A NUMERICAL STUDY OF THREE-DIMENSIONAL BREAKING WAVES AND TURBULENCE EFFECTS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Tso-Ren Wu

August 2004

A NUMERICAL STUDY OF THREE-DIMENSIONAL BREAKING WAVES AND TURBULENCE EFFECTS

Tso-Ren Wu, Ph.D.
Cornell University 2004

A three-dimensional numerical model has been developed to study wave-structure interactions. This model is based on solving the Reynolds Averaged Navier-Stokes (RANS) equations or filtered Navier-Stokes equations. As for solving the RANS equations, the Reynolds stresses in these equations are modeled by a nonlinear algebraic closure model. A nonlinear $k-\varepsilon$ model is solved to provide the information of turbulence kinetic energy, k, and the rate of the turbulence dissipation, ε. As for solving the filtered Navier-Stokes equations, because the small scale eddies are filtered by the spatial filter, and only the large eddies are solved explicitly, this method is called "Large Eddy Simulation" or LES. In this dissertation, the subgrid-scale Reynolds stress will be modeled by conventional Smagorinsky model (Smagorinsky, 1963). The finite volume two-step projection method is used to solve RANS and filtered Navier-Stokes equations. The forward time difference method is used to discretize the time derivative. The free-surface motion is tracked by using piecewise linear volume of fluid (VOF) method.

The numerical model is first validated by the solitary wave interacting with a circular cylinder. In order to have a better resolution around the circular cylinder, an unstructured mesh is used to discretize the spatial domain. The numerical results are well compared with the experimental data in terms of the time-history free surface elevations. The $k-\varepsilon$ model is then validated by the two-dimensional spilling breaker on a mild slope. The free-surface elevation and the turbulence characters are compared

with the laboratory data. Reasonable agreements are observed. Discussions are made on the spatial and temporal distributions of turbulence characteristics. The $k-\varepsilon$ model is then used to simulate a three-dimensional dam-break waves interacting with a square cylinder. The velocities and forces on the cylinder are well compared with the laboratory data. The spatial distribution of the turbulence intensity is presented and discussed.

In order to study the landslide generated waves, a moving solid algorithm is developed to describe the moving boundary condition of the slide motion on a fixed grid. The LES model is adopted to describe the three-dimensional turbulences. In total, forty five simulations are made by various slide initial elevations and weights. The numerical results are compared with the experimental data in terms of the time histories of runup and generated waves. Reasonably good agreements can be observed. The detailed discussions about complex three-dimensional flow patterns, velocity fields, free surface and shoreline deformations are presented and discussed.

Biographical Sketch

The author was born in Taipei, Taiwan on October 18, 1969. He received a Bachelor of Science degree in Soil and Water Conservation from National ChungHsing University (Taiwan) in 1992. Two years later, he received Master of Science degree in Civil and Environmental Engineering from National Central University (Taiwan), specializing in Water Resources.

In the fall of 1998, he enrolled in the Graduate School of Civil and Environmental Engineering of Cornell University to pursue his Ph.D. degree, focusing on the study of Coastal Dynamics and Computational Fluid Mechanics.

To my family

Acknowledgements

I am so thrilled that I have this opportunity to show my sincerest appreciation to my adviser, Professor Philip L.-F. Liu, for his continuous support and encouragement during my time as a PhD student at Cornell. I am most grateful for his constructive criticism of my research. His critical and rigorous attitude towards research taught me how to suspect my own ideas. I believe it will greatly benefit my future research, too. I would also like to thank my thesis committee members, Professor William Philpot of the School of Civil and Environmental Engineering and Professor Stephen D. DeGloria of the School of Crop and Soil Sciences for their valuable advice and suggestions on my course work and insightful comments on my thesis study.

The thesis has benefited from many people's works and efforts. The numerical model used in this study is modified from Truchas 1.8.4, which is developed and kindly provided by Dr. Douglas B. Kothe at Los Alamos National Laboratories. Special thanks are to Markus Bussmann, Jim M. Scilian, and Bryan Lally for assisting me to modify the Truchas source code. I would like to thank Professor C. E. Synolakis at University of Southern California for providing me with experimental landslide data by using the facility at Oregon State University. Also, I appreciate Professor Fredric Raichlen at California Institute of Technology for laboratory data analysis.

I happily thank Shih-Chun Hsiao, Patrick J. Lynett, Seung-Buhm Woo, Tom Hsu, and my fellow graduate students Qinghai Zhang, Khaled Abdullah Al-Banaa, Safak Ozkan, and Weihua Mo for the many chalkboard conversations during the meetings we have shared. I will miss the company of all the friendly fluids people. Liao Qian, In Mei Sou, Gustavo Zarruk, Evan Variano, Laurent Olivier Amoudry, Xiaoming Wang, Alexandra T. King, Dr. Jose Miguel Pasini, and Dr. Alejandro Orfila-Forster have all made my study at Cornell a pleasant time. I would like to thank Professor Inigo J. Losada Rodríguez for his kindness of checking my typos in this thesis.

Finally, I would like to thank my friends and family for their support and love. I thank my uncle Henry K. Chen for his encouragement to me; I also thank Shi-Ching Chen for all the joyful moments we have shared through the Internet. I want to express the gratitude from the bottom of my heart to my parents, His-Chang Wu and Ching-Chi Wu Liao, who have been giving me the invaluable support and infinite love in my life. With true love I thank my wife, Shu-Jen Hsu, for her love, patience, and care. She has made my years at Cornell great ones, and me a better man. I could not finish the study without her.

Table of Contents

List of Figures

Chapter 1

Introduction

In designing a coastal structure in the nearshore region, such as piles for a pier, a water intake structure or a submerged discharge pipeline, wave forces and ambient fluid conditions must be considered. In many situations, waves may be breaking and the interactions between breaking waves and structure may result in very complex three-dimensional turbulent flow fields. In the past, our knowledge of wave-structure interaction has relied heavily on laboratory experiments and empirical or semi-empirical correlations. However, it is well known that laboratory experiments suffer from constraints on the range of applicable physical parameters and scaling effects, not to mention the costs associated with careful experiments. In recent years, because of rapid advancement of computational power, several three-dimensional Navier-Stokes equation solvers have been developed and modified so that they can be used for wave hydrodynamics studies. With proper treatments on wave breaking induced turbulence and boundary conditions, these models have the potential to become useful research tools for better understanding the physical processes and engineering tools for the design of coastal structures.

In the wave-structure interaction studies, most of the cases considered active waves interacting with a passive structure. However, a structure might be a source to generate waves. The most important case is the landslide generated waves. Subaerial and submarine landslides may generate sizeable offshore and onshore propagating waves and cause significant danger due to coastal inundation with little warning time. The landslide generated waves usually are associated with huge wave forces and strong turbulence. The maximum runup height of the landslide generated wave becomes an important parameter in designing a coastal structure in the nearshore region. Therefore, there is a need to develop a numerical model which has the ability to simulate wave generation due to three-dimensional moving boundary and provide accurate estimations of the maximum runup height.

In this study, a numerical tool which is able to simulate three-dimensional complex flow situation with air-water interface as well as moving boundary problems will be presented. The accuracy of this numerical model will be examined and validated by many different test cases in terms of the turbulence characteristics, free surface elevation and velocity field. Many results that are not easy to be measured by the experiments will be presented and discussed.

1.1 Background of Wave-Structure Interaction

Breaking wave and structure interactions are important in the ocean and marine engineering. The entire interaction process includes the initiation of wave breaking, breaking, and wave interactions with surroundings (Rhee and Stern, 2002). In the last two decades, because of the advances of experimental techniques, the understanding of breaking waves has been greatly improved. The techniques of hot-film anemometer, Particle Image Velocimetry (PIV), and Laser Doppler Velocimeter (LDV) have been used to measure the velocity distribution under the breaking waves (Lin and Rockwell, 1994 and 1995; Skyner, 1996; Stive, 1980; Stive and Wind, 1982; Nadaoka and Kondo, 1982; Hattori and Aono, 1985; Mizuguchi, 1986; Nadaoka et al., 1989; Ting and Kirby, 1994, 1995, 1996). The turbulent fluctuations and turbulent kinetic energy can be also calculated from the PIV data (Chang and Liu, 1996). However, PIV approach does not provide the flow properties in the aerated region of the wave crest because of the presence of foam and bubbles (Nakagawa, 1983; Mizuguchi, 1986). Field studies suffer from the same difficulties as laboratory experiments do and are limited to site access and environmental variability.

For above-mentioned reasons, numerical studies of breaking waves have become increasingly popular. Numerical results can provide the details of the flow without the difficulties of scaling and measuring. Due to the limitation of computer speed, the early numerical simulation of breaking wave was mainly based on the depth-averaged equations. Boussinesq equations (Peregrine, 1967) and shallow water equations were usually employed. The energy dissipation due to the breaking processes was incorporated into momentum equations through simple dissipative terms (Schaffer et al., 1993; Johnson et al., 1996; Abbott et al., 1978, 1985; Zelt, 1991; Karambas and Koutitas, 1992). These approaches can save the computational efforts. However, they cannot predict the detailed characteristics of turbulent transport. Moreover, the vertical variations of velocity information are also lost due to the depth averaging processes.

Other approaches have been suggested to obtain the vertical distribution of velocities. For example, the boundary integral equation method (BIEM) based on the potential flow theory is a successful method to simulate nonlinear wave propagation up to the initiation of wave breaking. Before a wave breaks, the wave motion is essentially irrotational except within thin boundary layers on the bottom and free surface. Therefore, a potential theory is justified in simulating the flow motion (Longuet-Higgins and Cokelet, 1976 and 1978; Peregrine et al., 1980; Dommermuth et al., 1988).

The BIEM is mathematically elegant and numerically accurate. However, after wave breaking, the flow becomes highly rotational and complex. The potential flow assumption is no longer valid.

For incompressible Newtonian fluids, the flow motion can be described by the Navier-Stokes equations. Theoretically, the direct numerical simulation (DNS) (Orszag and Patterson, 1972; Rogallo, 1981) can be used to simulate breaking waves. However, due to the huge demand of computational resources, most of DNS applications focus only on the low Reynolds number flows within a small domain (Kim et al., 1987). Breaking waves are usually accompanied by the high Reynolds number and strong free surface deformation. The DNS is not feasible to solve the breaking wave problem currently or in the near feature.

Since computers are not fast enough to simulate high Reynolds turbulent flows, many researchers have developed several turbulent models to capture the turbulent characteristics within a manageable computation time. One of them is the Reynolds Averaged Navier-Stokes (RANS) equations. In the RANS equations, only the ensemble-averaged (mean) flow motion is described. The turbulence effects on the mean flow are represented by Reynolds stresses in the momentum equation. Lemos (1992) proposed a turbulence model to solve the RANS equations. In his study, the Reynolds stresses were modeled by solving balance equation for turbulence kinetic energy (k) and turbulence dissipation rate (ε). This method is called $k-\varepsilon$ turbulence model. However, Lemos did not present quantitative information on the accuracy of the predicted breaking wave characteristics. In addition, the information regarding the flow and turbulence fields was very limited. Lin and Liu (1998a, b) have recently performed a quantitative evaluation of their numerical model in the surf zone. However, due to the restriction of computational capabilities, their numerical model is restricted to 2D analysis. As mentioned at the beginning, the fluid fields of wave-structure interaction are mainly three-dimensional. Therefore, a three-dimensional numerical model with turbulence model is needed to solve the complex flow field in a 3D domain. In this study, a 3D $k-\varepsilon$ turbulence model will be presented, which will greatly improve the numerical capability of solving the turbulence flow in a 3D domain.

Another alternative to model the turbulence is the large-eddy simulation (LES) (Smagorinsy, 1963; Lilly, 1967; Deardorff, 1970 and 1974; Schumann, 1975; Li and Lin, 2001; Lin and Li, 2002). In LES, the larger three-dimensional unsteady turbulent motions are directly simulated, and the effects of the smaller-scale motions are modeled. In terms of the computational expense, LES lies between RANS models and DNS. Compared to DNS in high-Reynolds-number flows, because the explicitly representing the small-scale motions is avoided, the computational costs can be greatly reduced. Compared to RANS models, because the large-scale unsteady motions are represented explicitly, LES can be expected to be more accurate and reliable than RANS models for flows in which large-scale unsteadiness is significant (Pope, 2001). Therefore, in additional to presenting $k-\varepsilon$ turbulence model, the LES turbulence model will also be presented.

The movement of the structure might be an important source to generating waves and turbulence. This type of moving solid problem is especially important to the study of landslide generated waves. In order to have better understanding of the landslide generated waves, many models have been proposed to estimate the onshore and offshore wave heights. In the analytical study, Liu et al. (2003) derived an analytical solution to the forced long waves on a sloping beach. As for the semi-analytical studies, Striem and Miloh (1975), Murty (1979) transferred the energy from a sliding block into a solitary wave and calculated the height of the wave. Pelinovsky and Poplavsky (1996) and later Watts (1997) used force balance on a submerged solid body sliding down a slope to calculate the terminal velocity. The empirical studies are usually not general enough to be applied to different cases. The applications of analytical or semi-analytical studies are greatly limited by their assumptions and simplifications, such as the potential flow assumption and the long wave approximation.

As for the experimental studies, Watts (1998) performed a series of two-dimensional small-scale experiments with sliding blocks and granular masses but did not measure runup. Watts (2000) further expanded the wavemaker formalism to describe the general features of the water waves generated by two-dimensional underwater landslides. However, these efforts were focused on studying the two-dimensional offshore waves. In fact, experimental studies for 3D landslide generated waves are rare, since a 3D experiment requires a large domain to prevent waves reflected from the side walls.

In the numerical modeling, several numerical models simulate the rock-fall or landslide generated waves based on the depth-integrated approximation (Raney and Butler, 1975; Chiang et al., 1981; Townson and Kaya, 1988; Gozali and Hunt, 1989; Jiang and LeBlond, 1994; Grilli et al,, 2002). However, the depth-integrated approximation neglects the vertical acceleration that is very important in the wave generation zone where the water depth changes rapidly. Also, the depth-integrated approximation can not deal with a moving slide with a vertical wall. These constraints greatly limit the capability of solving the moving slide problems. In order to solve the full Navier-Stokes equations with the moving boundary problem, Heinrich (1991, 1992) developed a moving solid algorithm to solve the interfacial flow. In his study, the volume of fluid (VOF) method was adopted to track the air-water interface. A source function was added to the continuity and momentum equations to represent the moving boundary effects. Unfortunately, Heinrich's study stopped at two-dimensional landslide simulation.

The present numerical study concerns the solid landslide motion as well as the landslide generated breaking wave run-up and run-down on a slopping beach. A moving solid object algorithm similar to Heinrich's method will be developed. Forty five computational simulations with different initial elevations and specific weight of the solid slides will be conducted. Comparisons between the computational results and experimental data (Raichlen and Synolakis, 2003) for time histories of free surface fluctuations and the runup/rundown at various locations will be made.

1.2 Review of Turbulence Closure Models for RANS Equations

Turbulence model is important to simulating waves over a submerged structure. The choice of an appropriate turbulence model has a dominant influence on the success of modeling breaking waves. In the RANS equations, the most complex and complete classical turbulence model is the Reynolds stress equation model (RSM) (Launder et al., 1975). RSM is also called the second-order or second-moment closure model. This model solves seven extra equations which include six partial differential equations (PDE) for six Reynolds stress components and one PDE for the dissipation rate ε of turbulence energy. RSM is capable of representing many important mechanisms such as the anisotropy of turbulence in turbulent flows. However, it is currently not widely used because of the high cost of the computations. Many turbulence studies showed that RSM performs just as poorly as the $k-\varepsilon$ model does in some flows owing to the identical problem of modeling the ε-equation. RSM contains a few higher-order correlation terms that must be closed by closure models. Within all these higher-order correlation terms, the pressure-strain rate correlation term is the most difficult one because the proposed closure model is difficult to be verified directly by laboratory measurements. So far, at least five different closure models have been proposed based on different assumptions (Demuren and Sarkar, 1993). Unfortunately, none of them is completely satisfied with the complex flows when their modeling results are compared to the experimental data or DNS data (Demuren and Sarker, 1993). Since this modeling approach is computationally expensive, the application of this model is mainly for small scale problems.

Alternative to the RSM, the $k-\varepsilon$ model probably is the widest used and most validated turbulence model. Instead of solving seven extra equations in the RSM, $k-\varepsilon$ model solves the turbulence energy k only, which avoids the complicated closure problem in the pressure-strain rate correlation term. The basic assumption of $k-\varepsilon$ model is eddy-viscosity assumption. This assumption relates the Reynolds stresses to the local values of the averaged flow quantities. Conventionally, the linear isotropic eddy viscosity model is used for this purpose (Rodi, 1980). This linear eddy-viscosity approach to turbulence closure is extremely attractive from a computational point of view, especially in terms of numerical robustness, and has therefore enjoyed immense popularity with CFD practitioners. With the velocity and length scales determined from related differential transport equations, the approach also ensures that the fundamental mechanisms responsible for turbulence generation, dissipation and transport are accounted for.

In the past decades, many nonlinear $k-\varepsilon$ turbulence models have been developed in an attempt to improve the predictions from the classical, linear $k-\varepsilon$ model, and maintain the low computational costs. Pope (1975) first proposed a more general quadratic closure model. Speziale (1987) developed a workable quadratic model. Since then, a number of models have emerged (Yoshizawa, 1987; Rubinstein and Barton, 1990; Shih et al., 1993; Gatski and Speziale, 1993; Craft et al., 1997; Meric et al., 2001; Meric and Dick, 2002). Most models are quadratic, while those of Craft et al., Merci, and Merci and Dick, are cubic and that of Gatski and Speziale is quartic. In particular, the cubic fragments play an essential role in capturing the strong effects of curvature on the Reynolds stresses. As for a complex separation case, such as backward facing step case, researchers have found that a cubic model can predict more accurate and realistic results than linear and quadratic $k-\varepsilon$ model (Merci et al., 2001; Merci and Dick, 2002).

In this dissertation, the nonlinear $k-\varepsilon$ models will be applied to the RSM to simulate the three-dimensional wave-structure interaction problems. Numerical results will be presented in a variety of cases.

1.3 Review of Subgrid-Scale (SGS) Models for Large Eddy Simulation

The equations for LES are derived from NS equations with low-pass spatial filter applied. The flow motion is then divided into two parts: the resolved motion and the subfilter-scale motion (SFS) (Gullbrand and Chow, 2003). In the finite volume numerical implementation, the governing equations of the fluids are solved on a discrete grid. This implies that a discretization filtering operator has been applied. The discretization filtering operator is called implicit filter (Gullbrand and Chow, 2003). The scale of the flow motion smaller than the grid size is referred to the unresolved SFS motion, or subgrid-scale (SGS) motion (Zhou et al., 2001). For the SGS motion, a closure model is required to link the SGS motion effect to the resolved SFS. In most of the engineering applications, the implicit filter is the only one filter used in the LES modeling. However, the filter width, filter shape, and filter resolution are free parameters in LES. These variables might have effect to the model coefficients. Also, the numerical scheme as well as the numerical error might be significant. Lund and Kaltenbach (1995) proposed an explicit filter concept in LES modeling in the spectral domain to reduce those effects. Except the implicit filter, they apply an explicit filtering operator to the filtered governing equations first with the larger filter width than the width of implicit filter. The twice filtered results can be “reconstructed” to obtain the implicit filter results. Since the flow motion has been filtered twice, the truncation error can be reduced or eliminated (Lund and Kaltenbach, 1995; Lund, 1997). Carati et al. (2001) and Winckelmans and Jeanmart (2001) applied this explicit filtering method to their finite difference numerical models with careful distinguish between the discretization and filtering procedures. Gullbrand and Chow (2003) further proved that the explicit filtering can greatly improve the accuracy in the prediction of turbulence intensities which are important in the application of turbulent mixing prediction. However, an extra filtering and reconstruction make the computational cost higher, and the wider explicit filter width makes that it is very difficult to be applied to a domain with no-slip walls or with complex geometry. Gullbrand and Chow (2003) also found that the difference of the filtered velocities between with and without explicit filter is small. This indicates that if the turbulent mixing is not the major concern, the traditional LES that considers implicit filtering operator only is still a good choice. In current study, we shall focus on the traditional LES modeling. Hopefully in the near future, the LES with explicit filter can be applied to the complex geometry so it can be incorporated into our numerical model to improve the accuracy of turbulent intensity.

1.4 Free Surface Kinematics

In the study of wave-structure interaction, except for the turbulence issue, another important issue is the free surface kinematics. Flows with free surfaces are difficult because of the moving boundaries of the free surfaces. The position of the boundary is known only at the initial time step. The free surface position at later times has to be determined, and it is a part of the solution. To accomplish this, the free surface kinematic and dynamic boundary conditions must be used.

If the position of the free surface were known, the implementation of the dynamic boundary conditions would have not been a difficulty. However, finding the location of the surface must be done iteratively, which will greatly increase the complexity of the task. Therefore, in order to simplify the task, many efforts have been made to find the shape of the free surface efficiently.

The boundary fitting method (Lin and Li, 2002) is a mean to determine the free surface. In this method, the free surface is treated as a sharp interface. The boundary fitted grids are employed and advanced each time the free surface is moved. This method gives highly accurate results of tracking the free surface. However, due of the difficulty of tracing the breaking waves as well as the complex structures, this method is not attractive in current study.

Other methods are Marker-and-Cell (MAC) (Harlow and Welch, 1965) and Volume-of-Fluid (VOF) (Hirt and Nichols, 1981) schemes. Different from the boundary fitting method, these schemes do not define the interface as a sharp boundary. The computation is performed on a fixed grid, which extends beyond the free surface. According to the MAC scheme, the shape of the free surface is determined by computing the fraction of each near-interface cell that is partially filled. This goal can be achieved by following the motion of the massless particles introduced at the free surface at the initial time. Alternatively, the VOF scheme solves a transport equation for the fraction of the cell occupied by the liquid phase.

The MAC and VOF schemes are attractive because they can accurately treat the complex phenomena like wave breaking. However, the computing effort of MAC scheme is daunting, especially in three dimensions because in addition to solving the equations governing the fluid flow, one has to trace the motion of a large number of marker particles.

In the VOF method, along with the conservation equations for mass and momentum, one only needs to solve one additional equation for the filled fraction of each numerical cell f, so that $f = 1$ in filled cells and $f = 0$ in empty cells, and $0 < f < 1$ indicates the interface of each fluid. This method has a very successful implementation on the simulation of overturning free surfaces (Lin and Liu, 1998a,b). However, due to the first order accuracy of this VOF scheme, the interface between air and water will be reconstructed by either a horizontal or a vertical plane. This piecewise constant interface reconstruction scheme is relatively crude, and will lose the accuracy after a complex wave-breaking. Rider and Kothe (1998) provided a new algorithm for the volume tracking of interfaces. Their method approximates interfaces as piecewise linear and with second-order spatial accuracy. In this study, we will use the VOF method but adopt the piecewise linear interface reconstruction scheme to improve the solution quality and accuracy on the free surface.

1.5 Scope of Present Study

The purpose of this study is to discover the physical phenomena of wave-structure interactions. Before the numerical results are acquired, a complete numerical program will be built. The numerical program is modified from Truchas 1.8.4 which is developed by Doug Kothe, Jim Sicilian and their Telluride team members at Los Alamos National Laboratory. The original program has the ability to simulate the incompressible flows with multi-fluid interfaces. The original code solves Navier-Stokes equations by adopting projection method and finite volume discretization method. However, the original program is not designed to deal with coastal problems; there is no suitable turbulence model for simulating breaking waves. Furthermore, there are only a few options for the boundary conditions. Therefore, modifications are needed for turbulence models, boundary conditions, internal wavemaker, numerical sponge layer, and moving solid algorithm. Truchas can become a powerful tool for studying breaking wave introduced coastal problems by adding these numerical functions to it.

In this dissertation, the mathematical basis of the model will be presented in Chapter 2. In Chapter 3, the details of the numerical algorithms and implementations of the models will be given, which include finite volume method, VOF algorithm, and projection method. A few numerical tests will then be conducted to evaluate the model performance. The accuracy of the free surface tracking technique, the conservation of the mass and energy will be tested by using different model problems. Different boundary conditions such as zero pressure open boundary condition and advective open boundary condition will be discussed. Sponge layer as well as the internal wavemaker numerical tools will be presented.

In Chapter 4, the numerical implementation of $k-\varepsilon$ turbulence model will be introduced. The numerical model will then be employed to study several breaking-wave and wave-structure problems. Two study cases will be used to evaluate the model accuracy and performance. The first case will be the spilling breaker on a sloping beach. In this 2D simulation, the turbulence characteristics will be validated by comparing them with the laboratory data. In the second case, a dam-break wave interacting with a square cylinder problem will be investigated. The result will be compared to the experimental data in terms of the wave force acting on the cylinder and time-history velocities.

In Chapter 5, the landslide generated waves will be simulated. In this chapter, the moving solid algorithm will be developed to simulate the landslide motion. The simulation results will be compared with the laboratory measurements in terms of the surface elevation and most the runup height. The detailed three-dimensional spatial distribution of the velocity field as well as the inundation curve will be presented and discussed.

Chapter 2

Formulations

To study landslide generated waves and wave impacts on the structures in the coastal region, we shall first present the governing equations and boundary conditions describing the flow field with air-water interfaces. Because of wave breaking, a suitable turbulence model shall also be considered. In this chapter, two turbulence models will be discussed. All cases studied in this thesis are based on the physics introduced in this chapter.

2.1 Navier-Stokes Equations

Following the Newton's Second Law, fluid motion of incompressible Newtonian fluid can be described by the Navier-Stokes Equations (NSE) in a bounded domain Ω:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{2.1}$$

$$\frac{D(\rho\mathbf{u})}{Dt}=\frac{\partial(\rho\mathbf{u})}{\partial t}+\mathbf{u}\cdot\nabla(\rho\mathbf{u})=-\nabla p+\nabla\cdot\tilde{\tau}+\rho\mathbf{g} \tag{2.2}$$

where $\mathbf{u}$ represents velocity vector, ρ density (mass per unit volume), $\tilde{\tau}$ the stress tensor, $\mathbf{g}$ the gravity force vector, t time, and p pressure.

Equation (2.1) represents the conservation of mass, and Equation (2.2) the conservation of momentum. In this study, the fluids are assumed to be Newtonian, so that the stress tensor is a function of the molecular viscosity μ and the rate of the strain $\nabla\mathbf{u}$:

$$\tilde{\tau}=\mu\left(\nabla\mathbf{u}+\nabla^{T}\mathbf{u}\right) \tag{2.3}$$

As the density of any fluid particle is constant,

$$\frac{D\rho}{Dt}=0 \tag{2.4}$$

the continuity equation as well as the momentum equation can be expressed as:

$$\nabla\cdot\mathbf{u}=0 \tag{2.5}$$

$$\frac{\partial(\rho\mathbf{u})}{\partial t}+\nabla\cdot(\rho\mathbf{u}\mathbf{u})=-\nabla p+\nabla\cdot\tilde{\tau}+\rho\mathbf{g} \tag{2.6}$$

The above equations can be also written in the tensor format:

$$\frac{\partial u_i}{\partial x_i}=0 \tag{2.7}$$

$$\frac{\partial\rho u_i}{\partial t}+\frac{\partial\rho u_i u_j}{\partial x_j}=-\frac{\partial p}{\partial x_i}+\frac{\partial\tau_{ij}}{\partial x_j}+\rho g_i \tag{2.8}$$

where $i, j=1, 2, 3$ for three-dimensional flows, and τ_{ij} is the molecular viscous stress tensor:

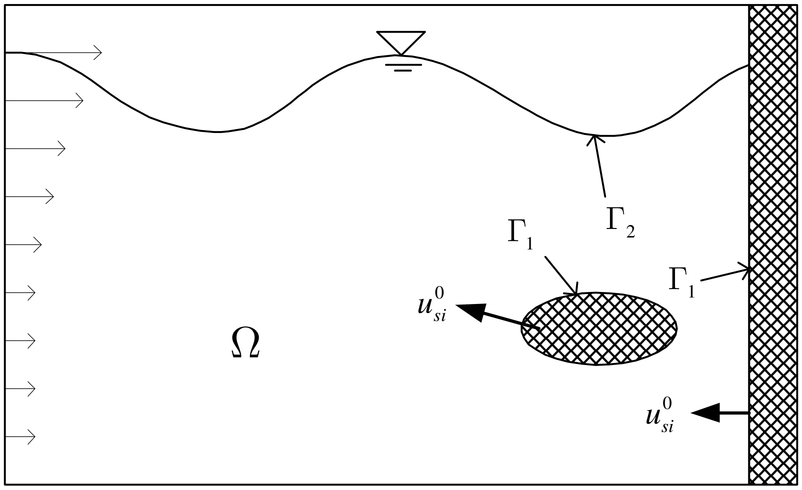


Figure 2.1: A sketch of the flow domain and boundaries. The gray parts indicate the solid material.

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{2.9}$$

Different flow conditions have different initial and boundary conditions. Along a solid boundary, Γ_1, (Figure 2.1) the fluid has the same velocity as that of the boundary does, u_{si}. Thus the boundary condition is:

$$u_i = u_{si} \tag{2.10}$$

Another type of boundary is the free surface, Γ_2, which is the interface between air and water. Along the free surface, both dynamic and kinematic boundary conditions are needed. The dynamic boundary condition describes the stresses acting on the free surface, and the kinematic boundary condition describes the movement of the free surface. As for the dynamic boundary conditions, without considering the surface tension, the normal and tangential stress components must be continuous and can be expressed as:

$$\left.\begin{aligned} -p+2\mu\frac{\partial u_n}{\partial n} &= S_n \\ \mu\left(\frac{\partial u_{T_k}}{\partial n}+\frac{\partial u_n}{\partial T_k}\right) &= S_{T_k} \end{aligned}\right\} \text{on } \Gamma_2 \tag{2.11}$$

where the subscripts n and T_k denote the outward free surface normal direction and two tangential directions (k =1, 2), respectively. S_n and S_{T_k} are the specific normal and tangential stress components induced by the air flow on the free surface. The kinematic boundary condition requires that the free surface be a sharp boundary separating the two fluids that allows no flow through it. The mathematical expression for the kinematic boundary may be derived from the equation that describes the surface. The surface can be expressed in terms of a mathematical expression of the form $F(\mathbf{x},t)=0$. The total derivative of the surface with respect to time would be zero on the surface:

$$\left.\frac{DF(\mathbf{x},t)}{Dt}=0\right|_{\text{on } F(\mathbf{x},t)=0 \text{ or on } \Gamma_2} \tag{2.12}$$

or

$$\frac{\partial F}{\partial t}+\mathbf{u}\cdot\nabla F=0 \qquad \text{on } F(\mathbf{x},t)=0, \text{ or on } \Gamma_2 \tag{2.13}$$

For the initial conditions, the whole flow field has to be given:

$$u_i(\mathbf{x},0)=u_i^0(\mathbf{x}) \tag{2.14}$$

Therefore, the initial pressure p^0 can be calculated from Equation (2.8) based on the initial velocity field $u_i^0(\mathbf{x})$, and the initial value of F can be represented as:

$$F(\mathbf{x},0) = F^0(\mathbf{x}) \tag{2.15}$$

The initial velocity and pressure fields have to satisfy the following conditions (see Figure 2.1 for a sketch of the fluid domain and boundaries):

$$\begin{aligned}
\frac{\partial u_i^0}{\partial x_i} &= 0, && \text{in } \Omega \\
u_i^0 &= u_{si}^0, && \text{on } \Gamma_1 \\
-p^0 + 2\mu \frac{\partial u_n^0}{\partial n} &= S_n^0, && \text{on } \Gamma_2 \\
\mu\left(\frac{\partial u_{T_k}^0}{\partial n} + \frac{\partial u_n^0}{\partial T_k}\right) &= S_{T_k}^0, && \text{on } \Gamma_2 \\
u_i^0 \frac{\partial F^0}{\partial x_i} &= 0, && \text{on } \Gamma_2
\end{aligned} \tag{2.16}$$

2.1.1 Initial and Boundary Conditions

The initial and boundary conditions described in the previous section represent the real physics. However, in practical computations, due to the limitation of computer capability, the resolution of the numerical algorithms used near the boundaries is usually too coarse to resolve the detailed flow field near the boundaries. Therefore, some assumptions and modifications have to be made so that the numerical model can produce reasonable results and reduce the computational cost.

The initial flow field u_i^0 can be specified by laboratory data or analytical solutions. For all the cases presented in this thesis, their initial flow conditions are specified with zero mean velocities and hydrostatic pressure.

The no-slip boundary condition indicates that the fluid velocity on the solid face is the same as the velocity of the solid face. Since this study will explore the physics of the moving object (landslide) problem, we shall discuss both the stationary and moving boundary conditions. The most common situation in the wave hydrodynamics study is the stationary solid wall. In this situation, the velocity on the boundary faces will be set to zero. However, adopting this boundary condition indicates that a fine resolution is required for a numerical model to resolve the viscous sub-boundary layer. If a coarse grid is used, the application of no-slip condition may result in the underestimated velocity near the boundary cells. Therefore, if the wall stress effect is not significant to the main study domain and a coarse grid is used, an alternative way is to use the free-slip boundary condition, which can be expressed as:

$$u_n = 0 \tag{2.17}$$

$$\frac{\partial u_T}{\partial n} = 0 \tag{2.18}$$

where u_n is the fluid velocity normal to the wall, u_T the fluid velocity tangential to the wall, and n the normal direction of the wall.

If the solid boundary is moving, e.g., landslide problems, the fluid boundary condition will be the Dirichlet type no-slip boundary condition. This means that we have to apply the solid movement velocity on the fluid face. However, in the numerical implementation, unless the mesh is remapping every time to fit the solid boundary, which is very expensive, the solid face rarely coincides with the fluid boundary. To conquer this, a moving solid algorithm (MSA) which combines the internal source function and partial cell treatment will be introduced in the Chapter 5. By using the moving solid algorithm, the moving boundary effects will be converted into an internal source function, and the boundary fitting problem will no longer exist.

2.1.1.1 Simplified free surface boundary dynamic condition

On the free surface, if the resolution is fine enough, the exact free surface dynamic boundary condition (Equation (2.11)) should be used and the detailed free surface boundary layer should be reproduced. However, free surface boundaries are not stationary. Therefore, on a fixed grid system, the resolution around the free surface usually is not fine enough to resolve the boundary layer. In addition, the numerical test reported by Nichols and Hirt (1971) shows that with the use of (2.11), fictitious oscillations on free surface may happen. Thus, in this study, a simplified free surface dynamic boundary condition will be adopted. The simplified free surface dynamic boundary condition assumes that the dynamic stress from the air side is small and can be neglected. So the pressure on the free surface will be set to zero:

$$p = 0, \qquad \text{on } \Gamma_2 \tag{2.19}$$

Because the stresses caused by air are neglected ($S_n = 0$, $S_{T_k} = 0$) and the pressure has been set to zero, from Equation (2.11), this simplified boundary condition implies that $\partial u_n / \partial n = 0$ and $\left(\partial u_{T_k} / \partial n + \partial u_n / \partial T_k \right) = 0$. Nichols and Hirt (1971) found that such boundary conditions produced rather accurate free surface information when the grid size is larger than the thickness of the free surface boundary layer.

2.1.1.2 Incident and radiation boundary conditions

Because of the limitation of computational resources, the incident wave boundary and advective open boundary conditions will be employed to reduce the computational domain size. By specifying the fluid velocity and mass on the lateral boundary, waves can be sent into the fluid domain through the boundary (see Figure 2.1, the left side boundary). In this study, the analytical solutions for a solitary wave (Lee et al., 1982) and Cnoidal waves (Wiegel, 1960) will be used to send waves into the fluid domains. Moreover, through the understanding of energy propagation speed, we can specify the outgoing flow properties to allow wave propagating out of the computational domain:

$$\frac{\partial \phi}{\partial t} + C_g \frac{\partial \phi}{\partial n} = 0 \tag{2.20}$$

where ϕ is a physical variable associated with waves, e.g. fluid velocities, density, … etc; C_g is the group velocity of waves. For long waves, the group velocity can be estimated by

$$C_g = \sqrt{g(h+\eta)} \tag{2.21}$$

where h is the local still water depth and η is the free surface elevation. Therefore, the application of present advective open boundary condition will be limited to long waves.

Equation (2.20) is also called radiation boundary condition. We shall discuss the details of the numerical algorithm in the next chapter.

2.1.2 Turbulence Modeling

Most flows encountered in the coastal engineering problems are turbulent. Turbulent flows are characterized by randomly fluctuating velocity fields. The scale of these fluctuations can be small and the frequency can be high. It is usually too computationally expensive to simulate the detailed turbulence fluctuations directly in practical engineering calculations. Therefore, instead of solving the details of the fluctuations, the exact governing equations can be averaged or filtered to remove those small scale motions. Solving the resulting modified equations becomes computationally less expensive. However, the modified equations contain additional unknown variables that need to be modeled in order to achieve “closure”.

The Reynolds-averaged Navier-Stokes (RANS) equations represent transport equations for the mean flow quantities only. The turbulence of all scales must be modeled. The approach of permitting a solution for the mean flow variables greatly reduces the computational effort. The Reynolds-averaged approach is generally adopted for practical engineering calculations, and uses models such as $k-\varepsilon$ turbulence model.

The Large-eddy-simulation (LES) provides an alternative approach in which the large eddies are computed in a time-dependent simulation that uses a set of “filtered” equations. In finite volume method, the filter size is usually taken as the mesh size. Like Reynolds averaging, the filtering process creates additional unknown terms that must be modeled in order to achieve closure. Compared to the RANS equations, the attraction of LES is that, by modeling less of turbulence (or solving more), the error induced by the turbulence model will also be reduced. Also, it is easier to find a “universal” model for the small scales which tend to be more isotropic than the large eddies.

In the following chapters, both RANS and LES will be presented. $k-\varepsilon$ and Smagorinsky models are selected to be the closure models of RANS and LES respectively.

2.2 Reynolds Averaged Navier-Stokes Equations

In the Reynolds averaging, the solution variables in the exact Navier-Stokes equations are decomposed into the mean (ensemble averaged) and fluctuating components. In this study, $\langle \ \rangle$ denotes the mean (ensemble averaged) quantities, and the prime " ' " represents the turbulent fluctuations. The velocity components are:

$$u_i = \langle u_i \rangle + u_i^{'} \tag{2.22}$$

where $\langle u_i \rangle$ and u_i are the ensemble mean and exact (instantaneous) velocity components. ($i = 1,2,3$). Likewise, the pressure components are:

$$p = \langle p \rangle + p^{'} \tag{2.23}$$

Assuming the turbulent fluctuations are random, and fluid particle density is constant, we have $\langle u_i^{'} \rangle = \langle p^{'} \rangle = 0$. By substituting (2.22), (2.23) into Equation (2.1), (2.5) and (2.6), and taking the ensemble average of the resulting equations, we obtain the governing equations for the mean flow field:

$$\frac{\partial \langle u_i \rangle}{\partial x_i} = 0 \tag{2.24}$$

$$\begin{aligned} &\frac{\partial \rho \langle u_i \rangle}{\partial t} + \frac{\partial \rho \langle u_i \rangle \langle u_j \rangle}{\partial x_j} \\ &\qquad = -\frac{\partial \langle p \rangle}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial \langle u_i \rangle}{\partial x_j} + \frac{\partial \langle u_j \rangle}{\partial x_i}\right)\right] + \frac{\partial}{\partial x_j}\left(-\rho \langle u_i^{'} u_j^{'} \rangle\right) + \rho g_i \end{aligned} \tag{2.25}$$

Equation (2.24) and (2.25) are called "Reynolds-averaged Navier-Stokes" (RANS) equations. They have the similar form as the Navier-Stokes equations. An additional term, $-\rho \langle u_i^{'} u_j^{'} \rangle$, represents the effects of turbulence. These Reynolds stresses, $-\rho \langle u_i^{'} u_j^{'} \rangle$, must be modeled in order to "close" Equation (2.25). One simple and natural way is to assume that the effect of turbulence can be represented by the enhanced viscosity. This leads to the eddy-viscosity assumption for the Reynolds stress:

$$-\rho\left\langle u_i^{'}u_j^{'}\right\rangle = \mu_t\left(\frac{\partial\langle u_i\rangle}{\partial x_j}+\frac{\partial\langle u_j\rangle}{\partial x_i}\right)-\frac{2}{3}\rho\delta_{ij}k \tag{2.26}$$

where μ_t is the turbulent viscosity (also called the eddy viscosity), k is the turbulence kinetic energy (TKE), and δ_{ij} is the Kronecker delta ($\delta_{ij}=1$ if $i=j$; and $\delta_{ij}=0$ if $i\neq j$):

$$k=\frac{1}{2}\left\langle u_i^{'}u_i^{'}\right\rangle \tag{2.27}$$

The last term in Equation (2.26) is required to guarantee that both sides of the equation remain correctly when the normal components are considered. Although the eddy-viscosity assumption, or say the "hypothesis", is not always true, it is easy to implement and can provide reasonably good results for many flows.

The mean-momentum equation with the turbulent-viscosity hypothesis (i.e., Equation (2.26) becomes:

$$\begin{aligned}&\frac{\partial\rho\langle u_i\rangle}{\partial t}+\frac{\partial\rho\langle u_i\rangle\langle u_j\rangle}{\partial x_j}\\&\quad=-\frac{\partial}{\partial x_i}\left(\langle p\rangle+\frac{2}{3}\rho k\right)+\frac{\partial}{\partial x_j}\left[\mu_{eff}\left(\frac{\partial\langle u_i\rangle}{\partial x_j}+\frac{\partial\langle u_j\rangle}{\partial x_i}\right)\right]+\rho g_i\end{aligned} \tag{2.28}$$

where $\mu_{eff}(\mathbf{x},t)=\mu+\mu_t(\mathbf{x},t)$ is the effective viscosity. Again, this equation has the same form as the Navier-Stokes equations with u, μ, and p being replaced by $\langle u\rangle$, μ_{eff}, and $\langle p\rangle+\frac{2}{3}\rho k$ respectively.

Turbulence is characterized by two parameters: the TKE k, or a velocity scale $\vartheta=\sqrt{2k}$, and a length scale ℓ. The dimensional analysis shows that:

$$\mu_t=C_\mu\rho\vartheta\ell \tag{2.29}$$

where C_μ is a dimensionless constant.

The governing equation for turbulence kinetic energy can be derived as follows (Tennekes and Lumley, 1972): First, one multiplies Navier-Stokes equations (2.6) by the fluctuating velocity components and sums all the resulting equations. Applying the same procedure to the RANS equation (2.28) and subtracting these two resulting equations, we obtain the balance equation for turbulence kinetic energy k (Tennekes and Lumley, 1972):

$$\begin{aligned} \frac{\partial \rho k}{\partial t}+\frac{\partial \rho \langle u_j \rangle k}{\partial x_j} &= \frac{\partial}{\partial x_j}\left(\mu \frac{\partial k}{\partial x_j}\right)-\frac{\partial}{\partial x_j}\left(\frac{\rho}{2}\langle u'_j u'_i u'_i \rangle+\langle p' u'_j \rangle\right) \\ &\quad -\rho \langle u'_i u'_j \rangle \frac{\partial \langle u_i \rangle}{\partial x_j}-\mu \left\langle \frac{\partial u'_i}{\partial x_k}\frac{\partial u'_i}{\partial x_k}\right\rangle \end{aligned} \tag{2.30}$$

In words, for the balance of the TKE, we have (Tennekes and Lumley, 1972):

The time rate of change of k *+ the advective transport of* k *= the diffusive transport of* k *by viscosity + (Transport of* k *by Reynolds stress + Transport of* k *by pressure) + Turbulence production – Rate of dissipation of* k*.*

The terms on the left-hand side of this equation and the first term on the right-hand side need no modeling. The last term represents the product of the density and the dissipation ε, the rate at which turbulence energy is converted into internal energy. The equation for the dissipation will be given in a later section.

The second term on the right-hand side represents turbulent diffusion of the kinetic energy. It is almost always modeled by a gradient diffusion assumption (Ferziger and Peric, 2002):

$$-\left(\frac{\rho}{2}\langle u'_j u'_i u'_i \rangle+\langle p' u'_j \rangle\right) \approx \frac{\mu_t}{\sigma_k}\frac{\partial k}{\partial x_j} \tag{2.31}$$

where σ_k is a turbulent Prandtl number whose value is approximately unity.

The third term on the right-hand side of Equation (2.30) is the rate of production of turbulence kinetic energy of the mean flow, which transfers the kinetic energy from the mean flow to the turbulence. Based on the eddy-viscosity hypothesis, this term can be written as:

$$P_k = -\left\langle u_i' u_j' \right\rangle \frac{\partial \langle u_i \rangle}{\partial x_j} \approx \frac{\mu_t}{\rho} \left(\frac{\partial \langle u_i \rangle}{\partial x_j} + \frac{\partial \langle u_j \rangle}{\partial x_i} \right) \frac{\partial \langle u_i \rangle}{\partial x_j} \tag{2.32}$$

Since the right-hand side of Equation (2.30) can be calculated from quantities that will be computed, the development of the turbulence kinetic energy equation is complete.

As mentioned above, another equation is required to determine the length scale of the turbulence. The relation of dissipation ε, k and ℓ will be discussed in the next section.

2.2.1 The $k-\varepsilon$ model

The viscous dissipation term in Equation (2.30) can be expressed as:

$$\begin{aligned} -\mu \left\langle \frac{\partial u_i'}{\partial x_k} \frac{\partial u_i'}{\partial x_k} \right\rangle &= -2\mu \left\langle e_{ij}' \cdot e_{ij}' \right\rangle \\ &= -2\mu \left(\left\langle e_{11}'^2 \right\rangle + \left\langle e_{22}'^2 \right\rangle + \left\langle e_{33}'^2 \right\rangle + 2\left\langle e_{12}'^2 \right\rangle + 2\left\langle e_{13}'^2 \right\rangle + 2\left\langle e_{23}'^2 \right\rangle \right) \end{aligned} \tag{2.33}$$

in which

$$e_{ij}' = \frac{1}{2} \left(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i} \right) \tag{2.34}$$

The dissipation of turbulence kinetic energy is caused by work done by the smallest eddies against viscous stresses. The rate of dissipation per unit mass is then denoted by:

$$\varepsilon = 2\nu \left\langle e_{ij}' \cdot e_{ij}' \right\rangle \tag{2.35}$$

By the equilibrium assumption, the rates of production and destruction of turbulence are in near-balance (Tennekes and Lumley, 1972). The relation of dissipation ε, k and ℓ is (Tennekes and Lumley, 1972):

$$\varepsilon \approx \frac{k^{3/2}}{\ell} \tag{2.36}$$

The idea is that, at high Reynolds numbers, the turbulence kinetic energy cascades from the largest scales to the smallest scales (Kolmogorov, 1941). Equation (2.36) is based on an estimation of the inertial energy transfer.

It is possible to develop a balance equation for the viscous dissipation ε (Bradshaw et al., 1981). However, the exact ε equation contains many unknown and unmeasurable terms. Therefore, we shall not derive it here. The standard or the most commonly used equation for ε is (Launder and Spalding, 1974; Rodi, 1980):

$$\frac{\partial \rho \varepsilon}{\partial t} + \frac{\partial \rho \langle u_j \rangle \varepsilon}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right) + \frac{\varepsilon}{k}\left(C_{\varepsilon 1}\rho P_k - C_{\varepsilon 2}\rho\varepsilon\right) \tag{2.37}$$

in which σ_ε, $C_{\varepsilon 1}$, and $C_{\varepsilon 2}$ are empirical coefficients.

The equation of k can be obtained by summarizing Equation (2.30), (2.31), and (2.32):

$$\frac{\partial \rho k}{\partial t} + \frac{\partial \rho \langle u_j \rangle k}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right) + \rho P_k - \rho\varepsilon \tag{2.38}$$

In this model, the eddy viscosity is expressed as:

$$\mu_t = \rho C_\mu \sqrt{k}\,\ell = \rho C_\mu \frac{k^2}{\varepsilon} \tag{2.39}$$

The model based on Equation (2.37)~(2.39) is called the $k-\varepsilon$ model (Launder and Spalding, 1974). This model has been widely used and contains five empirical constants C_μ, σ_k, σ_ε, $C_{\varepsilon 1}$, and $C_{\varepsilon 2}$. The standard $k-\varepsilon$ model employs values for the constants that are calibrated by comprehensive data fitting for a wide range of turbulent flows (Rodi, 1980):

$$C_\mu = 0.09\,;\ \sigma_k = 1.00\,;\ \sigma_\varepsilon = 1.30\,;\ C_{\varepsilon 1} = 1.44\,;\ C_{\varepsilon 2} = 1.92 \tag{2.40}$$

An extended Boussinesq relationship is applied to compute the Reynolds stresses with the $k-\varepsilon$ model:

$$\begin{aligned} -\rho\left\langle u_i' u_j' \right\rangle &= \mu_t \left(\frac{\partial \langle u_i \rangle}{\partial x_j} + \frac{\partial \langle u_j \rangle}{\partial x_i} \right) - \frac{2}{3}\rho k \delta_{ij} \\ &= 2\mu_t E_{ij} - \frac{2}{3}\rho k \delta_{ij} \end{aligned} \tag{2.41}$$

where E_{ij} is the mean flow strain rate and defined as follow:

$$E_{ij} \equiv \frac{1}{2}\left(\frac{\partial \langle u_i \rangle}{\partial x_j} + \frac{\partial \langle u_j \rangle}{\partial x_i} \right) \tag{2.42}$$

Equation (2.41) is the so-called linear isotropic eddy viscosity model. Equation (2.41) shows that this form has an extra term on the right-hand side which involves δ_{ij}, the Kronecker delta. The term is to make the formula applicable to the normal Reynolds stresses in which $i = j$. Based on the continuity, the sum of all the normal stress is zero:

$$\begin{aligned} 2\mu_t E_{ii} &= 2\mu_t \left(\frac{\partial \langle u_1 \rangle}{\partial x_1} + \frac{\partial \langle u_2 \rangle}{\partial x_2} + \frac{\partial \langle u_3 \rangle}{\partial x_3} \right) \\ &= 2\mu_t \nabla \cdot \langle \mathbf{u} \rangle = 0 \end{aligned} \tag{2.43}$$

However, in any flow the sum of the normal stress $-\rho\left(\left\langle u'^2 \right\rangle + \left\langle v'^2 \right\rangle + \left\langle w'^2 \right\rangle\right)$ is equal to minus twice the turbulence kinetic energy per unit volume $(-2\rho k)$. The equal third allocated to each normal stress component to ensure that their sum always has a physically correct value.

Because the linear isotropic eddy viscosity assumption utilized in Equation (2.41) may not represent the correct physics for anisotropic turbulence in complex turbulent flows, Pope (1975) proposed a more general nonlinear algebraic Reynolds stress closure model. In this model, the Reynolds stresses are the function not only the linear terms of the strain rate of the mean flow but also the higher-order terms. Shih et al. (1996) proposed a set of coefficients for all quadratic terms and calculated by using the turbulent flow over a step. Lin and Liu (1998a,b) adopted the similar approach and proposed a set of empirical coefficients for studying the breaking waves. The nonlinear eddy viscosity model as well as the coefficients proposed by Lin and Liu (1998a,b) will be adopted in this study:

$$\begin{aligned}
\left\langle u_i' u_j' \right\rangle = & -C_d \frac{k^2}{\varepsilon}\left(\frac{\partial\langle u_i\rangle}{\partial x_j}+\frac{\partial\langle u_j\rangle}{\partial x_i}\right)+\frac{2}{3}k\delta_{ij} \\
& -\frac{k^3}{\varepsilon^2}\left\{ C_1\left(\frac{\partial\langle u_i\rangle}{\partial x_l}\frac{\partial\langle u_l\rangle}{\partial x_j}+\frac{\partial\langle u_j\rangle}{\partial x_l}\frac{\partial\langle u_l\rangle}{\partial x_i}-\frac{2}{3}\frac{\partial\langle u_l\rangle}{\partial x_k}\frac{\partial\langle u_k\rangle}{\partial x_l}\delta_{ij}\right)\right. \\
& \qquad C_2\left(\frac{\partial\langle u_i\rangle}{\partial x_k}\frac{\partial\langle u_j\rangle}{\partial x_k}-\frac{1}{3}\frac{\partial\langle u_l\rangle}{\partial x_k}\frac{\partial\langle u_l\rangle}{\partial x_k}\delta_{ij}\right) \\
& \qquad \left. C_3\left(\frac{\partial\langle u_k\rangle}{\partial x_i}\frac{\partial\langle u_k\rangle}{\partial x_j}-\frac{1}{3}\frac{\partial\langle u_l\rangle}{\partial x_k}\frac{\partial\langle u_l\rangle}{\partial x_k}\delta_{ij}\right)\right\}
\end{aligned} \tag{2.44}$$

Where C_1, C_2 and C_3 are empirical coefficients. The values of these coefficients from Lin and Liu (1998a,b) are:

$$C_1 = 0.0054,\ C_2 = -0.0171,\ C_3 = 0.0027 \tag{2.45}$$

However, Lin and Liu (1998a,b) found that under the extreme complex flow conditions, the model using constant C_d, C_1, C_2 and C_3 may predict unphysical situations, such as negative turbulence velocity or unbounded Reynolds stress components. In order to enforce the correct physics in complex flows, certain realizability requirements are necessary. The coefficients thus have been modified in the following ways to satisfy the realizability requirements:

$$C_d = \frac{2}{3}\left(\frac{1}{7.4 + 2S_{\max}}\right), \; C_1 = \frac{1}{185.2 + 3D_{\max}^2}$$
$$C_2 = -\frac{1}{58.5 + 2D_{\max}^2}, \; C_3 = \frac{1}{370.4 + 3D_{\max}^2} \tag{2.46}$$

where

$$S_{\max} = \frac{k}{\varepsilon}\max\left[\left|\frac{\partial\langle u_i\rangle}{\partial x_i}\right|(\text{indices not summed})\right],$$
$$D_{\max} = \frac{k}{\varepsilon}\max\left[\left|\frac{\partial\langle u_i\rangle}{\partial x_j}\right|\right] \tag{2.47}$$

The modification will ensure the non-negativity of turbulence velocity and bounded Reynolds stress.

This nonlinear model has been successfully utilized to simulate breaking waves on a sloping beach (Lin and Liu, 1998a,b) and will be used in this study.

2.2.2 Initial Conditions of $k-\varepsilon$ Model

In most cases, the initial condition of the mean flow field is specified with the zero mean velocities and hydrostatic pressure. Physically, the turbulence field should also be zero since nothing has happened at this stage. However, because the production term in k equation (2.38) is proportional to k per se, the turbulence will remain zero if it is zero initially. In order to prevent this, an "initial seeding" method is imposed to this study. The initial seeding means that we impose a very small value on k as the initial condition. The initial values for k and ε are modeled as follows:

$$k = \frac{1}{2} u_t^2 \tag{2.48}$$

where $u_t = 2\delta c_i$. δ is an arbitrary small number. Suggested by Lin (1998), δ is chosen to be $2.5e-3$ in this thesis. c_i is an empirical velocity, and c_i is determined as:

$$c_i = \mathrm{Max}\left(1, \sqrt{g \frac{H^2}{(h+H)}}\right) \tag{2.49}$$

where Max() indicates the largest component of the two arguments. H is the incident wave height on the boundary and h is the still water depth. The initial value of ε is estimated by Equation (2.39) with an empirical relationship between μ and μ_t: $\mu_t = 20\mu$.

The determinations of the initial values for k and ε are purely empirical. However, from the numerical test we have done in this study, we found that if $k \geq \varepsilon$, the solution is insensitive to the initial values of k and ε. However, if $k << \varepsilon$, the generation of k will be greatly delayed near the breaking point. The initial values of k and ε in this study are about $k \approx 20\varepsilon$, and the solutions are insensitive to the initial seeding values.

2.2.3 Boundary Conditions of $k-\varepsilon$ Model

The balance equations for k and ε require the following boundary conditions:

1. Inflow (or the inlet) boundary conditions,
2. Outflow (or the outlet) boundary conditions,

3. Free surface boundary conditions,
4. Wall boundary conditions.

We shall discuss all these boundary conditions in the following sections.

2.2.3.1 Inflow boundary conditions

The distributions of k and ε have to be given on the inflow faces. The values of k and ε can be given by the experiment data, DNS data, the analytical equations, or the initial seeding values.

2.2.3.2 Outflow boundary conditions

On the outflow faces, the same boundary conditions as those in the mean flow will be applied. For instance, an advective boundary condition will be applied to both mean and $k-\varepsilon$ fields.

2.2.3.3 Free surface boundary conditions

On the air-water interface, it is difficult to prescribe the boundary conditions for k and ε because the detailed interaction between air and water during the wave breaking is not clear. The only physical phenomenon of turbulence on the free surface we observe is that the turbulence does not transport across the free surface. This implies that the normal flux of k and ε should be zero on the free surface:

$$\frac{\partial k}{\partial n}=0, \quad \frac{\partial \varepsilon}{\partial n}=0 \tag{2.50}$$

2.2.3.4 Wall boundary conditions

Theoretically speaking, turbulent kinetic energy should be zero on the wall. However, if this boundary condition is applied, a grid size must be small enough to resolve the laminar sub-layer for the k equation. At high Reynolds number, this boundary condition is not practical. In most of engineering cases, the grid size cannot adequately resolve the turbulent boundary. Therefore, the alternative boundary conditions for k and ε equations have to be employed.

One reasonable boundary condition is the "law of the wall" boundary condition derived from the wall function. In the logarithmic region of the boundary layer, the velocity profile can be expressed as:

$$u^+ = \frac{\langle u_T \rangle}{u_\tau} = \frac{1}{\kappa} \ln\left(Ey^+\right) \tag{2.51}$$

where $\langle u_T \rangle$ is the mean velocity parallel to the wall; κ is called the von Karman constant ($\kappa = 0.41$); E is an empirical constant related to the thickness of the viscous sublayer ($E \approx 9.0$ in a boundary layer over a smooth flat plate); y^+ is a dimensionless distance from the wall:

$$y^+ = \frac{\rho u_\tau y}{\mu} \tag{2.52}$$

where y is the distance from the wall. u_τ is the friction velocity given by:

$$u_\tau = \sqrt{\frac{|\tau_w|}{\rho}} \tag{2.53}$$

where τ_w is the shear stress at the wall.

Assuming that the flow is in local equilibrium, meaning that the production and dissipation of turbulence are balanced:

$$\varepsilon = P = -\left\langle u^{'} v^{'} \right\rangle \frac{d\left\langle u \right\rangle}{dy} \tag{2.54}$$

where u and x are local spanwise components, v and y are local streamwise components.

Based on the dimensional analysis, the mean velocity gradient in this region can be expressed as:

$$\frac{d\left\langle u \right\rangle}{dy} = \frac{u_\tau}{\kappa y} \tag{2.55}$$

with Equation (2.54), we have:

$$\varepsilon = P = -\left\langle u^{'} v^{'} \right\rangle \frac{d\left\langle u \right\rangle}{dy} = \frac{u_\tau^3}{\kappa y} \tag{2.56}$$

From Equation (2.41), the eddy viscosity ν_t can be expressed as:

$$\nu_t = -\frac{\left\langle u^{'} v^{'} \right\rangle}{\frac{d\left\langle u \right\rangle}{dy}} = \kappa u_\tau y \tag{2.57}$$

Substituting Equation (2.56) and (2.57) into Equation (2.39), we have:

$$k = \frac{u_\tau^2}{\sqrt{C_d}} \tag{2.58}$$

From Equation (2.56) and (2.58), the k and ε values of the boundary cells can be obtained from the ambient computational cells. The numerical implementation will be introduced in the next chapter.

2.3 Large Eddy Simulation

As mentioned above, an alternative way to model turbulence is to “filter” out the small scale eddies and explicitly solve the large eddies. This method is called “Large Eddy Simulation” or LES.

Compared with the RANS approach, the LES does not model all the turbulent motions. Instead, only small eddies are modeled. Since large eddies are more problem-dependent, they will be calculated explicitly by solving the filtered Navier-Stokes equations. Large eddies are affected by the geometries and flow conditions. However, small eddies are less dependent on the geometry, and tend to be more isotropic and problem independent. Also, the chance of finding a universal subgrid model is much higher when only small eddies are modeled.

From the other point of view, when designing a coastal structure, an engineer should pay more attention on the large scale motions since large eddies generally contain much more energy than the small scale ones do. The momentum, mass, and energy are transferred mostly by large eddies. Therefore, it is desirable to treat the large eddies more explicitly than the small eddies. Solving only the large eddies along with the subgrid model costs much less than using DNS of the same flow.

In the following sections, the details of the governing equations for LES will be presented. A subgrid model will be introduced and applied to the numerical model.

2.3.1 Filtering

In DNS, the gird size has to be small enough to resolve the Kolmogorov scale. In LES, a low-pass filter is performed so that the filtered velocity field can be resolved on the relatively coarse grid. A filtered variable (denoted by an overbar) is defined by (Leonard, 1974):

$$\bar{\phi}(x)=\int \phi(x')G(x,x')dx' \tag{2.59}$$

where G is the filter function. In LES, G can be a Gaussian, a box (a local average), or a cutoff (used in the spectrum method) filter. Each filter has a specified filter width Δ associated with it. Roughly, eddies of size smaller than Δ are small eddies and need to be modeled. The specified filter function G satisfies the normalization condition:

$$\int G(x,x')dx'=1 \tag{2.60}$$

In this study, since the finite-volume method (FVM) is used, it implies that FVM provides the filtering operation:

$$\bar{\phi}(x)=\frac{1}{V}\int_V \phi(x')dx',\ x'\in V \tag{2.61}$$

where V is the volume of a computational cell. Therefore, the filter function G here is specified as:

$$G(x,x')=\begin{cases} 1/V & \text{for } x'\in V \\ 0 & \text{otherwise} \end{cases} \tag{2.62}$$

2.3.2 Filtered Navier-Stokes Equations

The filtered Navier-Stokes equations can be derived from applying the filter function to Equations (2.5) and (2.6). A set of equations similar to the RANS equations can be obtained. The filtered continuity and momentum equations are shown as follows:

$$\overline{\left(\frac{\partial u_i}{\partial x_i}\right)} = \frac{\partial \bar{u}_i}{\partial x_i} = 0 \tag{2.63}$$

$$\frac{\partial\left(\rho \bar{u}_i\right)}{\partial t} + \frac{\partial\left(\rho \overline{u_i u_j}\right)}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \rho g_i + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right)\right] \tag{2.64}$$

where $\bar{u}$ and $\bar{p}$ are filtered velocity and pressure fields, respectively. It is important to address here that, the filtered product $\overline{u_i u_j}$ is different from the product of the filtered velocity $\bar{u}_i \bar{u}_j$:

$$\overline{u_i u_j} \neq \bar{u}_i \bar{u}_j \tag{2.65}$$

The quantity on the left side of this equation is not easy to compute. A model to approximate the difference between these two quantities has to be introduced. The difference is the residual-stress tensor defined by

$$-\rho \tau_{ij}^{R} \equiv -\rho\left(\overline{u_i u_j} - \bar{u}_i \bar{u}_j\right) \tag{2.66}$$

In LES, τ_{ij}^{R} is also called the subgrid-scale Reynolds stress, which is analogous to the Reynolds-stress tensor:

$$-\rho\left\langle u_i' u_j'\right\rangle = -\rho\left(\left\langle u_i u_j\right\rangle - \left\langle u_i\right\rangle\left\langle u_j\right\rangle\right) \tag{2.67}$$

The residual kinetic energy is

$$k_r \equiv \frac{1}{2}\tau_{ii}^{R} \tag{2.68}$$

and the anisotropic residual-stress tensor is defined by

$$\tau_{ij}^{r} \equiv \tau_{ij}^{R} - \frac{2}{3}k_r \delta_{ij} \tag{2.69}$$

The isotropic residual stress is included in the modified filtered pressure

$$\overline{p} \equiv \overline{p} + \frac{2}{3} k_r \tag{2.70}$$

Substituting Equations (2.69) and (2.70) into Equation (2.64), the filtered momentum equation can be rewritten as:

$$\frac{\overline{D}\left(\rho \overline{u}_j\right)}{\overline{D}t} = -\frac{\partial \overline{p}}{\partial x_j} + g_i + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right)\right] - \frac{\partial \tau_{ij}^r}{\partial x_i} \tag{2.71}$$

where the substantial derivative based on the filtered velocity is:

$$\frac{\overline{D}}{\overline{D}t} \equiv \frac{\partial}{\partial t} + \overline{\mathbf{u}} \cdot \nabla \tag{2.72}$$

2.3.3 Smagorinsky Model

In Equation (2.71), the subgrid-scale Reynolds stress contains the local average of the small scale field, therefore, the subgrid model should be based on the local velocity field. The most commonly used subgrid scale model is the Smagorinsky model (Smagorinsky, 1963). It is essentially a linear eddy viscosity model:

$$\tau_{ij}^r = -\nu_t\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right) = -2\nu_t \overline{S}_{ij} \tag{2.73}$$

The Smagorinsky model relates the residual stress to the rate of filtered strain. The coefficient $\nu_t\left(\mathbf{x}, t\right)$ is the subgrid-scale eddy viscosity of the residual motions. Based on the dimensional analysis, the subgrid-scale eddy viscosity is modeled as:

$$\begin{aligned}\nu_t &= \ell_S^2 \overline{\mathbf{S}} \\ &= \left(C_S \Delta\right)^2 \overline{\mathbf{S}}\end{aligned} \tag{2.74}$$

where ℓ_S is the Smagorinsky length scale, which is a product of the Smagorinsky coefficient C_S and the filter width Δ; $\overline{\mathbf{S}}$ is the characteristic filtered rate of strain:

$$\overline{\mathbf{S}} \equiv \left(2\overline{S}_{ij}\overline{S}_{ij}\right)^{1/2} \tag{2.75}$$

Under the isotropic turbulence condition, the Smagorinsky coefficient $C_S \approx 0.2$. However, in general, C_S is not a constant; its value varies from 0.1 to 0.2 in different flows. The present simulations have used a value of 0.15. Δ is the filter width. In finite volume discretization, Δ is the grid size:

$$\Delta = \left(\Delta x_1 \times \Delta x_2 \times \Delta x_3\right)^{1/3} \tag{2.76}$$

where Δx_1, Δx_2, and Δx_3 are the three components of the grid lengths.

2.3.4 Resolution Issues

As mentioned before, proper grid spacing is required to resolve all "large" turbulent scales in the flow, or 80% of the total flow energy. Therefore, what is appropriate grid spacing Δ to resolve filtered velocity field $\overline{\mathbf{u}}$ adequately? To answer this question, it is useful to introduce a length scale ℓ_{EI} as the demarcation between the anisotropic large eddies and the isotropic small eddies. Since the Smagorinsky model is based on the isotropic eddy assumption, the grid spacing Δ should be smaller than ℓ_{EI} (Pope, 2001)

$$\Delta < \ell_{EI} \tag{2.77}$$

The eddies in the largest size range are characterized by the length scale ℓ_0. In high Reynolds number flow condition (Tennekes and Lumley, 1972),

$$\ell_{EI} \approx \frac{1}{6}\ell_0 \approx \frac{1}{6}L_{11} \tag{2.78}$$

where L_{11} is the longitudinal integral length and $L_{11} \approx \ell_0$ at high Reynolds number.

One way to justify the grid spacing Δ is that Δ should be some ratio of a lengthscale L, which is defined as a function of turbulence kinetic energy k and turbulent dissipation rate ε:

$$L \equiv \frac{k^{3/2}}{\varepsilon} \tag{2.79}$$

At high Reynolds number, L_{11}/L or l_0/L tend asymptotically to a value of 0.43 (Pope, 2001):

$$\frac{L_{11}}{L} \approx \frac{l_0}{L} = 0.43 \tag{2.80}$$

Based on a channel flow study (Baggett et., al., 1997), 80% of the total flow energy is resolved and good results can be obtained (compared to the DNS results) if the ratio between Δ and L is chosen as:

$$\frac{\Delta}{L} \approx \frac{1}{10} \tag{2.81}$$

The above-mentioned equations (2.81) and (2.80) show that a grid spacing smaller than the one-twentieth of the local largest eddy size yields a good result from LES.

$$\frac{\Delta}{\ell_0} \approx \frac{1}{20} \tag{2.82}$$

Gullbrand and Chow (2003) also suggested that the appropriate grid resolution for a simulation should consider the following two concerns: First, the grid should be fine enough to resolve important physical characteristics. Second, the grid should be fine enough to obtain a solution that is not greatly affected by the numerical errors. As addressed in Equation (2.81), a 10 x 10 grid is required in the local cross-section to resolve a local largest eddy. In this study, the local largest eddy size is obtained from the laboratory data or from the grid refinement. Determining the local largest eddy size before obtaining the final numerical solutions is important, and this can guarantee the solution quality. The numerical error, on the other hand, is more difficult to be determined especially in a non-uniform, or unstructured grid. However, in chapter 3, an estimation of the numerical error based on the uniform grid will be present, thus the numerical error can be roughly estimated.

2.3.5 Near-Wall Treatment

In the near-wall viscous sub-layer region, the largest local turbulent eddy sizes are limited by the viscous scales. From Equation (2.82), the well resolved LES requires grids nearly as fine as those used in DNS. This restriction can be applied not only to the wall-normal direction but to the streamwise direction. Near-wall resolution requirement clearly limits the application of LES to simulating high Reynolds number flows. Therefore, a modeling strategy is required to solve the practical applications. Instead of modeling every detail in the near-wall region, this study uses a wall function approach to reduce the number of computational cells. Cabot and Moin (2000) derived a set of near-wall damping functions and used them to approximate the eddy viscosity at the first cell adjacent to the wall. The eddy viscosity ν_t is obtained from a RANS type mixing-length eddy viscosity model with near-wall damping:

$$\frac{\nu_t}{\nu} = \kappa y_w^+ \left(1 - e^{-y_w^+ / A}\right)^2 \tag{2.83}$$

where $y_w^+ = y_w u_\tau / \nu$ is the distance to the wall in wall units, $\kappa = 0.41$ and $A = 19$ (Cabot and Moin, 2000). The filtered velocity field is assumed to satisfy the no-slip condition on the wall.

Chapter 3

Numerical Implementation of Navier-Stokes Equations and Model Testing

In the previous chapter, we have introduced the physics and equations relevant to wave-structure interactions and landslide problems. In order to simplify the presentation, we choose to postpone the discussion on the turbulence models in this chapter. Numerical implementations of the turbulence models will be presented in Chapter 4.

Instead, in this chapter, we will focus on the method for solving the Navier-Stokes equations. The current numerical model is modified from Truchas 1.8.4, which is developed by Doug Kothe, Jim Sicilian and their Telluride team members at Los Alamos National Laboratory. The original program solves Navier- Stokes equations by the projection method with finite volume discretization. Their program has the ability of dealing with multi-fluids by using volume-of-fluid (VOF) method. However, the original program is not designed for studying coastal problems, and no suitable turbulence model is used to simulate strong turbulence during wave-breaking processes. Therefore, we shall take the advantage of this existing Navier-Stokes equation solver, and add adequate turbulence models as well as several other features so as to address the coastal problems with complex flow conditions.

We will first discuss the finite volume discretization method. It is followed by the discussion on a spatial interpolation method of the physical quantities. The volume of fraction interface tracking method, the two-step projection method, boundary conditions, internal wavemaker, numerical sponge layer as well as a testing case will be discussed in the successive sections.

3.1 Finite Volume Discretization Method

In order to numerically solve the balance equations introduced in the previous chapter, a spatial discretization method is needed. There are three major spatial discretization methods in the field of computational fluid dynamics (CFD): finite difference method (FDM) (Richtmyer and Morton, 1967; Strikwerda, 1989), finite element method (FEM) (Zienkiewicz, 1977), and finite volume method (FVM) (Vinokur, 1989; Ferziger and Peric, 1996).

The finite difference method is probably the most widely used method, and is very simple and effective to be applied on structured orthogonal meshes. Particularly a high-order numerical algorithm can be easily developed for regular meshes. The disadvantage of FDM is that the conservation law is not enforced unless special care is taken. Moreover, its restriction to simple geometries is a significant disadvantage in dealing with complex flows (Rhie and Chow, 1983). FEM, on the other hand, can effectively be used to solve parabolic or elliptic partial differential equations (PDEs) on the complex unstructured meshes. However, accurate FEM solutions to hyperbolic PDEs are achievable only within the last decade.

On an orthogonal mesh, FVM is very similar to FDM. On the other hand, for a unstructured mesh, both FVM and FEM employ the integral (weak) form of the conservation equations. Since FVM is conservative, it is suitable for complex geometries, and hyperbolic systems can also be easily discretized as other systems. Without using any interpolation coefficient or shape function, the basic form of FVM is much "cleaner" than FEM. In this study, we will adopt FVM to discretize the flow domain.

Consider a general conservation equation for an arbitrary quantity ϕ:

$$\frac{\partial \phi}{\partial t}+\nabla\cdot(\mathbf{u}\phi)=S(\phi) \tag{3.1}$$

where S is a source term which is generally dependent upon ϕ. The finite volume method starts from the volume integral form of the PDE, e.g., for equation (3.1), we have,

$$\int\left[\frac{\partial \phi}{\partial t}+\nabla\cdot(\mathbf{u}\phi)=S(\phi)\right]dV \tag{3.2}$$

By applying the Gauss divergence theorem, Equation (3.2) can be expressed as

$$\int\frac{\partial \phi}{\partial t}dV+\oint d\mathbf{A}\cdot(\mathbf{u}\phi)=\int S(\phi)\,dV\,, \tag{3.3}$$

The physical domain then can be divided into a number of discrete volume elements ("cells") denoted by subscript i.

In Equation (3.3), the volume integral can be viewed as the integral over each control volume (the cell) and the surface integral is over the boundary surface of the cell. Equation (3.3) can then be approximated by a discrete numerical scheme:

$$\frac{\phi_i^{n+1}-\phi_i^n}{\delta t}+\frac{1}{V_i}\sum_f\left[\mathbf{A}\cdot\mathbf{u}^n\right]_f[\phi]_f^n=S_i^n\,, \tag{3.4}$$

where n indicates the n^{th} time step, V_i is the i^{th} cell volume, and subscript f indicates the cell (control volume) face.

On the LHS of Equation (3.4), the surface integral has been approximated as a sum over discrete control volume faces f with an area vector $\mathbf{A}_f$; V_i is the control volume, and ϕ_i is a local volume-averaged value,

$$\phi_i = \frac{\int \phi dV}{\int dV} = \frac{1}{V_i} \int \phi dV \,, \tag{3.5}$$

and similarly for S_i,

$$S_i = \frac{\int S dV}{\int dV} = \frac{1}{V_i} \int S dV \,, \tag{3.6}$$

In Equation (3.4), a first-order forward time scheme has been adopted. For spatially second order schemes, ϕ is assumed to vary linearly in space. Hence ϕ_i is expressed as:

$$\phi_i = \phi\left(\mathbf{x}_c\right), \tag{3.7}$$

where $\mathbf{x}_c$ is the geometric centroid of the control volume. Similarly, the face quantities $\left\langle \phi \right\rangle_f$ are given at the geometric centroid of each face.

Because of the usage of FVM, it is nature to adopt the collocated arrangement which defines all the fluid properties at the cell centroids, including density, pressure, velocities, viscosities, k, ε,…, etc. However, from Equation (3.4), the FVM requires information not only at the cell centroid but also on the face centroid. The face centroid quantities may be a scalar (e.g., density), a vector (e.g. velocity vector), or a gradient (e.g., pressure gradient or velocity gradient). Therefore, an algorithm is required to convert cell centroid data to face centroids or to evaluate the quantity gradients at each cell centroid. This algorithm is crucial because it determines the accuracy of the discretization method. In this study, we are seeking a spatial reconstruction scheme whose accuracy is up to second order and can be easily applied to 3-D, unstructured meshes. The least square linear reconstruction (LSLR) method will be used in this study.

3.2 Least Square Linear Reconstruction (LSLR) Method

Consider a discrete scalar data ϕ at cell centroid $\mathbf{x}_i$, the first-order derivatives (e.g., $\nabla_i \phi_i$) on 2-D and 3-D fully-unstructured meshes can be made by least-square method. Barth (1995) first devised the least square algorithms for the linear and quadratic reconstruction of discrete data on unstructured meshes. Second (and higher) order accuracy has been demonstrated on highly irregular (e.g., random triangular) meshes.

In LSLR approach, Taylor series expansions are used to approximate ϕ_n^{TS} from the reference cell i at cell centroid $\mathbf{x}_i$ to each immediate neighbor cell n at centroid $\mathbf{x}_n$:

$$\begin{aligned} \phi^{TS}(\mathbf{x}_n) = \phi_n^{TS} &= \phi(\mathbf{x}_i) + (\mathbf{x}_n - \mathbf{x}_i) \cdot \nabla \phi(\mathbf{x}_i) + \dots \\ &= \phi_i + (\mathbf{x}_n - \mathbf{x}_i) \cdot \nabla \phi_i + \dots \\ &\approx \phi_n \end{aligned} \tag{3.8}$$

where ϕ_n is the definition of immediate neighbor cells, being those cells share at least one vertex with the reference cell i. Assume that only the first derivative terms are retained in the expansion (that is called a linear reconstruction),

$$\phi_n^{TS} = \phi_i + \left(\mathbf{x}_n - \mathbf{x}_i\right) \cdot \nabla\phi_i \approx \phi_n \tag{3.9}$$

the sum $\left(\phi_n - \phi_n^{TS}\right)^2$ over all n immediate neighbors is then minimized in the least squares sense:

$$\min \sum_n \left(\phi_n - \phi_n^{TS}\right)^2 \Rightarrow \sum_n \left(\phi_n - \phi_n^{TS}\right) \frac{\partial\left(\phi_n - \phi_n^{TS}\right)}{\partial \nabla \phi_i} = 0 \tag{3.10}$$

The above minimization yields N (= ndim * nneighbors) equations for the unknown components of $\nabla\phi_i$. Here 'nneighbors' is the total number of immediate neighbors, and ndim is the dimension of the system, i.e., the number of unknowns for $\nabla\phi_i$. If N < ndim the system is undetermined, if N = ndim, the system is solvable, and if N > ndim, the system is overdetermined. In general, the system is overdetermined, hence a minimizing solution has to be sought according to Equation (3.10).

The least square solutions of Equation (3.10) can be found by solving a linear system:

$$\left(\mathbf{A}^T \mathbf{W} \mathbf{A}\right) \mathbf{x} = \mathbf{A}^T \mathbf{W} \mathbf{b} \tag{3.11}$$

where $\mathbf{A}$,

$$\mathbf{A} = \begin{pmatrix} \left(x_k - x_i\right) & \left(y_k - y_i\right) & \left(z_k - z_i\right) \\ \vdots & \vdots & \vdots \\ \left(x_n - x_i\right) & \left(y_n - y_i\right) & \left(z_n - z_i\right) \end{pmatrix} \tag{3.12}$$

is a dense ndim * N matrix, and $\mathbf{W}$,

$$\mathbf{W} = \begin{pmatrix} w_k & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & w_n \end{pmatrix} \tag{3.13}$$

is a diagonal N * N matrix. The diagonal entries in $\mathbf{W}$ are individual weight

$w_k = w_k^d * w_k^g$, where w_k^g is geometric weight, and w_k^d is a data-dependent weight. The geometric weight w_k^g is defined as:

$$w_k^g = \frac{1}{\left(\mathbf{x}_k - \mathbf{x}_i\right)^2} \tag{3.14}$$

The data-dependent weight w_k^d is unity by default, and is the optional WEIGHT argument used in all discrete operator procedures. Therefore its value is determined at execution time. The vector $\mathbf{b}$ (length N) is given by

$$\mathbf{b} = \begin{pmatrix} (\phi_k - \phi_i) \\ \vdots \\ (\phi_n - \phi_i) \end{pmatrix} \tag{3.15}$$

and the solution vector $\mathbf{x}$ (length ndim) is

$$\mathbf{x} = \begin{pmatrix} \nabla_x \phi_i \\ \nabla_y \phi_i \\ \nabla_z \phi_i \end{pmatrix} \tag{3.16}$$

Where ∇_x, ∇_y, and ∇_z are x, y, and z components of the spatial gradients.

After carrying out the matrix-vector and matrix-matrix multiplications in Equation (3.11), a linear system

$$\mathbf{A}'\mathbf{x} = \mathbf{b}' \tag{3.17}$$

is obtained, where $\mathbf{A}'$ is a ndim * ndim matrix,

$$\mathbf{A}' = \begin{pmatrix} \sum_n w_n \delta x_{ni} \delta x_{ni} & \sum_n w_n \delta y_{ni} \delta x_{ni} & \sum_n w_n \delta z_{ni} \delta x_{ni} \\ \sum_n w_n \delta x_{ni} \delta y_{ni} & \sum_n w_n \delta y_{ni} \delta y_{ni} & \sum_n w_n \delta z_{ni} \delta y_{ni} \\ \sum_n w_n \delta x_{ni} \delta z_{ni} & \sum_n w_n \delta y_{ni} \delta z_{ni} & \sum_n w_n \delta z_{ni} \delta z_{ni} \end{pmatrix} \tag{3.18}$$

where $\delta x_{ni} = x_n - x_i$, $\delta y_{ni} = y_n - y_i$, $\delta z_{ni} = z_n - z_i$. The vector $\mathbf{b}'$ (length ndim) is given by

$$\mathbf{b}' = \begin{pmatrix} \sum_n w_n \delta x_{ni} \delta \phi_{ni} \\ \sum_n w_n \delta y_{ni} \delta \phi_{ni} \\ \sum_n w_n \delta z_{ni} \delta \phi_{ni} \end{pmatrix} \tag{3.19}$$

where $\delta \phi_{ni} = \phi_n - \phi_i$. The resulting ndim * ndim linear system is easily solved by LU decomposition.

We notice that the reference value ϕ_i in Equation (3.8) and (3.9) is not necessity the same as that at each cell centroid, it can be anywhere inside the fluid domain. If the reference value ϕ_f is on the cell face centroid, then Equation (3.9) can be modified as:

$$\phi_n^{TS} = \phi_f + \left(\mathbf{x}_n - \mathbf{x}_f \right) \cdot \nabla \phi_f \tag{3.20}$$

However, since ϕ_f is unknown in Equation (3.20), ϕ_f will be a part of the equation. Therefore:

$$\mathbf{A} = \begin{pmatrix} \left(x_k - x_f \right) & \left(y_k - y_f \right) & \left(z_k - z_f \right) & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \left(x_n - x_f \right) & \left(y_n - y_f \right) & \left(z_n - z_f \right) & 1 \end{pmatrix}$$

$$\mathbf{x} = \begin{pmatrix} \nabla_x \phi_f \\ \nabla_y \phi_f \\ \nabla_z \phi_f \\ \phi_f \end{pmatrix}$$

$$\mathbf{b} = \begin{pmatrix} \phi_k \\ \vdots \\ \phi_n \end{pmatrix}$$

and the matrices for the linear system are:

$$\mathbf{A}'=\begin{pmatrix}\sum_n w_n\delta x_{nf}\delta x_{nf} & \sum_n w_n\delta y_{nf}\delta x_{nf} & \sum_n w_n\delta z_{nf}\delta x_{nf} & \sum_n w_n\delta x_{nf}\\ \sum_n w_n\delta x_{nf}\delta y_{nf} & \sum_n w_n\delta y_{nf}\delta y_{nf} & \sum_n w_n\delta z_{nf}\delta y_{nf} & \sum_n w_n\delta y_{nf}\\ \sum_n w_n\delta x_{nf}\delta z_{nf} & \sum_n w_n\delta y_{nf}\delta z_{nf} & \sum_n w_n\delta z_{nf}\delta z_{nf} & \sum_n w_n\delta z_{nf}\\ \sum_n w_n\delta x_{nf} & \sum_n w_n\delta y_{nf} & \sum_n w_n\delta z_{nf} & \sum_n w_n\end{pmatrix} \quad (3.21)$$

$$\mathbf{b}'=\begin{pmatrix}\sum_n w_n\delta x_{nf}\phi_n\\ \sum_n w_n\delta y_{nf}\phi_n\\ \sum_n w_n\delta z_{nf}\phi_n\\ \sum_n w_n\phi_n\end{pmatrix} \quad (3.22)$$

$$\mathbf{x}=\begin{pmatrix}\nabla_{xf}\phi_f\\ \nabla_{yf}\phi_f\\ \nabla_{zf}\phi_f\\ \phi_f\end{pmatrix} \quad (3.23)$$

A linear system similar to Equation (3.17) with (ndim+1) * (ndim+1) is easily solved by the LU decomposition.

The Least square reconstruction method is attractive for the following reasons: First, it can be applied to any mesh topology including structured and unstructured meshes in one, two, or three dimensions. What we need to provide is the cell centroid discrete data values and the physical location of those known and unknown points. Second, the discrete data points can be arbitrarily weighted by the data-dependent weights w_k^d . When performing the water wave simulation, this arbitrary weighting function is especially useful for avoiding using information from the air part if the air stresses can be neglected. Third, this method has been demonstrated to provide second (and higher) order accuracy on highly irregular meshes (Barth, 1995). This accuracy satisfies the requirement of this study.

3.3 Interface Kinematics

We have introduced the finite volume discretization method used in this research and the way to perform the interpolation and to evaluate the gradient on a 3-D unstructured mesh. Because all the cases in this study are interfacial flows, we shall introduce the interface kinematic first in this section. As mentioned in Chapter 1, volume tracking algorithms have been chosen in this study. We found that volume tracking algorithms are useful for both 2-D and 3-D structured (Rider and Kothe, 1998; Kothe et al., 1996) and unstructured (Mosso et al., 1997; Kothe et al., 1999) meshes. Rider and Kothe (1998) devised and implemented volume tracking algorithms which reconstruct piecewise linear (planar) fluid interface from discrete fluid volume data. The volume tracking algorithm, developed by Kothe et al. (1999), has its robustness and computational efficiency, having at least the second-order accuracy in space, and maintainence of local conservation and compact interface width. The detailed algorithms will be discussed in the following sections.

3.3.1 VOF Equations

Starting from the mass conservation equation (Equation (2.1)), based on the Lagrangian invariance for each of the m^{th} fluid density, we obtain

$$\frac{\partial \rho_m}{\partial t} + \nabla \cdot \left(\mathbf{u} \rho_m \right) = 0 \tag{3.24}$$

The fluid density ρ_m can be expressed as the product of volume fraction f_m and constant fluid density ρ_k^0:

$$\rho_m = \frac{M_m}{V} = \left(\frac{V_m}{V}\right) \cdot \left(\frac{M_m}{V_m}\right) = f_m \rho_m^0 \tag{3.25}$$

where M_m is the mass of fluid m, V the cell total volume, V_m the volume occupied by fluid m, and f_m the volume fraction of a cell volume V occupied by fluid m:

$$f_m = \frac{V_m}{V} \tag{3.26}$$

and a cell density is related to the volume fraction:

$$\rho = \sum_m f_m \rho_m^0 \tag{3.27}$$

Equation (3.24) then can be written as:

$$\frac{\partial\left(f_m \rho_m^0\right)}{\partial t} + \nabla \cdot \left(f_m \rho_m^0 \mathbf{u}\right) = 0 \tag{3.28}$$

Since each ρ_m^0 is constant, f_m can be expressed as:

$$\frac{\partial f_m}{\partial t} + \nabla \cdot \left(f_m \mathbf{u}\right) = 0 \tag{3.29}$$

Equation (3.29) is called the VOF equation and can be discretized based on FVM (Equation (3.4)):

$$f_m^{n+1} = f_m^n - \frac{1}{V_i} \sum_f \delta t \left[\mathbf{A} \cdot \mathbf{u}^n\right]_f \left[f_m\right]_f^n \tag{3.30}$$

The second term on the RHS of Equation (3.30) denotes the volume fluxes across the cell faces of fluid m. However, unless small δt has been used, the error could be large if the volume fluxes are estimated by the volume fraction on cell faces (Figure 3.1). The volume fluxes can be estimated more precisely based on the geometric calculation:

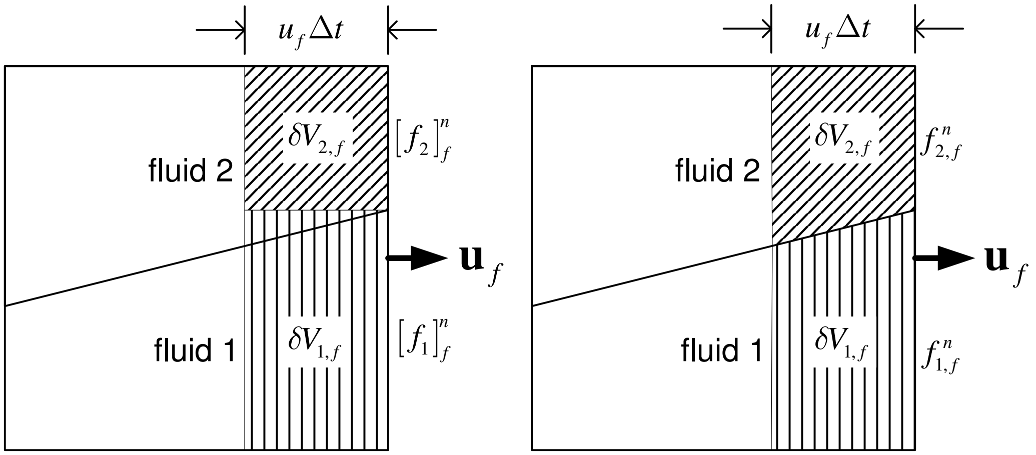


Figure 3.1: The advected mass through a cell face. The left one tends to have larger error than the right one.

$$f_m^{n+1} = f_m^n - \frac{1}{V_i}\sum_f \delta t\left[\mathbf{A}\cdot\mathbf{u}^n\right]_f f_{m,f}^n \tag{3.31}$$

where $f_{m,f}^n$ represents the volume fraction of δV_f^n associated with a particular material m through a cell face:

$$f_{m,f}^n = \frac{\delta V_{m,f}^n}{\delta V_f^n} \tag{3.32}$$

Detailed discussion will be presented in the next section.

We should point out that all the fluids inside a cell share the same velocity. In other words, one-field approximation has been invoked.

Equation (3.29) is therefore the evolution equation for the location of each fluid. The volume fractions f_m are bounded by $0 \le f_m \le 1$, where

$$f_m = \begin{cases} \text{1, inside fluid } m; \\ >0, <\text{1, at the fluid } m \text{ interface;} \\ \text{0, outside fluid } m. \end{cases} \tag{3.33}$$

Since fluid volumes are volume-filling, volume fraction must sum up to unity, $\sum_m f_m = 1$, throughout the domain. In seeking solutions to Equation (3.31), fluid volumes are marched forward in time.

3.3.2 Volume Tracking Algorithm

In order to solve the new time-step volume fractions f_m^{n+1} from Equation (3.31), we need the information of $f_{m,f}^n$. In this study, a multidimensional PLIC (piecewise linear interface calculation) (Rider and Kothe, 1998) is utilized to construct the plane in every interface cell. The plane is then used to estimate $f_{m,f}^n$.

The PLIC algorithm consists of two steps: a planar reconstruction and calculation of volume flux of each material across every cell face. As for the first step, the fluid-fluid interface within a cell is assumed to be a plane. Using the information on f_m^n, the orientations of the interfaces can be estimated. The orientations of the interface are evaluated as gradients of the f_m^n. As for the second step, the volume of each materials crossing every faces can be evaluated by multiplying the fluxes by Δt. This volume is then used to update the volume fraction in every cell and to transport other quantities in the momentum conservation equations. We shall discuss the detailed algorithm in the following sections.

3.3.2.1 Interface normal estimation

A normal vector to an interface between a material m and any number of other materials is defined as the gradient of the volume fraction data

$$\mathbf{n} = \nabla f_m \tag{3.34}$$

and can be normalized as:

$$\hat{\mathbf{n}} = \frac{\nabla f_m}{|\nabla f_m|} \tag{3.35}$$

The cell center and cell faces gradient calculation are carried out via the least square algorithms documented in chapter 3.2.

By assuming the interface geometry is piecewise linear (planar), the interface of each cell can be described as:

$$\hat{\mathbf{n}} \cdot \mathbf{x} - C_p = 0 \tag{3.36}$$

where $\mathbf{x}$ is a point anywhere on the plane and C_p is the plane constant to be determined.

3.3.2.2 Reconstructing the interface

Consider each fluid m, after having determined the interfacial normal vector $\hat{\mathbf{n}}$, the interface locations can be calculated iteratively. For each interface cell, the interface plane divides space into regions inside and outside the m^{th} fluid, depending upon the direction chosen for $\hat{\mathbf{n}}$. Based on the definition expressed in Equation (3.35), $\hat{\mathbf{n}}$ points into the fluid. Hence, $\hat{\mathbf{n}} \cdot \mathbf{x} - C_p$ (Equation (3.36)) will be positive for any point lying within the fluid, zero for any point $\mathbf{x}$ lying on the plane, and negative for any point $\mathbf{x}$ lying outside the fluid m. The fluid volume V_{tr} is the truncation volume lying within the fluid and separated by Equation (3.36). By changing C_p, V_{tr} will be changed. Our goal is to find the constant C_p iteratively (see Figure 3.2) so that a nonlinear function defined as:

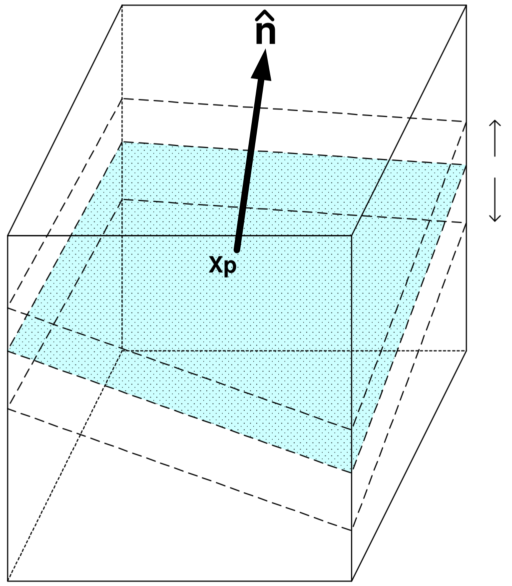


Figure 3.2: Locating the interface. The interface can be moved up and down by changing C (Equation (3.36)) and followed the direction of $\hat{\mathbf{n}}$. This is constrained by the volume conservation.

$$F\left(C_p\right) = V_{tr}\left(C_p\right) - f * V \approx 0 \tag{3.37}$$

becomes zero or smaller than a convergence criteria. Here $V_{tr}\left(C_p\right)$ is the fluid volume in the cell bounded by the interface plane (with constant of C_p), V is the cell volume and f is the volume fraction of the fluid. The algorithm of geometric calculation of the truncation volume V_{tr} can be found in Rider et al. (1998) and Kothe (1999).

When $F\left(C_p\right)$ is equal to zero or smaller than a certain tolerance, the interface plane is declared "reconstructed" in that cell. There are many root-finding algorithms, e.g., Bisection, Newton's method and Brent's method, are available to find the zero of this nonlinear function, but Rider and Kothe (1998) have found Brent's method (Press et al., 1986) to give the best results in practice.

After the interface plane is reconstructed, a geometric calculation of volume fluxes of different fluids across cell faces can be done based on the plane equation (3.36) of each cell.

Now our volume tracking algorithm template is summarized as follow:

1. Estimate the interface normal $\hat{\mathbf{n}}$ from discrete f_m data.
2. Find the plane constant C_p in Equation (3.36) by solving (3.37) to reconstruct the interfaces within the cell in a volume conservation manner.
3. A geometric calculation of the volume fluxes of different materials crosses cell faces based on $f_{m,f}^n$.
4. Update the cell center volume fraction from Equation (3.31).

3.3.3 Void Model

In our simulations, because both pressure gradients and momentum flux are often much smaller in the air regions of the fluid domain than in the water regions, a void model can be used to simplify the equations in the air regions.

Voids are used to represent regions where the Navier-Stokes equations are not solved. For example, instead of solving the Poisson pressure equation, the pressure in void regions is a constant or described by a simple ideal gas law $PM / \rho RT$, where P is the pressure of air, M is the mass, ρ is the density, R is the universal gas constant, and T is the temperature. When the density of air is described as zero ($\rho_{air} = 0$), the air becomes a void. Therefore, the air region is similar to the vacuum regions in the void treatment.

Once the void model is applied, the data-dependent weight w_k^d (Equation (3.13)) is zero in the air region:

$$\left(w_k^d\right)_{air} = 0 \tag{3.38}$$

Therefore, there is no momentum contribution from air. This implicitly implies that a stress-free assumption has been applied to the free surface.

3.4 Projection Method

The projection method has been widely used to solve the Navier-Stokes equations. This method was first proposed by Chorin (1968, 1969) and then used by Bell and coworkers to solve the constant-density (Issa, 1986) or variable-density (Rider and Kothe, 1998) incompressible NS equations. In this study, the projection method will be used to solve the NS, filtered NS, and RANS equations.

In the projection method, momentum equations (Equation (2.6)) can be described by two fractional steps:

$$\frac{\rho^{n+1}\mathbf{u}^{*} - \rho^{n}\mathbf{u}^{n}}{\Delta t} = -\nabla\cdot\left(\rho\mathbf{u}\mathbf{u}\right)^{n} + \nabla\cdot\left(\mu^{n}\left(\nabla\mathbf{u} + \nabla^{T}\mathbf{u}\right)^{n}\right) \tag{3.39}$$

$$\frac{\rho^{n+1}\mathbf{u}^{n+1} - \rho^{n+1}\mathbf{u}^{*}}{\Delta t} = -\nabla p^{n+1} + \rho^{n+1}\mathbf{g} \tag{3.40}$$

where Equation (3.39) is an explicit expression for the interim velocity $\mathbf{u}^{*}$, referred to as the predictor step. In Equation (3.39), all forces except gravity and pressure gradient are included. Equation (3.40) is termed the projection step. Combining Equation (3.39) and (3.40) exactly produces the time discretization of Equation (2.6):

$$\begin{aligned}\frac{\rho^{n+1}\mathbf{u}^{n+1} - \rho^{n}\mathbf{u}^{n}}{\Delta t} &= -\nabla\cdot\left(\rho\mathbf{u}\mathbf{u}\right)^{n} + \nabla\cdot\left(\mu^{n}\left(\nabla\mathbf{u} + \nabla^{T}\mathbf{u}\right)^{n}\right) \\ &\quad -\nabla p^{n+1} + \rho^{n+1}\mathbf{g}\end{aligned} \tag{3.41}$$

No additional approximation results from this decomposition.

Equation (3.40) relates $\mathbf{u}^{n+1}$ to $\mathbf{u}^{*}$. By adopting the solenoidal condition (Equation (2.5)), we have:

$$\nabla\cdot\frac{\nabla p^{n+1}}{\rho^{n+1}}=\nabla\cdot\left(\frac{\mathbf{u}^{*}}{\Delta t}+\mathbf{g}\right)\tag{3.42}$$

Equation (3.42) is named the Poisson Pressure Equation (PPE). New time-step cell-centered pressure p^{n+1} can be solved from Equation (3.42), and cell-face pressure gradient ∇p_f^{n+1} is calculated from a stencil correspondent to that of the density interpolation to faces. Equation (3.40) is then used to calculate the solenoidal face velocity field:

$$\mathbf{u}_f^{n+1}=\mathbf{u}_f^{*}-\Delta t\left(\frac{\nabla p_f^{n+1}}{\rho_f^{n+1}}-\mathbf{g}_f\right)\tag{3.43}$$

Finally, the pressure gradient on cell faces $\nabla p_f^{n+1}/\rho_f^{n+1}$ is interpolated to cell centers in order to obtain the new time-step velocity field $\mathbf{u}^{n+1}$ which satisfies the divergence free constrain enforced by Equation (2.5), from Equation (3.40).

In the following sections, the details of the algorithm used to solve Equation (3.39) and (3.40) will be presented.

3.4.1 Momentum Advection

After obtaining the volume tracking results (Section 3.3), the momentum advection $\nabla\cdot\left(\rho\mathbf{u}\mathbf{u}\right)^{n}$ can be evaluated by the face-centered velocities. Discretizing the advection term of Equation (3.39), we obtain:

$$\Delta t\int \nabla\cdot\langle\rho\mathbf{uu}\rangle^n\,dV \approx \sum_f \delta V_f^n \langle\rho\mathbf{u}\rangle_f^n = \sum_f M_f^n \mathbf{u}_f^n \tag{3.44}$$

where δV_f is the advected volume through each cell face, and M_f^n is the advected mass through each cell face. Consider Figure 3.1 that illustrates the advection of material across the right face of a cell containing an interface between two fluids. The advected volume is:

$$\delta V_f = \Delta t\mathbf{A}_f\cdot\mathbf{u}_f = \Delta t A_f \mathbf{u}_f\cdot\hat{\mathbf{n}}_f \tag{3.45}$$

where $\mathbf{u}_f\cdot\hat{\mathbf{n}}_f$ is the component of the solenoidal face velocity $\mathbf{u}_f$ normal to the face, and A_f is the face area.

The mass of each material m leaving a cell across face f can be expressed by multiplying the $\delta V_{m,f}$ and the corresponding densities:

$$M_{m,f} = \rho_m^0 \delta V_{m,f} \tag{3.46}$$

and with Equation (3.32):

$$M_{m,f} = \rho_m^0 \delta V_f f_{m,f} \tag{3.47}$$

The total mass leaving a cell face f is defined as:

$$M_f \equiv \sum_m M_{m,f} \tag{3.48}$$

Equation (3.44) can then be written as:

$$\begin{aligned}\Delta t\int \nabla\cdot(\rho\mathbf{uu})^n\,dV &\approx \sum_f M_f^n \mathbf{u}_f^n \\ &= \sum_f\sum_m M_{m,f}^n \mathbf{u}_f^n \\ &= \Delta t\sum_f\sum_m \rho_m^0\left(\mathbf{A}_f\cdot\mathbf{u}_f^n\right) f_{m,f}^n \mathbf{u}_f^n\end{aligned} \tag{3.49}$$

where $\left(\mathbf{A}_f\cdot\mathbf{u}_f^n\right) f_{m,f}^n$ is directly obtained from Equation (3.31), thus Equation (3.44) is consistent with mass advection. Note that Equation (3.49) also indicates an upwind value which is first order accurate approximation.

3.4.2 Momentum Diffusion

In this study, the fluids are assumed to be Newtonian fluids. Viscous forces are incorporated into the predictor estimate of the cell-centered velocity field by using n time step velocity field. This explicit approximation eliminates the need of solving a system of equations.

The net viscous stress on the control volume is calculated by applying the divergence theorem to the volume integral of the local stress. This reduces to a sum of the dot product of the face normal vector with the local velocity gradient multiplied by the face area:

$$\nabla\cdot\left(\mu^n\left(\nabla\mathbf{u}+\nabla^T\mathbf{u}\right)^n\right)=\sum_f \mu_f^n A_f\left[\hat{\mathbf{n}}_f\cdot\left(\nabla\mathbf{u}_f+\nabla^T\mathbf{u}_f\right)\right]^n \tag{3.50}$$

The velocity gradient is calculated by the least squares method (Chapter 3.2). The cell center viscosity μ is evaluated by using volume weighted technique.

$$\mu=\sum_m f_m\mu_m \tag{3.51}$$

The cell center viscosity is then harmonically averaged to face centroids by using an orthogonal approximation:

$$\mu_f=\frac{2}{1/\mu_{nb}+1/\mu_p} \tag{3.52}$$

where μ_{nb} is the neighbor cell viscosity, and μ_p is the cell center viscosity.

3.4.3 Projection

After obtaining the momentum advection and diffusion, the “predicted” velocity $\mathbf{u}^*$ can be calculated based on Equation (3.39). The predicted velocity $\mathbf{u}^*$ and ρ^{n+1} are then interpolated to cell faces to obtain $\mathbf{u}_f^*$ and ρ_f^{n+1}. Since we assume velocity does not vary discontinuous near an interface, the velocity interpolation can be done by the least square linear reconstruction (Chapter 3.2). However, because density varies discontinuously on an interface, we have to limit the size of the stencil. The cell face densities are calculated by averaging the cell densities on either side of a face.

The discretization of Equation (3.42) is:

$$\sum_f \left(\mathbf{A}_f \cdot \frac{\nabla p_f^{n+1}}{\rho_f^{n+1}} \right) = \sum_f \left(\mathbf{A}_f \cdot \left(\frac{\mathbf{u}_f^*}{\delta t} + \mathbf{g}_f \right) \right) \tag{3.53}$$

where cell face pressure gradient ∇p_f^{n+1} is calculated from the stencil same as that in the density interpolation to faces:

$$\nabla p_f^{n+1} = \frac{p_{nb}^{n+1} - p_p^{n+1}}{\delta x} \tag{3.54}$$

Here p_{nb}^{n+1} is the neighbor cell pressure, p_p^{n+1} the cell pressure, and δx the centroid-to-centroid distance.

We solve Equation (3.53) for the cell-centered pressure p^{n+1}. Equation (3.43) is then used to calculate the solenoidal face velocity field $\mathbf{u}_f^{n+1}$. The last step is to interpolate $\nabla p_f^{n+1} / \rho_f^{n+1} - \mathbf{g}$ to cell centers, and with Equation (3.40) and $\mathbf{u}^*$ to obtain cell center velocity field $\mathbf{u}^{n+1}$.

Equation (3.53), the Poisson Pressure Equation (PPE), is a set of linear algebraic equations for the pressure field that can be solved by iterative methods. The equation is basically elliptic in nature, and can be expressed in matrix notation as:

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{3.55}$$

where $\mathbf{A}$ is a matrix resulting from finite volume discretizations of the Laplacian. We can expect that $\mathbf{A}$ is sparse and symmetric in general. Thus when we are choosing a linear solution method, we should take advantage of this structure.

In this study, Krylov subspace methods (Golub and Van Loan, 1989; Saad 1996) are used. If a 3-D structured mesh system is used, it indicates that $\mathbf{A}$ is symmetric, conjugate gradient (CG) algorithm with symmetric successive over-relation (SSOR) (Varga, 1962) preconditioner (Strang, 1976; Strang, 1986; Golub and Van Loan, 1989) will be our choice. If 3-D unstructured grids are used, we use the preconditioned generalized minimal residual (GMRES) algorithm (Saad and Schultz, 1986) for our Krylov method. For GMRES, the preconditioning method is the two-level Schwarz multigrid method (Cai, 1995; Saad, 1996) with SSOR as the preconditioning preconditioner.

3.5 Computational Cycle

We now present the complete cycle to update the field variables within one time step. Given values of $\mathbf{u}^n$, $\mathbf{u}_f^n$, and p^n, we advance the solution to time $n+1$ in the following steps:

1. Solve Equation (3.31) for f_m^{n+1} using $\mathbf{u}_f^n$ and utilizing the multidimensional PLIC volume tracking algorithm. New time step cell density ρ^{n+1} is obtained via

f_m^{n+1} and Equation (3.27). A geometric calculation of volume fluxes of different materials crosses cell faces.

2. Obtain $\mathbf{u}^*$ by Equation (3.39) where $\nabla \cdot \left(\rho \mathbf{u}\mathbf{u}\right)^n$ is evaluated by using the volume flux information obtained directly from the previous step, and the viscous force is obtained via Equation (3.50).
3. $\mathbf{u}_f^*$ and ρ_f^{n+1} are obtained by interpolating $\mathbf{u}^*$ and ρ^{n+1} to cell faces.
4. Solve PPE (Equation (3.42) and (3.53)) to obtain cell center pressure p^{n+1}.
5. A solenoidal face velocity field $\mathbf{u}_f^{n+1}$ is obtained by Equation (3.43).
6. Finally, ($\nabla p_f^{n+1} / \rho_f^{n+1} - \mathbf{g}_f$) will be interpolated to cell center to obtain $\mathbf{u}^{n+1}$ from $\mathbf{u}^*$ via Equation (3.40).

The accuracy of this algorithm is first order in time and space. It is because we use forward differencing in time and upwind scheme in space. From the numerical experiments and several numerical simulations of breaking waves (Lin and Liu 1998a,b), this accuracy is high enough for studying the breaking-wave problems.

In terms of the boundary conditions, different steps require different boundary conditions and will be addressed in the next section.

3.6 The Implementation on Boundary Conditions

We have introduced several boundary conditions (BCs) in Chapter 2. Those boundary conditions are free-slip wall BC, no-slip wall BC, zero-pressure open BC, moving solid BC, incident wave BC, and advective open BC. Except the moving solid BC, which will be introduced in the next chapter, the numerical implementations of all the other BCs will be discussed here.

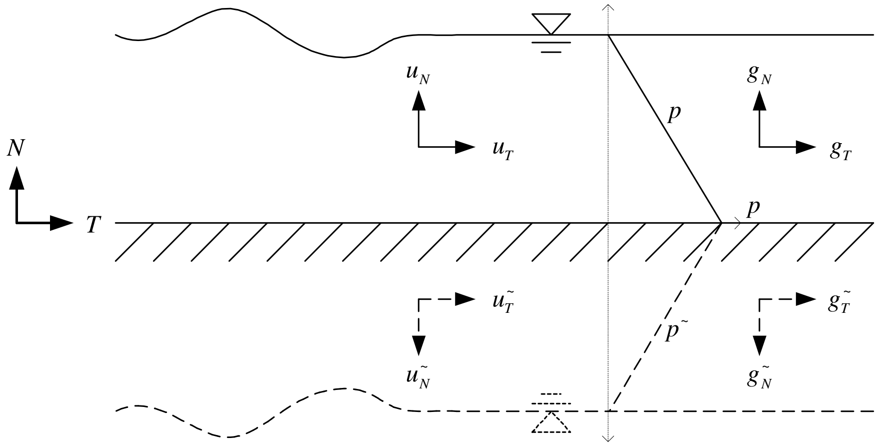


Figure 3.3: A sketch map of the free-slip boundary condition. The lower part is the image part.

Basically we can divide the boundary conditions into two categories: Dirichlet and Neumann type boundary conditions. For the Dirichlet type boundary conditions, the fluid quantities are specified on the boundary faces. Neumann type boundary conditions, on the other hand, specify the quantity gradients. Hence, the no-slip wall BC and pressure BC are the Dirichlet type boundary conditions, and free-slip wall BC is the Neumann type boundary condition. As for the incident wave and advective open boundary conditions, the time dependent volume and momentum fluxes are specified on the boundary faces, they can be categorized as the Dirichlet type boundary conditions. We shall discuss the details of each different boundary condition.

3.6.1 Free-Slip Stationary Wall Boundary Condition

The physical meaning of free-slip boundary condition is that the wall is impermeable and there is no tangential shear stress generated on the wall. From the other point of view, because this wall behaves like a mirror (Figure 3.3), it is called a symmetry plane too. Mathematically, velocity component normal to the wall is zero (Equation (2.17)), and the tangential velocity gradient components parallel to the boundary faces are zero (Equation (2.18)). Therefore, at the free-slip wall boundary faces, the convective fluxes of all quantities are zero.

In the numerical implementation, the free-slip wall boundary condition will be enforced via the least-squares spatial discretization scheme. In the VOF equation (Equation (3.31)), since $\mathbf{u}^n$ and $\mathbf{u}_f^n$ are obtained from the previous time-step, no boundary is required. The flux is zero on the boundary faces:

$$\left[\mathbf{A}\cdot\mathbf{u}^n\right]_{wall}=0 \tag{3.56}$$

where notation $[\ ]_{wall}$ indicates the wall boundary and $\mathbf{A}$ is the area vector of the wall. This zero volume flux condition will be implemented to momentum advection (Equation (3.49)) as well. In the momentum diffusion, $\partial u_T/\partial n=0$ will be applied in the predictor step on the boundary faces. Then, $u_n=0$ will be enforced via LSLR when interpolating $\mathbf{u}^*$ to $\mathbf{u}_{wall}^*$ as discussed above.

In Equation (3.43), since the velocity component normal to the wall is zero:

$$[u_N]_{wall}^{*}=[u_N]_{wall}^{n+1}=0 \tag{3.57}$$

and the gravity component normal to the wall is zero (because of the symmetry on the wall, see Figure 3.3),

$$[g_N]_{wall}=0 \tag{3.58}$$

it is equivalent to specifying a Neumann pressure boundary condition for the pressure correction:

$$\left[\nabla_N p_N\right]_{wall}^{n+1} = 0 \tag{3.59}$$

3.6.2 No-Slip Stationary Wall Boundary Condition

The physical meaning of this boundary condition is that viscous fluids are stuck on the solid wall boundary. The mathematical expression is shown in Equation (2.10) ($\mathbf{u} = \mathbf{u}_s = 0$) which is a Dirichlet boundary condition.

Similar to the free-slip boundary condition, because there is no flux through the wall, all the convective fluxes are zero. We can apply this boundary condition to VOF equation and momentum equation to obtain f_m^{n+1} and momentum advection.

Combining Equation (2.10) and the continuity equation (Equation (2.5)), the following equations can be derived:

$$\left[\frac{\partial \mathbf{u}_T}{\partial \mathbf{T}}\right]_{wall} = 0 \Rightarrow \left[\frac{\partial \mathbf{u}_n}{\partial \mathbf{n}}\right]_{wall} = 0 \tag{3.60}$$

where $\mathbf{u}_T$ is the velocities tangential to the wall; $\mathbf{u}_n$ the velocity normal to the wall, $\mathbf{T}$ the tangential vector of the wall, and $\mathbf{n}$ the normal vector of the wall.

Therefore, on a wall boundary, not only the zero velocity and zero flux will be applied, Equation (3.60) will also be implemented to the momentum diffusion. The rest of shear stresses on the wall face can be calculated by using one-side extrapolation approximation.

Because the predicted velocity on the wall boundaries is known as the wall speed,

$$\mathbf{u}_f^* = \mathbf{u}_{wall}^{n+1} \tag{3.61}$$

the pressure gradient on the wall face can be calculated explicitly via Equation (3.43):

$$\frac{\nabla p_{wall}^{n+1}}{\rho_{wall}^{n+1}} = \mathbf{g}_{wall} \tag{3.62}$$

For the same reason as that described in free-slip boundary condition, the gravity component normal to the wall is zero,

$$\left[g_N\right]_{wall} = 0 \tag{3.63}$$

hence, it is equivalent to specifying a Neumann boundary condition:

$$\left[\nabla_N p_N\right]_{wall}^{n+1} = 0 \tag{3.64}$$

3.6.3 Pressure Dirichlet Boundary Condition

For the wave simulations, we frequently specify $p_f = 0$ on one side of the domain to simulate an open boundary condition. This pressure Dirichlet open boundary condition physically indicates that the boundary is open to the atmosphere.

When the pressure is specified on the boundary faces, velocity then can not be prescribed. When solving the VOF equation (Equation (3.31)), since all the information is from the old time-step solutions ($\mathbf{u}^n$ and $\mathbf{u}_f^n$), no boundary condition is required. The volume flux information will be brought in to calculate the momentum advection of Equation (3.49). The boundary face velocity gradients $\nabla \mathbf{u}_f^n$ used to calculate the momentum viscous are extrapolated from the interior using LSLR. Then the predicted velocity $\mathbf{u}^*$ can be calculated. Up to this step, no boundary condition has been applied.

At the correct step, pressure Poisson equation (Equation (3.53)) is solved to obtain p^{n+1}. The zero pressure boundary condition is applied to the linear solution solver where the pressure gradients at the boundary faces are interpolated based on the known Dirichlet pressure boundary conditions and interior cell points. Then the pressure boundary condition will be used in Equation (3.43) to obtain $\mathbf{u}_f^{n+1}$. In the last step, the boundary condition will be implemented again when interpolating the cell face pressure gradients to cell centroid in order to obtain $\mathbf{u}^{n+1}$.

3.6.4 Incident Wave Boundary Condition

To send waves directly from one side of the boundaries, we specify the volume fluxes and boundary velocities on the boundary faces. The volume fluxes and boundary velocities are prescribed based on the time history of velocity and volume fraction:

$$\mathbf{u}_{wall}^{n+1} = \mathbf{u}_{wall}^{n+1}\left(\eta(t) + h, \mathbf{x}\right) \tag{3.65}$$

$$f_{m,wall}^{n+1} = f_{m,wall}^{n+1}\left(\eta\left(t\right) + h, \mathbf{x}\right) \tag{3.66}$$

where f_{wall} indicates the incident wave boundary faces, $\eta\left(t\right)$ is time dependent free surface displacement, h is the local still water depth, and $\left(\eta\left(t\right) + h\right)$ is the total water depth.

Equation (3.65) and (3.66) are then applied to the VOF equation (Equation (3.31)) and the advection equation (Equation (3.49)) to obtain cell center volume fraction f_m^{n+1}, and momentum advection. The rest of the calculation is the same as that in the no-slip boundary condition with boundary face velocities equal to the specified face velocities $\mathbf{u}_{wall}^{n+1}$.

3.6.5 Advective Open Boundary Condition

As we have discussed in Chapter 2, in order to save the computational time, in some cases we have to truncate the computational domain, and let the wave leave the domain through the boundary without reflection. This may be done with our knowledge of wave group velocity (Equation (2.18)). By defining ϕ in Equation (2.20) as volume fraction f, the new time step volume fraction can be obtained explicitly at the boundary faces:

$$f_{m,AOBC}^{n+1} = f_{m,AOBC}^{n} + C_g \Delta t \frac{\partial f_{m,AOBC}^{n}}{\partial x} \tag{3.67}$$

Here the subscript $_{m,AOBC}$ indicates the boundary faces, and we have assumed that x is the streamwise direction. $C_g = \sqrt{g\left(h+\eta\right)}$ is the long wave group velocity, which is independent of location on the boundary face.

Similarly, velocities can be expressed as:

$$\mathbf{u}_{m,AOBC}^{n+1} = \mathbf{u}_{m,AOBC}^{n} + C_g \Delta t \frac{\partial \mathbf{u}_{m,AOBC}^{n}}{\partial x} \tag{3.68}$$

Thus, this boundary can be implemented in the same way used in the incident wave boundary condition.

However, we shall point out here that the wave group velocity C_g is determined based on the long wave theory, this method is not suitable to deal with short waves. For short waves, another type of numerical method called “numerical sponge layer” will be presented in Chapter 3.9.

3.6.6 Discussion

There are two sets of physical variables in the Navier-Stokes equation: p and $\mathbf{u}$. Once the pressure boundary condition is specified, e.g. Dirichlet pressure boundary condition, at a boundary, velocity has to be determined by solving the NS equations. Similarly, once the velocity or velocity gradient is pre-described, the pressure has to be solved from NS equations.

An interesting phenomenon can be found at the boundary faces where the velocity Dirichlet boundary conditions are specified. If the free-slip or no-slip wall boundary conditions are applied, the pressure gradient normal to the boundary faces is Neumann pressure boundary condition (Equation (3.59) and (3.64)). However, if the wall boundary is lying on the bottom of a water tank, physically the hydrostatics pressure gradient normal to the wall is not zero there. Does this mean that those pressure Neumann boundary conditions are wrong?

As we have addressed before, once velocity boundary conditions are pre-described, pressure cannot be pre-described, and vice versa. Therefore, those pressure gradients on the boundary faces are not real boundary conditions. Actually they (Equation (3.59) and (3.64)) are the products of projection method, and have no physical meanings.

Then the next question is "where can we find the pressure gradient effects?". The answer is that they are hiding in $\nabla\cdot\left(\mathbf{u}^*/\Delta t\right)$ (Equation (3.42)). In the predict step, without considering the boundary conditions, the predicted velocity $\mathbf{u}^*$ contains no effect from pressure since pressure gradient term has been neglected at this step. However, we apply the boundary conditions, $\mathbf{u}_{wall}^{n+1}$ or $\nabla\mathbf{u}_{wall}^{n+1}$, to the boundary faces where $\mathbf{u}_{wall}^{n+1}$ and $\nabla\mathbf{u}_{wall}^{n+1}$ have already included the pressure gradient effect of the $n+1$ time step. Thus, $\mathbf{u}^*$ covers the effect of pressure gradient and the pressure gradient effect will be eliminated in the projection step.

3.7 Error Analysis and Numerical Stability

3.7.1 Error Analysis

Numerical solutions to a fluid flow problem are only approximate ones. The errors, therefore, can be defined as the difference between the approximated solutions and the true physical solutions. Errors always exist. However, our goal is to obtain desired accuracy with the least effort, or to obtain maximum accuracy with available resources.

The errors can be classified as modeling errors, discretization errors, and iteration errors. Modeling errors depend on the assumptions made when deriving the governing equations. They can be negligible or large. For laminar flows, the Navier-Stokes equations are accurate enough and the modeling errors are negligible. However, for turbulent flows, the modeling errors may be large. The modeling errors are also introduced by assumptions used in the boundary conditions. Modeling errors can only be evaluated by comparing the solutions with accurate experimental data or DNS results. However, before making the comparison, we have to make sure that the other error sources are small and negligible.

Discretization errors are introduced by the discretization of the fluid domain and can usually be reduced when the grid is refined. Grid refinement is the main tool in this study for improving the accuracy of a simulation. We usually perform a simulation on coarse meshes first to get an impression of the overall features of the solution. Then the grid is refined until there is no significant difference in the results between the previous and current refinements. These results are called “grid independent”.

Discretization errors can also be estimated mathematically. Since forward time scheme is adopted, the accuracy is first order in time. The nonlinear advection terms in Equation (3.44) are evaluated by using an upwind scheme which is a first-order accurate approximation and leaves the error to be second order. As for the momentum diffusion, the face velocity gradient is calculated by using LSLR, the accuracy is second order. Also, because the divergent theory is used to estimate the momentum stress, the accuracy is second order. Therefore, the leading errors will be fourth-order in the uniform grid system. Obviously, the discretization errors are dominated by the advection errors. Since the advection errors are second order, they have a similar effect on the dissipation. As we have discussed in Chapter 1, this study will simulate the turbulence effect using the eddy viscosity assumption. It is important to control the advection errors so that they will not spoil the turbulence simulations.

It is difficult to estimate the discretization errors from truncating the advection terms, especially on 3-D unstructured meshes and the utilization of finite volume discretization with least square scheme. However, it is well-known that finite volume methods can be reduced to finite difference methods on structured, orthogonal meshes. Therefore, the analysis can be started from error analysis of the finite difference method. Because the discretization errors are dominated by the advection terms, we shall focus our analysis on them. Considering the momentum advection on a x-y-z orthogonal coordinate system, the x-component of the solenoidal advection can be expressed as:

$$advection_x = u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} \tag{3.69}$$

The x-component of the truncation errors:

$$\begin{aligned} TE_{adv,x} = & \left(\frac{|u|\Delta x}{2} - \frac{|u|^2 \Delta t}{2} \right) \frac{\partial^2 u}{\partial x^2} + \\ & \left(\frac{|v|\Delta y}{2} - \frac{|v|^2 \Delta t}{2} \right) \frac{\partial^2 u}{\partial y^2} + \\ & \left(\frac{|w|\Delta z}{2} - \frac{|w|^2 \Delta t}{2} \right) \frac{\partial^2 u}{\partial z^2} + \ldots \end{aligned} \tag{3.70}$$

The y and z components can be expressed in a similar way. From the above expansion, the numerical dissipation can be characterized by the "numerical viscosities" defined below:

$$\begin{aligned} \nu_{n,xx} &\approx \left(\frac{|u|\Delta x}{2} - \frac{|u|^2 \Delta t}{2} \right), \\ \nu_{n,yy} &\approx \left(\frac{|v|\Delta y}{2} - \frac{|v|^2 \Delta t}{2} \right), \\ \nu_{n,zz} &\approx \left(\frac{|w|\Delta z}{2} - \frac{|w|^2 \Delta t}{2} \right) \end{aligned} \tag{3.71}$$

The numerical viscosities are determined by the grid size, fluid particle velocity and time step. In general, the numerical simulations show that the numerical viscosities are at least one order of magnitude smaller than the turbulence eddy viscosities. Therefore, the numerical errors will not spoil the simulation results.

The iteration errors are easy to be controlled. As a rule of thumb, the iteration errors should be at least an order of magnitude lower than discretization errors and can be defined as the differences between successive iterates:

$$\mathbf{r} = \mathbf{b}^{i} - \mathbf{b}^{i-1} \tag{3.72}$$

where $\mathbf{b}$, a vector defined in Equation (3.15), and the superscription i indicates the i^{th} interaction.

When the successive iterates do not differ more than a certain small number, the iterations stop and the solutions are converged. The stopping criterion is defined as:

$$\frac{\|\mathbf{r}\|}{\|\mathbf{b}^i\|} < \text{Stopping\_Criterion} \tag{3.73}$$

In this study, the Stopping_Criterion is equal to or smaller than 10^{-12} which is small enough to neglect the iteration errors.

3.7.2 Stability Analysis

Similar to the error analysis introduced in the previous section, the stability analysis is derived based on the same way as the one in the finite different scheme. For the linear PDE, the most widely used method to analysis the numerical stability is the von Neumann stability analysis method (Jaluria and Torrance, 1986). In this method, the initial error described by a series of Fourier modes will be substituted into the finite difference equation by assuming the initial error will be advected by the numerical scheme. For a stable scheme, the amplification factor of each Fourier mode has to be smaller than 1.0.

However, there exist nonlinear advection and diffusion terms in the momentum equations, the convectional von Neumann's method is not applicable directly. To solve this problem, a constant maximum velocity $Max(|\mathbf{u}|)$ will be used in the advection term, and a constant maximum effective viscosity $Max(\nu_{eff})$ will be used in the diffusion term. A standard von Neumann's analysis is then performed to obtain the following stability criteria,

$$\delta t_c < C_r \frac{d\xi}{Max(|\mathbf{u}|)} \tag{3.74}$$

$$\delta t_\mu < V_\mu \frac{(d\xi)^2}{Max(\nu_{eff})} \tag{3.75}$$

where δt_c is the time step restricted by the advection term. C_r is Courant number, which is defined as $C_r = Max(|\mathbf{u}|)\delta t / d\xi$, $d\xi$ is some measure of the cell size, δt_μ is the time step restricted by the diffusion term, V_μ is the viscous number which is defined as $V_\mu = Max(\nu_{eff})\delta t / (d\xi)^2$.

The stability analysis states that the time integration of advection terms is stable only when $C_r \le 1.0$. In this study, $C_r \le 0.45$ will be used. The viscous number V_μ is $1/2$, $1/4$, and $1/6$ in 1-D, 2-D, and 3-D, respectively, for uniform rectilinear cells.

3.8 Model Testing

3.8.1 Solitary Wave Propagation in Constant Water Depth

One of the classical benchmark problems for testing wave simulation model is the solitary wave propagation in a constant depth. The solitary wave has a finite amplitude and permanent shape. During the propagation, the nonlinearity is balanced with the frequency dispersion. Derived from the Boussinesq equations the solitary wave solution can be expressed as:

$$\eta(x,t) = a\,\mathrm{sech}^2\left[\sqrt{\frac{3H}{4h^3}}(x-ct)\right] \tag{3.76}$$

where η is the free surface elevation, h the still water depth, and $c = \sqrt{g(h+H)}$ is the wave celerity.

To test the numerical model, we have chosen the wave nonlinearity, defined as H / h, to be 0.1. In the numerical simulation, the still water depth $h = 1$ m is used. The numerical domain is $0 \leq x / h \leq 120$ in the streamwise direction, $0 \leq z / h \leq 1.22$ in vertical direction, and $0 \leq y / h \leq 0.25$ in the spanwise direction. In the streamwise direction, 480 cells with uniform size of $\Delta x / h = 0.25$ are used. In the vertical direction, 80 non-uniform cells with smallest cell size $\Delta z / h = 0.0048$ on the top of the domain. In the spanwise direction, we use only 3 cells with uniform size $\Delta y / h = 0.083$. The boundary conditions are incident wave BC at $x / h = 0$ faces, $p = 0$ at $z / h = 1.22$ and free-slip elsewhere. In this case, the turbulence model is turned off and the molecular viscosity is set to zero. Therefore, the model basically is solving the potential flow, which is the assumption of solitary wave analytical solution. During the simulation, the time step Δt is adjusted automatically constrained by the stability criteria (Chapter 3.7.2). Because there is no viscous terms in this case, the Δt is constrained by the advection processes only. The velocity vector (u, v, w) and free surface elevation η are specified at the left boundary based on the Boussinesq solutions derived by Lee et al. (1982). The velocity vector can be expressed as:

$$\begin{aligned}
u &= c\left\{\left[\frac{H}{h} + 3\left(\frac{H}{h}\right)^2\left(\frac{1}{6} - \frac{1}{2}\left(\frac{z}{h}\right)^2\right)\right]\frac{\eta}{H} - \left(\frac{H}{h}\right)^2\left(\frac{7}{4} - \frac{9}{4}\left(\frac{z}{h}\right)^2\right)\left(\frac{\eta}{H}\right)^2\right\} \\
v &= 0 \qquad\qquad (3.77) \\
w &= c\sqrt{\frac{3H}{h}}\frac{z}{h}\frac{\eta}{h}\tanh\left(\sqrt{\frac{3}{4}\frac{H}{h^3}}\left(x - ct\right)\right)\left\{1 + \frac{H}{2h}\left[1 - 7\frac{\eta}{H} - \left(\frac{z}{h}\right)^2\left(1 - \frac{3\eta}{H}\right)\right]\right\}
\end{aligned}$$

The air density has been set to zero, and thus is excluded from the iterations.

Figure 3.4 shows the comparisons between the computed and analytical solutions.

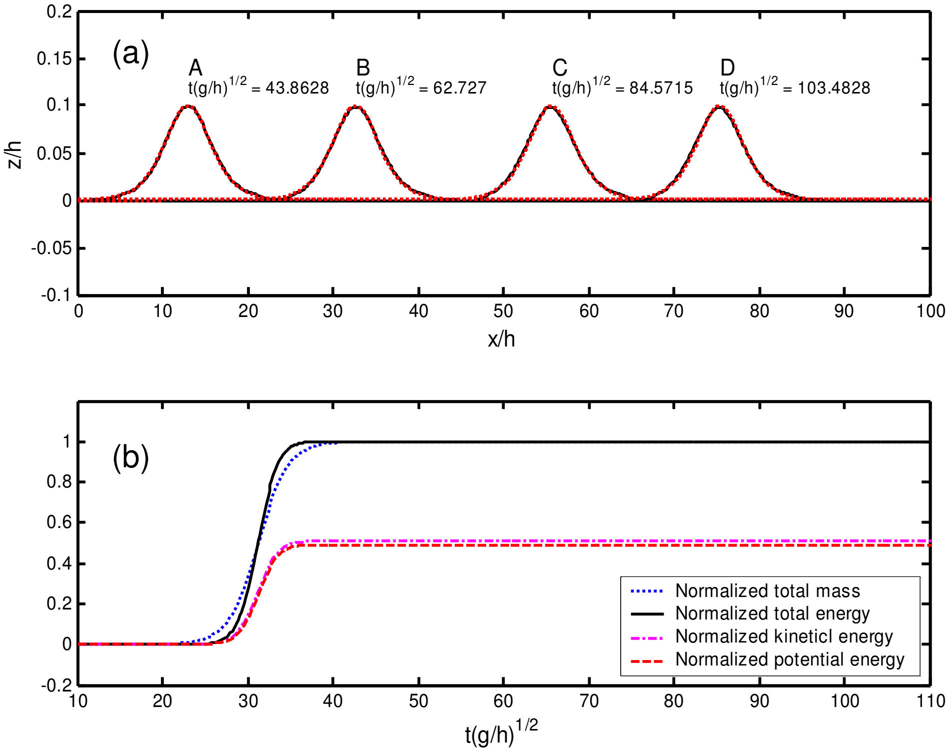


Figure 3.4: (a) Comparisons of solitary wave propagation in a constant water depth at different time $t\left(g/h\right)^{1/2}$. Solid lines: analytical solutions; Dot lines: numerical solutions. (b) Time histories of normalized mass, total energy, kinetic energy, and potential energy. The mass has been normalized by the calculated mass at $t\left(g/h\right)^{1/2}$ =40.0 and the energy has been normalized by the calculated total energy at $t\left(g/h\right)^{1/2}$ =40.0.

In Figure 3.4(a), we provide four snapshots of solitary wave profiles at the cross-section of $y/h = 0.125$ and $t\sqrt{g/h}$ = 43.863, 62.727, 84.572, and 103.483. In this case, the numerical solutions are uniform in the spanwise direction. From Figure 3.4(a) we can see that the numerical solutions match the analytical solutions very well in terms of the wave profiles and phase.

Figure 3.4(b) provides the normalized time history of mass and energy. Because the turbulence model has been turned off and there is no viscous effect in this case, the energy should be conserved. The total mass should also be conserved because the solitary wave is in the numerical domain all the time. From Figure 3.4(b), very good conservations can be seen. Based on the linear wave theory, the kinetic energy should equal to the potential energy exactly. However, due to the nonlinear effect, the simulation result shows the kinetic energy is slightly greater than the potential energy.

3.8.2 Conservation of Advective Open Boundary Condition

Here, we inspect the conservation of advective open boundary condition (AOBC) based on the same solitary wave profile as the one described in the previous section. The numerical setup is similar to the previous case except the domain is smaller ($0 \le x/h \le 40$) and the downstream boundary condition is AOBC. The face fluxes on the downstream are specified based on equation (2.18) with $C_g = \sqrt{g(h+\eta)}$, where η is the local wave height. The mass and energy conservation comparisons are shown in Figure 3.5. In Figure 3.5 we can see that the mass and energy have been smoothly transferred out of the computational domain after $t\sqrt{g/h} = 60$. Only less than 3% of the kinetic energy discrepancy can be seen after $t\sqrt{g/h} = 60$. This may be caused by the overestimation of the phase speed.

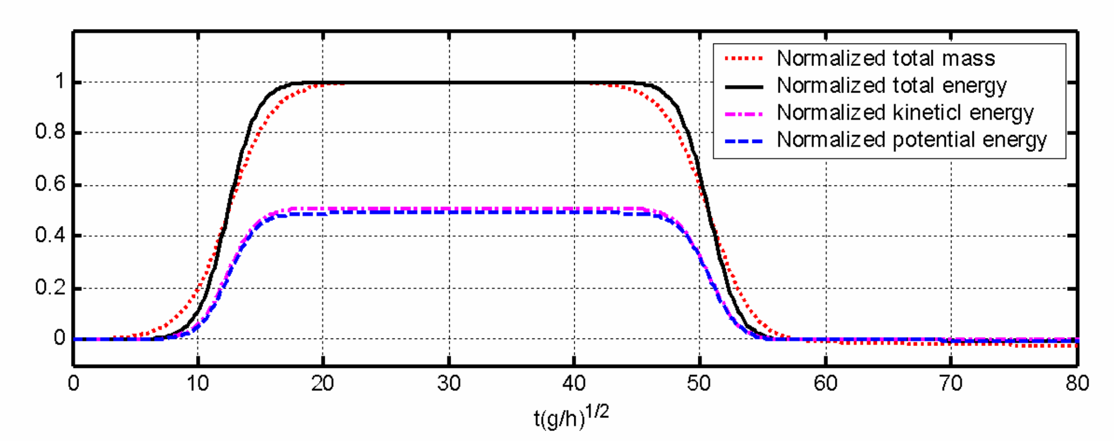


Figure 3.5: Conservation test of advective open boundary condition. The lines are time histories of normalized mass, total energy, kinetic energy, and potential energy. The mass has been normalized by the calculated mass at $t\left(g/h\right)^{1/2}=22.0$ and the energy has been normalized by the calculated total energy at $t\left(g/h\right)^{1/2}=22.0$.

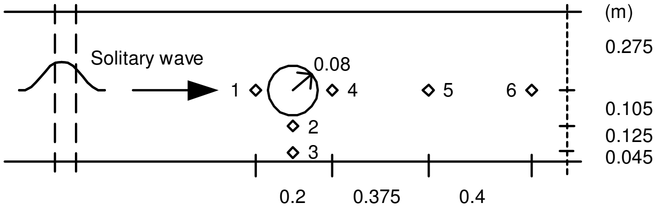


Figure 3.6: Model plan and installed depth gauges for a solitary wave overpassing a vertical circular cylinder.

The accuracy of advective open boundary condition is strongly dependent on the estimation of C_g. The over- or under-estimations of C_g might result in unphysical waves. From our experiences, the advective open boundary condition is suitable for long waves, for instance, solitary waves and cnoidal waves. As for the short waves, we need other treatments for the outgoing waves. One of the choices will be the numerical sponge layer. We shall discuss it later.

3.8.3 Solitary Wave Interaction with a Circular Cylinder

In the precious two cases, we have demonstrated the performance of our numerical model. However, in those cases, the flow conditions are nearly two-dimensional, and the meshes are orthogonal grid systems. Those waves are uniform in the spanwise direction. In this section, we shall study a three-dimensional case with an unstructured mesh.

3.8.3.1 Numerical setup and mesh

In this numerical simulation, the scattering of a solitary wave by a vertical circular cylinder is performed (Figure 3.6). The laboratory experiments performed by Antunes do Carmo et al. (1993) will be used to check the accuracy of the numerical solutions. The experiment is done in a channel with 0.55 m width. A water surface piercing cylinder, with 0.16 m of diameter, is installed in the center of the channel.

The numerical simulation is conducted in a half flume with a domain of $0 \le x \le 9$ (m), $0 \le y \le 0.275$ (m), $0 \le z \le 0.25$ (m), where x is the streamwise axis, y is the spanwise axis, and z is the vertical axis. The cylinder is located vertically at $x = 4.5\text{m}$ and $y = 0\text{m}$. An unstructured mesh is generated by Cubit 8.0 (provided by Sandia National Laboratories) with finer grids around the cylinder edge (Figure 3.7). The total cell number is 39800. We have employed six wave gauges at the following locations, in terms of (x, y) coordinates (in meters): gauge 1 = (4.220, 0.275), gauge 2 = (4.320, 0.170), gauge 3 = (4.320, 0.045), gauge 4 = (4.420, 0.275), gauge 5 = (4.795, 0.275), and gauge 6 = (5.195, 0.275). Figure 3.6 shows the numerical setup and the location of the gauges.

The undisturbed water depth h is 0.15 m. A solitary wave with wave height $a = 0.0375\text{m}$ is sent from left side boundary ($x = 0\text{m}$). In order to get rid of any possible interference, only no-flux wall boundaries are adopted. The numerical flume is long enough so that the reflected wave will not reach the gauges during the measurements. In this case, the nonlinearity of the solitary wave is not high ($a / h = 0.25$) and the wave length $\lambda = 11.24$ m (based on the 95% volume to the total wave volume) is much greater than the diameter (0.16 m). The turbulence model and the molecular viscous term are also turned off.

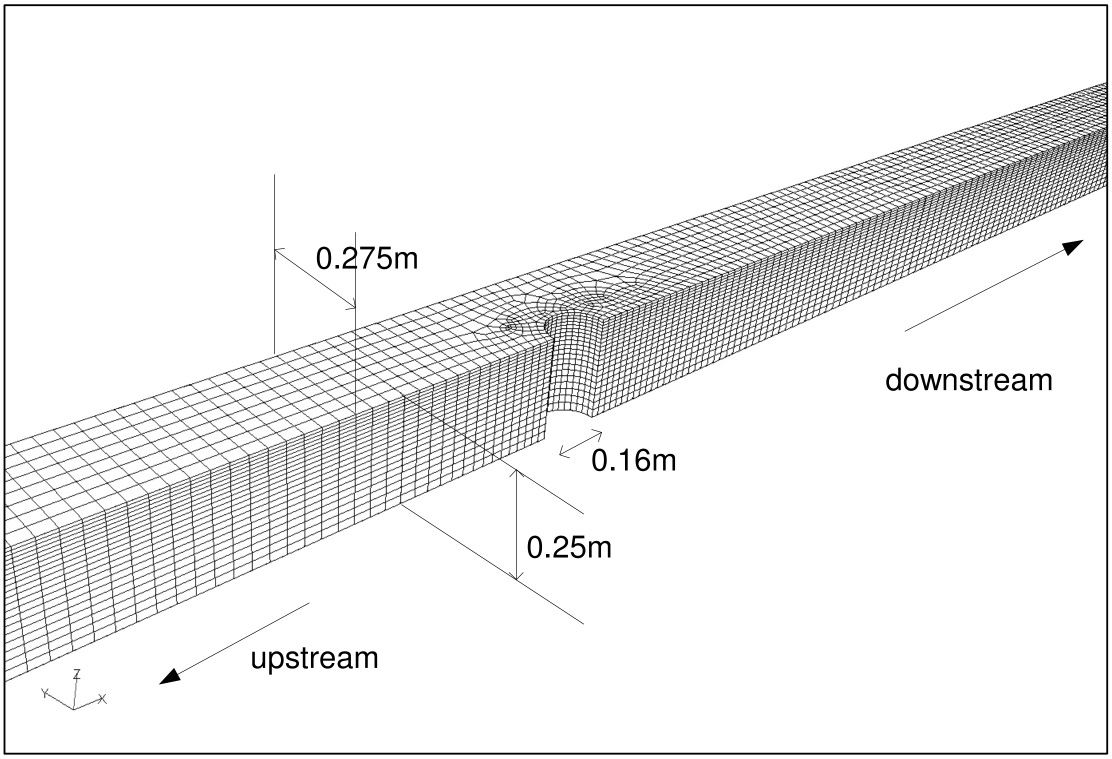


Figure 3.7: The unstructured mesh used in the simulation of a solitary wave overpassing a circular cylinder. (The mesh is generated by CUBIT 8.0).

3.8.3.2 Results and discussion

The numerical results show that short scattered waves are generated after the solitary wave impinges on the cylinder. These short waves propagate upstream, and the main wave recovers to solitary shape after a distance roughly 5 times of the cylinder diameter.

Figure 3.8 shows the time history free-surface elevations measured from the six wave gauges. The numerical solutions are compared to the laboratory data. The comparison shows the numerical simulation can provide high accuracy results in terms of the wave elevation and phase speed. Figure 3.8 also shows that this numerical model can capture both long waves and scattered short waves.

3.9 Internal Wave Maker and Numerical Sponge Layer

In the previous section, we have presented the numerical algorithm of solving the Navier-Stokes equations as well as the implementations of several boundary conditions. We also provide three cases to demonstrate the accuracy and capability of solving wave problems. In this section, the numerical implementations of the internal wavemaker and numerical sponge layer will be presented. The combination of these two numerical methods is very useful for studying the periodic wave problems.

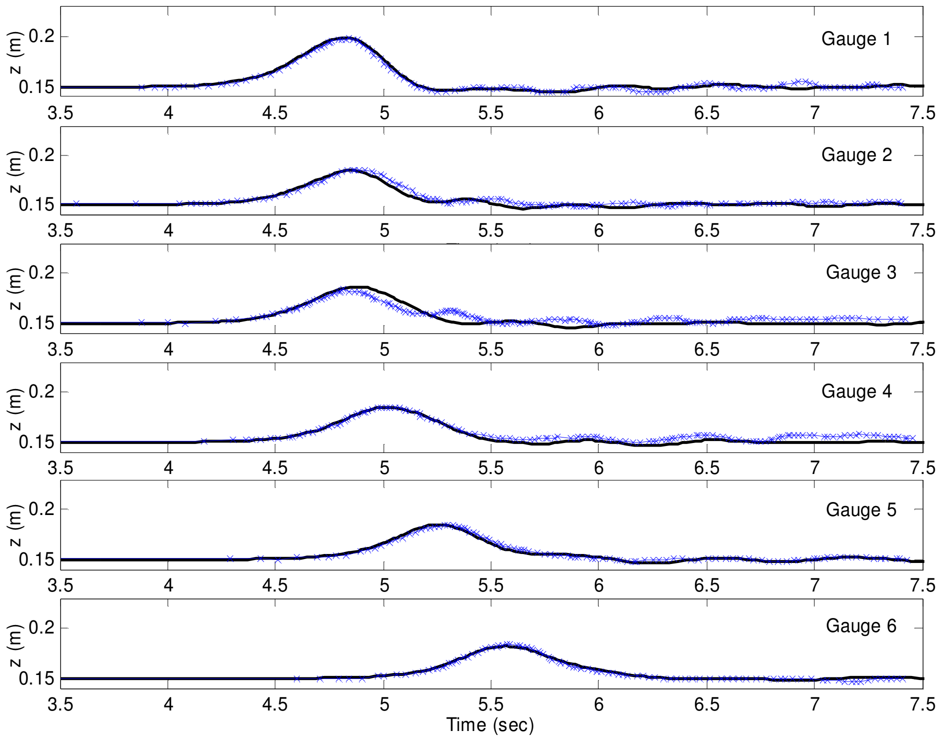


Figure 3.8: Comparison between numerical and experimental results for a solitary wave overpassing a vertical circular cylinder.

3.9.1 Introduction

It is a challenging task to generate waves through the boundary and at the same time absorbing reflected (or scattered) waves. In the laboratory experiment, active absorbing wavemaker, based on the linear wave theory, has been developed (Schaffer, 1994). They measure the surface elevation in front of the wave maker so that the incident and reflected waves can be separated. Based on the linear wave theory, the wavemaker can be adjusted to absorb the reflected waves.

For numerical simulations, many different methods can be employed to deal with the reflected waves. In the depth-averaged equation models, the incident and reflected waves can be separated based on the depth-averaged theories. For instance, Kobayashi et al., (1987) developed a boundary condition that can absorb the reflected waves and send the incident waves at the same time. With a similar approach, Van Dongeren and Svendsen (1997) developed a wave absorbing-generating boundary condition for two-dimensional shallow water equation model. However, since the boundary conditions are developed based on the shallow water assumption, they are not suitable to absorb the short waves. Another issue is that since all these absorbing boundary conditions assume the linear superposition of reflected and incident waves at the wave generating-absorbing boundaries, they can only be applied to small amplitude waves. Wei and Kirby (1995) found that this type of boundary conditions can lead to the accumulation of errors that may eventually contaminate numerical results in the entire domain after a long simulation.

For the numerical model based on the Navier-Stokes equations, Petit et al. (1994) and Lin and Liu (1998a) adopted the similar superposition concept as the depth-integrated equation methods to develop the so-called weakly reflecting boundary condition to send the incident waves and absorb the weakly reflected waves in their two-dimensional numerical flumes. However, because the linear wave assumption is still involved in the derivation, this approach can only be applied to small amplitude waves. Furthermore, Petit et al. (1994) also experienced excessive numerical errors when lengthy computations were made. In order to conquer this problem, Lin (1998) proposed a method to enforce the mass conservation by controlling the mass flux from the boundary based on the incident waves and the estimated reflected waves. This effort makes longer computations without introducing numerical instability. However, this approach is still only applicable to the weakly reflected waves. All the absorbing-generating boundary conditions require accurate estimation of the reflected waves. This estimation will be more difficult to make in a three dimensional numerical simulation since the waves may scattered in any horizontal direction.

Another idea for generating waves numerically is to use the source function inside the numerical domain. This idea is to introduce a source function in the mass conservation to increase or decrease the mass and so as to generate desired waves. This method is called internal wave-maker and was proposed by Larsen and Dancy (1983). In their original model, a line source is employed to generate waves based on the Boussinesq-type analytical solutions. In order to absorb the outgoing waves, Israeli and Orszag (1981) proposed the sponge-layer method. Iwata et al. (1996) and Lin and Liu (1999) extended the internal wavemaker and sponge layer method to generate any waves in their two dimensional numerical domain.

The internal wavemaker plus the sponge layer is more attractive than the wave absorbing-generating boundary condition since the source function inside the numerical domain will not interact with the reflected waves, and the sponge layer is able to absorb both short and long waves if the region of the sponge layer is long enough. In this study, the internal wavemaker and sponge layer method will be incorporated into current three-dimensional model. An example of Cnoidal waves will be presented.

3.9.2 Internal Wavemaker

The flow motion of the incompressible fluid is governed by the Navier-Stokes equation described in chapter 2. The numerical implementation with VOF algorithm has been presented in chapter 3. In order to generate desired waves numerically from the cells inside the numerical domain, a source function is added to the conservation equations. The mass source could be a point source, a line source, or a finite volume source. In this study, we will present the general finite volume source function in the conservation equations.

A mass source can be added through a source term in the mass conservation equation:

$$\nabla\cdot\mathbf{u}=\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=S\left(x,y,z,t\right)\ \text{in}\ V \tag{3.78}$$

where $S\left(x,y,z,t\right)$ is the source function inside the finite volume region V. Substituting equation (3.78) into equation (2.1) and (2.2), the mass conservation equation becomes:

$$\frac{\partial\rho}{\partial t}+\mathbf{u}\cdot\nabla\rho=-\rho S \tag{3.79}$$

Momentum equation becomes:

$$\frac{\partial\left(\rho\mathbf{u}\right)}{\partial t}+\nabla\cdot\left(\rho\mathbf{u}\mathbf{u}\right)=-\nabla P+\nabla\cdot\tilde{\tau}+\rho\mathbf{g}+\rho\mathbf{u}S \tag{3.80}$$

Therefore, in the projection method, a source function has to be incorporated into the predictor step:

$$\frac{\rho^{n+1}\mathbf{u}^{*}-\rho^{n}\mathbf{u}^{n}}{\Delta t}=-\nabla\cdot\left(\rho\mathbf{u}\mathbf{u}\right)^{n}+\nabla\cdot\left(\mu^{n}\left(\nabla\mathbf{u}+\nabla^{T}\mathbf{u}\right)^{n}\right)+\rho^{n}\mathbf{u}^{n}S^{n} \tag{3.81}$$

The Poisson Pressure equation needs to be modified as:

$$\nabla\cdot\frac{\nabla p^{n+1}}{\rho^{n+1}}=\nabla\cdot\left(\frac{\mathbf{u}^{*}}{\Delta t}+\mathbf{g}\right)-\frac{S^{n+1}}{\Delta t} \tag{3.82}$$

Once the mass is changed due to the source function, the free surface will response, and waves will be generated by gravity. This phenomenon is very similar to the water wave generated by underwater explosion. The source function in the computational domain is within a finite domain. The shape of finite volume source can be arbitrary. However, for current simulation purpose, we shall temporary focus on the 2-D (uniform in the lateral direction) source function. Hopefully we can derive the 3-D source function in the near future. As for the 2-D source region, the lateral width of the source region is equal to the channel width. The finite domain occupies several cells and its size can be represented as l_x, l_y, and l_z where subscript x indicates the streamwise direction, y the spanwise direction, and z the vertical direction. As addressed before, $l_y=W$, where W is the channel width. The centroid of the finite volume region can be regarded as the location as the internal wavemaker.

Lin and Liu (1999) proposed an equation to generate the desired waves.

$$\int_0^t\int_V S\left(x,y,z,t\right)dVdt=2\sum_{i=1}^{n}\int_0^t C_i\eta_i\left(t\right)dt \tag{3.83}$$

where S is the internal source function, C_i is the phase speed of the i^{th} target wave, and η_i is the free surface displacement of the i^{th} target wave. The factor “2” accounts for the face that waves are generated on both sides of finite source.

In this study, the internal source function is used to generate the cnoidal waves and solitary waves. For the cnoidal waves, Wiegel (1960) derived the analytical solution based on the Boussinesq equations. The cnoidal wave free surface elevation can be described as:

$$\eta(t) = y_t + Hcn^2\left[2K(m)\left(-\frac{t}{T} + t_s\right), m\right] \tag{3.84}$$

where y_t is the distance between the still water level and wave trough, H is the wave height, t_s is the initial time, cn is the Jacobi elliptic function associated with the cosine, and $K(m)$ is the complete elliptic function of the first kind with modulus m. Therefore, the source for generating the cnoidal waves can be described as:

$$S(t) = \frac{2C\eta W}{l_x l_y l_z} = \frac{2C}{l_x l_z}\left\{y_t + Hcn^2\left[2K(m)\left(-\frac{t}{T} + t_s\right), m\right]\right\} \tag{3.85}$$

where C is the wave celerity and can be evaluated as Wiegel (1960):

$$C = \left[1 - \frac{H}{h} + \frac{2}{m}\frac{H}{h}\left(1 - \frac{3}{2}\frac{E(m)}{K(m)}\right)\right]^{1/2}\sqrt{gh} \tag{3.86}$$

with wave length:

$$\lambda = \sqrt{\frac{16h^3}{3H} m \frac{C^2}{gh}} \cdot K(m) \tag{3.87}$$

and wave period:

$$T = \frac{\lambda}{C} \tag{3.88}$$

where E and K are the complete elliptic integrals of the first and second kind.

For a solitary wave, the free surface elevation is:

$$\eta(t) = H \operatorname{sech}^2\left[\sqrt{\frac{3H}{4h^3}}\left(-C\left(t-t_s\right)\right)\right] \tag{3.89}$$

Therefore, the internal source to generate a solitary wave is:

$$S(t) = \frac{2C}{l_x l_z} H \operatorname{sech}^2\left[\sqrt{\frac{3H}{4h^3}}\left(-C\left(t-t_s\right)\right)\right] \tag{3.90}$$

where C is the solitary wave celerity and can be expressed as:

$$C = \sqrt{g\left(h+H\right)} \tag{3.91}$$

3.9.3 Numerical Sponge Layer

The internal wavemaker generates waves in both upstream and downstream directions. In most of simulations, only one side of the waves is desired. Therefore, an open boundary condition is required to allow the undesired wave leave the computational domain without reflection from the boundary. As addressed in section 4.2, many researchers have developed wave absorbing-generating boundary condition (Kobayashi et al., 1987; Van Dongeren and Svendsen, 1997) or weakly reflected boundary condition (Petit et al., 1994; Lin and Liu, 1998a) to absorb the reflected waves. These methods are referred to as the active wave absorbing system because the system provides an active response to the waves (Troch and De Rouck, 1998). On the other hand, a passive system damps the wave motion. Due to the linear assumption, the error on predicting the wave height and velocities on the boundary faces will accumulate and contaminate the numerical results after a long run (Wei and Kirby ,1995). Here, we shall present a passive wave absorbing system that absorbs waves by using an efficient numerical sponge layer technique.

The idea of the numerical sponge layer is not new and can be explained as follows: when waves propagate into the sponge layer region, their velocities are gradually and artificially reduced by multiplying a absorption function $b(x)$. Larsen and Dancy (1983) first presented this method to damp out short waves in their model. Troch and De Rouck (1998) adopted this method in their two-dimensional numerical simulation model with a VOF algorithm. Troch and De Rouck (1998) compared two different types of absorption functions and suggested that an absorption function with very small gradient in most of the sponge layer region will produce smallest reflected waves. Their absorption function is shown below and will be adopted in our numerical model:

$$b(x)=\sqrt{1-\left(\frac{x-x1}{x_s}\right)^2} \tag{3.92}$$

where x_s is the sponge layer width, and $x1$ is the initial point of the sponge layer.

The absorption function is applied in the cells of the sponge layer after each time step calculation. However, our numerical simulation shows that applying the absorption function to the vertical velocity component w and leave the other two components adjusted by the numerical model itself generates least amount of reflecting waves. Our numerical experience also shows that the sponge layer width should be:

$$\lambda \le x_s \le 1.5\lambda \tag{3.93}$$

where λ is the wave length.

From equation (3.93) the width of the sponge layer is propositional to the wave length. Thus, longer wave length requires longer sponge layer. Therefore, using sponge layer is computational expensive for dealing long waves, such as solitary waves. A combination of the advective open boundary condition and sponge layer can handle both short and long waves and thus reduces the computational costs.

The sponge layer is attractive in our three dimensional simulations because the implementation is simple and independent of the wave propagation direction, which is a big challenge when applying the active absorbing boundary condition to a three dimensional domain. Furthermore, since the sponge layer method is efficient for a very wide range of the wave lengths, only the longest wave length has to be considered in designing the sponge layer. The sponge layer acts passively, therefore, it always increases the stability of numerical simulation, while the active absorbing boundary conditions might cause the instability. However, using a sponge layer will increase the numerical domain. On the other hand, only a limited number of grid cells are required within the sponge layer. There is no harm to use a coarser grid in the sponge layer region because the numerical damping is another kind of absorbing function.

3.9.4 Numerical Experiments

In order to test the internal wavemaker as well as the sponger layer introduced in the previous two sections, a case with cnoidal wave propagation on a constant water depth is studied. The model solves the Navier-Stokes equations with the VOF method tracking the free surface. In this case, a cnoidal wave train with a wave height $H = 0.04\,\text{m}$ and wave period $T = 2\,\text{sec}$ is generated on a constant water depth $h = 0.4\,\text{m}$ by a numerical internal wavemaker which is located at the center of the flume. Given wave height H, still water depth h, and wave period T, the modulus

m of the cnoidal wave can be obtained by solving equation (3.86), (3.87), and (3.88) iteratively. Thus, the surface elevation and the internal source function can be determined. In this simulation, the modulus m of the cnoidal wave is 0.467, wave celerity $C = 1.82\,\text{m/sec}$, wave length $\lambda = 3.65\,\text{m}$. The simulation is conducted in a numerical domain with 40 m long, 0.44 m high and 0.125 m wide. Free-slip wall boundary condition is applied to each boundary wall. Uniform grid is applied in the streamwise direction with $dx = 0.125\text{m}$. Non-uniform grid is applied in the vertical direction with finest grid located at the top of the domain and $(dz)_{finest} = 0.0064$ m. The internal wavemaker region is $12.3 \le x \le 12.7\ (\text{m})$, $0 \le y \le 0.125\ (\text{m})$, and $0.27 \le z \le 0.35\ (\text{m})$ (Figure 3.9). Because the internal wavemaker is uniform in the spanwise direction and symmetric boundary conditions are used on the lateral sidewalls, no lateral movement is going to be generated. In order to save the computational time, we choose to put only one cell in the lateral discretization. The sponge layers are applied to the upstream and downstream of the channel. The left side sponge layer region is $0 \le x \le 7.5\ (\text{m})$, and the right side sponge layer region is $32.5 \le x \le 40.0\ (\text{m})$ (Figure 3.9). In this case, three gauges are installed at the location of $x = 20\text{m}$, $x = 25\text{m}$, and $x = 30\text{m}$. Figure 3.10 shows the snapshots of the free surface elevations. The cnoidal waves are generated by the internal wavemaker and will be absorbed by the numerical sponge layers. Figure 3.11 shows the time history free surface elevation obtained from thee wave gauges. The time histories of free surface elevation are compared to the analytical solutions. The analytical solutions are the Boussinesq solution derived by Wiegel (1960). Figure 3.11 shows that the numerical results agree with the analytical solutions very well.

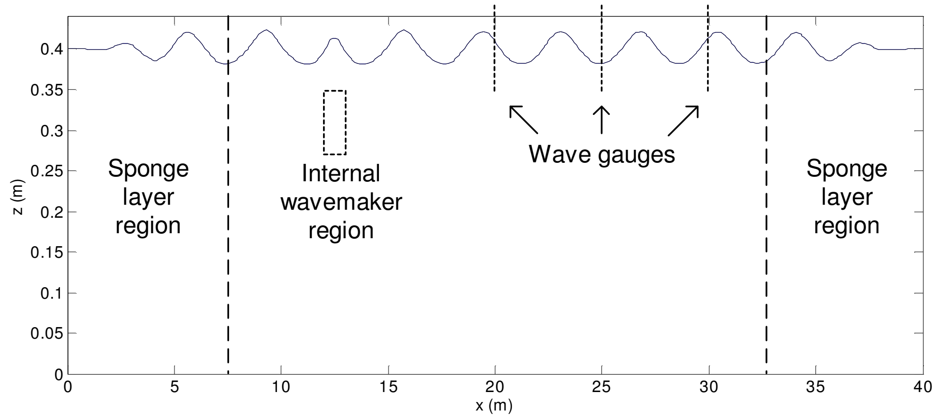


Figure 3.9: The numerical setup of cnoidal waves generated by using internal wavemaker. The cnoidal waves will be absorbed in the sponge layer region.

3.9.5 Summary

In section 3.9, we have introduced the internal wavemaker method to generate waves inside the fluid domain. By adding a mass source function to the mass and momentum equations, we are able to generate desired waves. The source functions for generating a solitary wave and cnoidal waves are presented. We also introduced the sponge layer to absorb the outgoing waves. The waves inside the sponge layer will be damped out by multiplying an absorbing function to the velocity field. An example of generating cnoidal wave inside a finite domain is demonstrated. The accuracy is discussed.

The combination of internal wavemaker and numerical sponge layer is very useful for studying many practical problems, such as the wave-structure interaction and periodic wave runup and rundown on an incline beach. These cases usually generate waves with a wide range of wave lengths which is difficult to handle by using the convectional wave generating-absorbing boundaries.

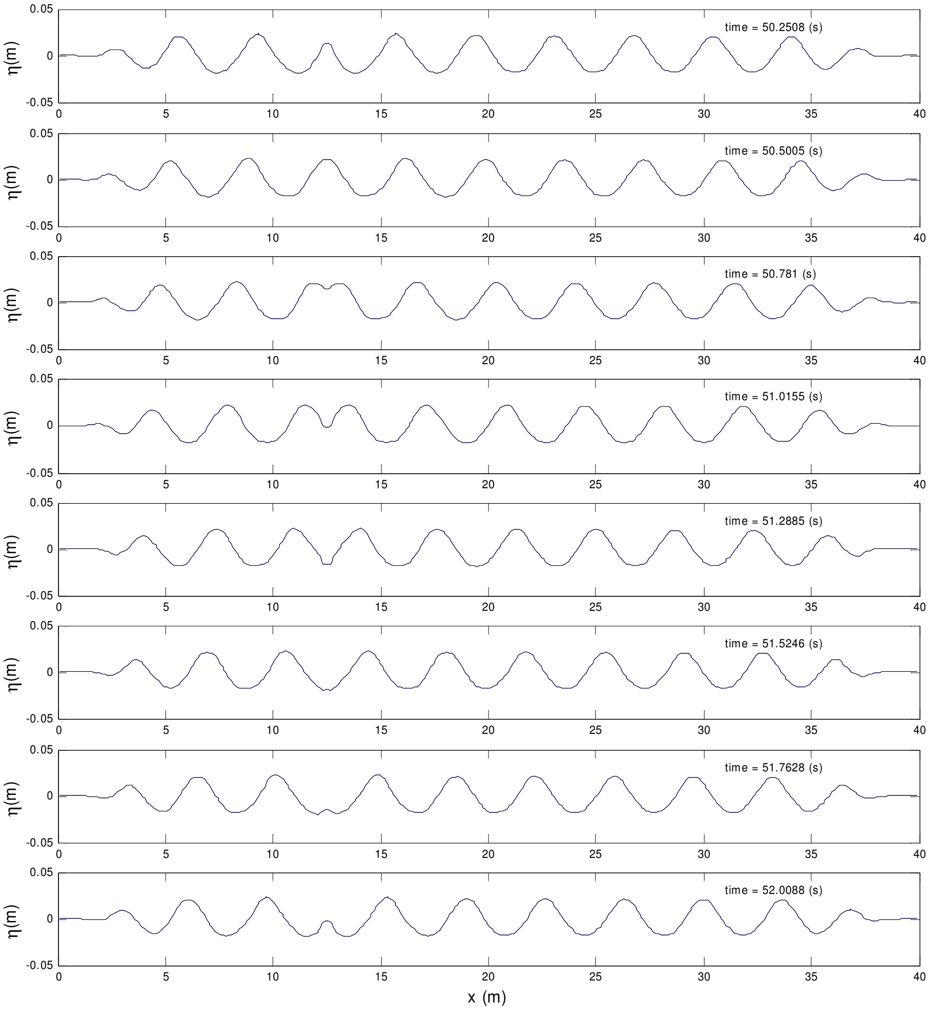


Figure 3.10: The snapshots of free surface elevations. The cnoidal waves are generated by the internal wavemaker and absorbed by the numerical sponge layers. The time sequence of snapshots cover one wave period ($T = 2.0\,\text{sec}$).

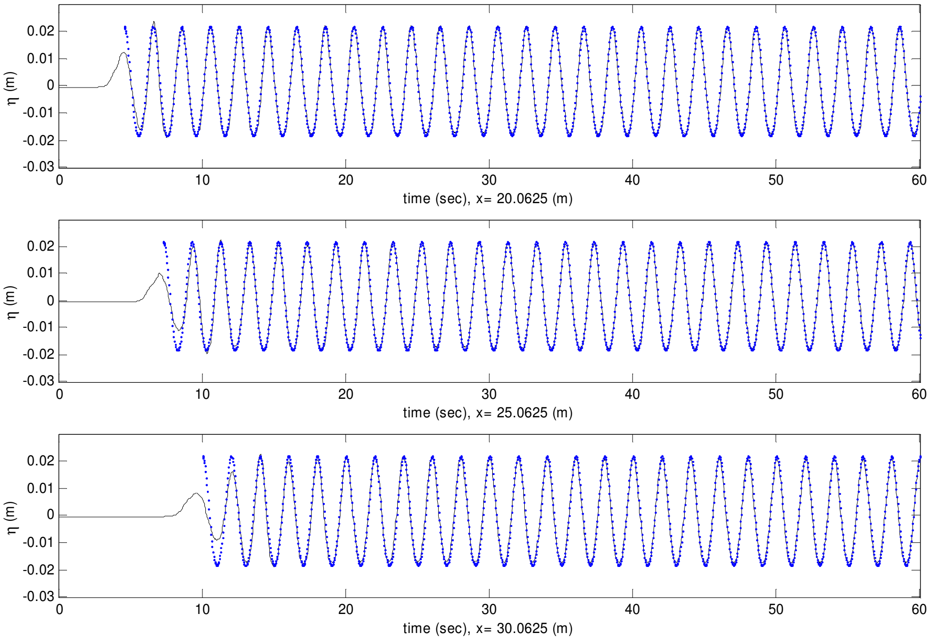


Figure 3.11: The time history free surface elevations obtain from three wave gauges. The solid lines are the numerical solutions. The dot lines are cnoidal wave analytical solution derived by Wiegel (1960).

Chapter 4

Numerical Implementation for $k-\varepsilon$ Turbulence Model

In this chapter, numerical implementations of $k-\varepsilon$ turbulence model, LES turbulence model, partial cell treatment, and moving solid algorithm will be presented. At the end of each section, a test case will be shown and discussed.

4.1 Discretizations for $k-\varepsilon$ Equations

In chapter 2, we have introduced the Reynolds-Averaged Navier-Stokes (RANS) equations (Equation 2.25). If the linear eddy viscosity assumption is applied, the normal stress of the turbulence can be converted into pressure, Equation (2.25) then can be expressed as:

$$\frac{\partial\left(\rho\left\langle u_i\right\rangle\right)}{\partial t}+\frac{\partial\left(\rho\left\langle u_i\right\rangle\left\langle u_j\right\rangle\right)}{\partial x_j}=\frac{\partial}{\partial x_j}\left[\mu_{eff}\left(\frac{\partial\left\langle u_i\right\rangle}{\partial x_j}+\frac{\partial\left\langle u_j\right\rangle}{\partial x_i}\right)\right]-\frac{\partial P}{\partial x_i}+\rho g_i \tag{4.1}$$

where $P=\left\langle p\right\rangle+{}^2\!/_3\rho k$, $\mu_{eff}=\mu+\mu_t$.

Comparing RANS equations (Equation (4.1)) to NS equations (Equation (2.8)), we observe that the structure of both sets of equations is the same except that the velocity in NS becomes mean velocity, pressure becomes effective pressure, and viscosity become effective viscosity in RANS. As for numerical implementations, the projection method (Chapter 3.4) can be directly used to solve RANS equations.

As for the nonlinear (quadratic) eddy viscosity model, the turbulence Reynolds stress tensor $\rho\left\langle u_i^{'}u_j^{'}\right\rangle$ will be calculated directly from the mean flow field based on Equation 2.43 and added to the original RANS equations (Equation 2.23).

In order to solve the turbulence viscosity, two additional equations of turbulence kinetic energy k and turbulence dissipation ε have to be solved. In the present solution algorithm, the time-advanced velocity field $\mathbf{u}^{n+1}$ is already available when the $k-\varepsilon$ equations are updated. Therefore, the production term P_k can be computed on the $n+1$ time level.

Based on the Gauss divergence theorem introduced in Chapter 3.1, the finite-volume discretized ε equation can be written as:

$$
\begin{aligned}
\frac{\partial}{\partial t}(\rho\varepsilon) &= \nabla\cdot\left[-\left(\langle\mathbf{u}\rangle\rho\varepsilon\right)+\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\nabla\varepsilon\right]+\frac{\varepsilon}{k}\left(C_{\varepsilon1}\rho P_k-C_{\varepsilon2}\rho\varepsilon\right)\\
&= \nabla\cdot\boldsymbol{\tau}_\varepsilon+\frac{\varepsilon}{k}\left(C_{\varepsilon1}\rho P_k-C_{\varepsilon2}\rho\varepsilon\right)\\
&(\text{for each cell})\\
&\Rightarrow\frac{\rho^{n+1}\varepsilon^{n+1}-\rho^n\varepsilon^n}{\delta t}=\left(\frac{1}{V}\sum_{f=1}^{nfc}\left\langle\boldsymbol{\tau}_\varepsilon\cdot\mathbf{A}\right\rangle_f^n\right)+\frac{\varepsilon^{n+1}}{k^n}\left(C_{\varepsilon1}\rho^{n+1}P_k^{n+1}-C_{\varepsilon2}\rho^n\varepsilon^n\right)\\
&\Rightarrow\varepsilon^{n+1}=\frac{1}{\dfrac{\rho^{n+1}}{\delta t}-\dfrac{\left(C_{\varepsilon1}\rho^{n+1}P_k^{n+1}-C_{\varepsilon2}\rho^n\varepsilon^n\right)}{k^n}}\left(\frac{\rho^n\varepsilon^n}{\delta t}+\left(\frac{1}{V}\sum_{f=1}^{nfc}\left\langle\boldsymbol{\tau}_\varepsilon\cdot\mathbf{A}\right\rangle_f^n\right)\right)\\
&\text{where }\boldsymbol{\tau}_\varepsilon=-\left(\langle\mathbf{u}\rangle\rho\varepsilon\right)+\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\nabla\varepsilon
\end{aligned}
\tag{4.2}
$$

Similarly, we can apply the Gauss divergence theorem to the k equation:

$$
\begin{aligned}
\frac{\partial}{\partial t}(\rho k) &= \nabla\cdot\left[-\left(\langle\mathbf{u}\rangle\rho k\right)+\left(\mu+\frac{\mu_t}{\sigma_k}\right)\nabla k\right]+\rho P_k-\rho\varepsilon\\
&= \nabla\cdot\boldsymbol{\tau}_\varepsilon+\rho P_k-\rho\varepsilon\\
&(\text{for each cell})\\
&\Rightarrow\frac{\rho^{n+1}k^{n+1}-\rho^n k^n}{\delta t}=\left(\frac{1}{V}\sum_{f=1}^{nfc}\left\langle\boldsymbol{\tau}_k\cdot\mathbf{A}\right\rangle_f^n\right)+\left(\rho^{n+1}P_k^{n+1}-\rho^{n+1}\varepsilon^{n+1}\right)\\
&\Rightarrow k^{n+1}=\frac{\delta t}{\rho^{n+1}}\left(\rho^n k^n+\left(\frac{1}{V}\sum_{f=1}^{nfc}\left\langle\boldsymbol{\tau}_k\cdot\mathbf{A}\right\rangle_f^n\right)+\rho^{n+1}P_k^{n+1}-\rho^{n+1}\varepsilon^{n+1}\right)\\
&\text{where }\boldsymbol{\tau}_\varepsilon=-\left(\langle\mathbf{u}\rangle\rho k\right)+\left(\mu+\frac{\mu_t}{\sigma_k}\right)\nabla k
\end{aligned}
\tag{4.3}
$$

For advection terms, the face fluxes of $\langle\mathbf{u}\rangle\rho k$ and $\langle\mathbf{u}\rangle\rho\varepsilon$ are evaluated based on the upwind scheme where the mass flux, $\langle\mathbf{u}\rangle_f^n\rho$, on each face is obtained directly from the VOF calculations. As for the diffusion terms, the cell face viscosities are interpolated from the ambient cell-centered viscosities. Therefore, a scheme similar to the central-difference scheme is applied to the viscosity calculations.

Equations (4.2) and (4.3) show that the cell centered advection and diffusion are evaluated by summing all fluxes through each face. The new time step ε^{n+1} is updated first, and $\left(C_{\varepsilon 1}\rho P_k - C_{\varepsilon 2}\rho\varepsilon\right)$ is treated in a semi-implicit way. In the k equation, the production and dissipation terms use the $n+1$ level information. Lemos (1992) found that when the values of P_k and ε at the advanced time level are used the performance of the model is improved in situation where the P_k - ε term dominates. Therefore, this semi-implicit scheme is suitable for current study.

No-flux boundary conditions ($\partial k/\partial n = 0$ and $\partial \varepsilon/\partial n = 0$) will be applied to the air-water interface cells when summing the face fluxes for the diffusion terms. The Law-of-the-Wall boundary condition will be applied to those cells adjacent to the impermeable walls, and will be used to update the cell-centered values of k^{n+1} and ε^{n+1}. We shall discuss the detail implementations of Law-of-the-Wall boundary condition in the next section. The cell face gradients, such as $\left\langle \nabla k \right\rangle_f$ and $\left\langle \nabla \varepsilon \right\rangle_f$, are evaluated by the Least square linear reconstruction (LSLR) introduced in Chapter 3.2. We remark that if the air density is set to zero, the air part data-dependent weight w_k^d is also zero to avoid using any information from air.

4.2 Implementation of Law-of-the-Wall Boundary Condition

The price of resolving the turbulent boundary layer is too high. One alternative way is to apply the law-of-the-wall boundary condition to the cells adjacent to a solid wall. The k and ε values are utilized at the location within the log-law region of the turbulent boundary layer, rather than on boundary faces. Equation (2.44) is used to calculate the friction velocity u_τ based on the known cell-centered mean tangential velocity $\langle u_T \rangle$ and the distance y between cell centroid and solid wall. The friction velocity u_τ can be solved iteratively by using the Newton-Raphson method:

$$X^{n+1} = X^n + \frac{\kappa - X^n \left(A + \ln X^n \right)}{1 + \frac{\kappa}{X^n}} \tag{4.4}$$

where the superscript n is the iteration number, κ is the von Karman constant ($\kappa = 0.41$), $X = u_\tau / \langle u_T \rangle$, and $A = \ln\left(E \rho u_\tau y / \mu \right)$, where y is the distance between cell centroid and solid wall and E is an empirical constant ($E \approx 9.0$).

After solving the friction velocity u_τ, the values of k, ε, and ν_t can be obtained by equation (2.51), (2.49), and (2.50), respectively.

4.3 Example 1: 2D Spilling Breakers

4.3.1 Introduction

Due to the mixing mechanism, breaking waves may cause many problems in the coastal region, such as beach erosion, sand drift, and damage to coastal structures. Breaking waves are generally classified as spilling, plunging, surging, and collapsing breakers. For regular incident waves on a plane beach, the breaker types depend primarily on the incident wave height, wave period and beach slope (Galvin, 1968; Goda, 1970). These breaker types can be characterized by the surf zone similarity parameter ξ_0 (Battjes, 1974), defined as below:

$$\xi_0 = \frac{\tan \beta}{\sqrt{H / \lambda}} \tag{4.5}$$

where $\tan\beta$ is the beach slope. Based on Galvin's experimental analysis, the break types can be classified as follow:

$$\begin{aligned} &\xi_0 < 0.5 && :\text{spilling breakers} \\ &0.5 < \xi_0 < 3.3 && :\text{plunging breakers} \\ &\xi_0 > 3.3 && :\text{surging breakers} \end{aligned} \tag{4.6}$$

This study focuses on the spilling breakers which can be seen in most surf zones. In the early studies, many researchers have carefully described the generation processes of the spilling breaker. Battjes (1988) found that a spilling breaker experiences a gradual change in shape and remains highly symmetric about its crest until breaking. Sorensen (1993) further described that spilling breakers typically occur on mildly sloping beaches and are characterized by turbulence and foaming at the crest and spilling down the front face of the wave and forming a whitecap. Basco (1985) had a detailed description for the flow at the wave crest. He found that the fluid particles at the crest of the wave accelerate to a speed greater than the wave celerity and then falls forward in the front of a small jet which is often obscured by foam and often not visible. Due to the limitation of the facilities, the early studies are mainly focused on the quantitative description of the breaking waves.

In the last few decades several experimental methods have been developed to measure the velocities accurately within a wave. Nadaoka et al. (1989) used a fiber-optic laser Doppler velocimeter (LDV) to study the turbulent flow field under a breaking wave on a 1:20 slope. Ting and Kirby (1995, 1996) employed LDV and wave gauges for plunging and spilling breaker on a 1:35 slope. They obtained high quality velocity and turbulence structure data in both type of breakers. In fact, Ting and Kirby's spilling breaker data will be used as the reference data in this study. Newer technology, like particle image velocimetry (PIV) can provide full velocity field. Chang and Liu (1999) used PIV to investigate turbulence in a spilling breaker, and characterized the importance of each term in the turbulence kinetic equation during the breaking process.

However, PIV as well as LDV measurements have the difficulty to measure the flow properties in the aerated crest region since the presence of foam and bubbles. Information in the crest region is important for estimating the radiation stresses and energy dissipation parameters used in the depth-integrated wave models. For these reasons, numerical studies have become popular because numerical solutions can provide details of the flows. Before wave breaking, the flow field is irrotational. Therefore the potential theory can apply (Peregrine, 1983; Battjes, 1988). Many numerical models have successfully modeled the shape of a wave crest in a plunging breaker, prior to its impingement on the free surface. However, during and after wave breaking, the flow field becomes highly rotational. Therefore, numerical models solving the Navier-Stokes equations have been developed. Lemos (1992) simulated periodic breaking waves on a sloping beach but he didn't present quantitative information on the accuracy of the predicted breaking wave characteristics. Lin and Liu (1998a,b) gave a profound quantitative evaluation of their numerical model in the surf zone. Their solutions have been carefully validated with the data measured by Ting and Kirby (1995, 1996) by using laser-Doppler velocimeter. In Lin and Liu's studies, a first order accurate (in time and space) finite difference numerical method was utilized to solve the Reynolds averaged Navier-Stokes equations in conjunction with a Reynolds stress model. Bradford (2000) studied the same spilling breaking case to evaluate the one and two equation turbulence models. In Bradford's study, the breaker height, location and undertow were examined.

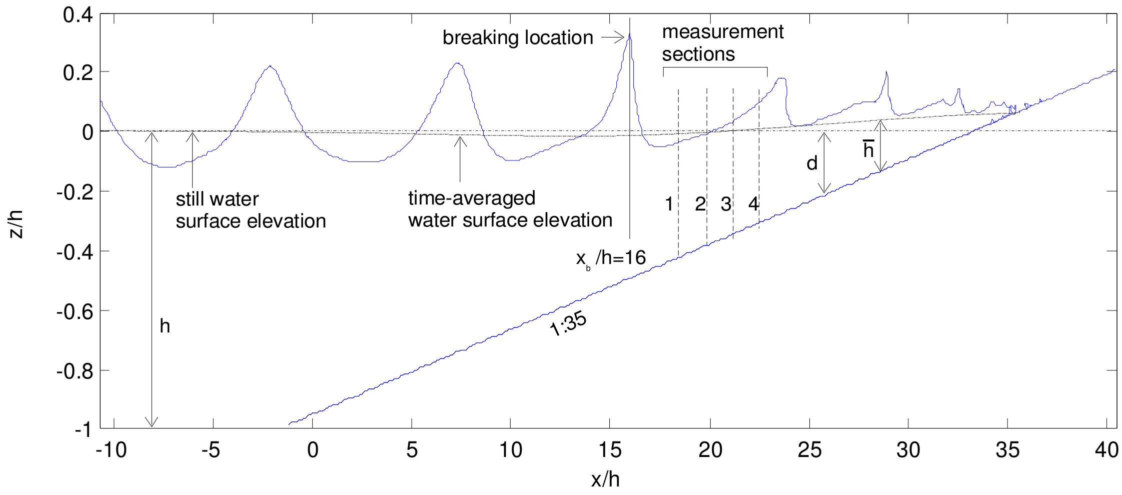


Figure 4.1: Sketch of experimental setup and computational domain. The still water depth is represented by h and the time-averaged water depth is represented by $\bar{h} = d + \bar{\eta}$, where d is local still water depth, $\bar{\eta}$ is the time-averaged water surface elevation. The beach slope is 1/35 and is connected to a constant water depth $h = 0.4$ m. The breaking location is at $x_b / h = 16.0$ and four measurement cross-sections are located at $(x - x_b)/h_b = 4.397$, 7.462, 10.528, and 13.618, where $h_b = 0.498h$ is the time-averaged water depth at the breaking location.

In this chapter, an approach similar to Lin and Liu (1998a,b) is used to model the surf zone dynamics. The $k - \varepsilon$ turbulence model is validated for spilling breakers over a sloping bed. The volume of fluid method is used to track the highly complex, discontinuous free surface. The solutions will be compared with the same laboratory data (Ting and Kirby, 1995; 1996) to examine the performance of the turbulence model. The simulation results are compared with laboratory measurements of the mean water level, trough and crest heights, and time-history turbulence kinetic energy in order to evaluate the capability of each approach in computing surf zone dynamics. Details of the breaking wave properties are presented and discussed.

4.3.2 Governing Equations

The governing equations used in this study are the Reynolds averaged Navier-Stokes equations (Equation 2.22 and 2.23) because RANS equations allow us to solve the breaking waves on a relative coarse grid (compared to DNS) and still consider the effects from turbulence. The nonlinear $k-\varepsilon$ turbulence model (Equation (2.37), (2.38), (2.39), and (2.44)) is used to simulate the complex turbulent flow. The movement of the free surface is tracked by using the volume of fluid (VOF) method (Equation (3.24)).

4.3.3 The Experimental and Numerical Setups

Experiments of spilling wave train on a slope were performed by Ting and Kirby in 1994 (Ting and Kirby, 1994; 1996). In their wave tank, a beach with a slope of 1/35 is connected to a region with a constant water depth $h = 0.4$ m. The coordinate system is chosen so that $x = 0$ is located at the position where the still water depth is $d_0 = 0.38$ m (see Figure 4.1). The incident waves are cnoidal waves with wave height $H = 0.125$ m in the constant water depth region and wave period $T = 2.0$ sec. The velocities and free surface displacement are measured at four vertical cross sections shoreward of the breaking point, which is located at $x_b = 6.4$m with time-averaged water depth $h_b = 0.199$ m. The data measured at $\left(x - x_b\right)/h_b$=4.397, 7.462, 10.528, and 13.618 (see Figure 4.1) are used for comparisons with the numerical results.

In the numerical simulations, the computational domain starts at $x = -4.3$ m and ends at $x = 16.2$ m. The numerical domain is 20.5 m long, 0.56 m high, 0.01 m wide, and is discretized by 800 x 1 x 70 cells with uniform grid in the streamwise direction (x-direction), $\Delta x = 0.0256$ m, and non-uniform grid in the vertical direction (z-direction). The fine grids with the minimum $\Delta z_{\min} = 0.0055\text{m}$ are placed near the top of the domain. The waves are assumed to approach perpendicularly to the beach. Therefore, the problem may be treated as 2D. In this simulation, the spanwise direction (y direction) is discretized into one cell, and free-slip wall boundary condition is applied. The time step is automatically adjusted during the computation to satisfy the stability constraints.

Cnoidal waves are input into the numerical domain by specifying the free surface elevation, η, and the cell-face velocities at the left boundary. The expressions of cnoidal waves derived by Wiegel (1960) are used to determine time dependent velocities and free surface elevation by specifying the wave period, T, wave height, H, and still water depth h. In addition, k and ε also specified at the boundary (see Chapter 2.2.3).

At the flat bottom, free-slip boundary condition is used. On the slope, the partial cell treatment and law-of-the-wall turbulence boundary conditions are implemented. On top of the numerical domain, $p = 0$ type open boundary condition (Chapter 3.6.3) is adopted. At the free surface, the pressure on the cell faces adjacent to air is set to zero (Chapter 3.3.3). Velocities on the free surface cells are extrapolated from the interior so the resulting tangential shear stress is zero. Advective and diffusive fluxes of k and ε are set to zero.

The initial condition for velocity field is zero everywhere. The initial values for k and ε are determined by the empirical procedures introduced in Chapter 2.2.2. In this study, the initial value of k is $k_{ini} = 7.8\times10^{-7}\text{m}^2/s^2$, and the initial value of ε is $\varepsilon_{ini} = 4.39\times10^{-8}\ \text{m}^2/\text{s}^3$. The turbulence quantities are very low and $\nu_t = 1.25\times10^{-6}\text{m}^2/\text{s} \approx \nu$. The simulations were found to be insensitive to the initial values of k and ε if $k \geq \varepsilon$. Same values of k, ε, and ν_t are used as the boundary conditions at the left boundary.

In the experiments, the measurements are taken 20 minutes after the first wave is generated, at that moment the waves have reached the quasi-steady state. The mean velocities and free surface displacement were computed by performing the phase average of the measured data. The phase-averaged quantities can be treated as the ensemble-averaged quantities and used to examine the numerical solutions. However, the turbulence field calculated by using phase-averaged method may be affected by random waves during the breaking process. Also, in the experiments, only two components of turbulence velocities are measured. Therefore, the turbulence kinetic energy has to be estimated by the empirical formula provided by Svendsen (1987), i.e., $k = \frac{2}{3}\left(\left\langle u'^2 \right\rangle + \left\langle w'^2 \right\rangle\right)$ where $\left\langle u'^2 \right\rangle^{1/2}$ and $\left\langle w'^2 \right\rangle^{1/2}$ are turbulence velocities in the horizontal and vertical direction, respectively. This approximation may introduce some uncertainties in the following comparisons.

The numerical simulation is run for 20 s, which allows 10 complete waves into the domain. The simulation time is chosen long enough to allow the waves to reflect off the beach and establish a return flow. However, the simulation can not be run for an arbitrarily long time since the specified cnoidal waves transports fluid into the domain, which eventually cause the mean water level to rise. The numerical solutions used in the following comparisons are from $t' = 18.2\text{s}$ to $t' = 20.2\text{s}$ with $t' = 0.0\text{s}$ representing the beginning of the numerical solution. This particular time $t' = 18.2\text{s}$ matches the time $t/T = 0$ in the phase-averaged experimental results.

4.3.4 Comparison of Experimental Data and Numerical Results

Numerical results were compared with laboratory data measured by Ting and Kirby

(1994). Some notations adopted in this thesis are presented as follow: The definition of mean (time-averaged) free surface elevation is:

$$\overline{\eta} = \frac{1}{T}\int_0^T \eta dt \tag{4.7}$$

where $\overline{\eta}$ is the time-averaged free surface elevation over one wave period T. η is the instantaneous water surface elevation. The local still water depth is represented by d, and the local time-averaged water depth is represented by $\overline{h} = d + \overline{\eta}$.

The simulation results of the mean velocities and free surface profiles indicate that the waves in the surf zone nearly, but not completely, reach the quasi-steady state. This situation suggests that the setup and setdown has not fully developed and the calculated mean water depth can be inaccurate. However, the major wave and turbulent characteristics under the breaking waves are captured by the numerical simulations.

Figure 4.2 to 4.5 show the comparisons of the mean free surface displacement, mean velocity, and turbulence intensity at four vertical cross sections in the surf zone. At each location, comparisons are made at four different elevations below the trough level for the mean velocities and turbulence intensity. In these figures, the left columns are laboratory measurements and the right columns are numerical results. Free surface displacements are normalized by the local time-averaged water depth $\overline{h}$. The mean velocities and turbulence intensities are normalized by the local phase velocity $C = \sqrt{g\overline{h}}$. The measured $\overline{h}$ is used in the normalization for both laboratory data and numerical results.

Figure 4.2 shows the comparison at $\left(x - x_b\right)/h_b = 4.397$, which is the measurement section closest to the breaking point. The comparisons of mean free surface and velocities are reasonably good (Figure 4.2 (a), (b), (c)). However, the numerical solution significantly overestimate the turbulence intensity. The pattern of the turbulence intensity in the simulation results (Figure 4.2(d)) is also different from that in the measurements (Figure 4.2(D)). This situation was also reported by Lin and Liu (1998a,b). The experimental data show no obvious correlation between the passage of the broken waves and the turbulence intensity. This observation is very different from the turbulence characteristics at other measurement sections in the surf zone (Figure 4.3(D), 4.4(D), and 4.5(D)).

Other researchers have found that the $k-\varepsilon$ model tends to overestimate the turbulence level near the breaking point (Lin and Liu, 1998a,b). The primary reason is that the turbulent closure model with the coefficients derived from quasi-steady turbulent flows cannot accurately predict the initiation of turbulence in a strong transient flow such as the initial phase of the breaking wave (Lin and Liu, 1998a,b). Fortunately, due to the turbulence dissipation mechanisms and the oscillatory nature of the mean flow under the breaking wave, the inaccuracy in the early stage of the simulation has little effect on the downstream locations.

Figure 4.3 shows the comparisons at $\left(x-x_b\right)/h_b=7.462$. The agreement between the numerical simulations and the laboratory measurements is better than those shown in Figure 4.2. Very good agreement can be seen in terms of the mean free surface elevation. In terms of the turbulence field, not only the turbulence pattern but also the turbulence intensity are accurately predicted by the numerical model. Both numerical and experimental results show that the turbulence intensity in the upper level is highly correlated to the surface profile.

Figure 4.4 and 4.5 show the comparisons at the cross sections $\left(x-x_b\right)/h_b=$ 10.528 and $\left(x-x_b\right)/h_b=13.618$, respectively. Again, the numerical simulations compare well with the laboratory measurements in terms of the mean free surface elevations, velocities and the pattern of the turbulence intensities.

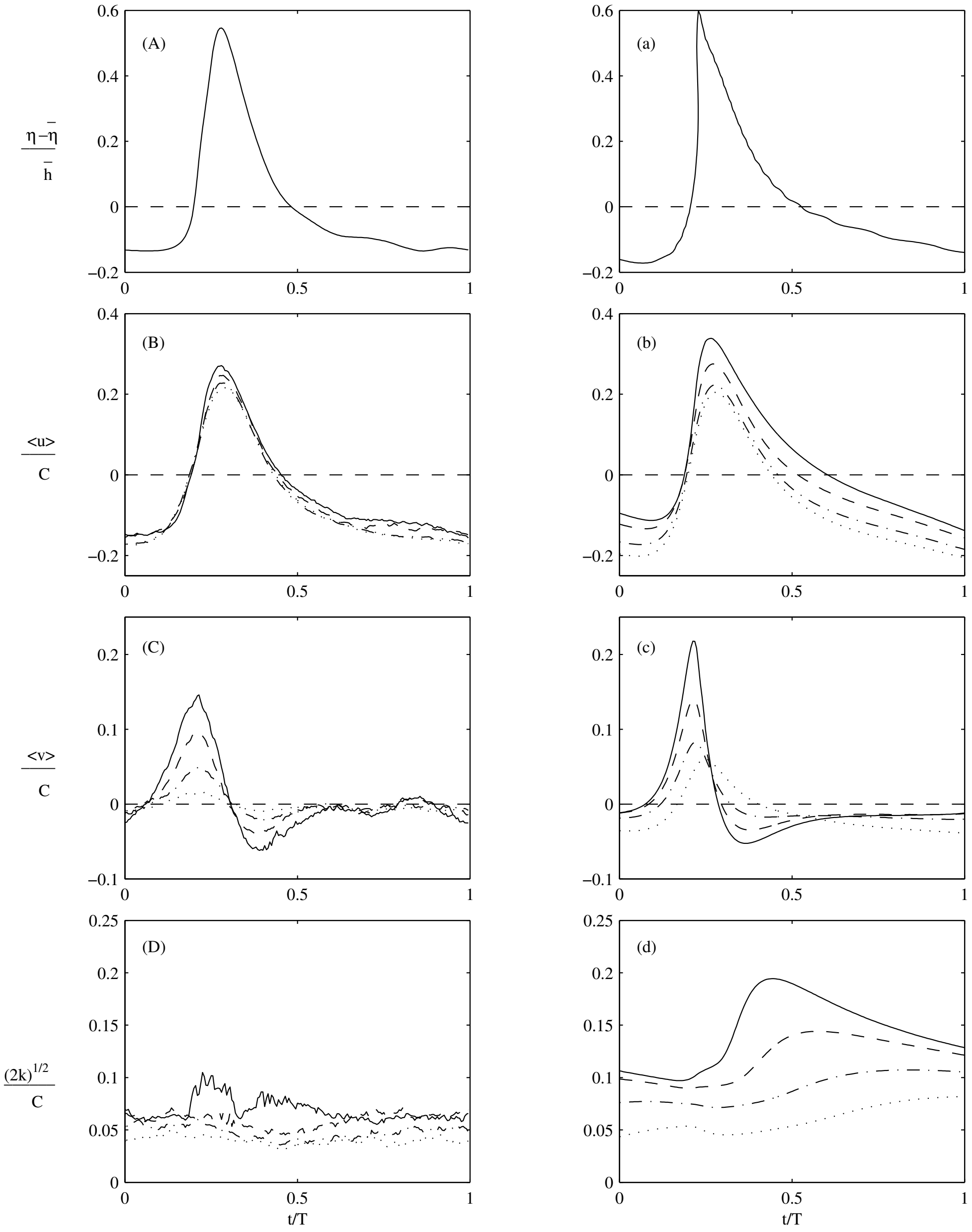


Figure 4.2: Comparisons of experimental data (A-D) and numerical results (a-d) at $(x-x_b)/h_b$ =4.397; $(y-\overline{\eta})/\overline{h}$ = -0.2623 (——), -0.4909 (– –), -0.7194 (–·–), -0.9080 (-------).

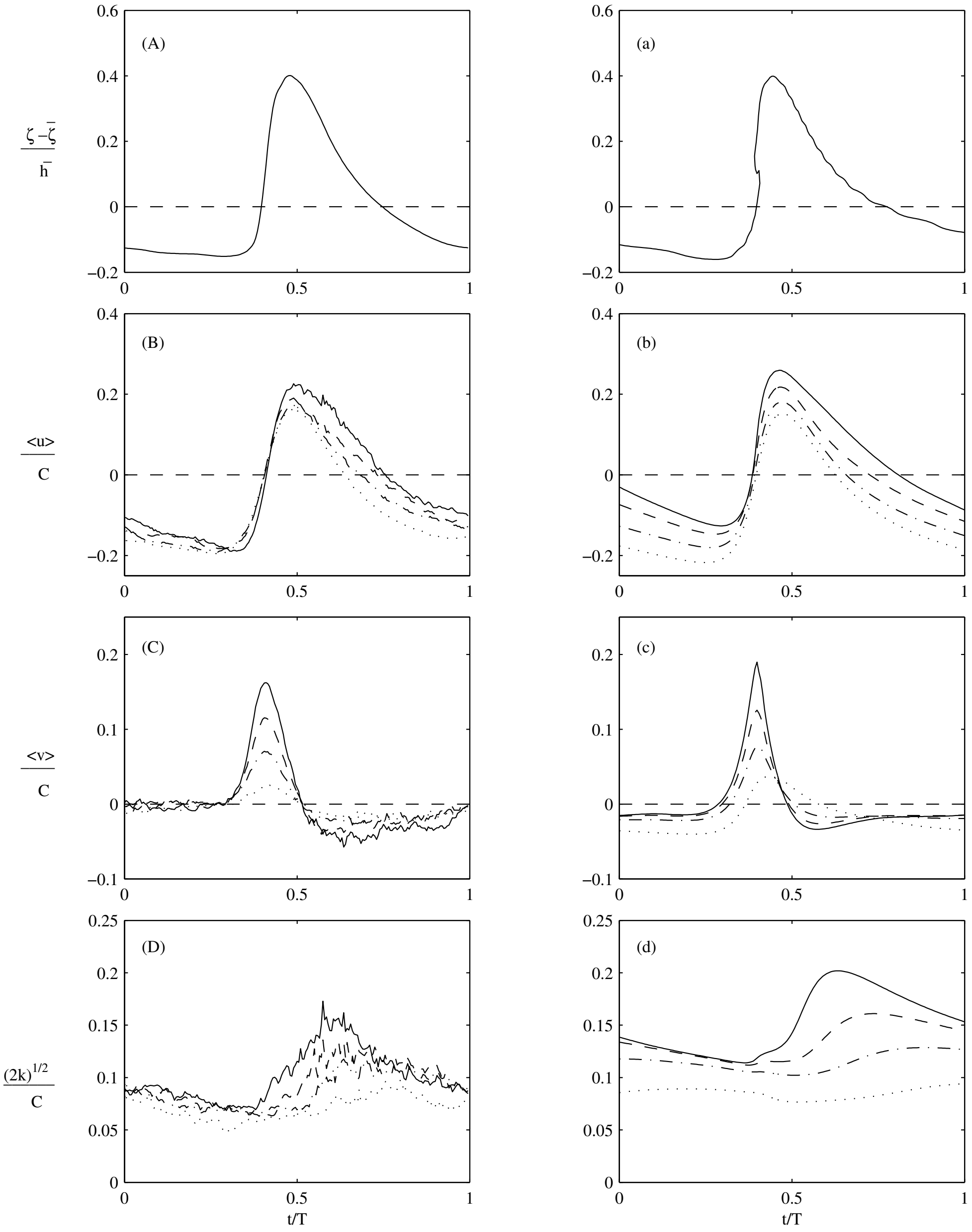


Figure 4.3: Comparisons of experimental data (A-D) and numerical results (a-d) at $(x-x_b)/h_b$ =7.462; $(y-\overline{\eta})/\overline{h}$ = -0.2957 (——), -0.4820 (– –), -0.6683 (– · –), -0.8857 (-------).

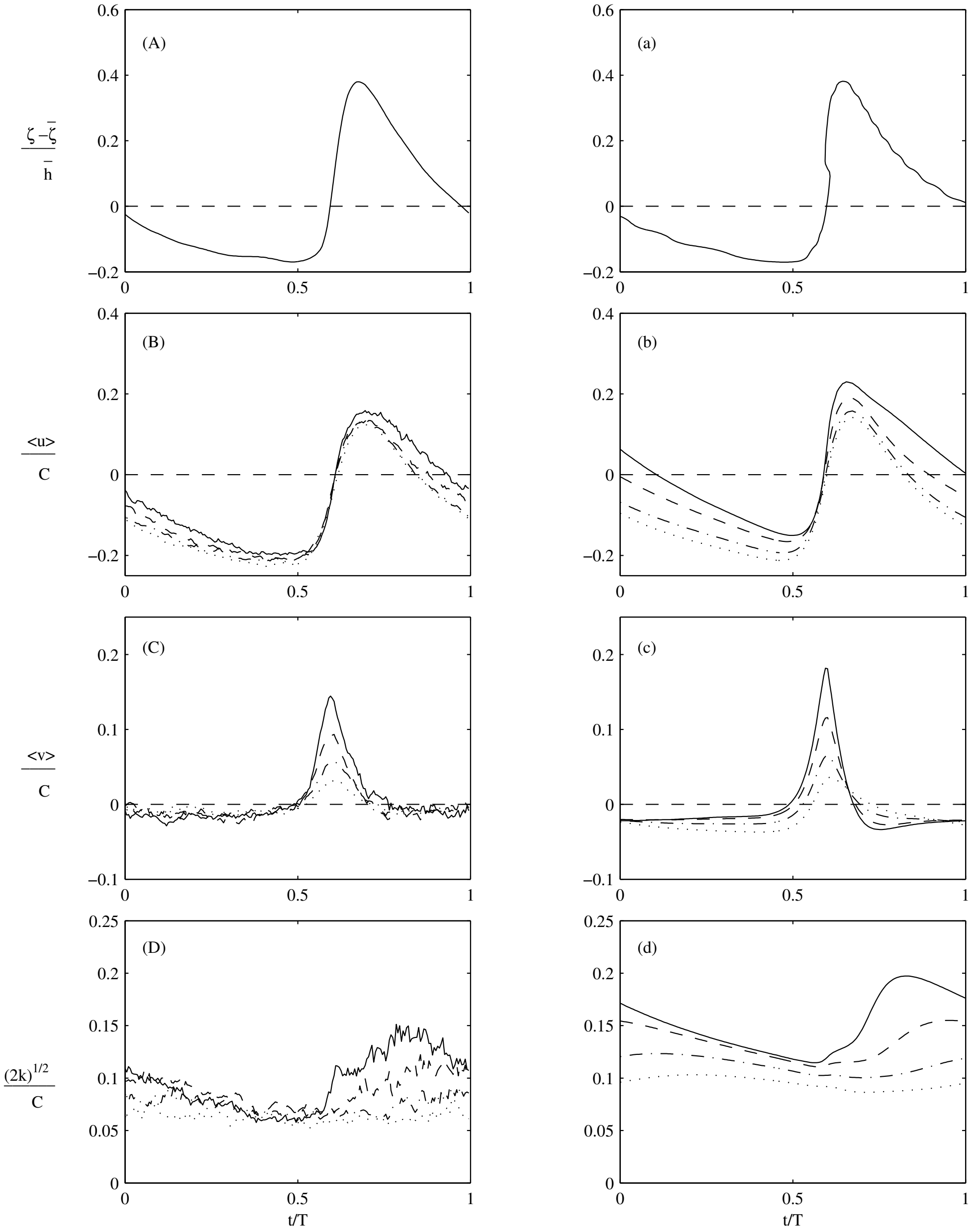


Figure 4.4: Comparisons of experimental data (A-D) and numerical results (a-d) at $(x-x_b)/h_b$=10.528; $(y-\overline{\eta})/\overline{h}=$ -0.2736 (——), -0.4764 (– –), -0.6791 (–·–), -0.8142 (-------).

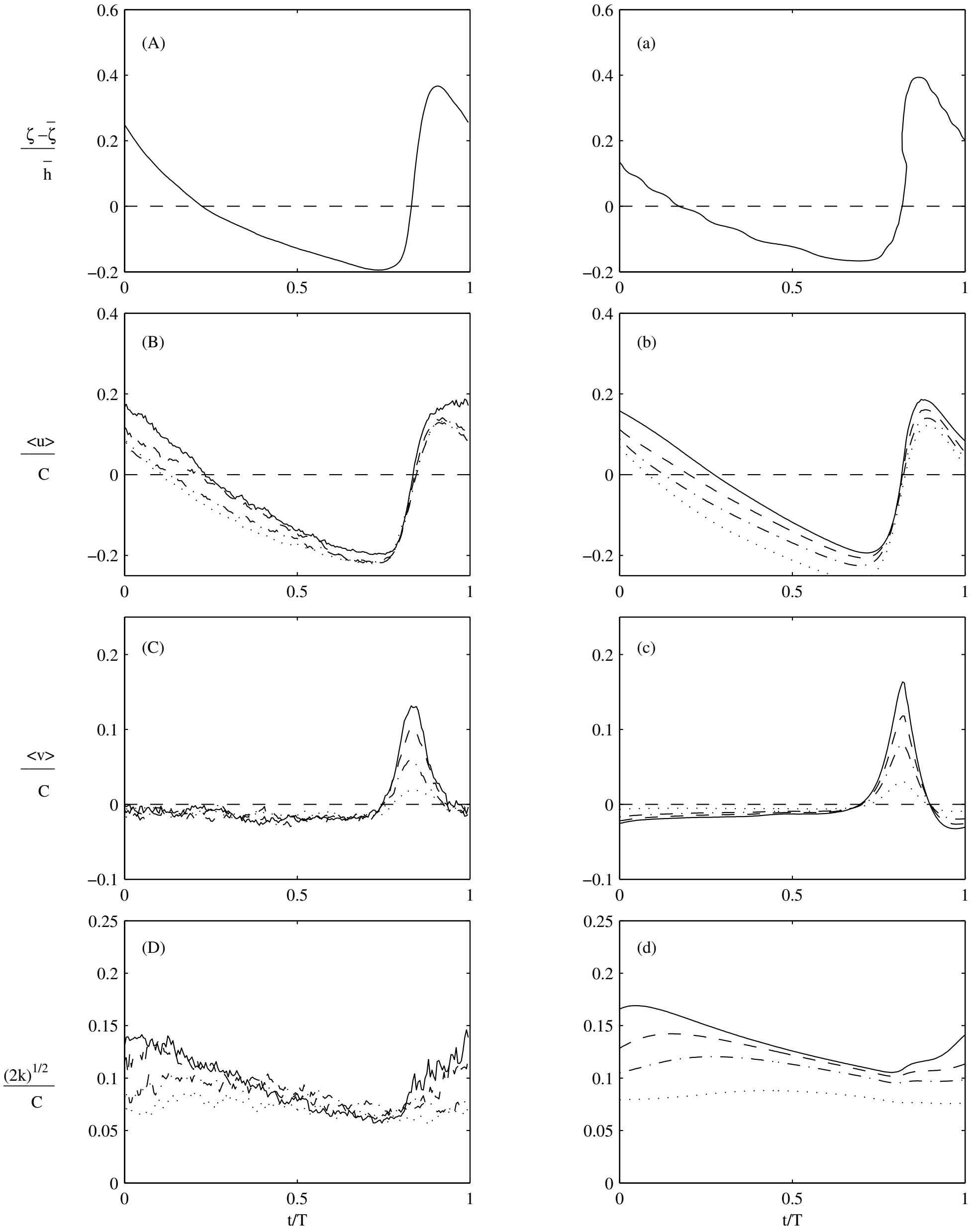


Figure 4.5: Comparisons of experimental data (A-D) and numerical results (a-d) at $(x-x_b)/h_b$=13.618; $(y-\overline{\eta})/\overline{h}=$ -0.3226 (——), -0.4729 (– –), -0.6233 (–·–), -0.8489 (-------).

Table 4.1: The comparisons of breaking wave heights and breaking locations.

	Laboratory Measurement (Ting and Kirby, 1994)	RNG model (Bradford, 2000)	Linear $k-\varepsilon$ model (Bradford, 2000)	Present nonlinear $k-\varepsilon$ model
x_b / h	16.0	11.83	12.78	16.0
A_b / h	0.41	0.36	0.35	0.33

4.3.5 Spatial Distribution of Turbulent Characteristics

One interesting topic is the determination of the breaking location and the breaking wave height. The breaking location is defined as the point at which the wave breaks. Ting and Kirby (1994) reported the breaking wave height (A_b) in the experiment is $A_b / h = 0.41$, at the location $x_b / h = 16.0$. Bradford (2000), who carried out numerical simulations with different turbulent closure models, reported that the breaking locations estimated by the linear Renormalized Group (RNG) $k-\varepsilon$ model and standard linear $k-\varepsilon$ model occurred much earlier than the laboratory measurements. However, the current nonlinear $k-\varepsilon$ turbulence model accurately predicts the breaking location (see Table 4.1). However, all the numerical models under-estimate the breaking wave height. This is because the turbulence levels near the breaking point are overestimated (see Figure 4.2(d)). These overestimations lead to exaggerated values of eddy viscosity ν_t and hence reduce the wave crest.

One advantage of the numerical simulation is its ability to display the mean velocity field and the turbulence field in the spatial domain. In this section, we shall present the spatial distributions of turbulent characteristics. In Figure 4.6, the spatial distributions at $t/T = 0.0$, 0.3, 0.6, and 0.9 of the normalized turbulent kinetic energy $k / g\left(h+a\right)$, turbulence dissipation rate $\varepsilon / g\sqrt{g(h+a)}$, mean vortices $\omega / \sqrt{g/\left(h+a\right)}$, turbulence production $P_k / g\sqrt{g\left(h+a\right)}$, and eddy viscosity ν_t / ν

are presented. In this case, a = 0.078 m and is the incident wave height evaluated from the still water. From Figure 4.6, the highest turbulence kinetic energy appears in the so-called “roller” region for the entire wave period. The “roller” has been defined as the aerated area of recirculating flow in the front of the turbulent bore (Battjes, 1988). In the “roller” region, the horizontal fluid particle velocity is roughly equal to the local phase velocity C. This “roller” concept has been adopted by Schaffer et al. (1993) in their depth-integrated numerical model to represent the wave breaking effects. Lin and Liu (1998a,b) further described the details of velocity profile in the “roller” region by solving RANS equations. Figure 4.6 shows that the pattern of turbulence dissipation rate (ε) is very similar to the pattern of turbulent kinetic energy (k). The highest turbulence dissipation also happens in the roller region. It indicates that the energy is mainly dissipated in the roller region. The vorticity (ω) is generated at the region lower than the roller region. Compared to the pattern of k and ε, the pattern of ω shows that ω is highly concentrated in the region very close to breaking wave fronts. The vortices mainly concentrate in the breaking wave fronts and move with the breaking wave regions, while the turbulences are left on the lee side of the waves. The fourth plot in Figure 4.6 is the normalized turbulence production (P_k). The plot reveals the sources of the turbulence kinetic energy. Similar to ω, P_k is highly concentrated in the region close to the breaking wave fronts. Turbulence production indicates the location of the turbulence source, and due to the transport and dissipation, the turbulence kinetic energy is no longer concentrated in the same region as the maximum turbulence production region and is left behind the wave crests. The fifth plot is the normalized eddy viscosity. The eddy viscosity ν_t is calculated based on the conventional eddy viscosity definition $\nu_t = C_d\left(k^2/\varepsilon\right)$. Similar to the turbulence kinetic energy, the largest eddy viscosity is in the roller region. After the passage of the wave crest, the eddy viscosity will be left behind the wave and gradually decayed by the turbulence dissipation mechanism. One obvious phenomenon is that the eddy viscosity decreases faster in the onshore region than the turbulence kinetic energy. This is because eddy viscosity is a function of turbulence length scale l and turbulence intensity $\sqrt{2k}$, i.e., $\nu_t \sim lk^{1/2}$, not only k but also l decrease as the water depth decreases.

Figure 4.6 also shows that the spilling breaking process is a local phenomenon. The effects introduced by the breaking waves decay significantly to the bottom. From Figure 4.2 (Dandd) ~ 4.5 (Dandd) we can see that the turbulence intensities change rapidly in the upper regions and become smooth or nearly constant curves in the lower regions. This indicates that the transport mechanisms are very strong in the upper region and very weak in the lower region.

Figure 4.7 illustrates the generation of turbulence intensity. $t/T = 0.0$ shows a wave is just passing the breaking location ($x/h = 16$) where low levels of turbulence are generated in the crest and front side of the wave. $t/T = 0.1$ shows that the turbulence has spread over the top of the crest and down the front face of the wave as it continued to break. Strong turbulence intensity can be seen at $t/T = 0.2$ and 0.3. This is due to the high shear rates at the wave front which generates significant levels of k at the lower front face of the wave. After that, the production of k continues as the wave transforms into a bore and overtakes the decaying turbulence from the previous breaking wave. This phenomenon of spilling waves has been observed by many researchers (Svendsen, 1984; Ting and Kirby, 1996; Bradford, 2000).

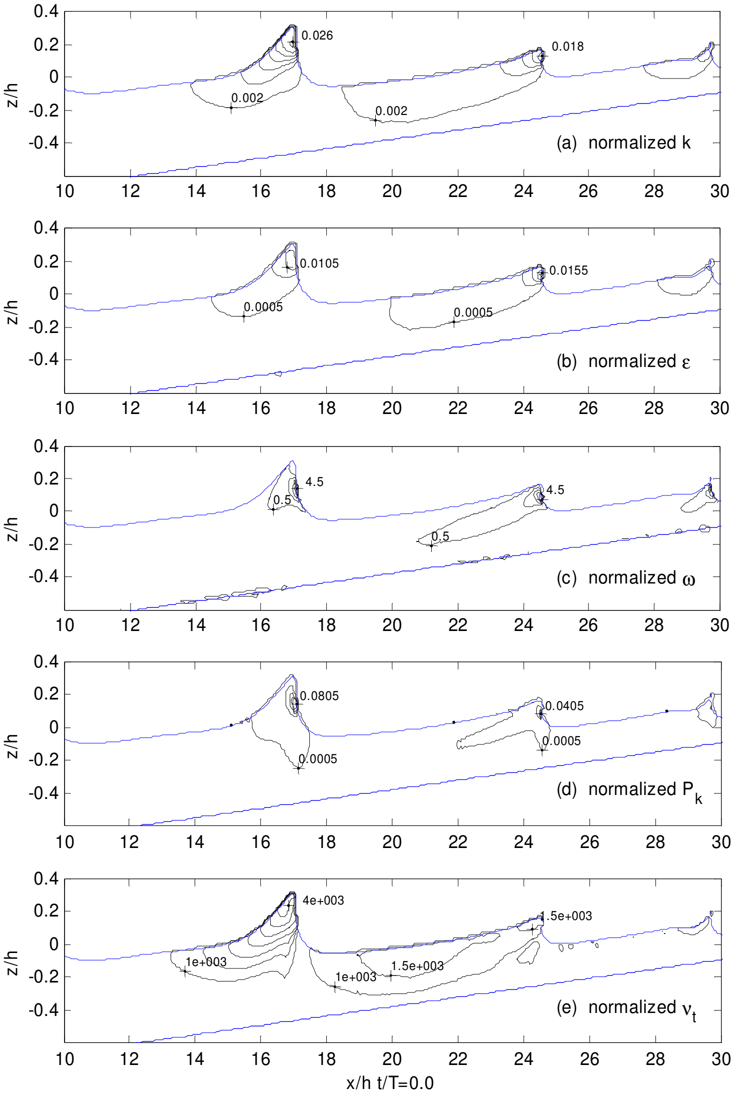


Figure 4.6: Computed normalized (a) turbulence kinetic energy $k/g\left(h+a\right)$ (with the interval of 0.05), (b) turbulence dissipation rate $\varepsilon/g\sqrt{g(h+a)}$ (with the interval of 0.005), (c) mean vortices $\omega/\sqrt{g/\left(h+a\right)}$ (with the interval of 1.0), (d) turbulence production $P_k/g\sqrt{g\left(h+a\right)}$ (with the interval of 0.02), and (e) eddy viscosity ν_t/ν (with the interval of 500) at t/T =0.0, 0.3, 0.6, and 0.9.

Figure 4.6 (continued)

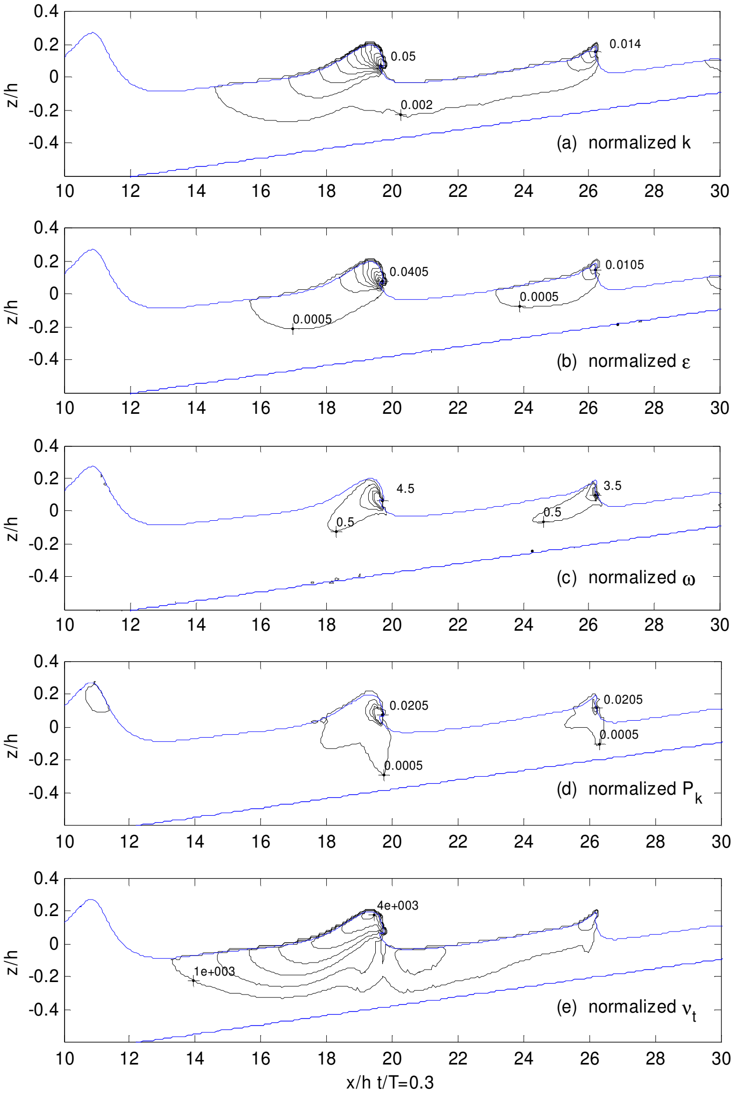


Figure 4.6 (continued)

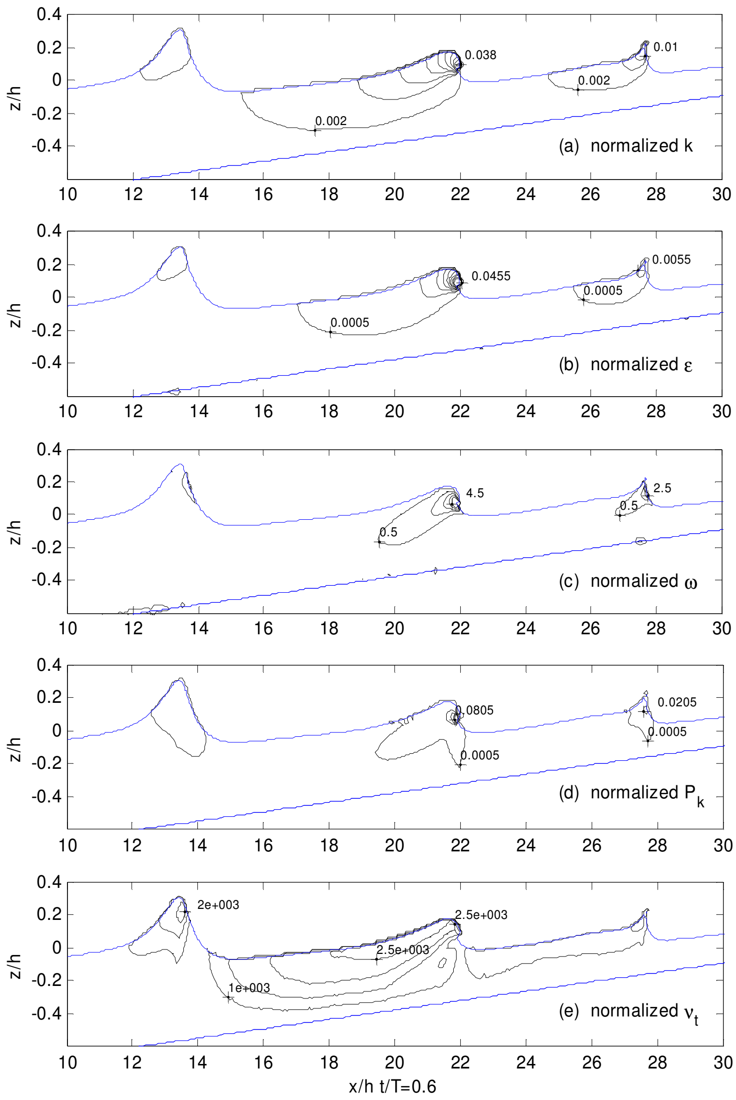
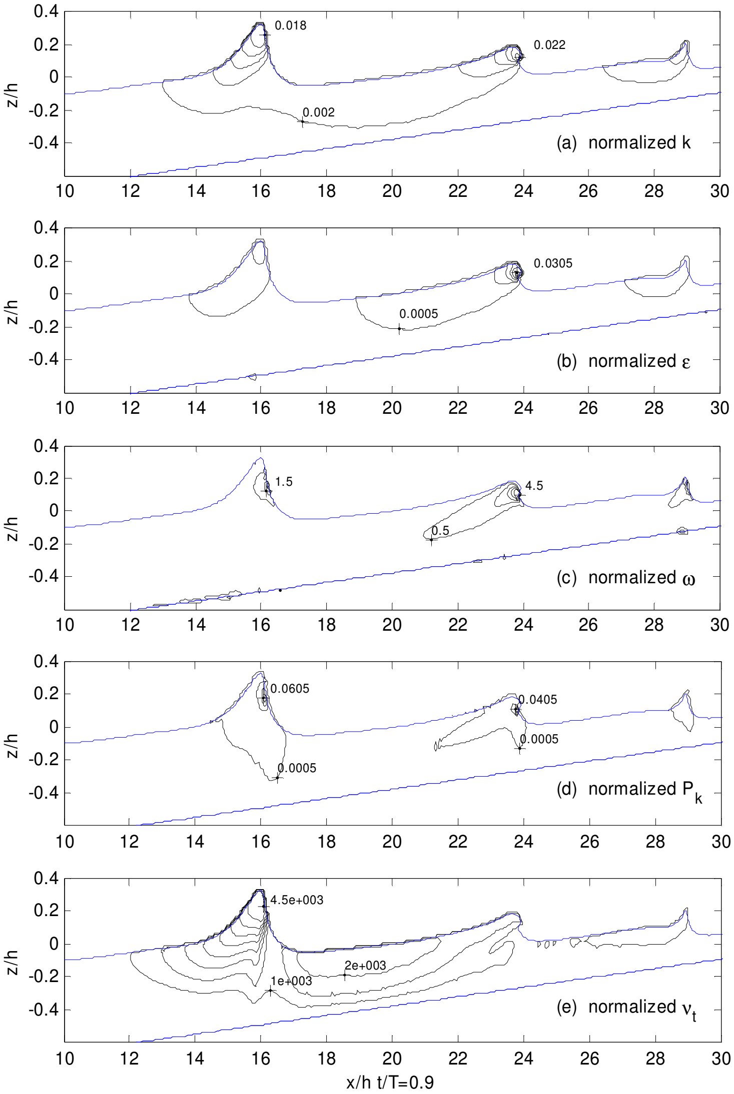


Figure 4.6 (continued)



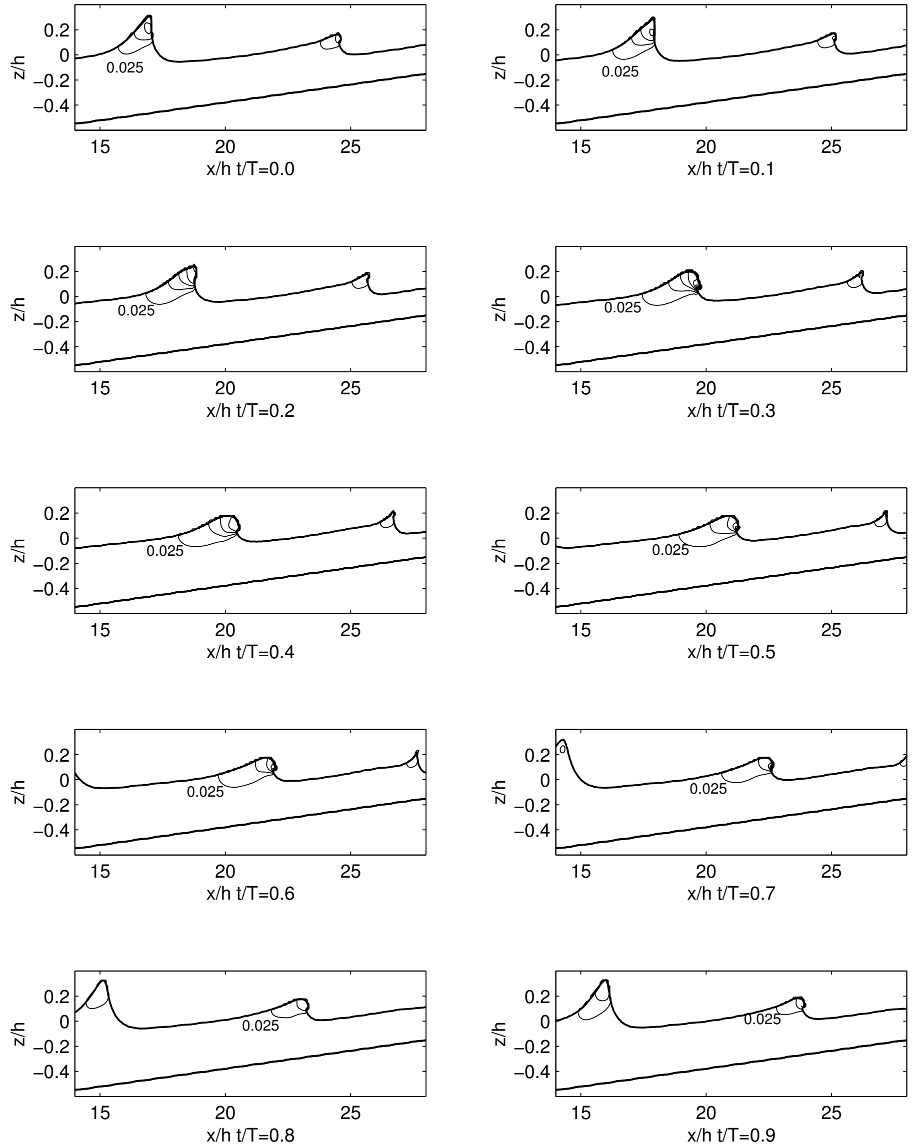


Figure 4.7: The snapshots of computed normalized turbulence intensities $\sqrt{2k}/\sqrt{g\left(h+a\right)}$ (with interval of 0.025) at $t/T = 0.0$ to 0.9.

4.3.6 Concluding Remarks

In this study, we have validated the $k-\varepsilon$ turbulence numerical model which can be used to simulate the breaking waves in the surf zone. By comparing the numerical solutions to the laboratory measurements, we found that the model results compare very well with the experimental data, especially in the inner surf zone.

We also provide the detail spatial distributions of turbulence characteristics which are not easy to measure in the laboratory experiments. From a series of snapshots, the generation, transport, and dissipation of turbulence intensity can be seen. Our study shows that the turbulence is mainly generated in the "roller" region. This is especially observed in the plots of turbulence production (Figure 4.6). The breaking process of the spilling breaker is a local phenomenon. In the bottom region, the turbulence levels are low and nearly constant. The turbulence production and mean vorticity are highly concentrated in the front face of the wave. However, the turbulence kinetic energy, turbulence dissipation rate and eddy viscosity are first transported from the breaking wave fronts into the wave crests and part of them will be left on the lee side of the waves.

4.4 Example 2: 3D Dam-Break Waves interacting with a Square Cylinder

In the previous section, we have already validated the accuracy of our numerical model for studying wave breaking. In this section, we shall apply our numerical model to study three-dimensional dam-break wave interacting with a square cylinder. The solution, using the nonlinear $k-\varepsilon$ turbulence model, will be compared in terms of the forces acting on the cylinder. Detailed turbulence transport will also be discussed.

4.4.1 Introduction

In designing a coastal structure in the nearshore region, such as the piles for a pier, a water intake structure or a submerged discharge pipeline, wave forces must be considered. In many situations, waves may break and interactions between waves and structure may also generate complex three-dimensional flow fields. In the past, our knowledge of wave-structure interaction has relied heavily on laboratory experiments and empirical or semi-empirical correlations. Based on the potential flow assumption some studies have been able to estimate the transmission and reflection coefficients (Mei and Black, 1969; Massel, 1983; Seabra-Santos et al., 1987; Ting and Kim, 1994; Zhuang and Lee, 1996; Tang and Chang, 1998; Huang and Dong, 1999), or the wave forces on a submerged rectangular box (Abul-Azm, 1994; Ertekin and Becker, 1996; Losada et al., 1996). Experimental measurements of free surface elevation for wave-structure interactions have also been reported (Grue, 1992; Rey et al., 1992; Beji and Battjes, 1993). Chyu and Rockwell (1996); Lin and Rockwell (1996); Sheridan et al. (1997); Chang et al. (2001) used more advanced Particle Image Velocimetry (PIV) technique and were also able to measure the velocities.

However, it is well known that the potential flow assumption may not be able to accurately estimate a complex flow field under a complex flow pattern, such as wave breaking. On the other hand, laboratory experiments also suffer from constraints on the range of testing physical parameters and scaling effects, not to mention the cost of performing careful experiments.

In this study, the force acting on the cylinder will be validated and compared with the laboratory experiments conducted at University of Washington. The turbulence models with the linear and nonlinear $k-\varepsilon$ models will be validated by comparing the predicted forces to the laboratory measurements. The detail transport of turbulence will be presented and discussed.

4.4.2 Governing Equations

In this case, since strong turbulence generated by the dam break bore interacting with the structure, the Reynolds averaged Navier-Stokes equations will be adopted as the governing equations (Equation 2.22 and 2.23). The nonlinear $k-\varepsilon$ model (Equation (2.37) ~ (2.44)) is used. The movement of the free surface is tracked by using the volume of fluid (VOF) method (Equation (3.24)).

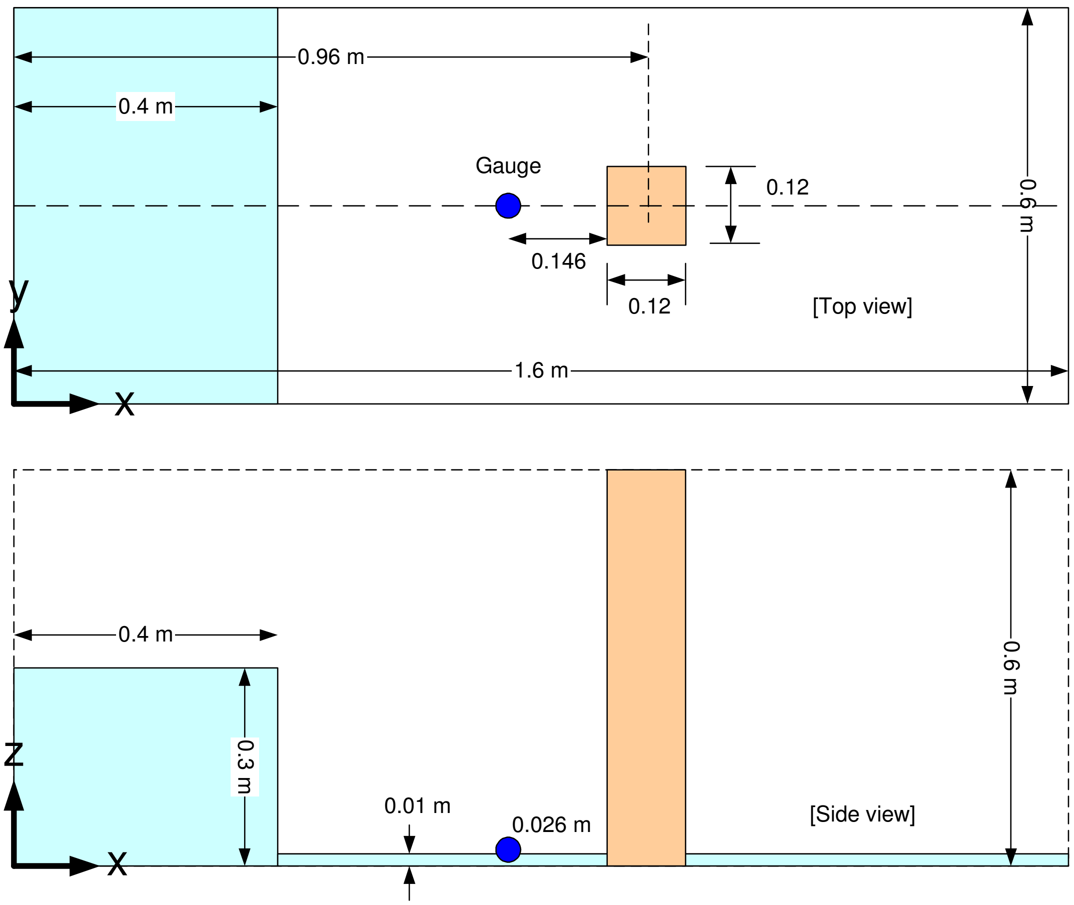


Figure 4.8: The numerical setup for dam-break wave interacting with a square cylinder.

4.4.3 The Experimental and Numerical Setups

In this section, the interaction of the dam-break wave with a square cylinder is simulated. The tank is 1.6 m long, 0.61 m wide, and 0.75 m tall. The volume of water initially contained behind a thin gate is 0.4 m x 0.61 m x 0.3 m. The structure, which is 0.12 m x 0.12 m x 0.75 m, is placed 0.5 m downstream of the gate and 0.24 m from the near sidewall of the tank. In the physical experiment, because it is impossible to completely drain the tank downstream of the gate, a layer of water (approximately 1 cm deep) always remains on the bottom of the tank and water also leaks through the gate before it is opened. Collected measurements included the time history of the net force on the cylinder and the time history of the fluid velocity at a fixed location. Forces were measured with a load cell and velocities were measured with a Laser-Doppler Velocimetery (LDV) system. The velocity measurement location is 0.146 m upstream of the center of the structure and 0.026 m off the floor of the tank.

Because the width of the square cylinder is within the same order of magnitude as those of the channel width and the initial bore height, the vortices are mainly generated at the corner around the square cylinder. This allows us to conduct the numerical simulation in a tank with half width to save the computational time. The numerical tank is 1.6 m long, 0.3 m wide, and 0.6 m tall with uniform grid $dx = dy = dz = 0.01$ m in all directions. See Figure 4.8 for details. Appropriate boundary conditions need to be specified. For the mean flow field, the free-slip boundary condition is imposed on the solid boundary. For the turbulence field, near the solid boundary, the log-law turbulent boundary condition is applied so that the values of k and ε can be expressed as functions of distance from the boundary and the mean tangential velocity outside the viscous sub-layer. On the free surface, the zero-gradient boundary conditions are imposed for both k and ε, i.e.,

$\partial k / \partial n = \partial \varepsilon / \partial n = 0$. A low level of k and ε for the initial condition is applied based on Equation (2.48) and (2.49).

4.4.4 Comparison of Experimental Data and Numerical Results

In order to examine the performance of the turbulence models, the numerical simulations with nonlinear $k-\varepsilon$ turbulence models are conducted. The force comparisons are shown in Figure 4.9. The numerical force is calculated by integrating the pressure on the structure and neglecting the shear force, which is normally an order of magnitude smaller than pressure force (Lin and Li 2003). Figure 4.10 is the time-history shear stress acting on the cylinder faces. This figure shows that the shear stress is about 1% of the normal stress and can be neglected when calculating the total force.

In order to evaluate the importance of turbulence effects on the net force on a square cylinder, we also show another two solutions in Figure 4.9. One solution is obtained by solving pure NS equations, and the other by solving filtered NS equations with LES turbulence model. The numerical implementation of LES model will be introduced in Chapter 5. In Figure 4.9, some features can be seen and will be addressed below.

Before the dam-break bore hits the square cylinder ($time$ < 0.31 sec), the differences between three numerical solutions (nonlinear $k-\varepsilon$, LES, and NS) are barely visible. That is because only a small amount of turbulence has been generated by then. However, after the bore hits the square cylinder, the flow condition is turbulent and three dimensional. Once the flow condition is turbulent, the turbulence dissipation will be under-estimated by solving pure NS equations. The difference, in terms of the bore phase, between these three numerical models is significant at $time \approx 1.25$ sec. Because of the under-estimation of the turbulence stress, the bore speed is over-estimated by solving the pure NS equations. As a result, the reflected bore from the end wall reaches the cylinder earlier than those in the other two solutions with turbulence models. With the same reason of under-predicting the turbulence stress, the net force predicted by solving pure NS equations contains stronger oscillation than those in the other two solutions with turbulence models at $time \approx 1.8$ sec.

An interesting phenomenon is that the solutions predicted by the nonlinear $k-\varepsilon$ and LES turbulence models are close to each other, especially the speed of the bore front movement. These two turbulence models have very different assumptions. The nonlinear $k-\varepsilon$ model is under the assumption of assemble averaging while LES is spatial filtering. However, these assumptions will not change the predicted traveling speed of the bore. This indicates that the solutions with nonlinear $k-\varepsilon$ model and LES model are close to the true DNS solution in terms of the traveling speed of the bore.

Figure 4.9 also shows that the turbulence effect on the net force on the square cylinder is minor. All the numerical solutions with or without a turbulence model predict very similar net forces in terms of the magnitude.

Overall, all the numerical results from turbulence models are very close to the laboratory measurements, especially in terms of the maximum positive and negative forces acting on the square cylinder. Figure 4.11 shows the comparison of time-history horizontal velocity at the gauge. Again, nonlinear $k-\varepsilon$ numerical models successfully predict the horizontal velocity.

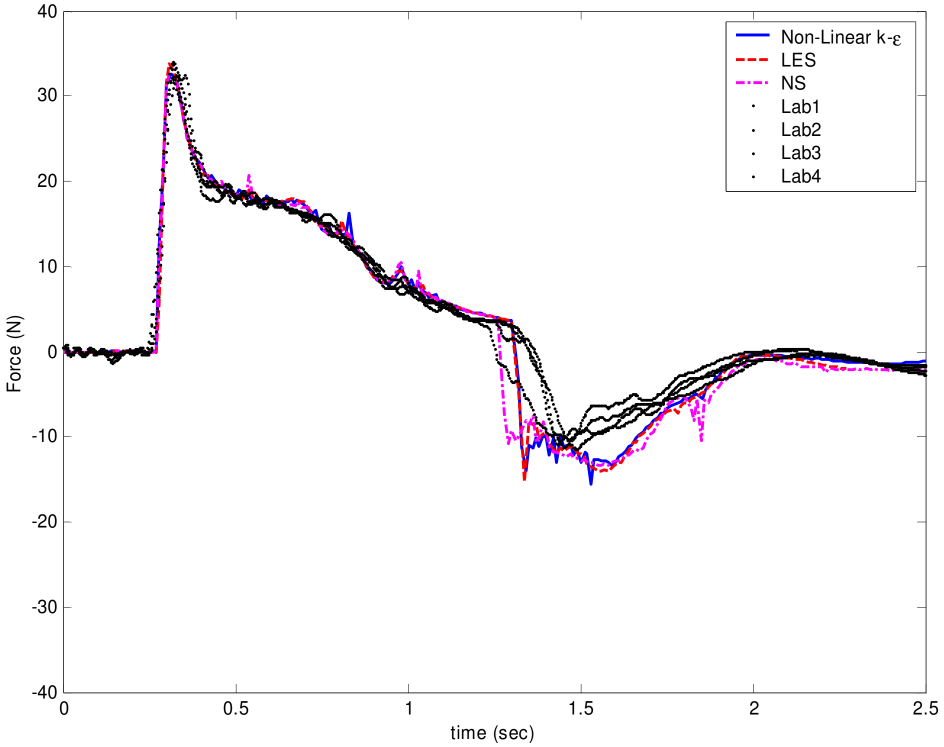


Figure 4.9: The time-history wave impact on a square cylinder. The numerical solutions of force on the square cylinder are validated with experimental data. The dot symbols are four sets of experimental data. The solid line is the numerical solution with nonlinear $k-\varepsilon$ model. The dashed line is the numerical solution with linear $k-\varepsilon$ model.

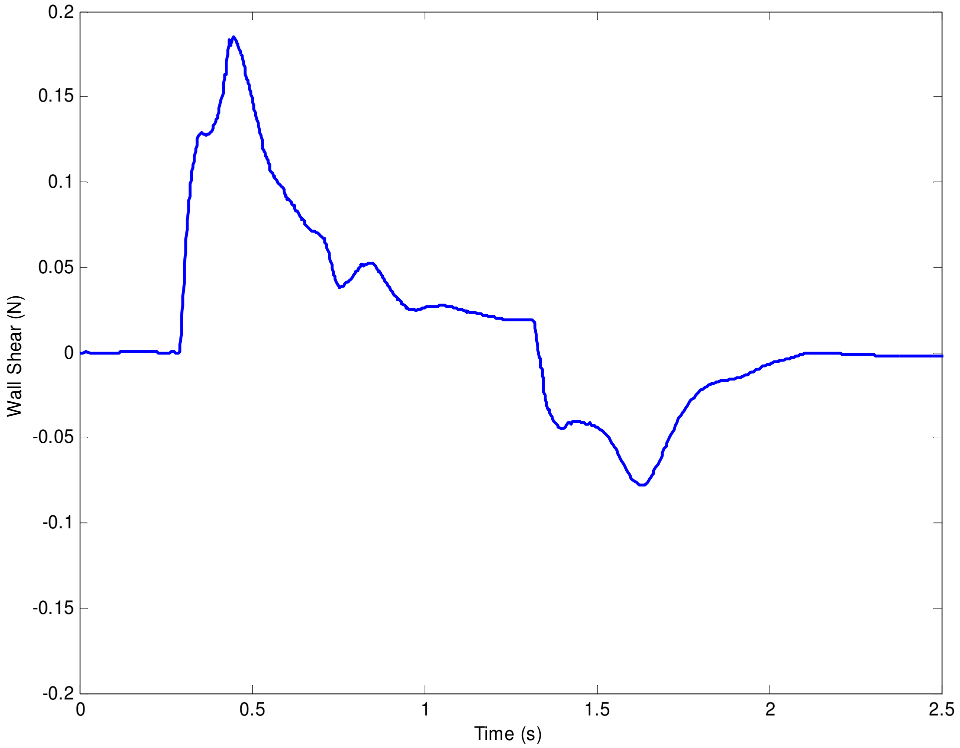


Figure 4.10: The time-history wall shear stress on a square cylinder. This solution is predicted by nonlinear $k - \varepsilon$ model.

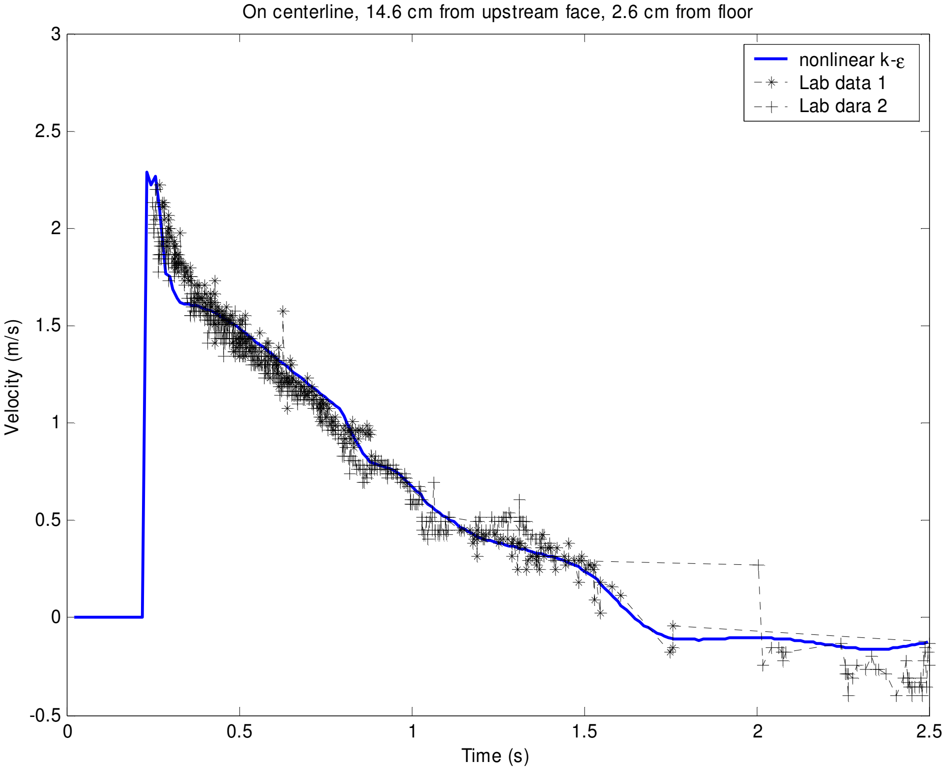


Figure 4.11: The time-history horizontal velocity. The gauge is located on the centerline; 14.6 cm from upstream face of the square cylinder, 2.6 cm from the floor. The dot symbols are two sets of experimental data. The solid line is the numerical solution with nonlinear $k-\varepsilon$ model. The dashed line is the numerical solution with linear $k-\varepsilon$ model.

4.4.5 Spatial Distribution of Turbulent Characteristics

In this section, the numerical solution obtained by solving the nonlinear $k-\varepsilon$ model will be presented. Figure 4.12 shows a sequence of snapshots of the free surface elevation as well as the turbulence intensity $\sqrt{2k}$. The initial numerical setup is shown at Time = 0.0 s. A “mushroom head” can be seen at Time = 0.15 s. Up to this stage, the flow condition is nearly 2D except for the weak shear stress generated by the side walls. The mushroom head bore will hit the floor before hitting the square cylinder and part of “air” will be trapped under the bore. The bore will hit the square cylinder and generate maximum net force at Time = 0.31 s. Part of the bore will be reflected by the square cylinder (Time = 0.36 s). The reflected water body will reach the maximum elevation at Time = 0.57 s. The maximum elevation has a chance to reach 0.48 m, which is higher than the initial impoundment (0.3 m). Due to the lower pressure in the wake zone, the separated bores on both sides of the square cylinder will merge together and generate strong turbulence. At Time = 0.68 s, the reflected bore on the cylinder face start collapsing. At Time = 0.80 s, two series of eddies begin to develop in the wake zone. The turbulence intensity is growing along with the developments of those eddies. At Time = 0.87 s, the reflected bore from the square cylinder is encountering the main dam-break bore, and a turbulence ring can be observed. After Time = 1.03 s, the bore will be reflected back from the end wall. The reflected bore will hit the floor and generate a rebounded wave at Time = 1.19 s and generate strong turbulence. After Time = 1.97 s, the turbulence intensity starts decaying gradually until hitting the upstream-side end wall.

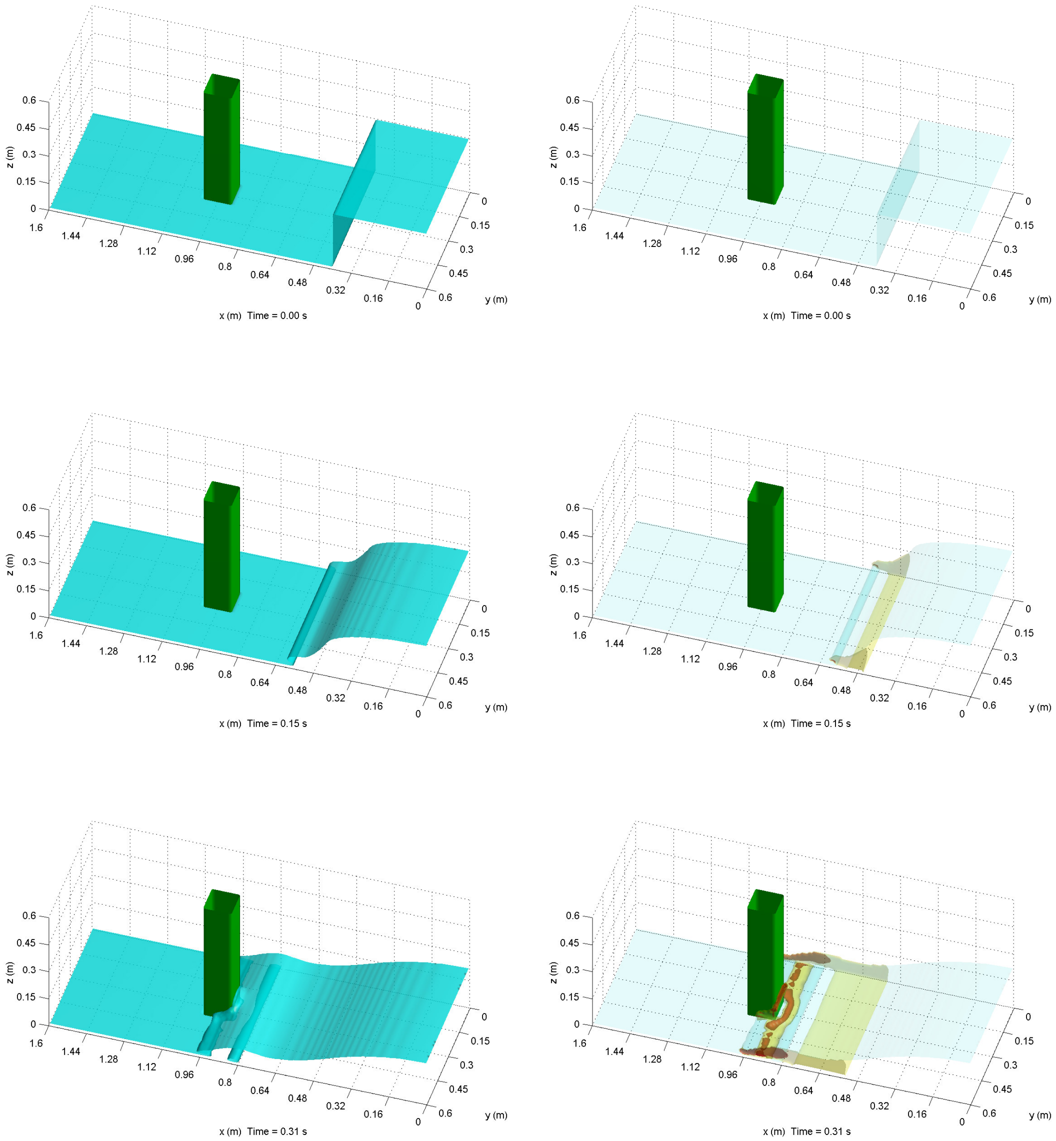


Figure 4.12: Snapshots of the free surface elevation (left column) and turbulence intensity $\sqrt{2k}$ (right column) by solving nonlinear $k-\varepsilon$ turbulence model. (Contour surfaces of turbulence intensity: [Yellow (Light gray): 0.125; Red (Gray): 0.25; Black (Dark gray): 0.375 m/s]).

Figure 4.12 (continued)

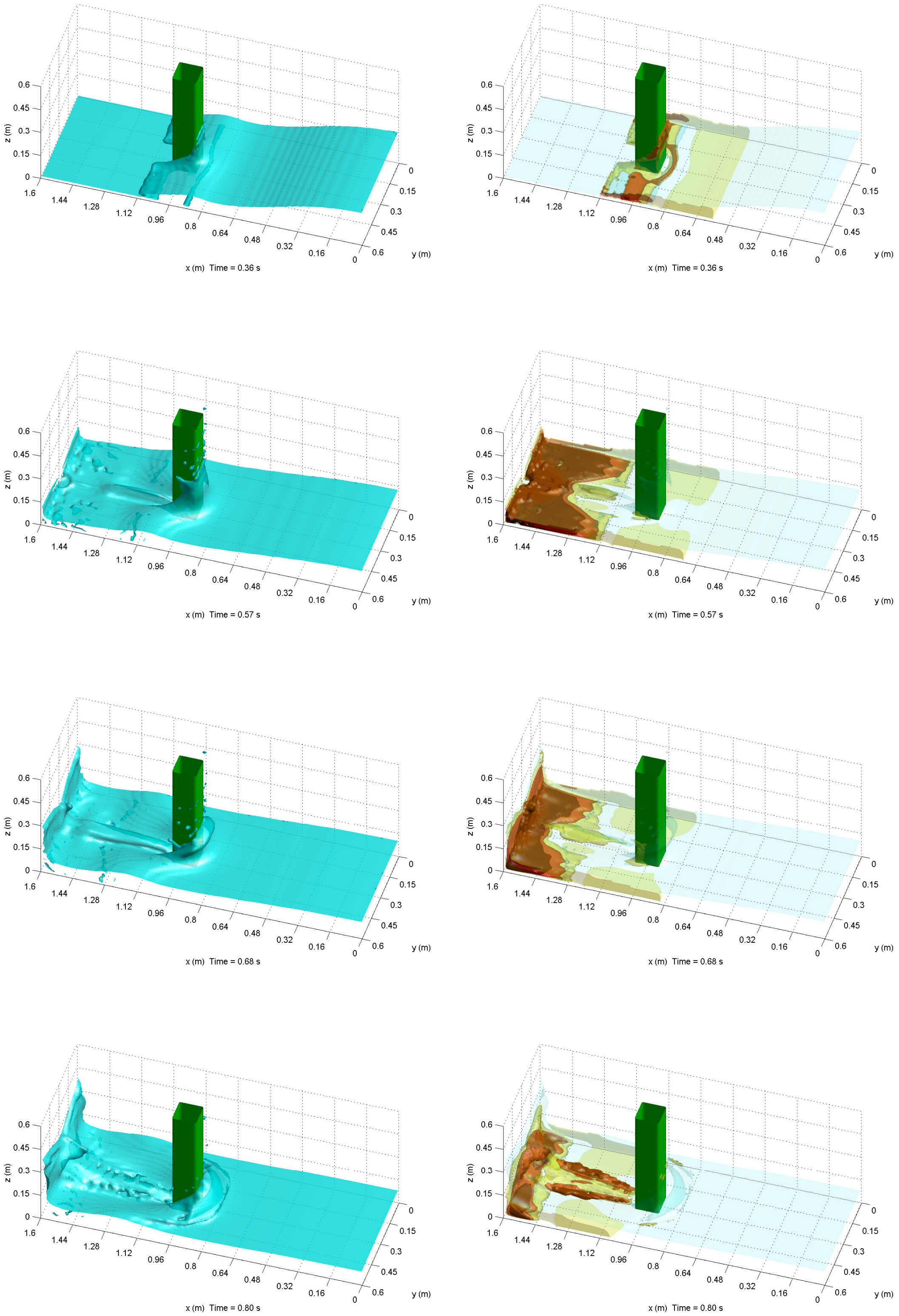


Figure 4.12 (continued)

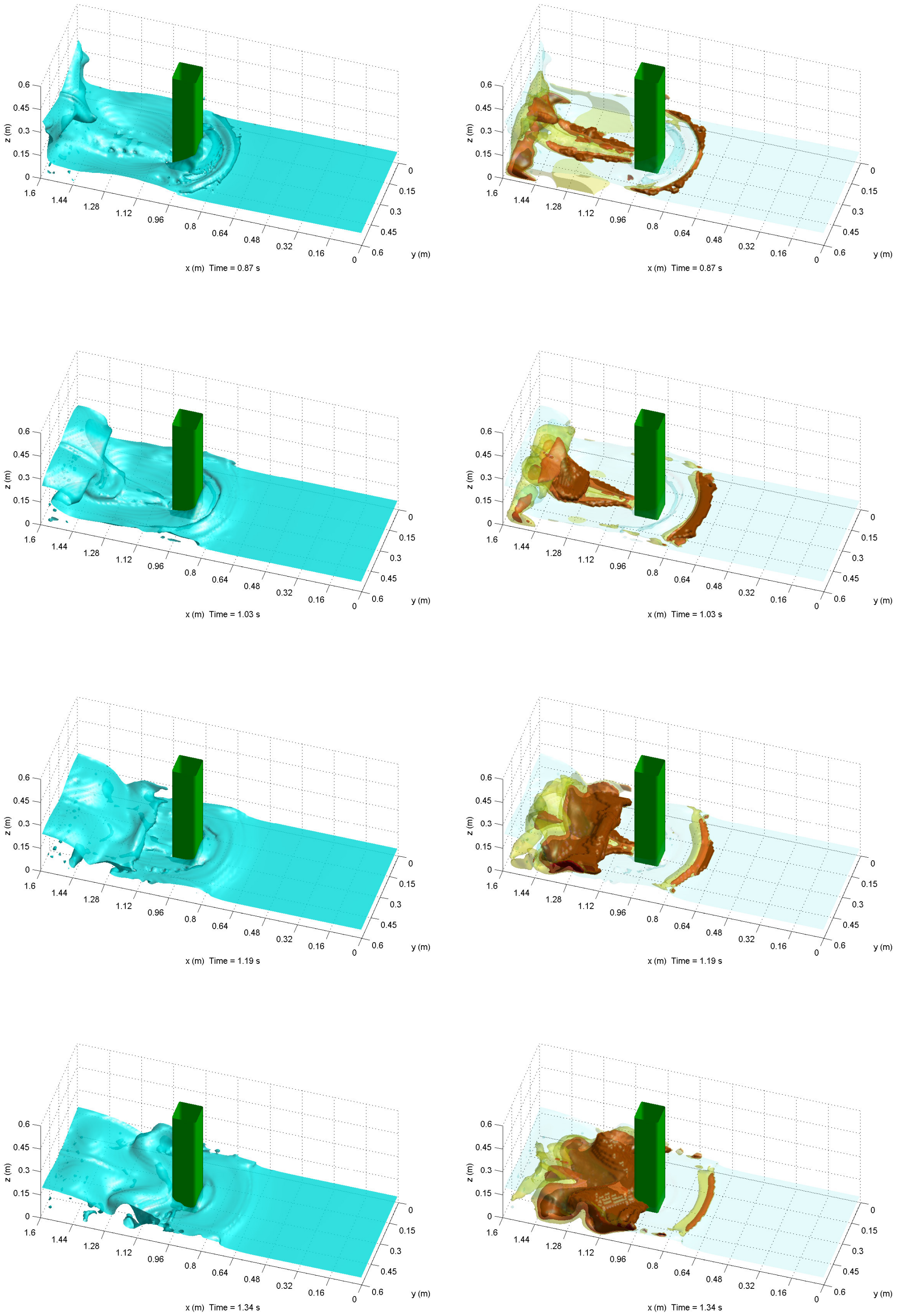
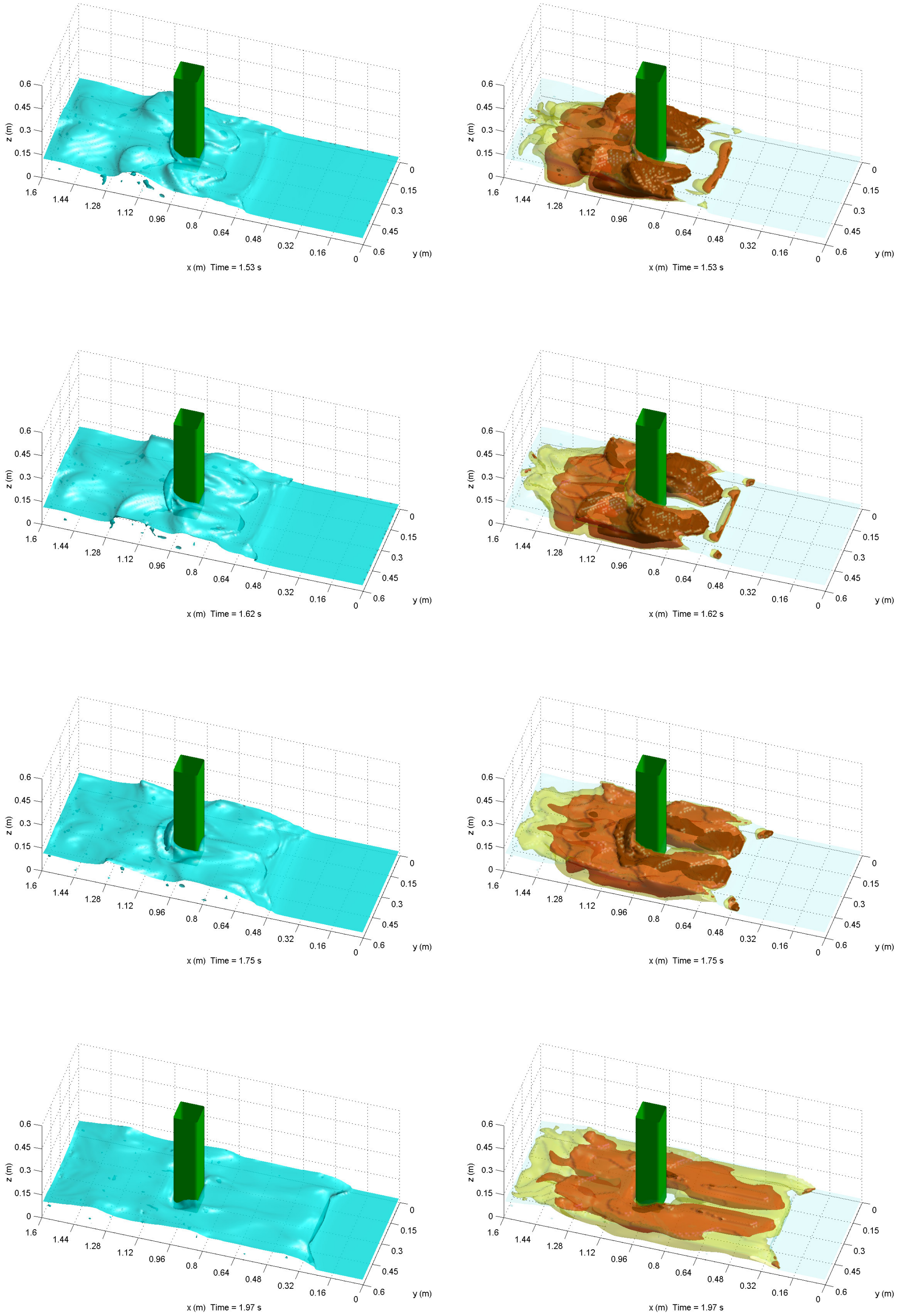


Figure 4.12 (continued)



4.4.5.1 Turbulence distribution at x, y, z cross-sections

Because the dam-break flow condition in this case is fully three-dimensional after Time = 0.31s, snapshots of cross-sections shall help us to understand the spatial distribution of the turbulence. In this study, we choose three cross-sections to discuss the turbulence distribution. There are cross-sections at: z = 0.045 m, y = 0.295 m, and x = 1.255 m. They will be shown in Figure 4.13, 4.14, and 4.15, respectively. In each cross-section snapshot, the turbulence intensity will be presented in a gray scale as well as the contour lines. The darker gray scale indicates stronger turbulence intensity.

In Figure 4.13, the x-y cross-section is chosen to be at z = 0.045 m. This is roughly 1/2 ~ 1/3 of the bore height in the wake zone. As we have addressed before, the bore will generate two sets of eddies in the wake region. This can be clearly observed from Time = 0.8 s ~ 1.03 s. At Time = 1.19 s, another stronger turbulence is generated at x = 1.25 ~ 1.35 m and y = 0.2 ~ 0.4 m. This is due to the reflected waves from the end wall. In general, the turbulence intensity at z = 0.045 m is in the region of $\sqrt{2k} \approx 0.2 \sim 0.4$ m/s, or $\sqrt{2k}/\sqrt{gH} \approx 0.117 \sim 0.233$, where $H = 0.3$ m is the impoundment.

Figure 4.14 shows the x-z cross-section at y = 0.295 m. This cross-section is very close to the centerline cross-section. At Time = 0.68 s, the bore is just reflected by the end wall. The turbulence intensity is as high as 0.6 m/s. At Time = 1.03 s, the reflected wave has a contact with the main bore at x =1.25 ~ 1.35 m and z = 0.05 ~ 0.075 m. This wave will be bounced up by the floor at Time = 1.19 s, and hit the square cylinder again at Time = 1.34 s. Inspected from Time = 0.68 ~ 1.62 s we can see that the turbulence intensity is about 0.2 m/s, which is within the same order of magnitude as that observed from z = 0.045 m cross-section.

Figure 4.15 shows the y-z cross-section at x = 1.255 m. This cross-section is chosen at twice as much as the cylinder width behind the squire cylinder where the turbulence condition is very complex. At Time = 0.68 s, the separated bores from two sides of the cylinder merge again and generate stronger turbulence at y = 0.25 ~ 0.35 m and z = 0.05 ~ 0.07 m. At Time = 0.8 s and 0.87 s, the jet-like eddies can be observed. The turbulence intensity is very high (about $\sqrt{2k} = 0.6$ m/s) compared to the ambient flow. After Time = 1.19 s, a part of the bore is reflected from the end wall and encounters the main bore. The turbulence intensity is $\sqrt{2k} = 0.2 \sim 0.6$ m/s.

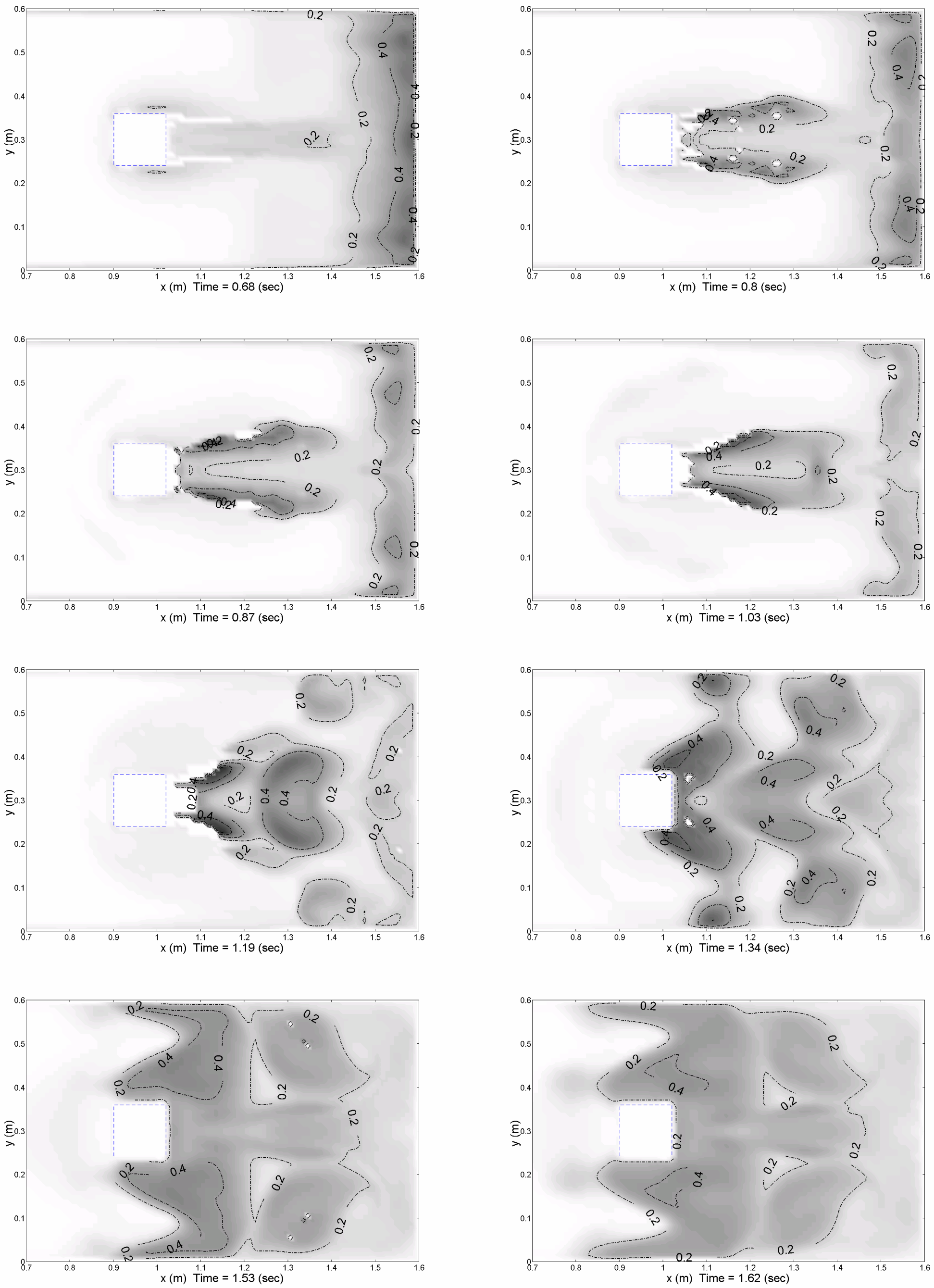


Figure 4.13: Snapshots of the turbulence intensity $\sqrt{2k}$ (m/s) on $z = 0.045$ m cross-section. Denser colors indicate stronger turbulence intensities whose values can be referenced to the contour lines (with interval = 0.2 m/s). The square dot-circle is the location of the square cylinder.

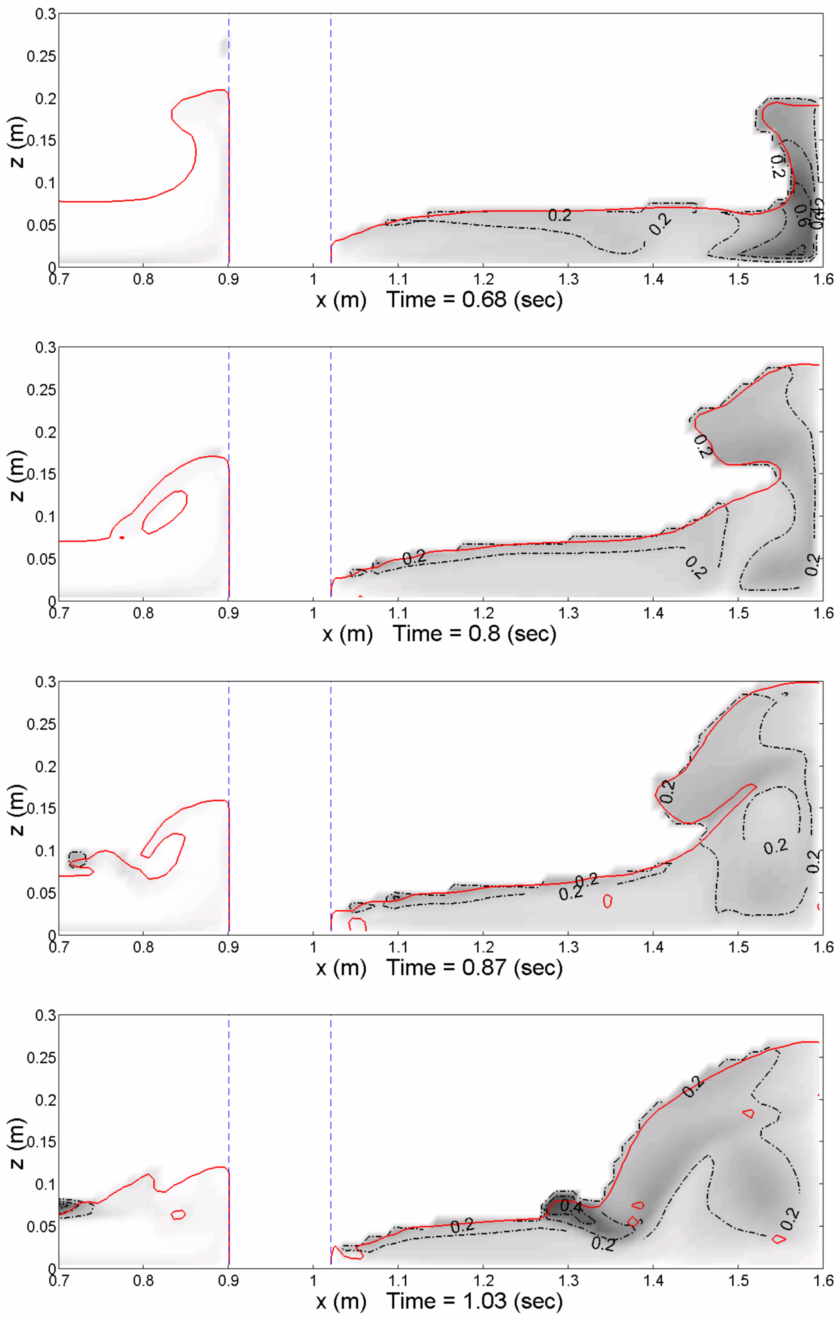
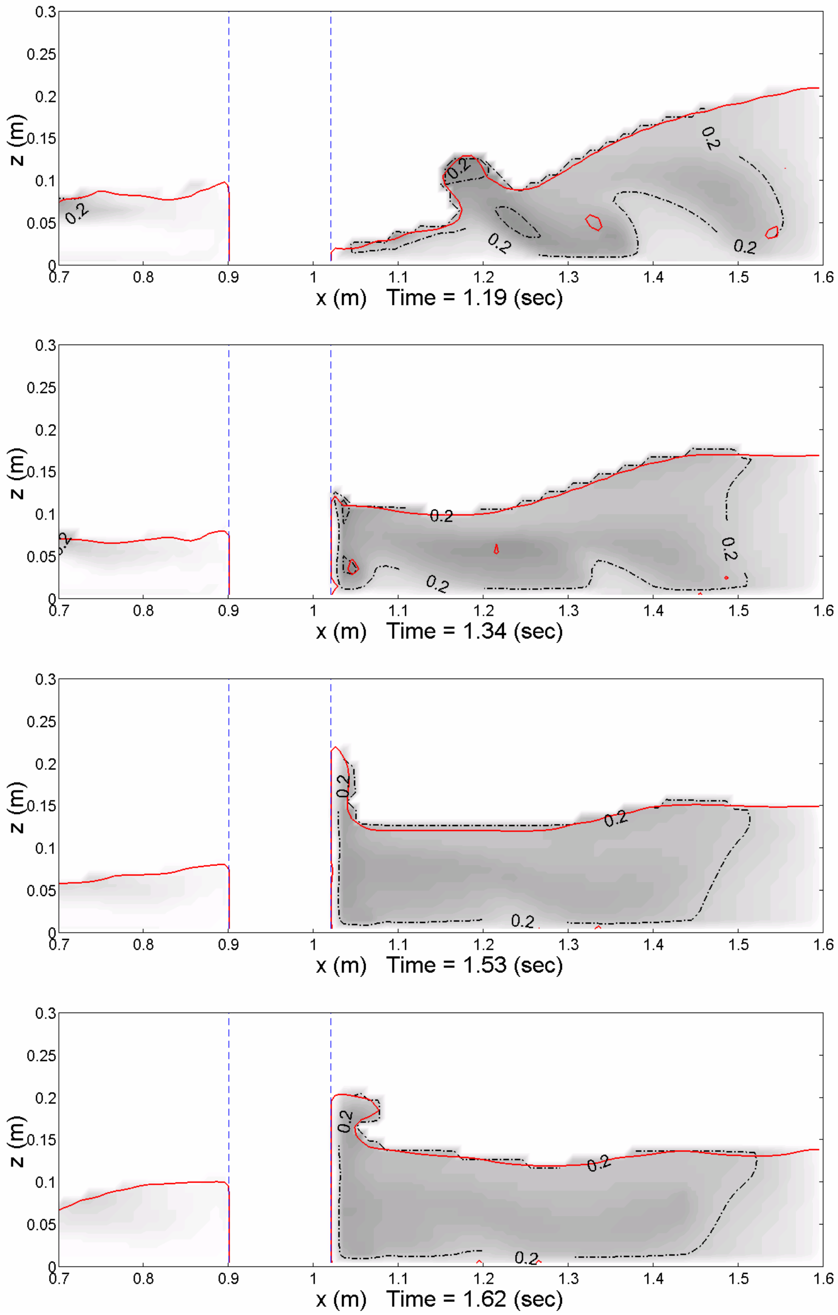


Figure 4.14: Snapshots of the turbulence intensity $\sqrt{2k}$ (m/s) on $y = 0.295$ m cross-section. Denser colors indicate stronger turbulence intensities whose values can be referenced to the contour lines (with interval = 0.2 m/s). The broken lines indicate the location of the square cylinder. The solid lines indicate the water surface where VOF of water = 0.5.

Figure 4.14 (continued)



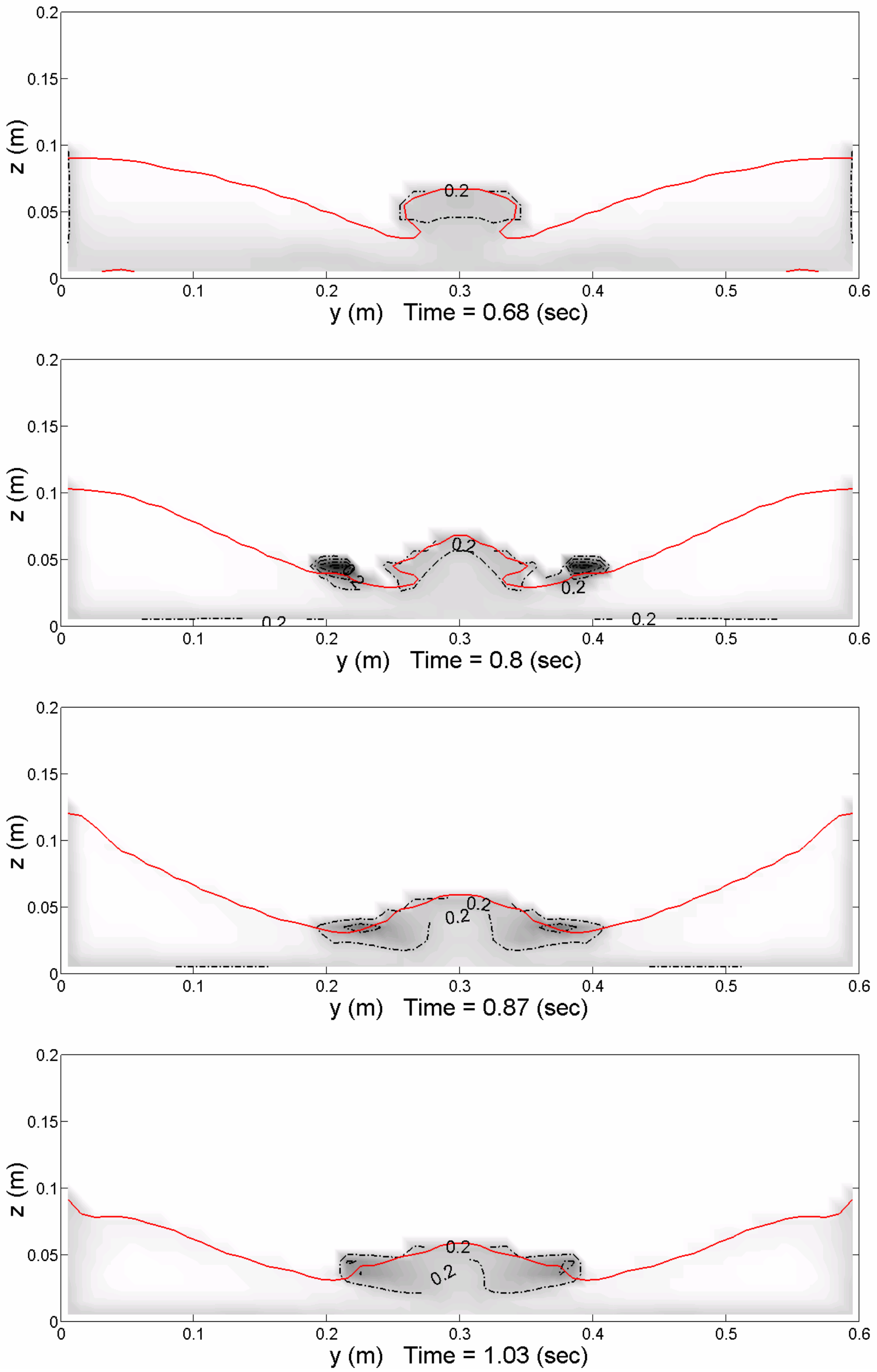
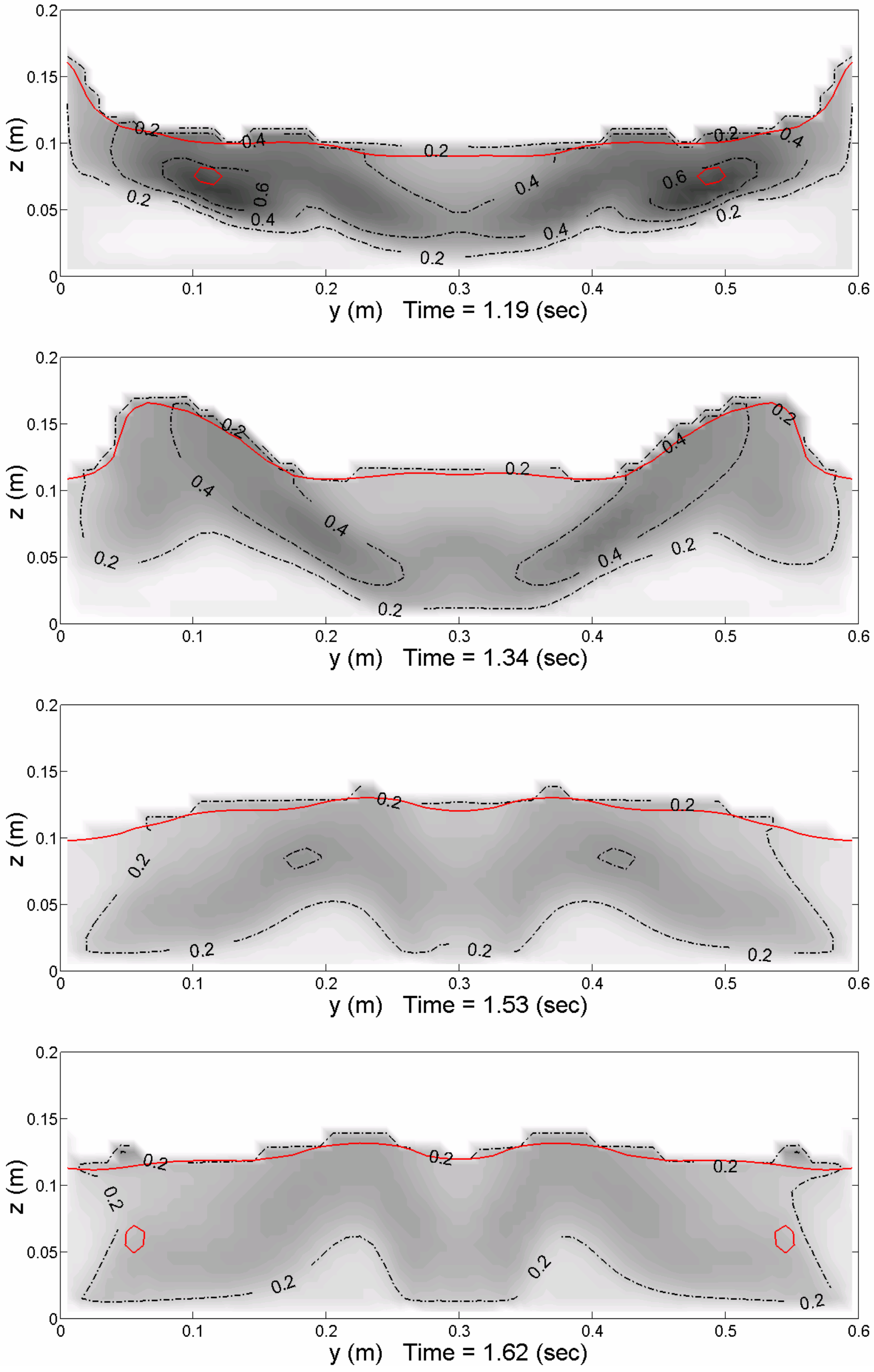


Figure 4.15: Snapshots of the turbulence intensity $\sqrt{2k}$ (m/s) on $x = 1.255$ m cross-section. Denser colors indicate stronger turbulence intensities whose values can be referenced to the contour lines (with interval = 0.2 m/s). The solid lines indicate the water surface where VOF of water = 0.5.

Figure 4.15 (continued)



4.4.6 Concluding Remarks

In this study, we have examined the performance of the three-dimensional numerical model by simulating a wave-structure interaction problem. The nonlinear $k-\varepsilon$ model is used to represent the turbulence effect on the mean flow and to trace the turbulence distribution.

In this dam-break case, because of the strong vortex at the edge of the cylinder and complex flow pattern with strong turbulence, the potential flow assumption is not suitable. The numerical simulations with turbulence model successfully predict the maximum wave force acting on the structure which is very important for designing a coastal structure. The simulation results show that the turbulence has minor effect on the net force on the square cylinder. However, the turbulence stress does affect the bore propagation speed. Both nonlinear $k-\varepsilon$ and LES modes predict a very similar bore traveling speed for the reflected bore from the end wall.

The nonlinear $k-\varepsilon$ model is also used to trace the spatial distribution of turbulence. Strong turbulence can be observed in eddies and in reflected waves, and the magnitude of turbulence intensity is in the region about $\sqrt{2k}=0.4$ m/s in this case.

Chapter 5

A Numerical Investigation of Runup from Three-Dimensional Sliding Masses

Landslide-generated tsunamis appear to be one of the major hazards in a coastal region. To better understand the landslide-generated waves and runup, we have developed a numerical model to simulate a sliding wedge in a 3.7 m-wide tank with a 1:2 plane slope in it. This solid wedge (a 91 cm horizontal length, 46 cm high and a 61 cm wide vertical face) was used to represent a landslide. The initial positions of the wedge ranged from totally aerial to fully submerged, and the weight was also varied over a wide range.

This numerical model is based on the Large-Eddy-Simulation (LES) approach. The Smagorinsky subgrid scale model is employed and the Volume of Fluid (VOF) method is used to track the free surface and shoreline movements. A numerical algorithm for describing the motion of the slide is developed. Numerical results are compared with the experimental data in terms of the time histories of runup and generated waves. Based on the numerical results, the detailed flow patterns, free surface and shoreline deformations are discussed.

5.1 Introduction

Tsunamis may be generated by several natural activities, such as earthquakes, volcano eruptions, and landslides. Mainly due in the disasters caused by slump-introduced tsunami of Papua New Guinea (PNG) in 1998 (Bardet et al., 2003), landslide-generated tsunamis have recently received great interest. In the PNG tsunami event, more than 2200 lives were lost (Davies, 1999; Kawata et al., 1999). The PNG tsunami is triggered by an earthquake of magnitude seven. Some recent investigations showed that for this size of earthquake, the tsunami wave high at shoreline was estimated to be 7 m to 8 m. However, the field watermark was up to 12 m. This indicated that the tsunami was generated by a massive underwater slump caused by the earthquake (Kawata et al., 1999; Synolakis et al., 2002).

Landslides in coastal areas have the potential to generate large tsunamis. The size of tsunamis depends mostly on the size of the sliding masses and the slope of the sea bottom. They may generate large waves with huge runup. One famous case is Lituya Bay tsunami happed in 1958 in Alaska. A huge volume of rocks falls down and produced a huge wave in the inlet. This wave then destroyed all vegetation up to 500 m at the opposite coast (Miller, 1960; Murty, 1977).

The landslide generated waves presents many complications, such as complex three-dimensional interfacial flow structures, moving boundary, and strong turbulence. The moving solid slides make the boundary condition depend on time. The strong turbulence generated by slide motion requires a very fine grid to resolve the turbulent structure, which makes the numerical simulation expensive especially in three-dimensional simulations.

The strong turbulence generated by the landslides requires a proper turbulence model. In this study, the traditional Smagorinsky model will be adopted, which is the most popular LES SGS model, and has been used in many breaking wave studies (Watanabe and Saeki, 1999; Christensen and Deigaard, 2001; Lin and Li, 2003).

The present numerical study concerns the solid landslide motion as well as the landslide generated breaking wave run-up and run-down on a slopping beach. A moving solid algorithm similar to Heinrich's method will be developed. The volume of fluid (VOF) methods similar to Rider and Kothe (1998) and Kothe et al. (1999) will be used to track the surface movement. LES turbulent model will be employed on the breaking wave simulations. Forty five computational simulations have been conducted with different initial elevations and specific weight of the solid slides. Comparisons between the computational results and experimental data for time histories of free surface fluctuations and the runup/rundown at various locations are made. Overall the agreement is very good. The simulation results for the velocity field, shoreline evolution and free surface profiles are discussed in detail.

5.2 Governing Equations and Boundary Conditions

The flow motions generated by a slide movement can be described by the Navier-Stokes equations. However, the size of the physical domain for the turbulent flows makes the direct numerical simulation (DNS) nearly impossible with available modern computers. The LES model (Deardorff, 1970) which solves the large-scale eddy motions and models the small-scale turbulent fluctuations becomes an attractive alternative. In the LES model, the governing equations can be seen in Equation (2.59) and (2.66). The Smagorinsky model (Smagorinsky, 1963) will be adopted to close the residual stress τ_{ij}^{r} in Equation (2.66).

5.2.1 Wall Boundary Condition

In this simulation, a near-wall damping function (Equation 2.83) derived by Cabot and Moin (2000) will be used at the cells adjacent to the solid walls. The Newton-Raphson iterative method (Equation 4.5) will be adopted to solve eddy viscosity ν_t:

5.3 Numerical Implementation

In the numerical model, the filtered equations are solved via the finite-volume two-step projection method introduced in Chapter 3. The forward time difference method is used to discretize the time derivative. To track the free-surface locations, the volume of fluid (VOF) method is used. In order to simulate the momentum and moving boundary effects introduced by landslides, the moving solid algorithm will be developed. In this section, we shall present the detail algorithm to solving moving solid problems.

5.3.1 Moving Solid Algorithm

Non-stationary boundaries are important in an increasing number of coastal applications, including sand drift and landslide problems. As for solving the moving boundary problem, curvilinear boundary-fitted coordinates and unstructured grids are two possible methods. However, the boundary-fitting method is complex in grid generation. Moving or deforming boundaries further complicate numerical flow simulations. When boundary fitted meshes are employed, the grid either has to be regenerated or deformed as the boundary geometry changes, adding considerable complexity to the computations. Another approach is the immersed boundary method (IBM). This method introduces a body-force field so that a desired velocity distribution can be assigned over a solid boundary (Mohd-Yusof, 1997; Goldstein, et al., 1993). This method allows the imposition of the boundary conditions on a given surface without coinciding with the computational grid. The major concern over this method is the interpolation of forcing over the grid. The IBM has been successfully applied to the internal combustion (IC) piston simulation (Fadlun et al., 2000; Verzicco et al., 2000). The IBM has the advantage of providing highly accurate results to the moving boundary problems. However, this method depends on the interpolated velocity on the boundary to evaluate the immersed force around the moving boundary. Therefore it requires fine resolution on the solid boundary. Instead of using immersed force in the fluid domain, Heinrich (1992) added a source function in the continuity equations to represent the solid movement to simulate two-dimensional landslide problems and the water surface profiles were compared well to the laboratory data. The moving solid algorithm which involves a source function in the VOF equations does not require any interpolation scheme and is independent of the grid system, thus it is very attractive in current study. In this simulation, a moving solid algorithm which is improved from Heinrich's method will be presented, and will be used to simulate three-dimensional subaerial and submerged landslide problems.

In order to simulate the movement of a moving solid, the partial grid algorithm (section 5.3.2) and an internal source function are added to the program. Consider a volume V containing an obstacle (Figure 5.1), which is defined by its volume $V_{obst(t)}$ and its surface $A_{obst(t)}$. If the obstacle volume increases, i.e. $dV_{obst}/dt > 0$, the volume of the fluids decrease, and vice versa. The conservation of mass in the volume V can be expressed as follow:

$$\oint_A \mathbf{u}\cdot\mathbf{n}dA = \frac{dV_{obst(t)}}{dt} \tag{5.1}$$

Then the new continuity equation for the volume V can be expressed as:

$$\nabla\cdot\mathbf{u} = \frac{1}{V}\frac{dV_{obst(t)}}{dt} = \phi\left(x, y, z, t\right) \tag{5.2}$$

where $\phi =$ internal source function. Substituting the new continuity equation, Equation (5.2), into the original governing equation, the momentum equation becomes:

$$\frac{\partial\left(\rho\bar{u}_i\right)}{\partial t} + \frac{\partial\left(\rho\bar{u}_i\bar{u}_j\right)}{\partial x_j} = -\frac{\partial\bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu_{eff}\left(\frac{\partial\bar{u}_i}{\partial x_j} + \frac{\partial\bar{u}_j}{\partial x_i}\right)\right] + \rho g_i + \rho\bar{u}_i\phi \tag{5.3}$$

The main procedure of moving solid algorithm can be expressed as:

1. Update the new time step solid body VOF (Figure 5.2).
2. Add a positive source function to the region where fluids will be pushed out, and a negative source function in the wake zone where fluids will be sucked in (Figure 5.3)
3. Solve the new time step velocities.

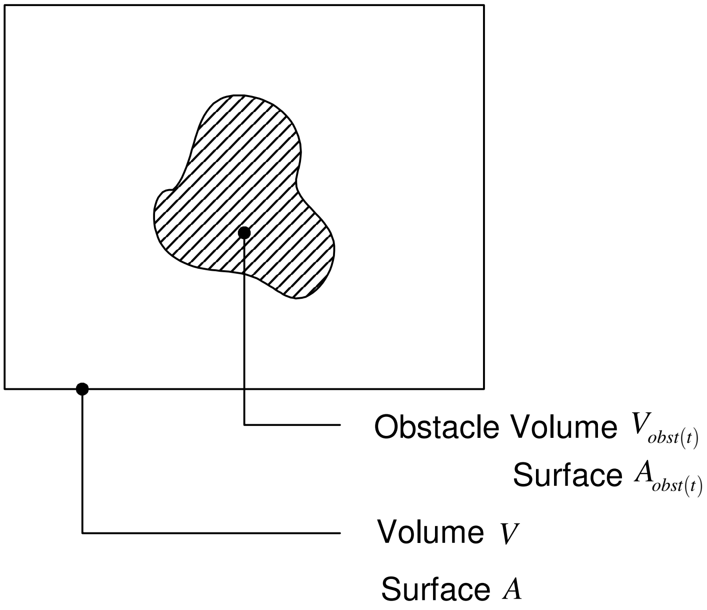


Figure 5.1: A solid body in the fluid domain.

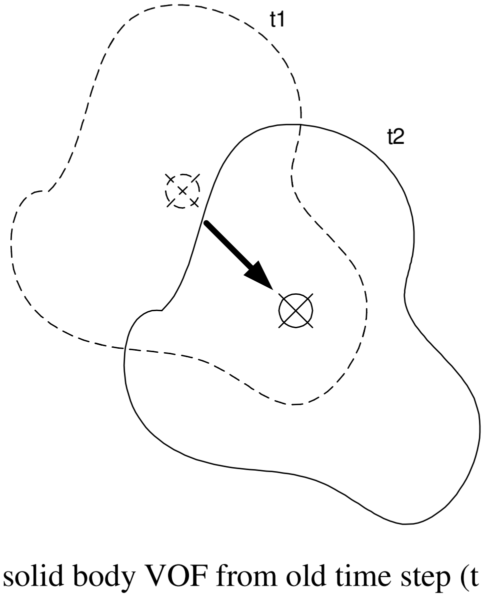


Figure 5.2: Update the solid body VOF from old time step (t1) to new time step (t2).

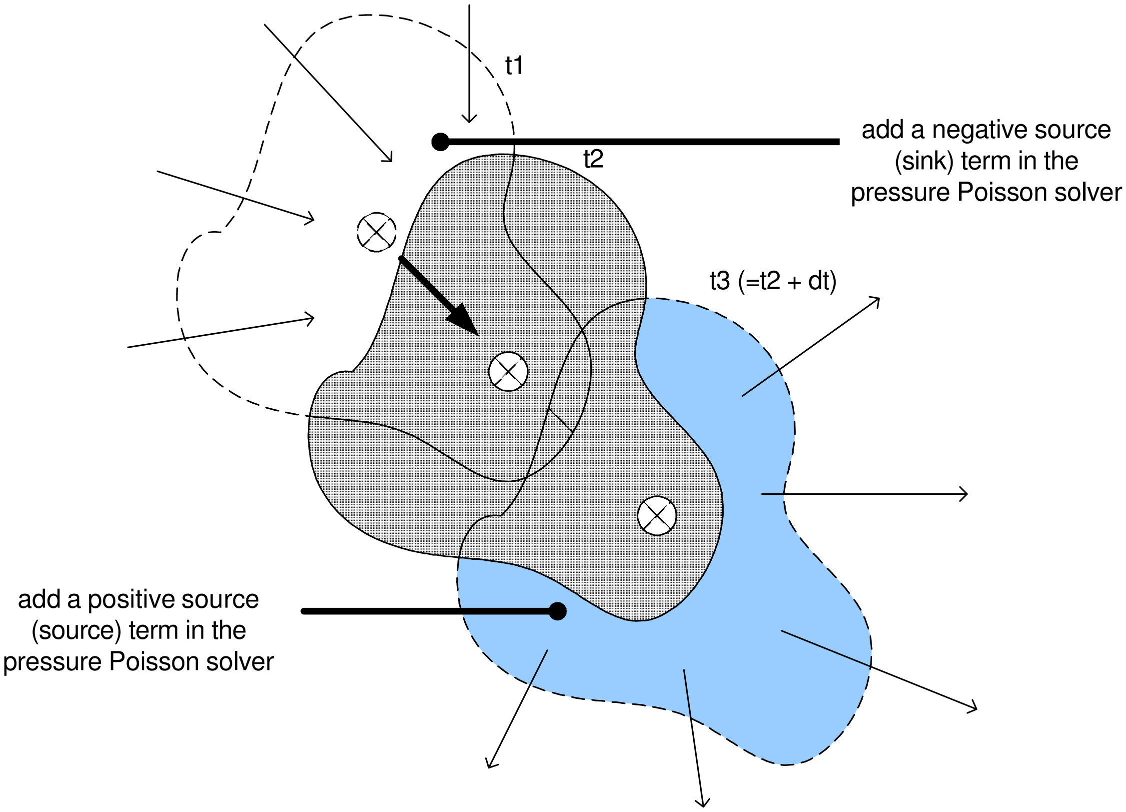
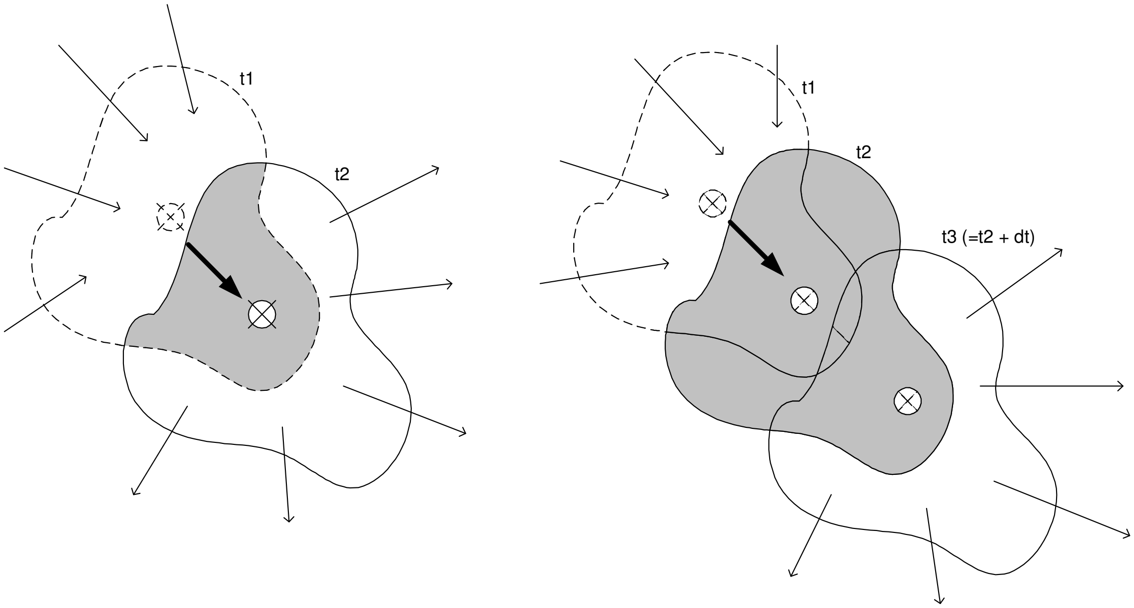


Figure 5.3: Add a positive source term in front of the solid body, and a negative source term (sink term) in the wake zone. Where dt = t2 – t1, t3 = t2 + dt.



Heinrich's Method Present Algorithm

Figure 5.4: The comparison between Heinrich's method and present moving solid algorithm.

The difference between present moving solid algorithm and the Heinrich (1992) one is that, the present algorithm updates the new time step solid VOF first, and then adds additional source functions in the areas where materials changed. However, in Heinrich's method, the source functions are added before updating the solid VOF. The present algorithm yields more accurate moving boundary effect (Figure 5.4).

One of the advantages of using the moving solid algorithm is that the grid system does not need to fit the solid boundary. Therefore, a part of the internal cells will be occupied by the solid material. In order to deal with a cell with partially solid materials and partial fluids, a simple partial cell treatment will be applied.

5.3.2 Partial Cell Treatment

The purpose of the partial cell treatment is to adopt moving solid algorithm function and to deal with unstructured interior obstacles. Since the obstacle inside the computational domain occupies a part of the interior fluid volume, the effective cell volume will be a fraction of the original cell volume:

$$\begin{aligned} V_{eff} &= \left(1 - f_{solid}\right)V \\ &= \theta V \end{aligned} \tag{5.4}$$

where V_{eff} is the effective cell volume, f_{solid} is the volume fraction occupied by solid material in each cell, V is the original cell volume, and θ is the effective volume fraction.

If a cell contains partial volume of solid material, the flow solver has to deal with it. Cell faces are defined either to be entirely closed, or not. Cell faces are "closed" only if at least one of the two immediately neighboring cells is entirely occupied by solid material. If the cell faces are "closed", the face velocity of the cell is set to zero, and the face pressure is no longer calculated in the pressure solution. On the other hand, if any face between two cells, containing at least a partial cell volume of fluid, is "open", the code solves the velocities and pressure gradients.

Compared to the conventional way to treat the irregular solid material that creates the "saw-tooth" boundary, the partial cell treatment introduced here is a better choice to represent the smooth real geometry of the boundary.

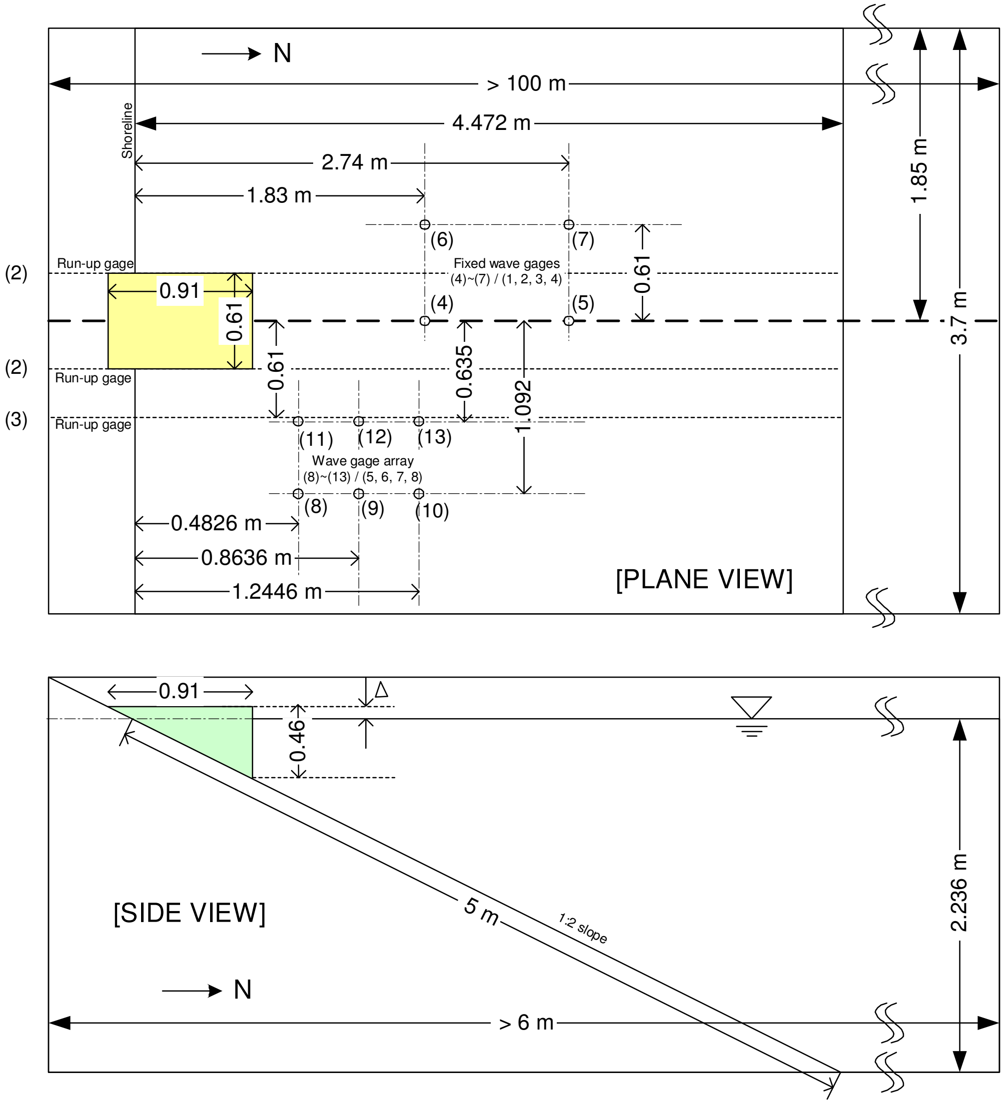


Figure 5.5: The Laboratory setup of the 3D landslide experiment.

5.4 Laboratory Setup

The numerical simulation results for a subaerial and a submerged wedge slide will be compared with the experimental data so as to gain confidence in the accuracy of the numerical model. The experiments were conducted in a wave tank at the Oregon State University. The wave tank has a length of 104 m, width of 3.7 m, and depth of 4.6 m. A plane slope (a beach with an inclination of two horizontal to one vertical) was installed near one end of the tank and a dissipating beach at the other end. In the experiments, two configurations of the wedge and one hemisphere on the slope were used: (1) the front face of the wedge vertical (Series A); (2) the wedge turned “end-for-end” so that for this orientation the top and front faces are neither horizontal nor vertical (Series B); (3) A hemisphere with 91.44 cm in diameter (Series H). However, due to the limited time, our numerical investigations only focused on Series A simulations. A solid wedge was used to represent the landslide (Figure 5.5). The triangular face has the following dimensions: a length of b = 91.44 cm, a front face a = 45.72 cm and a width of w = 61 cm.

In the laboratory experiments, the bodies move down the slope by gravity rolling on specially designed “Vee” shaped wheels (with low friction bearings). Therefore, the bodies could slide down unidirectionally. Near the bottom of the slope a “rubberized gorse hair” filled basket is mounted to the slope to catch the body before it can run out onto the horizontal bottom of the wave tank. One cable about 3 mm in diameter is used with a winch to retrieve the body and position it for the next experiment. A thin line connects the body to a rotating potentiometer that provides measurement of the body’s position and velocity time histories.

For a given initial wedge position, the specific weight γ (defined as $\gamma = \rho_{wedge} / \rho_{water}$) is varied from $\gamma = 1.5$ to $\gamma = 3.73$. The initial wedge elevation (Δ) ranged from subaerial to submerged ($-0.33 \le \Delta / b \le 0.5$). For series A, the maximum runup height was obtained by observing the video images of the leading edge of the runup tongue. One runup gauge is located at 30.5 cm to one side of the centerline (Gauge #2). The other runup gauge is located at 61 cm to the centerline (Gauge #3). In this way the runup time history could be obtained. In all cases the maximum runup was determined before wave reflections from the sidewalls returned to the centerline of the tank. Four wave gauges (Gauge #4 to Gauge #7) were installed in the offshore direction to measure the offshore waves. A wave gauge array with six wave gauges (Gauge #8 to Gauge #13) were installed in the lateral direction to measure lateral propagating waves.

5.5 Numerical Setup

The numerical simulations are carried out in a numerical wave tank. In this study, the numerical simulation is focused on simulating the type A landslide generated waves. The numerical tank is designed to simulate half of the original domain. In order to make sure that the half-domain simulations would not hurt the results, one subaerial landslide case has been conducted in both full and half domains. The results show that the differences between the full and half domains are smaller than 1% in terms of the time history runup heights and wave heights. This indicates that in these landslide simulations the effects of vortex shading on the runup height and wave heights are small and can be neglected, and it is save to adopt the half domain to save the computational time. Thus, the width of the numerical tank is 1.85 m.

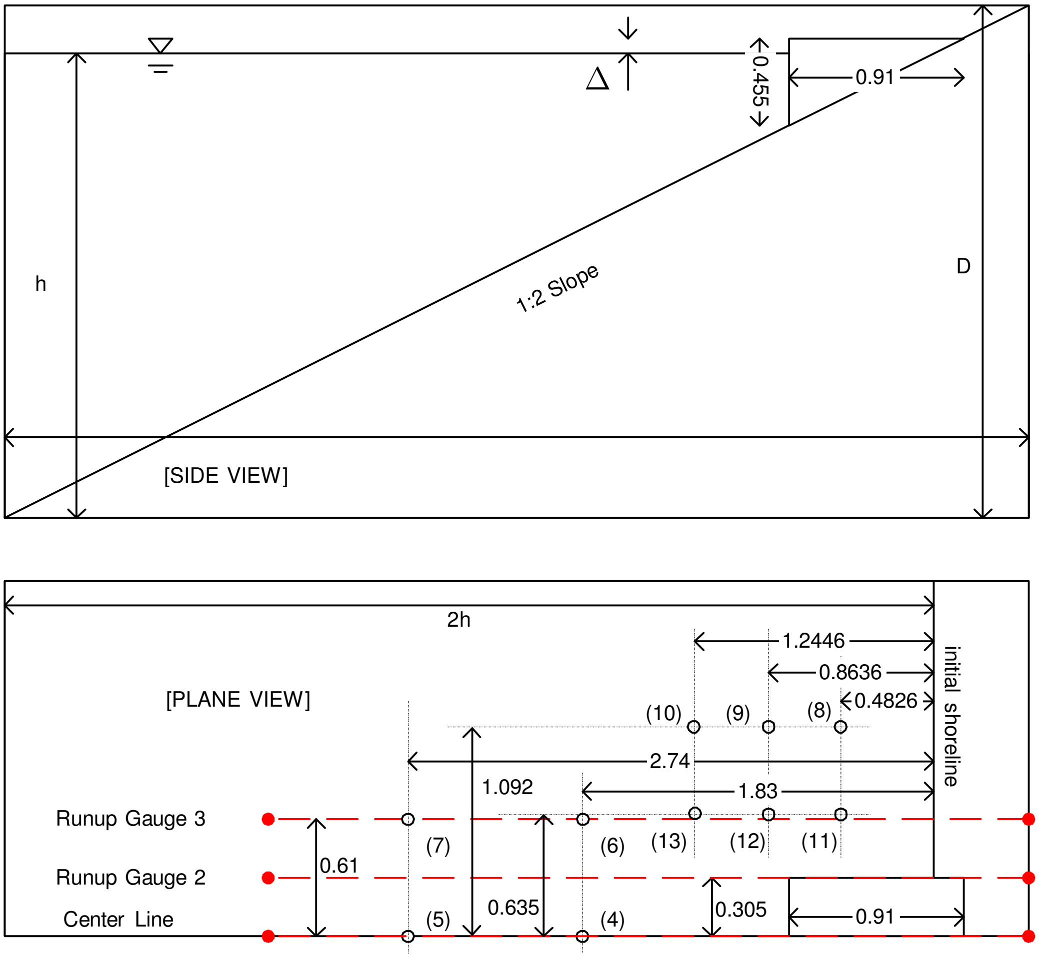


Figure 5.6: The numerical setup and gauge positions. Unit is in meter.

In this study, our discussion will be focused on the shoreline movement and near field surface profile. Therefore, the length of the domain is designed to be shorter than the laboratory setup. See Figure 5.6 for details. The maximum still water depth (h) varies for different cases and is proportional to the total displacement of the wedge. All these efforts are made to have a better resolution and save computational time. The number of grid points is fixed. For all cases in this study, the number of grid point is 60 in the streamwise direction, 50 in the spanwise direction, and 60 in the vertical direction. A non-uniform grid is used in the streamwise and vertical directions. The finest cell is located at the onshore corner (Figure 5.6) to have a finer resolution for the runup simulations. Coarser resolution is applied to the offshore deep water corner because the effect on the free-surface is less important. The boundary conditions of the side walls are impermeable free-slip walls. Therefore, the landslide generated waves will be reflected by the downstream wall, which will not be seen in the laboratory experiments. However, since the maximum and minimum runup as well as the major free-surface elevations will occur before the reflected waves reach the gauges, the reflecting wall boundary condition is still a good choice to simplify the numerical setup. The ceiling boundary condition is a zero-pressure open boundary condition. The 1:2 slope is designed to extend from upper-right corner to lower-left corner (Figure 5.6), so the solid material will occupy 50% of the cell volume on the slope. This design ensures that the effective cell volume will not be smaller than 50% of the total cell volume, therefore the time marching will not be limited by certain cells with extreme small effective cell volumes due to the Courant number restriction. The time step is determined dynamically based on the Courant number criteria which is 0.45 in this study.

Similar to the laboratory setup, three runup gauges are installed close to the moving wedge; four wave gauges numbered from #4 to #7 are installed in the offshore direction; and one wave gauge array with six wave gauges numbered from #8 to #11 is installed on the lateral side of the moving wedge. See Figure 5.6 for details. However, the gauge locations might not coincide with grid centroid. Thus, the nearest neighbor interpolation method is used to interpolate gauge data.

In this study, the displacement of the moving wedge is obtained from the laboratory measurements. Therefore, the numerical simulations share the same initial accelerations and friction coefficient with the laboratory experiments.

5.6 A Subaerial Landslide Simulation

In this study, we have conducted forty five numerical experiments based on different initial elevations and specific weights of the series A moving wedges. Before discussing all the numerical simulations, we shall inspect and verify the numerical solution with the laboratory gauge data. One subaerial and one submerged landslide simulations will be examined before we discuss all the numerical results. The free-surface profiles, velocity fields, and shoreline movements will be presented in the selected subaerial and submerged landslides. After that, the maximum runup heights of all the numerical solutions will be compared with the laboratory data. The cross-comparison of the shoreline movement will be presented and discussed.

In this section, a subaerial landslide simulation will be examined to verify the accuracy of the numerical model. The detailed observations will be presented and discussed.

5.6.1 Setup

In this subaerial landslide case, the initial elevation of the wedge is 0.454 m above the initial still water level ($\Delta = 0.454\text{m}$). The specific weight $\gamma = \rho_{soild} / \rho_{water} = 3.43$. The numerical simulation is conducted in a domain with size $(x, y, z)_{\text{Domain size}} = (6.6, 3.7, 3.3)$, where the unit is meter and (x, y, z) represent the directions of off-shore (x), lateral (y), and vertical (z) respectively. Non-uniform grids are used in x and z directions with the finest grid size $(dx, dz)_{finest} = (0.0391,\ 0.0196)$ located at the corner near the initial shoreline. A uniform grid was adopted in the lateral direction. The grid has been specially designed so that $dx = 2dz$. The slope was located diagonally across the x, z plan. Therefore, on the slope and without considering the moving wedge, the effective cell volume is always equal to 50% of the local cell volume. This design could help us to clearly identify the shoreline location without being interfered with the irregular effective cell volume.

The trajectory of the slide motion is obtained directly from the laboratory measurement. Figure 5.7 shows the time history of the displacement (S) and speed of the slide motion for current case.

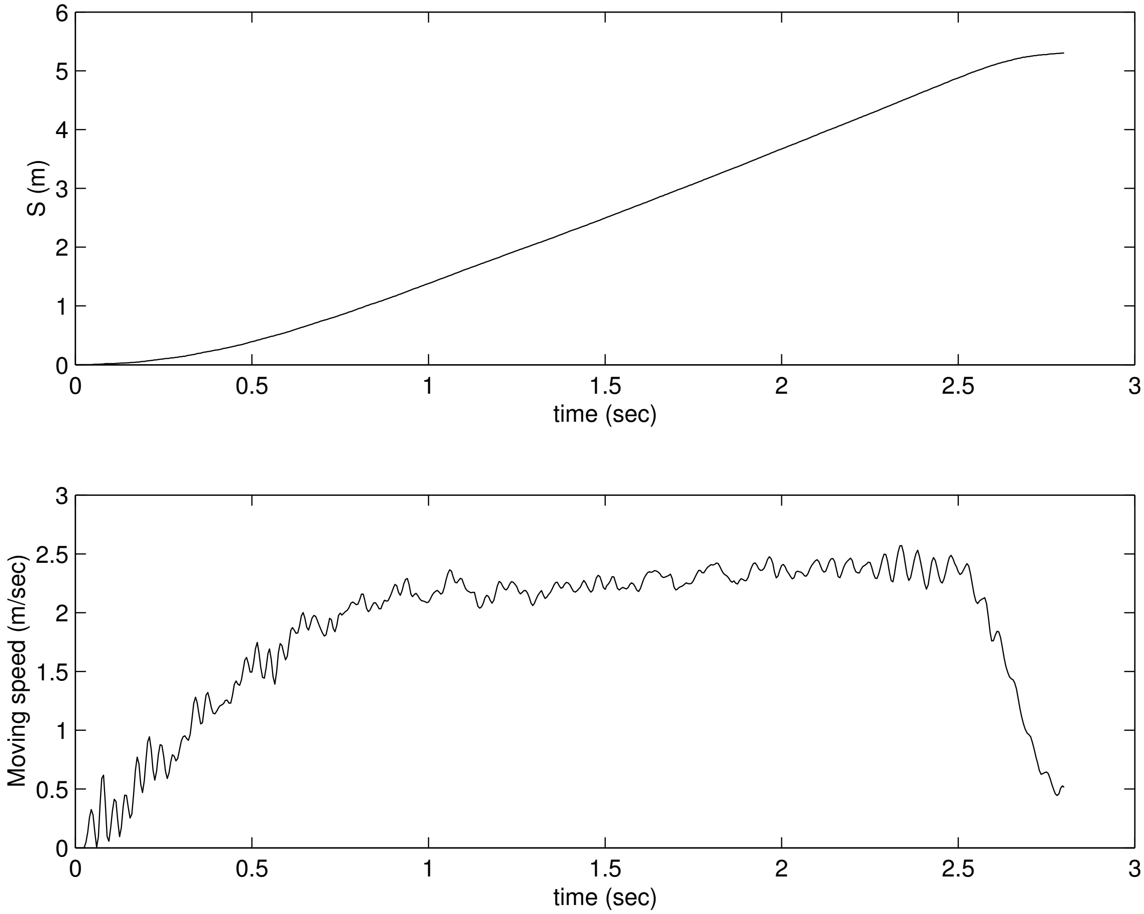


Figure 5.7: The displacement (S) and the speed of the moving slide. In this subaerial case, the initial wedge position (Δ) is 0.454 m above the still water surface. The specific weight (γ) of the wedge is 3.45.

5.6.2 Initial and Boundary Conditions

The initial condition is zero velocity everywhere including the moving slide. The boundary condition is no-flux and free-slip everywhere except on the slope and on the surface of the moving slide. Because of the partial cell treatment, the boundary condition of the slope and the surface of the slide will be treated as non-slip and no-flux that are very close to the real laboratory setup. The downstream face is specified as a free-slip wall, which will reflect energy and contaminate the solutions after a certain period. However, our goal is to investigate the maximum runup and the leading off-shore waves which happened before the reflecting waves reach the investigation area. Therefore, in this study, we still adopt the free-slip wall boundary condition on the downstream faces to simplify the setup of the problems.

5.6.3 The Runup Gauges

In the laboratory setup, two runup gauges are installed on the slope to provide the time history runup information. The first runup gauge (Gauge #2) is located right beside the edge of the wedge, the other one is located one wedge width away from the centerline cross-section. In the numerical simulation, we also installed two runup gauges at the same locations as the laboratory setup. Figure 5.8 shows the comparison between the numerical results and laboratory data. The solid lines are the numerical results and the broken lines are the experimental measurements. The numerical runup height is determined by the contour line where the water occupies 50% of the effective cell volume. A very good agreement is shown in the Gauge #2 comparison. Gauge #3 shows that the numerical solution slightly over-predicts the maximum runup height and the disagreement is about 10%. Overall, the numerical simulation is able to capture the maximum runup height in the near field region, and most importantly, the model is able to accurately predict the arrival time of the maximum runup height.

5.6.4 Four Fixed Wave Gauges

There are four fixed wave gauges installed in front of the slide and labeled as gauge #4 ~ #7 (see Figure 5.6). Figure 5.9 shows the comparison of wave gauge data. Again, the solid lines are the numerical solutions and the broken lines are the numerical measurements. Gauge #4 and #6 are the ones close to the shoreline. The comparison shows that the numerical simulation successfully predicts the leading wave height as well as the phase speed. Gauge #5 and #7 show that the numerical solutions under-predict the leading wave height. However, the numerical model is able to predict the phase of the waves.

5.6.5 Wave Gauge Array

Wave gauges #8 ~ #13 are installed on the side of the moving slide to record the lateral movement of landslide generated waves. Figure 5.10 shows the numerical results and laboratory data. The results show that the numerical solutions successfully predict the wave heights of the leading and secondary waves. The numerical solutions also accurately capture the phase of the waves. However, from gauge #8, #9, and #11, around time = 1.75 sec, the comparisons show that the numerical solutions cannot predict some small scale motions. This might be due to the resolution issue, or might also be the experimental errors because there are a lot of breaking and bubbles. Fortunately, these errors have no significantly effects on the predictions of the secondary waves.

5.6.6 Snapshots

From the comparisons with the laboratory gauge data, the landslide numerical model has been examined to provide useful information to investigate the subaerial landslide problems. In this section, we shall provide a sequence of snapshots of the surface elevation, velocities, and shoreline movements, which are not measured in the experiments.

5.6.6.1 Free-surface elevation

Figure 5.11 shows the snapshots of the free surface elevation. The initial shoreline is at the intersection of $x = 0$ and $z = 0$. The slide first pushes the water in front of the slide (time = 0.0 ~ 0.8 sec). Then the slide sinks into water and the water caves in due to the low pressure in the wake zone (time = 0.8 ~ 1.2 sec). With the slide keeping on sinking, the merged waves in the wake zone generate a strong reflecting wave (time = 1.2 ~ 3.0 sec). This strong motion is the key source to the maximum runup and to the largest leading wave towards the off-shore direction. After time = 3.0 sec, the solution is gradually contaminated by the waves reflected from the side walls and will be excluded from this discussion.

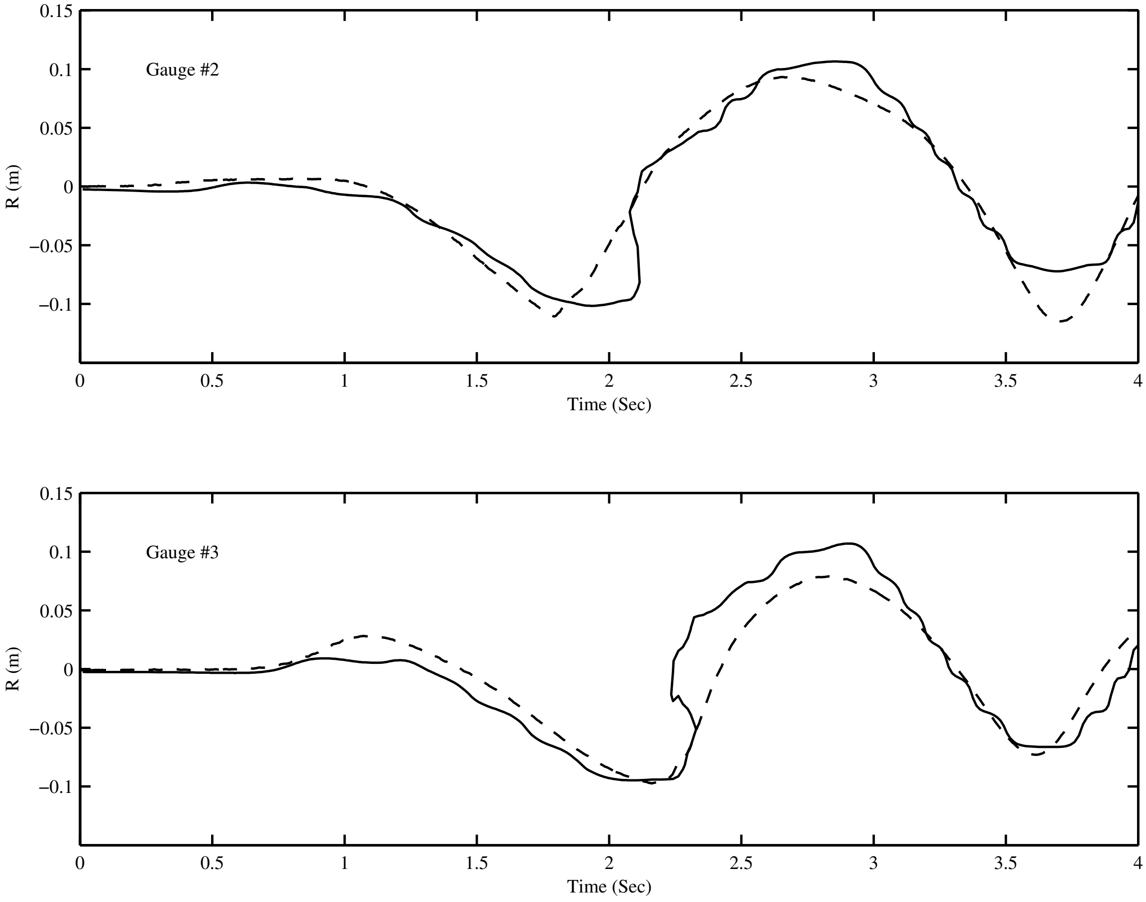


Figure 5.8: The comparison between numerical results (solid lines) and experimental data (broken lines) for the time history runup height at Gauge #2 and Gauge #3. $\Delta = 0.454\text{m}$. $\gamma = 3.43$.

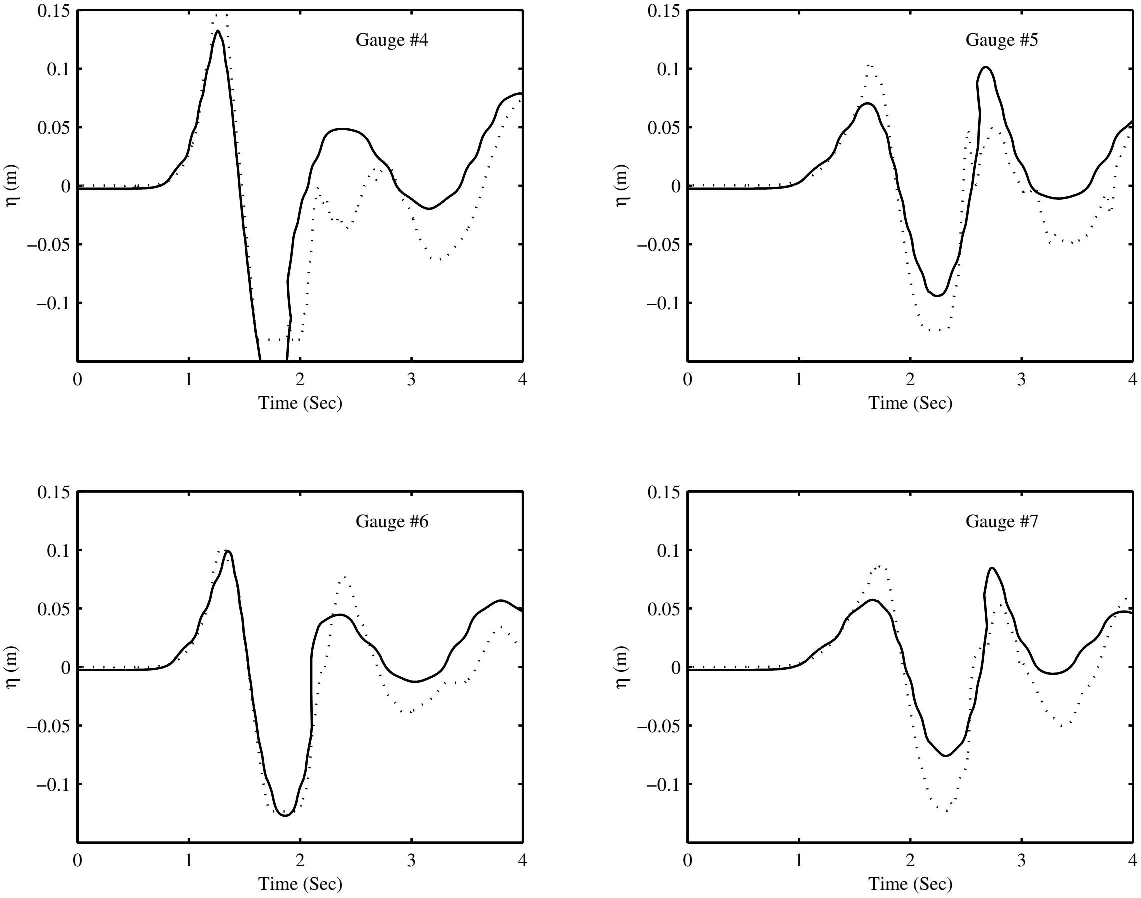


Figure 5.9: The comparison between numerical results (solid lines) and experimental data (broken lines) for the time histories of free surface fluctuations at wave gauge #4 ~ #7; Δ=0.454 m, γ=3.43. The coordinates for gauges are: Gauge #4: (x, y) = (1.83, 0); Gauge #5: (x, y) = (2.74, 0); Gauge #6: (x, y) = (1.83, 0.61); Gauge #7: (x, y) = (2.74, 0.61). The unit is in meter.

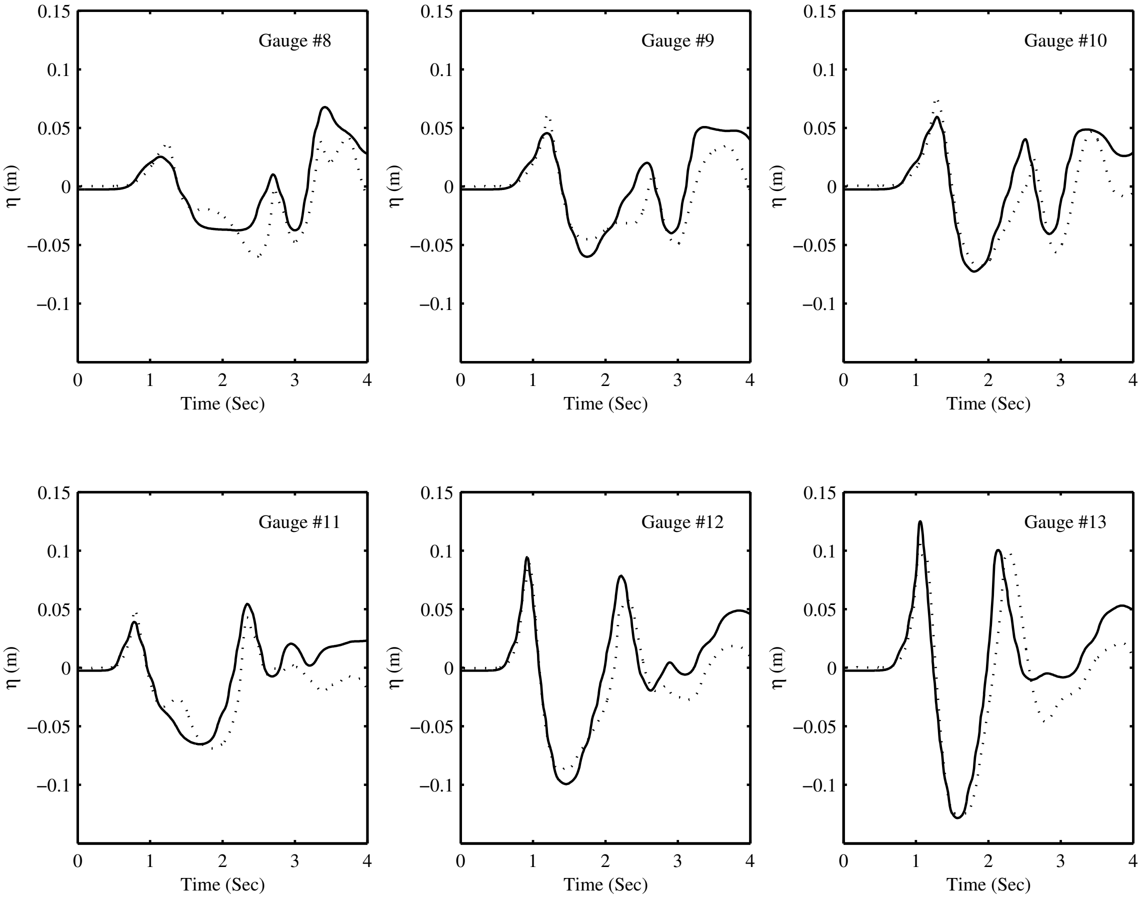


Figure 5.10: The comparison between numerical results (solid lines) and experimental data (broken lines) for the time histories of free surface fluctuations at wave gauge #8 ~ #13; Δ=0.454 m, γ=3.43. The coordinates for gauges are: Gauge #8: (x, y) = (0.4826, 1.092); Gauge #9: (x, y) = (0.8636, 1.092); Gauge #10: (x, y) = (1.2446, 1.092); Gauge #11: (x, y) = (0.635, 0.4826); Gauge #12: (x, y) = (0.635, 0.8636); Gauge #13: (x, y) = (0.635, 1.2446). The unit is in meter.

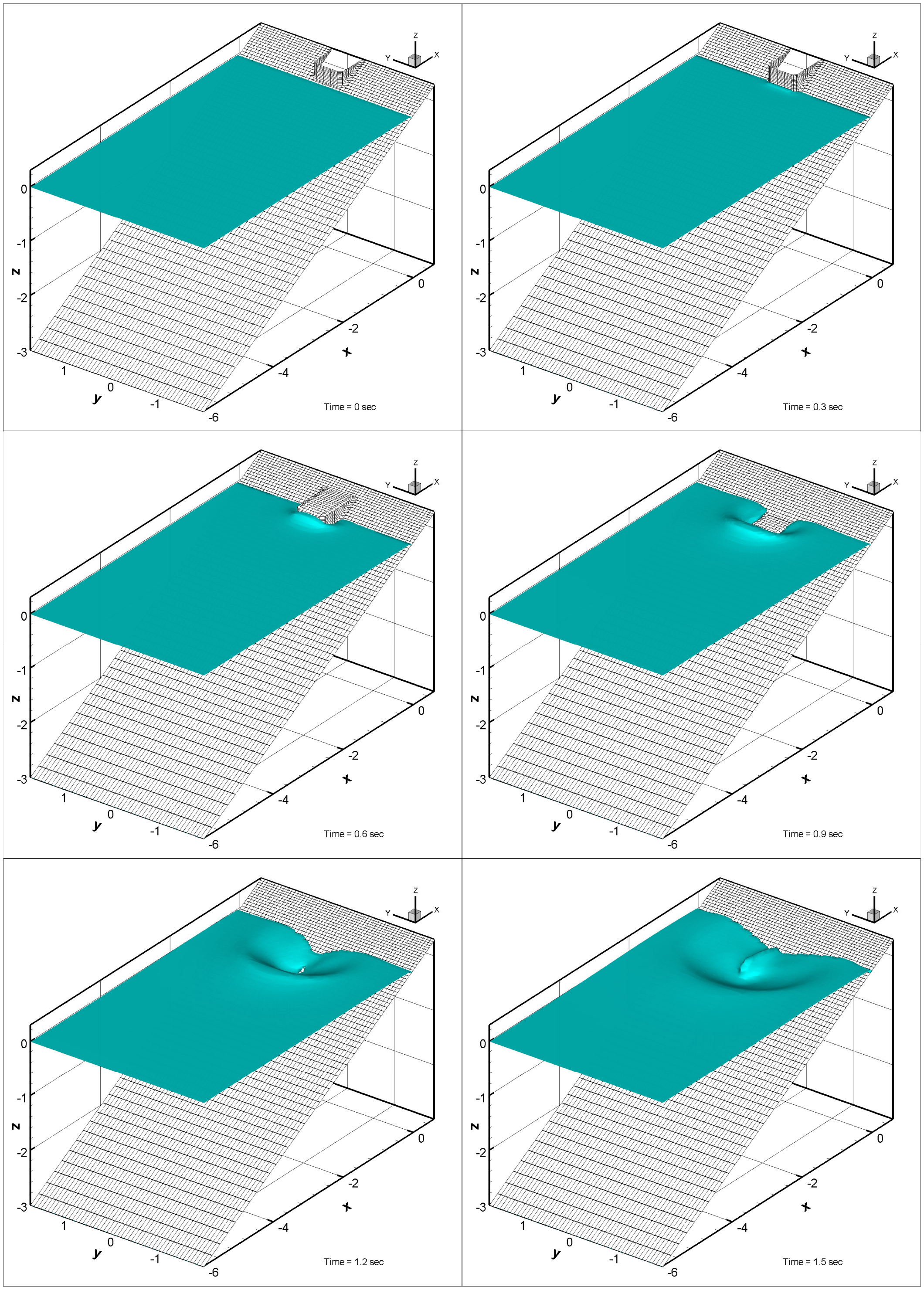
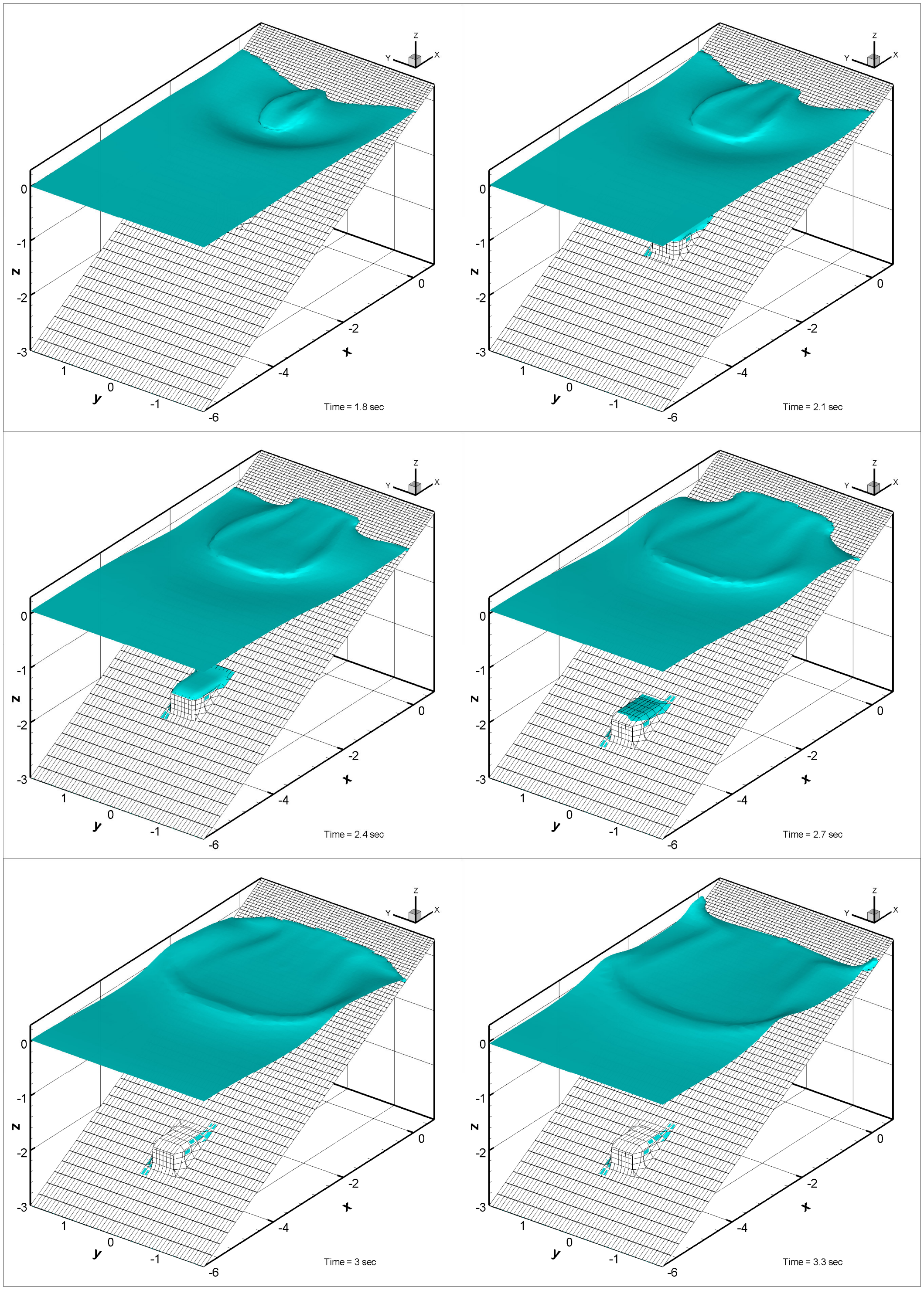


Figure 5.11: Snapshots of free-surface profile for the sliding wedge with $\Delta = 0.454$ m, and $\gamma = 3.43$. The unit is in meter.

Figure 5.11 (continued)



5.6.6.2 Velocities

Figure 5.12 presents the detailed velocity vectors at the centerline cross-section. From time = 0.0 ~ 0.8 sec, the velocities are generated by the pushing process of the slide. After time = 0.8 sec, the elevation of the top of the slide is lower than the still water depth, and part of the water floods into the wake area. At time = 1.2 sec, the slide is fully submerged. After that, the slide provides momentum to water not only from the pushing process but also from the low pressure in the wake area. The low pressure mainly results in adding a negative source function in the pressure-Poisson equation. A very complex flow pattern can be observed right after the slide has fully merged into water (time = 1.2 ~ 1.5 sec). This complex velocity distribution indicates that the depth-integrated equation models might have a larger error during this period. From time = 2.1 to 2.7 sec, the velocity distribution shows that a strong flow current has been generated by the slide motion, and has a thickness about 1.5 ~ 2.0 times the front face height (H_{slide}) of the slide. Above this flow current, there exists a returning current and generates a negative wave. Between these two currents, there exists an eddy which can be clearly seen from time = 2.1 sec to 2.7 sec. The size of the eddy depends on the moving speed of the slide. In this case, the eddy size is about one front face height (H_{slide}) of the slide. After t = 2.7 sec, the slide stops. However, the current behind the slide still keeps on moving with the decreasing strength.

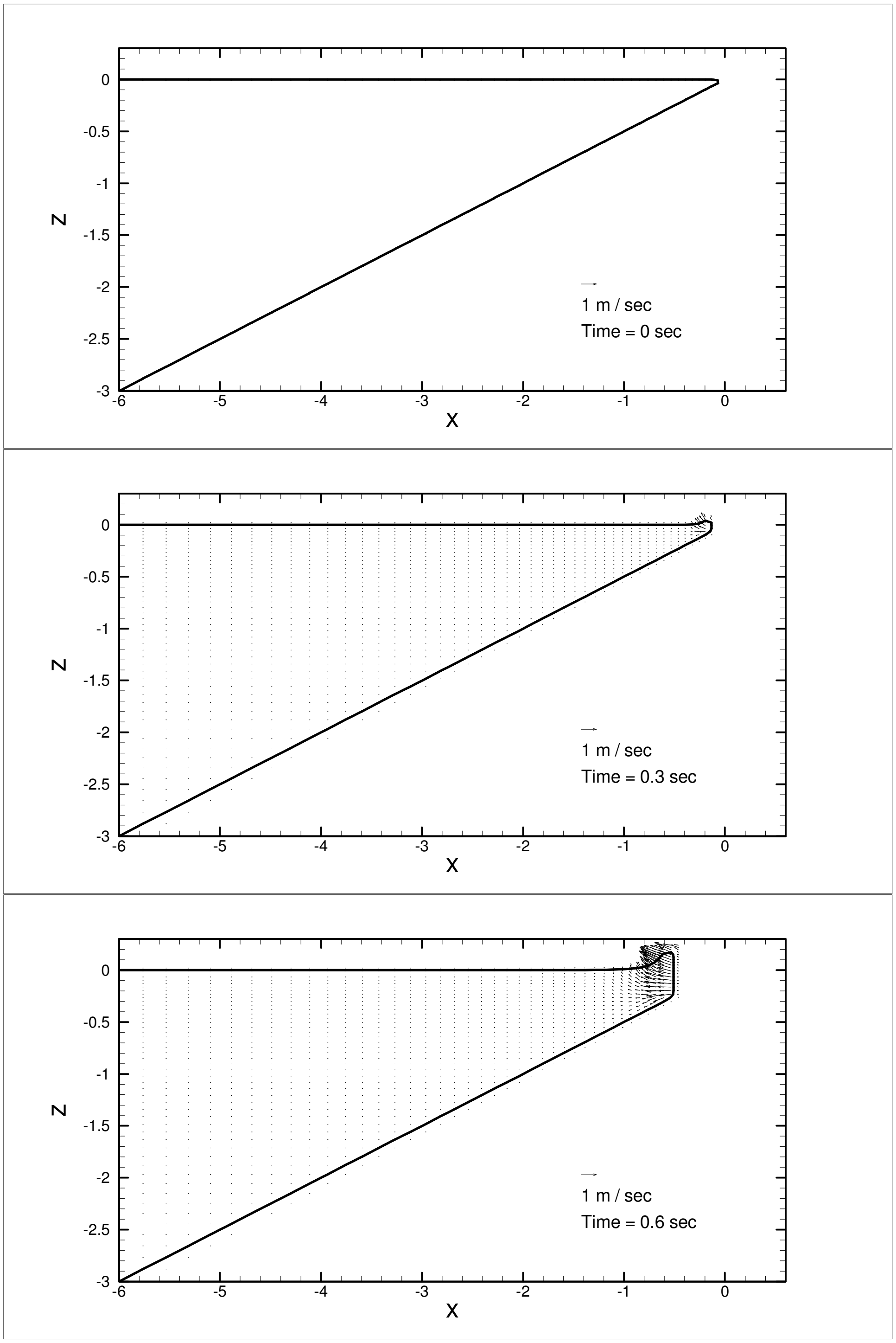


Figure 5.12: Snapshots of velocity vectors on the centerline vertical plane for the sliding wedge with Δ=0.454 m, and γ=3.43. The unit is in meter.

Figure 5.12 (continued)

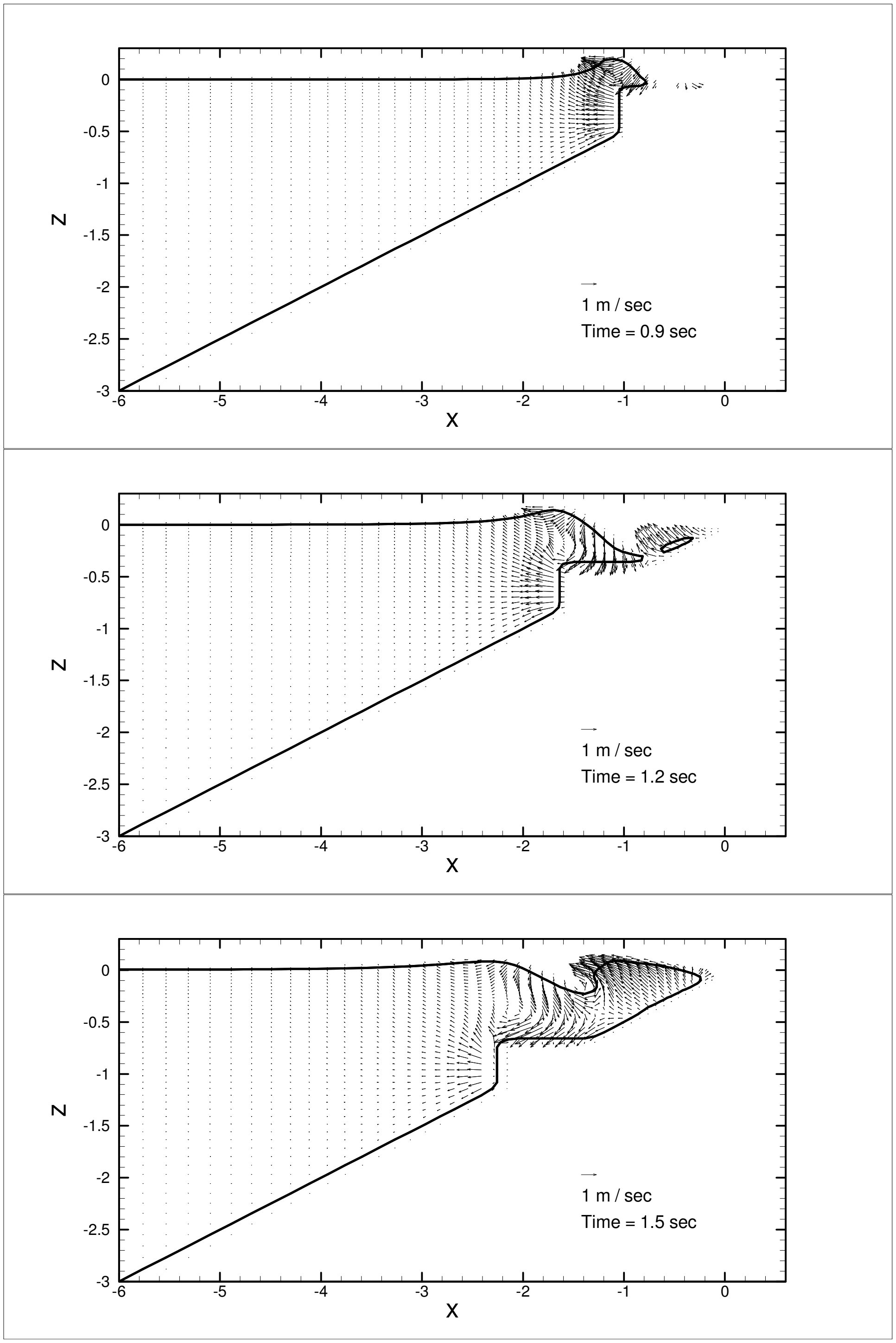


Figure 5.12 (continued)

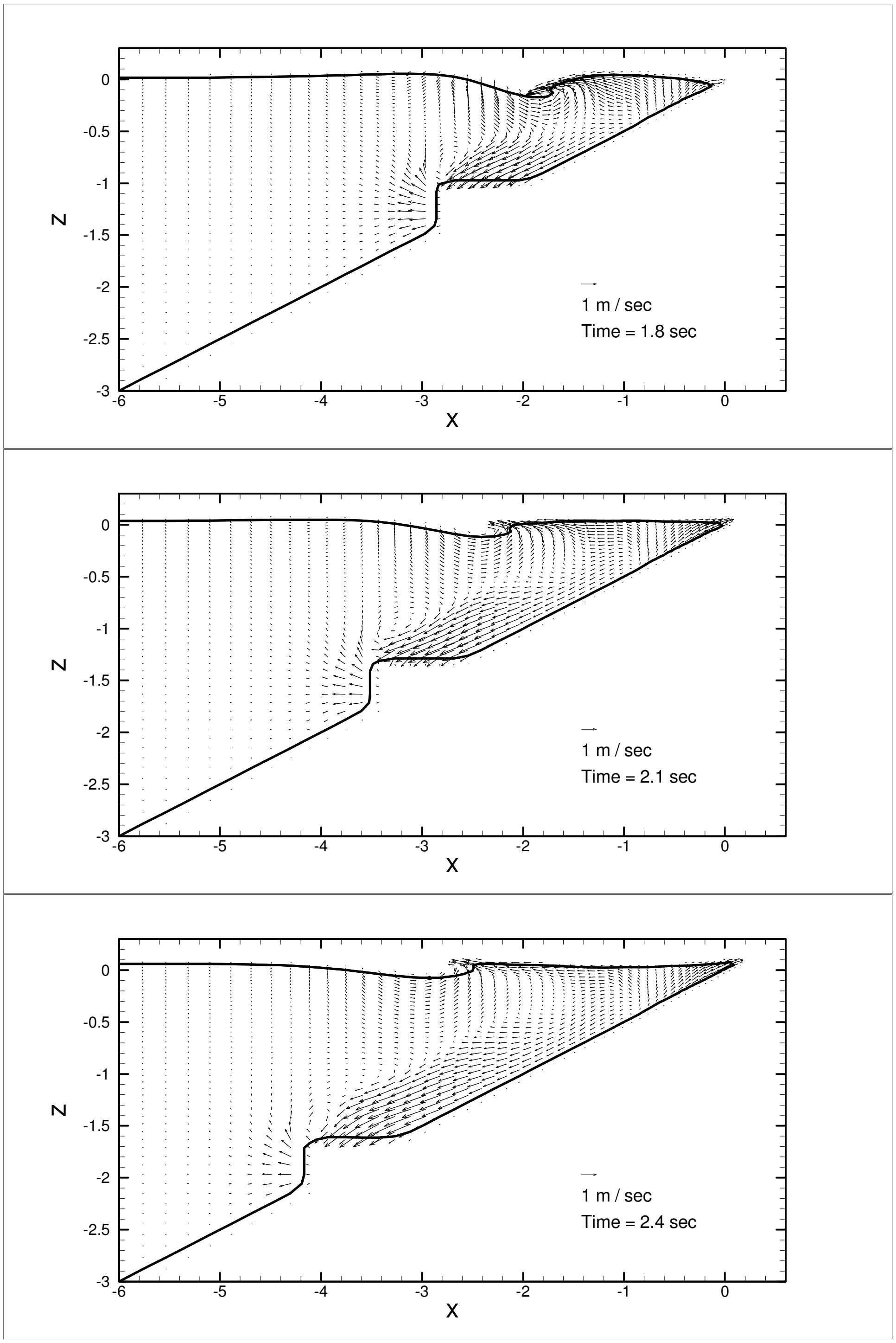
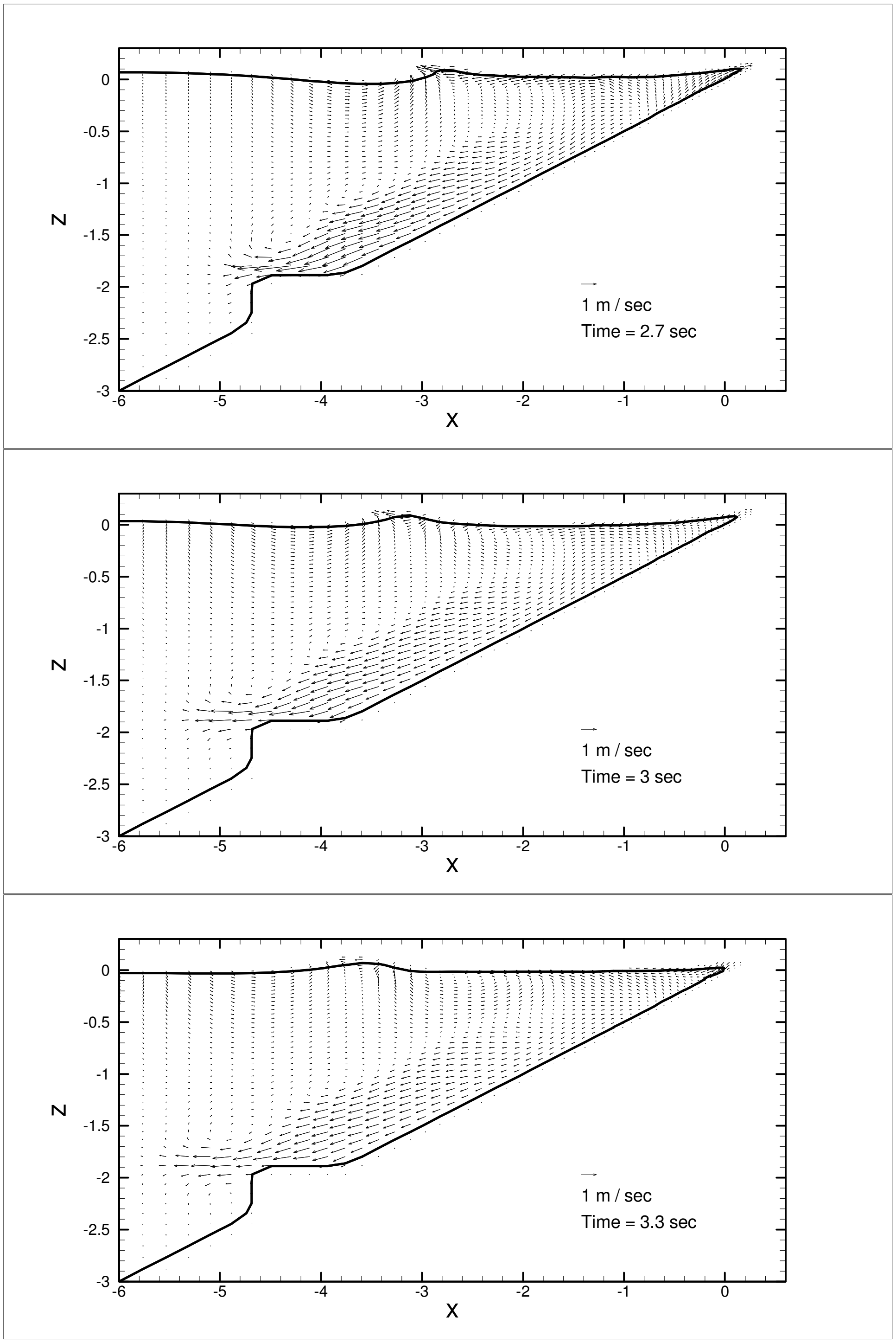


Figure 5.12 (continued)



5.6.6.3 Shoreline movement

Figure 5.13 shows the snapshots of the shoreline movement. The snapshot at time = 1.2 sec shows that the slide has fully submerged into water and an air bubble is trapped there. After time = 1.5 sec, the shoreline starts to move up, spreads out, and then reaches the maximum runup height at time = 2.7 sec. From the sequence of the snapshots we observe that at time = 2.7 sec, both the centerline and the near field region ($-1\text{m} \leq y \leq 1\text{m}$) reaches the maximum runup. Therefore, the intersection of the shoreline and z = 0 m at time = 2.7 sec can be treated as the maximum inundation area.

5.6.7 Cross-Sections

Here we provide the velocity distributions on different cross-sections at time = 1.5 sec. At time = 1.5 sec, the slide is just fully submerged into water and generates a complex flow pattern. As we have addressed before, at this moment, the depth-integrated type wave model will have a larger error.

- x cross-section

Figure 5.14 provides the velocity distribution from x = -3.0 m to x = -0.25 m, where initial shoreline is located at $(x, z) = (0, 0)$. On each plot, the vectors are the fluid particle velocity vectors, and the dot lines are the water boundaries. The upper solid lines indicate the interface between air and water, and the lower solid lines indicate the interface between water and solid walls. From x = -3.0 m to x = -2.5 m, the water has been pushed away and convected to the ambient fluid. The region effected by the moving slide has a length scale about 3 times the width of the slide (W_{slide}) laterally and 2 times the height of the slide (H_{slide}). From x = -2.25 m to x = -1.5 m, the cross-sections are interacting with the slide. An important feature is that a strong negative wave has been generated by the slide. At these cross-sections, the fluid motion is generally towards the direction of leaving slide. However, starting from x = -1.75 m, the returning currents at two sides of the slide can be observed. These returning currents become significantly and dominate the flow domain from x = -1.25 m ~ x = -0.25 m. These returning currents then hit each other and generate a strong reflecting wave.

- y cross-section

Figure 5.15 shows the cross-section from y = 0.0 m to y = 0.8 m, where y = 0.0 m is the centerline cross-section. From y = 0.0 m to y = 0.3 m, the velocity distributions clearly show how the water has been pushed in front of the moving slide, and how the water has been dragged by the slide in the wake zone. At y = 0.6 m and y = 0.8 m, the returning currents can be also observed right above the slope.

- z cross-section

Figure 5.16 shows the cross-section from z = -1.3 m to z = 0.0 m. From z = -1.3 m to z = -1.0 m, the plots show the region affected by the slide motion. Clearly, the largest velocity happens in front of the slide and within one W_{slide}. From z = -0.9 m to z = -0.2 m, we are able to see the details of how the fluids merge together in the wake area. At z = -0.1 m and z = 0.0 m, the complex flow patterns can be seen and the fluid motion is mainly toward the offshore direction.

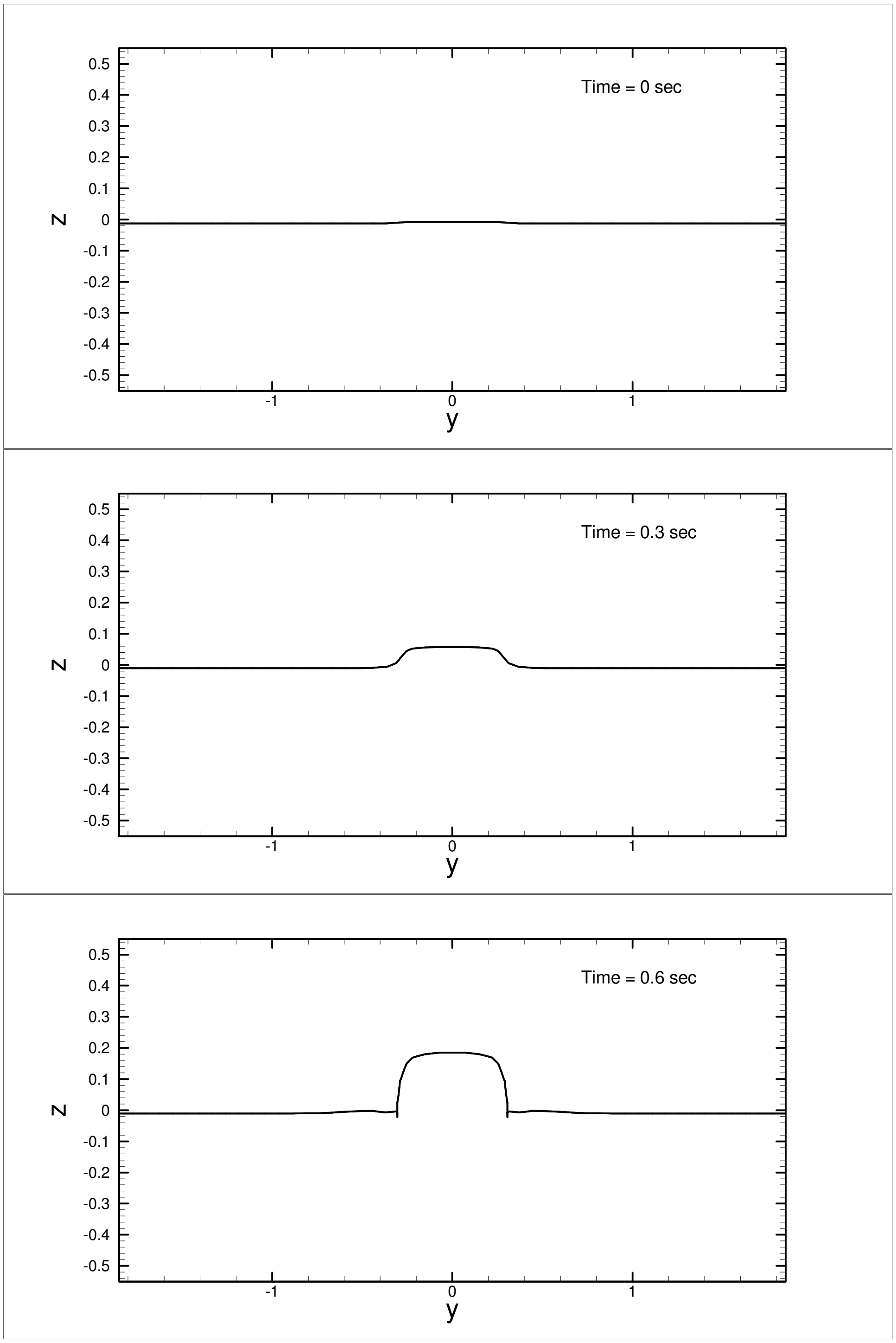


Figure 5.13: Snapshots of shoreline movement for the sliding wedge with Δ=0.454 m, and γ=3.43. The unit is in meter.

Figure 5.13 (continued)

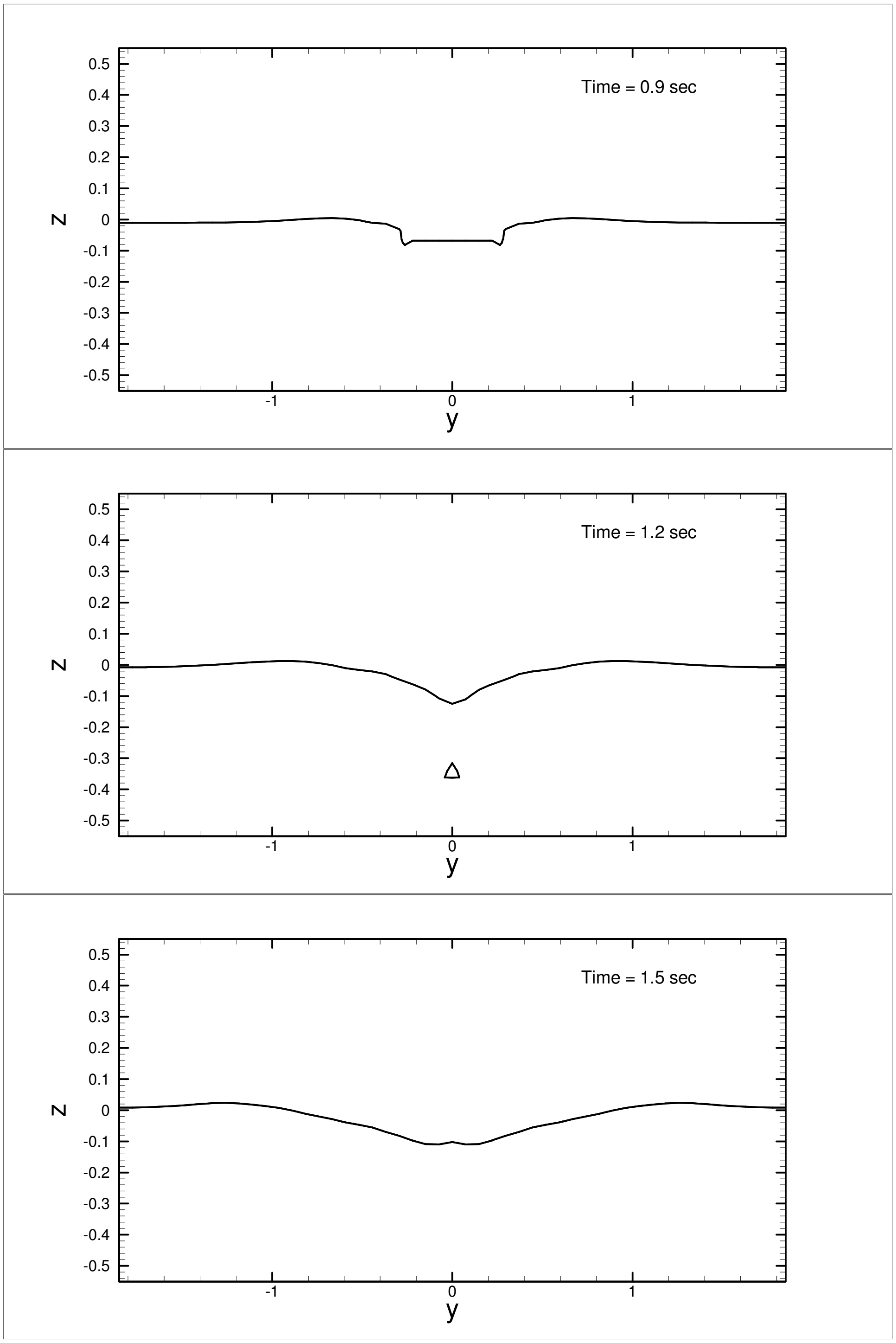


Figure 5.13 (continued)

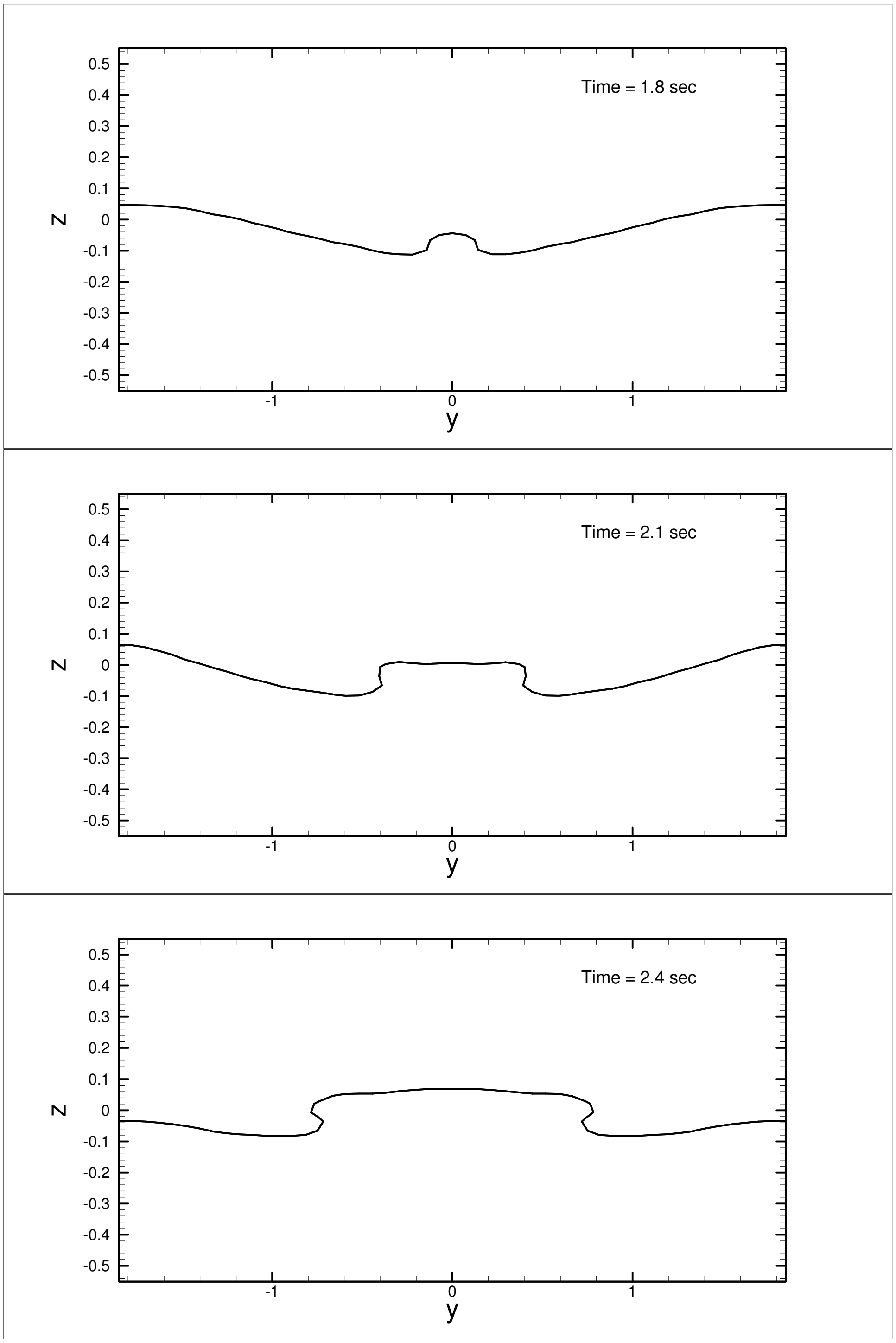
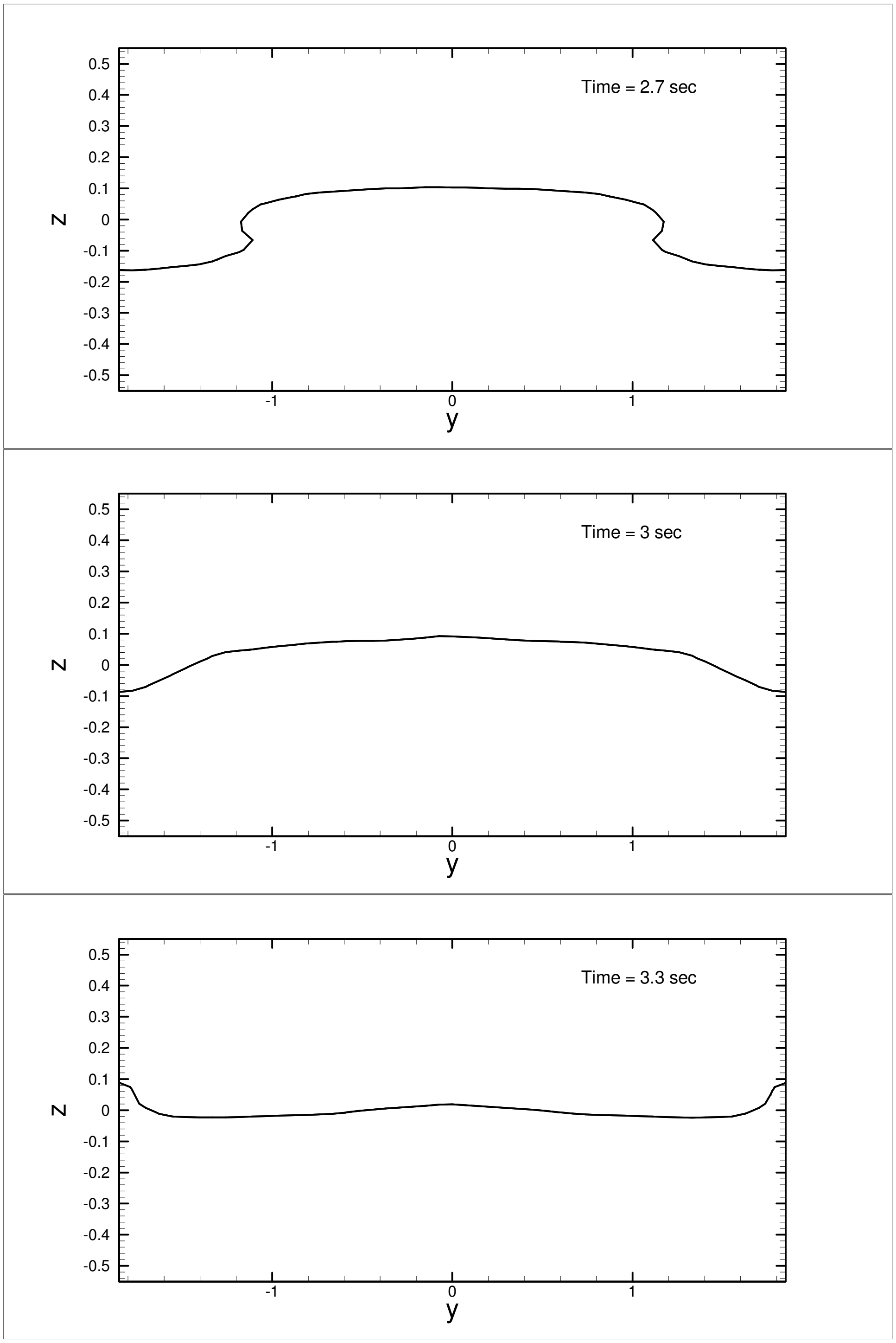


Figure 5.13 (continued)



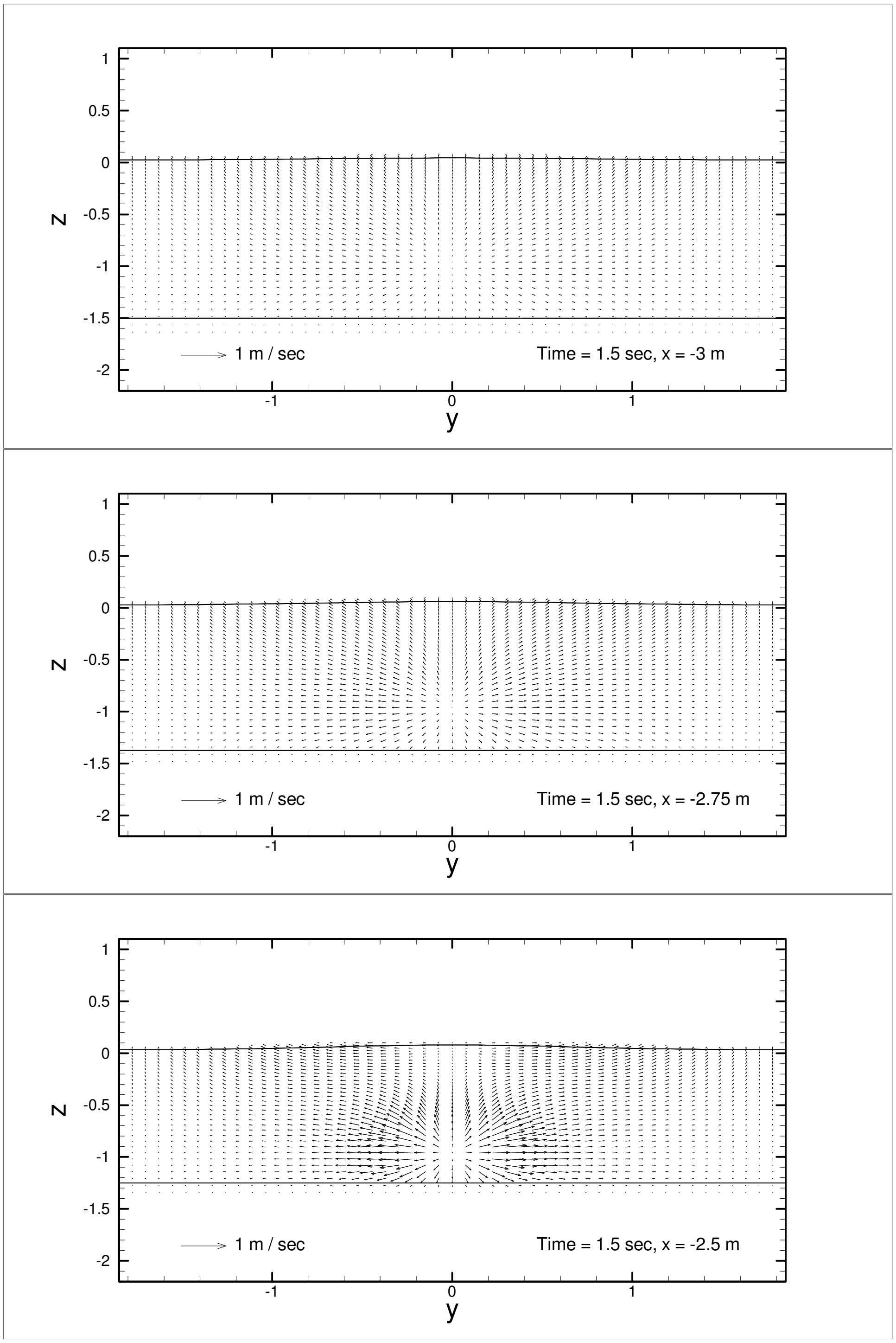


Figure 5.14: Snapshots of velocity vectors on the vertical x-planes at time = 1.5 sec for the sliding wedge with Δ=0.454 m, and γ=3.43. The unit is in meter. The magnitude of the reference vector indicates the speed of the wedge.

Figure 5.14 (continued)

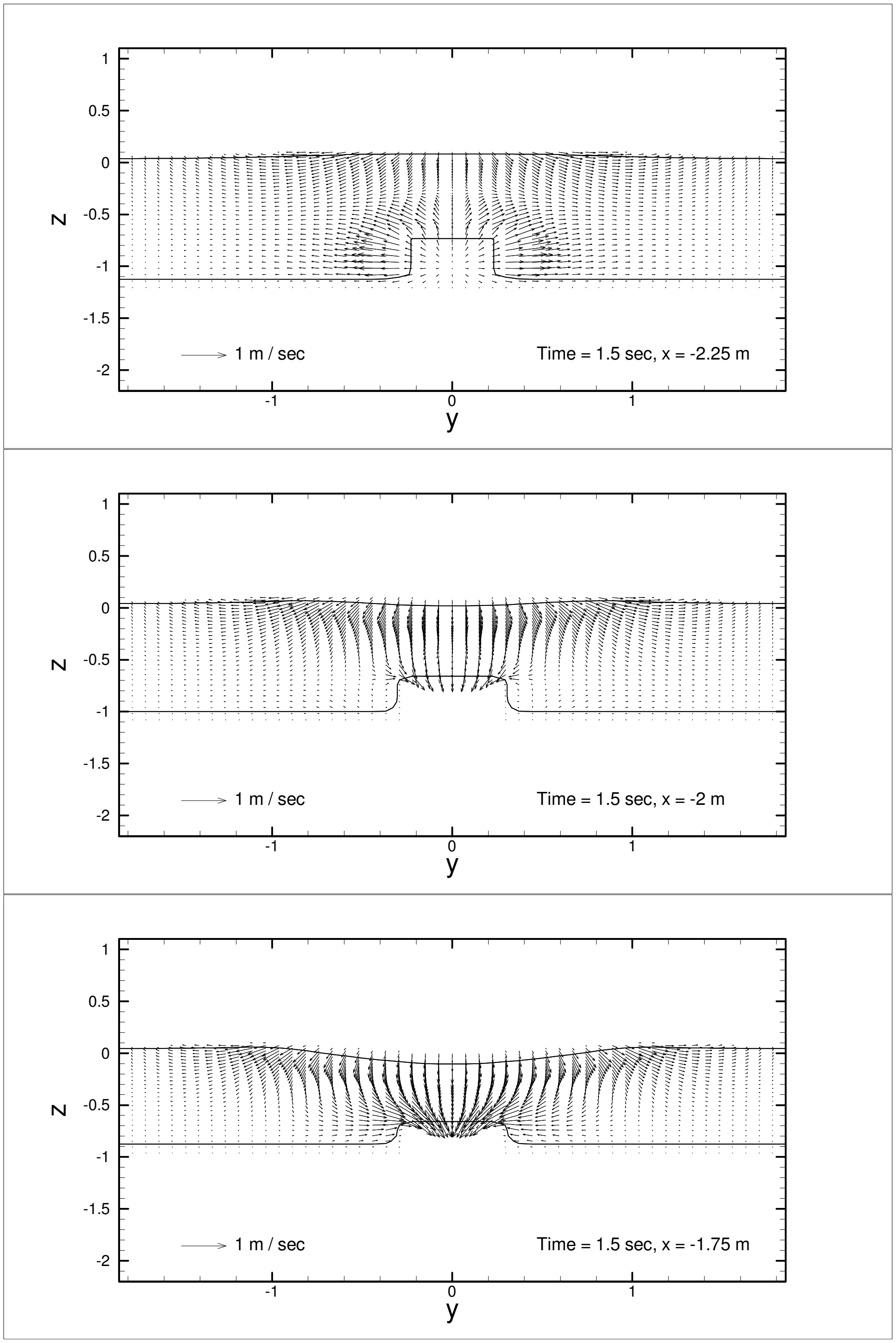


Figure 5.14 (continued)

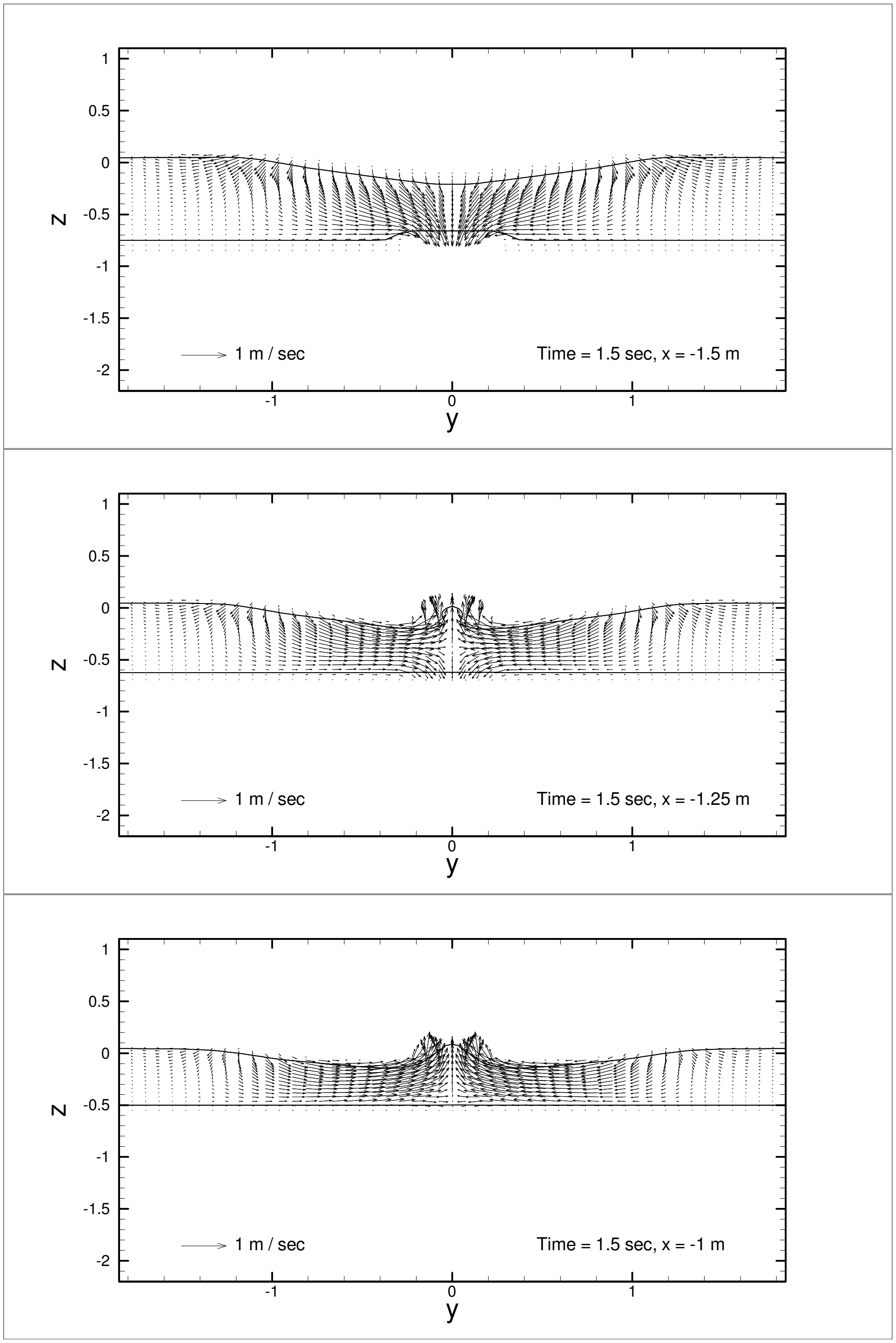
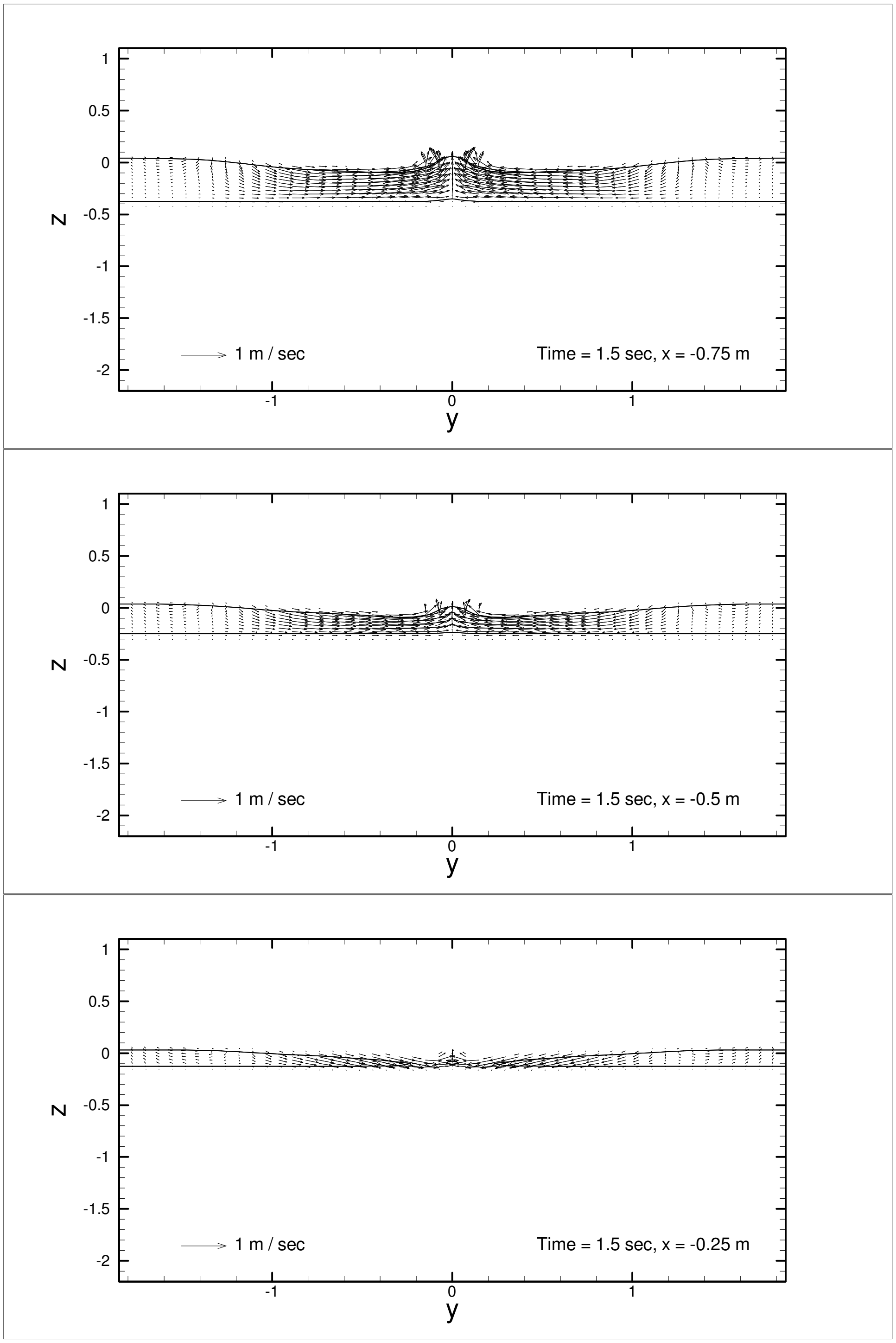


Figure 5.14 (continued)



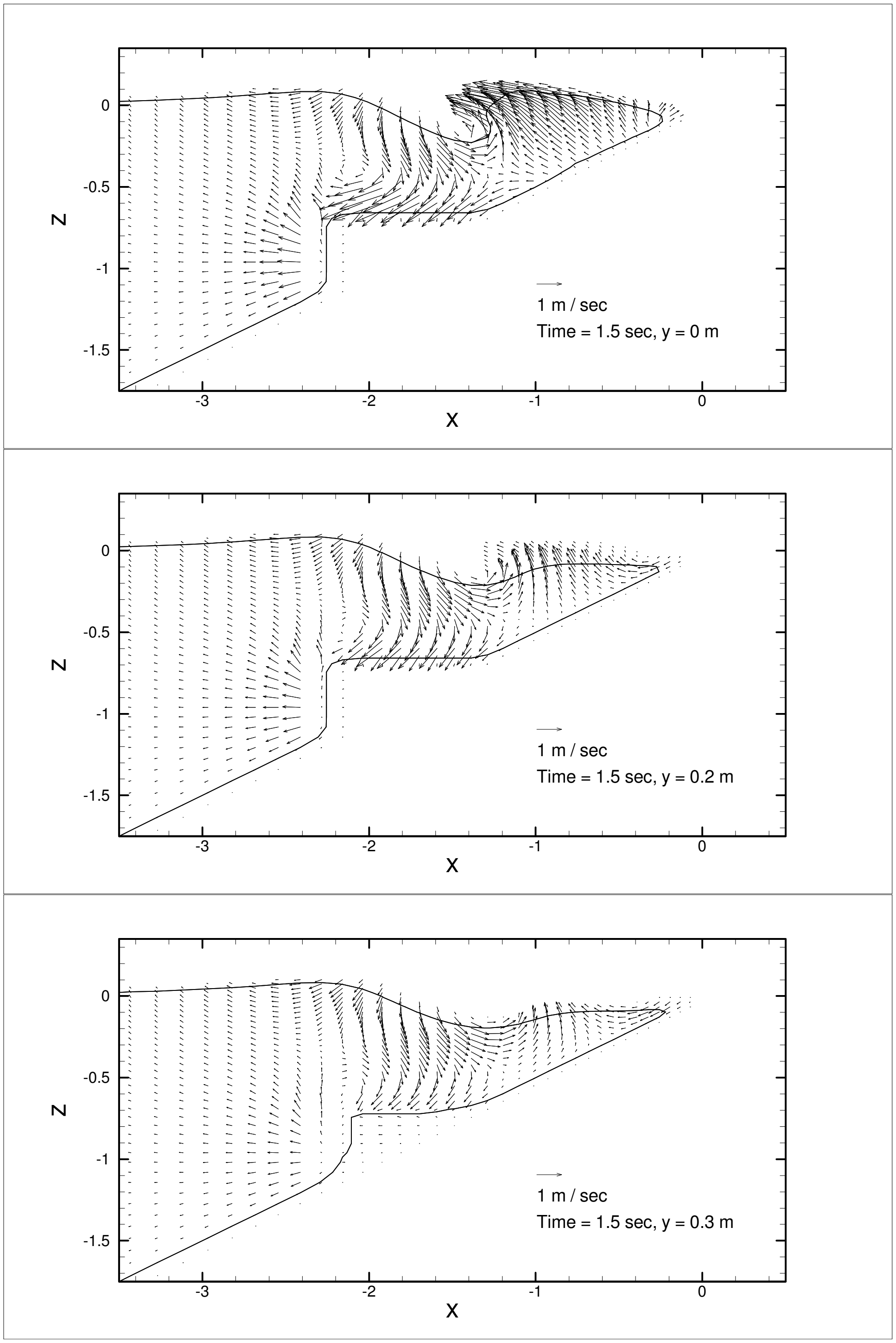
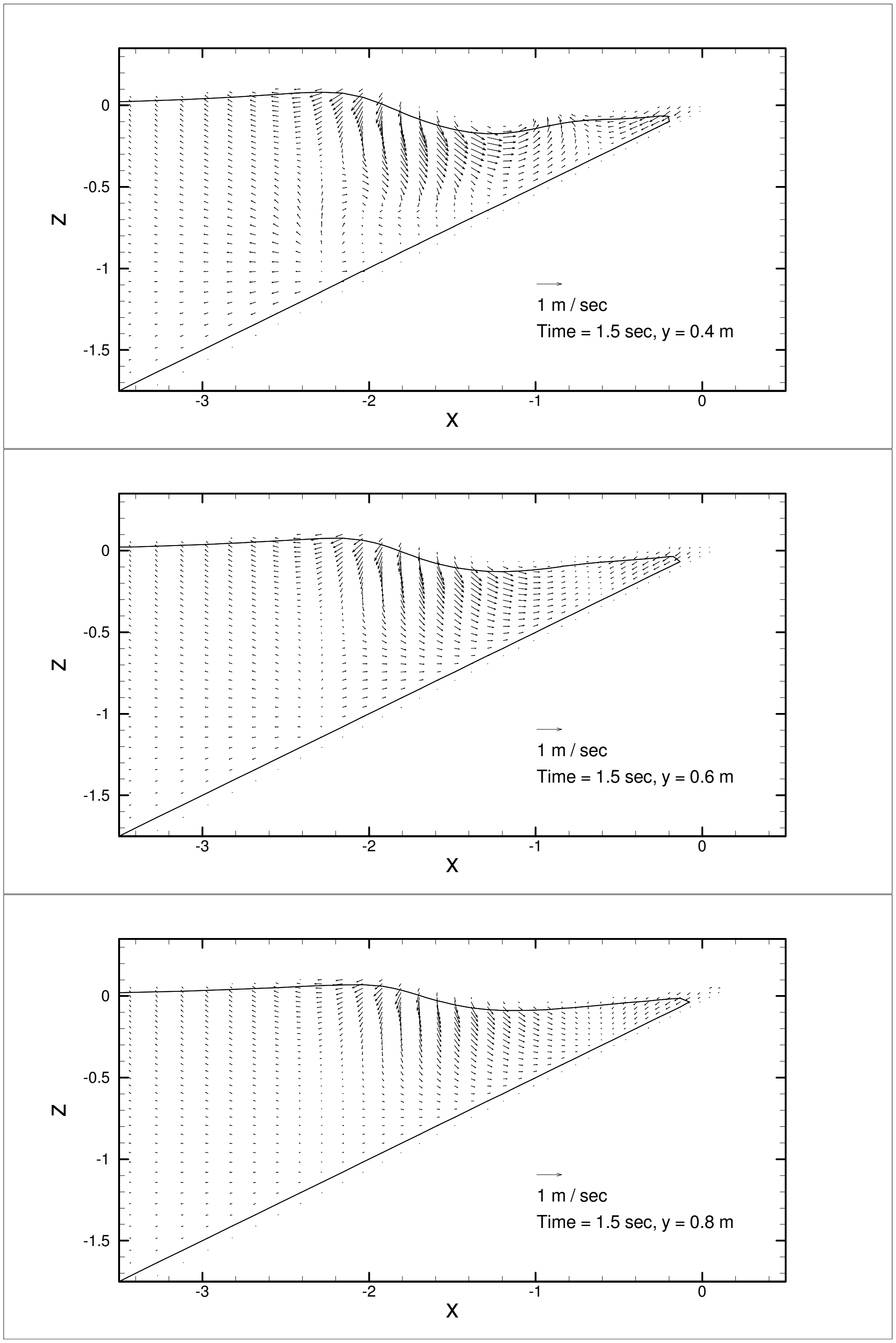


Figure 5.15: Snapshots of velocity vectors on the vertical y-planes at time = 1.5 sec for the sliding wedge with Δ=0.454 m, and γ=3.43. The unit is in meter. The magnitude of the reference vector indicates the speed of the wedge.

Figure 5.15 (continued)



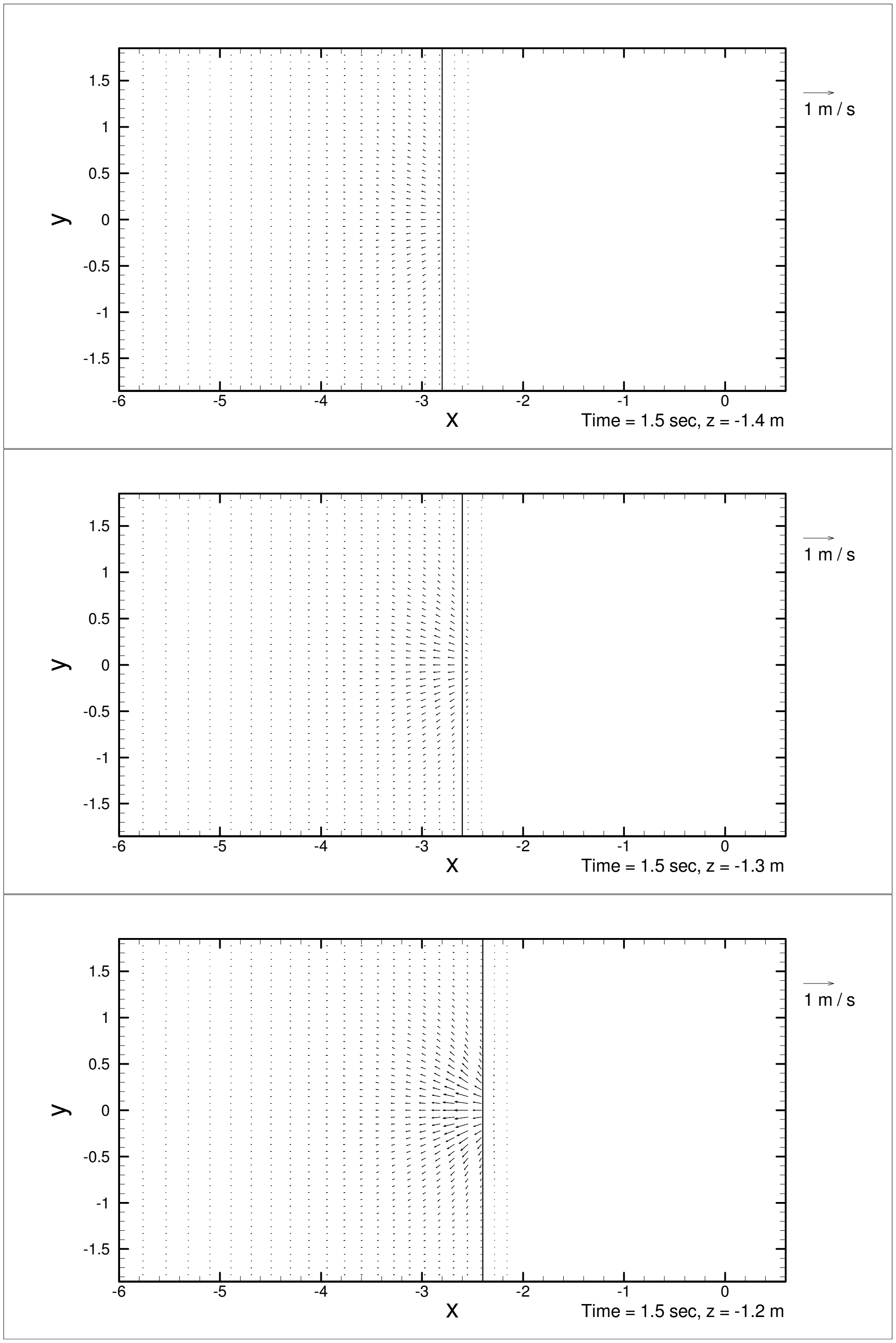


Figure 5.16: Snapshots of velocity vectors on the horizontal z-planes at time = 1.5 sec for the sliding wedge with Δ=0.454 m, and γ=3.43. The unit is in meter. The magnitude of the reference vector indicates the speed of the wedge.

Figure 5.16 (continued)

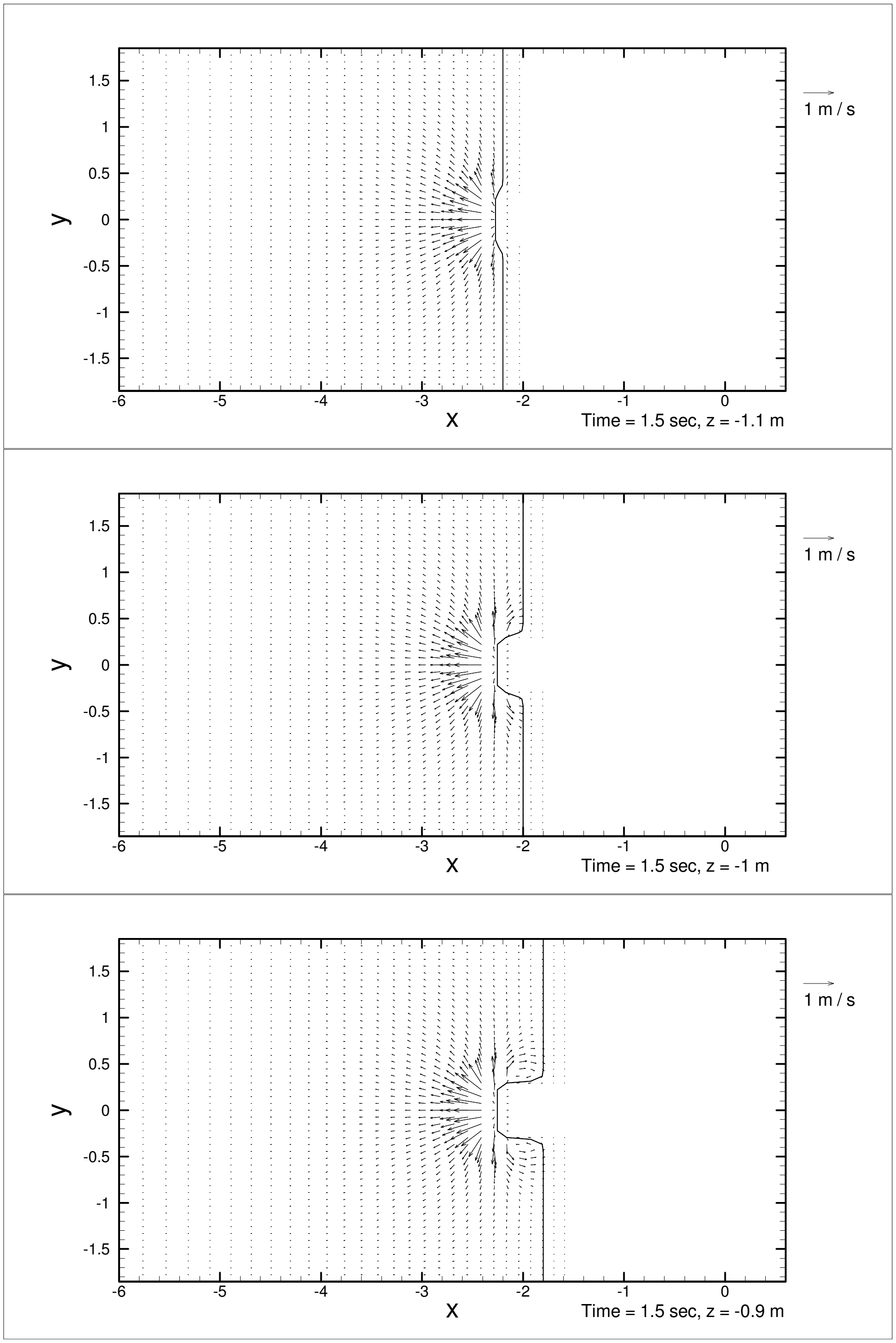


Figure 5.16 (continued)

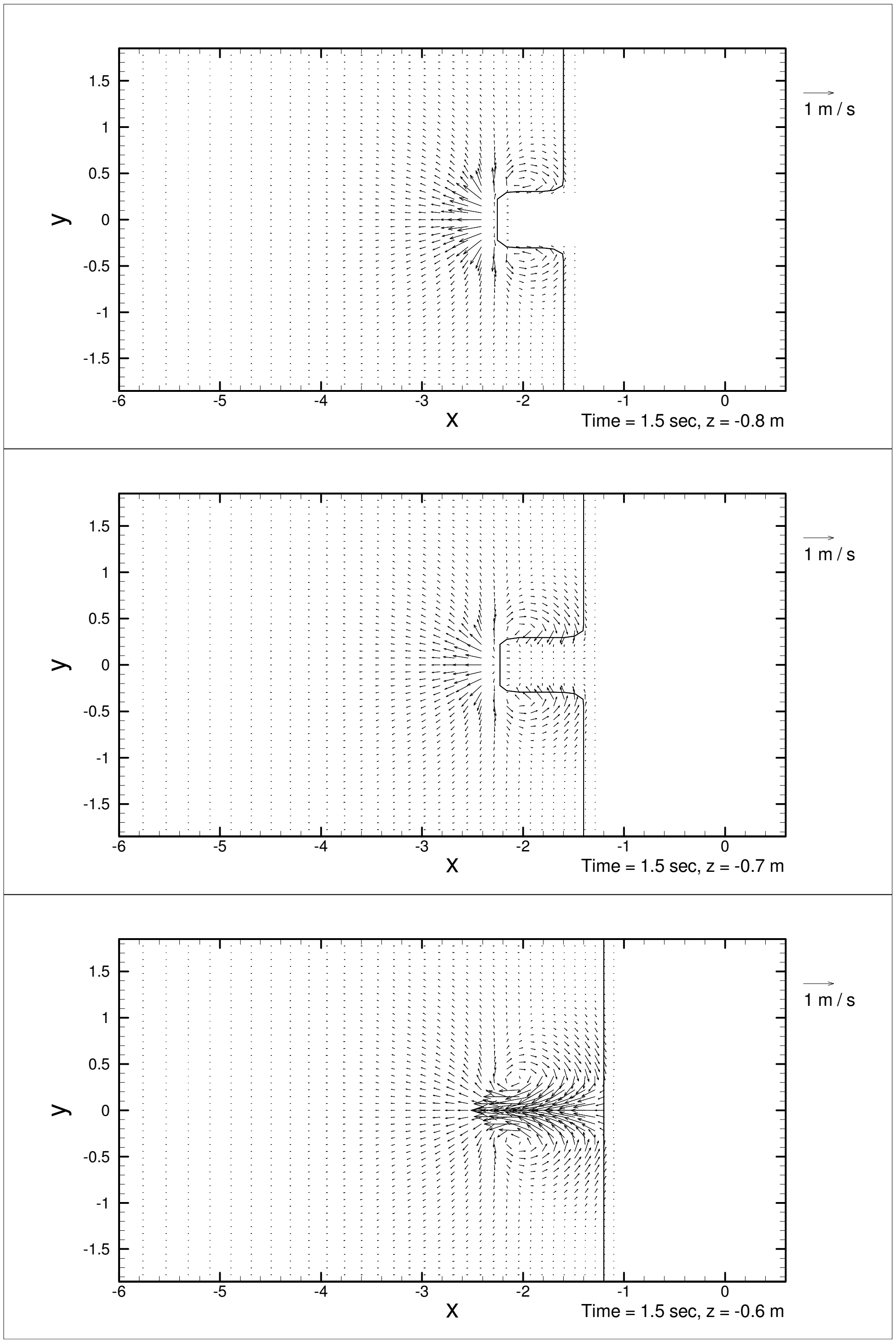


Figure 5.16 (continued)

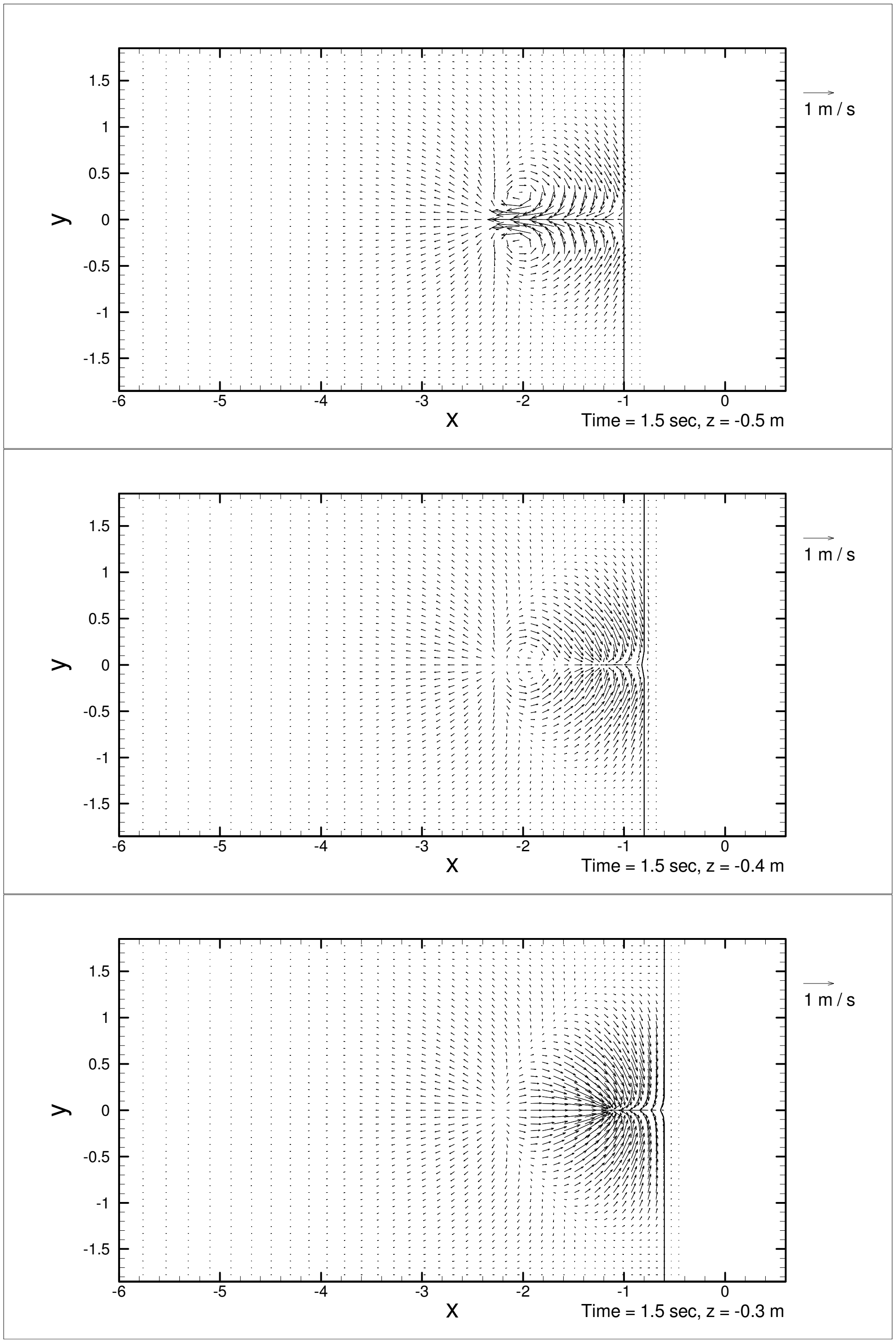
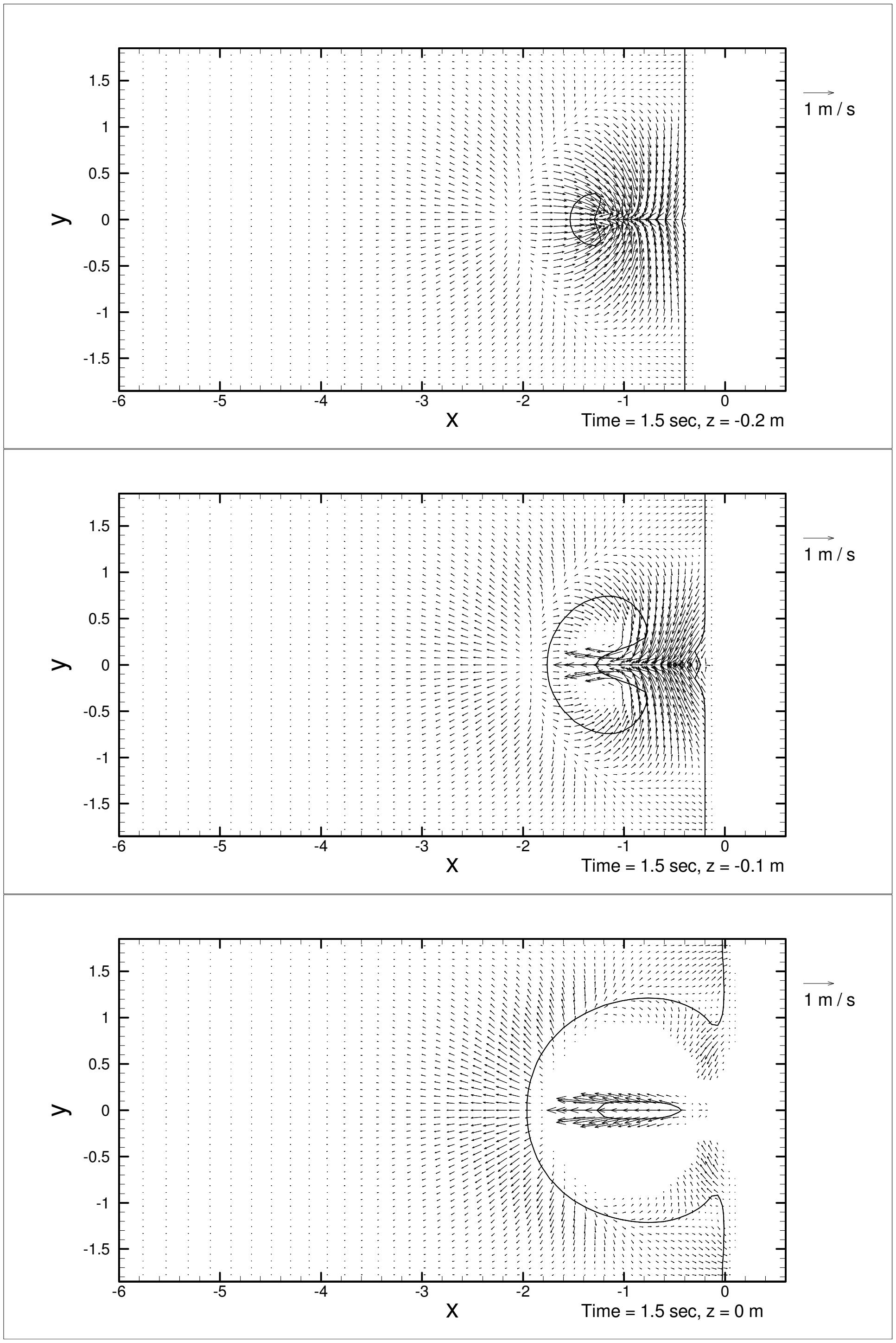
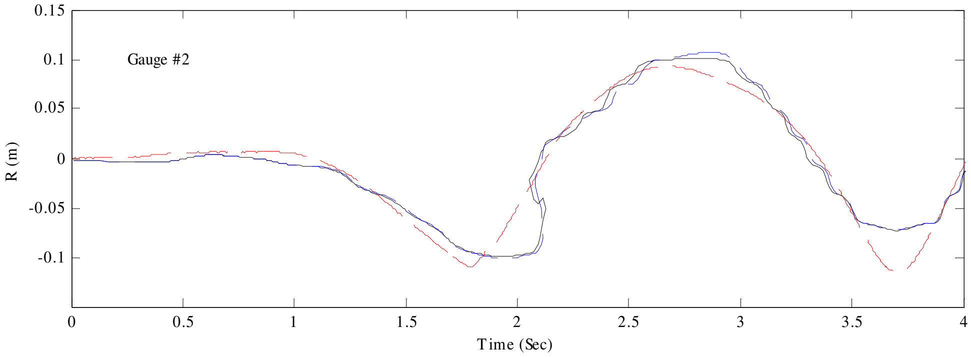


Figure 5.16 (continued)



5.6.8 Half Domain Simulation

From the x, y, and z cross-section figures, we have found that the velocities are nearly symmetric in the spanwise direction. This suggests that the vortex shading effect might not be significant to the whole flow field, especially to the runup and wave heights. In order to confirm this observation, we also have conducted a simulation in the same numerical tank but only with half width. Figure 5.17, 5.18, and 5.19 show the comparisons between full domain simulation, half domain simulation and experimental data in terms of runup heights and free surface elevations. The differences between the full domain and half domain simulation are less than 2% of the maximum runup height which is small and within one grid size. This test indicates that we can conduct the landslide simulation in the half-domain numerical tank and save the computational efforts. Therefore, the simulations will be conducted in half-domain numerical tanks.



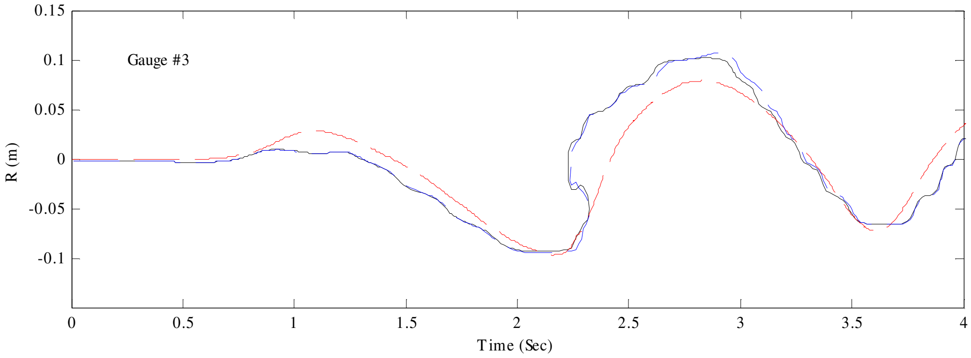


Figure 5.17: The comparison between full domain numerical results (solid lines), half domain numerical results (dashed lines), and experimental data (broken lines) for the time history runup height at Gauge #2 and Gauge #3. $\Delta = 0.454\text{m}$. $\gamma = 3.43$.

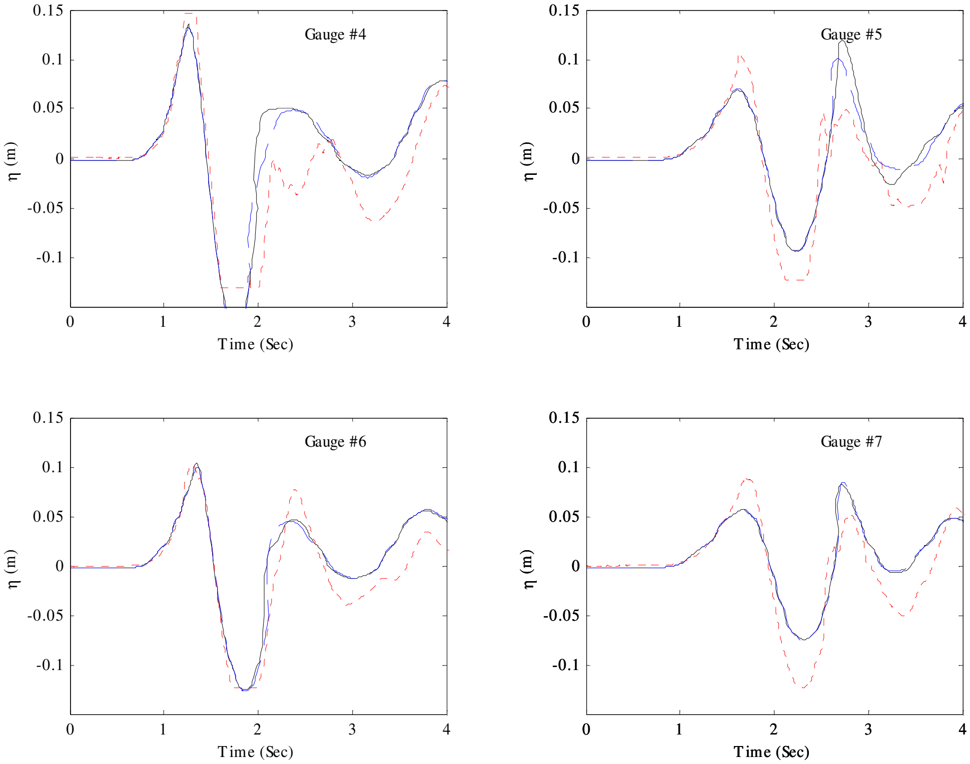


Figure 5.18: The comparison between full domain numerical results (solid lines), half domain numerical results (dashed lines), and experimental data (broken lines) for the time histories of free surface fluctuations at wave gauge #4 ~ #7; Δ=0.454 m, γ=3.43. The coordinates for gauges are: Gauge #4: (x, y) = (1.83, 0); Gauge #5: (x, y) = (2.74, 0); Gauge #6: (x, y) = (1.83, 0.61); Gauge #7: (x, y) = (2.74, 0.61). The unit is in meter.

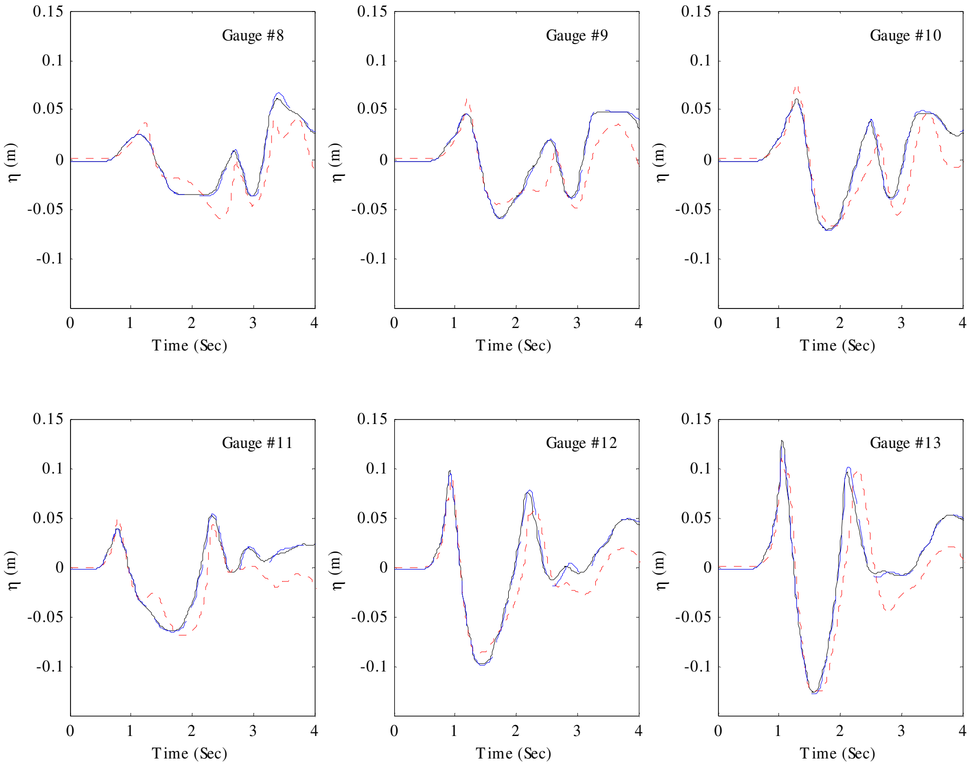


Figure 5.19: The comparison between full domain numerical results (solid lines), half domain numerical results (dashed lines), and experimental data (broken lines) for the time histories of free surface fluctuations at wave gauge #8 ~ #13; Δ=0.454 m, γ=3.43. The coordinates for gauges are: Gauge #8: (x, y) = (0.4826, 1.092); Gauge #9: (x, y) = (0.8636, 1.092); Gauge #10: (x, y) = (1.2446, 1.092); Gauge #11: (x, y) = (0.635, 0.4826); Gauge #12: (x, y) = (0.635, 0.8636); Gauge #13: (x, y) = (0.635, 1.2446). The unit is in meter.

5.7 A Submerged Landslide Simulation

In the previous section, we have verified the accuracy of a 3D subaerial landslide simulation. In this section, a 3D submerged landslide simulation is going to be inspected and discussed. Because there are many similarities between the subaerial and submerged landslide cases, we shall focus the descriptions and discussions on the differences between these two cases.

5.7.1 Setup

The initial elevation of the submerged landslide is 0.05 m below the still water level ($\Delta = -0.05\text{m}$). The specific weight γ is 2.79. The numerical simulation is conducted in a domain with size $(x, y, z)_{\text{Domain size}} = (5.4, 1.85, 2.7)$. In order to save computational efforts, we apply the symmetric boundary condition at $y = 0.0\text{m}$. Non-uniform grids are applied in x and z directions with the finest grid size $(dx, dz)_{finest} = (0.0388,\ 0.0194)$ located at the corner near the initial shoreline. A uniform grid is adopted in the lateral direction. Figure 5.20 shows the displacement (S) and the speed of the moving slide measured from the laboratory experiment.

5.7.2 Initial and Boundary Conditions

The submerged landslide case has the same initial and boundary conditions as those in the subaerial case.

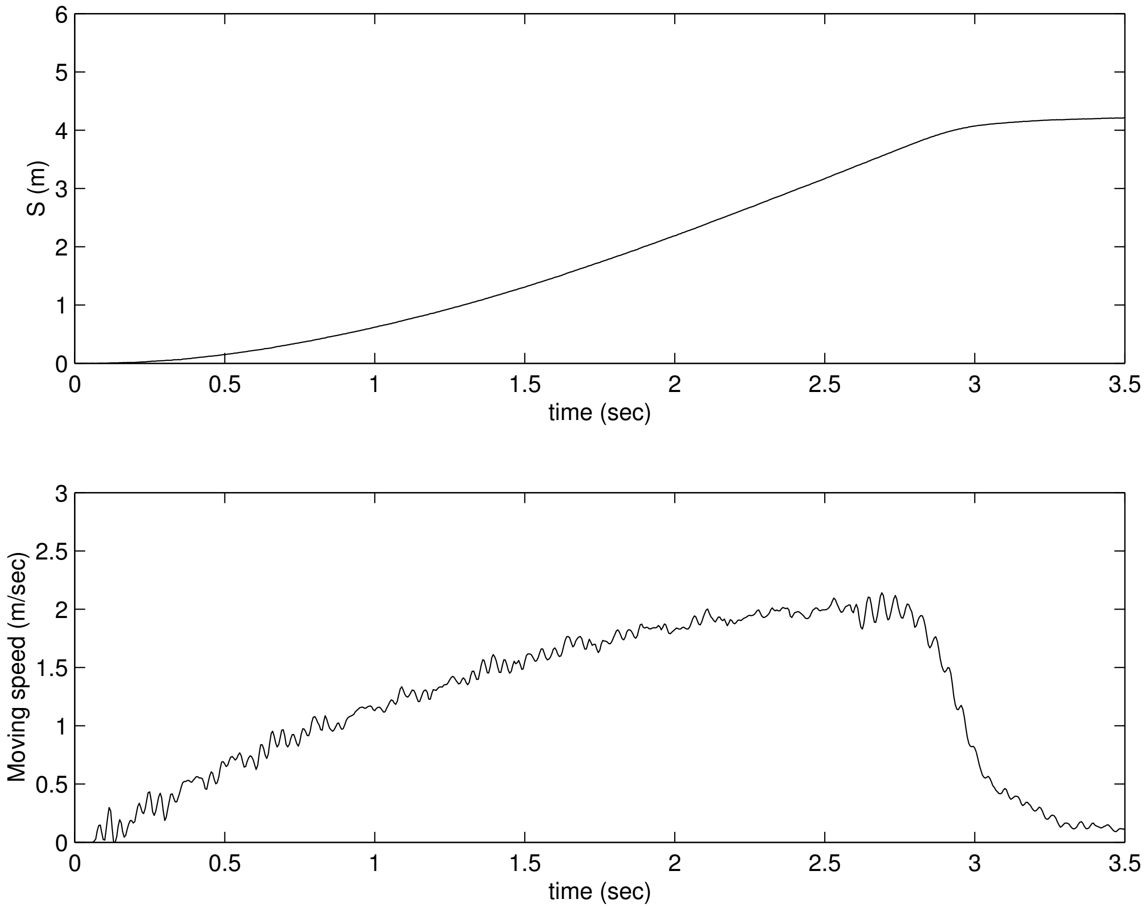


Figure 5.20: The displacement (S) and the speed of the moving slide. In this submerged case, the initial wedge position (Δ) is -0.05 m above the still water surface. The specific weight (γ) of the wedge is 2.79.

5.7.3 The Runup Gauges

Figure 5.21 shows the comparisons of the numerical solutions and laboratory data. From the comparison of Gauge #2, the numerical results appear to be in good agreement with the laboratory data. However, in the Gauge #3 comparison, the lateral movement of the runup in the laboratory experiment is slower than that in the numerical solution. The numerical solution shows that the time-history runup curves of Gauge #2 and Gauge #3 have a similar appearance. This observation can be also obtained in the subaerial simulation and measurement (Figure 5.8). However, in this submerged case, the laboratory measurements of Gauge #2 and Gauge #3 have very different shapes. Unfortunately, we currently do not have a good explanation for this disagreement. Except for the maximum runup, the rest of the data show a good agreement between the numerical solutions and the laboratory data.

5.7.4 Four Fixed Wave Gauges

Figure 5.21 shows the comparisons of wave gauge data. Gauge #4 ~ Gauge #7 are placed in the offshore region. The comparisons show a very good agreement between the numerical solutions and the laboratory data. The numerical model successfully predicts the maximum wave height in Gauge #6 and Gauge #7, and captures the phases of waves in all measurements.

From Gauge #4 to Gauge #7, the maximum wave height does not happen at the leading but at the secondary wave. This observation is different to that in the subaerial case. This shows that in the subaerial landslide, a large portion of energy is transferred from slide to water by the initial “pushing” process. However, in the submerged landslide case, the momentum generated by the initial pushing process will be transferred to the wake area, and has fewer effects on the free-surface waves.

5.7.5 Wave Gauge Array

Figure 5.23 shows the comparison of Gauge #8 ~ Gauge #13. These six wave gauges mainly monitor the energy transfer in the lateral direction. The comparisons show that before time = 3 sec, the numerical solution has a good agreement with the laboratory data. After time = 3 sec, the solution and data are contaminated by the reflected wave from the side walls, and will be excluded from current discussion.

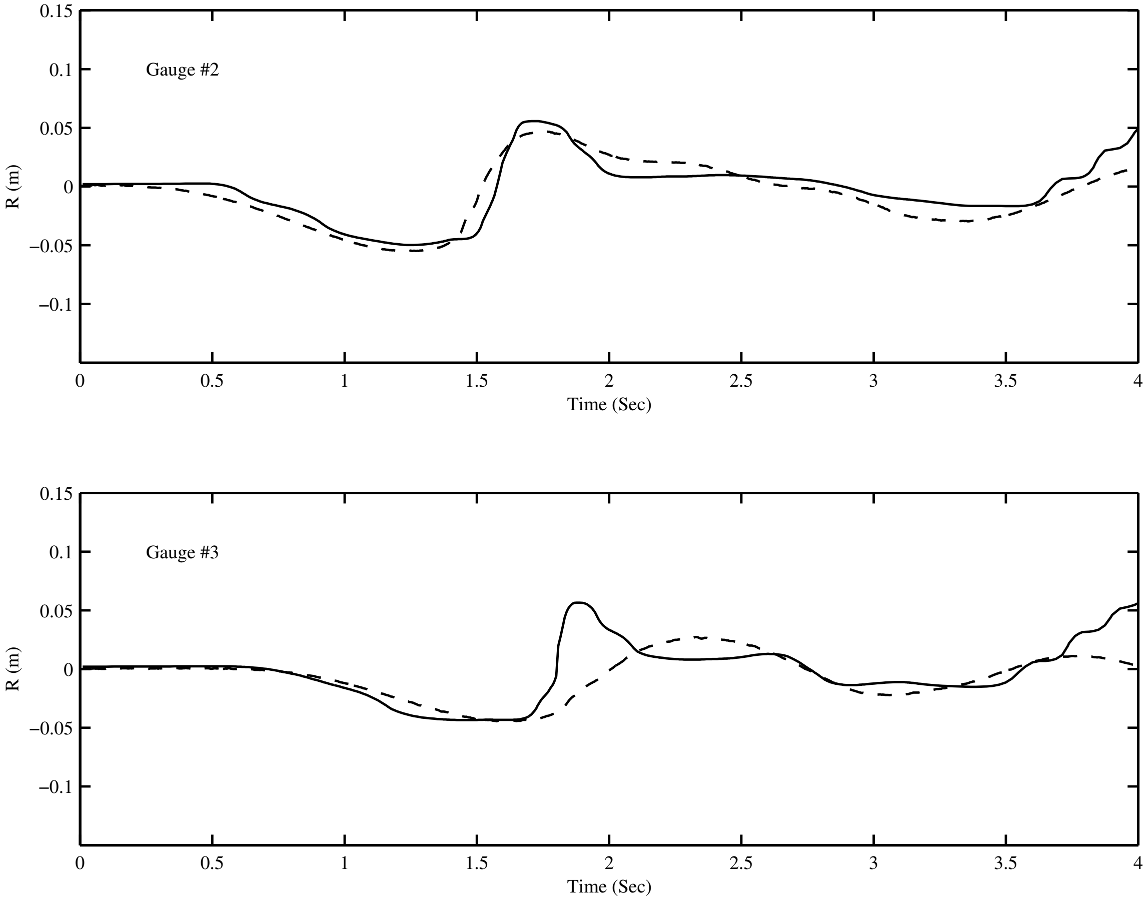


Figure 5.21: The comparison between numerical results (solid lines) and experimental data (broken lines) for the time history runup height at Gauge #2 and Gauge #3. $\Delta = -0.05\text{m}$. $\gamma = 2.79$.

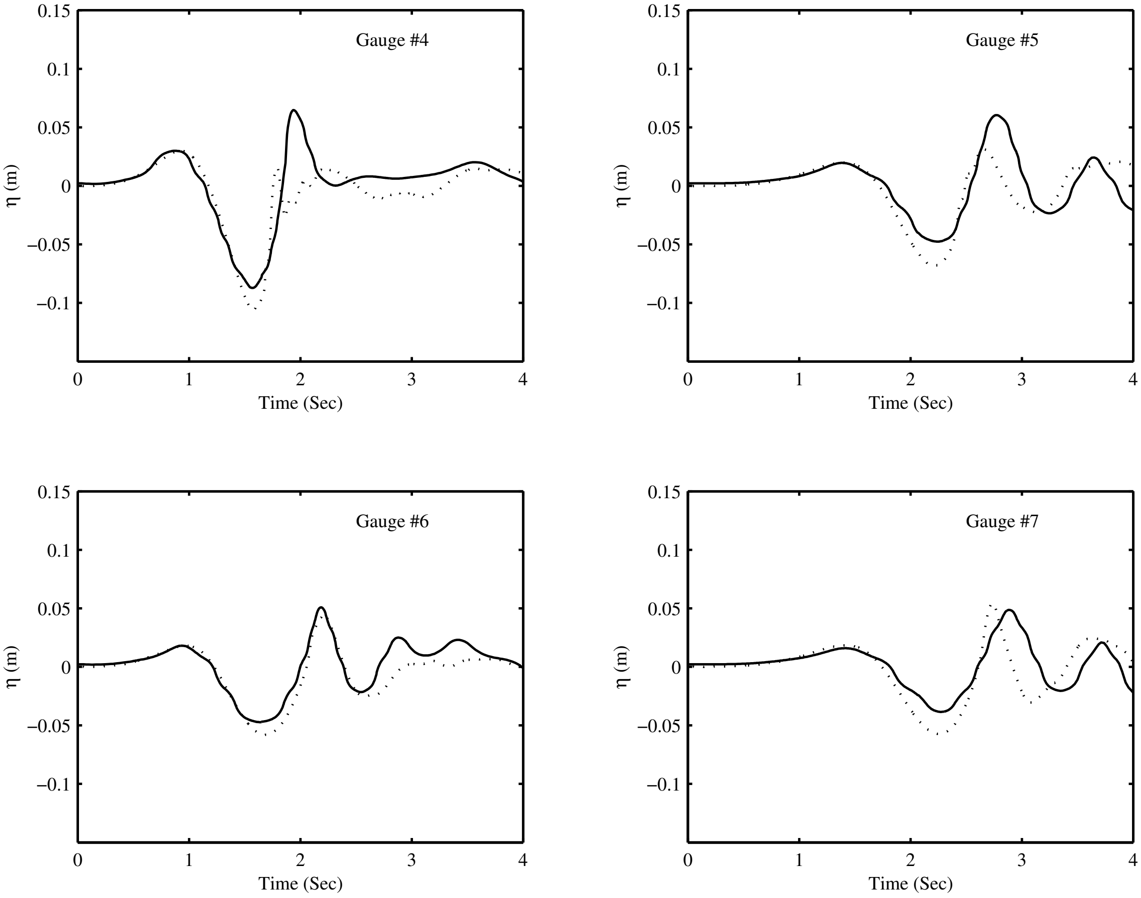


Figure 5.22: The comparison between numerical results (solid lines) and experimental data (broken lines) for the time histories of free surface fluctuations at wave gauge #4 ~ #7; Δ=0.454 m, γ=3.43. The coordinates for gauges are: Gauge #4: (x, y) = (1.83, 0); Gauge #5: (x, y) = (2.74, 0); Gauge #6: (x, y) = (1.83, 0.61); Gauge #7: (x, y) = (2.74, 0.61). The unit is in meter.

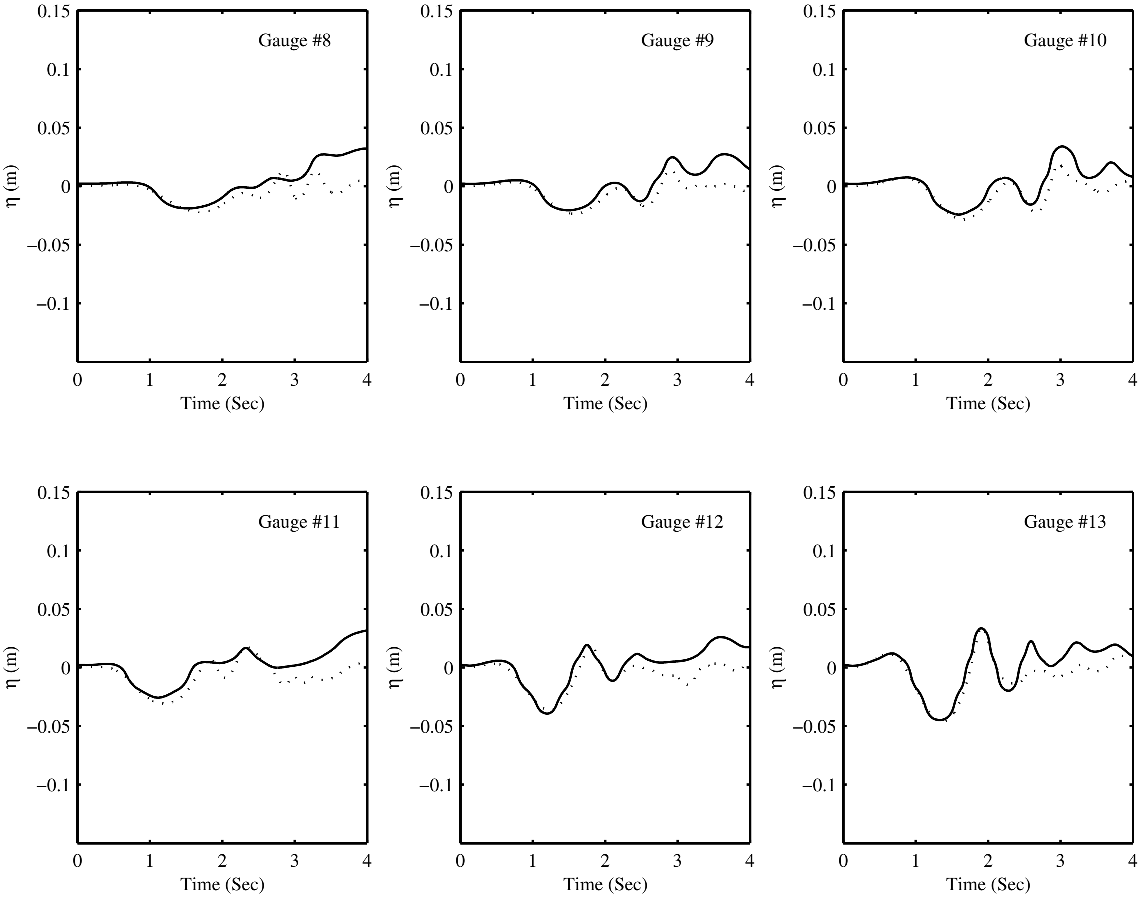


Figure 5.23: The comparison between numerical results (solid lines) and experimental data (broken lines) for the time histories of free surface fluctuations at wave gauge #8 ~ #13; Δ=-0.05 m, γ=2.79. The coordinates for gauges are: Gauge #8: (x, y) = (0.4826, 1.092); Gauge #9: (x, y) = (0.8636, 1.092); Gauge #10: (x, y) = (1.2446, 1.092); Gauge #11: (x, y) = (0.635, 0.4826); Gauge #12: (x, y) = (0.635, 0.8636); Gauge #13: (x, y) = (0.635, 1.2446). The unit is in meter.

5.7.6 Snapshots

We present the snapshots of free-surface elevation, velocities, and shoreline movements from Figure5.24 ~ Figure 5.26. The basic physical phenomena are very similar to those in the subaerial cases. In Figure 5.24, we can see that the free surface has smaller free-surface displacement than that in the subaerial one. This is because that the submerged landslide has a lower initial slide elevation and a smaller specific weight.

Figure 5.25 show the velocities at the centerline cross-section. Compared to the subaerial case, the main processes to generate waves and velocities are similar. One important observation is that the velocity on top of the slide has a speed faster than that in the moving slide. This is caused by the three-dimensional effect. We have found that the moving slide generates a three-dimensional eddy, and this eddy transfers the momentum to the centerline location and makes the high velocity jet on top of the slide. This phenomenon can be clearly seen in Figure 5.27.

Figure 5.28 shows the snapshots of the shorelines. Because of the slide movement, from time = 0.0 sec to 0.9 sec, the runup height has negative values. After time = 1.2 sec, a rebounding wave causes the shoreline to rise up. The maximum runup at the centerline happens at time = 1.75 sec. Compared to the subaerial case (Figure 5.15, time = 2.7 sec), different shoreline curves can be observed. In the subaerial simulation, the shoreline curve looks like a hump. However, in this subaerial case, the shoreline curve is more like a concave.

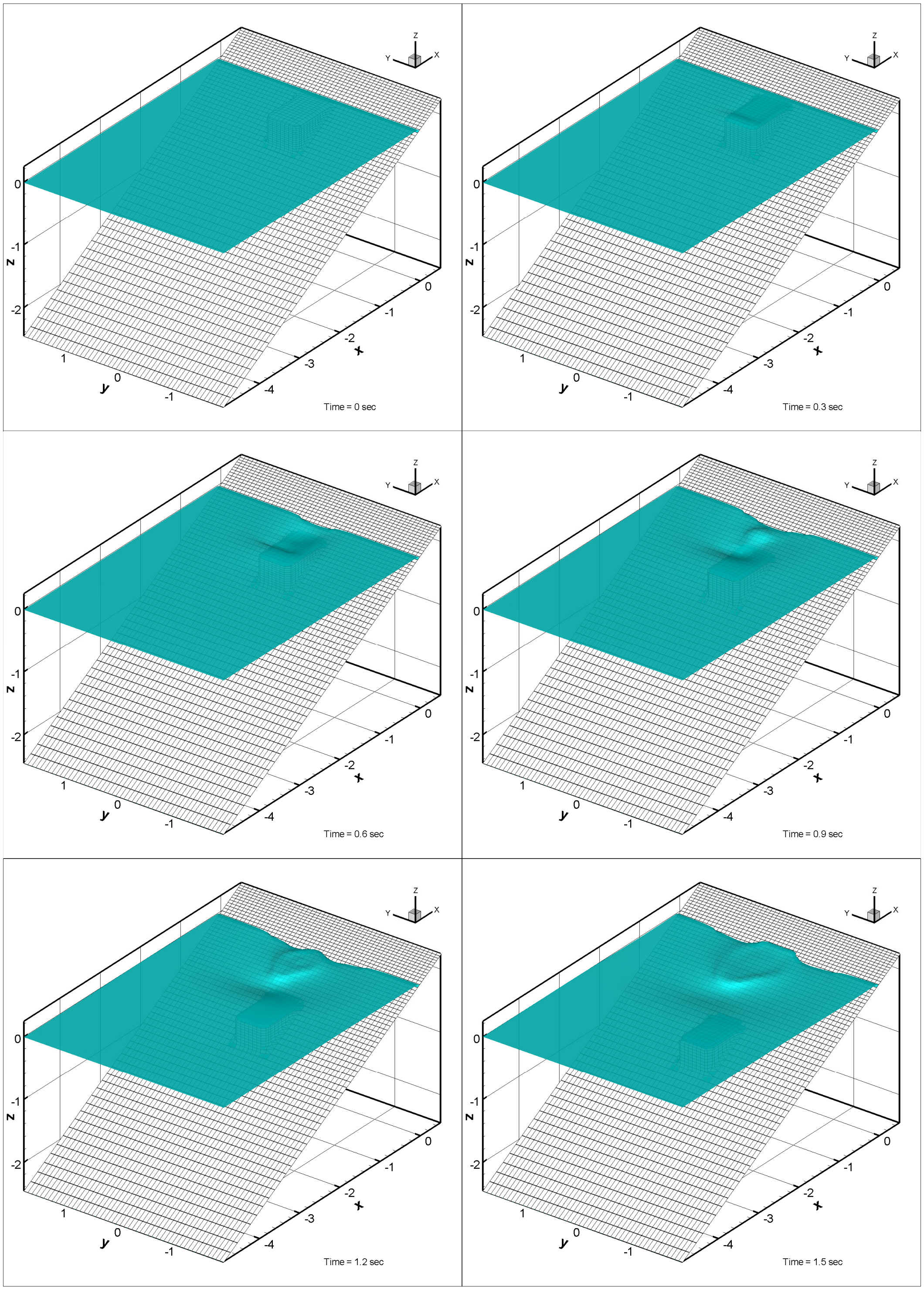
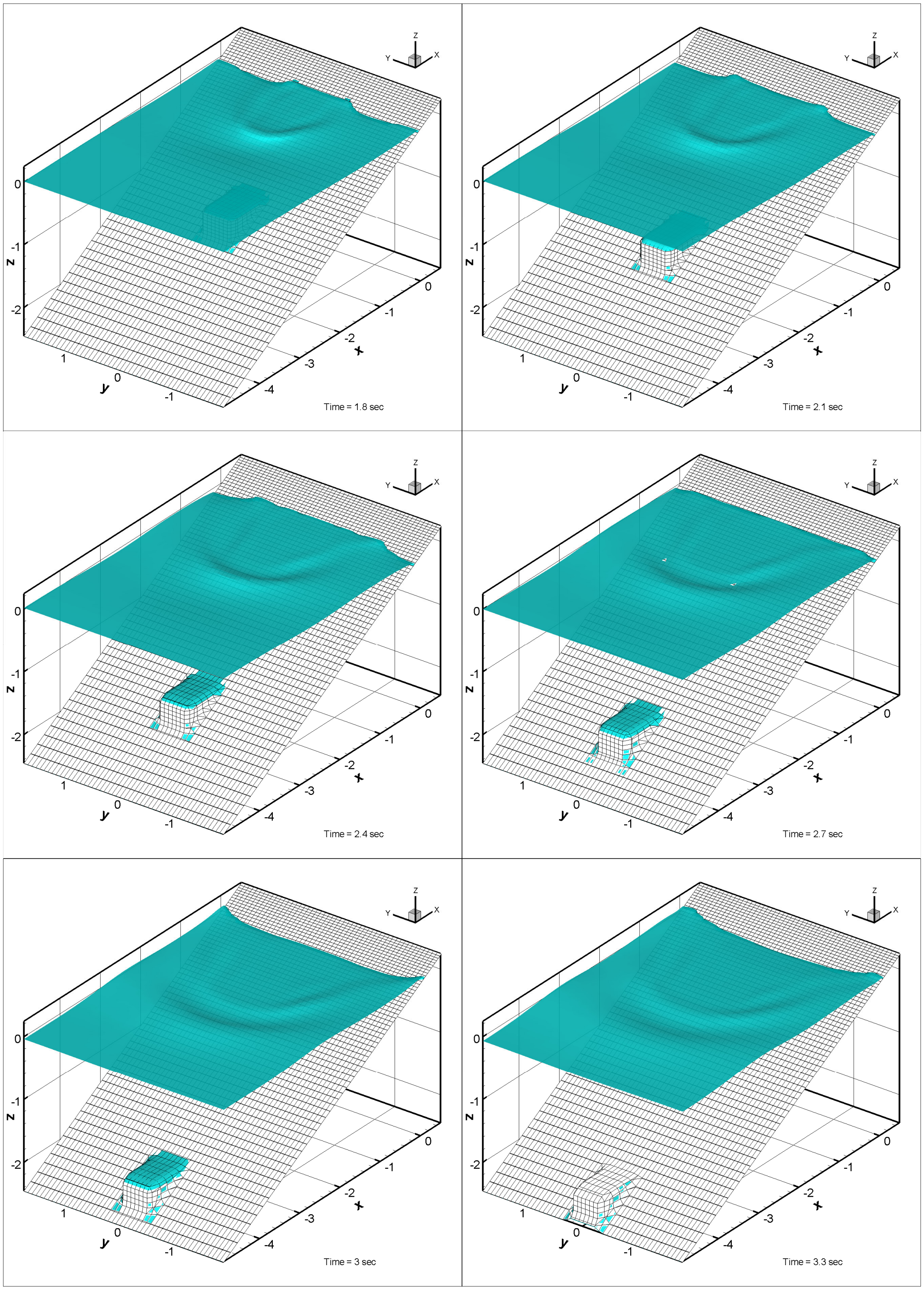


Figure 5.24: Snapshots of free-surface profile for the sliding wedge with Δ= -0.05 m, and γ =2.79. The unit is in meter.

Figure 5.24 (continued)



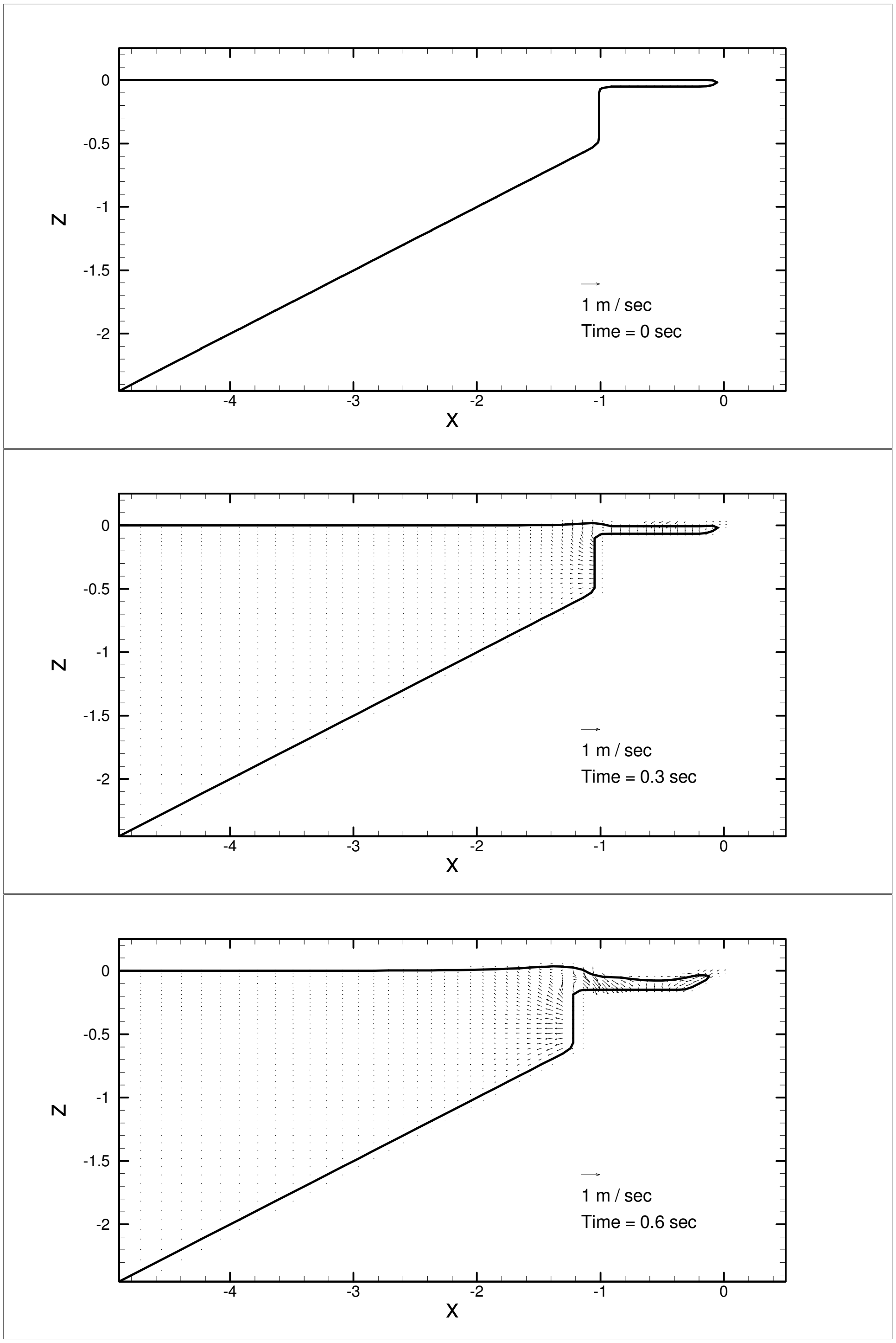


Figure 5.25: Snapshots of velocity vectors on the centerline vertical plane for the sliding wedge with Δ=-0.05 m, and γ=2.79. The unit is in meter.

Figure 5.25 (continued)

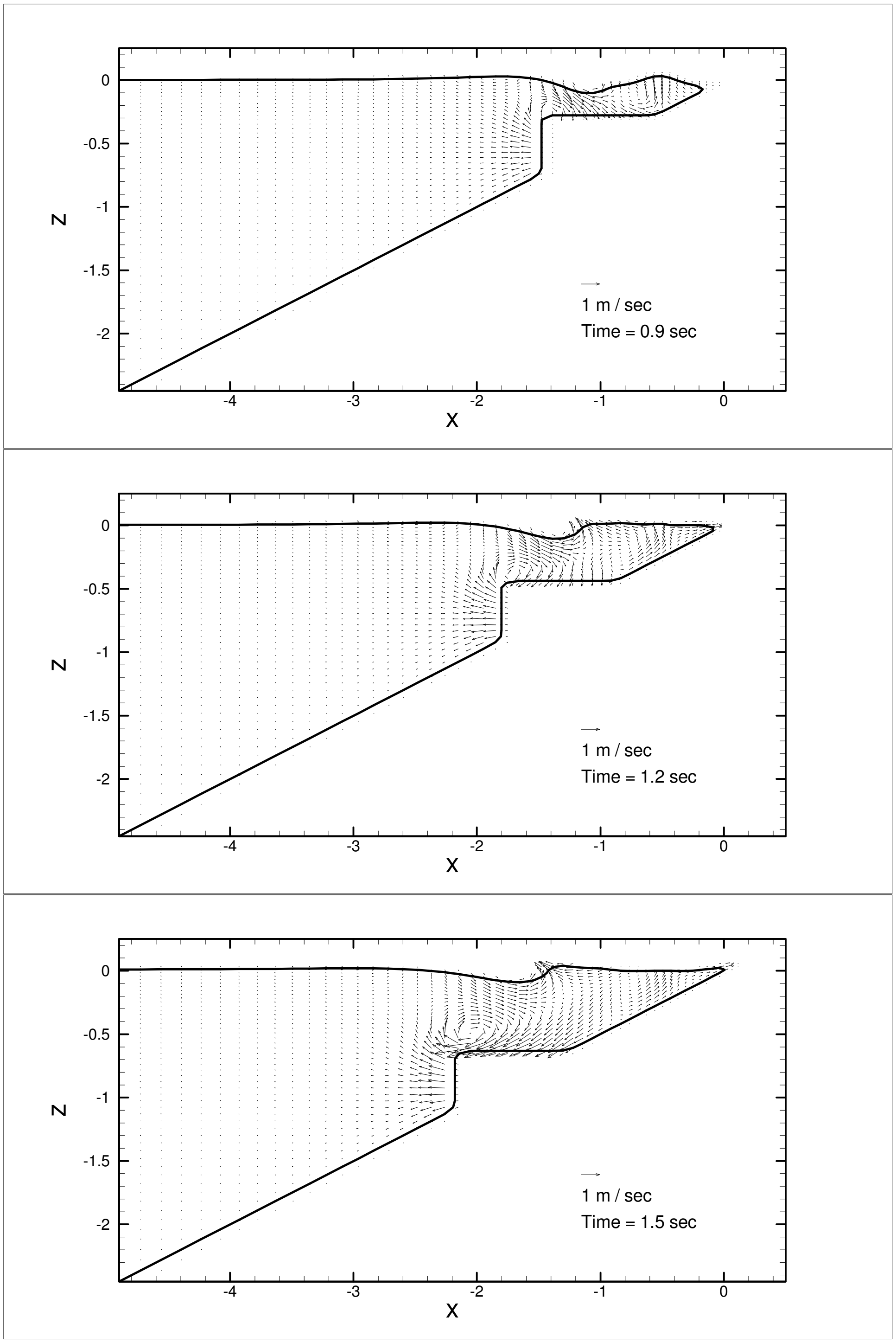
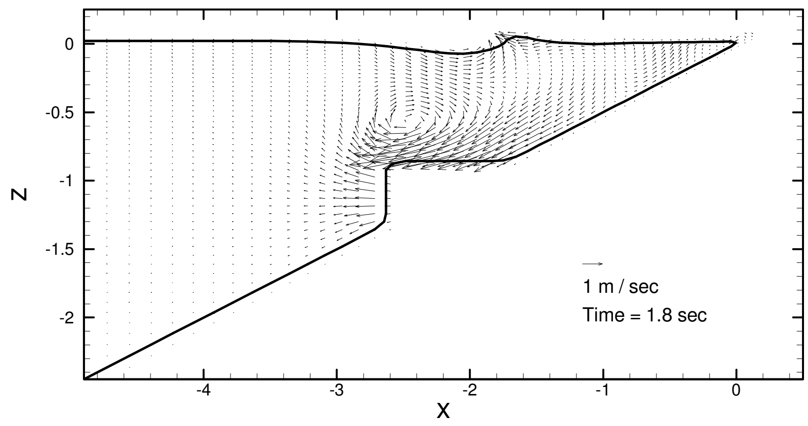
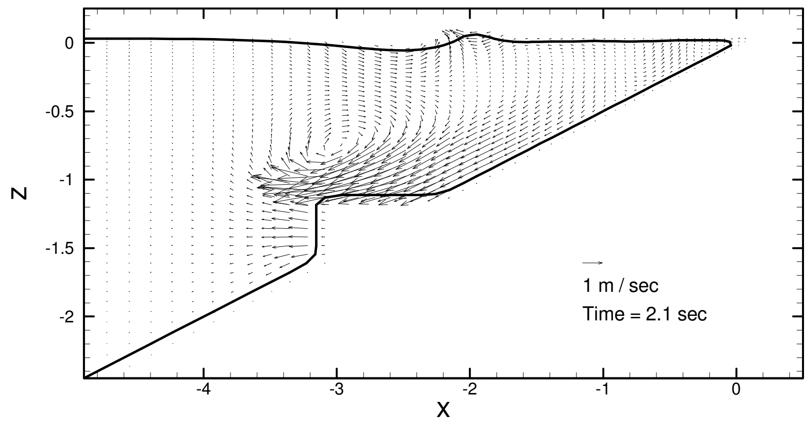


Figure 5.25 (continued)





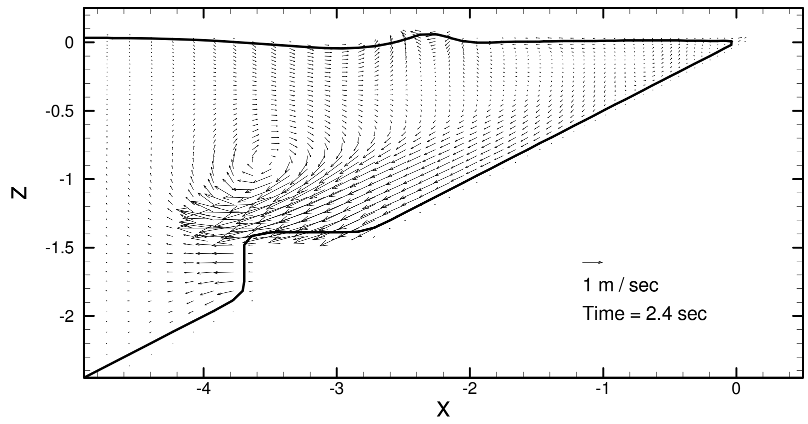
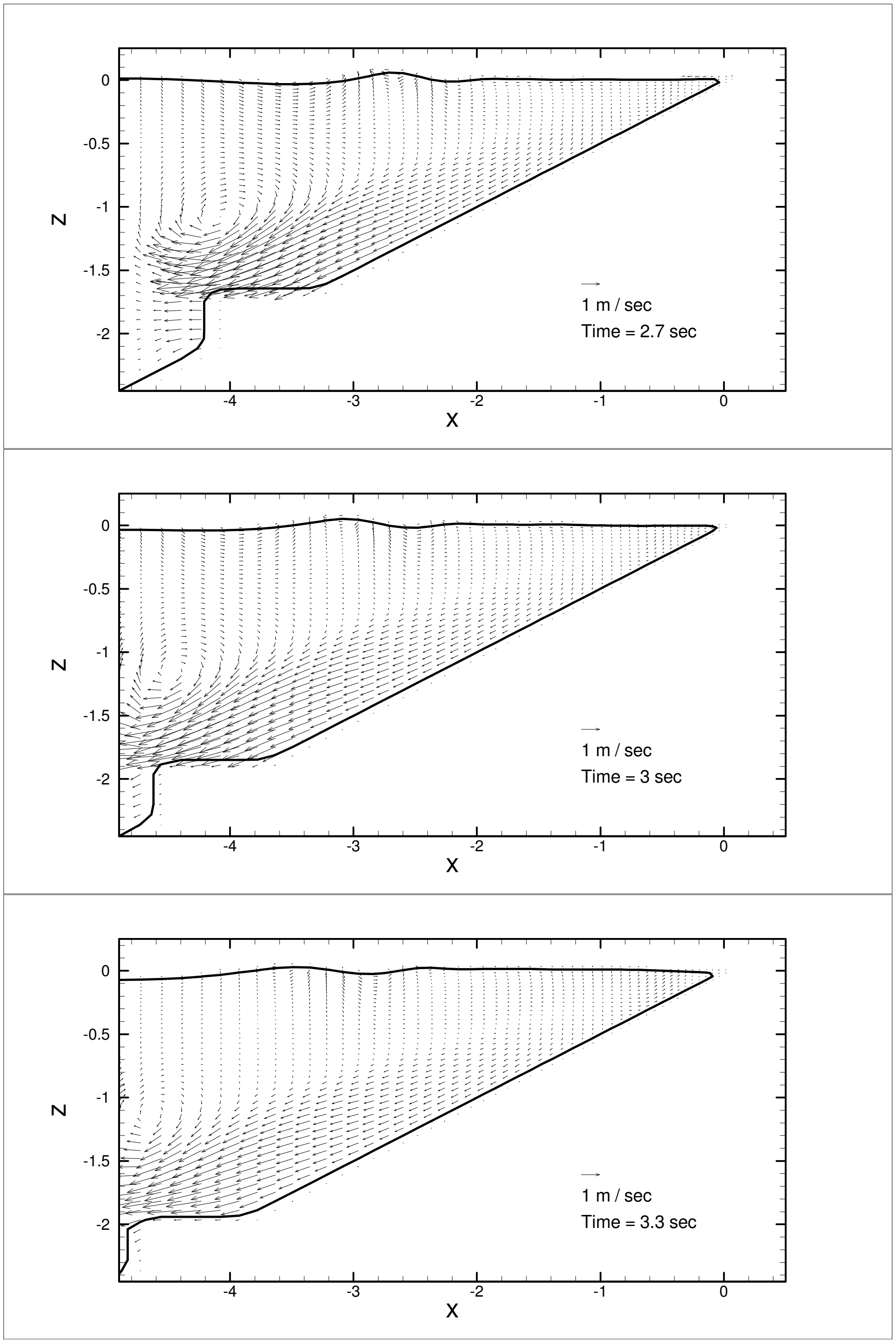


Figure 5.25 (continued)



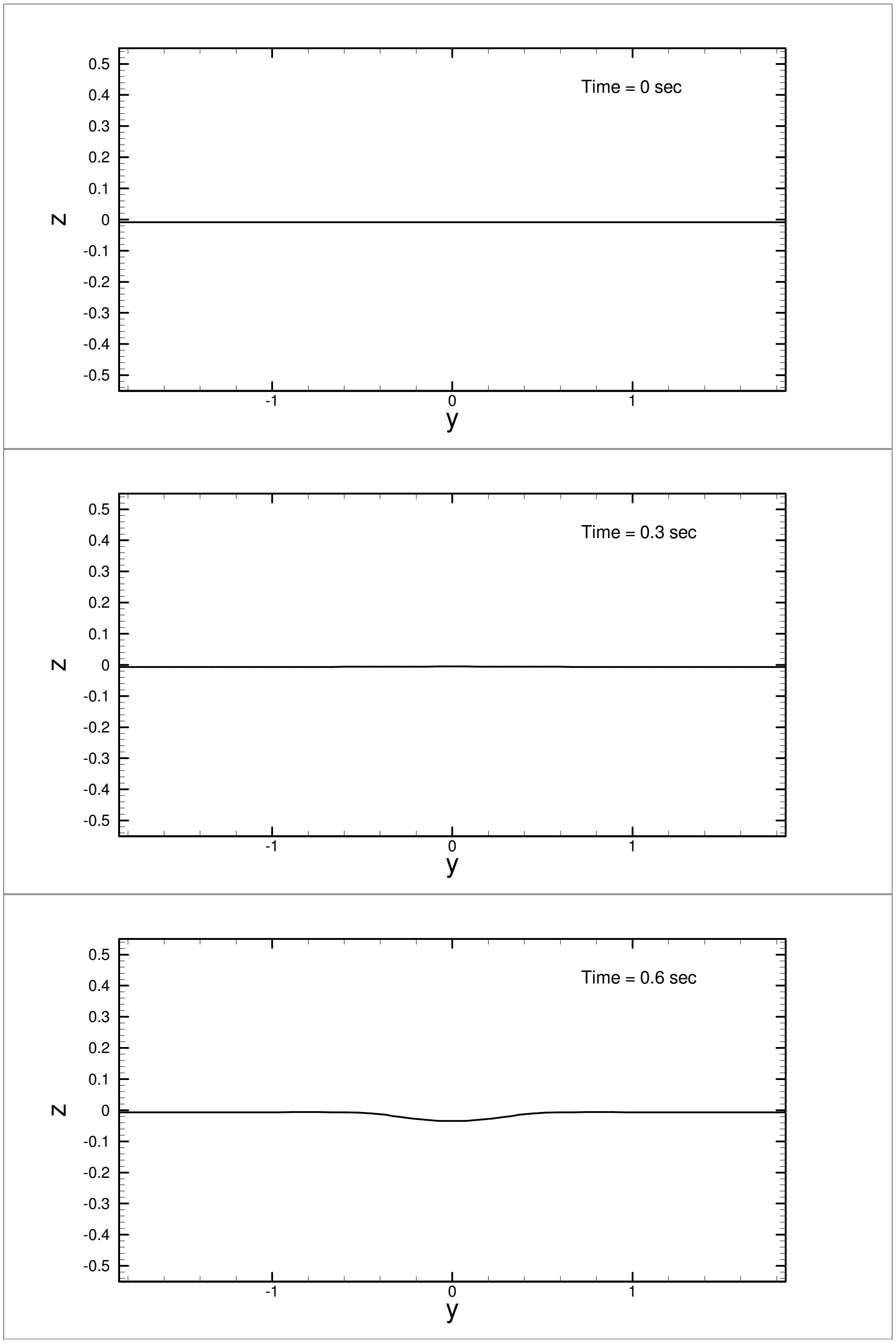


Figure 5.26: Snapshots of shoreline movement for the sliding wedge with Δ=0.454 m, and γ=3.43. The unit is in meter.

Figure 5.26 (continued)

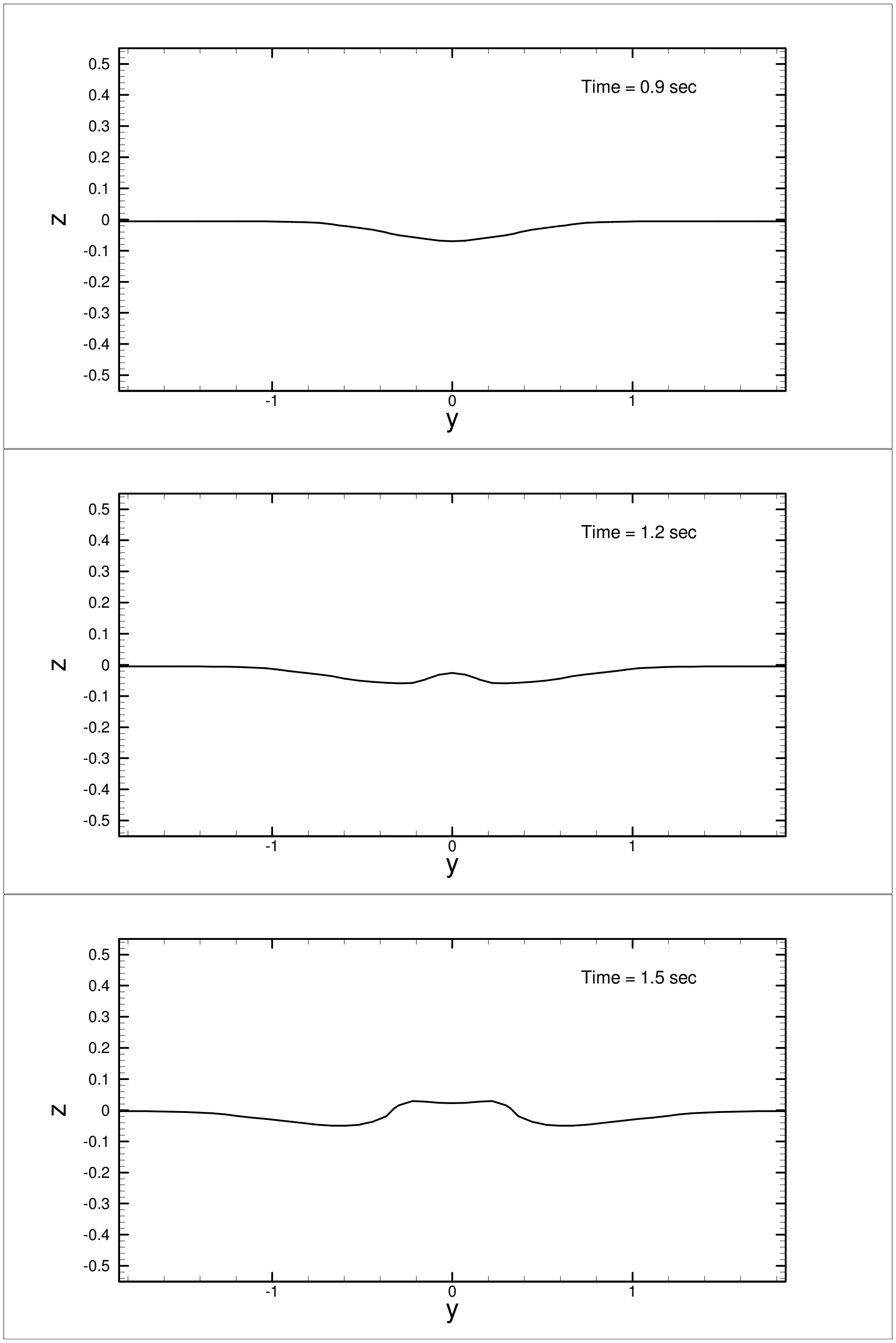


Figure 5.26 (continued)

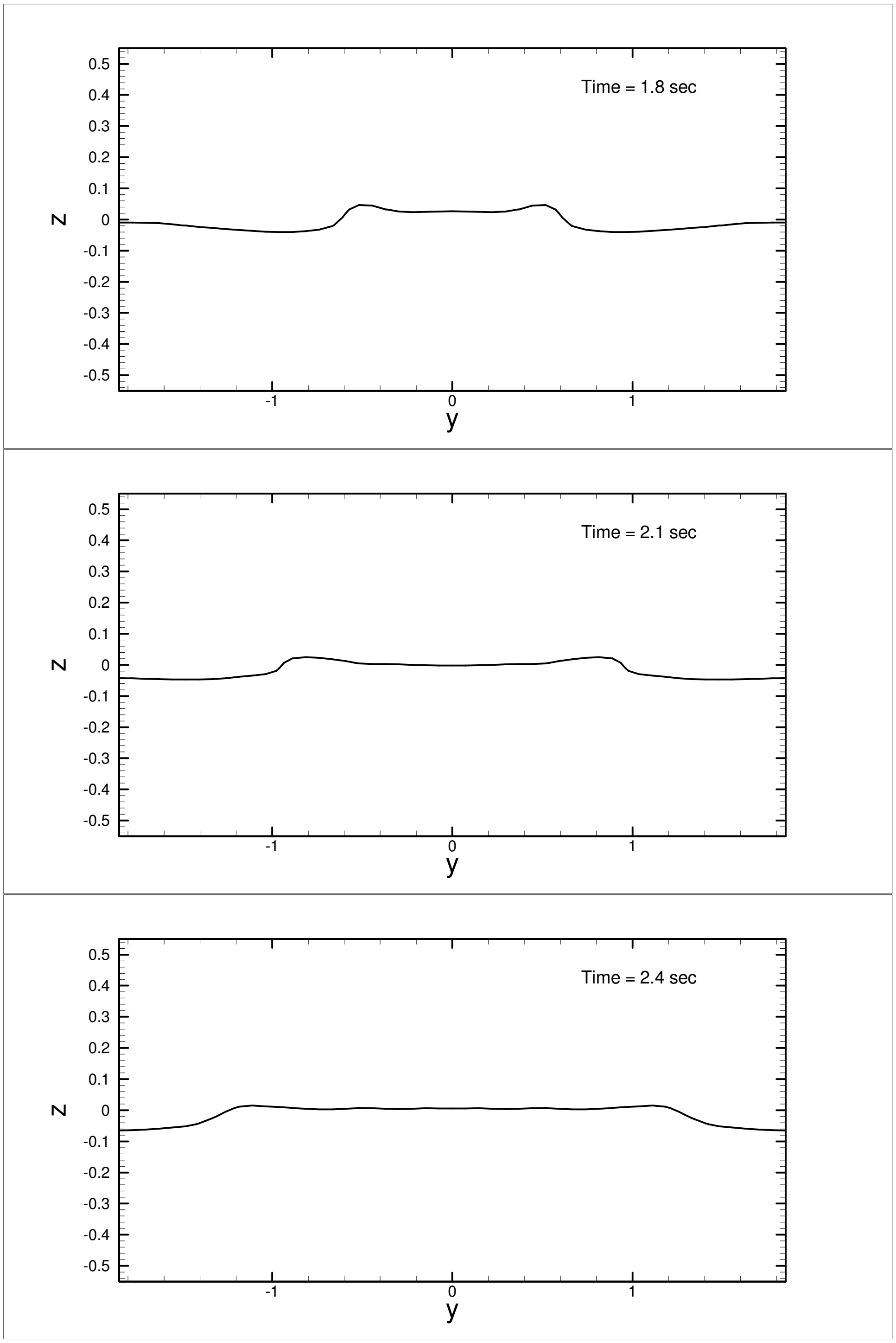
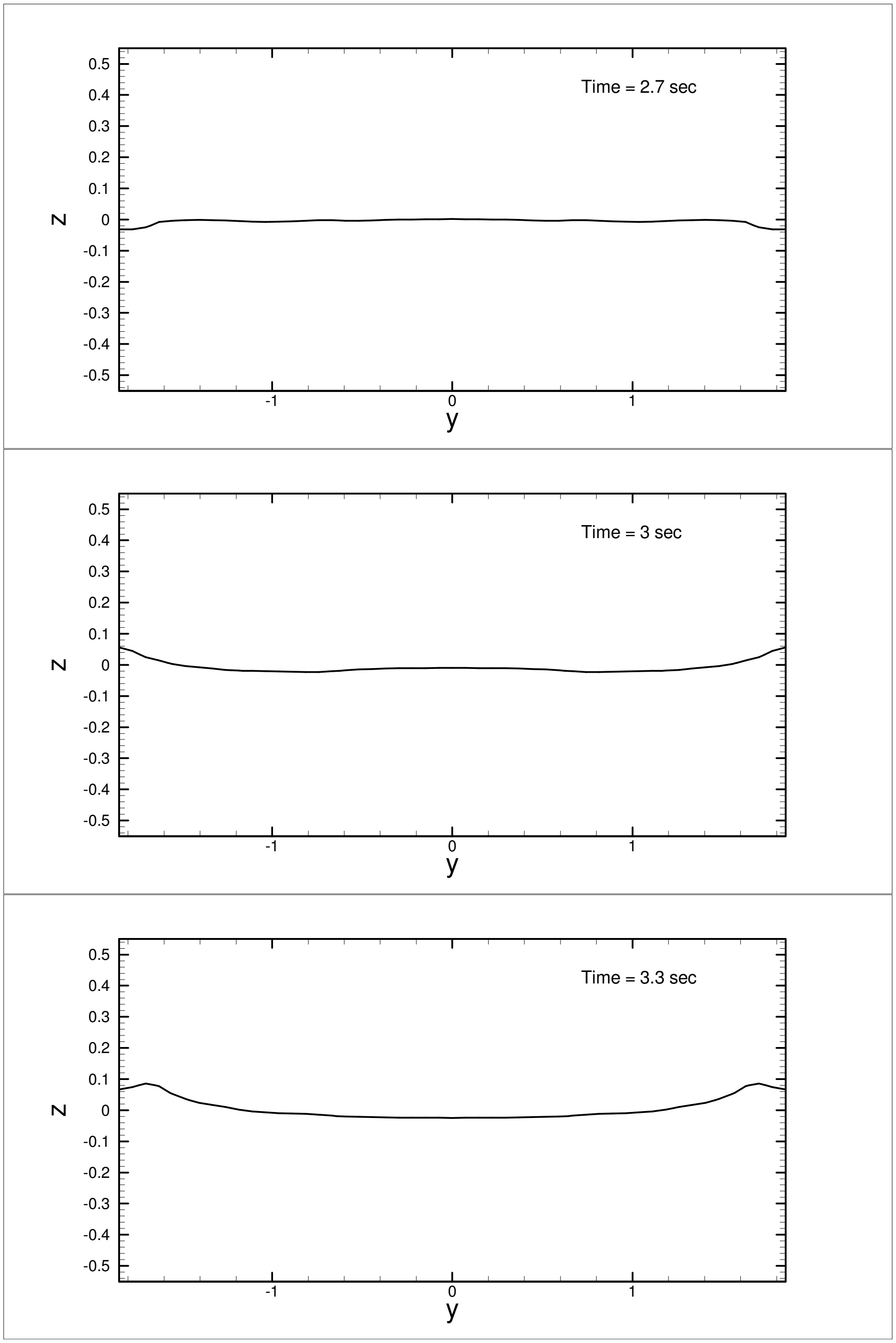


Figure 5.26 (continued)



5.7.7 Cross-Sections

In this section, the detailed velocity fields (time = 1.5 sec) at different cross-sections will be presented from Figure 5.27 to Figure 5.29. Because the major observations are close to that of the subaerial case, only the differences and important phenomena will be addressed.

From Figure 5.27, x = -2.0 m ~ -1.25 m, the three-dimensional effect which makes the momentum concentrate into the centerline can be observed. The velocities on the centerline cross-section are higher than those in the off-centroid cross-section (Figure 5.28, y = 0.0 m and y = 0.2 m). The three-dimensional movement can be also observed from Figure 5.29, at z = -0.7 m and z = -0.6 m.

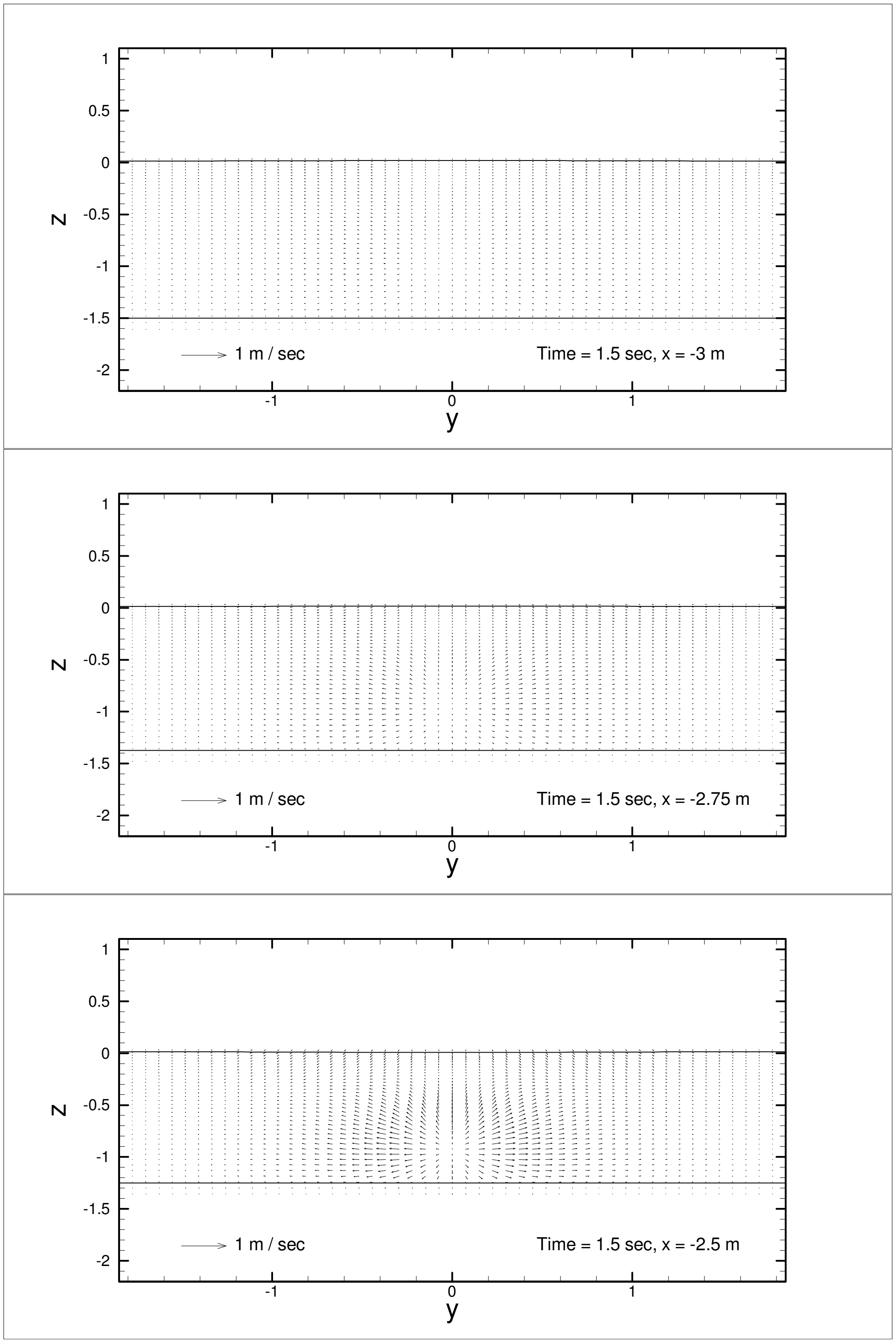


Figure 5.27: Snapshots of velocity vectors on the vertical x-planes at time = 1.5 sec for the sliding wedge with Δ=-0.05 m, and γ=2.79. The unit is in meter. The magnitude of the reference vector indicates the speed of the wedge.

Figure 5.27 (continued)

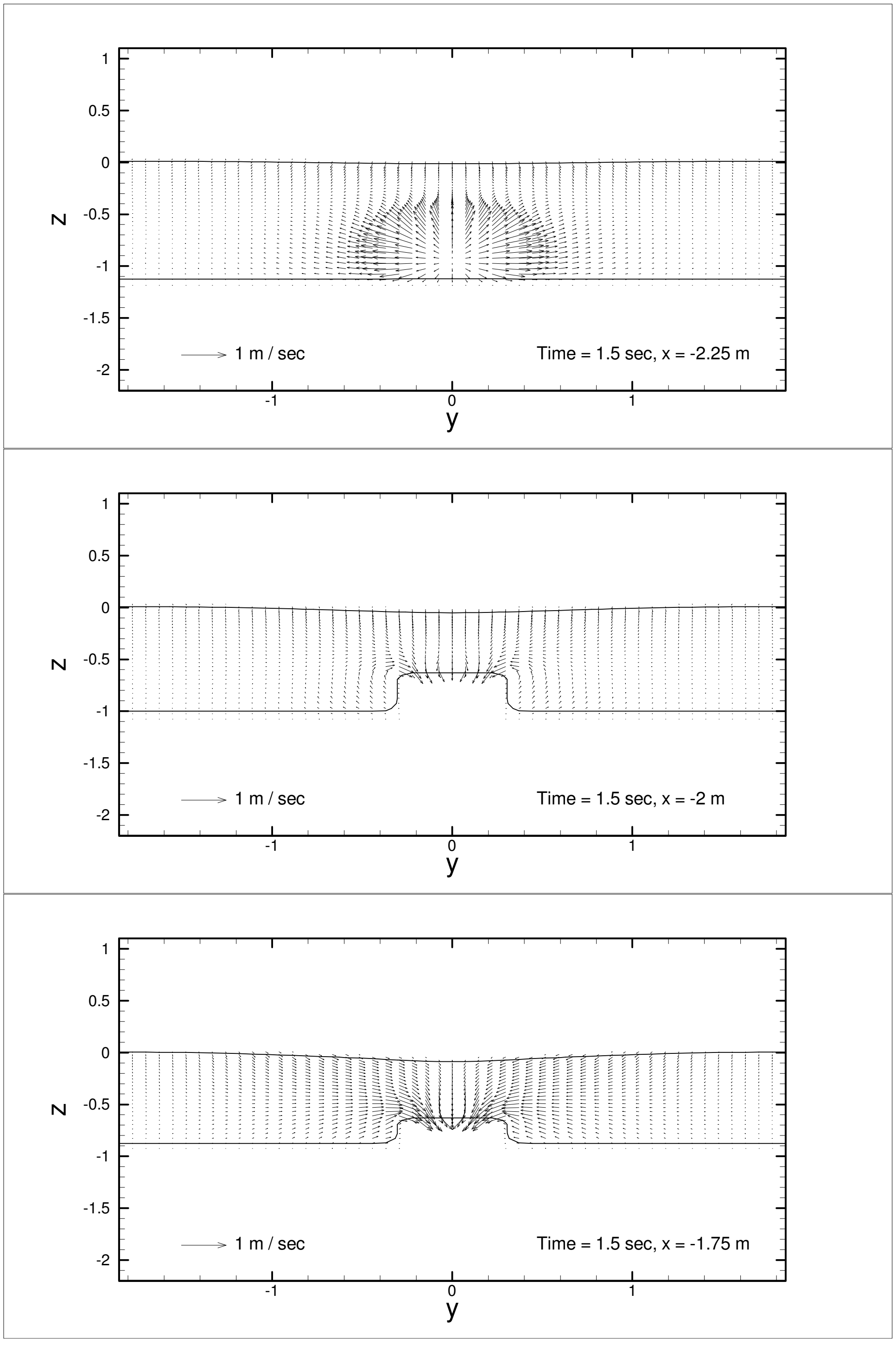


Figure 5.27 (continued)

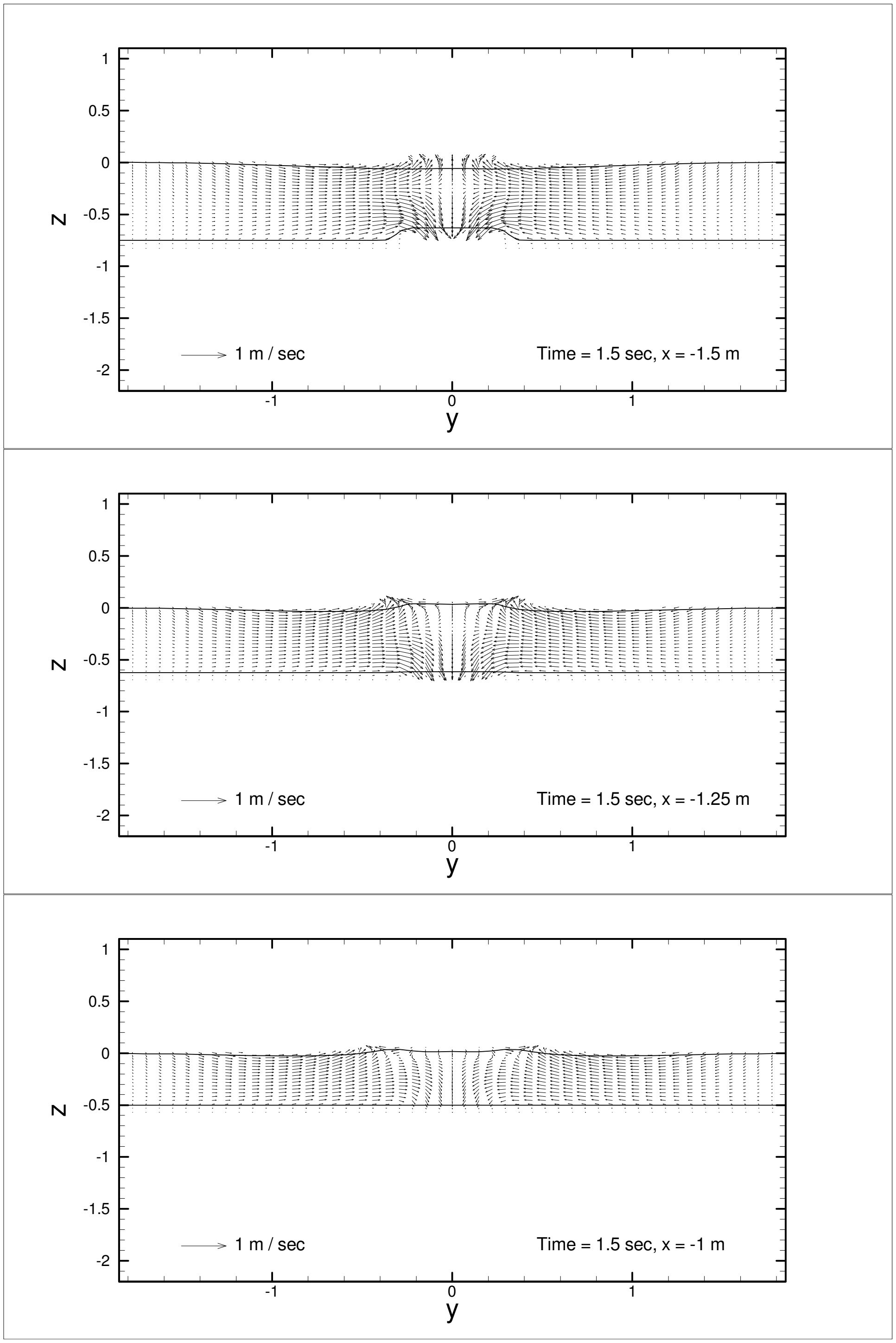
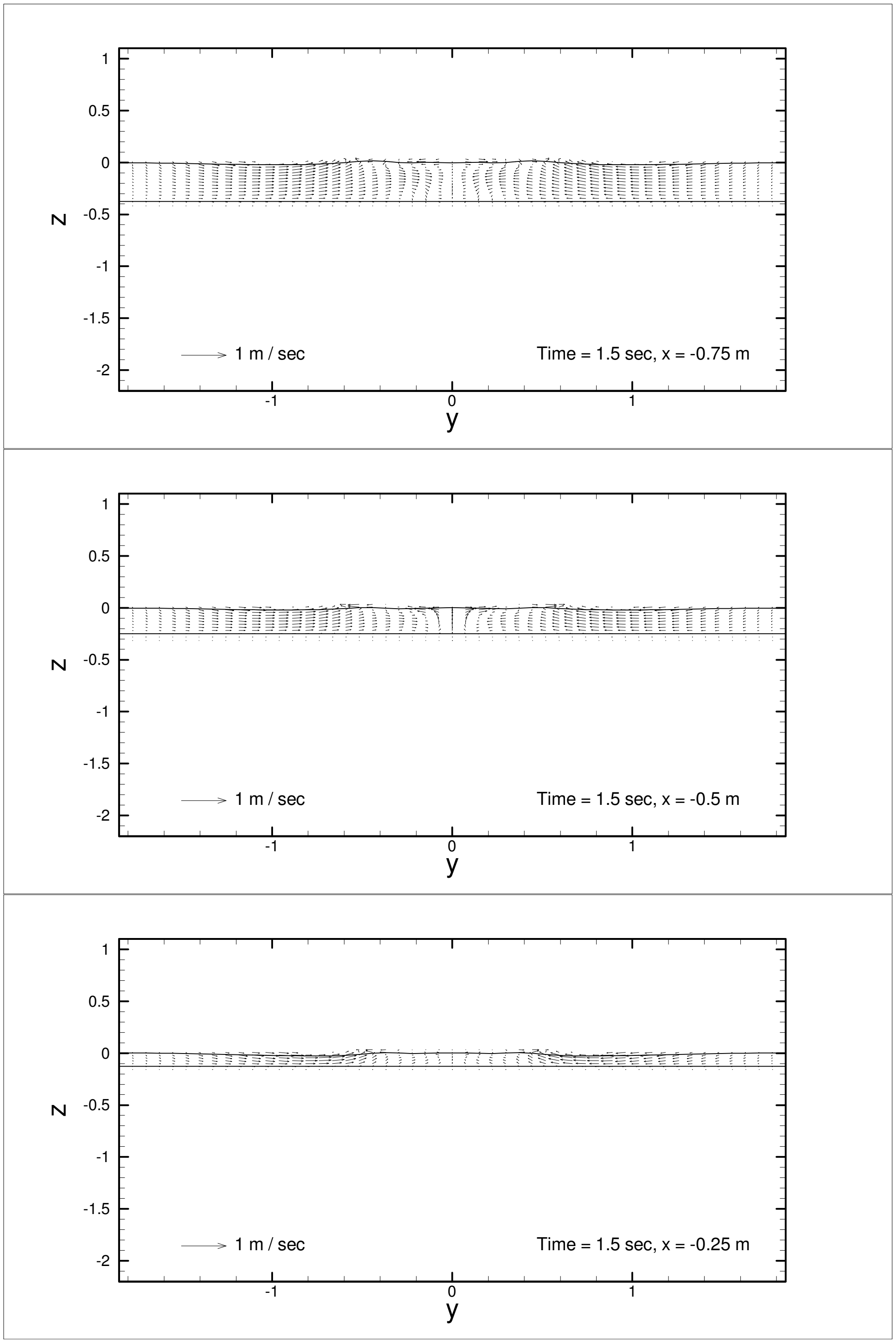


Figure 5.27 (continued)



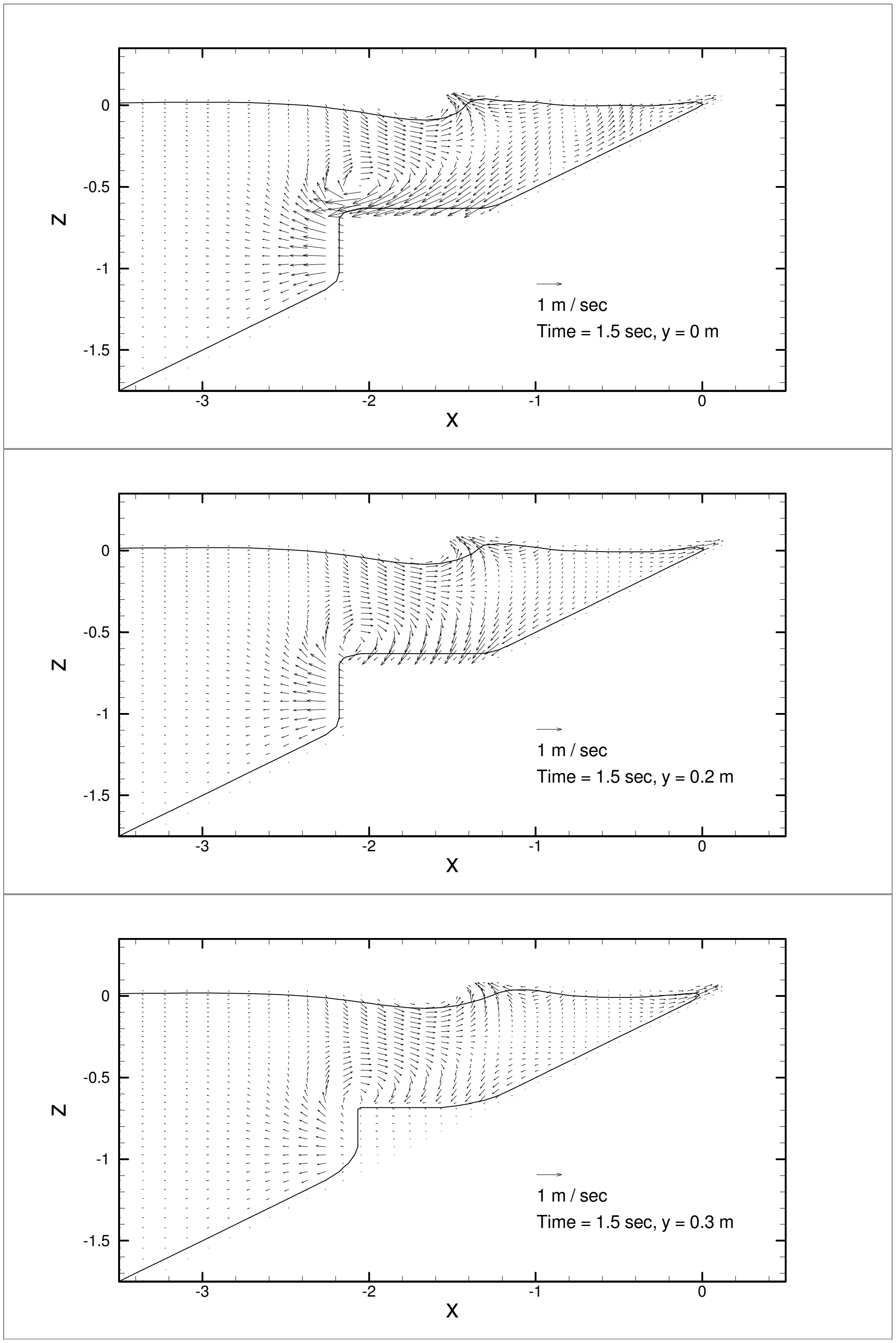
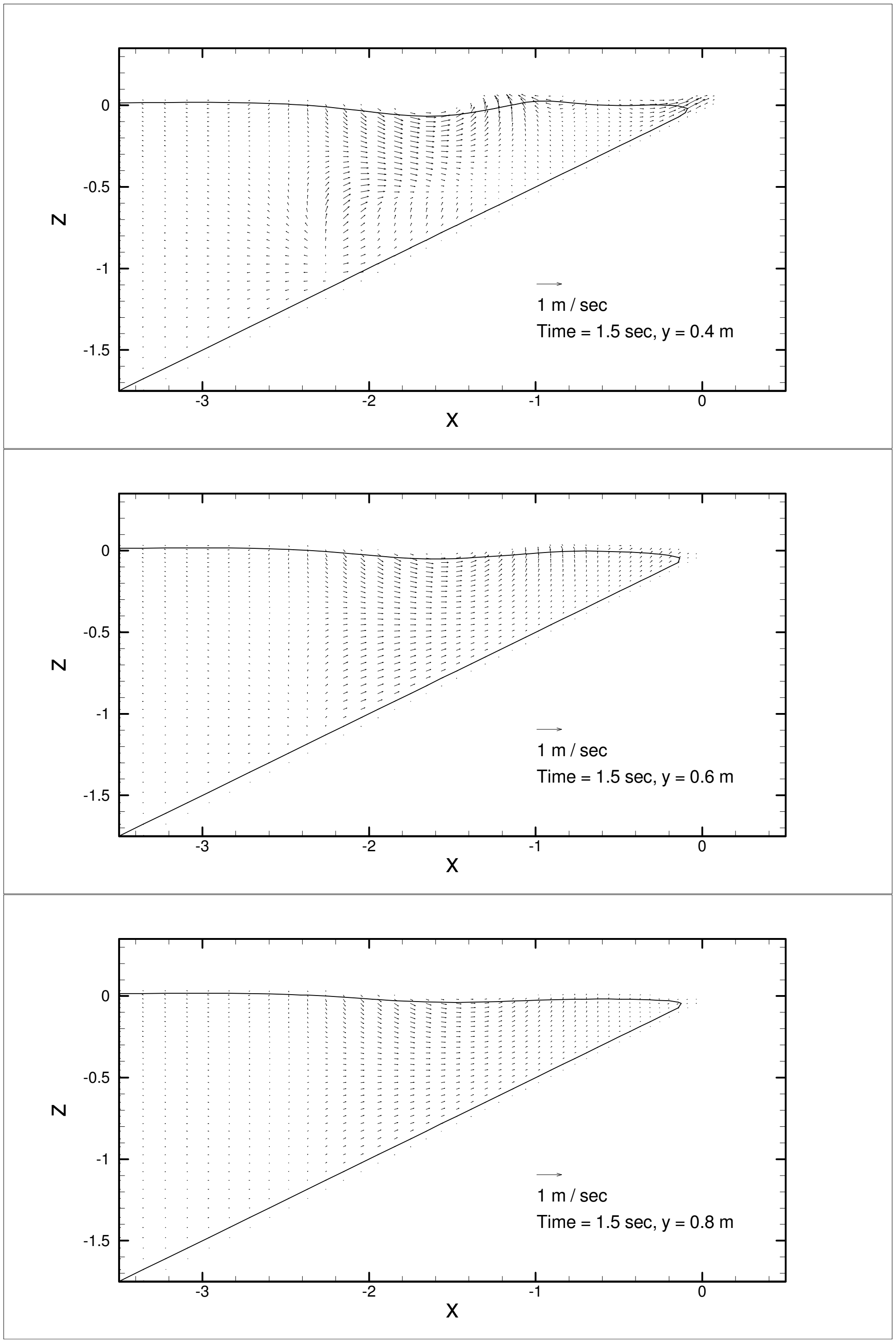


Figure 5.28: Snapshots of velocity vectors on the vertical y-planes at time = 1.5 sec for the sliding wedge with Δ=-0.05 m, and γ=2.79. The unit is in meter. The magnitude of the reference vector indicates the speed of the wedge.

Figure 5.28 (continued)



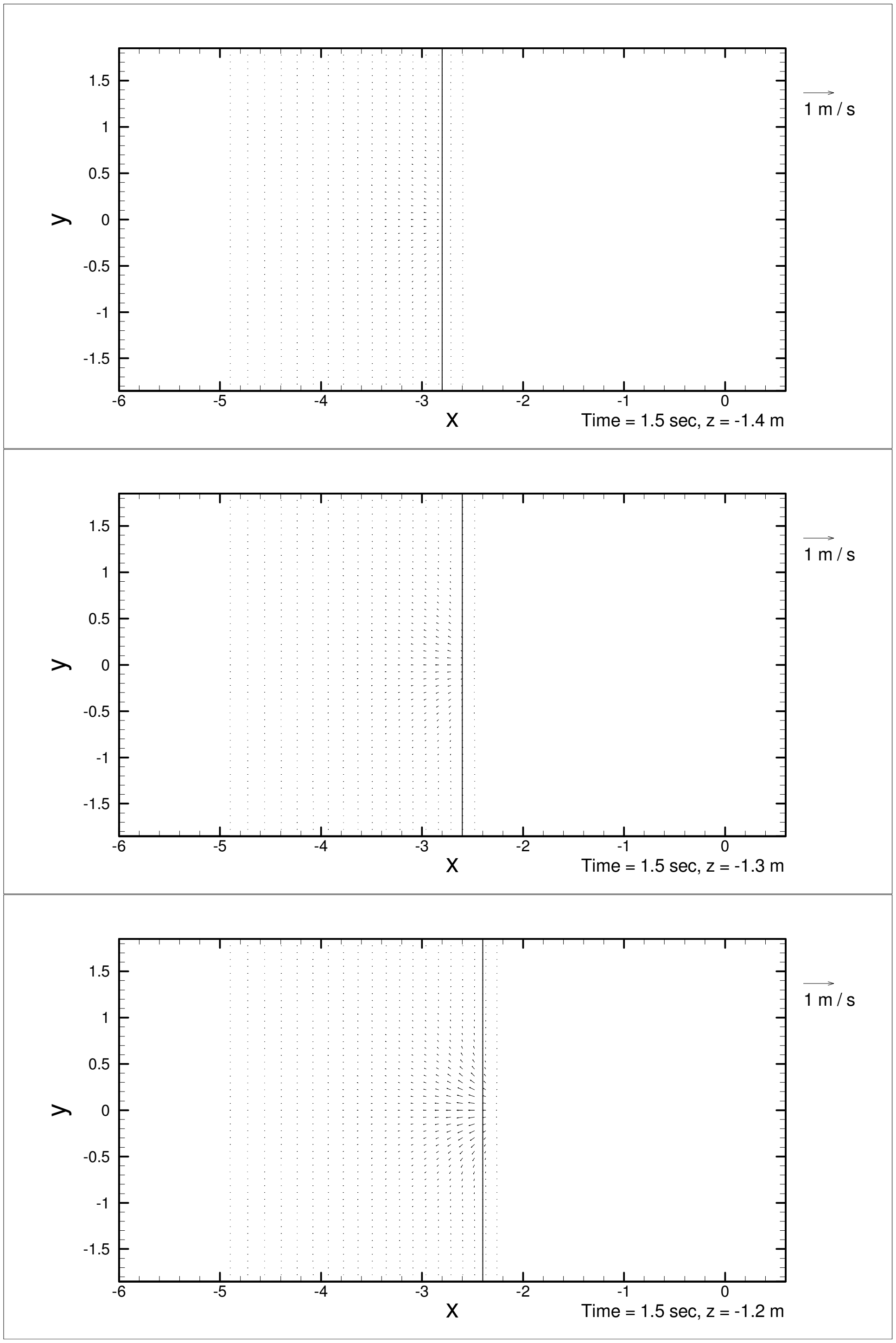


Figure 5.29: Snapshots of velocity vectors on the vertical z-planes at time = 1.5 sec for the sliding wedge with Δ=-0.05 m, and γ=2.79. The unit is in meter. The magnitude of the reference vector indicates the speed of the wedge.

Figure 5.29 (continued)

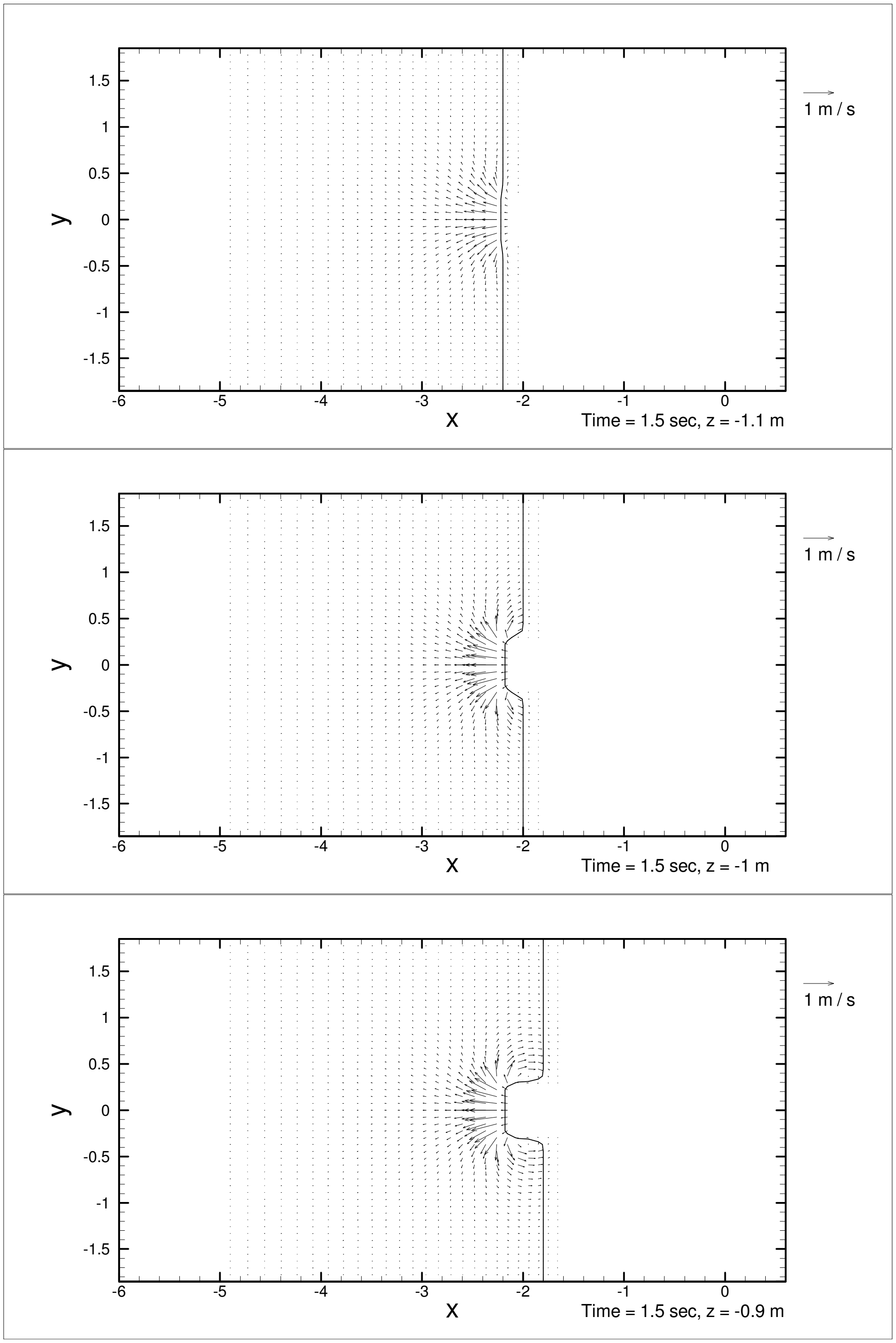


Figure 5.29 (continued)

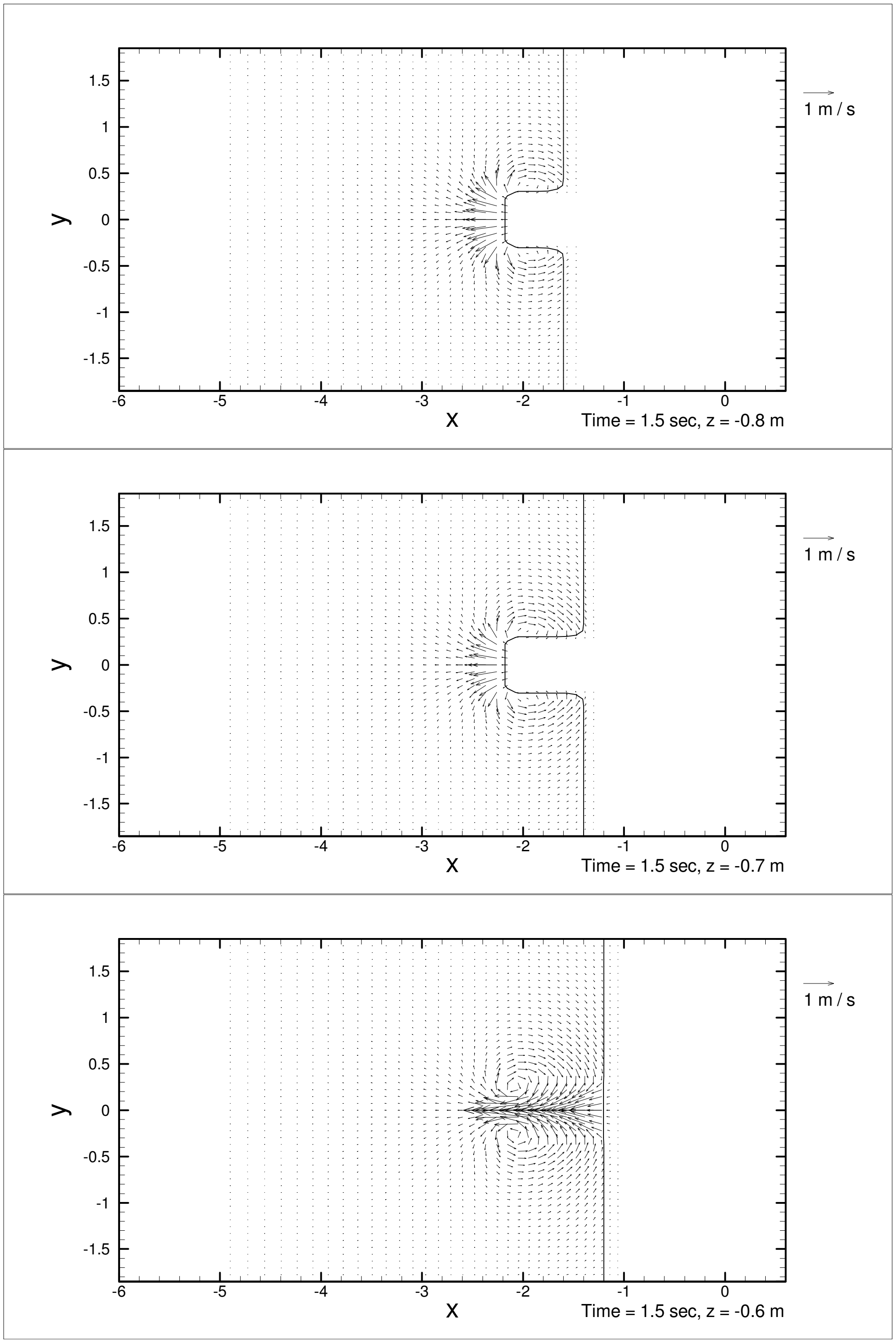


Figure 5.29 (continued)

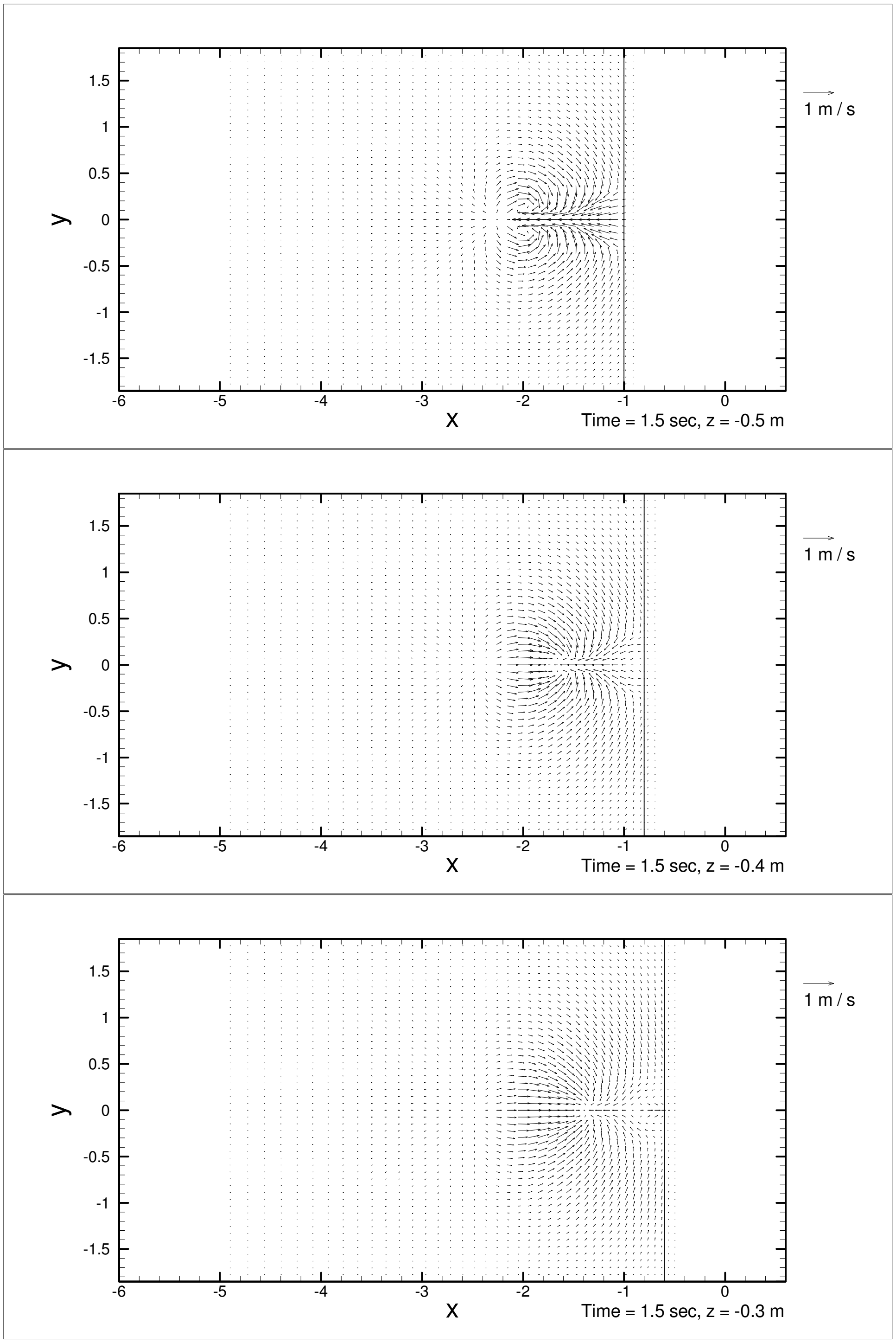
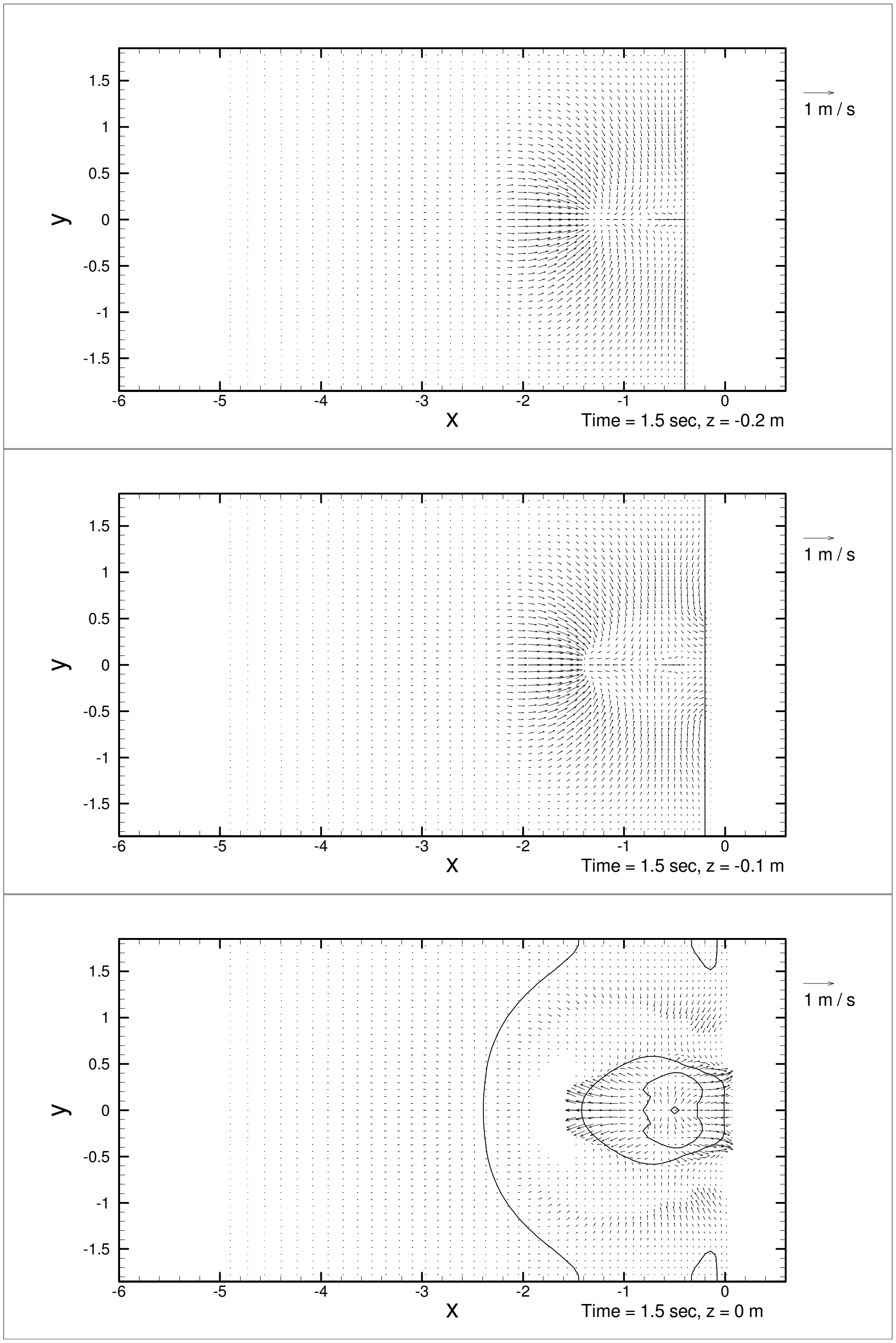


Figure 5.29 (continued)



5.8 Comparison between All Numerical Solutions and Laboratory Measurements

After the validations of the subaerial and submerged numerical simulations, the numerical model is then used to simulate 45 landslide cases for different initial slide elevations and specific weights. In this section, all the numerical solutions will be discussed and compared with the laboratory data.

Figure 5.30 shows R/B vs. $(\Delta/B)(\dot{u}/g)$, where R is the maximum runup height, B is the length of the slide, Δ is the initial elevation of the slide, $\dot{u}$ is the initial acceleration, and g is the gravity. The range of Δ is: $-0.3 \le \Delta \le 0.454$. In this study, because the trajectories of the moving slides are imposed directly from the laboratory data on the numerical model, both laboratory data and numerical solutions have the same $\dot{u}$. Figure 5.30 shows that the maximum runup heights predicted by the numerical model agree well with the laboratory data in the region where $(\Delta/B)(\dot{u}/g) > 0.04$ and $(\Delta/B)(\dot{u}/g) < -0.02$. However, the numerical model under-predicts the maximum runup height at the region where Δ is close to zero. This might be due to the concave curve of the shorelines, the same as those observed in the submerged case. Unfortunately, the exact reason is still unclear.

One important feature that can be observed from figure 5.30 is that the maximum runup height decays much faster in the region where $(\Delta/B)(\dot{u}/g) < 0.03$ than in the region where $(\Delta/B)(\dot{u}/g) > 0.03$. This implies that there is a strong energy dissipation mechanism (turbulence) which will cancel out the effect from large $(\Delta/B)(\dot{u}/g)$. This might indicate that the LES turbulence model is functioning well in the current study.

Figure 5.31 shows R/B vs. $(\Delta/B)\gamma$, where γ is the specific weight of the slide. Again, a very good agreement between the numerical solutions and the laboratory data in the region of $\left(\Delta / B\right)\gamma > 0.5$ and $\left(\Delta / B\right)\gamma^{-1} < -0.5$. The numerical model under-predicts the maximum runup height at the region where $\left(\Delta / B\right)\gamma$ is close to zero.

5.9 Cross Comparisons

In this section, the cross-comparisons of the shorelines and centerline cross-section velocity vectors will be presented. By fixing γ, we are able to see the effects of the initial slide elevation (Δ) on the shorelines (Figure 5.32) and the velocity vectors at the centerline cross-section (Figure 5.33). Also, by fixing Δ, we are able to see the effects of γ on the shorelines (Figure 5.34) and velocity vectors (Figure 5.35).

Figure 5.32 shows the effects of Δ on the shorelines. The selected specific weight (γ) is 3.43, and the time frame is selected when maximum runup height happens at the centerline cross-section. Observed from Figure 5.32, the higher slide initial elevation provides more momentum to the water and generates a higher runup. The convex type of shoreline curves can be seen at $\Delta \geq 0.2\text{m}$, and concave type of shoreline curves are shownung in the region where Δ is closer to zero ($-0.05 \leq \Delta \leq 0.05$). However, this feature disappears at $\Delta = -0.2$.

Figure 5.33 shows the velocity vectors at the centerline cross-sections. Observed from Figure 5.33, a similar feature appears in the different Δ in terms of the velocity vectors. As we have addressed in the discussion of subaerial landslide, the slide pushes the water in front of the slide and generates a current in the wake area. There exists an eddy in the region between the wake area and the upper returning current. The region affected by the movement of the slide is about one front-face height of the slide vertically and 1.5 times the front-face height horizontally. Due to the three dimensional vortex, the velocity on top of the slide is able to be faster than the moving speed of the slide ($\Delta = -0.2\text{m}$). This situation has been addressed in the discussion of the submerged landslide case.

Figure 5.34 and 5.35 show the effect of γ on the shoreline and velocity. Larger γ indicates larger potential energy. From Figure 5.34 we can see that larger γ generates a larger inundation area. Figure 5.35 shows the velocity vectors at the center-line cross-section. From the distances of the slide movement, we can see that the denser slide moves faster than the lighter slide (see γ = 2.14, 2.79 and 3.43) does. Therefore, the momentum provided by the denser slide is greater than that provided by the lighter one.

5.10 Conclusions

The numerical simulation has the advantage of observing phenomena cover the whole domain, which is not easy for the laboratory measurements especially for a large scale experiment with a moving obstacle. In this study, by adopting the displacement curve from the laboratory measurement, the numerical model is able to reproduce the near-field flow fields. The detailed numerical solutions are presented and discussed.

The relationship between maximum runup height and the initial elevation, initial acceleration and the specific weight of the slides is given in the results and discussion. The comparisons show that the numerical model has the capability to simulate the complex three-dimensional moving boundary type problem.

The landslide generated runup is caused by the “bounced” waves which are fully three-dimensional. Therefore, the two-dimensional simulations are not able to represent this mechanism and will significantly underestimate the maximum runup height and overestimate the offshore waves.

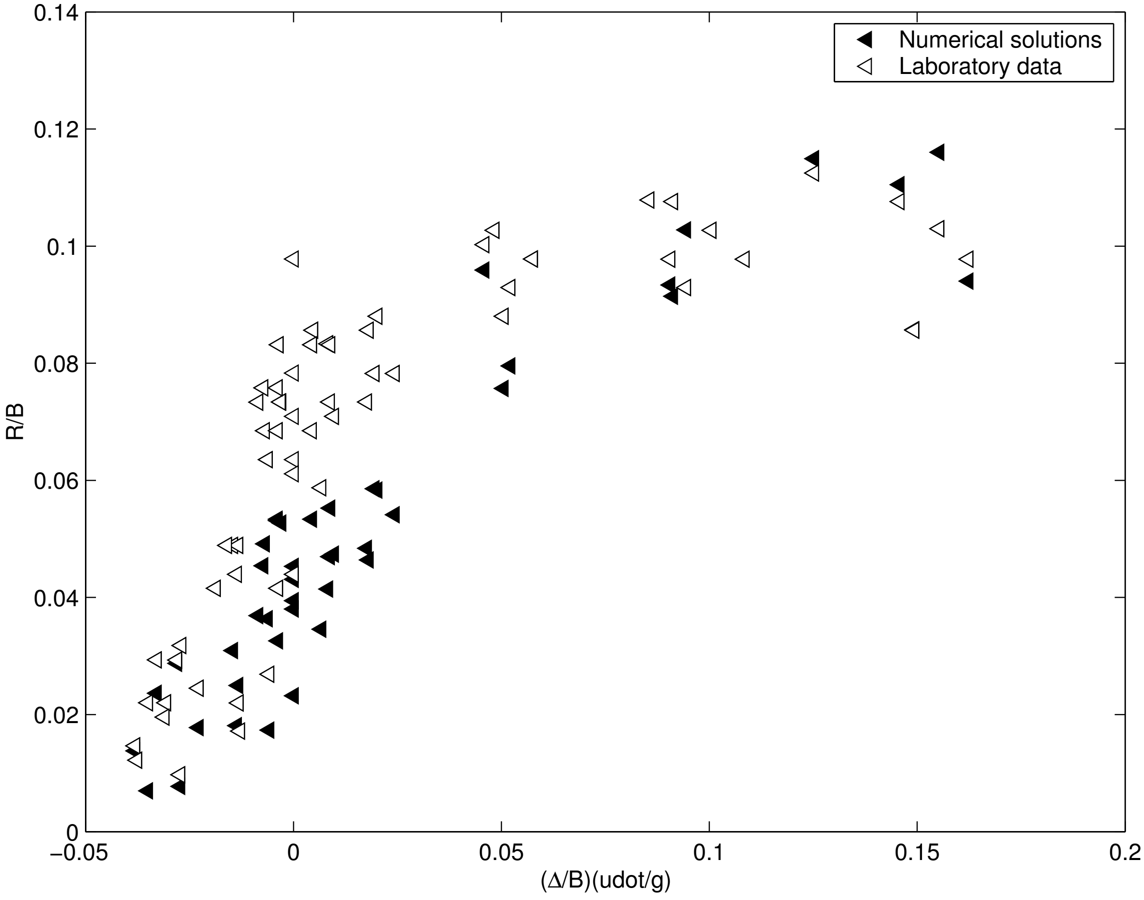


Figure 5.30: Comparison of all the numerical solutions to the laboratory data in terms of R/B vs. $(\Delta/B)(\dot{u}/g)$, where R is the runup height, B is the length of the wedge, Δ is the initial elevation of the wedge, $\dot{u}$ is the initial acceleration of the wedge measured by laboratory experiment, g is the gravity. The black triangles are numerical solutions. The white triangles are laboratory data.

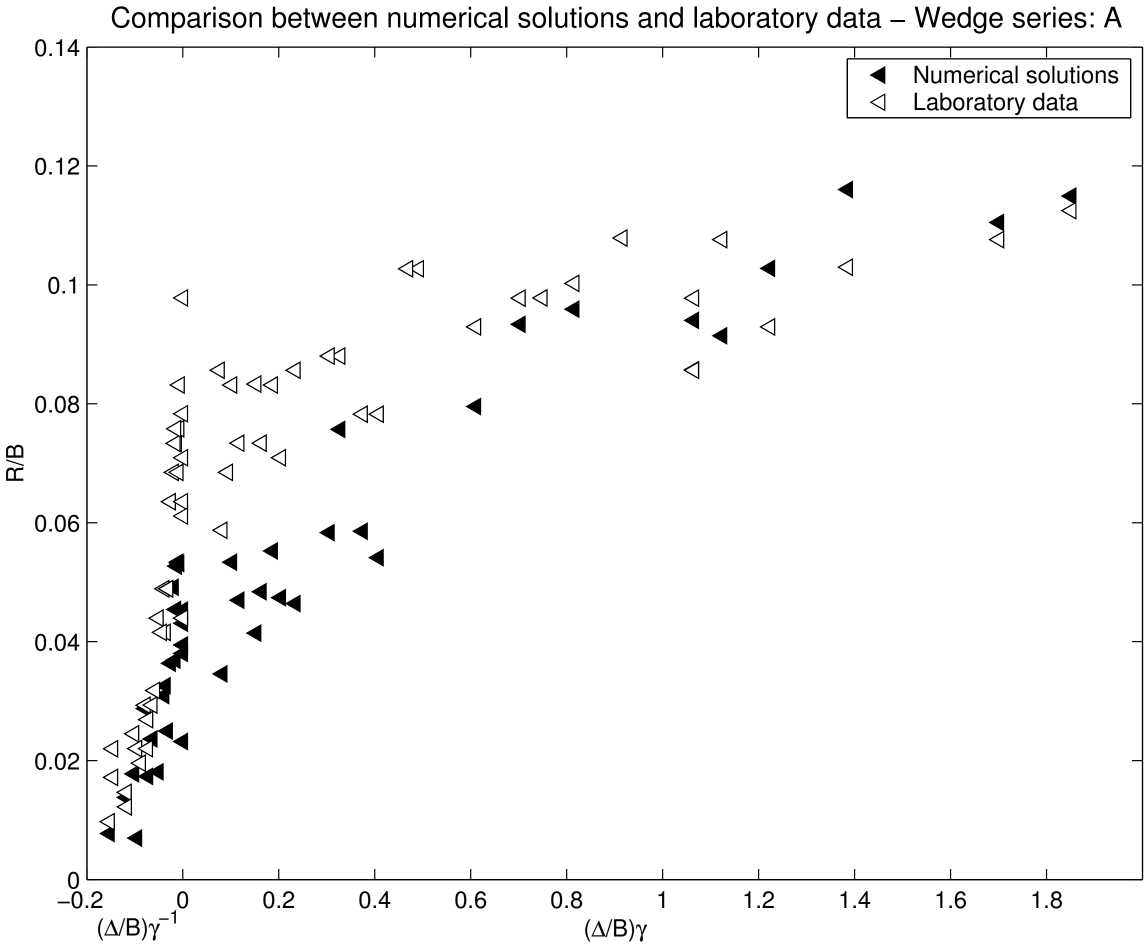
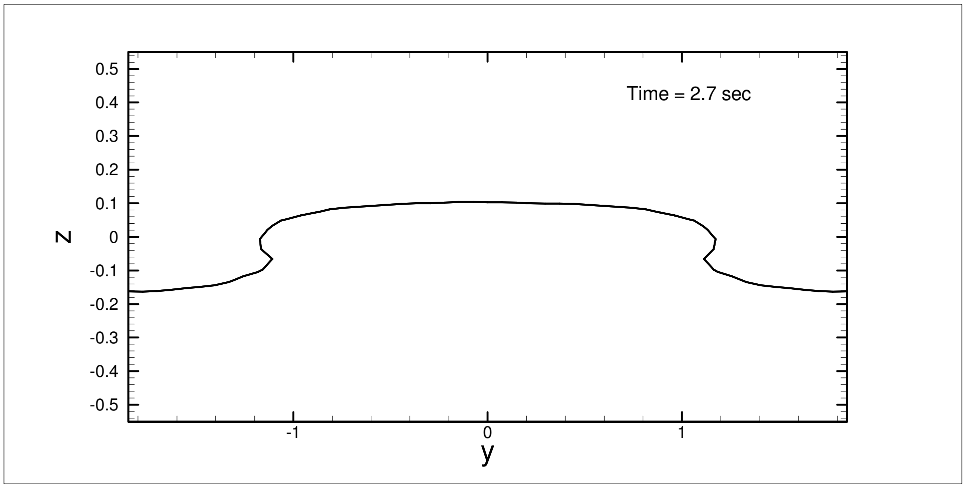
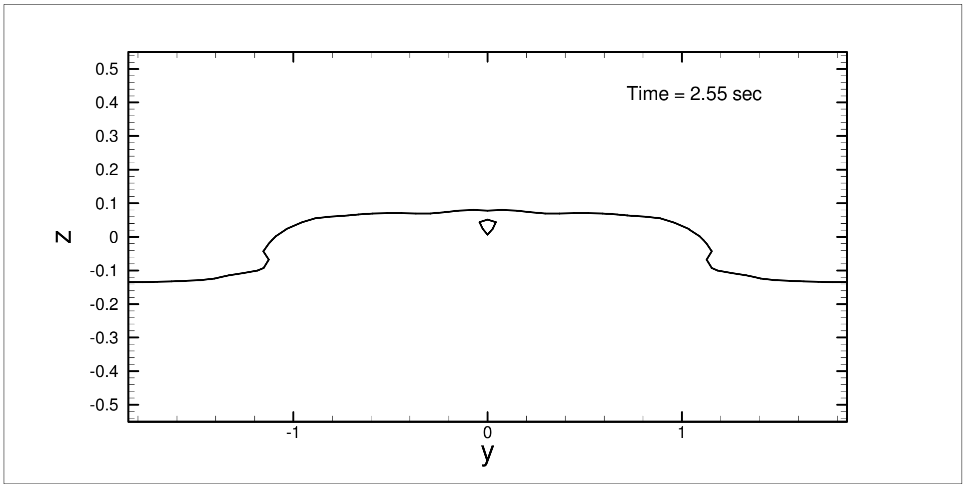


Figure 5.31: Comparison of all the numerical solutions to the laboratory data in terms of R/B vs. $(\Delta/B)\gamma$, where R is the runup height, B is the length of the wedge, Δ is the initial elevation of the wedge, γ is the specific weight of the wedge. The black triangles are numerical solutions. The white triangles are laboratory data.



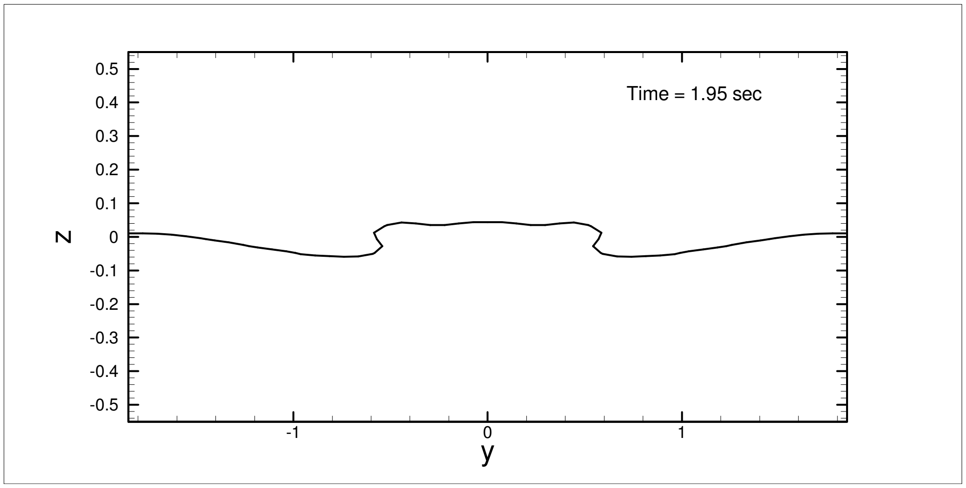
$\Delta = 0.454$ m, $\gamma = 3.43$.



$\Delta = 0.20$ m, $\gamma = 3.43$

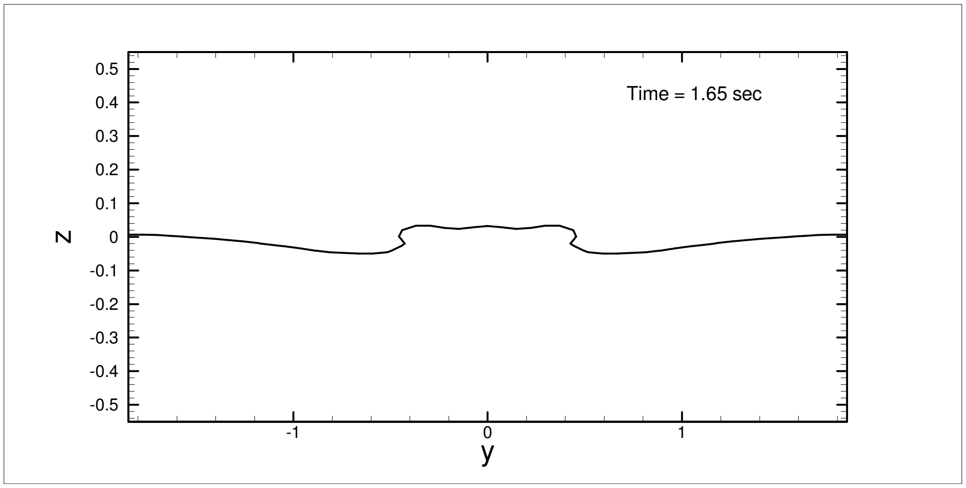
Figure 5.32: Cross comparisons of shoreline curves for a fixed wedge specific weight (γ =3.43). The time frame is chosen when the maximum runup height occurred. (Unit is in meter)

Figure 5.32 (continued)



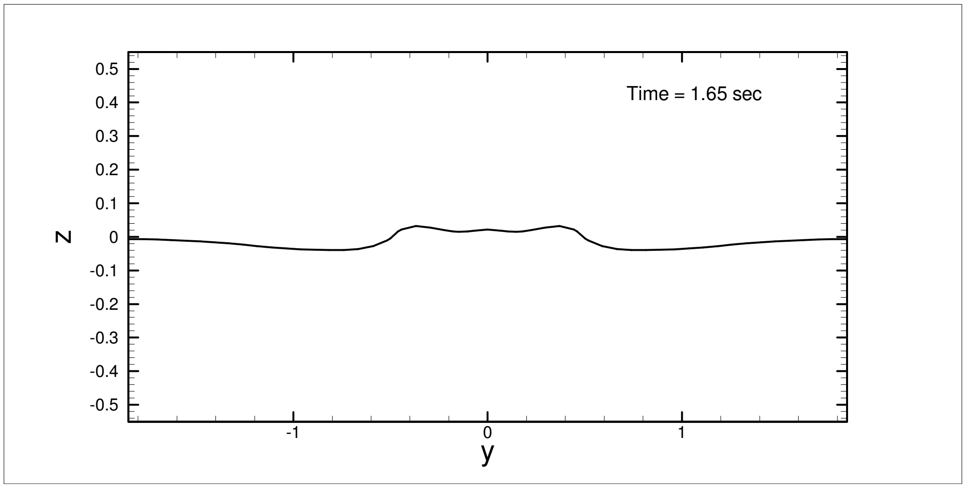
$\Delta = 0.05$ m, $\gamma = 3.43$.

.

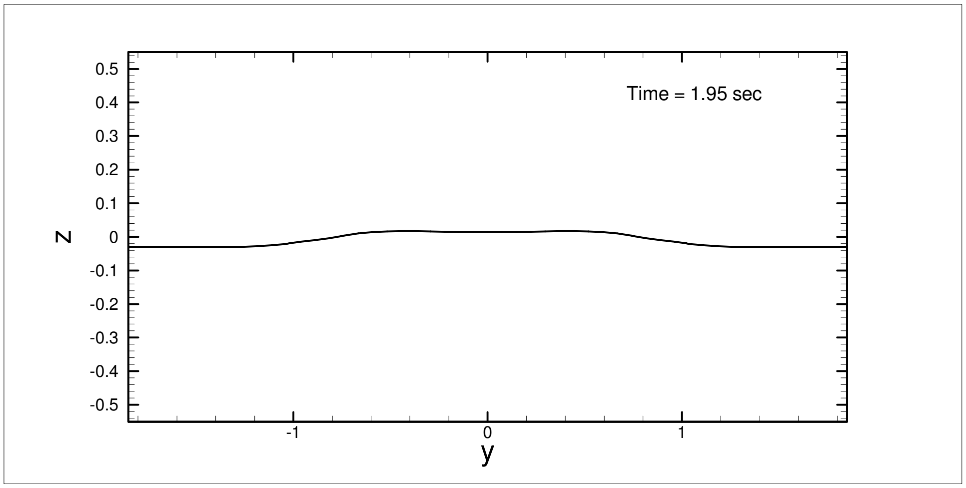


$\Delta = 0.0$ m, $\gamma = 3.43$.

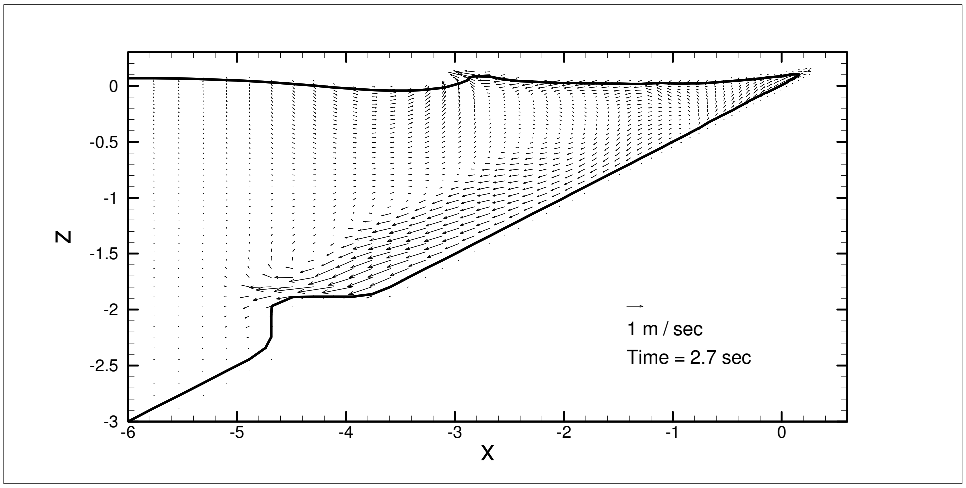
Figure 5.32 (continued)



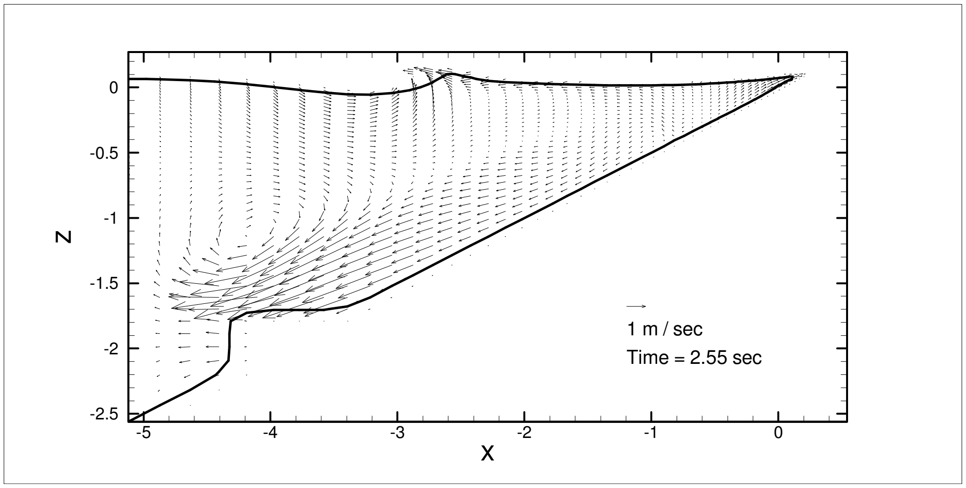
$\Delta = -0.05$ m, $\gamma = 3.43$.



$\Delta = -0.20$ m, $\gamma = 3.43$.



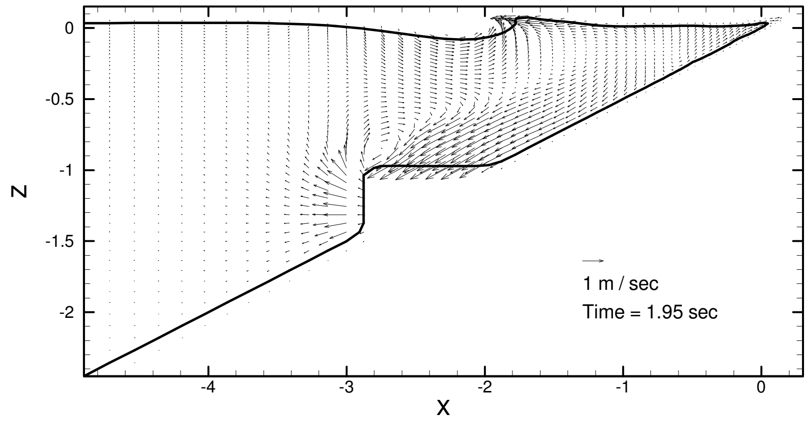
$\Delta = 0.454$ m, $\gamma = 3.43$.



$\Delta = 0.20$ m, $\gamma = 3.43$

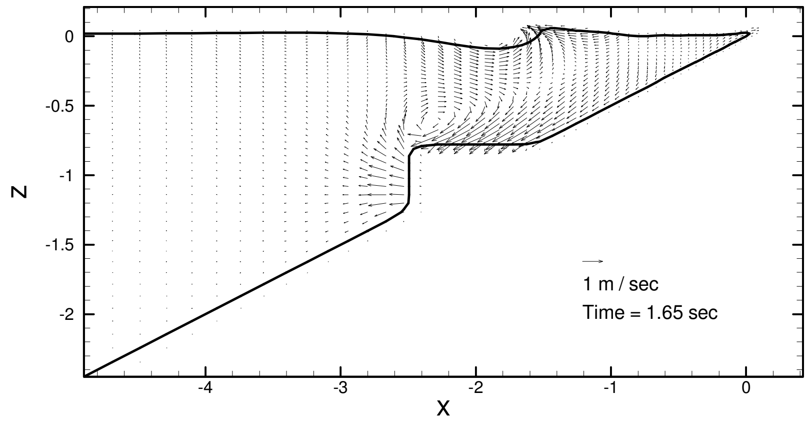
Figure 5.33: Cross comparisons of velocity vectors at centerline vertical plane for a fixed wedge specific weight (γ =3.43). The time frame is chosen when the maximum runup height occurred. (Unit is in meter)

Figure 5.33 (continued)



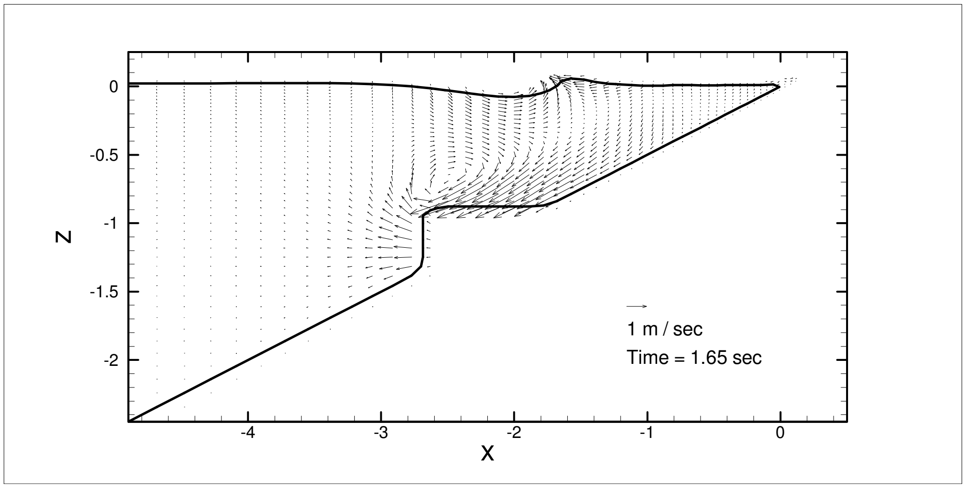
$\Delta = 0.05$ m, $\gamma = 3.43$.

.

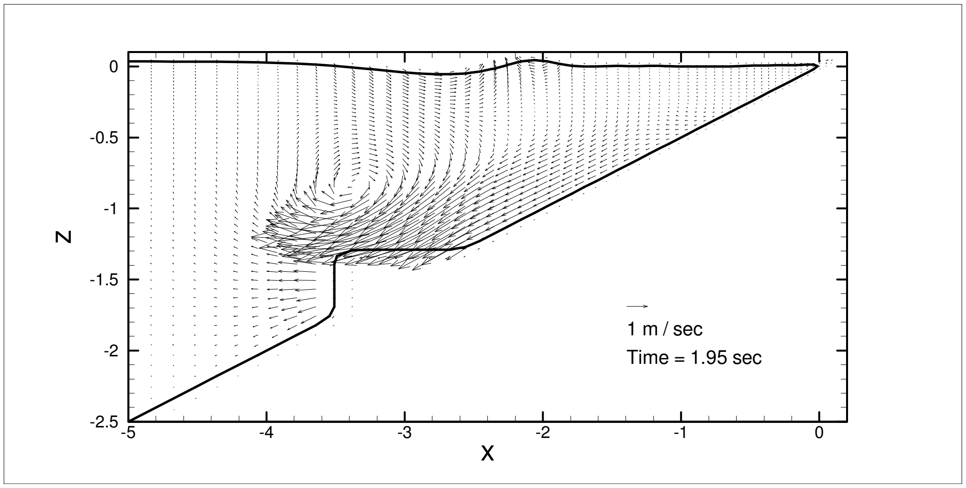


$\Delta = 0.0$ m, $\gamma = 3.43$.

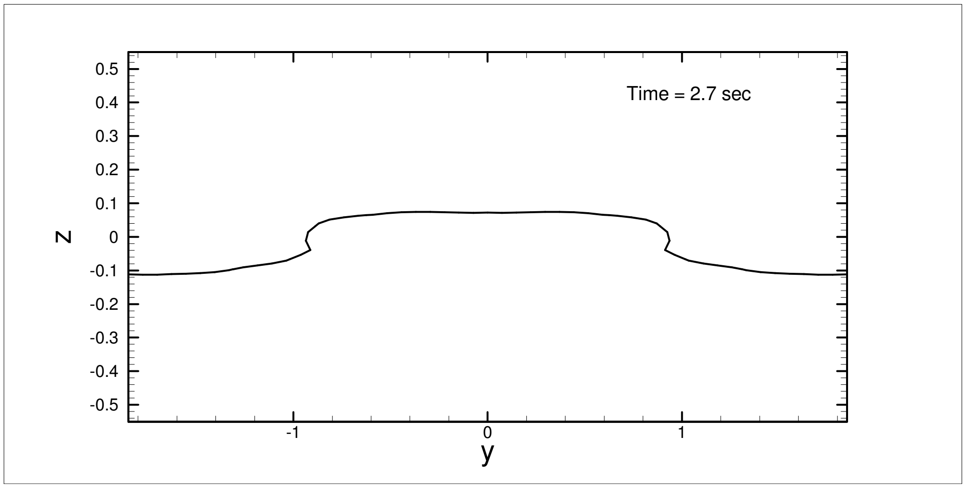
Figure 5.33 (continued)



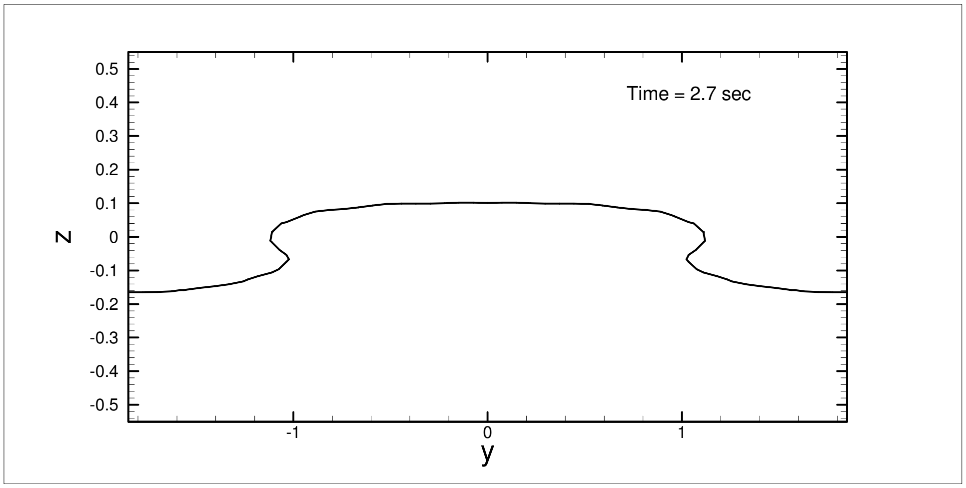
Δ = -0.05 m, γ = 3.43.



Δ = -0.20 m, γ = 3.43.



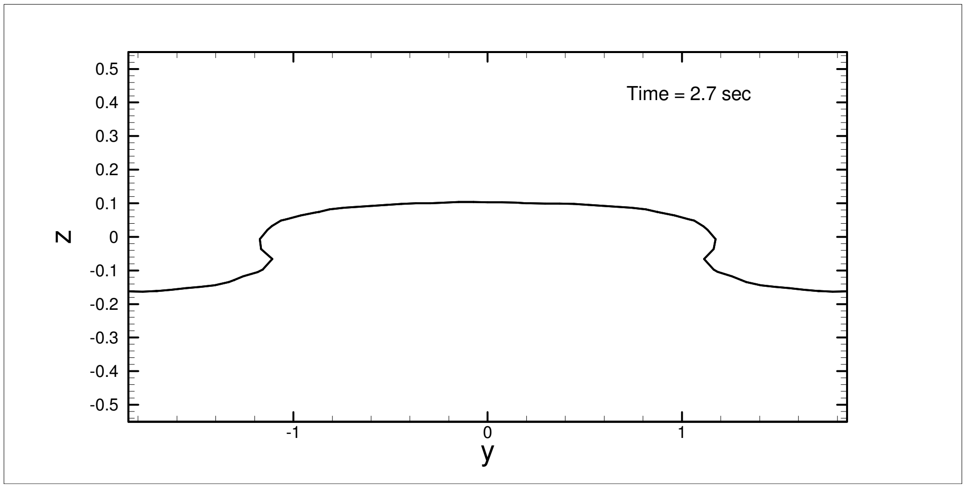
$\Delta = 0.454$ m, $\gamma = 2.14$.



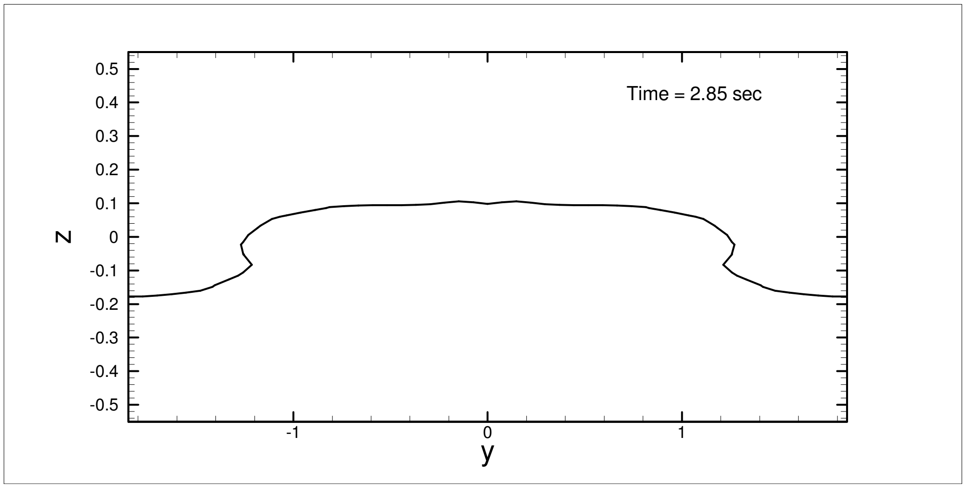
$\Delta = 0.455$ m, $\gamma = 2.79$.

Figure 5.34: Cross comparisons of shoreline curves for a fixed wedge initial position (Δ=0.454 m). The time frame is chosen when the maximum runup height occurred. (Unit is in meter)

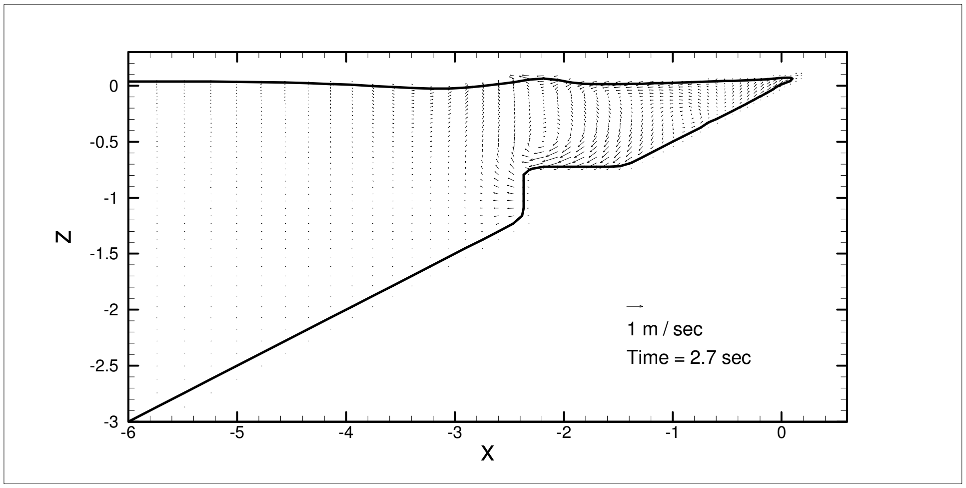
Figure 5.34 (continued)



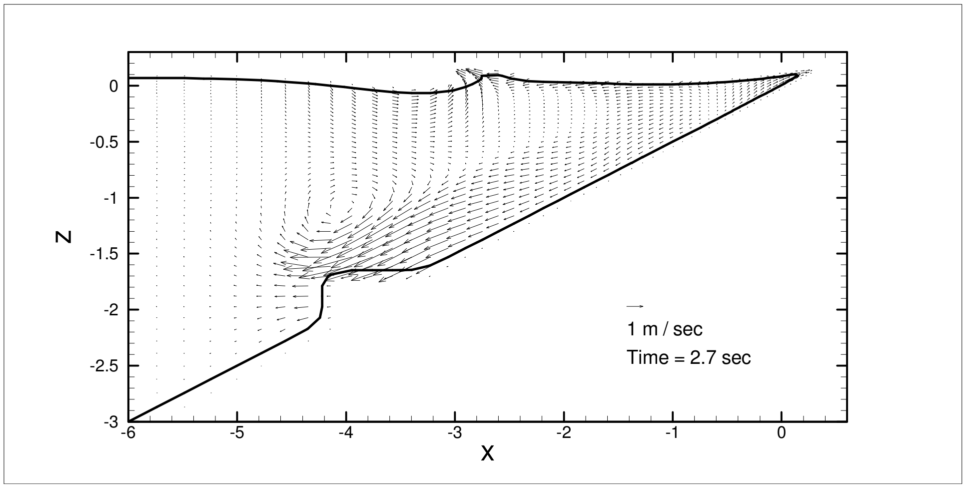
$\Delta = 0.455$ m, $\gamma = 3.43$.



$\Delta = 0.455$ m, $\gamma = 3.73$.



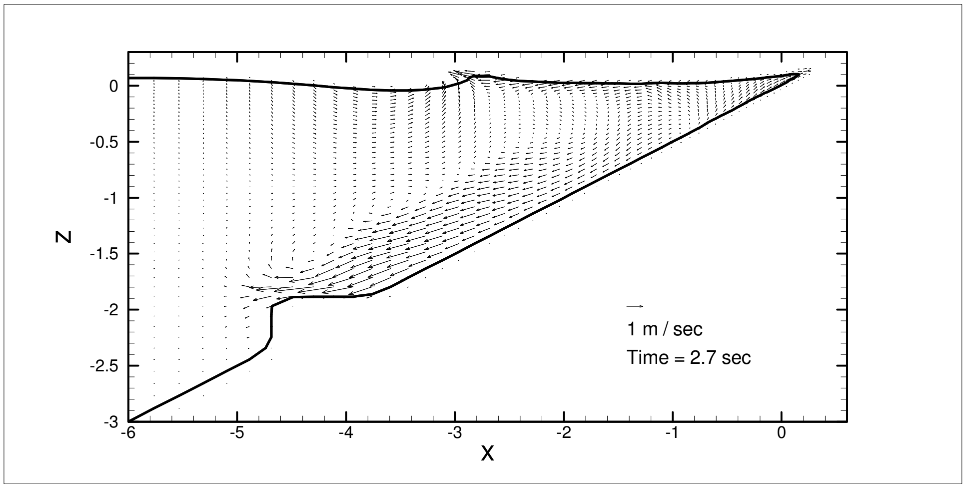
$\Delta = 0.454$ m, $\gamma = 2.14$.



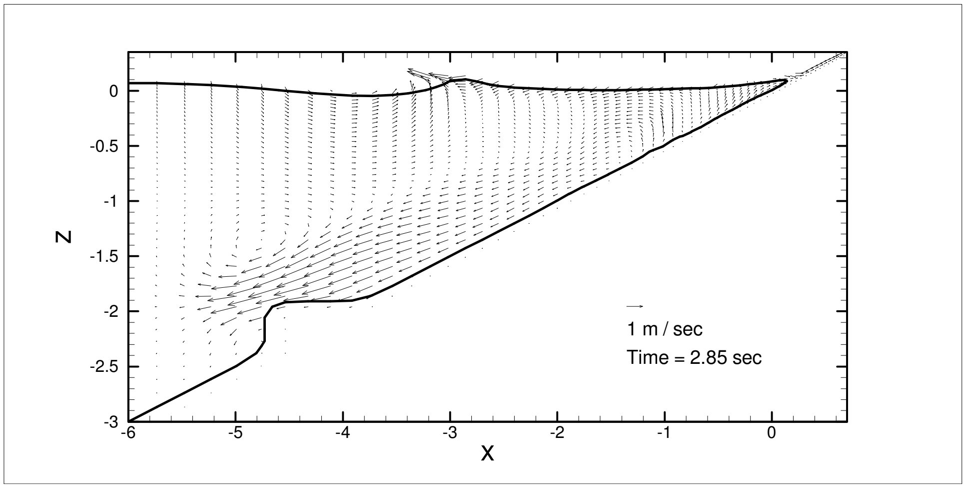
$\Delta = 0.455$ m, $\gamma = 2.79$.

Figure 5.35: Cross comparisons of velocity vectors at centerline vertical plane for a fixed wedge initial position ($\Delta = 0.454$ m). The time frame is chosen when the maximum runup height occurred. (Unit is in meter)

Figure 5.35 (continued)



$\Delta = 0.455$ m, $\gamma = 3.43$.



$\Delta = 0.455$ m, $\gamma = 3.73$.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In this thesis, a numerical model is developed to solve general three-dimensional water wave problems. The numerical model is based either on the Reynolds Averaged Navier-Stokes (RANS) equations for mean flow field or the Large Eddy Simulation (LES) for the filtered flow fields. For RANS equations, we adopt both linear and nonlinear $k-\varepsilon$ models as our turbulence closure models. As for the LES, the standard Smagorinsky model is chosen as the turbulence closure model.

On the free surface, the piecewise linear volume of fluid (VOF) method with second order accuracy is used to track the free surface movement. For the moving boundary problem, a moving solid algorithm is developed to solve the landslide generated waves.

The model's accuracy in simulating mean flow motions has been carefully examined by both 2D and 3D cases. In the 2D case, a solitary wave propagating in a constant water depth is simulated. The numerical solution is compared with the analytical solution. The conservation laws of the model have been inspected. For the 3D case, the simulation results of a non-breaking solitary wave interacting with a circular cylinder are verified against the experimental data. In this case, an unstructured grid is used. This case also demonstrates that model's capability of simulating flows in an irregular domain. The ability of using an irregular mesh is an important advantage of the finite volume discretization method. In order to deal with many complex wave conditions, many numerical tools have been developed for this model, such as incident wave boundary conditions, internal wavemaker, and numerical sponge layer. The accuracy of these numerical tools is examined. Overall, the performance of the numerical model for non-breaking waves is excellent. This demonstrates that the model is accurate enough in simulating mean flow motions.

The model is then employed to investigate breaking waves in the surf zone. A 2D spilling breaker on a sloping beach is simulated. In this case, the nonlinear $k-\varepsilon$ model is utilized to evaluate the strong turbulence effects during the breaking. The model is verified against the laboratory measurements in terms of the mean free surface displacement, mean velocities, initial breaking location, and turbulence intensity. From this simulation, we are able to evaluate the performance of the turbulence model. Except the early stage of the breaking, the nonlinear $k-\varepsilon$ model is able to predict the turbulence field. Some important characteristics of spilling breakers in the surf zone are summarized as below:

- The spilling breaking is confined in a small region very close to the wave crest area.
- The turbulence is generated in the region near the breaking wave front. This region is often referred to as "roller" region. Inside this region, the particle velocity near the free surface is close to the phase velocity.
- Because the turbulence is generated in the roller region, the breaking processes only have a local influence beneath the breaking wave.

A 3D case of dam-break waves interacting with a square cylinder is investigated to evaluate the performance of nonlinear $k-\varepsilon$ turbulence models in terms of the wave force acting on the structure and the time-history of velocity. The results are examined by comparing with the laboratory measurements. Overall, the velocities predicted by nonlinear $k-\varepsilon$ turbulence models are close to the laboratory data.

The wave force acting on the square cylinder is dominated by the normal stress. The shear stress acting on the cylinder faces is about 1% of the normal stress. Therefore, the shear stress can be neglected when calculating the net force on the cylinder. In this case, we also present the detailed turbulence generation and propagation with the dam-break bore.

From the 2D spilling breaker simulation, we validate the accuracy of the turbulence model. From simulations of the 3D dam-break wave interacting with a cylinder, we further demonstrate the capability of describing the turbulence field of current numerical model by utilizing nonlinear $k-\varepsilon$ turbulence model.

In Chapter 5, the LES turbulence model with a moving solid algorithm is used to simulate the landslide generated waves. The detailed moving solid algorithm is developed and used to simulate the momentum introduced by the movement of the slides. The landslide generated waves usually come along with the strong turbulence in the wake zone of the moving body. For this reason, the LES turbulence model with Smagorinsky SGS model is adopted. The numerical results are compared with the laboratory measurements in terms of the maximum runup height, the time-history runup height and time-history free surface elevation. The simulations show that the present model successfully predicts the runup in the near-field region. A total of 45 simulations are conducted by changing the initial elevation and specific weight of the slides. The overall comparisons are very good in terms of the maximum runup height and free surface elevation. Some important characteristics of landslide generated waves are summarized as below:

- The LES results illustrate the characteristics of the complex 3D free surface flows. Flows are three-dimensional, transient, rotational, and turbulent.
- The maximum runup is caused by the rebounding waves, which are the results of the convergent flows above the sliding wedge
- The runup and rundown are controlled by wedge size, submergence and initial motion time history.
- Higher initial elevation cases have higher runup.
- Heavier slides (with larger gamma values) generate larger inundation area.
- The region influenced by the sliding mass on the y-z plane is roughly 1.5~2.0 times the slide cross-section. Therefore, when the submergence is greater than 3.0 times the height of the slide, the slide becomes ineffective in generating waves.

6.2 Future Work

The numerical model presented in this thesis has been proven to be an accurate three-dimensional hydrodynamic model, and can serve for studying more practical coastal problems in the future. However, for a practical simulation, one of the most important issues is the computational cost. Currently, our simulations are confined in a numerical mash with about 70 x 70 x 70 grid numbers. A grid number greater than this limitation will usually require more than one week to finish one case on one single CPU, which is very time consuming. A parallel computation is a solution to this problem. In parallel programming, because multiple CPUs are working together at the same time and on the same simulation, the time (wall clock) can be reduced. The latest version, Truchas 2.0, has provided a full function in parallel computing in summer 2004. After adopting Truchas 2.0 and upgrading our computer facilities, we will have a chance to obtain a solution with a better resolution.

The success in developing a moving solid algorithm greatly extends the model capability to simulate wave-structure interaction problems. Except for the landslide problems, the algorithm can be used to simulate waves generated by a strong paddle movement or ship waves. Most of the coastal hydrodynamic problems with a moving solid boundary can be solved by the moving solid algorithm. One possible extension of the moving solid algorithm is to solve the "Bingham fluids". Many materials such as volcanic lava and submarine sediments behave approximately like a non-Newtonian fluid. These materials have a critical value of shear stress (or yield stress). Below this yield stress, they behave like a rigid body; above this yield stress, they behave like a viscous fluid. The Bingham plastic model is best-known for describing this viscoplastic behavior. Because the Bingham fluids will change from a solid-like material to a viscous fluid, a numerical solver shall have the capabilities of dealing with the fluid motion, the solid motion, and the phase change. Currently, our numerical model has satisfied the first two requirements. Hopefully in the near future, our model can be used to solve Bingham fluid problems such as "waves generated by flow slides" problems.

Bibliography

[1] Abbott, M. B., Petersen, H. M., and Skovgaard, O. 1978 On the numerical modeling of short waves in shallow water. J. Hydr. Res., 16(3), 173-203.

[2] Abbott, M. B., McCowan, A. D., and Warren, I. R. 1985 Accuracy of short-wave numerical models. J. Hydr. Engng., 110(10), 1287-1301.

[3] Abul-Azm, A. G. 1994 Diffraction through wide submerged breakwaters under oblique waves. Oc. Engng., 21, 683-706.

[4] Antunes do Carmo, J. S., Seabra-Santos, F. J. and Barthélemy, E. 1993 Surface waves propagation in shallow-water: A finite element model. Int. J. for Num. Meth. in Fluids, Vol. 16, 447 – 459.

[5] Baggett, J. S., Jimenez, J., and Kravchenko, A. G. 1997 Resolution requirements in large-eddy simulations of shear flows. In: Annual Research Briefs. Center for Turbulence Research, Standford, CA, 51-66.

[6] Bardet, J.-P., Synolakis, C. E., Davies, H. L., Imamura, F., and Okal, E. A. 2003 Landslide Tsunamis: Recent Findings and Research Directions. Pure Appl. Gepohys. 160, 1793-1809.

[7] Barth, T. J. 1995 Aspects of unstructured grids and finite-volume solvers for Euler and Navier Stokes equations, VKI/NASA/AGARD Special Course on Unstructured Grid Methods for Advection Dominated Flows, AGARD Publication R-787.

[8] Basco, D. R. 1985 A qualitative description of wave breaking. J. Wtrwy., Port, Coast., and Oc. Engng., ASCE, 111(2), 171-188.

[9] Battjes, J. A. 1974 Surf similarity. Proc. 14th Int. Coast. Engng. Conf., Copenhagen. ASCE, New York, 466-480.

[10] Battjes, J. A. 1988 Surf-zone dynamics. Ann. Rev. Fluid Mech., 20, 257-293.

[11] Beji, S. and Battjes, J. A. 1993 Experimental investigation of wave propogation over a bar. Coast. Engng. 19, 151-162.

[12] Bradford, S. F. 2000 Numerical simulation of surf zone dynamics. J Wtrwy., Port C-ASCE 126 (1): 1-13.

[13] Bradshaw, P., Cebeci, T., and Whitelaw, J. H. 1981 Engineering Calculation Methods for Turbulence Flow, Academic Press, London.

[14] Cabot, W. and Moin, P. 2000 Approximate wall boundary conditions in the large-eddy simulation of high Reynolds number flow. Flow Turb. Combust. 63, 269-291.

[15] Cai, C.-C. 1995 A family of overlapping Schwarz algorithms for nonsymmetric and indefinite elliptic problems. In D. E. Keyes, Y. Saad, and D. G. Truhlar, editors, Domain-Based Parallelism and Problem Decomposition Methods in Computational Science and Engineering, pages 1-19, Philadelphia, PA, SIAM.

[16] Carati, D., Winckelmans, G., and Jeanmart, H. 2001 On the modeling of the subgrid-scale and filtered-scale stress tensors in large-eddy simulation. J. Fluid Mech. 441, 119-138.

[17] Chang, K.-A. and Liu P. L.-F. 1996 Measurement of breaking waves using particle image Velocimetry. Proc. 25th Int. Conf. Coast. Engng., ASCE, 527-536.

[18] Chang, K.-A. and Liu, P. L.-F. 1999 Experimental investigation of turbulence generated by breaking waves in water of intermediate depth. Phys. of Fluids, 11,

11, 3390-3400.

[19] Chang, K.-A., Hsu, T.-J., and Liu, P. L.-F. 2001 Vortex generation and evolution in water waves propagating over a submerged rectangular obstacle Part I. Solitary waves. Coast. Engng. 44, 13-36.

[20] Chiang, W. L., Divorky, D., Parnicky, P., and Wier, W. 1981 Numerical model of landslide-generated waves. Tetra Tech Report T-3427, U.S. Department of Commerce, Pasadena, Calif.

[21] Chorin, A. J. 1968 Numerical solution of the Navier-Stokes equations. Math. Comp., 22, 745-762.

[22] Chorin, A. J. 1969 On the convergence of discrete approximations of the Navier-Stokes equations. Math. Comp., 23, 341-353.

[23] Christensen, E. D., and Deigaard R. 2001 Large eddy simulation of breaking waves, Coast. Engng. 42, 53-86.

[24] Chyu, C. K. and Rockwell, D. 1996 Near-wake structure of an oscillating cylinder: effect of controlled shear-layer vortices. J. Fluid Mech. 322, 21-49.

[25] Craft, T. J., Launder, B. E., and Suga, K. 1997 Prediction of turbulent transitional phenomena with a nonlinear eddy-viscosity model. Int. J. Heat Fluid Flow, 18, 15-28.

[26] Davies, H. L. 1999 Tsunami PNG 1998, National Library of Papua New Guinea, 48 pp.

[27] Deardorff, J. W. 1970 A numerical study of three-dimensional turbulent channel flow at large Reynolds numbers. J. Fluid Mech., 41(2), 452-480.

[28] Deardorff, J. W. 1974 Three-dimensional numerical study of the height and mean structure of a heated planetary boundary layer. Boundary-Layer Meterol. 7, 81-106.

[29] Demuren, A. O. and Sarkar S. 1993 Perspective: systematic study of Reynolds

stress closure models in the computations of plane channel flows. J. Fluids Engng., 115, 5-12.

[30] Dommermuth, D. G., Yue, D. K., Lin, W. M., and Rapp, R., J. 1988 Deepwater plunging breakers: a comparison between potential theory and experiments. J. Fluid Mech., 189, 423-442.

[31] Ertekin, R. C. and Becker, J. M. 1996 Nonlinear diffraction of waves by a submerged shelf in shallow water. In: Chakrabarti, S. et al. (Eds), Offshore Mech. and Arctic Engng. 1996, vol. I Pt. B, ASME, New York, 31-39.

[32] Fadlun, E. A., Verzicco, R., Orlandi, P., and Mohd-Yusof, J. 2000 Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. J. Comp. Phys. 161, 35-60.

[33] Ferziger, J. H. and Peric, M. 1996 Computational Methods for Fluid Dynamics. Springer-Verlag.

[34] Ferziger, J. H. and Peric, M. 2002 Computational Methid for Fluid Dynamics, Springer, 3rd Edition, 294-295.

[35] Galvin, C. J. 1968 Breaker type classification on three laboratory beaches. J. Geophys. Res. 73, 3651-3659.

[36] Gatski, T. B. and Speziale, C. G. 1993 On explicit algebraic stress models for complex turbulent flows. J. Fluid Mech., 254, 59-78.

[37] Goda, T. 1970 A synthesis of breaker indices. Trans. Jpn. Soc. Civ. Engng. 2, 227-230.

[38] Goldstein, D., Handler, R., and Sirovich, L. 1993 Modeling a no-slip flow boundary with an external force field, J. Com. Phys. 105, 354.

[39] Golub, G. H. and Van Loan, C. F. 1989 Matrix Computations. Johns Hopkins University Press.

[40] Gozali, S., and Hunt, B. 1989 Water waves generated by close landslides, J.

Hydr. Res., 27(1), 49-60.

[41] Grilli, S. T., Vogelmann, S., and Watts, P. 2002 Development of a 3D numerical wave tank for modeling tsunami generation by underwater landslides, Engng Analysis Boundary Elemt. 26(4), 301-313.

[42] Grue, J. 1992 Nonlinear water waves at a submerged obstacle or bottom topography. J. Fluid Mech. 244, 455-476.

[43] Gullbrand, J. and Chow, F. K. 2003 The effect of numerical errors and turbulence models in large-eddy simulations of channel flow, with and without explicit filtering, J. Fluid Mech. Vol. 495, pp. 322-341.

[44] Harlow, F. H. and Welch, J. E. 1965 Numerical calculation of time-dependent viscous incompressible flow. Phys. Fluids, 8, 2182-2189.

[45] Hattori, M. and Aono, T. 1985 Experimental study on turbulence structure under spilling breakers. In the Ocean Surface (ed. Y. Toba and H. Mitsuyasu), 419-424.

[46] Heinrich, P. 1991 Nonlinear numerical model of landslide-generated water waves. Int. J. Engng. Fluid Mech., 4(4), 403-416.

[47] Heinrich, P. 1992 Nonlinear water waves generated by submarine and aerial landslides. J. Wtrwy., Port, Coast., and Oc. Engng., ASCE, Vol. 118, No. 3, 249-266.

[48] Hirt, C. W. and Nichols, B. D. 1981 Volume of fluid (VOF) method for the dynamics of free boundaries. J. Comp. Phys., 39, 201-225.

[49] Huang, C.-J. and Dong, C.-M. 1999 Wave deformation and vortex generation in water waves propagating over a submerged dike. Coast. Engng. 37, 123-148.

[50] Israeli, M. and Orszag, S. A. 1981 Approximation of radiation boundary conditions. J. Comp. Phys., 41 115-131.

[51] Jiang, L., and LeBlond, P. H. 1994 Three-dimensional modeling of tsunami

generation due to a submarine landslide, J. Phys. Oceanog., 24, 559-572.

[52] Issa, R. I. 1986 Solution of implicitly discretized fluid flow equations by operator-splitting. J. Comput. Phys., 62, 40-65.

[53] Iwata, K., Kawasaki, K., and Kim, D.-S. 1996 Breaking limit, breaking and postbreaking wave deformation due to submerged structures. Proc., 25th Int. Conf. on Coast. Engng., ASCE, Reston, VA., 2335-2351.

[54] Jaluria, Y. and Torrance, K. E. 1986 Computational Heat Transfer. Washington, D. C., Hemisphere.

[55] Jiang, L., and LeBlond, P. H. 1994 Three-dimensional modeling of tsunami generation due to a submarine landslide, J. Phys. Oceanog., 24, 559-572.

[56] Johnson, B. D., Kobayashi, N., and Cox D. T. 1996 Formulation and validation of vertically 2-D shallow-water model. Proc. 25th Int. Conf. Coast. Engng., ASCE, 551-564.

[57] Karambas, Th. V. and Koutitas, C. 1992 A breaking waves propagation model based on the Boussinesq equations. Coastal Engng., 18, 1-19.

[58] Kawata, Y., Benson, B. C., Borrero, J. C., Borrero, J. L., Davies, H. L., DE Lange, W. P., Imamura, F., Letz, H., Nott, J., and Synolakis, C. E. 1999 Tsunami in Papua New Guinea was as Intense as First Thought, EOS. Trans. Am. Geophys. Union 80, 101, 104-105.

[59] Kim, J., Moin P., and Moser R. 1987 Turbulence statistics in fully developed channel flow at low Reynolds number. J. Fluid Mech., 177, 133-166.

[60] Kobayashi, N., Otta, A., and Roy, I. 1987 Wave reflection and run-up on porous slopes. J. Wtrwy., Port, Coast., and Ocean Engng., ASCE,113(3), 282-298.

[61] Kolmogorov, A. N. 1941 The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers. Dokl. Akad. Nauk SSSR 30, 299-303 [in Russian].

[62] Kothe, D. B., Rider W. J., Mosso, S. J., Brock, J. S., and Hochstein, J. I. 1996 Volume tracking of interfaces having surface tension in two and three dimensions. Technical Report AIAA 96-0859, AIAA 1996. Presented at the 34rd Aerospace Sciences Meeting and Exhibit.

[63] Kothe, D. B., Williams, M. W., Lam K. L., Korzewa, D. R., Tubesing, P. K., and Puckett E. G. 1999 A Second-order accurate, linearity-preserving volume tracking algorithm, for free surface flows on 3-D unstructured meshes, Proceedings of the 3rd ASME/JSME Joint Fluids Engineering Conference. FEDSM99-7109, July 18-22.

[64] Larsen, J. and Dancy, H. 1983 Open boundaries in short wave simulations-A new approach. Coast. Engng., 7, 285-297.

[65] Launder, B. E. and Spalding, D. B. 1974 The Numerical Computation of Turbulence Flows, Comp.. Meth. Appl. Mech. Engng., Vol. 3, pp. 269-289.

[66] Launder, B. E., Reece, G. T., and Rodi, W. 1975 Progress in development of a Reynolds stress turbulence closure. J. Fluid Mech., 68, 537-566.

[67] Lee, J.-J., Skjelbreia, E., and Raichlen, F. 1982 Measurement of velocities in solitary waves. J. of Wtrwy, Port, Coast., and Oc. Div., 108(WW2), 200-218.

[68] Lemos, C. M. 1992 Wave Breaking. Springer-Verlag.

[69] Leonard, A. 1974 Energy cascade in large eddy simulation of turbulent fluid flow. Adv. Feopgys. 18A, 237-248.

[70] Li, C.-W. and Lin, P. 2001 A numerical study of three-dimensional wave interaction with a square cylinder. Oc. Engng., 28, 1545-1555.

[71] Lilly, D. K. 1967 The representation of small-scale turbulence in numerical simulation experiments. In H. H. Goldstine (Ed.), Proc. IBM Scientific Computing Symposium on Environmental Sciences, IBM Form No. 320-1951, Yorktown Heights, New York, pp. 195-210.

[72] Lin, J.-C. and Rockwell, D. 1994 Instantaneous structure of a breaking wave. Phys. Fluids, 6(9), 2877-2879.

[73] Lin, J.-C. and Rockwell, D. 1995 Evolution of a quasi-steady breaking waves. J. Fluid Mech., 302, 29-44.

[74] Lin, J.-C. and Rockwell, D. 1996 Force identification by vortices fields: techniques based on flow imaging. J. Fluids Struct. 10, 663-668.

[75] Lin, P. 1998 Numerical Modeling of Breaking Waves. Phd Dissertation. Cornell University, Ithaca, N.Y.

[76] Lin, P. and Li, C. W. 2002 A σ -coordinate three-dimensional numerical model for surface wave propagation. Int. J. Numer. Meth. Fluids, 38:1048-1068.

[77] Lin, P. and Li, C. W. 2003 Wave-current interaction with a vertical square cylinder. Oc. Engng., 30, 855-876.

[78] Lin, P. and Liu, P. L.-F. 1998a A numerical study of breaking waves in the surf zone. J. Fluid Mech., 359, 239-264.

[79] Lin, P. and Liu, P. L.-F. 1998b Turbulence transport, vortices dynamics, and solute mixing under plunging breaking waves in surf zone. J. Geophys. Res., 103(C8), 15677-15694.

[80] Lin, P. and Liu, P. L.-F. 1999 Internal wave-maker for Navier-Stokes equations models. J. Wtrwy., Port, Coast., and Oc. Engng., 125 (4): 207-215.

[81] Liu, P. L.-F., Lynett, P. and Synolakis, C. E. 2003 Analytical solutions for forced long waves on a sloping beach. J. Fluid Mech., 478, 101-109.

[82] Longuet-Higgins, M. S. and Cokelet, E. D. 1976 The deformation of steep surface waves on water: I. A numerical method of computation. Proc. R. Soc. Lond. Ser. A, A 350, 1-26.

[83] Longuet-Higgins, M. S. and Cokelet, E. D. 1978 The deformation of steep surface waves on water: II. Growth of normal-mode instabilities. Proc. R. Soc.

Lond. Ser. A , 364, pp. 1-28.

[84] Losada, I. J., Silva, R., and Losada, M. A. 1996 3-D non-breaking regular wave interaction with submerged breakwaters. Coast. Engng., 28, 229-248.

[85] Lund, T. and Kaltenbach, H.-J. 1995 Experiments with explicit filtering for LES using a finite-difference model, Annual Research Brief, pp. 91-105. Center for Turbulence Research, NASA Ames-Stanford University.

[86] Lund, T. S. 1997 On the use of discrete filters for large eddy simulation. Annual Research Briefs, 83-95. Center of Turbulence Research, NASA Ames-Stanford Universuty.

[87] Massel, S. R. 1983 Harmonic generation by waves propagating over a submerged step. Coast. Engng., 7, 357-380.

[88] Mei, C. C. and Black, J. L. 1969 Scattering of surface waves by rectangular obstacles in waters of finite depth. J. Fluid Mech. 38, 499-511.

[89] Merci, B., Dick, E., Vierendeels, J., Roekaerts, D., and Peeters, T. W. J. 2001 A quasi-realizable cubic low-Reynolds eddy-viscosity turbulence model with a new dissipation rate equation. Combustion and Flame, 126 (1-2), 1533-1556

[90] Merci, B. and Dick, E. 2002 Predictive capabilities of an improved cubic k-epsilon model for inert steady flows. Flow turbulence and combustion, 68 (4), 335-358

[91] Miller, D. J. 1960 Giant waves in Lituya bay, U.S. Geol. Surv. Prof. Paper 354-C, 51-83.

[92] Mizuguchi, M. 1986 Experimental study on kinematics and dynamics of wave breaking. Proc. 20th Int. Conf. of Coast. Engng., ASCE, 589-603.

[93] Mohd-Yusof, J. 1997 Combined immersed boundaries/B-splines methods for simulations of flows in complex geometries, CTR Annual Research Briefs, NASA Ames/Stanford University, 1997.

[94] Mosso, S. J., Swartz, B. K., Kothe, D. B., and Ferrell, R. C. 1997 A parallel, volume-tracking algorithm for unstructured meshes. In P. Schiano, A. Ecer, J. Periaux, and N. Satofuka, editors, Parallel Computational Fluid Dynamics: Algorithms and Results Using Advanced Computers, 368-375, Capri, Italy, Elsevier Science.

[95] Murty, T. S. 1977 Seismic sea waves. Tsunamis, Bull. No. 198 of the Fisheries Research Board of Canada, Ottawa, 337 pp.

[96] Murty, T. S. 1979 Submarine slide-generated water-waves in Kitimat Inlet, British-Columbia, J. Geoph. Res, 84 (NC12), 7777-7779, 1979.

[97] Nadaoka, K. and Kondo, T. 1982 Laboratory measurements of velocity field structure in the surf zone by LDV. Coast. Engng. In Japan, 25, 125-146.

[98] Nadaoka, K., Hino, M., and Koyano, Y. 1989 Structure of the turbulent flow field under breaking waves in the surf zone. J. Fluid Mech., 204, 359-387.

[99] Nakagawa, T. 1983 On characteristics of the water-particle velocity in a plunging breaker. J. Fluid Mech., 126, 251-268.

[100] Nichols, B. D. and Hirt, C. W. 1971 Improved free surface boundary conditions for numerical incompressible-flow calculations, J. Comp. Phys., 8, 434-448.

[101] Orszag, S. A. and Patterson G. S. 1972 Numerical simulation of three-dimensional homogeneous isotropic turbulence. Phys. Rev. Lett., 28, 76-89.

[102] Pelinovsky, E. and Poplavsky, A. 1996 Simplified model of tsunami generation by submarine landslides, Phys. Chem. Earth., 21 (12) , 13-17.

[103] Peregrine, D. H. 1967 Long waves on a beach. J. Fluid Mech., 27(4), 815-827.

[104] Peregrine, D. H. 1983 Breaking waves on beaches. Ann. Rev. Fluid Mech., 15, 149-178.

[105] Peregrine, D. H., Cokelet, E. D., and McIver, O. 1980 The fluid mechanics of waves approaching breaking. Proc. 17th Int. Conf. Coastal Engng., ASCE,

512-528.

[106] Petit, H. A. H., Tonjes, P., van Gent, M. R. A., and ven den Bosch, P. 1994 Numerical simulation and validation of plunging breakers using a 2D Navier-Stokes model. Proc., 24th Int. Conf. on Coast. Engng., ASCE, Reston, Va., 511-524.

[107] Pope, S. B. 1975 A more general effective-viscosity hypothesis. J. Fluid Mech., 72, 331-340.

[108] Pope, S. B. 2001 Turbulent flows. Cambridge University Press.

[109] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. 1986 Numerical recipesin Fortran, Cambridge.

[110] Raney, D. C. and Butler, H. L. 1975 A numerical model for predicting the effects of landslide-generated water waves. U.S. Army Engineering Waterways Experiment Station Report H-75-1, U.S. Department of Commerce, Springfield, VA.

[111] Raichlen, F. and Synolakis, C.E. 2003 Run-up from three dimensional sliding mass, in Proceedings of the Long Wave Symposium 2003, (Briggs, M, Coutitas, Ch.) .XXX IAHR Congress Proceedings, ISBN-960-243-593-3, 247-256.

[112] Rey, V., Belzons, and M., Guazzelli, E. 1992 Propagation of surface gravity waves over a rectangular submerged bar. J. Fluid Mech. 235, 453-479.

[113] Rhee, S. H. and Stern, F. 2002 RANS Model for Spilling Breaking Waves. Journal of Fluids Engineering-Transactions of the ASME,124, 424-432.

[114] Rhie, C. M. and Chow, W. L. 1983 A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation. AIAA Journal, 21:1525-1532.

[115] Richtmyer, R. D. and Morton, K. W. 1967 Difference Methods for Initial Value Problems. Wiley-Interscience.

[116] Rider, W. J. and Kothe, D. B. 1998 Reconstructing Volume Tracking, J. Comp.

Phys., 141, 112-152.

[117] Rogallo, R. S. 1981 Numerical experiments in homogeneous turbulence. Technical Rep. TM81315, NASA.

[118] Rodi, W. 1980 Turbulence models and their application in hydraulics – A state-of-the-art review. I.A.H.R. publication.

[119] Rubinstein, R. and Barton, J. M. 1990 Nonlinear Reynolds stress models and the renormalization group, Phys. Fluids A, 2, 1472-1476.

[120] Saad, Y. 1996 Iterative Methods for Sparse Linear Systems. PWS Publishing Company.

[121] Saad, Y. and Schultz, M. H. 1986 GMRES: A generalized minimal residual algorithm for solving non-symetric linear systems. SIAM J. Sci. Stat. Comput., 7:856.

[122] Schaffer, H. A. 1994 Simultaneous generation and active absorbing of waves in flumed. Int. Symp. Waves – Physical and Numerical Modelling, Vancouver, Canada, pp. 90-99.

[123] Schaffer, H. A., Madsen, P. A., and Deigaard, R. 1993 A Boussinesq model for waves breaking in shallow water. Coast. Engng., 20, 185-202.

[124] Schumann, U. 1975 Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli. J. Comp. Phys. 18, 376-404.

[125] Seabra-Santos, F. J., Renouard, D. P., and Tempeerville, A. M. 1987 Numerical and experimental study of the transformation of a solitary wave over a shelf or isolated obstacle. J. Fluid Mech. 176, 117-134.

[126] Sheridan, J., Lin, J.-C., and Rockwell, D. 1997 Flow past a cylinder close to a free surface. J. Fluid Mech. 330, 1-30.

[127] Shih, T.-H., Zhu, J. and Lumley, J. L. 1993 A realizable Reynolds stress algebraic equation model, NASA TM 105993.

[128] Shih, T.-H., Zhu, J., and Lumley, J. L. 1996 Calculation of wall-bounded complex flows and free shear flows. International J. for Num. Meth. in Fluids, 23, 1133-1144.

[129] Skyner, D. 1996 A comparison of numerical predictions and experimental measurements of the internal kinematics of a deep water plunging wave. J. Fluid Mech., 315, 51-64.

[130] Smagorinsky, J. 1963 General circulation experiments with the primitive equations: I. The basic equations. Mon. Weather Rev. 91, 99-164.

[131] Sorensen, R. M. 1993 Basic wave mechanics. Wiley, New York.

[132] Speziale, C. G. 1987 On nonlinear k-l and k-epsilon models of turbulence. J. Fluid Mech. 178, 459-475.

[133] Strang, G. 1976 Linear Algebra and Its Applications. Harcourt Brace Jovanovich.

[134] Strang, G. 1986 Introduction to Applied Mathematics. Wellesley-Cambridge.

[135] Striem, H. L. and Miloh, T. 1975 Tsunamis induced by submarine slumping off the coast of Israel, Israel Atomic Energy Commission. 1975.

[136] Strikwerda, J. C. 1989 Finite Difference Scheme and Partial Differential Equations. Wadswork and Brooks/Cole.

[137] Stive, M. J. F. 1980 Velocity and pressure field of spilling breaker. Proc. 17th Int. Conf. Coast. Engng., ASCE, 547-566.

[138] Stive, M. J. F. and Wind, H. G. 1982 A study of radiation stress and set-up in the nearshore region. Coast. Engng., 6, 1-25.

[139] Synolakis, C. E., 2003 Tsunamis and seiches, in Earthquake Engineering Handbook, Chen, W-F and Scawthorn, C. (editors), CRC Press, 9-1 to 9-90.

[140] Synolakis, C. E., Bardet, J. P., Borrero, J. C., Davies, H., Okal, E. O., Silver, E. A., Sweet, and S., Tappin, D. R., 2002 Slump origin of the 1998 Papua New

Guinea Tsunami, Proc. Royal Society, London, Ser. A, 458, 763-789.

[141] Svendsen, I. A. 1984 Wave heights and set-up in a surf zone. Coast. Engng., 303-329.

[142] Svendsen, I. A. 1987 Analysis of surf zone turbulence. J. Geophys. Res., 92, 5115-5124.

[143] Tang, C. J. and Chang, J. H. 1998 Flow separation during solitary wave passing over submerged obstacle. J. Hydr. Engng. 124, 742-749.

[144] Tennekes, H. and Lumley, J. L. 1972 A first Course in Turbulence. Cambridge, MA: MIT Press.

[145] Townson, J. M. and Kaya, Y 1988 Simulations of the waves in Lake Botnen created by Rissa landslide, Proc. Inst. Civ. Engrs., 85(2), 145-160.

[146] Ting, F. C. K. and Kim, Y.-K. 1994 Vortex generation in water waves propagation over a submerged obstacle. Coast. Engng. 24, 23-49.

[147] Ting, F. C. K. and Kirby, J. T. 1994 Observation of undertow and turbulence in a laboratory surf zone. Coast. Engng., 24, 51-80.

[148] Ting, F. C. K. and Kirby, J. T. 1995 Dynamics of surf-zone turbulence in a strong plunging breaker. Coast. Engng., 24, 177-204.

[149] Ting, F. C. K. and Kirby, J. T. 1996 Dynamics of surf-zone turbulence in a spilling breaker. Coast. Engng., 27, 131-160.

[150] Townson, J. M. and Kaya, Y. 1988 Simulations of the waves in Lake Botnen created by Rissa landslide, Proc. Inst. Civ. Engrs., 85(2), 145-160.

[151] Troch, P. and De Rouck, J. 1998 Development of Two-Dimensional Numerical Wave Flume for Wave Interaction with Rubble Mound Breakwaters. Coast. Engng. Proceedings of Conference, Copenhagen, Denmark, June 22-26, pp. 1638-1649.

[152] Van Dongeren, A. R. and Svendsen, I. A. 1997 Absorbing-generating boundary

condition for shallow water models, J. Wtrwy. Port C-ASCE 123 (6) NOV-DEC: 303-313.

[153] Varga, R. 1962 Matrix Iterative Analysis. Prentice-Hall, Englewood Cliffs, NJ.

[154] Verzicco, R., Mothod-Yusof, J., and Orlandi, P. 2000 Large eddy simulation in complex geometric configurations using boundary body forces, AIAA Journal Vol. 38, 3, 427-433.

[155] Vinokur, M. 1989 An analysis of finite-difference and finite-volume formulations of conservation laws. J. Comp. Physics, 81:1-52.

[156] Watanabe, Y. and Saeki, H. 1999 Three-dimensional large eddy simulation of breaking waves, Coast. Engng., Vol. 41 Nos. 3and4 281-301.

[157] Watts, P. 1997 Water Waves Generated by Underwater Landslides, Ph.D. Thesis, California Institute of Technology, June, 1997.

[158] Watts, P. 1998 Wavemaker curves for tsunamis generated by underwater landslides, J. Wtrwy., Port, Coast., and Oc. Engrg., ASCE, 124 (3), 127-137.

[159] Watt, P., 2000 Tsunami Features of Solid Block Underwater Landslides, J. Wtrwy., Port, Coast., and Oc. Engrg., ASCE, 126, 144-152.

[160] Wei, G., and Kirby, J. T. 1995 Time-dependent numerical code for extended Boussinesq equations. J. Wtrwy., Port, Coast., and Oc. Engrg., ASCE, 121,251-161.

[161] Wiegel, R. L. 1960 A presentation of cnoidal wave theory for practical application. J. Fluid Mech., 7 (2), 273-286.

[162] Winckelmans, G. and Jeanmart, H 2001 Assessment of some models for LES without/with explicit filtering. In Direct and Large-Eddy Simulation IV (ed. B. Geurts, F. Friedrich and O. Metais), pp. 55-66. Kluwer.

[163] Yoshizawa, A. 1987 Statistical analysis of the derivation of the Reynolds stress from its eddy-viscosity representation, Phys. Fluids, 27, 1377-1387.

[164] Zelt, J. A. 1991 The run-up of nonbreaking and breaking solitary waves. Coast. Engng., 15, 205-246.

[165] Zhuang, F. and Lee, J. J. 1996 A viscous rotational model for wave overtopping over marine structure. Proc. 25th Int. Conf. Coastal Eng., ASCE, pp. 2178-2191.

[166] Zhou, Y., Brasseur, J. G., and Juneja, A. 2001 A resolvable subfilter-scale model specific to large-eddy simulation of under-resolved turbulence, Phys. Fluids 13, 2602-2610.

[167] Zienkiewicz, O. C. 1977 The Finite Element Method. McGraw-Hill, New York, NY.