City Research Online

City St George's, University of London

Citation: Zhou, J. (2010). Numerical investigation of breaking waves and their interactions with structures using MLPG_R method. (Unpublished Doctoral thesis, City University London)

This is the unspecified version of the paper.

This version of the publication may differ from the final published version. To cite this item please consult the publisher's version.

Permanent repository link: https://openaccess.city.ac.uk/id/eprint/1083/

Copyright and Reuse: Copyright and Moral Rights remain with the author(s) and/or copyright holders. Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge, unless otherwise indicated, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way. For full details of reuse please refer to City Research Online policy.

City Research Online: http://openaccess.city.ac.uk/ publications@citystgeorges.ac.uk

NUMERICAL INVESTIGATATION OF BREAKING WAVES AND THEIR INTERATIONS WITH STRUCTURES USING MLPG_R METHOD

By

Juntao Zhou

Supervisor

Prof. Qingwei Ma

A thesis submitted in fulfillment of

requirement of degree of

Doctor of Philosophy

School of Engineering and Mathematical Sciences

City University, London

June, 2010

CONTENTS

LIST OF FIGURES **5**

LIST OF TABLES **10**

ACKNOWLEDGEMENTS **11**

DECLARATION **12**

ABSTRACT **13**

LIST OF SYMBOLS **15**

1. INTRODUCTION **17**

1.1 Background 17

1.2 Objectives of the study 20

1.3 Outline of the thesis 20

2. LITERATURE REVIEW **22**

2.1 Mesh-based methods 22

2.2 Meshless methods 23

2.2.1 Smooth Particle Hydrodynamic method (SPH) 25

2.2.2 Moving Particle Semi-implicit method (MPS) 27

2.2.3 Meshless Local Petro-Galerkin method (MLPG) 29

2.3 Numerical methods for calculation pressure gradient in meshless methods 31

2.4 Numerical methods for implementation of solid boundary condition in meshless methods 33

2.5 Numerical methods for identifying the free surface 35

3. MATHEMATICAL MODEL AND NUMERICAL METHOD **38**

3.1 Governing equations and boundary conditions 38

3.2 Numerical Procedure 39

3.3 MLPG_R Formulation for 2D cases 41

3.3.1 Discretization of the governing equation for pressure 44

3.3.2 Numerical Technique for Domain Integration 46

3.3.3 Numerical implementation of solid boundary condition 49
3.3.4 Test the effectiveness of parameter α in Eq. (3.2.3)..... 53
3.4 MLPG_R Formulation for 3D cases 57
3.4.1 Numerical technique for evaluating domain integrals..... 59
3.4.2 Numerical technique for evaluating surface integrals 63
3.4.3 Effectiveness of the semi-analytical technique for surface integrals..... 67
4 NUMERICAL TECHUNIQUES FOR IDENTIFYING PARTICLES ON FREE SURFACE 71
4.1 Introduction..... 71
4.2 Mixed Particle Number Density and Auxiliary Function Method (MPAM) for 2D cases..... 72
4.3 Effectiveness of MPAM for identifying the free surface particles..... 75
4.4 Mixed Particle Number Density and Auxiliary Function Method (MPAM) for 3D cases..... 77
5. NUMERICAL SIMULATION OF DAM BREAKING.....80
5.1 Two dimensional dam breaking 80
5.1.1 Convergent investigation on different values of $\Delta x / \Delta \tau$ 84
5.1.2 Convergent investigation on different space increments Δx 86
5.2 Three-dimensional dam breaking 88
6. BREAKING WAVES OVER NON-FLAT SEABED.....100
6.1 2D breaking waves on a slope 101
6.1.1 Convergence investigation on different value of $\Delta x/\Delta \tau$ 102
6.1.2 Convergence investigation based on different value of Δx 103
6.2 2D breaking waves over a submerged step..... 109
6.3 3D breaking waves on a slope 112
6.3.1 Convergence investigation on different values of $\Delta x / \Delta \tau$ 113
6.3.2 Convergence investigation on different values of Δx 115
7. SIMULATION OF VIOLENT SLOSHING WAVES 118
7.1 Mathematical model and numerical techniques.....119

7.2 Numerical investigations on 2D sloshing waves 121
7.2.1 Convergence investigation and numerical validation 121
7.2.2 Behaviors of impact pressure in a baffled tank 128
7.3 Numerical investigations for 3D sloshing waves 133
8. 3D NUMERICAL INVESTIGATIONS ON VIOLENT WAVE IMPACT ON THE CYLINDER: OFFSHORE ENERGY STRUCTURE 139
8.1 Investigations of relationship between pressure impact peak and breaking waves . 141
8.2 Effects of Locations of Structures 143
8.3 Effects of different wave heights 145
9. CONCLUSIONS AND FUTURE WORK 147
9.1 Numerical techniques 148
9.2 Two dimensional and three dimensional sloshing cases 149
9.3 Interaction between breaking waves and a fixed cylinder 149
9.4 Future work 150
REFERENCES 151
APPENDIX A: PUBLICATION LIST 162
APPENDIX B: Gradient Calculation Scheme 163

LIST OF FIGURES

Fig.1.1.1 Wave breaking in sea……..……..…….…….17

Fig.1.1.2 Breaking wave impacting on a ship…….……..17

Fig.2.4.1 Illustration of the influence domain for wall particles in (a) BC1, (b) BC2 and (c) BC3 (solid circle: wall particles; hollow circle: water particles)……..……....32

Fig.3.0.1 Schematic of computational domain and the coordinate ….……..…….……..38

Fig.3.3.1 Illustration of particles, integration domain and support domain (a: wall particle; b: free surface particles and c: inner particles)…….…..…40

Fig.3.3.2 Illustration of division of an integration domain……...…….…...44

Fig.3.3.3 Illustration of the influence domain for wall particles (solid circle: wall particles; hollow circle: water particles)……...….49

Fig.3.3.4 Schematic of different solid boundaries…….……...….50

Fig.3.3.5 Schematic analysis of normal derivative for *d* boundary particle…….……...….52

Fig.3.3.6 Effects of different α on the PND in dam breaking cases …...…….…….....55

Fig.3.3.7 Configurations of particles for the cases with different value of α …….……57

Fig.3.4.1.Illustration of division of an integration domain……..…….59

Fig.3.4.2 Illustration of integration domain in θ-γ plane…….……..……64

Fig.3.4.3 A schematic view of the tank…….……67

Fig.3.4.4 Comparison of static pressure obtained by using the Gaussian quadrature and the semi-analytical technique for estimating the surface integral……..68

Fig.3.4.5 Comparison of wave profiles at different instants obtained by using the semi-analytical technique and the Gaussian quadrature with 32 and 72 Gaussian points, respectively……..……..……...70

Fig.4.2.1 Two typical examples of incorrect identification of free surface particles. (Solid circle represents free surface particle identified by the PNDM; hollow one

represents inner particle identified by the PNDM)……..…….74

Fig. 4.2.2 Local coordinate system at Particle *I* (inner circle denotes integration domain of the particle; outer circle denotes the support domain on it)…….……..…74

Fig.4.3.1 Comparisons of particle configurations obtained by using different free surface identification techniques (black color: free surface particles; grey or blue color: wall particles or inner particles)…….……...…76

Fig.4.3.2 Snapshots of the water particles in pre-breaking stage (Blue particles: inner water particle; Black particles: free surface particle; Grey particles: sloping seabed and Red line: experiment data (Li & Raichlen, 1998))……..….77

Fig.4.3.3 Snapshots of the water particles in the post-breaking stage (Blue particles: inner water particle; Black particles: free surface particle; Grey particles: sloping seabed)……..……78

Fig.4.4.1 Local domain at Node *I* for the definition of auxiliary function (the inner sphere denotes integration domain of the particle; the outer sphere denotes the support domain; the 6 coloured cylinders have the same diameter as the inner sphere)…….79

Fig.5.1.1 Dam breaking: problem definition.……...80

Fig.5.1.2(a) Snapshots of free surface at different time steps (grey particles: wall; the black ones: free surface particles) ……..…….81

Fig.5.1.2(b) Snapshots of free surface at different time steps (grey particles: wall; the black ones: free surface particles)…….……..82

Fig.5.1.3 Comparing the motion of the leading edge with other simulated results and experiment data……..83

Fig.5.1.4 Comparing the motion of the leading edge corresponding different values of $\Delta x / \Delta \tau$ ……...……..84

Fig.5.1.5 Non-physical occurrence with $\Delta x / \Delta \tau$ =1…….….85

Fig.5.1.6 Comparing free surface profiles corresponding to different values of $\Delta x / \Delta \tau$ …..86

Fig.5.1.7 Comparing free surface profiles corresponding to different values of Δx with $\Delta x / \Delta \tau = 2.5$ ……...87

Fig.5.2.1 Schematic view of the tank and the obstacle (unit: m)…….……....89

Fig.5.2.2 Details of the obstacle and the pressure transducers (unit: m)…….…….....89

Fig.5.2.3 General description of the system: top (top picture) and side (bottom picture) views

Fig.5.2.4 Snapshots of the free surface at (a) t=0.227s (b) t=0.493s and (c) t=0.701s…….....90

Fig.5.2.5 Comparison of the free surface profiles at $t \approx 0.4$s (a: experiment, b: FVM numerical results, both results from Klessfsman, et al, 2005; c: present method)……...91

Fig.5.2.6 Comparison of the free surface profiles at $t \approx 0.56$s (a: experiment, b: FVM numerical results, both results from Klessfsman, et al, 2005; c: present method)……...92

Fig.5.2.7 The time histories of free surface elevations at (a) for H2, (b) for H3 and (c) for H4……93

Fig.5.2.8 Pressure time histories at (a) for P1, (b) for P2 and (c) for P3…….……....….94

Fig.5.2.9 Pressure time histories at (a) for P5, (b) for P6, (c) for P7 and (d) for P8….……...96

Fig.5.2.10 Comparison of free surface profile at the same time (a) results without smoothing, (b) results with smoothing……..……..……..…98

Fig.5.2.11 Pressure time histories at (a) for P5, (b) for P6, (c) for P7 and (d) for P8……..99

Fig.6.0.1 Illustration of model set-up for the solitary wave….……...……..……..100

Fig.6.1.1 Comparison between experimental wave profiles (Li and Raichlen, 1998) and numerical results for different values of $\Delta x/\Delta\tau$ ……...104

Fig.6.1.2 Comparison between experimental wave profiles [Li and Raichlen (1998)] and numerical results obtained by using different Δx values when $\Delta x/\Delta\tau = 5$ …..……..105

Fig.6.1.3 Wave profiles in the post-breaking stage obtained by using different values of Δx at when $\Delta x/\Delta\tau = 5$ ……..…...106

Fig.6.1.4 Comparisons between laboratory photographs (Li and Raichlen, 2003) and numerical snapshots (right column: Blue particles: inner water particle; Black particles: free surface particle)…….…….108

Fig.6.2.1 Sketch of the problem about a solitary wave propagating to and over the step…..110

Fig.6.2.2 Snapshots of solitary wave evolution over the step at different time steps……..…100

Fig.6.2.3 Comparisons of wave elevations between the numerical results (line) and experimental data (mark) (Yasuda et al, 1997) at three different gauges (P1, P2 and P3)...……..……111

Fig.6.2.4 Velocity field around the step when wave crest is passing by…….112

Fig.6.3.1 Comparison between experimental wave profiles (Li and Raichlen, 1998) and numerical results for different values of $\Delta x/\Delta\tau$ ……..…….114

Fig.6.3.2 Comparisons between experimental wave profiles [Li and Raichlen (1998)] and numerical results obtained by using different Δx when $\Delta x/\Delta\tau = 5$ ……..…….116

Fig.6.3.3 Wave profiles in the post-breaking stage obtained by using different values of Δx when $\Delta x/\Delta\tau = 5$ ……..…….....117

Fig.7.2.1 Schematic view of experimental set-up and corresponding tank sizes……...…121

Fig. 7.2.2 Comparison between the experiment and numerical results……..123

Fig.7.2.3 Time histories of pressure with different values of $\Delta x/\Delta\tau$ for *Nz=30*……....…124

Fig.7.2.4 Time histories of pressure with different *Nz* for $\Delta x/\Delta\tau$ =2……..…..…124

Fig.7.2.5 Pressure in one period with different *Nz* for $\Delta x/\Delta\tau$ =2……...….125

Fig.7.2.6 Wave profile corresponding to the first peak shown in Fig. 7.2.5……..…126

Fig.7.2.7 Wave profile corresponding to the first trough value shown in Fig. 7.2.5……..…126

Fig.7.2.8 Wave profile corresponding to the second peak shown in Fig. 7.2.5…….….126

Fig.7.2.9 Wave profile corresponding to the second trough shown in Fig. 7.2.5……..…..127

Fig.7.2.10 Wave profile corresponding to the third peak shown in Fig. 7.2.5……..…..127

Fig.7.2.11 Sketch of baffle and the neighbouring water particles (black solid circles: baffle particles and wall particles; hollow circles: water particles; dash circles represent the initial influence domain; the solid curve represent effective influence domain)……..……..….129

Fig.7.2.12 Schematic view of tank and corresponding tank sizes (*A*: No baffle; *B*: Middle baffle and *C*: Left baffle)…….……...131

Fig.7.2.13 Comparison of pressure time histories at the point *a*……..…..132

Fig.7.2.14 Wave profiles corresponding to a primary pressure peak in the last period shown in

Fig. 7.2.12 for Case A, (b) for Case B and (c) for Case C…….132

Fig.7.2.15 Comparison of pressure time histories at the point *b*…….…...133

Fig.7.3.1 Schematic view and sizes of the tank…….....134

Fig.7.3.2 Snapshots of violent slosh wave (black particles: free surface particle; blue: velocity field)……..……...136

Fig.7.3.3 Comparisons of the time histories of impact pressure on different record points……..……137

Fig.8.0.1 The schematic view and details of tank and the structure……..…….……140

Fig.8.1.1 Different stages of wave impacting on the wind turbine structure (solitary wave height is 0.7, cylinder location is *C*)…….……141

Fig.8.1.2 Time histories of pressure acting on the wind energy structure (solitary wave height is 0.7, location: *C;* Point 1: 0.1 above the MWL; Point 2: 0.3 above the MWL)……...…….……142

Fig.8.1.3 Enlargement of free-surface particle distribution near structure viewed by *x*-*z* coordinates (solitary wave height is 0.7, location: *C*, $\tau \approx 13.44$)……...….142

Fig.8.2.1 Free surface profiles near the structures at $\tau \approx 13.2$, which are in different locations (solitary wave height is 0.7, a for Location A, b for Location B and c for Location C)…….…….……144

Fig.8.2.2 Pressure time histories at Point 2 on the structures located at different positions (Point 2: 0.3 above the MWL)……..144

Fig.8.2.3 Free surface profile near the structure at $\tau \approx 12.48$ at location B……...……145

Fig.8.3.1 Pressure time histories at Point 2 corresponding to different solitary wave heights (Point 2: 0.3 above the MWL)……..146

LIST OF TABLES

Table 3.4.1 Comparison of CPU time required by using different methods to evaluate the surface integral (GQ: Gaussian quadrature) for a static case……..…….67

Table 3.4.2 Comparison of CPU time required by using different method to evaluate the surface integrals (GQ: Gaussian quadrature) for a dynamic case……....70

Table 7.2.1 Calculation conditions of convergent investigation…….122

ACKNOWLEDGEMENTS

This study is sponsored by two projects, the first one is ‘Interaction between Breaking Waves and Three-dimensional Surface-Piercing Structures ‘funded by the Leverhulme Trust, UK and the second one is ‘Numerical investigation on violent sloshing loads in a FPSO tank’ funded by the ABS, USA , for which the author is most grateful.

I would like to express my sincere gratitude to my supervisor Dr. Q.W. Ma, for the guidance, support, assistance, friendship and understanding he has shown in helping me complete this work. His excellence in academics and personality will inspire me for my future. I would also like to thank him for his comments on the draft of this thesis.

A special thanks to Dr. Shiqiang Yan in our research group, who read the manuscript in detail. The valuable suggestions and discussions lead to a significant improvement of this thesis. I would also offer my thanks to my colleagues in the school, for their suggestions and discussion on my work.

Finally, and most importantly, I would like to thank my wife, Xiaoyan Lin, my son, Xiuheng Zhou, my parents and parents-in-law, and my brothers and sister. Without their encouragements and supports, this work would not have been completed.

DECLARATION

No portion of the work referred to in the thesis has been submitted in support of an application for other degree or qualification of this or any other university or other institute of learning.

I grant powers of discretion to the City University Library to allow this thesis to be copied in whole or in part without any reference to me. This permission covers only single copies made for study purpose subject to normal condition of acknowledgement.

ABSTRACT

Meshless Local Petrov-Galerkin method based on Rankine source solution (MLPG_R) has been developed by Dr. Qingwei Ma (Ma, 2005b) and has been used to simulate the nonlinear water wave problems in 2D cases without the occurrence of the breaking waves. In this thesis, MLPG_R method has been further developed to numerically simulate breaking waves and the interactions between breaking waves and structures in 2D and 3D cases. The main difference between this meshless method and conventional mesh-based methods is that the governing equations are solved in terms of particle interaction models, without the need of computational meshes. Therefore, this method avoids the time-consuming mesh generating and updating procedures which may be necessary and may need to be frequently performed in the mesh-based methods. Furthermore, in order to simulate the breaking waves well, several novel numerical techniques are developed and adopted. The numerical technique for implementing the solid boundary condition for meshless methods is proposed, which is more robust than others in terms of accuracy and efficiency. A technique for meshless interpolation (SFDI scheme) is adopted, which is as accurate as the more costly moving least square (MLS) method generally but requires much less computational time than the latter. A newly developed technique for identifying the free surface particles is presented, which is much more robust than those existing in literature. A semi-analytical method for numerical evaluation of integrals in a local domain and on its surface is presented to form the matrix for the algebraic equations, which makes it possible to modelling the 3D problems on personal computers.

The newly extended MLPG_R method is applied to simulate the waves generated by a wave maker and their propagations, overturning and breaking over flat and sloped seabed. And it is also applied to 2D and 3D dam breaking cases and violent sloshing cases. The convergence properties of this method in different cases are investigated. Some of the results have been validated by experimental data and numerical results obtained by other

methods. Satisfactory agreements are achieved. Based on these numerical investigations, a number of conclusions have been made, including that the breaking waves can cause large pressure with several peaks when they impact on structures; the behaviour of pressure strongly depends on the relative locations of structures to the breaking point of breaking waves. Breaking waves in a sloshing container can also cause more than one peaks, which is correlated with the direction change of water motion within the container. These investigations can give us better understanding of the impact pressure, breaking wave and interactions between breaking wave and structures.

LIST OF SYMBOLS

d	water depth
g	gravitational acceleration
h	wave height of solitary wave
p	pressure
$\vec{n}, \vec{\tau}$	normal and tangential vectors, respectively
r	distance between two partilces
r_0	cut-off radius
m_I	mass of particle I
n_I	particle number density of particle I
n^0	initial particle number density
n_I^*	particle number density of particle I at the intermediate step
t, τ	dimensional time and nondimensionlised time, respectively
$\vec{u}$	velocity vector of the flow
$\vec{u}^*$	velocity vector of the flow at the intermediate time step
u, v, w	x-,y- and z-directional velocity components of fluids, respectively
x,y,z	spatial coordinates in a Cartesian coordinate system, respectively
$\vec{x}$	spatial coordinate
B	width of tank
D_s	degree of the spatial dimensions
$[K]$	coefficient matrix of the pressure equations
$[F]$	coefficient matrix of the pressure equations
N	total number of particles in the support domain of targeted particle
Nz	particle number along z-direction
R_I	radius of integration domain of particle I

∇	gradient operator
∇^2	Laplace equation operator
$\frac{\partial}{\partial t}$	Partial time derivative in the Cartesian coordinate system
$\frac{D}{Dt}$	substantial derivative defined as $\frac{D}{Dt} = \frac{\partial}{\partial t} + \nabla\phi \cdot \nabla$
ν	kinematic viscosity
ρ	density of the fluid
Ω_I	integration domain of particle I
$\partial\Omega_I$	surface domain of integration domain of particle I
φ	test function
$\frac{D}{Dt}$	substantial derivative defined as $\frac{D}{Dt} = \frac{\partial}{\partial t} + \nabla\phi \cdot \nabla$
$\vec{U}$	velocity of the rigid boundary
$\vec{\dot{U}}$	acceleration of rigid boundary
Δt	time step
Φ	shape function
W	weight function
ψ	base function
θ, γ	angle parameters

1. INTRODUCTION

1.1 Background

Wave breaking (see, Fig.1.1.1) is a general phenomenon in nature. This phenomenon plays a vital role in air–sea interactions and wave-structure interactions. In addition, breaking waves may release huge amounts of energy, which may severely damage coastal structures, vehicles in the sea (see, Fig.1.1.2) and threaten the lives on them. Therefore, breaking waves and their interactions with structures have been of a great concern in offshore/coastal engineering.

Fig.1.1.1 wave breaking in sea
www.waterencyclopedia.com

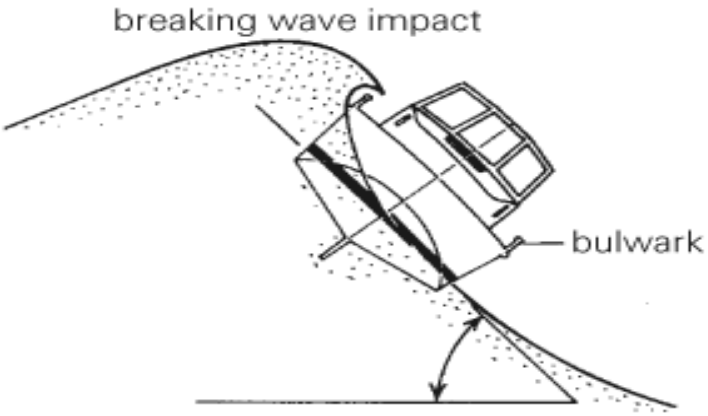


Fig 1.1.2 breaking wave impacting on a ship
www.answers.com

Many efforts have been made to achieve a good understanding of the phenomenon. Previous investigations on breaking waves mainly focused on the laboratory experiments or field observations. The experiments can produce very useful and reliable results for some cases but are generally very expensive. Furthermore, these results may be applicable only for particular cases. Alternatively, many numerical methods have been developed to address this issue. Accurate numerical simulation can provide detailed information on the hydrodynamics of breaking waves, which is not easily measured during physical experiments. Once the models have been validated, they can be employed to simulate more general and complicated cases. Thus, numerical modeling, instead of experimental study, may be preferred in the community.

To numerically simulate these problems, there are mainly two classes of theoretical models for cases with finite water depth. One is based on the general flow theory and the other is based on the potential theory. In the first class of models, the Navier-Stokes and continuity equations together with proper boundary conditions are solved, while in the second class, the Laplace's equation with fully nonlinear boundary conditions is dealt with. For brevity, the first class models will be called NS models and the second called FNPT (fully nonlinear potential theory) models in this work. Various numerical methods, such as finite element method and boundary element method have been adopted to solve the FNPT model to investigate the nonlinear water waves and their interaction with structures. However, due to ignoring the fluid viscosity and irrotational assumption, FNPT models usually can only model the overturning waves up to the jets hitting the free surface in front of it and can not model the post-breaking waves with which viscosity can be important. Therefore, if the cases with post-breaking are of main concern, NS models should be employed. On the other hand, numerical methods for simulation of breaking wave can be divided into two groups. i.e. mesh-based methods and meshless methods. The mesh-based method for nonlinear water waves mainly include finite difference method (Miyata, 1986; Lin and Liu, 1998); the finite element method (Ma and Yan, 2006), the boundary element method (Grilli, Guyenne and Dias, 2001; Biausser, et al, 2003, 2004; Guignard, Grilli, Marcer, and Rey, 1999) and the finite volume method (Devrard, et al, 2005a; Devrard, et al, 2005b). They all produce many impressive results. However, a limitation of

these methods is that a computational mesh/grid is required. Generally, those methods are based on either Eulerian description or Lagrangian description. For the methods based on Eulerian description, the computational mesh does not need to be updated but extra equations are required to identify the free surface. For the methods based on Lagrangian description, the mesh or grid may need to be updated repeatedly to follow the motion of the free surface, which is often a difficult and/or time-consuming task, particularly in case with breaking waves. Whereas, in the meshless methods, the fluid domain is discretised as particles, instead of computational mesh/grid in mesh-based methods, and a Lagrangian form of governing equation or boundary condition is imposed on every particle. The particles follow the motions of the fluid without the need of being regenerated repeatedly. Hence, meshless methods have high potential in modelling breaking waves. By far, many meshless methods have been reported in the literatures, such as Meshless Local Petro-Galerkin (MLPG) (see, for example, Atluri and Zhu, 1998, 2000; Atluri, and Shen, 2002; Batra & Ching, 2002; Lin and Atluri, 2000, 2001; Ma, 2005a, 2005b, 2007; Li & Atluri, 2008; Avila & Atluri, 2009), Moving Particle Semi-implicit method (MPS) (see, for instance, Koshizuka and Oka, 1996; Gotoh and Sakai, 2006), the Smooth Particle Hydrodynamic (SPH) (e.g. Monaghan, 1994, Shao 2006), the finite point method (Onate, Idelsohn, Zienkiewicz, Taylor and Sacco, 1996), the element free Galerkin method (Belytschko, Lu and Gu, 1994), the diffusion element method (Nayroles, Touzot and Vilon, 1992). Among them, the SPH, MPS and MLPG have been widely used to simulate nonlinear water waves by many authors. Ma (2005a, b; 2007) simulated nonlinear water waves, sloshing waves and freak waves by using the MLPG method. 2D and 3D breaking waves and the interactions with offshore structure have been modelled by Ma & Zhou (2009), Zhou & Ma (2010) and Zhou, Ma, Zhang & Yan (2009). The MPS method has been applied to simulate the collapse of a water column (Koshizuka and Oka, 1996), the shallow water sloshing waves (Naito and Sueyoshi, 2002), the breaking waves (Gotoh & Sakai, 1999, Khayyer & Gotoh, 2008a, 2008b and 2009) and the wave-body interaction (Gotoh and Sakai, 2006). The SPH method has been successfully used to simulate waves propagating towards beaches (Monaghan 1994, Lo and Shao, 2002) and many other cases. Although the MPS and SPH methods have produced many interesting and encouraging results, there are still some issues existing in these methods, such as, the oscillating of pressure (Sueyoshi & Naito,

2002, 2004a and Ma et al., 2009). Due to great flexibility of using different test functions, many others meshless methods can be considered to be special cases of the MLPG method (Ma, 2005b). Therefore the MLPG method is chosen in this study.

1.2 Objectives of the study

This study aims to extend the MLPG_R method based on the general flow theory to numerically simulate breaking waves and their interactions with 2D and 3D structures. The extended MLPG_R method is used to simulate 2D dam breaking cases, this case is for violent free surface flow similar to the green water or overtopping on floating bodies; a breaking wave over a step is investigated, this case is similar to the mitigation of tsunami effects or vertical breakwater to protect the coastal structures; violent sloshing waves in FPSO are simulated with different sloshing frequencies and different filling ratios and 3D violent wave impact on the wind energy structures are numerically simulated by MLPG_R method. The objectives of this study are centred on:

1. Developing an efficient and robust numerical scheme to track the free surface during the simulation of violent breaking wave cases;
2. Developing an efficient numerical method and procedure to simulate 2D and 3D breaking waves based on MLPG_R method;
3. Applying the developed method to investigate the interaction between breaking waves and fixed structures.

1.3 Outline of the thesis

This chapter introduces the background as well as the aims and objectives of this study. Other chapters are outline below: Chapter 2 reviews the literatures related to the numerical methods. The mathematical model and numerical procedure used on this study are presented in Chapter 3, together with the numerical implementations of solid boundary conditions. Chapter 4 gives the numerical techniques for identifying free surface. In Chapters 5-7, the developed method is validated using different cases and its convergence properties are discussed. Chapter 8 mainly focuses on a 3D application of the developed method, i.e. aiming

to investigating violent wave impact on offshore wind energy structure. Chapter 9 summarizes the conclusions and recommendations on the future work.

2. LITERATURE REVIEW

This chapter will review numerical methods regarding the modelling of breaking waves interacting with structures. Because the thesis focuses on the numerical simulation of breaking waves and their interactions with structures, only the related numerical methods are discussed below. These numerical methods may be split into two groups. One is mesh-based methods, in which the computational domains are discretized into meshes/grids; the other is the meshless methods where particles are used to represent the domains.

2.1 Mesh-based methods

In the mesh-based methods, the computational domain is discretised into many elements/grids. The computational mesh can be either fixed in space (the Eulerian formulation), follows the fluid flow (the Lagrangian formulation), or moves at an arbitrary velocity (the arbitrary Lagrangian - Eulerian fomulation).

The mesh-based methods in Eulerian formulation have been widely used to model wave problems. However, there are some big challenges when dealing with the violent breaking waves cases. One of them is how to track the free surface, which need to be determined during the simulation. In order to simulate the breaking wave cases, some interface capturing technologies have been developed, for example, Marker and Cell method (MAC, Harlow and Welch, 1965); Volume of Fluid method (VOF, Hirt and Nichols, 1981) and Level Set Method (LSM, Osher and Sethian, 1988), (more details in Section 2.5). In the methods, extra equations need to be solved, which will definitely add extra computational cost. Furthermore the practical calculation domain must be bigger than necessary when deal with the large distortions of interfaces between two different materials. This is because the computed domain should cover the area where the fluid may reach. Another problem is that the meshes are fixed, not moving with the fluid. Consequently the convection terms will exist in the formulation, and thus the numerical diffusion will be inevitably caused. Due to the facts, the methods are not very suitable for simulating violent breaking waves.

For the mesh-based methods in Lagrangian formulation, it overcomes the problems caused by fixed meshes. The non-linear convective terms no longer appear and the meshes need only to be generated in the regions of space occupied by the fluid. However, if the motion of the fluid becomes geometrically complex, the mesh undergoes severe distortion, the accuracy of this method is highly affected and the numerical methods become unstable. So the computational meshes need to be updated repeatedly to follow the motion of the free surface and need to be maintained good quantity. This is often a difficult and time-consuming task, particularly in the 3D cases with breaking waves.

To overcome this problem, the Arbitrary Lagrangian-Eulerian (ALE) approach was proposed (Hirt et al., 1974). The ALE formulation is a hybrid approach, in which the computational mesh does not need to adhere to particles or to be fixed in space but can be arbitrarily moved. Based on this description, both the Eulerian and Lagrangian methods are special cases of the ALE method. Therefore, the ALE formulations combine the merits of both Eulerian and Lagrangian formulations and alleviate their shortcomings. Of course, the governing equations are made more complex to account for the moving velocities of the mesh. Nevertheless, it does not yield accurate results when dealing with large deformations (Li and Liu, 2002) or fragmentations (Gotoh et al., 2005). And the convective transport effects in ALE often lead to spurious oscillation that needs to be stabilized by an artificial diffusion. The ALE formulation has been discussed and used in many publications. Huerta & Liu (1988), Henning & Peter (2000), Teng, Zhao & Bai (2001), Souli & Zolisio (2001), Fabián, Raúl & Srinivasan (2004), et al and Tanaka & Kashiyama (2006) are some examples of applications of this method related to the free surface problems.

2.2 Meshless methods

In contrast to the commonly used mesh-based method, a new type of numerical methods has been given many concerns in research and offshore/coastal engineering. They are meshless methods or gridless methods, which are expected to become superior to the conventional mesh-based methods. In meshless methods, the fluid domain is discretized into many particles without the need of any underlying mesh or nodal connectivity. Particles are used to represent the fluid and its motion; hence each particle has information about a set

of field variables such as mass, momentum and position. There is no mesh in the computational domain and so it does not need to deal with the meshes. As a result, it becomes easier to treat the large deformations, fluid fragmentation and coalescence compared to mesh-based methods. In meshless methods, particles can either be fixed (the Eulerian formulation) or moveable (the Lagrangian formulation or ALE formulation).

For the meshless methods in the Eulerian formulation, such as a gridless Euler / Navier-Stokes solution algorithm (Batina, 1993), it is relatively easier to apply to complex geometry than the mesh-based methods, because this method allows to use points which are more appropriately located. However numerical diffusion is inevitable during the calculation of convection.

For the meshless methods in the Lagrangian formulation (also called particle methods), particles move at the same velocity as the fluid velocity. Consequently, the free surface is tracked by following the free surface particles. Another advantage is that the convection term is not required in particle methods and, so, the numerical diffusion caused by it is avoided. Therefore, the difficulty in tracking free surface and the numerical diffusion due to the convection term in Eulerian form mesh-based method are overcome. Due to these facts, the particle methods are widely chosen to simulate the breaking waves and their interactions with structures. Yet, for some cases with the inflow and outflow, the particles initially located near the inlet or outlet of the domain may move away. If no special treatment is introduced, the particle methods may fail for such problems.

To overcome the numerical difficulties regarding inflow and outflow problems, Yoon et al (1999, 2001) proposed an Arbitrary Lagrangian-Eulerian (ALE) method, namely, MPS with a Meshless-Advection using Flow-direction Local-grid (MPS-MAFL). The method consists of two phases: the Lagrangian and the Eulerian phases. For the Lagrangian phase, the particle interaction model of the MPS method was applied to the differential operators and the moving interface was traced through the Lagrangian motion of computational points using MPS method; while, for the Eulerian phase, a high-order finite difference scheme (MAFL) was utilized to deal with the convection of fluid.

Many meshless methods have been developed and reported in the public, in which SPH, MPS and MLPG are most widely employed to simulate breaking waves and their interactions

with structures. The following reviews mainly emphasize on these three meshless methods.

2.2.1 Smooth Particle Hydrodynamic method (SPH)

Gingold & Monaghan (1977) and Lucy (1977) initially developed SPH for simulation of astrophysics problems. Their breakthrough was a method for the calculation of derivatives that did not require a structured computational mesh. In the SPH method, the spatial discretisation of state variables is provided by a set of points; SPH uses a kernel interpolation to approximate the field variables at any point in a domain. For example, an arbitrary function *f(x)* is approximated in a continuous form by an integral of the product of the function and a kernel function *W(x,h)* as follows:

$$\langle f(x)\rangle = \int f(x')W(x-x',h)dx' \tag{2.2.1}$$

where the angle brackets $\langle\rangle$ denote a kernel approximation, h is the smoothing length; x and x' are the position vectors, respectively. The numerical equivalent to Eq. (2.2.1) can be obtained by approximating the integral

$$\langle f(x)\rangle = \sum_{j=1}^{N} f(x_j)\frac{m_j}{\rho_j}W(x-x_j,h) \tag{2.2.2}$$

where N is the number of total neighbour particles of a particle located at x, ρ_j and m_j are the density and mass of particle j located at x_j, respectively. Furthermore, the spatial derivative such as the gradient and divergence can be similarly evaluated by

$$\nabla f(x) = \sum_{j=1}^{N} f(x_j)\frac{m_j}{\rho_j}\nabla W(x-x_j,h) \tag{2.2.3}$$

where $\nabla W(x-x_j,h)$ denotes the kernel gradient. One may notice that the gradient of a scalar field is only a function of the kernel gradient which is analytically known. The following symmetric gradient form with higher accuracy has been widely used (Monaghan, 1992; Liu & Liu, 2003; Liu & Liu, 2006):

$$\nabla f(x) = \rho\sum_{j=1}^{N} m_j\left(\frac{f(x)}{\rho^2}+\frac{f(x_j)}{\rho_j^2}\right)\nabla W(x-x_j,h) \tag{2.2.4}$$

Using the afore-mentioned function approximation techniques, it is possible to derive SPH

formulations for partial differential equations governing the fluid flows. The main applications of the SPH are summarized below.

Libersky et al (1993) and Randles & Libersky (1996) extended this method to study the solid mechanics problems. Monaghan (1994) extended it to simulate complicated free surface flows including the water wave propagation on beach, followed by Dalrymple & Rogers (2006) and others, Tulin & Landrini (2000) investigated plunging breakers. Solid bodies impacting on the water was modelled by Monaghan et al., (2003) and dam breaking simulations by Monaghan (1994) and Violeau & Issa (2007). In the original application of SPH in water wave, the fluid was assumed to be weakly compressible and an equation of state was introduced to calculate the pressure. Later, an incompressible SPH model has been put forward by Shao and Lo (2003), in which the pressure was calculated through a pressure Poisson equation derived from combinations of the mass and momentum equations. The turbulence model was considered in their model to simulate the breaking and overtopping waves. In Shao (2006), the widely used two-equation k-ε model was chosen as the turbulence model to be coupled with the incompressible SPH scheme. Shao (2006) reproduced cnoidal wave breaking on a slope under two different conditions: spilling and plunging. Good agreements were obtained between the numerical results and the experimental data. In Violeau & Issa (2007), a review of recently developed turbulent models adapted to the SPH method was presented, from a one-equation model involving mixing length to more sophisticated (and thus realistic) models like explicit algebraic Reynolds stress models (EARSM) or large eddy simulation (LES). The authors successfully applied mixing length and k–ε models to a turbulent free-surface channel. A 3D large eddy simulation (LES) model was also applied to the collapse of a water column in a tank. A sub-particle scaling technique using the Large Eddy Simulation method (LES) approach was introduced by Dalrymple & Rogers (2006) into SPH viscosity formulations to model breaking waves on beaches in two and three dimensions, green water overtopping of decks, and wave–structure interaction.

Although the SPH methods have been widely used in many different fields, there are some issues in the development of the methods. For example, the problem of free surface particle judgment (the discuss will be given in Section 2.5); the problem of big computational burden;

just as Dalrymple et al. (2010) and Issa et al. (2010) pointed out that "the requirement of high resolution in SPH needs very small time steps (O (10^{-5} s))" and since SPH is very time consuming, a massively parallel computing is required to do meaningful problems.

2.2.2 Moving Particle Semi-implicit method (MPS)

The MPS method was developed by Koshizuka, Tamako and Oka (1995) and Koshizuka, and Oka (1996). In this method, a particle interacts with others in its vicinity modelled by a weight function. The gradient and Laplacian operator in the Navier-Stokes equations are replaced by the particle interaction models, which are given below based on the particle *i* and its neighbouring particle *j*, whose coordinates are r_i and r_j, respectively:

$$\langle \nabla f \rangle_i = \frac{D_s}{n^0} \sum_{j \neq i} [\frac{f_j - f_i}{| r_j - r_i |^2} (r_j - r_i) W(| r_j - r_i |) \quad (2.2.6)$$

$$\langle \nabla^2 f \rangle_i = \frac{2D_s}{\lambda} \sum_{j \neq i} (f_j - f_i) W(| r_j - r_i |) \quad (2.2.7)$$

where *f* is a scalar quantity, *n* is the particle number density, which was proposed (Koshizuka, 1996) to approximate the fluid density and was defined as $n_i = \sum_{j \neq i} W(| r_j - r_i |)$, n^0 is the initial particle number density, D_s is the degree of the spatial dimensions (D_s =*2* for two-dimensional cases and D_s =*3* for three-dimensional cases), λ is a parameter to adjust a distributed quantity to the analytical results and given by $\lambda = \sum_{j \neq i} | r_j - r_i |^2 W(r_j - r_i)$. *W(r)* is a weight function; which is commonly given

$$W(r) = \begin{cases} \frac{r_0}{r} - 1, & r \leq r_0 \\ 0, & r > r_0 \end{cases} \quad (2.2.8)$$

where r_0 is the cut-off radius.

A semi-implicit time splitting scheme is adopted by the MPS method. Firstly, the velocities and locations of water particles at intermediate time step are predicted explicitly using the known variables at the previous time step; secondly, the pressure is solved implicitly based on the variables at the intermediate time step; thirdly, the velocities and coordinates of particles

at new time step are updated.

In the paper of Koshizuka and Oka (1996), the water column collapse was modelled and a acceptable agreement between the experimental data and numerical results was obtained. Since then, many researchers have applied the MPS method to deal with different problems. For example, Koshizuka, Nobe and Oka (1998) simulated the breaking wave over a slope. Two kinds of breaking waves, plunging and spilling breaker, were observed in the numerical results. Gotoh and Sakai (1999) simulated breaking waves over different seabed geometry, i.e. a uniform slope, a permeable uniform slope and a vertical wall with small step. Yoon, Koshizuka and Oka (1999) predicted the sloshing problems. This method has also been applied to resolve multiphase flow. Koshizuka, Ikeda and Oka (1999) analyzed the fragmentation process of a melt droplet in vapour explosions. Nonura, Koshizuka, Oka and Obata (2001) successfully simulated the droplet breakup behaviour. Yoon, Koshizuka & Oka (2001) calculated the bubble growth, departure and rise in nucleate pool boiling. There are also some applications in the interaction between violent free surface flow and structures. For instance, Naito and Sueyoshi (2003) simulated the shallow water sloshing, free rolling of floating bodies and motions of floating bodies in waves. The sub-particle-scale turbulence model, the solid-liquid and liquid-gas two phase flow models and the floating–bodies model were developed by Gotoh and Sakai (2006). In order to further increase the calculation efficiency, Naito and Sueyoshi (2003) proposed a method to simulate a large field with a restricted small calculation domain. Two effective technologies were presented. The first one was the wave absorption with the sidewall of calculation; the second one was an actuated bottom to simulate the infinite depth of water. Sueyoshi and Naito (2004b) calculated the 3D simulation of nonlinear fluid problems over one million particles with parallel computing on PC cluster. More applications can be found in Guo and Tao (2003), Zhang, Morita, Fukuda and Shirakawa (2005), Wang, Zheng et al. (2005), Xie, Koshizuka and Oka (2007).

In spite of the successful applications of the MPS method, there are still some numerical technologies need to be carefully investigated in order to get a stable solution when using the method. The first problem is related to the gradient operator in Eq. (2.2.6), which may cause some numerical instability as discussed by Koshizuka & Oka (1996) and Koshizuka, Nobe & Oka (1998). The detailed discusses will be presented in section 2.3. Another problem is

numerical implementations of solid boundary condition. Many researchers have made some improvements of numerical implementations of solid boundary conditions. The details will be given in section 2.4. The third problem is from the simple judgment rule for free surface particles. There is a special section about the free surface judgment technique given in section 2.5. In addition to these, the pressure is found by directly applying the Eq. (2.2.7), which is rough approximation to Laplacian operator.

Another problem is that the pressure solved by MPS method may have high frequency spurious fluctuations. To mitigate the pressure fluctuations, Sueyoshi and Naito (2002) used an averaged method; Sueyoshi and Naito (2004a) introduced an auxiliary computational procedure to reduce the oscillation; Hibi & Yabushita (2004) and Zhang, et al. (2006) employed more particles lays as solid boundary to obtain smooth pressure distribution.

2.2.3 Meshless Local Petro-Galerkin method (MLPG)

The MLPG method was proposed by Atluri & Zhu (1998) and was first extended to simulate the nonlinear water waves by Ma (2005a). In the MLPG, the unknown function f is approximated by a set of discretised variables and can be written as

$$f(x) = \sum_{j=1}^{N} \Phi_j(x) f(x_j) \tag{2.2.9}$$

where N is the number of total particles that affect particle x, $\Phi_j(x)$ is interpolation function called shape functions, which is formed by using moving least square (MLS) method (Atluri & Zhu, 1998; Ma, 2005a, b, 2007; Han & Atluri, 2004a, 2004b; Han et al 2006; Li, Shen, Han & Atluri, 2003). The derivative of an unknown function was found by direct differentiation of the shape function. The MLPG method is developed and based on a local symmetric weak form. For each particle, a local sub-domain is specified, which is a circle for two-dimensional and a sphere for three-dimensional cases. An equivalent weak form of governing equation is integrated over a local sub-domain. Through imposing the essential boundary conditions, the field variables (e.g. velocity, coordinate and etc) over the whole computational domain are solved by using a time marching procedure.

Due to the flexible choose of different trial and test functions, many variants of MLPG

methods have been developed by Atluri & Shen (2002). In that paper, six different MLPG methods have been compared numerically and the method using the Heaviside step function as the test function seems to be more promising than others. Many mixed numerical methods together with the MLPG method have also been developed to simplify the meshless implementation and to improve the efficiency. Atluri et al. (2004) proposed the so-called MLPG "mixed" finite volume method, in which both the displacements and displacement gradients were interpolated using the identical shape functions, independently. As a result, the continuity requirement on the trial function was reduced by one order, and the complex second derivatives of the shape function were avoided. Liu, Han & Atluri (2006b) developed a MLPG mixed collocation method for solving elasticity problems, in which the Dirac delta function was adopted as the test function, and therefore the equations were established at particles only. The traction boundary conditions were imposed by a penalty method, and the displacement boundary conditions were directly applied to the equations by the standard collocation approach. The MLPG mixed collocation method had achieved a great success, since it had a stable convergence rate, and higher efficiency than the MLPG finite volume method. Atluri, Liu & Han (2006) presented the MLPG mixed the finite difference method (FDM) to solve the solid mechanics problems, in which the displacements, displacement gradients, and stresses were interpolated independently using identical MLS shape functions. The MLPG mixed finite difference method successfully solved various elasticity problems with complex displacement and stress solutions.

In addition to its various applications to solid and general fluid problems, Ma (2005a) first extended the MLPG method to deal with nonlinear water wave problem and produced many encouraging results. In that paper, the simple Heaviside step function was adopted as the test function to formulate the weak form over local sub-domain, resulting in one in terms of pressure gradient. Following Ma (2005a), Ma (2005b) further developed a new form of the MLPG method called as a MLPG_R method, which is more suitable for modelling the nonlinear water waves. In MLPG_R method, the solution for Rankine sources rather than the Heaviside step function was taken as the test function. Based on this test function, a weak form of governing equations was derived, which did not contain any gradients of unknown functions and therefore made numerical discretisation of the governing equation relatively

easier and more efficient. A semi-analytical technique was also developed to evaluate the domain integral involved in this method, which dramatically reduce the CPU time spent on the numerical evaluation of the integral. Numerical tests showed that the MLPG_R method could be twice as fast as the MLPG method for modelling nonlinear water wave problems. The MLPG_R method has been successfully simulated the 2D freak waves, sloshing wave and various nonlinear water waves in Ma (2007).

Ma (2008) made another step forward in the development of the MLPG_R method for water waves. In that paper, a new meshless interpolation was suggested, which is as accurate as the moving least square method but much more efficient, particularly for computation of gradient of unknown functions. The method has been applied to solve 2D problem without wave breaking.

2.3 Numerical methods for calculation pressure gradient in meshless methods

In the numerical procedure of meshless methods, how to estimate the pressure and its gradient in terms of discrete value of pressure at nodes is an important step, which to some extent affects the efficiency and precision of the method. Atluri & Shen (2002) has reviewed the various meshless interpolation schemes. They are the Shepard function (SF), the partition of unity (PU), the reproducing kernel particle interpolation method (RKPM), radial basis function (RBF) and moving least square (MLS) method. Based on Atluri & Shen (2002) and Ma (2008), the Shepard function is the simplest method, which is similar to that used in the MPS method and has low order precision; the PU method is more computational expensive; The RKPM is equivalent to the MLS method if the basis and weight function used in them are the same; the RBF method has lower order accuracy than MLS if use the same number of nodes or need more nodes to achieve the same accuracy. Although the MLS method is widely used, it requires inverse of matrix or solution of a linear algebraic system and so is also quite time-consuming.

As mentioned above, the gradient operator Eq. (2.2.6) is used in the MPS method because of simplicity. However, the Eq. (2.2.6) may cause some numerical instability as discussed by Koshizuka & Oka (1996) and Koshizuka, Nobe & Oka (1998). A new pressure gradient form was proposed by Koshizuka et al. (1996 and 1998) to overcome the problem by

replacing f_i in Eq. (2.2.6) with the minimum value of *f* among the neighbouring particles satisfying $W(|r_j - r_i|) \neq 0$. Applying that equation to pressure it follows

$$\langle \nabla p \rangle_i = \frac{D_s}{n^0} \sum_{j \neq i} [\frac{p_j - p_i'}{|r_j - r_i|^2}(r_j - r_i) W(|r_j - r_i|) \tag{2.3.1}$$

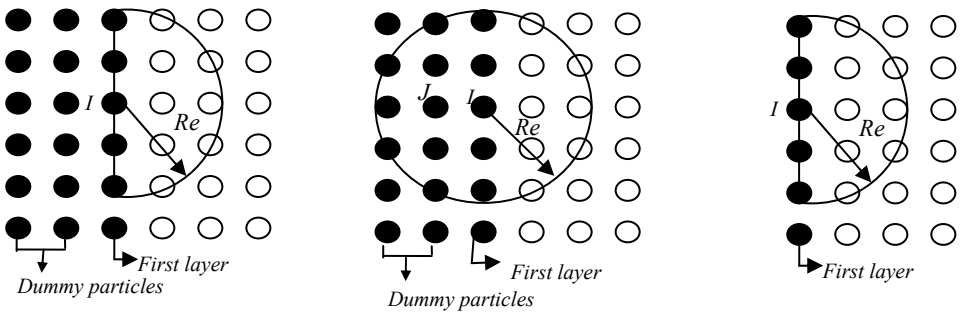
where $p_i' = \min(p_j)$, p is the pressure of nodes. The treatment improved the stability of the MPS method by ensuring that "the forces between particles are always repulsive because $p_j - p_i'$ is positive" (Koshizuka et al., 1998). However, this improvement is achieved by the sacrifice of conservation of momentum (Khayyer & Gotoh, 2009). From Eq. (2.3.1), one can notice that the force exerting on particle *j* by particle *i* due to the pressure gradient, will not equal to the force on particle *i* owing to particle *j* because of using the minimum value of pressure among the neighbouring particles rather than the pressure of particle *i*. To overcome the problem, Khayyer & Gotoh (2009) proposed an anti-symmetric equation for pressure gradient term given as follows

$$\langle \nabla p \rangle_i = \frac{D_s}{n^0} \sum_{j \neq i} [\frac{(p_i + p_j) - (p_i' + p_j')}{|r_j - r_i|^2}(r_j - r_i) W(|r_j - r_i|) \tag{2.3.2}$$

Khayyer & Gotoh (2009) made a simple numerical test for the evolution of an elliptical water drop using the Eqs. (2.2.6) and (2.3.2), and found that using Eq. (2.3.2), the conservation of momentum had been significantly improved. Ma (2008) also made some numerical investigations about the gradient formulation of Eq. (2.2.6) and found that the formulation was accurate for any distribution of nodes only if the function was a constant or accurate for a linear function only when the all nodes lied on intersection points of rectangular grid and the grid is square, which were hard to be met in practical problem simulations. Ma (2008) further presented a simplified finite difference interpolation (SFDI) scheme for meshless methods. This scheme is derived by using the Taylor series ignoring the terms of second and higher order derivatives. Therefore it has second order accuracy. The details will be given in Chapter 3.

2.4 Numerical methods for implementation of solid boundary condition in meshless methods

One of key problems in meshless methods is how to implement the solid boundary condition which strongly affects the accuracy convergence of the simulation of the nonlinear waves, particularly their interaction with solid boundaries (e.g. the wavemaker and the floating body). To do so, Koshizuka and Oka (1996) developed a method, which is used together with the MPS method. This method is referred as BC1 for brevity. In BC1, the solid boundary is modelled as several layers of particles (referred as wall particles) but only the particles in the first layer, which is the closest to the fluid, are involved in solving the boundary value problem (BVP) for the pressure. The wall particles in other layers are called as dummy particles. The same formulation for the pressure at the wall particles of the first layer as that for the fluid particles is used. The only contribution of the dummy particles is to calculate the particle number density in case the wall particle in the first layer would be judged as free surface particle because of small particle number density. So the dummy particles will not be considered as neighbour particle when solving the pressure governing equation. Therefore, the influence domain for these particles lies in one side of wall particles. This may reduce the overall accuracy. Fig 2.4.1a shows that all neighbour particles are distributed only on the right hand side of the wall particle *I*. The fact implies that the approximation to the pressure equation at the wall particle is of lower order accuracy than at the inner particles. A simple analysis can show that the BC1 can not generally ensure the pressure to satisfy the solid boundary condition. How good the approximation to the condition depends not only on the number of particles but also on how they are distributed. For example, when all the water particles are in line with their corresponding wall particles (Fig 2.4.1a) and the horizontal velocity of water particles adjacent to the wall is equal to the wall velocity, the solid boundary condition can be satisfied in the BC1 while it may not be in all other cases. In the problems about water waves, water particles are continuously moving and as a result, they may be sometimes in line with the wall particles but sometimes not. This implies that approximation to the solid boundary condition is sometimes better but sometimes worse and therefore possibly leads to spurious wiggles in the pressure time history.



(a) (b) (c)

Fig. 2.4.1 Illustration of the influence domain for wall particles in (a) BC1, (b) BC2 and (c) BC3 (solid circle: wall particles; hollow circle: water particles)

To overcome the problem, Hibi & Yabushita (2004) and Zhang, Morita, Fukuda & Shirakawa (2006) suggested another method (referred as BC2). In the BC2, all the wall particles including outer layers are considered in solving the BVP for the pressure. In this implementation, the neighbouring particles of the wall particles in the first layer are distributed on both sides of the layer and therefore enhance the accuracy to some extent. When the pressure equation is applied to the first layer wall particles, the neighbours of the wall particles do not only include those on their right (water particles) but also those on their left (wall particles in other two layers as shown in Fig 2.4.1b). However, after doing so, the number of unknown pressure values is larger than the number of discretised equations. In order to obtain sufficient number equations, the pressure at wall particles in other two layers is related to the pressure at wall particles by assuming that the pressure is linearly distributed, e.g., the pressure at Particle *J* in Fig 2.4.1b is expressed as

$$p_J = p_I - (\frac{\partial p}{\partial x})_I (x_J - x_I) \tag{2.4.1}$$

where $\left(\frac{\partial p}{\partial x}\right)_I = -\rho \dot{U}_x$, $\dot{U}_x$ is the acceleration of wall with respect to x direction. It is noted that although the approximation to the pressure equation in the BC2 may be more accurate than that in the BC1, the problem about variation in the degree of approximation to the solid boundary condition still remain. That is, the spurious wiggles may also appear.

In both BC1 and BC2, the physical solid boundary condition is only approximately satisfied and sometimes wiggles in the time history of pressure are observed (Hibi & Yabushita 2004 and Sueyoshi & Naito, 2004a). Ma (2005a, b) employed another method to

implement the boundary condition, in which the pressure gradient for the wall particles in the first layer is forced to satisfy the physical solid boundary condition. This is referred as BC3 (shown in Fig. 2.4.1c). In the BC3, the solid boundary condition of pressure is directly applied to the wall particles in the first layer. That is, the discretised equation for them is based on the solid boundary condition. It could be understood that the approximation to the solid boundary condition in the BC3 depends less on the distribution of water particles and yields much smoother pressure.

Apart from these, Monaghan (1994) applied artificial force acting on the fluid particles near the solid boundaries. Lo and Shao (2002) suggested to use a mirror technique. But these methods mainly ensure the velocity to satisfy the solid boundary condition. Atluri (1998) solved the boundary value problems with a penalty method, the penalty method works well for fixed (or with little movement) boundary problems. As pointed out by Ma (2005a), the penalty method may cause unaccepted errors, particularly near the free surface.

2.5 Numerical methods for identifying the free surface

Many surface tracking or capturing methods have been developed in mesh-based methods available to simulate the free surface, e.g., marker and cell (MAC, Harlow and Welch, 1965) method, volume of fluid (VOF, Hirt and Nichols, 1981) method, level set method (LSM, Osher and Sethian, 1988). MAC method is based on a Lagrangian approach and a set of massless marker particles whose position at any time step are used to reconstruct the interface. Accuracy requires a considerable amount of particles per grid cell, making the method computationally expensive, especially in 3D cases. VOF method is based on the Eulerian point of view instead. A discrete indicator (or color) function is used that corresponds to the cell volume occupied by fluid. The main issues include the difficulty in advecting a discontinuous indicator function and the accurate modeling of surface tension effects. An alternative to the indicator-function methods is the level set method (Sethian, 1996), which makes use of a function representing the distance to the liquid surface. Reconstruction of the free surface is conceptually simpler than with the VOF method. However, for violently moving free surfaces the level set function requires to be redefined regularly, and conservation of the amount of liquid cannot be guaranteed (Rider & Kothe, 1995).

Different from the aforementioned methods, meshless methods discretize the fluid by particles. Hence it is a more natural way to track the free surface by following the free surface particles. In different meshless methods, there are different techniques to do so. For example, in the SPH methods, the free surface particles are identified by the density; i.e., if the ratio of the density of a particle to the fluid density is less than a specified value, such as 1%, it is then identified as a free surface particle [e.g., Lo and Shao, (2002)]. Khayyer et al. (2008c) compared the violent free surfaces using Incompressible SPH (CISPH) and Corrected Incompressible SPH (CISPH), and found that obvious improvements have been achieved for the judgements of the free surface by using the CISPH method. However, one can note that from the compared photographs shown in that paper, the thickness of the free surface (normally the thickness is one particle layer) was still thick especially in the breaking and post-breaking regions, which indicated that the efficiency of the technique of judging free surface in SPH methods is low.

In MPS method, one simple technique is used based on the following parameter

$$\beta_I = n_I^* / n^0 \tag{2.5.1}$$

where n_I^* is the particle number density at particle *I* as defined in Section 2.2.2.

Due to the fact that no particle exists in the outer region of a free surface, the particle number density decreases on the free surface. So if $\beta_I < \beta$, then particle *I* will be judged as a free surface particle. Currently, there is no common agreement about how to specify the value of β. It is problem-dependent. Different researchers use different values. For example, it was 0.97 in Koshizuka and Oka (1996) and Gotoh and Sakai (2006) while it was 0.99 in Shao and Lo (2003). Results look to be promising in all the papers even with different values for β. Khayyer et al. (2007, 2008a and 2008b) compared the configuration of free surface and non-free surface particles in CMPS (Corrected MPS), CMPS-SBV (CMPS with Strain-Based Viscosity) and standard MPS (Koshizuka et al., 1996) using the simple rule. From the comparisons, it has concluded that there are much more improvements in terms of free particle particles judgement in CMPS-SBV and CMPS compared with the standard MPS. But in CMPS-SBV and CMPS, however there are still many incorrect judgements of free

surface particles. Some inner water particles were incorrectly judged as free surface particles, and many free surface particles were wrongly judged as inner water particles. So it is necessary to develop a more effective way to identify free surface particle.

3. MATHEMATICAL MODEL AND NUMERICAL METHOD

This chapter presents the governing equations, boundary conditions as well as the detailed the numerical procedures. The 2D and 3D MLPG_R formulations, including the numerical techniques for the domain integration, the gradient calculation scheme, are also presented.

Fig 3.0.1 shows the schematic of the computational domain and the coordinate system. The computational domain is chosen as a rectangle tank with a width of B. It may consist of two connected portions, the one with a flat seabed with a constant water depth of d and the other one with a sloping seabed with a sloping angle of θ. The lengths of these two portions are $L1$ and $L2$ respectively. The incident waves are generated by a wavemaker, which is mounted on the left side of the tank. A Cartesian coordinate system is used with the oxy plane on the mean free surface and with the z-axis being positive upwards.

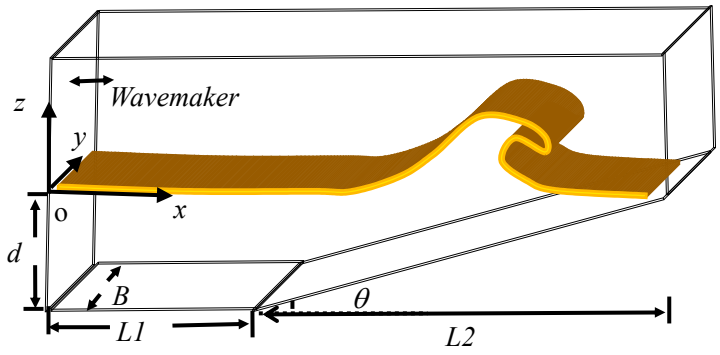


Fig 3.0.1 Schematic of computational domain and the coordinate

3.1 Governing equations and boundary conditions

The fluid is assumed to be incompressible and governed by the continuity and momentum equations as follows:

$$\nabla \cdot \vec{u} = 0 \qquad (3.1.1)$$

$$\frac{D\vec{u}}{Dt} = -\frac{1}{\rho}\nabla p + \vec{g} + \nu\nabla^2\vec{u} \qquad (3.1.2)$$

where $\frac{D}{Dt}$ is the substantial (or total time) derivative following fluid particles; $\vec{u}$ is the fluid velocity, ρ is the density of the fluid and p is the pressure, ν is the kinematic viscosity and g is the gravitational acceleration.

The Lagrangian forms of kinematic and dynamic conditions on the free surface ($z = \eta(x, y, t)$) are given as followed

$$\frac{D\vec{r}}{Dt} = \vec{u} \qquad (3.1.3)$$

and

$$p = 0 \qquad (3.1.4)$$

where $\vec{r} = x\vec{i} + y\vec{j} + z\vec{k}$ is the position vector related to the origin of the coordinate system. The relative atmospheric pressure in Eq. (3.1.4) has been taken as zero. On the rigid boundary, the following kinematic and dynamic boundary conditions are satisfied:

$$\vec{n}\cdot\vec{u} = \vec{n}\cdot\vec{U} \qquad (3.1.5)$$

and

$$\vec{n}.\nabla p = \rho\left(\vec{n}\cdot\vec{g} - \vec{n}\cdot\dot{\vec{U}} + \nu\,\vec{n}\cdot\nabla^2\vec{u}\right) \qquad (3.1.6)$$

where $\vec{n}$ is the inwards unit normal vector of the rigid boundary; $\vec{U}$ and $\dot{\vec{U}}$ are the velocity and the acceleration of the rigid boundary, respectively.

3.2 Numerical Procedure

The mathematical model formed by Eqs. (3.1.1.) ~ (3.1.6) is solved by a time splitted scheme similar to Ma (2005a, b). Suppose the velocity, pressure and the location at n^{th} time step are known, one can use the following numerical procedure to find them at $(n+1)^{th}$ time step.

(1) Calculate the intermediate velocity ($\vec{u}^*$) and position($\vec{r}^*$) of particles by using

$$\vec{u}^* = \vec{u}^n + \vec{g}\Delta t + \nu\nabla^2\vec{u}^n\Delta t \quad (3.2.1)$$

$$\vec{r}^* = \vec{r}^n + \vec{u}^*\Delta t \quad (3.2.2)$$

where the superscript * and *n* represent the intermediate step and n^{th} time step, respectively; Δt is the time step.

(2) Implicitly evaluate the pressure p^{n+1} using

$$\nabla^2 p^{n+1} = \alpha\frac{\rho^{n+1}-\rho^*}{\Delta t^2} + (1-\alpha)\frac{\rho}{\Delta t}\nabla\cdot\vec{u}^* \quad (3.2.3)$$

where α is an artificial coefficient between 0 and 1, and ρ^{n+1} and ρ^* are the fluid densities at $(n+1)^{th}$ time step and at the intermediate time step, respectively. For the incompressible flow, the density should be a constant and $\rho^{n+1}=\rho$, where ρ is the density of fluid specified initially. The density ρ^* at the intermediate step may not be the same as ρ because the velocity and position calculated in Eqs. (3.2.1) and (3.2.2) do not necessarily satisfy the continuity equation given in Eq. (3.1.1).

(3) Calculate the fluid velocity and thus update the position of the particles using

$$\vec{u}^{**} = -\frac{\Delta t}{\rho}\nabla p^{n+1} \quad (3.2.4)$$

$$\vec{u}^{n+1} = u^* + \vec{u}^{**} = \vec{u}^* - \frac{\Delta t}{\rho}\nabla p^{n+1} \quad (3.2.5)$$

$$\vec{r}^{n+1} = \vec{r}^n + \vec{u}^{n+1}\Delta t \quad (3.2.6)$$

(4) Go to (1) for the next time step

As it can be seen, the key task of this procedure is to solve Eq. (3.2.3) in order to evaluate the pressure. Many numerical methods, for example, finite difference method and finite element

method, may be adopted to solve this equation. But in this study, the MLPG_R method will be adopted.

In the meshless method, the computational domain is discretised by many particles. At each of the particles, a sub-domain is specified, which is a circle for 2D and a sphere for 3D cases. Eq. (3.2.3), after multiplying by an arbitrary test function φ, is integrated over the sub-domain, leading to

$$\int_{\Omega_I} \varphi \nabla^2 p^{n+1} d\Omega_I = \int_{\Omega_I} [\alpha \frac{\rho^{n+1} - \rho^*}{\Delta t^2} + (1-\alpha) \frac{\rho}{\Delta t} \nabla \cdot \vec{u}^*] \varphi \, d\Omega_I \tag{3.2.7}$$

where Ω_I is the area (2D problems) or volume (3D problems) of the sub-domain centred at particle *I*. Test function φ can be chosen as different functions, which will lead to different formulations used in different meshless methods. For example, if φ is defined as a Heaviside step function or a Rankine Source function, Eq. (3.2.7) will be changed to the governing equations used in the MLPG or MLPG_R methods (Ma, 2005a, b; Ma & Zhou, 2009; Zhou & Ma, 2010); or if φ is defined as a δ function, Eq. (3.2.7) will be the same as those used in the MPS method (Ikeda, 1999; Zhang, et al, 2005). The MLPG_R formulations for 2D and 3D will be detailed in the following subsection. Due to the facts that Rankine source function has different forms in 2D and 3D problems and that the integration domains of particles are also different in 2D and 3D problems, the MLPG_R formulations for 2D and 3D will be separately presented below for clarity.

3.3 MLPG_R Formulation for 2D cases

In the MLPG_R method, the particles are separated into three groups: those located on the rigid boundary (referred to as wall particles), those on the free surface (referred to as free surface particles) and others (referred to as inner particles), which are illustrated in Fig. 3.3.1. For each inner particle *I*, Eq. (3.2.7) is applied. The test function is taken as the Rankine source solution, i.e., the function φ satisfies $\nabla^2 \varphi = 0$, in Ω_I excluding the center and $\varphi = 0$ on $\partial\Omega_I$ which is boundary of Ω_I. For 2D cases, the expression of the Rankine source solution is

$$\varphi = \frac{1}{2\pi}\ln(r / R_I) \tag{3.3.1}$$

where r is the distance between a concerned point and the center of Ω_I and R_I is the radius of Ω_I.

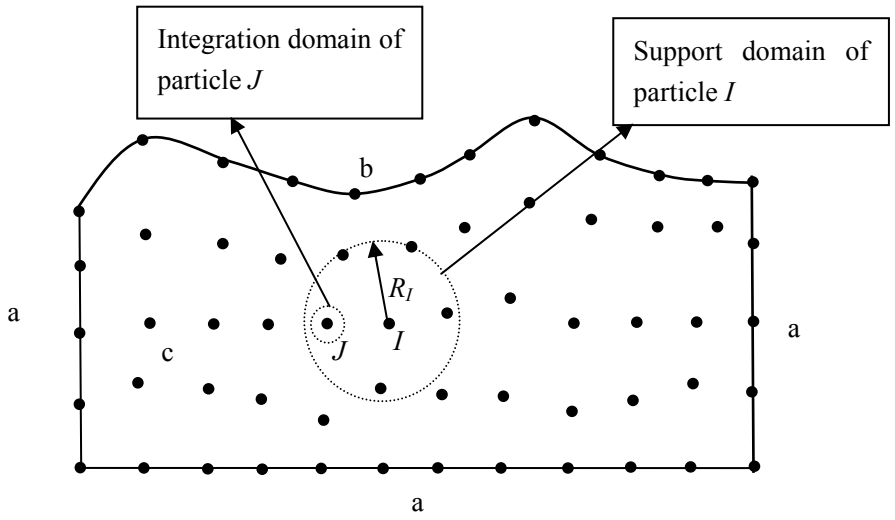


Fig. 3.3.1 Illustration of particles, integration domain and support domain

(a: wall particle; b: free surface particles and c: inner particles)

In Eq. (3.2.7), the second order derivative of unknown pressure and the gradient of the intermediate velocity are included. Numerical calculation of the derivative and gradient terms requires not only much computational time but also degrades the accuracy. In order to obtain a better form, Eq. (3.2.7) is changed, by adding a zero term $p\nabla^2\varphi$ and applying the Gauss's theorem, into

$$\begin{aligned}\int_{\partial\Omega_I+\partial\varepsilon}[\vec{n}\cdot(\varphi\nabla p)-\vec{n}\cdot(p\nabla\varphi)]dS &= \alpha\int_{\Omega_I}\frac{\rho-\rho^*}{\Delta t^2}\varphi d\Omega \\ &+(1-\alpha)[\int_{\partial\Omega_I+\partial\varepsilon}\frac{\rho}{\Delta t}\vec{n}\cdot(\varphi\vec{u}^*)dS - \int_{\Omega_I}\frac{\rho}{\Delta t}\vec{u}^*\cdot\nabla\varphi d\Omega]\end{aligned} \tag{3.3.2}$$

where $\partial\varepsilon$ is a small surface surrounding the center of Ω_I, which is a circle in 2D cases. The reason for adding $\partial\varepsilon$ is that the test function φ in Eq. (3.3.1) becomes infinite at r=0

and so the Gauss's theorem can not be used otherwise. ρ^{n+1} in Eq. (3.2.7) has been replaced by a function of ρ based on the discusses below Eq. (3.2.3). One can easily prove that taking $\varepsilon \to 0$ results in (Ma (2005b))

$$\int_{\partial\Omega_I+\partial\varepsilon} [\vec{n}\cdot(\varphi\nabla p)]dS = 0 \tag{3.3.3}$$

$$\int_{\partial\varepsilon} [\vec{n}\cdot(p\nabla\varphi)]dS = -p \tag{3.3.4}$$

$$\int_{\partial\Omega_I+\partial\varepsilon} \vec{n}\cdot(\varphi\vec{u}^*)dS = 0 \tag{3.3.5}$$

and

$$\int_{\Omega_I} \varphi d\Omega = -\frac{R_I^2}{4} \tag{3.3.6}$$

Using the results in Eqs. (3.3.3) ~ (3.3.6), Eq. (3.3.2) can be rewritten as:

$$\int_{\partial\Omega_I} \vec{n}\cdot(p\nabla\varphi)dS - p = \alpha\frac{\rho_I - \rho_I^*}{\Delta t^2}\frac{R_I^2}{4} + (1-\alpha)\int_{\Omega_I}\frac{\rho}{\Delta t}\vec{u}^*\cdot\nabla\varphi d\Omega \tag{3.3.7}$$

where it has been assumed that the increment of the density ($\rho - \rho^*$) is a constant within the sub-domain and so equal to its value at Particle *I*, which is acceptable not only because the density should not change much due to the change in the intermediate position of the particle but also because the small error caused due to the assumption is further reduced by multiplying the coefficient α that has a small value. The term may be evaluated in a more accurate way, for example by interpolation as done for the second term but such a way will not improve the accuracy significantly due to the reason.

It is obvious that Eq. (3.3.7) requires the intermediate density that is not computed in the MLPG_R method. Actually, the density can be replaced by a particle number density (PND) defined by Koshizuka and Oka (1996) in their MPS method as follows:

$$n_I = \sum_{j=1, j\neq I}^{N} W(|r_j - r_I|) \tag{3.3.8}$$

where W is a weight function in terms of the distance between Particle I and Particle j, which becomes zero when the distance is larger than a certain value. The domain, within which the weight function is not zero, is called support domain (see the example shown in Fig. 3.3.1). In the above equation, N is the total number of particles in the support domain of particle I. As indicated by Koshizuka and Oka (1996), the PND is related to the density by:

$$\rho_I = \frac{m_I n_I}{\int_{V_I} W(r) dV_I}, \tag{3.3.9}$$

where m_I is the mass of Particle I. The denominator of Eq. (3.3.9) is the integral of the weight function in the whole region V_I, excluding a central part occupied by Particle I, this integration is constant if the radius of the whole region V_I is fixed. After applying Eq. (3.3.9), Eq. (3.3.7) becomes

$$\int_{\partial\Omega_I} \vec{n} \cdot (p\nabla\varphi) dS - p = \alpha \frac{\rho}{\Delta t^2} \frac{n^0 - n_I^*}{n^0} \frac{R_I^2}{4} + (1-\alpha) \int_{\Omega_I} \frac{\rho}{\Delta t} \vec{u}^* \cdot \nabla\varphi d\Omega \tag{3.3.10}$$

where n^0 is the initial particle number density of fluid and n_I^* is the particle number density of Particle I at the intermediate step. Ma (2005b) has detailed the method to discretize Eq. (3.3.10), in which the pressure on the left hand side is interpolated by a moving least square method (MLS) and the integration of the second term on the right hand side is evaluated by a semi-analytical technique. For the purpose of completeness, the details will be simply explained as follows.

3.3.1 Discretization of the governing equation for pressure

The unknown function p needs to be approximated by a set of discretized variable. Herein the approximation may be written as

$$p(\vec{x}) \approx \sum_{J=1}^{N} \Phi_J(\vec{x}) \hat{p}_J \tag{3.3.11}$$

where N is the number of particles affecting the pressure at point $\vec{x}$; $\Phi_J(\vec{x})$ is interpolation function called shape function. $\hat{p}_J$ are particle variables but not necessarily equal to the

particle values of $p(\vec{x})$. Generally in meshless methods, a local approximation to the unknown function is assumed, which is expressed in terms of unknown variables corresponding to some randomly located particles nearby. This local approximation may be formulated in a variety of ways. One of them is to use a moving least-squre (MLS) method (Atluri and Shen, 2002), which is adopted in this study. With this method, the shape function is given by

$$\Phi_J(\vec{x}) = \sum_{m=1}^{M} \psi_m(\vec{x})[A^{-1}(\vec{x})B(\vec{x})]_{mJ} \tag{3.3.12}$$

With the base function being $\psi^T(\vec{x}) = [\psi_1, \psi_2, \psi_3] = [1, x, y]$ (M=3) for 2D cases, and with the matrixes $B(\vec{x})$ and $A(\vec{x})$ being defined as

$$B(\vec{x}) = \Psi^T W(\vec{x}) = [w_1(\vec{x} - \vec{x}_1)\psi(\vec{x}_1), w_2(\vec{x} - \vec{x}_2)\psi(\vec{x}_2), \cdots] \tag{3.3.13}$$

$$A(\vec{x}) = \Psi^T W(\vec{x})\Psi = B(\vec{x})\Psi \tag{3.3.14}$$

where $W(\vec{x})$ and Ψ are, respectively, expressed by

$$W(\vec{x}) = \begin{bmatrix} w_1(\vec{x} - \vec{x}_J) & 0 & \cdots & 0 \\ 0 & & & \\ \cdots & & & \\ 0 & & & w_N(\vec{x} - \vec{x}_J) \end{bmatrix} \tag{3.3.15}$$

and

$$\Psi^T = \left[\psi(\vec{x}_1, \vec{x}_2, \vec{x}_3, \cdots, \vec{x}_N)\right] \tag{3.3.16}$$

which shows each column of the matrix Ψ^T is the value of the base function ψ at a particular point. The weight function $w_J(\vec{x} - \vec{x}_J)$ may be chosen to be a spline function given by

$$w_J(\vec{x} - \vec{x}_J) = \begin{cases} 1 - 6(\frac{d_J}{r_J})^2 + 8(\frac{d_J}{r_J})^3 - 3(\frac{d_J}{r_J})^4, & 0 \le \frac{d_J}{r_J} \le 1 \\ 0 & , \frac{d_J}{r_J} > 1 \end{cases} \tag{3.3.17}$$

where r_J is the size of support domain of the weight function and $d_J = |\vec{x} - \vec{x}_J|$ is the distance between the particle *J* and point $\vec{x}$.

Using Eq. (3.3.11), Eq. (3.3.10) can be rewritten as

$$[K_{IJ}]\cdot[\hat{p}]=[F_I] \tag{3.3.18}$$

where

$$K_{IJ}=\int_{\partial\Omega_I}\Phi_J(\vec{x})\vec{n}\cdot\nabla\varphi dS-\Phi_J(\vec{x}) \tag{3.3.19}$$

$$F_I=\alpha\frac{\rho}{\Delta t^2}\frac{n^0-n^*}{n^0}\frac{R_I^2}{4}+(1-\alpha)\int_{\Omega_I}\frac{\rho}{\Delta t}\vec{u}^{(*)}\cdot\nabla\varphi d\Omega \tag{3.3.20}$$

in which Particle I are inner particles; and Particle J are those influencing Particle I, determined by the weight function. Using test function Eq. (3.3.1), the boundary integral in Eqs.(3.3.19) can be simplified as

$$\int_{\partial\Omega_I}\Phi_J(\vec{x})\vec{n}\cdot\nabla\varphi dS=\frac{1}{2\pi R_I}\int_{\partial\Omega_I}\Phi_J(\vec{x})dS \tag{3.3.21}$$

The integral over the domain in Eq. (3.3.20) can also be simplified as

$$\int_{\Omega_I}\frac{\rho}{\Delta t}\vec{u}^*\cdot\nabla\varphi d\Omega=\frac{\rho}{2\pi\Delta t}\int_0^{2\pi}\int_0^{R_I}u_r^*drd\theta \tag{3.3.22}$$

where u_r^* is the radial component of u^*.

3.3.2 Numerical Technique for Domain Integration

As shown in Eq. (3.3.22), the domain integration must be numerically evaluated, usually by using the Gaussian quadrature. To do so, more than 16 Gaussian points for 2D cases may be required to obtain satisfactory results, at which the intermediate velocities are estimated by employing the MLS method. Evaluation of the velocities at so many points is time-consuming. In order to make the method more efficient, the following semi-analytical technique is suggested:

1. Dividing an integration domain into several sub-domains;
2. Assuming intermediate velocities to linearly vary over each sub-domain;
3. Performing the integration over each sub-domain analytically.

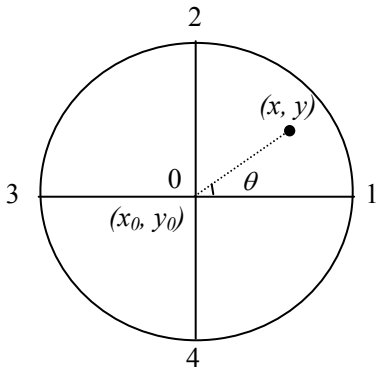


Fig. 3.3.2 Illustration of division of an integration domain

Let us consider a circular integration domain with a radius of R_I, centred at (x_0, y_0), as shown in Fig. 3.3.2 and divide it into four sub-domains, 0-1-2, 0-2-3, 0-3-4 and 0-4-1. Over each sub-domain, e.g., 0-1-2, the intermediate velocity components at arbitrary point (x, y) are assumed to be linear with respect to coordinates and given by

$$u^{(*)} = u_0^{(*)} + c_{ux}(x - x_0)/R_I + c_{uy}(y - y_0)/R_I \tag{3.3.23a}$$

$$v^{(*)} = v_0^{(*)} + c_{vx}(x - x_0)/R_I + c_{vy}(y - y_0)/R_I \tag{3.3.23b}$$

where $(u^{(*)}, v^{(*)})$ are the intermediate velocity components at any points (x, y) in the subdomain 0-1-2; and $(u_0^{(*)}, v_0^{(*)})$ are those at its centre. c_{ux}, c_{uy}, c_{vx} and c_{vy} are constants, which are determined in such a way that the velocity components equal to those at point 1 and 2. Taking the x-component as an example, one should have

$$c_{ux}(x_1 - x_0) + c_{uy}(y_1 - y_0) = (u_1^{(*)} - u_0^{(*)})R_I \tag{3.3.24a}$$

$$c_{vx}(x_1 - x_0) + c_{vy}(y_1 - y_0) = (v_1^{(*)} - v_0^{(*)})R_I \tag{3.3.24b}$$

which gives

$$c_{ux} = (u_1^{(*)} - u_0^{(*)}) \tag{3.3.25a}$$

$$c_{uy} = (u_2^{(*)} - u_0^{(*)}) \tag{3.3.25b}$$

c_{vx} and c_{vy} can be found similarly or obtained by replacing $(u_1^{(*)}, u_2^{(*)})$ with $(v_1^{(*)}, v_2^{(*)})$ in Eq. (3.3.25); and thus the velocities in this sub-domain are determined by Eq. (3.3.23). The velocities in other sub-domains can also be estimated in this way. The only difference is that

they may be related to the velocities at points 3 and 4, depending on which sub-domain is concerned. Consequently, the velocities at any points in the circle are determined by those at only five points (0, 1, 2, 3, and 4). Based on this, the integration in Eq. (3.3.22) over the circle can be rewritten as:

$$\int_0^{2\pi}\int_0^{R_I} u_r^{(*)} dr d\theta = \sum_{i=1}^{N_i}\int_{\vartheta_i}^{\vartheta_{i+1}}\int_0^{R_I} u_r^{(*)}(r,\theta) dr d\theta \tag{3.3.26}$$

(N_i=4 for four divisions and $\vartheta_5 = \vartheta_1$)

The integration over each sub-domain can be evaluated analytically using Eq. (3.3.23). For this purpose, Eq. (3.3.23) can be rewritten as

$$u^{(*)} = u_0^{(*)} + c_{ux}\vec{r}\cos\theta + c_{uy}\vec{r}\sin\theta \tag{3.3.27a}$$

$$v^{(*)} = v_0^{(*)} + c_{vx}\vec{r}\cos\theta + c_{vy}\vec{r}\sin\theta \tag{3.3.27b}$$

where the transformation of $(x - x_0)/R = \vec{r}\cos\theta$ and $(y - y_0)/R = \vec{r}\sin\theta$ have been employed. The relationship between the $u_r^{(*)}$ and $u^{(*)}$, $v^{(*)}$ is

$$u_r^{(*)} = u^{(*)}\cos\theta + v^{(*)}\sin\theta \tag{3.3.28}$$

Substituting Eq. (3.3.27) into Eq. (3.3.28) yields

$$u_r^{(*)} = u_0{}^{(*)}\cos\theta + v_0{}^{(*)}\sin\theta + c_{ux}\vec{r}\cos^2\theta + c_{uy}\vec{r}\cos\theta\sin\theta + c_{vx}\vec{r}\cos\theta\sin\theta + c_{vy}\vec{r}\sin^2\theta \tag{3.3.29}$$

It is easy to show that the integration of $u_0{}^{(*)}\cos\theta + v_0{}^{(*)}\sin\theta$ over whole circle becomes zero and can be omitted. The integration of other terms in Eq. (3.3.29) over sub-domain 0-1-2 is given by

$$\int_{\vartheta_1}^{\vartheta_2}\int_0^{R_I} u_r^{(*)}(r,\theta) dr d\theta = \frac{1}{4}R_I\begin{bmatrix}(c_{ux}+c_{vy})(\vartheta_2-\vartheta_1)\\ +(c_{ux}-c_{vy})(\sin\vartheta_2\cos\vartheta_2 - \sin\vartheta_1\cos\vartheta_1)\\ +(c_{uy}+c_{vx})(\sin^2\vartheta_2 - \sin^2\vartheta_1)\end{bmatrix} \tag{3.3.30}$$

Substituting the values of c_{ux}, c_{uy}, c_{vx} and c_{vy} and $\vartheta_1 = 0; \vartheta_2 = \pi/2$ into Eq. (3.3.30), it can be rewritten as

$$\int_0^{\pi/2}\int_0^{R_I} u_r^{(*)}(r,\theta) dr d\theta = \frac{1}{4}R_I\{(u_1^{(*)} - u_0^{(*)} + v_2^{(*)} - v_0^{(*)})\pi/2 + (u_2^{(*)} - u_0^{(*)} + v_1^{(*)} - v_0^{(*)})\} \tag{3.3.31}$$

Similar results can be obtained for other sub-domains and the sum of these gives the results for the whole domain. The finial integration over sub-domain 0-2-3 can be expressed as

follows,

$$\int_{\pi/2}^{\pi}\int_{0}^{R_I} u_r^{(*)}(r,\theta)drd\theta = \frac{1}{4}R_I\{(u_0^{(*)} - u_3^{(*)} + v_2^{(*)} - v_0^{(*)})\pi/2 - (u_2^{(*)} - u_0^{(*)} - v_3^{(*)} + v_0^{(*)})\} \tag{3.3.32}$$

The finial integration over sub-domain 0-3-4 can be expressed as follows,

$$\int_{\pi}^{3\pi/2}\int_{0}^{R_I} u_r^{(*)}(r,\theta)drd\theta = \frac{1}{4}R_I\{-(u_3^{(*)} - u_0^{(*)} + v_4^{(*)} - v_0^{(*)})\pi/2 - (u_4^{(*)} - u_0^{(*)} + v_3^{(*)} - v_0^{(*)})\} \tag{3.3.33}$$

The finial integration over sub-domain 0-4-1 can be expressed as follows,

$$\int_{3\pi/2}^{2\pi}\int_{0}^{R_I} u_r^{(*)}(r,\theta)drd\theta = \frac{1}{4}R_I\{(u_1^{(*)} - u_0^{(*)} - v_4^{(*)} + v_0^{(*)})\pi/2 - (-u_4^{(*)} + u_0^{(*)} + v_1^{(*)} - v_0^{(*)})\} \tag{3.3.34}$$

By adding the Eqs. (3.3.31) ~ (3.3.34) together, the final expression for Eq. (3.3.26) over the full circle can be obtained as follows,

$$\int_{0}^{2\pi}\int_{0}^{R_I} u_r^{(*)}(r,\theta)drd\theta = \frac{\pi}{4}R_I(u_1^{(*)} + v_2^{(*)} - u_3^{(*)} - v_4^{(*)}) \tag{3.3.35}$$

It can be seen that the integral over full circle is determined by the velocity only at four points rather than at least 16 Gaussian points if the Gaussian quadrature would be employed. Although the results depend on the number of sub-domains, numerical tests show that four sub-domains are good enough for 2D cases. It is noted that the above derivation is similar to that given in Ma (2005b), but the finial expression Eq. (3.3.35) is more compact.

3.3.3 Numerical implementation of solid boundary condition

In MLPG_R method, the solid boundaries are also discretized into many particles, but only one layer is used to represent the solid boundary (shown in Fig. 3.3.3).

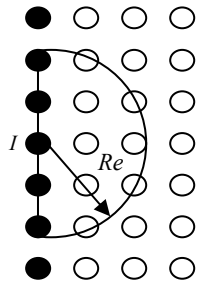


Fig. 3.3.3 Illustration of the influence domain for wall particles
(solid circle: wall particles; hollow circle: water particles)

The solid boundary condition (Ma 2005a, b), without considering the viscosity, is given as

$$\vec{n}.\nabla p = \rho\left(\vec{n}\cdot\vec{g} - \vec{n}\cdot\dot{\vec{U}}\right) \tag{3.3.36}$$

where $\dot{\vec{U}}$ is the acceleration of the solid boundary. This expression can be derived directly from Eqs. (3.1.2) and (3.1.6) by taking $\nu = 0$. When $\nu \neq 0$ as in the cases considered in this study, one may also derive the condition by using Eqs. (3.1.2) and (3.1.6) and obtain:

$$\vec{n}.\nabla p = \rho\left(\vec{n}\cdot\vec{g} - \vec{n}\cdot\dot{\vec{U}} + \nu\,\vec{n}\cdot\nabla^2\vec{u}\right). \tag{3.3.37}$$

It is obvious that one must compute the term $\nabla^2\vec{u}$ when applying this condition, which needs to estimate the second order derivative at the rigid boundary. Although there is no much difficulty to do so theoretically, the error associated with it is not easy to be suppressed in computational practice as the fluid particles located only on one side of the boundary. Therefore, it is better to avoid the computation of the second order derivative when possible. For this reason, Eqs. (3.1.5) and (3.2.5) are combined to give an alternative equation for the boundary condition as follows:

$$\vec{n}\cdot\nabla p^{n+1} = \frac{\rho}{\Delta t}\vec{n}\cdot(\vec{u}^{*} - \vec{U}^{n+1}) \tag{3.3.38}$$

where $\vec{u}^{*}$ is computed by Eq. (3.2.1) with $\vec{u}^{n}$ taken as $\vec{U}^{n}$. If the solid boundary is fixed, $\vec{n}\cdot\vec{U}^{n+1} = 0$ and so Eq. (3.3.38) can be rewritten as

$$\vec{n}\cdot\nabla p^{n+1} = \frac{\rho}{\Delta t}\vec{n}\cdot\vec{u}^{*} \tag{3.3.39}$$

Although one still needs to calculate $\nabla^2\vec{u}$, which is inevitable, to estimate $\vec{u}^{*}$, the second order derivative term is not explicitly involved in Eqs. (3.3.38) and (3.3.39) and does not need to be calculated again after $\vec{u}^{*}$ is available. The solid boundary condition of pressure (Eq. (3.3.38) or Eq. (3.3.39)) is applied directly to the wall particles. That is, the pressure equation Eq. (3.3.38) or Eq. (3.3.39) is discretized with the gradient of pressure approximated by Eq. (B6) or Eq. (B7) in 2D cases.

In the numerical simulations, the corresponding normal pressure derivative to the different

topographies should be found approximately using the gradient operator Eqs. (B6) and (B7) or the combinations of Eqs.(B6) and (B7) in 2D cases. The details about how to implement the solid boundary condition will be presented as follows. Taking the 2D breaking wave over a beach as example, the detailed implementations will be given based on different solid boundaries. The schematic set-up is plotted in Fig. 3.3.4, the whole solid boundary can be simply divided into four kinds of different solid boundary, namely, *a, b, c* and *d* boundaries.

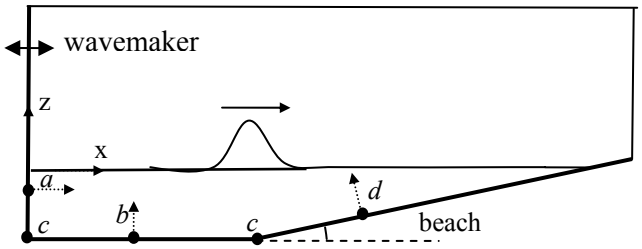


Fig. 3.3.4 schematic of different solid boundaries

For the wall particles on boundary *a*, the inwards normal pressure derivative, i.e. the pressure derivative in *x* direction will be found approximatively using the gradient operator Eq. (B6) from the neighboring particles. On this boundary the pressure condition Eq. (3.3.38) will be changed into

$$p_x^{n+1} = \frac{\rho}{\Delta t}(\vec{u}_x^* - \vec{U}_x^{n+1}) \tag{3.3.40}$$

where p_x is the pressure derivative in *x* direction; $\vec{u}_x^*$ is the intermediate velocity component in *x* direction calculated based on Eq. (3.2.1), $\vec{U}_x^{n+1}$ is the x-dimensional velocity component of the boundary at $(n+1)^{th}$ time step.

For the wall particles on boundary *b*, the inwards normal pressure derivative, i.e. the pressure derivative in z direction will be found approximately using the gradient operator Eq. (3.3.42) from the neighbouring particles. On this boundary the pressure condition Eq. (3.3.39) can be rewritten as

$$p_z^{n+1} = \frac{\rho}{\Delta t}\vec{u}_z^* \tag{3.3.41}$$

where p_z is the derivative of pressure in *z* direction; $\vec{u}_z^*$ is the intermediate velocity

component in z direction calculated based on Eq. (3.2.1), in which $\vec{u}_z^n = 0$ is assigned according to the kinematic condition.

For the wall particles on boundary *c*, which is the intersection points of two different solid boundaries, they are singularity points, which need to satisfy both boundary conditions. To overcome the numerical difficult in treating singularly points, the MLS is used to approximate the pressure at those points using their neighboring particles. The details can be seen in Eqs. (3.3.12) ~ (3.3.18).

For the wall particles on boundary *d*, the pressure derivative in the normal direction is the combination of the derivative in x direction and the derivative in z direction. The illustration is plotted in Fig. 3.3.5. The left side of Eq. (3.3.39) can be rewritten as

$$\vec{n} \cdot \nabla p = \frac{\partial p}{\partial n} = p_x \frac{\partial x}{\partial n} + p_z \frac{\partial z}{\partial n} = (-\sin\theta) p_x + \cos\theta p_z \tag{3.3.42}$$

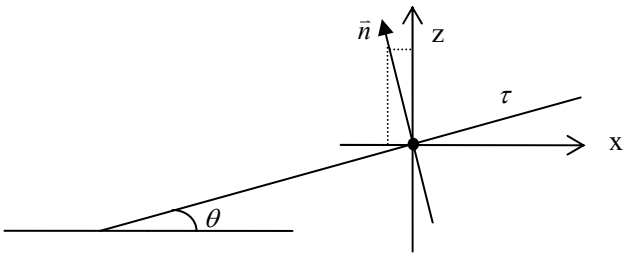


Fig. 3.3.5 Schematic analysis of normal derivative for *d* boundary particle

Using the Eqs. (B6) and (B7), Eq. (3.3.42) can be discretized to construct the pressure governing equation at the particles on the solid boundary. The normal velocity at the right side of Eq. (3.3.39), i.e. $u_{\vec{n}}^*$, can be calculated based on Eq. (3.2.1) with $u_{\vec{n}}^n = 0$ according to the kinematic condition.

For 2D cases with more complicated geometry of seabed, solid boundaries can be considered as a combination of *a, b, c* and *d* boundaries. The above four formulas can be used to implement the boundary conditions.

Zhou et al (2008) investigated three different numerical implementations of solid boundary in meshless methods. Two of them are from the MPS method, they are named as BC1 and

BC2 for brevity as discussed in Chapter 2.4; another one is from MLPG_R method, which is presented above and named as BC3. In that paper, 2D cases for solitary wave propagation in a flat tank have been investigated with the three different numerical implementations of solid boundary. Both the wave elevation and the pressure time history were studied. The particles were uniformly distributed at beginning with the same distance between water particles in both directions. The number of particles along z-direction is represented by *Nz*. Different particle numbers (*Nz*) have been investigated for BC1, BC2 and BC3, i.e. *Nz*=15, 20, 25, 30 and 40 for BC1 and BC2; *Nz*=15, 20 and 25 for BC3. The numerical results showed that the wave elevation and pressure on the rigid wall at fixed locations were different with $Nz \le 25$ for the BC1 and BC2, however, the results became to convergent until $Nz \ge 30$. On the other hand, the results obtained by using BC3 with *Nz=20* were almost the same as those with $Nz = 25$. So one can conclude that the results with $Nz = 20$ have already been convergent. Many comparisons between BC1, BC2 and BC3 have been made in that paper and conclusion has been drawn that to achieve the results with similar accuracy, the BC3 requires much less number of particles than the other two. Therefore, the BC3 may be recommended to achieve higher computational efficiency or accuracy.

3.3.4 Test the effectiveness of parameter α in Eq. (3.2.3)

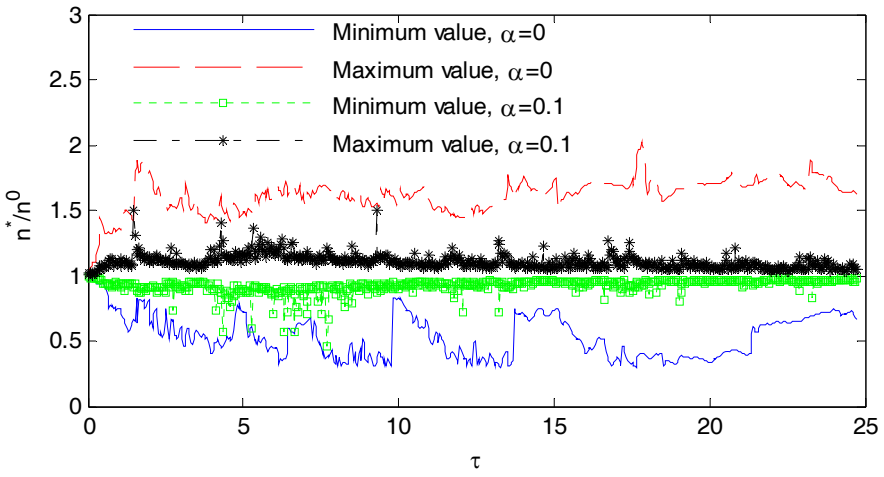
The formulation in Eq. (3.2.1) to Eq. (3.2.6) is different from the previous MLPG_R formulations by Ma (2005a, b and 2008) in two aspects. (1) The viscous term is considered in this work but not considered in the previous work. (2) There are two terms on the right hand side of Eq. (3.2.3) rather than one term in the previous publications. Nevertheless, if $\alpha = 0$, the formulation here become the formulation in Ma (2005a, b and 2008) automatically. As pointed out in cited papers, the viscous effect is neglectable if waves are not breaking. That is why the associated term is ignored in those papers that considered only non-breaking steep waves. This thesis aims to study the breaking waves and thus the viscosity is likely important. As a result, viscosity must be taken into account. Regarding the change in Eq. (3.2.3), it can be shown that the governing equation for the pressure have two forms: one corresponding to $\alpha = 0$ (Ma 2005a, b and 2008) and the other to $\alpha = 1$ [Koshizuka & Oka (1996); Idelsohn Storti and Onate (2001)], and both are derived by applying the continuity

equation for incompressible fluid. Without wave breaking, the formulation with $\alpha = 0$ can yield good results as demonstrated in the cited papers. However, when wave breaking occurs, splash and re-entry take place frequently. In such cases, none of the two forms on their own works very well. Assigning a nonzero small value to α can improve the results dramatically, which will be demonstrated in the following section. Such an approach has also been adopted by Zhang, Morita, Kenji and Shirakawa (2006) for the MPS method.

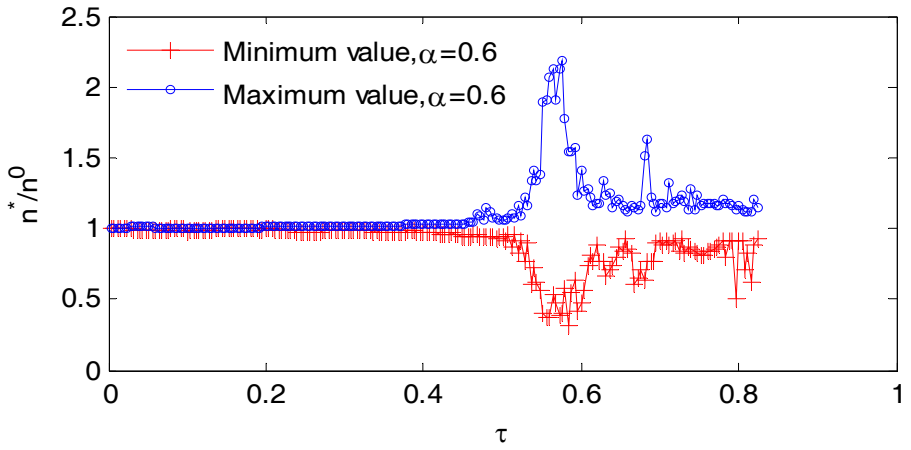
As discussed above, either of the form of Eq. (3.2.3) with $\alpha = 0$ or the one with $\alpha = 1$ does not work well in the cases for breaking waves. The root cause of the problem, based on the observation of numerical results, is that the distribution of the particles becomes over-distorted. This sub-section will present some results indicating that the problem may be overcome by choosing appropriate value for α.

For this purpose, one needs to define an indicator that reflects the level of distortion of particle distribution. The particle number density (PND) will be chosen as the indicator. That is because the change in the PND at inner particles defined in Eq. (3.3.8) can reflect the feature of particle distributions. At the beginning, the difference between the largest and the smallest PND of inner particles is a given value depending on the initial distribution of particles. It is very small if the initial distribution is almost uniform. With increase in the level of distortion of particle distribution, the difference between the largest and the smallest PND will grow. The larger the difference, the severer the distortion is.

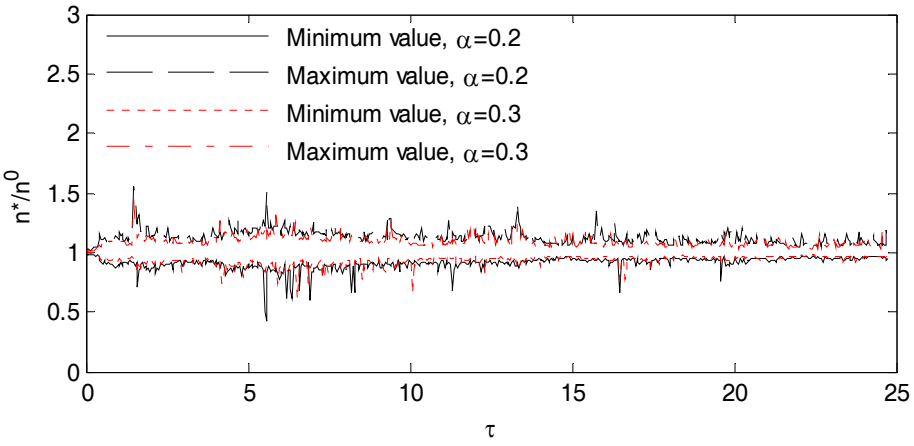
To show how the PND changes, a dam breaking problem is considered. This is a classical case for studying violent free surface flow, which has been numerically simulated by many researchers using various methods, such as Monaghan (1994) and Koshizuka and Oka (1996). The geometry used here is similar to that in Koshizuka and Oka (1996). In this case, the parameters with a length scale are nondimensionalised by the water depth *d* and the time is nondimensionalised by $\sqrt{d/g}$. So the ratio of the length to the height of the water column confined by a plate is 0.5 with the total length of the tank being 2. The plate is lifted off at τ =0 instantaneously. To model this case, the distribution of particles is in uniform at beginning and the particle number along the *z* direction is 50, which yields the total particle number being 1,551. And the time step $\Delta\tau$ is chosen as 0.004. Because the aim here is to show



(a)



(b)

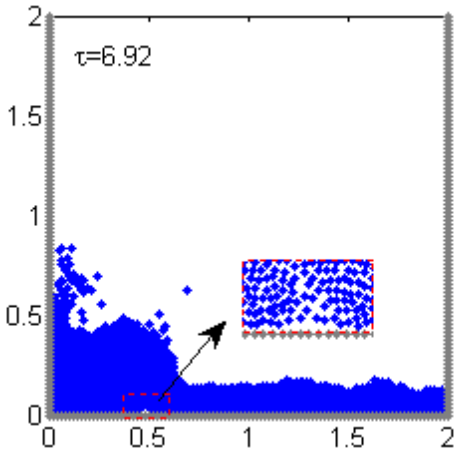


(c)

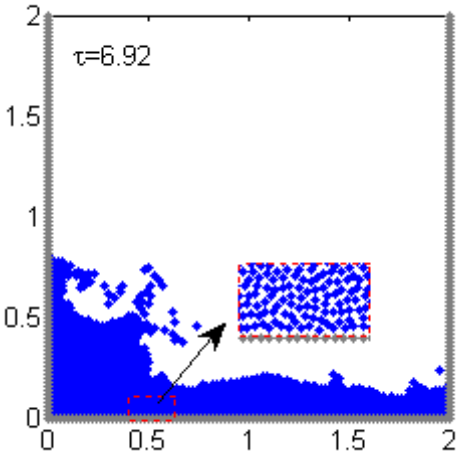
Fig. 3.3.6 Effects of different α on the PND in dam breaking cases

the effectiveness of α, the change in particle number and time step is not considered, though it is believed that the values chosen here are sufficient according to the numerical tests given below. The time histories of the ratios of the maximum and minimum values of the intermediate PND to the initial PND for inner particles excluding the free surface particle are recorded at each intermediate time step. They are plotted in Fig. 3.3.6 (a) for the cases with $\alpha = 0$ and α=0.1. One can see from this figure that the maximum ratio fluctuates at about 1.7 and the minimum ratio change around 0.5 for the case with $\alpha = 0$. The difference between them is about 1.2. On the other hand, the difference between the maximum and minimum ratios for the case with $\alpha = 0.1$ is about 0.4 and tends to be consistently smaller with increase of time. The effect of α is further illustrated by using Fig. 3.3.7, where the configurations of particles at the same instant for $\alpha = 0$ and $\alpha = 0.1$ are plotted and two small areas are enlarged in the figure. The distribution of the particles in the enlarged areas for $\alpha = 0.1$ is more uniform than that for $\alpha = 0$. These observations show that the particle distribution in the case with $\alpha = 0.1$ is better than in the case with $\alpha = 0$ and the former is likely to lead to better results.

The cases with other values of $\alpha > 0.1$ are also investigated. The similar results can be obtained with α=0.1, α=0.2 and α=0.3 shown in Fig. 3.3.6 (c). However, when $\alpha > 0.4$, the difference between the maximum and minimum PND is found to be large after a period of simulation even though it is small at the beginning. This feature is illustrated in Fig. 3.3.6 (b) for the case with $\alpha = 0.6$. It can be seen from this figure that the maximum and minimum ratios are almost the same from $\tau = 0$ to about $\tau = 0.5$ but in the area near $\tau = 0.57$ the difference reaches 1.5 with the maximum PND being 2.2 and the minimum PND being 0.7. Although the difference becomes smaller again after $\tau = 0.65$, the results for the flow is not right any more due to the error produced when it is large in the earlier steps. Based on these numerical investigations, the acceptable value of α is in the range from 0.1 and 0.3. In this study, α is chosen as 0.1. Nevertheless, the appropriate value may different for other cases not considered in this study.



(a) $\alpha = 0.0$



(b) $\alpha = 0.1$

Fig. 3.3.7 Configurations of particles for the cases with different value of α

3.4 MLPG_R Formulation for 3D cases

In MLPG_R method for 3D cases, the whole computational domain is also discritized by lots of particles, which are again separated into three groups: wall particles, free surface particles and inner particles. The MLPG_R formulations for 3D cases are similar to those described in Chapter 3.3. The main difference is the integration domain. Therefore, details will not be repeated here. Only the difference will be described as follows.

In the 3D formulation, the test function is also taken as the solution of Rankine source, the

expression of the solution for Rankine source for 3D cases is

$$\varphi = \frac{1}{4\pi}(1 - R_I / r) \tag{3.4.1}$$

Using the same techniques as those adopted in 2D cases, Eq. (3.3.2) will be simplified and the differences between the 2D and 3D are the coefficients in Eqs. (3.3.4) and (3.3.6). In 3D cases, they are

$$\int_{\partial\varepsilon} [\vec{n} \cdot (p\nabla\varphi)] dS = -R_I p \tag{3.4.2}$$

and

$$\int_{\Omega_I} \varphi d\Omega = -\frac{R_I^3}{6} \tag{3.4.3}$$

Using Eqs. (3.4.2) and (3.4.3), Eq. (3.3.2) can be rewritten as:

$$\int_{\partial\Omega_I} \vec{n} \cdot (p\nabla\varphi) dS - R_I p = \alpha \frac{\rho}{\Delta t^2} \frac{n^0 - n_I^*}{n^0} \frac{R_I^3}{6} + (1-\alpha) \int_{\Omega_I} \frac{\rho}{\Delta t} \vec{u}^* \cdot \nabla\varphi d\Omega \tag{3.4.4}$$

where the increment of the density is also replaced by the particle number density as done in 2D cases. Based on the experience in 2D cases, a small value coefficient α is taken as 0.1 for all 3D cases. The method to discritize Eq. (3.4.4) is very similar to 2D cases and therefore only brief description will be given in the following section for completeness.

Substituting Eq. (3.3.11) into Eq. (3.4.4), it follows that:

$$[K_{IJ}] \cdot [\hat{p}] = [F_I] \tag{3.4.5}$$

where $\hat{p}$ is the vector formed by the nodal value of pressure p_I and the elements of the Matrixes K_{IJ} and F_I are given by

$$K_{IJ} = \int_{\partial\Omega_I} \Phi_J(\vec{x}) \vec{n} \cdot \nabla\varphi dS - R_I \Phi_J(\vec{x}) \tag{3.4.6}$$

$$F_I = \alpha \frac{\rho}{\Delta t^2} \frac{n^0 - n^*}{n^0} \frac{R_I^3}{6} + (1-\alpha) \int_{\Omega_I} \frac{\rho}{\Delta t} \vec{u}^{(*)} \cdot \nabla\varphi d\Omega \tag{3.4.7}$$

where particle I is an inner water particle; and particle J are those influencing particle I, determined by the weight function Eq. (3.3.17). Employing with the test function Eq. (3.4.1),

the sphere surface integral in Eq. (3.4.6) can be rewritten as:

$$\int_{\partial\Omega_I} \Phi_J(\vec{x})\vec{n}\cdot\nabla\varphi dS = \frac{R_I}{4\pi}\int_0^{2\pi}\int_0^{\pi} \Phi_J(\vec{x})\sin\theta d\theta d\gamma \tag{3.4.8}$$

The domain integral over the sphere in Eq. (3.4.7) can also be simplified to:

$$\int_{\Omega_I} \frac{\rho}{\Delta t}(\vec{u}^{(*)}\cdot\nabla\varphi)d\Omega = \frac{\rho R_I}{4\pi\Delta t}\int_0^{R_I}\int_0^{2\pi}\int_0^{\pi} u_r^{(*)}\sin\theta dr d\theta d\gamma \tag{3.4.9}$$

where $u_r^{(*)}$ is the radial component of $\vec{u}^{(*)}$, θ and γ are defined in Fig. 3.4.1.

3.4.1 Numerical technique for evaluating domain integrals

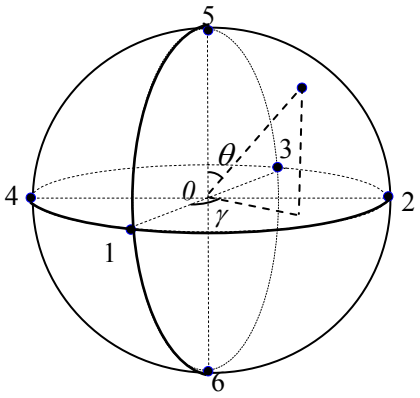


Fig. 3.4.1 Illustration of division of an integration domain

In previous Chapter 3.3.2, a semi-analytical technique has been applied to 2D problems and proved to be sufficiently accurate and more efficient than Gaussian quadrature. This technique will be extended to 3D cases here. For this purpose, let us consider a spherical domain with a radius of R_I, which is centred at (x_0, y_0, z_0) (Fig. 3.4.1) and divided into 8 sub-domains. Over each sub-domain, e.g. 0-1-2-5, the intermediate velocity components are assumed to be linear with respect to coordinates and given by

$$u^{(*)} = u_0^{(*)} + C_{ux}\frac{x-x_0}{R_I} + C_{uy}\frac{y-y_0}{R_I} + C_{uz}\frac{z-z_0}{R_I} \tag{3.4.10a}$$

$$v^{(*)} = v_0^{(*)} + C_{vx}\frac{x-x_0}{R_I} + C_{vy}\frac{y-y_0}{R_I} + C_{vz}\frac{z-z_0}{R_I} \tag{3.4.10b}$$

$$w^{(*)} = w_0^{(*)} + C_{wx}\frac{x - x_0}{R_I} + C_{wy}\frac{y - y_0}{R_I} + C_{wz}\frac{z - z_0}{R_I} \tag{3.4.10c}$$

where ($u^{(*)}, v^{(*)}, w^{(*)}$) are the intermediate velocity components at targeted point (*x, y, z*) in the sub-domain 0-1-2-5; and $(u_0^{(*)}, v_0^{(*)}, w_0^{(*)})$ are those at its centre point 0. C_{ux}, C_{uy}, C_{uz}, C_{vx}, C_{vy}, C_{vz} and C_{wx}, C_{wy}, C_{wz} are constants, which are determined in such a way that the velocity components equal to those at Point 1, 2 and 5. Taking the *x*-component as an example, one should have

$$C_{ux}(x_1 - x_0) + C_{uy}(y_1 - y_0) + C_{uz}(z_1 - z_0) = (u_1^{(*)} - u_0^{(*)})R_I \tag{3.4.11a}$$

$$C_{ux}(x_2 - x_0) + C_{uy}(y_2 - y_0) + C_{uz}(z_2 - z_0) = (u_2^{(*)} - u_0^{(*)})R_I \tag{3.4.11b}$$

$$C_{ux}(x_5 - x_0) + C_{uy}(y_5 - y_0) + C_{uz}(z_5 - z_0) = (u_5^{(*)} - u_0^{(*)})R_I \tag{3.4.11c}$$

which yields

$$C_{ux} = u_1^{(*)} - u_0^{(*)} \tag{3.4.12a}$$

$$C_{uy} = u_2^{(*)} - u_0^{(*)} \tag{3.4.12b}$$

$$C_{uz} = u_5^{(*)} - u_0^{(*)} \tag{3.4.12c}$$

Similarly, if taking the *y*-component as an example, one should have

$$C_{vx}(x_1 - x_0) + C_{vy}(y_1 - y_0) + C_{vz}(z_1 - z_0) = (v_1^{(*)} - v_0^{(*)})R_I \tag{3.4.13a}$$

$$C_{vx}(x_2 - x_0) + C_{vy}(y_2 - y_0) + C_{vz}(z_2 - z_0) = (v_2^{(*)} - v_0^{(*)})R_I \tag{3.4.13b}$$

$$C_{vx}(x_5 - x_0) + C_{vy}(y_5 - y_0) + C_{vz}(z_5 - z_0) = (v_5^{(*)} - v_0^{(*)})R_I \tag{3.4.13c}$$

which will lead to

$$C_{vx} = v_1^{(*)} - v_0^{(*)} \tag{3.4.14a}$$

$$C_{vy} = v_2^{(*)} - v_0^{(*)} \tag{3.4.14b}$$

$$C_{vz} = v_5^{(*)} - v_0^{(*)} \tag{3.4.14c}$$

Similarly, if taking the *z*-component as an example, one should have

$$C_{wx}(x_1 - x_0) + C_{wy}(y_1 - y_0) + C_{wz}(z_1 - z_0) = (w_1^{(*)} - w_0^{(*)})R_I \tag{3.4.15a}$$

$$C_{wx}(x_2 - x_0) + C_{wy}(y_2 - y_0) + C_{wz}(z_2 - z_0) = (w_2^{(*)} - w_0^{(*)})R_I \tag{3.4.15b}$$

$$C_{wx}(x_5 - x_0) + C_{wy}(y_5 - y_0) + C_{wz}(z_5 - z_0) = (w_5^{(*)} - w_0^{(*)})R_I \tag{3.4.15c}$$

which will yield

$$C_{wx} = w_1^{(*)} - w_0^{(*)} \tag{3.4.16a}$$

$$C_{wy} = w_2^{(*)} - w_0^{(*)} \tag{3.4.16b}$$

$$C_{wz} = w_5^{(*)} - w_0^{(*)} \tag{3.4.16c}$$

Substituting Eqs. (3.4.12), (3.4.14) and (3.4.16) into Eq. (3.4.10) results in

$$u^{(*)} = u_0^{(*)} + (u_1^{(*)} - u_0^{(*)})\frac{x - x_0}{R_I} + (u_2^{(*)} - u_0^{(*)})\frac{y - y_0}{R_I} + (u_5^{(*)} - u_0^{(*)})\frac{z - z_0}{R_I} \tag{3.4.17a}$$

$$v^{(*)} = v_0^{(*)} + (v_1^{(*)} - v_0^{(*)})\frac{x - x_0}{R_I} + (v_2^{(*)} - v_0^{(*)})\frac{y - y_0}{R_I} + (v_5^{(*)} - v_0^{(*)})\frac{z - z_0}{R_I} \tag{3.4.17b}$$

$$w^{(*)} = w_0^{(*)} + (w_1^{(*)} - w_0^{(*)})\frac{x - x_0}{R_I} + (w_2^{(*)} - w_0^{(*)})\frac{y - y_0}{R_I} + (w_5^{(*)} - w_0^{(*)})\frac{z - z_0}{R_I} \tag{3.4.17c}$$

Thus the velocities $(u^{(*)}, v^{(*)}, w^{(*)})$ in the sub-domain 0-1-2-5 are estimated by Eq. (3.4.17). The velocities in other sub-domains can also be estimated in this way. The only difference is that they may be related to the velocities at Points 3, 4 and 6, depending on which sub-domain is concerned. Consequently, the velocities at any point in the sphere are determined by those at only 7 points (0, 1, 2, 3, 4, 5 and 6). The relationship between $u_r^{(*)}$ and $(u^{(*)}, v^{(*)}, w^{(*)})$ is listed as follows:

$$u_r^{(*)} = u^{(*)} \sin\theta\cos\gamma + v^{(*)} \sin\theta\sin\gamma + w^{(*)} \cos\theta \tag{3.4.18}$$

Based on this, the spherical integration in Eq. (3.4.9) can be discretized at every sub-domain:

$$\int_0^{R_I}\int_0^{2\pi}\int_0^{\pi} u_r^{(*)} \sin\theta dr d\theta d\gamma = \sum_{j=1}^{M_j}\sum_{i=1}^{N_i}\int_{\theta_j}^{\theta_{j+1}}\int_{\gamma_i}^{\gamma_{i+1}}\int_0^{R_I} u_r^{(*)} \sin\theta dr d\theta d\gamma \tag{3.4.19}$$

where $N_i = 4$, $M_j = 2$ and $\gamma_5 = \gamma_1$.

Substituting Eqs. (3.4.17) and (3.4.18) into Eq. (3.4.19), the spherical integration evaluated analytically at the first sub-domain 0-1-2-5 is given by

$$\int_0^{R_I}\int_0^{\pi/2}\int_0^{\pi/2} u_r^{(*)} \sin\theta dr d\theta d\gamma =$$

$$\frac{\pi}{4} R_I (u_0^{(*)} + v_0^{(*)} + w_0^{(*)}) + \frac{\pi}{12} R_I (u_1^{(*)} - u_0^{(*)} + v_2^{(*)} - v_0^{(*)} + w_5^{(*)} - w_0^{(*)}) \tag{3.4.20}$$

$$+ \frac{1}{6} R_I (u_2^{(*)} + u_5^{(*)} - 2u_0^{(*)} + v_1^{(*)} + v_5^{(*)} - 2v_0^{(*)} + w_1^{(*)} + w_2^{(*)} - 2w_0^{(*)})$$

Consequently in the same way, the spherical integration at sub-domain 0-2-3-5 can be obtained by

$$\int_0^{R_I}\int_0^{\pi/2}\int_{\pi/2}^{\pi} u_r^{(*)} \sin\theta dr d\theta d\gamma =$$

$$\frac{\pi}{4} R_I (-u_0^{(*)} + v_0^{(*)} + w_0^{(*)}) + \frac{\pi}{12} R_I (-u_3^{(*)} + u_0^{(*)} + v_2^{(*)} - v_0^{(*)} + w_5^{(*)} - w_0^{(*)}) \tag{3.4.21}$$

$$+ \frac{1}{6} R_I (-u_2^{(*)} - u_5^{(*)} + 2u_0^{(*)} + v_3^{(*)} + v_5^{(*)} - 2v_0^{(*)} + w_3^{(*)} + w_2^{(*)} - 2w_0^{(*)})$$

the spherical integration at sub-domain 0-3-4-5 can be obtained by

$$\int_0^{R_I}\int_0^{\pi/2}\int_{\pi}^{3\pi/2} u_r^{(*)} \sin\theta dr d\theta d\gamma =$$

$$\frac{\pi}{4} R_I (-u_0^{(*)} - v_0^{(*)} + w_0^{(*)}) + \frac{\pi}{12} R_I (-u_3^{(*)} + u_0^{(*)} - v_4^{(*)} + v_0^{(*)} + w_5^{(*)} - w_0^{(*)}) \tag{3.4.22}$$

$$+ \frac{1}{6} R_I (-u_4^{(*)} - u_5^{(*)} + 2u_0^{(*)} - v_3^{(*)} - v_5^{(*)} + 2v_0^{(*)} + w_3^{(*)} + w_4^{(*)} - 2w_0^{(*)})$$

the spherical integration at sub-domain 0-4-1-5 can be obtained by

$$\int_0^{R_I}\int_0^{\pi/2}\int_{3\pi/2}^{2\pi} u_r^{(*)} \sin\theta dr d\theta d\gamma =$$

$$\frac{\pi}{4} R_I (u_0^{(*)} - v_0^{(*)} + w_0^{(*)}) + \frac{\pi}{12} R_I (u_1^{(*)} - u_0^{(*)} - v_4^{(*)} + v_0^{(*)} + w_5^{(*)} - w_0^{(*)}) \tag{3.4.23}$$

$$+ \frac{1}{6} R_I (u_4^{(*)} + u_5^{(*)} - 2u_0^{(*)} - v_1^{(*)} - v_5^{(*)} + 2v_0^{(*)} + w_1^{(*)} + w_4^{(*)} - 2w_0^{(*)})$$

the spherical integration evaluated analytically at the first sub-domain 0-1-2-6 is given by

$$\int_0^{R_I}\int_{\pi/2}^{\pi}\int_0^{\pi/2} u_r^{(*)} \sin\theta dr d\theta d\gamma =$$

$$\frac{\pi}{4} R_I (u_0^{(*)} + v_0^{(*)} - w_0^{(*)}) + \frac{\pi}{12} R_I (u_1^{(*)} - u_0^{(*)} + v_2^{(*)} - v_0^{(*)} - w_6^{(*)} + w_0^{(*)}) \tag{3.4.24}$$

$$+ \frac{1}{6} R_I (u_2^{(*)} + u_6^{(*)} - 2u_0^{(*)} + v_1^{(*)} + v_6^{(*)} - 2v_0^{(*)} - w_1^{(*)} - w_2^{(*)} + 2w_0^{(*)})$$

the spherical integration at sub-domain 0-2-3-6 can be obtained by

$$\int_0^{R_I}\int_{\pi/2}^{\pi}\int_{\pi/2}^{\pi} u_r^{(*)} \sin\theta dr d\theta d\gamma =$$
$$\frac{\pi}{4} R_I(-u_0^{(*)} + v_0^{(*)} - w_0^{(*)}) + \frac{\pi}{12} R_I(-u_3^{(*)} + u_0^{(*)} + v_2^{(*)} - v_0^{(*)} - w_6^{(*)} + w_0^{(*)}) \tag{3.4.25}$$
$$+ \frac{1}{6} R_I(-u_2^{(*)} - u_6^{(*)} + 2u_0^{(*)} + v_3^{(*)} + v_6^{(*)} - 2v_0^{(*)} - w_3^{(*)} - w_2^{(*)} + 2w_0^{(*)})$$

the spherical integration at sub-domain 0-3-4-6 can be obtained by

$$\int_0^{R_I}\int_{\pi/2}^{\pi}\int_{\pi}^{3\pi/2} u_r^{(*)} \sin\theta dr d\theta d\gamma =$$
$$\frac{\pi}{4} R_I(-u_0^{(*)} - v_0^{(*)} - w_0^{(*)}) + \frac{\pi}{12} R_I(-u_3^{(*)} + u_0^{(*)} - v_4^{(*)} + v_0^{(*)} - w_6^{(*)} + w_0^{(*)}) \tag{3.4.26}$$
$$+ \frac{1}{6} R_I(-u_4^{(*)} - u_6^{(*)} + 2u_0^{(*)} - v_3^{(*)} - v_6^{(*)} + 2v_0^{(*)} - w_3^{(*)} - w_4^{(*)} + 2w_0^{(*)})$$

the spherical integration at sub-domain 0-4-1-6 can be obtained by

$$\int_0^{R_I}\int_{\pi/2}^{\pi}\int_{3\pi/2}^{2\pi} u_r^{(*)} \sin\theta dr d\theta d\gamma =$$
$$\frac{\pi}{4} R_I(u_0^{(*)} - v_0^{(*)} - w_0^{(*)}) + \frac{\pi}{12} R_I(u_1^{(*)} - u_0^{(*)} - v_4^{(*)} + v_0^{(*)} - w_6^{(*)} + w_0^{(*)}) \tag{3.4.27}$$
$$+ \frac{1}{6} R_I(u_4^{(*)} + u_6^{(*)} - 2u_0^{(*)} - v_1^{(*)} - v_6^{(*)} + 2v_0^{(*)} - w_1^{(*)} - w_4^{(*)} + 2w_0^{(*)})$$

Added all the results together, the Eq. (3.4.19) can be rewritten as

$$\int_0^{R_I}\int_0^{2\pi}\int_0^{\pi} u_r^{(*)} \sin\theta dr d\theta d\gamma = \frac{\pi}{3} R_I(u_1^{(*)} + v_2^{(*)} + w_5^{(*)} - u_3^{(*)} - v_4^{(*)} - w_6^{(*)}) \tag{3.4.28}$$

It can be seen that the integral over each local spherical domain is determined by the velocities only at six points rather than at least at 64 Gaussian points if the Gaussian quadrature would be employed.

3.4.2 Numerical technique for evaluating surface integrals

The integral of Eq. (3.4.8) regarding the shape function is an integral over a curve in 2D cases, which was estimated by using Gaussian quadrature in Ma & Zhou (2009). For 3D cases, if using the same way, it will need $8M^2$ Gaussian points assuming the spherical surface is divided into 8 quadrants, where M is the number of Gaussian points corresponding

to each variable of the integral in one quadrant. Later, it will be shown that $M \geq 3$, indicating that at least 72 Gaussian points over the whole spherical surface of the local domain are required, which would be very time consuming if the Gaussian quadrature is still used. Therefore, a new numerical technique is developed to evaluate the spherical surface integral in Eq. (3.4.8), which is similar to that for evaluating the domain integral described above but based on a different approximate function. To introduce the technique, the integral in Eq. (3.4.8) is rewritten as

$$\int_0^{2\pi}\int_0^{\pi} \Phi \sin\theta d\theta d\gamma = \sum_{i=1}^{N_i}\sum_{j=1}^{M_j} \int_{\theta_j}^{\theta_{j+1}} \int_{\gamma_i}^{\gamma_{i+1}} \Phi \sin\theta d\theta d\gamma \tag{3.4.29}$$

where $N_i = 4$, $M_j = 2$ and $\gamma_5 = \gamma_1$ and the subscript of Φ is dropped without affecting its meaning. That is, the integration is first performed over each of 8 quadrants and then the results are added together, similar to what have been done for obtaining Eq. (3.4.28).

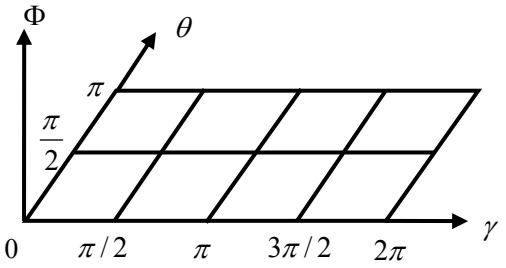


Fig.3.4.2 Illustration of integration domain in θ - γ plane

The shape function Φ in the integrand in Eq. (3.4.29) is the function of θ - γ and thus it may be transformed to the θ - γ plane that may be split into 8 patches as illustrated in Fig. 3.4.2. It is noted, however, that the shape function Φ is the same for any values of γ when $\theta = 0$ or $\theta = \pi$ because it corresponds to a single point 5 or 6 in Fig. 3.4.1. Consider the shape function Φ in the quadrant 1-2-5 in Fig. 3.4.1 or the patch near the origin in Fig. 3.4.2, we may approximate it by using

$$\Phi = A + B\theta + C\gamma + D\theta\gamma \tag{3.4.30}$$

where *A, B* , *C* and *D* are constants. They can be determined by using the fact that the shape

function should be equal to Φ_5 at $\theta = 0$ and $\gamma = 0$ or $\gamma = \pi/2$; equal to Φ_1 at $\theta = \pi/2$ and $\gamma = 0$; and equal to Φ_2 at $\theta = \pi/2$ and $\gamma = \pi/2$, where the subscripts of Φ refer to the number in Fig. 3.4.1, that is

$$\Phi_5 = A \tag{3.4.31a}$$

$$\Phi_5 = A + C\pi/2 \tag{3.4.31b}$$

$$\Phi_1 = A + B\pi/2 \tag{3.4.31c}$$

$$\Phi_2 = A + B\pi/2 + C\pi/2 + D(\pi/2)^2 . \tag{3.4.31d}$$

It follows that

$$A = \Phi_5 \tag{3.4.32a}$$

$$C = 0 \tag{3.4.32b}$$

$$B = \frac{2}{\pi}(\Phi_1 - \Phi_5) \tag{3.4.32c}$$

$$D = \frac{4}{\pi^2}(\Phi_2 - \Phi_1) \tag{3.4.32d}$$

Substituting Eq. (3.4.32) into Eq. (3.4.30) yields

$$\Phi = \Phi_5 + \frac{2}{\pi}(\Phi_1 - \Phi_5)\theta + \frac{4}{\pi^2}(\Phi_2 - \Phi_1)\gamma\theta \tag{3.4.33}$$

which is the approximate expression of the shape function in the first sub-domain 1-2-5 in Fig. 3.4.1 or the patch near the origin in Fig. 3.4.2.

The integral of Eq. (3.4.29) over the sub-domain 1-2-5 is then given by

$$\int_0^{\pi/2}\int_0^{\pi/2} \Phi \sin\theta d\theta d\gamma = \frac{1}{2}\Phi_1 + \frac{1}{2}\Phi_2 + (\frac{\pi}{2} - 1)\Phi_5 \tag{3.4.34}$$

Similarly, the integration of Eq. (3.4.29) over sub domain 2-3-5 is given by

$$\int_0^{\pi/2}\int_{\pi/2}^{\pi}\Phi\sin\theta d\theta d\gamma=\frac{1}{2}\Phi_2+\frac{1}{2}\Phi_3+(\frac{\pi}{2}-1)\Phi_5 \tag{3.4.35}$$

Similarly, the integration of Eq. (3.4.29) over sub domain 3-4-5 is given by

$$\int_0^{\pi/2}\int_{\pi}^{3\pi/2}\Phi\sin\theta d\theta d\gamma=\frac{1}{2}\Phi_3+\frac{1}{2}\Phi_4+(\frac{\pi}{2}-1)\Phi_5 \tag{3.4.36}$$

Similarly, the integration of Eq. (3.4.29) over sub domain 4-1-5 is given by

$$\int_0^{\pi/2}\int_{3\pi/2}^{2\pi}\Phi\sin\theta d\theta d\gamma=\frac{1}{2}\Phi_4+\frac{1}{2}\Phi_1+(\frac{\pi}{2}-1)\Phi_5 \tag{3.4.37}$$

Similarly, the integration of Eq. (3.4.29) over sub domain 1-2-6 is given by

$$\int_{\pi/2}^{\pi}\int_0^{\pi/2}\Phi\sin\theta d\theta d\gamma=\frac{1}{2}\Phi_1+\frac{1}{2}\Phi_2+(\frac{\pi}{2}-1)\Phi_6 \tag{3.4.38}$$

Similarly, the integration of Eq. (3.4. 29) over sub domain 2-3-6 is given by

$$\int_{\pi/2}^{\pi}\int_{\pi/2}^{\pi}\Phi\sin\theta d\theta d\gamma=\frac{1}{2}\Phi_2+\frac{1}{2}\Phi_3+(\frac{\pi}{2}-1)\Phi_6 \tag{3.4.39}$$

Similarly, the integration of Eq. (3.4. 29) over sub domain 3-4-6 is given by

$$\int_{\pi/2}^{\pi}\int_{\pi}^{3\pi/2}\Phi\sin\theta d\theta d\gamma=\frac{1}{2}\Phi_3+\frac{1}{2}\Phi_4+(\frac{\pi}{2}-1)\Phi_6 \tag{3.4.40}$$

Similarly, the integration of Eq. (3.4. 29) over sub domain 4-1-6 is given by

$$\int_{\pi/2}^{\pi}\int_{3\pi/2}^{2\pi}\Phi\sin\theta d\theta d\gamma=\frac{1}{2}\Phi_4+\frac{1}{2}\Phi_1+(\frac{\pi}{2}-1)\Phi_6 \tag{3.4.41}$$

The integration over other sub domains can be obtained in the similar way and the sum of these results over whole spherical plane domain is given by

$$\int_0^{2\pi}\int_0^{\pi}\Phi\sin\theta d\theta d\gamma=2(\Phi_1+\Phi_2+\Phi_3+\Phi_4)+(2\pi-4)(\Phi_5+\Phi_6) \tag{3.4.42}$$

It is noted that the integral in Eq. (3.4.42) is only related to the shape function at six points, compared with at least 72 points if the Gaussian quadrature is employed. Therefore, the CPU time spent on evaluation of the integral will be dramatically reduced, which will be demonstrated in the later section.

3.4.3 Effectiveness of the semi-analytical technique for surface integrals

As pointed above, the main aim of developing this semi-analytical technique for evaluating the surface integral given in Eq. (3.4.42) is to reduce the computational time while maintaining the same or similar accuracy of results. This will be demonstrated in this section by modelling two examples using the Gaussian quadrature and the semi-analytical technique for estimating the surface integral with all others being the same. Again in following cases, the parameters with a length scale are nondimensionalised by the water depth d, the pressure by $\rho g d$ and the time by $\sqrt{d/g}$. In the first example, a static case is considered in which the solution for pressure is just the hydrostatic pressure. A tank with the height, length and width being 2, 3 and 1, respectively, is used. A schematic view of the tank is shown in Fig.3.4.3, where grey particles and blue particles represent wall particles and water particles, respectively. To make it clearer, only those of boundary particles on the right wall, the left wall and the bottom are plotted in the Fig.3.4.3. The particles are uniformly distributed initially in all the directions. The number of particles along z-direction is represented by Nz, Nz is equal to 10 in this case. The total water particles are 2,900 excluding the particles on boundaries and 5,260 including those on the boundaries.

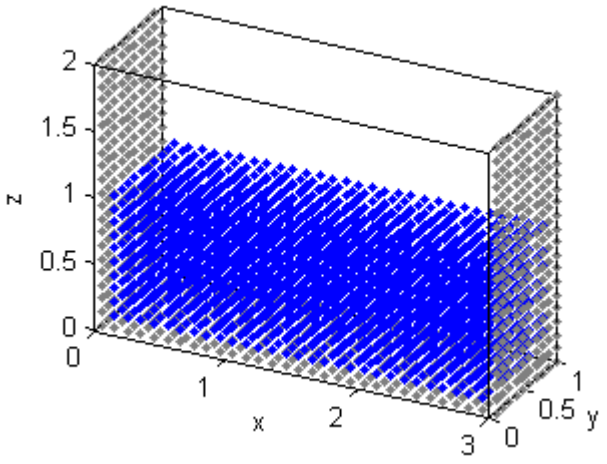
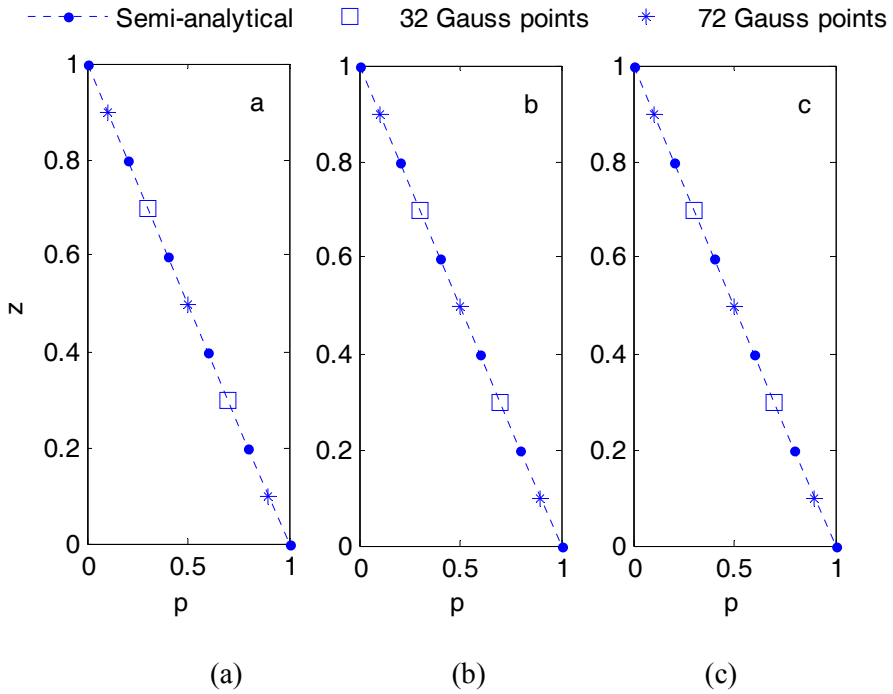


Fig.3.4.3 A schematic view of the tank

Table 3.4.1 Comparison of CPU time required by using different methods to evaluate the surface integral (GQ: Gaussian quadrature) for a static case

Method	Semi-analytical	32 Gauss points (GQ)	72 Gauss points (GQ)
Average CPU time in one time step	1	5.67	11.17



(a) (b) (c)

Fig.3.4.4 Comparison of static pressure obtained by using the Gaussian quadrature and the semi-analytical technique for estimating the surface integral

Figs. 3.4.4a, b and c present the pressure distribution along a vertical line in the middle of the tank after running 200, 400 and 600 time steps, respectively. It can be seen from these figures that the results are almost the same, which indicates that the semi-analytical technique produces the same results as the Gaussian quadrature. The CPU time required for forming Matrix $\boldsymbol{K}$ and $\boldsymbol{F}$ in Eq. (3.4.5) in the cases shown in Fig. 3.4.4 is summarised in Table. 3.4.1, where the CPU time is divided by the CPU time required by the semi-analytical technique. This table shows that the CPU time required by the Gaussian quadrature is at least 5 times longer than that required by the semi-analytical technique.

In the second example, the propagation of a solitary wave is modelled, again by using the semi-analytical technique and Gaussian quadrature with 32 (*M=2*) and 72 (*M=3*) Gaussian points, respectively. The solitary wave is generated by a piston-type wavemaker according to the theory given by Goring (1978). The numerical water tank and coordinate system are shown in Fig. 3.0.1, the water depth at the flat seabed is *1*; the width of the tank is 0.5*; L1=10* and the angle of the slope is 1:15. The wave height targeted is h =0.45. For this case, the particle number along z-axis, i.e., *Nz,* is 20, which yields 89,492 particle totally. These cases are run up to 500 time steps with the length of 0.02 each step. Wave profiles at 6 different instants are compared in Fig. 3.4.5. From this figure, one may see that the results of Gaussian quadrature with 32 Gaussian points are very different from these of Gaussian quadrature with 72 Gaussian points. The later is very similar to those obtained by using the semi-analytical technique. This indicates that at least 72 Gaussian points are required to compute the integral in Eq. (3.4.8) in this case. It is highly likely that such a number is needed for modelling other 3D violent waves, which has been confirmed for cases we considered so far. This also indicates that the semi-analytical technique can produce results that are as accurate as the Gaussian quadrature with 72 Gauss points. Similar comparison of CPU time to that shown in Table. 3.4.1 is made in Table. 3.4.2. This table demonstrates that the CPU time required by the semi-analytical technique is only 1/17.65 of that required by the Gaussian quadrature with 72 Gauss points to achieve acceptable results. Although the CPU time required for the case with 32 (*M=2*) Gaussian points is only half of that with 72 (*M=3*), the accuracy of the former is not satisfactory.

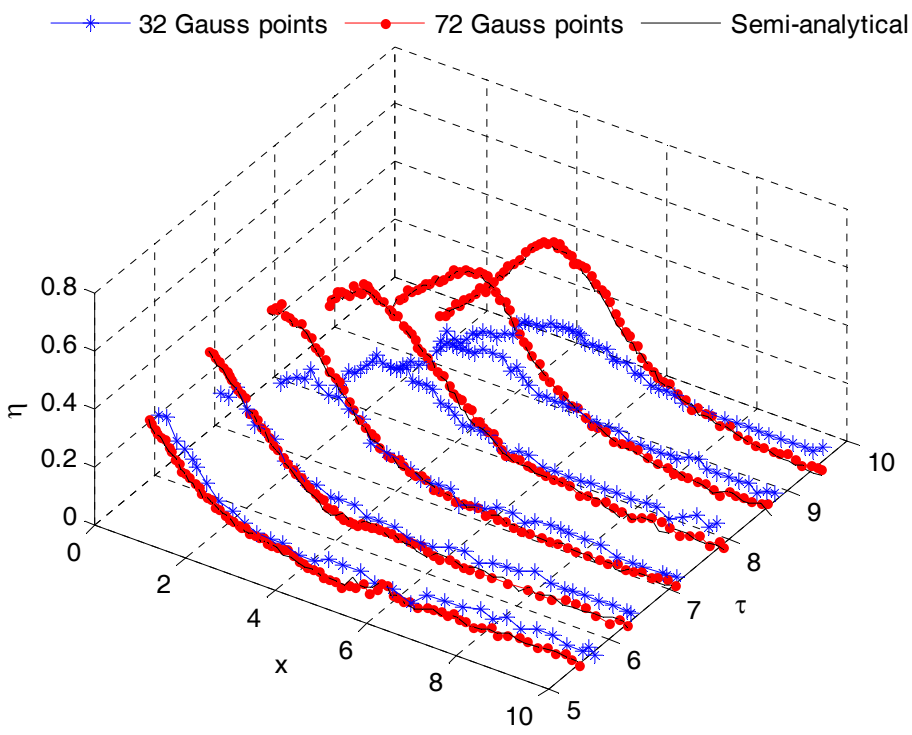


Fig. 3.4.5 Comparison of wave profiles at different instants obtained by using the semi-analytical technique and the Gaussian quadrature with 32 and 72 Gaussian points, respectively.

Table 3.4.2 Comparison of CPU time required by using different method to evaluate the surface integrals (GQ: Gaussian quadrature) for a dynamic case

Method	Semi-analytical	32 Gauss points (GQ)	72 Gauss points (GQ)
Average CPU time in one time step	1	8.57	17.65

Investigations on these static and dynamic cases demonstrate that the semi-analytical technique spends only a small fraction of the CPU time spent by the Gaussian quadrature to achieve the results with a similar accuracy. This advance allows to model 3D breaking waves on a normal PC within a reasonable time, which has proved impossible if using the Gaussian quadrature.

4 NUMERICAL TECHUNIQUES FOR IDENTIFYING PARTICLES ON FREE SURFACE

The mathematical models and some key numerical techniques adopted in MLPG_R method, including the detailed discretizations of pressure governing equation for inner particles have been described in Chapter 3. This chapter presents the numerical technique developed to identify free surface particles.

4.1 Introduction

In order to find solution for pressure by using the governing equation Eq. (3.3.10) for 2D cases and Eq. (3.4.4) for 3D cases in Chapter 3, all the particles need to be sorted into three groups: those on rigid boundaries (referred as wall particles), those on the free surface (referred as free surface particles) and others (referred as inner particles). The wall particles are always attached to the rigid boundary in this modeling. Eq. (3.3.38) or (3.3.39) is applied to wall particles. Eq. (3.3.10) or Eq. (3.4.4) is applied to the inner particles. The pressure at the free surface particles is specified by the condition in Eq. (3.1.5).

For non-breaking waves, one can assume that the particles initially on the free surface remain on the free surface. Therefore, the free surface particles need only to be specified at the first step and there is no need to identify them during the simulation.

However, for the cases with breaking waves as considered in this study, the fluid particles initially on the free surface may not remain on the free surface during calculation. Actually, the particles initially within the fluid can emerge on the free surface and the particles initially on the free surface can immerge into the inner fluid domain due to wave breaking and splashing. Consequently, it is necessary to identify which particle is on the free surface when modeling breaking-wave cases.

There is a similar requirement on identifying the free surface in mesh-based numerical

models for breaking waves, as discussed in Section 2.5. Nevertheless, the identification of free surface particles remains to be a big challenge in meshless methods, particularly for those of true meshless models without any background mesh, like the MLPG_R method.

4.2 Mixed Particle Number Density and Auxiliary Function Method (MPAM) for 2D cases

The approach to identify the free surface used in the MPS is named as Particle Number Density Method, abbreviated to PNDM in this study. In PNDM, there is no standard rule for how to assign the value of β. However, all the results associated with MPS method seem to be promising with different values of β, like the discusses in Chapter 2.5.

The PNDM method is also tested by using the MLPG_R method in this study. However, the results showed that the simple technique based on different values of β is not very robust. There are always many particles that are incorrectly identified (i.e., free surface particles are identified as inner particles or vise versa). The incorrect identification could not be rectified by simply choosing a different value for β. Similar observation has also been described by Lee and Park (2007). To shed some light on the reason for this, one may consider the examples shown in Fig.4.2.1, which illustrates particles around a steep wave crest. Particle *I* is near the crest of a steep wave. In this area, the particles on the free surface are generally much closer to each other than in other areas and so the particle number density associated with them can be very higher. As a result, Particle *I* may be incorrectly identified as an inner particle rather than a free surface particle by the PNDM. On the other hand, the neighbor particle of Particle *J* is quite far from it. The value of the particle number density can be relatively smaller and so it may be incorrectly considered as a free surface particle even though it should be an inner particle. In addition to these situations, two or three splashed particles may be very close to each other and so may also be incorrectly identified as inner particles though they should be considered as free surface particles for the sake of solving pressure.

To improve the robustness of identification of free surface particles, a new approach has been suggested (Ma & Zhou, 2009). The approach is based on three auxiliary functions. The first auxiliary function is defined by

$$fsp_a(I)=\begin{cases}1, & NA\geq 1\\ 0, & NA=0\end{cases} \tag{4.2.1}$$

where NA represents the number of free surface particles existing in the support domain (D_f) of Particle I in previous time step as shown in Fig. 4.2.2. The second auxiliary function is

$$fsp_b(I)=\begin{cases}1, & NB=4\\ 0, & NB\leq 3\end{cases}, \tag{4.2.2}$$

where NB represents the number of quadrants occupied by the neighbor particles of Particle I in a local coordinate system originating at Particle I, as shown in Fig. 4.2.2.

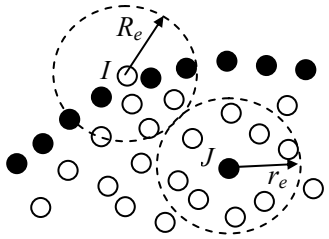


Fig. 4.2.1 Two typical examples of incorrect identification of free surface particles. (Solid circle represents free surface particle identified by the PNDM; hollow one represents inner particle identified by the PNDM; dash circle represents support domain).

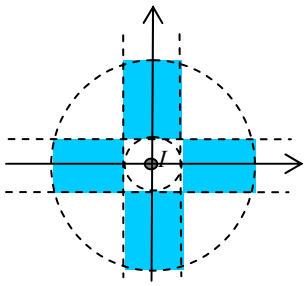


Fig. 4.2.2 Local coordinate system at Particle I (inner circle denotes integration domain of the particle; outer circle denotes the support domain on it)

The third auxiliary function is given by

$$fsp_c(I) = \begin{cases} 1, & NC = 4 \\ 0, & NC \le 3 \end{cases}, \tag{4.2.3}$$

where *NC* represents the number of colored rectangles in Fig. 4.2.2 in which there is at least one fluid particle.

If one of following conditions is met when they are checked sequentially, Particle *I* is identified as a free surface particle.

(a) no inner particle in D_f except for *I*;

(b) $\beta_I < 0.97$ and $fsp_a(I) = 1$;

(c) $\beta_I \ge 0.97$, $fsp_a(I) = 1$ and $fsp_b(I) = 0$;

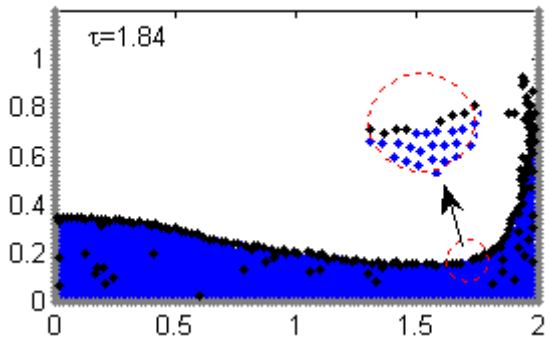
(d) $\beta_I \ge 0.97$, $fsp_a(I) = 1$ and $fsp_c(I) = 0$.

Satisfying the condition (a) indicates that the particle concerned is in the group of particles which belong to the part of splashing fluid. The condition (b) identifies those free surface particles with at least one neighbour particle on the free surface. The condition (c) and (d) picks up those free surface particles with sufficient large number of neighbour particles but with no particle in a sufficient large part of its support domain. This approach will be called as Mixed Particle Number Density and Auxiliary Function Method, abbreviated to MPAM.

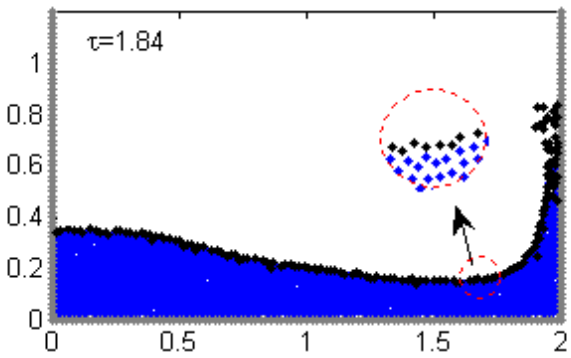
Using this new approach, the wrong identification of the free surface particles may be suppressed. For the Particle *I* in Fig. 4.2.1, there is more than one free surface particle in its support domain, and a large part of its support domain is not occupied by any particle. As a result, $fsp_a(I) = 1$, $fsp_b(I) = 0$ and it will be identified, by applying Condition (b) or (c), as a free surface particle no matter what is the value for β. For Particle *J* shown in Fig. 4.2.1, there are no free surface particles existing in its support domain and so $fsp_a(I) = 0$. Therefore, even though $\beta_I < 0.97$ for the particle, no one of four conditions will be met and thus Particle *J* is not identified as a free-surface particle.

4.3 Effectiveness of MPAM for identifying the free surface particles

In this sub-section, some numerical results will be presented to show that the MPAM works better than the PNDM. Two cases are used for this purpose. The first case is that used in Chapter 3.3.3, which is a typical 2D dam breaking case. The case is simulated by using the MPAM and the PNDM, respectively. Particle configurations obtained by using the two different techniques at an instant are shown in Fig. 4.3.1. Fig. 4.3.1 (a) is the results corresponding to the PNDM while Fig. 4.3.1 (b) is these corresponding to the MPAM. This figure clearly demonstrates that many inner particles are incorrectly identified as free surface particles by the PNDM but are correctly judged as inner particles by the MPAM. It also shows that the PNDM assigns some free surface particles to inner particles in the enlarged area but the MPAM does not make such a mistake.



(a) PNDM



(b) MPAM

Fig. 4.3.1 Comparisons of particle configurations obtained by using different free surface identification techniques (black color: free surface particles; grey or blue color: wall particles or inner particles)

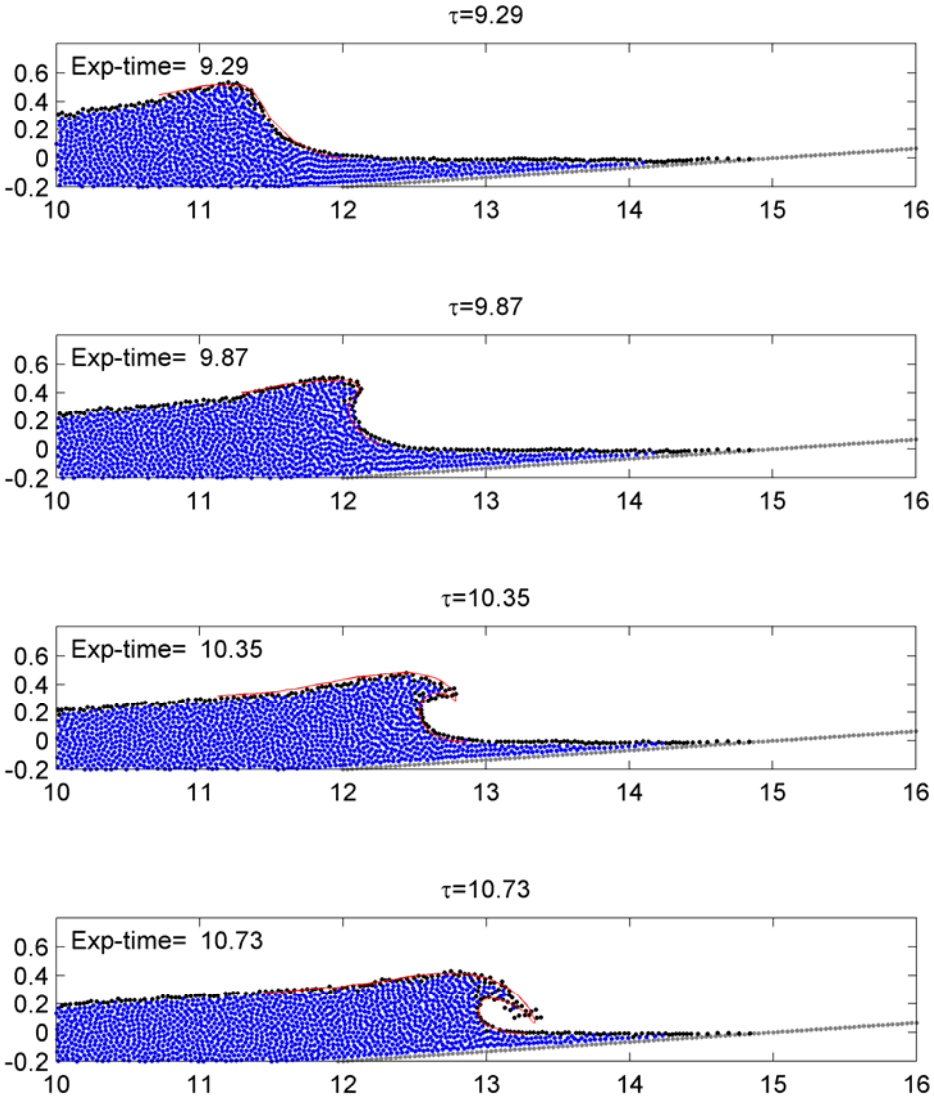


Fig. 4.3.2 Snapshots of the water particles in pre-breaking stage
(Blue particles: inner water particle; Black particles: free surface particle; Grey particles: sloping seabed and Red line: experiment data (Li & Raichlen, 1998))

The second case considered here is that used in Chapter 5.1 which is the solitary wave propagates from flat seabed to a mild beach. Here a series of particle configurations in the area of wave crest profiles are focused on to show the effectiveness of MPAM, others details can be referred to Chapter 5.1. The results in Fig. 4.3.2 and Fig. 4.3.3 are obtained based on the particle number along *z* direction on the flat seabed being 30, i.e. *Nz=30*. Four constant snapshots together with the experiment data (Li & Raichlen, 1998) are plotted in Fig. 4.3.2,

from which one may notice that the free surface particles (the black ones) have been accurately tracked regardless of how violent deformability the wave profiles are. Even for the wave profiles in the post-breaking situations, the MPAM can still find the free surface particles accurately as shown in Fig. 4.3.3. Furthermore, the free surface particles around the air cavity are correctly identified as well. From the wave propagation to formation of plunging jet and from pre-breaking stage to post-breaking stage, there is almost only one-particle layer identified as free surface particle. Hence the free surface boundary conditions can be correctly applied to the free surface particles, which to some extent guarantee the high accurate numerical simulation of breaking wave using MLPG_R method.

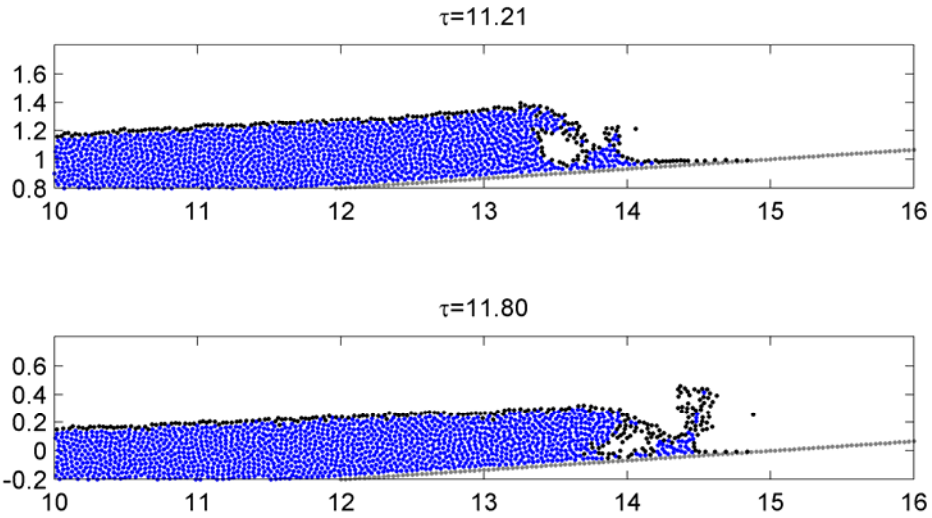


Fig. 4.3.3 Snapshots of the water particles in the post-breaking stage
(Blue particles: inner water particle; Black particles: free surface particle;
Grey particles: sloping seabed)

4.4 Mixed Particle Number Density and Auxiliary Function Method (MPAM) for 3D cases

In the above sub-section, a technique (called as MPAM) for identifying free surface particles and its effectiveness have been presented. This method can be easily extended to 3D cases. In MPAM for 2D cases, the support domain of every targeted particle is divided into four sub-domains and four rectangles. Then based on the local neighbour particles' configurations and particle number density, the targeted particle will be judged as free surface

particle or not. In MPAM for 3D cases, some necessary modifications have been made to simulate the 3D violent breaking waves, the auxiliary functions in the MPAM method have also been extended to 3 dimensional forms (Zhou & Ma, 2010). The support domain of a targeted particle, which is a sphere in 3D cases, is divided into eight sub-domains and six cylinders. Then again, based on the local neighbour particles' configurations and particle number density, the targeted particle will be identified as free surface particle or not. The details are follows:

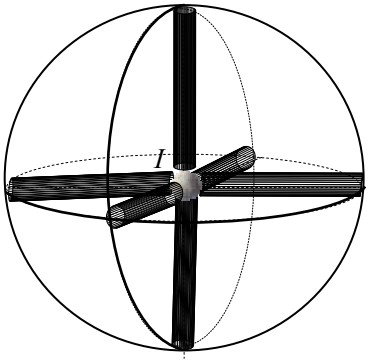


Fig. 4.4.1 Local domain at Node *I* for the definition of auxiliary function
(the inner sphere denotes integration domain of the particle; the outer sphere denotes the support domain; the 6 coloured cylinders have the same diameter as the inner sphere)

Similar auxiliary functions should also be defined in 3D MPAM. The first auxiliary function, which is the same as Eq. (4.2.1) for 2D cases, is defined by

$$fsp_a(I) = \begin{cases} 1, & NA \geq 1 \\ 0, & NA = 0 \end{cases} \tag{4.4.1}$$

where NA represents the number of free surface particles existing in the support domain (D_f) of Node I in previous time step. The second auxiliary function is now given by

$$fsp_b(I) = \begin{cases} 1, & NB = 8 \\ 0, & NB \leq 7 \end{cases}, \tag{4.4.2}$$

where NB represents the number of quadrants occupied by the neighbor particles of Node I in a local coordinate system originating at Node I, as shown in Fig. 4.4.1.

The third auxiliary function is changed to

$$fsp_c(I)=\begin{cases}1, & NC=6\\ 0, & NC\le 5\end{cases}, \tag{4.4.3}$$

where NC represents the number of colored cylinders in Fig. 4.4.1 in which there is at least one fluid particle.

If one of following conditions is met when they are checked sequentially, Node I is identified as a free surface particle.

(a) no inner particle in D_f except for I;

(b) $\beta_I < \beta$ and $fsp_a(I)=1$;

(c) $\beta_I \ge \beta$, $fsp_a(I)=1$ and $fsp_b(I)=0$;

(d) $\beta_I \ge \beta$, $fsp_a(I)=1$ and $fsp_c(I)=0$.

In 2D cases, β is equal to 0.97; but in 3D cases, the value of β is slightly adjusted to 0.9, which can give better results according to our numerical tests. The effectiveness of MPAM in 3D cases will be shown in later chapters related to 3D cases.

5. NUMERICAL SIMULATION OF DAM BREAKING

As a classic test case for violent free surface flows, dam breaking cases have been experimentally and numerically studied by many researchers. In practices, the dam breaking problems can be approximately regarded as a green water problem. The knowledge revealed by studying dam breaking benefits the understanding of the green water impact. This chapter presents the numerical simulations of dam breaking using the MLPG_R method. Both 2D and 3D cases are considered. The accuracy and convergence properties of the MLPG_R in such problems are of most concern. The parameters with a length scale are nondimensionalised by the initial water depth d and the time by $\sqrt{d/g}$.

5.1 Two dimensional dam breaking

A 2D dam breaking case has been presented in previous chapters for verifying some numerical techniques. More details will be presented here to show the overall accuracy of the MLPG_R methods. The problem definition is sketched in Fig. 5.1.1 with a rigid gate holding the water column initially. The vertical particle number Nz is chosen as 50 in this case, the total number of particle and water particle number are 1,551 and 1,250, respectively, and the time step is 0.008.

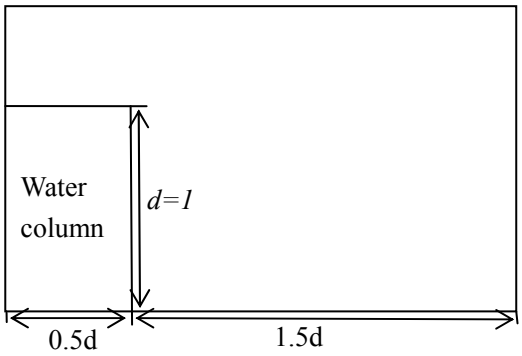


Fig. 5.1.1 Dam breaking: problem definition

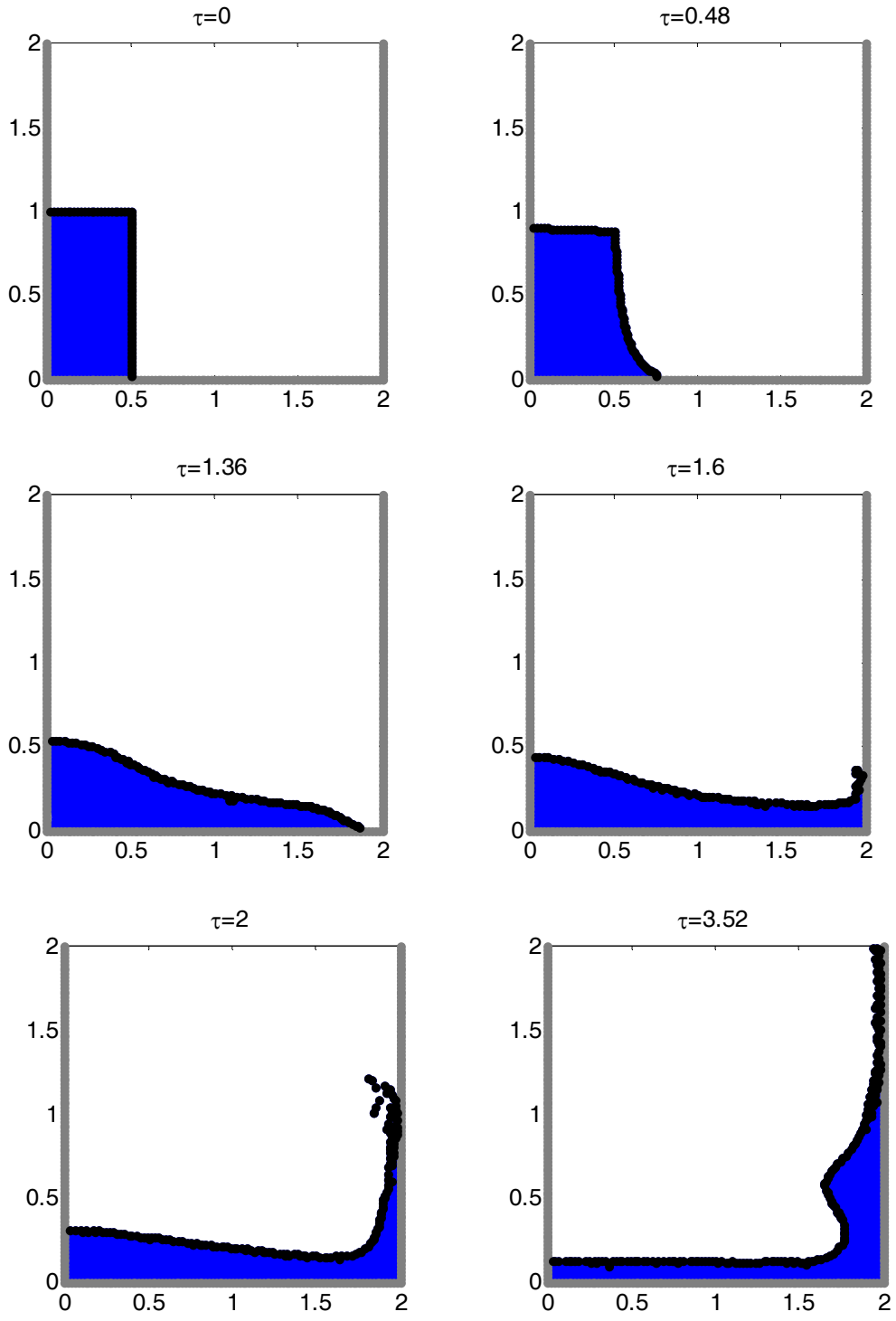


Fig. 5.1.2(a) Snapshots of free surface at different time steps
(grey particles: wall; the black ones: free surface particles)

Configurations of the free surface at different time steps are shown in Fig. 5.1.2. The water column begins to collapse when the rigid gate is suddenly removed at $\tau = 0$. At $\tau = 0.48$ and $\tau = 1.36$ the collapsing water is running on the bottom of the tank, then

impinge on the right wall just before $\tau = 1.6$. From $\tau = 1.6$ to τ =4, the water impinges on the right wall and goes up along the right wall forming a thin film attached to the vertical wall. The water column gradually lose its momentum and begin to come down at τ =3.52. The falling water jet hits the free surface after τ =4 and a reflected water jet is formed at τ = 4.8; the jet hits on the left wall at τ =5.36. The water goes up along the left wall after the impinging at τ =6.48. In order to show the efficiency of free surface particles identification method (MPAM), the free surface particles are also plotted separately using the black color. Through Fig. 5.1.2, one can clearly see that the free surface particles are tracked accurately regardless of how violent the free surface profile is.

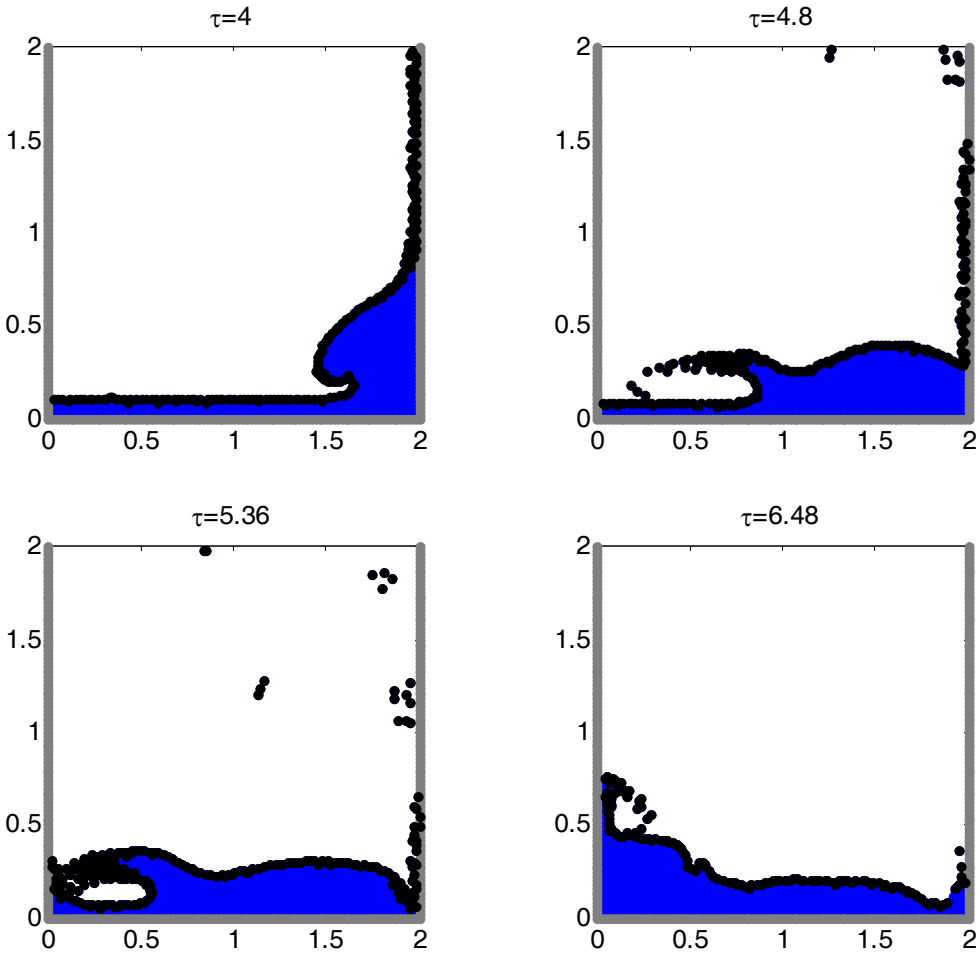


Fig. 5.1.2(b) Snapshots of free surface at different time steps
(grey particles: wall; the black ones: free surface particles)

The predicted horizontal location of the free surface along the bottom wall are compared

with the experimental data from Martine and Moyse (1952) & Koshizuka et al. (1995) and other numerical results, i.e. Miyata & Park (1995), Hirt & Nichols (1981) and Lv et al (2010). For the purpose of comparisons, they are all plotted in Fig. 5.1.3. As can be seen, the present results agreed well with other two numerical ones, i.e. Miyata & Park (1995) and Hirt & Nichols (1981). Furthermore, the three numerical results seem to be different from the experimental data. This is mainly due to two facts: one is that the initial friction of right-hand side of the water column is different between the experiments and simulations when the rigid gate is suddenly removed; the other one is that the effect of air on the free surface is not taken into account in the numerical modeling. Obviously, the results of Lv et al (2010) are much closer to the experimental data thanks to the consideration of gaseous phase with complex numerical methods.

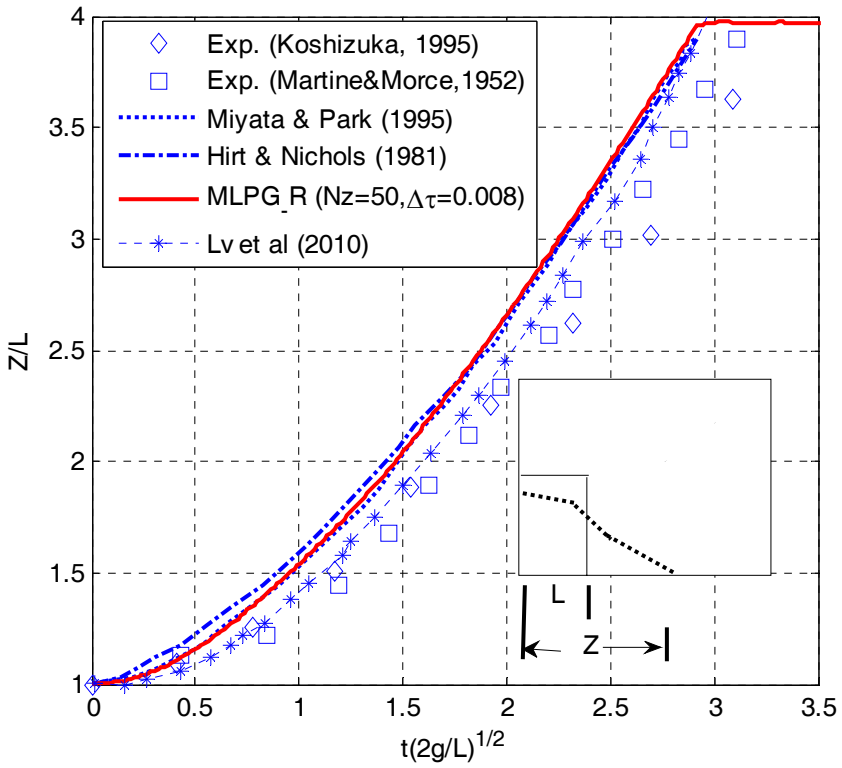


Fig. 5.1.3 Comparing the motion of the leading edge with other simulated results and experiment data

5.1.1 Convergent investigation on different values of $\Delta x / \Delta \tau$

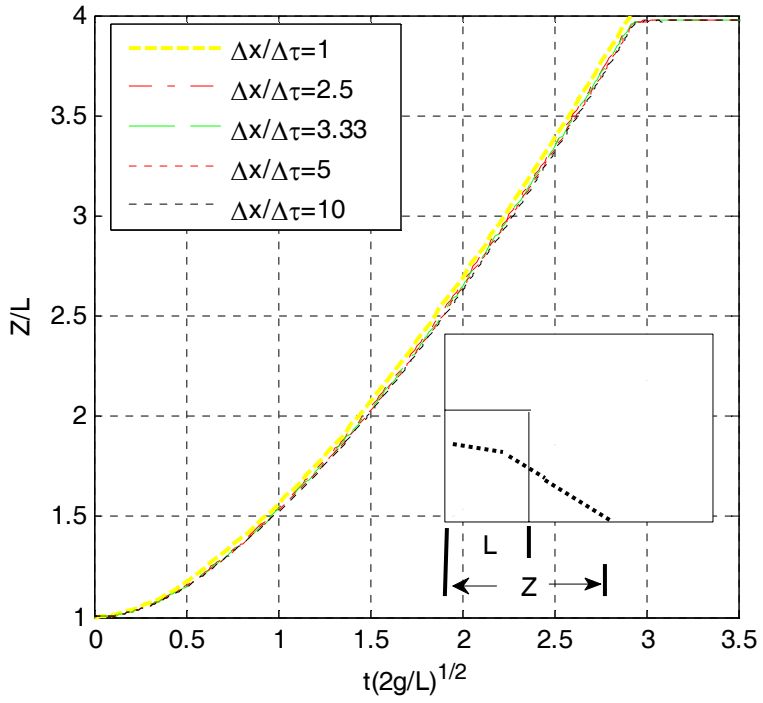


Fig. 5.1.4 Comparing the motion of the leading edge corresponding different values of $\Delta x / \Delta \tau$

Convergent tests are now shown on the case in Fig. 5.1.4. Following the previous experience at Chapter 4 and Chapter 5, the different ratios of Δx to $\Delta \tau$ are investigated. In the investigations, the representative distance Δx is chosen as 0.02, corresponding to the particle number along *z*-direction (*Nz*) being 50. The values of $\Delta x / \Delta \tau$ are chosen as 1, 2.5, 3.33, 5 and 10, respectively. The motion of the leading edge corresponding to different values of $\Delta x / \Delta \tau$ is plotted in Fig. 5.1.4. One can see that there is no obvious difference between these results except minor difference existing in the results of $\Delta x / \Delta \tau = 1$ compared to others. Furthermore, when $\Delta x / \Delta \tau = 1$, there is a non-physical phenomenon happening, which some water particles penetrate the solid wall when the water jet hitting the right wall shown in Fig. 5.1.5. The probable reason may be that the time step is a little bigger; the water particles cannot be effectively stopped by the interaction between the wall particles and water particles. Based on these reasons, one may conclude that results of $\Delta x / \Delta \tau \geq 2.5$ are considered to

model the process well and are convergent. The free surface profiles are also numerically investigated to show how the results are affected by the values of $\Delta x / \Delta \tau$. The comparisons are plotted in Fig. 5.1.6. From the time τ=1.2 to τ=4, i.e. from the water column beginning to collapse to water column coming down along the right side wall after hitting the right wall, the four sets of numerical results are in good agreement with each other. But after that from the last two sub-figures in Fig. 5.1.6, obvious difference can be observed between the results of $\Delta x / \Delta \tau$=1 and others results, the jet shape and location are different and a little lower than others at τ=4.8. Hence based on above numerical investigations, similar conclusion can be drawn that the results of $\Delta x / \Delta \tau \geq 2.5$ may be considered as convergent for simulation of free surface profiles in the 2D dam breaking case.

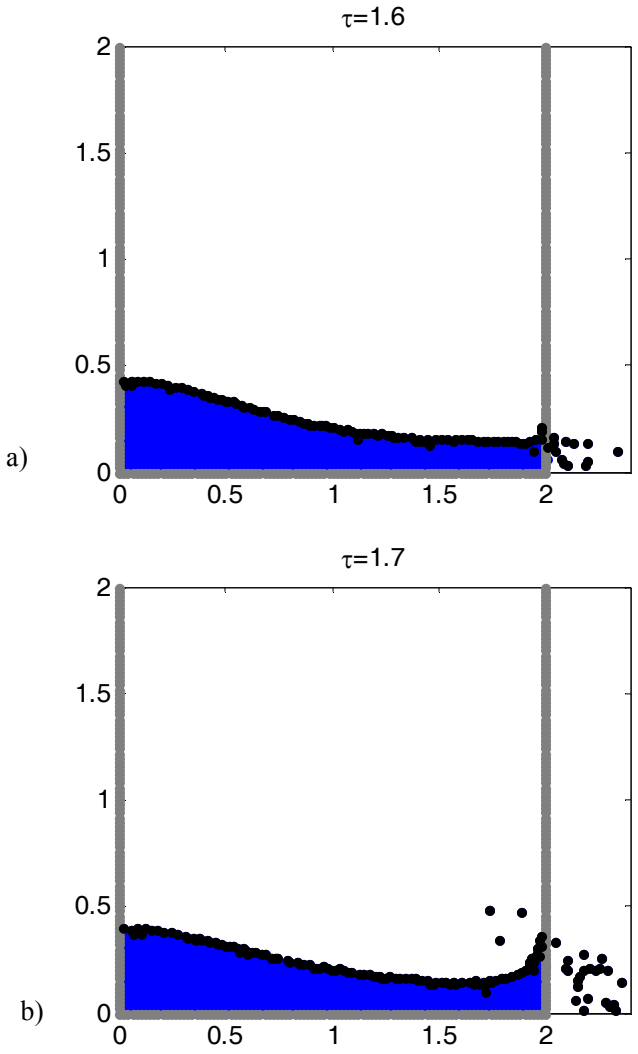


Fig. 5.1.5 Non-physical occurrence with $\Delta x / \Delta \tau$=1

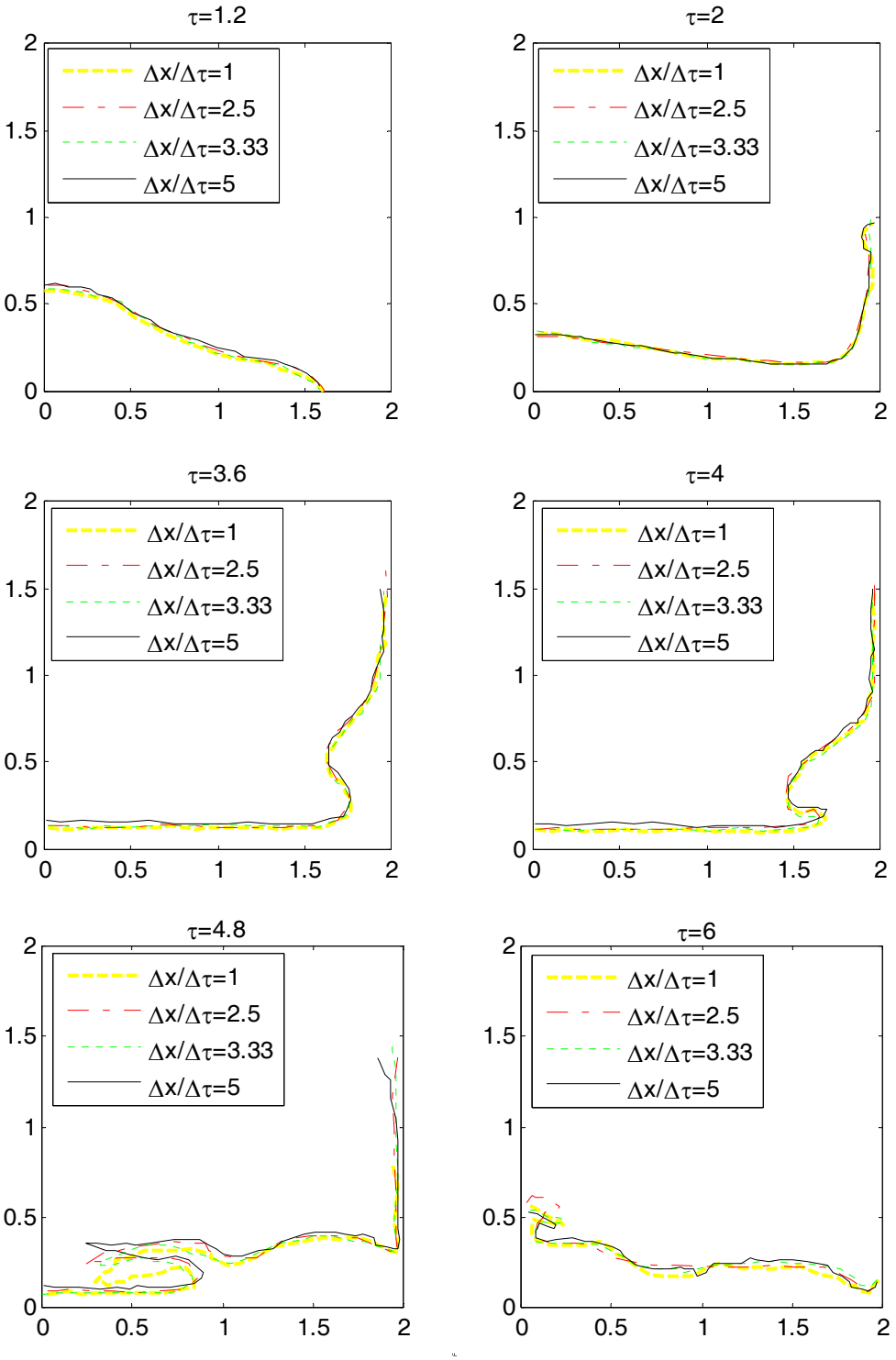


Fig. 5.1.6 Comparing free surface profiles corresponding to different values of $\Delta x / \Delta \tau$

5.1.2 Convergent investigation on different space increments Δx

The effect of Δx on the convergence properties of the MLPG_R in modeling the case shown in Fig. 5.1.3 is also investigated. In the investigation, constant $\Delta x / \Delta \tau$ =2.5 is chosen.

The values of Δx are 0.04, 0.025, 0.02 and 0.0167, respectively. The corresponding particle numbers along z-axis (Nz) are 25, 40, 50 and 60, respectively.

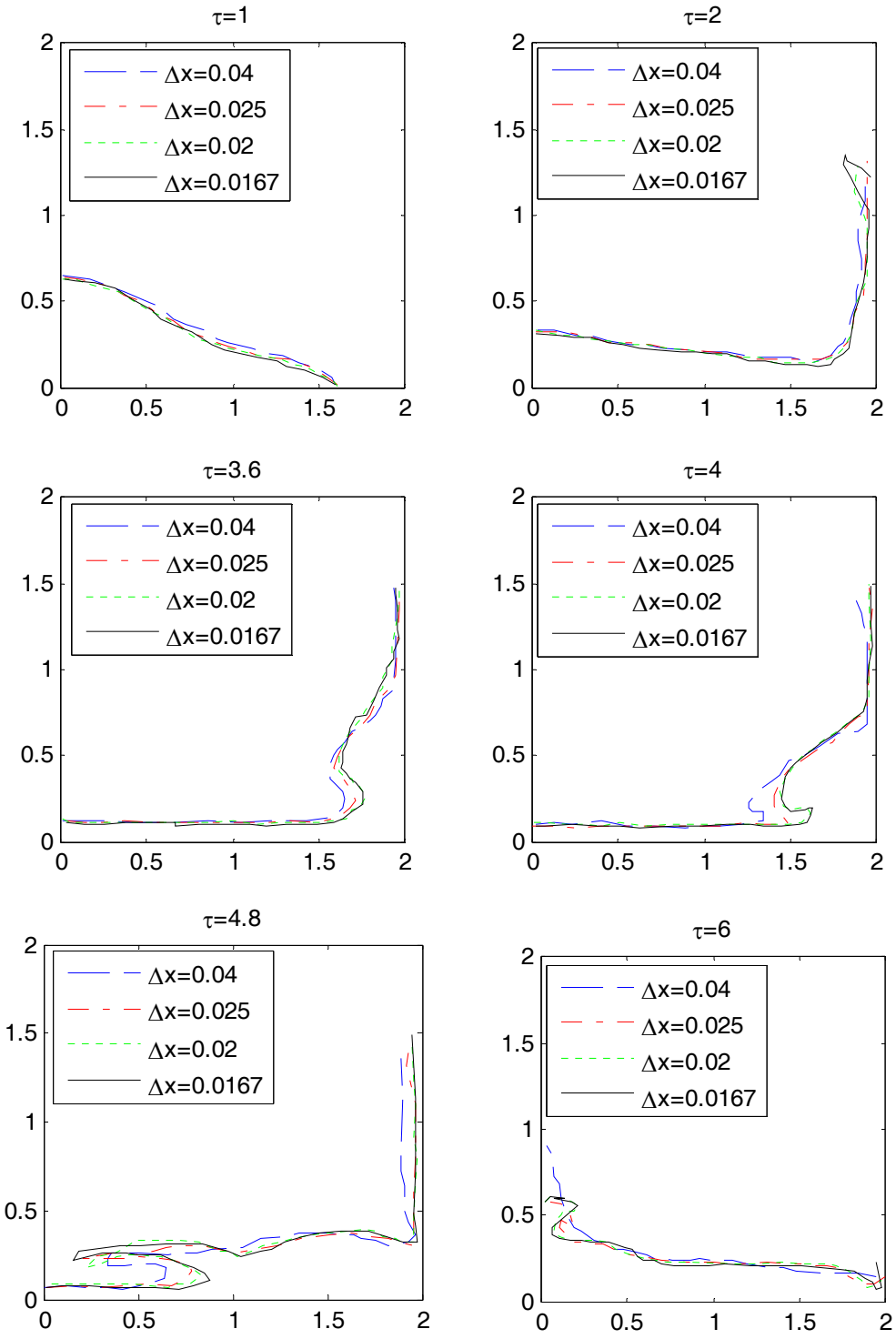


Fig. 5.1.7 Comparing free surface profiles corresponding to different values of Δx with $\Delta x / \Delta \tau = 2.5$

Six instant snapshots of the free surface profiles are shown in Fig. 5.1.7. It can be seen that the four sets of results are very close to each other, although there is some small differences between the results of Δx =0.04 and others after the water column beginning to come down along the right wall. Hence, $\Delta x \le 0.04$ with $\Delta x / \Delta \tau \ge 2.5$ can be used to find the satisfied results for the free surface profiles in the 2D dam breaking case.

5.2 Three-dimensional dam breaking

Similar to 2D dam breaking problem, 3D cases have also been studied experimentally and numerically by many researchers, such as Kleefsman et al. (2005). For convenience of comparing numerical results with experimental data, the parameters presented in this subsection are all dimensional. In the case, a rectangle tank (Fig. 5.2.1) with dimensions of $3.22 \times 1 \times 1$ m is used. A water column, whose length, width and height are 1.228m, 1m and 0.55m, respectively, is initially held on the right side of the tank by a rigid gate. A box-type obstacle is mounted facing the water column. The distance between the centre of the obstacle and the left vertical wall of the tank is 0.744 m. On the surface of the obstacle, various pressure transducers are installed. The locations of those pressure transducers are illustrated in Fig. 5.2.2. In the experiment by Kleefsman et al. (2005), four probes (whose locations are H1, H2, H3 and H4 shown in the Fig. 5.2.1 and Fig. 5.2.3) are used to record the time history of the wave height. The geometry of the system is described in Fig. 5.2.3. Based on the convergence investigation of 2D dam breaking case and considering 3D computational efficiency, in this case, the spatial distance between two particles are taken as 0.022m (N_z=25, dimensionless space increment $\Delta x = 0.04$) and the time step *dt* as 0.00189s (equivalent dimensionless time step $\Delta \tau = 0.008$) is used. The total number of water particles and total number of particle numbers are 63,000 and 87,589, respectively. The simulation time step is 4,200, yielding duration up to 6 seconds. The total CPU time is about 31 hours in normal PC with Inter (R) Core(TM) 2 Duo CPU E7500 @ 2.93GHz (one CPU is used) and 2.93G RAM in Microsoft system.

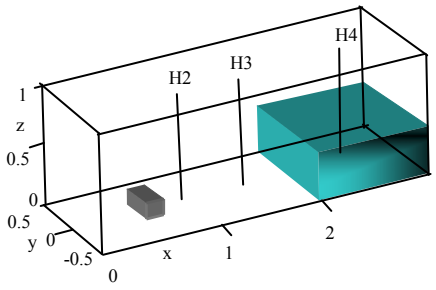


Fig. 5.2.1 Schematic view of the tank and the obstacle (unit: m).

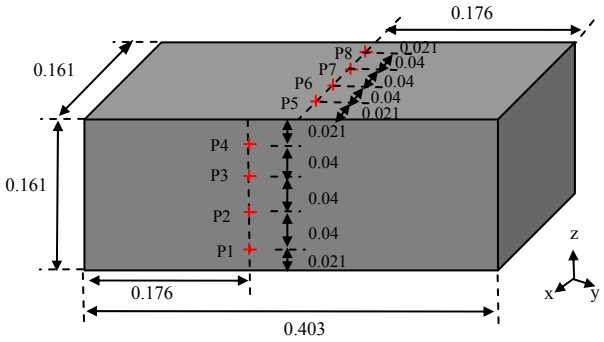


Fig. 5.2.2 Details of the obstacle and the pressure transducers (unit: m)

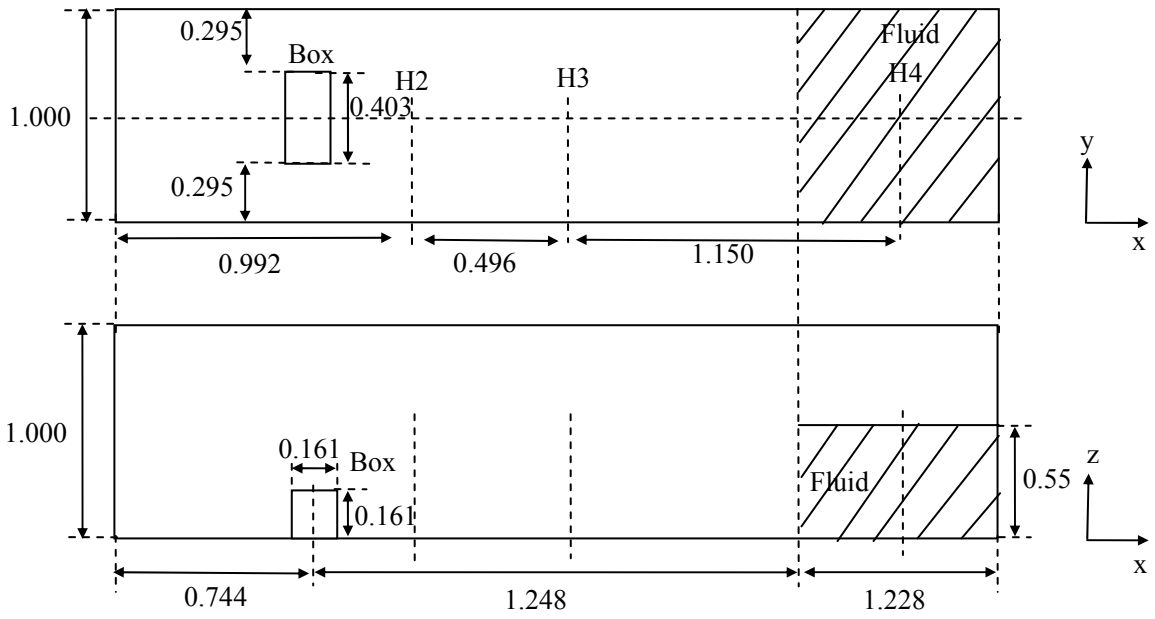


Fig. 5.2.3 General description of the system: top (top picture) and side (bottom picture) views

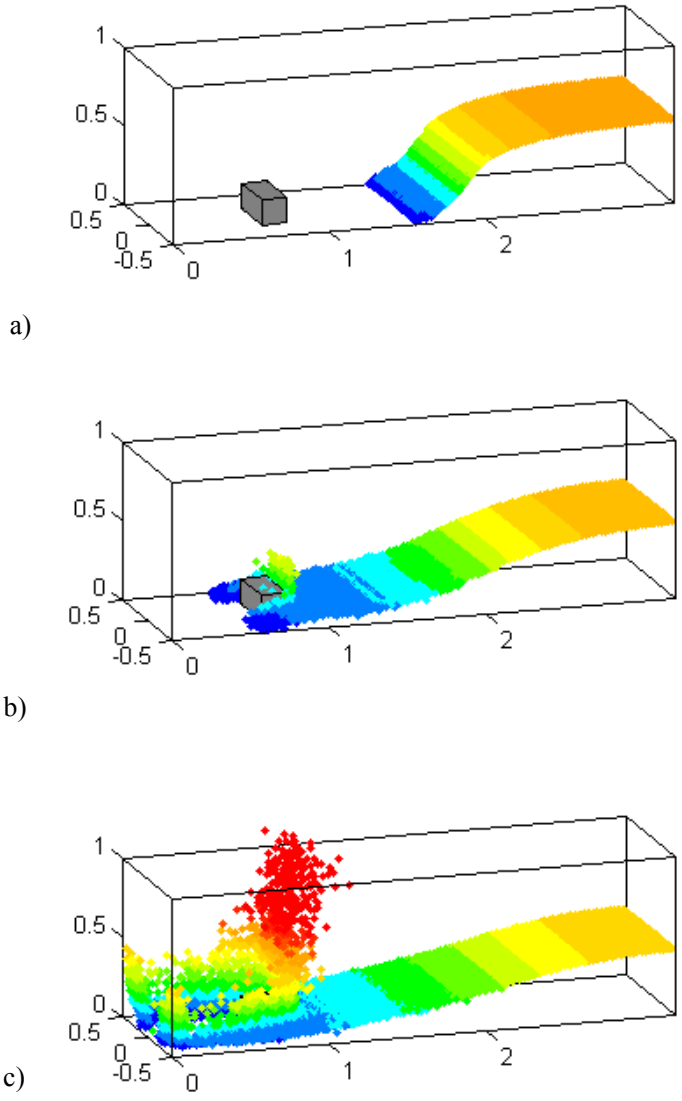
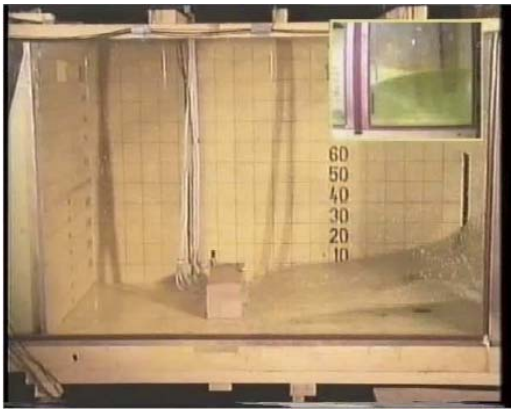


Fig. 5.2.4 Snapshots of the free surface at (a) t=0.227s (b) t=0.493s and (c) t=0.701s.

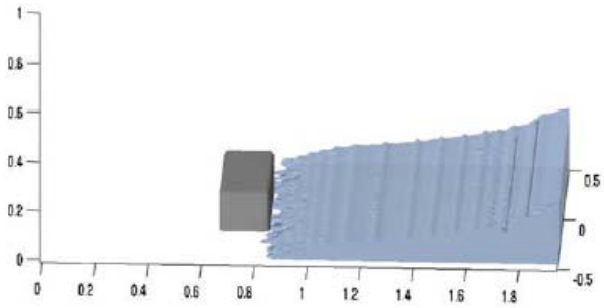
The numerical simulation starts when the rigid gate is released instantaneously. Some snapshots of the free surface profiles are shown in Fig. 5.2.4 which demonstrates typical stages of the wave evolution. In this figure, different color highlights different water height. When the collapsed water column hits the obstacle with very high speed (Fig. 5.2.4b), an impact occurs on the surface of the obstacle. After that, the water near the front side of the obstacle (facing impact) is reflected by the obstacle causing relatively higher elevations (Fig. 5.2.4c). The free surface profiles at two instants $t \approx 0.4$s and $t \approx 0.56$s are compared with the experimental photographs and the numerical results (with Finite Volume Method) by Kleefsman, et al. (2005) in Fig. 5.2.5 and Fig. 5.2.6. Only the free surfaces near the obstacle

are plotted for clarity. From the figures, it is found that the present results are very similar to the Kleefsman's numerical ones and both the numerical results are also similar to the experimental profiles.

a)



b)



c)

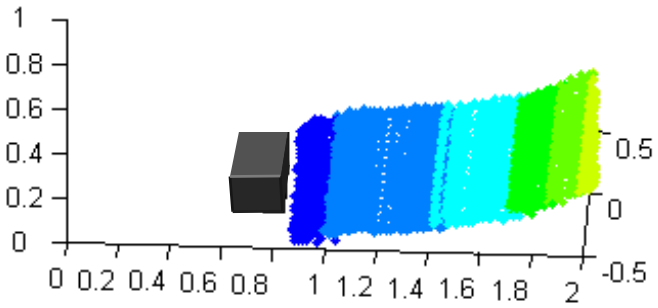
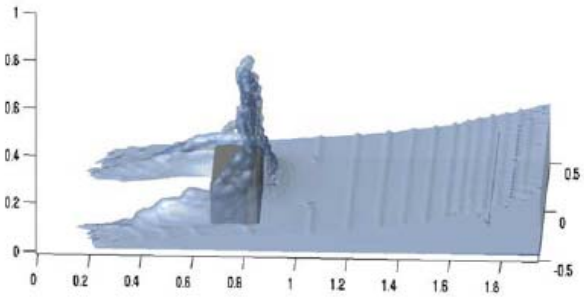


Fig. 5.2.5 Comparison of the free surface profiles at $t \approx 0.4$s (a: experiment, b: FVM numerical results, both results from Klessfsman, et al, 2005; c: present method)

a)

b)



c)

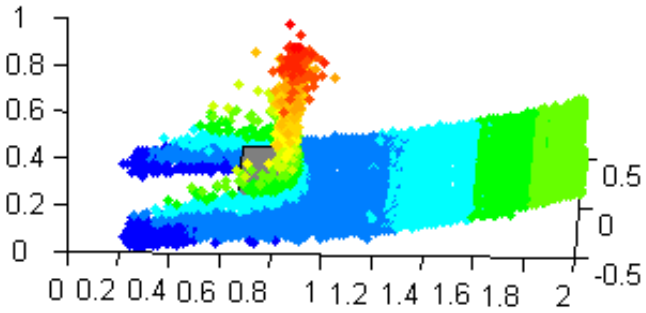


Fig. 5.2.6 Comparison of the free surface profiles at $t \approx 0.56$s (a: experiment, b: FVM numerical results, both results from Klessfsman, et al, 2005; c: present method)

The comparisons shown in Fig. 5.2.5 and Fig. 5.2.6 demonstrate an acceptable accuracy of the MLPG_R method in predicting the spatial distribution of the free surface elevation at specific time steps. The comparison is also made for the time histories of the wave height recorded at different positions of the tank in order to explore its accuracy in modeling temporal distribution of physical quantities. Some results are shown in Fig. 5.2.7 for the wave height histories recorded by Probes H2, H3 and H4. It is observed from Fig. 5.2.7 that the numerical results correlate very well with the experimental data. However, a visual

difference exits. Considering complexity of this problem, these agreements can be considered as acceptable.

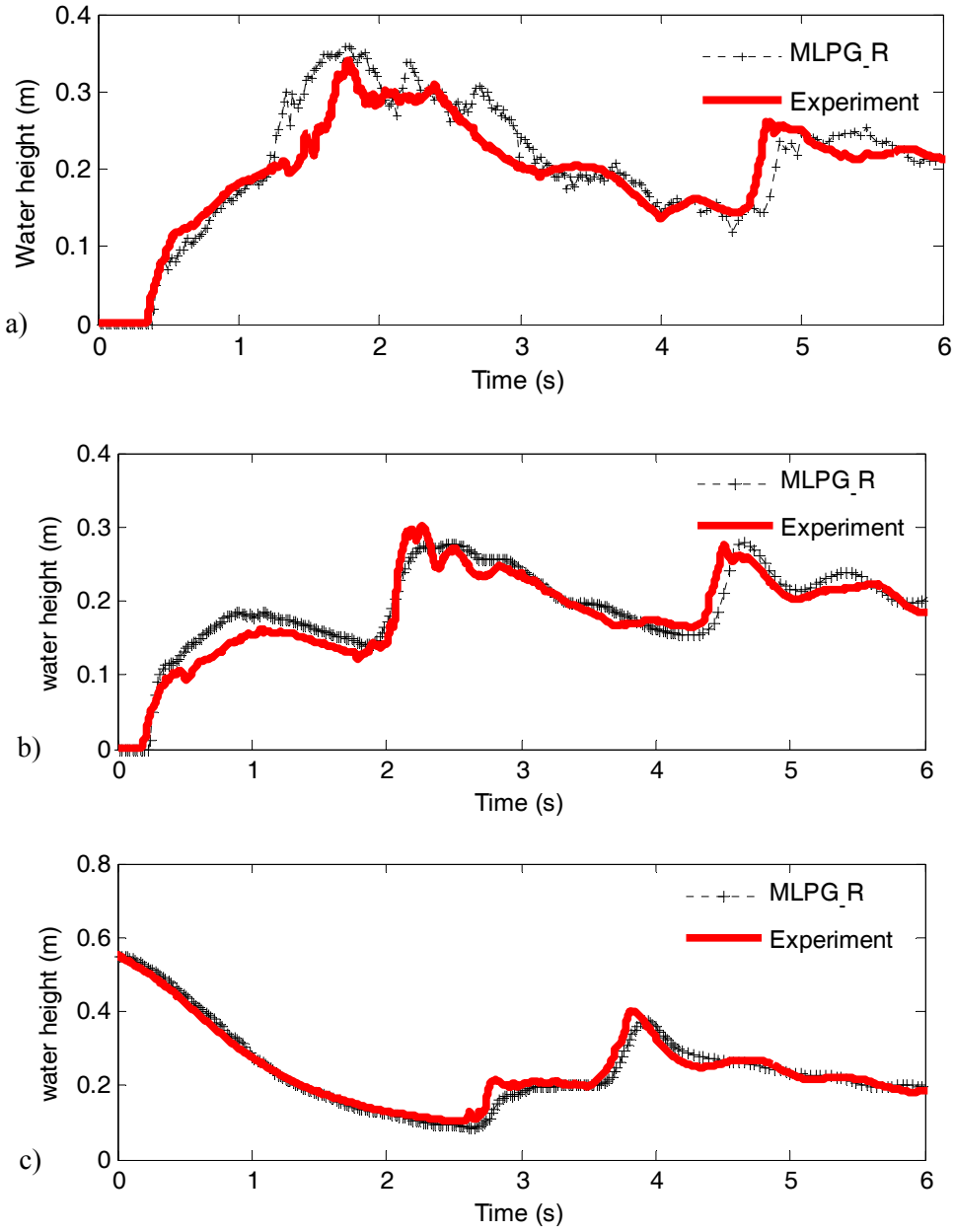


Fig. 5.2.7 The time histories of free surface elevations at (a) for H2, (b) for H3 and (c) for H4

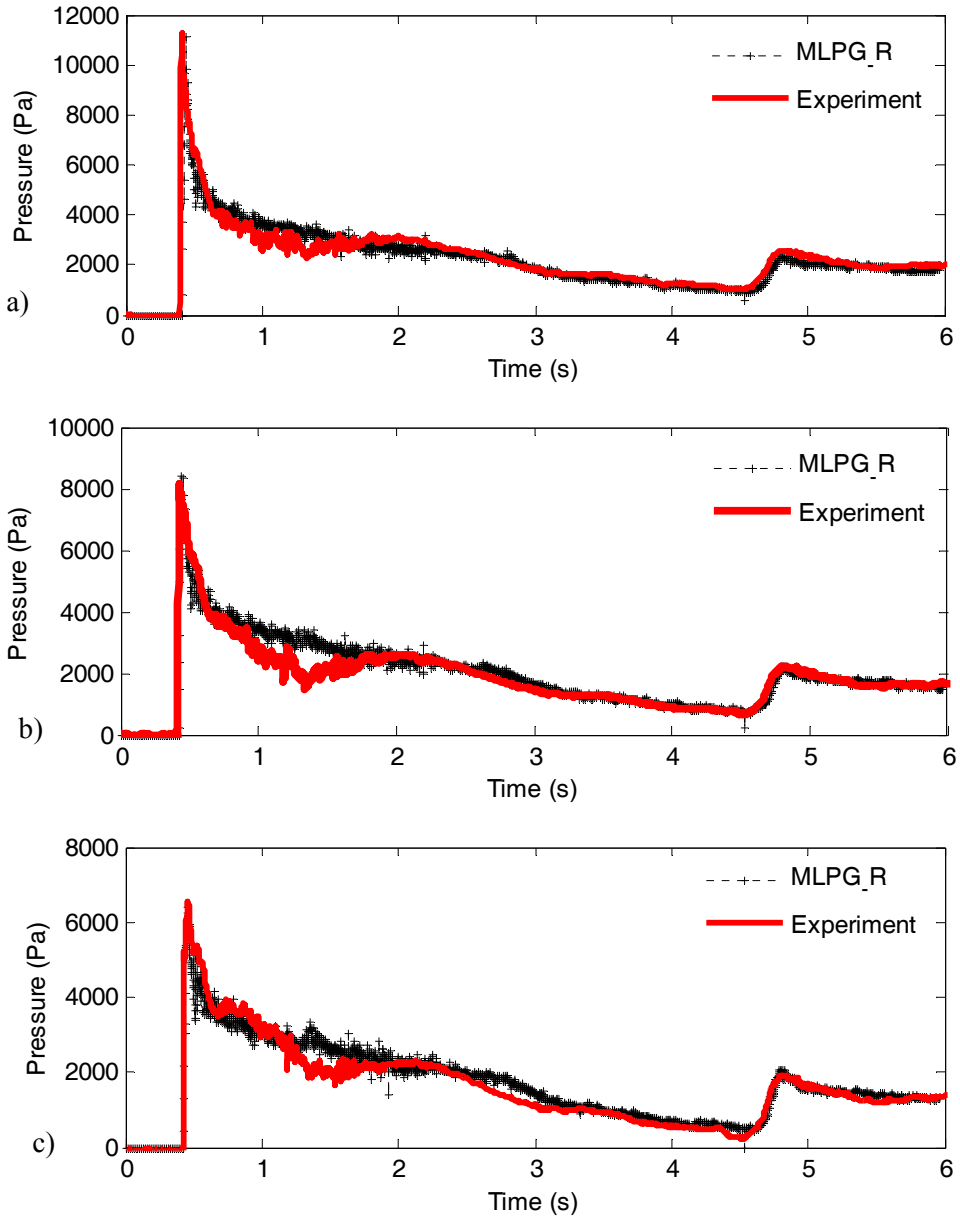


Fig. 5.2.8 Pressure time histories at (a) for P1, (b) for P2 and (c) for P3

Apart from the wave profiles, another important quantity is the pressure. It is also investigated. Fig. 5.2.8 displays the time histories of pressure acting at three points on the front side of the obstacle shown in Fig. 5.2.2. For the purpose of comparison, the corresponding experimental data (Klessfsman, et al, 2005), recorded by the pressure transducers located at the same places, are also plotted. From the Fig. 5.2.8, one can notice that there are two pressure rises in each curve. The first pressure rise is generated due to the primary impact on the obstacle after the water column collapses while the second rise occurs when the returning wave front hits the obstacle again from the right after it is reflected by the

left and then right walls. The first rise is extreme large and last for a very short period, demonstrating the typical impulsive feature caused by impact. The second rise is not so large and grows relatively slowly. One can also see that the shapes of the numerical curves are largely similar to the experimental ones. Particularly, the numerical method does not only predict the first pressure rise well, but also give a good estimation to the second rise of experimental data. The comparisons between the numerical results and the experimental data or numerical results from other methods further demonstrate the capability and the accuracy of the MLPG_R method in modeling wave height and impact pressure caused by violent water waves.

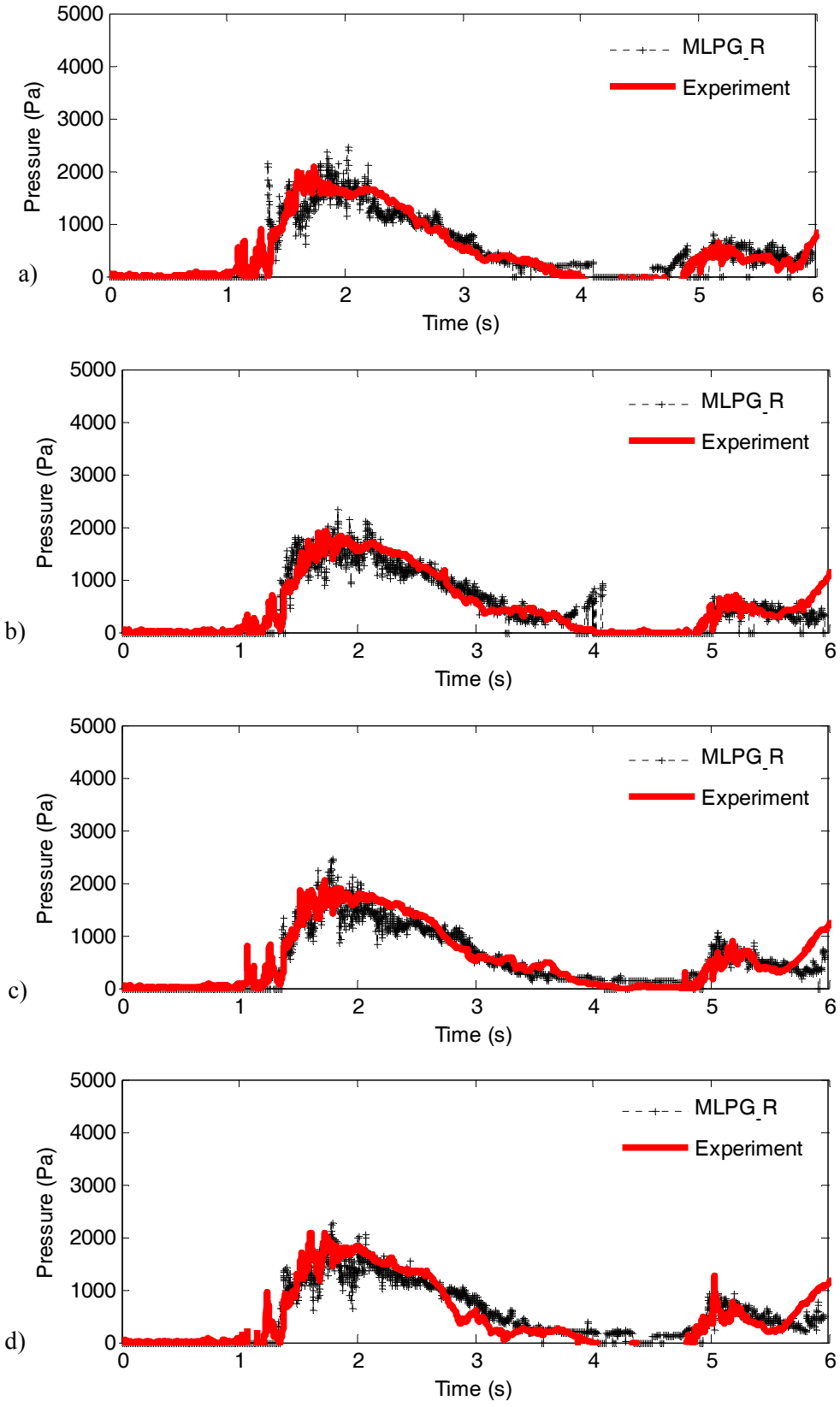


Fig. 5.2.9 Pressure time histories at (a) for P5, (b) for P6, (c) for P7 and (d) for P8

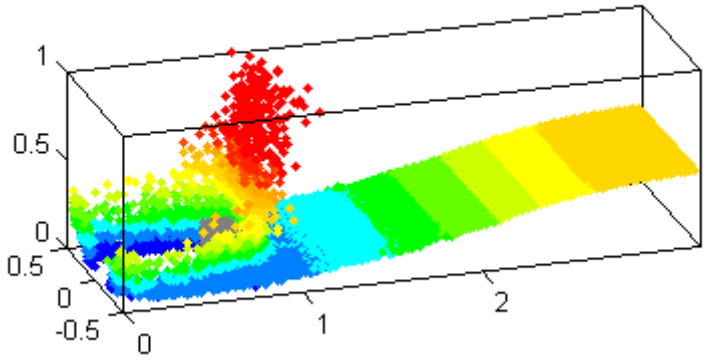
Our attention is also paid to the pressure recorded on the top of the obstacle. Some results are displayed in Fig. 5.2.9 with the corresponding experimental data, which show that good agreements have been achieved. However, it is found that the curves are not as smooth as those shown in Fig. 5.2.8. To mitigate the fluctuation, the numerical smooth technique suggested by Ma (2005b) may be used, in which the water particles velocity are updated by

$$u_{i,new}^{n+1} = (1-\kappa)u_i^{n+1} + \kappa\sum_{j=1}^{N}\Phi_j u_j^{n+1} \tag{5.2.1}$$

where $u_{i,new}^{n+1}$ is the new velocity of particle *i* after being smoothed, u_i^{n+1} and u_j^{n+1} are the velocities of particle *i* and particle *j* at the (n+1) time step, respectively, and particle *j* are those influencing particle *i,* determined by the weight function Eq. (3.3.17). κ is an artificial coefficient; *N* is the total neighboring particle number. Φ_j is the shape function implemented by the moving least square method (the details can be seen in the Eqs. (3.3.12) ~ (3.3.18)). Based on the numerical tests, κ is chosen as 0.1.

In order to show the effect of the smooth technique, two snapshots at the same instant but from different simulations, one without considering the smooth technique and the other applying the smooth technique, are plotted in Fig. 5.2.10 for comparison. From the Fig. 5.2.10(a), one may notice that the splashed water particles are very scattered if the smooth technique is not applied. While the smooth technique seems to make the splashed water become more regular (Fig. 5.2.10 (b)). Other than the free surface, the pressure time histories of the top wall particles in Fig. 5.2.2 are also investigated and the results are plotted in Fig. 5.2.11. From this figure, one can see that the pressure curves become much smoother than those in Fig. 5.2.9, but the pressure peak in the time histories is a little lower than the experimental data. The reason may be that the some energy is dissipated because of the numerical viscosity caused by the smoothing.

(a)



(b)

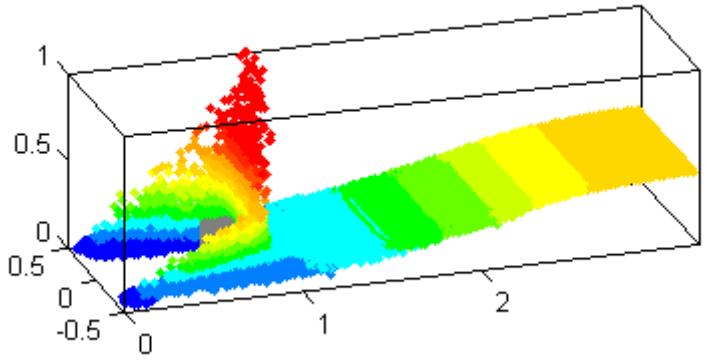


Fig. 5.2.10 Comparison of free surface profile at the t=0.68s
(a) results without smoothing, (b) results with smoothing

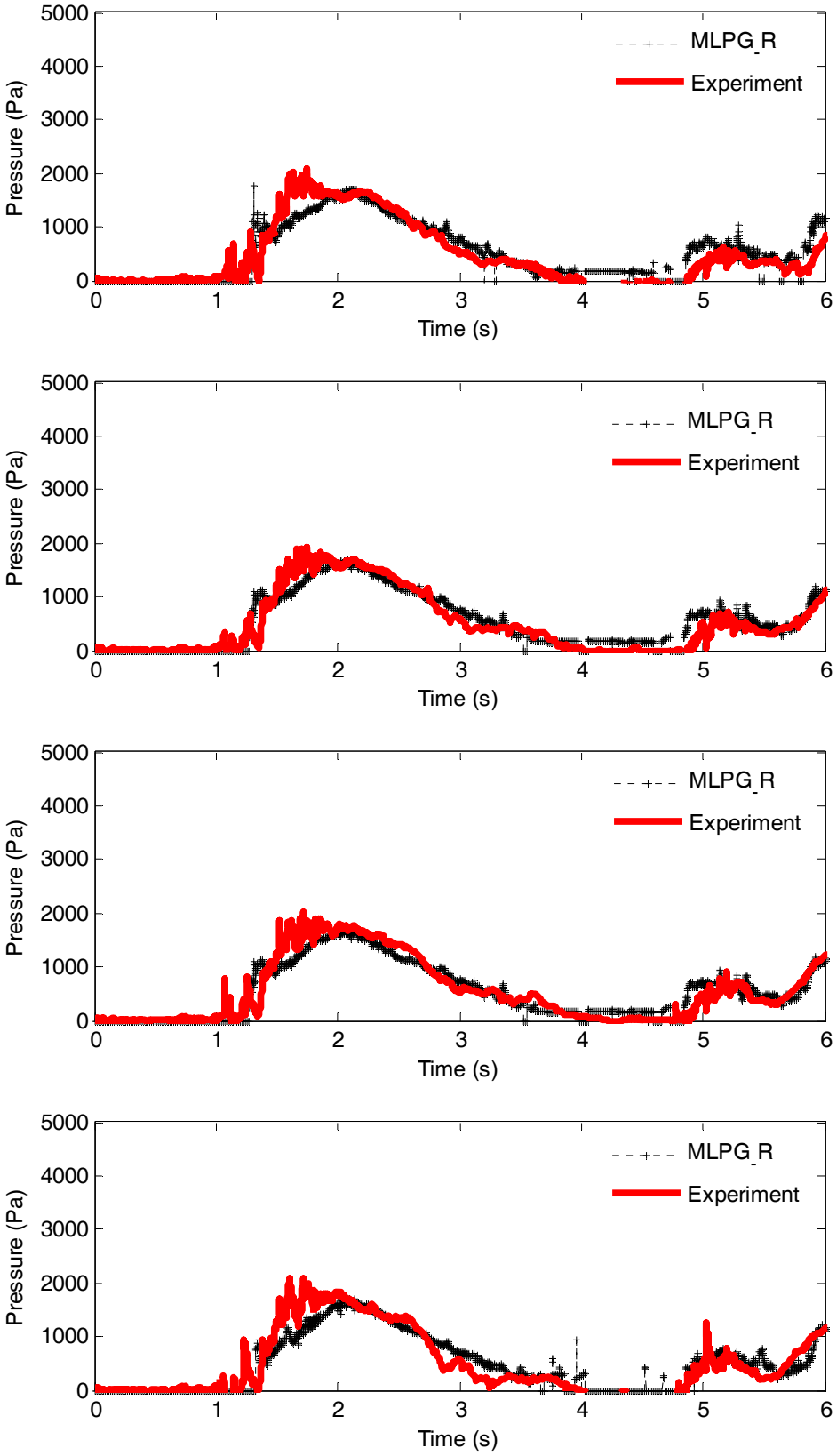


Fig. 5.2.11 Pressure time histories at (a) for P5, (b) for P6, (c) for P7 and (d) for P8

6. BREAKING WAVES OVER NON-FLAT SEABED

In this chapter, the breaking wave over non-flat seabed will be simulated by using the MLPG_R method combining the newly developed numerical techniques, e.g. the scheme to identify free surface particles. The schematic view of the tank is shown in Fig. 6.0.1 for 2D cases and a Cartesian coordinate system is adopted where the *x*-axis is positive pointed from wavemaker to the beach, the *z*-axis is positive upwards, the origin of the coordinate is located on the free surface at the toe of slope unless mentioned otherwise. The water waves are generated by a piston-type wavemaker mounted at the left side of the numerical wave tank. Similar to the cases presented in Chapter 5, all the parameters with a length scale are nondimensionalised by the water depth d and the time by $t \to \tau\sqrt{d/g}$, unless mention otherwise.

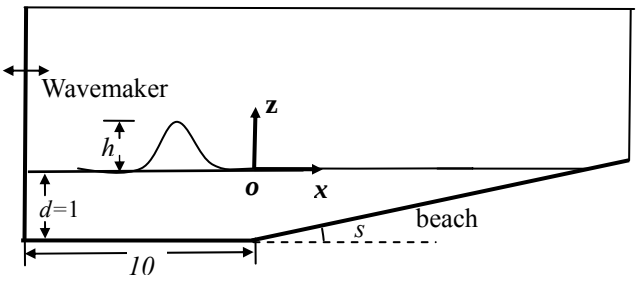


Fig. 6.0.1 Illustration of model set-up for the solitary wave

In mesh-based methods, the convergence property is related to the element (or cell size) size in meshes or grids and time step ($\Delta\tau$). In the MLPG_R method, there is no mesh at all. Therefore, element (or cell) size is not relevant. A similar quantity is the distance between particles. Unfortunately, the distance continuously varies when modeling water waves. To bypass the problem, the initial distance of two adjoining particles is chosen as a representative

distance. As the initial distribution of particles is uniform in the study, the representative distance is equal to the difference in the *x*- or *z*- coordinates of two adjoining particles, which is denoted by Δx in the following discussion. The number of particles along z-direction is represented by N_z.

The two parameters (Δx and $\Delta \tau$) can be considered directly for studying the convergence property. They can also be replaced by other two parameters, such as $\Delta x/\Delta \tau$ and Δx. The second way is chosen herein, because the later is related to the well-known Courant number that has a form of $\Delta \tau = c_t \Delta x$ ($\Delta t = c_t \, \Delta x/\sqrt{gd}$ in dimensional form) with c_t being a constant, as used by a number of researchers studying wave waters, for instance, Grilli, Guyenne and Dias (2001) and Yan and Ma (2009).

According to the linear wavemaker theory (Goring, 1978), the solitary wave can be generated by specifying the wavemaker motion as follows,

$$S(\tau) = \frac{H}{\kappa}(\tanh \chi + \tanh \kappa\lambda) \tag{6.0.1}$$

$$\chi = \kappa\left[c\tau - S(\tau) - \lambda\right] \tag{6.0.2}$$

$$\lambda = \log\left(1/\varepsilon_z\right)\left(1+\sqrt{1-\varepsilon_z}\right)/\kappa \tag{6.0.3}$$

where $S(\tau)$ is the displacement of the wavemaker; H is the solitary wave height; $\kappa = \sqrt{3H/4}$; $c = \sqrt{1+H}$ is the celerity; ε_z is a coefficient which is assigned to be 0.002 in this study.

6.1 2D breaking waves on a slope

The case considered here is about the propagation of solitary waves over a beach with a slope of 1:15. The experimental data for this case are available in Li and Raichlen (1998). Its experimental set-up is similar to that illustrated in Fig. 6.0.1, in which the distance between the wavemaker and the toe of beach is 10 and the height of the solitary wave is *h* is 0.45. In order to focus on the overturning and wave breaking process, which takes place over the beach, it is assumed that the instant corresponding to the wave crest being at the toe of the beach is $\tau = 0$ for this case.

6.1.1 Convergence investigation on different value of $\Delta x/\Delta\tau$

Firstly, how the ratio, $\Delta x/\Delta\tau$, affects numerical results will be investigated. For this purpose, the representative distance, Δx, is taken as 0.05, corresponding to *Nz* =20, which yields the total particle number of 5893. The values of $\Delta x/\Delta\tau$ are chosen as 2.5, 5.0, 7.0 and 9.0, respectively. The propagation of the solitary wave is illustrated in Fig. 6.1.1 by snapshots at several instants, which are obtained by using different values for $\Delta x/\Delta\tau$. For the purpose of comparison and validation, the experimental data from Li and Raichlen (1998) are also plotted together. It can be seen that at about $\tau = 9.29$, the wave crest becomes very steep; then at $\tau = 9.87$, a plunging jet starts to be formed; and after this, the jet is moving forward and nearly impacts on the other part of the water surface in its front at time $\tau = 10.73$. There is no experimental data after this instant. For validation purpose, we only present the results until this instant here.

It can be seen that the numerical wave profiles generally agree well with the experimental ones except for those corresponding to $\Delta x/\Delta\tau = 2.5$. In addition, the numerical results for $\Delta x/\Delta\tau = 5$ and $\Delta x/\Delta\tau = 7$ are a bit closer to the experimental data than that for $\Delta x/\Delta\tau = 9$. That seems to imply that for a certain value of Δx, the time step should not be too large or too small. It is easy to understand why the time step should not go beyond the upper limit but the reason why it is also subjected to the lower limit from the point of view of accuracy needs to be further studied. Nevertheless, one should not choose too small time step in practice in order to save computational time if possible and so the lower limit may not be of a huge concern.

It is worth noting that Yan and Ma (2009) and Grilli, Guyenne and Dias (2001) both have concluded that one may take $c_t = 0.4$ in $\Delta\tau = c_t\Delta x$ to simulate 2D overturning solitary waves using fully nonlinear potential theory for the finite element and boundary element simulations, respectively, which corresponds to $\Delta x/\Delta\tau = 2.5$. This investigation seems to suggest that the time step for the MLPG_R method based on the NS equation should be smaller (about half in this case) than that in the methods for potential theory to achieve

similar accuracy. The information is useful because it helps researchers to guess the suitable time step a prior from experience in using the potential theory.

6.1.2 Convergence investigation based on different value of Δx

Secondly, the convergence properties against the value of Δx are also investigated. For this purpose, three different values for Δx are chosen. They are 0.05, 0.04 and 0.033, respectively. $\Delta x/\Delta\tau = 5$ is used for all cases. The corresponding numerical results are plotted in Fig. 6.1.2 together with the experimental data of Li and Raichlen (1998). This figure indicates that all these results up to $\tau = 10.73$ have an acceptable agreement with the experimental one. It also indicates that $\Delta x = 0.05$ is sufficient for the results in Fig. 6.1.1.

Nevertheless, it does not mean all the values are equally good beyond that instant. In order to show this, some snapshots for later instants are plotted in Fig 6.1.3, which illustrates the wave profiles after the jet impacts on the front part of the water surface, called as post-breaking stage. As can be observed, another jet is formed and a cavity behind the jet appears after the impact. It is also observed that the profiles for $\Delta x = 0.04$ (Nz=25) and $\Delta x = 0.033$ (Nz=30) are quite similar but there is a significant discrepancy between the one for $\Delta x = 0.05$ and others. Although the experiential data is not available for the profiles in these instants, one may deduce that $\Delta x = 0.05$ may not be sufficiently small to model the post-breaking waves. In other words, more particles are required to model the waves in this stage.

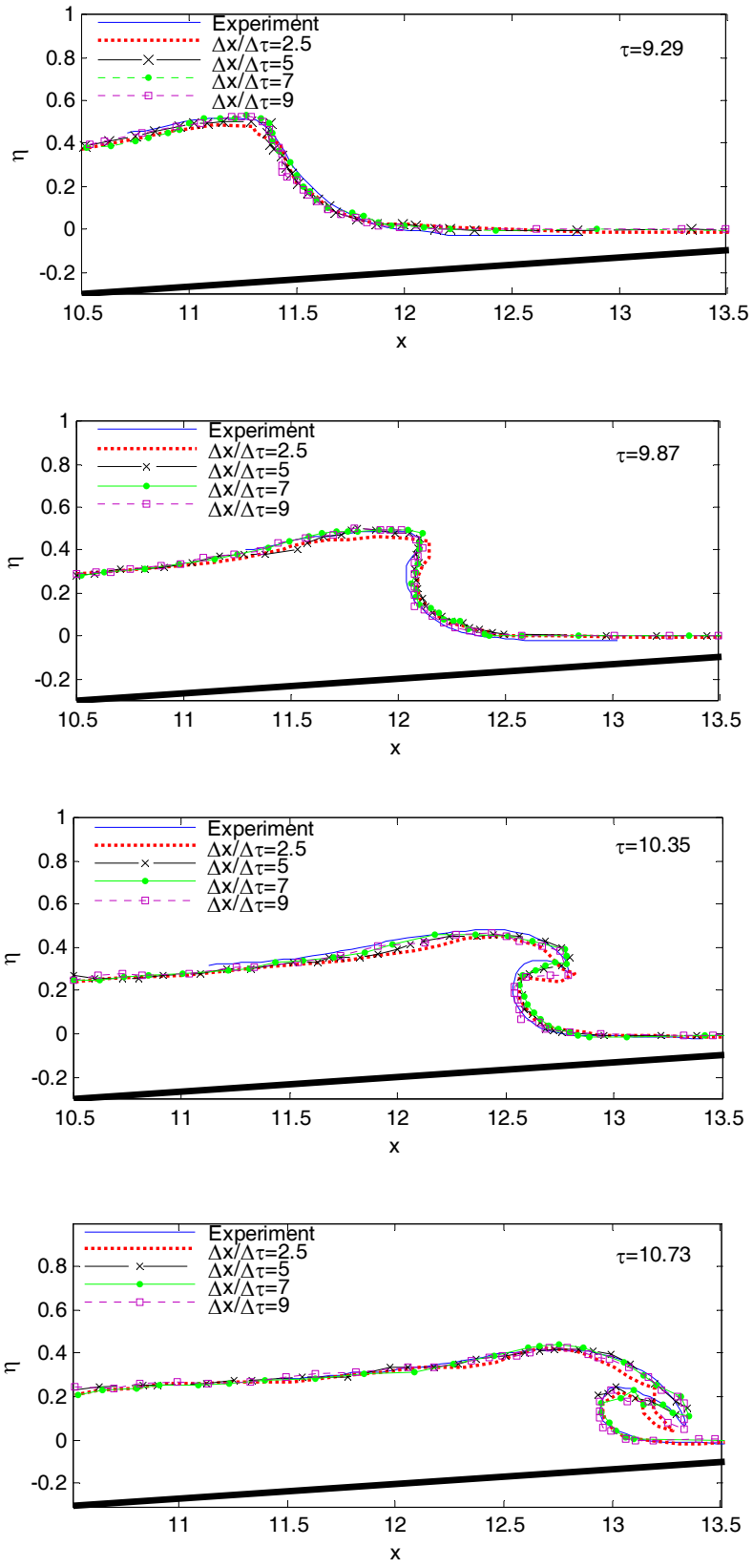


Fig. 6.1.1 Comparison between experimental wave profiles (Li and Raichlen, 1998) and numerical results for different values of $\Delta x/\Delta \tau$

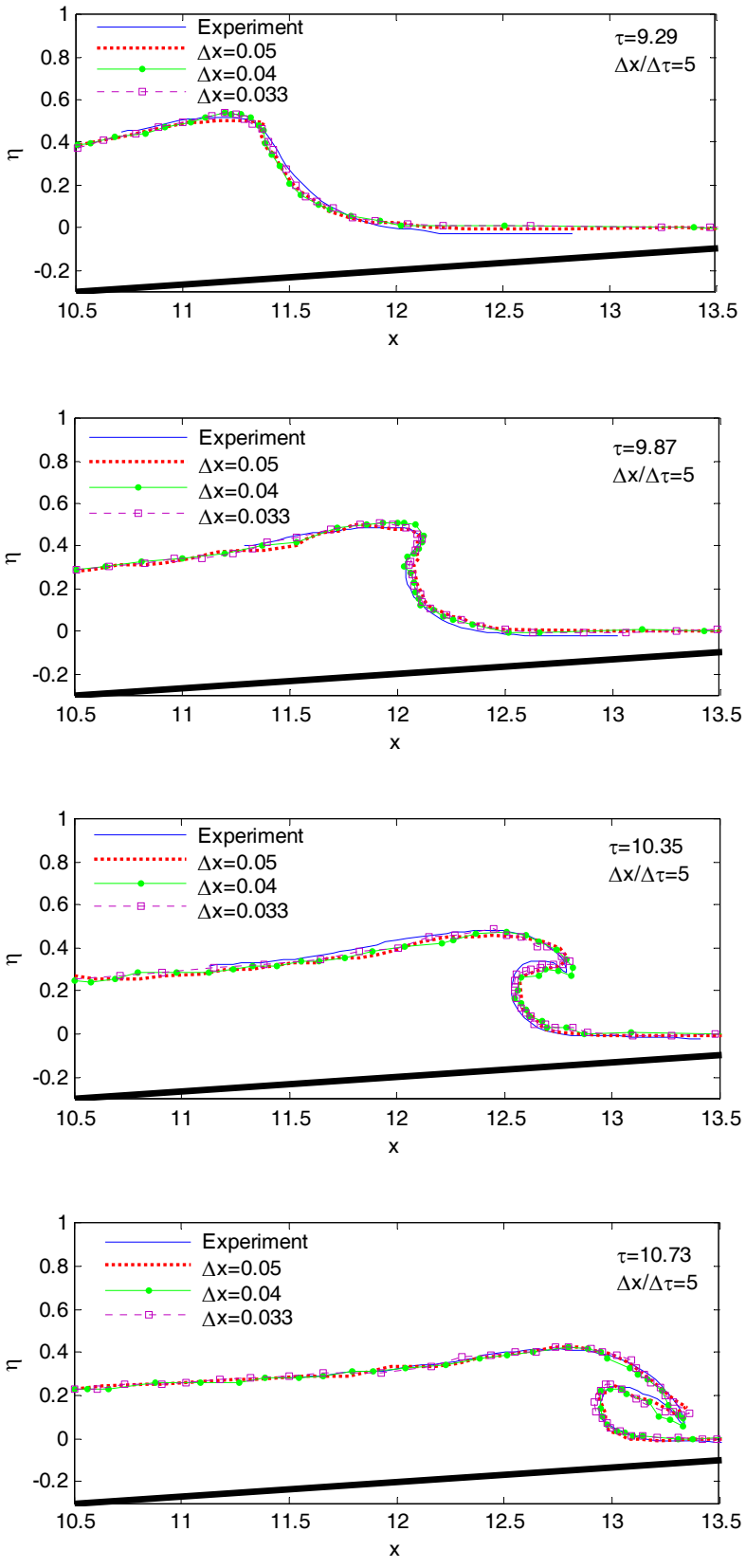


Fig. 6.1.2 Comparison between experimental wave profiles [Li and Raichlen (1998)] and numerical results obtained by using different Δx values when $\Delta x/\Delta\tau = 5$

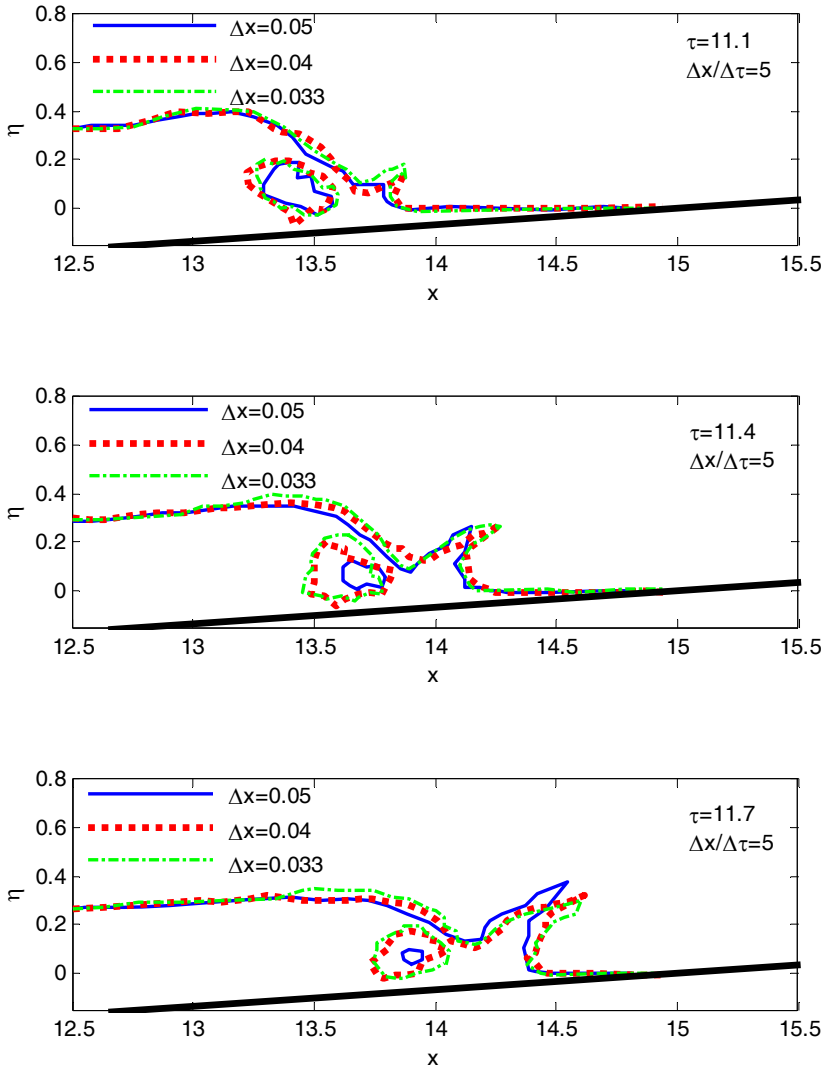


Fig. 6.1.3 Wave profiles in the post-breaking stage obtained by using different values of Δx at when $\Delta x/\Delta\tau = 5$

In order to further test the MLPG_R method for simulating the post-breaking wave stage, another case is considered, in which the incident solitary wave height is 0.4 and the slope is 1/15. In the case, the post-breaking stage will be concentrated to highlight for comparisons as indicated above, more particles should be needed to simulate the post-breaking stage. So *Nz=40* is chosen in this case, which yields the total particle number being 22,610. $\Delta x/\Delta\tau = 5$ is used to choose the time step. The comparisons of wave profiles with laboratory photographs (Li & Raichlen, 2003) are shown in Fig. 6.1.4, where laboratory photographs are shown on the left column and numerical simulation results are on the right column. From the figure, it can be seen that the numerical snapshots agree well with the laboratory photographs. Especially for the Fig. 6.1.4 (a) and (am), (b) and (bm) and (c) and (cm), the numerical snapshots are almost the same as the laboratory photographs.

After the plunging jet hits the free surface, a cavity appears where the air is enclosed by the water in the experiments. However, in the numerical modeling, the air is not considered. That may be the reason why there is some differences in the two snapshots Fig. 6.1.4 (d) and (dm), which may further affect the shapes of air cavity and jet after that. To improve the accuracy at this stage, the multi-phase flow model should be introduced in MLPG_R method, which will be done in the future work.

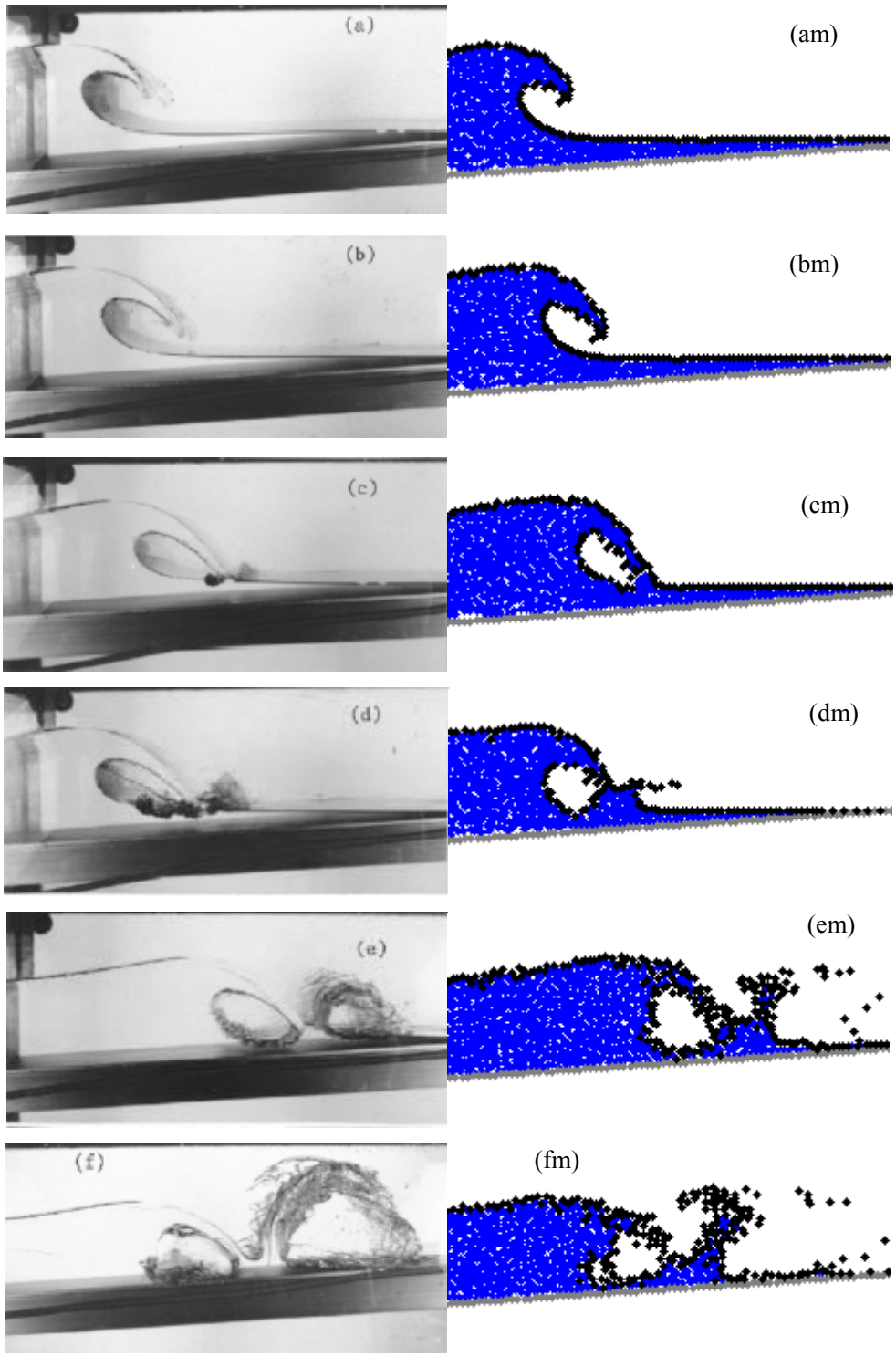


Fig. 6.1.4 Comparisons between laboratory photographs (Li and Raichlen, 2003) and numerical snapshots (right column: Blue particles: inner water particle; Black particles: free surface particle)

6.2 2D breaking waves over a submerged step

To further validate the method, another case is considered, which has been experimentally studied by Yasuda, Mutsuda and Mizutani (1997). The sketch and the coordinate system are illustrated in Fig. 6.2.1, in which there are three wave gauges P1, P2 and P3 located at *x*= 6.45, 8.11 and 9.74, respectively, to measure the wave elevations at the different positions. The solitary wave height is *h=0.423*. The particle number (*Nz*) along the z-axis is 33 in the part before the step and is 5 over the step, respectively. The total particle number used in this case is 14,806. The representative distance is chosen as $\Delta x = 0.03$. Taking $\Delta x/\Delta\tau = 5$, the time step is 0.006.

The solitary wave is generated by the piston-type wavemaker amounted at the left side of the tank in the same way as above. At earlier stage, the solitary wave propagates towards the step without deformation. When it is near the step, a part of the wave is reflected and other is transmitted to the area over the step. The front of the transmitted wave becomes steep and a jet is formed. Then the wave breaking occurs. To illustrate the process, four snapshots of solitary wave profiles over the step at different time steps are shown in Fig. 6.2.2. One can observe that the wave crest becomes steep at $\tau = 6.67$; a plunging jet is formed and almost hits the free surface at $\tau = 7.81$. After the jet impacts the free surface, an cavity appears and the second plunging jet is formed at $\tau = 8.43$. Another new air cavity appears after the second jet hits the free surface at $\tau = 9.27$. This process is largely similar to that for the solitary wave propagating over a sloped beach.

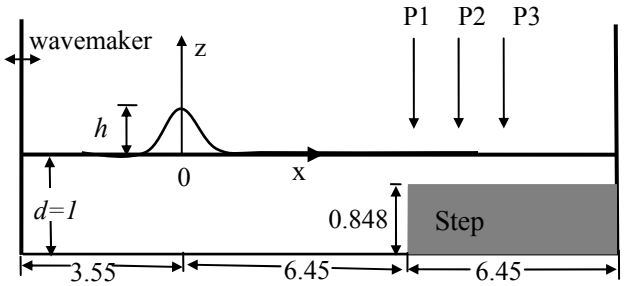


Fig. 6.2.1 Sketch of the problem about a solitary wave propagating to and over the step

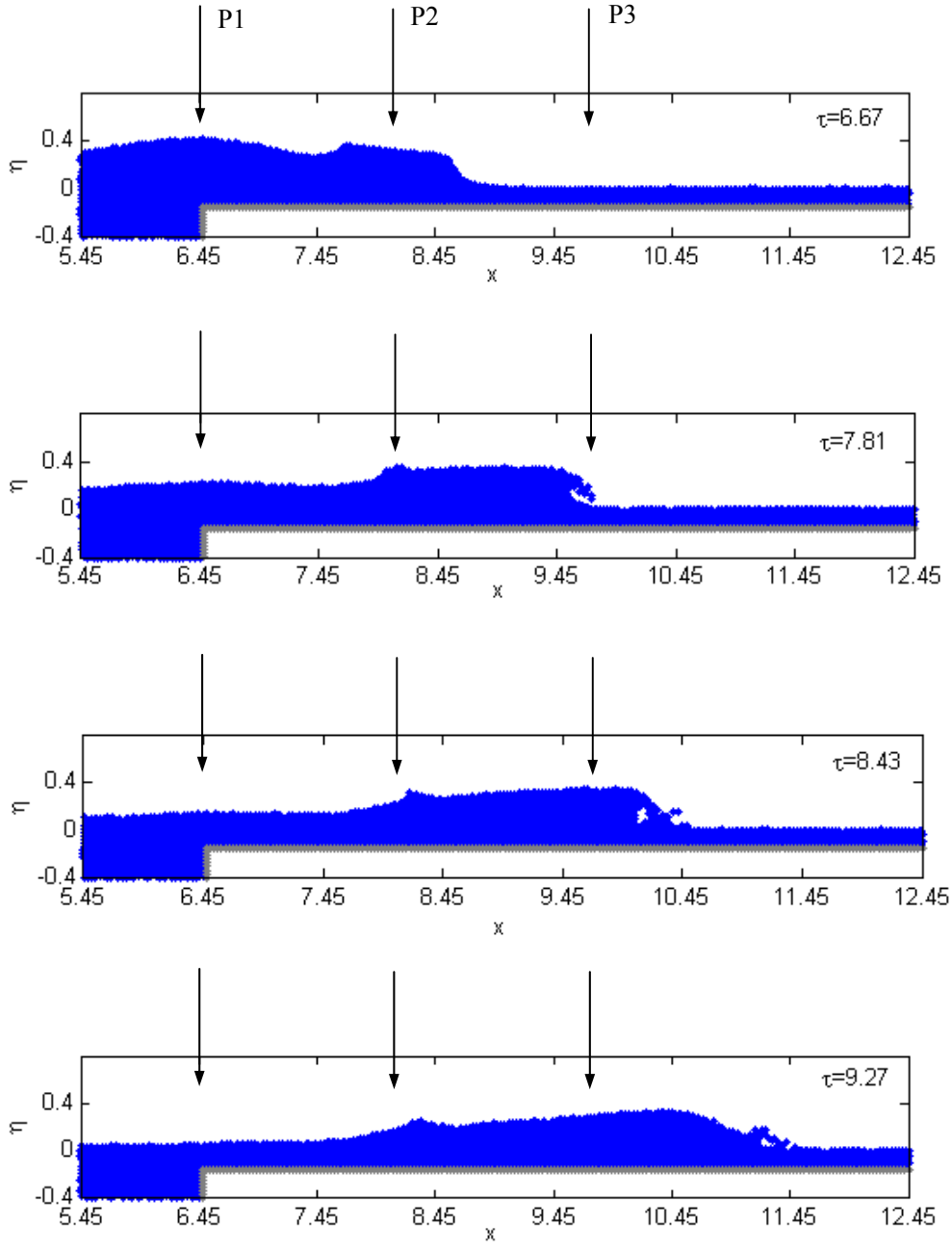


Fig. 6.2.2 Snapshots of solitary wave evolution over the step at different time steps

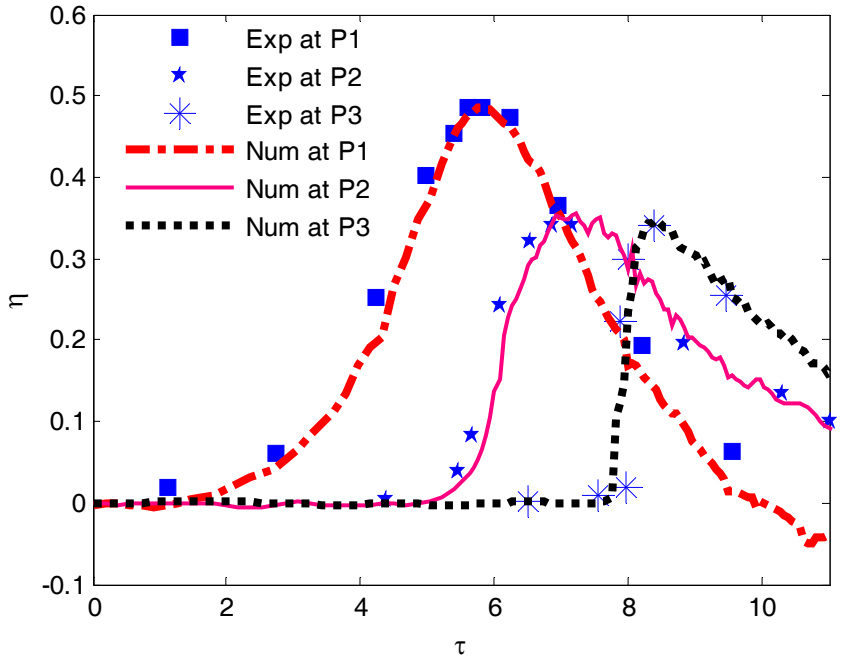


Fig. 6.2.3 Comparisons of wave elevations between the numerical results (line) and experimental data (mark) (Yasuda et al, 1997) at three different gauges (P1, P2 and P3)

The time histories of the wave elevations recorded at the three wave gauges are shown in Fig. 6.2.3, together with the corresponding data from Yasuda, Mutsuda and Mizutani (1997). As can be seen from the figure, the wave form recorded at Gauge P1 is still similar to a full solitary wave form, but the maximum wave height is higher than the height at x=0. At Gauges P2 and P3, the wave elevations grows quickly but the wave height is relatively smaller compared with the height at $x=0$. This indicates that the wave has become very steep or broken before the points. As can also be seen, the numerical results are in very good agreement with the experimental data (Yasuda, Mutsuda and Mizutani 1997) at all the gauges. That again shows that the MLPG_R method works well in the cases with violent breaking waves.

Except the wave elevation, the velocity field around the step tip is also shown in Fig. 6.2.4 when wave crest is passing by, from which it can be clearly seen that due to the existence of vertical step, water particles before the step have obvious upwards velocity and consequently increase the wave elevation. It also explains the phenomena in Fig 6.2.3 why the total wave elevation at wave gauge P1 is higher than the solitary wave height. A clockwise eddy can also

be observed above the step, the reason is that water particles are quickly passing by the step tip, which leads to relative less water particles and the relative lower pressure area above the step. Therefore, some water particles flow back from the right side to the step tip as a result of pressure difference.

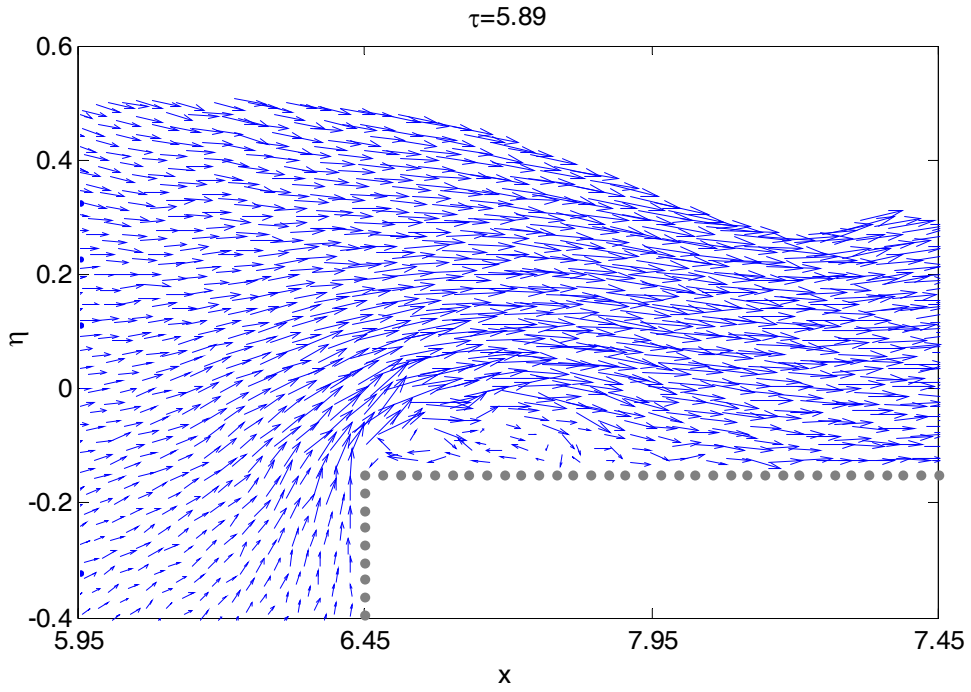


Fig. 6.2.4 Velocity field around the step when wave crest is passing by

6.3 3D breaking waves on a slope

The experimental results of 3D breaking waves have not been found in the public domain to the best of the knowledge. However, one may agree that all 2D experiments may also be considered as a y-independent 3D problem to validate 3D MLPG_R methods. For this purpose, the same 2D case in Chapter 5.1 will be simulated in a 3D numerical wave tank. The width of the numerical wave tank is 0.5, the others parameters are the same as those shown in Fig. 6.0.1. In order to save the CPU time for this 3D case, computational process is split into two stages. The first stage starts from wave generation and ends when the wave propagates just to the toe of the slope (x=0 in Fig. 6.0.1). The wave propagation in this stage is simulated by using the 2D model. The second stage continues from the end of the first stage,

and the 3D model is employed. It is noted that the 3D model can be used for the entire process, though it may need a longer CPU time.

Following the 2D convergence property investigations, similar investigations have also been made for the 3D case and corresponding results will be presented as following section.

6.3.1 Convergence investigation on different values of $\Delta x / \Delta \tau$

Similar to 2D investigations in Chapter 6.1, the numerical tests will be carried out, i.e. $\Delta x / \Delta \tau = 2.5, 5, 7$ and 9 and with fixed representative distance $\Delta x = 0.05$, which yields 89,492 particles totally. For the purpose of better comparisons between the numerical results and experimental data from Li and Raichlen (1998), which should be considered as average wave profiles with respect to y-coordinates because they are drawn from the photographs taken during laboratory experiments from a side, the resulting wave profiles near the section when overturning occurs at four instants and at three different transverse positions (y=0, y=0.25 and y=0.5) are shown in Fig. 6.3.1. One can observe that before just overturning (Fig. 6.3.1 a and b), all the numerical results are almost the same as the corresponding experimental data. At the third instant ($\tau = 10.35$), the numerical results for $\frac{\Delta x}{\Delta \tau} = 5$ and $\frac{\Delta x}{\Delta \tau} = 7$ are close to each other and both have good agreement with the experiment, but $\frac{\Delta x}{\Delta \tau} = 2.5$ and $\frac{\Delta x}{\Delta \tau} = 9$ seem have a bit quicker forward velocities compared to the experiment. This observation indicates that the convergence features of the 3D MLPG_R method are similar to those for the 2D case. For better simulation of breaking wave cases, especially for the post-breaking stages, the results seem to imply that for a certain value of representative distance, the time step should not be too large or too small.

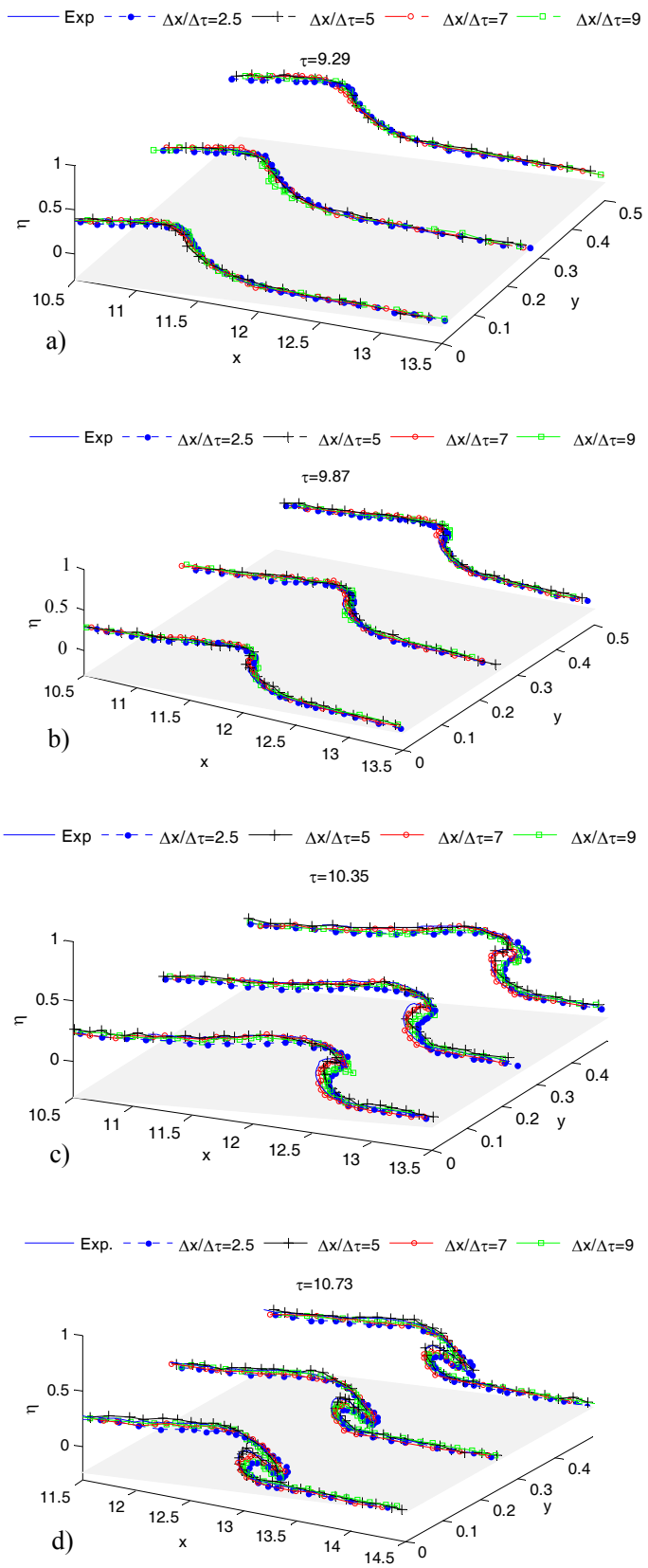


Fig. 6.3.1 Comparison between experimental wave profiles (Li and Raichlen, 1998) and numerical results for different values of $\Delta x/\Delta\tau$

6.3.2 Convergence investigation on different values of Δx

In this subsection the results obtained by using the three different values of Δx, i.e., 0.05, 0.04 and 0.033 (corresponding to *Nz*=20, 25 and 30, respectively), with $\Delta x / \Delta \tau = 5$ as discussed. The wave profiles near the section when overturning occurs at four instants and at three different transverse positions (y=0, y=0.25 and y=0.5) are shown in Fig. 6.3.2. The four instants are those as in Fig. 6.3.2, corresponding when the wave becomes very steep, just overturning, a jet appearing and a jet touching the water surface in its front. The experimental data from Li and Raichlen (1998) are also shown in the Fig. 6.3.2 for comparisons. The figure demonstrates that the results corresponding to different Δx values are very similar to each other and that all of them agree quite well with the experimental profiles. This also demonstrates that the value of $\Delta x = 0.05$ used in Fig. 6.3.1 is suitable.

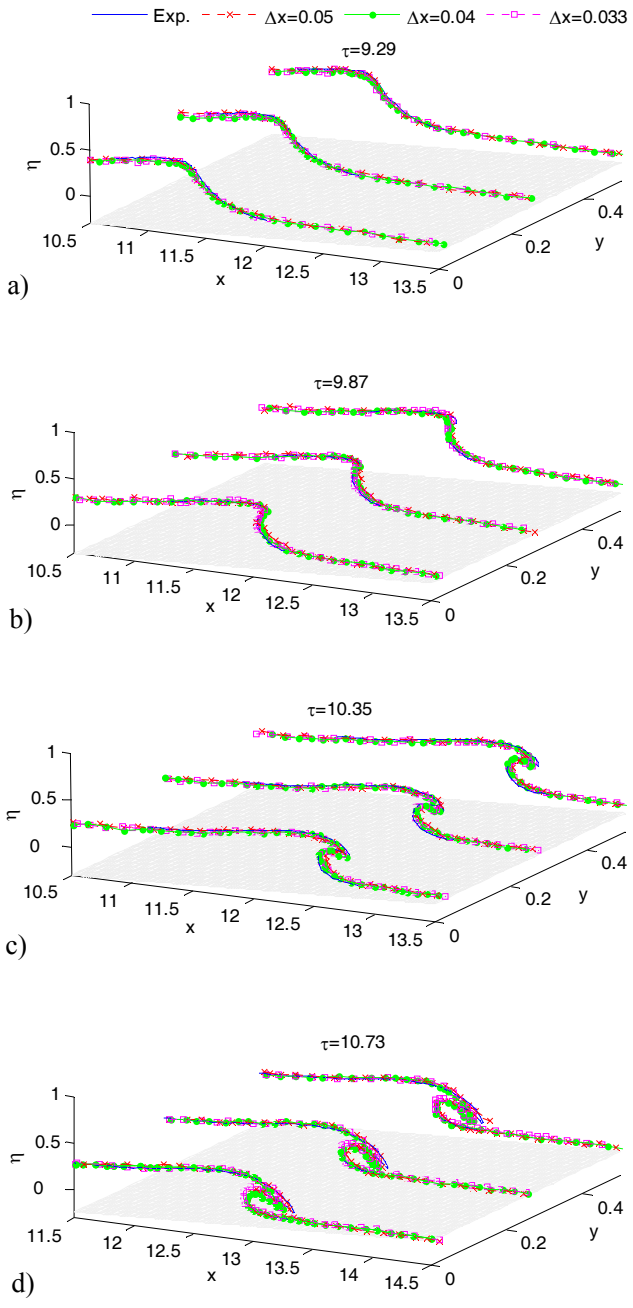


Fig. 6.3.2 Comparisons between experimental wave profiles [Li and Raichlen (1998)] and numerical results obtained by using different Δx when $\Delta x / \Delta \tau = 5$

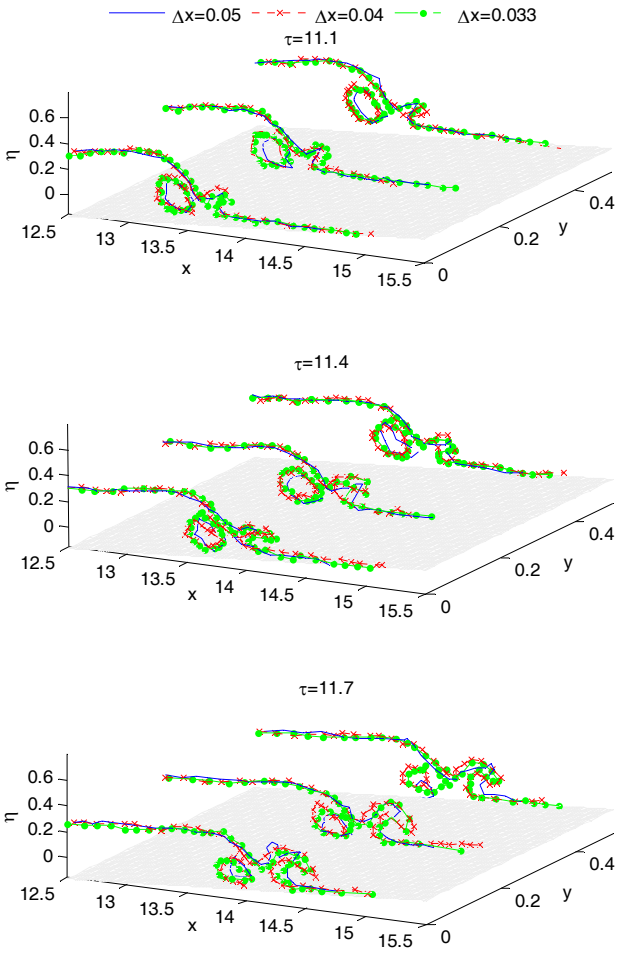


Fig. 6.3.3 wave profiles in the post-breaking stage obtained by using different values of Δx when $\Delta x / \Delta \tau = 5$

The wave profiles at the post-breaking stage are plotted in Fig. 6.3.3. At this stage, the plunging jets hit on the free surface in the front and cause splash, leading to a cavity formed behind the new jet. In addition, the profiles for Δx =0.04 (*Nz=25*) and Δx =0.033 (*Nz=30*) are very similar to each other, but have visible discrepancy with those for Δx =0.05, especially in the shape of the cavities. This indicates that one may need more particles to model the post-breaking waves. It is noted that the profiles at different y-coordinates are not the same, unlike what we have seen in Fig. 6.3.2. That is perhaps because some unknown random factors arise due to the turbulence at the post-breaking stage. The similar phenomenon can be observed in laboratory experiments. Further investigations are required to find the reasons in future.

7. SIMULATION OF VIOLENT SLOSHING WAVES

With the increasing demand for oil and natural gas, large FPSOs (Floating Production, Storage and Offloading System) have recently been developed quickly. Large FPSOs with liquid tanks of large volume may suffer from random waves frequently on the real seas. Due to the excitations from random waves, liquid sloshing in tanks has complicated behaviours. In some severe sea states, violent sloshing may yield very huge impact pressure to tank structure, even cause the damage of tank structures. Therefore the violent sloshing waves in tanks have been given lots of concerns in the past decades. Much effort has been devoted to numerically or experimentally study the violent wave motion in sloshing tanks and much useful knowledge has been accumulated. However, there are still lots of uncertainties associated with violent sloshing, for example, the pressure impact, the baffle effect on the sloshing wave and 3 directional effects. Because violent sloshing is a strong nonlinear problem, which may involve many complicated physical phenomena and procedure, such as breaking wave, splashing, formation of air pocket and air bubbles, behaviours of impact pressure and dynamic coupling with structural response. Due to these, it is difficult to exactly predict violent sloshing motions. Therefore, the laboratorial experiments were considered as main tools to study the practical information of sloshing problems. However, it is not always possible to obtain the required information, such as the continuous spatio-temporal details of dynamic distributions of water flow (velocity field) and pressure, which are essential for studying dynamic features of sloshing motion. Over the past decades, the numerical methods have been developed to address the issue. For instance, Kishev et al (2006) and Akyildiz & Unal (2006) adopted the finite difference method to investigate pressure impact problem due to the violent sloshing; Okamoto and Kawahara (1990) used the finite element method to analysis the 2D sloshing problems; Djavareshkian & Khalili (2006) made the comparison of finite volume and pendulum models for simulation of sloshing; Naito &

Sueyoshi (2002), Sueyoshi (2009) and Pan et al (2008) adopted MPS method to simulate different sloshing problems; SPH method was used by Delorme et al (2005) and Kim (2007) to investigate the problems.

This chapter will present our work on modelling sloshing waves by using the MLPG_R method. First the mathematical model and numerical technique will be presented in Chapter 7.1; two dimensional sloshing simulations, convergence investigation and numerical validation will be given in Chapter 7.2; the last section 7.3 focuses on three dimensional sloshing simulations and corresponding conclusions are given.

7.1 Mathematical model and numerical techniques

Similar to many numerical methods focusing on the sloshing problem, a moving coordinate system, following the motion of the tank, is employed. In other words, the tank is fixed with respective to the coordinate system. Thus the fluid flow relative to the tank will be considered. The governing equations in the relative system are given in the following Lagrangian form:

$$\nabla \cdot \vec{u} = 0 \tag{7.1.1}$$

$$\frac{D\vec{u}}{Dt} = -\frac{1}{\rho}\nabla p + \vec{g} + \nu\nabla^2\vec{u} - \vec{A}_s \tag{7.1.2}$$

The last term in Eq. (7.1.2), $\vec{A}_s$, represents the acceleration induced by the motion of the tank, which may include the translational and angular motions and expressed by

$$\vec{A}_s = \frac{d\vec{U}}{dt} + 2\vec{\Omega}\times\vec{u} + \frac{d\vec{\Omega}}{dt}\times\vec{r} + \vec{\Omega}\times\left[\vec{\Omega}\times\vec{r}\right] \tag{7.1.3}$$

where $\vec{\Omega}$, $\vec{U}$ are the angular velocity and translational velocity vector, respectively, $\vec{r}$ is the position vector of the point concerned relative to the centre of the rotation (e.g. centre of gravity of a ship); The compressibility is ignored in the above equations. The position of the fluid particle relative to the system is determined by

$$\frac{D\vec{r}}{Dt} = \vec{u}\ . \tag{7.1.4}$$

At all the free surface particles, pressure is considered as equal to relative atmospheric pressure, which may be taken as zero, i.e.,

$$p = 0 \tag{7.1.5}$$

On the rigid boundary, the following kinematic and dynamic boundary conditions are

satisfied:

$$\vec{n}\cdot\vec{u}=\vec{n}\cdot\vec{U} \tag{7.1.6}$$

and

$$\vec{n}.\nabla p=\rho\left(\vec{n}\cdot\vec{g}-\vec{n}\cdot\dot{\vec{U}}+\nu\,\vec{n}\cdot\nabla^2\vec{u}\right) \tag{7.1.7}$$

where $\vec{n}$ is the normal direction of the boundary surface.

The mathematical model formed by Eqs. (7.1.1) to (7.1.6) is solved with a time marching procedure, which is similar to the numerical procedure for the fixed coordinate system detailed in Chapter 3. Only a summary is given below. Suppose the velocity, pressure and the position of a particle at *n–th* time step ($t=t_n$) are known, they are found at *(n+1)-th* time step by the following step:

(1) Calculate the intermediate velocity ($\vec{u}^*$) and position ($\vec{r}^*$) of particles using

$$\vec{u}^{(*)}=\vec{u}^{(n)}+\Delta t\left[\vec{g}+\nu\nabla^2\vec{u}-\vec{A}_s\right]^{(n)} \tag{7.1.8}$$

$$\vec{r}^*=\vec{r}^*+\vec{u}^*\Delta t \tag{7.1.9}$$

where the superscript n represents *n-th* time step; Δt is the length of time step.

(2) Find the pressure p^{n+1} by solving

$$\nabla^2 p^{n+1}=\alpha\frac{\rho}{\Delta t^2}\frac{n^0-n^*}{n^0}+(1-\alpha)\frac{\rho}{\Delta t}\nabla\cdot\vec{u}^* \quad \text{for inner particles} \tag{7.1.10}$$

and

$$\vec{n}\cdot\nabla p^{(n+1)}=\frac{\rho}{\Delta t}\vec{n}\cdot\vec{u}^{(*)} \quad \text{on a rigid boundary} \tag{7.1.11}$$

(3) Calculate the fluid velocity and therefore update the position of the particles using

$$\vec{u}^{(**)}=-\frac{\Delta t}{\rho}\nabla p^{(n+1)} \tag{7.1.12}$$

$$\vec{u}^{(n+1)}=\vec{u}^{(*)}+\vec{u}^{(**)} \tag{7.1.13}$$

$$\vec{r}^{\,n+1}=\vec{r}^{\,n}+\vec{u}^{\,n+1}\Delta t \tag{7.1.14}$$

(4) Go to the next time step

There is only one difference between the procedure above and the procedure in Chapter 3.2. The difference is the last term in Eq. (7.1.8), which is the acceleration of sloshing waves;

while no this term in Eq. (3.2.1).

In the following cases, the fluid in the sloshing tank is chosen as water. Similar to what have done before, the parameters with a length scale are nondimensionalised by the water depth d, the pressure by ρgd and the time by $\sqrt{d/g}$ where g is gravity acceleration. In all the cases, the particles are uniformly distributed at the beginning with the same distance between the particles in all directions.

7.2 Numerical investigations on 2D sloshing waves

The accuracy of the MLPG_R method in modelling 2D sloshing problems is first investigated by comparing with experimental data. Corresponding studies on the sloshing problem are first presented to reveal its convergence feature in such problems, though the convergence property of the MLPG_R methods have been studied by using many cases presented in Chapter 5 and Chapter 6. Apart from this, the effects of the baffle in a tank on the sloshing problems, which have not been systematically studied using NS-based numerical model, are mainly focused. Some interesting finding will be revealed.

7.2.1 Convergence investigation and numerical validation

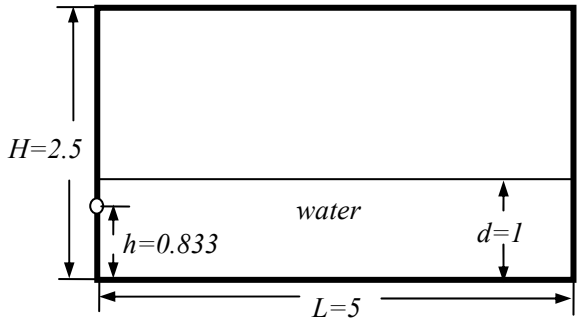


Fig. 7.2.1 Schematic view of experimental set-up and corresponding tank sizes

To validate the MLPG_R method in sloshing problems, some experimental results in public domain, i.e. Kishev et al (2006), is used. The case considered here is a rectangle tank in translational motion. The filling rate is 40%. The corresponding experimental set-up and some experimental parameters are shown in Fig. 7.2.1. A pressure recorded point is located on the

left side of the tank with 0.833 above the bottom. The tank's motion is specified by:

$$X(\tau) = a\sin(\omega\tau) \tag{7.2.1}$$

$$U(\tau) = \dot{X}(\tau) = a\omega\cos(\omega\tau) \tag{7.2.2}$$

where a and ω are the amplitude and frequency of the tank motion, respectively, which are chosen as 0.41667 and 0.46328 rad, respectively. The parameters are the same as those for one of the cases in Kishev et al (2006), where they presented some experimental results.

Table 7.2.1 Calculation conditions of convergent investigation

Nz	Δx	$\Delta x/\Delta\tau$	$T/\Delta\tau$	Nz	Δx	$\Delta x/\Delta\tau$	$T/\Delta\tau$
25	0.04	10	3390	35	0.02857	10	4746
25	0.04	8	2712	35	0.02857	8	3797
25	0.04	5	1695	35	0.02857	5	2373
25	0.04	2.5	848	35	0.02857	2.5	1187
25	0.04	2	678	35	0.02857	2	949
25	0.04	1.67	566	35	0.02857	1.67	793
30	0.033	10	4068	40	0.025	10	5424
30	0.033	8	3254	40	0.025	8	4339
30	0.033	5	2034	40	0.025	5	2712
30	0.033	2.5	1017	40	0.025	2.5	1356
30	0.033	2	813	40	0.025	2	1085
30	0.033	1.67	680	40	0.025	1.67	904

It has been pointed out that the pressure histories have a random feature in violent sloshing waves. This has been demonstrated by many experimental works in literature (Kim, 2001). In numerical analysis, this is reflected by the high frequency fluctuation. It is difficult to precisely catch these fluctuations. In order to get more realistic results, it has been suggested and employed in the literature that the time histories are smoothed in the post-processing. One

way to do so is to average the pressure in the neighbourhood, as proposed by Kim (2001). This method will be used in the study. The formula for this purpose is given by

$$p(t_l) = \sum_{j=l-2}^{l+2} p(t_j)/5 \tag{7.2.3}$$

where $p(t_l)$ is the pressure at time t_l.

The accuracy of the MLPG_R is demonstrated in Fig. 7.2.2, which compares the pressure time history between the numerical results and the experimental data by Kishev et al (2006). In the numerical simulation, Nz=30, $\Delta x/\Delta\tau$ =2 are used. Since the time origin in the experiment is not given in the reference, the time axis for numerical results has been slightly adjusted to match the pressure peak value. From this figure, it is found that two results are largely close to each other, which indicates that the MLPG_R method can yield results with the main feature.

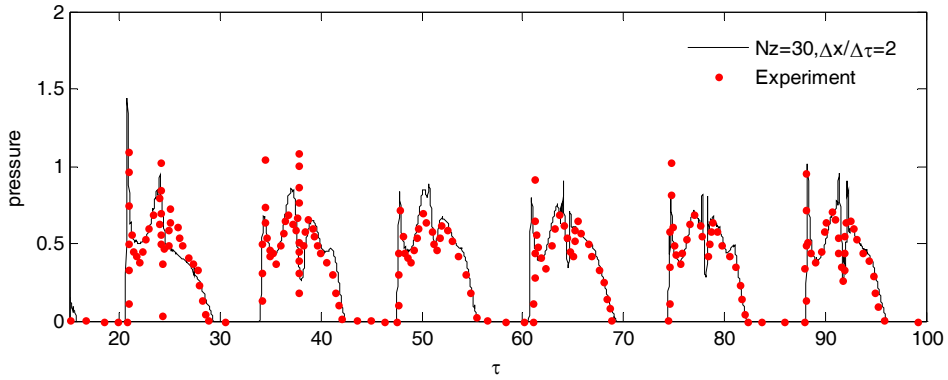


Fig. 7.2.2 Comparison between the experiment and numerical results

In order to investigate the convergence property of the MLPG_R method, two important parameters ($\Delta x/\Delta\tau$ and Δx) are mainly considered. As indicated before, Δx is related to Nz by Δx =1/Nz, thus, the effect of Δx can be converted to that of Nz. According to the previous numerical investigation experience, their ranges in this investigation are, respectively, chosen as [1.67, 10] and [0.025, 0.04]. The detailed configurations are given in Table 7.2.1. Ten periods were computed for each case. In order to clearly show the comparisons of different pressure curves, the pressure time histories only in the last four periods are shown in the figures displayed below.

The convergence property against $\Delta x/\Delta\tau$ is first considered. Some results are shown in Fig. 7.2.3 with different colours representing different values of $\Delta x/\Delta\tau$, in which Δx=0.033 (*Nz=30)*. One can see that six pressure curves are very similar to each other except the regions near some pressure peaks. It is also found that bigger the value of $\Delta x/\Delta\tau$ is, more significant the pressure oscillation is. Further investigations are needed to suppress the fluctuations. Furthermore, the pressure changes very quickly within several time steps when the impact occurs. So in order to simulate the phenomenon, the time step selected should not be too big. Based on the above reasons, the $\Delta x/\Delta\tau$ may be chosen between 2 and 5.

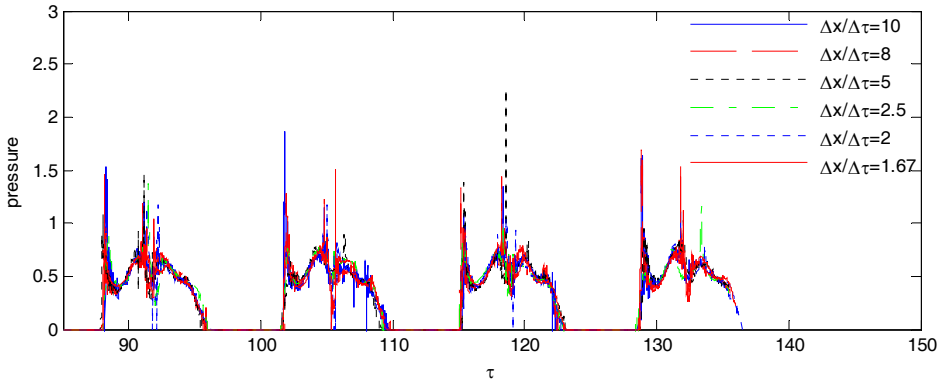


Fig. 7.2.3 Time histories of pressure with different values of $\Delta x/\Delta\tau$ for *Nz=30*

Numerical investigation is also carried out to see how the different values of *Nz* affect the numerical results. Based on the investigations shown in Fig.7.2.3, $\Delta x/\Delta\tau$=2 is used here. The pressure time histories in the last four periods in the cases with different *Nz* (i.e. *Nz* is equal to 25, 30, 35 and 40, respectively) are plotted in Fig. 7.2.4 Similar to Fig. 7.2.3, numerical results are in good agreement with each other except in the region around the pressure peaks. The pressure histories are also compared with the corresponding experimental data (Kishev et al, 2006). Some results are shown in Fig. 7.2.5. For clarity, only results in one period are shown. From the Fig. 7.2.5, it can be seen that the numerical results are very similar to the experimental data except for *Nz*=25. Based on the tests, the results for *Nz*=30, $\Delta x/\Delta\tau$=2 can be accepted as convergent results for the violent sloshing case.

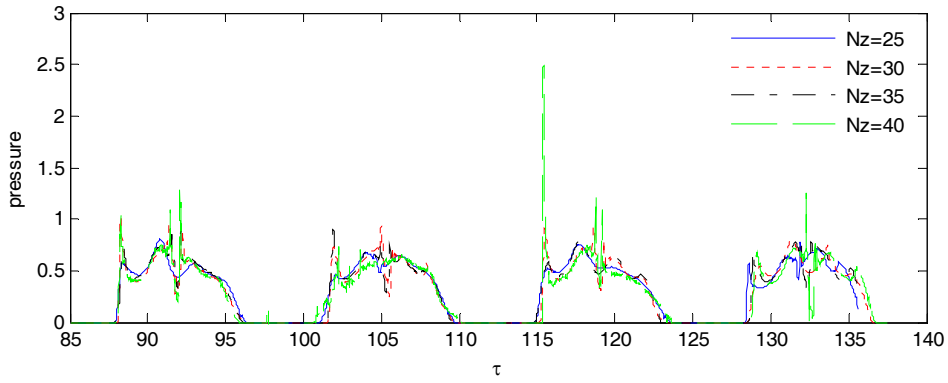


Fig. 7.2.4 Time histories of pressure with different *Nz* for $\Delta x / \Delta \tau = 2$

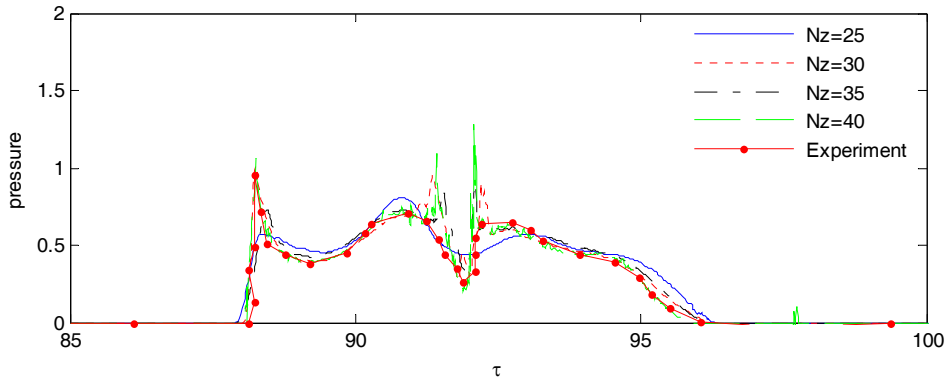


Fig. 7.2.5 Pressure in one period with different *Nz* for $\Delta x / \Delta \tau = 2$

In order to show the correlation between the peak pressure and the motion of sloshing waves, the snapshots obtained by numerical results using *Nz*=30, $\Delta x / \Delta \tau = 2$ corresponding to Fig. 7.2.2 are given in Figs. 7.2.6 ~ 7.2.10, where the black particles are the free surface particle. The first pressure peak appears when time is close to 88.2684. The wave profile shown in Fig 7.2.6 demonstrates that the major peak of the pressure takes place before the highest run-up is reached. The reason may be that the major peak is created by the rapidest change in the moving direction of water from horizontal to vertical during the short period after the fast moving water strikes the wall. The rapid change of the moving direction causes the rapid change of momentum and therefore the large increase of pressure near the wall.

With the tank further moves to the left, the run-up along the left tank wall continues increasing. During this period, the water kinetic energy is gradually converted into potential energy, which reduces the corresponding dynamic pressure on the tank wall. Fig. 7.2.7 exhibits the highest run-up at $\tau = 89.2685$, which is the time when the first trough value of pressure in Fig. 7.2.5 occurs. After this instant, the run-up is falling down while the pressure

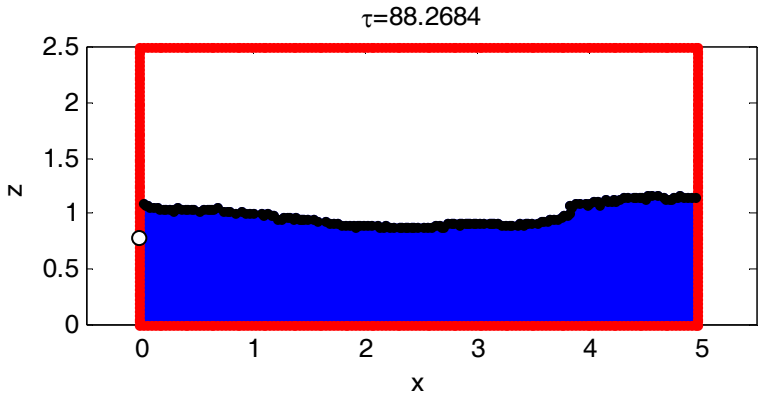


Fig. 7.2.6 Wave profile corresponding to the first peak shown in Fig. 7.2.5

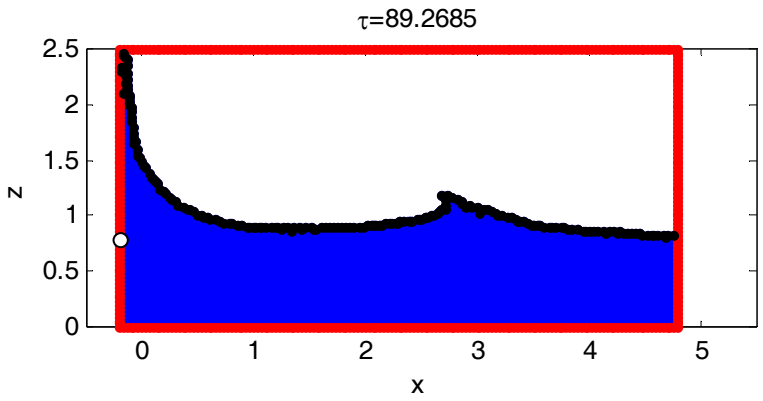


Fig. 7.2.7 Wave profile corresponding to the first trough value shown in Fig. 7.2.5

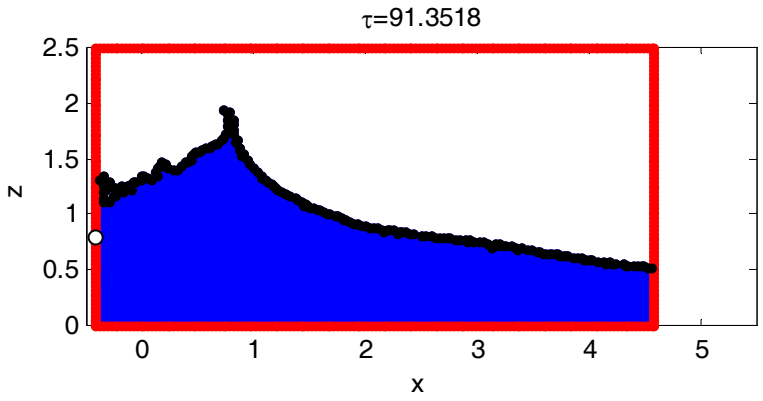


Fig. 7.2.8 Wave profile corresponding to the second peak shown in Fig. 7.2.5

is rising until reaching the second peak in the period. The reason for the adverse movements

of water and pressure may be explained as follows. While the run-up goes down, the falling-down water is constrained by both the wall and the water in the lower region. The result is that the falling-down water is forced to change its moving direction from vertical to horizontal and therefore the pressure near the wall is increased due to the variation of momentum caused by the direction change. Fig. 7.2.8 shows the water profile when the pressure reaches the second peak in Fig. 7.2.5. At this time, the tank moves to the maximum

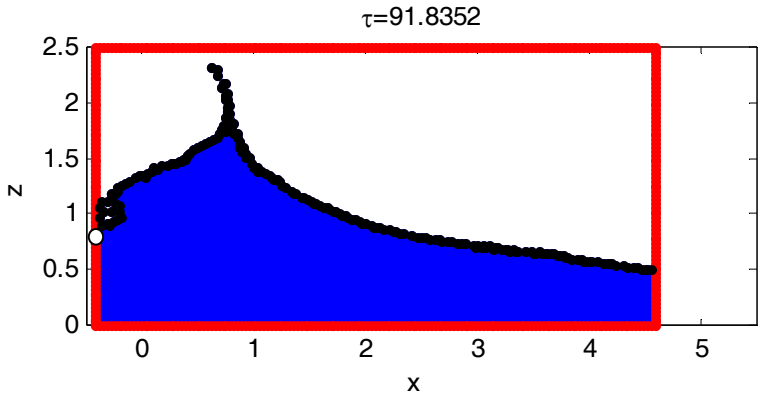


Fig. 7.2.9 Wave profile corresponding to the second trough shown in Fig. 7.2.5

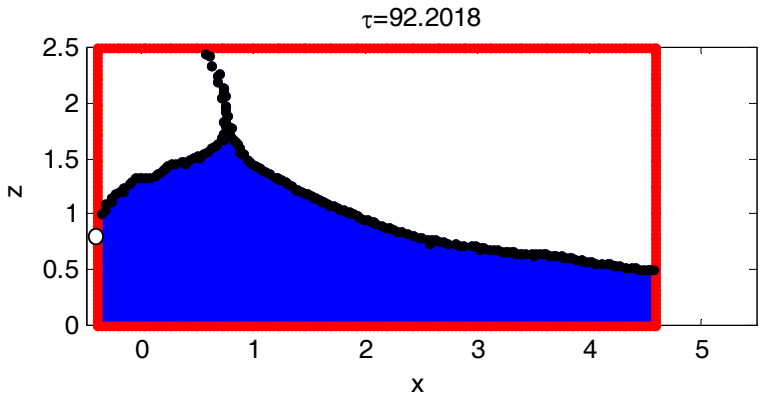


Fig. 7.2.10 Wave profile corresponding to the third peak shown in Fig. 7.2.5

displacement to left side. The moving acceleration and the force reach the maximum value, which causes the rapid change in the moving direction of water from leftward to rightward and yields the second pressure peak. With the continuous falling-down of the water level, the black particles (the free surface face where pressure is zero) quickly approaches the pressure recorded point's level (*h=0.8333*), Fig. 7.2.9 exhibits the wave profile corresponding to the second trough in the pressure history. On the left side of Fig. 7.2.9, one can clearly notice

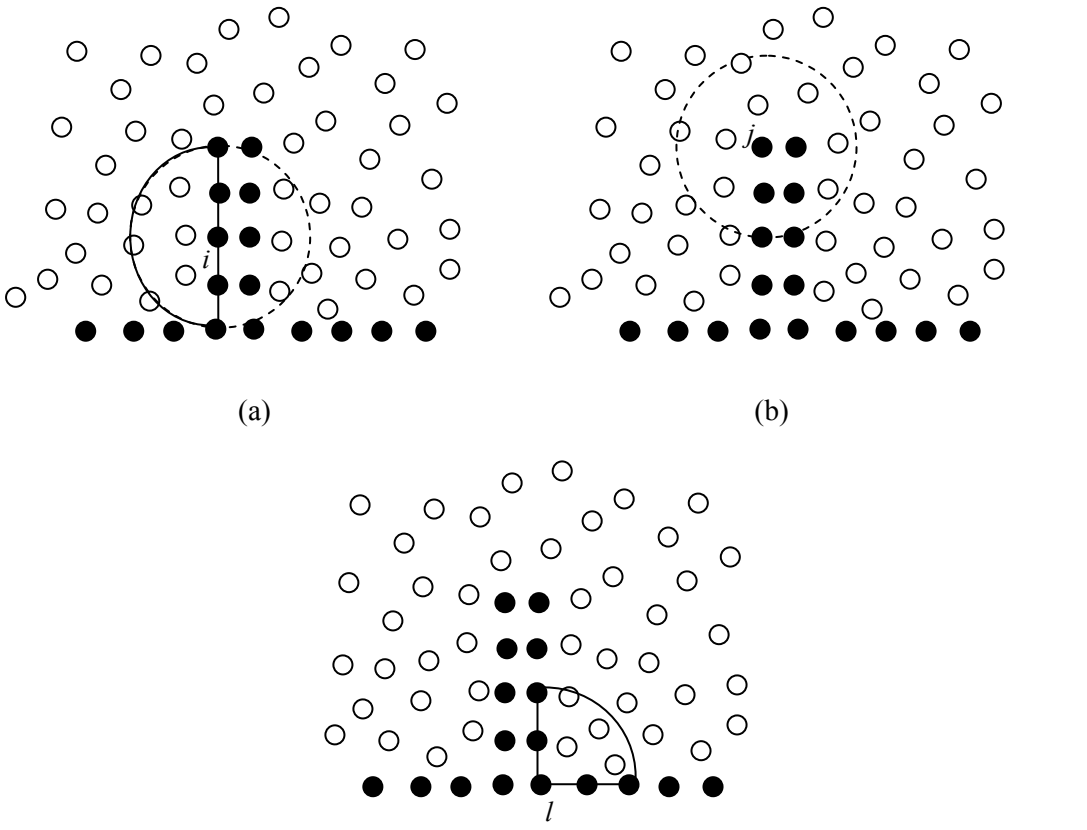
that the free surface is below $z<1$. With the tank movement changing from left to right, the wave crest reaches its maximum height (shown in Fig. 7.2.10) and then falls down; this situation is exactly similar to the one where the second pressure peak occurs. The change of momentum from vertical to horizontal yields the third pressure peak.

7.2.2 Behaviors of impact pressure in a baffled tank

In many applications, there are baffles in liquid tanks to support tank walls or suppress the liquid motion, for example, large girders in FPSOs. The baffles on the walls affect the motion of liquid in tanks and so the pressure distribution. Cases for tanks with the same size but different installations of a baffle will be presented here to illustrate how the pressure would be affected by a baffle when sloshing is violent. It is noted that the investigation is preliminary. A detailed investigation will be carried out in the near future.

In the cases with a baffled tank, special numerical algorithms to treat the baffle structures are required in the MLPG_R calculation. The basic ideas for numerical implementation of the boundary conditions on baffle walls are similar to those used for the rigid wall. The baffles are discretized into two columns of particles. The sketch of baffle, vertically installed on the lower horizontal tank wall, and the water particles around it is shown in Fig. 7.2.11, in which the vertical black wall particles represent the baffle and the horizontal row of black wall particles are those on horizontal tank wall. Due to the existing of the baffle, the particles on the baffle surface or those surrounding it may be affected. In the MLPG_R method, the influence domain for any particle *i* is generally defined by using a cut-off radius. All particles falling in the region with a distance not large than the cut-off radius are considered to directly influence the particle *i*. By using this way, the influence domain for a particle on the baffle surface may include particles on the other side of the baffle, which, physically, do not directly affect particle *i* due to the obstruct of the baffle. Some typical examples are shown in Fig. 7.2.11. For clarity, the discussion on the particles on the baffle surface and the fluid particles surrounding the baffle is separated.

For a particle on the baffle but not laying on the ends of the baffle, e.g. Particle *i* in Fig. 7.2.11(a), its initial influence domain is a full circle with a constant cut-off radius (dashed circle in the figure). However, due to the obstruct of the baffle, the particles falling in the right



(a) (b)

(c)

(d) (e)

Fig. 7.2.11 sketch of baffle and the neighbouring water particles
(black solid circles: baffle particles and wall particles; hollow circles: water particles; dash circles represent the initial influence domain; the solid curve represent effective influence domain)

semi-circle of the influence domain do not directly affect Particle *i*, only the fluid particle at the left semi-circle can interact with it. Therefore, the effective influence domain should be

the left semi-circled area (solid semi-circle as shown in Fig. 7.2.11(a)). For the particles located on the top of the baffle e.g. particle *j* shown in Fig. 7.2.11 (b), the effective influence domain is equal to the initial influence domain. For the particles at the bottom of baffle e.g. particle *l* shown in Fig. 7.2.11 (c), the effective influence domain is just a quarter circle because of the baffle existence.

The special treatment for the fluid particles surrounding the baffle is discussed here. For a fluid particle close to the baffle surface but away from the ends of the baffle, e.g. Particle *k* shown in Fig. 7.2.11 (d), the initial influence domain is a full circle (represented by dashed circle in the figure). Due to the baffle effect, the particles located on the other side (left side for particle *k*) of the baffle should be excluded from the initial influence domain. Hence only part of the initial influence domain may be considered as the effective influence domain, i.e. the plane surrounded by the solid curve and corresponding right side of for the particle *k*. For the fluid particle near the top of the baffle (shown in Fig. 7.2.11 (e)), the effective influence domain is equal to the initial influence domain. It should be noted that the technique described above can also be used to other problems with very thin structures, either submerged or surface-piercing.

By using the special treatment indicated above, the baffle effect on the sloshing problems is investigated. The length and height of the tank considered for this purpose are 3 and 2, respectively. The water height is 1, corresponding filling ratio is 50%. Three different conditions are used in the investigation, i.e. without baffle (Fig. 7.2.12A), with a vertical baffle located in the middle (Fig. 7.2.12B) and with a vertical baffle located at a position other than the middle of the tank (Fig. 7.2.12C). The height of the baffle in both the baffled tanks is half of the water depth. The tanks are subjected to the rotational motion defined by:

$$\theta(\tau) = \theta_{\max} \sin(\varpi\tau) \tag{7.2.4}$$

where the rotational amplitude being $\theta_{\max} = 8$ degree and frequency is ϖ =0.9115 rad close to the natural frequency of tank without baffle. Two points on the left wall of the tank, i.e. *a* and *b* are selected as the pressure recorded points. The corresponding vertical coordinate of *a* and *b* is *z=1* and *z=0.5*, respectively. The vertical particle number along *z*-direction (*Nz*) is chosen as 30 when distributing the initial particles, the corresponding Δx is 0.033333. The total particle number is

2,970; the time step is chosen as 0.016667.

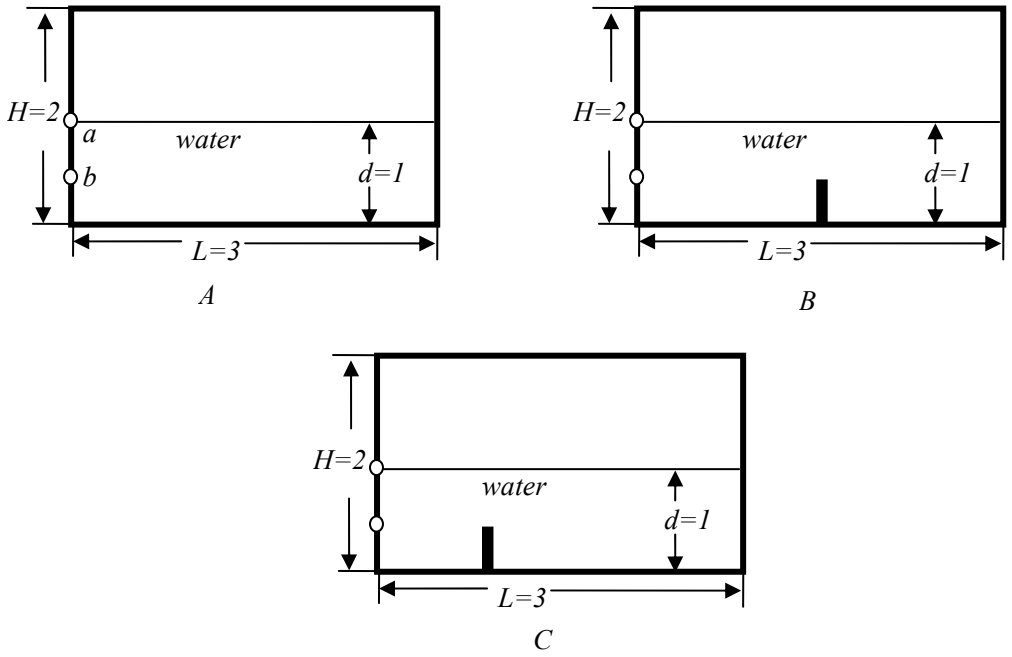


Fig. 7.2.12 Schematic view of tank and corresponding tank sizes
(*A*: No baffle; *B*: Middle baffle and *C*: Left baffle)

The total simulating time is 19 periods; it takes about 0.7 CPU hours to complete each case in normal a PC with Inter (R) Core(TM) 2 Duo CPU E7500 @ 2.93GHz (one CPU is used) and 2.93G RAM in Microsoft system. The pressure time histories recorded at point *a* in the last 7 periods are shown in Fig. 7.2.13. From the figure, the primary peak value from the case *A* and *C* are very close, while that from the case *B* is the minimum among the three cases. Nevertheless, there are significant differences in the profiles of the curves in this figure. For the case *A,* the pressure time history recorded at point *a* has one primary pressure peak and an obvious minor pressure peak in every period. However, for case *B* and case *C*, the minor pressure peaks are not very obvious. The wave profiles when the maximum pressure value in the last period occurs are plotted in Fig. 7.2.14. Due to the baffle effect, the wave profiles are quite different from each other. It can be observed that the run-up in the tank without baffles (case *A*) is the highest. The run-up for case *B* is the lowest, the run-up for case *C* is the between the other two.

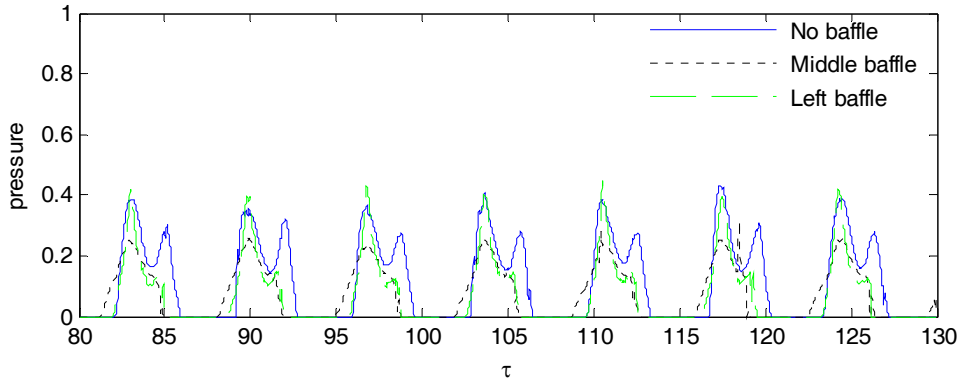


Fig. 7.2.13 Comparison of pressure time histories at the point *a*

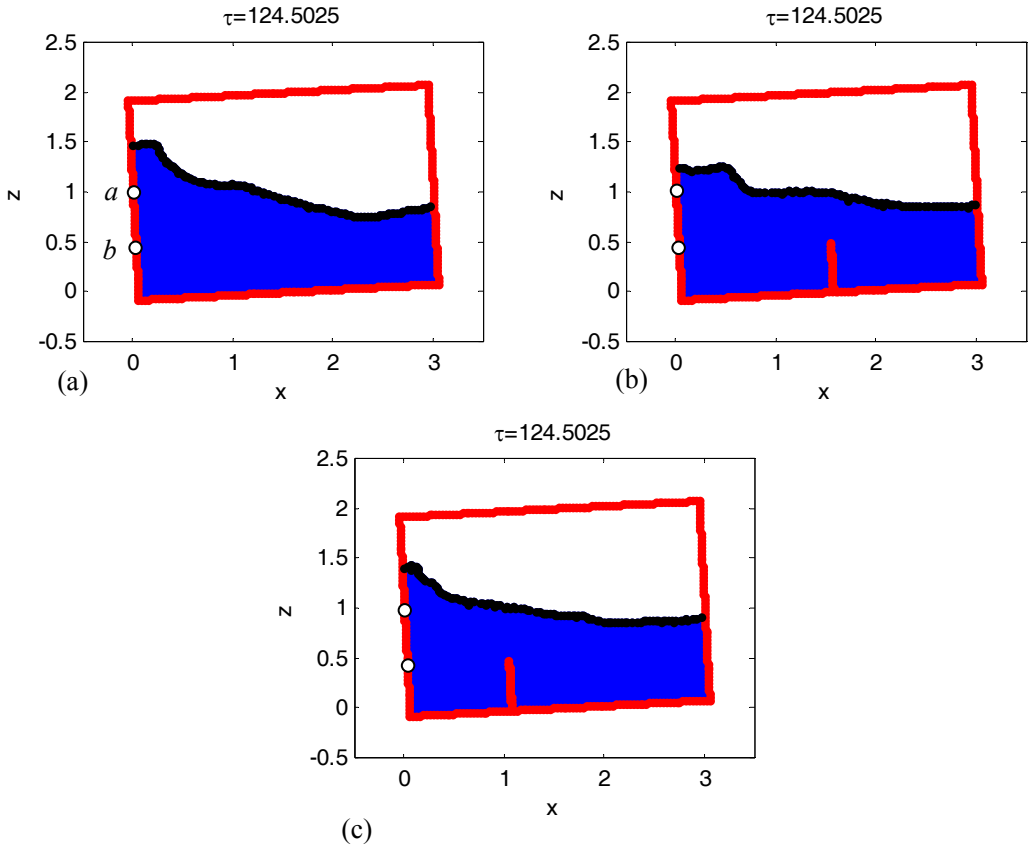


Fig. 7.2.14 Wave profiles corresponding to a primary pressure peak
in the last period shown in Fig. 7.2.12
(a) for Case A, (b) for Case B and (c) for Case C

Fig. 7.2.15 exhibits the pressure time histories at point *b* in the three cases. The primary peak in case *A* has the maximum value compared to the others. For the case C, the primary peak is close to that in the Case *B*. That means the pressure peak at Point *b* is reduced due to the existing of baffle.

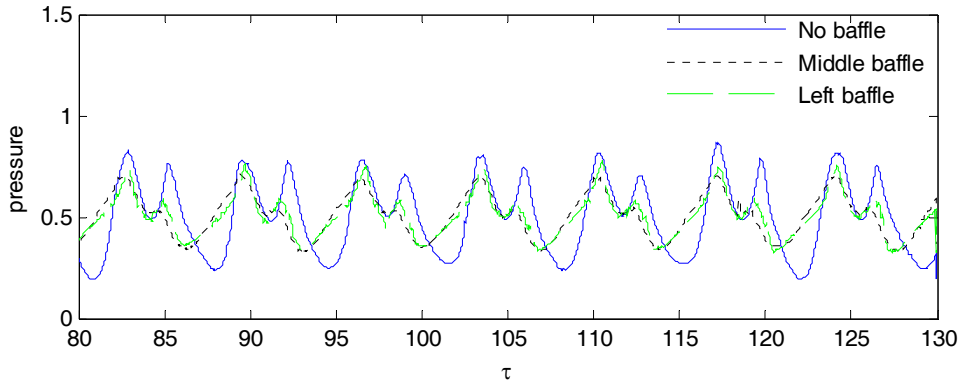


Fig. 7.2.15 Comparison of pressure time histories at the point *b*

These cases appear to demonstrate that the baffles can affect the motion of sloshing waves and therefore the largest pressure. Nevertheless, just like the discussions in Ma et al (2009) that one should bear in mind that there are many factors leading to these results. At least, two factors would play a role: the resistance of the baffle to the movement of the water in the tank and the change of the natural frequency caused by it. The first factor can always dissipate the wave motion and so reduce the pressure magnitude. The role of the second factor depends on the size and position of the baffle and also on the frequency of the excitation. If the baffle is high enough, the natural frequency of the baffled tank is significantly different from that of the smooth tank without baffles. Under such a condition, when the exciting frequency is near the natural frequency of the smooth tank and sufficiently away from that of the baffled tank, the motion of the wave and so the pressure may grow in a long time interval (Wu et al, 1998) and so is larger in the smooth tank. On the other hand, when the exciting frequency is close to the natural frequency of the baffled tank and sufficiently away from that of the smooth tank, the motion of the wave and so the pressure in the baffled tank are not necessarily smaller. More work is apparently required to further understand the effect of a baffle on sloshing waves.

7.3 Numerical investigations for 3D sloshing waves

In this section, the MLPG_R method is adopted to simulate the violent sloshing wave in a three dimensional container with 6 degrees of freedom of motion. Since the main objectives of the thesis is to develop the method, rather than investigating 3D sloshing problem, only one

case is presented here for demonstration. The schematic view and dimensions of tank are shown in Fig. 7.3.1. The filling rate is 50%. There are two columns of particles to record the pressure. One column is on the left front corner of the tank; the other one is at the left middle of the tank wall. The heights of P1 and P4 are 0; the heights of P2 and P5 are 0.95 and the heights of P3 and P6 are 1.7. The tank are excited by both translational and rotational motions, which are decomposed to,

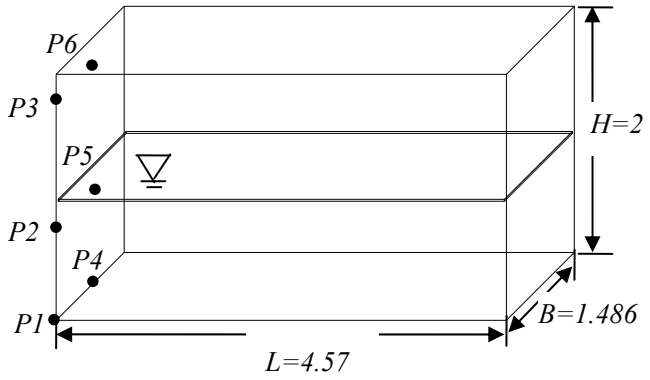


Fig. 7.3.1 Schematic view and sizes of the tank

The periodic translational excitations are as follows:

Surge motion: $x(\tau) = 0.044\sin(0.655\tau - 1.66)$ (7.3.1)

Sway motion: $y(\tau) = 0.03\sin(0.655\tau + 2.27)$ (7.3.2)

Heave motion: $z(\tau) = 0.1455\sin(0.655\tau + 0.86)$ (7.3.3)

The periodic rotational excitations are:

Roll motion: $\theta_1(\tau) = 0.044\sin(0.655\tau - 1.68)$ (7.3.4)

Pitch motion: $\theta_2(\tau) = 0.035\sin(0.655\tau - 2.33)$ (7.3.5)

Yaw motion: $\theta_3(\tau) = 0.011\sin(0.655\tau - 0.052)$ (7.3.6)

For this case, the representative distance Δx is 0.05, which yields the total number of fluid particles and total number of particles being 52,200 and 68,670, respectively. The time step is chosen as 0.012. The total simulation time is about 10 periods, which takes nearly 66.6

CPU hours on normal PC with Linux system Inter (R) Core (TM)2 CPU 6300 @ 1.86GHz. Based on Wu, Ma and Eatoch Taylor (1998) the natural frequencies in the three dimensional cases are given by

$$\varpi_{mn}^2 = \sqrt{(\frac{m\pi}{L})^2 + (\frac{n\pi}{B})^2}\ \tanh\sqrt{(\frac{m\pi}{L})^2 + (\frac{n\pi}{B})^2}, m, n = 0,1,2... \qquad (7.3.7)$$

where L and B are the length and width of the tank, respectively. Among them, the terms with *m=1,3,5..., n=0* and *n=1, 3, 5, ..., m=0* correspond to the symmetric motions in the x and y directions, respectively. Thus in this case, $\varpi_{10} = 0.6403$ and $\varpi_{01} = 1.433$ are the lowest natural frequencies in *x*-direction and *y*-direction, respectively.

Fig. 7.3.2 shows snapshots of the particles location in different instants of the simulation, in which the black particles represents the free surface particles. Due to the fact that the exciting frequency is close to the lowest natural frequency in *x*-direction of the tank, the free surface seems to be violent. The sloshing water hits the ceiling of the tank. The plunging jet is formed which generates many water droplets. One of the wave profiles before hitting the ceiling is shown in the Fig. 7.3.2 (a). It can be observed that the whole free surface is tilted due to the motion of tank and rush to the ceiling of the tank. One can also see that the plunging jets are formed after hitting the ceiling of the tank in Fig. 7.3.2 (b) and (c) and that fragmentation of the plunging jet occur. In addition the free surface, represented by black particles in Fig. 7.3.2, is very clear even after the plunging jet is formed, which again implies that the technique MPAM works well even in 3D cases.

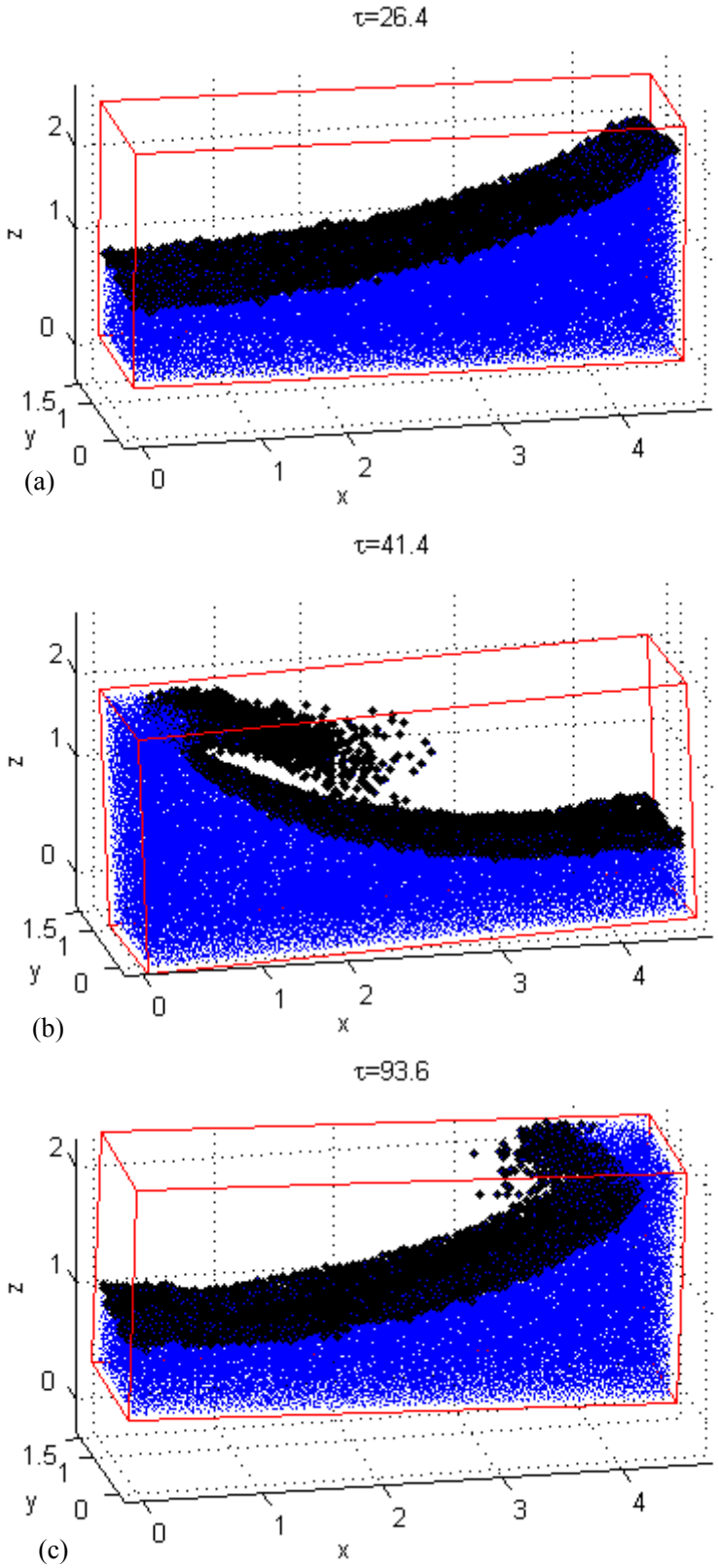


Fig. 7.3.2 Snapshots of violent slosh wave
(black particles: free surface particle; blue: velocity field)

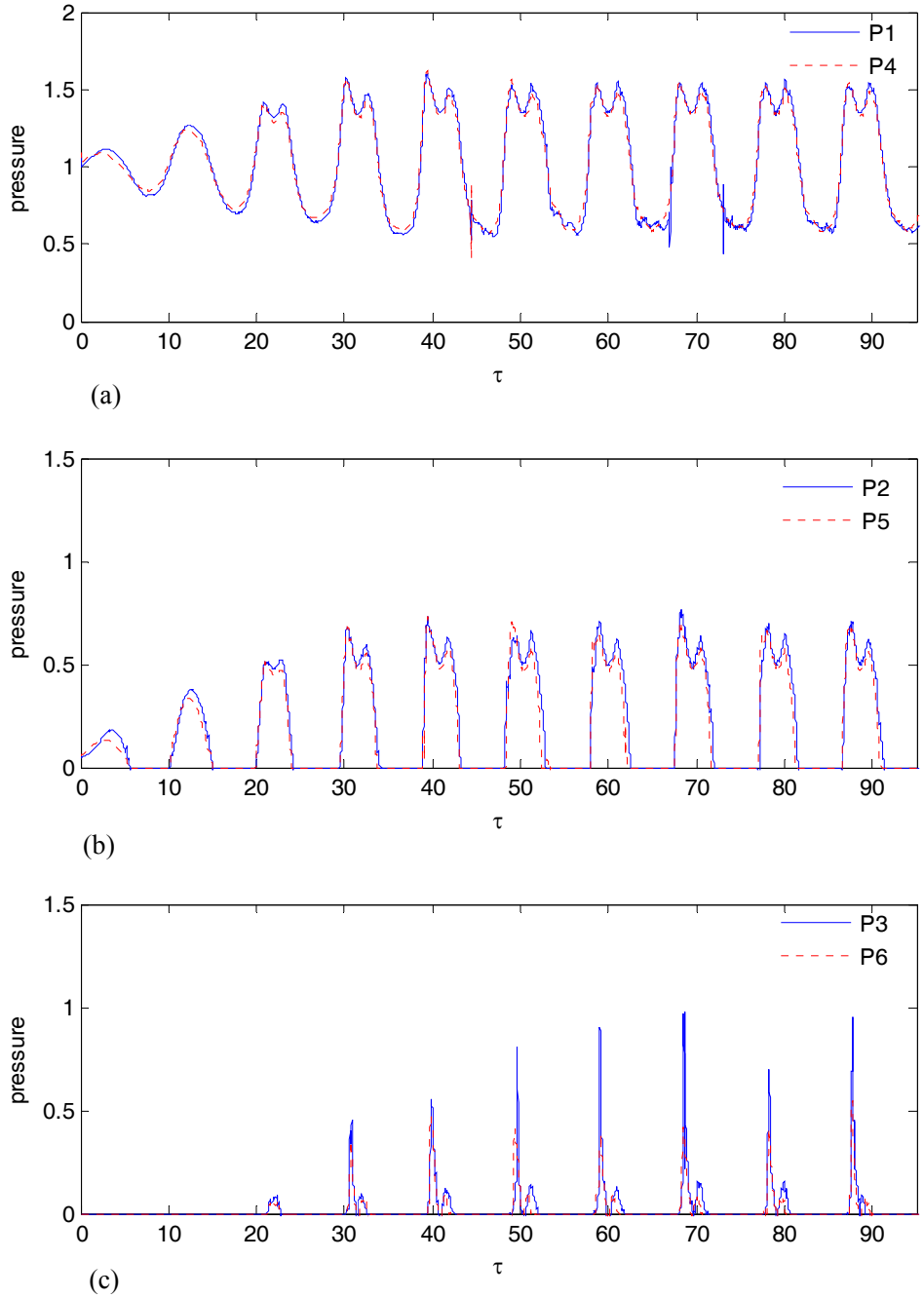


Fig. 7.3.3 Comparisons of the time histories of impact pressure on different record points

The pressure time histories recorded at different points are compared and some results are shown in Fig. 7.3.3. From Fig. 7.3.3 (a) and (b), one can see that the pressure curves become nearly periodic from the sixth exciting periods. There is no obvious difference existing in the pressure time histories between P1 and P4, and P2 and P5. However in the Fig. 7.3.3 (c), one may observe that the pressure curves for P3 and P6 are significantly different, particularly the crest value. The pressure peaks recorded at P3 are much bigger

than those at P6. That is because P3 is at the corner, where more momentum change leads to bigger pressure peak. The above 3D case demonstrates the capability of MLPG_R method in simulating violent sloshing cases.

8. 3D NUMERICAL INVESTIGATIONS ON VIOLENT WAVE IMPACT ON THE CYLINDER: OFFSHORE ENERGY STRUCTURE

Offshore wind energy has been considered to make significant contribution to provision of energy in future. A great deal of knowledge about onshore wind energy utilization has been accumulated. Although such knowledge may be very helpful for the development of offshore wind energy technologies, there are many issues which do not matter for onshore wind energy exploitation but that must be addressed for offshore wind energy system. One of them is the violent wave impact on the structures under action of breaking waves. The breaking waves may result in large impact pressure/forces on them and cause their damage. To avoid such damage and to guide the design of these structures, a good understanding of the interaction between breaking waves and offshore wind energy structures is required.

Efforts have been made to investigate breaking waves and violent wave impacts acting on structures. Due to the high degree of complexity of such problems, numerical simulations are rarely seen in the early studies and, therefore, the laboratorial experiments or field observations are mainly performed, such as Mogridge and Jamieson, (1980); Blackmore and Hewson, (1984); Chan and Melville, (1988); Kirkgöz, (1982, 1991, 1995); Neelamani, et al (1999); Bullock, Crawford, Hewson, Walkden and Bird, (2001). These experiments produced very useful results for some specific cases but are generally expensive. In this chapter, MLPG_R method is presented to simulate the interaction between breaking waves and 3D offshore wind energy structures.

In this study, the wind energy structure is considered as a vertical cylinder. To model its interaction with breaking waves, a solitary wave, which is generated by a piston wavemaker and overturns near the cylinder due to a sloping seabed, is considered as a representative. The motion equation of the wavemaker are Eqs. (6.0.1) - (6.0.3) for generating such waves. The bottom geometry is chosen as sloping seabed with different slopes at different positions as

sketched in Fig. 8.0.1. The offshore wind energy structure has a diameter of 4.3m and the water depth at its position is 10m.

As indicated above, the simulation is carried out in a numerical wave tank. For convenience, the water depth near the wavemaker is d. The sloping seabed starts from $x=5d$ and truncated at $x=10d$. For $x>10d$, a flat seabed is $0.5d$ under the mean free surface. The radius of the cylinder is $0.1d$ and the width of the tank is taken as $1d$ for this investigation, as shown in Fig. 8.1.1.

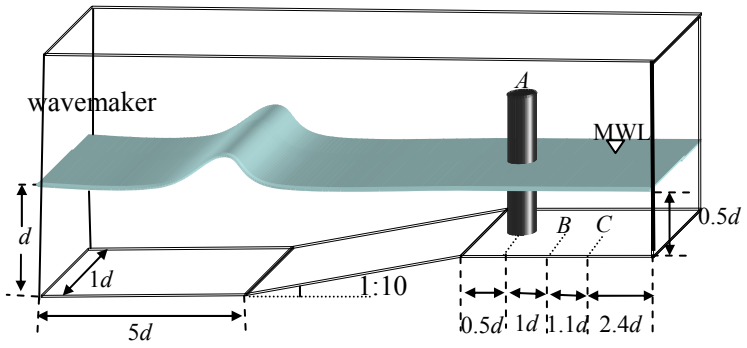


Fig. 8.0.1 Schematic view and details of tank and the structure

(*A, B* and *C* are the different locations of cylinder)

Again, the parameters presented below are nondimensionalised using the same method indicated before, for convenience of discussion. The spatial distance between two particles is taken as 0.0667, which yields the total number of particles of 55,888. The time step is 0.012. The computation was run on a PC with two 1.86GHz processors (one is used) and 4.8G RAM in Linux system and needs 21.1 hours. Two points, i.e. Point 1 and Point 2, on the front side of the cylinder are assigned to record pressure; they are 0.1 and 0.3 above the mean water level (MWL).

8.1 Investigations of relationship between pressure impact peak and breaking waves

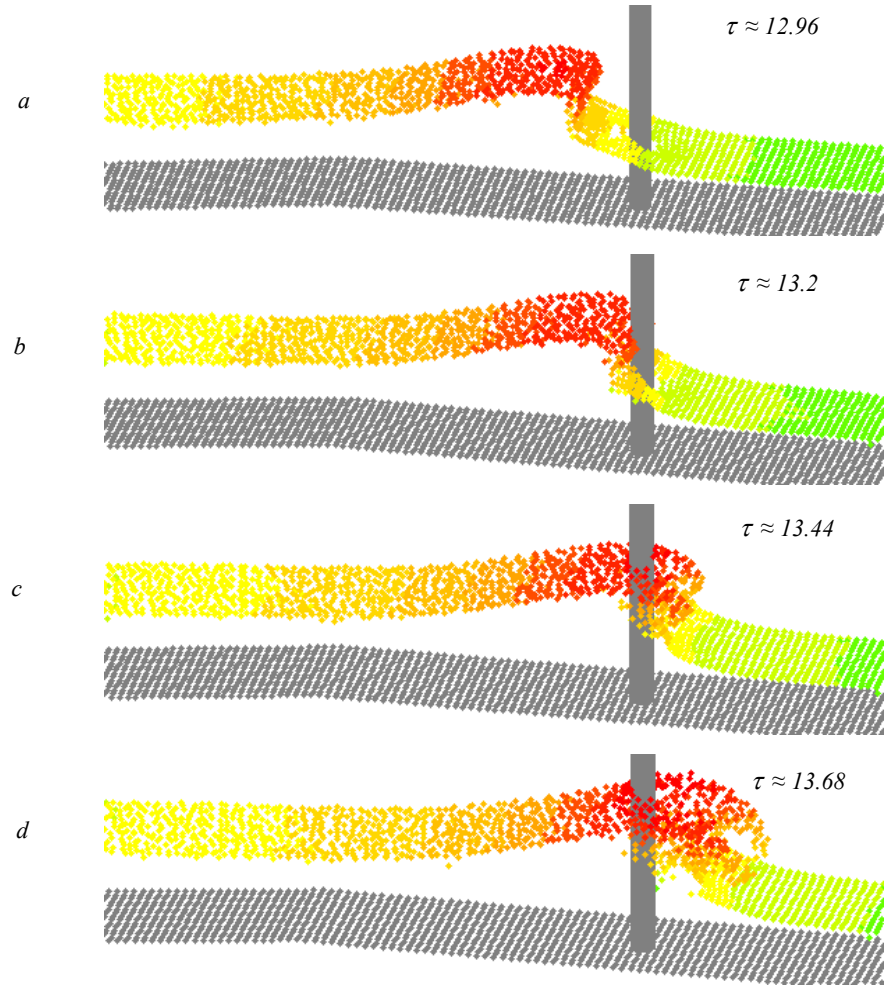


Fig. 8.1.1 Different stages of wave impacting on the wind turbine structure

(solitary wave height is 0.7, cylinder location is *C,* which is shown in Fig. 8.0.1)

In the first case considered here, the solitary wave height is taken as 0.7. The structure is centered at location *C*, whose distance is 2.6 from where the sloping seabed truncated. Some snapshots of the free surface near the structure are illustrated in Fig.8.1.1, where τ represents the dimensionless time and the gray particles on the bottom represent the seabed geometry and colorful particles are free surface particles with different color denoting different wave elevation. The solitary wave is generated by the wavemaker on the left side of the tank and propagates toward the structure. During its propagation, the solitary wave becomes steeper and overturning (Fig. 8.1.1a). The overturning wave hits the structure (Fig.

8.1.1b) in very short time and results in a huge pressure on the structure. To view how large the pressure could be, the time histories of the pressure recorded at the two points on the front surface facing the incoming waves are plotted in Fig. 8.1.2. As can be seen from this figure, at $\tau \approx 13.2$, when the overturning jet hits the structure (Fig. 8.1.1b), the pressure at both two points reaches their first crests in very short time. At this moment, the pressure acting on the Point 1 and Point 2 are very close. These pressure values are much larger than the maximum hydrostatic pressures on the structure. One may also find that for the pressure at Point 2, the pressure suddenly increase to its peak value. This is clearly due to the impact from the breaking waves.

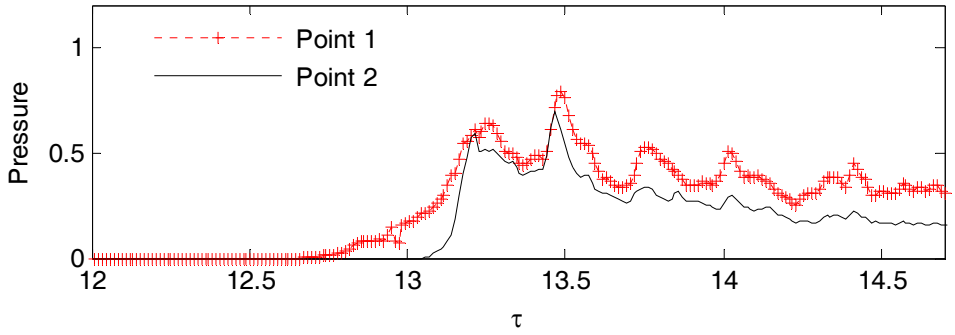


Fig. 8.1.2 Time histories of pressure acting on the wind energy structure (solitary wave height is 0.7, location: *C;* Point 1: 0.1 above the MWL; Point 2: 0.3 above the MWL)

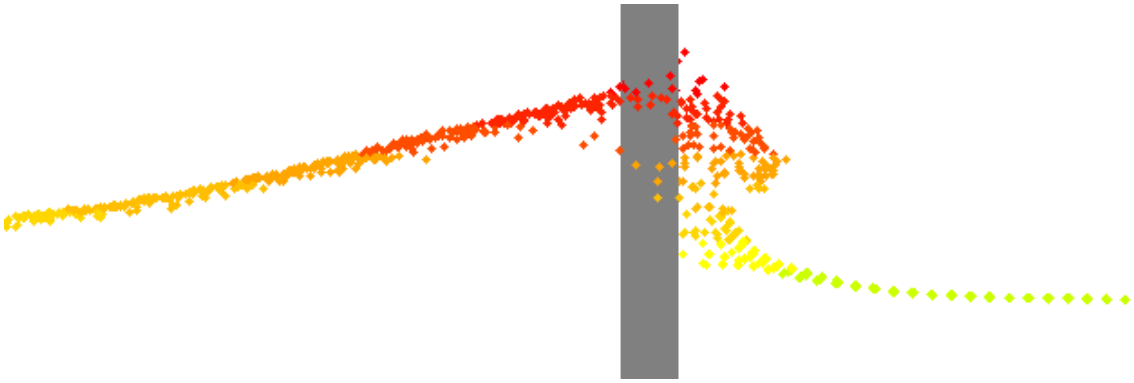


Fig. 8.1.3 Enlargement of free-surface particle distribution near structure viewed by *x*-*z* coordinates (solitary wave height is 0.7, location: *C*, $\tau \approx 13.44$)

Fig. 8.1.1 also illustrates the corresponding results after the plunging jet hits the structure. Fig. 8.1.1c shows that the major jet has been split into two jets when passing through the

structure. At this moment, not only the breaking jet, but also the main body of the wave crest hit the structure. It is clearer in Fig. 8.1.3 which displays the enlargement of the free surface particles around the structure. This impact causes a second peak value of the pressure, which is larger than the first one, as can be seen from Fig. 8.1.2 ($\tau \approx 13.44$). After this, the plunging jets keep moving forward and touch the free surface as shown in Fig. 8.1.1d. Due to this, a third peak value of pressure occurs as shown in Fig. 8.1.1d ($\tau \approx 13.68$). This case reveals a different behaviour of the pressure time history from that of non-breaking waves, in which the peak value of pressure accompanied with the occurrence of the wave crest. For the cases with breaking waves, there may be two or more pressure peaks corresponding to one wave crest. Furthermore, it takes very short time for the pressure changes from zero to a large value at the beginning of the impact. This causes a high rate of the change of the pressure and may lead to the damage of the structure. Due to this fact, the interactions between structures and breaking waves need to be carefully investigated.

8.2 Effects of Locations of Structures

For regular non-breaking oscillating waves without considering seabed effects, the feature of the interaction between structures and waves may not be affected by the location of the structures. However, for the cases with breaking waves, the effect of the location must be considered. This is understandable. For example, for the structures located near the position where the breaking waves occurs (breaking points), the results must differ from those in the case with the structures being located far away from the breaking point. The effects of the location of the structures are further investigated in this section.

For this purpose, the structures at three different locations, i.e. A, B and C as sketched in Fig.8.0.1, are considered. All other parameters are the same as those in Fig. 8.1.1. The free surface profiles at $\tau \approx 13.2$ in these cases are shown in Fig. 8.2.1. From this figure, it is found that the location and time where an overturning jet appears are almost the same in these cases. However, in the cases with the structure located at A and at B (Fig. 8.2.1a, b), the overturning occurs after the structure while for the case shown in Fig. 8.2.1c, the breaking jet starts at the same position as the structure. In addition, the structures have separated the breaking jet into two and results in turbulent breaking waves in the first two cases.

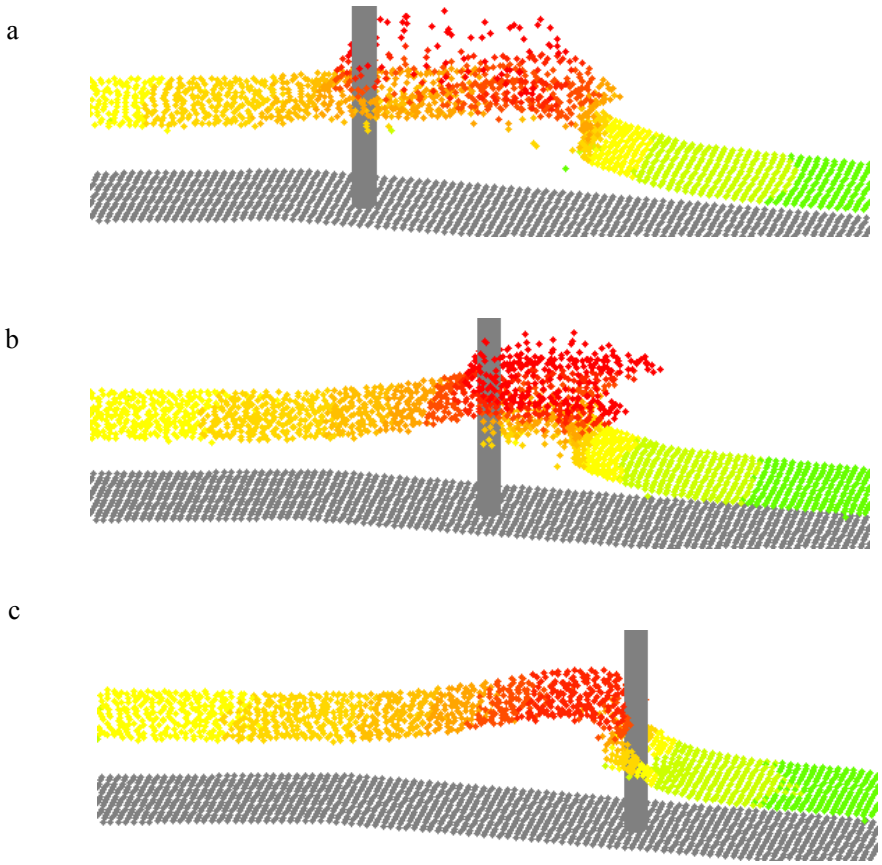


Fig. 8.2.1 Free surface profiles near the structures at $\tau \approx 13.2$, which are in different locations (solitary wave height is 0.7, a for Location A, b for Location B and c for Location C)

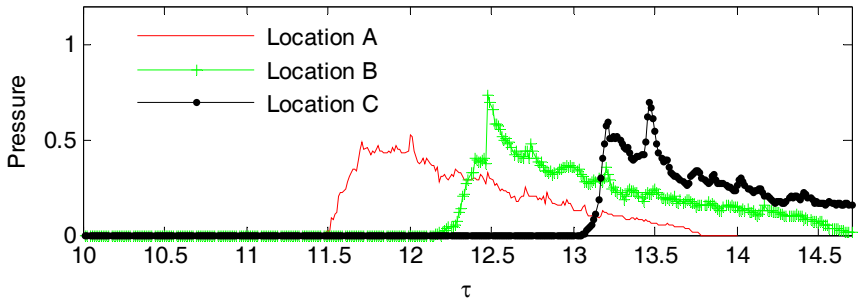


Fig. 8.2.2 Pressure time histories at Point 2 on the structures located at different positions (Point 2: 0.3 above the MWL)

The corresponding pressure time histories recorded at Point 2 are displayed in Fig. 8.2.2. It is found that for the case with the structure at location A, the pressure increases more slowly than in the other two cases. It can also be found that the maximum pressures in

different cases are different. For Location A, the value is the smallest while for Location B, it is the largest. The reason may be that the wave becomes vertical at the Location B. From the Fig. 8.2.3, one can note that the wave crest is very steep when time is close to 12.48. The whole vertical water column hit the structures at the same time, which yields the maximum pressure peak. This is not like the case shown in Fig. 8.2.1c and Fig.8.1.1, where the overturning jet hits the structure first and then the rest part of the water follows. Therefore, the maximum pressure in this case is larger than that shown in Fig. 8.1.1. For the same reason, there is no clear second peak observed in the case of Location B. Based on this observation, one may conclude that the pressure on the structure strongly depends on where the structure is located and its largest value can be reduced by selecting favorable location for it. It is noted that this investigation is preliminary. A detailed investigation will be carried out in the near future.

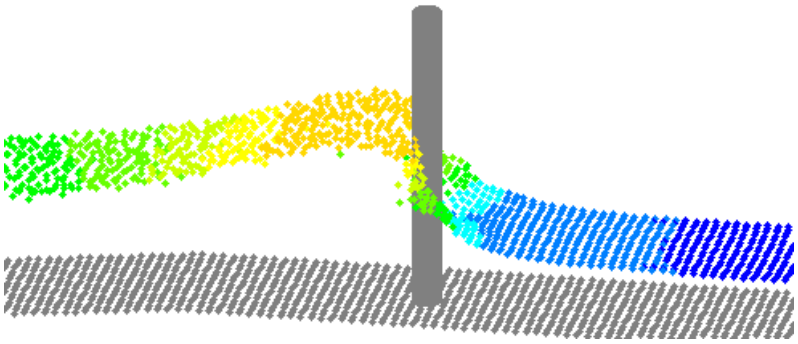


Fig. 8.2.3 Free surface profile near the structure at $\tau \approx 12.48$ at location B

8.3 Effects of different wave heights

Investigation is also made on the effect of the wave heights (H) on the wave impact on the wind energy structures. For this purpose, different solitary wave heights, ranging from 0.45 to 0.7, are specified. Other parameters are the same as the case shown in Fig.8.2.1. The wind energy structure is located at Location A, near the position where the sloping seabed truncated.

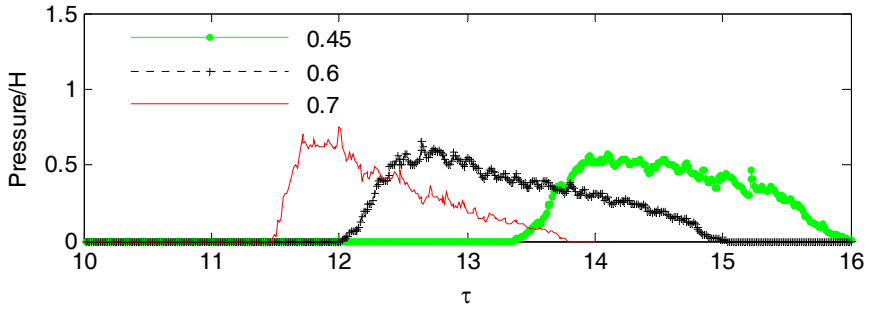


Fig. 8.3.1 Pressure time histories at Point 2 corresponding to different solitary wave heights (Point 2: 0.3 above the MWL).

Fig. 8.3.1 illustrates the pressure time histories recorded at Point 2 for the cases with different wave heights. As can be seen, the maximum pressure appears earlier as wave height increase. This is because the solitary wave with higher wave height has larger celerity and therefore travels faster. In addition, it is found that the nondimensionlized pressure profiles are different for different wave heights, being flatter for smaller wave height.

9. CONCLUSIONS AND FUTURE WORK

In this study, the MLPG_R method based on the Navier-Stokes model has been further developed in order to simulate breaking waves and their interactions with structures. In this method, the computational domain is discretized into many particles without meshes. The pressure governing equation is integrated over the sub-domain of each particle, which is a circle for 2D cases or a sphere for 3D cases. In order to solve the boundary value problem for pressure, comparative studies on three numerical implementations of solid boundary condition are carried out and the best one is the imposing of the condition about the normal derivative of pressure directly on the boundary particles. The test results indicate that the pressure wiggles can be suppressed, while other numerical implementations likely lead to the larger wiggles unless the number of particles is very large. For the purpose of simulating the breaking waves, a numerical technique for identifying the free surface named Mixed Particle Number Density and Auxiliary Function Method (MPAM) is developed, which is able to accurately track the free surface in 2D and 3D violent breaking waves. In order to increase the computational efficiency of present method, semi-analytical methods for surface and domain integration in 3D cases are presented, which makes it possible to run the 3D cases in normal PCs when using MLPG_R method.

Numerical results obtained by the MLPG_R method have been compared with experimental data and the results from other methods in the public domain. The various cases with pre-breaking and post-breaking waves and with or without structures have been simulated and validated The cases included the breaking wave of solitary wave generated by a piston wavemaker, dam breaking cases, sloshing cases and the interactions between breaking waves and offshore wind energy structure. Good agreements have been achieved. The convergence property of this method has been also investigated for different cases and shows the good convergence property.

The more achievements details and conclusions are summarised below. It is then followed

by some recommendations on future work.

9.1 Numerical techniques

In order to achieve high computational efficiency and robustness, a couple of numerical techniques are developed in this work. For the purpose of tracking the free surface particles accurately, a new technique based on three auxiliary functions, named as MPAM (Mixed Particle Number Density and Auxiliary Function Method), is developed in this project. In this method, each particle has a local coordinate system and a support domain. The first auxiliary function is to judge if there is a free surface particle in its support domain based on local particle configurations at previous time step. The second auxiliary function is based on how the local particles distribute in its support domain; the third auxiliary function is also based on the configurations of local particles. There are four conditions formed based on the particle number density and the three auxiliary functions, if one of four conditions is met when they are checked sequentially, then the particle concerned will be identified as a free surface particle. With this new technique, the free surface can be identified accurately according to the numerical tests on the 2D and 3D cases.

In addition, semi-analytical methods for surface and domain integration in 3D cases are developed. The semi-analytical methods are divided into three steps: the first one is to divide an integration domain into several sub-domains; the second one is to assume functions to linearly vary over each sub-domain; the last one is to perform the integration over each sub-domain analytically. The numerical comparisons of surface integrals using the semi-analytical method and Gaussian quadrature method are made to investigate the effectiveness of the semi-analytical technique for surface integrals. Two different cases are considered: one is to solve the hydrostatic pressure; the other one is to simulate the propagation of solitary wave. Numerical results show that the time spent by Gaussian quadrature method is at least 11 times more than the one spent by the semi-analytical method with the same precision. With these new approaches, the solution for 3D problems can be performed on normal PCs.

9.2 Two dimensional and three dimensional sloshing cases

For this purpose, breaking sloshing waves in a container generated by oscillating the container are considered. According to investigations on sloshing waves, it is found that there are three pressure peaks and two trough values in each impact period under certain condition (40% filling rate and oscillating frequency closing the natural frequency of the container). The large and major pressure peak is first generated and then the pressure rapidly decreases to its first minimum value. After that, the pressure rises again to its second minor peak value and drop down quickly to the second minimum value. And then again quickly rises to the third pressure peak. The investigations have also demonstrated that the major peaks of the pressure do not occur at the appearance of the highest run-up; instead they take place before the highest run-up is reached. The reason may be that the major peaks are created by the rapidest change in the moving direction of water from horizontal to vertical during the short period after the fast moving water strikes the wall. The rapid change of the moving direction causes the rapid change of momentum and therefore the large increase of pressure near the wall. The reason of the second peak may be the rapid change of moving direction of water from leftward to rightward, which lead to the rapid change of momentum and cause the second peak. The generation of the third peak can also be explained by the change in the direction of water motion, which occurs when the water fall down and is forced to change towards horizontal direction by the container wall. The effects of baffles on the sloshing waves have also been investigated and it is found that the baffles can not always reduce the pressure amplitude in violent sloshing cases. This indicates that the variation of natural frequency due to a baffle must be carefully considered if it is large enough.

9.3 Interaction between breaking waves and a fixed cylinder

The MLPG_R method is used to preliminarily study the interaction between breaking waves and a fixed cylinder, which may be considered as the tower for offshore wind turbines. In the investigations, the cylinder is installed at the shallow water region and a typical sloped seabed is considered in the numerical wave tank. The solitary wave is generated by the piston wavemaker at the left side of numerical wave tank. Due to the seabed effects, the wave

overturning and breaking will occur. From the investigations, some interesting features of interaction between structures and breaking waves have been discovered. For example, It is found that two pressure peaks in one impact may be created if a large overturning wave hitting the structure and that the largest pressure does not correspond to the moment when the wave crest arrives at the structure but occurs after that. It is also found that the features of the pressure during impact depend on the relative position between the structure and breaking point of breaking waves. These findings may provide good guidelines on how to consider the impact pressure during the design of the structures.

9.4 Future work

Although the project has made significant advance in understanding of interactions between breaking waves and structures, many uncertainties still exist.

Because the viscous effects have an important role in the breaking and post-breaking stages, it is necessary to introduce turbulent models into MLPG_R method and further investigations need to be carried out on implementation of different and/or new turbulent models.

In modeling post-breaking wave stage case, there is obvious difference for the shape of air cavity between the experimental photographs and numerical results. The possible reason may be only the single phase flow considered. So the multiphase flow model will be introduced into numerical model, and the air compressibility should be taken into account when breaking wave occurs.

For the simulation of realistic 3D problems, larger computational domain and more particles are needed, in which the computational load significantly increases. Hence, the parallel computation should be considered for the cases of 3D calculations to increase the computational efficiency, especially in the case of the simulation of 3D multi-phase problems.

In this study, fixed obstacles are considered in the numerical model. To model some practical problems, some related numerical methods should be included to deal with the moving obstacles in 2D and 3D cases. MLPG_R method can be further used to investigate the coastal sediment transport.

REFERENCES

Atluri, SN, Zhu, T., 1998, A New Meshless Local Petrov-Galerkin (MLPG) Approath in Computational Mechanics, *Computational Mechanics*, Vo1 22, pp 117-127.

Atluri, SN, Shen, S., 2002, The Meshless Local Petrov-Galerkin (MLPG) Method: A Simple & Less-costly Alternative to the Finite Element and Boundary Element Methods, *CMES: Computer Modelling in Engineering & Sciences,* Vol 3 (1), pp 11-52.

Atluri, S. N., Han, Z. D., Rajendran, A. M., 2004, A New Implementation of the Meshless Finite Volume Method, Through theMLPG "Mixed" Approach, *CMES: Computer Modeling in Engineering & Sciences*, vol. 6, no. 6, pp. 491-514.

Atluri, SN, Liu, HT., Han, ZD., 2006, Meshless Local Petrov-Galerkin (MLPG)Mixed Finite DifferenceMethod for Solid Mechanics, *CMES: Computer Modeling in Engineering & Sciences*, vol. 15, no. 1, pp. 1-16.

Atluri, S.N., Zhu, T., 2000, New Concepts in Meshless Methods, *International J. Numerical Methods in Engineering,* Vol. 47 (1-3), pp. 537-556.

Avila. R., Atluri. SN., 2009, Numerical Solution of Non-steady Flows, Around Surfaces in Spatially and Temporally Arbitrary Motions, by using the MLPG method, *CMES: Computer Modelling in Engineering & Sciences*, Vol. 54, No.1, pp. 15-64.

Batina, T., 1993, A gridless Euler/Navier-Stokes solution algorithm for complex aircraft applications, *AIIA paper 93-0333*, American Institute of Aeronautics and Astronautics.

Batra, R.C., Ching, HK., 2002, Analysis of Elastodynamic Deformations near a Crack/Notch Tip by the Meshless Local Petrov-Galerkin (MLPG) Method, *CMES: Computer Modeling in Engineering & Sciences*, Vol. 3 (6), pp. 717-730.

Belytschko, T, Lu,Y.Y., Gu, L., 1994, element-Free Galerkin Methods, *Int. J. Numr. Meth. Eng.*, Vol 37, pp 229-256.

Biausser, B., Grilli, ST., Fraunie, P., 2003, Numerical Simulations of Three-dimensional Wave Breaking by Coupling of a VOF Method and A Boundary Element Method. In *Proc. 13th Offshore and Polar Engng. Conf.* (ISOPE03, Honolulu, USA, May 2003), 333-339.

Biausser B., S.T. Grilli, Fraunié P., Marcer, R., 2004, Numerical analysis of the internal

kinematics and dynamics of three-dimensional breaking waves on slopes. *Intl. J. Offshore and Polar Engng.*, 14(4), 247-256.

Blackmore, P.A., Hewson, P.J., 1984, Experiemnts on full-scale wave impact pressures, *Coastal Engineering* Volume: 08 Issue: 4, pp: 331-346.

Dalrymple, RA., Gomez-Gesteira, M., Rogers, BD., Panizzo, A, Zou, S, Crespo A, Cuomo, G and Narayanaswamy, M., 2010, Smoothed Particle Hydrodynamics for Water Waves, Ch 13 in ADVANCES IN NUMERICAL SIMULATION OF NONLINEAR WATER WAVES (ISBN: 978-981-283-649-6 or 978-981-283-649-7), edited by QW Ma, *The world Scientific Publishing Co.*

Dalrymple, RA., Rogers BD., 2006, Numerical modeling of water waves with SPH method. *Coastal Engineering*, 53, pp 141-147.

Delorme, L., Iglesias, A. S., Perez S.A., 2005, Sloshing Loads Simulation in LNG Tankers with SPH, *International conference on Computational Methods in Marine Engineering*, MARTINE, 2005.

Devrard, D., Marcer, R., Grilli, ST, Fraunie, P, Rey, V., 2005a, Experimental Validation of a Coupled BEM-Navier-Stokes Model for Solitary Wave Shoaliing and Breaking, *Proc. 5th Intl. on Ocean Wave Measurement and Analysis*, pp 166-176

Devard, D., Marcer, R., Grilli, ST., ASCE, M, Fraunie, P., Rey V., 2005b, Experimental validation of a coupled BEM-Navier-Stokes model for solitary wave shoaling and breaking, *Ocean Waves Measurement and Analysis, Fifth International Symposium Waves 2005*, 3rd-7th, 2005. Madrid, Spain.

Fabián Duarte., Raúl Gormaz., Srinivasan, Natesan., 2004, Arbitrary Lagrangian Eulerian method for Navier-Stokes equations with moving boundaries, *Comput. Meth. Appl. Mech. Eng.*, 193: 4819-4836.

Goring, DG., 1978, Tsunamis the Propagation of Long Waves onto A Shelf, *Report No. KH-R-38, W.M.Kech Laboratory of Hydraulics and Water Resources*, California Institute of Technology, Pasadena, CA. pp 337.

Gotoh, H., Sakai, T., 1999, Lagangian Simulation of Breaking Waves Using Particle Method, *Coastal Engineer Journal*, Vol 41, Nos 3 & 4, pp 303-326.

Gotoh, H., Ikari, H., Memita, T., Sakai, T., 2005, Lagrangian particle method for simulation

of wave overtopping on a vertical seawall, *Coast. Eng. J*. 47 (2 & 3), 157-181.

Gotoh, H., Sakai, T., 2006, Key Issues in the Particle Method for Computation of Wave Breaking, *Coastal Engineer*, Vol 53, pp 171-179.

Grilli, ST., Guyenne, P., Dias, F., 2001., A fully non-linear model for three-dimensional overturning waves over an arbitrary bottom, *Int. J. Numer. Meth. Fluid,* Vol 35, pp 829–867.

GINGOLD, RA., MONAGHAN, JJ., 1977, Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly notices of the Royal Astronomical Society*, Vol. 181, pp: 375-389

Guo, J., Tao, Z., 2003, Modidied Moving Particle Semi-implicit Meshless Method for Incompressible Fluid*, J. of Thermal Science*, Vol. 13, No.3, pp 226-234.

Guignard, S., Grilli, S.T., Marcer, R., Rey, V., 1999, Computation of shoaling and breaking waves in nearshore areas by the coupling of BEM and VOF methods. *In Proc. 9th Offshore and Polar Engng. Conf.* (ISOPE99, Brest, France, May 1999), Vol. III, 304-309

Harlow, FH., Welch, J.E., 1965, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, *Phys. Fluid 8,* 2182–2189.

Han. Z. D., Atluri, SN., 2004a, Meshless Local Petrov-Galerkin (MLPG) approaches for solving 3D Problems in elasto-statics, *CMES: Computer Modeling in Engineering & Sciences*, vol. 6 no. 2, pp. 169-188.

Han. Z. D., Atluri, SN., 2004b, A Meshless Local Petrov-Galerkin (MLPG) approaches for solving 3- dimensional elasto-dynamics, *CMC: Computers, Materials & Continua*, vol. 1 no. 2, pp. 129-140.

Han Z. D., Liu H. T., Rajendran, AM., Atluri, SN., 2006, The Applications of Meshless Local Petrov- Galerkin (MLPG) Approaches in High-Speed Impact, Penetration and Perforation Problems, *CMES: Computer Modeling in Engineering & Sciences*, vol. 14, no. 2, pp. 119-128.

Henning, B., Peter, W., 2000, Arbitrary Lagrangian Eulerian finite element analysis of free surface flow, *Comput. Methods Appl. Mech. Engrg.* 190: 95-109.

Hibi, S., Yabushita, K., 2004, A Study on Reduction of Unusual Pressure Fluctuation of MPS Method, *J. Kansai Soc. N. A., Japan*, No 241, pp 125-131.

Hirt, C. W., Nichols, B.D., 1981, Volume of fluid (VOF) method for the dynamics of free

incompressible flow of fluid with free surface, *Phys. Fluid 8:2182-2189.*

Hirt, C. W., Amsden, A. A., Cook, J. L., 1974, an arbitrary lagrangian-Eulerian computing method for all flow speeds, *Journal of computational physics*, 14, pp. 227-253.

Huerta, A., Liu W. K., 1988, Viscous flow with large free surface motion, *Comput. Meths. Appl. Mech. Engrg, 69:277-324.*

Issa R., Violeau, D., Lee, E.-S., Flament, H., 2010, Modelling Nonlinear Water Waves with RANS and LES SPH Models, Ch 14 in ADVANCES IN NUMERICAL SIMULATION OF NONLINEAR WATER WAVES (ISBN: 978-981-283-649-6 or 978-981-283-649-7), edited by QW Ma, The world Scientific Publishing Co.

Khayyer, A., Gotoh, H., 2007, Applicability of MPS method to breaking and post-breaking of solitary waves*, Annual Journal of Hydraulic Engineering*, *JSCE* 51, pp: 175-180.

Khayyer, A., Gotoh, H., 2008a, Development of CMPS method for accurate water-surface tracking in breaking waves, *Coastal Engineering Journal*, 50(2), 179-207.

Khayyer, A., Gotoh, H., 2008b, Refined simulation of solitary plunging breaker by CMPS method*, Annual Journal of Hydraulic Engineering*, JSCE, 52, 121-126.

Khayyer, A., Gotoh, H., Shao, S.D., 2008c, Corrected Incompressible SPH method for accurate water-surface tracking in breaking waves, *Coastal Engineering,* 55(3), 236-250.

Khayyer, A., Gotoh, H., 2009, Modified MPS methods for prediction of 2D wave impact pressure, *Coastal Engineering,* 56(4), 419-440.

Khayyer, A., Gotoh, H., Shao, S.D., 2009, Enhanced predictions of wave impact pressure by improved incompressible SPH methods, *Applied Ocean Research*, 31(2), 111-131.

Khayyer, A., Gotoh, H., 2009, Wave impact pressure calculations by improved SPH methods, *International Journal of Offshore and Polar Engineering,* 19(4), 300-307.

Khayyer, A., Gotoh, H., 2010, A higher order Laplacian model for enhancement and stabilization of pressure calculation by the MPS method, *Applied Ocean Research*, doi: 10.1016/j.apor.2010.01.001, (article in press).

Kirkg□z, M.S., 1982, Shock pressure of breaking waves on vertical walls. *Journal of the Waterway, Port, Coastal and Ocean Divison*, v 108, n WW1, pp 81-95.

Kirkgoz, M.S., 1995, Breaking wave impact on vertical and sloping coastal structures, *Ocean Engineering*, v 22, n1, pp 35-48.

Klessfsman, KMT., Fekken, G., Veldman, AEP., Iwanowski, B., Buchner, B., 2005, A Volume-of-Fluid Based Simulation Method for Wave Impact Problems, *Journal of Computational Physics*, Vol 206, pp 363-393.

Koshizuka, S., Tamako, H., Oka, Y., 1995, A Particle Method for Incompressible Viscous Flow with Fluid Fragmentation, *Comput. Fluid Dynamics J.,* 4, pp 29-46.

Koshizuka, S., Oka, Y., 1996, Moving-Particle Semi-Implicit Method for Fragmentation of Incompressible Fluid, *Nuclear Science and engineering*: Vol 123, pp 421-434.

Koshizuka, S., Nobe, A., Oka, Y., 1998, Numerical analysis of breaking waves using the moving particle semi-implicitmethod. *International Journal forNumericalMethods in Fluids* 26, 751–769.

Koshizuka, S., Ikeda, H., Oka, Y., 1999, Numerical analysis of fragmentation mechanisms in vapour explosions, *Nuclear Engineering and Design* 189, 423-433.

Lee, B-H., and Park, J.-C., 2007, Numerical Simulation of Impact Loads Using a Particle Method, MPS, *Proceeding of ISOPE*, pp: 2029-2036.

Li, S., Atluri, SN., 2008, The MLPG mixed collocation method for material orientation and topology optimization of anisotropic solids and structures, *CMES: Computer Modeling in Engineering & Sciences*, 30 (1): 37-56.

Li, S.F., Liu. W.K., 2002, Meshfree and particle methods and their applications. *Applied Mechanics Review*, 55, 1-34.

Li Q., Shen,S., Han ZD., Atluri, SN., 2003, Application of Meshless Local etrov-Galerkin (MLPG) to Problems with Singularities, and Material Discontinuities, in 3-D Elasticity, *CMES: Computer Modelling in Engineering &Sciences*, Vol 4(5), pp 571-585.

Li, Y., Raichlen, F., 1998, Breaking criterion and characteristics for solitary waves on slope (discussion), *Journal of Waterway, Port, Coastal, and Ocean Engineering,* Vol. 124, 329-333.

Libersky, L.D., Petschek, A.G., Carney, T.C., Hipp, J.R., Allahdadi, F.A., 1993, High Strain Lagrangian Hydrodynamics: A Three-Dimensional SPH Code for Dynamic Material

Response, *Journal of Computational Physics*, Vol. 109, Issue 1, November, pp. 67-75.

Lin, H., Atluri, SN., 2000, Meshless Local Petrov-Galerkin (MLPG) method for convection-diffusion problems, *CMES: Computer Modelingin Engineering & Sciences*, Vol. 1 (2), pp. 45–60.

Lin, H., Atluri, SN., 2001, The Meshless Local Petrov-Galerkin (MLPG) Method for Solving Incompressible Navier-Stokes Equations, *CMES: Computer Modelling in Engineering &Sciences*, Vol 2(2), pp 117-142.

Lin, P. Z., Liu, Philip L.-F., 1998, A numerical study of breaking waves in the surf zone, *J. Fluid Mech, vol.* 359, pp. 239-264.

Lin, W.M., Newman, J.N., Yue, D. K. P., 1984, Nonlinear forced motion of floating bodies, *In Proc. 15th Symp. On Naval Hydrology*, Hamburg, Germany, pp 33-49.

Liu GR., Liu MB., 2003, Smoothed particle hydrodynamics: a meshfree particle method. *World Scientific*, Singapore.

Liu MB., Liu GR., 2006, Restoring particle consistency in smoothed particle hydrodynamics. *Appl Numer Math,* 56(1):19–36.

Liu HT., Han ZD., Rajendran, AM., Atluri, SN., 2006a, Computational Modeling of Impact Response with the RG Damage Model and the Meshless Local Petrov-Galerkin (MLPG) Approaches, *CMC: Computers, Materials,& Continua*,vol. 4, no.1, pp. 43-53.

Liu, HT., Han, ZD., Atluri, S.N., 2006b, Meshless Local Petrov-Galerkin (MLPG) Mixed Collocation Method for Elasticity Problems, *CMES: Computer Modeling in Engineering & Sciences*, vol.14, no. 3, pp. 141-152.

Lo, Edmond YM, Shao Songdong., 2002, Simulation of near-shore solitary wave mechanics by an incompressible SPH method, *Applied Ocean Research,* Vol 24, pp 275-286.

Lucy, L., 1977, A numerical approach to the testing of the fission hypothesis, *Astronomical Journal*, Vol. 82, pp: 1013-1024.

Lv, X., Zou, QP., Zhao, Y., Reeve, D., 2010, A novel coupled level set and volume of fluid method for sharp interface capturing on 3D tetrahedral grids, *Journal of Computational Physics*, Vol. 229, pp: 2573-2604.

Ma, Q.W., 1998, Numerical simulation of nonlinear interaction between structures and steep waves, *PhD Thesis, Department of Mechanical Engineering, University College London,*

UK.

Ma, Q.W., Patel M.H., 2001, On the nonlinear forces acting on a floating spar platform in ocean waves, *Appl. Ocean Res.*, 23:29-40.

Ma, QW., Wu, GX, Eatoch-Taylor, R,. 2001a, Finite element simulation of fully nonlinear interaction between vertical cylinders and steep waves. Part 1: Methodology and numerical procedure, *Int. J. Numer. Meth. Fluids,* 36: 265-285.

Ma, QW., Wu, GX, Eatoch-Taylor, R,. 2001b, Finite element simulation of fully nonlinear interaction between vertical cylinders and steep waves. Part 2: numerical results and validation, *Int. J. Numer. Meth. Fluids,* 36: 287-308.

Ma, QW 2005a, Meshless Local Petrov-Galerin Method for Two-dimensional Nonlinear Water Wave Problems, *Journal of Computational Physics*, Vol 205, Issue 2, pp 611-625.

Ma, Q.W., 2005b, MLPG Method Based on Rankine Source Solution for Simulating Nonlinear Water Waves, *CMES: Computer Modelling in Engineering &Sciences*, Vol 9, No 2, pp 193-209.

Ma, Q.W., 2007, Numerical Generation of FreakWaves Using MLPG_R and QALE-FEM Methods, *CMES: Computer Modelling in Engineering &Sciences*, Vol 18, No 3, pp 223-234.

Ma, Q.W., 2008, A New Meshless Interpolation Scheme for MLPG_R Method, *CMES: Computer Modelling in Engineering & Sciences*, Vol 23, No 2, pp 75-89.

Ma, Q.W., Duan, Wen Yang, *Zhou, J.T.*, Zheng, Xing, Yan, S., 2009, *Numerical Study on Impact Pressure due to Violent Sloshing Waves*, *Proceedings of ISOPE 2009, ISBN 978-1-880653-53-1, Vol 3, pp. 71-76.*

Ma, Q.W., Yan, S., 2006, Quasi ALE Finite Element Method for Nonlinear Water Waves, *Journal of Computational Physics*, 212, pp 52-72.

Ma, Q.W.,Yan, S., 2009, QALE-FEM for Numerical Modelling of Nonlinear Interaction between 3D Moored Floating Bodies and Steep Waves*, International Journal for Numerical Methods in Engineering,* Vol. 78, pp. 713-756.

Ma, Q.W., Zhou, JT., 2009, MLPG_R Method for Numerical Simulation of 2D Breaking Waves, *CMES: Computer Modelling in Engineering &Sciences*, Vol 43, No 3, pp 277-303.

Martin, J.C., Moyce, W, J., 1952, An Experimental Study of the Collapse of Liquid Columns on a Rigid Horizontal Plane, *Philos. Trans. R. Soc. London Ser. A, 244-312.*

Miyata, H., 1986, Finite-difference simulation of breaking waves, *Journal of Computational Physics, Vol* 65, issue 1, pp. 179-214.

Miyata, H., Park, J.C., 1995, Ch. 5 Wave Breaking Simulation, *Advances in Fluid Mechanics, Potential Flow of Fluids*. In: Rahman, M., UK., pp. 149-176.

Monaghan, JJ., 1992, Smoothed Particle Hydrodynamics, *Annu. Rev. Astron. Astrophy.*, Vol 30.

Monaghan, JJ., 1994, Simulation Free Surface Flows with SPH, *JOURNAL OF COMPUTATIONAL PHYSICS*, Vol 110, pp 399-406.

MONAGHAN, JJ., KOS, S., ISSA, N., 2003, Fluid Motion Generated by Impact, *J. Waterway, Port, Coastal, and Ocean Engrg*, ASCE, 129, 6, pp: 250-259.

Naito, S., Sueyoshi, M., Ohahsi, T., 2003, Study on Particle Method with Small Size Calculation Domain*, J. Kansai Soc. N. A*., Japan, No.240, pp 109-144.

Nayroles, B., Touzot, G., Villon, P., 1992, Generalizing the Finite Element Method, Diffuse Approximation and Diffuse Elements, *Computational Mechanics*, Vol 10, pp 307-318.

Naito, S, and Sueyoshi, M., 2002, A Numerical Analysis of Viloent Free Surface Flow by Particle Method, *the proceeding of ISOPE PACOMS*, pp 219-226.

Nonura, K., Koshizuka, S., Oka, Y., Obata, H., 2001, Numerical Analysis of Droplet Breakup Beharior using Particle Method, *Journal of Nuclear Science and Technology*, Vol.38, No.12, pp. 1057-1064.

Okamoto, T., Kawahara, M., 1990, Two-dimensional sloshing analysis by Lagrangian finite element method, *International Journal of Numerical Methods in Fluids*, Vol. 11, pp. 453-477.

Onate, E., Idelsohn, S., Zienkiewicz, OC., Taylor, RL., Sacco, C., 1996, A stabilized Finite Point Method for Analysis of Fluid Mechanics Problems, *Comput. Methods Appl. Mech. Engrg*.139: pp 315-346.

Osher S., Sethian J., 1988, Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton--Jacobi Formulations, *Journal of Computational Physics, 79, pp. 12--49.*

Pan, XJ., Zhang HX., Lu YT., 2008, Numerical simulation of viscous liquid sloshing by moving-particle semi-implicit method, *Journal of Marine Science and Application*, Vol 7,

Num 3, pp: 184 ~ 189.

Randles P.W., Libersky L.D., 1996, Smoothed particle hydrodynamics: Some recent improvements and applications. *Computer methods in applied mechanics and engineering*, Vol 139, pp: 375-408.

Rider, W.J., Kothe, D.B., 1995, Stretching and tearing interface tracking methods. *AIAA paper 95-1717.*

Sellountos, E.J., Sequeira, A., Polyzos, D., 2009, Elastic transient analysis with MLPG(LBIE) method and local RBFs, *CMES: Computer Modeling in Engineering & Sciences,* Vol. 41, No. 3, pp. 215-242.

Sethian J.A., 1996, Level Set Methods: Evolving Interfaces in Geometry, *Fluid Mechanics, Computer Vision and Materials Science.* Cambridge University Press.

Sladek, J., Sladek, V., Tan, CL., Atluri, S.N., 2008, Analysis of Transient Heat Conduction in 3D Anisotropic Functionally Graded Solids, by the MLPG Method, *CMES: Computer Modeling in Engineering & Sciences,* Vol. 32 (3): 161-174.

Sladek. J., Sladek. V., Wünsche, M,. Zhang. Ch., 2009, Interface Crack Problems in Anisotropic Solids Analyzed by the MLPG, *CMES: Computer Modelling in Engineering & Sciences*, Vol. 54, No.2, pp. 223-252.

Shao, S., Lo, EYM., 2003, Incompressible SPH Method for Simulating Newtonian and Non_Newtonian with A Free Surface, *Adv. Water Resour.* 26, pp 787-800.

Shao, S., 2006, Simulation of breaking wave by SPH method coupled with k–ε model, *Journal of Hydraulic Research,* Vol. 44, No. 3 (2006), pp. 338–349.

Souli, M., Zolisio, J. P., 2001, Arbitrry Lagrangian Eulerian and free surface methods in fluid mechanics, *Comput. Methods Appl. Mech. Engrg.*, 191: 451-466.

Sueyoshi, M., Naito, S., 2002, A Study of Nonlinear Fluid Phenomena with Particle Method (part 2) Two Dimensional Hydrodynamic Forces, *J. Kansai Soc. N. A., Japan,* No.237, pp 181-186.

Sueyoshi, M., Naito, S., 2004a, An Improvement on Pressure Calculaation Scheme of MPS Method, *J.Kansai Soc. N.A., Japan,* No 242: 53-59.

Sueyoshi, M., Naito, S., 2004b, 3 Dimensional Simulation of Nonlinear Fluid Problem by Paritlce Method Over One Million Particels Parrallel Computing on PC Cluster, *J. Kansai*

Soc. N. A., Japan, No.241, pp 133-142.

Sueyoshi, M., 2009, Numerical Simulation of Tank Sloshing with Thin Plate Structures by using a Particle Method, *Proceeding of 19th ISOPE conference*, Osaka, Japan, pp: 303 ~ 307.

Sussman, M., SMEREKA, P., OSHER, S., 1994, A Level Set Approach for Computing solutions to Incompressible Two-Phase Flow, *J. Comp. Phys.* 114, pp: 146-159.

Tanaka, S., Kashiyama, K., 2006, ALE finite element method for FSI problems with free surface using mesh re-generation method based on background mesh, *International Journal of Computational Fluid Dynamics,* Volume 20, Issue 3 & 4 March 2006 , pages 229 – 236.

Teng, B., Zhao M., Bai W., 2001, Wave fiffraction in a current over a local shoal. *Coast. Eng.,* 42: 163-172.

TULIN, M. P., LANDRINI, M., 2000, Breaking waves in the Ocean and around Ships. *Proc. 23rd ONR Symposium on Naval Hydrodynamics*, Val de Reuil, France.

Violeau, D., Issa, R., 2007, Numerical modelling of complex turbulent free-surface flows with the SPH method: an overview, *Int. J. Numer. Meth. Fluids*; 53:277–304.

Wang, Q., Zheng, Y., Chen, C., Fujimoto, T., Chiba, N., 2005, Efficient Rendering of Breaking Waves using MPS Method, *J Zhejiang Univ Science A*. 7(6): 1018-1035.

Wu, G.X., Ma, Q.W., R. Eatock Taylor., 1998, Numerical simulation of sloshing waves in a 3D tank based on a finite element method, *Appl. Ocean Res*. Vol. 20 337–355.

Xie, H., Koshizuka, S., Oka, Y., 2007, Modeling the wetting Effects in Droplet Imoingement using Particle Method, *CMES: Computer Modelling in Engineering &Sciences*, Vol. 18, No. 1, pp 1-16.

Yan, S., Ma, Q.W., 2009, QALE-FEM for modelling 3D overturning waves, *International Journal for Numerical Methods in Fluids*, Volume 63 Issue 6, pp : 743-768.

Yasuda, T., Mutsuda, H., Mizutani, N., 1997, Kinematics of overturning solitary waves and their relations to breaker types, *Coastal Engng.*, 29, 317-346.

Yoon, H.Y., Koshizuka, S., Oka, Y., 1999, A particle-gridless hybrid method for incompressible flows, *Int. J. Numer. Meth. Fluid*, 30, 407–424.

Yoon, H.Y., Koshizuka, S., Oka, Y., 2001, Direct Calculation of Bubble Growth, Departure, and Rise in Nucleate Pool Boiling, *International Journal of Multiphase Flow* 27, pp 277-298.

Zhang, S., Morita, K., Kenji, F., Shirakawa, N., 2006, An Improved MPS Method for Numerical Simulations of Convective Heat Transfer Problems, *INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS*, pp 51: 31-47.

Zheng, J., Long, S., Xiong, Y., Li, G., 2009, A Finite Volume Meshless Local Petrov-Galerkin Mehtod for Topology Optimization Design of the Continuum Structures, *CMES: Computer Modelling in Engineering & Sciences,* Vol. 42, No.1, pp. 19-34.

Zhou, JT., Ma, Q.W., Yan, S., 2008, Numerical Implementation of Solid Boundary Condition in Meshless Methods*, Proceedings of ISOPE, 2008,* Vancouver, BC, Canada, Vol 3, pp 16-23.

Zhou, JT., Ma, Q.W., Zhang, L., Yan, S., 2009, Numerical Investigation of Violent Wave Impact on Offshore Wind Energy Structures using MLPG_R Method, *Proceedings of ISOPE*, ISBN 978-188065353-1, Vol 3, pp. 503-509.

Zhou, JT., Ma, Q.W., 2010, MLPG Method based on Rankine source solution for Modelling 3D Breaking Waves, *CMES: Computer Modelling in Engineering &Sciences*, Vol 56, No 2, pp 179-210.

APPENDIX A: PUBLICATION LIST

The following papers have been published, which are based on the research of this study.

[1] Zhou, J.T and Ma, Q.W. , "MLPG Method based on Rankine source solution for modelling 3D Breaking Waves", *Computer Modeling in Engineering & Sciences (CMES),* Vol. 56, No. 2, pp. 179-210, 2010.

[2] Ma, Q.W. and Zhou, J.T , "MLPG_R Method for Numerical Simulation of 2D Breaking Waves", *Computer Modeling in Engineering & Sciences (CMES)*, Vol. 43, No. 3, pp. 277-304, 2009.

[3] *Zhou, J.T.*, Ma, Q.W., Zhang, L. and Yan, S., 2009. "*Numerical Investigation of Violent Wave Impact on Offshore Wind Energy Structures using MLPG_R Method*", *Proceedings of ISOPE 2009, ISBN 978-1-880653-53-1, Vol 3, pp. 503-509*.

[4] Ma, Q.W., Duan, Wen Yang, *Zhou, J.T*, Zheng, Xing, Yan, S., 2009, "*Numerical Study on Impact Pressure due to Violent Sloshing Waves*", *Proceedings of ISOPE 2009, ISBN 978-1-880653-53-1, Vol 3, pp. 71-76.*

[5] *Zhou, J.T*, Ma, Q.W., and Yan, S., 2008, "Numerical Implementation of Solid Boundary Conditions in Meshless Methods", *Proceeding of ISOPE 2008*, ISBN 978-1-880653-70-8.

[6] Sriram, V, Yan, S, *Zhou, J.T* and Ma, Q.W, 2010. "*APPLICATIONS OF MLPG_R & SALE/QALE-FEM FOR WAVE - STRUCTURE INTERACTIONS*", 9th International Conference on Hydro-science and Engineering (ICHE) 2010.

APPENDIX B: Gradient Calculation Scheme

Before updating the velocity following the Eq. (3.2.5), the pressure gradient value should be evaluated numerically. Accuracy of the gradient calculation scheme will significantly affect the overall accuracy of MLPG_R method. Herein, the gradient of pressure is estimated by using the method developed by Ma (2008). According to numerical tests, this method leads to higher order of accuracy than those used by the MPS method, particularly in the cases where the particles are distributed irregularly (Ma, 2008). Only brief description will be given in the following section for completeness.

Assume a general function $f(\vec{r})$ is expanded into a Taylor series near point $\vec{r}_i$:

$$f(\vec{r}) = f(\vec{r}_i) + (\nabla f)_{\vec{r}_i} \cdot (\vec{r} - \vec{r}_i) + O(|\vec{r} - \vec{r}_i|^2) \tag{B1}$$

Moving the $f(\vec{r}_i)$ from the right side to the left side of Eq. (B1) results in

$$f(\vec{r}) - f(\vec{r}_i) = (\nabla f)_{\vec{r}_i} \cdot (\vec{r} - \vec{r}_i) + O(|\vec{r} - \vec{r}_i|^2) \tag{B2}$$

Multiplying $\frac{(\vec{r}_{J,x_m} - \vec{r}_{i,x_m})}{|\vec{r}_J - \vec{r}_i|^2} W(|\vec{r}_J - \vec{r}_i|)$ on both sides of Eq. (B2), taking the sum of equations at all the relevant particles and ignoring the error term, it follows that

$$\begin{aligned} \sum_J^N [f(\vec{r}_J) - f(\vec{r}_i)] \frac{(\vec{r}_{J,x_m} - \vec{r}_{i,x_m})}{|\vec{r}_J - \vec{r}_i|^2} W(|\vec{r}_J - \vec{r}_i|) &= \sum_J^N \frac{(\vec{r}_{J,x_m} - \vec{r}_{i,x_m})^2}{|\vec{r}_J - \vec{r}_i|^2} W(|\vec{r}_J - \vec{r}_i|)(f_{,x_m})_{\vec{r}_i} \\ &+ \sum_J^N \sum_{k=1,k\neq m}^{D_s} (\vec{r}_{J,x_k} - \vec{r}_{i,x_k})(f_{,x_k})_{\vec{r}_i} \frac{(\vec{r}_{J,x_m} - \vec{r}_{0,x_m})}{|\vec{r}_J - \vec{r}_i|^2} W(|\vec{r}_J - \vec{r}_i|) \end{aligned} \tag{B3}$$

where *N* is the total particle number in the support domain, particle *J* is the particle affecting particle *i* based on the distance between them, *W* is weight function and D_s is the dimensional number.

Alternatively, Eq. (B3) can be rewritten as

$$(f_{,x_m})_{\vec{r}_i} + \sum_{k=1,k\neq m}^{D_s} a_{i,mk} (f_{,x_k})_{\vec{r}_i} = C_{i,m} \qquad (m = 1,2,...,D_s) \tag{B4}$$

where *m=1,2* and *k=1,2*, x_k for *k=1,2* is *x* and z coordinates, respectively, and

$$C_{i,m} = \frac{1}{n_{i,x_m}} \sum_{J}^{N} [f(\vec{r}_J) - f(\vec{r}_i)] \frac{(\vec{r}_{J,x_m} - \vec{r}_{i,x_m})}{|\vec{r}_J - \vec{r}_i|^2} W(|\vec{r}_J - \vec{r}_i|) \tag{B5a}$$

$$a_{i,mk} = \frac{1}{n_{i,x_m}} \sum_{J}^{N} \frac{(\vec{r}_{J,x_m} - \vec{r}_{i,x_m})(\vec{r}_{J,x_k} - \vec{r}_{i,x_k})}{|\vec{r}_J - \vec{r}_i|^2} W(|\vec{r}_J - \vec{r}_i|) \tag{B5b}$$

and

$$n_{i,x_m} = \sum_{J \neq i}^{N} \frac{(\vec{r}_{J,x_m} - \vec{r}_{i,x_m})^2}{|\vec{r}_J - \vec{r}_i|^2} W(|\vec{r}_J - \vec{r}_i|) \tag{B5c}$$

Solving the system in Eq. (B4), one can find the gradient components. For example, in 2D cases, the formulas are listed as follows:

$$(f_x)_{\vec{r}_i} = \frac{C_{i,1} - a_{i,12} C_{i,2}}{1 - a_{i,12} a_{i,21}} \tag{B6}$$

$$(f_z)_{\vec{r}_i} = \frac{C_{i,2} - a_{i,21} C_{i,1}}{1 - a_{i,12} a_{i,21}} \tag{B7}$$

The 3D gradient calculation scheme is presented as follows. Solving the system in Eq. (B5), the 3D gradient components can be found and Eq. (B5) can be rewritten as

$$[E]\{F\} = [G]\{\delta f\} \tag{B8}$$

where

$$\{F\} = [(f_{,x})_{\vec{r}_i}, (f_{,y})_{\vec{r}_i}, (f_{,z})_{\vec{r}_i}]^T \tag{B9}$$

$$\{\delta f\} = \left[f(\vec{r}_1) - f(\vec{r}_i), f(\vec{r}_2) - f(\vec{r}_i), \cdots, f(\vec{r}_J) - f(\vec{r}_i), \cdots\right]^T \tag{B10}$$

[G] is an $3 \times N$ matrix with its components defined as

$$G_{mJ} = \frac{1}{n_{i,x_m}} \frac{(\vec{r}_{J,x_m} - \vec{r}_{i,x_m})}{|\vec{r}_J - \vec{r}_i|^2} W(|\vec{r}_J - \vec{r}_i|), \tag{B11}$$

[E] is an 3×3 matrix with its components defined as

$$E_{mk} = a_{i,mk} \quad (m \neq k) \quad and \quad E_{mm} = 1 \tag{B12}$$

For more details about this formulation, readers are referred to Ma (2008).