIR	H&N+LHF
	MACHO	COSMIC	MACHO	COSMIC	FMFPA-3D	
32 × 32 × 64	1.20×10^{-2} <i>1.51</i>	1.17×10^{-2} <i>1.51</i>	1.16×10^{-2} <i>1.53</i>	1.17×10^{-2} <i>1.52</i>	9.95×10^{-3} <i>1.61</i>	6.20×10^{-2} <i>1.07</i>
64 × 64 × 128	4.21×10^{-3} <i>1.45</i>	4.10×10^{-3} <i>1.44</i>	4.01×10^{-3} <i>1.51</i>	3.97×10^{-3} <i>1.49</i>	3.27×10^{-3} <i>1.82</i>	2.95×10^{-2} <i>1.05</i>
128 × 128 × 256	1.54×10^{-3}	1.51×10^{-3}	1.41×10^{-3}	1.40×10^{-3}	9.27×10^{-4}	1.42×10^{-2}

Table 4.6: Error (4.45) for the 3D shearing flow test (single vortex in the $x - y$ plane with laminar pipe flow in the z -direction). CFL is equal to 1.0. The column with the header CLC-CBIR/FMFPA-3D shows data taken from [45] for comparison. Rate of convergence (4.47) is written in italics between mesh entries.

Table 4.6 shows the geometrical error (4.45). The data in the column with the header CLC-CBIR/FMFPA-3D are taken from [45] for comparison. This study includes the im-

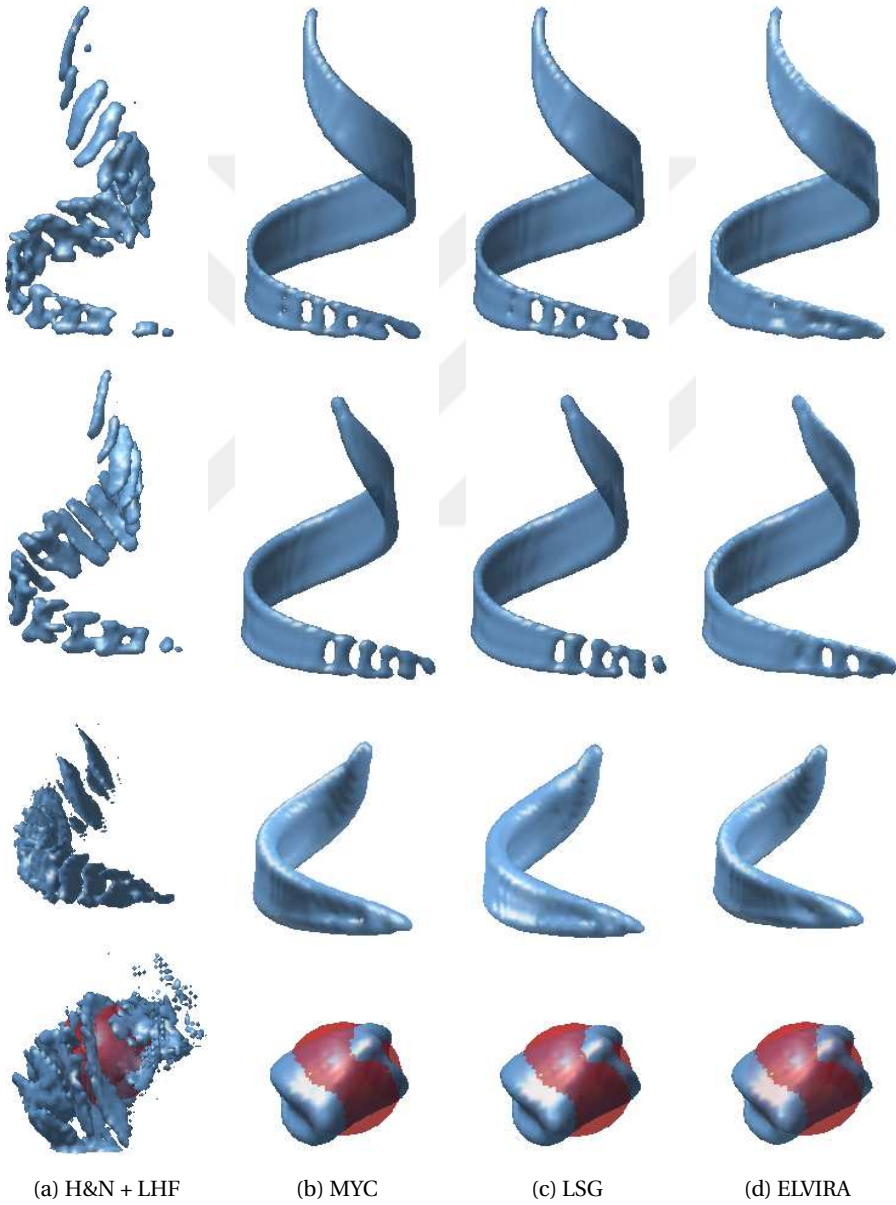


Figure 4.11: 3D reversible shearing flow test: a combination of single vortex flow and laminar pipe flow. Only the second half of the test is depicted through several snapshots taken at several instances from $t = 3$ (plots at the top) to $t = 6$ (plots at the bottom). Results are obtained on a $64 \times 64 \times 128$ uniform grid at $CFL = 1.0$. The PLIC methods are combined with the MACHO advection scheme. Geometries in blue color indicate numerical results, and spheres in red color in the plots at the bottom indicate the exact solution.

plementation of the coupled Conservative Level-Contour and Cubic-Bezier-based Interface Reconstruction (CLC-CBIR), and the Face-Matched Flux Polyhedra (FMFPA-3D) for unsplit advection in 3D. The first-order accuracy of the H&N + LHF method is obvious. Although the CLC-CBIR/FMFPA-3D combination of [45] outperforms the other PLIC-VOF modules, there is not a considerable difference in terms of both magnitude of geometrical error and rate of convergence. Analogous with the previous test, the performance of the advection schemes MACHO and COSMIC is similar for the same reconstruction method. Considering the reconstruction schemes, ELVIRA shows a slightly better performance than LSG in terms of magnitude of error, though both schemes have similar order of accuracy.

4.4.6. APPLICATION EXAMPLE: PROPAGATING RIENECKER-FENTON WAVES

In the final test, we will assess the performance of the new VOF modules consisting of several reconstruction and advection schemes in a practical application. For this purpose, the VOF modules are incorporated into the CFD flow solver ComFLOW. We will generate Rienecker-Fenton waves [97] in shallow water in 2D, and monitor wave propagation throughout the domain. We particularly focus our attention on investigating the effect of various VOF modules on energy dissipation in these simulations. Two Rienecker-Fenton waves with the same period but different heights are generated; see Table 4.7. WAVE2 has a steepness of 2.0% and WAVE10 has a steepness of 10.8% which indicates strong nonlinear behavior in the flow. Both waves are started from rest, and within the first three periods, wave heights are gradually increased until full heights are reached.

Waves	<i>period</i> T(s)	<i>height</i> H(m)	<i>length</i> L(m)	<i>steepness</i> H/L(%)
WAVE2	4	0.5	24.7	2.0
WAVE10	4	3.0	27.6	10.8

Table 4.7: Characteristics of the Rienecker-Fenton waves.

The length of the domain in the direction of propagation is defined in such a way that both waves do not reach the end of the domain during the simulations. This procedure guarantees that there is no reflection in the computational domain, and hence the solution is not perturbed. The duration of the simulations should allow us to have a comprehensive picture regarding wave damping. Therefore, a stable wave system for a large number of wave periods is required. In this analysis, we performed simulations for 200 seconds to have a stable wave system for at least 18 consecutive wave lengths, and correspondingly a domain length of 2000 meters is considered sufficient taking the fastest propagating wave component into account. The water depth in all simulations is 10 meters. Three uniform grid resolutions are considered for the grid convergence study: $1m \times 1m$, $0.5m \times 0.5m$ and $0.25m \times 0.25m$.

Figures 4.12 and 4.13 show wave elevation as a function of the horizontal position at time $t = 200s$ on three uniform grid resolutions for WAVE2 and WAVE10, respectively. The first 500m of the full computational domain is plotted, since this part provides am-

ple insight concerning the dissipation property of the VOF schemes. The results are obtained by using three PLIC algorithms with the MACHO advection scheme; the Mixed Youngs-centered (MYC) scheme [49], the least squares gradient (LSG) technique [44] and the efficient least squares VOF interface reconstruction algorithm (ELVIRA) [46]. Also, the analytical results from the Rienecker-Fenton theory and the Hirt-Nichols' VOF with local height function (H&N + LHF) are plotted in the figures.

To analyze the results quantitatively, we will use two error definitions. The first error is the relative difference in wave height ε_H between the numerical results and the prescribed value for each wave given in Table 4.7. The second error is the relative shift ε_L between the numerical and analytical positions in the x -direction at which waves reach highest elevations. The expressions for ε_H and ε_L are given by

$$\varepsilon_H = \frac{|H_n - H|}{H} \times 100, \quad \varepsilon_L = \frac{|x_{\eta_n} - x_{\eta_a}|}{L} \times 100. \quad (4.53)$$

Table 4.8 shows maximum ε_H and ε_L values in Figs. 4.12 and 4.13. The rate of error reduction as the grid is refined is obtained using (4.47) and written in italics and green color between mesh entries.

Waves	Resolution	MYC		ELVIRA		LSG		H&N+LHF	
		ε_H	ε_L	ε_H	ε_L	ε_H	ε_L	ε_H	ε_L
WAVE2	$1m \times 1m$	30.5	12.9	27.2	12.9	29.5	12.7	48.9	8.9
		<i>3.12</i>	<i>0.70</i>	<i>3.04</i>	<i>0.72</i>	<i>2.47</i>	<i>0.64</i>	<i>1.78</i>	<i>0.36</i>
	$0.5m \times 0.5m$	3.5	7.9	3.3	7.8	5.3	8.1	14.2	6.9
		<i>0.37</i>	<i>3.30</i>	<i>0.40</i>	<i>3.28</i>	<i>0.72</i>	<i>3.33</i>	<i>0.94</i>	<i>2.52</i>
	$0.25m \times 0.25m$	2.7	0.8	2.5	0.8	3.2	0.8	7.4	1.2
WAVE10	$1m \times 1m$	66.5	35.1	62.3	31.4	65.1	35.2	80.3	49.6
		<i>1.12</i>	<i>0.12</i>	<i>1.33</i>	<i>0.08</i>	<i>1.19</i>	<i>0.13</i>	<i>0.78</i>	<i>0.57</i>
	$0.5m \times 0.5m$	30.5	32.2	24.7	29.7	28.4	32.1	46.7	33.3
		<i>1.65</i>	<i>2.08</i>	<i>1.59</i>	<i>2.28</i>	<i>1.64</i>	<i>2.30</i>	<i>0.73</i>	<i>1.87</i>
	$0.25m \times 0.25m$	9.7	7.6	8.2	6.1	9.1	6.5	28.1	9.1

Table 4.8: ε_H (%) and ε_L (%) values computed by (4.53). Rate of convergence (4.47) is written in italics and green color between mesh entries. Results are shown for WAVE2 and WAVE10 using four VOF schemes on three uniform grid resolutions of $1m \times 1m$, $0.5m \times 0.5m$ and $0.25m \times 0.25m$. The MYC, ELVIRA and LSG reconstruction methods are combined with the MACHO advection scheme.

For both WAVE2 and WAVE10 the results show that the resolution of $1m \times 1m$ is very insufficient, which is expected since this resolution corresponds to, for example, only 3 cells per wave wave height and 27 cells per wave length for WAVE10. On the finest grid with the resolution of $0.25m \times 0.25m$ all the VOF schemes show better performances.

If we look at the ε_L values in Table 4.8, we see that the performance of H&N + LHF for both WAVE2 and WAVE10 is comparable to those of the PLIC-VOF schemes. For WAVE2 the ε_L value from H&N + LHF on the coarsest grid is 8.9% while the smallest ε_L value from the PLIC schemes is 12.7% produced by LSG. For the same wave on the finest grid

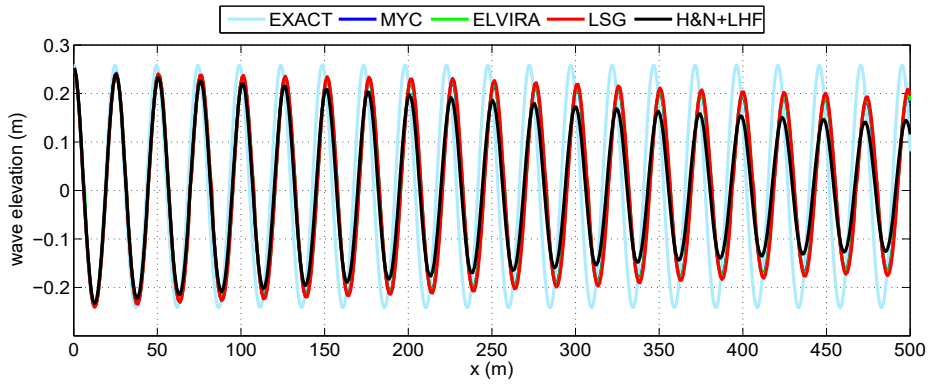
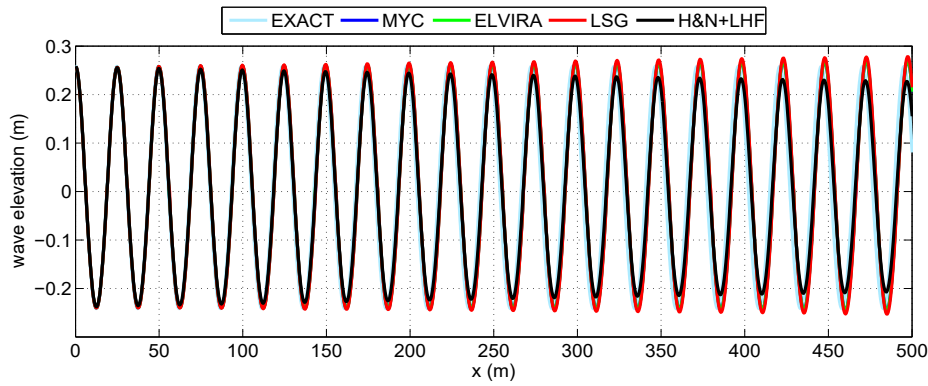
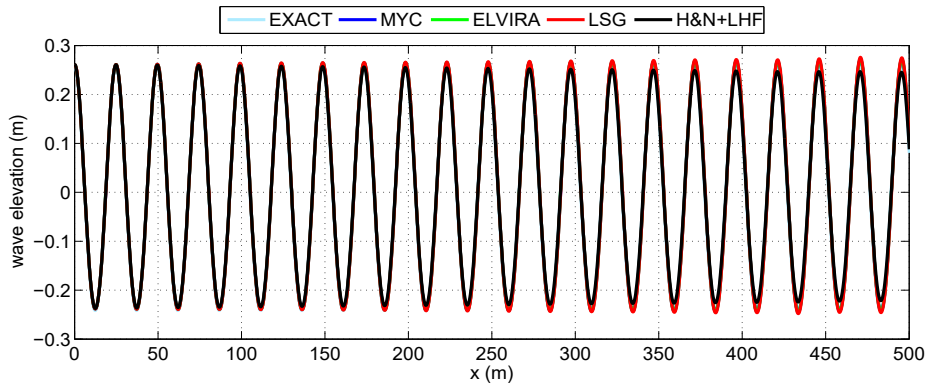
(a) Grid resolution is $\Delta x = \Delta z = 1m$ (b) Grid resolution is $\Delta x = \Delta z = 0.5m$ (c) Grid resolution is $\Delta x = \Delta z = 0.25m$

Figure 4.12: Wave elevations as a function of horizontal location at time $t = 200s$ for WAVE2. The MYC, ELVIRA and LSG reconstruction methods are combined with the MACHO advection scheme.

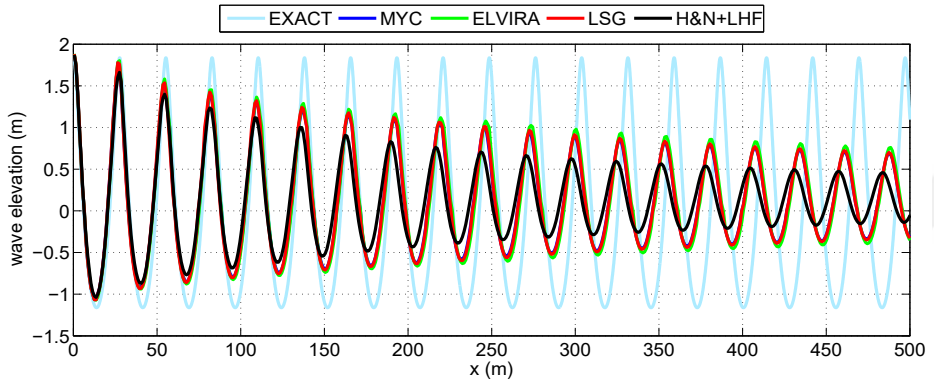
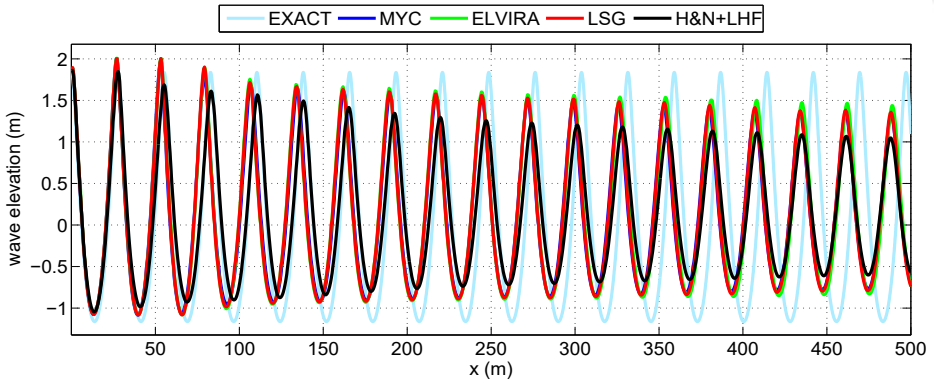
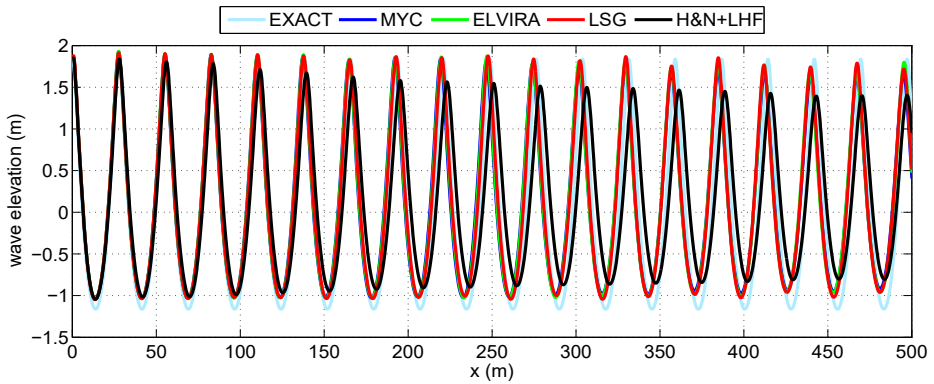
(a) Grid resolution is $\Delta x = \Delta z = 1m$ (b) Grid resolution is $\Delta x = \Delta z = 0.5m$ (c) Grid resolution is $\Delta x = \Delta z = 0.25m$

Figure 4.13: Wave elevations as a function of horizontal location at time $t = 200s$ for WAVE10. The MYC, ELVIRA and LSG reconstruction methods are combined with the MACHO advection scheme.

H&N + LHF produces an ε_L value of 1.2% while the value from PLIC schemes is 0.8%. Although H&N + LHF initially starts with the smallest ε_L value on the coarsest grid, its rate of convergence is somewhat poorer compared to PLIC methods, and it ends up with a larger but comparable error on the finest grid. For WAVE10 which is more challenging and more non-linear, H&N + LHF produces larger ε_L values on all the resolutions. On the finest grid the 9.1% error from H&N + LHF is larger than, but still comparable to, the 6.1% error from ELVIRA.

The values of ε_H in Table 4.8 depict, however, a different picture. On all the resolution levels the PLIC schemes demonstrate a clearly superior performance. For WAVE2 ELVIRA produces the smallest ε_H value of 27.2% on the coarsest grid while H&N + LHF produces 48.9%. On the finest grid this value reduces to 2.5% with ELVIRA and 7.4% with H&N + LHF. For WAVE10 ELVIRA is again the best performing method. On the resolution of $1m \times 1m$ the ε_H value from ELVIRA is 62.3% while that from H&N + LHF is 80.3%. As the grid is refined in two levels, the error from ELVIRA decreases at a rate of 1.33 and 1.59, while the error from H&N + LHF decreases at a rate of 0.78 and 0.73. On the finest grid the ε_H value from ELVIRA is only 8.2% while that from H&N + LHF is 28.1%.

For WAVE2 the wave signals obtained with the three PLIC methods are almost the same for all the three grid resolutions. For WAVE10 PLIC methods perform again very similarly. Both errors indicate that ELVIRA performs better compared to others but only very slightly.

Figure 4.14 illustrates the results to compare the performance of the advection methods. Here wave elevation as a function of the horizontal position is plotted at time $t = 200s$ on three uniform grid resolutions for WAVE10. The MACHO, EI-LE and COSMIC advection methods are used in combination with the ELVIRA reconstruction scheme. The results demonstrate that all the three advection methods resulted in almost the same profiles at the three grid resolutions. When these advection methods were combined with other PLIC reconstruction schemes, we again observed similar performances from the advection schemes but preferred not to put those results here for the sake of brevity. This observation is valid for WAVE2 as well.

4.5. CONCLUDING REMARKS

Several PLIC-VOF methods have been implemented in the CFD tool ComFLOW, and their performances have been demonstrated in a number of test cases. In these tests the PLIC-VOF methods have also been compared to the original VOF implementation in COMFLOW, the H&N + LHF method. The test cases included a set of standard advection problems and an application example with Rienecker-Fenton waves. In the advection tests velocity fields are defined in such a way that advected fluid bodies return to their original shapes and locations at the end of the simulation. Therefore, the volume fraction distribution at the end of the simulation is expected to be equal to the original volume fraction distribution, which can be considered as the exact solution. The difference between these two distributions gives an estimation of accuracy of a VOF method. In the application example with Rienecker-Fenton waves, spurious energy dissipation in the computational domain was monitored, which manifests itself as loss of wave height accompanied by phase shift. The results demonstrated that PLIC-VOF methods outperform the H&N + LHF method by a clear margin without increasing computational costs

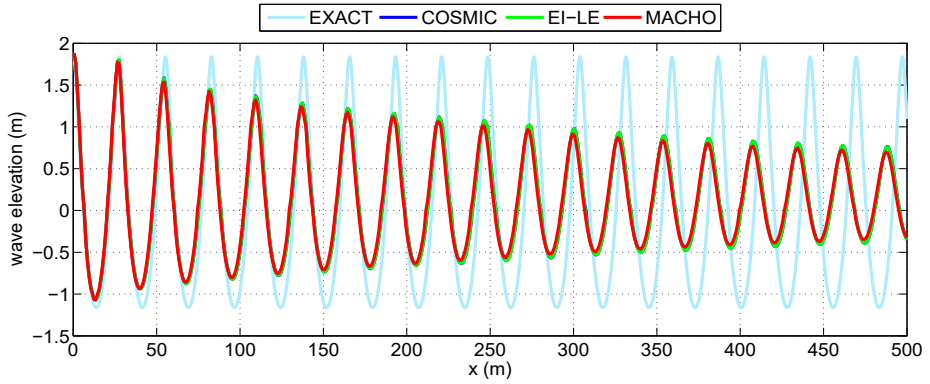
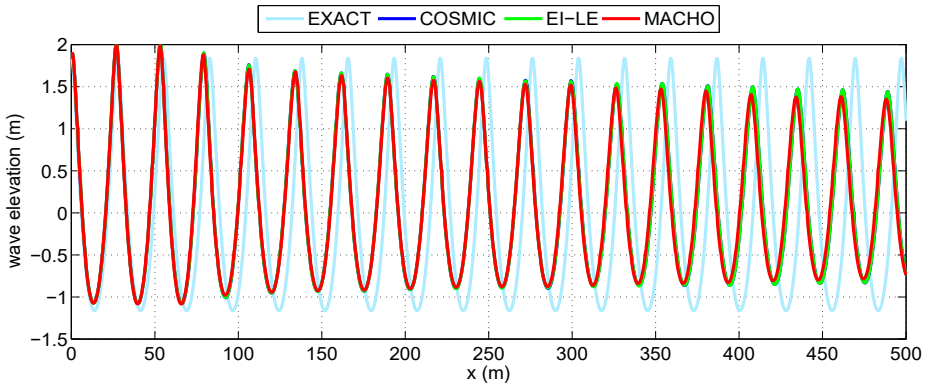
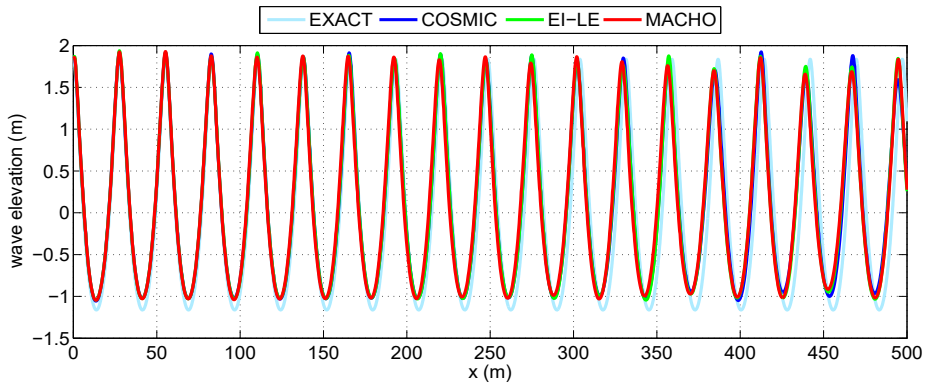
(a) Grid resolution is $\Delta x = \Delta z = 1m$ (b) Grid resolution is $\Delta x = \Delta z = 0.5m$ (c) Grid resolution is $\Delta x = \Delta z = 0.25m$

Figure 4.14: Wave elevations as a function of horizontal location for WAVE10. The COSMIC, EI-LE and MACHO advection schemes are combined with the ELVIRA reconstruction method.

in a disproportionate manner. In the advection tests the PLIC-VOF methods showed a superior performance over the H&N + LHF method both qualitatively and quantitatively, and in simulations of Rienecker-Fenton waves a significant reduction in errors in terms of loss of wave height as well as phase shift was observed. It is possible to conclude that although the H&N + LHF method may be adequate to model the free surface of low-steepness water waves, it has a limited ability to model the free surface of more nonlinear waves which are actually interesting in practical applications.

The PLIC-VOF methods used in this work do not involve highly complex mathematical procedures with a large number of algorithmic tasks. Therefore, difficulty in implementing these methods was modest. Although direction splitting advection is easier to implement than unsplit advection especially in 3D, it requires at least three interface reconstruction sweeps to complete a volume tracking update. To overcome this increase in computational cost, we opted to implement reconstruction techniques with slightly lower accuracy compared to some of the contemporary PLIC schemes, but are considerably cheaper as well as easier to implement. In the end this strategy allowed us to achieve a satisfying level of accuracy in an economical manner.

REFERENCES

- [1] G. Tryggvason, R. Scardovelli, and S. Zaleski, *Direct numerical simulations of gas-liquid multiphase flows* (Cambridge University Press, 2011).
- [2] G. Yeoh and J. Tu, *Computational techniques for multiphase flows* (Butterworth-Heinemann, 2009).
- [3] F. Harlow and J. Welch, *Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface*, Phys. Fluids **8**, 21822189 (1965).
- [4] B. Daly, *A technique for including surface tension effects in hydrodynamic calculations*, J. Comput. Phys. **4**, 97117 (1969).
- [5] J. Glimm, O. Bryan, R. Menikoff, and D. Sharp, *Front tracking applied to Rayleigh-Taylor instability*, SIAM J. Sci. Stat. Comput. **7**, 230251 (1986).
- [6] S. Chen, D. Johnson, and P. Raad, *The surface marker method*, (in: L.C. Wrobel and C.A. Brebbia (Eds.), Computational Modelling of Free and Moving Boundary Problems, vol. 1, Fluid flow, Walter de Gruyter & Co, 1991) pp. 223–234.
- [7] S. Unverdi and G. Tryggvason, *A front tracking method for viscous incompressible flow*, J. Comput. Phys. **100**, 2537 (1992).
- [8] S. Osher and J. Sethian, *Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations*, J. Comput. Phys. **79**, 1249 (1988).
- [9] M. Sussman, E. Fatemi, P. Smereka, and S. Osher, *An improved level set method for incompressible two-phase flows*, Comput. Fluids **27**, 663680 (1998).
- [10] M. Sussman and E. Fatemi, *An efficient, interface preserving level set re-distancing algorithm and its application to interfacial incompressible fluid flow*, SIAM J. Sci. Comput. **20**, 1165 (1999).

- [11] D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell, *A hybrid particle level set method for improved interface capturing*, J. Comput. Phys. **183**, 83 (2002).
- [12] A. Iafrati, A. D. Mascio, and E. Campana, *A level set technique applied to unsteady free surface flows*, Int. J. Num. Meth. Fluid **35**, 281 (2001).
- [13] T. Yabe, F. Xiao, and T. Utsumi, *The constrained interpolation profile (CIP) method for multi-phase analysis*, J. Comput. Phys. **169**, 556 (2001).
- [14] H. Takewaki, A. Nishiguchi, and T. Yabe, *Cubic interpolated pseudoparticle method (CIP) for solving hyperbolic-type equations*, J. Comput. Phys. **61**, 261 (1985).
- [15] H. Takewaki and T. Yabe, *The cubic-interpolated pseudo particle (CIP) method: application to nonlinear and multi-dimensional hyperbolic equations*, J. Comput. Phys. **70**, 355 (1987).
- [16] D. Jacqmin, *Calculation of two-phase Navier-Stokes flows using phasefield modeling*, J. Comput. Phys. **155**, 96 (1999).
- [17] H. Ding, P. Spelt, and C. Shu, *Diffuse interface model for incompressible two-phase flows with large density ratios*, J. Comput. Phys. **226**, 2078 (2007).
- [18] P. Woodward and P. Colella, *The numerical solution of two-dimensional fluid flow with strong shocks*, J. Comput. Phys. **54**, 115 (1984).
- [19] D. Torres and J. Brackbill, *The Point-Set method: front-tracking without connectivity*, J. Comput. Phys. **165**, 620 (2000).
- [20] C. Hirt, J. Cook, and T. Butler, *A Lagrangian method for calculating the dynamics of an incompressible fluid with a free surface*, J. Comput. Phys. **5**, 103 (1970).
- [21] M. Fritts and J. Boris, *The Lagrangian solution of transient problems in hydrodynamics using a triangular mesh*, J. Comput. Phys. **31**, 173 (1979).
- [22] V. Dyadechko and M. Shashkov, *Moment-of-fluid interface reconstruction*, Tech. Rep. LA-UR-05-7571, Los Alamos National Laboratory (2005).
- [23] C. Hirt and B. Nichols, *Volume of fluid (VOF) method for the dynamics of free boundaries*, J. Comput. Phys. **39**, 201 (1981).
- [24] J. Ramshaw and J. Trapp, *A numerical technique for low-speed homogeneous two-phase flow with sharp interfaces*, J. Comput. Phys. **21**, 438 (1976).
- [25] N. Ashgriz and J. Poo, *FLAIR: Flux line-segment model for advection and interface reconstruction*, J. Comput. Phys. **93**, 449 (1991).
- [26] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti, *Modelling merging and fragmentation in multiphase flows with SURFER*, J. Comput. Phys. **113**, 134 (1994).

- [27] O. Ubbink, *Numerical prediction of two fluid systems with sharp interfaces*, Ph.D. thesis, University of London (1997).
- [28] M. Darwish, *A new high-resolution scheme based on the normalized variable formulation*, Numerical Heat Transfer, Part B **24**, 353 (1993).
- [29] S. Zalesak, *Fully multidimensional flux-corrected transport algorithms for fluids*, J. Comput. Phys. **31**, 335 (1979).
- [30] B. Leonard, *The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection*, Comp. Meth. in Appl. Mech. and Eng. **88**, 17 (1991).
- [31] S. Davis, *Flux difference splittings and limiters for the resolution of contact discontinuities*, Appl. Math. Comput. **65**, 3 (1994).
- [32] M. Rudman, *Volume-tracking methods for interfacial flow calculations*, Int. J. Numer. Meth. Fluids **24**, 671 (1997).
- [33] H. Jasak, H. Weller, and A. Gosman, *High resolution NVD differencing scheme for arbitrarily unstructured meshes*, Int. J. Numer. Meth. Fluids **31**, 431 (1999).
- [34] G. Price, *A piecewise parabolic volume tracking method for the numerical simulation of interfacial flows*, Ph.D. thesis, The University of Calgary (2000).
- [35] W. Noh and P. Woodward, *SLIC (Simple Line Interface Calculation)*, (in: A.I. van der Vooren and P.J. Zandbergen (Eds.), Lecture Notes in Physics, vol. 59, Springer-Verlag, New York, 1976) pp. 330–340.
- [36] A. Chorin, *Flame advection and propagation algorithms*, J. Comput. Phys. **35**, 1 (1980).
- [37] D. Youngs, *Time-dependent multi-material flow with large fluid distortion*, (in: K.W. Morton and M.J. Baines (Eds.), Numerical Methods for Fluid Dynamics, Academic Press, New York, 1982) pp. 273–285.
- [38] P. Lötstedt, *A front tracking method applied to burger's equation and two-phase porous flow*, J. Comput. Phys. **47**, 211 (1982).
- [39] M. Sussman and E. Puckett, *A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows*, J. Comput. Phys. **162**, 301 (2000).
- [40] T. Ménard, S. Tanguy, and A. Berlemont, *Coupling level set/VOF/ghost fluid methods: Validation and application to 3D simulation of the primary break-up of a liquid jet*, Int. J. Multiphas. Flow **33**, 510 (2007).
- [41] M. Jemison, E. Loch, M. Sussman, M. Shashkov, M. Arienti, M. Ohta, and Y. Wang, *A coupled level set-moment of fluid method for incompressible two-phase flows*, J. Sci. Comput. **54**, 454 (2013).

- [42] B. Parker and D. Youngs, *Two and three dimensional Eulerian simulation of fluid flow with material interfaces*, Technical Report 01/92, UK Atomic Weapons Establishment (1992).
- [43] T. Barth and D. Jespersen, *The design and application of upwind schemes on unstructured meshes*, in *Proceedings of the 27th AIAA Aerospace Sciences Meeting and Exhibit*, 9-12 January 1989, Reno (USA) (1989).
- [44] W. Rider and D. Kothe, *Reconstructing volume tracking*, J. Comput. Phys. **141**, 112 (1998).
- [45] J. López, C. Zanzi, P. Gómez, F. Faura, and J. Hernández, *A new volume of fluid method in three dimensions-Part 2: Piecewise-planar interface reconstruction with cubic-Bézier fit*, Int. J. Numer. Meth. Fluids **58**, 923 (2008).
- [46] J. Pilliod and E. Puckett, *Second-order accurate volume-of-fluid algorithms for tracking material interfaces*, J. Comput. Phys. **199**, 465 (2004).
- [47] B. Swartz, *The second-order sharpening of blurred smooth borders*, Math. Comput. **52**, 675 (1989).
- [48] P. Liovic, M. Rudman, J.-L. Liow, D. Lakehal, and D. Kothe, *A 3D unsplit-advection volume tracking algorithm with planarity-preserving interface reconstruction*, Comput. Fluids **35**, 1011 (2006).
- [49] E. Aulisa, S. Manservigi, R. Scardovelli, and S. Zaleski, *Interface reconstruction with least-squares fit and split advection in three-dimensional Cartesian geometry*, J. Comput. Phys. **225**, 2301 (2007).
- [50] S. Mosso, C. Garasi, and R. Drake, *A smoothed two- and three-dimensional interface reconstruction method*, Comput. Visual Sci. **12**, 365 (2009).
- [51] D. Benson, *Volume of fluid interface reconstruction methods for multi-material problems*, Appl. Mech. Rev. **55**, 151 (2002).
- [52] T. Marić, H. Marschall, and D. Bothe, *vofoam – A geometrical Volume-of-Fluid algorithm on arbitrary unstructured meshes with local dynamic adaptive mesh refinement using OpenFOAM*, arXiv preprint arXiv:1305.3417, available online (<http://arxiv.org/abs/1305.3417>) (2013).
- [53] A. Cervone, S. Manservigi, and R. Scardovelli, *An optimal constrained approach for divergence-free velocity interpolation and multilevel VOF method*, Comput. Fluids **47**, 101 (2011).
- [54] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, and S. Zaleski, *Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimensional flows*, J. Comput. Phys. **152**, 423 (1999).
- [55] R. Scardovelli and S. Zaleski, *Direct numerical simulation of free-surface and interfacial flow*, Annu. Rev. Fluid Mech. **31**, 567 (1999).

- [56] R. Scardovelli and S. Zaleski, *Interface reconstruction with least-square fit and split Eulerian-Lagrangian advection*, Int. J. Numer. Meth. Fluids **41**, 251 (2003).
- [57] D. Youngs, *An interface tracking method for a 3D Eulerian hydrodynamics code*, Technical Report AWRE/44/92/35, Atomic Weapons Research Establishment (1987).
- [58] T. Barth, *Aspects of unstructured grids and finite-volume solvers for Euler and Navier-Stokes equations*, AGARD Publication R-787, VKI/NASA/AGARD Special Course on Unstructured Grid Methods for Advection Dominated Flows (1995).
- [59] S. Popinet, *An accurate adaptive solver for surface-tension-driven interfacial flows*, J. Comput. Phys. **228**, 5838 (2009).
- [60] Z. Wang, J. Yang, and F. Stern, *A new volume-of-fluid method with a constructed distance function on general structured grids*, J. Comput. Phys. **231**, 3703 (2012).
- [61] G. Miller and P. Colella, *A conservative three-dimensional Eulerian method for coupled solid-fluid shock capturing*, J. Comput. Phys. **183**, 26 (2002).
- [62] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical recipes in Fortran 90* (Cambridge University Press, 1996).
- [63] H. Ahn and M. Shashkov, *Multi-material interface reconstruction on generalized polyhedral meshes*, Tech. Rep. LA-UR-07-0656, Los Alamos National Laboratory (2007).
- [64] R. Scardovelli and S. Zaleski, *Analytical relations connecting linear interfaces and volume fractions in rectangular grids*, J. Comput. Phys. **164**, 228 (2000).
- [65] J. López and J. Hernández, *Analytical and geometrical tools for 3D volume of fluid methods in general grids*, J. Comput. Phys. **227**, 5939 (2008).
- [66] B. Leonard, A. Lock, and M. MacVean, *Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes*, Mon. Weather Rev. **124**, 2588 (1996).
- [67] G. Strang, *On the construction and comparison of difference schemes*, SIAM J. Numer. Anal. **5**, 506 (1968).
- [68] S. van der Pijl, A. Segal, C. Vuik, and P. Wesseling, *Computing three-dimensional two-phase flows with a mass-conserving level set method*, Comput. Visual. Sci. **11**, 221 (2008).
- [69] D. Lörstäd and L. Fuchs, *High-order surface tension VOF-model for 3D bubble flows with high density ratio*, J. Comput. Phys. **200**, 153 (2004).
- [70] S. van der Pijl, A. Segal, C. Vuik, and P. Wesseling, *A mass-conserving Level-Set method for modelling of multi-phase flows*, Int. J. Numer. Meth. Fluids **47**, 339 (2005).

- [71] S.-J. Lin and R. Rood, *Multidimensional flux-form semi-Lagrangian transport schemes*, Mon. Weather Rev. **124**, 2046 (1996).
- [72] G. Weymouth, D. Dommermuth, K. Hendrickson, and D.-P. Yue, *Advances in Cartesian-grid methods for computational ship hydrodynamics*, in *26th Symposium on Naval Hydrodynamics, 17-22 September 2006, Rome (Italy)* (2006).
- [73] G. Weymouth and D.-P. Yue, *Conservative Volume-of-Fluid method for free-surface simulations on Cartesian-grids*, J. Comput. Phys. **229**, 2853 (2010).
- [74] S. Mosso, B. Swartz, D. Kothe, and S. Clancy, *Recent enhancements of volume tracking algorithms for irregular grids*, Technical Report LA-CP-96-227, Los Alamos National Laboratory (1996).
- [75] K. Shahbazi, M. Paraschivoiu, and J. Mostaghimi, *Second order accurate volume tracking based on remapping for triangular meshes*, J. Comput. Phys. **188**, 100 (2003).
- [76] J. López, J. Hernández, P. Gómez, and E. Faura, *A volume of fluid method based on multidimensional advection and spline interface reconstruction*, J. Comput. Phys. **195**, 718 (2004).
- [77] S. Garrioch and B. Baliga, *A PLIC volume tracking method for the simulation of two-fluid flows*, Int. J. Numer. Meth. Fluids **52**, 1093 (2006).
- [78] P. Liovic and D. Lakehal, *A Newton-Krylov solver for remapping-based volume-of-fluid methods*, SIAM J. Sci. Comput. **31**, 865 (2008).
- [79] E. Puckett, A. Almgren, J. Bell, D. Marcus, and W. Rider, *A high-order projection method for tracking fluid interfaces in variable density incompressible flows*, J. Comput. Phys. **130**, 269 (1997).
- [80] M. Sussman, *A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles*, J. Comput. Phys. **187**, 110 (2003).
- [81] M. Sussman, K. Smith, M. Hussaini, M. Ohta, and R. Zhi-Wei, *A sharp interface method for incompressible two-phase flows*, J. Comput. Phys. **221**, 469 (2007).
- [82] D. Kothe, W. Rider, S. Mosso, J. Brock, and J. Hochstein, *Volume tracking of interfaces having surface tension in two and three dimensions*, in *Proceedings of the 34th AIAA Aerospace Sciences Meeting and Exhibit, 15-18 January 1996, Reno (USA)* (1996).
- [83] J. Li, *Calcul d'interface affine par morceaux (piecewise linear interface calculation)*, C.R. Acad. Sci. Paris, série IIb (Paris) **320**, 391 (1995).
- [84] V. Gopala and B. van Wachem, *Volume of fluid methods for immiscible-fluid and free-surface flows*, Chem. Eng. J. **141**, 204 (2008).
- [85] S. Afkhami and M. Bussmann, *Height functions for applying contact angles to 3D VOF simulations*, Int. J. Numer. Meth. Fluids **61**, 827 (2009).

- [86] Z. Wang, J. Yang, B. Koo, and F. Stern, *A coupled level set and volume-of-fluid method for sharp interface simulation of plunging breaking waves*, Int. J. Multi-phase Flow **35**, 227 (2009).
- [87] D. de Niem, E. Kührt, and U. Motschmann, *A volume-of-fluid method for simulation of compressible axisymmetric multi-material flow*, Comput. Phys. Commun. **176**, 170 (2007).
- [88] E. Aulisa, S. Manservigi, R. Scardovelli, and S. Zaleski, *A geometrical area-preserving Volume-of-Fluid advection method*, J. Comput. Phys. **192**, 355 (2003).
- [89] H. Tang, L. Wrobel, and Z. Fan, *Tracking of immiscible interfaces in multiple-material mixing processes*, Comput. Mater. Sci. **29**, 103 (2004).
- [90] B. Leonard, A. Lock, and M. MacVean, *Extended numerical integration for genuinely multidimensional advective transport insuring conservation*, (in: C. Taylor and P. Durbetaki (Eds.), Numerical Methods in Laminar and Turbulent Flow, vol. 9, Pineridge Press, New York, 1995) pp. 1–12.
- [91] B. Leonard, M. MacVean, and A. Lock, *The flux integral method for multidimensional convection and diffusion*, Appl. Math. Modelling **19**, 333 (1995).
- [92] R. Leveque, *High-resolution conservative algorithms for advection in incompressible flow*, SIAM J. Numer. Anal. **33**, 627 (1996).
- [93] J. Gerrits, *Dynamics of liquid-filled spacecraft*, Ph.D. thesis, University of Groningen (2001).
- [94] J. Bell, P. Colella, and H. Glaz, *A second-order projection method for the incompressible Navier-Stokes equations*, J. Comput. Phys. **85**, 257 (1989).
- [95] D. Harvie and D. Fletcher, *A new volume of fluid advection algorithm: The stream scheme*, J. Comput. Phys. **162**, 1 (2000).
- [96] J. Hernández, J. López, P. Gómez, C. Zanzi, and F. Faura, *A new volume of fluid method in three dimensions-Part 1: Multidimensional advection method with face-matched flux polyhedra*, Int. J. Numer. Meth. Fluids **58**, 897 (2008).
- [97] M. Rienecker and J. Fenton, *A Fourier approximation method for steady water-waves*, J. Fluid Mech. **104**, 119 (1981).

5

VALIDATION

IN this chapter, results from validation studies are presented. We apply the numerical methods, explained in the previous chapters, to two tests: breaking dam and wave run-up on a semi-submersible. Particular attention will be paid to the performance of the VOF methods and the GABC in the numerical simulations. Results from the numerical simulations will be compared with the experiments. The setup of the numerical and experimental tests will be given below in detail.

5.1. SEMI-SUBMERSIBLE TEST

The first test is the wave run-up on a semi-submersible. A semi-submersible is a specialized floating vessel used as drilling rig, crane vessel, production and accommodation facility. This vessel consists of a platform-type deck supported by structural columns that connect the deck and watertight pontoons located below the water level, see Fig. 5.1 for the images of two semi-submersibles. While the deck is kept well away from the wave impact, the hull structure is submerged at a deep draft, which makes the semi-submersible generally show far less motions in rough waters than mono-hull vessels. Because of their good stability and seakeeping characteristics they are especially suitable for rather harsh conditions. Another advantage of this vessel is the large, open deck designed to accommodate their colossal cargo. This deck may be used as working space and it may contain various types of machinery such as drilling equipments.

Wave impact on a semi-submersible is a challenging test because it involves many complex physical and numerical phenomena: generation of steep water waves, non-reflecting boundary conditions, free-surface modeling, turbulence effects, air entrapment, fluid-structure interaction among others. It is important to note that observing relative effects of each of these phenomena on wave impact loads is a complicated and much-needed work in itself, and outside the scope of this study. Here we will merely focus our attention on the effects of free-surface modeling as various VOF modules will be used in the numerical simulations. Performances of these modules will be compared to the experimental results in terms of free surface elevation and wave impact pressure



Figure 5.1: Photographs of two semi-submersibles.

at various measurement positions. At the outflow boundaries where waves are propagating out of the computational domain, the GABC is applied. Therefore, we will also keep an eye on the performance of this boundary condition. Next, we will first give some information concerning experimental modeling, and afterwards we will describe the setup of the numerical simulation.

5.1.1. EXPERIMENTAL MODELING

The tests were carried out at Maritime Research Institute in the Netherlands (MARIN) in 2008. Several experimental parameters were specified, such as the design of the semi-submersible and the generated sea states, to provide a comprehensive analysis regarding the physics behind wave impact loads on a semi-submersible. In this section, we present some details of the test considered in this study. The 1:50 scale of the semi-submersible was modeled and its position was fixed in the High Speed Basin with dimensions 200m long by 4m wide by 4m deep. Fig. 5.2a shows the photograph of the semi model in the work shop and Fig. 5.2b in the experiment. The basin is equipped with a flap-type wave generator at one end and a beach at the other end. The center of the test set-up was located 100m away from the wave flap (note that, for simplicity, all the details are written in prototype values in the remainder of this section unless stated otherwise).

The surface elevations were measured using resistance type wave probes at various locations on and around the model. Additionally, piezo-type pressure transducers were attached to the structure to monitor wave impact pressures for the duration of the tests, see Fig. 5.3 for the positions of the measurement instruments which will be utilized to compare the numerical results to the experimental results and for the general plan of the semi-submersible at full scale. Also, video recordings were made during the model tests



(a) The 1:50 model of the semi-submersible in the work shop.

(b) Wave run-up and impact on the forward column and deck in regular waves.

Figure 5.2: Photographs of the semi-submersible model in the work shop and in the experiment.

using various types of cameras. The measurements ceased as soon as reflected waves from the beach started to arrive at the semi model in the basin.

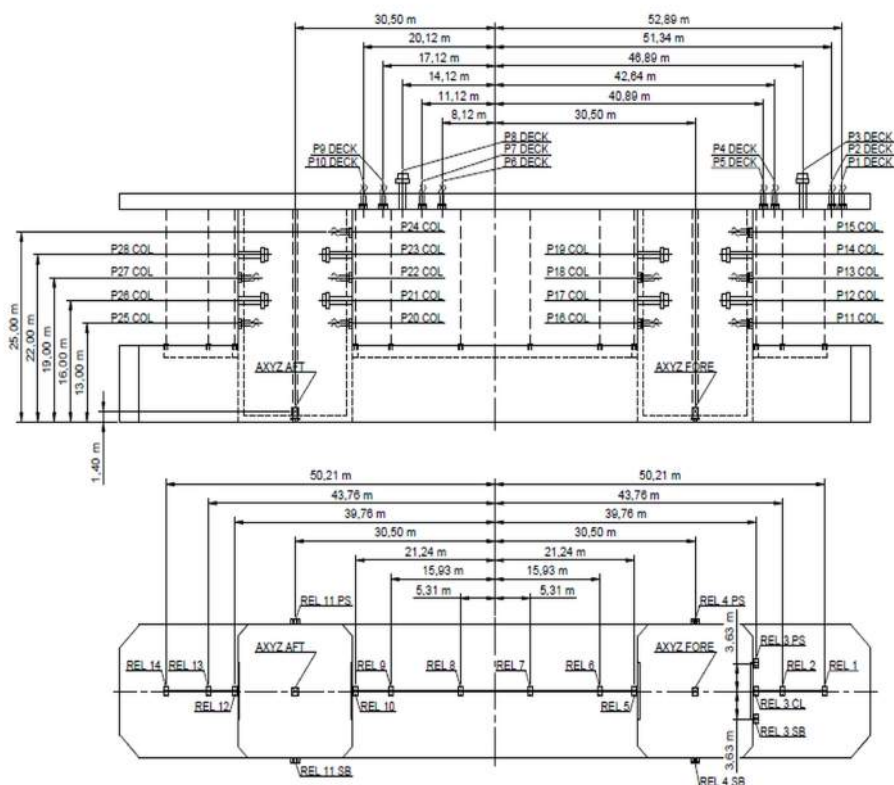


Figure 5.3: General plan of the semi-submersible at full scale with the locations of the wave probes and pressure transducers. Waves come from the right.

The waves were produced with a flap-type wave generator at one end of the basin. Several sea states were generated including regular and irregular long crested head waves. Here we will consider only two regular waves shown in Table 5.1.

Waves	<i>period</i> T(s)	<i>height</i> H(m)	<i>length</i> L(m)	<i>steepness</i> H/L(%)
W9	9	7.5	130	5.8
W11	11	14.7	199	7.4

Table 5.1: Characteristics of the 5th-order Stokes waves.

5.1.2. NUMERICAL SIMULATION

The setup of the computational domain is given in Fig. 5.4. The structure is a semi-submersible consisting of two columns, a pontoon under water and a deck above water. The distance between the inflow boundary Γ_W and the semi-submersible is one wavelength. Similarly we also add a distance of one wavelength on the trailing side of the semi-submersible. Considering the length of the semi-submersible along this direction, 114.5m, the size of the domain along the x -direction becomes $Dx = 1L + 114.5 + 1L$ where L denotes the wavelength. This suggests that Dx is 374.5m in the simulation with the wave W9, and 512.5m with the wave W11. Along the y -direction the full width of the High Speed Basin is modeled, which is $Dy = 200m$, and the semi-submersible is located at the center along this direction. The dimension of the computational domain in the z -direction is $Dz = 210m$ and the water depth is 180m.

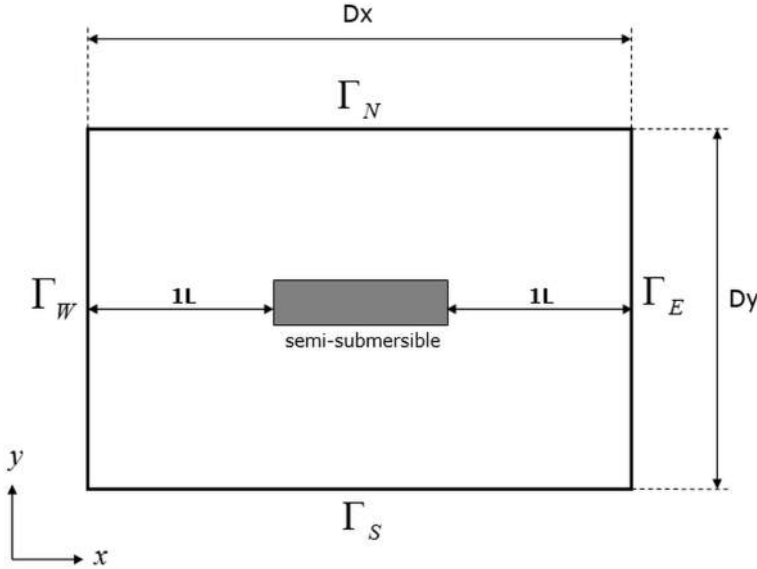


Figure 5.4: Setup of the computational domain on the $x - y$ plane.

To capture the details of the flow near the structure, a finely spaced grid in all directions is used as sketched in Fig. 5.5a. However, far away from the structure, the grid becomes progressively coarser. Two grid resolutions are considered for both waves. In the simulation with the wave W9, a resolution of $1.5m$ near the structure on a stretched grid with a total number of 1.2 million cells, and a resolution of $0.85m$ near the structure on a stretched grid with 3.9 million cells are generated in the computational domain. In the simulation with the wave W11, the stretched grid has a resolution of $1.2m$ near the structure and a total number of 1.8 million cells, while the refined grid for this wave has a resolution of $0.73m$ near the structure and a total number of 4.3 million cells. It is clear that grid refinement study was not performed on an ideal setting. If we take f as a refinement factor and include time in the refinement as well, we will have a work increase of a factor of f^4 between any two levels in an ideal grid refinement study. Considering the durations and total point counts of the numerical simulations considered in the current work, this grid refinement procedure causes substantial computational cost. With the computing power at our disposal this procedure was extremely challenging. Instead we increased the total number of cells only mildly and applied a more aggressive stretching near the semi-submersible to see the effect of grid refinement on the solution.

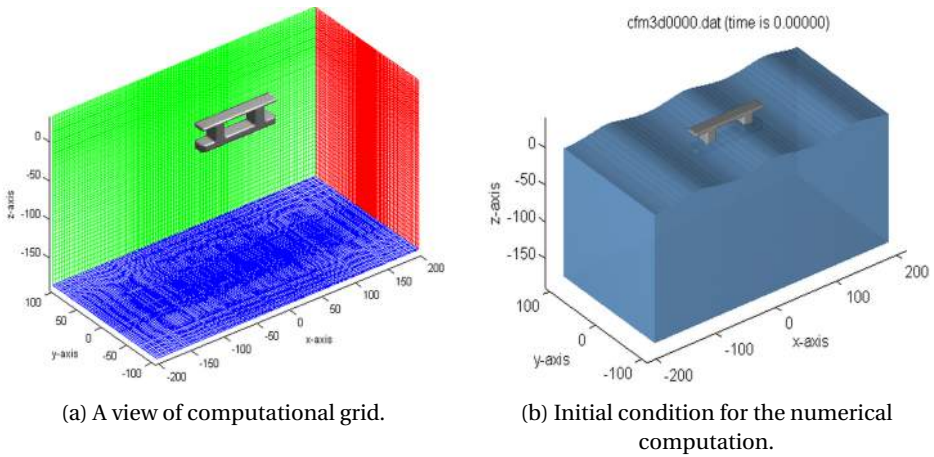


Figure 5.5: Setup of the numerical simulation.

At $t = 0$ the wave is generated and initialized everywhere in the computational domain according to the considered wave theory, see Fig. 5.5b for an illustration of the initial condition. In the numerical simulations we used the fifth-order Stokes theory [1]. At every time step after the initial condition, the flow variables are calculated according to the fifth-order Stokes theory, and prescribed at the inflow boundary Γ_W . On the east boundary Γ_E the generating and absorbing boundary condition (GABC) is applied. The boundary conditions applied on the other boundaries of the computational domain are explained in Chapter 2.

5.1.3. RESULTS

The durations of the numerical simulations are 33 seconds for W9 and 38 second for W11. The experiments, on the other hand, are performed for a duration of 1850 seconds. Therefore, at first, we had to specify a short time window from each experiment during which the wave field is fully developed but the effect of the sidewalls and the beach is not dominant. After analyzing time traces of the measured wave elevations, $t = 1100s$ is selected as the starting point for W9, and $t = 1025s$ is selected for W11. During subsequent time windows, comparisons are made between the numerical simulations and the experiment.

RESULTS FOR W9

In this section, we present the results from the semi-submersible test with the wave W9. Numerical and experimental results are compared in terms of wave elevation and wave impact pressure at various positions, see Fig. 5.3 for the locations of the measurement instruments.

In the numerical simulations, we use three PLIC methods: MYC, LSG and ELVIRA. These methods are combined only with the MACHO advection scheme. The reason for this choice was explained in Chapter 4: the performance of the MACHO scheme is very close to that of the COSMIC scheme provided that the order of advection along each coordinate is alternated at every time step. This procedure reduces directional bias, and as a result MACHO performs similarly to the inherently symmetric COSMIC. This result combined with the fact that MACHO is considerably cheaper than COSMIC makes the choice between these methods clear. The PLIC methods combined with the MACHO advection scheme are also compared with the HN&LHF method. The same numerical parameters are set in all the numerical simulations to ensure that the analysis is based on fair results. The numerical simulations are performed on two grid resolutions to observe the effect of grid refinement on the results, more information about the grids is given above.

Figures 5.6 and 5.7 illustrate the surface elevation at four measurement positions: in front of the forward column (REL1), the forward column run-up (REL3CL), between the two columns (REL8) and the aft column run-up (REL10). At all the measurement positions, we do not observe a significant difference between the four VOF methods on either of the grid resolutions. The results also indicate that grid refinement improves the agreement between the numerical methods and experiment at three positions, namely REL1, REL3CL and REL8, while the results change only slightly at REL10. On both grids, the numerical methods generally estimate the phases successfully. The crest amplitudes are, however, somewhat overestimated.

Concerning the performance of the absorbing boundary condition, the free surface elevation at especially REL10 gives some clues as this position is the closest to the outflow boundaries. Careful analysis of the results points out that the crest and/or trough elevations do not increase or decrease in a dramatic fashion during the simulation. Changes in the crest elevation, which is more dramatically present in the experiment, can be attributed to the effect of diffraction.

Figures 5.8 to 5.10 show the wave impact pressure at six measurement positions: P11COL, P13COL, P15COL, P17COL, P23COL and P27COL. Wave impact pressures are

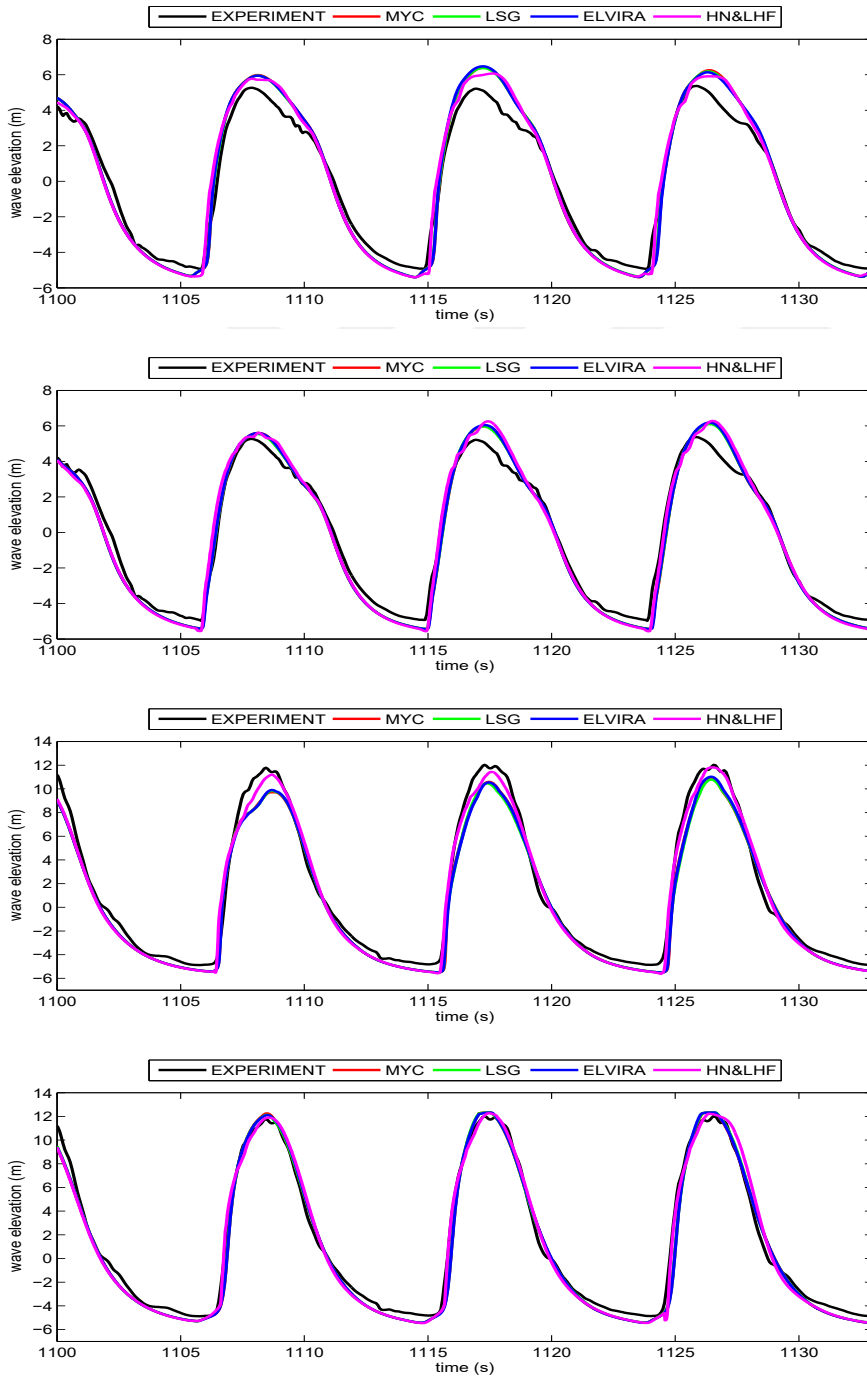


Figure 5.6: The free surface elevation at Wave Sensors REL1 and REL3 for the wave W9. From top to bottom results are shown for REL1 (coarse grid), REL1 (fine grid), REL3 (coarse grid), and REL3 (fine grid).

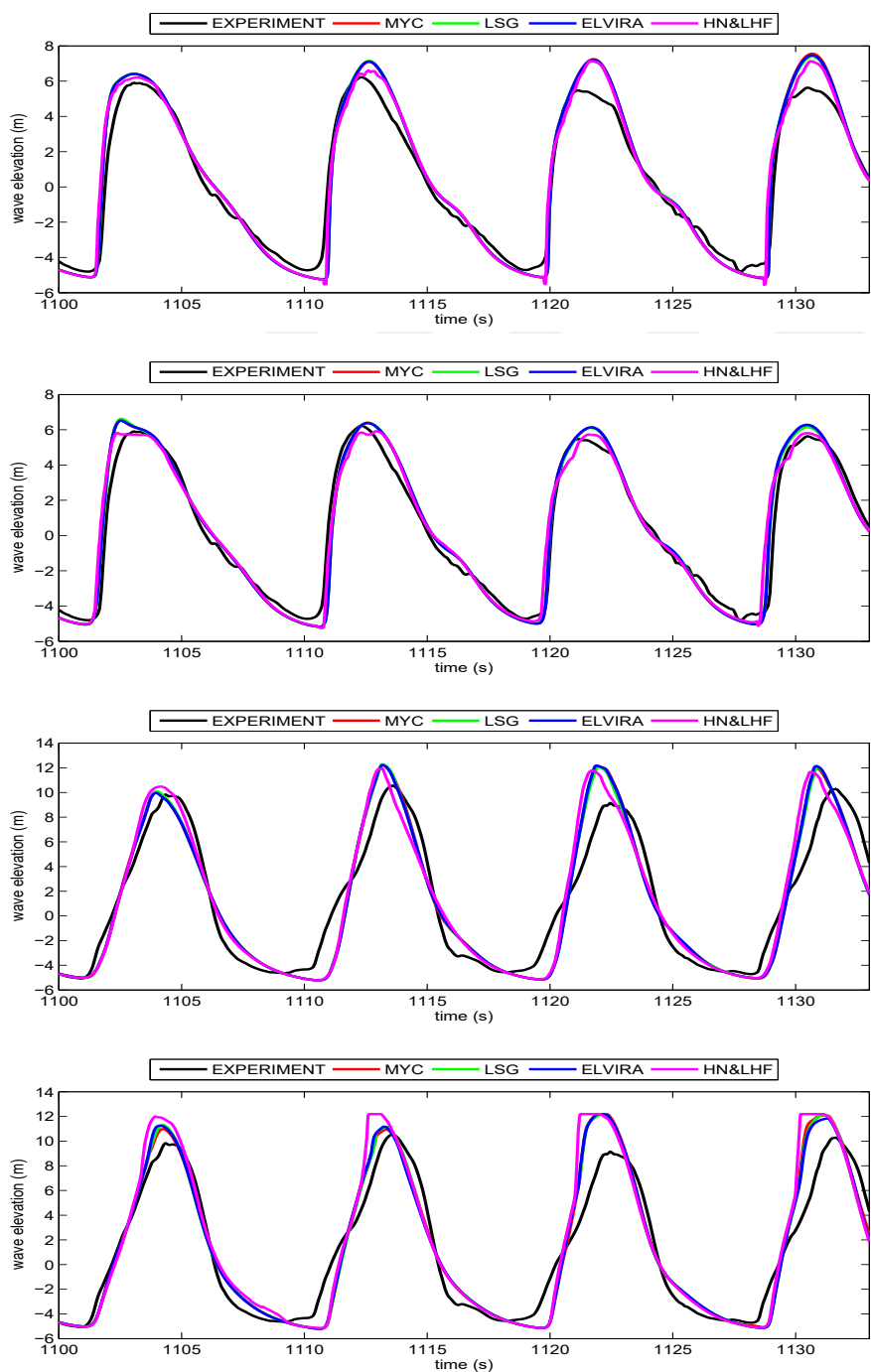


Figure 5.7: The free surface elevation at Wave Sensors REL8 and REL10 for the wave W9. From top to bottom results are shown for REL8 (coarse grid), REL8 (fine grid), REL10 (coarse grid), and REL10 (fine grid).

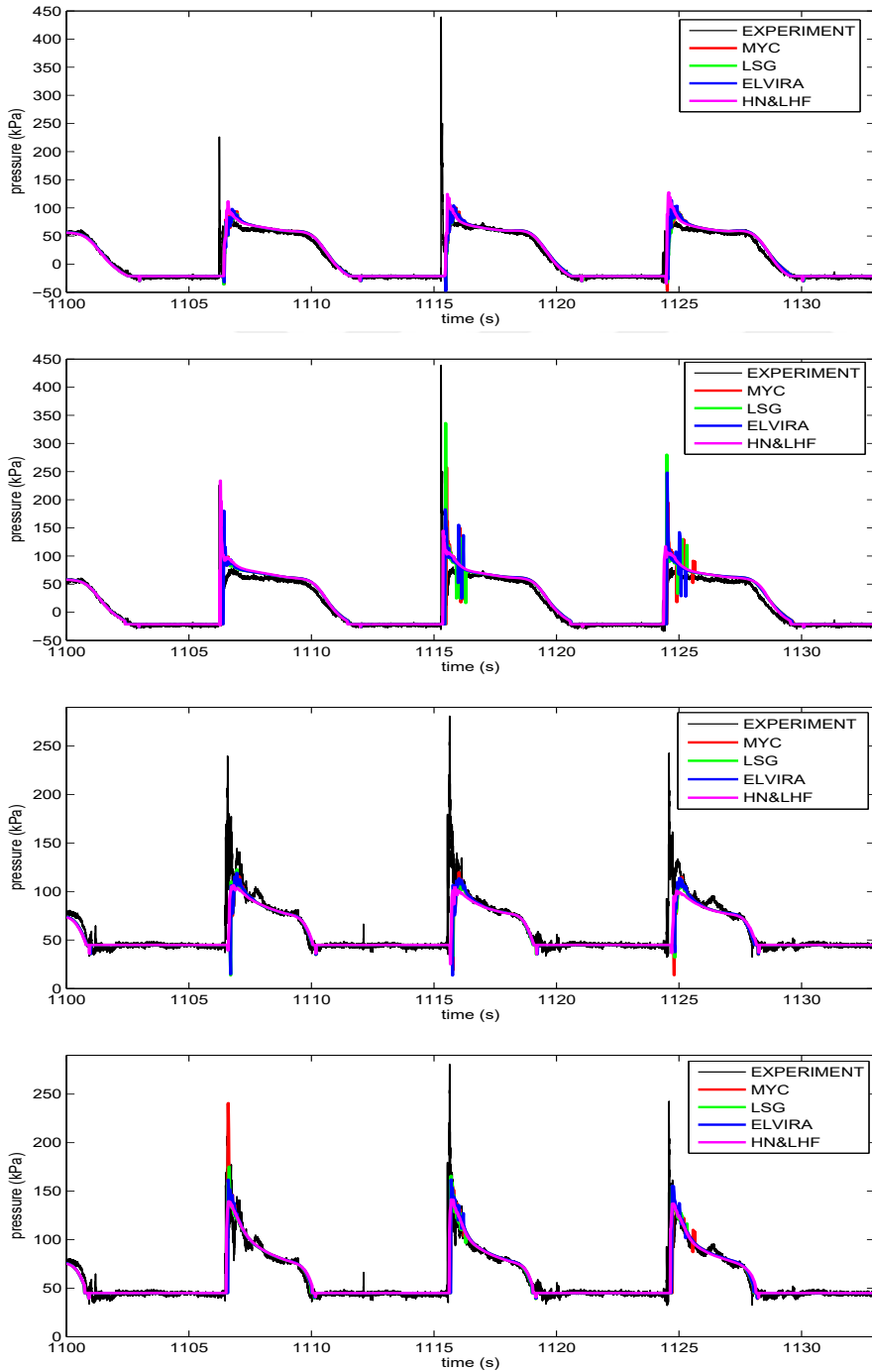


Figure 5.8: Wave impact pressure at Pressure Sensors P11COL and P13COL for the wave W9. From top to bottom results are shown for P11COL (coarse grid), P11COL (fine grid), P13COL (coarse grid), and P13COL (fine grid).

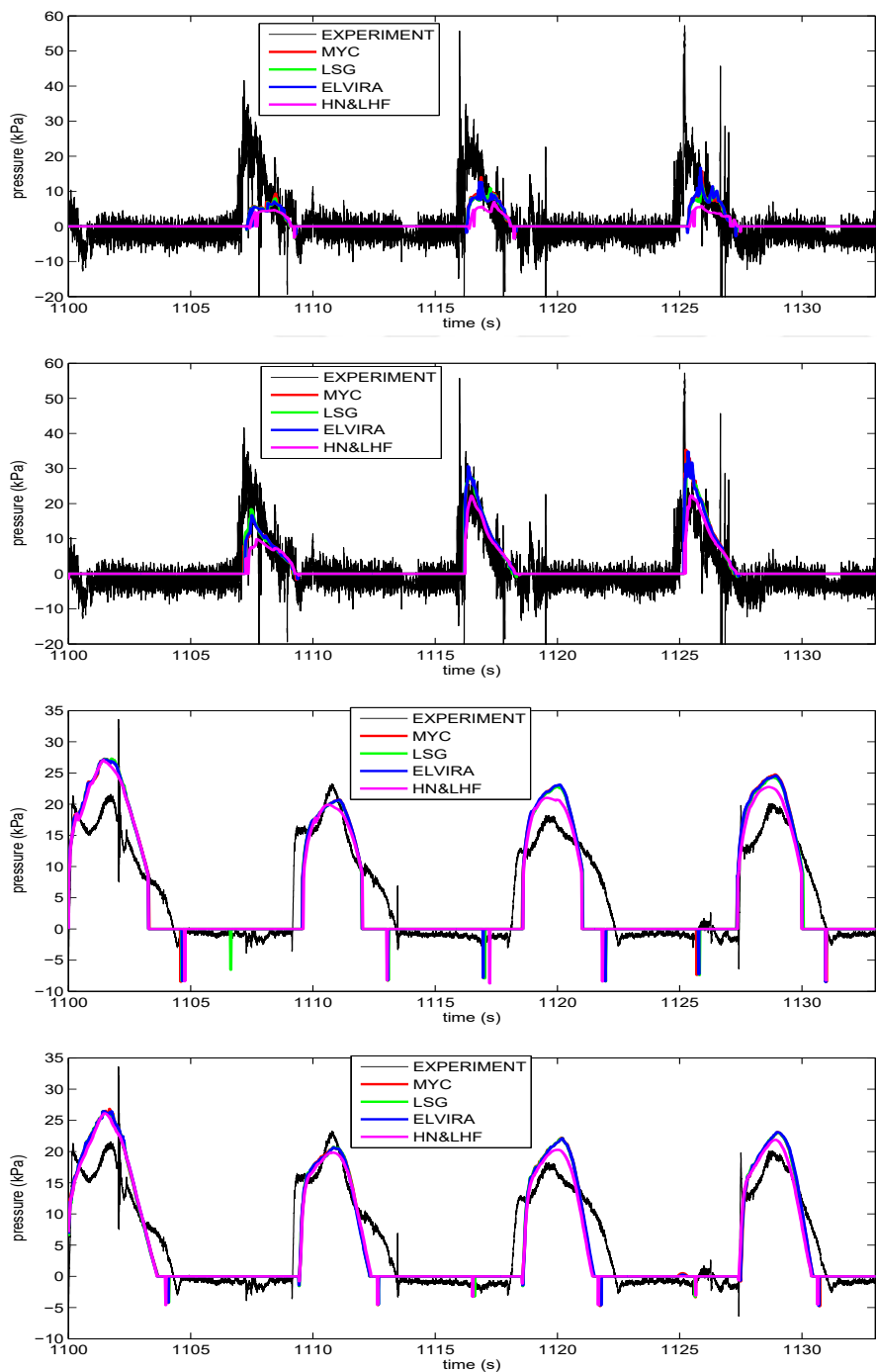


Figure 5.9: Wave impact pressure at Pressure Sensors P15COL and P17COL for the wave W9. From top to bottom results are shown for P15COL (coarse grid), P15COL (fine grid), P17COL (coarse grid), and P17COL (fine grid).

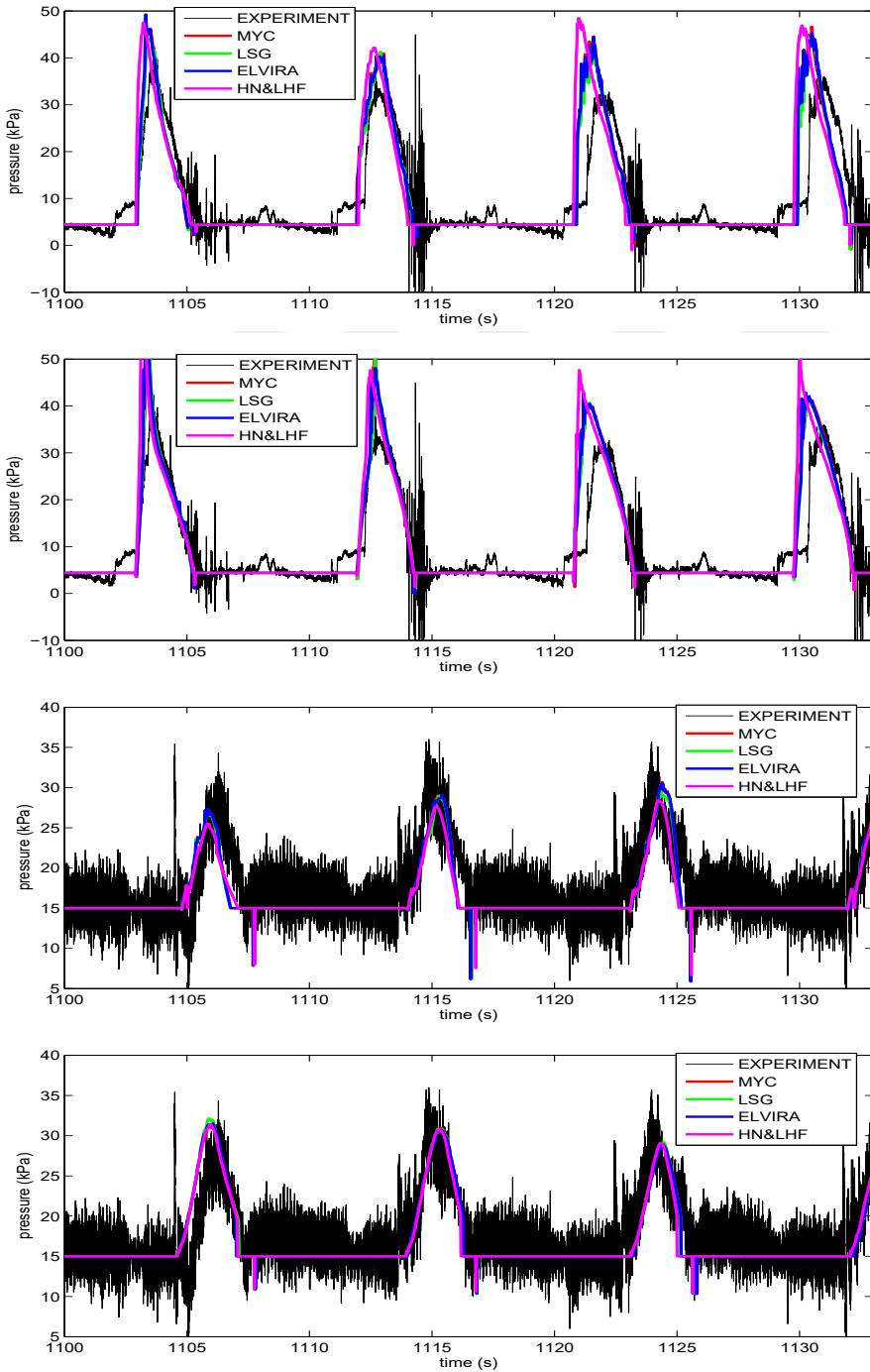


Figure 5.10: Wave impact pressure at Pressure Sensors P23COL and P27COL for the wave W9. From top to bottom results are shown for P23COL (coarse grid), P23COL (fine grid), P27COL (coarse grid), and P27COL (fine grid).

presented only on the two columns, since the waves did not touch the underside of the deck in the numerical simulations. Results from the pressure sensors in front of the forward column demonstrate the superior performance of the PLIC-MACHO combinations over the HN&LHF method. On the coarse grid, PLIC-MACHO methods perform similarly to the HN&LHF at two lower positions (P11COL and P13COL), but much better at the top position (P15COL). When the grid is refined, results from all the methods improve. In this case, high pressure peaks are better captured with the PLIC-MACHO methods. Among the PLIC methods, however, we do not observe a clear difference. In front of the forward column, the numerical methods tend to underestimate the pressure peaks on both grids, while behind the forward column they overestimate the impact pressure. In front of and behind the forward column, we do not observe a phase shift, which is consistent with the wave elevation results at this position.

In front of the aft column at the position P23COL, the numerical methods overestimate the impact pressure while resulting in a slight phase shift. The phase shift was also observed in the wave elevation results at this position. Behind the aft column at the position P27COL, all the VOF methods produce similar values for the impact pressure. We also see that the agreement between the numerical and experimental results is better on the fine grid at this measurement position.

It is important to note that there is a clear difference in wave impact pressure between the PLIC-MACHO methods and HN&LHF, for example in front of the forward column at the pressure sensor P15COL, although all the VOF methods at this position yield similar wave elevations. This outcome was also encountered in the simulation with the wave W11. We will take a closer look at this situation when the results from W11 are presented in the next section.

RESULTS FOR W11

In the previous section, three PLIC methods were combined with the MACHO advection scheme in the semi-submersible test with the wave W9. Taking into account the results from this test and the analysis made in Chapter 4, we will choose only one PLIC-MACHO combination in the following test: MACHO&LSG. The MACHO&LSG module will be compared to the HN&LHF method as this was the standard VOF method in the ComFLOW code before this study. Simulations are performed on two grid resolutions, and the comparison is made in terms of wave elevation and wave impact pressure as in the previous test. The wave that we generate now, W11, is not only larger and higher but also steeper than W9. This consequently necessitates using a larger number of grid cells and a bigger computational domain compared to the previous test.

Figure 5.11 shows snapshots throughout the numerical simulation with the wave W11. We can observe some critical moments in these snapshots: waves running up on the forward and aft columns, water level on the columns dropping, area between the two columns being dry or completely submerged.

Figures 5.12 and 5.13 show the wave elevation from the four wave sensors, in front of the forward column (REL1), the forward column run-up (REL3CL), between the two columns (REL8) and the aft column run-up (REL10). The results indicate that wave elevations at four locations from the numerical simulations and the experiment are in good agreement. We observe that grid refinement does not change the results considerably, and both VOF methods yield similar wave elevations.

The free surface elevation from the wave sensor REL10 demonstrates that reflection from the outflow boundaries does not affect the results as any increase or decrease in the crest and/or trough elevations does not take place during the simulation.

Figures 5.14 to 5.16 show the wave impact pressure at the six measurement positions two of which are on deck and the rest are on the columns: P4DECK, P5DECK, P13COL, P15COL, P24COL and P28COL. Results from the pressure sensors on the deck demonstrate the superior performance of the MACHO&LSG over the HN&LHF method. This is also valid for the pressure sensors in front of the forward column, P13COL and P15COL. In front of the aft column from the sensor P24COL, a clear difference is observed only during the third impact which takes place at around $t = 1054$ s. Both VOF methods result in similar impact pressures at the position P28COL behind the aft column.

Although the wave elevations from both VOF methods at the four measurement positions are hardly different, there is a noticeable difference in wave impact pressure. For example, pressure sensors P13COL, P15COL, P4DECK and P5DECK record higher impact pressures when MACHO&LSG is used especially on the fine grid, although the wave sensor REL3CL records similar surface elevations. To understand where this difference comes from, we need to take a closer look at the results in front of the forward column. Figure 5.17 illustrates three snapshots which are taken when the first wave runs up on the forward column and afterwards hits the underside of the deck. First of all, we observe that the way the wave hits the underside of the deck and propagates in the negative x -direction on the surface of the underside of the deck is different. This difference is a manifestation of how the free surface is modeled in these VOF methods. These VOF methods have different free surface reconstruction and advection procedures, which yields different fluid configurations after the transportation of the fluid at each time step is completed. Second of all, we also observe that there are differences in velocity values in the z -direction from the two VOF methods (similar differences were also noticed in velocity values in the x -direction). The difference in the fluid topology results in differences in cell labels, which consequently affects the type of equations solved in each computational cell. Thus, the resulting values for velocity and pressure becomes different. This shows that although the VOF method is the part of the overall code which only transports fluid from one grid cell to another, velocity and pressure are unavoidably affected by the fashion this transportation takes place.

5.2. BREAKING DAM TEST

The second test we consider is the breaking dam. Resembling green water flow on the deck of a ship, this test is easy to setup as the geometry of the structure is simple and there is no need for special inflow or outflow boundary conditions. As the flow is rather violent and a significant amount of topological change occurs in fluid configuration, the breaking dam is useful to test the stability and robustness of the overall numerical code as well as the VOF method.

5.2.1. EXPERIMENTAL MODELING

The breaking dam experiment was carried out at MARIN in 2006. In a large tank with dimensions of $3.22\text{m} \times 1\text{m} \times 1\text{m}$ in the x , y and z directions respectively, a solid box of

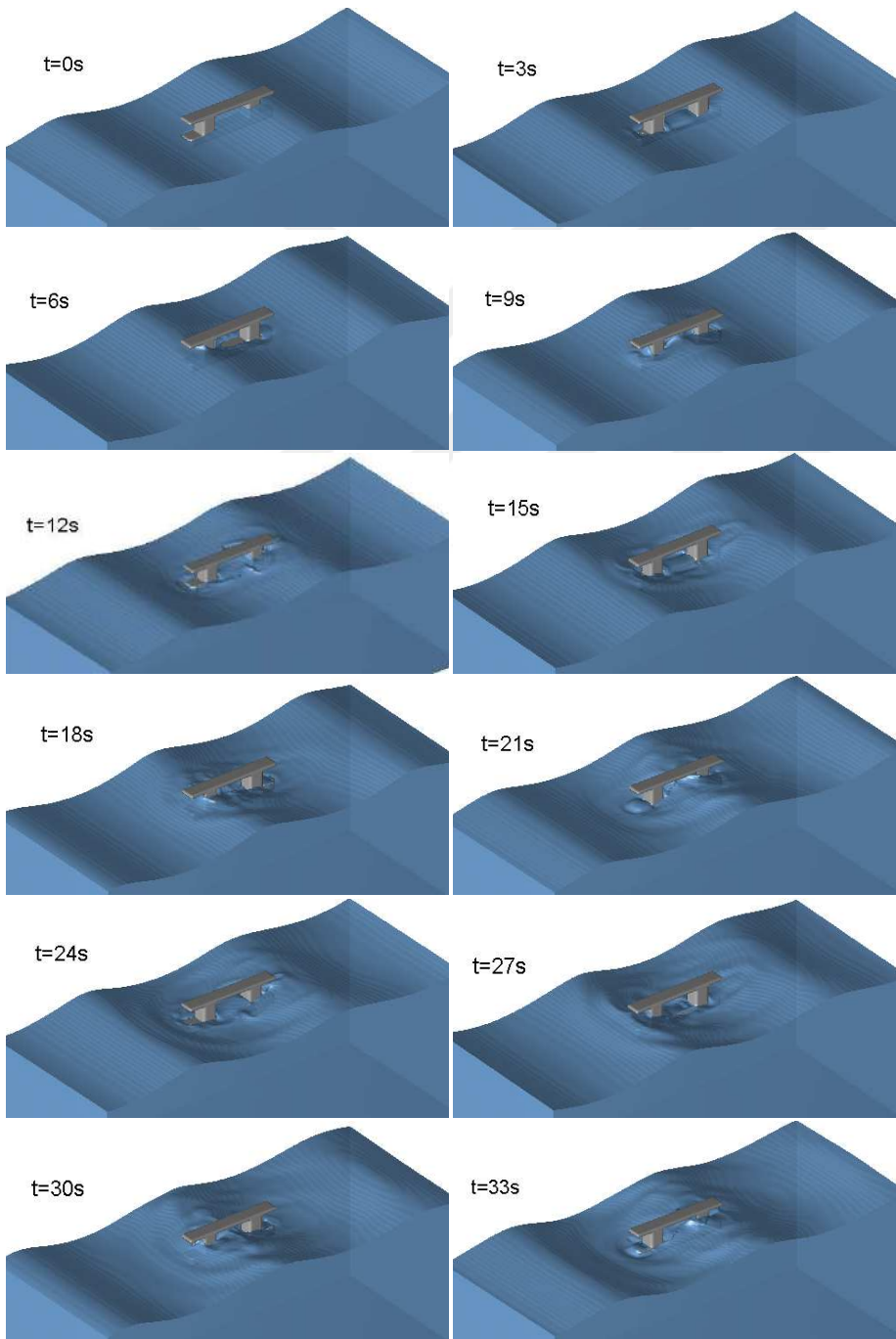
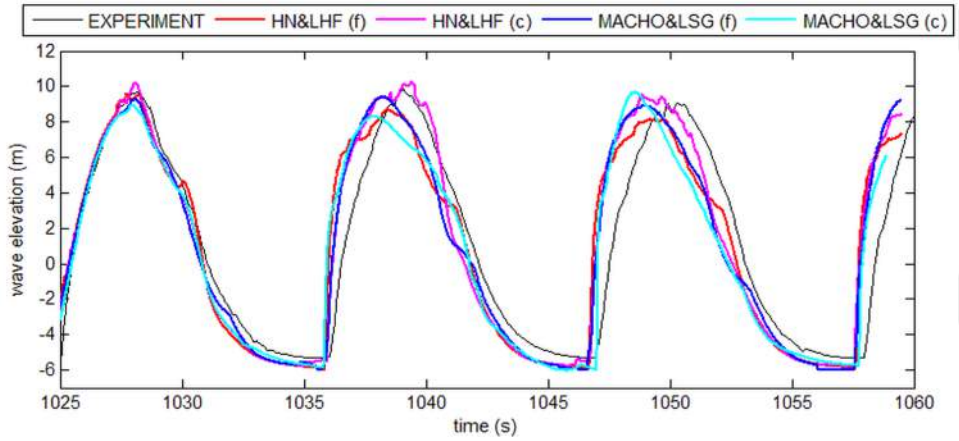
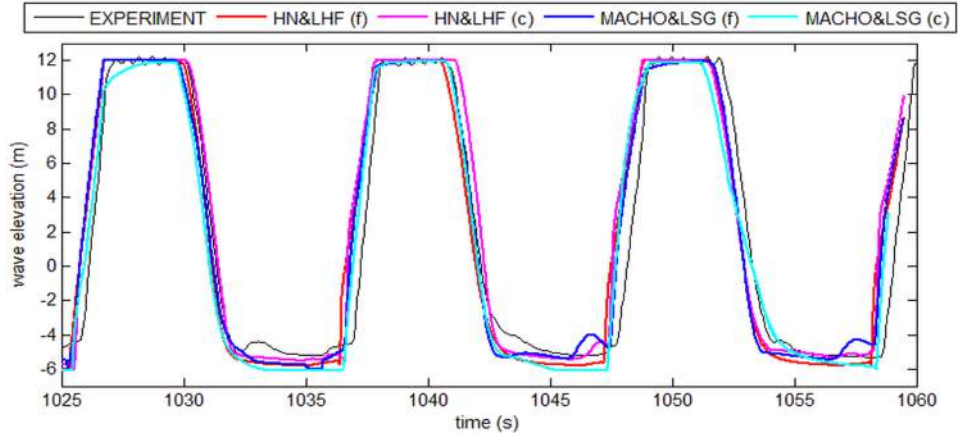


Figure 5.11: Snapshots of the numerical simulation at various instances in time for the wave W11. Waves are propagating from the left to the right. The VOF module MACHO&LSG is used on the fine grid.

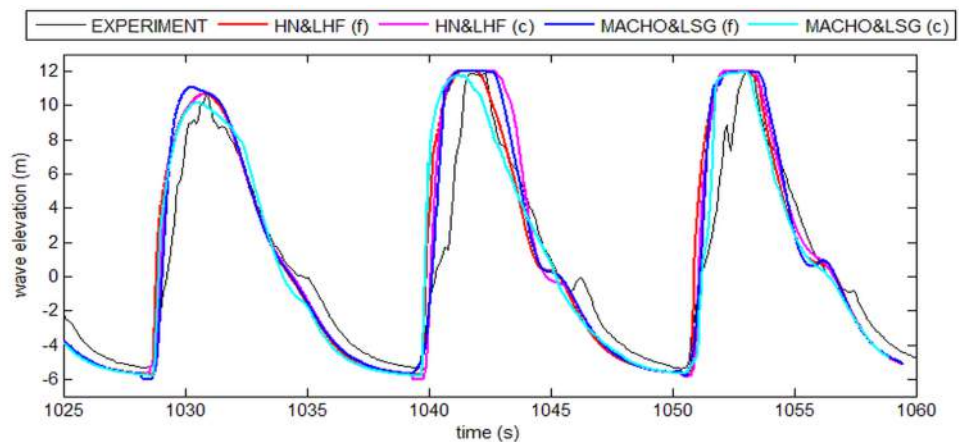


(a) Wave Sensor REL1

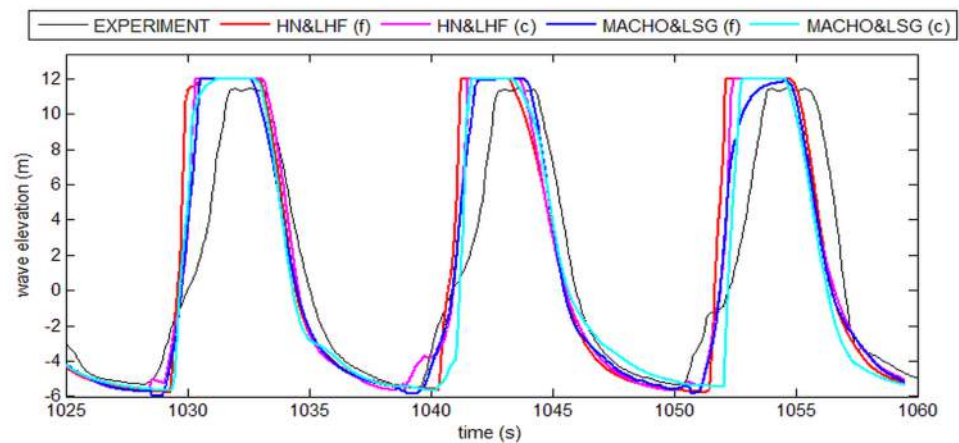


(b) Wave Sensor REL3CL

Figure 5.12: The free surface elevation at Wave Sensors REL1 and REL3CL for the wave W11. f stands for the fine grid, and c stands for the coarse grid.

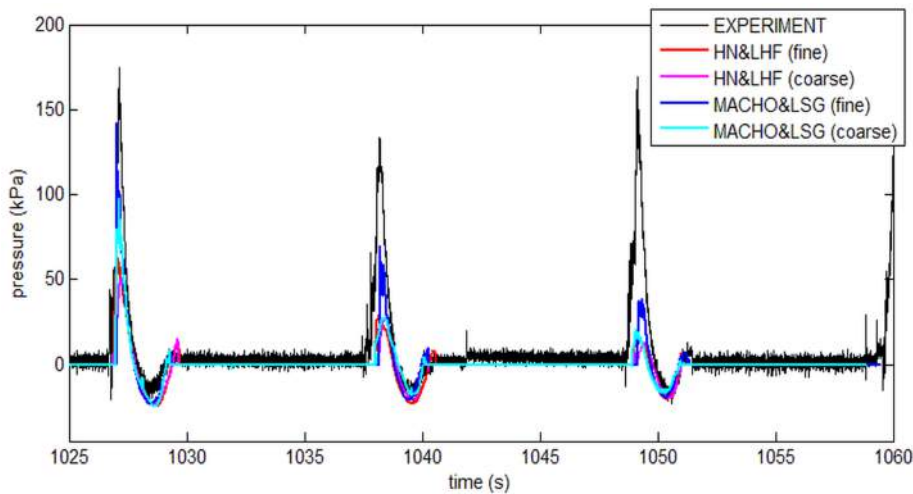


(a) Wave Sensor REL8

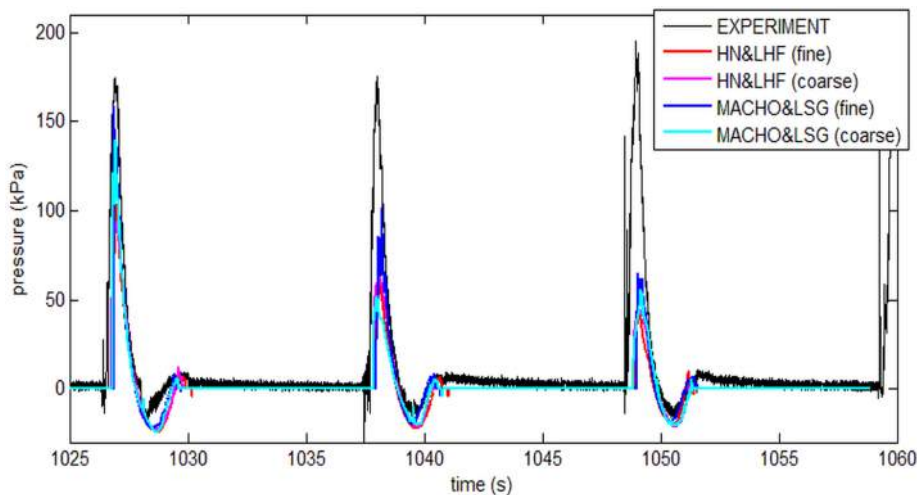


(b) Wave Sensor REL10

Figure 5.13: The free surface elevation at Wave Sensors REL8 and REL10 for the wave W11. *f* stands for the fine grid, and *c* stands for the coarse grid.

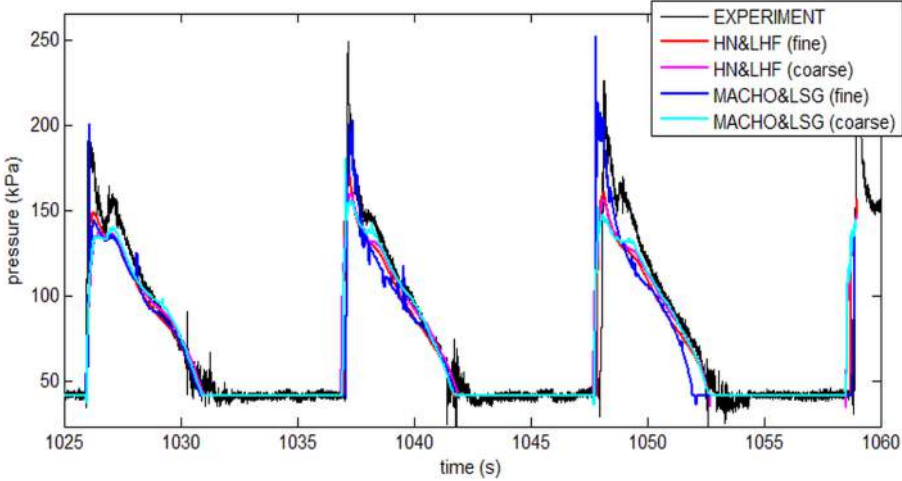


(a) Pressure Sensor P4DECK

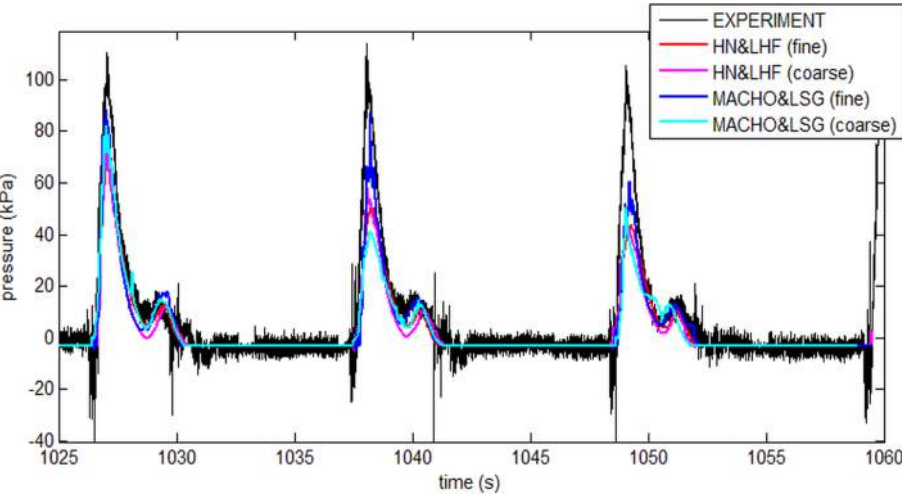


(b) Pressure Sensor P5DECK

Figure 5.14: Wave impact pressure at Pressure Sensors P4DECK and P5DECK for the wave W11.

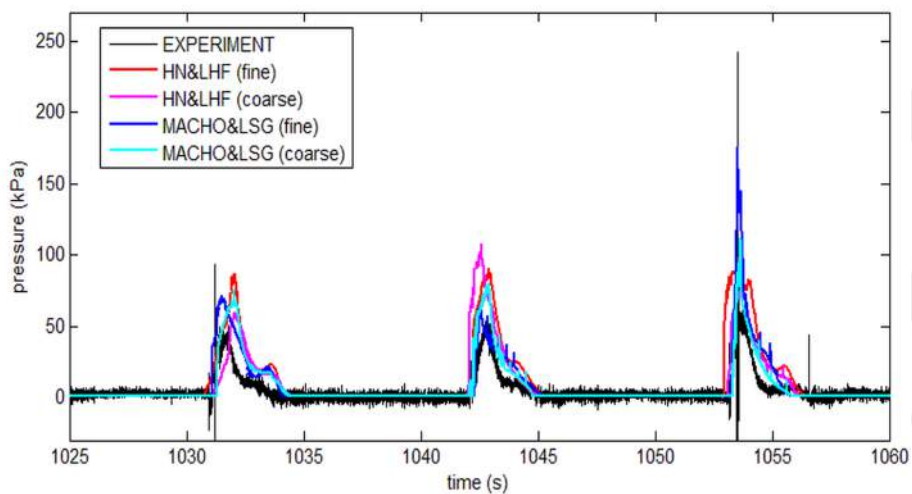


(a) Pressure Sensor P13COL

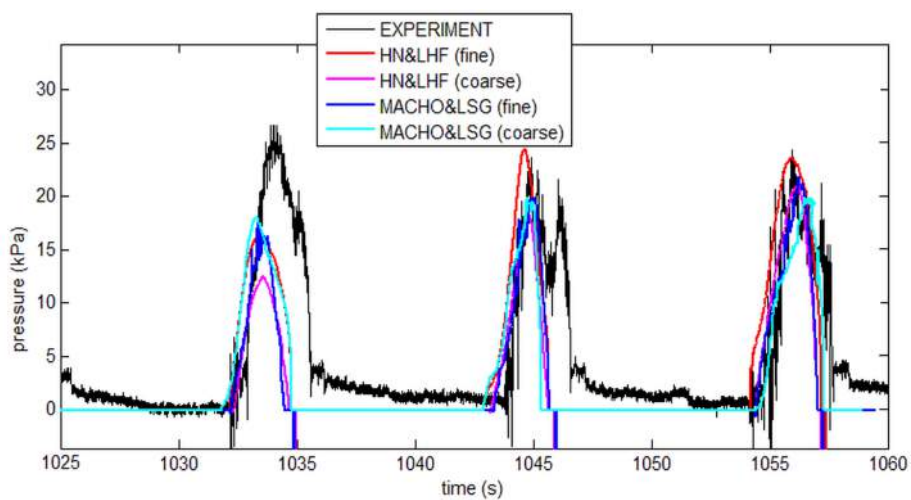


(b) Pressure Sensor P15COL

Figure 5.15: Wave impact pressure at Pressure Sensors P13COL and P15COL for the wave W11.



(a) Pressure Sensor P24COL



(b) Pressure Sensor P28COL

Figure 5.16: Wave impact pressure at Pressure Sensors P24COL and P28COL for the wave W11.

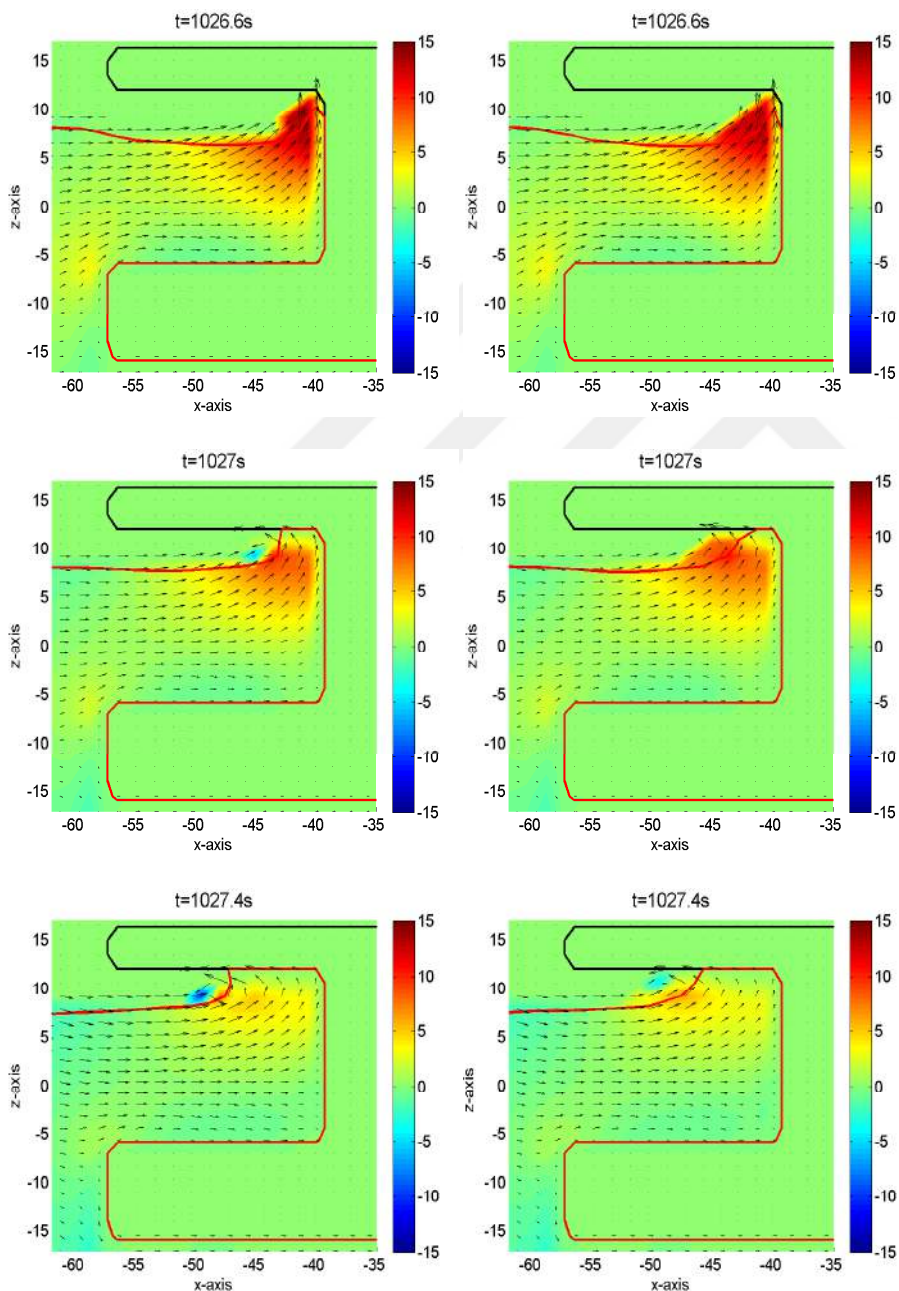


Figure 5.17: First wave run-up on the forward column. Snapshots are shown at three instances in time. Snapshots on the left column are taken from the simulation with the HN&LHF method, and the snapshots on the right with the MACHO&LSG method. Colorbar corresponds to the velocity component in the z -direction. Arrows indicate velocity vectors at various points in space.

$0.16m \times 0.4m \times 0.16m$ is placed at the bottom. At $x = 2m$ a lock-gate is positioned. Upstream of the lock-gate $1.22m \times 1m \times 0.55m$ of water is held at rest. Once the lock-gate is pulled up almost instantaneously, the water starts to collapse as a result of gravity generating a strong vertical velocity which is converted into a large horizontal component near the bottom.

The water level inside the tank is measured using resistance type wave probes at various locations. Additionally, piezo-type pressure transducers were installed on the box and on the left wall located at $x = 0m$ to record impact pressures for the duration of the test. Video recordings were also made during the model tests.

Experiments were carried out with different setups. For example, an experiment was conducted in an empty tank without the box. In another one, a balcony was installed on the left tank wall located at $x = 0m$. Rotation of the box with respect to the collapsing water front was also varied. The box was placed transverse to the flow or with an angle of 45° (quartering box). Here we will consider the case where the box is placed transverse to the flow as depicted in Fig. 5.18.

5.2.2. NUMERICAL SIMULATION

The breaking dam test conducted in this work is a single-phase flow: there is only water in the tank. The effect of the second phase on the results is thoroughly investigated by Wemmenhove [2]. In the numerical simulations the first-order upwind scheme is used for the spatial discretization, and the first-order Forward Euler scheme is used for the time integration. The simulations are performed for a duration of 6 seconds. During the simulation the CFL number fluctuates around 0.5 but sometimes reaches a maximum value of 0.77.

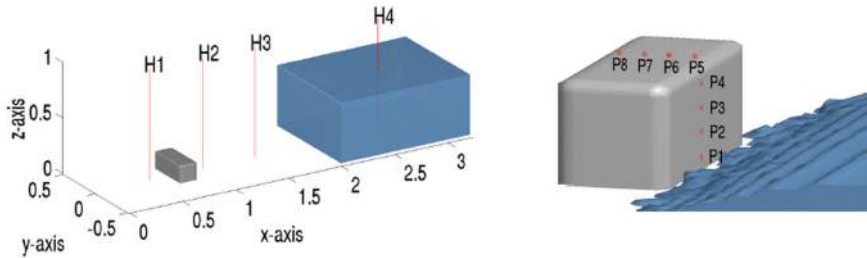


Figure 5.18: Locations of the measurement devices for water height and pressure.

The setup of the computational domain is given in Fig. 5.18. Wave probes H1 to H4 record the water elevation, and pressure sensors P1 to P8 which are installed on the box record the impact pressure. We use two VOF methods in the numerical simulations, HN&LHF and MACHO&LSG. Two grid configurations are considered: the coarse grid has $118 \times 38 \times 34$ cells in the x , y and z directions, respectively. One level grid refinement results in a number of $236 \times 76 \times 68$ cells in the domain. Both grids are slightly stretched towards the bottom of the tank.

5.2.3. RESULTS

Figure 5.19 shows the water height records at various locations: in the reservoir (H4), between the reservoir and the box (H3), just in front of the box (H2) and just behind the box (H1). When we look at the water height probe in front of the box H2, we observe two peaks from the experiment at around $t = 1.8s$ and $t = 4.7s$. The first peak occurs when the water front returns back from the left tank wall and meets with the probe, see Fig. 5.20 for snapshots of the numerical simulation obtained by using MACHO&LSG on the grid of $236 \times 76 \times 68$ cells. Afterwards this front moves towards the right tank wall, runs up, returns back and meets with the probe for a second time. Apart from a slight delay for the second one, both peaks are captured well in the numerical simulations. At probes H3 and H4 numerical methods also agree well with the experiment. Among the four probes the most chaotic behavior from the numerical methods is inspected at probe H1. When we look at the snapshots taken at $t = 0.54s$ and $t = 0.72s$ in Fig. 5.20, we observe that when the water front hits the box, it moves in three main directions: a part moves towards the cavity between the box and the right side wall, another part moves into the cavity between the box and the left side wall and another part moves towards the ceiling in a jet-like motion. After a while these three parts merge into each other in the area behind the box causing a rather complicated and violent flow. Water probe H1 starts to measure water height at around $t = 1.0s$, and until $t = 3.0s$ this violent flow reigns in the area behind the box. This time window is where the numerical methods differ relatively largely from the experiment.

Figure 5.21 illustrates the time traces of impact pressure at four points on the box: P1, P3, P4 and P6. The pressure developments at points P1, P3 and P4 indicate that the moment of the initial impact on the box is very well captured in the numerical simulations. After the initial impact the water front runs up on the left tank wall and propagates back to the right tank wall. After it returns from the right tank wall, it impinges on the box for a second time. The second impact takes place at around $t = 4.7s$ in the experiment. The instant of the second impact on the box is captured in the numerical simulations with a slight delay of about $0.2s$. We also observe that grid refinement does not help decrease this delay.

The magnitude of the initial impact pressure is successfully captured in the numerical simulations at lower points on the box P1 and P3, but is underpredicted at the top point on the front side of the box P4. Figure 5.21 also demonstrates that the magnitude of the second impact pressure calculated by the numerical methods is also in good agreement with the experiment. Compared to other transducers P6 is the point where we see the largest differences between the simulations and experiment. As described earlier and can be seen in Fig. 5.20, the top side of the box where P6 transducer is located interacts with a rather violent and complicated flow between $t = 1.0s$ and $t = 2.0s$. During this time window significant amount of topological changes take place in the fluid in a single time step. These changes affect cell labels which affect the type of equations solved in each computational cell. This chain of events in the flow solver may need further modifications as it somewhat fails to provide an acceptable correspondence with the experiment. See [2] and [3] for further analysis on this subject.

Contrary to the semi-submersible test, results from the breaking dam simulations did not favor a VOF method. Both VOF methods yield similar results in terms of impact

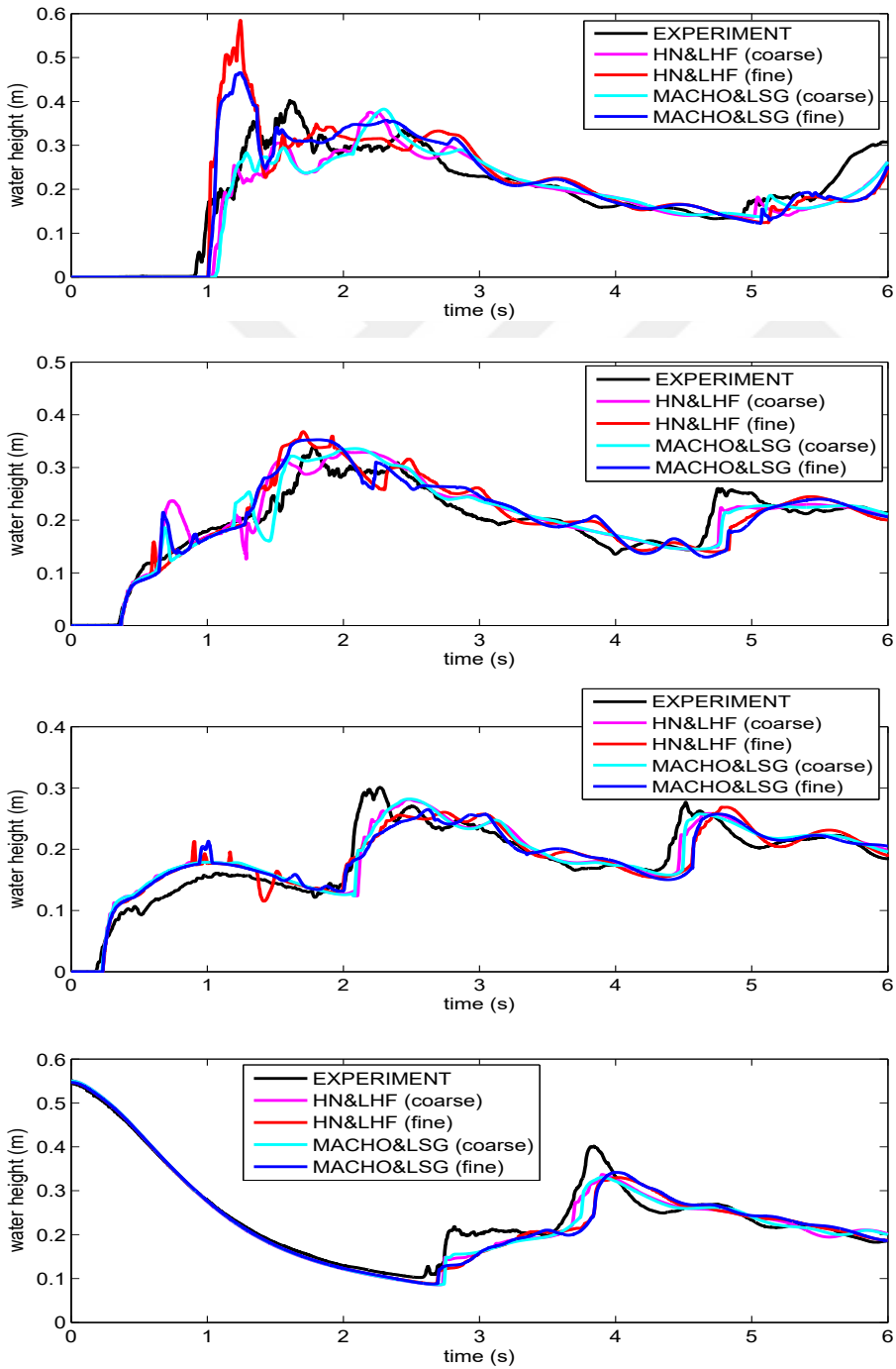


Figure 5.19: Time traces of water height from four water height probes in the tank: from top to bottom H1, H2, H3 and H4.

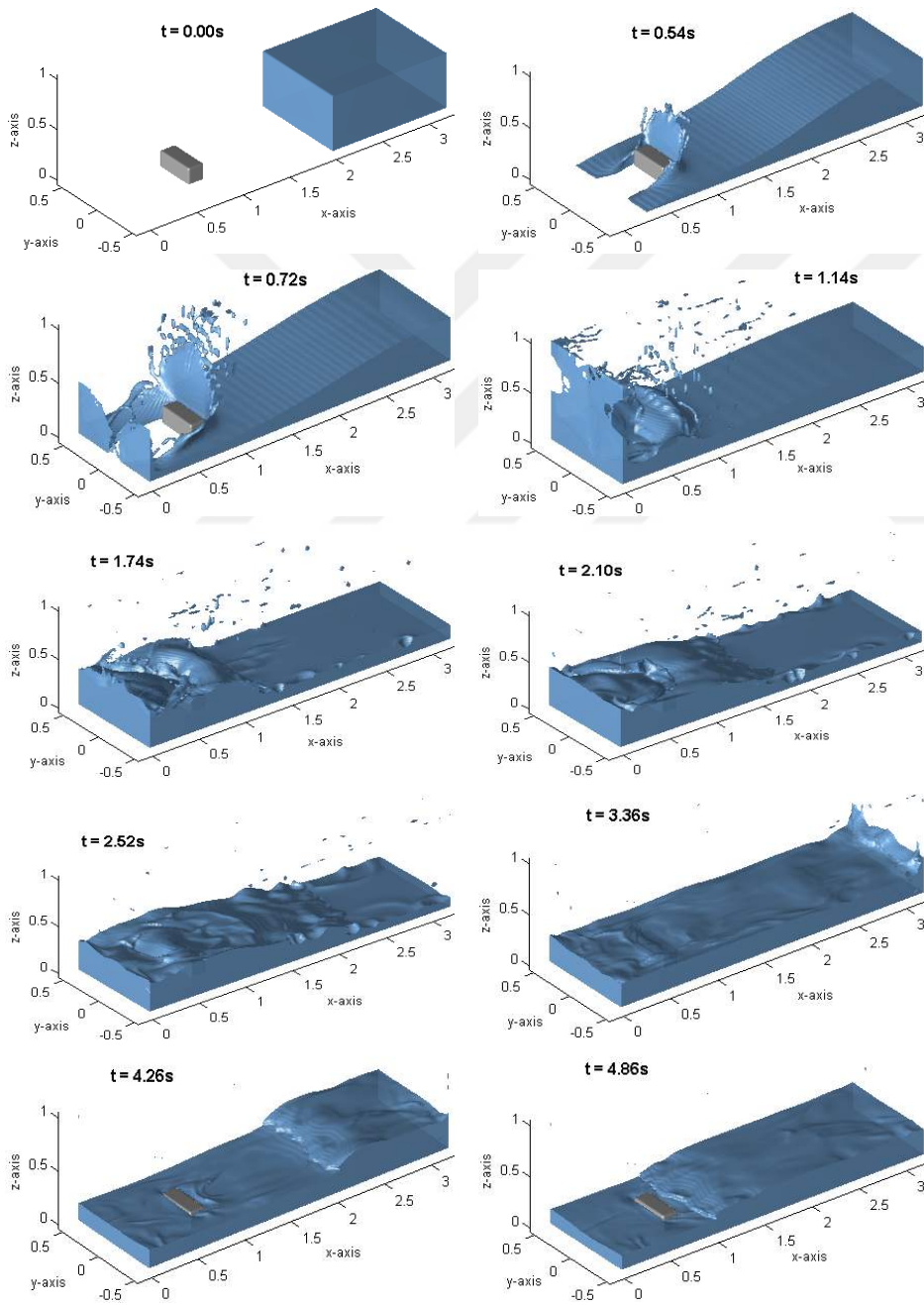


Figure 5.20: Snapshots of the critical instances in the breaking dam test. MACHO&LSG is used in this simulation on a grid with $236 \times 76 \times 68$ cells in the x , y and z directions, respectively.

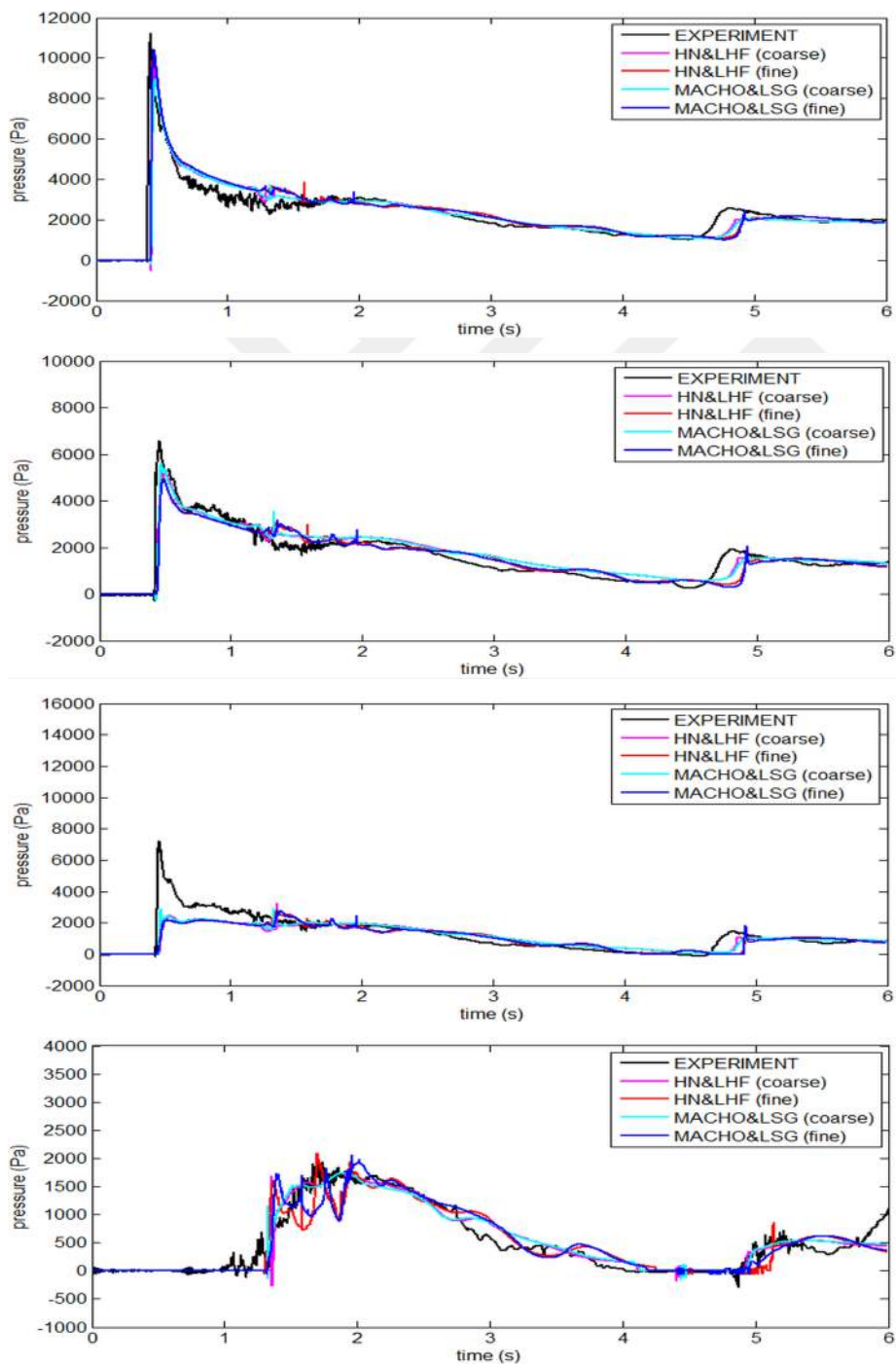


Figure 5.21: Time traces of pressure from four pressure transducers on the box: from top to bottom P1, P3, P4 and P6.

pressure and water height. In a future work, the breaking dam test can be further studied by considering other experimental setups.

REFERENCES

- [1] L. Skjelbreia and J. Hendrickson, *Fifth order gravity wave theory*, in *Proceedings of the 7th Conf. of Coastal Engineering, The Hague (The Netherlands)* (1960) pp. 184–196.
- [2] R. Wemmenhove, *Numerical simulation of two-phase flow in offshore environments*, Ph.D. thesis, University of Groningen (2008).
- [3] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, and B. Buchner, *A Volume-of-Fluid based simulation method for wave impact problems*, *J. Comput. Phys.* **206**, 363 (2005).

6

SUMMARY AND CONCLUSIONS

THIS thesis presented the outcome of a 4-year doctorate study on two popular topics in CFD simulation of interfacial and free surface flows: free surface modeling and transparent boundary conditions. Numerical methods that address these topics were implemented in the CFD simulation tool ComFLOW. This program is mainly designed for complex free surface flows in the offshore industry. Therefore, modeling the free surface in the most accurate, robust and efficient manner constitutes one of the fundamental features of the overall program. This was one of the main subjects of this thesis. Typical problems in offshore industry involve interactions between the environment and numerous man-made structures. This generally takes place in a compact physical domain around the offshore structures and involves free surface water waves whose motion and behavior can sometimes become quite chaotic. In order to carry out CFD simulations for wave-structure interaction generation and absorption of these waves in computational domain is significant. Moreover and not surprisingly, the need for generating and absorbing waves over the same boundary simultaneously (this gives the boundary condition the attribute of transparency) arises, which makes the problem even more interesting and difficult. A chapter in this thesis was dedicated to this problem, and results demonstrating the performance of such a boundary condition were included in the chapter as well. Below we will summarize our findings and conclusions stemming from the results in this work.

6.1. FREE SURFACE MODELING

Spurious energy dissipation in numerical wave simulations due to numerical diffusion is a widely-known problem, which manifests itself as loss of wave height accompanied by phase shift and can have several causes. The numerical diffusion can be due to the discretization method used for the governing equations applied inside the flow domain, but can also be due to numerical modeling of the moving free surface and the boundary conditions applied at the free surface. Among these the free surface treatment is an important factor, and its effect on wave dissipation was studied extensively. A technique

to model an arbitrary 3D free surface efficiently is the volume-of-fluid (VOF) method. Throughout years researchers have developed numerous techniques within the VOF framework, which can be classified into three categories: donor-acceptor formulation, high resolution differencing schemes and line techniques (explicit geometrical reconstruction of the free surface). Line techniques can also be divided into three classes: piecewise constant (SLIC), linear (PLIC) and parabolic (PPIC) interface calculation. The VOF implementation in ComFLOW before this work was the donor-acceptor method introduced by [Hirt and Nichols](#) [1] and a local height function (LHF) [2] to overcome the bottlenecks which originate from this VOF technique such as violation of mass conservation and spurious flotsam and jetsam [3]. Although implementation of the LHF reduced the amount of flotsam and jetsam considerably and improved mass conservation [4], the overall VOF method produced a significant wave damping. After carefully analyzing strengths and weaknesses of many VOF techniques in the literature, explicit geometrical reconstruction and advection of the free surface was the approach that was eventually followed. For reconstruction and advection several schemes were studied and afterwards implemented, and a combination of a reconstruction and an advection scheme was referred to as a PLIC-VOF module. Performance of the PLIC-VOF modules were investigated through a number of test problems including simulation of propagating Rienecker-Fenton waves [5] where strong nonlinear behavior was dominant in the flow. In all the test cases PLIC-VOF modules outperformed the previous VOF implementation by a clear margin without increasing computational costs in a disproportionate manner. Especially in simulations of Rienecker-Fenton waves a significant reduction in errors in terms of loss of wave height as well as phase shift was observed.

In addition to these tests, implementation of the PLIC-VOF modules were also validated in two tests which have strong relevance to practical situations in life: wave impact on a semi-submersible and a breaking dam. Both tests involve many complex physical and numerical phenomena, such as generation of steep water waves, generating and absorbing boundary conditions, free-surface modeling, turbulence effects, air entrapment and fluid-structure interaction. Among these phenomena, rather violent fluid flow which implies significant amount of topological change in fluid configuration in both cases, was useful to test especially robustness as well as accuracy of the VOF schemes. Numerical results were compared to the measurements in terms of free surface elevation and wave impact pressure at various locations. In the semi-submersible test, especially at elevated locations where waves first run up and then impact on the structure, PLIC-VOF modules showed better correspondence with the measurements in terms of pressure. This was due to more accurate approximation of the wave run-up with PLIC-VOF modules. In other locations where hydrostatic effects were dominant in the flow, VOF schemes produced similar results. The breaking dam test did not favor a VOF method. Nonetheless, we were able to test the limits of robustness of the VOF schemes in a very complex flow, and the agreement between the measurements and simulations was good.

It is important to note that reconstructing the free surface by simply adopting a piecewise linear representation does not always result in a high-order approximation since the accuracy of the reconstruction depends critically on the calculation of the free surface normal. In that regard, researchers have continuously been working on develop-

ing more accurate PLIC schemes, which indicates possibility and room for further improvements. Among these attempts Moment-of-fluid (MOF) scheme by [Dyadechko and Shashkov](#) [6] deserves attention. Furthermore, the numerical framework of ComFLOW allowed us to adopt a direction splitting strategy for the advection of the free surface, which may also be the focal point of further research. Comparing the effects of unsplit and directionally split advection schemes on wave damping and wave impact on structures can be an interesting exploration.

6.2. GENERATION AND ABSORPTION OF FREE SURFACE WATER WAVES

Numerical simulations of wave phenomena necessarily have to be carried out in a limited computational domain. This implies that incoming waves should be prescribed properly, and the outgoing waves should leave the domain without causing reflections. Sometimes both tasks need to be performed over the same boundary simultaneously. In this thesis we present a method building upon a strategy addressing one aspect of this problem: generating and absorbing long-crested irregular waves propagating perpendicular to the boundary [7]. The Sommerfeld condition [8] formed the backbone of this strategy, which is perfectly absorbing for a single wave component with one phase speed. The phase speed is substituted by an approximation of the linear dispersion relation in terms of the wave number. Subsequently the wave number is replaced by the second-order derivative of the velocity potential in the vertical direction. The innovative aspect of this design is that the phase speed featuring in the boundary condition, which is usually unknown due to dispersion, is now determined from the local flow solution in an implicit fashion. In addition, the resulting condition causes small reflection for a range of wave components. Stability of the boundary condition was also investigated, and several criteria were formulated afterwards. Allowing incoming and outgoing waves to travel simultaneously over the same boundary is achieved by following the idea of [Carpenter](#) [9]. In that respect the incoming wave is prescribed with a non-zero right-hand side which this time applies the same operators to the incoming wave potential.

In this work we preferred Higdon operators [10, 11] as the starting point of the derivation. The reason for this decision lies in the fact that the first-order Higdon operator is non-reflecting for a wave impinging on the boundary with an angle of incidence, while the Sommerfeld condition provides full absorption only at the normal incidence. This allowed us to generate and absorb oblique regular and long-crested irregular waves. However, looking out at the sea, we rarely observe waves traveling in a specific direction. On the contrary, the sea surface is rather chaotic with waves of different characteristics propagating in different directions. Disregarding this very dominant physical behavior may limit the accuracy of synthesizing random seas in reality. Therefore, the boundary condition needed a further modification such that generating and absorbing short-crested waves become possible. This is achieved by considering the second-order Higdon operator which has much more favorable directional properties than the first-order variant. Into the second-order operator we then incorporated the improvements concerning dispersive effects that were explained previously. After performing stability analysis on the conditions several criteria were obtained. In the end, all these steps

resulted in a versatile and accurate boundary condition for various sea states that may possess both dispersive and directional features.

During numerical discretization we shifted our attention towards how to incorporate the boundary condition into the pressure Poisson equation and how to solve the consequent expression. The design of the pressure Poisson equation and the numerical framework of the overall code allowed us to accomplish this task in a number of steps without causing an incommensurate difficulty in solving the system of equations. A Krylov subspace solver, BiCGSTAB, with ILU(p) factorization as preconditioner was capable enough to solve the linear system.

The performance of the boundary conditions was assessed in a number of tests by comparing results to reference solutions. In the numerical experiments carried out here, reference solutions were obtained by solving the problem in a much larger domain with the same discretization in space and time. First, the capability of the boundary conditions to account for only dispersive effects of the waves was investigated. In the second test only directional effects of the waves were the focal point. Following these tests a short-crested wave is simulated in the computational domain, which combined both effects in the same test. In the final test circular concentric waves are generated with an oscillating solid sphere impacting on water surface in a tank. In all the tests the boundary condition that was presented in this work showed a good performance demonstrating its capability of accounting for dispersive and directional effects of free surface water waves. In the validation study where fifth-order Stokes waves with large amplitudes were impacting on a semi-submersible, we also observed a good performance from the boundary condition.

The design of the boundary condition is quite flexible to accommodate further extensions. Including the effects of flow currents, for example, can diversify the strengths of the condition. It is important to note that linear potential theory was utilized in several points through derivation. It is highly recommended to explore other possibilities for building non-linear effects in the design. This can permit us to work with rather extreme sea states.

6.3. EPILOGUE

In this thesis we focused our attention on two main topics: (1) Reducing energy dissipation in numerical simulations of free surface water waves through implementation of a PLIC-VOF method, and (2) Accounting for dispersive and directional effects of free surface water waves by means of developing and applying a high-order generating and absorbing boundary condition. The methods were implemented in the CFD computation tool ComFLOW, which has many users from industry and the fundamental objective of simulating complex free surface flows. Therefore, while accomplishing these tasks, we had to make sure that certain requirements which the fabric of ComFLOW program instantaneously had laid before us were also satisfied: accuracy, efficiency, stability, generality and robustness. Anybody who is one way or another involved in numerical analysis knows that satisfying these requirements when developing and/or implementing a numerical technique can be a challenge. In the end the methods explained throughout this thesis have shown to be able to address their corresponding problems while meeting these criteria to a satisfying extent. And anybody who is one way or another involved

in numerical analysis also knows that there is always room for improvement. In that respect, several ideas were briefly mentioned based on our humble experience. As much as the author of this thesis enjoyed the time spent on this endeavor, he would also like to stay in touch with the ComFLOW community to witness and contribute to the journey of this program in the future.

REFERENCES

- [1] C. Hirt and B. Nichols, *Volume of fluid (VOF) method for the dynamics of free boundaries*, J. Comput. Phys. **39**, 201 (1981).
- [2] J. Gerrits, *Dynamics of liquid-filled spacecraft*, Ph.D. thesis, University of Groningen (2001).
- [3] M. Rudman, *Volume-tracking methods for interfacial flow calculations*, Int. J. Numer. Meth. Fluids **24**, 671 (1997).
- [4] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, and B. Buchner, *A Volume-of-Fluid based simulation method for wave impact problems*, J. Comput. Phys. **206**, 363 (2005).
- [5] M. Rienecker and J. Fenton, *A Fourier approximation method for steady water-waves*, J. Fluid Mech. **104**, 119 (1981).
- [6] V. Dyadechko and M. Shashkov, *Moment-of-fluid interface reconstruction*, Tech. Rep. LA-UR-05-7571, Los Alamos National Laboratory (2005).
- [7] P. Wellens, *Wave simulation in truncated domains for offshore applications*, Ph.D. thesis, Delft University of Technology (2012).
- [8] A. Sommerfeld, *Partial differential equations in physics* (Academic Press, 1949).
- [9] K. Carpenter, *Note on the paper 'Radiation conditions for the lateral boundaries of limited-area numerical models' by M.J. Miller, A.J. Thorpe*, Quart. J. R. Meteorol. Soc. **108**, 717 (1982).
- [10] R. Higdon, *Absorbing boundary conditions for difference approximations to the multi-dimensional wave equation*, Math. Comput. **47**, 437 (1986).
- [11] R. Higdon, *Numerical absorbing boundary conditions for the wave equation*, Math. Comput. **49**, 65 (1987).

A

FORMULAE FOR PLANE CONSTANT AND CUT VOLUME

IN a typical volume of fluid (VOF) method, for a given discrete volume fraction field in a computational domain, the interface normal in a cell is calculated using the data from the surrounding cells. Following this step, the planar interface within the cell is located in such a way that local volume conservation is satisfied. Finding the position of the planar interface for a given interface normal is generally referred to as the inverse problem. After this process, the amount of fluid volume to be advected through each control volume face is computed. For this purpose, the fluid volume truncated by the planar interface within each donating flux region is obtained. This procedure is generally referred to as the forward or direct problem. For both the forward and inverse problem, researchers use various algorithms. [Scardovelli and Zaleski \[1\]](#) introduced analytical expressions, thus eliminating the need for an iterative scheme and therefore generally reducing CPU time. In this report, we simply follow their idea. As they derived the analytical formulae for a unit cube, we extend their analysis to a structured orthogonal grid of hexahedral cells.

Consider a cell with dimensions Δx_1 , Δx_2 , Δx_3 in a Cartesian coordinate system (x_1, x_2, x_3) such as shown in Fig. A.1. The equation for the plane $IJKL$ can be given as

$$m_1 x_1 + m_2 x_2 + m_3 x_3 = \alpha \quad (\text{A.1})$$

where $\mathbf{m} = (m_1, m_2, m_3)$ is the normal vector to the plane and α is the plane constant. Here, the components of the normal vector are assumed to be positive and the normal vector is pointing towards the vertex G . In this configuration, Gueyffier *et al* [2] showed that the cut volume $ABFEIJKL$ illustrated in color below the plane can be calculated by

the equation

$$V = \frac{1}{6m_1m_2m_3} \left[\alpha^3 - \sum_{i=1}^3 H(\alpha - m_i\Delta x_i) (\alpha - m_i\Delta x_i)^3 + \sum_{i=1}^3 H(\alpha + m_i\Delta x_i - \alpha_{\max}) (\alpha + m_i\Delta x_i - \alpha_{\max})^3 \right] \quad (\text{A.2})$$

where $\alpha_{\max} = m_1\Delta x_1 + m_2\Delta x_2 + m_3\Delta x_3$ and the Heaviside step function $H(y)$ is defined as

$$H(y) = \begin{cases} 0 & \text{for } y < 0 \\ 1 & \text{for } y > 0 \end{cases} \quad (\text{A.3})$$

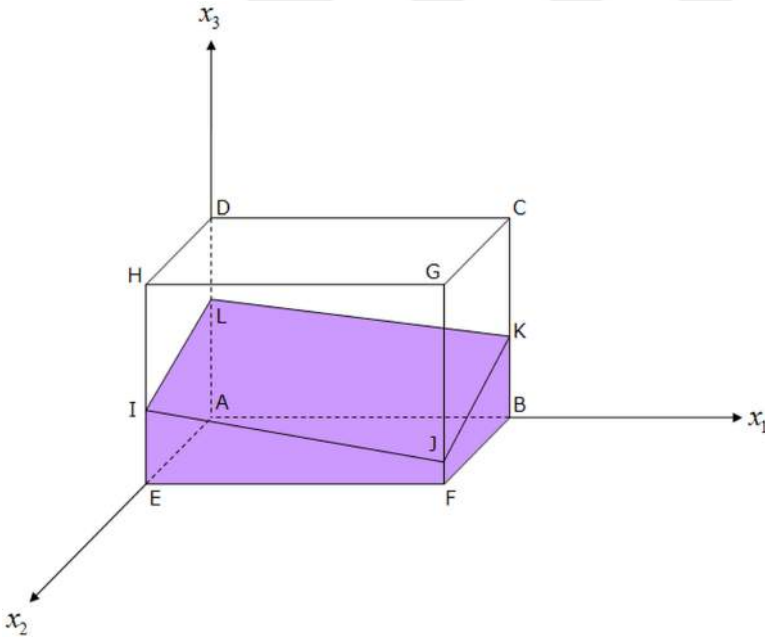


Figure A.1: A computational cell truncated by a planar interface. The cut volume is below the plane $IJKL$ and illustrated in color.

The function V is equal to zero when $\alpha = 0$ and it is equal to the volume of the cell $V_{\max} = \Delta x_1\Delta x_2\Delta x_3$ when $\alpha = \alpha_{\max}$. The expression for V changes behavior when the plane passes through one of the corners in the cell and this is characterized by the Heaviside step function in Eq. (A.2). To specify an order in which the corners in the cell are crossed, we make a simple rearrangement among the indices such that $m_1\Delta x_1 \leq m_2\Delta x_2 \leq m_3\Delta x_3$. In this case, the relation between $m_1\Delta x_1 + m_2\Delta x_2$ and $m_3\Delta x_3$ controls the order only for a certain range of α values. Outside this range, the order is invariant with respect to whether $m_1\Delta x_1 + m_2\Delta x_2$ is smaller or larger than $m_3\Delta x_3$. This range will be taken into account while deriving analytical expressions from Eq. (A.2). First, the explicit relations for the forward problem will be derived: given the cell dimensions, $\mathbf{m}$

and α , find the cut volume V . Then, the inverse problem will be considered: given the cell dimensions, $\mathbf{m}$ and the cut volume V , find the plane constant α .

A.1. FORWARD PROBLEM

For the forward problem in 3D, the cut volume V varies in the range $0 \leq V \leq V_{\max}$ while α varies in the range $0 \leq \alpha \leq \alpha_{\max}$. The expressions for the volume of interest are given below.

1. for $0 \leq \alpha < m_1 \Delta x_1$:

$$V = \frac{\alpha^3}{6m_1 m_2 m_3} \quad (\text{A.4})$$

2. for $m_1 \Delta x_1 \leq \alpha < m_2 \Delta x_2$:

$$V = \frac{3\alpha \Delta x_1 (\alpha - m_1 \Delta x_1) + m_1^2 \Delta x_1^3}{6m_2 m_3} \quad (\text{A.5})$$

3. for $m_2 \Delta x_2 \leq \alpha < m$:

$$V = \frac{\alpha^2 (3m_1 \Delta x_1 + 3m_2 \Delta x_2 - \alpha) + m_1^2 \Delta x_1^2 (m_1 \Delta x_1 - 3\alpha) + m_2^2 \Delta x_2^2 (m_2 \Delta x_2 - 3\alpha)}{6m_1 m_2 m_3} \quad (\text{A.6})$$

4. For the fourth interval there are two possibilities,
one for $m = \min(m_1 \Delta x_1 + m_2 \Delta x_2, m_3 \Delta x_3) = m_3 \Delta x_3$,
and the other for $m = \min(m_1 \Delta x_1 + m_2 \Delta x_2, m_3 \Delta x_3) = m_1 \Delta x_1 + m_2 \Delta x_2$:

- (a) for $m = m_3 \Delta x_3 \leq \alpha \leq \alpha_{\max}/2$:

$$V = \frac{-2\alpha^3 - 3\alpha (m_1^2 \Delta x_1^2 + m_2^2 \Delta x_2^2 + m_3^2 \Delta x_3^2 - \alpha \alpha_{\max})}{6m_1 m_2 m_3} + \frac{m_1^3 \Delta x_1^3 + m_2^3 \Delta x_2^3 + m_3^3 \Delta x_3^3}{6m_1 m_2 m_3} \quad (\text{A.7})$$

- (b) for $m = m_1 \Delta x_1 + m_2 \Delta x_2 \leq \alpha \leq \alpha_{\max}/2$:

$$V = \frac{\Delta x_1 \Delta x_2 (2\alpha - m_1 \Delta x_1 - m_2 \Delta x_2)}{2m_3} \quad (\text{A.8})$$

5. For the fifth interval there are also two possibilities,
one for $m = \min(m_1 \Delta x_1 + m_2 \Delta x_2, m_3 \Delta x_3) = m_3 \Delta x_3$,
and the other for $m = \min(m_1 \Delta x_1 + m_2 \Delta x_2, m_3 \Delta x_3) = m_1 \Delta x_1 + m_2 \Delta x_2$:

(a) for $\alpha_{\max}/2 < \alpha \leq \alpha_{\max} - (m_1 \Delta x_1 + m_2 \Delta x_2)$:

use Eq. (A.8).

(b) for $\alpha_{\max}/2 < \alpha \leq \alpha_{\max} - m_3 \Delta x_3$:

use Eq. (A.7).

6. for $\alpha_{\max} - m < \alpha \leq \alpha_{\max} - m_2 \Delta x_2$:

$$V = \frac{-\alpha^3 + 3\alpha(2m_1 \Delta x_1 m_2 \Delta x_2 + \alpha m_3 \Delta x_3 - m_3^2 \Delta x_3^2) + m_3^3 \Delta x_3^3}{6m_1 m_2 m_3} - \frac{\Delta x_1 \Delta x_2 m_{12}}{2m_3} \quad (\text{A.9})$$

7. for $\alpha_{\max} - m_2 \Delta x_2 < \alpha \leq \alpha_{\max} - m_1 \Delta x_1$:

$$V = \frac{3m_1 \Delta x_1 (-\alpha^2 + \alpha(m_1 \Delta x_1 + 2m_{23}) - m_2 \Delta x_2 m_{12} - m_3 \Delta x_3 m_{13}) - m_1^3 \Delta x_1^3}{6m_1 m_2 m_3} \quad (\text{A.10})$$

8. for $\alpha_{\max} - m_1 \Delta x_1 < \alpha \leq \alpha_{\max}$:

$$V = \frac{(\alpha - \alpha_{\max})^3 + 6m_1 \Delta x_1 m_2 \Delta x_2 m_3 \Delta x_3}{6m_1 m_2 m_3} \quad (\text{A.11})$$

In these equations m_{12} , m_{13} , m_{23} and m are defined as

$$\begin{aligned} m_{12} &= m_1 \Delta x_1 + m_2 \Delta x_2, \\ m_{13} &= m_1 \Delta x_1 + m_3 \Delta x_3, \\ m_{23} &= m_2 \Delta x_2 + m_3 \Delta x_3, \\ m &= \min(m_{12}, m_{13}, m_{23}). \end{aligned} \quad (\text{A.12})$$

A.2. INVERSE PROBLEM

The ranges for V and α in the forward problem are also valid in the inverse problem. The analytical expressions for the inverse problem in three dimensions are given below.

1. for $0 \leq V < V_1$:

$$\alpha = \sqrt[3]{6m_1 m_2 m_3 V} \quad (\text{A.13})$$

2. for $V_1 \leq V < V_2$:

$$\alpha = \frac{1}{2} \left(m_1 \Delta x_1 + \sqrt{m_1^2 \Delta x_1^2 + \frac{8m_2 m_3}{\Delta x_1} (V - V_1)} \right) \quad (\text{A.14})$$

3. for $V_2 \leq V < V_3$:

$$P(\alpha) = -\alpha^3 + 3m_{12}\alpha^2 - 3(m_1^2\Delta x_1^2 + m_2^2\Delta x_2^2)\alpha + m_1^3\Delta x_1^3 + m_2^3\Delta x_2^3 - 6m_1m_2m_3V = 0 \quad (\text{A.15})$$

4. For the fourth interval there are two possibilities, one for $V_3 = V_{31}$ and the other for $V_3 = V_{32}$:

(a) for $V_3 = V_{31} \leq V \leq V_{\max}/2$:

$$P(\alpha) = -2\alpha^3 + 3\alpha_{\max}\alpha^2 - 3(m_1^2\Delta x_1^2 + m_2^2\Delta x_2^2 + m_3^2\Delta x_3^2)\alpha + m_1^3\Delta x_1^3 + m_2^3\Delta x_2^3 + m_3^3\Delta x_3^3 - 6m_1m_2m_3V = 0 \quad (\text{A.16})$$

(b) for $V_3 = V_{32} \leq V \leq V_{\max}/2$:

$$\alpha = \frac{m_3V}{\Delta x_1\Delta x_2} + \frac{m_1\Delta x_1 + m_2\Delta x_2}{2} \quad (\text{A.17})$$

5. For the fifth interval there are also two possibilities, one for $V_3 = V_{31}$ and the other for $V_3 = V_{32}$:

(a) for $V_{\max}/2 < V \leq V_{\max} - V_{31}$:

use Eq. (A.16).

(b) for $V_{\max}/2 < V \leq V_{\max} - V_{32}$:

use Eq. (A.17).

6. for $V_{\max} - V_3 < V \leq V_{\max} - V_2$:

$$P(\alpha) = -\alpha^3 + 3m_3\Delta x_3\alpha^2 + 3(2m_1\Delta x_1m_2\Delta x_2 - m_3^2\Delta x_3^2)\alpha + m_3^3\Delta x_3^3 - 3m_1\Delta x_1m_2\Delta x_2m_{12} - 6m_1m_2m_3V = 0 \quad (\text{A.18})$$

7. for $V_{\max} - V_2 < V \leq V_{\max} - V_1$:

$$P(\alpha) = \alpha^2 - (m_1\Delta x_1 + 2m_{23})\alpha + \frac{m_1^2\Delta x_1^2}{3} + m_2\Delta x_2m_{12} + m_3\Delta x_3m_{13} + \frac{2m_2m_3V}{\Delta x_1} = 0 \quad (\text{A.19})$$

8. for $V_{\max} - V_1 < V \leq V_{\max}$:

$$\alpha = \sqrt[3]{6m_1m_2m_3(V - \Delta x_1\Delta x_2\Delta x_3)} + \alpha_{\max} \quad (\text{A.20})$$

In these equations V_1 , V_2 , V_{31} , V_{32} and V_3 are defined as

$$\begin{aligned} V_1 &= \frac{m_1^2 \Delta x_1^3}{6m_2 m_3}, \\ V_2 &= V_1 + \frac{\Delta x_1 \Delta x_2 (m_2 \Delta x_2 - m_1 \Delta x_1)}{2m_3}, \\ V_{31} &= \frac{m_3^2 \Delta x_3^2 (3m_{12} - m_3 \Delta x_3) - 3m_3 \Delta x_3 (m_1^2 \Delta x_1^2 + m_2^2 \Delta x_2^2) + m_1^3 \Delta x_1^3 + m_2^3 \Delta x_2^3}{6m_1 m_2 m_3}, \quad (\text{A.21}) \\ V_{32} &= \frac{\Delta x_1 \Delta x_2 (m_1 \Delta x_1 + m_2 \Delta x_2)}{2m_3}, \\ V_3 &= \begin{cases} V_{31} & \text{for } m = m_3 \Delta x_3 \\ V_{32} & \text{for } m = m_{12}. \end{cases} \end{aligned}$$

In the previous relations, we notice that $P(\alpha)$ can be a linear, quadratic or cubic function. Among the three cases, finding the roots of the cubic polynomial $P(\alpha)$ is the most challenging. [Scardovelli and Zaleski \[1\]](#) explain a simple procedure for this purpose. Here, we will recall this procedure. Consider a cubic polynomial given by

$$P(\alpha) = a_3 \alpha^3 + a_2 \alpha^2 + a_1 \alpha + a_0 = 0. \quad (\text{A.22})$$

If we divide this polynomial by a_3 , we will have

$$P(\alpha) = \alpha^3 + \frac{a_2}{a_3} \alpha^2 + \frac{a_1}{a_3} \alpha + \frac{a_0}{a_3} = 0. \quad (\text{A.23})$$

Then, we let

$$\begin{aligned} p_0 &= \frac{1}{3} \left(\frac{a_1}{a_3} \right) - \frac{1}{9} \left(\frac{a_2}{a_3} \right)^2 \\ q_0 &= \frac{1}{6} \left(\frac{a_1}{a_3} \frac{a_2}{a_3} - 3 \left(\frac{a_0}{a_3} \right) \right) - \frac{1}{27} \left(\frac{a_2}{a_3} \right)^3. \end{aligned} \quad (\text{A.24})$$

Using the relations given in Eq. (A.24), we let

$$\theta = \frac{1}{3} \cos^{-1} \left(\frac{q_0}{\sqrt{-p_0^3}} \right). \quad (\text{A.25})$$

Then, the root of interest can be found by

$$\alpha = \left(\sqrt{3} \sin \theta - \cos \theta \right) \sqrt{-p_0} - \frac{1}{3} \left(\frac{a_2}{a_3} \right). \quad (\text{A.26})$$

REFERENCES

- [1] R. Scardovelli and S. Zaleski, *Analytical relations connecting linear interfaces and volume fractions in rectangular grids*, J. Comput. Phys. **164**, 228 (2000).
- [2] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, and S. Zaleski, *Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimesional flows*, J. Comput. Phys. **152**, 423 (1999).

B

COMPUTATION OF FLUX VOLUME

AFTER reconstructing the free surface, flux volumes across each cell face are computed. Since the orientation of the planar interface is arbitrary, the geometric procedure to calculate the flux volume can become highly complicated. In this report, we focus our attention on exploiting analytical expressions given in the previous chapter to calculate the flux volume across a cell face for a direction splitting advection algorithm. For the sake of simplicity, we will consider only one geometric volume configuration as the approach below can be easily used for other configurations as well.

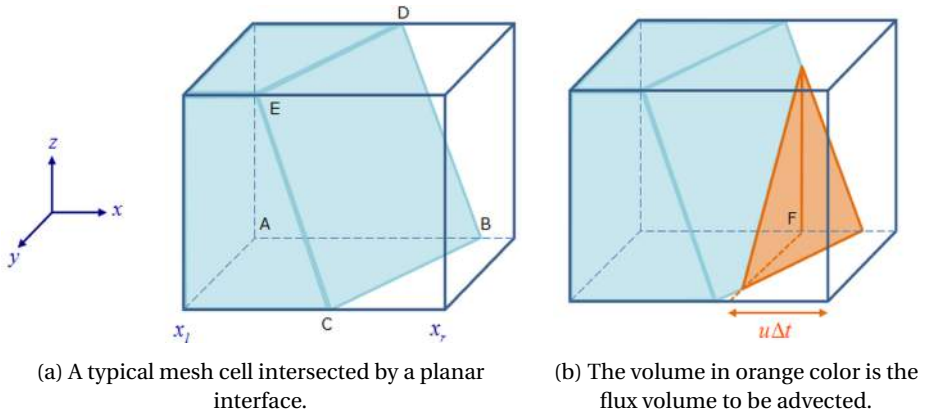


Figure B.1: Specifying the amount of flux volume in a mesh cell.

In this configuration, computation of the flux volume is demonstrated at only one cell face since the procedure is the same at the other cell faces. This face is the right cell face on the x direction in a typical mesh cell such as displayed in Fig. B.1a. The cell has dimensions Δx , Δy and Δz in the x , y and z directions, respectively. In addition, x_l denotes the position of the left cell face along the x direction and x_r denotes the position of the right cell face.

The free surface is represented by a planar interface $BCDE$ which cuts through the cell and the resulting fluid volume is illustrated in blue color in Fig. B.1a. All points on the interface satisfy

$$m_x x + m_y y + m_z z = \alpha \quad (\text{B.1})$$

where m_x , m_y and m_z are the components of the normal vector $\mathbf{m}$ and α is the plane constant. To determine the amount of fluid volume to be advected across the right cell face, the cell is also intersected by the plane $x = x_r - u\Delta t$, see Fig. B.1b. Here, u is the x -component of the velocity which is defined at the center of the right cell face and pointing outwards from the cell while Δt is the time step. The fluid volume shown in orange color and enclosed by $u\Delta t \times \Delta y \times \Delta z$ will be advected through the right cell face.

In the configuration given in Fig. B.1, the flux volume to be advected is shown in orange color. To calculate this volume, we will use the analytical expressions derived in the previous chapter. The expression for the volume stated by Eq. (A.2) is a function of the dimensions, plane normal and plane constant, for example, the fluid volume in Fig. B.1a can be computed by $V(\Delta x, \Delta y, \Delta z, \mathbf{m}, \alpha)$. Note that the function V is valid only if all components of the normal vector are positive. If one or more components of the normal are negative, the geometry can be rotated using simple mirror reflections until standard orientation is reached, i.e., all components are positive. Using this function we can compute the flux volume via one of the two methods given below.

- Since the plane constant is obtained with respect to the origin A of the cell, we need to make a coordinate transformation such that a new plane constant is computed with respect to the origin F of the fluxing region $u\Delta t \times \Delta y \times \Delta z$. If we have

$$x = (\Delta x - u\Delta t) + x', \quad (\text{B.2})$$

we can see that x' measures the distance from F . Substituting this transformation into Eq. (B.1) yields

$$m_x x' + m_y y + m_z z = \alpha' \quad (\text{B.3})$$

where $\alpha' = \alpha - (\Delta x - u\Delta t)$ is the plane constant of the interface with respect to F . We can now calculate the flux volume using

$$V_{FLUX} = V(u\Delta t, \Delta y, \Delta z, \mathbf{m}, \alpha'). \quad (\text{B.4})$$

- Alternatively we can subtract the volume illustrated in blue color in Fig. B.1b from the total fluid volume in the cell in order to find the flux volume. This procedure can be written as

$$V_{FLUX} = F^s \Delta x \Delta y \Delta z - V(\Delta x - u\Delta t, \Delta y, \Delta z, \mathbf{m}, \alpha) \quad (\text{B.5})$$

where F^s is the fluid fraction in the cell and $V(\Delta x - u\Delta t, \Delta y, \Delta z, \mathbf{m}, \alpha)$ is the fluid volume illustrated in blue color in Fig. B.1b.

We can easily observe that there may not be any fluid within $u\Delta t \times \Delta y \times \Delta z$ in some cases depending on the orientation and location of the interface. This is illustrated in Fig. B.2. If $m_x > 0$, then the existence of fluxing across the right cell face is controlled by

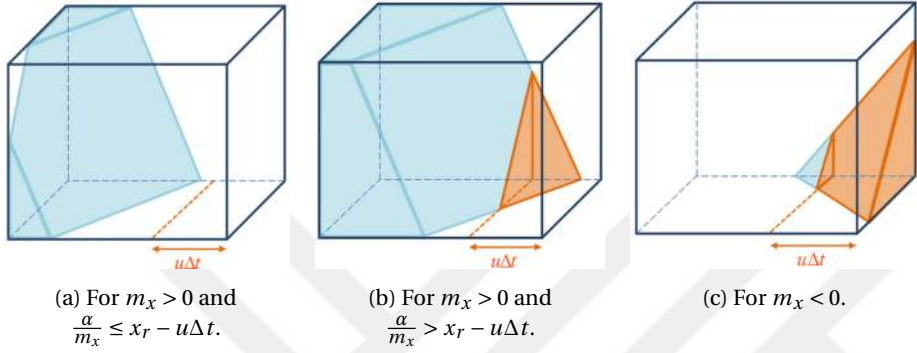


Figure B.2: Volume configurations for $m_x x + m_y y + m_z z = \alpha$ where all the components of the normal are nonzero.

the relation between the intersection point of the interface with the x -direction (α/m_x) and $\Delta x - u\Delta t$. If $m_x > 0$ and $\alpha/m_x \leq \Delta x - u\Delta t$, then there is not any fluid volume to be advected, see Fig. B.2a. If $m_x > 0$ and $\alpha/m_x > \Delta x - u\Delta t$, then the fluid volume in orange color will be transported across the right cell face, see Fig. B.2b. If $m_x < 0$ such as shown in Fig. B.2c, then there is always an amount of fluid volume to be transported. In this case we can mirror the geometry such that we have $m_x > 0$, and afterwards we can use the function V to calculate the flux volume. Eventually we arrive at the conclusion that Eq. (A.2) can safely be used to determine the flux volume irrespective of the orientation or location of the planar interface. Pseudocode for the case discussed here is given below.

Algorithm 1 Pseudocode for the flux computation for $m_x x = \alpha$

```

if  $m_x > 0$  then
  if  $\frac{\alpha}{m_x} \leq x_r - u\Delta t$  then comment: See Fig. B.2a
     $V_{FLUX} = 0$ 
  else comment: See Fig. B.2b
     $V_{FLUX} = F^s \Delta x \Delta y \Delta z - V(\Delta x - u\Delta t, \Delta y, \Delta z, \mathbf{m}, \alpha)$ 
  end if
else comment: See Fig. B.2c
   $V_{FLUX} = V(u\Delta t, \Delta y, \Delta z, \mathbf{m}, \alpha)$ 
end if

```

SUMMARY

THIS thesis presented the outcome of a four-year doctorate study on two popular topics in CFD simulation of interfacial and free surface flows: free surface modeling and absorbing boundary conditions. Numerical methods that address these topics were implemented in the CFD simulation tool ComFLOW.

Spurious energy dissipation in numerical wave simulations is a widely-known problem, which manifests itself as loss of wave height accompanied by phase shift, and can have several causes such as, the discretization method used for the governing equations applied inside the flow domain, numerical modeling of the moving free surface, and the boundary conditions applied at the free surface. Among these, the free surface treatment is an important factor, and its effect on wave dissipation was studied extensively. The VOF implementation in ComFLOW before this work was the donor-acceptor method introduced by [Hirt and Nichols](#) [1], and a local height function (LHF) [2] to overcome the bottlenecks which originate from this VOF technique such as violation of mass conservation, and spurious flotsam and jetsam [3]. Although implementation of the LHF reduced the amount of flotsam and jetsam considerably and improved mass conservation [4], the overall VOF method produced a significant amount of wave damping. After carefully analyzing strengths and weaknesses of many VOF techniques in the literature, several methods were implemented, and their designs were explained in detail. Performance of these methods were investigated through a number of test problems including simulation of propagating waves where strong nonlinear behavior was dominant in the flow. In all the test cases, the VOF methods implemented in this work outperformed the previous VOF implementation by a clear margin without increasing computational costs in a disproportionate manner. Especially in simulations of propagating waves, a significant reduction in wave damping was observed.

Numerical simulations of wave phenomena necessarily have to be carried out in a limited computational domain. This implies that incoming waves should be prescribed properly, and the outgoing waves should leave the domain without causing reflections. Sometimes both tasks need to be performed over the same boundary simultaneously. In this thesis we presented a boundary condition building upon the work in [5]. The Sommerfeld condition [6] formed the backbone of the work in [5], which is perfectly absorbing for a single wave component with one phase speed. The phase speed was substituted by an approximation of the dispersion relation from the linear theory in terms of the wave number. Subsequently the wave number was replaced by the second-order derivative of the velocity potential in the vertical direction. The innovative aspect of this design is that the phase speed featuring in the boundary condition, which is usually unknown due to dispersion, is now determined from the local flow solution in an implicit fashion. Allowing incoming and outgoing waves to travel simultaneously over the same boundary was achieved by following the idea of [Carpenter](#) [7].

In this work we preferred Higdon operators [8] as the starting point of the design.

The reason for this decision lies in the fact that the first-order Higdon operator is non-reflecting for a wave impinging on the boundary with an angle of incidence, while the Sommerfeld condition provides full absorption only at the normal incidence. This allowed us to generate and absorb oblique regular and long-crested irregular waves. However, looking out at the sea, we rarely observe waves traveling in a specific direction. On the contrary, the sea surface is rather chaotic with waves of different characteristics propagating in different directions. Therefore, the boundary condition needed a further modification such that generating and absorbing short-crested waves become possible. This was achieved by considering the second-order Higdon operator which has much more favorable directional properties than the first-order variant. Into the second-order operator we then incorporated the improvements concerning dispersive effects that were explained previously. The performance of the boundary conditions was assessed in a number of tests. In all the tests the boundary condition that was presented in this work showed a good performance demonstrating its capability of accounting for dispersive and directional effects of free surface water waves.

Finally, results from validation studies were presented. Two tests were considered: breaking dam, and wave run-up on a semi-submersible. A good agreement between the results from the numerical simulations and the experiments was observed.

REFERENCES

- [1] C. Hirt and B. Nichols, *Volume of fluid (VOF) method for the dynamics of free boundaries*, J. Comput. Phys. **39**, 201 (1981).
- [2] J. Gerrits, *Dynamics of liquid-filled spacecraft*, Ph.D. thesis, University of Groningen (2001).
- [3] M. Rudman, *Volume-tracking methods for interfacial flow calculations*, Int. J. Numer. Meth. Fluids **24**, 671 (1997).
- [4] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, and B. Buchner, *A Volume-of-Fluid based simulation method for wave impact problems*, J. Comput. Phys. **206**, 363 (2005).
- [5] P. Wellens, *Wave simulation in truncated domains for offshore applications*, Ph.D. thesis, Delft University of Technology (2012).
- [6] A. Sommerfeld, *Partial differential equations in physics* (Academic Press, 1949).
- [7] K. Carpenter, *Note on the paper 'Radiation conditions for the lateral boundaries of limited-area numerical models' by M.J. Miller, A.J. Thorpe*, Quart. J. R. Meteorol. Soc. **108**, 717 (1982).
- [8] R. Higdon, *Absorbing boundary conditions for difference approximations to the multi-dimensional wave equation*, Math. Comput. **47**, 437 (1986).

SAMENVATTING

DIT proefschrift presenteert de uitkomst van een vierjarig promotieonderzoek naar twee populaire onderwerpen in CFD simulatie van vrije wateroppervlak stromingen: vrije wateroppervlak modellering en absorberende randvoorwaarden. In de CFD simulatietool ComFLOW zijn numerieke methoden geïmplementeerd die betrekking hebben op deze onderwerpen.

Ongewenste dissipatie van golfenergie in numerieke golf simulaties is een algemeen bekend probleem, dat zich manifesteert als het verlies aan golfhoogte begeleid door een faseverschuiving. Dit kan verschillende oorzaken hebben, zoals de discretizatie methode die wordt gebruikt voor de stromingsvergelijkingen toegepast in het rekendomein, numerieke modellering van het vrij bewegend wateroppervlak, en de toegepaste randvoorwaarden bij het vrije wateroppervlak. Van deze oorzaken is de vrije wateroppervlakte behandeling een belangrijke factor en daarom is dit effect op golfdissipatie uitgebreid onderzocht. Voordat dit onderzoek begon, was de VOF implementatie in ComFLOW de donor-acceptor methode, geïntroduceerd door [Hirt and Nichols](#) [1]. Deze methode werd gecombineerd met een lokale hoogte functie (LHF) [2] om de knelpunten, afkomstig van deze VOF techniek te overbruggen, zoals het schenden van massabehoud en ongewenste *flotsam en jetsam* [3]. Hoewel de implementatie van de LHF de hoeveelheid ongewenste *flotsam en jetsam* aanzienlijk vermindert en massabehoud verbetert [4], produceerde de VOF methode een aanzienlijke golfdemping. Na een zorgvuldige analyse van sterke en zwakke punten van de vele VOF technieken in de literatuur, werden verschillende methoden toegepast, en hun ontwerpen in detail toegelicht. Prestaties van deze werkwijzen werden onderzocht door een aantal testproblemen waaronder simulatie van voortplantende golven waar sterk niet-lineair gedrag dominant voorkomt in de stroming. In alle testgevallen presteren de VOF methoden die geïmplementeerd zijn in dit onderzoek beter dan de voorgaande VOF implementatie zonder een disproporitionele toename van de rekenkosten. Vooral in simulaties van voortplantende golven werd een significante vermindering van golfdemping waargenomen.

Numerieke simulaties van golfverschijnselen worden noodzakelijkerwijs in een beperkt rekendomein uitgevoerd. Dit heeft tot gevolg dat inkomende golven goed moeten worden voorgeschreven en dat de uitgaande golven het domein moeten verlaten zonder dat reflecties optreden. Soms moeten beide taken gelijktijdig uitgevoerd worden op dezelfde rand van het domein. In dit proefschrift presenteren we een randvoorwaarde voortbouwend op het werk in [5]. De Sommerfeld randvoorwaarde [6] vormt de ruggengraat van het werk in [5], die perfect is voor het absorberen van een golfcomponent met een fasesnelheid. De fasesnelheid werd vervangen door een benadering van de dispersie relatie op basis van lineaire theorie. Vervolgens werd het golfgetal in de dispersie relatie vervangen door de tweede afgeleide van de snelheidspotential in verticale richting. Het innovatieve aspect van dit ontwerp is dat de fasesnelheid in de randvoorwaarde, die meestal onbekend is vanwege dispersie, nu bepaald wordt door de oplossing van de lo-

kale stroming op een impliciete wijze. Het toestaan van inkomende en uitgaande golven gelijktijdig op dezelfde rand, werd bereikt door het volgen van het idee van [Carpenter](#) [7].

In dit werk geven we de voorkeur aan Higdon operators [8] als uitgangspunt van het ontwerp. De reden voor deze beslissing is dat de eerste orde Higdon operator niet reflecteert wanneer de golf de rand bereikt onder een invalshoek, terwijl de Sommerfeld randvoorwaarde alleen volledige absorptie als de propagatierichting gelijk is aan de normaalrichting op de rand. Dit maakt het mogelijk om schuine regelmatige en langkammige onregelmatige golven te genereren en te absorberen. Echter, kijkend naar de zee, zien we zelden golven voortplanten in een bepaalde richting. Integendeel, het zeeoppervlak is nogal chaotisch met golven van verschillende kenmerken die zich voortplanten in verschillende richtingen. Daarom was er nog een wijziging nodig aan de randvoorwaarde die het mogelijk maakt om kortkammige golven te absorberen en te genereren. Dit werd bereikt door te de tweede orde Higdon operator te beschouwen die veel gunstigere richtingseigenschappen heeft dan de eerste orde variant. In de tweede orde operator hebben we vervolgens de verbeteringen met betrekking tot dispersieve effecten die eerder werden toegelicht opgenomen. De prestaties van de randvoorwaarden werden beoordeeld in een aantal tests. In alle tests liet de randvoorwaarde die werd gepresenteerd in dit werk een goede prestatie zien, waarmee werd gedemonstreerd dat dispersieve en richtings effecten van golven goed worden afgehandeld.

Tenslotte werden de resultaten van de validatie studies gepresenteerd. Twee tests werden beschouwd: het breken van een dam en een oplopende golf op een semi-submersible. Een is een goede overeenkomst tussen de resultaten van de numerieke simulaties en de experimenten.

ACKNOWLEDGEMENTS

In my journey towards this moment I have had help from many people. If I forget to mention any of them here, I extend my apologies. First of all I would like to thank my mother and father, Güllü and İbrahim Düz, for always remembering me in their prayers, supporting and motivating me, and doing everything they can to provide for our family. I am also grateful to my brothers, Orhan, Ali and Belya, for their support and encouragements. I can not forget my only sister, Zeliha. She has always been the light of our family. My special thanks also go for my mother- and father-in-law, Buthina and Rafe; my sisters-in-law, Reham and Dalia; and my brothers-in-law Seiph, Abdullah and Muhammed. I hope that you will always be as cheerful and full of life as you are now.

I would like to thank my promoters, Rene H.M. Huijsmans and Arthur E.P. Veldman. I do not remember the number of reports and papers that I have sent to Arthur throughout my study, but he has always been very patient with me and provided very constructive comments and ideas. I enjoyed and benefited considerably from the conversations we had during the last five years. During a project trip in Houston in 2013, I had a major health problem and had to go to the hospital. During this very difficult time Rene was with me and helped me a lot. I will always be grateful to him for being there for me in the hospital as well as for guiding and supporting me during my study.

My research associates, Mart Borsboom and Peter Wellens from Deltares, always made inspiring and fruitful critiques about my research. They were always available to help me understand the elusive topics I often encountered. I also thank my colleagues in the ComFLOW-3 project, Roel Luppés, Henri van der Heiden and Peter van der Plas, for being good friends and research partners.

After I came to the Netherlands, I had the pleasure of meeting very welcoming and friendly people. Among these, Vehbi Kuzu, has a special place as he was the first person I met in Leiden, and helped me find my very first home. I am also thankful to Hasan İnan from Fatih Foundation in Leiden, who deserves a special mention for his peaceful friendship. İsmail Kuzu, Burak Nallı, Fatih Olgun, Yasin Olgun, Zübeyr Kuzu, and Selman Tayyar are among those who deserve my sincere gratitude. I am also grateful to the friends in Rotterdam, especially Ramazan Güngören and İlhan Döne. The period that Serhan Afacan was my flatmate in Leiden was very memorable and entertaining for which I am thankful to him. I should also mention many friends from İstanbul especially Fatih Keçeci, Mesut Haskul, Muhammet Emin, and Hasan Kaya. I am also indebted to one of my closest friends, Fatih Altunbaş, for his support and prayers.

Last but not least, I would like to thank my wife, Bissan Rafe. Her unending love, encouragement and patience were the backbone of this achievement. She also read the thesis a couple of times, and gave me very constructive comments improving the quality of this document. Thank you Bissan Rafe for everything.

Bülent Düz, Lunteren, September 2015

CURRICULUM VITÆ

Bülent Düz was born in İspir, Turkey in 1982. After graduating from Açık Öğretim İmam Hatip high school, he studied Mechanical Engineering at Yıldız Technical University in İstanbul. He then started his M.Sc. in Mechanical Engineering at Boğaziçi University in İstanbul. His thesis was on the topic of fluid-structure analysis of turbomachinery blades. In 2010 he started his PhD study at Delft University of Technology. Since April 2014 he has been working as a researcher at the R&D department of the Maritime Research Institute in the Netherlands (MARIN). His research interests include free surface water waves, free surface modeling, wave impact on structures, absorbing boundary conditions, and domain decomposition methods.