

ABSTRACT

MARJANI, ALIASGHAR. Advancement in Numerical Method for Optimizing Ocean Wave Energy Conversion. (Under the supervision of Dr. Billy L. Edge).

The main purpose of this study is to enhance the Smoothed Particle Hydrodynamics (SPH) method that can more accurately simulate the hydrodynamic forces on a structure and can be used for determining efficient designs for wave energy devices. A desirable method is one that is accurate and easy to apply. Smoothed particle hydrodynamics is a method used in various fields of study. Unlike the finite difference method (FDM), SPH is a Lagrangian mesh-free method in which each particle moves according to the property of the surrounding flow and governing conservation equations, and carries the properties of water such as density, pressure and mass.

SPH is validated by several experimental data sets in 2D and 3D. Basic fluid mechanics demonstrates that the boundary layer generation and flow separation around a blunt object play key roles in wake generation behind the object. It is known that the boundary layer thickness depends inversely on the Reynolds number. Hence, using very small particles is necessary to simulate the ultra-thin boundary layer. On the other hand it is impossible to have very fine particles for a problem with large domain. Our new algorithms and techniques minimize the total number of required particles and consequently reduce computational cost when fine particles are employed locally and mirror boundary condition is applied for symmetric geometries.

© Copyright 2013 by Aliasghar Marjani

All Rights Reserved

Advancement in Smoothed Particle Hydrodynamics for
Applying to Ocean Wave Energy Conversion

by
Aliasghar Marjani

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Civil Engineering

Raleigh, North Carolina

2013

APPROVED BY:

Dr. Billy L. Edge
Co-chair of Advisory Committee

Dr. Margery F. Overton
Co-chair of Advisory Committee

Dr. Kumar Mahinthakumar

Dr. Tarek Echekki

BIOGRAPHY

Aliasghar Marjani was born on 21st of March 1983 in Damghan, Iran. He earned his bachelor in Civil Engineering at Amirkabir University of Technology (Tehran Polytechnic) in 2004. He continued his study with a two-year master's degree in Water Resources and Environmental Engineering in Sharif University of Technology working on 3D-Numerical Modeling of Hydrodynamics of Hyper Saline Lake Urmia. Then in 2008, he moved to the U.S. to pursue his PhD in Coastal and Ocean Engineering under supervision of Dr. Billy L. Edge. Improving the efficiency of a numerical model, Smoothed Particle Hydrodynamic, in computing wave force on coastal structures at high Reynolds number was the main part of his study.

ACKNOWLEDGMENTS

I would like to express my sincerest gratitude to my advisor **Dr. Billy L. Edge** for his unforgettable supervision, scientific hints, advice, and his continued support and encouragement. His invaluable advice toward becoming a capable professional in my field of study is one of my greatest assets that will carry throughout my life. My special acknowledgments to **Dr. Robert A. Darlymple** and his research group for their guidance, constructive suggestions and assistance.

I also thank **Dr. Margery L. Overton**, **Dr. Kumar Mahinthakumar**, and **Dr. Tarek Echekki** for participating in my doctoral committee. I thank all the professors in the School of Civil, Construction, and Environmental Engineering, the North Carolina State University, especially **Dr. Margery F. Overton** and **Dr. Vernon C. Matzen**.

Last and most, I express my deepest affection for my parents, **Tayyebeh** and **Reza**, to whom I dedicate this work, and for my sisters, **Mahdokht** and **Mahya**, who have always supported me. It has not been easy living so far away from them. But, their love and unwavering support has made my journey much easier.

TABLE OF CONTENTS

| | |
|---|------------|
| LIST OF FIGURES | vii |
| LIST OF TABLES | xii |
| CHAPTER 1 | 1 |
| 1 INTRODUCTION..... | 1 |
| 1.1 Background..... | 1 |
| 1.2 Study Objectives | 8 |
| CHAPTER 2 | 14 |
| 2 NUMERICAL SIMULATION | 14 |
| 2.1 Role of Numerical Simulation..... | 14 |
| 2.2 Procedure of Numerical Simulations..... | 15 |
| 2.3 Meshless Particle Methods..... | 17 |
| 2.4 Smoothed Particle Hydrodynamics (SPH) | 20 |
| 2.5 Basics of SPH | 30 |
| 2.6 Governing Equation..... | 34 |
| 2.7 Standard SPH formulation | 36 |
| 2.8 Boundary Condition | 61 |
| 2.9 Nearest Neighboring Particle searching (NNPS)..... | 73 |
| 2.10 Floating Objects | 77 |
| 2.11 Computing Force on an Object | 79 |
| 2.12 Variable- <i>h</i> | 79 |

| | | |
|------------------|--|------------|
| 2.13 | Dynamic Load Balancing (Processors border) | 85 |
| CHAPTER 3 | | 87 |
| 3 | VALIDATION | 87 |
| 3.1 | Test 1: Dam Breaking | 87 |
| 3.2 | Test 2: Dam Breaking Wave Impact on a Small Structure (in 3D)..... | 99 |
| 3.3 | Regular Wave Propagation in 2D | 104 |
| 3.4 | Test Case 4: Force on a Flat Rectangular Plate | 114 |
| 3.5 | Test Case 5: Tsunami Wave Impact on a Single Circular Cylinder | 119 |
| 3.6 | Test 6: Flow Past a Cylinder in 2D | 135 |
| CHAPTER 4 | | 142 |
| 4 | ENHANCEMENT IN THE EFFICIENCY OF SPH IN 3D MODELING | 142 |
| 4.1 | Introduction | 142 |
| 4.2 | Variable Underlying Cells | 144 |
| 4.3 | Dividing Wall | 157 |
| 4.4 | Mirror Boundary Condition | 164 |
| 4.5 | Combination of Methods | 189 |
| CHAPTER 5 | | 191 |
| 5 | CONCLUDING REMARKS AND FUTURE WORK | 191 |
| 5.1 | Summary..... | 191 |
| 5.2 | Conclusions | 194 |
| 5.3 | Future work | 197 |

| | |
|---|------------|
| REFERENCES..... | 198 |
| APPENDICES | 208 |
| APPENDIX A (LOAD BALANCING ALGORITHM)..... | 209 |
| APPENDIX B (PISTON WAVEMAKER FORMULATION FOR SOLITARY WAVE)..... | 214 |
| APPENDIX C (SOLITARY WAVE: EXPERIMENT DATA CONVERSION).. | 218 |

LIST OF FIGURES

| | |
|---|----|
| Figure 1-1 – PowerBuoy, 40 kiloWatt WEC (http://www.oceanpowertechnologies.com)..... | 4 |
| Figure 1-2- Pelamis Convertor | 5 |
| Figure 1-3- Oyster, a surge converter..... | 5 |
| Figure 2-1- Official homepage of SPHysics code | 28 |
| Figure 2-2- Particle approximations and influence domain of particle i | 32 |
| Figure 2-3- Density field for a dam break test comparing the effect of density filter. (a) without any filter, (b) with Shepard filter, (c) with MLS filter, and (d) with diffusive term..... | 41 |
| Figure 2-4- Time Variation of the system energy during the collapse of water column | 47 |
| Figure 2-5- Quadratic kernel function and its derivative [Crespo, 2008]..... | 52 |
| Figure 2-6- Cubic spline kernel function and its first derivative [Crespo, 2008]..... | 53 |
| Figure 2-7- Wendland kernel function and its first derivative [Crespo, 2008]..... | 54 |
| Figure 2-8- Monaghan-Kos repulsive force formulation | 63 |
| Figure 2-9- Dynamic boundary particle arrangement | 67 |
| Figure 2-10- Ghost particle boundary condition (a) free-slip condition, (b) no-slip condition..... | 69 |
| Figure 2-11- fixed ghost particle, interpolation points and body nodes | 70 |
| Figure 2-12- Periodic lateral boundary (plan view) The kernel support of particle i is continued through periodic boundary | 72 |
| Figure 2-13- All-pair search algorithm for NNPS in 2D..... | 74 |
| Figure 2-14- Linked-list algorithm of NNPS | 75 |
| Figure 2-15- Initial particle configuration of two-dimensional simulation of wedge water entries [Oger et al., 2006] | 83 |
| Figure 2-16- Omidvar et al. (2011) replaced a heavy particle with a mass m with four lighter ones with a mass of $m/4$ (in 2D)..... | 84 |
| Figure 2-17- Dynamic load balancing for a beach problem. Dashed lines present the border of processors | 85 |
| Figure 3-1- Experimental geometry: Side view (upper sketch), plan view (lower sketch) [Gomez-Gesteira and Dalrymple, 2004] | 89 |

| | |
|--|-----|
| Figure 3-2-Comparison between experimental data (dots) and first-attempt SPH simulation ($dp=0.0225$ m and using Dynamic particle BC) (solid line) by author. The horizontal velocity at the gauge installed in front of the obstacle (upper plot) and the net force on the obstacle (lower plot) | 91 |
| Figure 3-3- Comparison between experimental data (dots) and second-attempt numerical simulation ($dp=0.01$ m and using Dynamic particle BC) (solid line). The horizontal velocity at the gauge installed in front of the obstacle (upper plot) and the net force on the obstacle (lower plot) | 92 |
| Figure 3-4- Velocity time series in the flow direction at interested point for selected particle size [Rooney et al., 2011]..... | 93 |
| Figure 3-5- Comparison between experimental data (dots) and third-attempt SPH simulation ($dp=0.0225$ m and using Repulsive force BC) (solid line)..... | 96 |
| Figure 3-6- Total force on the circular cylinder | 99 |
| Figure 3-7- Geometry of the 3D water collapse experiment with measurement positions (unit: meter). Zoom of the box is shown on the right [Lee et al., 2010]..... | 100 |
| Figure 3-8- Fluid pressure is recorded at points P1 to P8 on the surface of the Obstruction | 100 |
| Figure 3-9- In-detailed description of system. Top picture: plan view, Bottom picture: side view ... | 101 |
| Figure 3-10- Water height time history of the 3D dam-break flow at (a): H2 and (b): H4 | 102 |
| Figure 3-11- Pressure time history of the 3D dam break flow at P1 (top) and P3 (bottom) | 103 |
| Figure 3-12- Initial setting of regular wave propagation on a simple beach | 104 |
| Figure 3-13- Wave height (meter) at three locations x_1 , x_2 , and x_3 . (a) with repulsive force BC and (b) with dynamic boundary particles | 105 |
| Figure 3-14- Water surface elevation at (a) $x_1= 5.1$ m, (b) $x_2=14.4$ m, and (c) $x_3= 17$ m | 107 |
| Figure 3-15- Horizontal particle velocity at (a) $(x_1,z_1) = (5.1$ m, 0.1m above the bottom), (b) $(x_2,z_2) = (14.4$ m, 0.1 m above the bottom), and (c) $(x_3,z_3) = (17$ m, 0.3 m above the bottom) | 108 |
| Figure 3-16- Vertical particle velocity at (a) $(x_1,z_1) = (5.1$ m, 0.1m above the bottom), (b) $(x_2,z_2) = (14.4$ m, 0.1 m above the bottom), and (c) $(x_3,z_3) = (17$ m, 0.3 m above the bottom) | 109 |
| Figure 3-17- Wave runup on the sloped beach by SPH | 111 |
| Figure 3-18- The distance between the bottom of cylinder and SWL, h_{b-SWL} , and wave height of two tests. left sketch: Case #1 and right sketch: Case #2 [photo from Prasad (1994)]. | 112 |

| | |
|--|-----|
| Figure 3-19- Vertical force on the pipe by SPH (solid line) and Experiment (red squares), (a) Case #1 and (b) Case #2 | 113 |
| Figure 3-20 - Fluid particles stays on the cylinder for a longer time than do in reality | 114 |
| Figure 3-21- Plan view of wave tank and the position of rectangular barge..... | 115 |
| Figure 3-22- Numerical result of water surface elevation 1 m downstream of the wavemaker..... | 116 |
| Figure 3-23- Horizontal force on the barge in x and y direction by SPH..... | 116 |
| Figure 3-24- Plan view of single cylinder and all measuring instruments positions | 121 |
| Figure 3-25- A snapshot from the SPHysics output at t=4.8 s | 122 |
| Figure 3-26- Experimental (dot line) and Numerical (solid line) wave amplitude versus time at gauges 4,5,6,7,8,9. Dynamic boundary particle is applied here. | 124 |
| Figure 3-27- Experimental (dot line) and Numerical (Solid line) wave amplitude versus time at gage 5, 6, 7 when repulsive force is applied..... | 128 |
| Figure 3-28- Experimental (dot line) and Numerical (solid line) horizontal velocity versus time ant ADV1 to ADV5 | 129 |
| Figure 3-29- Force on the cylinder versus time: experimental (dot line), SPH with repulsive force BC (Solid line) and SPH with Dalrymple BC (dashed line) | 132 |
| Figure 3-30- flow separation and wake generation as flow past a cylinder at different Reynolds number [Photo from Behara and Mittal (2011)] | 134 |
| Figure 3-31- Flow past a single cylinder within a periodic lattice [photo from Morris et al. (1997)] | 136 |
| Figure 3-32- Comparison of present study (solid line) and Morris (dashed line) velocity profile along paths 1 and 2 for Re=1 | 137 |
| Figure 3-33- Comparison of present study (solid line) and Morris (dashed line) velocity profile along paths 1 and 2 for Re=0.03 | 138 |
| Figure 3-34- Velocity field and separation angle at $Re=20$ for a circular cylinder in a steady flow (present study)..... | 139 |
| Figure 3-35- Velocity magnitude (m/s) as predicted by SPH in 2D at Re=100 | 141 |
| Figure 4-1- Variable mass particles with spatially uniform kernel length, h. The number of finer particles within the support circle is about 3.5 times greater than that of coarser ones..... | 145 |

| | |
|---|-----|
| Figure 4-2- The total number of particles within the support domain is in the same order everywhere when each particles has its own smoothing length | 146 |
| Figure 4-3- uni-size underlying cells is not efficient when variable- h is used | 149 |
| Figure 4-4- Many particles should be transferred to the adjacent processors, if the size of underlying cells is much greater than the local h | 150 |
| Figure 4-5- (a) fixed-size underlying cells vs. (b) variable-size underlying cells | 151 |
| Figure 4-6- Dynamic load balancing process and border shifting with fixed-size cell | 153 |
| Figure 4-7- Dynamic load balancing process and border shifting with variable-size cell and employing a marker as processors border..... | 156 |
| Figure 4-8- completely different size of particle are located at two sides of the moving wall..... | 158 |
| Figure 4-9- Two types of moving walls, a. type1: one degree freedom and b. type2: two degrees freedom | 159 |
| Figure 4-10- Initial particle arrangement with variable h and variable cell size. Colors represent the particle mass | 161 |
| Figure 4-11- Total force on the cylinder by variable- h , constant- h and experiment | 161 |
| Figure 4-12- Experimental (dots) and Numerical (solid line) wave height at the measuring points.. | 162 |
| Figure 4-13- Experimental (dots) and Numerical (solid line) particle velocity in x -direction at the measuring points | 163 |
| Figure 4-14- Periodic boundary condition is not suitable for modeling the half-basin | 165 |
| Figure 4-15- Two divided parts are congruent with respect to symmetry line in a symmetric problem | 166 |
| Figure 4-16- Particles in the first underlying cell are mirrored with respect to the symmetry line... | 168 |
| Figure 4-17- Initial particle configuration of half-basin dam breaking test by mirror boundary condition | 169 |
| Figure 4-18- Time history of dam breaking, left panel: half-basin, right panel: full basin | 170 |
| Figure 4-19- Comparison between half-size model and full-size model and experiment for $dp=0.0225$ m, (a) Flow velocity in front of the structure, (b) Total force on the obstacle..... | 173 |
| Figure 4-20- Comparison between half-size model and full-size model and experiment for $dp=0.01$ m, (a) Flow velocity in front of the structure, (b) Total force on the obstacle..... | 174 |

| | |
|---|-----|
| Figure 4-21- Initial particle configuration for half-basin solitary wave impacting a circular cylinder | 175 |
| Figure 4-22- Time history of solitary wave impact on a cylinder by half-basin technique | 176 |
| Figure 4-23- Total force on the single cylinder by experiment (dots), SPH: full-basin (dashed lone) and SPH: half-basin (solid line with star) | 177 |
| Figure 4-24- Water surface elevation at 6 wave gauges by experiment (dots) and SPH: half-basin (solid line) | 178 |
| Figure 4-25- particle velocity in the direction of flow at 6 ADVs by experiment (dots) and SPH: half-basin (solid line) | 179 |
| Figure 4-26- Total horizontal force on the single cylinder under different test settings, (a) $d=0.6$ m, $H=0.24$ m, (b) $d=0.6$ m, $H=0.36$ m, (c) $d=0.75$ m, $H=0.30$ m | 181 |
| Figure 4-27- Total horizontal force on the single cylinder when $d=0.75$ m, $H=0.30$ m | 184 |
| Figure 4-28- Total horizontal force on the single cylinder, SPH output (solid line), Experimental data converted by Zhang (dots), Experimental data converted by Marjani (triangles). (a) when $d=0.45$ m, $H=0.40$, (b) when $d=0.60$ m, $H=0.24$, (c) when $d=0.60$ m, $H=0.36$, and (d) when $d=0.75$ m, $H=0.30$ | 186 |
| Figure 4-29- Initial particle configuration of combined model. Warm colors represent light particles and cold colors represent the coarse ones. | 190 |
| Figure 4-30- Total horizontal force on the cylinder. Dashed line: full-basin; dots: half-basin and solid line: half-basin & variable-h | 190 |

LIST OF TABLES

| | |
|---|-----|
| Table 3-1- The wave condition: wave period (T) and wave height (H) and the distance between the bottom of pipe and the still water level..... | 112 |
| Table 3-2- A cylinder in steady flow, Reynolds number (Re), velocity (V_0) and Speed of sound (c) | 136 |
| Table 3-3- Numerical and experimental values of the separation angle (θ), normalized length of the wake (L/D) and drag coefficient (C_D) for $Re=20$ | 140 |
| Table 4-1- Numerical settings for various tests of solitary wave impact on a single cylinder | 180 |
| Table 4-2- Estimated drag coefficient C_d and inertia coefficient C_m for the impact of plane solitary wave on a single cylinder by Zhang (2009)..... | 183 |
| Table 4-3- The horizontal force on the half cylinder in the normal direction to the wave direction. | 185 |

Chapter 1

1 Introduction

1.1 Background

About 93 percent of today world's energy utilization relies on fossil energy sources such as coal and oil. Burning of these nonrenewable fuels produces hazardous pollutant which can result in environmental damages. Renewable energy sources, such as wind, solar, and wave have been considered as possible alternatives to fossil fuels [Hildenbrand and Hirota, 2010]. Due to the recent climate changes, renewable energy sources play an important role to meet the world's future energy demands.

The oceans contain huge amounts of untapped energy. The ocean has been considered one of the most reliable and renewable energy sources. Most people know well about hydroelectric dam, a gravity-based source of energy, while the ocean energy as an exploitable wave-based source of energy is not yet well-known [Vining, 2007]. Oceans offer their abundant and promising energy in various forms such as tidal currents, marine currents,

thermal currents and waves [Vining, 2007, CEC¹]. Ocean currents and waves have mostly been studied as the most commercially feasible energy resources. Solar energy produces wind and wind blows over the surface of ocean and waves are created [Vining, 2007]. These waves can travel hundreds of miles with little loss of energy. Waves lose their energy near the shoreline because of bottom friction and breaking [Vining, 2007].

Under the Marine Renewable Energy Research Development Act, the U.S. has committed \$200 million toward novel wave-energy technology from 2008 to 2012. In addition, the U.S. Department of Energy has provided \$55 million during past three years for the research, development and application of ocean energy [Hildenbrand and Hirota, 2010]. The Electric Power Research Institute states that the U.S. coasts contain about 2,100 TWh/yr (Terra Watt hour per year). If this amount of energy can be caught, that would be enough for the whole U.S. demand. First of all, it's impossible to harness all the available energy, and second, the energy is not distributed evenly. If merely 15% of wave power can be extracted and the net output was reduced by 81% due to plant availability and power train efficiency, the delivered power would be about 4.6 KW/m for a typical wave period of 10 seconds and wave height of 3 meters.

Although the wave energy harnessing was first patented in 1977 and some preliminary applications for wave energy were constructed in France about 1910, wave energy began to be commercialized in the 1990s. Wave Energy Convertors (WECs) were invented to capture the mechanical ocean energy and convert it to other kinds of energy such as electrical power.

¹ The California Energy Commission: <http://www.energy.ca.gov/oceanenergy/index.html>

WECs can be grouped into four general categories: surge converters, oscillating water columns, overtopping devices, and floating devices [Vining, 2007, Coastal Wiki²]. Oscillating water surface caused by wave oscillation produces mechanical motion in floating devices. Floating devices can be divided into two groups, point and linear absorbers. Point absorbers are buoy-type WECs which catch the wave energy from all directions. As waves migrate, the buoy part moves up and down and this motion is converted to power by mechanical or hydraulic connections to a linear turbine or through electromagnetic induction. These devices are located offshore, at or near the water surface. Figure 1-1 shows a utility scale WEC floating device that is being deployed for testing in Spain, and New Jersey.

² Coastal Wiki: http://www.coastalwiki.org/coastalwiki/Wave_energy_converters

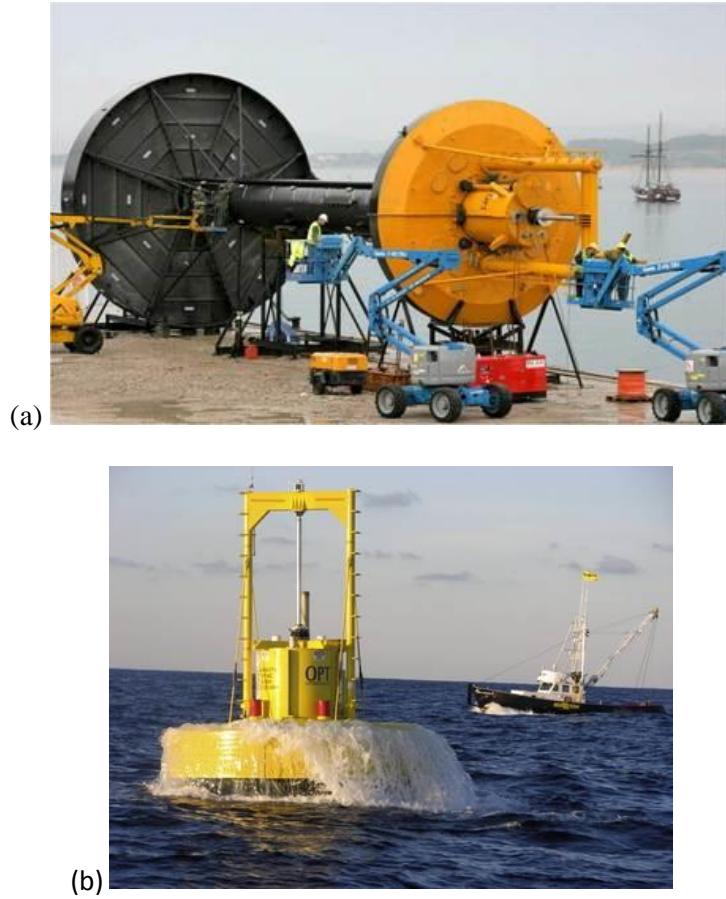


Figure 1-1 – PowerBuoy, 40 kiloWatt WEC (<http://www.oceanpowertechnologies.com>)

Linear absorber (or wave attenuator) is a long and multi-segmented floating device. This anchored device is positioned perpendicularly to incoming waves and articulated motion of this device drives a generator through hydraulic connection. Pelamis, shown in Figure 1-2, is unique among wave energy convertors [Vining, 2007]. This WEC lies on the water surface and perpendicular to the wave direction.



Figure 1-2- Pelamis Convertor

Figure 1-3 presents Oyster as a surge converter. This type of device catches wave energy directly from the surging motion of waves. The oscillation between a flapping membrane and a fixed point converts the wave energy into usable mechanical and electrical energy.

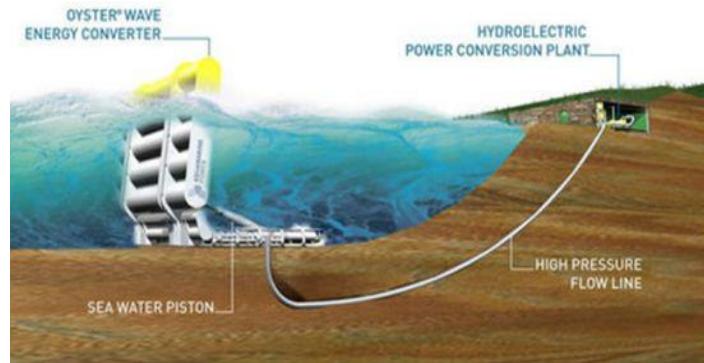


Figure 1-3- Oyster, a surge converter

All above WECs can be considered as fixed or moving marine structures. Under severe water-structure interactions, the evaluation of global and local force and momentums is necessarily important. The design of marine structures and their safety in operation relies on

the prediction of local and global loads on them. Specifically under extreme sea conditions, it is indeed a fundamental concern for ocean/coastal structures. Several types of impact flows are involved in those situations, e.g. slamming, sloshing, flooding and water on deck flows [Marrone et al., 2011]. The fluid-structure interaction causes wave force, diffracted and reflected waves [Kim et al., 2007]. A circular pier on pile is a common structure in ocean/coastal regions. One of the famous problems in fluid mechanics area is a flow past a cylinder which exists in a couple of wave energy converters, such as PowerBuoy. The generation of reflected and diffracted waves disturbs the incident wave as it impinges on the cylinder, generally called diffraction. The pressure field around the cylinder changes by the process of diffraction, thus the total force and momentum on the body will be influenced [Kim et al., 2007]. Therefore, for the case of multiple WECs the flow field is fully complex due to the motion of each device.

At the moment, there is no broad understanding and knowledge of the behavior of WECs under extreme ocean condition, so much work is still on going to improve their potential performance. Simulating the offshore flow field with different arrangement of fixed or floating devices, which involves complex interactions of physical processes, is not an easy task to do in laboratories. Besides, making severe wave condition may not be possible in laboratories and sea trials are unavoidable in such cases. As discussed before, the impacts are characterized by strong dynamics and large deformation and analytical solutions are only available for simple geometrical and flow field situations. On the other hand, interpreting the experimental observation is often difficult due to the mixture of various physical effects. In this context, numerical simulations can help analysis [Marrone et al., 2011].

Computational Fluid Dynamics (CFD) is a well-developed science these days and has many successes and applications in several areas of engineering. Numerical modeling is another possible approach for studying the heaving wave energy devices or any other type of marine structures and understanding their behavior in real ocean condition by solving momentum, mass and energy conservation equations. Two different general approaches are typically used in CFD, Eulerian and Lagrangian. In Eulerian form, the most frequently used method, any properties of fluid are computed through the time at a spatially fixed point. Thanks to over fifty-years of study of this method, this description is fully understood. Most of commercial software/codes employ Eulerian approach via finite element, finite difference and finite volume approaches. However, Eulerian method has weakness in simulating complicated problems with large fragmentation and complex free-surface flow. Since explicit treatment of the free-surface is required, Eulerian approaches are less well suited to simulate free-surface flow.

On the contrary, in the Lagrangian method, one can follow the history of an individual fluid parameter through the time and any flow variable is expressed at particle place through the time. Particle and meshless methods, which employ Lagrangian approach, are used in complex problems as described above.

Smoothed Particle Hydrodynamics (SPH), a fully Lagrangian and particle technique, was initially developed by Lucy (1997) and Gingold and Monaghan (1977) in astrophysics to study the collision of galaxies and the impact of bolides on planets. In this method, each property of fluid (such as density, velocity, internal energy, etc.) carried by particles are computed by continuity and conservation of momentum and energy equation at each time

step. Thanks to its Lagrangian nature, SPH is well suited to free-surface flow problems with large deformation and distorted surfaces. Some of SPH application is to study gravity currents (Monaghan, 1996; Monaghan et al., 1999), wave propagation (Monaghan, 1994; Monaghan and Kos, 1999, 2000), wave breaking (Dalrymple and Rogers, 2006; Khayyer et al., 2008), multi-phase flow (Monaghan and Kocharyan, 1995), simulation of the impact of solids (Benz and Asphaug, 1994, 1995) and wave impact on a structure (Gomez-Gesteira and Dalrymple, 2004; Lee et al., 2010). Recently, SPH was widely used for wave impact studies of offshore structures (Fontaine, 2002; Dalrymple et al., 2002; Gomez-Gesteira et al., 2012). The major advantages of SPH, arising directly from its Lagrangian nature, tackle complications related with lack of symmetry, possible large voids and fragmentation in the field, and no special treatment for free-surface in order to simulate highly nonlinear violent flows. Moreover, relative simplicity and accuracy in comparison with other CFD methods are the main preferences of using SPH (Monaghan, 2005a). A detailed description of SPH equation will be expressed in chapter 2.

1.2 Study Objectives

The main purpose of this study is to develop and enhance a hydrodynamic model that can accurately simulate the hydrodynamic forces on a structure and can be used for determining efficient designs for wave energy devices. Smoothed Particle Hydrodynamics (SPH) is a Lagrangian method to approximate numerical solution of fluid dynamics. SPHysics is an

existing open source FORTRAN code which uses SPH method to model fluid mechanics problems. Modeling boundaries is one of the most controversial topics in the SPH method.

Stationary or free-moving objects in the ocean continuously encounter dynamic forces from waves, pressure and currents. Fluid motion will exert forces on all boundaries, specifically in fully dynamic situations.

A couple of examples will be presented in chapter 3. All examples prove that SPH is a reliable approach in estimating kinematics and dynamics of a problem if the resolution is high (or particle size is small) enough. Two dam break tests show that SPH is reliable in computing velocity, water surface and the wave force on a structure. In addition SPH estimates water surface elevation and particle velocity in the wave field both in 2D and 3D precisely. Defining small particles, simulations show that SPH computes the wave force on a flat plate accurately. It is well known that a three-dimensional simulation will be very expensive if the basin and object are very large.

Generally, a flat plate is known as a streamlined object in which the friction force due to surrounding fluid motion is dominant. For blunt bodies, on the contrary, the form drag force is much more important than friction force. When a bluff body such as a vertical or horizontal circular or square cylinder is located in moving water, the total force is dictated by pressure drag (or form drag). At high Reynolds numbers, a wake is generated on the downstream side of a submerged body. As fluid flow past an object, the fluid pressure increases at upstream side and drops at the wake area. The form drag arises from the pressure gradient between two sides of an object. The size of wake plays a key role in the magnitude of pressure gradient. Small wake reduces the pressure gradient and inversely, large wake

increases it. The flow pattern around the object, the boundary layer and the separation point or the angle of separation affects the shape and size of wake behind the object. Later, it will be discussed that the numerical resolution (particle size) should be small enough to model the boundary layer correctly.

A couple of experiments in two dimensions were carried out to study the flow behavior around a circular or square cylinder at low Reynolds number. Morris et al. (1997) compared the velocity and pressure field with a Finite Element Method (FEM) solution and found a good agreement between SPH and FEM results. The Reynolds number in their study is in the range of 0.01-1.0. Takeda et al. (1994) studied higher Reynolds number in 2D. Isothermal viscous flows around a cylinder were simulated using SPH method for $Re=6, 10, 20, 30, 55$ with initial particle numbers about 20,000 and 46,000. The results were compared with mirror-symmetric isothermal flows obtained from calculation by the FDM. The agreement in drag coefficient is very good and both experiment and SPH gives the same separation angle and length of circulation. They believe that the possible Reynolds number increases proportional to particle number N for two dimensional flow and to $N^{2/3}$ for three dimensional flow. As an example, flow with $Re \sim 1000$ to 5000 is possible to simulate if calculations are done with 100 times the particles used in their study. Therefore, for higher Reynolds number 2-5 million particles are required. In addition to finer resolution, a proper turbulent model is necessary for complex motion around an object. Marrone et al. (2012) focused on higher Reynolds number about $Re=100$ to $Re=1000$. They investigated the wake properties and vortex street generation behind the cylinder in a 2D steady current. They compared the separation angle, wake length and drag coefficient with experimental results of Conteanceau

and Bouard (1977) for $\text{Re} = 10, 20$ and 40 . They found a very good agreement between SPH outputs and experimental observations. For an appropriate choice of particle size they employed the following criteria

$$\text{Re}_{dp} = \frac{U dp}{\nu} \quad (1-1)$$

in which U is inflow current speed and ν is fluid kinematic viscosity and Re_{dp} should be of order $O(1)$. Based on above criteria, 1.2 million particles were defined for the test of $\text{Re}=100$ -200. Definitely for higher Reynolds number, $\text{Re}=1000$, more particles are necessary.

Besides, the boundary layer thickness around a cylinder (or any object) depends on the square root of Reynolds number inversely. As Reynolds number increases, the thickness of boundary layer decreases. Therefore, to simulate the boundary layer correctly, the particle size should be comparable with its thickness and consequently finer size is required.

In real world, specially, for ocean waves, Reynolds numbers are in the range of $O(10^4$ - $10^5)$. Regarding above statement about boundary layer thickness, enough-fine particles are essential for practical applications. It will be shown that to simulate a flow past a cylinder in three dimensions, hundreds of millions particles are unavoidable which is almost impossible to do with current technology. Therefore, the general scope of this study is to improve SPH in such a way that large 3D tests at high Reynolds number can be simulated. This enhancement has two aspects, one to enable SPH to model these types of problems and the other is to speed up the calculation. Both reduce the computational cost of SPH algorithms.

One approach is particle refinement. As coarser particles approach the interest area they are divided into smaller particles and increases resolution locally. This method is very hard to

implement in SPH. The other approach is to define finer particles around interest point and coarser ones far from critical points.

Oger et al. (2006) used very small particle at the interest area and then gradually increase the particle size till reach a specific value. This approach avoids some extra treatments and corrections due to variable particle size (will be discussed more in detail in chapter 2). Although this approach decreases the total number of particles, it is slow in most cases, specifically, for the 3D cases in which the difference between minimum and maximum size is noticeable. Inventing a new algorithm to treat this drawback is one of the objectives of this study. The parallel version of SPH loses its efficiency when at least one core includes more than 10^5 particles. The key point to make the parallel version more efficient and useful is to ensure that the total number of particles is almost same for all processors. According to our experience, when the difference in size of the finest and coarsest particle increases, a couple of processors contain more than 10^5 particles that is not desirable.

Most of the problems we are dealing with are geometrically symmetric. That means half-basin problem can be simulated instead of the whole domain. Mirror-type boundary is a new technique added to SPH in this research to model the symmetric geometries. Chapter 4 discusses this type of boundary in detail and proves that the half-basin model is able to generate the output of full-basin model. This innovative technique can give a speedup of about 2.5.

The resulting numerical model can be applied specifically to simulate WECs operation. Many WECs have been invented in recent years, many of which have been tested in the laboratory to simulate ocean effects prior to testing in the ocean. It is always economical to

build test scale models for physical testing at laboratory scale. Using a numerical tool will greatly facilitate designing of devices prior to laboratory testing. This enables applying forces that may be impossible in laboratory settings. Simulating these devices and exploring their efficiency numerically is one of the topics of study in the coastal area.

Chapter 2

2 Numerical Simulation

2.1 Role of Numerical Simulation

Understanding and building a physical or engineering system requires a high-level process of modeling, simulations, analysis, design, testing prototypes, fabrication and manufacturing. Numerical simulation has become a common approach for solving these types of complex practical problems in engineering and science. Numerical simulator translates important aspects of physical phenomena into a discrete form of mathematical description, solves the equation on a computer, visualize the problem regarding the requirements of the scientists. Thanks to the increasing computer power these days, modern numerical methods solve the problem all in detail without making too many assumptions as it was done when a problem was solved traditionally by theoretical approach. Numerical simulations introduce an alternative tool, instead of expensive, dangerous or even time-consuming experiments in large laboratories or on site. Numerical simulations can be repeated easily to provide insightful and complete information that cannot be directly

observed via other methods such as in laboratory experiments. It acts also as a bridge between theory and experimental observation by giving a validation for theories or assisting in discovery of new phenomena. [Liu and Liu, 2007]

2.2 Procedure of Numerical Simulations

With some possible assumptions and simplifications, any physical problem can be interpreted to mathematical models. These mathematical models consist of governing equations, proper boundary conditions (BC) and initial condition (IC). To numerically solve the governing equation, the whole domain should be divided into discrete components. Domain discretization represents a continuum problem domain with a finite number of components which makes the main frame of computational solver. Traditionally, this frame is a set of grid or mesh consisting of some nodes where the field variables are evaluated. After all, to simulate a problem numerically, the decomposed domain and numerical algorithm is translated into a computer code in a programming language. In this process, the accuracy and efficiency (speed and hardware) are two important programmer's concerns [Liu and Liu, 2007].

In the area of fluid mechanics, the governing equations are established from the conservation laws, which tells that some specific field variables, such as mass, momentum and energy must be conserved during a physical process. Most of the time, these governing equations are partial or ordinary differential equations which cannot be solved readily

analytically. Computational fluid dynamics (CFD) deals with the techniques of solving approximately these differential equations. [Liu and Liu, 2007]

2.2.1 Grid-based methods

Lagrangian and Eulerian description are two fundamental approaches for describing the physical governing equations. In the Eulerian grid-based method typically represented by well-known and widely used FEM (Finite Element Method), grid is always attached to the material during the simulation. Thus, it moves with the material. Therefore, relative movement of connecting nodes results in compression, expansion or any general deformation of a mesh element. One of the advantages of this method is to easily track a moving boundary and interface. This method is very popular and successful where the deformation is not large as that in fluid flows. However, Lagrangian method is not easy to apply for the case with extremely distorted mesh. [Liu and Liu, 2007 and Crespo, 2008]

FVM is typically represents the Eulerian description. Contrary to the Lagrangian grid, in Eulerian approach, the grid or mesh is fixed on the space, but not to the material. Thus, all grid nodes and mesh cells stay fixed in space and do not move or are not deformed with time while the simulated object is flowing across the mesh. The nature of this method does not let us to analyze the time history of field variables at a fixed point on the material. Also, it is not very easy to simulate the irregular or complicated geometry in this method. Besides, accurately determination of the free surface position, deformable boundaries and material interfaces are very difficult. One of the limitations of the grid-based method is that constructing a regular grid for complex geometry is not a straightforward task and it usually

needs some additional complicated mathematical transformation that can be even more expensive than the original problem [Liu and Liu, 2007]. In addition, Eulerian description is not proper if the main goal of the simulation is a set of discrete physical particles rather than a continuum, such as interactions of stars, dynamic movement of molecules, the behavior of millions atoms in an equilibrium or non-equilibrium situation.

2.3 Meshless Particle Methods

Regarding mentioned difficulties and disadvantages of grid-based methods, the development of the next generation of computational methods (Meshfree methods) is the recent strong interest of many scientists. The main idea of meshfree methods is to develop an accurate and robust numerical solution for governing equation without defining any mesh. Since the meshfree methods consists of a set of arbitrarily distributed particles rather than a pre-defined grids, these methods are more likely to be used for the problems that are complicated for traditional grid-based ones. Meshfree methods are widely used for solid mechanics as well as fluid dynamics [Liu and Liu, 2007 and Crespo, 2008]. Generally, meshfree particle methods (MPMs) are referred to the category of meshfree methods that use particles to represent the state of whole system. In the field of fluid dynamics, each particle carries the properties of system such as mass, velocity, pressure, density, energy, shape of domain (position of each particle), etc. Typically, most of MPMs are inherently Lagrangian methods. MPMs can facilitate the simulation of problems with large deformation and complex geometry. The connectivity between the particles is not required in this type of

method. All one needs is to fill the computational domain with particles. One distinct MPM, smoothed particle hydrodynamics (SPH), is a very potent method for CFD problems governed by the Navier-Stokes equations. Some examples of meshless methods are briefly introduced here [Omidvar, 2010]:

- Dissipative Particle Dynamics (DPD), first developed by Hoogerbrugge and Koelmann (1993) to simulate microscopic hydrodynamic phenomena, applied Molecular Dynamics (MD) in which the particles interact according to Newton's law. Simulating macroscopic non-Newtonian flow properties of the fluid based on its microscopic structure, was the major reason of using DPD. Some applications of DPD can be summarized as modeling the rheological properties of concrete (Sim and Martys, 2004), Multicomponent flows (Coveney and Novik, 1997), porous flow (Hoogerbrugge and Koelman, 1993) and colloidal suspensions (Koelman and Hoogerbrugge, 1993). A major drawback of DPD is not to conserve energy which causes the system to be unsustainable under a temperature gradient [Omidvar 2010 and Espanol, 1997].
- Regarding its name, Meshless Finite Element Method (MFEM) is very similar to finite element method with a shape function, depending only on the nodes position, whose domain is divided into spherical polyhedral elements (Idelsohn et al., 2003). In this method, the space is divided into elements with continuity of the shape function but with discontinuity of the derivatives. This method was applied by Idelsohn et al., 2003 for fluid-structure interaction and by Xuan et al. (2004) and Ho et al. (2005) for electromagnetic simulations.

- Moving Particle Semi-implicit method (MPS) is firstly invented by Koshizuka and Oka (1996) to simulate incompressible free-surface flows. This method is the most similar method to the Smoothed Particle Hydrodynamics on the basis of particle approximation. A difference between SPH and MPS is that the kernel gradient is not calculated in MPS. A semi-implicit predictor-corrector approach obtains the solutions. Nuclear engineering (Koshizuka et al., 2001), coastal engineering (Gotoh and Sakai, 1999, Khayyer and Gotoh, 2009) are some of MPS application since its first introduction. The major disadvantages of MPS are unphysical pressure fluctuation and non-conservation of momentum.
- Smoothed Particle Hydrodynamics (SPH), was applied first by Lucy (1977) and Monaghan and Gingold (1977) in the area of astrophysics. In this method each particle carries the fluid information such as mass, velocity, pressure, density and energy. Due to its Lagrangian feature and its efficient computational techniques, it is widely used in many areas of solid and fluid mechanics. Boundaries are defined as a set of particles which are treated the same as fluid particle. This nature of SPH helps us to simulate floating objects easily. Employing an explicit integration scheme and avoiding particle penetration result in smaller time step which causes high computational cost as one of SPH disadvantages. Since the equations and formulation of SPH are very easy to implement or modify, SPH was found as the best choice for simulation of the free-surface problem specifically with large-deformation or complex geometry.

2.4 Smoothed Particle Hydrodynamics (SPH)

Smoothed particle hydrodynamics, is a truly meshfree and Lagrangian method in which the system is presented by a set of particles possessing material properties and move according to the governing equations. A vast variety of Lagrangian formulations have been proposed to calculate the local density, pressure, acceleration and velocity of the fluid. Since its first use was in astrophysical problem by Lucy (1977) and Gingold and Monaghan (1977), SPH has been comprehensively studied and extended to other fields including fluid flow with large nonlinear deformation [Liu and Liu, 2007]. SPH can be regarded as the oldest modern meshfree particle method [Liu et al., 2008 and Liu and Liu, 2010]. This method had been used for about ten years merely for astrophysical problems [Marrone, 2011]. Basics of SPH came from the probability theory, and statistical mechanics are widely employed for numerical estimation. Those initial formulations did not conserve linear and angular momentum. However, reasonably good results were obtained for many astrophysical. Several modifications have been proposed to improve the original formulation [Liu and Liu, 2010]. For instance, Monaghan and Gingold (1997) introduced a new SPH algorithm that conserves both linear and angular momentum, when they found non-conservative origin. A lot of efforts were put on the SPH method on the numerical aspects in accuracy, efficiency, stability and convergence.

These days, SPH is used in broad range of physics and engineering problems, from solid mechanics to multiphase flows. Monaghan applied SPH for simulating free-surface inviscid flows [Marrone, 2011]. Due to its Lagrangian nature, SPH is very suitable to handle very

large deformation and fragmentation. The equation of mass, momentum and energy are a set of ordinary differential equations which are very easy to understand and solve. For instance, a force between pair of particle comes from the pressure gradient [Monaghan, 1992].

2.4.1 Advantages

This method has a couple of noteworthy advantages [Monaghan, 2005a]. Contrary to finite element methods, interface problems are easy to study with SPH, if multi-materials, each described by its own set of particles, exist. Pure advection is fully treated in SPH, while modern finite difference methods produced corrupted results when a large constant velocity is superposed [Monaghan, 2005a]. Connecting the continuum and fragmentation in natural way is possible with smoothed particle hydrodynamics. Consequently, the best existing procedure for study of fragile structure in solid mechanics is SPH [Benz and Asphaug, 1994, 1995]. The resolution in SPH can be adoptable by time and space suited for astrophysical and geometrical problems where enormous variations in the relevant length scale are common (see e.g. the simulation of the Moon formation (Benz et al., 1986) or formation of stars (Bate et al., 1995, 2003).

In SPH, equations are only solved for the location in where particles or materials exist. The density of particles increase in places where the material presents and computational efforts is only focused in those regions. This feature leads to significantly reduction in memory storage and computational cost when the complex problems such as stars, drops or fragments are dealing with. Due to its similarity to molecular dynamics, including or adding multi-physics is straightforward in SPH [Crespo, 2008 and Monaghan, 2005a]. For instance,

if very small perturbation occurs in a problem, finite difference is preferable regarding the low accuracy of SPH. However, SPH shows up its advantages where large perturbations happen. Complex physics can often be included in the code with trivial effort within few days, while similar modification may take many months or years to write in other computational methods [Monaghan, 2005a]. Initial conditions can be easily programmed without any complex gridding scheme as is common in finite elements methods [Crespo, 2008]. Poisson equation is not solved for the pressure computation in fluid mechanics area, which requires much more complicated algorithm to be solved, specifically in parallel computing. Weakly compressible assumption facilitates finding fluid pressure according to local fluid pressure.

2.4.2 Disadvantages

Generally, SPH is slower computationally in comparison with other grid-based methods. The assumption of weakly compressible in fluid mechanics problem connects the time steps to the speed of sound. Higher speed of sound reduces the time step remarkably. Boundary condition implementation is still tough and not accurate enough, and some methods suffer from fluid particle penetration into boundaries. Particle disorder seriously influences the typically simple interpolation method in SPH. Although SPH produces reasonable results for first derivative, Bonet and Lok (1999) recommended gradient correction. It is recommended to apply special techniques when higher order derivatives are included.

2.4.3 Classical SPH and liquids

Basically, SPH is a technique for approximating a continuum media. Hence, it can be applied for a broad range of fluid mechanics problems [Monaghan, 2005a]. The classical SPH method was originally developed for hydrodynamics problem in which the field variables such as density, pressure, velocity, energy, and etc. are computed by a set of partial differential equation [Liu M.B., Liu G.R., 2010]. The key point in SPH formulation is the local interpolation around each particle using a smoothing function or kernel function. According to this method of interpolation, any field variable can be expressed or averaged in terms of its value at a set of disordered particles. The SPH formulation consists in two major steps, kernel approximation and particle approximation. In the first step, a function and/or its derivatives in continuous form as integral representation is presented. In this step, a function and/or its derivatives are approximated using that smoothing function and its derivatives. In the particle step, the whole domain is initially presented by a set of distributed particles. After this step, the approximation of field variables at the location of each particle is calculated by a summation of corresponding variable over the nearest neighbor points [Liu M.B., Liu G.R., 2010].

Although SPH initially was applied to gas dynamic problems, it has also been used for incompressible fluids by assuming that fluid as weakly compressible with a robust equation of state [Monaghan, 1994 and Monaghan, 2005a]. Monaghan used a stiff equation of state, so-called Tait's equation, to estimate the pressure based on fluid density considering the fluid as a weakly compressible (WCSPH) [Monaghan, 1994]. The classical SPH formulation

suffers from several weaknesses such as inaccuracy, instability and incorrect behavior of boundary condition [Omidvar, 2010; Marongiu et al., 2008]. As mentioned above, Monaghan was the first person who employed smoothed particle hydrodynamics to simulate free-surface flows [Marrone, 2011] and has put much effort to improve this method in the computational fluids science since 1980s [Omidvar, 2010].

Artificial viscosity was proposed by Monaghan (1992) to treat slightly the lack of stability. His formulation conserves the linear and angular momentum and stabilizes the numerical algorithm and handle high Mach number shocks [Monaghan, 1992]. Sometime the artificial viscosity causes particle penetration in SPH [Monaghan, 1989]. Moreover, very large effective shear velocity or velocity divergence may produce some problems including velocity shear (e.g. differentially rotating disks) [Cha and Whitworth, 2003]. Beside, Omidvar states that standard SPH formulation does not handle wave propagation in an intermediate depth and long channel [Omidvar, 2010].

Regarding the mentioned equation of state, the pressure is a function of fluid density variation and speed of sound. For fluid like water, the magnitude of sound speed is smaller than the real value, but still large enough to restrict the density fluctuation [Gomez-Gesteira et al., 2010]. This popular approach has been tested and validated in several test cases ranging from sloshing flows [Marsh et al., 2010; Colagrossi, 2005; Souto-Iglesias et al., 2004] to gravity wave propagation [Antuono et al., 2011; Landrini et al., 2007; Dalrymple and Rogers, 2006; Monaghan and Kos, 1999] and impact flows [Marrone et al., 2011; Ferrari et al., 2010; Oger et al., 2006; Trivellato et al., 2006; Colagrossi and Landrini, 2003].

2.4.3.1 ISPH

Similar to the approach adopted in Moving Particle Semi-implicit (MPS) model [Koshizuka and Oka, 1996], Cummins and Rudman (1999) proposed incompressible SPH (ISPH). ISPH utilizes and alternative approach to enforce incompressibility based on a two-step projection method. In this approach, the Poisson pressure equation, derived from an approximate projection, is solved to obtain a divergence-free velocity field [Marrone, 2011]. Later, Shao and Lo (2003) and Ellero et al (2007) developed other methods to enforce incompressibility strictly. Shao & Lo (2003) introduced a solution for Navier-Stokes equation and the incompressible mass conservation applying prediction-correction temporal method without considering incompressibility in the prediction step. Although this ISPH formulation conserves the linear momentum, it does not preserve angular momentum which can affect the results when fully nonlinear violent free-surface flows are simulated (e.g. wave breaking or water splashing) [Omidvar, 2010]. Khayyer et al. (2008) introduced a Corrected ISPH (CISPH) formulation, using the CSPH method by Bonet and Lok (1999) and the ISPH approach by Shao and Lo (2003), which ensures the conservation of angular momentum.

The major difference between ISPH and WCSPH is that the former uses fluid velocity rather than speed of sound in CFL (Courant-Friedrichs-Lewi) condition, which leads to a bigger time step. While, ISPH requires considerably tough works each time step on solving a complex algebraic system of Poisson equation, which is truly complex for an efficient parallelized computation [Marrone, 2011 and Omidvar, 2010]. In addition, to assign the dynamic boundary condition (DBC), the detection of the free-surface is required in incompressible scheme. Clearly, it is not a simple process to find the surface elevation in

three-dimensional vigorous flows, such as strong splashing water or large fragmentation [e.g. Marrone et al., 2010; Haque and Dilts, 2007]. It is known that the accuracy of the solution of Poisson equation, necessary for incompressible methods, depends on the free surface configuration. Thus, small errors in finding the particles at the surface may lead to different flow dynamics [Marrone, 2011]. On the other hand, WCSPH as an explicit central scheme suffers from unphysical high frequency oscillation in pressure. Therefore, it needs an artificial viscosity term for numerical stability.

It is very common to smooth out the density oscillation by a density filter, either Shepard filter or Moving Least Squares (MLS) interpolant, and to re-assign a new density to each particle following Colagrossi and Landrini (2003) or Belytschko et al. (1998) and Dilts (1999). These techniques are very effective but one of the MLS disadvantages is that some parameters need to be tuned. Besides, the hydrostatic pressure is not preserved after a long-time simulation [Marrone, 2011]. Recently, on 2009, Molteni and Colagrossi found that adding a conservative diffusive term in the continuity equation can eliminate spurious noise inside the pressure field. Finally, the Riemann solvers were introduced by Moussa et al. (1999) to smooth the pressure field instead of using artificial viscosity.

There are two different types of equation exist for calculating density in fluid mechanics problem. One is the formula that was used primarily in SPH by Lucy 1977 and mostly used in astrophysics context. In this formulation, the density of each particle at each time step is directly computed by interpolating the density of neighboring particles. The other formulation is derived from direct discretization of the continuity equation. In the second method, the density variation of particles is computed according to the velocity divergence.

Since the former method causes a density drop near the free surface or probably solid boundaries, this is not the best option for fluid calculations [Monaghan, 1992]. According to the later formulation, the density changes when particles move relative to each other [Gomez-Gesteira et al. 2010].

2.4.3.2 SPHysics

SPHysics is an open source code written in Fortran language which follows the Smoothed Particle Hydrodynamics formulation of Monaghan (1992). Several multinational researchers from different universities, such as Johns Hopkins University (USA), the University of Vigo (Spain), the University of Manchester (UK), and the University of Roma La Sapienza (Italy), developed this code. Its prior version was released on August 2007 and now, it is broadly used in different fields of studies by several researchers. The source code including a comprehensive manual file and various test cases is now downloadable from its official webpage³, see Figure 2-1. Now, several serial and parallel versions of this code are available in 2D and 3D. GPUSPH and DualSPHysics are also released during the current research.

³ <http://wiki.manchester.ac.uk/SPHysics>

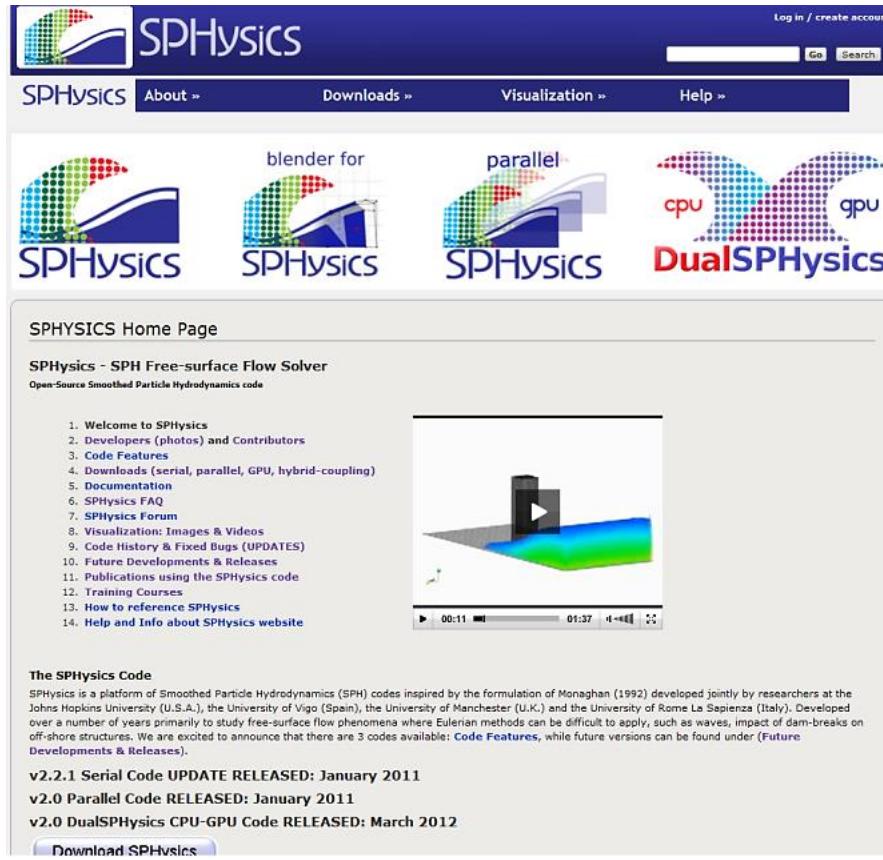


Figure 2-1- Official homepage of SPHysics code

Alexis Héault developed GPUSPH primarily for studying the lava cooling. His original contribution was presented at the 3rd SPHERIC workshop in Lausanne, Switzerland in 2008. GPUSPH follows the same governing equations of the SPHysics, instead, it uses Nvidia's CUDA language for coding Nvidia graphics cards to do heavy computations associated with SPH. Its tremendous speedup has motivated researchers to put more effort on this code; however, CUDA is not a widely-used programming language.

DualSPHysics is a set of C++, CUDA and Java codes designed to model real-life engineering problems. It combines CPU and GPU for SPH calculations. Similar to GPUSPH,

it follows SPHysics governing equations to study the free-surface problems where Eulerian method is very complex to apply.

2.4.4 SPH Formulation

So far, a broad review about the history of meshfree methods and specifically smoothed particle hydrodynamic has been discussed. Some general features of meshless methods were introduced. In this section, the basic idea and necessary formulation of SPH are presented more in detail. The SPH approximation including general concept of this method, kernel approximation (integral interpolant) and discretized particle approximation, will be reviewed. The meaning of support and influence domain is described. In addition, the tricks used to interpret the Navier-stokes in SPH form will be discussed and new schemes, methods and formulation will be introduced and the advantages and disadvantages of them will be discussed [Liu and Liu, 2007].

As mentioned before, hydrodynamics problems are mathematically described as a set of partial differential equations (PDE) of field variables, density, velocity, energy, etc. Generally, except for some very simple cases, finding the analytical solution of these PDEs is not an easy task or even possible. Hence, people put an effort in seeking and describing the solution numerically. To do so, the domain, where the PDEs are defined, needs to be discretized. Then a function approximation is applied to convert the PDEs into a set of ordinary differential equations as functions of time only. Finally, the best method is chosen among conventional finite difference method for time integration [Liu and Liu, 2007].

In the SPH method, first the domain is discretized by a set of arbitrarily distributed particles which are not connected to each other as in grid-based methods. These finite numbers of macroscopic particles which carry the properties of fluid every time, presents the whole continuum media. Then an *integral representation method* is applied to approximate the field function. In the next step, called *particle approximation*, the above integral representation is interpreted into discretized form and the field function and its derivative is computed with summations over all particles surrounding a specific particle within its *support domain*. Finally, the mentioned particle approximation is employed to all field functions in the governing equations (PDEs) to make a set of ODEs in discretized form with respect to time. These ODEs are integrated using an explicit and stable enough integration algorithm to present the time history of all field variables.

2.5 Basics of SPH

The key idea of SPH is an integral interpolation that allows any function to be expressed in terms of its value in the surrounding domain (Ω) [Lucy, 1977; Monaghan, 1985 and 1992; Omidvar, 2010]. The linear approximation of any function $\langle f(\mathbf{r}) \rangle$ is defined by

$$\langle f(\mathbf{r}) \rangle = \int_{\Omega} f(\mathbf{r}') W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}' \quad (2-1)$$

Where $d\mathbf{r}'$ is a differential volume element and W , the weighting function, is so-called smoothing kernel function or smoothing function or smoothing kernel or simply kernel in many literature [Liu et al., 2003; Morris, 1996, Liu and Liu, 2010; Monaghan, 1992; Fulk,

1994]. The kernel length, h , determines the size of kernel function that is a domain of influence around point r . The size of h , typically, is chosen higher than initial particle space, dp . In 2D and 3D, the support domain is a circle or sphere, respectively, with a radius of kh , where k depends on the kernel definition (k is 2 in our study). Generally, h is constant, while in some application this parameter can vary in time and space [Monaghan, 1992]. If a delta function was used as kernel function, the interpolant reproduces f exactly. The kernels are, practically, functions that tend to delta function if the kernel length, h , approaches zero [Monaghan, 2005a]. Since W is not a Dirac delta function, the integral interpolant reproduces f approximately. The kernel function which is an even function should satisfy a number of conditions because the performance of an SPH model depends on the choice of the weighting function [Crespo, 2008]. First of all, this function should be positive inside the domain Ω (*positivity*). Secondly, kernel function should result zero for outside the influence domain (*compact support*). Normalization condition or *unity condition* states

$$\int_{\Omega} W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}' = 1 \quad (2-2)$$

That means the constants are interpolated exactly [Monaghan, 2005a]. The fourth condition is *delta function property*, that is

$$\lim_{h \rightarrow 0} W(\mathbf{r} - \mathbf{r}', h) = \delta(\mathbf{r} - \mathbf{r}') \quad (2-3)$$

Finally, a proper kernel function should be sufficiently smoothed and should decrease monotonically. It is shown in the related literature that the kernel approximation has second

order accuracy, $O(h^2)$ [Monaghan, 1982 and 1992; Hernquist and Katz; 1989; Fulk, 1994; Liu and Liu, 2007].

$$\langle f(\mathbf{r}) \rangle = f(\mathbf{r}) + O(h^2) \quad (2-4)$$

For numerical simulations, the integral form can be approximated by a summation over the mass element. This summation interpolant is

$$f(\mathbf{r}_i) = \sum_j f_j W(\mathbf{r} - \mathbf{r}_j) V_j \quad (2-5)$$

where the summation is over all particles within compact support domain of particle i . Particle j has mass m_j , position \mathbf{r}_j , density ρ_j and velocity \mathbf{v}_j . Therefore, $V_j = m_j / \rho_j$ in (2-5) represents the volume of particle j . Figure 2-2 illustrate a schematic picture of the support domain of kernel function W for particle i in 2D.

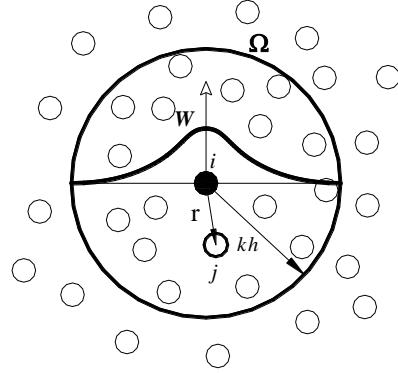


Figure 2-2- Particle approximations and influence domain of particle i

The SPH formulation allows derivatives to be approximated readily. One of the advantages of SPH method is that using a differentiable kernel obtains the derivatives of

interpolant (2-5) by ordinary differentiation; in contrast the grid-based method is one in which the derivatives are calculated from neighboring points, which is extremely complicated for irregularly located SPH particles. Therefore, the derivatives of function (2-5) can be estimated by

$$\langle \nabla f(\mathbf{r}_i) \rangle = \sum_j f_j \nabla_i W(\mathbf{r}_{ij}) V_j \quad (2-6)$$

where \mathbf{r}_{ij} is the distance between particle i and j , $\mathbf{r}_i - \mathbf{r}_j \cdot \nabla_i W(\mathbf{r}_{ij})$ represents the gradient of kernel taken with respect to the position of i

$$\nabla_i W_{ij} = \frac{\mathbf{x}_i - \mathbf{x}_j}{r_{ij}} \frac{\partial W_{ij}}{\partial r_{ij}} = \frac{\mathbf{x}_{ij}}{r_{ij}} \frac{\partial W_{ij}}{\partial r_{ij}} \quad (2-7)$$

And similarly the divergence of and vector can be expressed as

$$\langle \nabla \cdot \mathbf{f}(\mathbf{r}_i) \rangle = \sum_j \mathbf{f}_j \cdot \nabla_i W(\mathbf{r}_{ij}) V_j \quad (2-8)$$

To obtain higher accuracy, typically, two following formula are applied. Regarding two mathematical identities,

$$\rho \nabla(f) = \nabla(\rho f) - f \nabla \rho \quad (2-9)$$

$$\nabla(f) = \rho \nabla\left(\frac{f}{\rho}\right) - \frac{1}{\rho} f \nabla \rho \quad (2-10)$$

Substituting above identities into (2-6) yields

$$\langle \nabla f(\mathbf{r}_i) \rangle = \frac{1}{\rho_i} \sum_j m_j (f_j - f_i) \nabla_i W_{ij} \quad (2-11)$$

$$\langle \nabla f(\mathbf{r}_i) \rangle = \rho_i \sum_j m_j \left(\frac{f_i}{\rho_i^2} + \frac{f_j}{\rho_j^2} \right) \nabla_i W_{ij} \quad (2-12)$$

Monaghan (1992) presents several formulas for scalar field gradients or vector divergence. The well-known feature of (2-11) is that to give zero gradient of a constant field off, while (2-12) does not.

2.6 Governing Equation

2.6.1 Continuity

A fluid dynamics problem can be solved through three fundamental physical conservation laws, conservation of mass, momentum and energy. As mentioned before, two approaches, Eulerian description (spatial description) and Lagrangian description (material description), are available for describing the physical governing equations. The key difference between these two approaches is that in Lagrangian form, the total derivative does not include the convective part as Eulerian form does. The SPH equations of motion are derived based on Lagrangian form of all above conservation laws. The general form of mass continuity equation in Lagrangian form is

$$\frac{d\rho}{dt} + \rho \nabla \cdot \mathbf{v} = 0 \quad (2-13)$$

when the liquid is considered as incompressible, (2-13) reduces to

$$\nabla \cdot \mathbf{v} = 0 \quad (2-14)$$

and consequently a Poisson equation will be introduced to the Navier-Stokes equation for the pressure field. Therefore, the pressure solution is obtained through the solution of an algebraic system [Marrone, 2011]. To do so, all the boundaries of fluid should be known and generally, it is not a simple and accurate computation for free surface in meshfree context. To overcome with this problem, fluid is assumed as weakly-compressible. This is very similar to those cases in which a compressible fluid is modeled under the constraint of negligible compressibility effects. Hence, an equation of state is introduced to link the pressure and fluid density together. Regarding weakly-compressible assumption, SPH formulation is considered as a fully-explicit method.

2.6.2 Momentum Equation (Acceleration Equation)

The momentum conservation equation in a continuum media is written as

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla P + \mathbf{F} + \text{viscous term} \quad (2-15)$$

where \mathbf{v} , P , ρ are the velocity, pressure and density, respectively. \mathbf{F} is the body force which is equal to gravitational acceleration, $\mathbf{g} = (0,0,-9.81) \text{ m/s}^2$ in SPH formulation.

2.7 Standard SPH formulation

2.7.1 Conservation of mass

It is mentioned that using summation (2-5), any quantity can be approximated. If density, ρ is substituted in (2-5), the density is interpolated by so-called *summation density* method

$$\rho_i = \sum_j m_j W_{ij} \quad (2-16)$$

This equation simply implies that the density of each particle can be estimated by the weighted average of the densities of the neighbor particles inside the support domain. Another way to find the density is to calculate the density acceleration, $d\rho / dt$. Regarding the following identities which is commonly used in classic Lagrangian formulation

$$\frac{d\rho}{dt} = -\rho \nabla \mathbf{v} = -\nabla(\rho \mathbf{v}) + \nabla \rho \cdot \mathbf{v} \quad (2-17)$$

Using the gradient operation in SPH form, equation (2-6), and after some mathematical work, (2-17) becomes

$$\frac{d\rho_i}{dt} = \sum_j m_j \mathbf{v}_{ij} \nabla_i W_{ij} \quad (2-18)$$

In which the notation $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ has been used. This method is called *continuity density* method. There are advantages and disadvantages for both approaches of continuity density and summation density. In contrast with the continuity density method, the density summation conserves the mass exactly [Monaghan, 1992; Fulk, 1994]. However, the summation density suffers from serious boundaries deficiency due to the particle

inconsistency. Thus, this method is not the best choice in fluid calculations since it leads to a density drop near the boundaries [Gomez-Gesteira et al., 2010]. Another negative point of density summation formulation is its higher computational cost. Density value should be known before other parameters can be evaluated as density summation technique is applied. Whereas, the continuity density approach calculates the density acceleration as it computes other terms. Some modifications have been proposed to reduce the errors of above methods. The density filter is the most straightforward option.

2.7.2 Density filters

Although SPH is very accurate in computing the kinematics of a problem, typically, large oscillation is observed in the pressure field of particles. This issue becomes more serious close to boundaries (including surface), where the accuracy suffers from particle deficiency at boundaries. Since the pressure is computed by the fluid density, the most straightforward and computationally least expensive method to smooth out the pressure is to perform a filter over the density of particles and to re-assign a density to each of them [Gomez-Gesteira et al., 2010; Colagrossi and Landrini, 2003; Dilts, 1999; Belytschko et al., 1998]. There are two types of corrections available, zero order Shepard filter and first order Moving Least Squares (MLS).

Shepard filter is the most quick and simple method which is applied every 20 to 40 time steps [Liu and Liu 2010; Randles and Libersky, 1996; Chen and Beraun, 2000]. At each application of the Shepard filter, the new density is re-assigned as

$$\rho_i^{new} = \frac{\sum_j m_j W_{ij}}{\sum_j \left(\frac{m_j}{\rho_j} \right) W_{ij}} \quad (2-19)$$

MLS approach was firstly developed by Dils (1999) and applied by Colagrossi and Landrini (2003). Similar to Shepard filter, MLS should be also applied every 20-40 time steps. The first order corrected kernel function, W_{ij}^{MLS} , is computed

$$W_{ij}^{MLS} = B(\mathbf{x}_i) \cdot (\mathbf{x}_i - \mathbf{x}_j) W_{ij} \quad (2-20)$$

so that in 2D it becomes

$$W_{ij}^{MLS} = [B_0(\mathbf{x}_i) + B_{1x}(\mathbf{x}_i)(x_i - x_j) + B_{1z}(\mathbf{x}_i)(z_i - z_j)] W_{ij} \quad (2-21)$$

where the correction vector B is given by

$$B(\mathbf{x}_i) = \begin{bmatrix} B_0 \\ B_{1x} \\ B_{1z} \end{bmatrix} = A^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (2-22)$$

$$A = \sum_j W_{ij} \tilde{A} V_j \quad (2-23)$$

$$\tilde{A} = \begin{bmatrix} 1 & (x_i - x_j) & (z_i - z_j) \\ (x_i - x_j) & (x_i - x_j)^2 & (z_i - z_j)(x_i - x_j) \\ (z_i - z_j) & (x_i - x_j)(z_i - z_j) & (z_i - z_j)^2 \end{bmatrix} \quad (2-24)$$

2.7.3 Diffusive term

Antuono et al. (2010) state that although the density filters give good results, for long time simulations, it does not conserve the total volume the particles due to improperly filtering of

hydrostatic component. Antuono et al. (2010) and Marrone et al. (2011) were able to extend the work by Molteni and Colagrossi (2009). In this approach numerical diffusion terms are added both in the density and the thermal energy equations. In order to avoid unphysical effects, all diffusive terms have to vanish as the numerical accuracy increases. This can be tuned by multiplying the diffusive terms by appropriate coefficients. Antuono et al. (2010) showed that the range of variability of such coefficients is narrow. Another approach to reduce density (pressure) field fluctuation is to increase numerical accuracy through Riemann solvers. The strong advantage of the Riemann solvers is that it does not need any parameters to control diffusion. However, Ferrari et al. (2009) pointed out that Riemann solvers are often too diffusive/dissipative, therefore are not proper for vigorous free-surface deformation [Antuono et al., 2010]. $\delta - SPH$ is a new formulation proposed by Marrone et. al (2011) in which the diffusive term is added to the continuity equation

$$\frac{d\rho_i}{dt} = \rho_i \sum_j (u_i - u_j) \cdot \nabla_i W_{ij} V_j + \delta h c_0 \sum_j \psi_{ij} \cdot \nabla_i W_{ij} V_j \quad (2-25)$$

where

$$\psi_{ij} = 2(\rho_i - \rho_j) \frac{\mathbf{r}_{ij}}{r_{ij}^2} - \left[\langle \nabla p \rangle_i^L + \langle \nabla p \rangle_j^L \right] \quad (2-26)$$

and $\langle \nabla p \rangle^L$, the normalized density gradient is defined as

$$\langle \nabla p \rangle_i^L = \sum_j (\rho_i - \rho_j) L_i \nabla_i W_{ij} V_j \quad (2-27)$$

$$L_i = \left[\sum_j (\mathbf{r}_i - \mathbf{r}_j) \otimes \nabla_i W_{ij} V_j \right]^{-1} \quad (2-28)$$

δ , and h are diffusion coefficient, kernel length, respectively. c_0 , the speed of sound at the reference density, will be described in detail later. Marrone et al. (2011) set the value of diffusion coefficient equal to 0.1 in all their simulations. Antuono et al. (2010) explained that the diffusive term becomes trivial as space resolution, h , goes to zero. Figure 2-3 compares the effect of different density filter formulations on a simple case of free-surface dam-break flow. Note the severe pressure gradient in Figure 2-3-a which presents standard model without any filter; however more regular density distribution can be observed applying a density filter.

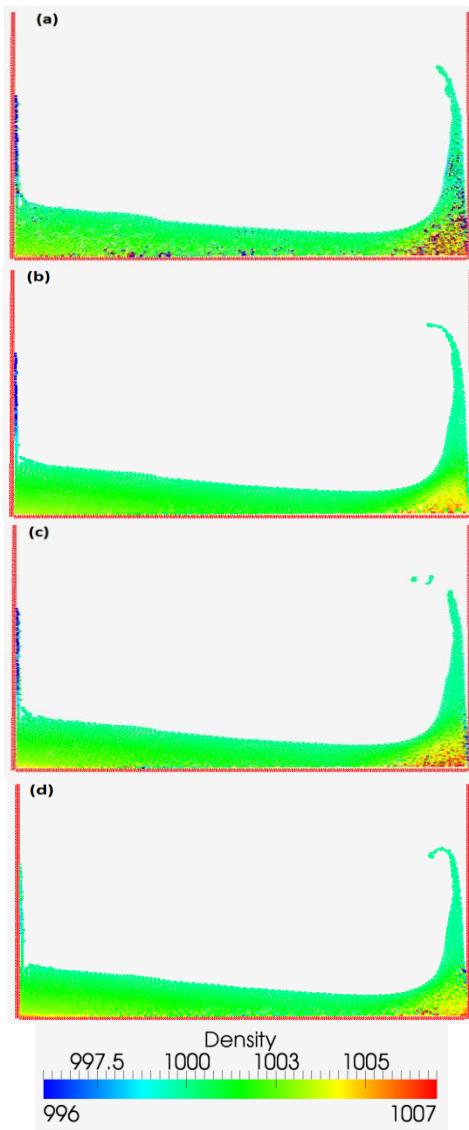


Figure 2-3- Density field for a dam break test comparing the effect of density filter.
(a) without any filter, (b) with Shepard filter, (c) with MLS filter, and (d) with diffusive term

2.7.4 Conservation of momentum

The pressure gradient term in momentum equation, (2-15), can be written in the form of (2-29) employing (2-6)

$$\nabla p_i = \sum_j m_j \frac{p_j}{\rho_j} \nabla_i W_{ij} \quad (2-29)$$

then, the momentum equation for an inviscid flow in SPH form becomes

$$\frac{d\mathbf{v}_i}{dt} = - \frac{1}{\rho_i} \sum_j m_j \frac{p_j}{\rho_j} \nabla_i W_{ij} \quad (2-30)$$

But, (2-30) does not conserve the linear and angular momentum exactly, since the force on i due to j is not necessarily equal-opposite to that of i on j , because $p_i \neq p_j$. Hence, to conserve the linear and angular momentum completely, a symmetric form of momentum is required. Using the mathematical identity of (2-31), symmetrized momentum equation is expressed by (2-32).

$$\frac{\nabla p}{\rho} = \nabla \left(\frac{p}{\rho} \right) + \frac{p}{\rho^2} \nabla \rho \quad (2-31)$$

$$\frac{d\mathbf{v}_i}{dt} = - \sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) \nabla_i W_{ij} \quad (2-32)$$

Although some authors (Bonet and Lok, 1999) preferred (2-33), equation (2-32) is the most commonly used to express the gradient pressure in SPH.

$$\frac{d\mathbf{v}_i}{dt} = - \sum_j \frac{1}{\rho_i \rho_j} m_j (p_i + p_j) \nabla_i W_{ij} \quad (2-33)$$

2.7.5 Viscosity term

A numerical scheme has to be stable and the viscosity term is a key component in preventing instabilities in fluid motion. Three different types of diffusion are commonly used in smoothed particle hydrodynamics: 1) artificial viscosity, 2) laminar viscosity and 3) turbulence modeling (laminar viscosity + Sub-Particle Scale (SPS) Turbulence).

2.7.5.1 Artificial Viscosity

Monaghan proposed this formulation for the first time [Monaghan, 1985] and it is frequently used for hydrodynamics problem due to its simplicity. The momentum equation in SPH form after adding artificial viscosity, Π_{ij} , is expressed

$$\frac{d\mathbf{v}_i}{dt} = - \sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \Pi_{ij} \right) \nabla_i W_{ij} \quad (2-34)$$

Π_{ij} and its different parameters for fluid mechanics problems are defined

$$\Pi_{ij} = \begin{cases} \frac{\alpha \bar{c}_{ij} \mu_{ij}}{\bar{\rho}_{ij}} & \mathbf{v}_{ij} \cdot \mathbf{r}_{ij} < 0 \\ 0 & otherwise \end{cases} \quad (2-35)$$

$$\bar{\rho}_{ij} = (\rho_i + \rho_j) / 2 \quad (2-36)$$

$$\bar{c}_{ij} = (c_i + c_j) / 2 \quad (2-37)$$

$$\mu_{ij} = \frac{h \mathbf{v}_{ij} \cdot \mathbf{r}_{ij}}{r_{ij}^2 + \eta^2} \quad (2-38)$$

where c_k is the speed of sound of particle k and α , the non-dimensional artificial viscosity coefficient, is in the order of 0.01 in our simulations. $\eta^2 = 0.01h^2$ is chosen to avoid

singularity when two particles are approaching each other. This expression produces a shear and bulk viscosity due to the linear term of velocity difference [Monaghan, 1985]. This method conserves both linear and angular momentum. Laminar and SPS turbulence viscosity are other options for modeling viscosity which are more closely resemble the governing equations.

2.7.5.2 Laminar Viscosity

Gonzalez et al. (2009) analyzed, tested and explored the properties of the most referred WCSPH⁴ laminar viscosity models, Takeda et al. (1994), Morris et al. (1997) and Monaghan-Cleary-Gingold (1998, 2005b). Results prove that their accuracy is reasonable for both linear type flows and those of more complex nature such as Lamb-Oseen vortex evolution. Generally, the conservation of momentum equation with laminar viscosity term in Lagrangian form is expressed

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla P + g + \nu_{kv} \nabla^2 \mathbf{v} \quad (2-39)$$

where Morris et al. (1997) and Lo and Shao (2002) simplified the laminar stress term to

$$(\nu_{kv} \nabla^2 \mathbf{v})_i = \sum_j m_j \left(\frac{2\nu_{kv} \mathbf{r}_{ij} \nabla_i W_{ij}}{\bar{\rho}_{ij} |\mathbf{r}_{ij}|^2} \right) \mathbf{v}_{ij} \quad (2-40)$$

in which $\bar{\rho}_{ij}$, the averaged density, is expressed by (2-36) and ν_{kv} is the kinematic viscosity of laminar flow ($\sim 10^{-6} \text{ m}^2/\text{s}$).

⁴ Weakly Compressible Smoothed Particle Hydrodynamics

2.7.5.3 Laminar Viscosity and Sub-Particle Scale (SPS) Turbulence

The viscosity term (2-35) was originally included in momentum equation to represent the viscosity and also to avoid particle interpenetration. In addition, for free-surface flows, it plays a key role in stabilizing the numerical scheme. However, in some cases it is too dissipative and affects the shear in the fluid [Dalrymple and Rogers, 2006]. Basically, the governing equations are averaged over a length scale comparable to smoothing length and that is sufficient to model the large-scale eddies. For eddies smaller than particle size, a turbulent model is needed to take into account their effect on flow field. First Gotoh et al., 2001 and then Rogers and Dalrymple (2004) replaced the standard SPH viscous formulation by introducing a Sub-Particle Scale (SPS) technique. Using a Large Eddy Simulation (LES) model [Christensen, 2006; Crespo, 2008], this sub-grid scaling scheme, which uses a Smagorinsky [Smagorinsky, 1963] eddy viscosity, is very similar to Lo and Shao's (2002) formulation. Gotoh et al. (2001) applied SPS in the Moving Particle Semi-implicit (MPS) technique. However, SPS technique for a compressible fluid requires a spatial averaging methodology such as that Dalrymple and Rogers (2006) used, that illuminates introducing new terms into the conservation of mass equation. The momentum conservation in this approach is

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla P + g + \nu_{kv} \nabla^2 \mathbf{v} + \frac{1}{\rho} \nabla \bar{\tau} \quad (2-41)$$

where the third term on RHS of (2-41), the laminar viscosity, can be treated following Eq. (2-40) and $\bar{\tau}$ represents the SPS stress tensor for compressible fluid

$$\frac{\bar{\tau}}{\rho} = 2\nu_t \tilde{S}_{ij} - \frac{2}{3} \tilde{S}_{kk} \delta_{ij} - \frac{2}{3} C_s \Delta p^2 \delta_{ij} |\bar{S}_{ij}|^2 \quad (2-42)$$

where the turbulence eddy viscosity, ν_t , is determined by standard Smagorinsky model

$$\nu_t = (C_s \Delta p)^2 |\bar{S}_{ij}| \quad (2-43)$$

in which the Smagorinsky constant, $C_s=0.12$, Δp is the initial particle spacing, $C_I=0.0066$ and the local strain rate is given by $|\bar{S}_{ij}| = (2\tilde{S}_{ij}\tilde{S}_{ij})^{1/2}$ and the Favre-filtered rate of strain tensor is

$$\tilde{S}_{ij} = -\frac{1}{2} \left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i} \right) \quad (2-44)$$

Finally, following Dalrymple and Rogers (2006), the momentum equation in SPH notation becomes

$$\frac{d\mathbf{v}_i}{dt} = -\sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \frac{\tau_i^*}{\rho_i^2} + \frac{\tau_j^*}{\rho_j^2} \right) \nabla_i W_{ij} + \sum_j m_j \left(\frac{2\nu_{kv} \mathbf{r}_{ij} \nabla_i W_{ij}}{\bar{\rho}_{ij} |\mathbf{r}_{ij}|^2} \right) \mathbf{v}_{ij} + \mathbf{g} \quad (2-45)$$

where ν_{kv} is kinematic viscosity.

2.7.6 Energy Equation

During a simulation, kinetic, potential and thermal energy (e_i) of each particle are calculated by particle velocity, elevation and energy equation, respectively. The summation of kinetic, potential and thermal energy over all particles yields the total energy of system.

$$\frac{de_i}{dt} = \frac{1}{2} \sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \Pi_{ij} \right) \mathbf{v}_{ij} \nabla_i W_{ij} \quad (2-46)$$

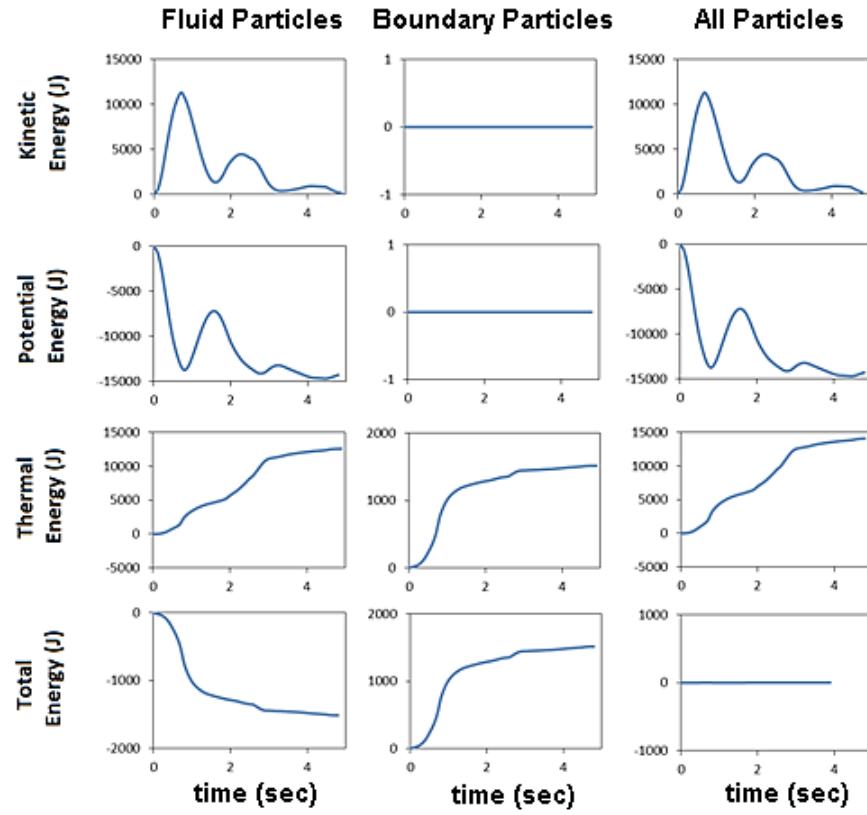


Figure 2-4- Time Variation of the system energy during the collapse of water column

Figure 2-4 presents the different types of energy of a simple dam breaking. This figure shows the kinetic, potential, thermal and total energy of fluid, boundary and all particles in 12 graphs. They show how different types of energy vary in the domain and finally, total energy changes only 0.3%.

2.7.7 Equation of State (EOS)

As discussed previously, in WCSPH, a fluid is no longer fully incompressible. Weakly compressible assumption permits us to apply a robust equation of state to link the pressure of

a particle to its density which is truly faster and easier than solving the complex equation of Poisson. A so-called Tait's equation of state, Eq. (2-47) is frequently used in SPH following Monaghan (1994) and Batchelor (1974).

$$p = B \left[\left(\frac{\rho}{\rho_0} \right)^{\gamma} - 1 \right] \quad (2-47)$$

in which $B = c_0^2 \rho_0 / \gamma$ is a constant related to the bulk modulus of elasticity of the fluid [Crespo, 2008]. ρ_0 , the reference density is usually taken as the density of fluid at the free surface i.e. 1000 Kg/m³. The power of γ is typically 7 in fluid dynamic simulation and the minus one in (2-47) ensures zero pressure at the free surface.

When a particle approaches another one, the density of both particles increases locally; therefore, the pressure of them grows dramatically regarding Eq. (2-47). Then, they repel each other, according to the momentum equation. Similarly, when the density at a specific point decreases, the pressure drop to a negative value and that makes attraction between particles. This fact causes the particle at very top layer of fluid stay at the surface. The specific entropy (S) does not influence the pressure in Eq. (2-47) and it is purely as a function of density. While it can be negligible for liquid, S has a significant effect on the pressure of gaseous media.

Due to compressibility assumption of the fluid, the speed of sound of each particle which is the square root of the derivative of the pressure with respect to the density is

$$c^2(\rho) = \frac{\partial P}{\partial \rho} = \frac{B\gamma}{\rho_0} \left(\frac{\rho}{\rho_0} \right)^{\gamma-1} \quad (2-48)$$

Substituting ρ_0 in (2-48) gives the speed of sound at reference density which is needed for calculating B.

$$c_0^2 = c^2(\rho_0) = \frac{B\gamma}{\rho_0} \quad (2-49)$$

Typically, in the SPH simulation, the speed of sound at rest is assigned and fixed at the initial time under the following constrain:

$$c_0 \geq 10 \max |\mathbf{u}| \quad (2-50)$$

where $\max |\mathbf{u}|$ is the expected maximum velocity in the flow time evolution. In this way, Mach number, the ratio of maximum velocity and speed of sound, is about $\text{Ma}=0.1$. Under this condition, it is reasonable to consider the fluid as a weakly compressible media and expect the variation in the density within 1%, i.e. $\Delta\rho/\rho_0 \sim 1\%$. Later, it will be shown that the time step is connected to the speed of sound. Thus, the real value of sound speed in water, $c_0=1500$ m/s, leads to very small time steps which causes large computational cost. Note that the wave celerity can be faster than the flow velocity in presence of gravity wave. In shallow water regime, the wave celerity is approximately expressed by $c_w^2 = gd$, in which d is the water depth. Therefore, the speed of sound is taken

$$c_0 = 10 \sqrt{gd} \quad (2-51)$$

The Morris formula, another form of the equation of state, can be derived by linearizing the Tiat's equation and neglecting the higher order terms of $\Delta\rho/\rho_0$:

$$\begin{aligned}
P &= B \left[\left(\frac{\rho_0 + \Delta\rho}{\rho_0} \right)^\gamma - 1 \right] = B \left\{ 1 + \gamma \frac{\Delta\rho}{\rho_0} + O \left[\left(\frac{\Delta\rho}{\rho_0} \right)^2 \right] - 1 \right\} \approx \\
&\approx B \gamma \frac{\Delta\rho}{\rho_0} = c_0^2 (\rho - \rho_0)
\end{aligned} \tag{2-52}$$

Note that, in contrast with Tiat's formula, Morris equation is not a function of γ and usually the speed of sound is set to c_0 during simulation [Marrone, 2011].

2.7.8 Particle position (XSPH)

Regarding basic kinematic laws, the location of each particle can be estimated by its velocity and current position

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i \tag{2-53}$$

Monaghan (1992 and 1994) modified (2-53) with an average velocity characteristic of the neighborhood of particle i .

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i + \varepsilon \sum_j \frac{m_j \mathbf{v}_{ji}}{\bar{\rho}_{ij}} W_{ij} \tag{2-54}$$

where the average velocity, $\bar{\rho}_{ij}$, is expressed by (2-36) and constant ε ranges between 0 and 1 (often, $\varepsilon = 0.5$). XSPH increases dispersion in the media of nearly incompressible flows in order to avoid particles clumping together. This method keeps particles moving more smoothly, so, it is efficient to obtain particle location according to XSPH instead of (2-53).

2.7.9 Choice of Kernel Function

In section 2.5, Kernel function and its importance in SPH interpolation were described. It was discussed that a kernel function must have some essential properties. Any function that satisfies those conditions can be considered as a kernel function in SPH integral interpolation. Many researchers and scientists have introduced or tried different types of smoothing functions. Liu and Liu (2007) and Fulk and Quinn (1996) review the characteristic of some of the most frequently used functions in the SPH literature in detail. Here, those which are used in this study are introduced briefly. In the following section, it will be discussed that how some corrections can increase the SPH interpolation and discretized Navier-Stokes formulation. In all following equations, q is r/h , where r is the distance between two particles and h is the kernel length.

2.7.9.1 Quadratic

Johnson et al. (1996) used this second-order smoothing function to model a severe impact problem. Since the first derivative of this function monotonically increases from zero to two (Figure 2-5), it prevents particle penetration or clustering in compression problems. The force from pressure gradient increases as particles are approaching and decreases as they are moving apart.

$$W(r, h) = \alpha \left[\frac{1}{4}q^2 - q + 1 \right] \quad 0 \leq q \leq 2 \quad (2-55)$$

where α is $3/(2\pi h^2)$ in 2D and $15/(16\pi h^3)$ in 3D. Figure 2-5 illustrates this function and its first derivative.

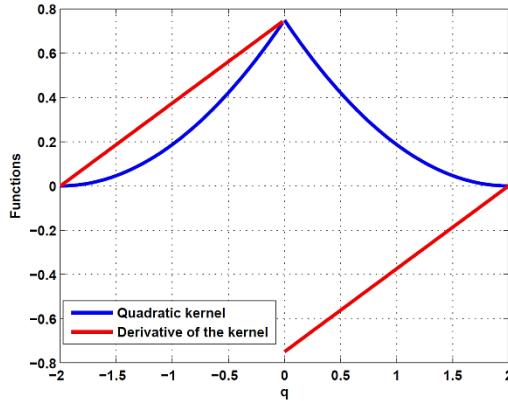


Figure 2-5- Quadratic kernel function and its derivative [Crespo, 2008]

2.7.9.2 Cubic Spline

The cubic spline (B-spline) kernel is the most common one in SPH literature and has been broadly employed by researchers as it resembles a Gaussian function while having narrower compact support. This third-order kernel function goes to zero for $r > 2h$ and this reduces the computational cost. This function has a potential of tensile instability (see section 2.7.11).

$$W(r, h) = \alpha \begin{cases} 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3 & 0 \leq q < 1 \\ \frac{1}{4}(2-q)^3 & 1 \leq q < 2 \\ 0 & q \geq 2 \end{cases} \quad (2-56)$$

where α is $10/(7\pi h^2)$ in 2D and $1/(\pi h^3)$ in 3D. Figure 2-6 illustrates this function and its first derivative.

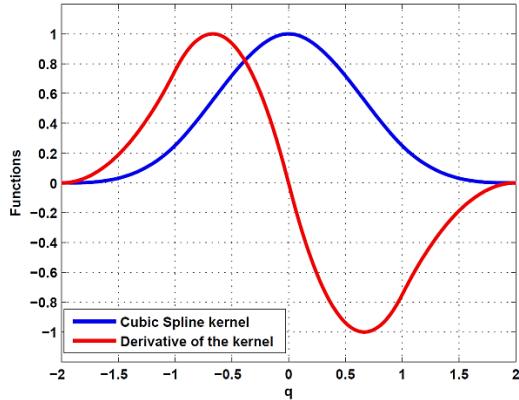


Figure 2-6- Cubic spline kernel function and its first derivative [Crespo, 2008]

2.7.9.3 Wendland (Quintic)

Generally, the higher-order kernels yields greater accuracy of the SPH scheme, although using higher order kernels increases computation time. Literature [Panizzo et al., 2007] proves that this forth-order function gives bring the best compromise between accuracy and computational cost. Figure 2-7 plots equation (2-57).

$$W(r, h) = \alpha \left(1 - \frac{q}{2}\right)^4 (2q + 1) \quad 0 \leq q \leq 2 \quad (2-57)$$

where α is $7/(4\pi h^2)$ in 2D and $7/(8\pi h^3)$ in 3D.

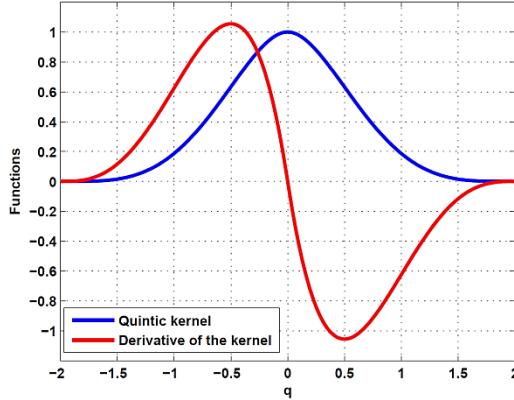


Figure 2-7- Wendland kernel function and its first derivative [Crespo, 2008]

Figure 2-7 shows that the first derivative of Wendland function also has a minimum around $q=0.5$, that means this kernel can lead to tensile instability as the distance between a pair of particles goes beneath $h/2$. Monaghan (2000) provided a correction to stop particle clustering for those kernels that have minimum in their first derivatives.

2.7.10 Choice of Kernel Length

The error of replacing integral interpolant (2-1) with summation (2-5) is normally $O(h^2)$ [Monaghan, 1992]. Therefore the first rule to increase the accuracy is to decrease h , that is $h \rightarrow 0$. Studying the error in summation interpolant, Monaghan (2005a) showed that the dominant error terms vanish if $h=dp$ and are small if $h > dp$. Moreover, to ensure the error in interpolation integrals tends to zero, the number of particles inside the kernel should go to infinity [Monaghan, 2000], $dp / h \rightarrow 0$. Although mathematics and one's intuition agree that infinite number of particles inside the support domain increases the accuracy, vast number of interactions inside the kernel increases the computational cost significantly from

programming point of view. Gathering all above hints with numerical simulation experiences demonstrates that a choice of $h=(1.2-1.6) dp$ is the best choice to satisfies the accuracy and avoid expensive simulations.

2.7.11 Tensile Instability

Swegle and Attaway (1995) studied the SPH stability via a one-dimensional Von Neumann analysis. They found that SPH suffers from a tensile instability that is not from the numerical time integration algorithm.

Tensile instability causes particle clustering in SPH. Regarding momentum equation, when particles have a negative pressure they may attract each other and clustering occurs. This statement does not happened in the real world where attraction is prevented by repulsive force between the atoms. Regarding the fact that the first derivative of some kernel functions goes to zero as inter-particle spacing decreases, the repulsive force becomes trivial resulting in unrealistically clumping. Monaghan (2000) showed how this kind of instability can be removed by introducing an artificial stress which, in the case of fluid, it is an artificial pressure. Monaghan added the artificial pressure as

$$\frac{p_k}{\rho_k^2} = \frac{p_k}{\rho_k^2} \left(1 + R_k f^n \right) \quad (2-58)$$

where k represents the particle i or j in the momentum equation and constant n is usually 4. A suitable form for f which increases as the particle separation decreases is

$$f = \frac{W(r/h)}{W(\Delta p/h)} \quad (2-59)$$

in which Δp is the distance at which the first derivative of the kernel function reaches the local minimum and r is the present inter-particle spacing. For instance, the cubic spline function has a minimum at $q=r/h=2/3$, thus $\Delta p = 1/1.5$. Monaghan (2000) suggests $\Delta p = 1/1.3$. The factor R is calculated

$$R_k = \begin{cases} 0.006 \frac{P_k}{\rho_k^2} & P_k > 0 \\ 0.6 \frac{P_k}{\rho_k^2} & P_k < 0 \end{cases} \quad (2-60)$$

where $R = R_i + R_j$ and finally, the momentum equation becomes

$$\frac{d\mathbf{v}_i}{dt} = -\sum_j m_j \left(\left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) (1 + c f_{ij}^4) \right) \nabla_i W_{ij} + viscous \text{ term} + g \quad (2-61)$$

2.7.12 Kernel Normalization

Since in SPH hydraulics problems the domain is finite and a free surface exists, a correction of the kernel function W is necessary due to lack of particle. The kernel function of particles near the boundaries and free surface is truncated due to the absence of neighboring particles and no longer, the integral of W inside its influence domain is one. However, this issue can be treated by correcting the kernel function W itself or its gradient or both of them. A couple of corrective terms for fixing the mentioned errors are available in the literature [Randles and Libersky, 1996; Belytschko et al., 1998; Bonet and Lok, 1999; Vila, 1999; Chen and Beraun, 2000; Bonet and Kulasegaram, 2000]. Following sections will introduce the most commonly used techniques to avoid errors.

2.7.12.1 Kernel Correction

A kernel correction was proposed by Liu et al. (1997) and alternatively, introduced by Bonet and Lok (1999). Bonet and lok (1999) consider that the linear correction is unsuitable for computational purpose. They suggest using constant correction rather than linear one. By constant correction a vector can be expressed as

$$\mathbf{f}_i = \frac{\sum_j \frac{m_j}{\rho_j} \mathbf{f}_j W_{ij}}{\sum_j \frac{m_j}{\rho_j} W_{ij}} \quad (2-62)$$

2.7.12.2 Kernel gradient correction

Bonet and Lok (1999) proposed the kernel gradient correction to ensure that the gradient of a velocity field is correctly evaluated. In this method, the corrected kernel gradient $\tilde{\nabla} W_{ij}$, Eq. (2-63), should be used in momentum equation to find the forces on each particle instead of the regular kernel gradient ∇W_{ij} .

$$\tilde{\nabla} W_{ij} = \mathbf{L}_i \nabla W_{ij} \quad (2-63)$$

For the sake of clarity, consider a 2D medium the kernel gradient is

$$\begin{bmatrix} \tilde{\nabla} W_x \\ \tilde{\nabla} W_z \end{bmatrix} = \begin{bmatrix} L_i(1,1) & L_i(1,2) \\ L_i(2,1) & L_i(2,2) \end{bmatrix} \begin{bmatrix} \nabla W_x \\ \nabla W_z \end{bmatrix} \quad (2-64)$$

$$\mathbf{L}_i = \mathbf{M}_i^{-1} \quad (2-65)$$

$$\mathbf{M}_i = \sum_j^{np_i} \frac{m_j}{\rho_j} \nabla W_{ij} \otimes (\mathbf{x}_i - \mathbf{x}_j) \quad (2-66)$$

Where np_i is the number of particles interacting with particle i . More details on constructing matrix M is expressed in Gomez-Gesteira et al. (2010). Note that both matrices M and L are identity when particle i is positioned far from any kinds of boundary including free-surface and there is no correction on the kernel gradient. On the contrary, around where the distribution of particles is not symmetric anymore, both M and L are not identity matrix anymore. Frankly, the SPH property of conserving the momentum fails when any operation of kernel gradient correction is employed.

Gomez-Gesteira et al. (2010) also explained the combination of kernel and kernel gradient correction briefly.

2.7.13 Time Integration

Four numerical schemes, (1) Predictor-Corrector, (2) Verlet (3) Sympletic (4) Beeman, are implemented in current SPHysics to develop the solution of the SPH equations in time. Since the spatial accuracy is in the order of $O(h^2)$, it is recommended to use at least second-order accurate scheme in time. Beeman is a fourth-order scheme. Since only Predictor-Corrector and Symplectic are used in this study, these two algorithms are briefly described here.

2.7.13.1 Predict-Corrector scheme

At the predict level, the parameters are evaluated in time as

$$\left\{ \begin{array}{l} \rho_i^{n+1/2} = \rho_i^n + \frac{\Delta t}{2} \Delta \rho_i^n \\ \mathbf{v}_i^{n+1/2} = \mathbf{v}_i^n + \frac{\Delta t}{2} \mathbf{F}_i^n \\ \mathbf{r}_i^{n+1/2} = \mathbf{r}_i^n + \frac{\Delta t}{2} \mathbf{v}_i^n \end{array} \right. \quad (2-67)$$

Then pressure at mid-time is computed by the equation of state using the density at $n+1/2$. Then these values are corrected using the forces at the mid step

$$\left\{ \begin{array}{l} \rho_i^{n+1/2} = \rho_i^n + \frac{\Delta t}{2} \Delta \rho_i^{n+1/2} \\ \mathbf{v}_i^{n+1/2} = \mathbf{v}_i^n + \frac{\Delta t}{2} \mathbf{F}_i^{n+1/2} \\ \mathbf{r}_i^{n+1/2} = \mathbf{r}_i^n + \frac{\Delta t}{2} \mathbf{v}_i^{n+1/2} \end{array} \right. \quad (2-68)$$

At the end of time step, the values are calculated following:

$$\left\{ \begin{array}{l} \rho_i^{n+1} = 2\rho_i^{n+1/2} - \rho_i^n \\ \mathbf{v}_i^{n+1} = 2\mathbf{v}_i^{n+1/2} - \mathbf{v}_i^n \\ \mathbf{r}_i^{n+1} = 2\mathbf{r}_i^{n+1/2} - \mathbf{r}_i^n \end{array} \right. \quad (2-69)$$

Accepting a small error, Monaghan (1989) used the midpoint value of previous time step instead of computing the value at step n , which saves time. Note that \mathbf{V}_i represent the XSPH velocity of particle i , Eq. (2-54).

2.7.13.2 Symplectic scheme

In the absence of dissipation (friction or viscosity), this scheme preserves the properties of the Lagrangian equations. This method is known as kick-drift-kick form of time integration in which drift is the change in particle position and kick is the change in velocity due to the forces. In this method, the density and position acceleration at the middle of time

step is firstly calculated

$$\left\{ \begin{array}{l} \rho_i^{n+1/2} = \rho_i^n + \frac{\Delta t}{2} \Delta \rho_i^n \\ \mathbf{r}_i^{n+1/2} = \mathbf{r}_i^n + \frac{\Delta t}{2} \mathbf{v}_i^n \end{array} \right. \quad (2-70)$$

then $P^{n+1/2}$ is computed by EOS using $\rho^{n+1/2}$. In the next step, $\mathbf{F}_i^{n+1/2}$, the velocity acceleration, is obtained by a sweep over the particles. Finally the velocity and position of each particle are predicted at the end of time step

$$\left\{ \begin{array}{l} \mathbf{v}_i^{n+1} = \mathbf{v}_i^{n+1/2} + \frac{\Delta t}{2} \mathbf{F}_i^{n+1/2} \\ \mathbf{r}_i^{n+1} = \mathbf{r}_i^{n+1/2} + \frac{\Delta t}{2} \mathbf{v}_i^{n+1} \end{array} \right. \quad (2-71)$$

Using \mathbf{v}_i^{n+1} and \mathbf{r}_i^{n+1} , $\Delta \rho_i^{n+1}$ is calculated and then the pressure is computed by ρ_i^{n+1} .

2.7.13.3 Variable time step

A suitable time step ensures that the particles do not penetrate to the neighbors and the time-integration is stable. Three different parameters control the time-step, internal and external forcing terms, CFL (Courant-Fredrich-Levy condition) and viscous diffusion term [Monaghan, 1989 and Monaghan et al., 1999]. The variable time step is calculated according to

$$\Delta t = CFL \cdot \min(\Delta t_f, \Delta t_v) \quad (2-72)$$

where the forcing term, Δt_f , and viscous term, Δt_v , are

$$\Delta t_f = \min \sqrt{h / |f_i|} \quad (2-73)$$

$$\Delta t_v = \min \frac{h}{c_s + \max \left| \frac{h \mathbf{v}_{ij} \cdot \mathbf{r}_{ij}}{\mathbf{r}_{ij}^2} \right|} \quad (2-74)$$

In the case of parallel computing, the minimum time-step of whole domain is used in all processors.

2.8 Boundary Condition

Two general types of boundary condition have been implemented in SPHysics, (i) solid wall boundary and (ii) periodic open boundary. Despite all advantages of using SPH and its widespread application, the appropriate modeling of boundaries is not still clear enough [Crespo et al., 2007]. In the SPH formulations, solid walls do not appear in a natural way. To model any object in SPH the condition of no particle penetrating into the impermeable boundaries should be satisfied. Close to any fixed boundary, fluid velocity falls to zero while pressure and density do not. Some virtual force should be provided to create the limits of system and keep all fluid particles inside that domain. Several techniques are now available in literature about simulating the walls, e.g. ghost particles [Randles and Libersky, 1996; Colgarossi and Landrini, 2003], dynamic boundary condition [Gomez-Gesteira and Dalrymple, 2004; Crespo et al., 2007], Repulsive force [Monaghan and Kos, 1999; Monaghan and Kajtar, 2009]. Each method has its advantages and disadvantages. The idea behind each method and its formulation will be introduced in detail here.

2.8.1 Repulsive force

In this case the particles that constitute the boundary stop fluid particle by exerting forces on them, in analogy with the central forces among molecules. Several formulas have been proposed as repulsive force in SPH.

2.8.1.1 Lennard-Jones boundary force

The Lennard-Jones [Monaghan, 1994] is a well-known equation used in MD (Molecular Dynamics) to find the central force between a pair of particles. Using this theory the fixed boundary particles repel incident fluid particle with a central force proportional to the distance between the particles, given that the distance between them is less than initial particle spacing, r_0 .

$$LJ \text{ - Force} = K_{LJ} \left(\left(\frac{r_0}{r} \right)^{p_1} - \left(\frac{r_0}{r} \right)^{p_2} \right) \quad (2-75)$$

Monaghan (1994) suggested a magnitude of K_{LJ} as $5gd$, where g is the acceleration of gravity and d is a characteristic water depth.

2.8.1.2 Monaghan and Kos boundary Force

This formula, developed by Monaghan and Kos (1999) and modified by Rogers et al. (2008), exert a force on fluid particles normal to the boundary. Figure 2-8 illustrates a fluid particle (green dot) close to boundary particles (red dots). In Monaghan-Kos approach only particles 1 and 2 exert force on the fluid particle because it locates between those two boundary particles.

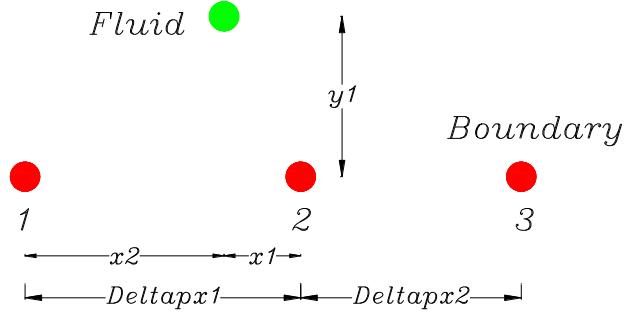


Figure 2-8- Monaghan-Kos repulsive force formulation

This force/unit mass (\mathbf{f}) is given by

$$\mathbf{f} = \mathbf{n} R(y) P(x) \varepsilon(z, u_{\perp}) \quad (2-76)$$

where \mathbf{n} is the unit normal vector. $P(x)$ is to ensure that a fluid particle experiences constant force as it moves between two adjacent boundary particles. It does not change the maximum value of force on a particular fluid particle. $P(x)$ is computed by

$$if \quad x < \Delta p_x \quad P(x) = \frac{1}{2}(1 + \cos(\pi x / \Delta p)), \quad (2-77)$$

else $P(x)=0$. Hence, $P(x)$ has a value between 0 and 1. $R(y)$ is presented by

$$if \quad q < 1 \quad R(y) = A \frac{1}{\sqrt{q}}(1 - q), \quad (2-78)$$

else $R(y) = 0$. q is equal to $y/(2\Delta p)$ where Δp is the initial particle spacing and y is the normal distance between fluid particle and the boundary surface. A in (1-1) affects the maximum value of force from a boundary particle on the fluid one, and the rest of equation increases from zero as q decreases. Parameter A has a dimension of acceleration

(velocity²/length). This equation should prevent the maximum velocity of the whole domain (v_{\max}). On the other hand, the sound speed is defined as, $c \geq 10 v_{\max}$ (or $v_{\max} \leq 0.1c$), so, A is presented by [Monaghan and Kos, 1999]

$$A = \frac{0.01 c_i^2}{h} \quad (2-79)$$

Finally, the function $\varepsilon(z, u_{\perp})$, suggested by Rogers et al. (2008), adjust the magnitude of the force regarding the normal velocity of water particle to the boundary and the local water depth

$$\varepsilon(z, u_{\perp}) = \varepsilon(z) + \varepsilon(\perp) \quad (2-80)$$

where

$$\varepsilon(z) = \begin{cases} 0.02 & z \geq 0 \\ |z/d| + 0.02 & -d \leq z \leq 0 \\ 1 & |z/d| > 1 \end{cases} \quad (2-81)$$

and

$$\varepsilon(u_{\perp}) = \begin{cases} 0 & u_{\perp} \geq 0 \\ \frac{20u_{\perp}}{c_s} & |20u_{\perp}| < c_s \\ 1 & |20u_{\perp}| > c_s \end{cases} \quad (2-82)$$

In which d is the local water depth and z is the elevation above the still-water level. u_{\perp} , the component of fluid velocity in normal direction to the boundary surface, is expressed

$$u_{\perp} = n \cdot (\mathbf{v}_{FP} - \mathbf{v}_{BP}) \quad (2-83)$$

Although this method is suitable for any flat surfaces even with sharp edges, it is not an easy task to use for curvature, any concave or convex surfaces (or line in 2D). To fix this problem, Monaghan and Kajtar (2009) proposed a new method.

2.8.1.3 Monaghan and Kajtar boundary Force

The particle boundary force proposed by Monaghan and Kajtar (2009) is another boundary force in which the summation of all central forces from boundary particles on a fluid particle determines the normal force on the fluid. In this method it is not needed to find the local normal vector of boundary. They suggested that if the distance between boundary particles is about 3-4 times less than that of initial spacing of fluid particles, the resultant of all central forces from boundary particles on the fluid one is normal to the surface and the tangential force is relatively zero. The force/mass on fluid particle j due to boundary particle k is presented by

$$\mathbf{f}_{jk} = \frac{K}{\beta} \frac{\mathbf{r}_{jk}}{r_{jk}^2} W(r_{jk}/h) \frac{2m_k}{m_j + m_k} \quad (2-84)$$

Monaghan and Kajtar suggested $K=0.01c^2$, where c is the sound speed. β ensures that by changing the space of boundary particle (or the ratio of boundary and fluid particle spacing) the force on a fluid particle is invariant. Here, W is the 1D cubic/quintic Wendland Kernel. In the present study the 1D quintic Wendland kernel is used,

$$W(q) = \frac{1.77}{32} \begin{cases} (1 + 2.5q + 2q^2)(2 - q)^5, & 0 \leq q \leq 2 \\ 0, & q > 2 \end{cases} \quad (2-85)$$

in which $q = r_{jk} / h$. This method has its own negative points. First, the number of boundary particles should be more than normal to have reasonable accuracy. It, consequently, increases the simulation time which is not our desire. Second, this formula is very sensitive to the distance of particles. For short distance between fluid and boundary particle, huge force is created to stop particle penetration. This fact causes big forces at initial time and significant particle shooting can be occurred close to the boundaries. To fix this problem Monaghan and kajtar suggested a damping (or relax) time. During this time, which is typically about 0.5-1 second, viscosity is set higher than normal value. With this period, the whole system has enough time to reach the static and stable situation. In fact the simulation starts after damping time. Similarly, this additional time increases the total simulation cost.

2.8.2 Dynamic Particles

This method was first presented by Dalrymple and Knio (2000). Later, it was used to study the interaction of fluid-coastal structure [Gomez-Gesteira and Dalrymple, 2004; Gomez-Gesteira et al., 2005]. The same equation of continuity and state, as the fluid particle verify, are solved for these dynamic particles. Since their position remains unchanged or is externally imposed, they are not involved in conservation of momentum. Simple implementation is one of the most interesting advantages of this method, since they can be calculated inside the same loops as fluid particles. All particles are considered as fluid particle and once the initial condition is introduced to the code, no extra consideration is needed. Commonly, boundary particles are located in staggered form, Figure 2-9.

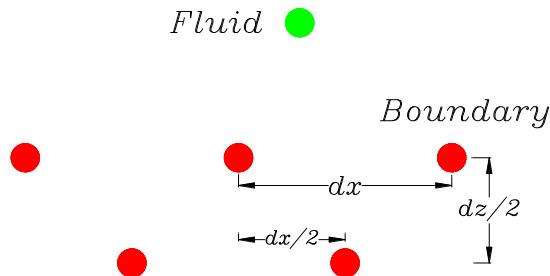


Figure 2-9- Dynamic boundary particle arrangement

Fluid particle penetration into the boundary is the most negative property of dynamic boundary particles (DBPs). When a fluid particle approaches the boundary fast enough, this method may not stop it completely. There are two ways to reduce the possibility of penetration, choosing smaller time step (or smaller CFL number (<0.2) in variable time step) or define more layers of DBPs. Both options increase computational cost, finally. Regarding the test cases presented in the next chapter, DBPs give more reasonable results than repulsive force methods do.

When a fluid particle moves toward a boundary, the local density increases and consequently, the pressure increases regarding the equation of state. Conservation of momentum, (2-32), implies that increasing in pressure of particle results a central force which stops or repels particles. This force reduces the velocity of fluid particle in vicinity of boundary or bounces them back. The above process causes large fluctuation of pressure where solid boundaries exist.

None of above methods satisfies the no-slip boundary condition purely. In both dynamic particles and repulsive force methods, a wall viscosity is defined to simulate the no-slip condition, but this viscous force does not ensure zero velocity at boundary surface.

2.8.3 Ghost particles

Ghost boundary particle was proposed by Randles and Libersky (1996) to have more accurate pressure field in boundary neighborhood. This method is also more reliable for simulating no-slip condition than other methods. It is very similar to ghost cells used in finite difference calculations that are outside the fluid domain but alongside the actual boundaries. In this approach, the properties of boundary particles including their positions are dictated by fluid particle condition at each time step. When the distance between a fluid particle and the boundary becomes less than the kernel smoothing length, then a ghost particle is created outside of the system, constituting the specular image of the incident one. The density and pressure of the virtual particle is equal to that of fluid one, while the normal velocity is in an opposite direction. To enforce free-slip and no-slip condition the tangential velocity of virtual particle is in the same and opposite direction of that of fluid one, respectively [De Leffe et al., 2011]. Figure 2-10 shows the velocity vector of standard ghost particles in both free-slip and no-slip condition.

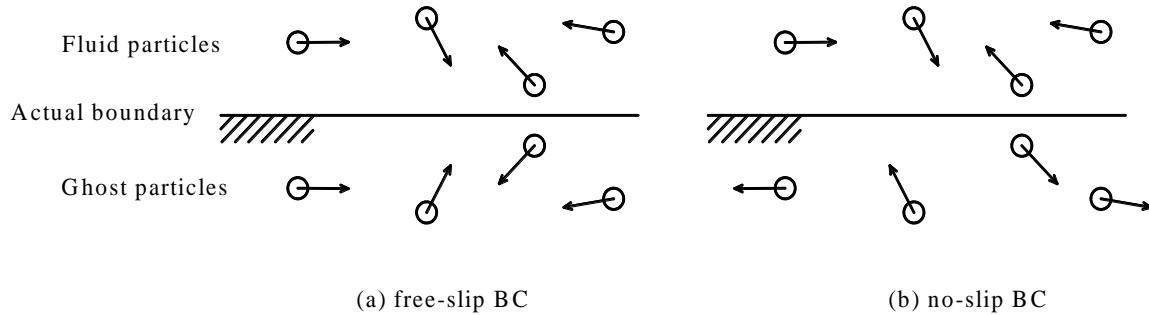


Figure 2-10- Ghost particle boundary condition (a) free-slip condition, (b) no-slip condition

In this way, the normal component of fluid velocity in both cases is zero as required and the tangential component has zero value and non-zero value for no-slip condition and free-slip condition, respectively. Since the pressure (or density) of ghost boundary particles is equal to the corresponding fluid ones, the pressure field in vicinity of solid walls is much less fluctuated and more smoothed than other approaches. Following above statement, the Neumann boundary condition is enforced truly, i.e., $\partial p / \partial n = \rho \mathbf{g} \cdot \mathbf{n}$ where \mathbf{n} is the normal unit vector to the solid boundary and \mathbf{g} is the body force field. While in other methods, even dynamic boundary particles, the Neumann boundary condition is not necessarily satisfied.

Ghost particle method has two major disadvantages which complicate its implementation in the code. First, due to the uncertain location and motion of fluid particles with respect to solid boundaries, the number of ghost particles varies in each time step. Second, where the boundary has sharp edges, such as saw teeth or w shape, the location of ghost particles is not easy to compute. To improve these two mentioned drawbacks, the fixed ghost particles method was proposed by Marrone et al. (2011).

In this new technique, ghost particles are not created at each time step anymore, while they are built only once at the initial time and are fixed in the frame of reference of the boundary [Marrone et al., 2011]. To place the fixed ghost particles, the body nodes discretized the body boundary into piecewise spline. Then the outward normal and tangent unit vector can be analytically computed. Those body nodes are spaced with a prescribed distance, ds . The first layer of fixed ghost boundary particles are created by duplicating the body nodes out of the fluid domain in the normal direction, at a distance of $ds/2$ from the body boundary. Then this new spline is duplicated at the distance of ds in the normal direction to build the second spline. The nodes along these lines are fixed ghost particles, see Figure 2-11.

One of the robust privileges of using fixed ghost particle instead of the standard ones is that their arrangement is always uniform and fixed with respect to the body surface and does not depend on the fluid particle position. This feature facilitates modeling of complex geometries [De Leffe et al., 2011].

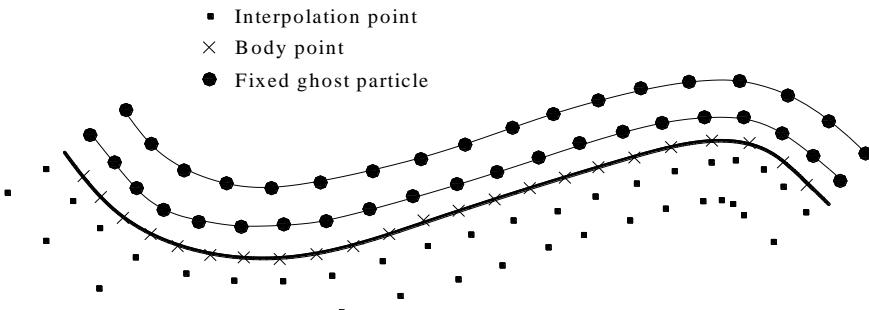


Figure 2-11- fixed ghost particle, interpolation points and body nodes

To compute the quantities, density or pressure, at each ghost particle, an interpolation point is assigned to it, Figure 2-11. The position of these interpolation points is found by monitoring a ghost particle into the fluid domain. By a moving-least interpolation of the fluid particle values, the interpolation point quantities are evaluated at each time step. In this way, both Dirichlet and Neumann condition are possible to enforce. The pressure of a ghost particle (p_G) using an MLS interpolated is calculated as

$$p_G = \sum_{j \in fluid} p_j W^{MLS}(r_j) V_j + 2d_{GA}\rho g \cdot n \quad (2-86)$$

where d_{GA} is the distance between the ghost particle and actual body surface and W^{MLS} can be computed by equations (2-20)-(2-24). Then the density of ghost particle is obtained by the equation of state. In addition, either free-slip or no-slip conditions can be easily implemented along solid boundaries. To ensure the boundary condition $\vec{u} \cdot \vec{n} = 0$, the normal component of the fluid velocity is reversed during the mirroring procedure. To enforce no-slip and free-slip condition, the tangential component of velocity should be reversed and unaltered, respectively.

2.8.4 Periodic BC (inflow/outflow condition)

Open boundaries can be simulated through periodic lateral boundary condition. In this type of boundary, particles near the lateral boundaries interact with those on the complementary open boundary on the other side of domain. Regarding Figure 2-12, when particle i locates near an open boundary, its kernel support extends beyond the lateral boundary.

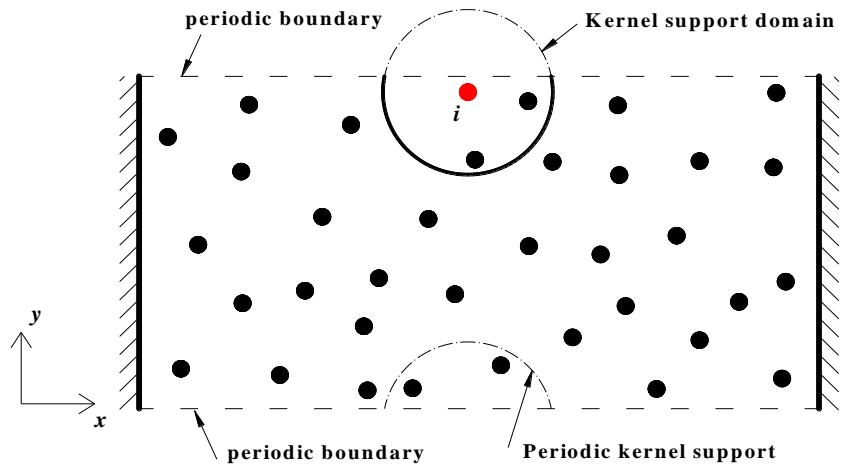


Figure 2-12- Periodic lateral boundary (plan view)
The kernel support of particle i is continued through periodic boundary

With periodic boundaries, the kernel support domain of a particle is continued through the lateral boundary so that it interacts with other particles within its influence domain but on the other side of the domain, Figure 2-12. In this way, if a particle leaves the domain from one lateral side it appears or enters from the other side.

Periodic boundary is one of the techniques that can help to model inflow and outflow in SPH. For instance, if generating current in y direction is desired (Figure 2-12), a pressure gradient or a virtual body force impulses particles to move in this direction. When a particle leaves the domain from the top side, it is replaced from the bottom boundary with the same x position. Any restriction for density, pressure or velocities can be considered at inflow or outflow. This approach of modeling inflow and outflow is much more straightforward than other existing methods, such as creating particles at inflow and deleting those which leave

the domain from the outlet [Mahmood et al., 2011]. Morris et al., 1997 and Safdari Shadloo et al., 2011 used periodic boundary method to create inflow and outflow in their study.

2.9 Nearest Neighboring Particle searching (NNPS)

Regarding to the fact that kernel function has a compact support, not all particles interact with each other. Only a finite number of particles within the support domain of a particle interact with it and are used in the particle approximations. Generally, these particles are called nearest neighboring particles (NNP) of the interested particle and the process of finding the NNP is referred to as nearest neighboring particle searching (NNPS). Since the position of particles varies through the time, this process should be conducted every step. Generally, three approaches are popular in SPH for NNPS, all-pair search, linked-list search algorithm and tree search algorithm.

All-pair search is the easiest and slowest technique in which the distance r_{ij} from i to other particles j is computed. If this distance is smaller than the kernel support dimension, particle j belongs to the influence domain of i . If constant smoothing length is defined for all nodes, then, clearly particle i is in the support domain of j . Then all equations are solved for this pair of particles. The number of operations in this method is in the order of $O(N^2)$, N being the number of particles. According to the fact that this process is essential at each time step, this approach is not efficient at all for the problem with huge number of particles.

Figure 2-13 presents all-pair approach schematically.

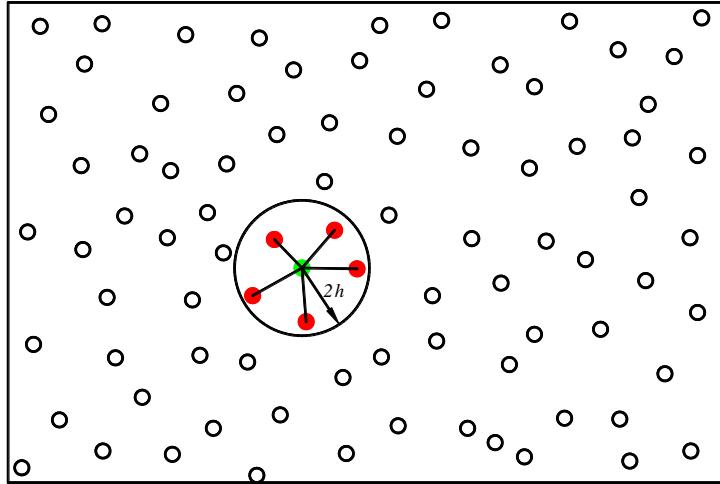


Figure 2-13- All-pair search algorithm for NNPS in 2D

2.9.1 Linked-list

Since all-pair method is not suitable for large problems with huge number of particles, Linked-list method was proposed by Monaghan and Gingold (1983). Monaghan and Gingold noted that linked-list technique works well when spatially constant smoothing length is used and it saves remarkable computational time. In 2D (or 3D) SPHysics, the whole domain is divided in squares (or cubes) cells of side $2h$ following [Monaghan and Lattanzio, 1985 and Gomez-Gesteira et al., 2009]. Hence, for searching the nearest neighboring particles of a given particle inside a cell, only the interaction with the particles of neighboring cells are considered. Since the smoothing length is $2h$, choosing this value as the size of cells guarantees that all particles inside the cells beyond neighboring cells are not in its support domain. In comparison with all-pair approach, this technique diminishes the computational

time remarkably and reduces the number of calculations per time step from N^2 operations to $N \log N$. Figure 2-14 illustrates the underlying cells.

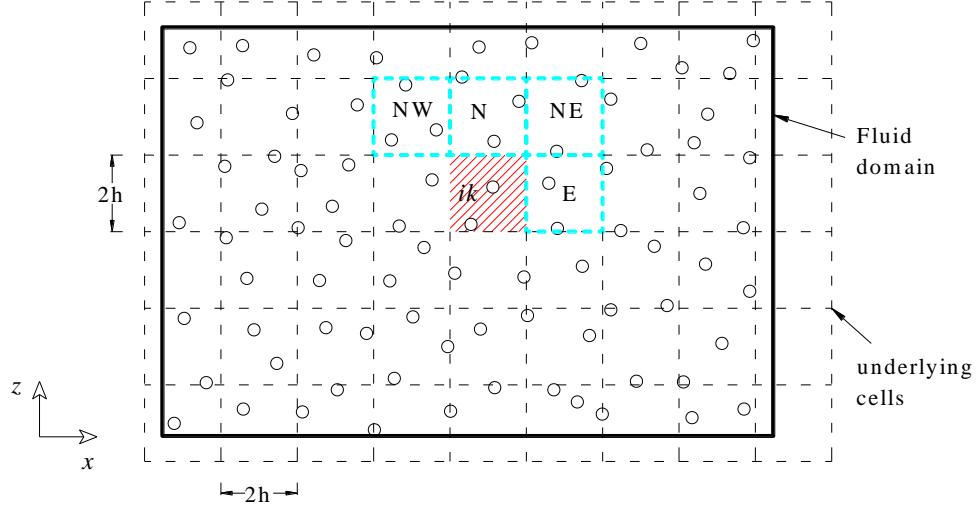


Figure 2-14- Linked-list algorithm of NNPS

For each z level, the 2D-SPHysics sweeps through the cells along the x -direction. Around each cell, such as cell ik in Figure 2-14, only 4 cells of E, NE, N and NW are checked to eliminate repeating the particle interactions. If the distance between any particle in these cells and a specific particle in cell k is less $2h$, then all equations are solved for this pair of particles. If above procedure is done for all cells in the whole domain, all cells are linked to their neighbor cells. Although one cell does not search inside the west or south-west cells in the explained process, the west cell searched and checked its surrounding cells at the previous steps. For instance, the cell $i=5$ and $k=4$ does not check the cell $i=4$ and $k=3$ because the cell (4,3) checked its north-west cell, which is $i=5$ and $k=4$, in its searching process. A very similar protocol is employed in 3D version of SPH.

As stated above this approach is very suitable when constant smoothing length is applied. As an example, if kernel length is defined as $h=(1.4)(dp)$, the typical number of particles in a square (or cube) cell is about 8 (or 22) in 2D (or 3D). Variable- h , discussed later, is an interesting tool to avoid large number of particles in 3D where small particle size is required for higher accuracy. In many cases we do not need high resolution and high accuracy everywhere in the fluid domain but only in a specific region. As an example, if the interested study is to investigate the wave impact on an object, higher accuracy is only crucial for the area around the object and high resolution is not necessary for the location far from it. Therefore, finer particle is defined close to the interested object.

2.9.2 Tree Search

Tree search algorithm works well for problems with variable kernel length. Ordered trees are created according to the particle positions. Once the tree structure is defined, it is used to find the NNP efficiently. The complexity of this method is in the order of $N(\log N)$. A detailed description of tree search algorithm can be find in Hernquist and Katz (1989) and Liu and Liu (2007). Since programing of this algorithm is more complicated than linked-list method and it is not as compatible as linked-list with current SPHysics structure, some people prefer to use linked-list instead of tree search even if variable- h is applied. Oger et al. (2006) employed linked-list technique in their study where variable- h method is used to simulate wedge water entries in 2D.

If liked-list is chosen for variable- h technique, the maximum h in the whole domain should be used to build the underlying squares (or cubes in 3D). Consequently, the number of

particle inside a square or cube becomes higher than normal. Following the above example, if particle size varies from $dp=0.03$ to $dp=0.05$, the cube size will be $(2)(1.4)(0.05)= 0.14$. The number of finer particle inside a cube is approximately equal to 100, which is 5 times more than normal. This increases the time of nearest neighbor particle search. One solution for reducing the computational time is to use different h for each processor in parallel computing.

2.10 Floating Objects

As discussed in chapter 1, our main goal is to have a trustable model to simulate any objects, specially, wave energy convertors. It is also mentioned that many of these devices include a floating part, such as PowerBuoy, Pelamis or Oyster. So, it is necessary to explain how floating objects are modeled in SPH. In SPH in general, a boundary is defined as a set of particles that exert force on fluid particles. Oppositely, water particles also exert the same force in opposite direction on the boundary particles. The summation of all force on the entire body will express the motion of the object using rigid body dynamics formulation [Rogers et al., 2010; Omidvar, 2010]. Following Monaghan et al. (2003) for modeling the floating objects, the objects are treated as rigid body. Therefore, the linear and angular momentum of the object will explain the linear motion and rotation of it. The force on each boundary particle is calculated by summing up all contributions from all surrounding fluid ones inside its kernel support area. The force per unit mass experienced by a boundary particle k is computed by

$$\mathbf{f}_k = \sum_{i \in WP} \mathbf{f}_{ki} \quad (2-87)$$

where WP denotes water particles and \mathbf{f}_{ki} is the force per unit mass from water particle i on boundary particle k . Then the force exerted by a water particle on a boundary particle is given based on the principle of equal and opposite action and reaction,

$$m_k \mathbf{f}_{ki} = -m_i \mathbf{f}_{ik} \quad (2-88)$$

Equation (2-88) facilitates determining the force on an object using the calculated force from the boundary on the fluid. Then, the total force on the rigid body is computed by

$$\mathbf{f}_{total} = \sum_{k \in BP} m_k \mathbf{f}_k \quad (2-89)$$

Regarding the basics of rigid body dynamics, the motion of the moving object in the transitional and rotational degree of freedom are presented by

$$M \frac{d\mathbf{V}}{dt} = \mathbf{f}_{total} \quad (2-90)$$

$$I \frac{d\Omega_I}{dt} = \sum_{k \in BPs} m_k (\mathbf{r}_k - \mathbf{R}_0) \times \mathbf{f}_k \quad (2-91)$$

in which I is the moment of inertia, \mathbf{V} is the velocity and Ω_I is the rotational velocity of the object whose direction is perpendicular to the object motion and \mathbf{R}_0 is the center of the object. Equations (2-90) and (2-91) are integrated in time to find the value of linear and angular velocity for the beginning of next time-step. Then the velocity of each boundary particle of rigid body is given by

$$\mathbf{v}_k = \mathbf{V} + \Omega_I \times (\mathbf{r}_k - \mathbf{R}_0) \quad (2-92)$$

Then the motion of each particle of floating body is predicted by integrating Eq. (2-92) in time. Monaghan et al. (2003) showed that this technique conserves both linear and angular momentum.

2.11 Computing Force on an Object

Fluid motion in vicinity of an object exerts force on it. Two general methods of calculating force exist. One is very similar to floating body calculation and the other one is using the fact that force is equal to pressure times the surface area.

In the first approach the total force on the object is predicted by Eq. (2-89). The summation of contributions from all water particles yields the final force on the body. In the second method, the global force on an object is given by

$$\mathbf{f}_{surf} = \int_{\partial \Omega_B} \mathbf{T} \cdot \mathbf{n} \, ds \quad (2-93)$$

where \mathbf{n} is the unit outward normal to the solid profile and \mathbf{T} is the stress tensor. The first method is employed in the current study.

2.12 Variable- h

All topics that have been introduced so far are based on the constant smoothing length assumption. Regardless of any criterion, the kernel length of each particle is always constant at everywhere. Under variable- h condition, the smoothing length of particles can be variable in space, in time or in both time and space. The first step in dynamic adaptivity was initiated

by Monaghan for astrophysical simulation, in which the density was the criterion to change the resolution. In their simulation h was constant in space but varied with time. The most common rules was to take

$$h \propto \frac{1}{\sqrt[\lambda]{\langle \rho \rangle}} \quad (2-94)$$

$$\langle \rho \rangle = \frac{\sum_j \rho_j}{n} \quad (2-95)$$

where λ is the number of dimensions and n is the number of particles around an specific point. In SPH, h determines the resolution and the number of neighbors contributing to the properties at a point. Therefore, the efficiency and accuracy will increase if h adapts according to the local particle number density [Monaghan, 1992]. Several authors have used local smoothing length to study and improve the formulation and accuracy of this extension of the SPH algorithm [Hernquist and Katz, 1989; Evrard, 1988; Benz et al., 1990]. Typically, the smoothing length is defined

$$\frac{dh_i}{dt} = - \left(\frac{h_i}{v\rho_i} \right) \frac{d\rho_i}{dt} \quad (2-96)$$

Momentum can be conserved if only the kernel is symmetric [Monaghan, 1992] which is not if variable smoothing length is applied. Consider two particles i and j have different smoothing length, h_i and h_j . Assuming $h_i > h_j$, it can be possible that particle j locate in support domain of particle i , while i is outside of j influence area. To conserve the momentum, the total from i on j should be equal to that of j on i . A symmetric kernel can be obtained by

defining any of standard kernels with h replaced by symmetric combination of smoothing length of a pair of particles. This combination can be the arithmetic mean, minimum (h_i, h_j) or maximum of (h_i, h_j). If the distance between two particles is less than the symmetric kernel length, then all equations are solved for them, otherwise it is assumed that they are not in their influence domain.

The kernel value can be computed in two ways. One is to calculate W_{ij} regarding the symmetric kernel length that is $W(r_{ij}, h_{\text{symmetric}})$. An alternative is to employ the average of two kernels, one with h_i and one with h_j .

$$\tilde{W}_{ij} = \frac{1}{2} \left(W(r_{ij}, h_i) + W(r_{ij}, h_j) \right) \quad (2-97)$$

Therefore, the discrete form of equation (2-96) becomes

$$\frac{dh_i}{dt} = - \left(\frac{h_i}{v\rho i} \right) \sum_j m_j \mathbf{v}_{ij} \cdot \nabla_i \tilde{W}_{ij} \quad (2-98)$$

Recently, Fledman and Bonet (2007) introduced a general procedure of particle refinement. In this approach, a set of daughter particles is replaced by particle i with a certain pattern centered at the position of i . According to particle replacement, the total number of particles varies during the time. Note that once particle i is replaced by daughter particles, they will not merge to recreate a coarser particle.

When the kernel length varies in space, some other terms need to be added to the equations. Nelson and Papaloizou (1994) presented the pressure force on particle i when smoothing length varies spatially.

$$\begin{aligned}
F_{P,i} = & - \sum_{j=1}^N m_i m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} \right) \frac{1}{2} \left[\frac{\partial W(r_{ij}, h_i)}{\partial r_i} \Bigg|_{h_i \text{constant}} + \frac{\partial W(r_{ij}, h_j)}{\partial r_i} \Bigg|_{h_j \text{constant}} \right] \\
& - \sum_{k=1}^N \sum_{j=1}^N m_k m_j \left(\frac{P_k}{\rho_k^2} \right) \frac{1}{2} \left[\frac{\partial W(r_{kj}, h_k)}{\partial h_k} \frac{\partial h_k}{\partial r_i} + \frac{\partial W(r_{kj}, h_j)}{\partial h_j} \frac{\partial h_j}{\partial r_i} \right]
\end{aligned} \quad (2-99)$$

The second term involving derivatives of the smoothing lengths appears due to the temporal and spatial variability of the h_i s. These types of terms are referred as $\vec{\nabla}h$ terms. A comprehensive momentum equation including viscous terms is described in Nelson and Papaloizou (1994). Studying energy conservation, they assessed the effect of neglecting or including $\vec{\nabla}h$.

In most cases we do not need high resolution and accuracy everywhere but only in an interest area. If high resolution is used in the whole fluid domain, it will lead to a very expensive computation. Generally, accuracy and fine resolution is necessary in the impact area. Hence, the particles are initially defined so that its size is small at impact area and bigger elsewhere. As discussed above, spatially and temporally variable h imply some additional terms including $\vec{\nabla}h$ and dh/dt in the standard formulation to satisfy the conservation laws. Oger et al. (2006) assume that each particle carries its own h which is constant in time. Therefore all terms with time derivative of h , dh/dt , are always equal to zero, whereas the terms involving $\vec{\nabla}h$ are not trivial since particles with different smoothing length are interacting during their Lagrangian evolution. For simplicity, they remove those terms from SPH formulation, instead, in order to reduce the errors in that study, the initial particle configuration is chosen so that the smoothing length varies slowly in space. In addition, the smoothing length at interest area is uniform locally. Note that to search for

nearest neighbor particles, the linked-list algorithm is applied as used in SPHysics. Figure 2-15 shows initial particle setup of Oger et al. (2006) study.

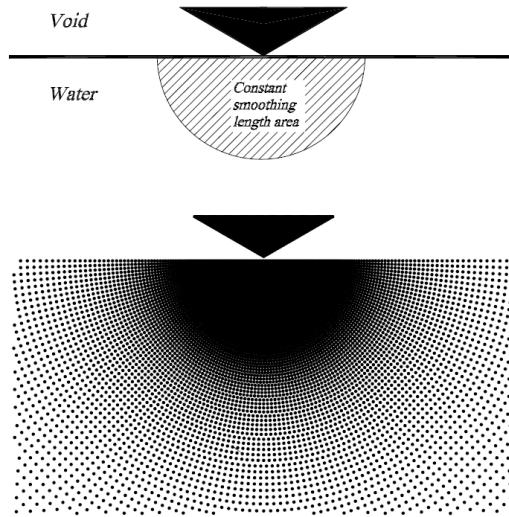


Figure 2-15- Initial particle configuration of two-dimensional simulation of wedge water entries [Oger et al., 2006]

Omidvar et al. (2011) proposed variable mass instead of spatially or temporally variable h or particle refinement at the impact area. A pre-defined variable mass distribution with a constant h is used with small particles at impact region and coarser ones for the rest of the domain. The coarse particles mass is m and they are typically replaced with four lighter ones with a mass of $m/4$ (in 2D), see Figure 2-16.

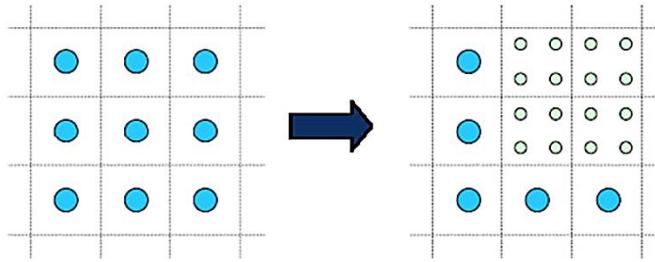


Figure 2-16- Omidvar et al. (2011) replaced a heavy particle with a mass m with four lighter ones with a mass of $m/4$ (in 2D)

This method has two major drawbacks which are not mentioned in their study. First is that using only lighter particle with large smoothing length does not improve the accuracy. As mentioned above, the magnitude of h plays the key role in SPH accuracy. Besides, as Monaghan (2000) states and regarding our personal experience, the best choice for the ratio of smoothing length and particle size, h/dp , is 1.3-1.5. Any value less than 1.2 decreases the particle mobility and any value above 1.8 causes unreal particle motion. Omidvar et al. (2011) define h as 1.3 dP , where dP is the size of coarser particles. Thus, the ratio of h/dp for lighter particles becomes 2.6. Note that in 2D, the size of a heavier particle is twice of a lighter one.

Second and probably the most important one is the cost of NNPS process. The underlying grid of linked-list process is built based on the smoothing length of larger particles. Thus, the number of particles within one square is more than normal (about 3 times greater in 2D and 8 times greater in 3D). This problem causes significant difference in computational cost. This difference becomes much more tremendous in 3D version. Although this approach reduces

the total number of particle and consequently the total computational cost, it takes long time for those small particles to find the nearest neighbor particles.

2.13 Dynamic Load Balancing (Processors border)

In parallel SPHysics, each processor has its own particles and each processor carries out all its jobs separately every time step. Then, all data transfers at the end of time step. Unless the problem is a simple rectangle or a tank completely full of particles, generally, when a problem is initialized, particles are not equally divided between all processors. Some processors have more or less number of particles than others, see Figure 2-17-a. For instance, a problem including a sloping beach has more particles offshore than inclined part or in a dam breaking problems, water is initially kept at one side of the domain while other part is completely empty.

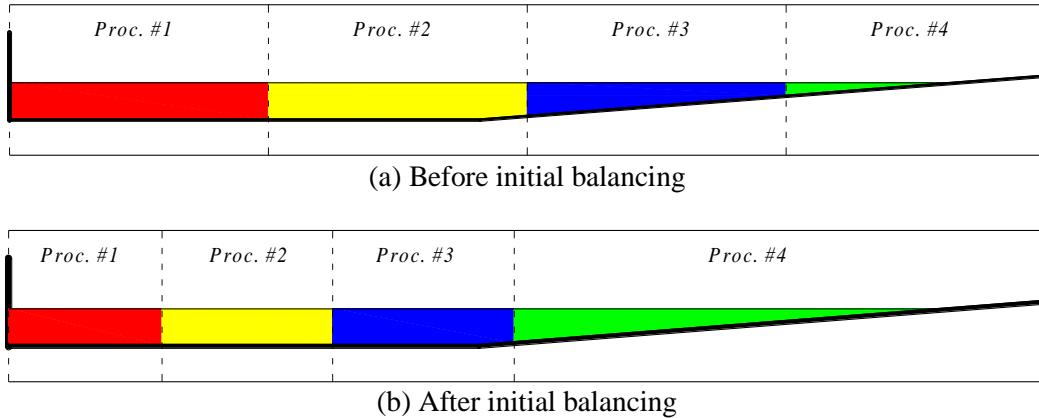


Figure 2-17- Dynamic load balancing for a beach problem.
Dashed lines present the border of processors

In addition, during the simulation, the number of particles allocated to each processor dynamically changes due to the motion of particles. Therefore, a dynamic load balancing algorithm is necessary to keep a roughly equal number of particles on each processor. The current parallel SPHysics applies a very basic form of load balancing which happens at two steps

- 1) Initially- before the main loop starts
- 2) Dynamically- during the simulation

The present method and the modified algorithm by author will be explained in Appendix A in detail.

Chapter 3

3 Validation

The accuracy and convergence of SPH for free-surface flows is examined by comparing its calculation with some existing experimental observations or analytical solutions. Several items including water surface, velocities, pressure and integrated force on an object are evaluated in these following tests.

3.1 Test 1: Dam Breaking

3.1.1 Test 1-1: Dam Breaking Impact on a Square Cylinder (in 3D)

Studying and understanding the influence of shore protection and vegetation on severe flooding event such as tsunami or hurricanes is important in protecting coastal area and communities [Rooney et al., 2011]. While the assessing of this effect can be complicated, numerical models aid engineers in sorting out the flow structure in presence of coastal vegetation. The first step is to investigate the flow dynamics around one vegetative stem using a numerical model. One technique to model one vegetative stem is to simulate flow

impact on a single obstacle. Yeh and Petroff [Raad P., Mitigation of local tsunami effects⁵] conducted an experimental study of a single bore impacting on square pile obstruction.

The experimenters referred this test as “bore in a box”, in which a dam break problem is confined within a rectangular box. Figure 3-1 shows the geometry of the problem from top and side view. The experimental tank was an open-top cube with 1.6 m long, 0.61 m wide and 0.75 m high. The volume of water behind the gate at one end of the box was initially 0.4 m long, 0.3 m high. The obstacle is a tall square column with dimensions of 0.12 m x 0.12 m x 0.75, located 0.5 m downstream of the gate and 0.24 m from one sidewall of the tank. This experiment is categorized as a dam break over a wet bed that is an initial layer of water (\approx 1 cm deep) exists on the bottom of the tank, since it was difficult to completely drain the tank downstream of the gate initially.

⁵ <http://lyle.smu.edu/waves/mitigation.html>

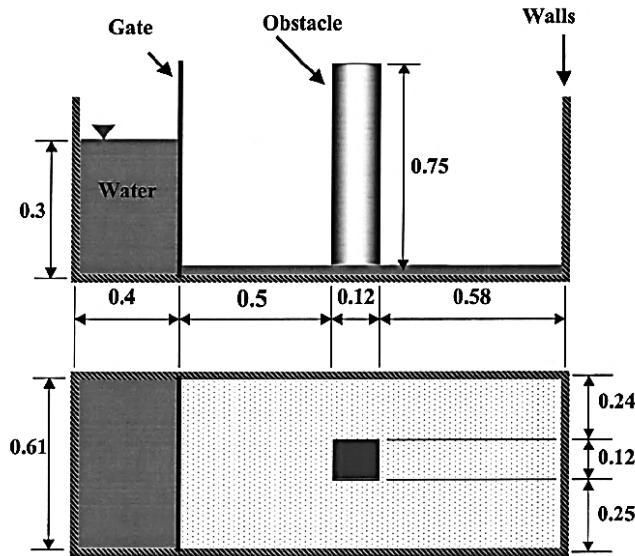


Figure 3-1- Experimental geometry: Side view (upper sketch), plan view (lower sketch) [Gomez-Gesteira and Dalrymple, 2004]

Experimental observation included the time history of the fluid velocity at a specific location and the net force on the structure. A laser Doppler Velocimeter (LDV) was located 14.6 cm upstream of the center of the obstacle and 2.6 cm off the bottom.

Previously, Raad at Southern Methodist University compared the experimental data with a numerical model using his ELMMC-3D (Three dimensional Eulerian-Lagrangian Marker and Microcell). The resulting wave force on the structure and velocities are in very good agreement with Yeh and Petroff experiment observation.

Gomez-Gesteira and Dalrymple (2004) simulated the same experiment by standard SPH formulation. They chose cubic spline as the kernel function and applied Monaghan artificial viscosity in the momentum equation with $\alpha = 0.01$. Tait's equation of state and XSPH presents the pressure and position of each particle. Dynamic boundary particles are applied as

boundary condition for all walls. Particle size is set to $dx=dy=dz=0.0225$ (m) and the constant smoothing length is $h=0.0331$ (m). The density of particles is initialized by

$$\rho = \rho_0 \left(1 + \frac{\rho_0 g (H - z)}{B} \right)^{\frac{1}{\gamma}} \quad (3-1)$$

where H = initial water depth and z = vertical distance from the bottom and $\gamma = 7$. All particles are initially placed in staggered pattern, such that the total number of particle is 34,000. They conclude that using SPH with above settings gives the flow velocity at a given position perfectly matched with experiment. In addition the 3D version of the model was able to reproduce the exerted force on a structure due to the collision of a wave with it.

A test was carried out by author with the same configuration as Gomez-Gesteira and Dalrymple (2004) used in their study.

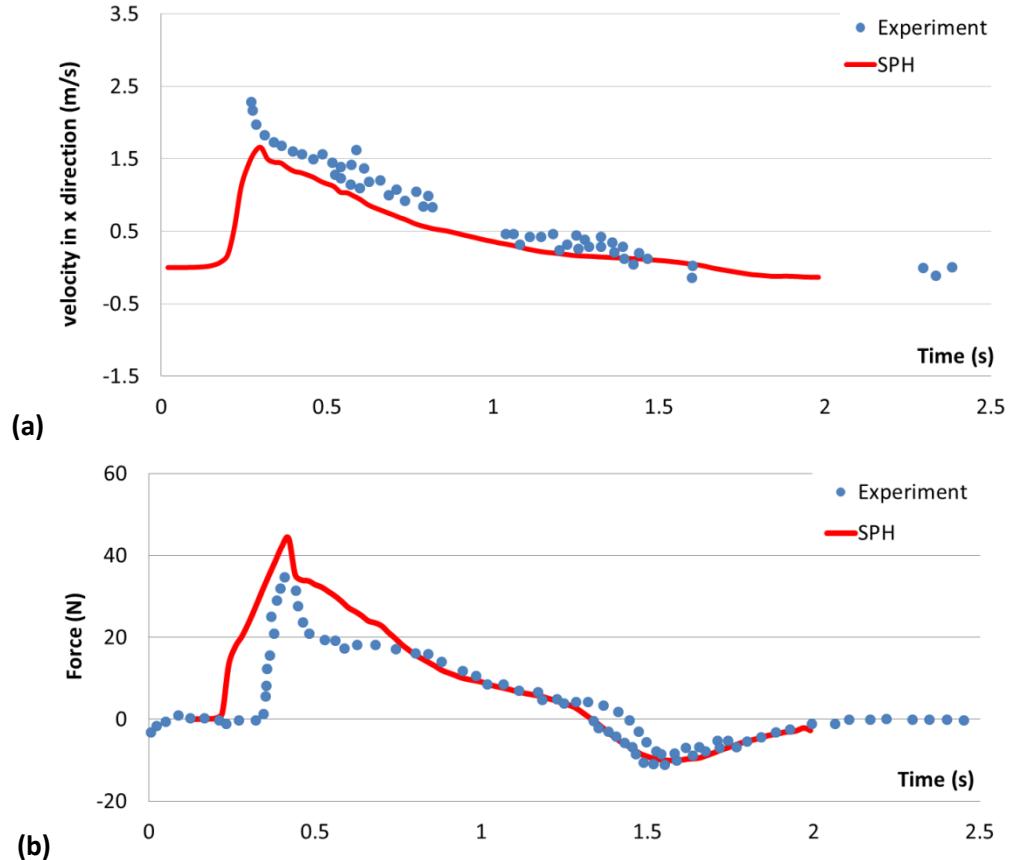


Figure 3-2-Comparison between experimental data (dots) and first-attempt SPH simulation ($dp= 0.0225$ m and using Dynamic particle BC) (solid line) by author. The horizontal velocity at the gauge installed in front of the obstacle (upper plot) and the net force on the obstacle (lower plot)

Figure 3-2-a shows that SPH method with given configuration underestimates the velocity specifically the maximum value at $t= 0.29$ s, while Gomez-Gesteira and Dalrymple output was perfectly fitted to the experiment data. Besides, the graph corresponding to the exerted force on the pile is not in good agreement for the time less than about 0.7 sec. Figure 3-2-b suggests that SPH closely predicts the net force for the moments greater than 0.7 sec. For the second try, the same problem was run with finer particle, $dp=0.01$ m. In this case, particles are not scattered in staggered form and they are normally located in cube form.

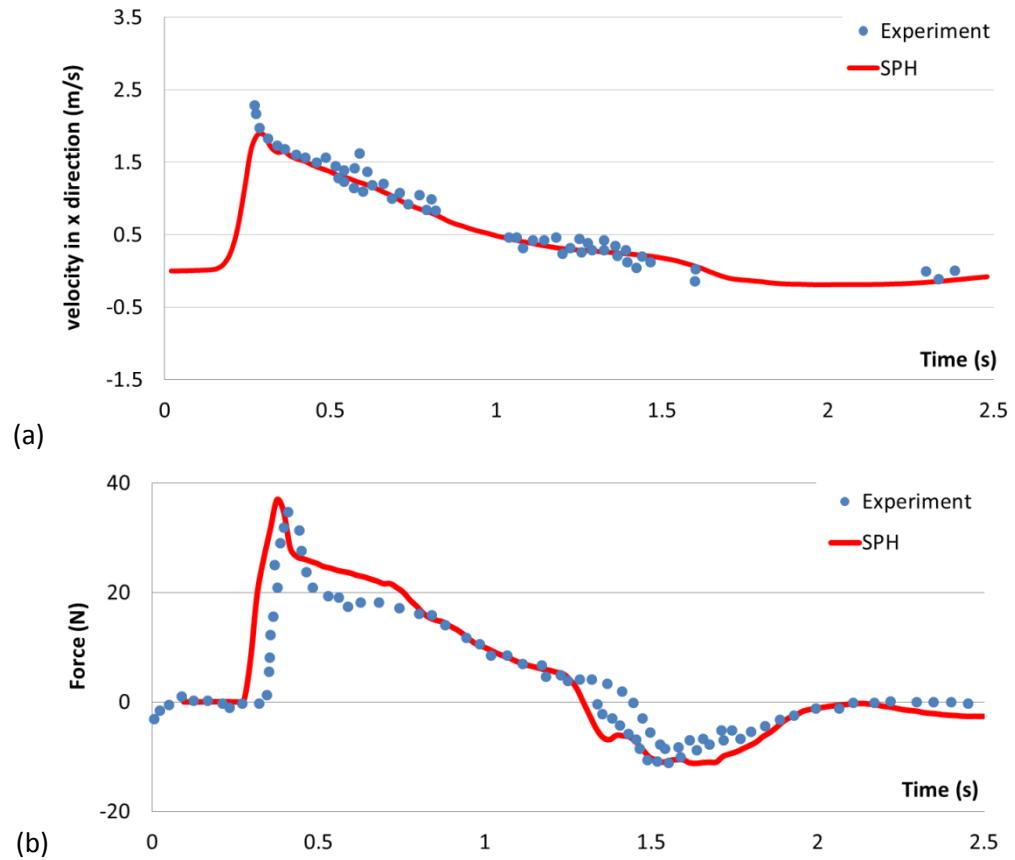


Figure 3-3- Comparison between experimental data (dots) and second-attempt numerical simulation ($dp=0.01$ m and using Dynamic particle BC) (solid line). The horizontal velocity at the gauge installed in front of the obstacle (upper plot) and the net force on the obstacle (lower plot)

Figure 3-3 denotes how increasing the resolution can improve the SPH results. Figure 3-3-a demonstrates that $dp=0.01$ m estimates both the maximum value of velocity and general shape of velocity graph much better than coarser particles ($dp=0.0225$ m). In addition, the predicted force on the structure is much more compatible with observation for the time less than 0.7 sec. SPH calculates the maximum force of impact on the column much closer to the experimental data when finer particle ($dp=0.01$ m) is employed.

Rooney et al. (2011) simulated the same experiment with GPUSPH. They validated the GPUSPH for this kind of free-surface flow and also determined the effects of particle size on the accuracy of output. In general, it is expected to have more accurate result with finer particles. Their study proved that this statement is not essentially true, although initial particle spacing impacts the accuracy of results. All walls including six sides of the obstacle, four sides of the tank and bottom were filled with boundary particles exerting Lennard-Jones boundary force as Monaghan, 1994 did. They tried various particle spacing in the range of 0.02 m to 0.0035 m. Figure 3-4 presents the velocity time history of their results for a couple of selected particle spacing.

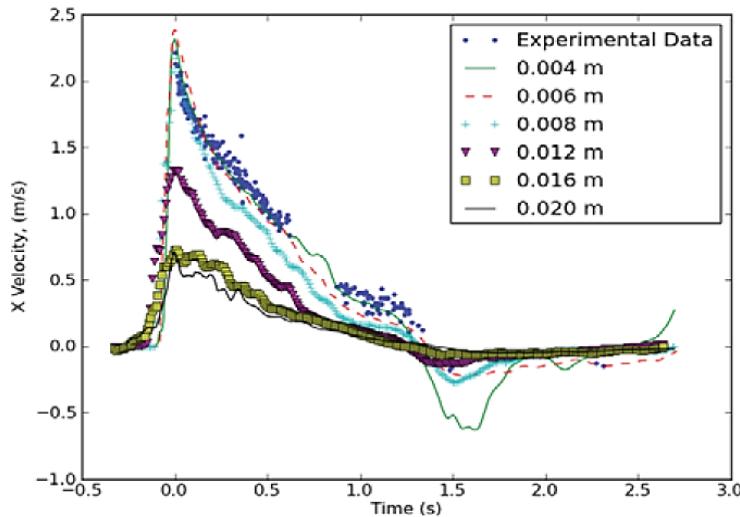


Figure 3-4- Velocity time series in the flow direction at interested point for selected particle size [Rooney et al., 2011]

Figure 3-4 proves that reaching the maximum velocity of 2.2 m/s is not feasible using $dp=0.02$ m as Gomez-Gesteira and Dalrymple (2004) used or even smaller size as

$dp=0.016$ m or $dp=0.012$ m. Furthermore, it implies that smaller particle causes higher velocity and more accurate value for the maximum velocity. On the other hand, very fine particles such as $dp=0.004$ m does not predict the returning water velocity ($t=1.4-1.8$ sec of Figure 3-4) correctly.

The reason of tremendous different between the velocity at the given point can be the existing of thin layer of water in the channel downstream of lock gate. Since one layer of fluid particles is located at that area, the total mass of water resting on the floor varies as a function of particle size. When heavier mass exist after the lock, more momentum is dissipated from dam break flow. In other words if light particles rest in front of the wave, less energy dissipation happens and the wave front moves faster with less friction. Then the wave reaches the end of the tank and water splashing occurs. Thus severe impact happens and higher run-up is seen at the end of the tank, which cause higher potential energy. This higher potential energy may cause higher returning velocity during $t=1.4-1.8$ sec.

The force analysis of their simulations proves that the most accurate particle spacing tends to be in the range of 0.01- 0.014 m. Very fine particles underestimate the maximum force on the pile and also too many fluctuations of the force are seen after wave-obstacle impacting.

3.1.1.1 Boundary condition: Repulsive force

Another test was attempted to simulate using repulsive force as boundary condition instead of dynamic particles. All settings were as the same as the first simulation and particle

size is $dp=0.0225$ m. Figure 3-5 presents the velocity in flow direction at the interested point and total net force on the obstruction.

Comparison of dynamic particles BC with repulsive force yields that generally, dynamic particles method explains the kinematics of the problem more accurately than repulsive force methods do. Although estimated force by the repulsive force method is not as good as that of dynamic particles for the time greater than 1.2 s, the overall behavior of force on the pile by repulsive force is reasonable.

Note that the maximum velocity of our study by repulsive force is about twice of what Rooney et al. (2011) found from their study from the same configuration. One of the probable reasons of this discrepancy is the location at which the particle velocity is recorded. The monitoring point of experiment is located about 2.6 cm away from the bottom. In our study, the size of particles is also considered in data extraction and half h is added to 2.6 cm. As an example, if the particle size is 0.02 m, the extraction point is located $(x,z)=(0.75,0.026+0.02*1.4/2)=(0.75,0.04)$.

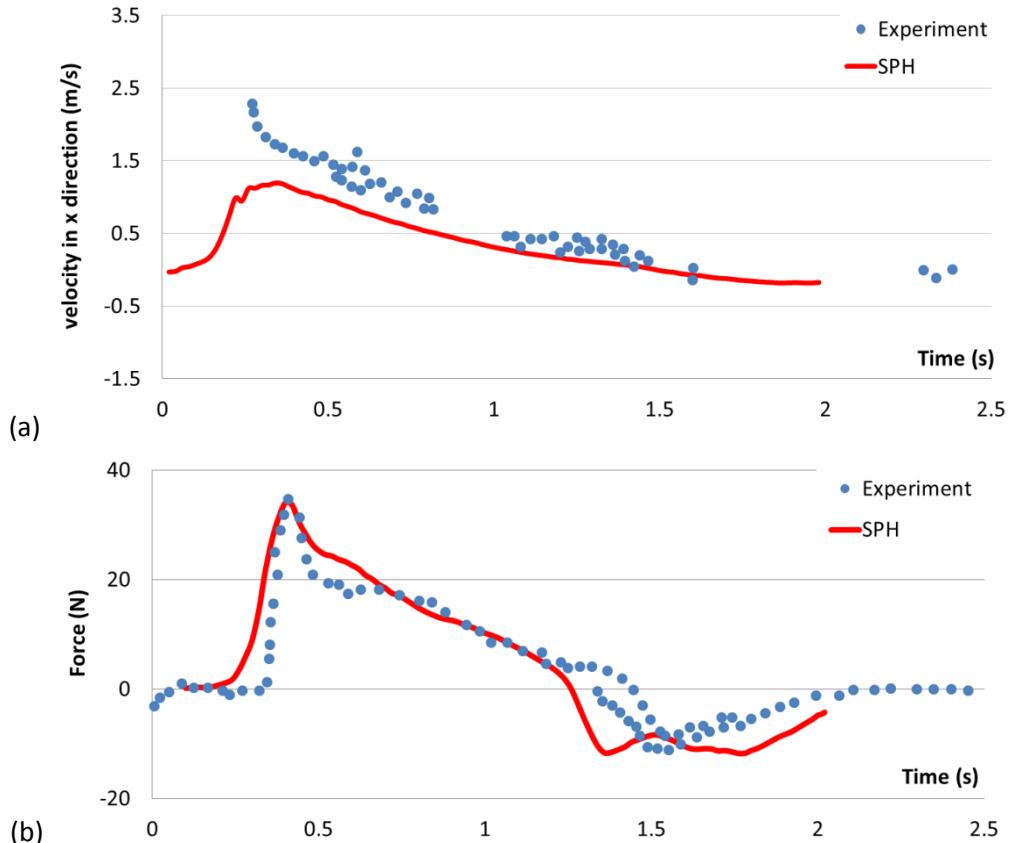


Figure 3-5- Comparison between experimental data (dots) and third-attempt SPH simulation ($d_p=0.0225$ m and using Repulsive force BC) (solid line)

3.1.2 Test 1-2: Dam breaking Impact on a Circular Cylinder (in 3D)

In addition to square pile, several dam break experiments have been carried out with circular cylinder [Arnason, 2005; Nouri et al., 2010]. Arnason (2005) and Nouri et al. (2010) studied the dam break wave impact on both square and circular cylinder. The major difference between their studies is the initial water depth downstream side of the dam. Nouri et al. (2010) attempted a dry bed downstream of the dam while, Arnason studied a wet bed dam break. Both recorded a couple of characters of test such as pressure and total force on

the obstacle, wave celerity and bore wave height. Since the overall size of Arnason's experiments is much smaller than that of Nouri et al., Arnason experiments are used for present study.

In this part the accuracy of SPHysics is tested for a circular object encountered a dam break impact and also the effect of very coarse particle size is investigated. Later in the current section, the impact of a solitary wave on a circular cylinder will be discussed. It will be shown that SPHysics cannot estimate the total force on the pile accurately if the particle size is chosen 0.03 m due to the largeness of wave basin.

Arnason (2005) explored the dam break flow and impact on a square and circular cylinder on a wet bottom. The water depth after the dam is 2 cm in all attempts and different pile sizes and still water depths behind the dam were tried. One of those experiments is chosen here to be simulated by SPHysics. The cylinder diameter is 14 cm and the lock water depth is 21 cm. The tank width and length is 0.61 m and 16.6 m and the circular cylinder is located 10.1 m downstream of the gate. To reduce the total number of particles, the length of numerical lock water is smaller than that of actual experiment, thus the mass of water behind the numerical dam is less than that of experiment which causes less momentum or less potential energy. This assumption may cause less numerical force on the cylinder as $dp=0.01$ m.

Figure 3-6 presents the total force on the obstacle and compare it with experimental records. This test was simulated twice, once with $dp=0.01$ m and another time with $dp=0.03$ m. Note that these results are from some primary simulations and the accuracy can be improved by modifying parameters, but here the accuracy is not our first concern; whereas,

the effects of relatively coarse particle on the total calculated force on the cylinder is our first priority in this study.

Although SPH did not predict the total force perfectly, the general trend is compatible with experimental observation. Regarding Figure 3-6, SPH underestimates and overestimates the total force using $dp=0.01$ m and $dp=0.03$ m, respectively. This test shows that $dp=0.03$ m is not suitable enough for a test in which a circular cylinder is located in such a high Reynolds number flow. The initial huge kick in the plot corresponding to $dp=0.03$ m does not exist if dp is 0.01 m. Consider that total number of particles is about 97K and 1.3M if particle size is set to 0.03 m and 0.01 m, respectively. The model with coarser particles takes about 5 hours to run on 16 processors and the model with finer particles takes about 34 hours to get done on 32 processors of RENCI cluster. Note that above timing information is based on 8 seconds of real time modeling. It implies that to have reasonable outcomes small enough particle size is essential. On the other hand, employing fine particles is not possible to run when the size of domain is huge. Test case 5 is an example of large-domain problems. If dp is set 0.01 m for test case 5, the total number of particles will be about 27M. Regardless of computational time, the required memory and CPU are not economical.

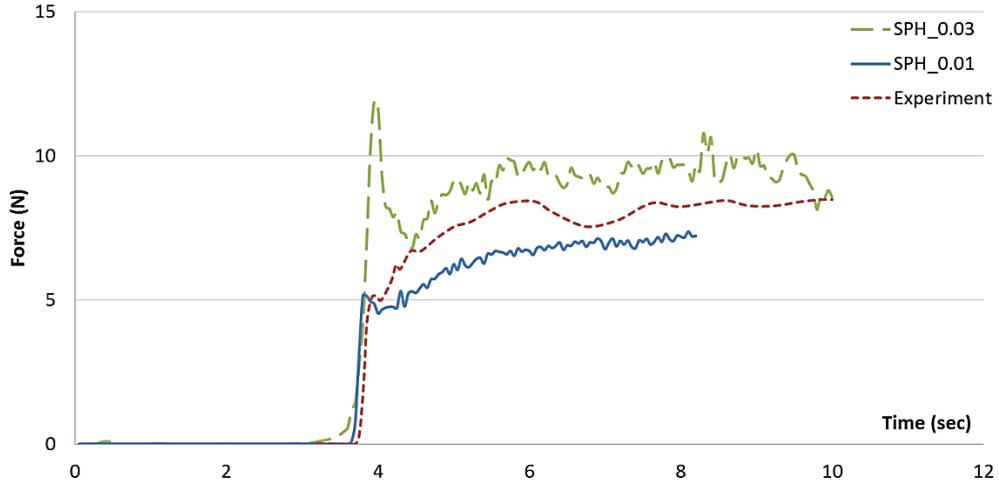


Figure 3-6- Total force on the circular cylinder

3.2 Test 2: Dam Breaking Wave Impact on a Small Structure (in 3D)

Lee et al. (2010) applied both weakly compressible (WCSPH) and truly incompressible SPH (ISPH) to 3D water collapse in a tank and river dam spillway. They compared WCSPH and ISPH with experiment and discussed the accuracy of each method. Boundary condition is assumed through the technique of wall particles and fictitious particles [Lee et al., 2010]. In WCSPH, no particular condition is required for the free surface, while pressure of surface particles must be set to zero if incompressible SPH is applied. Due to complex algorithm of solving Poisson equation on multiple cores, they run ISPH on a single processor and WCSPH on multi processors. The geometry of 3D dam break in the tank is presented below.

Regarding Figure 3-7, a small rectangular box is located in an open roof tank [see Kleefsman et al., 2005]. The geometry and measurement points are presented in Figure 3-7.

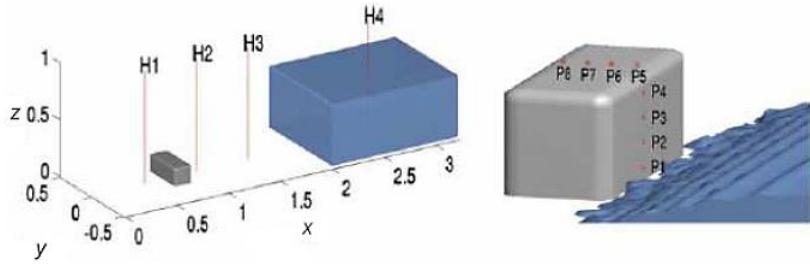


Figure 3-7- Geometry of the 3D water collapse experiment with measurement positions (unit: meter).
Zoom of the box is shown on the right [Lee et al., 2010]

Water heights are monitored at points H1 to H4 and water pressure sensors are installed on points P1 to P8 (details are illustrated in Figure 3-8 and Figure 3-9 schematically). Although in the experiment, water is held by a gate and it is released by pulling up the gate, there is no gate in numerical simulation and water is released immediately when the simulation begins. Figure 3-9 presents the detail of all dimensions of the problem from top and side view.

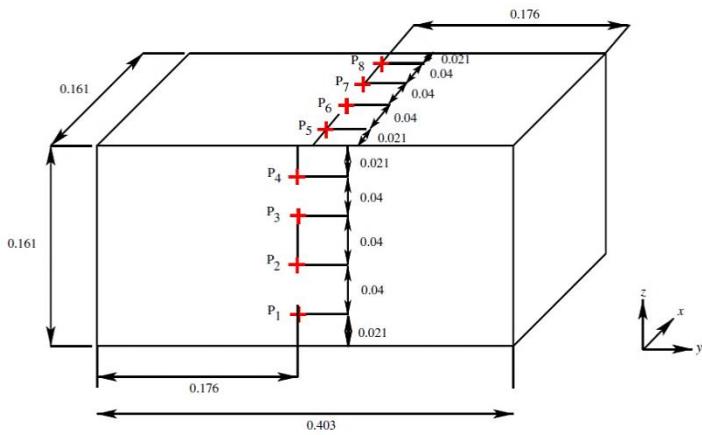


Figure 3-8- Fluid pressure is recorded at points P1 to P8 on the surface of the Obstruction

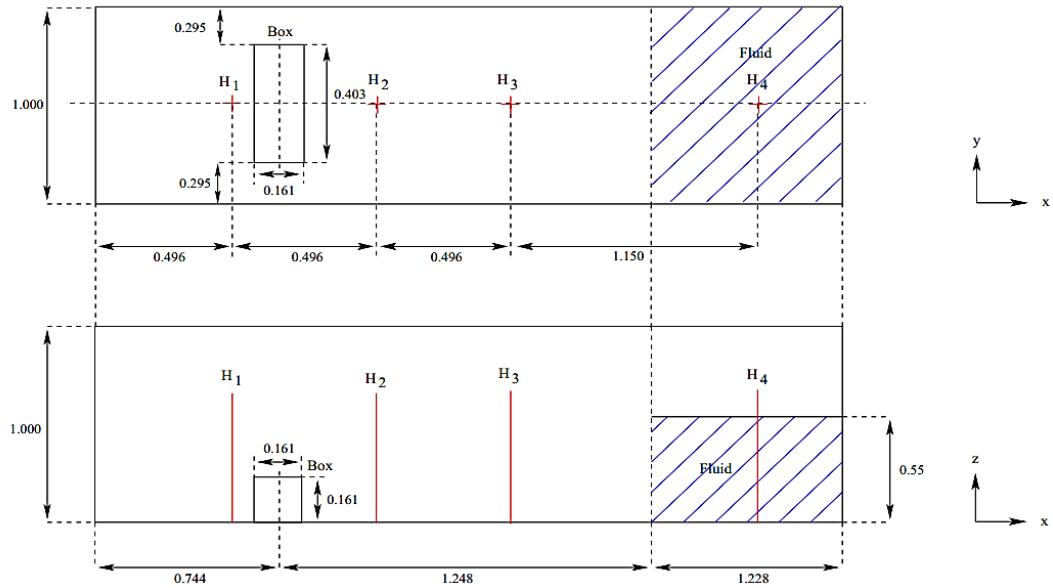


Figure 3-9- In-detailed description of system. Top picture: plan view, Bottom picture: side view

For WCSPH, they defined particle size and smoothing length as $dp=0.018$ m and $h=0.0275$ m. They compared all SPH results with experiment and VOF (Volume of Fluid) results in Kleefsman et al. (2005). Lee et al. (2010) compared water surface elevation at H2 and H4 and fluid pressure at P1, P3, P5 and P7.

In our study, the same test case was run with weakly compressible SPH approach and using repulsive force as wall boundary condition. Monaghan and Kajtar (2009) formulation was used for the box and Monaghan and Kos (1999) method was used for all other walls. The rest of settings are identical to Lee et al. (2010) description of their simulation.

Figure 3-10 compares the water height at points H2 and H4 of our simulation, Lee et al. (2010) study and experimental data.

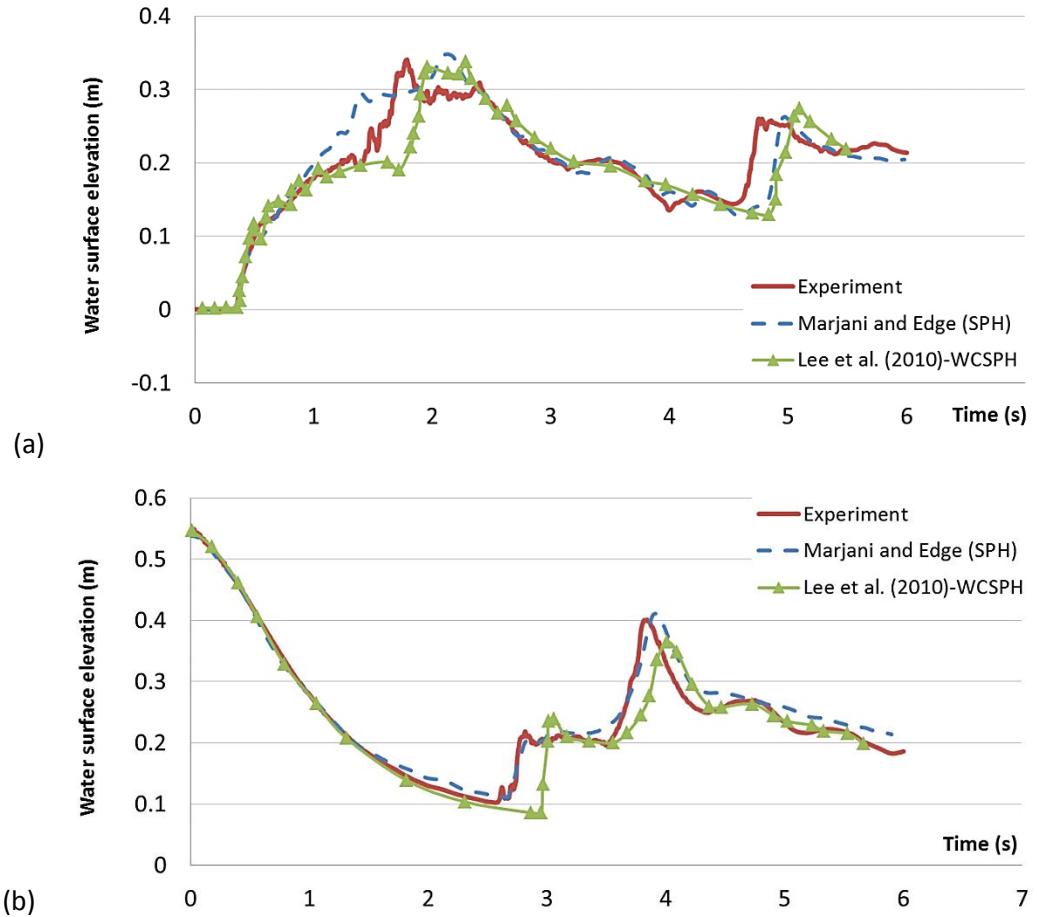


Figure 3-10- Water height time history of the 3D dam-break flow at (a): H2 and (b): H4

Figure 3-10 demonstrates that our results for water heights are in better agreement with observation in comparison with Lee et al. (2010), generally. This comparison is seen clearly in Figure 3-10-b for the point H4.

Figure 3-11 presents the fluid pressure at points P1 and P3 and compares them with experimental results and Lee et al. (2010) study. All comparisons are done with weakly compressible SPH results of Lee et al. (2010).

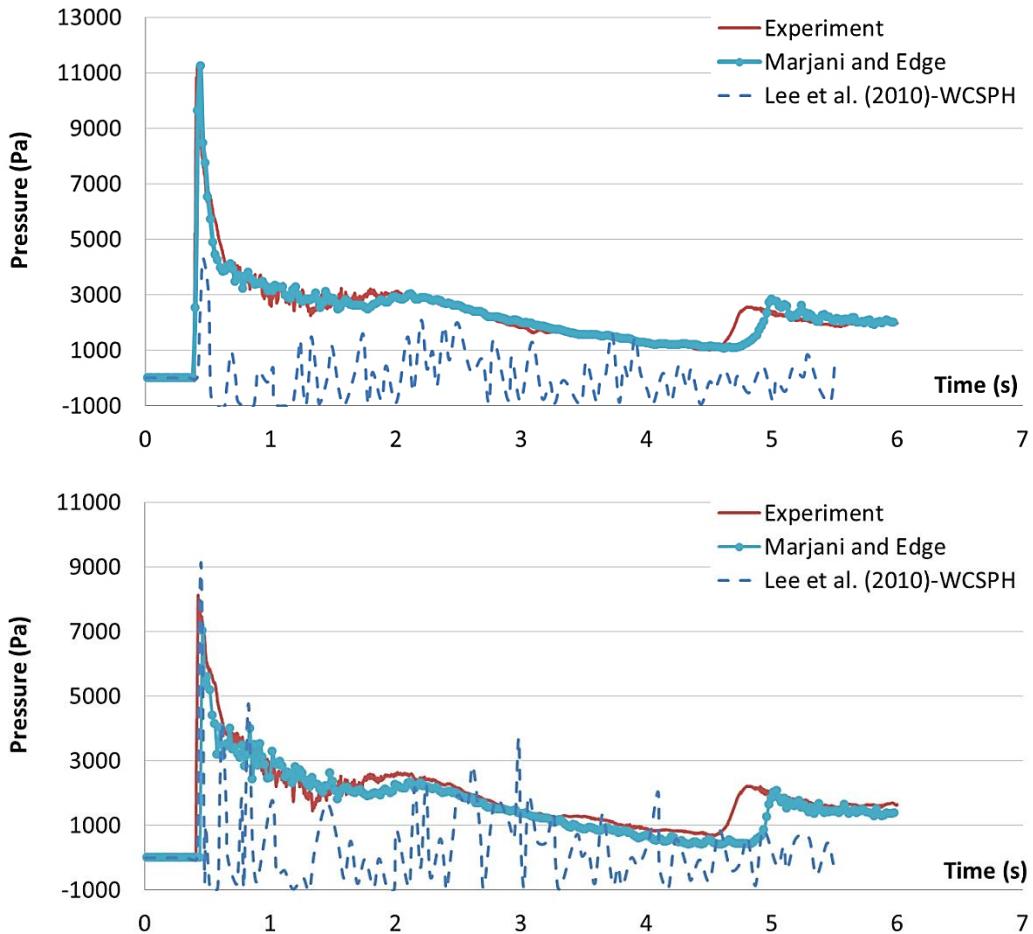


Figure 3-11- Pressure time history of the 3D dam break flow at P1 (top) and P3 (bottom)

Figure 3-11 shows a very good agreement between our results and experimental data. In addition it shows that Lee et al. (2010) WCSPH model does not predict the pressure on the object well. Although the ISPH model of Lee et al. (2010) does not calculate the maximum pressure neither at P1 nor at P3 correctly, its general behavior is compatible with experiment.

3.3 Regular Wave Propagation in 2D

3.3.1 Test 3-1: Wave Propagation on a 2D Beach

In this problem, the SPH is tested for its ability of predicting kinematics and only the water surface elevation and particle velocity are evaluated. First of all, the behavior of generally different types of boundary condition on wave making and wave propagating on a beach is studied. The geometry of problem is sketched in Figure 3-12.

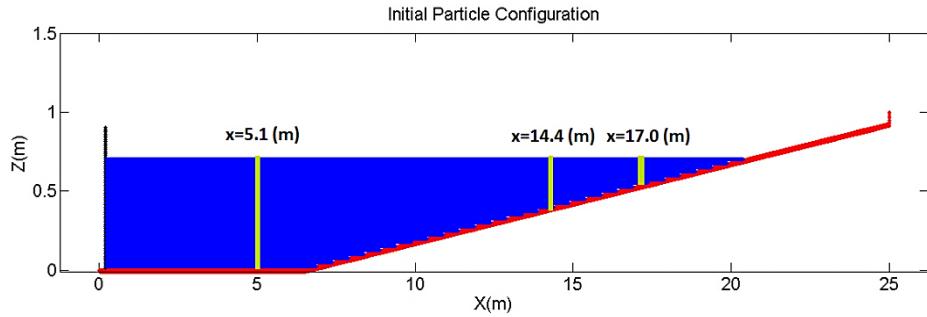


Figure 3-12- Initial setting of regular wave propagation on a simple beach

The length of flat bottom is 6.5 m and total length of wave tank is 25 m in which the inclined part with the slope of 1:20 has a length of 13.5 m. The off-shore still water depth is 0.7 m. The desired data is recorded at three locations along the tank, $x_1=5.1$ m, $x_2=14.4$ m, and $x_3=17.0$ m. A piston-type wavemaker, located at $x=0.0$ m, produces linear wave in the tank with the period of 2 s and wave height of 12 cm at offshore ($x_1=5.1$ m). Particle size and kernel length is set to 0.015 m and 0.025 m, respectively, thereby the total number of

particles is about 45000. This 2D simulation is run on 8 processors. Figure 3-13 presents the wave height at three interested points (x_1 - x_3).

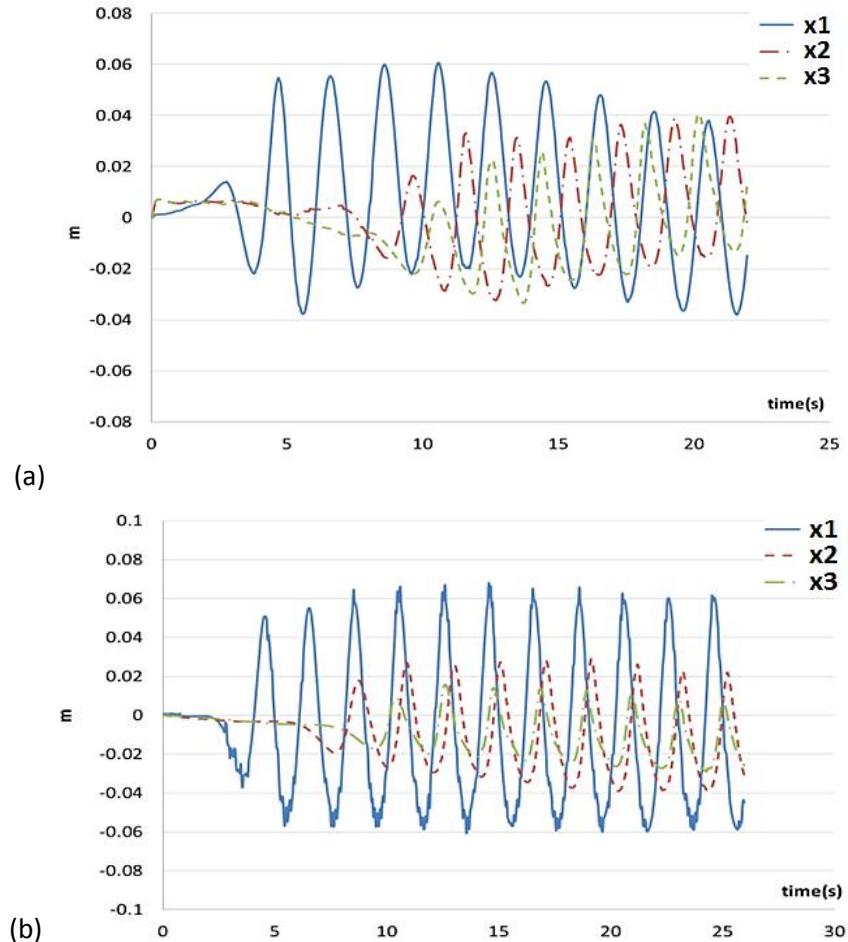


Figure 3-13- Wave height (meter) at three locations x_1 , x_2 , and x_3 .
(a) with repulsive force BC and (b) with dynamic boundary particles

Based on an unknown reason, repulsive force BC does not produce regular wave as consistent as dynamic boundary particles do, see Figure 3-13. When Monaghan and Kos (1999) formula is applied for all boundaries, the mean surface water level varies in time,

while dynamic BPs technique keeps the mean water level almost unchanged during the simulation, Figure 3-13-b. Moreover, the desired wave height is 12 cm while the wave height by repulsive force is about 8 cm. It may be expected that dynamic particles performs more accurately than repulsive force since the nature of those particles is identical to fluid particles and they act exactly similar to fluid particles, whereas boundaries exert fictitious force on the fluid in repulsive force approaches. Hereafter, only dynamic boundary particle is used when a problem involves any types of wave propagation, although this method makes more friction close to the boundaries. Experiences show that this method creates a better shape of water surface elevation and more correct particle velocities.

De Padova et al. (2008) evaluated their simulation by SPH method with the wave propagation experiment implemented by De Serio and Mossa (2006). Wave height and particle velocity in x and z direction are recorded at three mentioned locations and compared with experimental observation. Similar simulation and comparison are conducted here.

Insomuch as Dalrymple boundary condition (Dynamic boundary particle) is known as a sticky method that is water particles tend to stick to the walls, time step must be chosen very small. Hence, CFL number is set to 0.1 in the present modeling. Several simulations prove that the coefficient of artificial viscosity (α) should be small (in the order of 0.001). Padova et al. (2008) set $\alpha = 0.01$ in their investigations. However our simulations demonstrated that $\alpha = 0.01$ dissipates the wave energy and wave height on the inclined surface significantly. Figure 3-14 presents the wave height at three interested points.

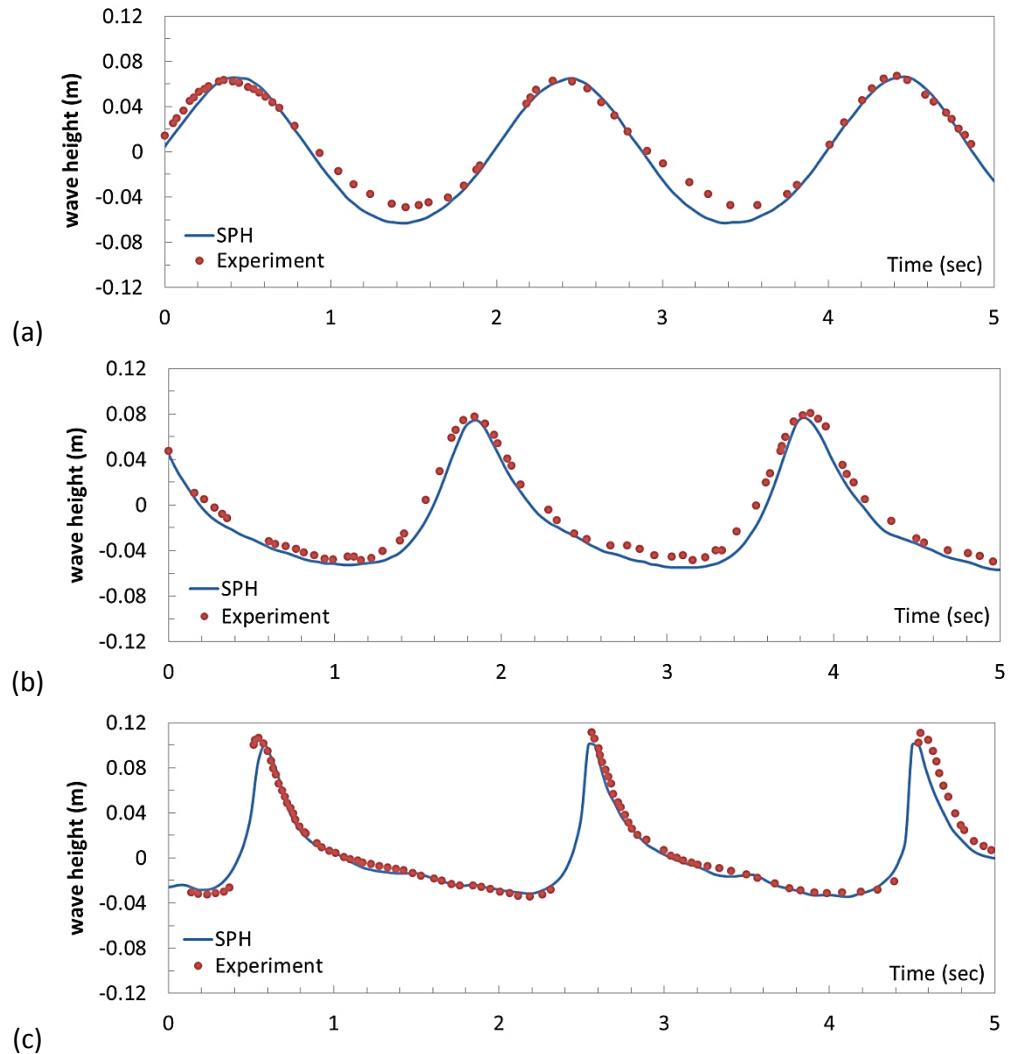


Figure 3-14- Water surface elevation at (a) $x_1= 5.1$ m, (b) $x_2=14.4$ m, and (c) $x_3= 17$ m

SPH produces the wave height or water surface very close to the real data. This compatibility is much more obvious at $x_2=14.4$ m and $x_3=17$ m, see Figure 3-14-b and Figure 3-14-c.

Particle velocity vector at different locations of the tank can be compared with laboratory observation. Following figures present the horizontal (u) and vertical (w) components of

particle velocity at the same locations of monitored water surface. The velocities are recorded 0.1 m above the bottom at locations x_1 and x_2 , and 0.3 m above the bottom at x_3 .

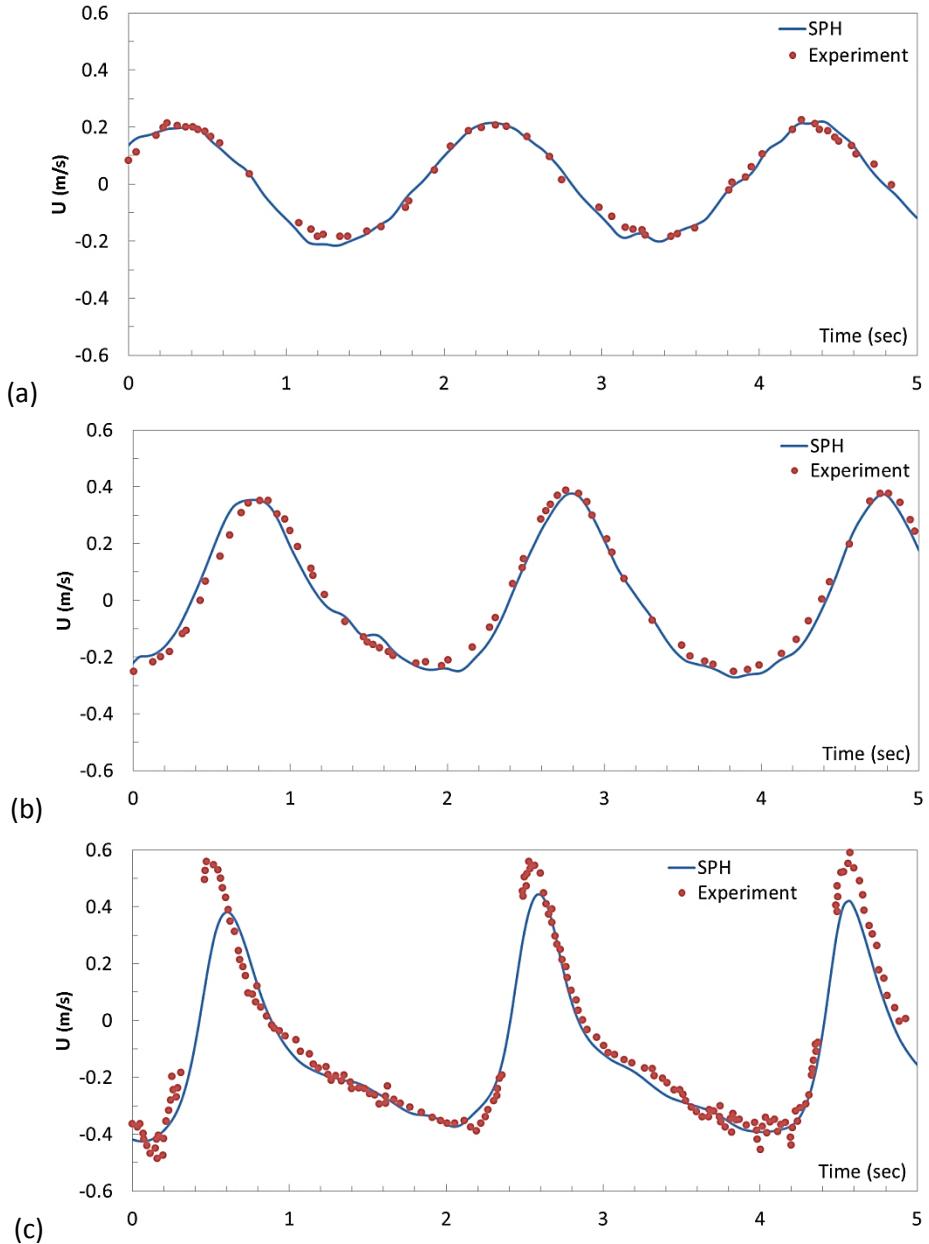


Figure 3-15- Horizontal particle velocity at (a) $(x_1, z_1) = (5.1 \text{ m}, 0.1 \text{ m above the bottom})$,
(b) $(x_2, z_2) = (14.4 \text{ m}, 0.1 \text{ m above the bottom})$, and (c) $(x_3, z_3) = (17 \text{ m}, 0.3 \text{ m above the bottom})$

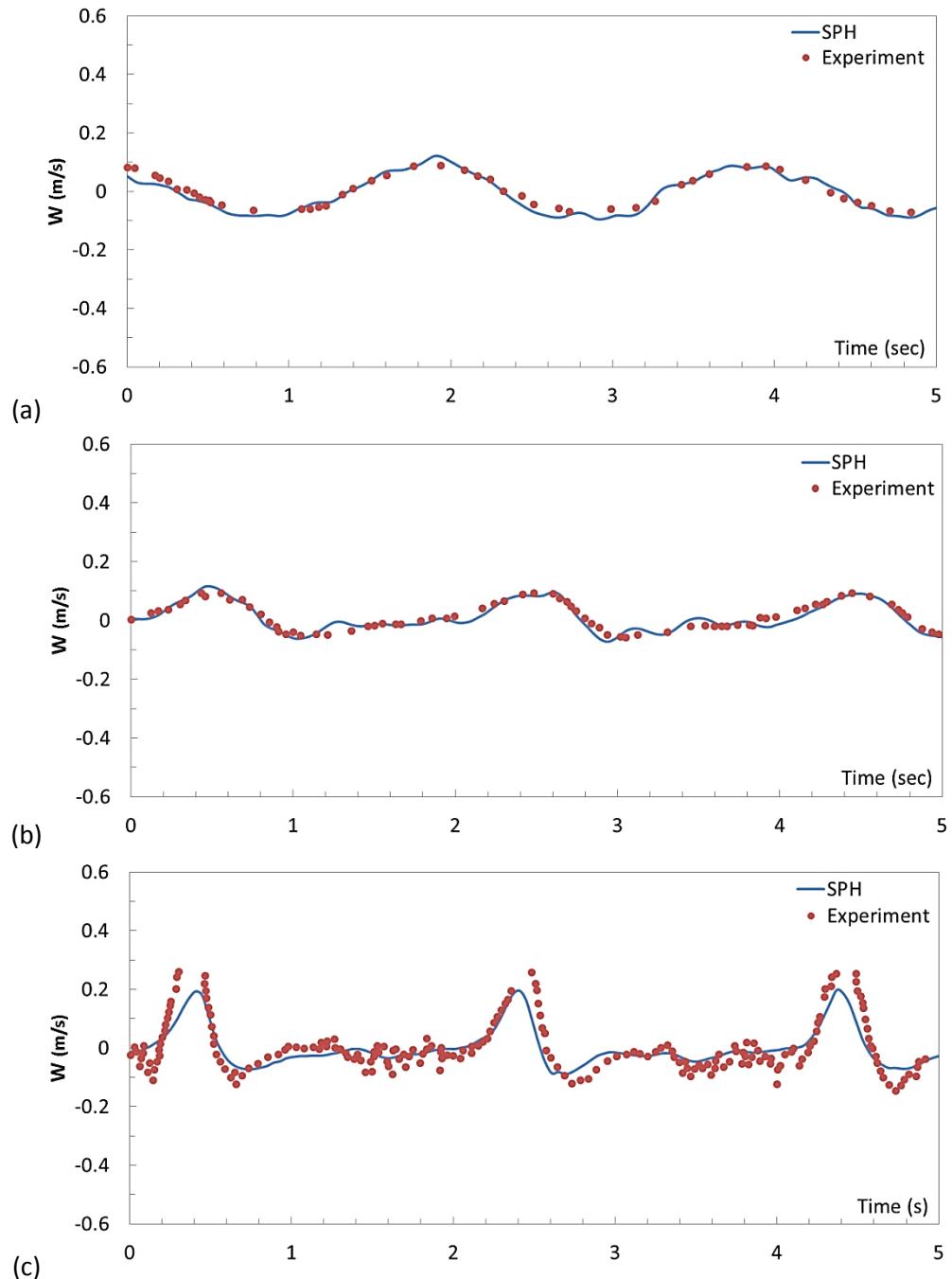


Figure 3-16- Vertical particle velocity at (a) $(x_1, z_1) = (5.1 \text{ m}, 0.1 \text{ m above the bottom})$,
(b) $(x_2, z_2) = (14.4 \text{ m}, 0.1 \text{ m above the bottom})$, and (c) $(x_3, z_3) = (17 \text{ m}, 0.3 \text{ m above the bottom})$

Figure 3-15 and Figure 3-16 show that SPH is accurate enough in the aspect of kinematics. Vertical and horizontal velocities at both flat and sloped beach are in reasonably good agreement with observations. Velocity components of flat parts are perfectly predicted by SPH, see Figure 3-15-a, Figure 3-15-b, Figure 3-16-a, and Figure 3-16-b. This agreement can be improved if finer particle is used.

3.3.2 Test 3-2: Wave Runup on a Plain Beach

The other term that can be examined is the magnitude of wave runup at the shoreline. Wave runup is the maximum elevation of wave uprush above the still water level and consists of two parts: rise in mean water level due to wave action (wave setup), and fluctuation about the wave setup [CEM⁶]. Although the wave runup was not studied experimentally for the above test case, there are some equations that estimate the runup for different types of beach and wave condition. Based on laboratory data, Hunt (1959) empirically determined the runup as a function of beach slope (β), wave height (H_0) and wave length (L_0),

$$R = H_0 \frac{\tan \beta}{\sqrt{H_0/L_0}} \quad (3-2)$$

To check the runup, a similar test to the previous one was conducted with finer particle, $dp=0.005$ m which is one-third of the first try. To decrease the computational cost, the still water depth was reduced to 0.6 m. The wave height and period are identical to previous case.

⁶ Coastal Engineering Manual

Given $H_0=0.12$ m, $L_0=4.36$ m, and $\beta = 2.86^\circ$, Hunt formula gives the runup about 3.6 cm. Figure 3-17 shows a snapshot of the simulation. The output states that the runup based on SPH simulation is about 3.8 cm which is in good agreement with Hunt's formula.

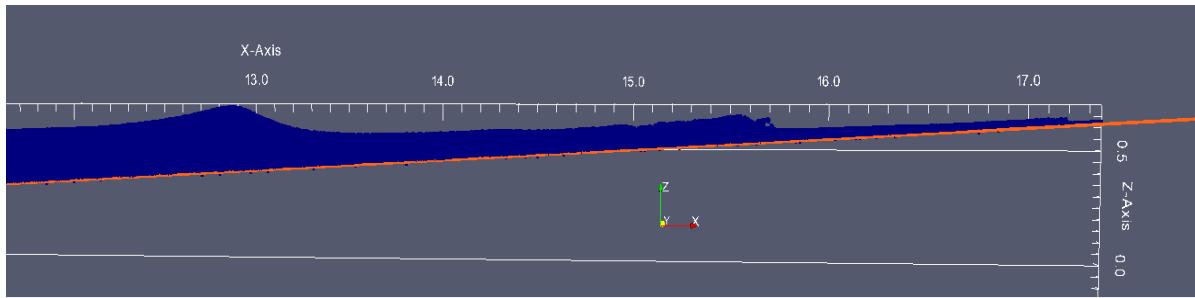


Figure 3-17- Wave runup on the sloped beach by SPH

3.3.3 Test 3-3: Wave Force on a Horizontal Pipe in 2D

The force on a horizontal cylinder due to wave motion is computed in 2D and compared with experimental studies. On 1994, Prasad experimentally studied the impact of wave on a small horizontal cylinder [Prasad, 1994]. Different wave condition and different location of cylinder with respect to still water level were tried in these tests. Two different tests are picked for our simulations. Still water depth in both tests is 55 cm and the pipe diameter is 4.2 cm. Simulations were done with 2D-parallel SPHysics and particle size was set to 0.005 m, so, the total number of particles is about 220,000. The wave condition and the location of pipe in these two tests are presented in.

Table 3-1- The wave condition: wave period (T) and wave height (H) and the distance between the bottom of pipe and the still water level

| Case # | T (sec) | H (cm) | h_{b-SWL} (cm) |
|--------|---------|--------|------------------|
| 1 | 1.1 | 17.5 | 0.5 |
| 2 | 1.5 | 18.4 | -4.5 |

Schematic picture of the position of pipe with respect to the SWL and wave height are drawn in Figure 3-18. h_{b-SWL} in presents the distance between the bottom of pipe and SWL.

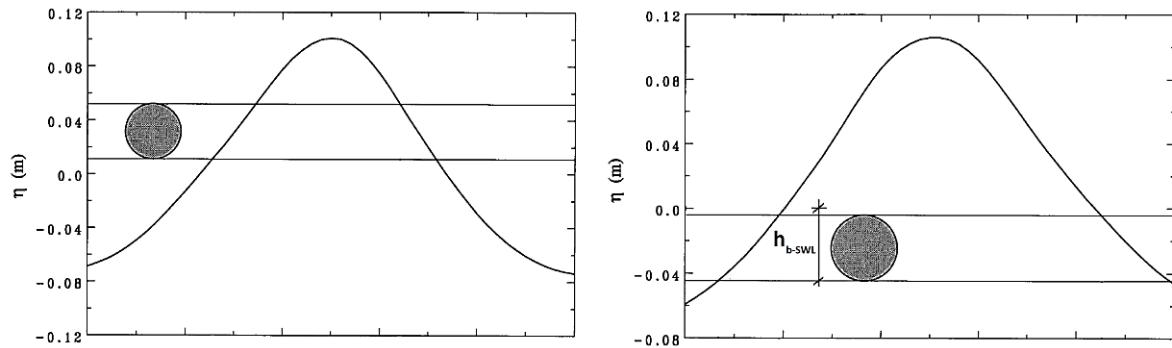


Figure 3-18- The distance between the bottom of cylinder and SWL, h_{b-SWL} , and wave height of two tests. left sketch: Case #1 and right sketch: Case #2 [photo from Prasad (1994)].

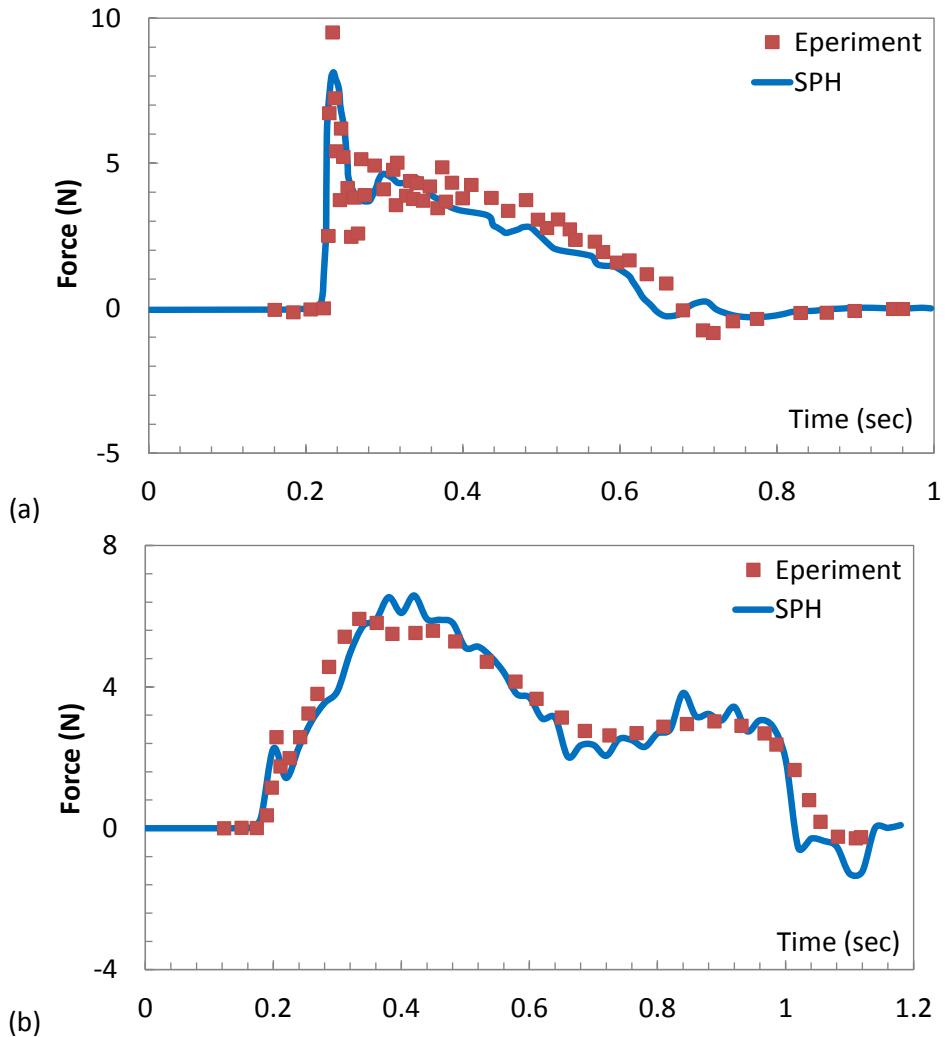


Figure 3-19- Vertical force on the pipe by SPH (solid line) and Experiment (red squares), (a) Case #1 and (b) Case #2.

Figure 3-19 compares the estimated vertical wave force on the pipe by SPH with experimental observation. They prove that using small enough particles leads to acceptable accuracy of force calculation. In case #2, cylinder is completely under the water when the wave crest reaches it while case #1 is a pure slamming-type test. SPH underestimates the vertical force in case #1, Figure 3-19-a, for a small period of time. That can be explained by

the fluid particles stuck on the top of the pipe for a while. In reality, those particles fall down much more quickly. Figure 3-20 illustrates this phenomenon for case #1.

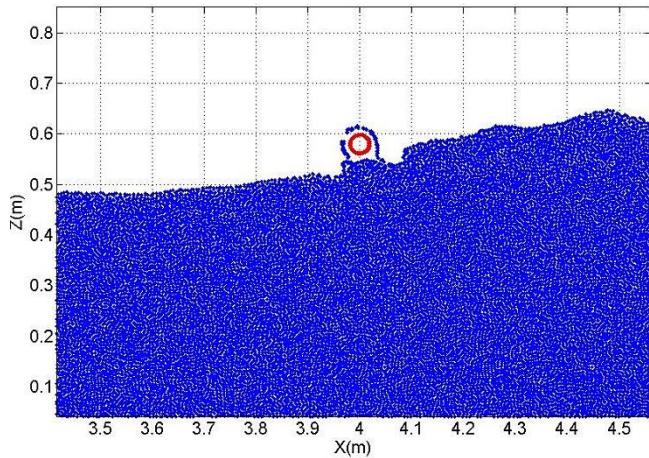


Figure 3-20 - Fluid particles stays on the cylinder for a longer time than do in reality

3.4 Test Case 4: Force on a Flat Rectangular Plate

Relying on previous tests, SPH is absolutely trustworthy for kinematics. The main scope of this study focuses on computing force on a structure. Hence, the benchmarks related to force-object interaction can be at the center of our interest.

Barter et al. (1958) carried out a vast number of experiments on the force of regular wave on four different types of plates, rectangular barge, rounded barge, slotted rectangular barge and flat plate. They present the absolute value of maximum and minimum horizontal force on the object. One of those experiments in which the still water depth is 0.3 m, wave period is 1.09 s, wave height is 0.06 m, and a rectangular barge is located 0.15 m below the water

surface is considered here. The length, width and thickness of barge are about 0.25 m, 0.76 m, and 0.047 m, respectively. This experiment is simulated by 3D-parallel SPHysics. The flat bottom of tank is about 6 m which is much shorter than the experimental one and the slope of the inclined is 1/20. Dalrymple boundary condition is applied and the particle size is 0.03 m, $h=0.04$ m, so the total number of particles is more than 280,000. Since Dalrymple is used, the CFL number is set to 0.15 to reduce the time step. Small time steps prevent the water particles to stick to the boundaries. The plate is defined by two layers of particles in staggered pattern. This configuration makes the thickness of plate 0.06 m which is bigger than the actual size. Prior simulations prove that if thinner plate was used, both buoyancy and horizontal were not compatible with that of observation. The expected buoyancy force based on the actual size of plate is about $9810 \times 0.048 \times 0.76 \times 0.25 = 90$ N. The average buoyancy force on one-layer plate is about 50 N which is equal to almost half of 90 N. Figure 3-21 illustrates a plan view of the wave tank and position of plate.

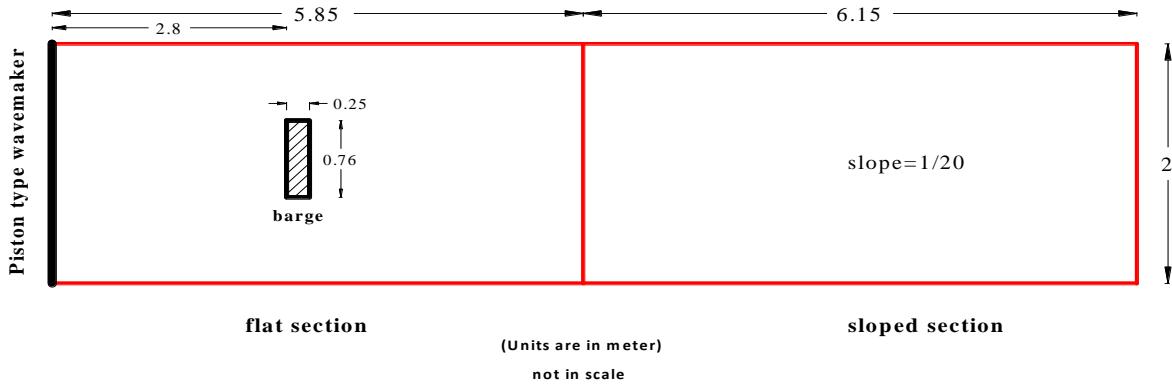


Figure 3-21- Plan view of wave tank and the position of rectangular barge

Figure 3-22 shows the water surface elevation 1 m downstream of the wavemaker on the flat bottom. This proves that SPHysics produces the sinusoidal wave very well when dynamic boundary particles are utilized. The wave height is equal to 0.06 m as expected. Figure 3-23 presents the horizontal force on the barge due to the incoming waves.

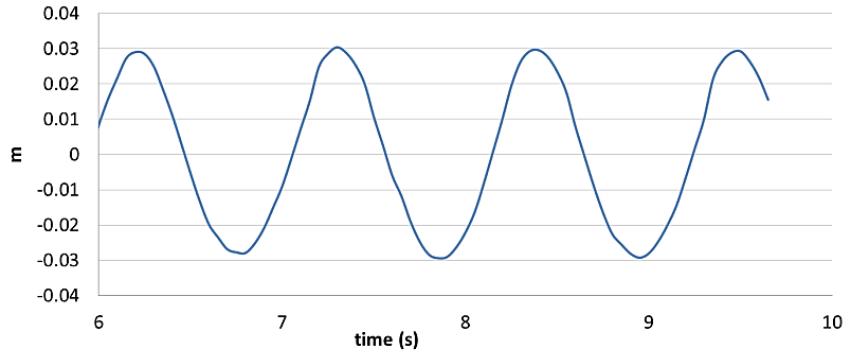


Figure 3-22- Numerical result of water surface elevation 1 m downstream of the wavemaker

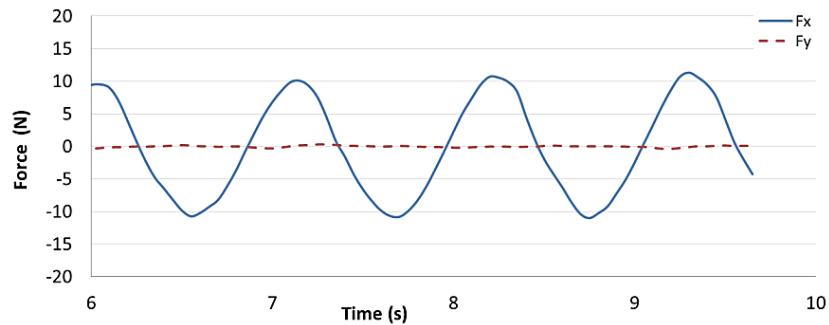


Figure 3-23- Horizontal force on the barge in x and y direction by SPH

The absolute value of numerical maximum and minimum force on the barge is about 10 N. Barter et al. (1958) recorded the maximum and minimum force as 9.97 N and 9.4 N,

respectively. Slightly difference between the SPH calculation and reality may arise from the fact that the numerical barge is a bit thicker than the real one. The numerical result for both wave height and force on the barge are in very good agreement with the data recorded by Barter et al. (1958). The slightly difference between the SPH force calculation and the reality is probably due to three facts, the numerical barge is a little bit thicker than the experimental one, Dalrymple boundary condition produces more friction at boundaries, and particle size is not small enough. Smaller particles may produce less friction in vicinity of barge body. Note that the wave height and numerical barge thickness (0.06 m) are only twice of particle size (0.03 m); nonetheless, the accuracy is fully acceptable.

A flat plate is scientifically considered as a blunt body. A body in constant-speed flow experience a drag force consisted of two components: frictional drag (or viscous drag) and pressure drag (or form drag). Viscous drag comes from the frictional force inside the boundary layers between the body surface and surrounding flow. Form drag is related to the wake generation behind the body and it is usually less sensitive to Reynolds number than the frictional drag. Note that both types of drag are due to fluid viscosity that means if a body placed in an inviscid fluid it does not feel any drag at all. Pressure drag becomes more important in separated flows, while frictional drag is important for attached flows (no separation happens). Regarding experiments, the pressure is not uniform downstream of separation and has lower value than the pressures on the forward face of the body. This difference in the pressure of two sides of a body is the source of drag force. The surface area of body exposed to the flow plays an important role in frictional drag. Whereas, the pressure drag is related to cross-sectional area of the body.

An object is called streamlined if the drag force is dominated by viscous part and is called bluff (or blunt) if it is dominated by pressure part. A wake is created on backward face of any types of body at high Reynolds number. The wake is very small for streamlined bodies while it is large for bluff ones. The adverse pressure gradient around a bluff body causes flow separation. This separation enlarges the wake size, and pressure in the wake drops due to eddy formation that increases the pressure drag portion. Therefore, the body shape determines that a flow is viscous-drag dominant or pressure-drag dominant. A fish and an airfoil at small angles of attack are examples of streamlined bodies, whereas a cylinder, sphere, or an airfoil at large angles of attack are categorized as bluff bodies⁷.

Regarding above description of blunt and streamlined bodies, the flat plate in current test case is considered streamlined object that means friction force is dominant and the drag force due to small wake behind the plate is trivial and separation does not happen.

⁷ Reference: Kundu et al, 2011 and http://www.princeton.edu/~asmits/Bicycle_web/blunt.html

3.5 Test Case 5: Tsunami Wave Impact on a Single Circular Cylinder

3.5.1 Experimental and Numerical Settings

Tsunami wave are generated by large movement or any other disturbances on the ocean's floor, but earthquakes are the most common cause. Tsunami may happen close to the shoreline or travel thousands of miles if the source is in the deep ocean. Offshore-generated tsunamis can travel long distance across an ocean with negligible energy dissipation. That huge amount of energy causes severe damages to the coastlines. Since they are natural hazard that can occur any time in coastal area, it is very vital to study their physics. Generally, two terms of runup on the coast and impact force on structures are considered in the elevation of tsunami damage.

A series of experiments were carried out by Liu and Yim at Oregon State University during 2002-2007. They chose solitary wave for its close relationship to tsunami wave behavior offshore. Various still water depth ($d=0.75, 0.6$, and 0.45 m), wave amplitude to water depth ratio ($\alpha=\text{wave height/water depth}=0.2-0.6$), and three types of solitary wave were tried in this study. Water surface elevation, particle horizontal velocity, dynamic pressure on cylinder surface, and the total force/momentum on the cylinder were monitored during each experiment. Figure 3-24 depicts a plan view of the single cylinder case and the position of ADVs and wave gages. The details of all experiments can be found in Zhang (2009).

The 3D-parallel SPHysics is used to simulate one of their experiments with a single cylinder cases. In this experiment, water depth is $d=45$ cm and $\alpha=0.4$. Cylinder diameter is 1.22 m and the length, width and depth of basin is 48.8, 26.5 and 2.1 m, respectively. The center of cylinder is located 21 m away from the wavemaker.

Center of the cylinder = (21.085 m, 0); diameter of the cylinder = 4 ft = 1.219 m

● Tower for supporting wave gages

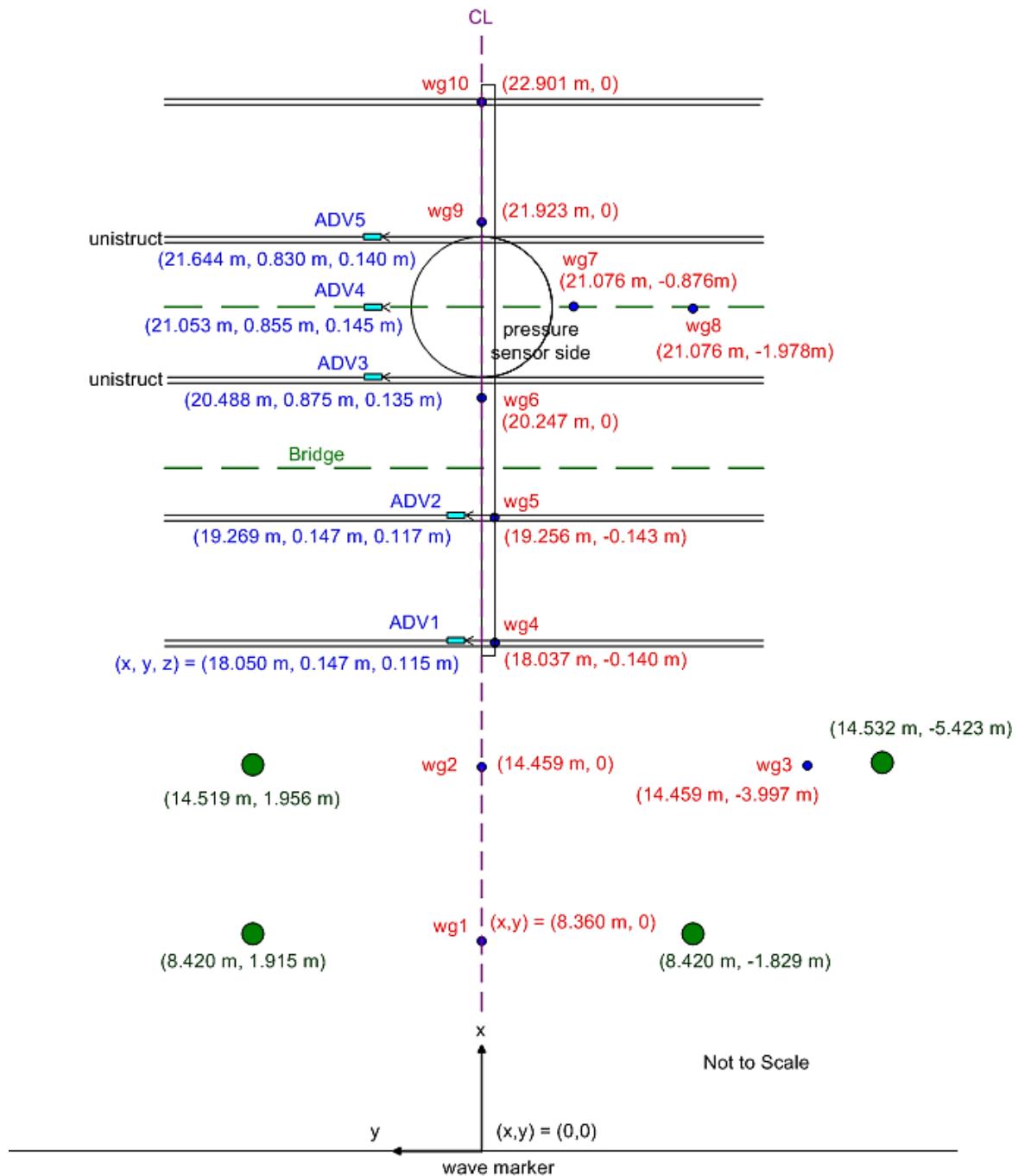


Figure 3-24- Plan view of single cylinder and all measuring instruments positions

To reduce the number of water particles in numerical simulation, the length and width of basin are reduced to 15.7 and 4.0 m, respectively. Since the width is reduced from 26.5 m to 4 m, periodic boundary condition is chosen for lateral boundaries in order for the simulation to become more similar to the experiment. In the SPHysics simulation, the length of flat bottom is 10 m and the center of cylinder is located 8.4 m away from the wavemaker which is located at $x=0.0$ m.

Regarding previous discussion, Dalrymple boundary condition is used in this test. In addition to computing the total force on the cylinder by SPHysics, particle horizontal velocity is recorded at all ADVs' location of experiment (See Figure 3-24). Henceforth, all experimental data are extracted from Zhang's, 2009 graphs. Figure 3-25 shows a snapshot of SPHysics simulation.

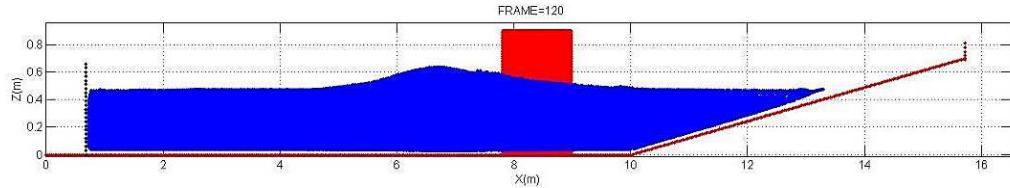


Figure 3-25- A snapshot from the SPHysics output at $t=4.8$ s

The non-breaking wave form of a typical solitary wave is described by Dean and Dalrymple (1991)

$$\eta(x, t) = H \operatorname{sech}^2(k(x - Ct)) \quad (3-2)$$

The parameter k (wave number) is described by

$$k = \sqrt{\frac{3}{4} \frac{H}{d^3}} \quad (3-3)$$

where H and d are the wave height and still water depth, respectively. Wave celerity, C , is governed by [Dean and Dalrymple, 1991]

$$C = \sqrt{gd} (1 + \alpha / 2) \quad (3-4)$$

The horizontal and vertical component of particle velocity under the solitary wave is presented below

$$u = C \left\{ \alpha \left[1 + \frac{\alpha}{4} \left(3 \left(\frac{z}{d} \right)^2 - 5 \right) \right] \right\} \quad (3-5)$$

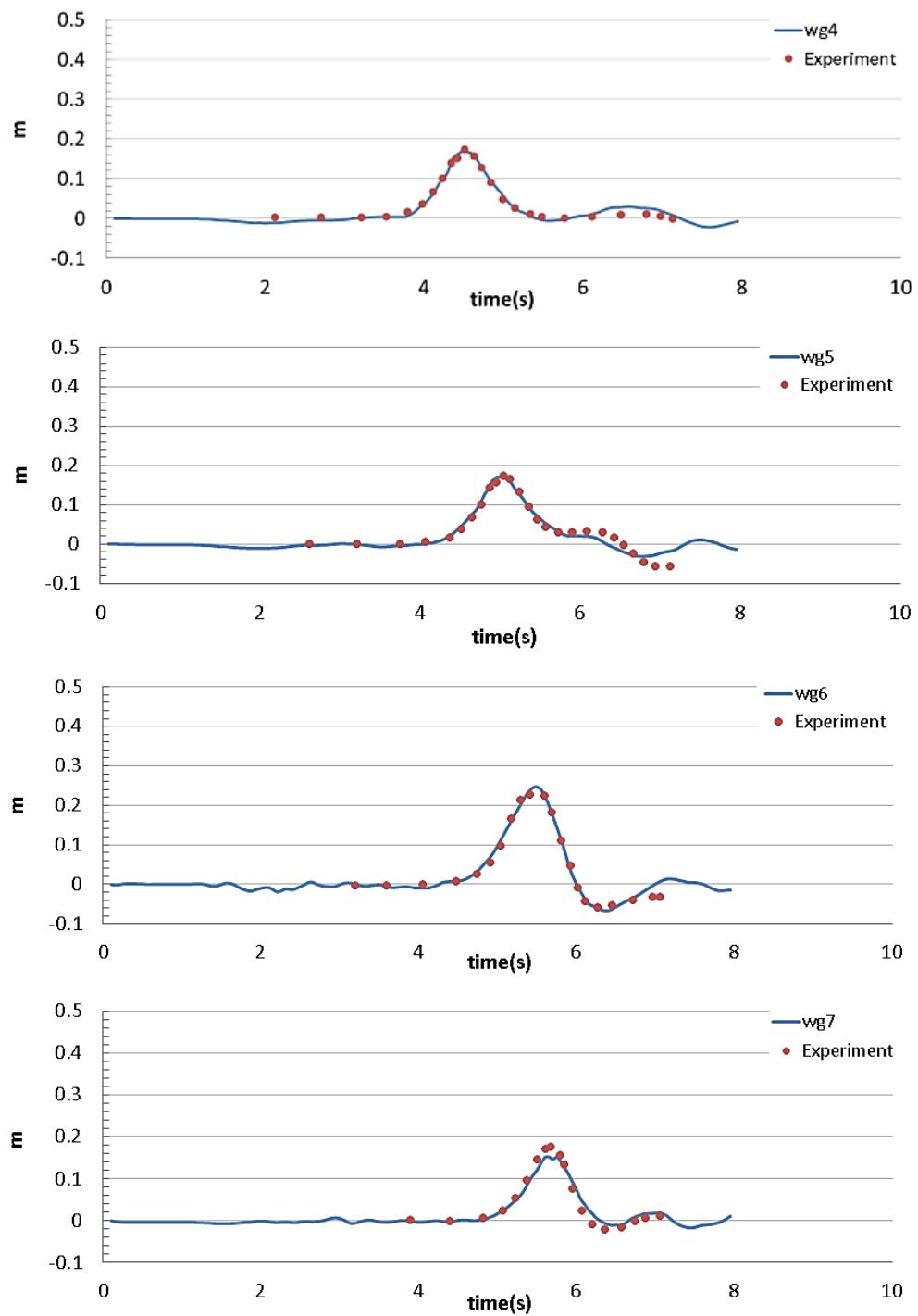
$$w = C \sqrt{3\alpha} \frac{z}{d} \frac{\eta}{d} \tanh \left(\sqrt{\frac{3}{4} \frac{H}{d^3}} (x - Ct) \right) \left\{ 1 + \frac{\alpha}{2} \left[1 - \frac{7\eta}{H} - \left(\frac{z}{d} \right)^2 \left(1 - \frac{3\eta}{H} \right) \right] \right\} \quad (3-6)$$

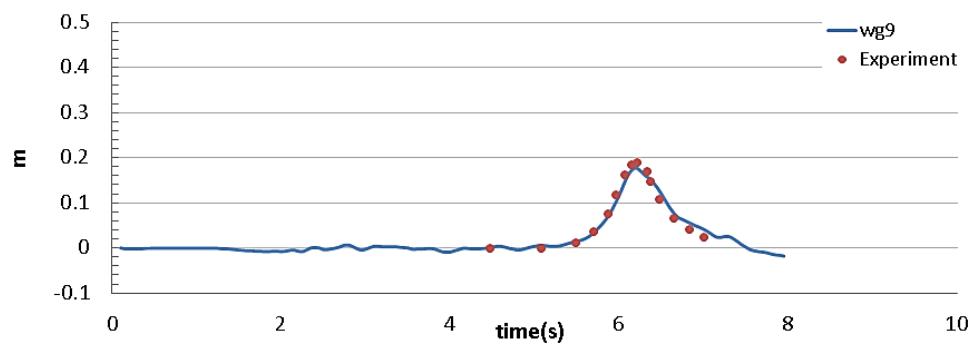
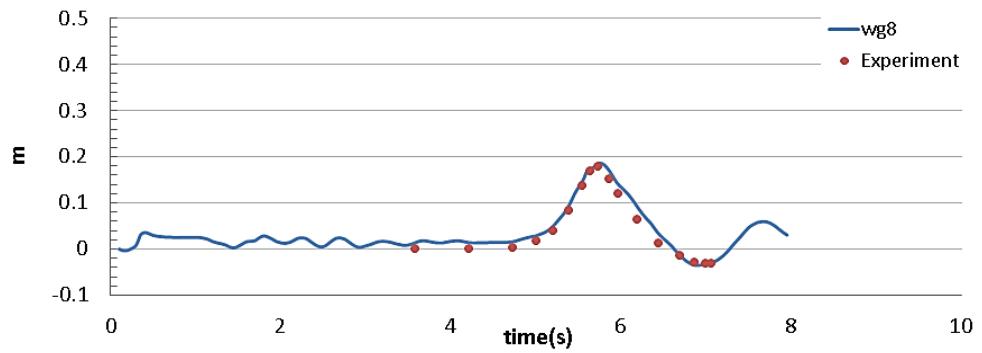
The motion of wavemaker is very crucial in generating a desirable solitary wave. The motion of a piston-type wavemaker is dictated by the equations described in Goring (1979). See Appendix (B) for more details about wavemaker formulation.

3.5.2 Results

Generally, a problem consists of kinematic and dynamic parameters. The wave amplitude (water surface elevation) and particle velocity, as the kinematic terms, are compared with experiment in Figure 3-26 and Figure 3-28. The wave amplitude at 6 wave gage locations are compared with experimental records in Figure 3-26.

Figure 3-26- Experimental (dot line) and Numerical (solid line) wave amplitude versus time at gauges 4,5,6,7,8,9. Dynamic boundary particle is applied here.





All plots in Figure 3-26 prove that SPH can predict the water surface in very good agreement with real data. This perfect compatibility exists both for the location far from the cylinder and close to it and all the time before and after the wave-structure impact. Note that this flawless fitness was not reachable when repulsive force was applied. Figure 3-27 illustrates the wave amplitude at only three wave gages, #5, #6, #7. Comparing two different types of boundary condition implies that repulsive force boundary condition is not even able to generate the solitary wave accurately, although the wave height is in good agreement with theory and experiment.

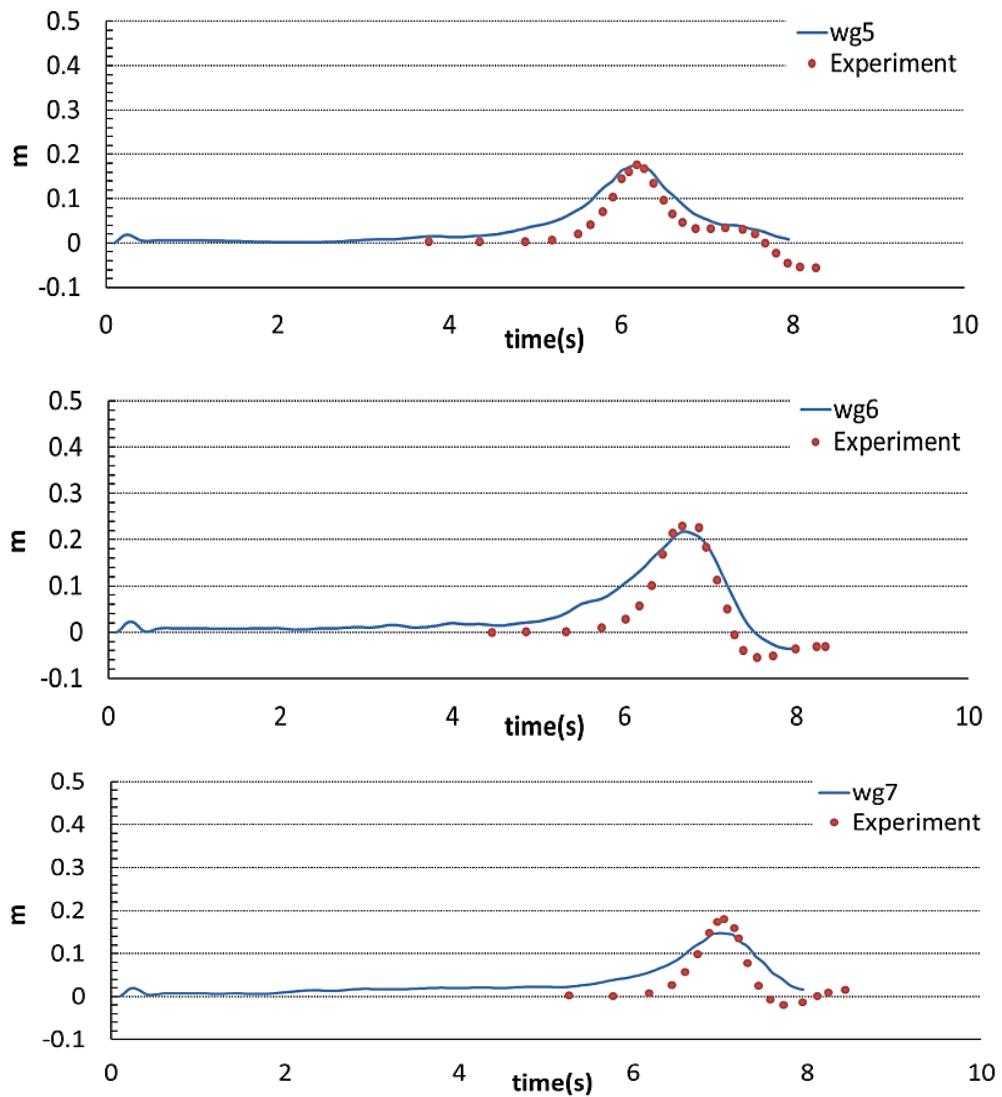
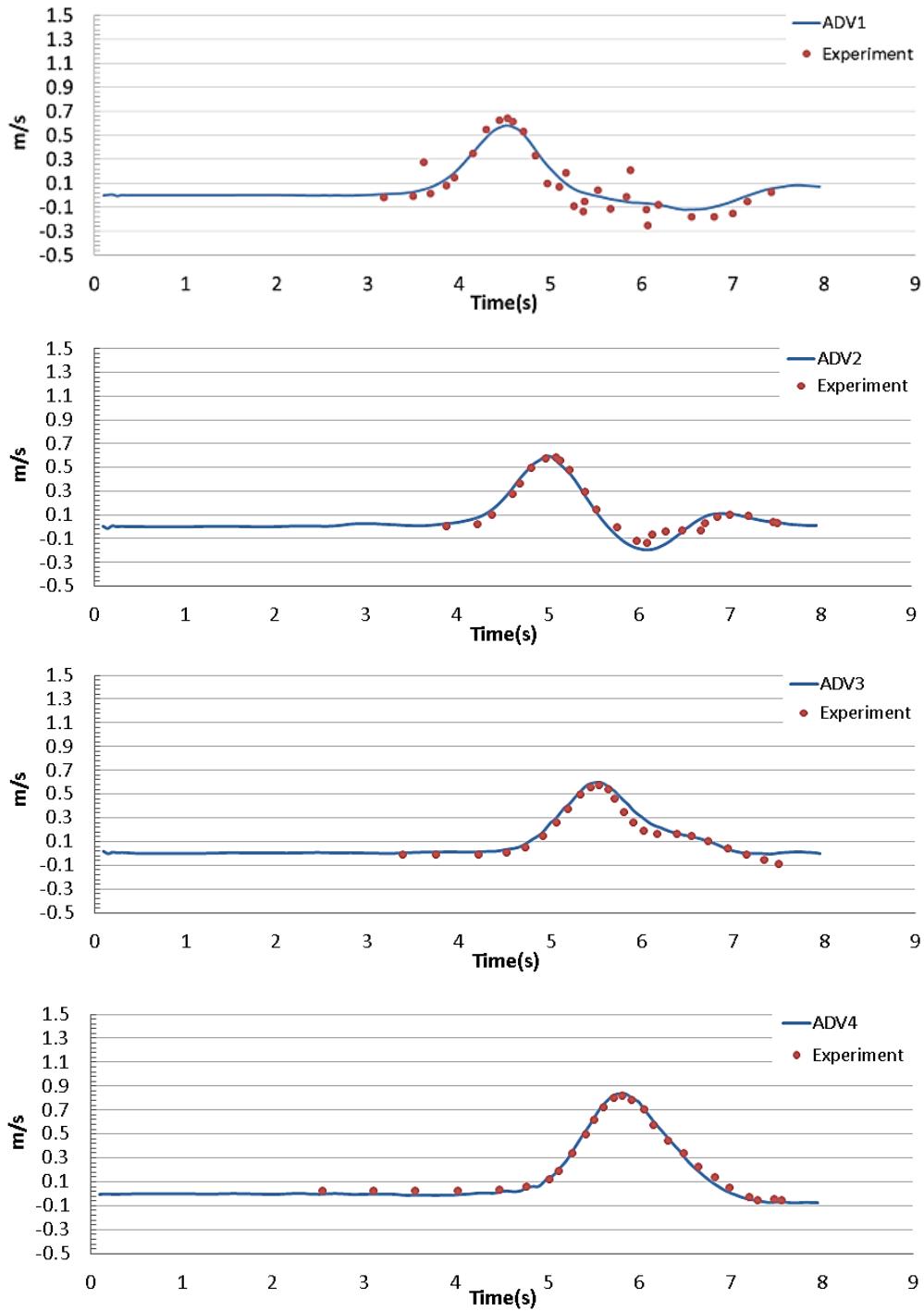


Figure 3-27- Experimental (dot line) and Numerical (Solid line) wave amplitude versus time at gage 5, 6, 7 when repulsive force is applied.

Figure 3-28 presents the numerical and experimental horizontal particle velocity at 5 ADV locations.

Figure 3-28- Experimental (dot line) and Numerical (solid line)
horizontal velocity versus time ant ADV1 to ADV5



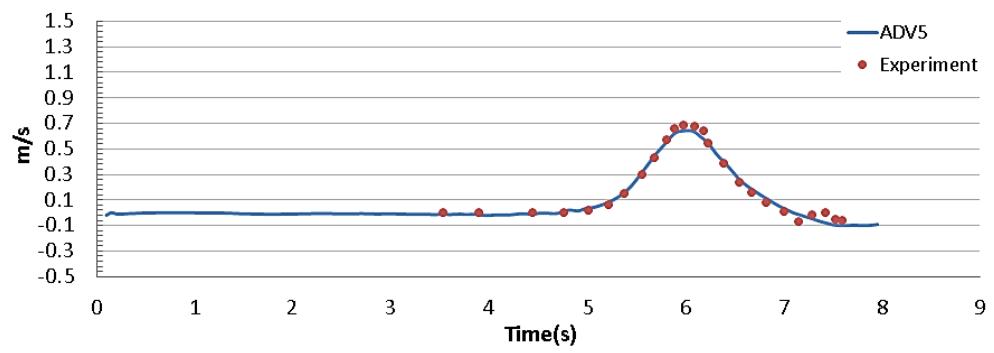


Figure 3-28 demonstrates the flawless compatibility between SPH output and experimental observations. When dynamic boundary particle is applied in SPH, it predicts the kinematic of the problem perfectly, even when the particle size is not extremely fine. However, SPH overestimates the total force on the cylinder given $dp=0.03$ m. Figure 3-29 compares the SPH estimation for the wave force on the pier and the experimental data reported by Zhang (2009).

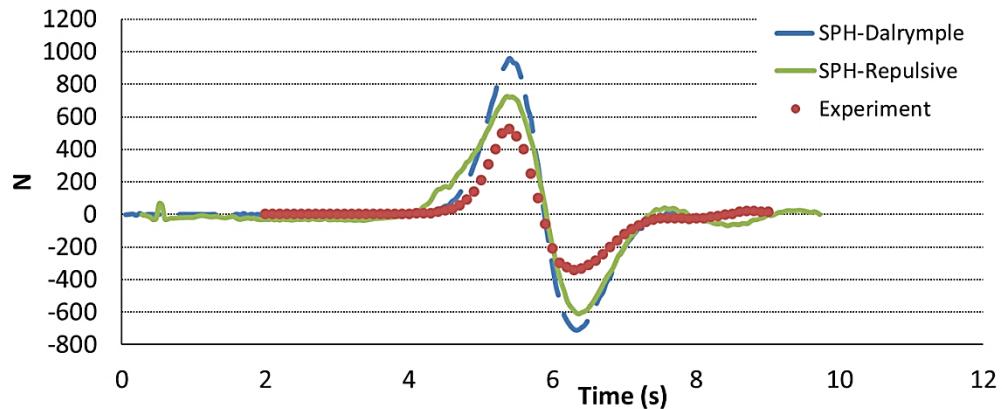


Figure 3-29- Force on the cylinder versus time: experimental (dot line), SPH with repulsive force BC (Solid line) and SPH with Dalrymple BC (dashed line)

Figure 3-29 shows that SPHysics calculates the magnitude of total maximum and minimum force about twice of the experiment reported by Zhang (2009). Note that it is well known that dynamic boundary particles produce strong friction and this can be of the sources of mentioned discrepancy. Several trials proved that even with lower wall viscosity, the magnitude of the force is computed about twice of the experiment. Regarding to the discussion in test case 2, a circular cylinder is categorized as a bluff body. For a bluff body,

pressure drag force plays more significant role than friction force does. That might be the reason of no change in the total force even with an inviscid fluid.

Due to the shape of object in this test, the boundary layer generation, separation point and the wake behind the cylinder become much more important at this point. As previously argued, the boundary layer thickness is a function of Reynolds number. If Reynolds number increases, the boundary layer becomes thinner. Using the information of particle velocity at ADV#1 and the diameter of cylinder and fluid viscosity, the Reynolds number of this problem is about $Re = (0.7)(1.22)/10^{-6} = 8.5 \times 10^5$, which is an extreme Reynolds number and the flow around the cylinder is under turbulent condition. Note that the ADVs record the velocity near the bottom, obviously, the velocity at mid depth or close to the surface should be greater than what we use in Re. number calculation.

Based on basic fluid mechanics, the contribution of pressure force in drag force is directly linked to the shape and the size of wake. The fluid pressure drops remarkably inside the wake and consequently the pressure difference of front and behind of the cylinder increases. This pressure gradient produces the form drag force. If wake is very small, the pressure gradient becomes trivial. As the wake grows, wider area behind the object has low pressure, so, the pressure gradient becomes higher. Hence, the generation of wake is very important for the force calculation. The separation point (or angle) determines the wake size. Figure 3-30 shows that for $Re=1 \times 10^4 - 5 \times 10^4$ the separation angle, θ_s , is less than 90° while for $Re > 6.0 \times 10^4$ it is bigger than or equal to 90° . Therefore, the wake size is bigger when $Re < 5 \times 10^4$.

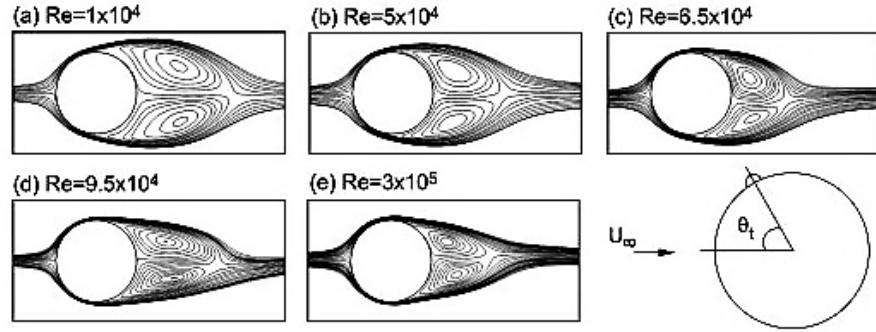


Figure 3-30- flow separation and wake generation as flow past a cylinder at different Reynolds number [Photo from Behara and Mittal (2011)]

It can be shown that the boundary layer thickness is in the order of

$$\bar{\delta} \sim \frac{D}{\sqrt{Re}} \quad (3-7)$$

In our problem, the boundary layer thickness is around 0.0013 m, which is extremely smaller than the particle size, $dp=0.03$ m. Regarding to the fact that the wake generation is the major phenomena, the particle size can be assumed critical here.

3.6 Test 6: Flow Past a Cylinder in 2D

To assess small particle necessity, in a couple of more tests, the flow pattern around a cylinder standing in a steady current is simulated in 2D. Morris et al. (1997) modified the SPH method to improve it for low Reynolds number flow. They showed that new viscosity treatment, kernel function, equation of state and fixed ghost boundary particles produced as accurate for following benchmarks, Couette flow, Poiseuille flow and flow through a periodic lattice of cylinder. They compared their test with either FEM (Finite Element Method) calculations or analytical solutions.

In the current study, two of their simulations are reproduced here. Here, the standard formulation of SPH is applied. Dynamic particle is used for circular cylinder which operates slightly different from ghost particles. Similar to Morris et al. (1997), flow is driven by a pressure gradient, and periodic boundary conditions are applied to all four sides of domain to make an infinite periodic array. Figure 3-31 depicts the geometry of the problem, where $a=0.02$ m and $L=0.1$ m. Particle size (spacing) is 0.002 m as Morris et al. chose. Morris et al. picked two value as Reynolds number, $Re=0.03$ and $Re=1$, where Reynolds number is defined

$$Re = \frac{V_0 L}{\nu} \quad (3-8)$$

Kinematic viscosity, ν , is $10e-6$ m^2s^{-1} in both tests. Table 3-2 presents the parameters of each test.

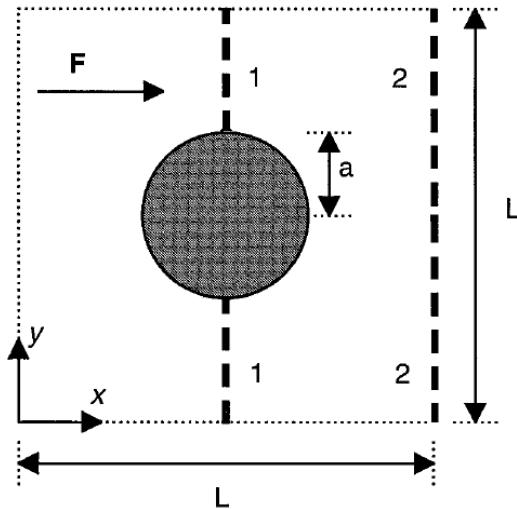


Figure 3-31- Flow past a single cylinder within a periodic lattice
[photo from Morris et al. (1997)]

Table 3-2- A cylinder in steady flow, Reynolds number (Re), velocity (V_0) and Speed of sound (c)

| Case # | Re | V_0 (m/s) | c (m/s) |
|--------|------|----------------------|----------------------|
| 1 | 1 | 5×10^{-5} | 5.7×10^{-4} |
| 2 | 0.03 | 1.5×10^{-4} | 1×10^{-2} |

Velocity distribution by Morris et al. (1997) and the present study are compared by plotting values along two paths described in Figure 3-31. Figure 3-32 and Figure 3-33 presents a comparison of velocity profiles obtained by Morris et al. (1997) and Marjani and Edge for lines 1 and 2. Our simulation is compatible with that of Morris et al. (1997), specifically for $Re= 0.03$. Both Figure 3-32 and Figure 3-33 show that the velocity by Morris et al. is slightly lower than ours. This may occurs due to the fact that Morris et al. applied fixed ghost boundary particles for simulating no-slip boundary, while in the present study

dynamic boundary condition is applied. Dynamic particles always have zero velocity while a ghost particle adapts its virtual velocity with adjacent fluid particles to ensure zero tangent velocity at boundary surface.

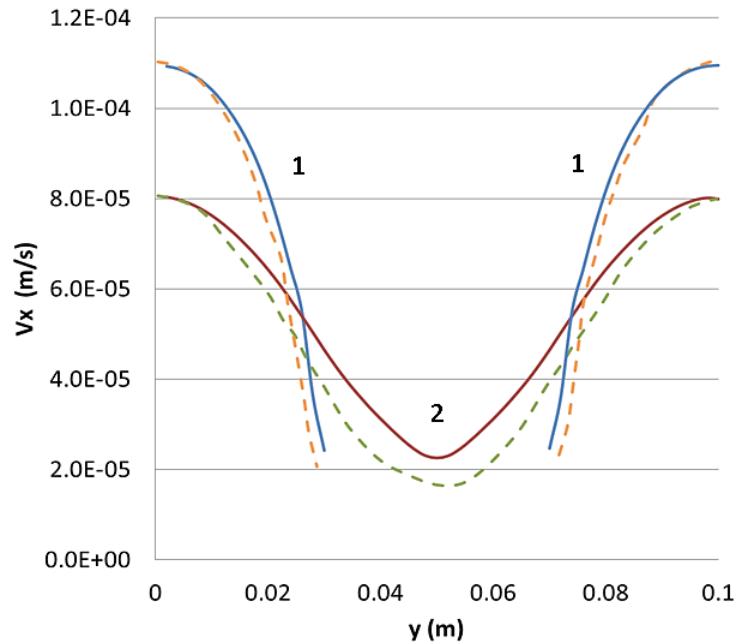


Figure 3-32- Comparison of present study (solid line) and Morris (dashed line) velocity profile along paths 1 and 2 for $Re=1$

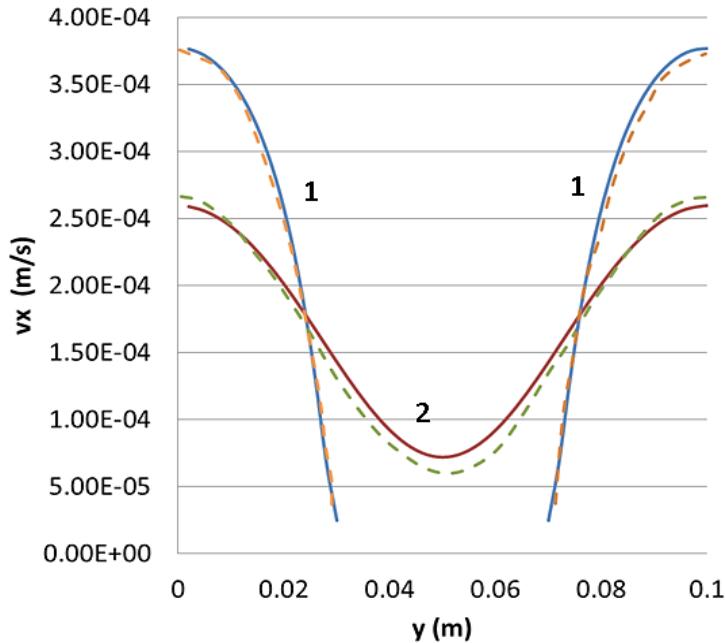


Figure 3-33- Comparison of present study (solid line) and Morris (dashed line) velocity profile along paths 1 and 2 for $Re=0.03$

A practical problem has a much higher Reynolds number. Therefore, higher Reynolds number of 10, 100 and 1000 are tried in the next attempts. Marrone et al. (2012) adopted $\delta - SPH$ scheme proposed by Antuono et al. (2010) and added a background pressure to the equation of state and applied the proposed specific treatment by De Leffe et al. (2011) for velocity field of fixed-ghost boundary particles to improve the SPH ability at Reynolds number ranging 10 to 1000.

Considering above modifications, a 2D flow past a circular cylinder in a steady current was studied by Marrone et al. (2012). Flow separation angle, the length of wake behind the cylinder and drag coefficient are compared with experimental values [Coutanceau and Bouard, 1977]. Here, we reproduced their output only for $Re=20$. The drag coefficient, C_D , is

obtained by $C_D=2F_x/\rho U^2 D$, where D is the diameter of cylinder and is always set to 1. The angle of separation, as shown in Figure 3-34, is defined where the tangential velocity in vicinity of cylinder changes abruptly its sign. Similar procedure is applied to determine the wake length L .

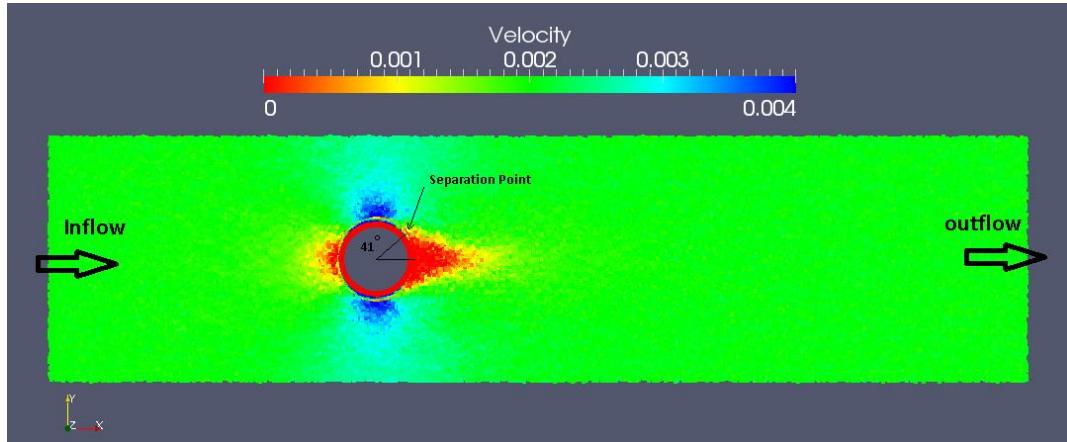


Figure 3-34- Velocity field and separation angle at $Re=20$ for a circular cylinder in a steady flow (present study)

Marrone et al. (2012) state that the minimum spatial spacing for grid-based solvers is fixed by the Reynolds-cell number, $Re_{\Delta x} = U \Delta x / v$. To avoid extra numerical diffusion and too small time step, $Re_{\Delta x}$ should be in the order of unity. Marrone et al. (2012) extended this criterion to their simulation considering particle size dp as $\Delta x \cdot U$, the steady inflow velocity, is equal to 2×10^{-3} (m/s) and kinematic viscosity is 10^{-4} (m^2/s) in the present study. Regarding above criterion, the particle size in this case should not be bigger than 1/20. Here, $dp=1/50$ m is used as an appropriate selection. In the present study, the initial condition of pressure and field velocity is dictated by potential flow theory. The model has periodic boundary condition

in flow direction to simulate inflow and outflow. After initial time, a constant velocity is assigned to those particles entering from left side to reach a steady flow. Figure 3-34 shows a snapshot of our simulation, presenting the constant inflow on the left and outflow on the right and shows that the separation angle is about 41° . Table 3-3 compares the current simulation with that of Marrone et al. (2012) and experimental data reported in Marrone et al. (2012).

Table 3-3- Numerical and experimental values of the separation angle (θ), normalized length of the wake (L/D) and drag coefficient (C_D) for $Re=20$.

| | θ | L/D | C_D |
|--------------------|--------------|-------|-------|
| SPH Marjani | 41.0° | 0.70 | 2.20 |
| SPH Marrone et al. | 43.0° | 0.75 | 2.19 |
| Experiment | 42.3° | 0.73 | 2.22 |

Several reasons may cause the small difference between our simulation and that of Marrone et al. (2012). For example, Marrone et al. (2012) used a more viscous flow ($\nu=10^{-3}$ m²/s) and consequently higher speed 2×10^{-2} m/s. Using more viscous fluid leads to more friction between fluid and the circular cylinder which cause separation at a larger angle. The other reason can be their slightly different approach of producing steady flow [for more information, refer to Marrone et al. (2012)]. Finally, ghost boundary particle is much more capable to simulate the no-slip boundary in comparison with dynamic boundary particle approach.

The smoke lines behind the cylinder for $Re=100$ is shown in Figure 3-35. Here, the velocity and kinematic viscosity are chosen 0.1 m/s and 0.001 m²/s, respectively to have $Re=100$ and regarding Reynolds-cell number formula, dp is 0.03 m. Marrone et al. (2012)

selected $dp=0.01$ m for their attempt and had 1.2 million particles which corresponds to a resolution $D/\Delta p=100$. They also carries out a numerical simulation for $Re=1000$ including totally 5.1 million particles.

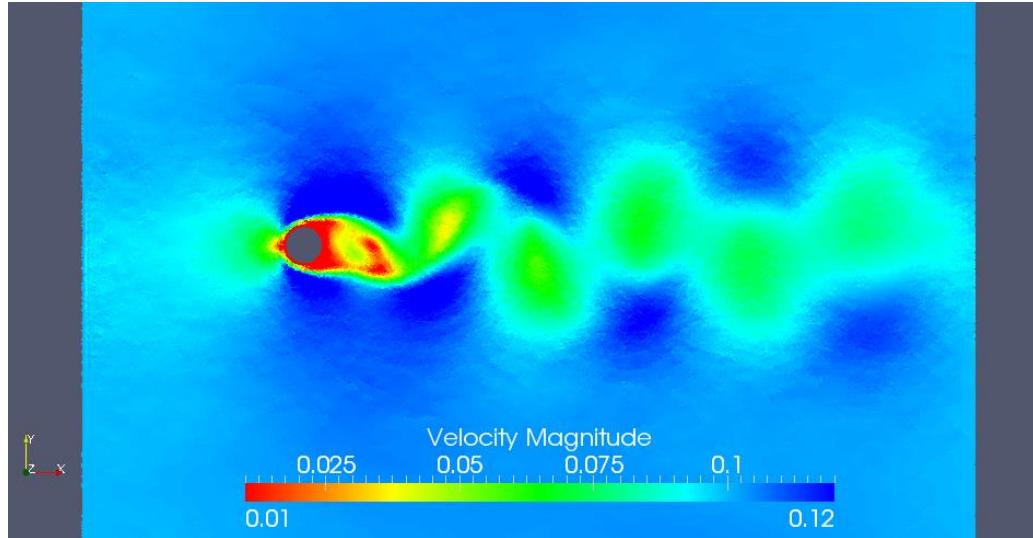


Figure 3-35- Velocity magnitude (m/s) as predicted by SPH in 2D at $Re=100$

A real and practical problem such as wave impact is known as a severe phenomenon in which Reynolds number is much higher than 100. As discussed previously in test case 5, the Reynolds number is in the order of $10^5\text{-}10^6$. Therefore, regarding Reynolds-cell number formula and tiny thickness of boundary layer at high Reynolds number, particle size should be smaller than 0.03 m as used previously. On the other hand, it is impossible to use smaller particles, such as 0.01 m, in the whole domain in a huge 3D tank. In the next chapter we will propose some techniques which lets us to use small particles and moreover, to speed up SPH computations.

Chapter 4

4 Enhancement in the Efficiency of SPH in 3D Modeling

4.1 Introduction

In chapter 3, several tests were simulated by SPH and compared with corresponding experimental data. It was demonstrated that SPH is highly robust in predicting kinematics, specifically for low Reynolds number flow. An optimum size of particle plays a key role in accuracy and efficiency of smoothed particle hydrodynamics approach. Coarse particles decrease the accuracy while fine particles increase the computational cost tremendously in three-dimensional simulation. For instance, test case 1-2 of chapter 3 proves that $dp = 0.03$ m is not small enough for accurate calculation of bam break impact on a circular cylinder.

Basic fluid mechanics believes that the pressure drag force dominates the friction force at high Reynolds number as the wake size behind that grows up. Therefore, small enough particle is essential for accurate generation of the wake. As an example, in some cases $dp = 0.01$ m or even $dp = 0.005$ m is needed, while it is impossible to use this small size for a huge 3D domain. This inability is not only because of computational time, but also the required hardware. As described in the previous chapter, total number of particle for a basin

like test 5, solitary wave impact on circular cylinder, would be about 27 million, which is impossible to be simulated with current technology.

In this chapter, either some existing techniques are improved or some new methods are suggested to improve the accuracy and efficiency of SPH remarkably. The following techniques individually or the combination of them will make it feasible to use smaller particle in large domain. These methods are briefly named 1) *variable underlying cells*, 2) *dividing wall* and 3) *mirror-type boundary condition*.

In the first method the whole domain is divided into a number of subdivisions. Each part has its own underlying cube (or square in 2D) size. It was described in chapter 2 that the domain is divided into underlying cells for NNPS process.

A dividing wall separates the fluid particles into two sides. In this case, different particle size can be used on each side without considering interaction between the particles with different smoothing length.

A mirror boundary prepares a condition in which a symmetric problem can be divided in two sections and only one half of domain is modeled. This is undoubtedly important for saving time and generally, reducing the computational cost such as physical memory and number of processors.

4.2 Variable Underlying Cells

4.2.1 Introduction

As discussed above and shown in chapter 3, small enough particle is essential to increase the accuracy. Our assumption is that the main reason in overestimating of SPH is the large particle size. In this chapter the effect of particle size on the prediction of solitary wave force on a circular cylinder by smoothed particle hydrodynamic method is investigated.

Since this specific problem has a huge basin and is modeled in three dimensions, filling the whole domain with fine particles is not reasonable. In most cases we do not need high resolution and accuracy everywhere but only in an interest area. Generally, accuracy and fine resolution is necessary in the impact area. The current study follows the method that Oger et al. (2006) applied in their study. The particles are initially defined so that its size is small at impact area and bigger elsewhere and the particle size and consequently the smoothing length varies very smoothly from the fine area to the coarse one and vice versa. Hence, the terms involving $\vec{\nabla}h$ are trivial. In addition, a kernel length is assigned to each particle which is constant in time, which implies that all terms with time derivative of h , dh/dt , are always equal to zero.

Because of two major drawbacks, the approach of Omidvar et al. (2011) was not applied in the current study. They proposed a predefined variable mass distribution with a constant h . The mass of coarse particles is m and they are typically replaced with eight lighter ones with a mass of $m/8$ (in 3D). Since the smoothing length of lighter particles is still h , the number of

particles within the support domain of each smaller particle is about 8 times more than normal. This increases the running time of those processors which possess lighter particles remarkably. Figure 4-1 describes that how using a large kernel length can increase the number of interactions between particles within their support domain tremendously.

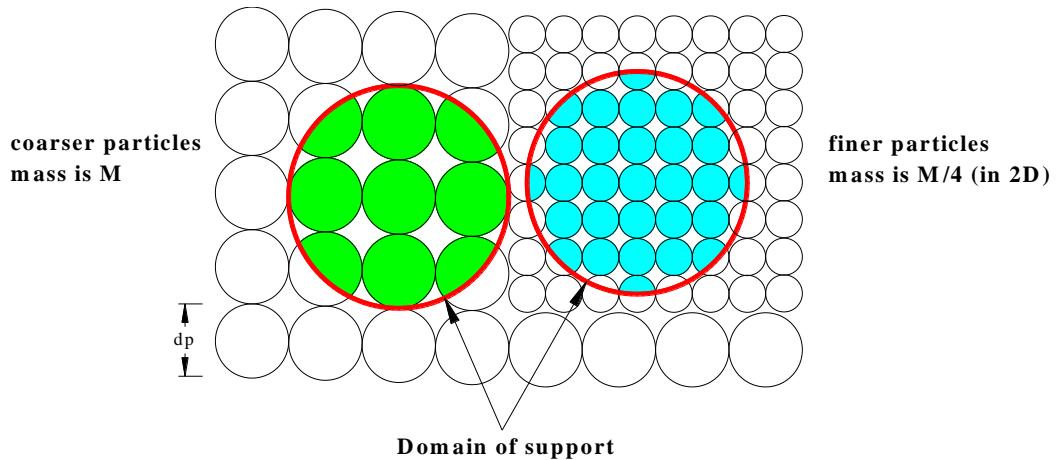


Figure 4-1- Variable mass particles with spatially uniform kernel length, h . The number of finer particles within the support circle is about 3.5 times greater than that of coarser ones.

Figure 4-1 presents the noticeable difference between the total number of fine and coarse particles within a fixed-size support domain. It shows that the number of lighter particles (blue circles) is about 3.5 time more than coarser ones (green circles). According to permutation theory, the total number of interactions between blue circles is about $29!/(29-2)! \times 2 = 406$, while the total number of interactions between green circle is only about $9!/(9-2)! \times 2 = 36$, given the number of green and blue circles are 9 and 29, respectively. Based on above simple calculation, using light particle with uniform smoothing length can increase the

total interactions and running time up to $406 / 36 \approx 11$. This ratio will be much greater if the calculations are redone for three dimensions.

Although locally replacing coarse particles by lighter ones decreases the total number of particles significantly, it increases the computational time due to massive interactions between fine particles. Thus, applying variable h is highly recommended to decrease the number of interactions. In variable h approach, each particle has its own smoothing length which is defined initially. In the present study, the smoothing length of each particle is defined as a function of particle size which is equal to the local particle spacing. The best choice for kernel length is $1.5\Delta p$. The smoothing length does not vary in time.

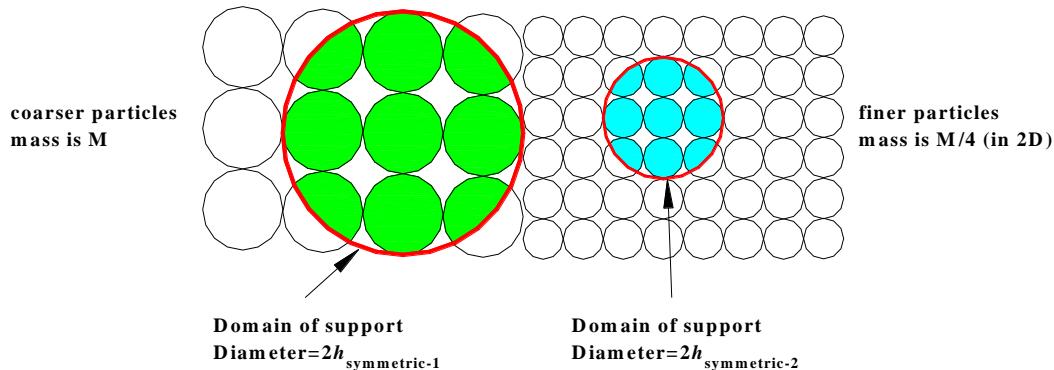


Figure 4-2- The total number of particles within the support domain is in the same order everywhere when each particles has its own smoothing length

In variable h approach, the symmetric combination of kernel length (h_s) plays a key criterion in NNPS process. This combination can be the arithmetic mean, a minimum, or maximum of (h_i, h_j) . Here, the minimum of h_i and h_j is chosen for the symmetric smoothing length. One of the advantages of using variable h is to keep the number of particles within

one's support domain the same everywhere, see Figure 4-2. Other than the first drawback, when a large smoothing length is assigned to a small particle, some unphysical errors are observed. In addition, Omidvar et al. (2011) introduced a virtual pressure to reduce particle clumping at the interface. According to above discussion, the particle size varies smoothly in space to avoid the errors due to ignoring $\vec{\nabla}h$ and smoothing length of each particle is linked to initial particle spacing.

Feldman and Bonet (2007) introduced a general procedure of particle refinement where a particle i is substituted with a set of daughter particles with a certain pattern centered at the position of i . Since the daughter particles are not merging after the refinement, the total number of particles increases rapidly through the time. Moreover, if variable h is assigned to the particles, a number of additional derivative terms related to kernel function should be included in the equation, as discussed before. If a fixed h is used everywhere, then too many interactions happen due to above discussion.

4.2.2 Fixed-size Underlying Cells

It was described in chapter 2 that the whole physical domain is divided into uni-size cubes in 3D (or squares in 2D). This division will speed up the process of NNPS tremendously. Using these cells, one particle does not need to check all particles in the domain at each time step to examine if it is inside its influence domain or not. In standard SPH, where the particle size is a constant value everywhere, the dimensions of underlying cells are uniformly defined $2h$. During this process, it is identified that each particle belongs to which cell and each cell knows its particles. Therefore, to find the nearest neighbor

particles, only those particles inside the surrounding cells are checked. Choosing $2h$ as the size of underlying cells guarantees that those particles beyond the surrounding cell are not in support domain of one particle. Thus, it is not required to check all particles.

Above technique is not efficient when variable- h approach is applied. Figure 4-3 illustrates a sample of problems with variable- h and uni-size underlying cells schematically in 2D. This sketch proves that why applying uni-size underlying cells for variable- h is not rational. In this figure, particle size varies from coarser ones on the left to the finer ones on the right side of domain, while the underlying cells are created based on the coarser particles. So, the number of particles inside the cell on the right side, where the fine particles reside, is much greater than normal. That means a fine particle on the right side of domain still needs to do a time-consuming task of checking too many particles inside the surrounding cells unnecessarily.

Figure 4-3 also demonstrates that only a few numbers of particle are inside the support domain of a fine particle on the right side, while it should check a huge number of points inside the surrounding cells. This task will be much more time-consuming in three dimensions.

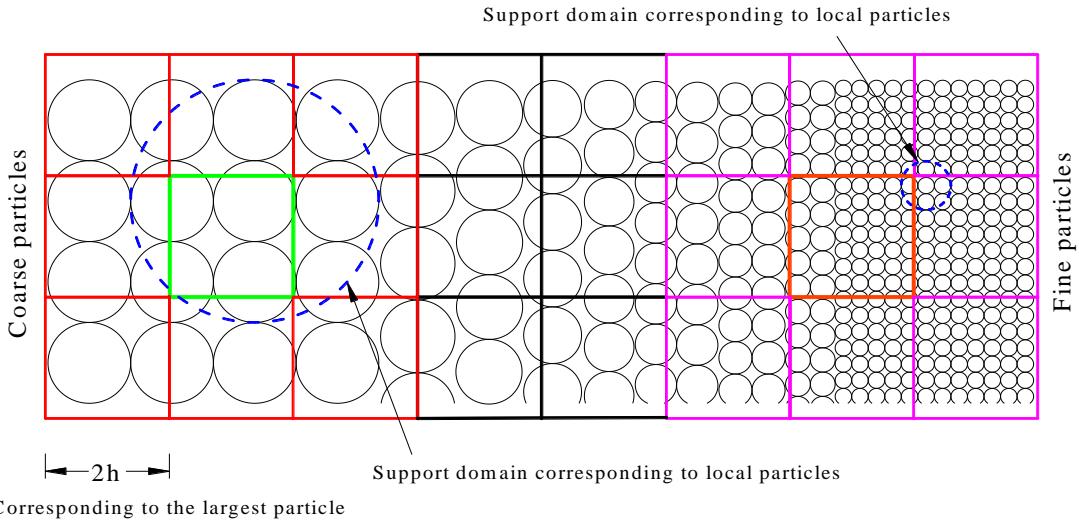


Figure 4-3- uni-size underlying cells is not efficient when variable- h is used

Another disadvantage is related to particle transferring between processors. It is known that the information of particles is transferred from one processor to its adjacent ones twice every time step. Regarding Figure 4-4, those particles within the first column and first row of one processor are transferred to the processor on the left and down side, respectively. When calculations are carried out in the goal processor, the results are sent back to the sender one. If the underlying cells are big in comparison with the local influence area, numerous numbers of particles are transferred whereas only a few of them will be inside the influence domain of other particles in the guest processor.

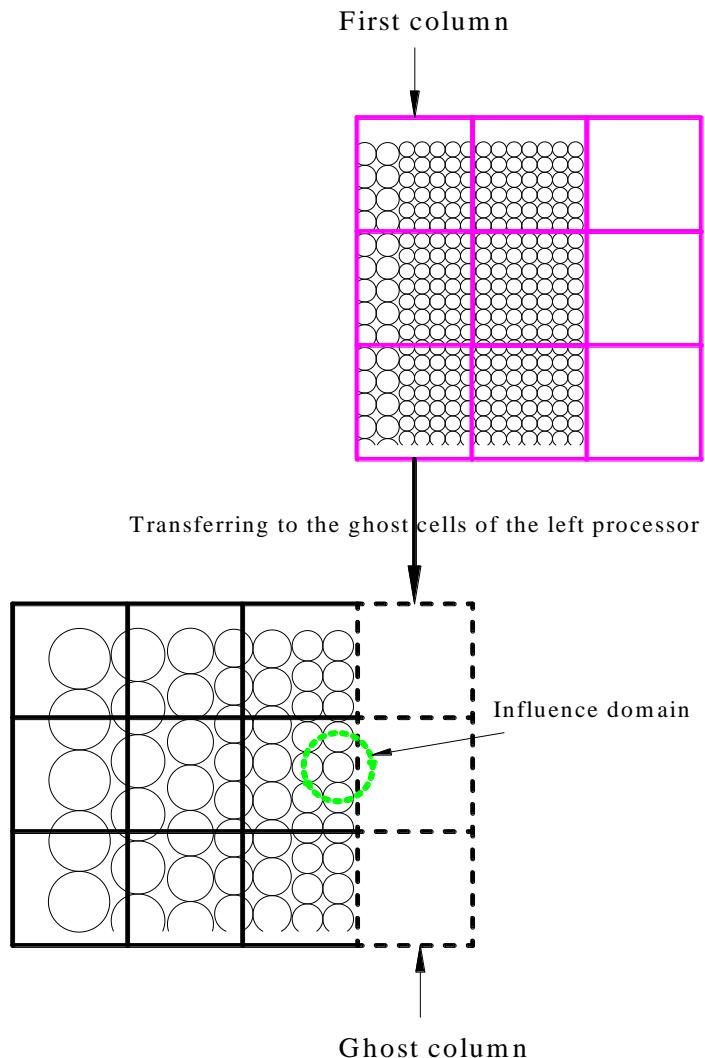


Figure 4-4- Many particles should be transferred to the adjacent processors, if the size of underlying cells is much greater than the local h

4.2.3 Variable-size Underlying Cells

Section 4.2.2 verified that fixed underlying cell size causes lots of undemanded tasks that extend the particle transferring and NNPS processes. The parallel version of SPHysics

facilitates growing up the idea of using variable-size cells. The new suggested approach links the size of underlying cells to the local kernel length. Since each processor completes all calculations independently after particle transferring, each processor can have its own value for underlying cells. This value can be similar or even equal for a couple of processors which contain almost similar particle size. Note that the size of the cells does not change in time but it may be different from one processor to another one. Figure 4-5 illustrates the initial idea of choosing variable cell size for three neighbor processors schematically.

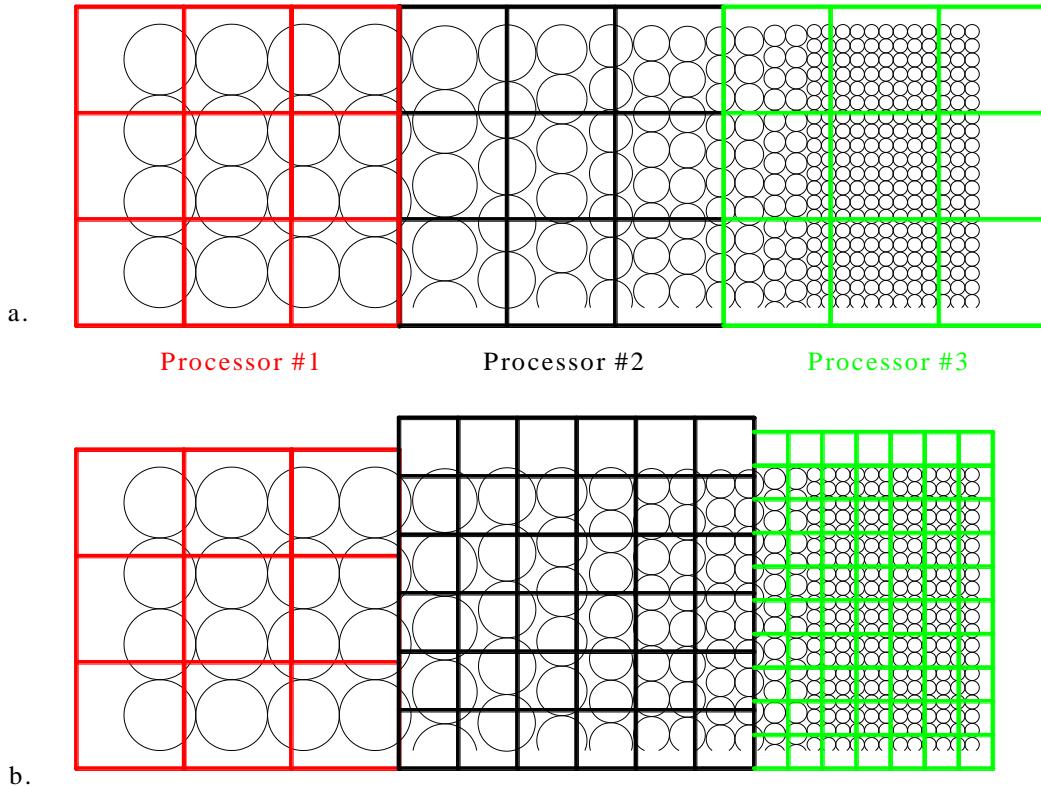


Figure 4-5- (a) fixed-size underlying cells vs. (b) variable-size underlying cells

Processor #1 has the largest cells, processor #2 has medium-size cell, and processor #3 has the smallest cell, see Figure 4-5-b. Therefore, the number of particles inside all cells is almost uniform in the whole domain. In addition, comparing Figure 4-5-a with Figure 4-5-b confirms that the transferring particles from one processor to its neighbor will be very optimal using variable cells. As an example, focus on the first column of processor #3 in both Figure 4-5-a and Figure 4-5-b. Particle transferring from processor #3 to processor #2 in Figure 4-5-a takes about 5 times more than that of Figure 4-5-b according to the number of particles in the first column.

Note that data transferring is an expensive task in parallel computing. Several simulations show that one processor spend almost equal or even more time on data transferring than SPH calculations. Since data transferring is done several times at each time step, optimizing this process is a significant step in improving SPH efficiency. A 3D wave propagation test in which the particle size varies from 0.01 m to 0.02 m is carried out to investigate the effect of variable cells on data transferring expense. At the first try the constant cells are set to the kernel length of coarse particles while for the second try, the size of cells are linked to the particle size locally. This test shows that the variable cells can decrease the cost of transferring up to 3.5 times. It is highly worthy to notice that in this test the total time spent on data transferring is about 1.2 times greater than SPH calculation as fixed cells are defined. The ratio of data transferring time to SPH computation time becomes about 0.75 when variable cells are generated. Note that the variable cells also reduce the SPH computation time noticeably (around two times). Twelve processors are utilized to model a wave basin with 0.3 m width and 0.45 m water depth.

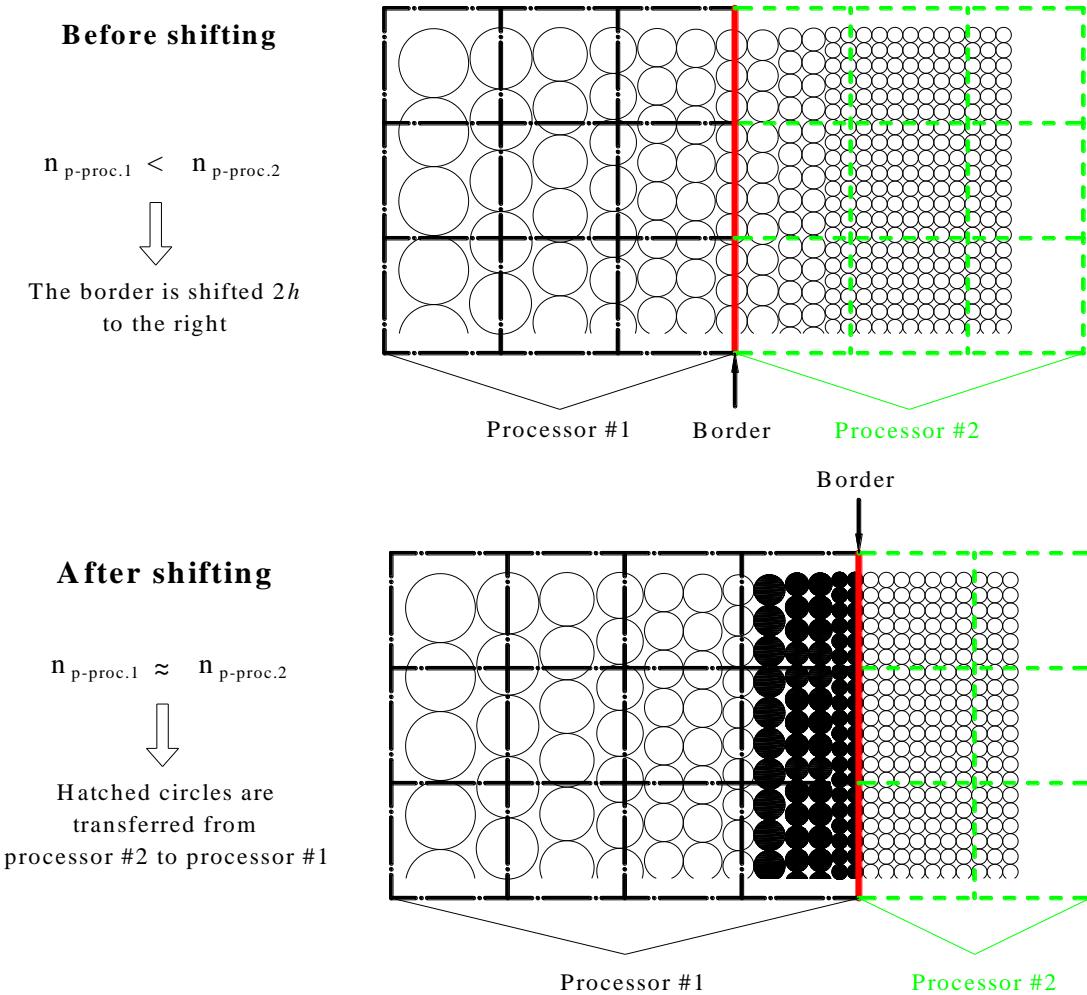


Figure 4-6- Dynamic load balancing process and border shifting with fixed-size cell

The above procedure has a fatal flaw. It is discussed in chapter 2 that the dynamic load balancing reduces the difference in the number of particles inside the processors. In this technique, the number of particles inside a pair of neighbor processors is compared. If the difference exceeds the limit, the border of processors is shifted $2h$ towards the processor with more points. Consequently, the number of columns on the processor with higher number of

particle reduces by 1, in contrast, that of the other processor increases by 1. Then particles are transferred between processors according to the new borders. Figure 4-6 presents the above procedure clearly. This is feasible when all cells have equal size. In variable-size cells, the cell size of two adjacent processors is not necessarily identity.

To fix the above drawback, a marker determines the border of processors in variable-size cells approach. Consequently, this marker guides the dynamic load balancing. As the difference in the number of particles of two processors becomes more than the limit, the marker, representing the border, moves slightly towards the processor containing more particles. This movement is less than the size of underlying cells of each processor. Any particles on the left side of the marker belong to the left processor and those on the right side belong to the right processor. Moreover, as the steps of border motion are smaller than the cells size, it yields a neat balance between processors. If the border had a shift equal to $2h$, that causes a huge number of particles transferring to the receiver processor. This may increase the number of particles in the receiver processor tremendously and force the border to go back to the previous position at the next step and consequently, the border has a uselessly back and forth motion.

In the novel method, the cells of adjacent processors may have an overlap which is not problematic. The idea of creating the marker is to avoid this issue. Figure 4-7 depicts how a marker plays an important role in determining the border of processors and in load balancing process. The solid circles on the left belong to processor #1 and smaller hatched circles on the right belong to processor #2. Some columns might be eliminated or created, as marker moves around the overlap area. In Figure 4-7-d, the most left column of processor #2 is

removed as the marker passed it or a new column is added to the right side of processor #1 as the marker passed the last column, see Figure 4-7-f.

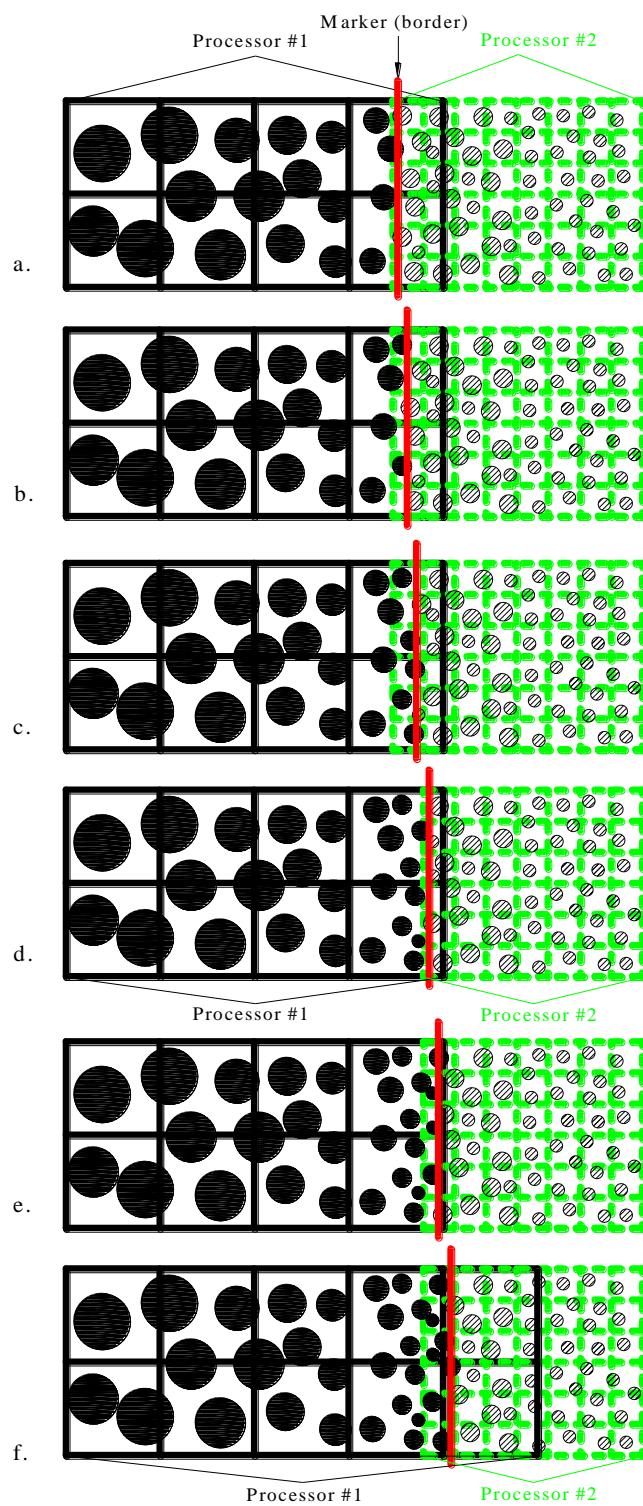


Figure 4-7- Dynamic load balancing process and border shifting with variable-size cell and employing a marker as processors border

4.2.4 Results and Conclusions

A couple of regular or solitary wave propagation on a beach tests prove that applying variable-size cell can increase the running time up to 3 for three-dimension tests and up to 2 for two-dimension tests. As the range of particle size in the whole domain increases, the advantage of using variable-size cell becomes much more rational. Comparing the running time of different simulations of solitary wave propagation in 3D with and without applying variable cells shows that when the largest particle is 0.03 m and the finest one is 0.015 m, the speedup is almost 3. As the finest particle increases to 0.02 m, the Speedup becomes about 2.2. These tests show that how effective and beneficial is to define the variable cells. For instance, a four-day running simulation can be reduced to two-day or even one and half-day if the size of underlying cells varies spatially as a function of particle spacing.

4.3 Dividing Wall

The main scope of this study is to propose the methods that facilitates and makes it possible to define small particle at critical locations and to reduce the running cost by decreasing the total number of particle. In the last section, the variable h methodology was employed and the innovative technique of variable underlying cells accelerates it noticeably.

Another technique which is effective in some cases such as wave propagation is *dividing wall*. This technique can be useful only for those cases in which the whole domain is occupied by fluid. Therefore, it is not helpful for a dam break test. Since our main focus is the solitary wave impact on the circular cylinder, this tool is considerably profitable. A

dividing wall divides the whole area in two parts. The physical process on only one side is our interest and the accuracy on the other side is not important at all. Consider a test in which the regular wave or solitary wave force on a horizontal or vertical cylinder is studied. The important area is only around the pipe or cylinder while the processes on the sloped beach do not matter. So, a wall can separate the whole domain in two sections. The area on the downstream of wall can be filled with very large particle. The size of these particles can be much different from the other side and their function is only to fill the downstream area. Note that this wall can move freely. The wave transfers its energy to the wall and forces it to move. The moving wall transfers its energy to the large particles and finally wave breaks on the beach. Figure 4-8 depicts the general idea of moving diving wall.

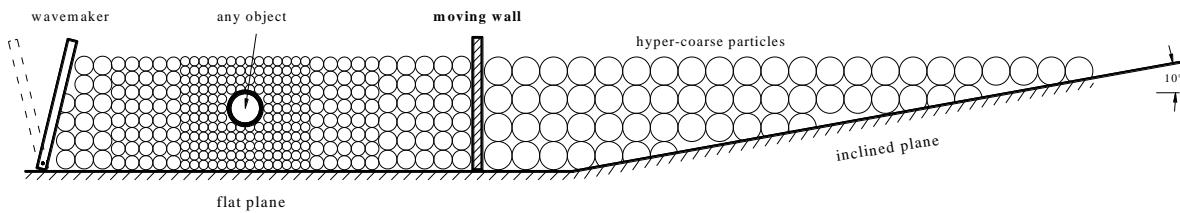


Figure 4-8- completely different size of particle are located at two sides of the moving wall

According to different size of particle on both sides of the wall, the smoothing length is also differs from upstream to downstream side of the wall. The interaction between those fluid particles in the vicinity of wall reduces significantly due to the existence of the wall and its thickness. The idea of locating a moving wall between two areas was built to reduce this interaction. The smoothing length of fluid particles on one side of the wall differs from that

of another side. Existence of the wall reduces the interaction of particles with completely different smoothing length to decrease the error of ignoring kernel length gradient terms.

Two types of walls can be defined. The motion of first type has one degree of freedom of a rigid body and the second type has two degrees of freedom. The second type is more accurate and appropriate when regular wave needs a back and forth motion, while, the first type is enough for solitary wave motion. Figure 4-9 illustrates these two types.

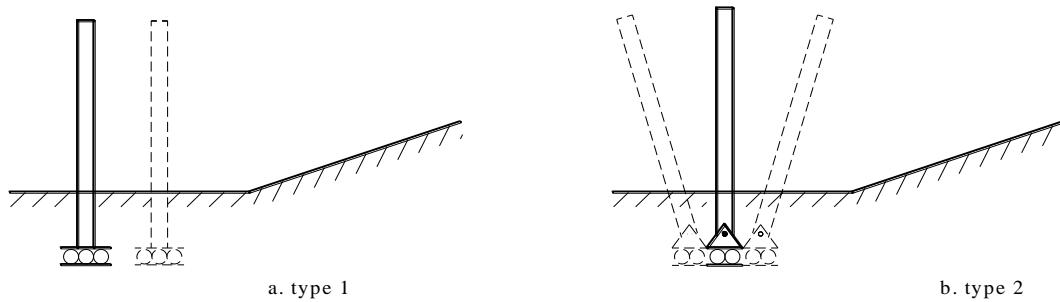


Figure 4-9- Two types of moving walls, a. type1: one degree freedom and
b. type2: two degrees freedom

Moving wall can also be combined with the variable smoothing length. This technique becomes very helpful and makes huge difference in total number of particles when the width of wave basin or the water depth is much greater than the average particle size. A simple example will clarify it. The solitary wave test has a basin with a width of 4 m and still water depth of 0.45 m. let us assume that the moving wall locates 1 m upstream of inclined plane and the slope of beach is 10 degree. The downstream side of the wall requires 510k particles with a size of 0.02 m to be filled while only 64k particles with a size of 0.04 m can fill it. Assuming the optimum number of particle within one processor is 30,000, 17 processors are

required for the former case but only 2 processors are enough for the latter case. It should be considered that reducing in the number of processors affects the communication time between them. Since at each time step processors send and receive their information to each other, a remarkable portion of computational time corresponds to transferring data between cores. Other than these all, the number of particle defines the size of output files which might be critical for storage or visualizing and making animation. This simple example shows that how effective is to use a dividing wall to reduce the computational cost tremendously.

4.3.1 Impact of Solitary Wave on a Single Circular Cylinder

Test case 5 in chapter 3 is simulated here. Variable smoothing length with variable underlying cells is combined with moving wall for the current attempt. Particle size varies from 0.03 m at the wavemaker and decreases gradually to 0.015 m close to the cylinder which is the impact region, then, it increases to 0.03 m as getting far from the critical area and close to the moving wall. Particles on downstream side of wall have a size of 0.04 m. The kernel length is defined as 1.5 times particle size and the underlying cell size is 4 times particle size which is bigger than $2h$. Figure 4-10 illustrates the initial particle configuration. Colors represents the particle mass. Combining separating wall and variable h makes it possible to have as fine particles as 0.015 m around the obstacle and totally 1.33 million particles. The model took about 34 hours to run on 64 processors of RENCI cluster.

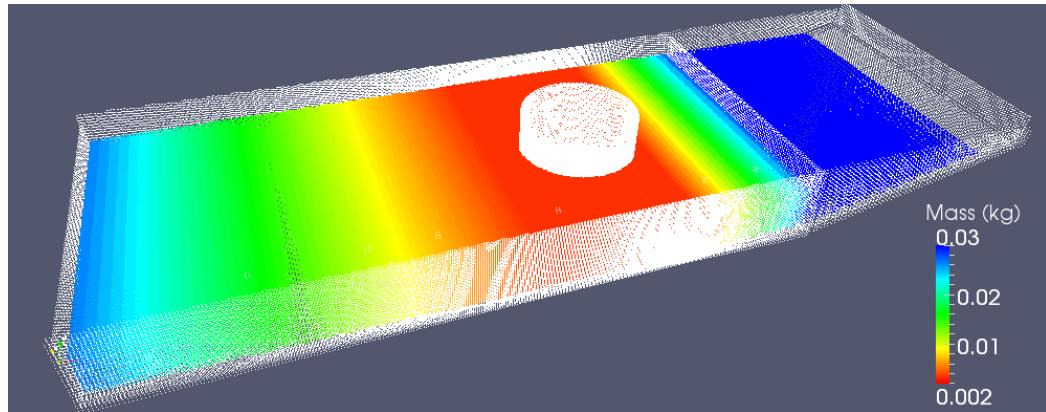


Figure 4-10- Initial particle arrangement with variable h and variable cell size.
Colors represent the particle mass

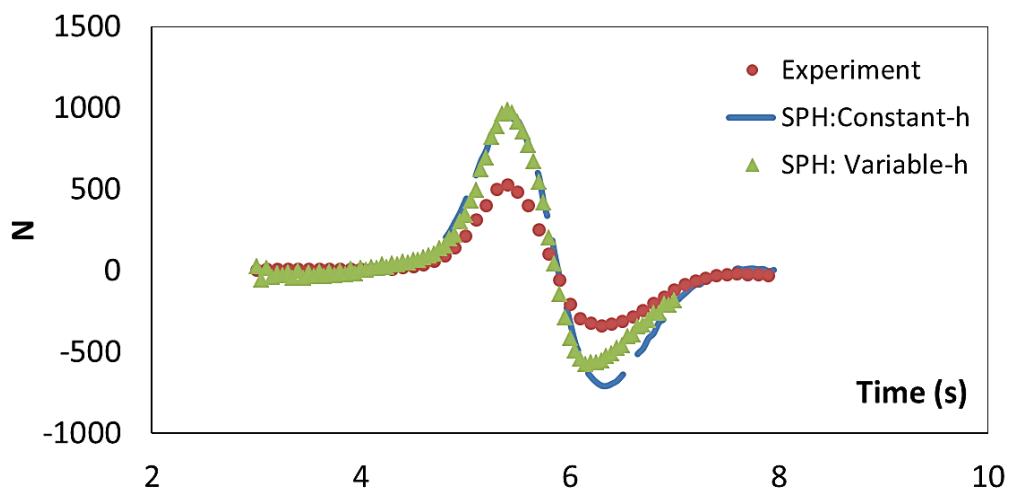


Figure 4-11- Total force on the cylinder by variable- h , constant- h and experiment

Above model finds the total horizontal force on the cylinder very similar to the model in which the particle size is constantly equal to 0.03 m. Figure 4-11 compares the total force on the cylinder when smoothing length is variable and constant with experimental observation.

Therefore, the above test concludes that the particle size of 0.03 m is appropriate for modeling the above test. This conclusion was the beginning step of searching for the actual reason of this discrepancy. More experiments of OSU will be simulated in the next section.

Figure 4-12 and Figure 4-13 validates the accuracy of wave height and particle velocity in the direction of flow at measuring locations, respectively.

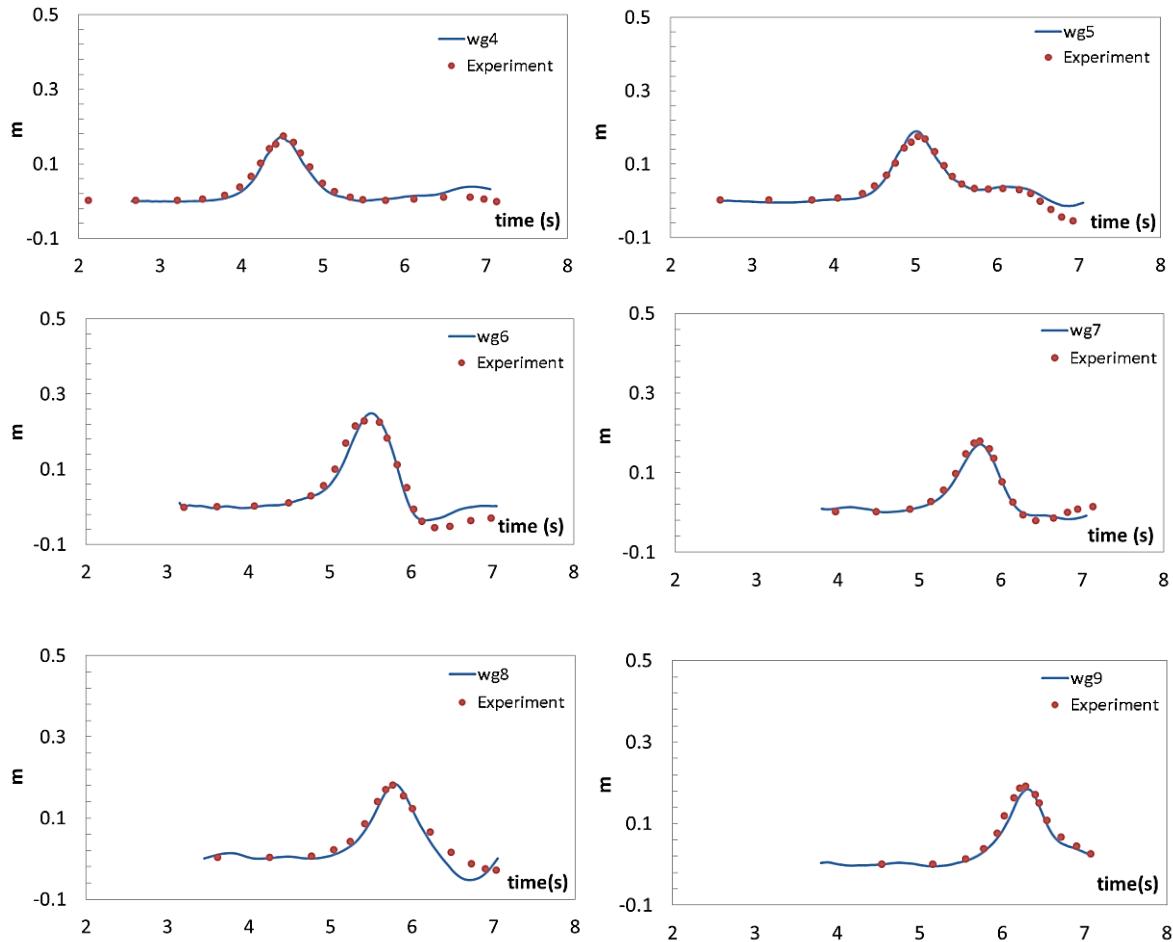


Figure 4-12- Experimental (dots) and Numerical (solid line) wave height at the measuring points

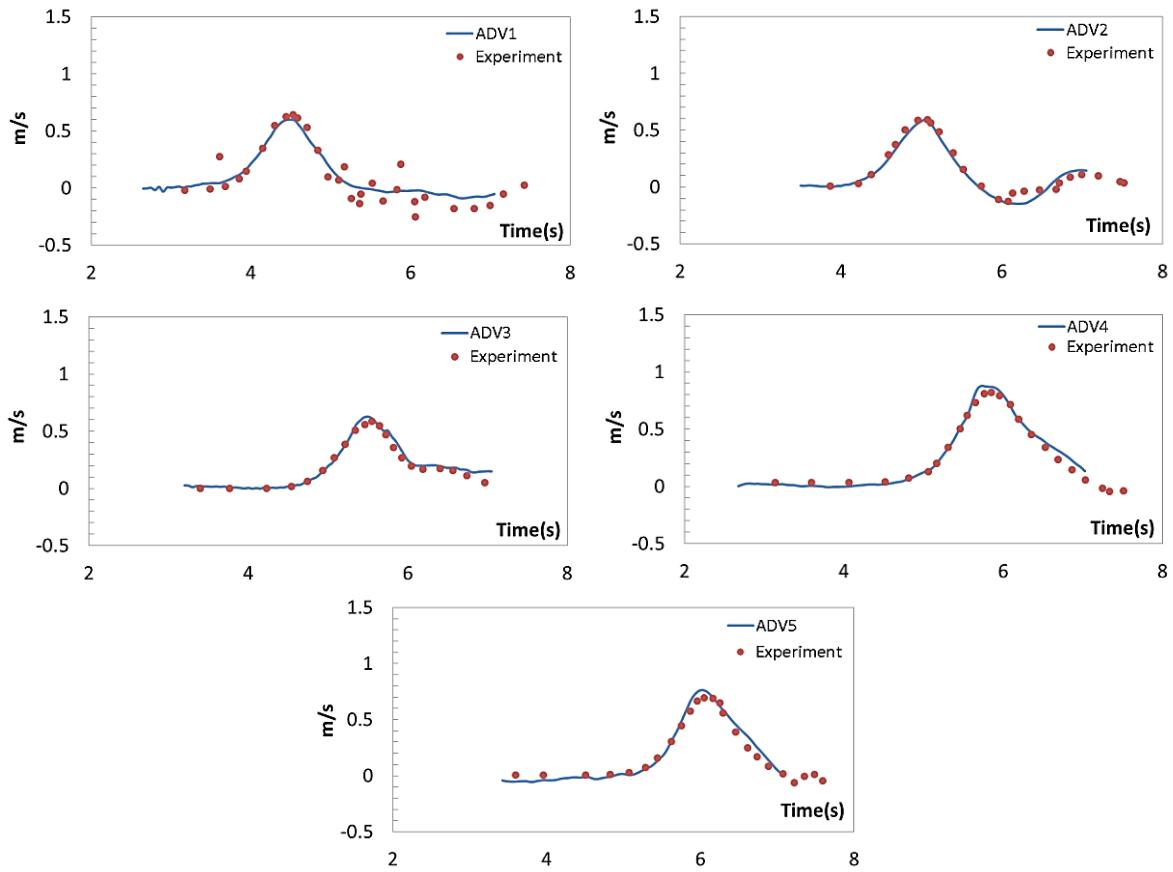


Figure 4-13- Experimental (dots) and Numerical (solid line) particle velocity in x -direction at the measuring points

Figure 4-12 and Figure 4-13 demonstrate that variable- h plus variable-size underlying cells technique predicts the kinematics of problem accurately.

4.4 Mirror Boundary Condition

4.4.1 Periodic BC vs. Mirror-type BC

Most of the time, an interested 3D test is a symmetric problem which can be halved into two sections. Then, only one-half of a problem is simulated. All 3D benchmarks in chapter 3 are symmetric, i.e. one-half basin can be modeled. This approach is very common in CFD and Eulerian mesh-based methods, while the current SPHysics is not able to do it. This technique is definitely one of the best approaches to reduce the total number of particles by half. In this way, either a problem can be modeled with half number of processors or it takes less time to be modeled with the same number of cores. It is worthy to repeat that reduction in the number of processors influences the communications between cores which is truly effective on the computational cost.

Note that mirror boundary is different from periodic one. Periodic boundary condition makes an infinite domain in the direction in which the periodic BC is applied. Figure 4-14 shows that what happens if periodic boundary condition is used for the half-basin dam break test.

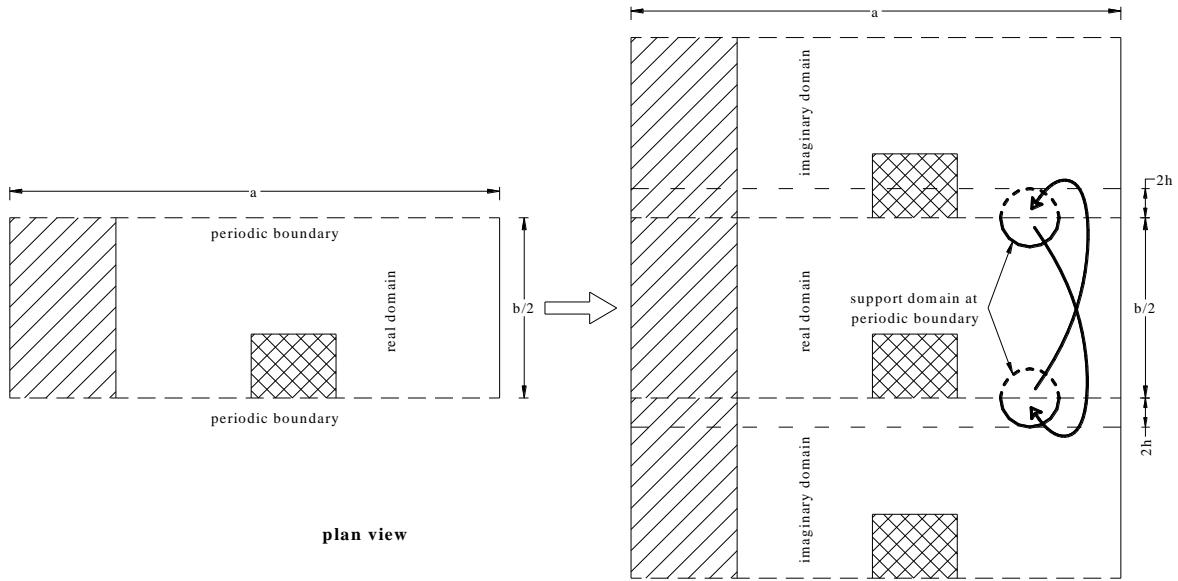


Figure 4-14- Periodic boundary condition is not suitable for modeling the half-basin

To create a periodic domain, SPHysics transfers the information of particles in the first row of underlying cells to the other side of the domain as the most last imaginary row. Reversely, it transfers the information of particles resided in the last row of underlying cells to the other side of the domain as the most first imaginary cell. If a particle leaves one side it will enter the domain from the other side, see Figure 4-14.

In contrast, a mirror-type boundary condition is able to rebuild the original domain from the half basin. Employing mirror-type boundary, a symmetric domain can be divided into two congruent parts with respect to the symmetry line. The last released version of SPHysics is not able to mirror a half-basin problem to simulate the whole domain.

4.4.2 Methodology

Regarding Figure 4-15, it is assumed that ideally, two congruent parts of a symmetric problem behave completely similar and each particle in one side has a mirror-image particle with similar properties. Colors in Figure 4-15 represent the properties of particles.

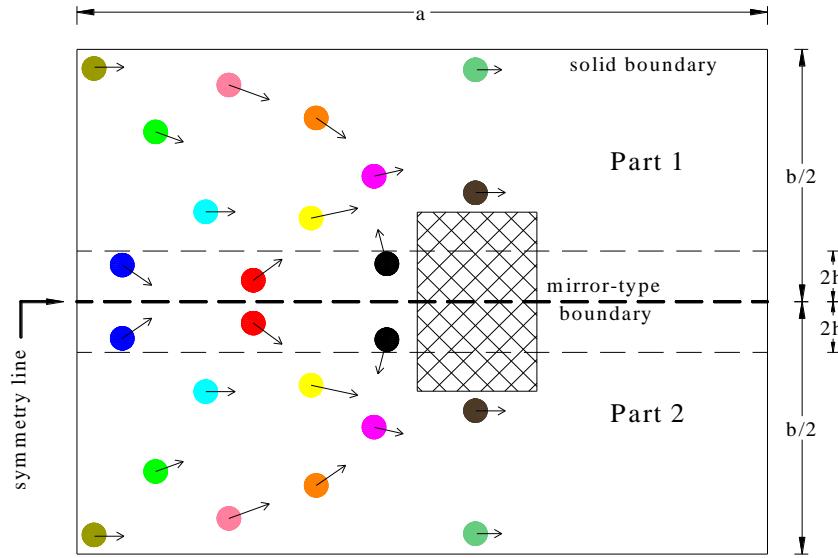


Figure 4-15- Two divided parts are congruent with respect to symmetry line in a symmetric problem

Note that the particles of part 1 interact with those in the same part and at most with those particles located in the last row of underlying cells of part 1. They never interact with the particles beneath the last row of part 2. In other words, only the particles in the first row of part 1 interact with the particles of last row of part 2. Since each particle has an image in the opposite side, all particles in the first row of part 1 may interact with their images in the last row of part 2. Therefore, to divide the original problem into two congruent parts and do the

computation only for part 1, the information of particles in the last row of part 2 is needed at each time step. Thus, they use their own information as that of particles in part 2.

An imaginary cell, called cell number 0, is created spatially before cell number 1. At each time step, those particles in the first cell are mirrored with respect to the symmetry line. These image particles are definitely located in the imaginary cell. All properties of that particle (including pressure, density, mass and velocity in tangential and vertical direction) are copied to the imaginary one except the velocity vector component normal to the symmetry line. The normal velocity is reversed. Two mathematical conditions should be satisfied at the symmetry line. Normal fluid velocity must be zero

$$\vec{v} \cdot \vec{n} = 0 \quad (4-1)$$

where \vec{n} is a horizontal unit vector normal to the symmetry line. It results that the normal velocity components are zero at the boundary. All other variables are symmetric relative to the boundary

$$\frac{\partial \alpha}{\partial n} = 0 \quad (4-2)$$

Figure 4-16 depicts this procedure. After each time step, image particles in the cell 0 are removed and at the beginning of next step new image particles are created based on the updated properties of real particles. This approach reduces the number of particles tremendously, specially, when the basin is very wide in the direction perpendicular to the symmetry line, such as Figure 4-16.

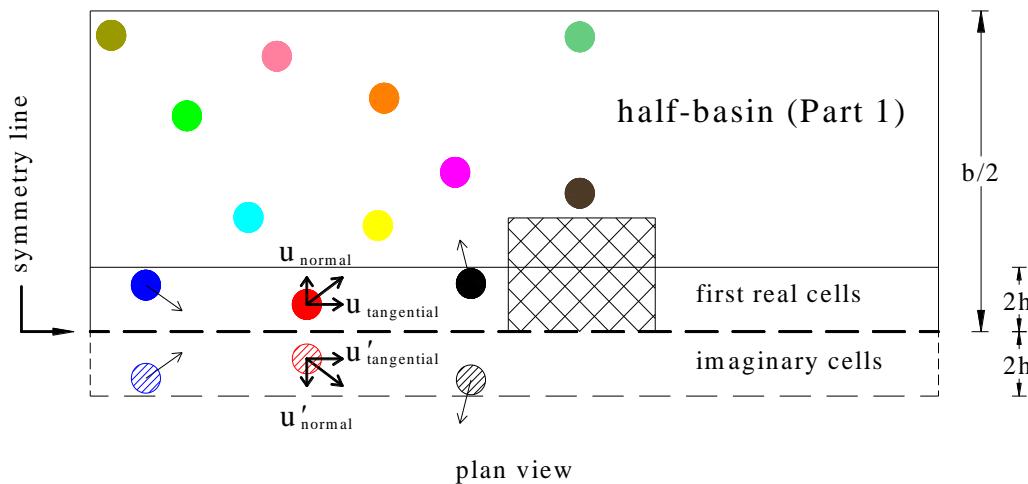


Figure 4-16- Particles in the first underlying cell are mirrored with respect to the symmetry line

4.4.3 Dam Breaking in 3D (Half-Basin)

Test 1-1 of chapter 3 is remodeled here. In that test, a dam break force on a tall structure was compared with real data. Since the rectangular obstacle is located on the symmetry line, this test can be modeled with mirror boundary. All settings including particle size, initial condition, initial particle configuration, kernel function, etc., are similar to the original test except the geometry of problem which is divided by two along the symmetry line. Similar to the original test, total force in flow direction on the object and particle velocity in front of object are validated by experiment. Besides, the half-basin model is compared with full size test. Figure 4-17 shows the initial configuration of half-basin dam break test.

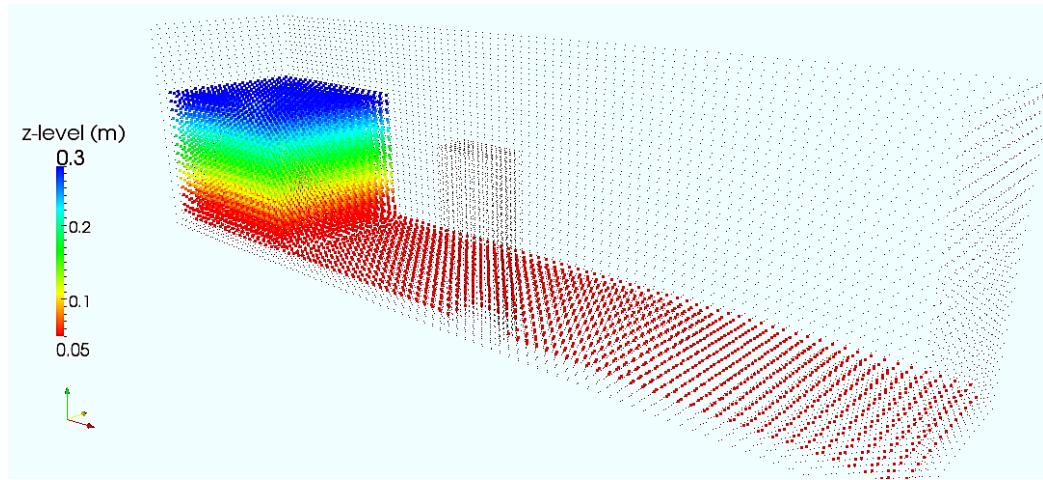
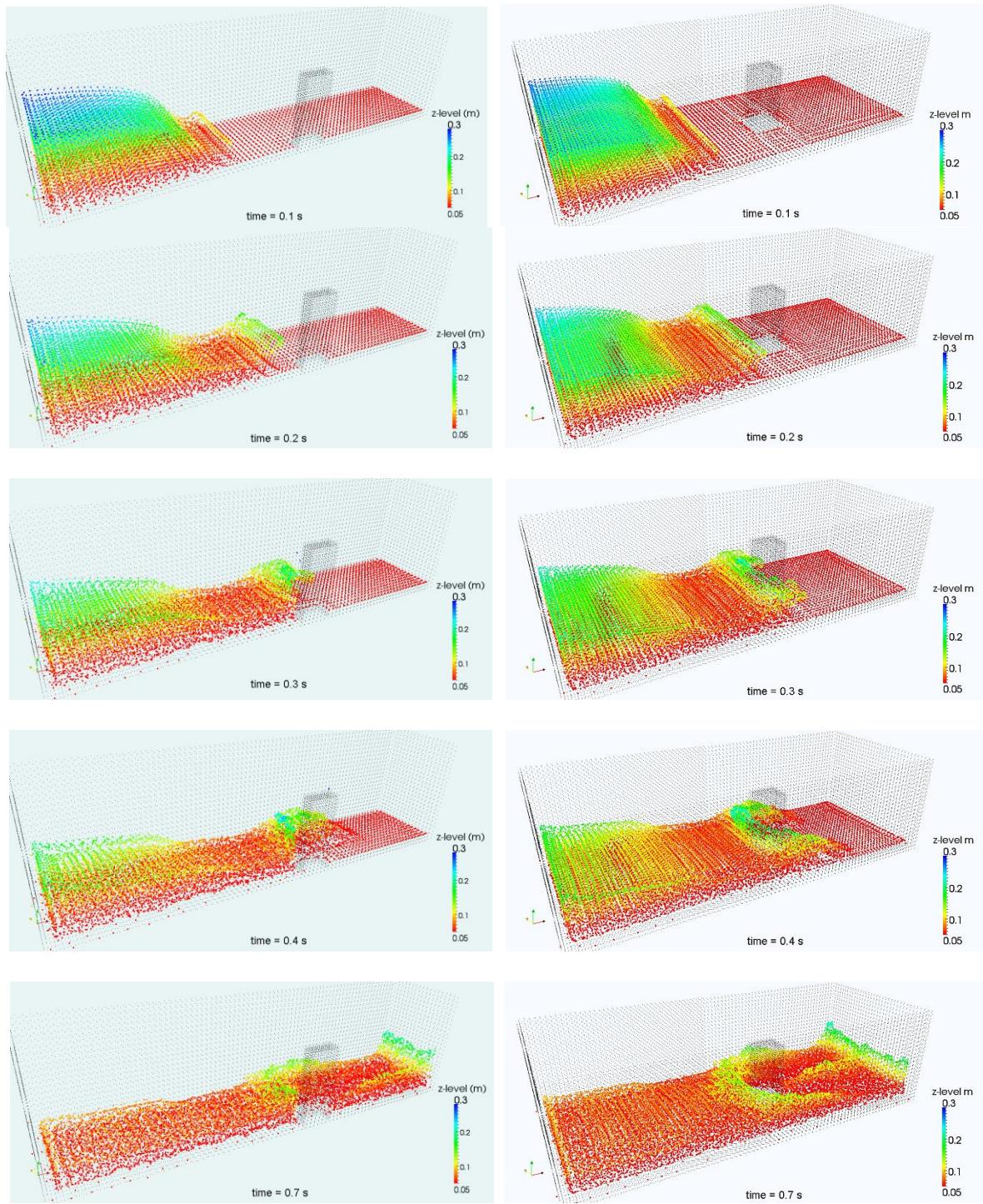


Figure 4-17- Initial particle configuration of half-basin dam breaking test by mirror boundary condition

The width of rectangular box is 0.34 m and its length is equal to the original problem, 1.68 m. Total number of particle in the present test is about 22K which is expectedly equal to the half of original test, which is about 46K. According to the fact that linked-list algorithm yields $N_p \log (N_p)$ interactions between particles at each time step, reducing the number of particles by a factor of 2 decreases the total number of interactions by a factor greater than 2. In another word, the relation between computational time and the number of particles is not a linear function. As an example, the original dam break test takes about 9600 seconds to get done while the half-basin dam break takes about 3600 seconds which is about 0.37 times of full basin. Although the number of particles included in the half basin model is about the half of full size test, its running time is less than the half of running time of full size model. Figure 4-18 visualizes the half-basin versus full-basin model in time. Colors represent the water z -level.

Figure 4-18- Time history of dam breaking, left panel: half-basin, right panel: full basin



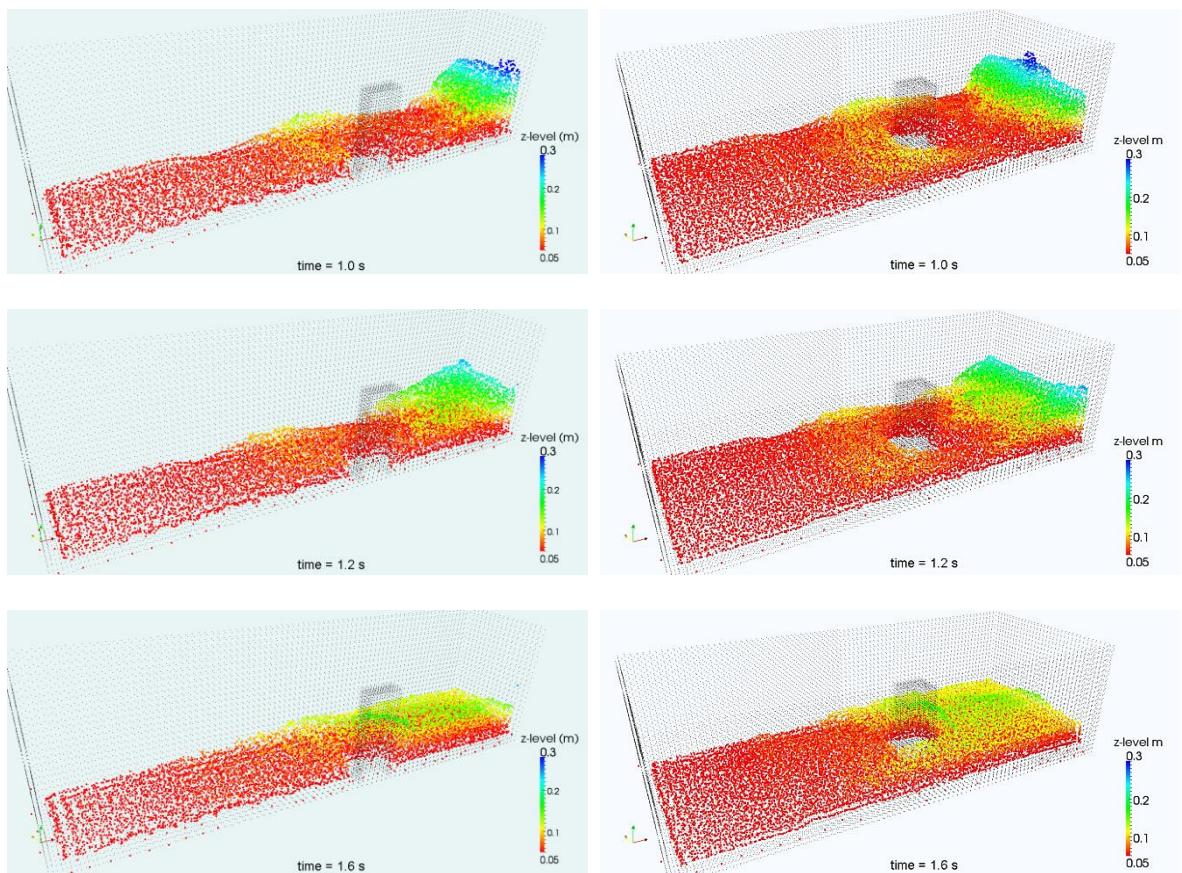


Figure 4-19 and Figure 4-20 compares the outputs of half-basin model with full-basin one and experimental data for two different particle size, $d_p=0.0225$ m and $d_p=0.01$ m. The fluid velocity in the direction of flow is in very good agreement with full-basin model and experiment. As discussed in previous chapter, the maximum velocity will be increased if the particles size decreases. The importance of the current comparison is only the perfect similarity between full-basin and half-basin model.

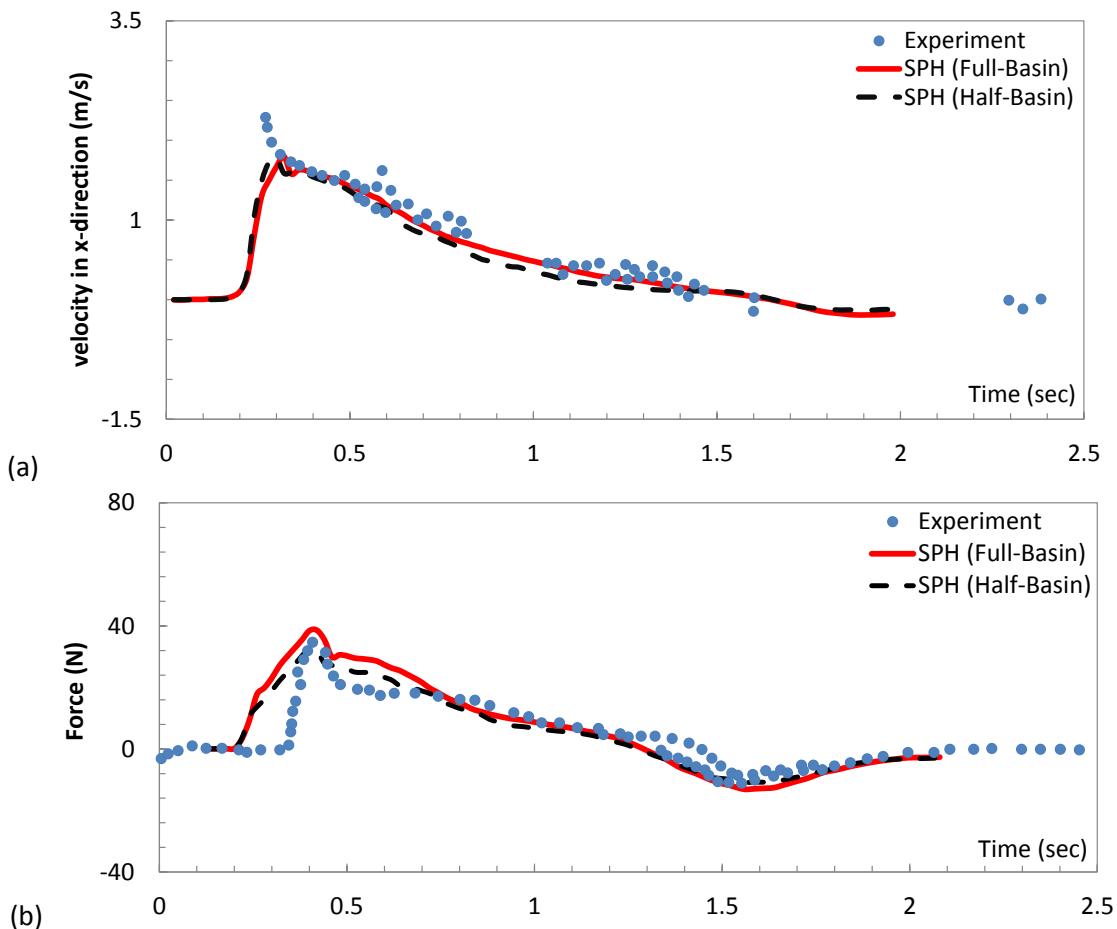


Figure 4-19- Comparison between half-size model and full-size model and experiment for $d_p=0.0225$ m, (a) Flow velocity in front of the structure, (b) Total force on the obstacle

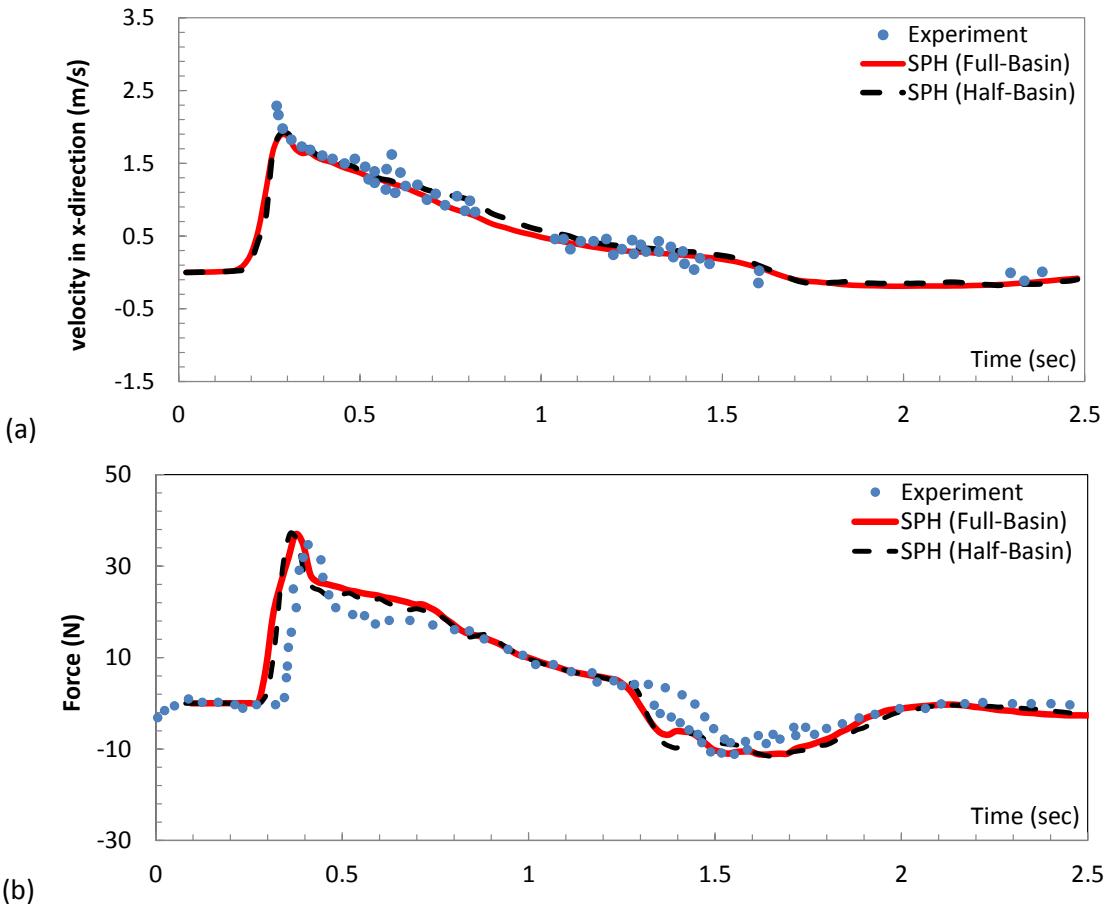


Figure 4-20- Comparison between half-size model and full-size model and experiment for $dp=0.01$ m,
(a) Flow velocity in front of the structure, (b) Total force on the obstacle

4.4.4 Impact of Solitary Wave on a Single Circular Cylinder (Half-Basin)

The symmetric properties of test 5 in chapter 3 let us to apply half-basin model for this problem. Section 4.3.1 concluded that $dp=0.03$ m gives the best trade-off between accuracy and computational cost. The width of half-basin model is 4 m which presents a full-basin problem of 8 m wide. A wider basin reduces the effect of lateral solid walls on the flow

pattern around the cylinder. Hence, the half-basin model allows choosing a wider basin than previous. Total number of particles is about 650K which is equal to half of full-size domain with a width of 8 m. The model takes about 6 hours to run on 64 processors of the RENCI cluster. The same model, which was run on 96 processors, took almost the same time to get done. Although increasing the number of processors decreases the number of allocated particles to each core, the communications between processors increases. For greater number of particles, increasing the total cores may speed it up. Initial particle configuration is shown in Figure 4-21.

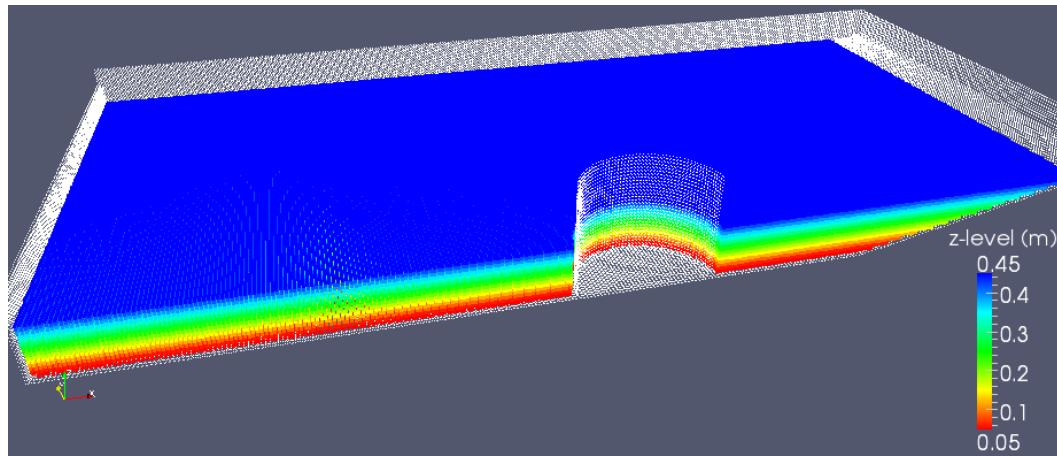


Figure 4-21- Initial particle configuration for half-basin solitary wave impacting a circular cylinder

Figure 4-22 visualizes the above simulation in time. Again, the color represents the z -level.

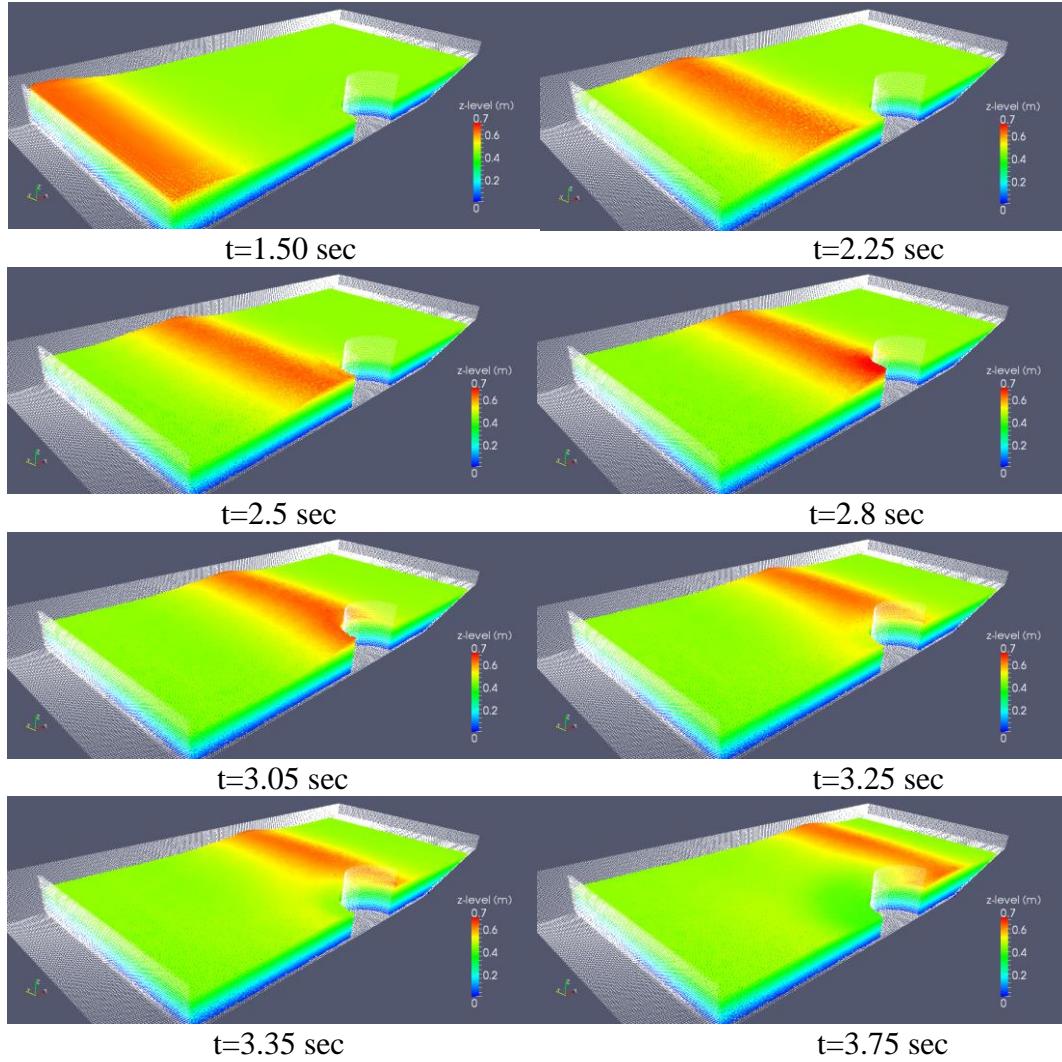


Figure 4-22- Time history of solitary wave impact on a cylinder by half-basin technique

Figure 4-23 compares the laboratory observation with the half basin model. The only obvious difference between full-basin and half-basin model is the minimum force. Half-size model yields slightly smaller minimum force than full-size model does. This difference is also seen for other cases with deeper water. Among all scenarios, the worst case belongs to $d=0.45$ m and $\alpha = H / d = 0.4$ in which the difference between the minimum forces is about

15% of maximum force. Later, it will be shown that the estimated force by half basin-model is in very good agreement with correctly converted experimental force, see Figure 4-28.

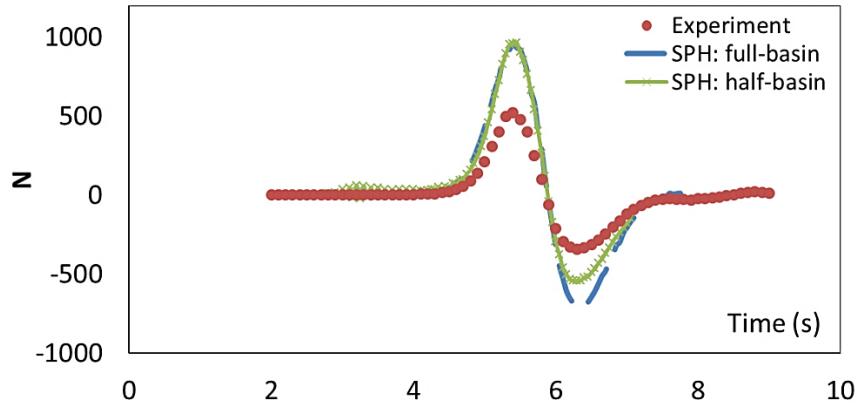


Figure 4-23- Total force on the single cylinder by experiment (dots), SPH: full-basin (dashed line) and SPH: half-basin (solid line with star)

Figure 4-24 and Figure 4-25 present the water surface elevation and particle velocity of half-size model. They confirm that the half-basin model can exactly reproduce the full-size model. So hereafter, all other solitary wave tests are simulated in half-basin size. Moreover, comparing Figure 4-24 with Figure 3-26 and Figure 4-25 with Figure 3-28 demonstrates that the half-basin model is slightly more accurate than the full-basin model. This preference might be achieved by enforcing the symmetry at the symmetric line which may not be satisfied necessarily in the full-size model.

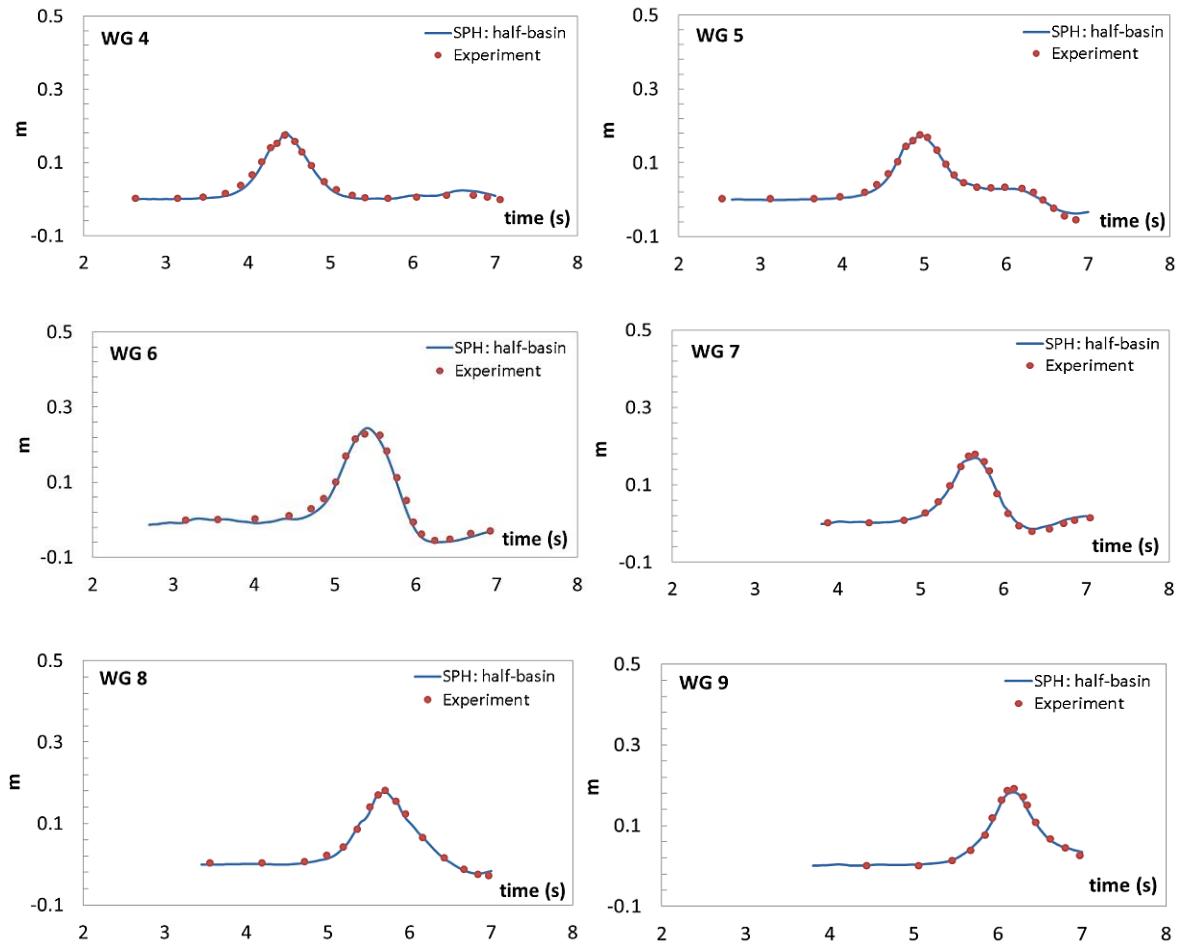


Figure 4-24- Water surface elevation at 6 wave gauges by experiment (dots) and SPH: half-basin (solid line)

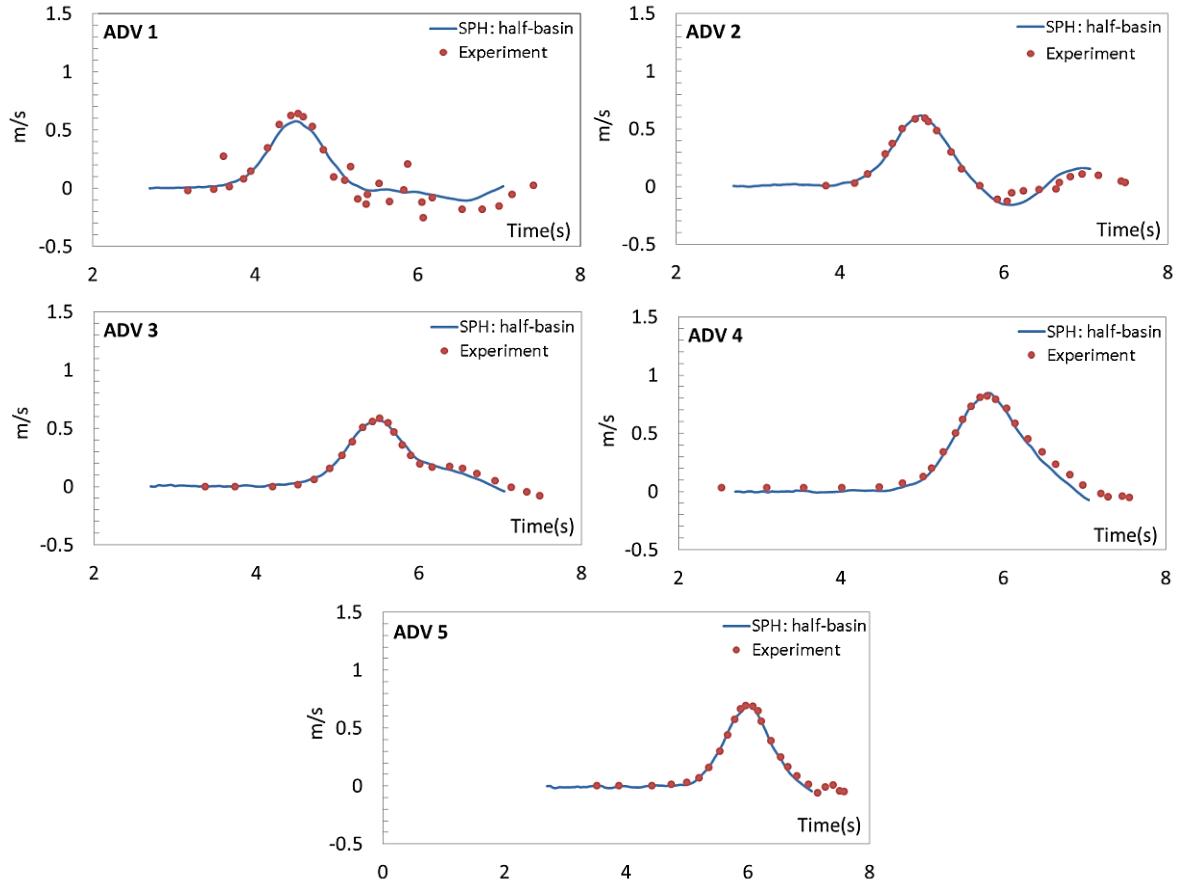


Figure 4-25- particle velocity in the direction of flow at 6 ADVs by experiment (dots) and SPH: half-basin (solid line)

4.4.5 The Impact of Solitary Wave on a Single Cylinder (More Tests by Half-Basin Model)

All above simulations agree that SPH predicts the kinematics very well while it computes the total horizontal force about twice of the Zhang's experimental data. In other words, the reported force by Zhang (2009) is about half of SPH calculation. It is very suspicious and hard to believe that SPH can predict the velocity field and water surface in the whole

domain, specifically around the cylinder, perfectly, while there is a huge gap between experimental data and numerical results for horizontal force. Note that, even using $dp=0.015$ m close to the cylinder did not change the computed force significantly. To find the actual reason of this suspicious discrepancy, some more simulations were carried out about the impact of solitary wave on single cylinder.

As mentioned in test 5 of chapter 3, various wave conditions and water depths were tried at OSU laboratory. Here, three more cases are simulated by 3D-parallel SPHysics. Particle size in all current simulations is 0.03 m. The settings of current tests are presented in Table 4-1. Applying mirror boundary condition, the half-basin model is used for all following tests.

Table 4-1- Numerical settings for various tests of solitary wave impact on a single cylinder

| Test No. | Water depth (m) | Wave height (m) | Numerical half-basin width | Total number of particles |
|----------|--------------------|--------------------|-------------------------------|------------------------------|
| 1 | 0.60 | 0.24 | 3.5 | 770 K |
| 2 | 0.60 | 0.36 | 3.5 | 770 K |
| 3 | 0.75 | 0.30 | 3.5 | 970 K |

Thanks to mirror-type boundary condition, it is possible to model those tests with deep still water which were not very easy to simulate with full-basin model previously. Figure 4-26 compares the computed total horizontal force on the cylinder with Zhang, 2009 graphs.

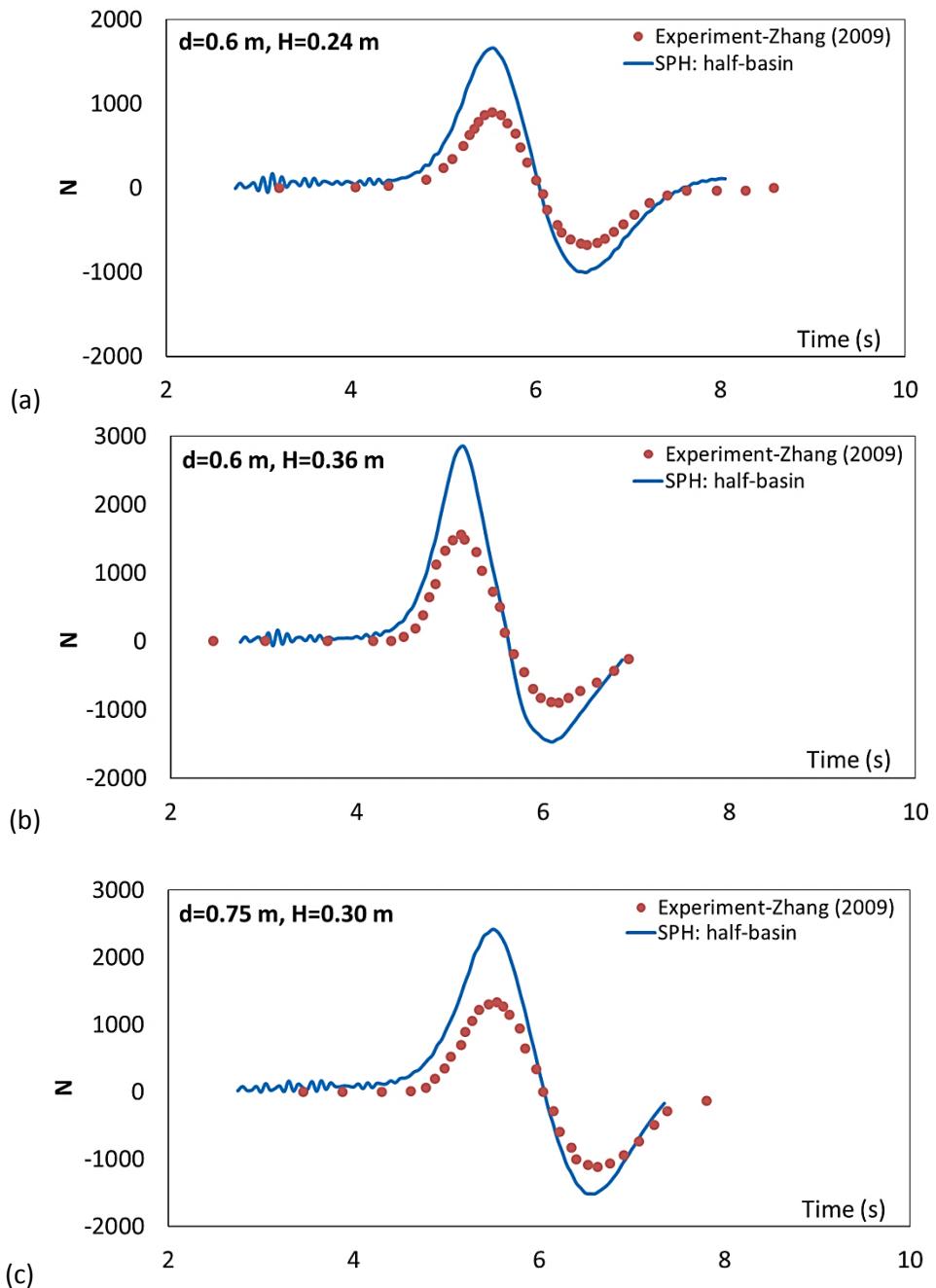


Figure 4-26- Total horizontal force on the single cylinder under different test settings,
(a) $d=0.6 \text{ m}, H=0.24 \text{ m}$, (b) $d=0.6 \text{ m}, H=0.36 \text{ m}$, (c) $d=0.75 \text{ m}, H=0.30 \text{ m}$

All above plots agree that the maximum force reported by Zhang (2009) is almost about the half of SPH calculation. This is very suspicious since kinematics of problem is predicted very accurately by SPH. Further study of Zhang's thesis and OSU reports demonstrate that the force on the cylinder is reported as two strain gauge measurements. Many calibration tests were conducted to find a relationship between strain gauge value and actual force on the cylinder. Zhang (2009) analyzed the strain gauge values and found the total force on a cylinder based on his converting formula. Note that there are two strain gauges installed, one is on the upper side of cylinder and the other on the bottom side. Remember that the ratio of SPH calculation and Zhang's data is two. It is very probable that this factor of two comes from the error in Zhang's conversion process. Morison equation can give us some general idea about the order of expected force on a single cylinder under a given wave condition, although this equation is only fully trusted when the diameter of cylinder is very small comparing with wave length.

Zhang used depth-average horizontal velocity, Equation (1-1), and Morison equation to find the drag and inertia coefficient (C_d and C_m).

$$\bar{u} = \frac{C \eta(x, t)}{d + \eta(x, t)} \quad (4-3)$$

where C is the wave celerity and d is water depth and η is the wave elevation. Table 4-2 reports some of estimated C_d and C_m for a single cylinder. In this table the Reynolds number of each test is calculated and the drag and inertia coefficient based on Sorenson's, 2006 formulation are also mentioned.

Table 4-2- Estimated drag coefficient Cd and inertia coefficient Cm for the impact of plane solitary wave on a single cylinder by Zhang (2009)

| Water depth <i>d</i> (m) | Wave Height <i>H</i> (m) | Drag Coefficient, <i>C_d</i> | | Inertia Coefficient, <i>C_m</i> | |
|-----------------------------|-----------------------------|--|----------|---|----------|
| | | Zhang | Sorenson | Zhang | Sorenson |
| 0.45 | 0.18 | 2.33 | 0.7 | 0.35 | 1.5 |
| 0.60 | 0.24 | 2.12 | 0.7 | 0.47 | 1.5 |
| | 0.36 | 1.91 | 0.7 | 0.31 | 1.5 |
| 0.75 | 0.30 | 2.10 | 0.7 | 0.54 | 1.5 |

First, the inertia coefficients by Zhang are all less than one which is very questionable, because, it is known that $C_m = 1 + K$, where K is a positive number. Thus, the inertia coefficient should have a value higher than one. Since the motion of solitary wave is a highly accelerated flow, the portion of inertia force is significant. As an example, for $d=0.6$ m and $H=0.36$ m and given $C_d=0.7$ and $C_m=1.2$, the maximum of drag force, inertia force and total force on the cylinder are 380 N, 2900 N and 3000 N, respectively. This simple example shows the importance of the inertia force in this specific problem. According to the Morison equation, the total force on the cylinder should be about 3000 N which is much closer to the SPH estimation in comparison with experimental data reported by Zhang. Therefore, the above discussion enhanced the idea of incorrectness of the experimental force calculated by Zhang, 2009. The other evidence can be found in Mo's thesis (2010).

Mo's thesis is the ONLY document that is linked and referenced on the official website of NEES (Network for Earthquake Engineering Simulation) and related to the above experiments. Mo only reports the experimental data of the test with $d=0.75$ m and $H=0.3$ m.

It is interesting that his experimental force, which is very close to SPH calculations, is about twice of that of Zhang. Figure 4-27 plots the total force on the cylinder computed by SPH versus the experimental data reported by Zhang and Mo.

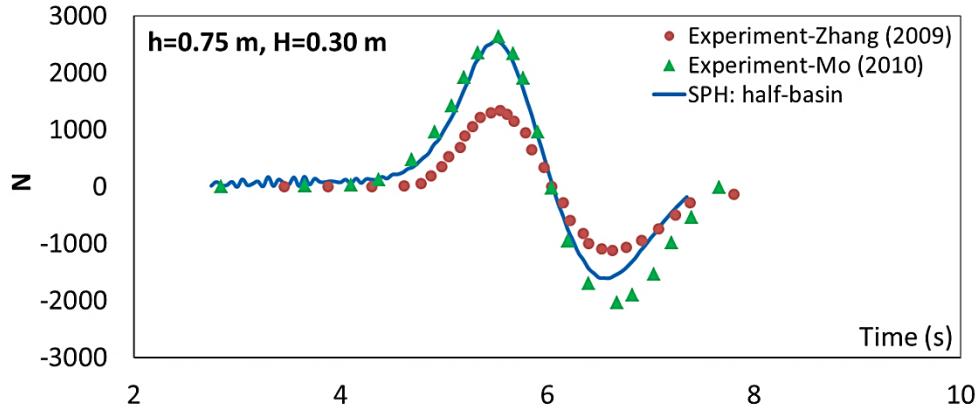


Figure 4-27- Total horizontal force on the single cylinder when $d=0.75$ m, $H=0.30$ m

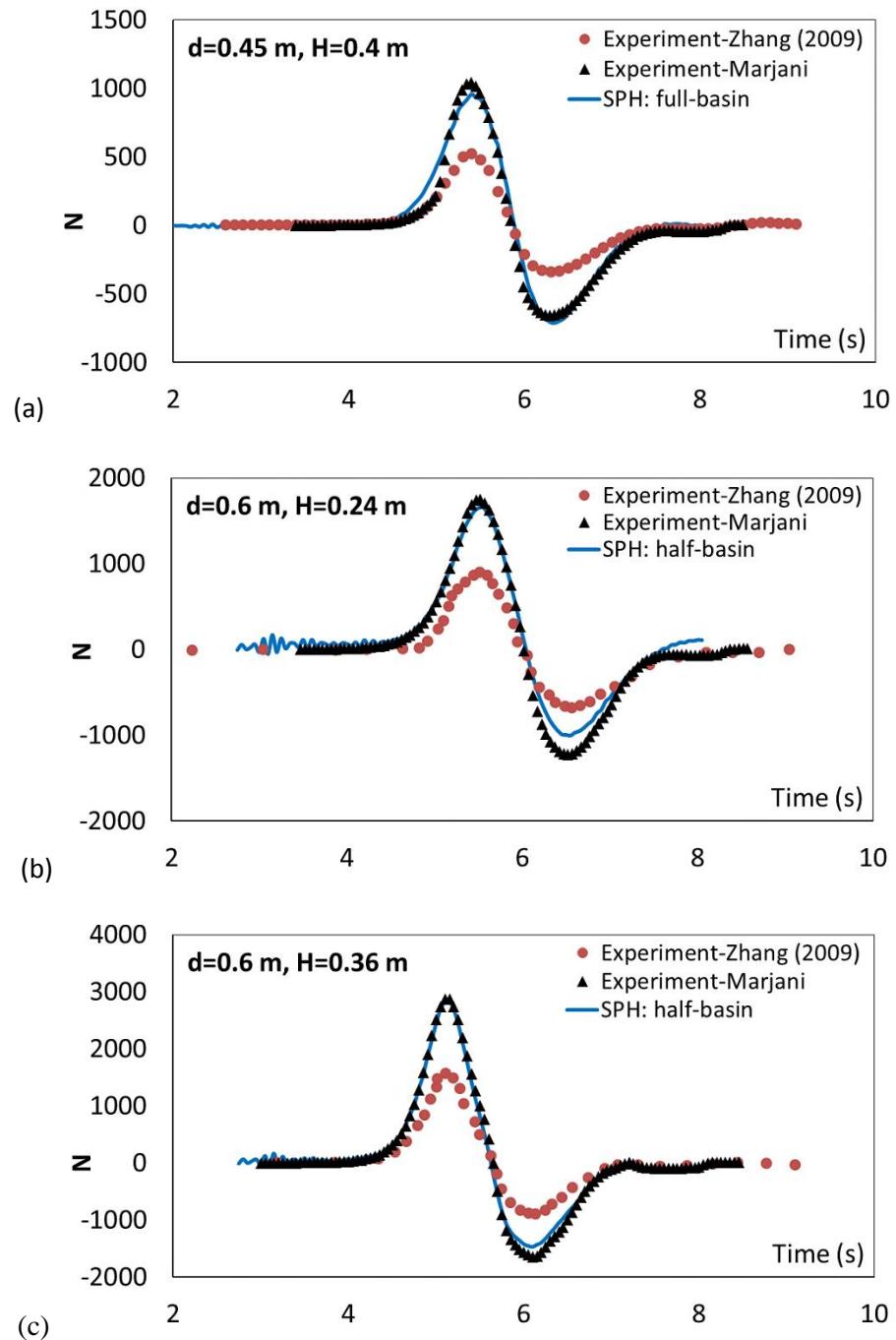
For all cases with different water depth of 0.75 m, 0.6 m and 0.45 m, the total force on the half cylinder in the direction normal to the wave propagating (F_y) is computed when the fluid is at the rest which should be approximately equal to the hydrostatic force. Obviously, the total force on the full cylinder in the same direction is expected to be zero always. Table 4-3 reports the SPH computation and hand-calculation hydrostatic force. This may be considered as a double-checking the correctness of the Fortran subroutine that integrates the force around an object.

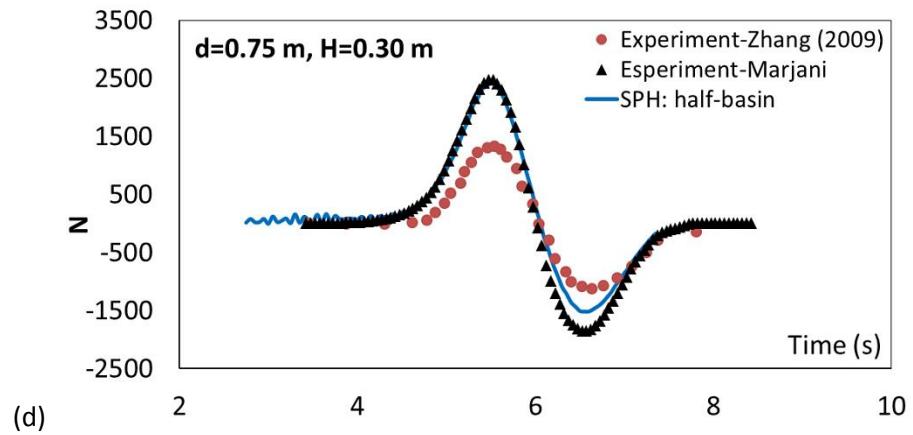
Table 4-3- The horizontal force on the half cylinder
in the normal direction to the wave direction

| Water depth d (m) | F_y (N) | |
|------------------------|-----------|--------|
| | SPH | Theory |
| 0.45 | 1198 | 1192 |
| 0.60 | 2134 | 2120 |
| 0.75 | 3280 | 3310 |

Finally, the strain gauge raw data from NEES website are converted into force unit directly. The raw data of four different tests are converted here and compared with SPH simulation and Zhang's converted data, see Figure 4-28. Figure 4-28-a presents the full-basin model while others correspond to half-basin model. Appendix C describes the conversion process in detail.

Figure 4-28- Total horizontal force on the single cylinder, SPH output (solid line), Experimental data converted by Zhang (dots), Experimental data converted by Marjani (triangles). (a) when $d=0.45$ m, $H=0.40$, (b) when $d=0.60$ m, $H=0.24$, (c) when $d=0.60$ m, $H=0.36$, and (d) when $d=0.75$ m, $H=0.30$





4.5 Combination of Methods

It is very helpful to combine all above techniques to speed up SPH more. For instance, the numerical simulation in section 4.3.1 has 1.33 million particles and it takes about 34 hours to run on 64 processors. In that test, only variable- h plus variable cells and dividing wall are combined together. Adding the mirror boundary condition can reduce the number of particle by half which leads to speedup of more than 2. A test similar to that of section 4.3.1 is simulated by combination of predefined variable cell, dividing wall and mirror boundary condition. Thus the particle size varies from $dp=0.015$ m close to the cylinder to $dp= 0.03$ m far from it. The width of half basin is 1.5 m and total number of particles is 670K. This model took about 15 hours to get done on 48 processors of RENCI cluster. Employing less number of cores, half-basin model speed the running up to 2.3 which is 15% higher than 2. Also, consider that the number of cores reduced from 64 to 48. Figure 4-29 illustrates the initial particle configuration. Colors represent the mass of particles. Figure 4-30 plots the total horizontal force by the combined model with the full-size and half-size model as presented in section 4.3.1.

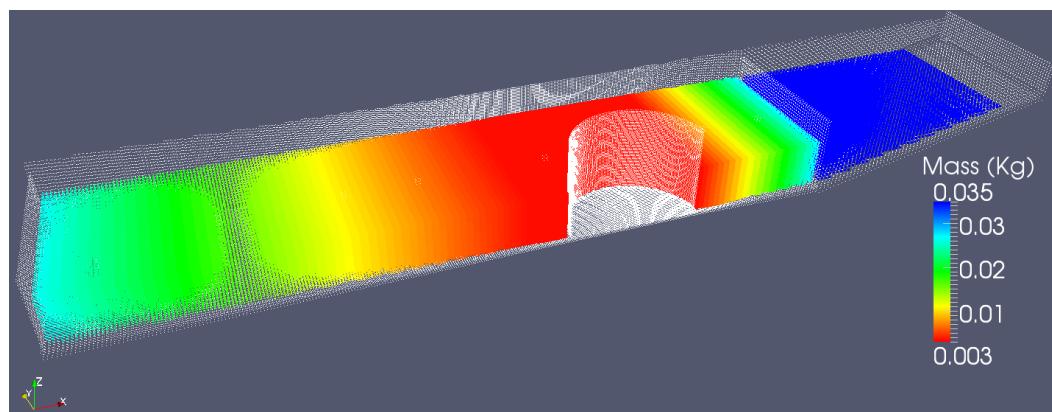


Figure 4-29- Initial particle configuration of combined model. Warm colors represent light particles and cold colors represent the coarse ones.

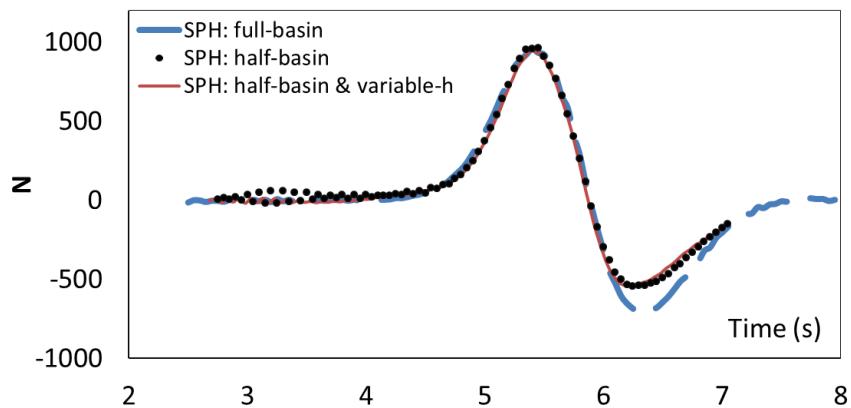


Figure 4-30- Total horizontal force on the cylinder.
Dashed line: full-basin; dots: half-basin and solid line: half-basin & variable-h

Chapter 5

5 Concluding Remarks and Future Work

5.1 Summary

In this study, the state-of-the-art of classic Smoothed Particle Hydrodynamics methodology has been presented and modified to investigate the free-surface flows interacting with rigid structures. Severe water-structure interactions, the evaluation of global and local force and momentums play leading roles in coastal engineering and maritime. An invulnerable and safe design of marine structures and their safety in operation relies on the numerical estimation of all types of loads on them. The fluid-structure interaction causes wave force, diffracted and reflected waves. The pressure field around the cylinder changes by the process of diffraction, thus the total force and momentum on the body will be influenced.

Combination of Lagrangian approach with a meshless character motivates us to choose such a numerical model since no special treatment for free-surface and complex geometry is needed. Therefore, fully non-linear flows under extreme sea condition including violent wave breaking, fluid fragmentation, slamming, flooding and sloshing are very easy to be modeled.

Although this method was firstly used in astrophysics context about forty years ago, it has been applied in fluid mechanics only in two decades. SPH attracts an increasing attention of researchers and numerical modelers recently. Innovative numerical schemes and different techniques are introduced continuously to improve its robustness, simplicity and accuracy. Implementing new physics, formula or algorithm is very simple in SPH. The first part of this study is dedicated to describing the basic concept of SPH and its formulation. Several recent techniques and achievements related to present research activity and the advantages and disadvantages of them are also reviewed.

SPHysics follows SPH formulation to simulate free-surface flows. Serial and parallel versions of SPHysics are used here to study and validate the accuracy and efficiency of different techniques through simple two and three dimensional tests. Water surface elevation, particle velocity, fluid pressure and force components on an object are evaluated by experimental data or available numerical simulations. The major attention of this thesis is to assess the capability of SPH in computing the force on a structure.

Initially, SPH was tested by several scenarios of dam breaking. The first tests is involving the collapse of a dam on an almost dry bed and the successive impact with a tall square shape column, comparing the results with the experimental data given by Raad for horizontal force on the structure and velocity in the close proximity of it. SPH, using dynamic particles as solid walls, is acceptably accurate in both terms when particle size is chosen 0.0225 m; however, it is much more precise when smaller particle, 0.01 m, is applied. Moreover, the test is also re-simulated by Monaghan and Kos (1999) repulsive boundary condition and the particle size of 0.0225 m. Comparison of results proves that dynamic particle technique

estimates the particle velocity much more accurately than repulsive force does and the general trend of force by dynamic particles is more compatible with laboratory observation.

Another dam break test on a wet bottom and in a relatively large tank shows that SPH does not accurately estimate the impact force on a circular cylinder when 100K particles initially spaced 0.03 m are employed. Although smaller particle can produce more acceptable result, all SPH calculations should be done for about 1.3M particles at each time step if dp is 0.01 m. SPH predicts the water surface and fluid pressure of a dam break on a small box precisely as the particle size is 0.018 m. Present simulation computes the pressure around the box much more perfect than the weakly compressible SPH model of Lee et al (2010) does. This tremendous discrepancy is still unclear.

In the next step, sinusoidal wave generation and its propagation are evaluated by SPH in 2D using $dp=0.015$ m. Water surface elevation and the velocity vector at offshore and two locations at the onshore are compared with the experiment. Similar to dam break test, dynamic boundary particles produce much more accurate regular wave in comparison with repulsive force method. Probably, this privilege is related to the fact that in contrast to repulsive force tool, dynamic boundary particles are treated as fixed fluid particle. In addition, the numerical wave runup on the beach agrees the Hunt's formula and laboratory data. Defining $dp=0.005$ m, SPH is indeed accurate in computing the wave force on a horizontal pipe under two different scenarios, specifically for the case in which the pipe is partially submerged.

SPH presents the force on streamlined objects such as a rectangular flat plate successfully. Astonishingly, this model can find the horizontal force on a flat plate under the wave

propagation correctly even with dp is as large as 0.03 m. Since the size of domain is not very large, the total number of particles, 280k, is not enormous. In some cases, even the choice of $dp=0.03$ m produces huge number of particles. Test 5 in chapter 3 presents the impact of solitary wave on a circular cylinder. About 900k particles with the size of 0.03 m fill the numerical basin which is smaller than actual test. Although SPH generates the water surface, particle velocity and the horizontal force precisely, calculations take about 35 hours of 64 processors.

5.2 Conclusions

The benchmarks used in this work demonstrate that existing SPH formulation is accurate enough to simulate fluid mechanics phenomena. One of the well-known drawbacks of SPH is its highly expensive computational cost. Due to its fully explicit character, WSPH is considered as a very slow tool. According to this disadvantage, 3D-SPH can be used only for small scale test rather than large or practical applications. Not only the computational time, but also the number of processors and required memory are not very important. Not all machines have 48-64 cores with enough amount of physical memory. Therefore, some techniques are introduced in chapter 4 that can reduce the number of particles or the computational cost.

About 27 million particles are needed to simulate the test 5 in chapter 3 if particle size is set to 0.01 m everywhere. Even choosing $dp=0.015$ m creates about 7 million particles. It is discussed that the predefined and smoothly variable kernel length is one of the tools used to

define fine particles at the critical regions and coarse particles elsewhere. Sections 4.2.2 and 4.2.3 describes the disadvantages of applying fixed underlying cells for variable mass particles specifically in 3D. Nearest Neighbor Particle Searching process loses its efficiency when variable kernel length is applied with fixed size cell. Moreover, enormous number of particles will be transferring between cores unnecessarily. Thanks to parallel processing feature, variable underlying cells is allocated to each processor as a function of average particle size. It is shown that this technique is able to increase the SPH efficiency up to 3 in three dimensions.

The other technique that has not been implemented previously in SPHysics is a mirror boundary. All benchmarks in this PhD thesis have symmetric geometries. Some of applications in coastal engineering have symmetric properties, while we have not been able to use this property to reduce the computational cost. Actually, all 3D tests in the present study are geometrically symmetric. Mirror type boundary is introduced for the first time and is used in two 3D tests, dam breaking impact on the square column and solitary wave impact on circular cylinder. It is worthy to note that linked-list algorithm reduces the number of particle interactions in a processor to $N_p \log N_p$. That means given the same number of processors, reducing the processors' load by half will give a speedup more than 2. Besides, mirror boundary leads to less particle transferring activities which are considerably time consuming task. Simulating the half basin can decrease the number of demanded cores by half, given the same running time. Sometime, limitation of hardware becomes a serious concern rather than running time. Note that periodic boundary condition cannot generate the

full basin by multiplying the half basin. Periodic boundary condition only duplicates the basin infinitely in a specific direction.

Due to a suspicious error in reported experimental force on the singular cylinder by Zhang (2009), it was tried to convert the raw data to the force unit in the current study. Four different scenarios associated to various water depth and wave height are converted. Mirror boundary condition gives us the opportunity of simulating the deeper cases in a shorter time. The computed force in all cases is in very good agreement with the converted data.

Combination of these two techniques definitely increases the speedup. The half-basin of solitary wave test is modeled when particle size varies from 0.03 m to 0.015 m. Presented results in chapter 4 state that combining mirror boundary condition with variable kernel function can give a speedup of 2.3 even by reducing the number of cores from 64 to 48.

It should not be forgotten that the idea of dividing wall decreases the total number of particles in the case of solitary wave noticeably. As an example, the solitary wave propagation after the impact time and the wave process on the beach is not important in test 5. As discussed, the freely-moving wall let us to use much larger particles on the downstream side of the wall.

A new load balancing algorithm is proposed in appendix A. This approach follows the existing algorithm however it is much faster and smarter when a model includes variable mass particle. It assigns higher priority to the borders corresponding to greater load difference in the whole domain.

5.3 Future work

Although the accuracy of half-basin model is acceptable, a slight difference is captured between the minimum force of solitary wave, resulted by half basin and full size model. The reason of this discrepancy still remains an open question which should be explored in future. Under the condition of severe flow velocity in the direction normal to the mirror boundary, the entering of fluid particle in imaginary side is a possible consideration. Further modifications can improve the accuracy of this type of boundary.

Another issue is the accuracy of smoothly variable kernel function. The particle size changes from large to small gradually and vice versa in the whole domain to reduce the error of ignoring the gradient of h . The variable- h method does not predict the water surface, fluid velocity and impact force as perfectly as constant- h model does. For much further accuracy, it is highly recommended to consider the additional terms related to gradient of h to the existing SPH formulation.

Since most of wave energy convertors, such as PowerBuoy, Pelamis, Oyster, and etc., have symmetric geometry, mirror-type boundary condition will be very helpful in speeding up the simulation of their operation. Some primary simulations of flap-type and buoy-type convertors (similar to Oyster and PowerBuoy) have been done in this study that are not presented in this document. Studying the properties of various types of devices, testing different shape or geometry of each individual one or comparing the efficiency of devices is considered as a real application of SPH in coastal engineering.

References

- Antuono, M., Colagrossi, A., Marrone, S., Molteni, D. (2010), Free-surface flows solved by means of SPH schemes with numerical diffusive terms, *Computer Physics Communications*, Vol. **181**, pp. 532–549
- Antuono, M., Colagrossi, A., Marrone, S., and Lugni, C. (2011), Propagation of gravity waves through an SPH scheme with numerical diffusive terms, *Comp. Phys. Comm.*, Vol. **182**, pp. 866–877
- Arnason, H. (2005), *Interactions between an incident bore and a free-standing coastal structure*, PhD Thesis, University of Washington, Seattle
- Barter, E.F., McNown, J.S., Stair, L.D. (1958), Wave Forces on Submerged Structures, Journal of the Hydraulics Division, *Proceedings of the ASCE*, Paper 1833
- Batchelor, G.K. (1974), *Introduction to fluid dynamics*, Cambridge University Press., U.K.
- Bate, M.R., Bonnell, I.A. and Bromm, V. (2003), The formation of a star cluster predicting the properties of stars and Brown Dwarfs, *Mon. Not. R. Astron. Soc.*, Vol. **399**, pp. 577–99
- Bate, M.R., Bonnell, I.A. and Price, N.M. (1995), Modeling accretion in protobinary systems, *Mon. Not. R. Astron. Soc.*, Vol. **277**, pp. 362–76
- Behara, S., Mittal, S. (2011), Transition of the boundary layer on a circular cylinder in the presence of a trip, *Journal of Fluid and Structures*, Vol. **27**, pp. 702-715
- Belytschko, T., Krongauz, Y., Dolbow, J., and Gerlach, C. (1998), On the completeness of meshfree particle methods, *Journal of Numer. Meth. Engng*, Vol. **43.5**, pp. 785–819
- Benz, W. and Asphaug, E. (1994), Impact simulations with fracture. I. Methods and tests, *International Journal of solar system studies*, Vol. **107**, pp. 98-116
- Benz, W. and Asphaug, E. (1995), Simulations of brittle solids using smooth particle hydrodynamics, *Journal of Computer Physics Communications*, Vol. **87**, pp. 253-265
- Benz, W., Bowers, R., Cameron, A., and Press, W. (1990), Dynamic mass exchange in doubly degenerate binaries. i - 0.9 and 1.2 solar mass stars, *ApJ*, Vol. **348**, pp. 647
- Benz, W., Slattery, W.L., and Cameron, A.G.W. (1986), The origin of the moon and the single impact hypothesis, *Icarus*, Vol. **66**, pp. 515–35

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

Bonet, J., and Lok, T.S.L. (1999), Variational and momentum preservation aspects of Smoothed Particle Hydrodynamics formulations, *Computer Methods in Applied Mechanics and Engineering*, Vol. **180**, pp. 97-115

Cha, S.H., and Whitworth, A.P. (2003), Implementations and tests of Godunov-type particle hydrodynamics, *Monthly Notice of the Royal Astronomical Society*, Vol. **340** (1), pp. 73-90

Chen, J.K., Beraun, J.E. (2000), A generalized smoothed particle hydrodynamics method for nonlinear dynamic problems, *Comput. Meth. Appl. Mech. Engrg*, Vol. **190**, pp. 225–239

Christensen, E.D. (2006), Large eddy simulation of spilling and plunging breakers, *Coastal Engineering*, Vol. **53**, pp.463-485

Cleary, P.W. (1998), Modelling confined multi-material heat and mass flows using sph, *Applied Mathematical Modelling*, Vol. **22** (12), pp. 981-993

Colagrossi, A. (2005), *A meshless lagrangian method for free-surface and interface flows with fragmentation*, PhD thesis, Universit`a di Roma La Sapienza

Colagrossi, A. and Landrini, M. (2003), Numerical simulation of interfacial flows by smoothed particle hydrodynamics, *Journal of Computational Physics*, Vol. **191**, pp. 448-475

Coutanceau, M. and Bouard, R. (1977), Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. Part 1. Steady flow, *J. Fluid Mechanics*, Vol.**79**, part 2: pp. 231-256

Coveney, P.V. and Novik, K.E. (1997), Computer simulations of domain growth and phase separation in two-dimensional binary immiscible fluids using dissipative particle dynamics, *Physical review*, Vol. **5**, pp. 5134-5141

Crespo, A.J.C. (2008), *Application of the Smoothed Particle Hydrodynamics model SPHysics to free-surface hydrodynamics*, PhD thesis, Department of Applied Physics, University of Vigo, Spain

Crespo, A.J.C., Gómez-Gesteira, M. and Dalrymple, R.A. (2007), 3D SPH simulation of large waves mitigation with a dike, *Journal of Hydraulic Research*, Vol. **45**, pp. 631-642

Cummins, S.J. and Rudman, M. (1999), An SPH projection method, *Journal Computational Physics*, Vol. **152**, pp. 584-607

Dalrymple, R.A. and Knio, O. (2001), SPH Modelling of Water Waves, *Proc. Coastal Dynamics*, ASCE, Lund, Sweden, pp. 779-787

Dalrymple, R.A. and Herault A. (2009), Levee Breaching with GPU-SPHysics Code, *4th SPHERIC Workshop*, Nantes

Dalrymple, R.A. and Rogers, B.D. (2006), Numerical modeling of water waves with the SPH method, *Journal of Coastal Engineering*, Vol. **53**, pp. 141-147

Dalrymple, R. A., Knio, O., Cox, D. T., Gesteira, M., and Zou, S. (2002), Using a Lagrangian particle method for deck overtopping., *Proc., Waves 2001, ASCE*, Reston, Va., pp. 1082–1091

Dean, R.G., Dalrymple, R.A. (1991), *Water Wave Mechanics for Engineers & Scientists*, World Scientific Pub Co Inc

De Leffe, M., Le Touze, D., and Alessandrini, B. (2011), A modified no-slip condition in weakly-compressible SPH, *6th International SPHERIC workshop*, Germany

De Padova, D., Dalrymple, R.A., Mossa, M., and Petrillo, A.F. (2009), SPH Simulations of Regular and Irregular Waves and their Comparison with Experimental Data, submitted to *J. of Fluid Dynamics*

De Serio, F., and Mossa, M. (2006), Experimental study on the hydrodynamics of regular breaking waves, *J. of Coastal Engineering*, vol. **53**, pp. 99-113

Dilts, G.A. (1999), Moving-Least-Squares-Particle Hydrodynamics- I. Consistency and Stability, *Journal of Numer. Meth. Engng*, Vol. **44**, pp. 1115–1155

Ellero, M., Serrano, M., and Espan˜ol, P. (2007), Incompressible smoothed particle hydrodynamics, *Journal of Comp. Phys.*, Vol. **226**, pp. 1731–1752

Español, P. (1997), Dissipative particle dynamics with energy conservation, *Europhysics Letters*, Vol. **40** (6), pp. 631-636

Evrard, A.E. (1988), Beyond N-body: 3D-cosmological gas-dynamics, *MNRAS*, Vol. **235**, pp. 911-934

Feldman, J. and Bonet, J. (2007), Dynamic refinement and boundary contact forces in SPH with applications in fluid flow problems, *J. Numer. Meth. Engng.*, Vol. **72**, pp. 295–324

Ferrari, A., Dumbser, M., Toro, E.F., Armanini, A. (2009), A new 3D parallel SPH scheme for free surface flows, *Journal of computers and Fluids*, Vol. **38**, pp. 1203-1217

Fontaine, E. (2000), On the use of smoothed particle hydrodynamics to model extreme waves and their interaction with structures., *Proc. Rogue Waves 2000*

Fulk, D.A. (1994), *A Numerical Analysis of Smoothed Particle Hydrodynamics*, PhD thesis, Air Force Institute of Technology

Fulk, D.A., Quinn, D.W. (1996), An Analysis of 1-D Smoothed Particle Hydrodynamics Kernels, *Journal of Computational Physics*, Vol. **126**, pp. 165-180

Gingold, R.A. and Monaghan J.J (1977), Smoothed Particle Hydrodynamics, *Monthly Notice of the Royal Astronomical Society*, Vol. **235**, pp. 911-934

Gomez-Gesteira, M. and Dalrymple, R.A. (2004), Using a 3D SPH method for wave impact on a tall structure, *Journal of Waterway, Port, Coastal and Ocean Engineering*, Vol. **130** (2), pp. 63-69

Gómez-Gesteira, M., Cerqueiro, D., Crespo, A.J.C. and Dalrymple, R.A. (2005), Green water overtopping analyzed with an SPH model, *Ocean Engineering*, Vol. **32** (2), pp. 223-238

Gómez-Gesteira, M., Crespo, A.J.C., Rogers, B.D., Dalrymple, R.A., Dominguez, J.M., Barreiro, A. (2012), SPHysics – development of a free-structure fluid solver – Part 2: Efficiency and test cases, *J. of Computers & Geosciences*, Vol. **48**, pp. 300-307

Gómez-Gesteira, M., Rogers, B.D., Dalrymple, R.A., Crespo, A.J.C. and Narayanaswamy, M. (2009), User guide for the SPHysics code v2.0, <http://wiki.manchester.ac.uk/sphysics>

Gómez-Gesteira, M., Rogers, B.D., Dalrymple, R.A. and Crespo, A.J.C. (2010), State-of-the-art of classical SPH for free-surface flows, *Journal of Hydraulic Research*, Vol. **48**, pp. 6-27

Gonzalez, L.M., Sánchez, J.M., Maciá, F., Souto-Iglesias, A. (2009), Analysis of WCSPH laminar viscosity models, *4th International SPHERIC workshop*, Nantes, France

Goring, D.G. (1979), *Tsunami-The Propagation of Long Waves onto a Shelf*, PhD thesis, California Institute of Technology, California, USA

Gotoh, H. and Sakai, T. (1999), Lagrangian simulation of breaking waves using particle method, *Coastal Engineering Journal*, Vol. **41**, pp. 303-326

Gotoh, H., Shibihara, T. and Sakai, T. (2001), Sub-particle-scale model for the MPS method-Lagrangian flow model for hydraulic engineering, *Journal of Computational Fluid Dynamics*, Vol. **9** (4), pp. 339-347

Haque, A. and Dilts, G.A. (2007), Three-dimensional boundary detection for particle methods, *Journal of Comp. Phys.*, Vol. **226**, pp. 1710–1730

Héroult, A., Bilotta, G., and Dalrymple, R.A. (2010), SPH on GPU with CUDA, *Journal of Hydraulic Research*, Vol. **48** (Extra Issue), pp. 74-79

Hernquist, L. and N. Katz (1989), TreeSPH: A Unification of SPH with the Hierarchical Tree Method, *Astrophysical Journal Supplement*, Vol. **70**, pp. 419-446

Hildenbrand, K. and Hirota, A. (2010), A primer on Wave Energy Devices, A report by Oregon Sea Grant, <http://seagrant.oregonstate.edu/sgpubs/onlinepubs/g10003.pdf>

Ho, S.L., Yang, S., Wong, H.C., Lo, E.W.C. and Ni, G. (2005), Refinement computations of electromagnetic fields using FE and meshless methods, *IEEE Transactions on Magnetics*, Vol. **41**, pp. 1456-1459

Hoogerbrugge, P.J. and Koelman, J.M.V.A. (1992), Simulating microscopic hydrodynamic phenomena with Dissipative Particle Dynamics, *Europhysics Letters*, Vol. **19**, pp. 155-160

Hunt, I.A. (1959), Design of Seawalls and Breakwater, *Journal of Waterways and Harbor Division*, Vol. **85**, No. WW3, pp. 123-152

Idelsohn, S.R., Oñate, E., Calvo, N. and Del Pin, F. (2003), The meshless finite element method, *International Journal for Numerical Methods in Engineering*, Vol. **58** (6), pp. 893-912

Issa, R., Violeau, D., 2006, SPHERIC: Test case 2, 3D dambreaking,
https://wiki.manchester.ac.uk/spheric/images/SPHERIC_Test2_v1p1.pdf

Johnson, G.R., Stryk, R.A., and R., B.S. (1996), SPH for high velocity impact computations, *Comput. Methods Appl. Mech. Engineering*, Vol. **139**, pp. 347-373

Kim, N.H., Park, M.S., and Yang, S.B. (2007), Wave Force Analysis of Vertical Circular Cylinder by Boundary Element Method, *Journal of civil Engineering*, Vol. **11**, No. 1, pp. 31-35

Khayyer, A. and Gotoh, H. (2009), Modified Moving Particle Semi-implicit methods for the prediction of 2D wave impact pressure, *Coastal Engineering Journal*, Vol. **59**, pp. 419-440

Khayyer, A., Gotoh, H., and Shao, S.D. (2008), Corrected incompressible SPH method for accurate water-surface tracking in breaking waves, *Journal of Coastal Engineering*, Vol **55** (3), pp. 236-250

Kleefsman, K.M.T., Fekken, G., Veldman, A.E.P., Iwanowski, B., and Buchner, B. (2005), A Volume-of-Fluid based simulation method for wave impact problems, *J. Comput. Phys.*, Vol. **206**, pp. 363–393

Koshizuka, S. and Oka, Y. (1996), Moving particle semi-implicit method for fragmentation of incompressible fluid, *Nuclear Science and Engineering*, Vol. **123**, pp. 421-434

Koshizuka, S., Chikazawa, Y. and Oka, Y. (2001), A particle method for elastic and visco-plastic structures and fluid-structure interactions, *Computational Mechanics*, Vol. **27**, pp. 97-106

Kundu, P.K., Cohen, I.M., and Dowling, D.R. (2011), *Fluid Mechanics*, Academic Press, Fifth edition

Landrini, M., Colagrossi, A., Greco, M., and Tulin, M.P. (2007), Gridless simulations of splashing proces and nearshore bore propagation, *Journal of Fluid Mech*, Vol. **591**, pp. 183–213

Lee, E-S., Violeau, D., Issa, R., and Polix, S. (2010), Application of weakly compressible and truly incompressible SPH to 3-D Water collapse in waterworks, *Journal of Hydraulics Research*, Vol. **48**, Extra Issu, pp. 50-60

Liu, G.R. and Liu, M.B. (2007), *Smoothed Particle Hydrodynamics a meshfree particle method*, World Scientific

Liu M.B., Liu, G.R. (2010), Smoothed Particle Hydrodynamics (SPH): an Overview and Recent Developments, *Arch. Comput. Methods Eng.*, Vol. **17**, pp. 25-76

Liu, M.B., Liu, G.R., Lam, K.Y., Zong, Z. (2003), Smoothed particle hydrodynamics for numerical simulation of underwater explosion, *Journal of Computational Mechanics*, Vol. **30** (2), pp. 106-118

Liu, M.B., Liu, G.R., Zong, Z. (2008), An Overview on Smoothed Particle Hydrodynamics, *International Journal of Computational Methods*, Vol. **5**, No. 1, pp. 135-188

Liu, W., Li, S., and Belytschko, T. (1997), Moving least square Kernel Galerkin method (I) methodology and convergence, *Comput. Methods Appl. Mech. Engineering*, Vol. **143**, pp. 113-154

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

Lucy, L.B. (1977), A numerical approach to the testing of fusion process, *Astronomical Journal*, Vol. **88**, pp. 1013-1024

Mahmood, O., Kassiotis, C., Violeau, D., Ferrand, M., Denis, C. (2011), Effect of wall boundary treatment in SPH for modelling turbulent flows with inlet/outlet, *6th International SPHERIC workshop*, Germany

Marongiu, J.C., Leboeuf, F. and Parkinson, E. (2008), Riemann Solvers and efficient boundary treatments: an hybrid SPH-finite volume numerical method, *Proceeding of 3rd International SPHERIC Workshop*; Switzerland, pp. 101-108

Marrone, S. (2011), *Enhanced SPH modeling of free-surface flows with large deformations*, PhD Thesis, Theoretical and Applied Mechanics, University of Rome

Marrone, S., Colagrossi, A., Le Touzé, D., and Graziani, G. (2010), Fast free-surface detection and level-set function definition in SPH solvers, *Journal of Comp. Phys.*, Vol. **229**, pp. 3652–3663

Marrone, S., Antuono, M., Colagrossi, A., Colicchio, G., Le Touze, D., and Graziani, G., (2011), δ -SPH model for simulating violent impact flows, *Journal of Comput. Mehodts Appl. Mech Engrg*, Vol. **200**, No. 13, pp. 1526-1542

Marrone, S., Antuono, M., Colagrossi, A., Colicchio G., and Graziani, G. (2012), SPH modeling of viscous flows around cylinders from $Re=10$ to $Re=1000$, *7th International SPHERIC Workshop*, Prato, Italy, pp. 28-30

Marsh, A.P., Oger, G., Le Touze, D., Guibert, D. (2011), Validation of a conservative variable-resolution SPH scheme including $\text{grad}(h)$ terms, *6th ERCOFTAC SPHERIC Workshop*, Hamburg, Germany, pp. 86–93

Mo, W. (2010), *Numerical Investigation of Solitary Wave Interaction with Group of Cylinders*, PhD thesis, Cornell University

Molteni, D. and Colagrossi, A. (2009), A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH, *Journal of Comp. Phys. Comm.*, Vol. **180**, pp. 861–872

Monaghan, J. J. (1985), Particle methods for hydrodynamics, *Comput. Phys. Rep.*, Vol. **3**, pp. 71–124

Monaghan, J.J. (1988), An introduction to SPH, *Computer Physics Communication*, Vol. **48**, pp. 89-96

Monaghan, J.J. (1989), On the Problem of penetration in particle methods, *Journal Computational Physics*, Vol. **82**, pp. 1-15

Monaghan, J.J. (1992), Smoothed Particle Hydrodynamics, *Annual Review of Astronomy and Astrophysics*, Vol. **30**, pp. 534-574

Monaghan, J.J. (1994), Simulating free surface flows with SPH, *Journal of Computational Physics*, Vol. **110**, pp. 290-311

Monaghan, J.J. (1996), Gravity currents and solitary wave, *Physica D: Nonlinear Phenomena*, Vol. **98**, pp. 523-533

Monaghan, J.J. (2000), SPH without Tensile Instability, *Journal Computational Physics*, Vol. **159**, pp. 290-311

Monaghan, J.J. (2005a), Smoothed Particle Hydrodynamics, *Reports on Progress in Physics*, Vol. **68**, pp. 1703-1759

Monaghan, J.J. (2005b), Smoothed Particle Hydrodynamics simulations of shear flow, *Monthly Notices of the Royal Astronomical Society*, Vol. **365**, pp. 199-213

Monaghan, J.J., Cas, R.F., Kos, A., and Hallworth, M. (1999), Gravity currents descending a ramp in a stratified tank, *Journal of Fluid Mechanics*, Vol. **379**, pp. 39-70

Monaghan, J.J. and Gingold, R.A. (1983), Shock Simulation by the Particle Method SPH, *J. Comp. Phys.*, Vol. **52**, pp. 374-389

- Monaghan, J.J. and Kajtar, J.B. (2009), SPH particle boundary forces for arbitrary boundaries, *Comp. Phys. Comm.*, Vol. **180**, pp. 1811-1820
- Monaghan, J.J. and Kocharyan, A. (1995), SPH simulation of multi-phase flow, *Computer Physics Communications*, Vol. **87**, pp. 225-235
- Monaghan, J.J. and Kos, A. (1999), Solitary waves on a Cretan beach, *Journal of Waterway, Port, Coastal and Ocean Engineering*, Vol. **125**, pp. 145-154
- Monaghan, J.J. and Kos, A. (2000), Scott Russell's wave generator, *Physics of Fluids*, Vol. **12 (3)**, pp. 622-630
- Monaghan, J.J., Kos, A. and Issa, N. (2003), Fluid motion generated by impact, *Journal of the Waterway Port, Coastal and Ocean Division*, Vol. **129 (6)**, pp. 250-259
- Monaghan, J.J. and Lattanzio, J.C. (1985), A refined method for astrophysical problems, *Astron. Astrophys.*, Vol. **149**, pp. 135-143
- Morris, J.P. (1996), *Analysis of smoothed particle hydrodynamics with applications*, PhD Thesis, Monash University, Australia
- Morris, J.P., Fox, P.J., and Zhu Y. (1997), Modeling Low Reynolds Number Incompressible Flows Using SPH, *Journal of Computational Physics*, Vol. **136**, pp. 214-226
- Moussa, B.B., Lanson, N., Vila, J.P. (1999), Convergence of meshless methods for conservation laws: applications to Euler equations, *Int. Ser. Numer. Math.*, Vol. **129**, pp. 31–40
- Nelson R.P., Papaloizou, J.C.B. (1994), Variable Smoothing Lengths And Energy Conservation In Smoothed Particle Hydrodynamics, *Monthly Notices of the Royal Astronomical Society*, Vol. **270**, No. 1, P. 1
- Nouri, Y., Nistor, I., and Palermo, D. (2010), Experimental Investigation of Tsunami Impact on Free Standing Structures, *Coastal Engineering Journal*, Vol. **52**, No. 1, Pp. 43–70
- Oger, G., Doring, M., Alessandrini, B. and Ferrant, P. (2006), Two Dimensional SPH simulations of wedge water entries, *Journal of Computational Physics*, Vol. **213**, pp. 803-822
- Omidvar, P. (2010), *Wave Loading on Bodies in the Free Surface Using Smoothed Particle Hydrodynamics (SPH)*, PhD thesis, Department of Mechanical, Aerospace and Civil Engineering, University of Manchester, UK
- Omidvar, P., Stansby, P.K., and Rogers, B.D. (2011), Wave body interaction in 2D using smoothed particle hydrodynamics (SPH) with variable particle mass, *International Journal for Numerical Methods in Fluids*, Published online in Wiley Online Library

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

Prasad, S. (1994), *Wave Impact Forces on a Horizontal Cylinder*, PhD thesis, Department of Civil Engineering, University of British Columbia, Canada

Raad P., Mitigation of local tsunami effects, <http://lyle.smu.edu/waves/mitigation.html>

Randles, P.W., Libersky, L.D. (1996), Smoothed particle hydrodynamics: Some recent improvements and applications, *Comput. Meth. Appl. Mech. Engrg.*, Vol. **138**, pp.375–408

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

Rogers, B.D., Dalrymple, R.A., and Stansby, P.K. (2008), SPH modeling of floating bodies in the surf zone, *Proceeding of 31st International Conference on Coastal Engineering (ICCE)*; Germany, pp. 204-215

Rooney, E.A., Wei, W.-Y., Irish, J., Weiss, R., Dalrymple, R.A., Herault, A., and Bilotta G. (2011), Testing accuracy and convergence of GPUSPH for free-surface flows, *6th International SPHERIC workshop*, Germany

Safdari Shadloo, M., Zainali, A., Sadek, S.H., and Yildiz, M. (2011), Improved Incompressible Smoothed Particle Hydrodynamics method for simulating flow around bluff bodies, *Comput. Methods Appl. Mech. Engrg.*, Vol. **200**, pp. 1008–1020

Sims, J.S. and Martys, N.S. (2004), Simulation of sheared suspensions with a parallel implementation of QDPD, *Journal of Research of the National Institute of Standards and Technology*, Vol. **109** (2), pp. 267-277

Shao, S. and Lo, E.Y.M. (2003), Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface, *Advances in Water Resources*, Vol. **26**, pp. 787-800

Smagorinsky, J. (1963), General circulation experiments with the primitive equations: I. The basic Experiment, *Monthly Weather Review*, Vol. **91**, pp. 99-164

Sorensen, R. M. (2006), *Basic Coastal Engineering*, Springer Science + Business Media

Souto-Iglesias, A., Pérez Rojas, L., and Zamora Rodríguez, R. (2004), Simulation of anti-roll tanks and sloshing type problems with smoothed particle hydrodynamics, *Ocean Engineering*, Vol. **31**, pp. 1169–1192

Swegle, J.W. and Attaway, S. (1995), On the feasibility of using Smoothed Particle Hydrodynamics for underwater explosion calculation, *Comput. Mech.*, Vol. **17**, pp. 151-168

Takeda, H., Miyama, S.M., and Sekiya, M. (1994), Numerical Simulation of Viscous Flow by Smoothed Particle Hydrodynamics, *Progress of Theoretical Physics*, Vol. **92**, No. 5, pp. 939-960

Trivellato, F., Bertolazzi, E., and Colagrossi, A. (2006), Flow solvers for liquid–liquid impacts, Vorticity and Turbulence Effects in Fluid Structure Interactions, Southampton: WIT Press. Chap. 11, pp. 261–287, ISBN: 1-84564-052-7

Vila, J.P. (1999), On particle weighted methods and Smoothed Particle Hydrodynamics, *Mathematical Models and Methods in Applied Sciences*, Vol. **9**, pp. 161-209

Vining, J.G. (2007), *Ocean wave energy converters: Overview, legal and economic aspects and direct-driver power take-off*, Master thesis, University of Wisconsin-Madison

Wendland, H. (1995), Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, *Advances in computational Mathematics*, Vol. **4** (1), pp. 389- 396

Xuan, L., Zeng, Z., Shanker, B. and Udpal, L. (2004), Element-free Galerkin method for static and quasi-static electromagnetic field computation, *IEEE Transactions on Magnetics*, Vol. **40**, pp. 12-20

Zhang, W. (2009), *An Experimental Study and a Three-Dimensional Numerical Wave Basin Model of Solitary Wave Impact on A Vertical Cylinder*, Master of Ocean Engineering Thesis, Oregon State University, USA

Coastal Wiki: http://www.coastalwiki.org/coastalwiki/Wave_energy_converters

DualSPHysics homepage: <http://dual.sphysics.org/>

GPUSPH homepage: <http://www.ce.jhu.edu/dalrymple/GPUSPH/Home.html>

http://www.princeton.edu/~asmits/Bicycle_web/blunt.html

<http://geography.about.com/od/hazardsanddisasters/a/tsunami-overview.htm>

NEES website, Tsunami Wave Impact Forces on Cylinders: <http://nees.org/warehouse/project/325>

The California Energy Commission: <http://www.energy.ca.gov/oceanenergy/index.html>

Renewable ocean energy for North Carolina, a briefing for development of a work plan, University of North Carolina, Coastal studies institute, Manteo, NC, Personal contact, 2010.

Appendices

Appendix A (Load Balancing Algorithm)

The present method can be explained as below. Let name n_{FP} as the total number of resident particles on processor I , n_{FP_East} as the number of resident particles on the east neighbor rank and n_{FP_West} as the number of resident particles on the west neighbor rank, Figure A-1.

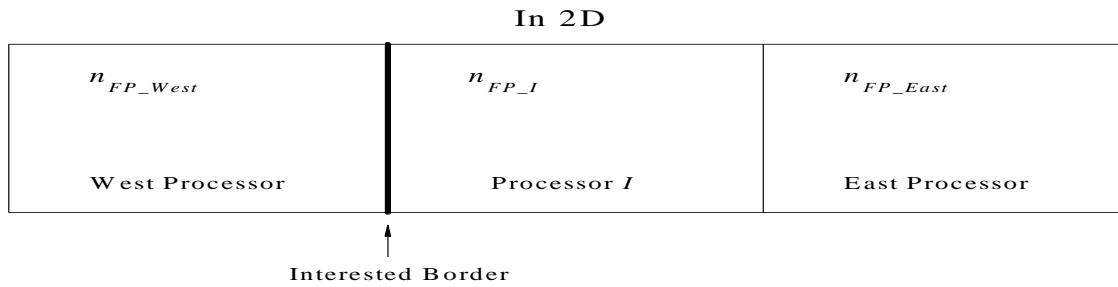


Figure A-1- Dynamic load balancing- Algorithm of adjusting the location of borders

Following criterion expresses the motion of partitions between cores (ranks or processors). The number of particles on processor I is compared only with its west processor. If adjusting is needed, then partition is moved only by one $2h$.

If $(n_{FP_I} - n_{FP_west}) > n_{FP_Max_difference}$ move the left border to the left by $2h$

If $(n_{FP_west} - n_{FP_I}) < -n_{FP_Max_difference}$ move the left border to the right by $2h$

Where $n_{FP_max_difference}$ is a given criterion for adjusting decision. A 2D dam break test proves that the above load balancing process can lead to 40% increase in speedup. In 3D, the decision of adjusting the borders is taken for a column or a row of processors. As an

example, see Figure A-2, the borders of processors 1&2, 4&5 and 7&8 move together, either all move or none of them moves.

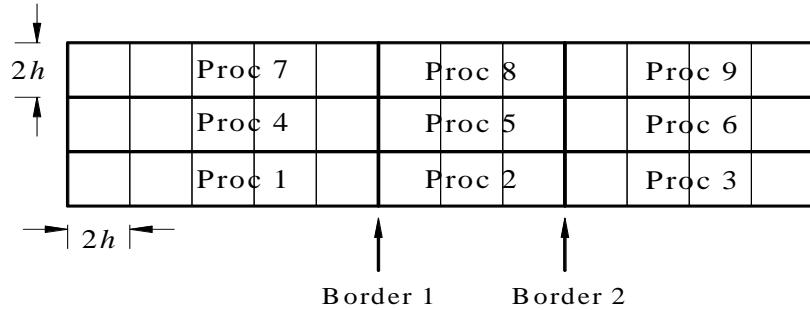


Figure A-2- Dynamic load balancing algorithm in 3D

A similar criterion as in 2D is checked for each adjacent processors in a row. If the border between rank 1&2 should be moved to the right then the parameter *i_shift* for these two processors is set to one. If it is needed to be moved to the left then *i_shift*=-1. In the case of no adjustment, *i_shift* is zero. Finally, the summation of all *i_shifts* of one column (or row) dictates the motion of border. If the resultant is a negative (positive) number, the whole borders in corresponding column move to the left (right). If becomes zero, no shifting is required. Given $n_{FP_max\ difference}= 1000$, following example shows how this method works,

$$np_proc1= 5000, np_proc2= 2000, \quad n_{FP_difference} = -3000 \quad \rightarrow \quad i_shift_1= -1$$

$$np_proc4= 4000, np_proc5= 15000, \quad n_{FP_difference} = 11000 \quad \rightarrow \quad i_shift_2= 1$$

$$np_proc7= 6000, np_proc8= 1000, \quad n_{FP_difference} = -5000 \quad \rightarrow \quad i_shift_3= -1$$

Finally, the summation of *i_shifts* is -1, that means border 1 in Figure A-2 is moved to the left. Shifting border 1 to the left decreases the size of processors 1, 4 and 7 and increases the

size of processors 2, 5 and 8 reversely. Consequently, it decreases the load of processors 1, 4 and 7 and increases the load of ranks 2, 5 and 8. Although this shifting decreases the difference between the number of particles of processors 1 and 2 or 7 and 8, it increases the difference between the number of particles of 4 and 5 undesirably. In addition the number of particles of processor 5 will become enormous that causes magnificent computational cost. Since all processors are synchronized during the computation, processor 5 increases the total simulation time due to its huge number of particles. This problem is very common when variable- h is used with linked-list NNPS algorithm. Therefore, it is much efficient to decrease the load of processor 5 which have the maximum number of particles. To do so, a new algorithm is defined in which i_shift is

$$i_shift_new = \frac{n_{FP_difference}}{n_{FP_max\ difference}} \quad (A-1)$$

In this way those borders that have higher difference have much more priority or power to decide about final shifting. With this new formulation the above example becomes

$$\begin{aligned} np_proc1 &= 5000, np_proc2 = 2000, n_{FP_difference} = -3000 \rightarrow i_shift_1 = -3000/1000 = -3 \\ np_proc4 &= 4000, np_proc5 = 15000, n_{FP_difference} = 11000 \rightarrow i_shift_2 = 11000/1000 = 11 \\ np_proc7 &= 6000, np_proc8 = 1000, n_{FP_difference} = -5000 \rightarrow i_shift_3 = -5000/1000 = -5 \end{aligned}$$

Finally, the summation of $i_shifts = i_shift_1 + i_shift_2 + i_shift_3 = +3$, that means the border 1 in Figure A-2 should be moved to the right instead of left as we found previously. In this way the number of particles in processor 5 decreases as we desire.

The explained new method is much more appropriate than original SPHysics method specifically when variable- h approach is applied in which particle number density varies tremendously in space. Although it improved the process of border adjusting, more issues

should be considered. Assume that at one stage border 1 and border 2 in Figure A-2 should be shifted to the left and right, respectively. In this case, the size of processors in the middle column, that is proc. 2, 5 and 8, will become larger and will have more loads in comparison with adjacent processors, proc. 1, 4, 7, 3, 6, and 9. Thus, in the next step both border 1 and border 2 will be back to the previous position, that is, border 1 will be shifted to the right and border 2 will be shifted to the left and this will be repeated in future steps. To improve this, a new algorithm is suggested.

In this method one border does not move back to its previous location within two sequent steps. This algorithm (the border adjusting in x -direction) can be explained in following steps:

- 1) The difference between the numbers of particles of two adjacent processors is calculated,
- 2) The absolute value in step (1) is computed,
- 3) The maximum absolute difference from step (2) in each column of processors is found,
- 4) The average of those maximum values are computed,
- 5) If the maximum value of one border (or column) is higher than the above average, that column is eligible to be shifted,
- 6) It should be checked that the movement of that border is not in the opposite direction of previous shifting.

Following figure and calculation provide an example of above steps.

| | | | | |
|--------|----|--------|----|--------|
| Proc 7 | -5 | Proc 8 | 4 | Proc 9 |
| Proc 4 | 11 | Proc 5 | -8 | Proc 6 |
| Proc 1 | -3 | Proc 2 | 6 | Proc 3 |

↑ ↑

Border 1 Border 2

Figure A-3- Border adjusting, the new algorithm (Marjani)

Figure A-3 shows nine processors and the i_shift_new is written at each border. The maximum of absolute i_shift_new of border (1) and (2) is 11 and 8, respectively. The average of 11 and 8 is 9.5. Since the maximum i_shift_new of border (1) is higher than the average, only this border is eligible for shifting. In the current step, this border should be moved to the right. If it was shifted to the left at previous step we cannot move it to the right at this step. In this new method, those borders that do not have great differences are not eligible to shift.

Appendix B (Piston Wavemaker Formulation for Solitary Wave)

The wave generation theory described here is only applicable to long waves, such as solitary or cnoidal waves, which propagate with constant form [Goring, 1979]. The wavemaker consists of a flat plate which is moved horizontally and its displacement-time history is prescribed by following formulations. The time history of the motion of the wave plate that produces the wave is called *trajectory* by Goring and presented by $\xi(t)$. At $t=0$ the wave plate begins to move from the rest $\xi = 0$ along the trajectory. The goal is to find the trajectory function which produces a predefined long wave $\eta(x,t)$ which propagates with constant celerity c . The basic term to be matched is the velocity of the plate at all times and locations, $d\xi/dt$, with the field particle velocity under that particular wave. According to the fact that the particle velocity is approximately constant over the depth for long waves, the depth averaged velocity is used here,

$$\frac{d\xi}{dt} = \bar{u}(\xi, t) \quad (\text{B-1})$$

Continuity simply shows that the averaged velocity over depth (d) is:

$$\bar{u}(x, t) = \frac{c \eta(x, t)}{d + \eta(x, t)} \quad (\text{B-2})$$

which in terms of the plate velocity, equation (B-1), becomes

$$\frac{d\xi}{dt} = \frac{c \eta(x, t)}{d + \eta(x, t)} \quad (\text{B-3})$$

To obtain the trajectory, $\xi(t)$, equation (B-3) should be integrated. It is assumed that the wave is defined in below from

$$\eta(\xi, t) = H f(\varphi) \quad (\text{B-4})$$

In which H is the wave height and

$$\varphi = k(ct - \xi) \quad (\text{B-5})$$

The derivative of (B-5) with respect to time yields

$$\frac{d\varphi}{dt} = kc - \frac{d\xi}{dt} \quad (\text{B-6})$$

And after some calculations it can be shown that

$$\frac{d\xi}{d\varphi} = \frac{\dot{\xi}}{k(c - \xi)} \quad (\text{B-7-1})$$

$$\text{or} \quad \frac{d\xi}{d\varphi} = \frac{H f(\varphi)}{kd} \quad (\text{B-7-2})$$

where $\dot{\xi} = d\xi / dt$ and finally the generation equation is

$$\xi(t) = \frac{H}{kd} \int_0^\varphi f(\alpha) d\alpha \quad (\text{B-8})$$

Now, for a solitary wave, the wave function in equation (B-4) is

$$f(\varphi) = \sec h^2 \varphi \quad (\text{B-9})$$

and $k = \sqrt{3H / 4d^3}$ and $c = \sqrt{g(h + H)}$. Substituting (B-9) in (B-8) and performing the integration gives

$$\xi(t) = \frac{H}{kd} \tanh \varphi = \tanh k(ct - \xi) \quad (\text{B-10})$$

where the value of φ and each time step is computed by iterative equation which derived from Newton's rule (for more details see Goring thesis)

$$\varphi^{n+1} = \varphi^n - \frac{-k\xi + \frac{H}{d} \tanh \varphi^n}{1 + \frac{H}{d} \sec^2 \varphi^n} \quad (\text{B-11})$$

The initial value of ξ in (B-11) is the negative value of stroke over two (i.e. $-S/2$), see Figure B-1. Stroke can be calculated by subtracting the magnitude of (B-10) at $t = -\infty$ from its value at $t = +\infty$ to yield

$$S = \frac{2H}{kd} \quad (\text{B-12})$$

Regarding Figure B-1, the duration of motion of wavemaker is defined as (B-13), therefore, the initial time is set to $-\tau/2$,

$$\tau = 2t_0 + S/c \quad (\text{B-13})$$

where $t_0=3.8/kc$. Interested reader can see Goring (1979) for more details of formula derivations. If the motion of wavemaker is prescribed by above formulation, a smooth solitary wave is generated without any small wave at its tail.

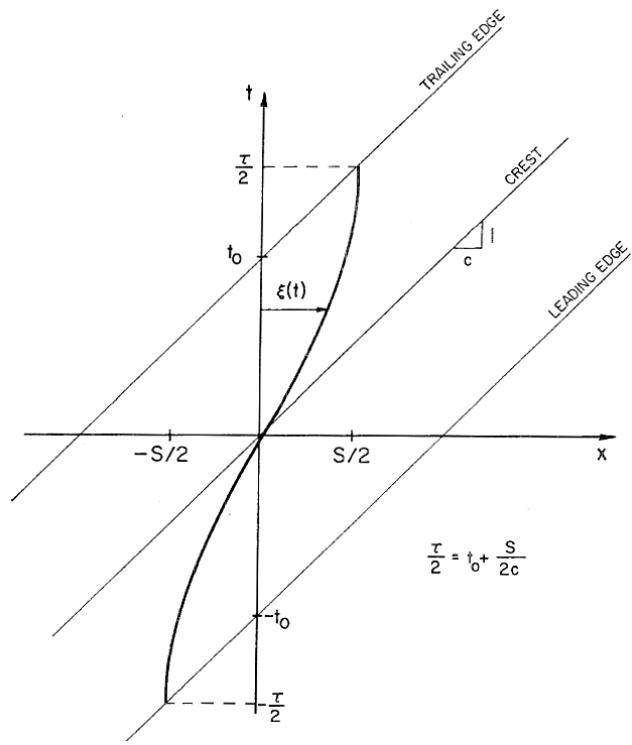


Figure B-1- Wave plate trajectory for a typical solitary wave [Goring, 1979: figure 3.7]

Appendix C (Solitary Wave: Experiment Data Conversion)

The cylinder is designed in a way that it is efficiently able to measure the horizontal wave force and overtopping momentum. Truss members connect the thin-shell aluminum cylinder to a core shaft. Four strain gauges are attached to the shaft to record the magnitude of strain due to external forces. These strain gauges are installed on the top and bottom of the core shaft in x and y direction. Since the cross frame is supported by a ball-and-socket universal joint, only horizontal force is transmitted without bending momentum. The shaft itself is held by an internal supporting structure. There is 1 cm gap between the wave basin floor and the external cylindrical shell [Zhang, 2009].

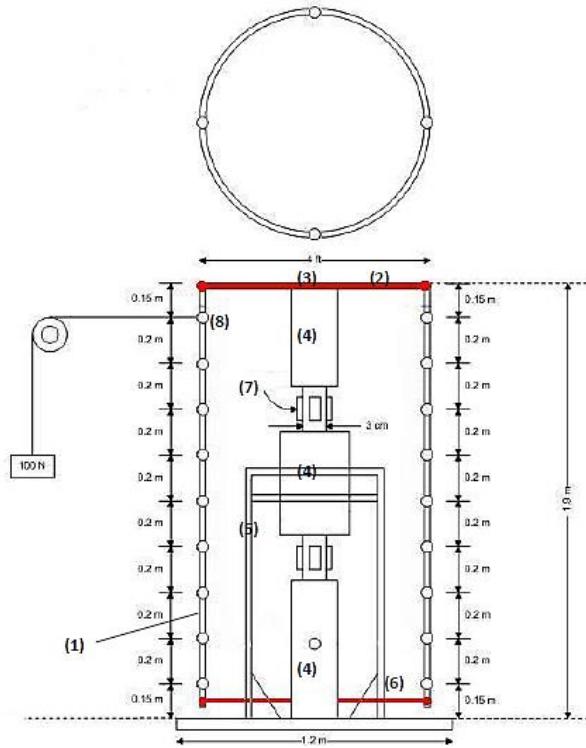


Figure C-1- Components of cylindrical structure: (1) thin shell, (2) rigid cross bar, (3) universal joint, (4) core shaft, (5) supporting structure, (6) truss members, (7) strain gauges, (8) load attachment
 [photo and caption from Zhang (2009)]

Strain gauges 1 and 3 (SG01 and SG03) are installed on the top and measure the strain associated to the force exerted from the upper cross bars while strain gauge 2 and 4 (SG02 and SG04) measure the corresponding strain from the lower bars as installed on the bottom location. SG01 and SG02 are in y direction and SG03 and SG04 are attached in x direction. Since in our study, the force in the direction of wave (x -direction) is only needed, we need the raw data of SG03 and SG04. Figure C-1 shows the cylindrical structure schematically.

The correlation between stain gauge value and total horizontal force the cylinder is yield by a set of dry load test called calibration tests. Known load ranging 0-1530 N are applied to

nine elevations at four vertical sides of the shell. Using basics of structural analysis, the portion of each strain gauge from the load is calculated. The initial value of strain gauge 3 and 4 are not zero at zero-magnitude load. Thus, the initial value is subtracted from the measured values. The total force on each gauge versus the corrected strain gauge values are plotted, see Figures C-2 and C-3. Figure C-2 represents to the loading in positive x direction and Figures C-3 represents to the loading in negative x direction.

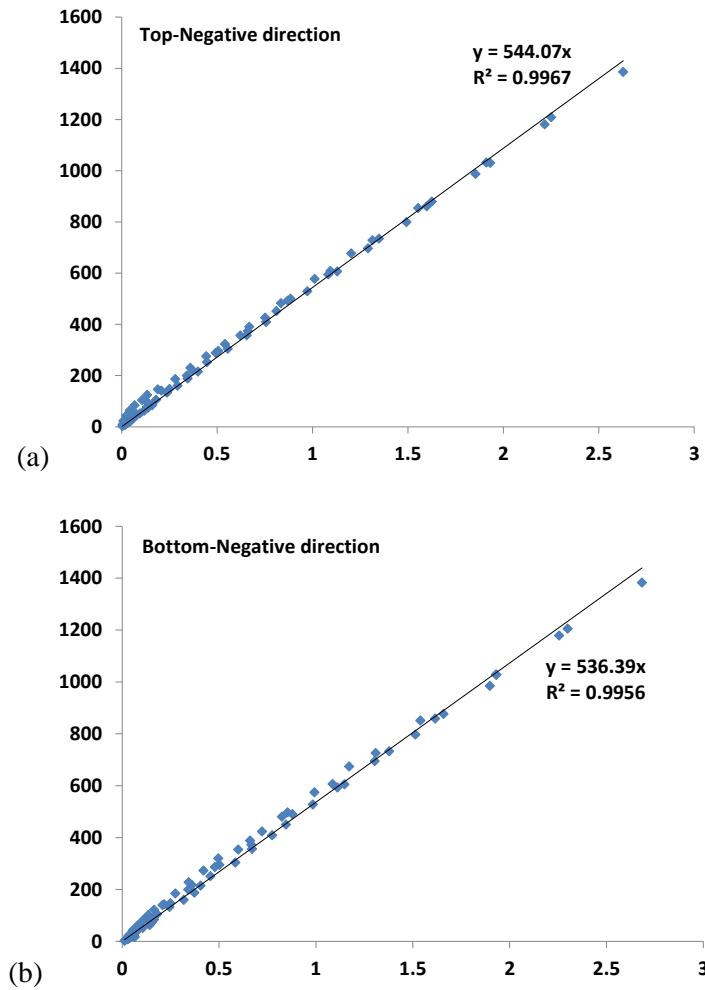


Figure C-2- Least-square fit of the (a) top-force and (b) bottom force strain gauge data as the load exerted in negative x direction

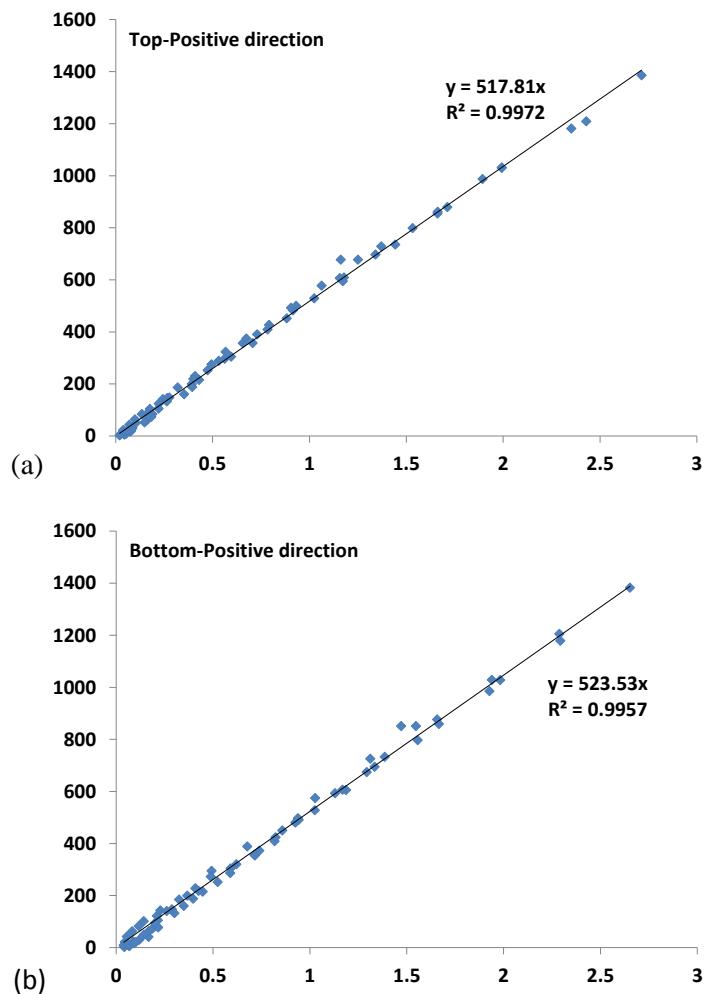


Figure C-3- Least-square fit of the (a) top-force and (b) bottom force strain gauge data as the load exerted in positive x direction

Based on data analysis of results of calibration test, the following relationship between the strain gauge value and actual force is yielded

$$FT_x = Coeff_{03} \cdot SG\ 03 \quad (B-1)$$

$$FB_x = Coeff_{04} \cdot SG\ 04 \quad (B-2)$$

If the magnitude of strain gauge is positive, the value of $Coeff_{SG}$ is read from Figure C-2; otherwise Figure C-3 determines the magnitude of $Coeff_{SG}$. Finally, the summation of FT_x and FB_x gives the total horizontal force on the structure in x direction. When a real solitary wave test is run, the strain gauge data is recorded with a frequency of 1000. The magnitude of $Coeff_{SG}$ is computed according to the strain gauge value. A simple program is written in FORTRAN to read the raw data and find the coefficients and convert them to real force unit using above equations and plots.

Program conversion

```

integer ntime
real pres(42), ratio3_4,ratio4_3,coef,coef3,coef4,dt,time

ntime = 120033
freq = 1000
dt = 1.0/freq

open(23,file='force_raw.txt',status='old')
open(24,file='force_converted')
npp = 0
time = 0.0
read(23,*,end=300) std, WMstart, SG3_ini,SG4_ini,
&                               SG1,SG2, pres(1:42)
do ii=2,ntime
    read(23,*,end=300) std, WMstart, SG3,SG4, SG1,SG2,
pres(1:42)
    npp=npp+1

    SG3 = SG3 - SG3_ini
    SG4 = SG4 - SG4_ini

    if(abs(SG3).gt.1.0e-3.and.abs(SG4).gt.1.0e-3) then
        call coefficient(SG3,coef,3)
    endif
    if(abs(SG4).gt.1.0e-3.and.abs(SG3).gt.1.0e-3) then
        call coefficient(SG4,coef,4)
    endif

```

```

        time = time + dt
        if(mod(npp,50).eq.0.and.time.gt.20.and.time.lt.40.) then

            write(24,*) time - 20.0 , coef3*SG3 + coef4*SG4
        endif

    enddo ! ii=1,ntime

300     close (23)

end

subroutine coefficient(SG,coeff,SG_num)

real SG,coeff
integer SG_num

if(SG.gt.0.0) then
    if(SG_num.eq.3) then
        coeff = 544.07
    else
        coeff = 536.39
    endif
else
    if(SG_num.eq.3) then
        coeff = 517.81
    else
        coeff = 523.53
    endif
endif

return
end subroutine

```