Doctoral Dissertation

Boundary Treatment for Particle-based Fluid Simulation with Solid Wall and Open Boundary Conditions

September 2021

Doctoral Program in Advanced Information Science and Engineering
Graduate School of Information Science and Engineering
Ritsumeikan University

BUI Tien Thanh

Doctoral Dissertation Reviewed
by Ritsumeikan University

Boundary Treatment for Particle-based Fluid Simulation with Solid Wall and Open Boundary Conditions

（粒子法流体シミュレーションにおける固定境界条件と開境界条件の設定方法）

September 2021

2021 年 9 月

Doctoral Program in Advanced Information Science and Engineering
Graduate School of Information Science and Engineering
Ritsumeikan University

立命館大学大学院情報理工学研究科
情報理工学専攻博士課程後期課程

BUI Tien Thanh

ブイ　ティエン　タイン

Supervisor: Professor NAKATA Susumu

研究指導教員：仲田　晋教授

Abstract

Open boundary conditions and solid wall boundary conditions are main topics of boundary treatment in particle-based fluid simulation. Appropriate boundary conditions (BCs) can significantly enhance the simulation results and reduce computation costs. This thesis mainly deals with three issues related to BCs of particle-based fluid simulation: (1) the complication in implementing open BCs, (2) the challenge in generating wall particles along implicit surfaces and (3) the difficulty of efficient determination of appropriate particle motion around free surfaces and implicit surfaces. These works aim to develop simplified schemes related to BCs that achieve fast computation without loss of accuracy.

On the first topic, simplified schemes for open BCs with inflow and outflow boundaries are proposed. The first scheme, a nonreflecting outlet boundary condition (NROBC) considered a combination of inflow/outflow algorithm and periodic boundary conditions. This approach helps conservation of total mass and insertion of inflow particles without re-arranging process. The second scheme is developed to overcome the challenges of different sizes of inlet and outlet. A formulation that estimates physical quantities at inflow/outflow zones using standard scheme of smoothed particle hydrodynamics is developed. This scheme can avoid complicated extrapolation process and contributes to simple and fast estimation. The advantages of the δ^+- smoothed-particle hydrodynamics (SPH) scheme, specifically the particle shifting technique (PST), were successfully used to correct the position, velocity, and pressure of particles. Thus, instability errors of particles were prevented. The proposed technique easily addressed the challenges of inflow boundary conditions with prescribed or nonprescribed values.

On the second topic, we proposed schemes for generation of layered boundary particles along implicit surfaces. In the first approach, two central problems related to wall boundary penetration and the truncated support of kernel function have been overcome. Boundary particles are uniformly distributed, along with implicit curves, thereby considering the effect of the boundary. The second approach is an improved boundary treatment technique to compare with the first approach. The techniques provide full treatment for complicated geometries even though at sharp corners without any additional treatment.

On the third topic, a modification of the PST is proposed to reduce computation costs in the free surface treatment. New formulations for particle shifting that prevents tensile instability and for solid BCs with implicit surfaces are proposed in order to achieve stability at the interaction of fluid and obstacles. This approach is based on the idea of controlling the effect of particles shifting and it can be a significant breakthrough to minimize costly processes related to ghost particles in the computation.

These proposed techniques were validated using several benchmark test cases, and the results obtained are consistent with reference solutions. The developed approaches are well suited for wall boundaries and the evolution of the flow field. The results demonstrate the robustness and versatility of the proposed techniques for various simulations.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Professor. Susumu Nakata, for his guidance, excellent support throughout my journey at Ritsumeikan University. I am very grateful to him for his recommendations and advice during my research project. Without his help, I would not have been able to finish my doctoral thesis successfully.

I also would like to thank Professor. Yoshihisa Fujita for sharing his valuable time and experience in our discussions. I am grateful to him for his advice and comments to improve my research project.

I would like to thank all my friends, Vietnamese, my Ritsumeikan friends, and Japanese laboratory friends who always helped and supported me in the life.

Finally, I dedicate this thesis to my parents, my wife Chu Dieu Hien, my daughter Bui Truc Linh, and my dear family members. They encourage and support me during my Ph.D. studies. The time and effort necessary to complete this research would not have been possible without their encouragement, dedication.

Contents

Contents iv

List of Figures vii

List of Tables xi

Chapter 1 Introduction 1

1.1 Introduction 1

1.2 Thesis objectives 2

1.3 Related works 3

1.4 Thesis outline 5

Chapter 2 Smoothed particle hydrodynamics (SPH) method with implicit surface 6

2.1 Mathematical formulations 6

2.1.1 SPH Formulations 6

2.1.2 Kernel function 7

2.2 Weakly compressible SPH 8

2.3 Time integration 10

2.4 The particle shifting technique (PST) 11

2.5 Boundary treatment 11

2.5.1 Open boundary conditions 11

2.5.2 Solid wall boundary condition 13

2.6 Implicit surface 14

Chapter 3 Simplified approaches in open boundary conditions 17

3.1 Non-reflecting outlet boundary condition 17

3.1.1 The proposed techniques 17

3.1.2 Simulation test cases 20

3.2 A simplified approach of open boundary conditions 27

3.2.1 The proposed techniques 27

3.2.2 Simulation test cases 29

Chapter 4 Improved boundary treatment in solid wall boundary conditions 35

4.1 Boundary treatment techniques using implicit surface 35

4.1.1 The proposed techniques 35

4.1.2 Simulation test cases 37

4.2 Improved boundary treatment technique for implicit surfaces 42

4.2.1 The proposed techniques 42

4.2.2 Simulation test cases 47

Chapter 5 A flexible boundary treatment for implicit surfaces 60
5.1 The proposed techniques 60
5.1.1 Modification of the particle shifting technique 63
5.1.2 The particle shifting technique with implicit surface 64
5.2 The simulation test cases 65
5.2.1 Flow past a circular cylinder 65
5.2.2 Dam break simulation 67
5.2.3 Fluid simulation in computer graphics 72
Chapter 6 Conclusions and future work 77
6.1 Conclusions 77
6.2 Future works 78
References 79

List of Figures

Figure 2.1 Influence region of the smoothing function W for particle i with a radius of kh....7

Figure 2.2 Periodic boundary condition....12

Figure 2.3 Implicit surface modeling....15

Figure 3.1 Initial sketch of the computational domain: different colours are associated to different sets of particles....18

Figure 3.2 Initial Computational domain of the open-channel flow....21

Figure 3.3 Particles distribution and velocity field at $\mathrm{t(g/H)1/2} = 100$ at $Re = 10$, 100 and 200....22

Figure 3.4 Comparisons between analytical solution and numerical results at $\mathrm{t(g/H)1/2} = 100$ for $Re = 10$, 100 and 200 at $x = 0$, $x = H$ and $x = 2H$....23

Figure 3.5 The Mean Square Error Percent (MSEP) for $Re = 10$ with the proposed technique and inflow/outflow by I. Federico [29]....23

Figure 3.6 The Mean Square Error Percent (MSEP) for $Re = 100$ and $Re = 200$ with the proposed technique....23

Figure 3.7 Particles distribution and velocity field at $\mathrm{t(g/H)1/2} = 100$ at $Re = 10$, 100 and 200....25

Figure 3.8 Comparisons between analytical solution and numerical results at $\mathrm{t(g/H)1/2} = 100$ for $Re = 10$, 100 and 200 at at $x = 0$, $x = H$, and $x = 2H$.....26

Figure 3.9 The Mean Square Error Percent (MSEP) for $Re = 10$ with the proposed technique and inflow/outflow by I. Federico [29]....26

Figure 3.10 The Mean Square Error Percent (MSEP) for $Re = 100$ and $Re = 200$ with the proposed technique....27

Figure 3.11 Sketch of the simplified approach of open boundary conditions, where kh is the kernel support....28

Figure 3.12 Sketch of the domain used for the backward-facing step simulations....30

Figure 3.13 Axial velocity for $\mathrm{Re} = 389$....31

Figure 3.14 (a) Velocity profile at the inlet and (b) reattachment length for Re = 389....31

Figure 3.15 Comparison of velocities at four different locations for $\mathrm{Re} = 389$.....32

Figure 3.16 Sketch of the domain used for the backward-facing step simulations.....33

Figure 3.17 Axial velocity for $\mathrm{Re} = 100$.....33

Figure 3.18 Comparison of the axial velocities at four different marked positions, P1–P4, for $\mathrm{Re} = 100$.....34

Figure 4.1 The distribution of fluid particles(blue), fluid particles close the wall (black circle) and grid points particles (green circle)...36

Figure 4.2 Generation of wall particles using proposed technique..36

Figure 4.3 Computational domain for 2-D flow past a circular cylinder...............................37

Figure 4.4 Time history of drag and lift coefficients for flow past a circular cylinder Re = 20 and 200...38

Figure 4.5 2-D flow past a cylinder: velocity field and pressure field at Re $= 20$................39

Figure 4.6 2-D flow past a cylinder: velocity field and pressure field at Re $= 200$ 39

Figure 4.7 2D flow past a cylinder: the velocity vector field at Re $= 200$. 40

Figure 4.8 A circular cylinder without a sector 41

Figure 4.9 Generation of wall particles..41

Figure 4.10 Time history of drag and lift coefficients for flow past the body at Re = 100 41

Figure 4.11 Vorticity field at $tUD = 5$ 41

Figure 4.12 2-D flow past the body: velocity field and pressure field at Re $= 100$ 42

Figure 4.13 Improved boundary treatment technique with implicit surface..........................43

Figure 4.14 Process of constructing ghost particles using IWFF: (a)–(i) sequential steps in the process..45

Figure 4.15 Process of constructing ghost particles using IWFG: (a)–(i) sequential steps in the process..46

Figure 4.16 Wall particles generated by IWFG along implicit curve...................................47

Figure 4.17 Cylinder model within a periodic lattice. 48

Figure 4.18 Paths for comparing the SPH and FEM solutions 48

Figure 4.19 Velocity magnitude around a cylinder and contour plots of the velocity magnitude for Re =1. 49

Figure 4.20 Comparison of the SPH and FEM velocity profiles along paths 1 and 2 for Re = 1..49

Figure 4.21 Comparison of the SPH and FEM pressure profiles along path 3 for Re = 1. 49

Figure 4.22 The velocity magnitude contours of the IBT for (a) Re =20, (b) Re = 30, and (c) Re = 40...51

Figure 4.23 Pressure curves along the path indicated in the inset for Re = 20. 51

Figure 4.24 Pressure curves along the path indicated in the inset for Re = 30. 52

Figure 4.25 Pressure curves along the path indicated in the inset for Re = 40. 53

Figure 4.26 Computational domain for the flow past an implicit surface. 55

Figure 4.27 Boundary treatment for the implicit surface at angles 1 and 2 (Fig. 4.26) using the IBT and BT techniques as follows: (a) improved wall particles from fluid particles, (b) improved wall particles from ghost particles, (c) wall particles from fluid particles, and (d) wall particles from ghost particles.. 55

Figure 4.28 Velocity magnitude around an implicit surface and contour plots of the velocity magnitude using the IBT and BT techniques for Re = 1. 56

Figure 4.29 Comparison of the SPH velocity profiles along path 1 (**Figure 4.26**) using the IBT and BT techniques for Re = 1. 56

Figure 4.30 Comparison of the SPH pressure profiles along path 2 (**Figure 4.26**) using the IBT and BT techniques for Re = 1. 56

Figure 4.31 The contour plots of velocity magnitude, particle positions and velocity magnitude using the IBT technique for Re=30 and Re=40 58

Figure 4.32 Comparison of the pressure profiles along the path 2 in Fig. 4.26 using the IBT and BT techniques for Re = 30 and Re = 40. 58

Figure 5.1 Simulation with the conventional PST 61

Figure 5.2 Normal vector of free surface particles 62

Figure 5.3 Particle distribution during the simulation 63

Figure 5.4 Two-dimensional flow past a cylinder: (a) velocity field and (b) particle distribution surrounding the implicit surface at $Re = 200$ 66

Figure 5.5 Time history of lift and drag coefficient..66

Figure 5.6 Sketch of 3D dam-break simulation 68

Figure 5.7 Numerical simulation of 3D dam-break simulation at different times T 69

Figure 5.8 Comparison of the time histories of water height evolution at point A 70

Figure 5.9 Comparison of the time histories of water height evolution at point B................70

Figure 5.10 Comparison of boundary particles in the proposed technique and conventional technique in dam-break simulation. 71

Figure 5.11 Dam break simulation past a solid dragon...73

Figure 5.12 Comparison of boundary particles in the proposed technique and conventional technique on a solid dragon 73

Figure 5.13 Fluid drops on sphere 74

Figure 5.14 Comparison of boundary particles in the proposed technique and conventional technique on solid sphere...74

Figure 5.15 Comparison of boundary particles in the proposed technique and conventional technique on solid bunny 75

Figure 5.16 Fluid sphere drops on solid bunny..76

List of Tables

Table 4.1 Comparison of the SPH and FEM pressures for Re = 1. 50
Table 4.2 Comparison of the SPH and MBT pressures for Re = 20. 52
Table 4.3 Comparison of the pressure values of SPH and the MBT technique for Re = 30. 53
Table 4.4 Comparison of the pressure values of SPH and the MBT technique for Re = 40. 54
Table 4.5 Comparison of the SPH pressure for Re = 1. 57
Table 4.6 Comparison of pressure for Re=30 59
Table 4.7 Comparison of pressure for Re=40 59
Table 5.1 Comparison of CD and CL values using different method for $Re = 200$ 67

Chapter 1 Introduction

1.1 Introduction

Nowadays, computer graphics has become a central part of our lives; it is used in movies, games, computer-aided design, virtual simulators, visualization, shape modeling, fluid simulation, and so forth. The development of three-dimensional (3D) content has been considered the main goal of computer graphics, and it has been extensively studied in several ways. There have been various 3D shape representations in computer graphics such as voxel grids [1, 2], point clouds [3,4], mesh-based [5,6], integrated surface patches [7], and implicit fields [8,9]. Presently, an implicit surface has risen to be a promising 3D shape representation. The creation of an implicit surface can be evaluated from 3D points at arbitrary resolutions, and the mesh/surface can be extracted by applying the classical marching cubes algorithm. This output representation enables shape recovery at arbitrary resolutions, and it is continuous and can handle different topologies. An important application of implicit surfaces is surface reconstruction from scanned surface points [10,11], and a number of surface reconstruction techniques have been proposed in the last two decades. The functions of implicit surfaces consist of wavelets [12], radial basis functions [13], B-spline bases [14] and piecewise polynomial [15]. Recently, several new approaches based on learning-based 3D reconstruction for implicit surfaces have attracted increasing attention in the research community [8,9]. It expresses the huge potential of 3D implicit representation and its application in various fields.

Fluid phenomena are an interesting topic in computer graphics. Various fluid phenomena include wind, weather, and ocean waves simulated in computational fluid dynamics. Several numerical methods have achieved outstanding results, such as SPH [16,17,39,83], Position based dynamics[18,84], Moving particle semi-implicit (MPS) [77,78], Vortex method [79,80] and so on. However, the 3D geometry in their research is typically represented by particles in Akinci [16,17] or polygonal mesh [18, 39,81]. Thus, these approaches require a large amount of memory to store particles or meshes, vertices, and so on throughout the simulation process. Moreover, determining whether a point is on the interior or exterior of an object is difficult. Presently, investigating a simulation with an implicit surface is unusual. Some pioneering research has been conducted to apply particle-based fluid simulation with implicitly defined walls [19,20]. When representing an obstacle using implicit surfaces, the smoothness of the object is obtained much easier than in other 3D obstacles. Hence, particle-based fluid simulation with implicit surface has probably become a new trend in the future.

Among particle-based methods, the SPH method is a robustness model for fluid animation in computer graphics [82,83,85,86]. SPH has emerged to be a fully-fledged technique in fluid animation with versatile effects for interactive free-surface scenarios. The SPH method was originally developed as a grid-less numerical method for simulating astrophysical problems [Lucy, 1977 [21]; Gingold & Monaghan, 1977 [22]]. Unlike Eulerian computational techniques, such as the finite volume and finite difference methods, SPH does not require a grid because derivatives are approximated using a kernel function to solve the governing equations. Boundary conditions are both crucial and challenging in this method [48,87,88,89]. It can consider a key aspect of numerical simulations and can seriously influence the correct and accurate representation of the solution. Thus, this research focuses on dealing with boundary treatment for the SPH method.

1.2 Thesis objectives

This thesis presents several developments and improvements to the SPH method to obtain good accuracy and fast computation. The objectives of this thesis can be briefed as follows:

- Propose simplified approaches to handle the complication in implementing open boundary conditions
- Propose new boundary treatments to overcome the challenge in generating wall particles along implicit surfaces.

- Propose a technique to address the difficulty of efficient determination of appropriate particle motion around free surface and implicit surfaces.

1.3 Related works

Over the past four decades, SPH has successfully been used to simulate complex problems ranging from magnetohydrodynamics and solid mechanics to fluid mechanics, including free surfaces, because of the simplicity of formulas and stability of the overall simulation [81,87,88,90,91,92]. Boundary conditions are a challenge in this method. Appropriate boundary conditions can dramatically enhance simulation results and computational time. Many approaches in boundary treatment proposed in both open boundary and solid wall boundary conditions.

Several authors have expressed an interest in incorporating open boundary conditions into the SPH method. In several SPH simulations, the simplest and fastest technique, i.e., the use of periodic boundary conditions [23,24], is often used in which particles are recycled with particles passing through the outlet and reinserted at the inlet. This technique is limited and inappropriate for simulations with different inflow/outflow cross-sectional lengths. Additionally, violations resulting from the outlet velocity field after reinsertion can cause an unstable state in long runtime simulations.

Lastiwka et al. [25] proposed permeable boundary conditions; however, this is difficult to implement with a free surface. A different approach relies on a generalization of the unified semi-analytical boundary condition method introduced by Ferrand et al. [26] and Leroy et al. [27]. This technique has been used to incorporate unsteady open boundaries into incompressible SPH and weakly compressible SPH models. Tafuni et al. [28] used mirrored particles at the fluid zone and a higher-order interpolation process of updating the values of inflow/outflow particles. However, these approaches have a high computational cost and are complicated. Federico et al. [29] presented an implementation of open boundary conditions based on inflow/outflow buffer layers with the inflow particle information assigned at the beginning of the simulation. Alvarado-Rodrıguez et al. [30] used a reservoir buffer to prevent a reflection of the velocity field at the outlet.

Vacondio et al. [31] introduced open boundary conditions using Riemann invariants. Tarksalooyeh [32] constructed a pre-inlet domain to feed the inflow region. Molteni et al. [33] and Altomare et al. [34] proposed a sponge layer to avoid spurious reflections from waves. Liu [35] used Taylor series expansion to extrapolate the values of inlet and outlet particles. Wang et al. [36] and Negi [37] selected a characteristic-based method to apply as a non-reflecting boundary condition in a WCSPH model. Most previous studies have usually opted to update inflow/outflow particle information via extrapolation from fluid particles. This results in complicated procedures and is time consuming.

In terms of solid wall boundary treatment, it is always challenging, especially for complicated geometry with sharp corners. To simplify the treatment of boundary conditions, a single layer of boundary particles can be used to represent the wall boundary in Monaghan [38], where the repulsive force is enforced by the Lennard–Jones (LJ) potential between fluid and boundary particles. This method can avoid the penetration of fluid particles through obstacles but suffers from spurious pressure behavior near the wall.

Harada et al. [39] proposed a polygon wall boundary condition to represent objects that eliminate boundary particles. This technique obtained good results in the case of planar wall boundaries but resulted in large pressure oscillations for non-planar geometries.

To accurately represent a wall boundary, the approaches are utilized with several layers of boundary particles to manage the deficiency of the support domain of the fluid particles close to a wall such as a mirror [46,47] and dummy particles [48,49]. Mirror particles can obtain a smooth pressure distribution in two and three dimensions. Yildiz et al. [47] proposed a multiple boundary tangent method to simulate 2D channel flow with an obstacle. This method can be extended to three dimensions. However, the boundary treatment is complicated and time consuming to be generalized to 3D. Adami et al. [48] used several layers of dummy particles to simulate complex geometries with slip and no-slip boundary conditions and obtained good results. Marrone et al. [49] proposed a technique called fixed ghost particles that performs well for simulating flows around bluff bodies for Reynolds (Re) numbers in the regime of Re ∈ [1, 2400]. However, this technique requires special treatment for sharp corners in complicated geometries.

Zhang et al. [50] proposed an initial boundary particle arrangement technique, where boundary particles are automatically and uniformly generated on polygons, complying with the shape of the boundary. However, this technique only applies to the precomputing step, and it has difficulty in handling sharp corners.

In all previously introduced techniques, the geometries of objects are represented by a polygon model or boundary particles. This leads to high memory consumption and high computational costs. Another approach to represent the geometries of objects that is promising in computer graphics comprises of so-called implicit surfaces. The main idea of implicit surfaces is to use implicit functions to define an object. Therefore, no polygonal mesh or boundary particles disappear during simulation, which helps reduce the large computational time. Nakata et al. [19] successfully applied fluid simulation to a 3D scene with implicit surfaces using SPH. Their simulation with a 3D dragon and trefoil knot opened the possibility of using SPH for many applications in computer graphics.

1.4 Thesis outline

In this thesis, the SPH theory as well as a WCSPH method and implicit surfaces are presented in Chapter 2. Simplified approaches for open boundary conditions were developed and described in Chapter 3. In Chapter 4, we proposed some new approaches for generating boundary particles for a solid body using implicit surfaces. A flexible boundary treatment for implicit surfaces is presented in Chapter 5. In Chapter 6, the conclusion of the research work is shown. Future works for further improvement are also suggested.

Chapter 2 Smoothed particle hydrodynamics (SPH) method with implicit surface

SPH is one of the earliest particle methods in computational mechanics. According to G.Liu [51], the key idea of the SPH method is as follows;

- A set of arbitrarily distributed particles without using any mesh that provides the connectivity of these particles. (***Meshfree***)
- Kernel approximation serves as the integral function representation. (***Integral function representation***)
- The integral representation of the field function and its derivatives with summations over all corresponding values at neighboring particles in a local domain called the support domain. (***Compact support***)
- Particle approximation at every time step, using the current local distribution of particles. (***Adaptive***)
- Field functions in PDEs to produce a set of ODEs in discretized form with respect to time only. (***Lagrangian***)
- The time history of all field variables for all particles. (***Dynamic***)

2.1 Mathematical formulations

2.1.1 SPH Formulations

The formulation of SPH is based on the integral representation which is called kernel approximation of field functions. From the statement that any function $f(x)$ can be represented in integral form, we can deduce the following:

$$f(x) = \int_{\Omega} f(x')\,\delta(x - x')dx' \tag{2.1}$$

where $f(x)$ is a function of the 3D position vector x , Ω represents the volume (contain x) of the integral, and $\delta(x - x')$ is the Dirac delta function defined by the convention:

$$\delta(x - x') = \begin{cases} 1, x = x' \\ 0, x \neq x' \end{cases} \tag{2.2}$$

Instead of using the Delta function $\delta(x - x')$, a smoothing function $W(x - x', h)$ is substituted into $f(x)$, and the integral representation of $f(x)$ becomes

$$f(x) = \int_{\Omega} f(x')\, W(x - x', h) dx' \tag{2.3}$$

where W is the so-called smoothing kernel function or kernel function; h represents the smoothing length which is used to determine the influence region with radius kh of smoothing kernel W for particle i, as shown in **Figure 2.1**

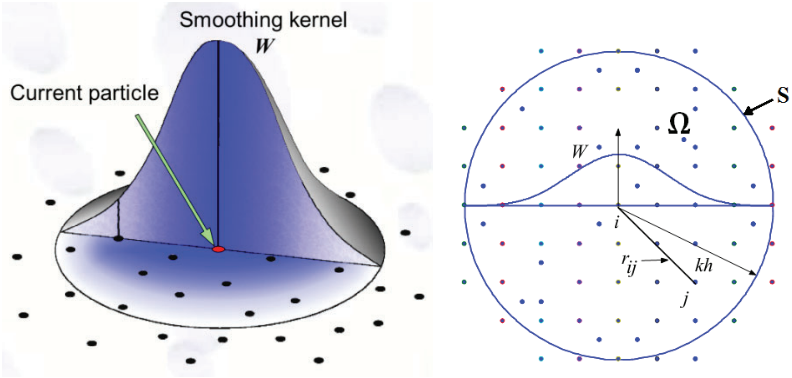


Figure 2.1 Influence region of the smoothing function W for particle i with a radius of kh

2.1.2 Kernel function

The kernel function significantly contributes to the value of the physical property in the reference particle. The kernel function must satisfy the following properties [52]:

Normalization: $\int_{\Omega} W(x - x', h)\, dx' = 1$

Positivity: $W(x - x', h) \geq 0$ inside the domain Ω

Compact support:$W(x - x', h) = 0$ out of the domain Ω

Delta function behavior: $\lim_{h \to 0} W(x - x', h) = \delta(x - x')$

Monotonically decreasing behavior: $W(x - x', h)$

A function that complies with these properties can be used as a smoothing kernel function. The selection of the kernel function is considered critical for determining the accuracy of the entire computation. A wide range of kernel functions such as Cubic spline kernel [93,94], Gaussian kernel [95,96], Quartic kernel [52,97], Wendland kernel [98,99,100], Quintic kernel [38,100]. In this work, we choose the Quintic spline kernel because of its stability and accuracy [38].

Quintic spline kernel (Morris 1997) – M6 spline

$$W(R, h) = \alpha_d \begin{cases} (3-R)^5 - 6(2-R)^5 + 15(1-R)^5 & for\ 0 \le R < 1 \\ (3-R)^5 - 6(2-R)^5 & for\ 1 \le R < 2 \\ (3-R)^5 & for\ 2 \le R < 3 \\ 0 & for\ R > 0 \end{cases} \tag{2.4}$$

Where α_d is $120/h$, $7/478\pi h^2$ and $3/359\pi h^3$ in one, two- and three-dimensional space, respectively. R is the relative distance between two particles.

2.2 Weakly compressible SPH

A theoretically incompressible flow becomes compressible flow by allowing 1% of density fluctuations to approach the weakly compressible concept. The incompressibility of fluid can be approximated by assuming a compressible fluid with a large speed of sound, typically Mach number of 0.1 to keep the density fluctuation small. The advantage of WCSPH is because it is robust and easier to program because of its explicit method. Pressure is computed explicitly according to the equation of state in Eq (2.7). However, because of the numerical speed of sound, the use of much smaller time steps increases the computational time and cost, and the use of the equation of state may cause large and nonphysical pressure fluctuations.

The governing equations for WCSPH in its Lagrangian form are

$$\frac{D\rho_i}{Dt} = -\rho_i \nabla \cdot \boldsymbol{u}_i, \tag{2.5}$$

$$\frac{D\boldsymbol{u}_i}{Dt} = -\frac{1}{\rho}\nabla \mathrm{p} + \frac{\mu}{\rho}\nabla^2 \boldsymbol{u}_i + \boldsymbol{F}_g, \tag{2.6}$$

where ρ, **u**, p, μ, and F represent the density, velocity, pressure, dynamic coefficient of viscosity, and body force, respectively.

To overcome the limitations of WCSPH, several variants of the WCSPH model have been proposed. The δ −SPH scheme was proposed by Antuono et al. [53,54]. This scheme adds a proper artificial diffusive term to the continuity equation to avoid spurious numerical high-frequency noise in the pressure field. Recently, the δ^+ −SPH scheme proposed by Sun et al. [53] has emerged as the most effective variant of the SPH model. This scheme is an improvement to the δ −SPH scheme. The main modification is the use of the particle shifting technique (PST) to remove tensile instabilities, which generate numerical cavitation. Therefore, this scheme ensures simulation accuracy and stability. The δ^+ −SPH model is written as:

$$\begin{cases} \dfrac{D\rho_i}{Dt} = -\rho_i \sum_j (\boldsymbol{u}_j - \boldsymbol{u}_i)\nabla_i W_{ij} V_j + \delta h c_0 \sum_j \psi_{ij} \nabla_i W_{ij} V_j \\ \dfrac{D\boldsymbol{u}_i}{Dt} = -\dfrac{1}{\rho_i}\sum_j \left(p_j + p_i\right)\nabla_i W_{ij} V_j + \dfrac{\mu}{\rho_i}\sum_j \pi_{ij}\nabla_{ij} V_j + \boldsymbol{F}_g \\ p_i = c_0^2(\rho_i - \rho_0) \end{cases} \tag{2.7}$$

where ρ_i, p_i, and $\boldsymbol{u}_i$ are the density, pressure, and velocity associated with the *i*-th particle, respectively, $\boldsymbol{F}_g$ represents the body force, V_i is the *i*-th particle volume defined as m_i/ρ_i; W_{ij} represents the Quintic kernel function and *j* denotes the particles inside the support domain of particle *i*; the gradient $\nabla_i W_{ij}$ indicates the differentiation with respect to the position of *i*-th particle; h is the smoothing length; $\psi_{ij}$and π_{ij} are diffusive and viscous terms ;μ is the dynamic viscosity; ρ_0 is a reference density; and c_0 is the speed of sound. c_0 is usually chosen such that:

$$c_0 \geq 10 \max\left(U_{max}, \sqrt{p_{max}/\rho_0}\right) \tag{2.8}$$

where U_{max} and p_{max} are the maximum expected velocity and pressure, respectively. According to the weakly compressible approach, this ensures a less than 1% density variation; additionally, the Mach number of the flow should be 0.1 or less.

The diffusive term and viscous term are:

$$\psi_{ij} = 2(p_j - p_i)\frac{\boldsymbol{r}_{ij}}{|\boldsymbol{r}_{ij}|^2} - \left[\langle\nabla\rho\rangle_i^L + \langle\nabla\rho\rangle_j^L\right] \tag{2.9}$$

$$\pi_{ij} = 2(n+2)\frac{(\boldsymbol{u}_j - \boldsymbol{u}_i)(\boldsymbol{r}_j - \boldsymbol{r}_i)}{|\boldsymbol{r}_j - \boldsymbol{r}_i|^2}$$

respectively, where $\boldsymbol{r}_i$ are the position associated with the i-th particle, and the renormalized density gradient is defined as

$$\langle\nabla\rho\rangle_a^L = \sum_b (p_b - p_a) L_a \nabla_a W(r_b) dV_b \tag{2.10}$$

$$L_a = \left[\sum_b (\boldsymbol{r}_b - \boldsymbol{r}_a) \otimes \nabla_a W(r_b) dV_b\right]^{-1}$$

The parameter δ is set equal to 0.2 in all simulations, and n is the number of dimensions

2.3 Time integration

We adopted the leapfrog method for the integration time. For stability, the time step Δt fulfilled the following criteria:

$$\begin{cases} \Delta t_v = 0.125\, min\frac{h_i^2}{\nu}, \Delta t_a = 0.25\, min\sqrt{\frac{h_i}{\|\mathbf{a}_i\|}}, \Delta t_c = CFL\, min\frac{h_i}{c_0}, \\ \Delta t = min(\Delta t_v, \Delta t_a, \Delta t_c), \end{cases} \tag{2.11}$$

where $\|\mathbf{a}_i\|$is the particle acceleration and the Courant–Friedrichs–Lewy (CFL) constant was set to CFL $= 1.5$ for all test cases.

2.4 The particle shifting technique (PST)

The PST was derived by Lind et al. [56] and is indispensable to SPH simulations [53,57]. In the δ^{+}-SPH scheme [53], the PST is modified to be well suited for WCSPH, as shown in Eq. (2.12).

$$\begin{cases} \dfrac{D\boldsymbol{r}_i}{Dt} = \boldsymbol{u}_i \,, \boldsymbol{r}_i^{*} = \boldsymbol{r}_i + \delta\boldsymbol{r}_i \\ \delta\boldsymbol{r}_i = -CFL\,.\,Ma\,.\,(2h_i)^2 \displaystyle\sum_j \left[1 + 0.2\left(\frac{W_{ij}}{W_{\Delta x}}\right)^4\right] \nabla_i W_{ij} \frac{m_j}{(\rho_i + \rho_j)} \end{cases} \tag{2.12}$$

Here, the symbols $\boldsymbol{r}_i$ and $\boldsymbol{u}_i$ indicate the position and velocity of the i-th particle, respectively, whereas $\boldsymbol{r}^{*}$ indicates the corrected position after applying particle shifting $\delta\boldsymbol{r}_i$. Ma is the Mach number of the flow.

After time integration, the particle position is shifted slightly by an amount $\delta\boldsymbol{r}_i$ to implement the PST. The corrected position is calculated and obtained at every time step. The pressure and velocity fields are corrected using Taylor series approximation (see Ref. [56]):

$$\phi_i^{new} = \phi_i^{old} + (\nabla\phi)\,.\,\delta\boldsymbol{r}_i + O(\,\delta\boldsymbol{r}_i^{2}), \tag{2.13}$$

where ϕ is a general variable. To avoid shifting errors, a limit is imposed on the particle shifting distance $\delta\boldsymbol{r}_i$:

$$\frac{\|\delta\boldsymbol{r}_i\|}{\Delta x} \leq 0.05 \tag{2.14}$$

2.5 Boundary treatment

2.5.1 Open boundary conditions

Open boundary conditions are difficult topics that have been the subject of numerous studies by various authors.

A periodic boundary condition [23,24] is considered as the simplest and fastest technique in which particles are recycled with particles passing through the computational domain as shown in **Figure 2.2**. It is applied for inlet and outlet particles in the direction of flow. Particles crossing the outlet are vice-versa into the flow domain at the inlet from the same y-coordinate positions with the same field values. The checking and adjustment for the x-direction are as follows:

If $x_i \geq x_{max}$then the new position of x_i is $x_i - L_x$

If $x_i \leq x_{min}$ then the new position of x_i is $x_i + L_x$

where,$L_x = x_{max} - x_{min}$is the length between two periodic boundaries. This checking and adjustment can also be used for y- and z-directions by changing the coordinate x to y and z.

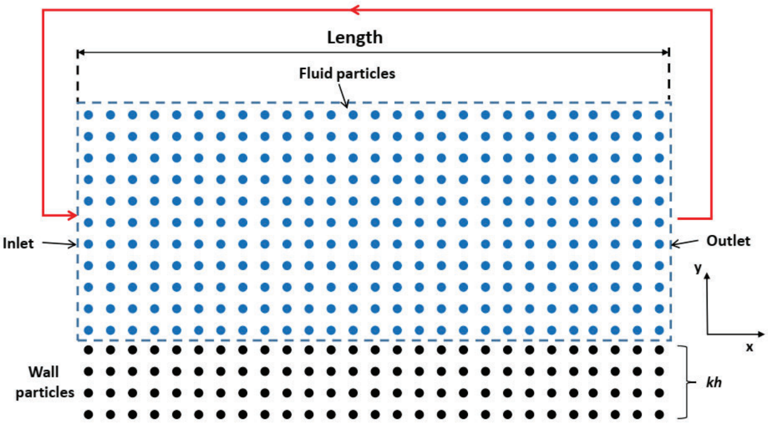


Figure 2.2 Periodic boundary condition

This technique is limited and is inappropriate for simulations with different inflow/outflow cross-sectional lengths. Additionally, violations resulting from the outlet velocity field after reinsertion can cause an unstable state in long runtime simulations.

A widely-used approach is the inflow/outflow boundary condition, where the computing domain is divided into four sets of particles as follows: fluid, wall, inflow, and outflow particles. The inflow zone is placed in front of the fluid zone so that the attached zone covers a region as wide as the kernel support. Inflow particles move according to their velocity until they cross the inflow zone and become fluid particles. In terms of fluid particles, their information evolves according to the SPH governing equations. The fluid particles, which cross the fluid zone, will become outflow particles. Outflow particles, which cross the outflow zone, will be eliminated, creating new inflow particles at the inflow region. In the case of the different velocity profiles at the outlet zone, the number of particles inserted into the computational domain is not as equal as eliminated particles. It leads to a loss of total mass, where the conservation of mass is violated. Moreover, the array structure of particles is distributed sequential, as described in [29]: inflow, fluid, outflow, and boundary particles at every time step. For simulation with the above inflow/outflow algorithm, a rearrangement process for the structure of total particles is required at every time step, resulting in a high computational cost.

2.5.2 Solid wall boundary condition

Three widely used solid boundary conditions have been used in SPH, namely, the repulsive force [38,39,19] , mirror particle [46,47] and dummy particle [48,49,50]

The first one, the repulsive force simplified Bernoulli trial (SBT) algorithm, was first proposed by Monaghan [38]. A line of ghost or virtual particles is created and located right on the solid boundary to produce a highly repulsive force to particles near the solid boundary. This algorithm prevents these fluid particles from penetrating the solid boundary. The force is calculated using a similar approach to the LJ from the molecular force in classic molecular dynamics, which can be written as follows:

$$PB_{ij} = \begin{cases} D\left[\left(\frac{r_0}{r_{ij}}\right)^{n_1} - \left(\frac{r_0}{r_{ij}}\right)^{n_2}\right]\frac{x_{ij}}{r_{ij}^2} & \frac{r_0}{r_{ij}} \leq 1 \\ 0 & \frac{r_0}{r_{ij}} > 1 \end{cases} \tag{2.15}$$

where PB_{ij} and r_{ij} are the repulsive force and distance between a solid particle and fluid particle near the solid boundary, respectively. Parameters n1 and n2 are usually taken as 12 and 4, respectively. r_0 is the cutoff distance. D is a problem-dependent parameter and should be chosen to be on the same scale as the square of the largest velocity.

Another approach was proposed by Harada [39] and developed with implicit surface in Nakata [19]. However, the repulsive force SBT algorithm cannot address the boundary deficiency problem in the support domain of fluid particles close to the boundary, which causes a large disturbance and instability to flow near the boundaries.

The second boundary, the mirror particle method was proposed by Morris [23]. The idea of this technique is to select particles close to the wall, whose kernel support has been truncated, and take a reflection on the other side of the wall. These mirror particles have information similar to that of their corresponding fluid particles, except that the velocity was computed using the equation below.

$$\boldsymbol{w} = (\boldsymbol{u}_i + \boldsymbol{u}_{if})/2 \tag{2.16}$$

where $\boldsymbol{u}_i$ and $\boldsymbol{u}_{if}$ represent the velocities for fluid particle i and its corresponding mirror particle, respectively. $\boldsymbol{w}$ is the wall velocity. However, the mirror particle method has several disadvantages, such as the distribution of reflected particles is dense and nonuniform with initial particle distribution. Furthermore, at a corner, it does not generate mirror particles or require some special treatment for this generation. Additionally, this technique requires an update at each time step.

In the dummy particle method [48,49,50], several layers of dummy particles are created along the solid wall at the initial state. These particles have zero velocity or impose a no-slip boundary condition through the entire simulation [47,48]. The number of layers depends on the radius of the kernel support to ensure full kernel support of particles located close to the wall. This approach has the drawback of using zero velocity particles and consequently problems in enforcing free-slip conditions.

2.6 Implicit surface

The implicit surfaces are mathematical models used in computer graphics to represent shapes used for geometric modeling, scientific simulation, and data visualization. For the past few decades, several approaches for implicit surface modeling techniques have been proposed. Blinn [40] and Nishimura et al. [41] introduced the blobby and meta-ball surface used for shape modeling in computer graphics where the functions are the sum of radially symmetric Gaussian function. Some primal operations related to implicit surfaces such as blending, Boolean operations, warping, and metamorphosis are presented in [42,43].For recent surface reconstruction techniques as radial basis functions [13], B-spline bases [14], and piecewise polynomial [15].

Presently, implicit surface modeling has been widely used in surface reconstruction methods, especially applications in computer graphics. Implicit surfaces have become a great way to represent objects, such as complicated obstacles, organic objects, morphing, and constructive solid geometry.

An implicit surface is mathematically defined as the zero-level set of points for an implicit function $f(x) = 0$, where $f(x)\colon \mathbb{R}^3 \rightarrow \mathbb{R}$. Interior and exterior points of such surfaces are given opposite signs.

As shown in the **Figure 2.3**, an input can be a point cloud or a polygon model. A polygonal model is composed of polyhedrons so that the intersection of the line of sight and the model of concern can be obtained easily, the same as in modeling and rendering. The beauty of using implicit surfaces, as shown in **Figure 2.3**, is that it can determine whether a point is inside or outside a surface and can represent the surface with function all over the space. For instance, in a 3D space, since the surface is defined implicitly, providing a specific coordinate can be efficiently tested to determine whether a point is inside, outside, or sprawling out the surface.

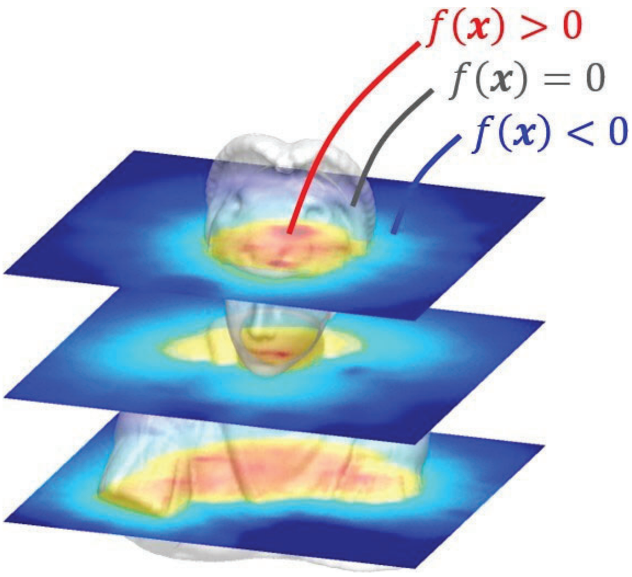


Figure 2.3 Implicit surface modeling

In contrast to traditional parametric surfaces or polygonal meshes, implicit surfaces can easily describe smooth, intricate, and articulable shapes. Moreover, points can easily be determined as interior or exterior for an implicit surface as described in G. Yngve et al.[44], which is considered challenging for parametric surfaces and polygonal meshes.

Unlike obstacles represented as particles or polygonal meshes, the nearest distance between a particle and an obstacle can easily be estimated using the Taubin distance [45]:

$$d = \frac{|f(x, y, z)|}{\|\nabla f(x, y, z)\|}, \tag{2.17}$$

and the normal vector is defined as

$$n = \frac{\nabla f(x, y, z)}{\|\nabla f(x, y, z)\|}. \tag{2.18}$$

Chapter 3 Simplified approaches in open boundary conditions

3.1 Non-reflecting outlet boundary condition

Open boundary conditions pose an interesting challenge in numerical simulations. It helps to ensure the flow moving continuity and physical. Many researches recently focus on creating the computational domain with the inflow and outflow zones where they can be defined as being upstream and downstream of the fluid domain, respectively. These researches usually eliminate the outflow particles go out the computational domain and at the same time create inflow particle at inlet. However, this technique is unsatisfied with the conservation of total mass and it requires a re-arranging process of particles in simulations with the same inflow/outflow cross-sectional lengths [29]. Periodic boundary condition can be efficiently deal with this problem. However, violations resulting from the outlet velocity field after reinsertion can cause an unstable state in long runtime simulations.

To address this issue, we propose a non-reflecting outlet boundary condition (NROBC) for particle-based fluid simulation as a combination of inflow/outflow algorithm and periodic boundary condition. The NROBC inherits the advantage of the periodic boundary condition in the sense that the number of particles is constant. Our boundary condition supports both cases, prescribed and non-prescribed, and the loss of the accuracy is small even if the quantities are non-prescribed at the inflow zone.

3.1.1 The proposed techniques

The computing domain is divided into four sets of particles as follows: fluid, wall, inflow, outflow particles as **Figure 3.1**. Assuming x- and y-axes that are along horizontal and vertical axis, respectively. In a similar way to the in/out-flow algorithm, inflow zone is placed in front of the fluid zone so that the attached zone covers a region as wide as the kernel support kh. Inflow particles move according to their velocity until they cross the inflow zone and become fluid particles. In terms of fluid particles, their information such as velocity, pressure evolves in accordance with the SPH governing equations. The fluid particles which cross the fluid zone will become outflow particles. Unlike most of inflow/outflow algorithms, outflow particles which cross outflow zone will be immediately transferred to the opposite end similarly to periodic boundary condition (see the red arrow).

Generally, most inflow/outflow algorithms eliminate outflow particles that cross the outflow region and create new inflow particles at the inflow region [25,28,29]. In the case of the different velocity profiles at the outlet zone, the number of particles inserted to the computational domain is not as equal as eliminated particles. It leads to a loss of total mass where conservation of mass is violated. Besides, the array structure of particles is distributed sequential as described in [29]: inflow particles, fluid particles, outflow particles and boundary particles at every time step. For the simulation with the above inflow/outflow algorithm, a re-arranged process for the structure of total particles is required at every time step which leads to high computational cost.

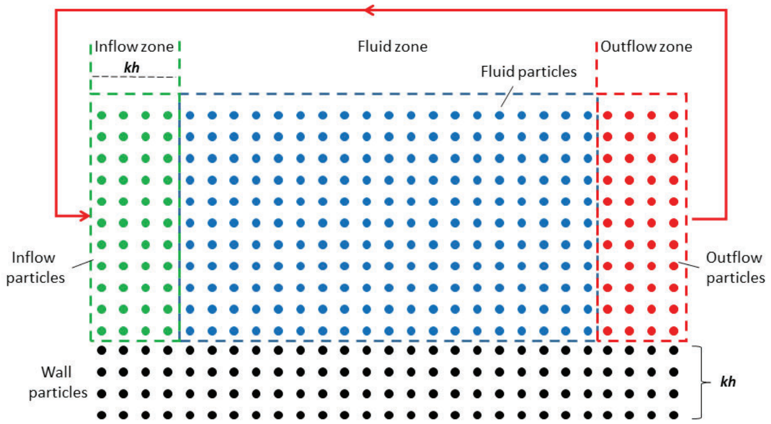


Figure 3.1 Initial sketch of the computational domain: different colours are associated to different sets of particles

The advantage of our approach is to preserve the total number of particles throughout the simulation by inserting inflow particles immediately into the computational domain after passing the outlet zone. In addition, the management of total particles are based only on the type of particles. Hence, this avoids the re-arranged process for the structure of total particles at every time step.

Due to the hybrid of in/out-flow boundaries and periodic boundary condition, this approach is flexible to solve various kinds of test cases whether prescribed boundary condition or non-prescribed boundary condition.

+ ***Boundaries with prescribed values***

Let us assume that velocity and pressure conditions at the inlet are prescribed. The velocity and pressure are enforced the desired values. The inflow particles that cross the inflow zone become fluid particles and they will be treated as fluid particles. The fluid particles pass through fluid zone become outflow particles and they will be treated as outflow particles. The outflow particles are either possible to impose specific outflow conditions as inflow particles or handling as fluid particles. Outflow particles that pass through outflow zone will be re-inserted at inlet zone with same y-coordinate positions and the values are same the desired values at the inflow zone.

+ ***Boundaries with non-prescribed values***

Let us assume that velocity and pressure conditions at the inlet are non-prescribed. The in/out-flow particles will be treated as fluid particles that evolve following the SPH governing equations. On the contrary, the values of new inflow particles are calculated as Eq. (3.1) and Eq. (3.2). (see detail in Ref [37,101])

$$\rho_i^{inlet} = \frac{\sum_{j\in fluid}^{N} \rho_j \frac{m_j}{\rho_j} W_{ij}}{\sum_{j\in fluid}^{N} \frac{m_j}{\rho_j} W_{ij}} \tag{3.1}$$

$$u_i^{inlet} = \frac{\sum_{j\in fluid}^{N} u_j \frac{m_j}{\rho_j} W_{ij}}{\sum_{j\in fluid}^{N} \frac{m_j}{\rho_j} W_{ij}} \tag{3.2}$$

Using a renormalization process from only fluid particles may help the inflow particles updating newest information from fluid particles at re-inserted positions. Thus, it ensures the stability of inflow zone.

3.1.2 Simulation test cases

Two test cases of viscous open-channel flow in the laminar regime were simulated with prescribed value and non-prescribed value. The results obtained through simulation by I. Federico [29] have been used here as a reference solution.

In this simulation, fluid flow moves with a distribution of velocity $u(x, y)$ for 2D channel flow as equation:

$$u(x, y) = \frac{\rho g s_0}{2\mu}(2Hy - y^2) \tag{3.3}$$

where ρ is density, g is the gravity acceleration, s_0 is the bottom slope, H is the surface depth, μ is the dynamic viscosity and x and y being the horizontal and vertical coordinates whose origin is located at the channel bottom.

The Reynolds number is calculated by equation:

$$Re = \frac{\rho U H}{\mu} \tag{3.4}$$

where velocity U is evaluated be the average velocity profile

$$U = \frac{1}{H}\int_0^H u(y)dy \tag{3.5}$$

The length of fluid domain is $L = 2H$ and the slope $s_0 = 0.001$. For both cases, the fluid particles are initialized with analytical solution by Eq. (3.3). The sound speed is selected equal $10u\ (y = H)$. The model of simulation depicts as **Figure 3.2.**

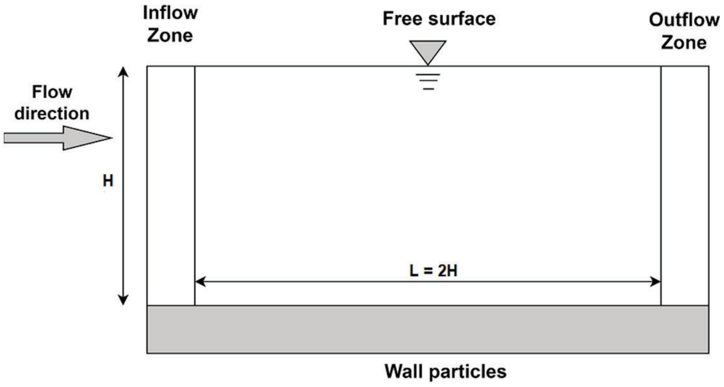


Figure 3.2 Initial Computational domain of the open-channel flow

3.1.2.1Viscous open-channel flow with prescribed boundary condition

In this case, the initial boundary condition is imposed as follows:

$$
\begin{aligned}
&y_f(t=0) = y_i(t) = H \\
&u_f(y,t=0) = u_i(y,t) = \frac{\rho g s_0}{2\mu}(2Hy - y^2) \qquad (3.6) \\
&p_f(y,t=0) = p_i(y,t) = \rho g(y-H)
\end{aligned}
$$

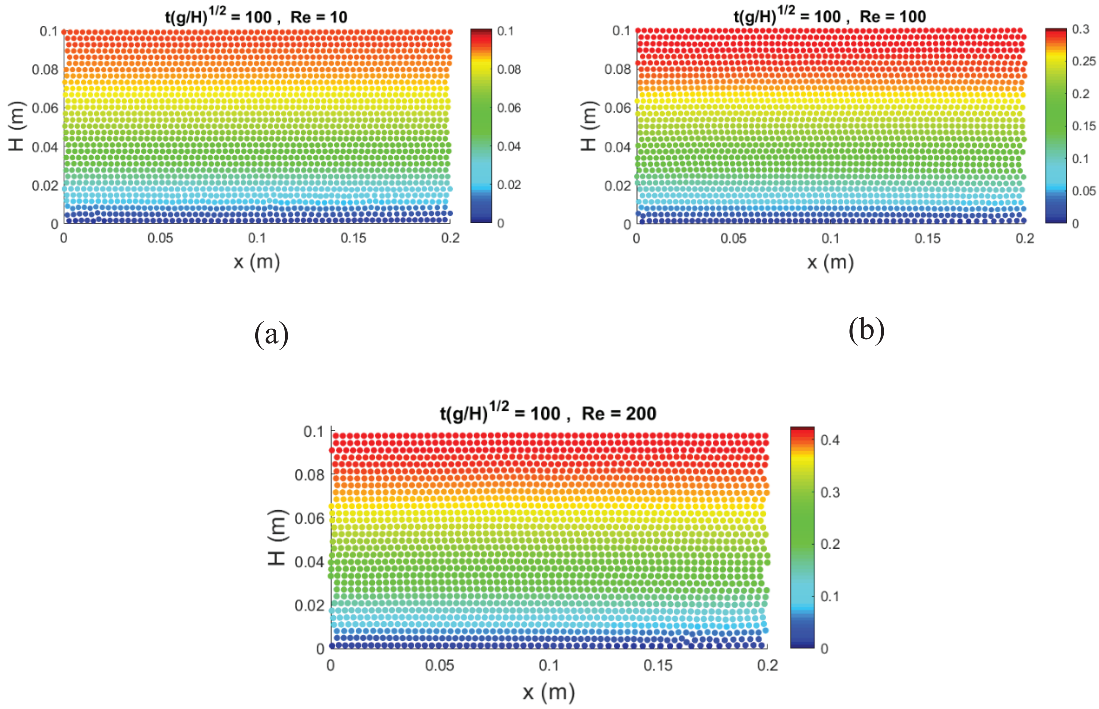
Where f and i denote the fluid and inflow particles. Velocity and pressure at the inflow zone are enforced desired values throughout simulation. On the contrary, at the outflow zone, the outflow particles are treated as fluid particles where initial information the same as the inflow particles and then their information evolves in accordance with the SPH governing equations.

At the channel bottom, we are enforced a no-slip condition through several layers of ghost particles as in Ref [48]. The objective of the simulation is to verify the velocity field of fluid particles throughout the simulation and compare with the analytical solution in Eq.(3.3).

The simulation is carried out for a long enough time in order to check the stability of the fluid flow. The particles are initially distributed with a resolution $4\Delta x = 4H/125$ with 2000 particles and applied for $Re = 10{,}100$ and 200, respectively.

Figure 3.3 illustrates particles distribution and velocity field at t(g/H)$^{1/2}$=100 for $Re = 10, 100$ and 200, respectively. It can be seen that the flow develops in almost parallel layers over the entire computational domain. The results obtained using the proposed technique are in good agreement with the inflow/outflow algorithm by Federico [29] throughout the flow domain.

In order to verify the stability of proposed technique, we carry out a comparison between the analytical solution in Eq. (3.3) and the numerical results obtained by SPH at three different x-positions, $x = 0$ (inflow threshold), $x = H$ (middle of the fluid domain), and $x = 2H$ (outflow threshold),. **Figure 3.4** shows a good agreement between analytical and numerical profiles. The performance of the proposed technique is approximately the same analytical solution at all three different x-positions. Therefore, it demonstrates the stability of this technique throughout the computational domain.



(a)

(b)

(c)

Figure 3.3 Particles distribution and velocity field at t(g/H)$^{1/2} = 100$ at $Re = 10$, 100 and 200.

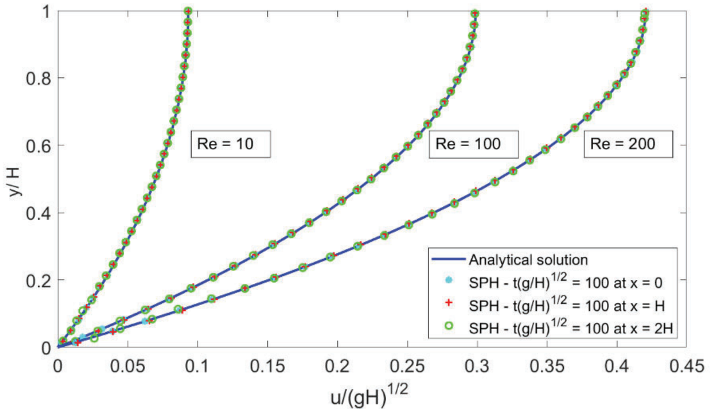


Figure 3.4 Comparisons between analytical solution and numerical results at $\mathrm{t(g/H)^{1/2}} = 100$ for $Re = 10$, 100 and 200 at $x = 0$, $x = H$ and $x = 2H$.

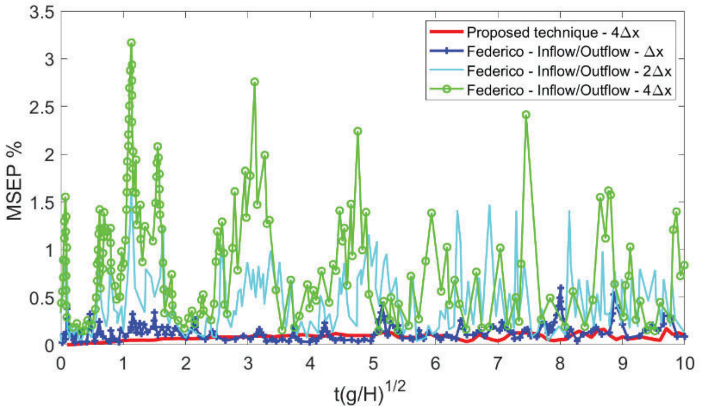


Figure 3.5 The Mean Square Error Percent (MSEP) for $Re = 10$ with the proposed technique and inflow/outflow by I. Federico [29].

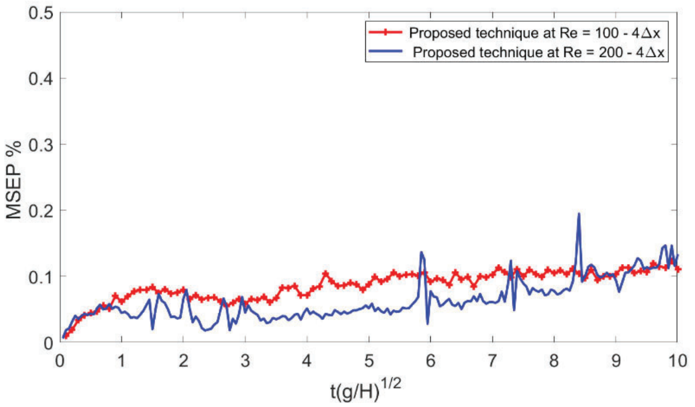


Figure 3.6 The Mean Square Error Percent (MSEP) for $Re = 100$ and $Re = 200$ with the proposed technique

To check the convergence of velocity field between the proposed technique and analytical solution, a mean square error is calculated by Eq. (3.7) at x = H (middle of the fluid domain) with a resolution $4\Delta x = 4H/125$

$$RMSEP = \frac{1}{N}\sum_{j=1}^{N}\left(\frac{u_j^a - u_j^n}{u_j^a}\right)^2 \times 100\% \tag{3.7}$$

Where, u^a and u^n are the analytical and numerical velocity, respectively. N is the number of velocity values.

Figure 3.5 illustrates the MSEP of the velocity field of the proposed technique and inflow/outflow by I. Federico [29]. As the figure, it can be seen that the MSEP of the proposed technique is low with the maximum error is about 0.1 % at the resolution $4\Delta x$. The performance of the MSEP is significantly lower to compare with inflow/outflow in I. Federico [29] at three different spatial resolutions even though the resolution Δx with 31000 particles. **Figure 3.6** also depicts the MSEP of the velocity field of proposed technique at $Re = 100$ and 200 with a peak error about 0.2 %.

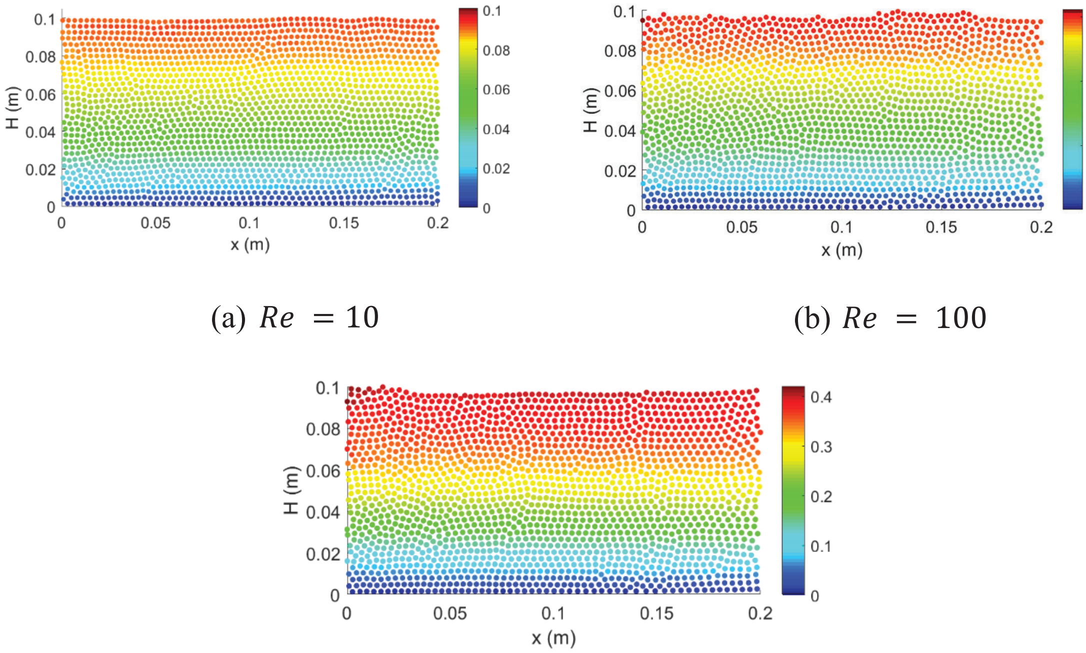
3.1.2.2 Viscous open-channel flow with non-prescribed boundary condition

In SPH, there are many simulations where the fluid flow has driven by a body force **F** in Eq. (2.7). Therefore, the fluid particles obtain the propagation and evolution of both velocity and pressure fields during the simulation. It leads to unpredictable values at the inflow/outflow region. In order to demonstrate the effectiveness and applicability of the proposed technique. In this case, we assume that the velocity and pressure of the inflow zone are non-prescribed. In/out-flow particles are treated the same as fluid particles. The initial state is imposed as follows:

$$\begin{aligned}
&y_f(t=0) = y_i(t=0) = y_o(t=0) = H \\
&u_f(y,t=0) = u_i(y,t=0) = u_o(y,t=0) = \frac{\rho g s_0}{2\mu}(2Hy - y^2) \\
&p_f(y,t=0) = p_i(y,t=0) = p_o(y,t=0) = \rho g(y - H)
\end{aligned} \tag{3.8}$$

Where f , i and o denote the fluid ,inflow and outflow particles .The outflow particles pass through outflow zone will be vice versa inflow zone and their information is calculated as Eq. (3.1) and Eq. (3.2).

The ghost particles are also enforced a non-slip condition. The simulation is carried out as same as an above test case with a resolution $4\Delta x = 4H/125$ for $Re = 10$, 100 and 200, respectively.



(a) $Re = 10$

(b) $Re = 100$

(c) $Re = 200$

Figure 3.7 Particles distribution and velocity field at $t(g/H)^{1/2} = 100$ at $Re = 10$, 100 and 200.

Figure 3.7 illustrates particles distribution and velocity field at $t(g/H)^{1/2} = 100$ for $Re = 10$, 100 and 200, respectively. As the figure, a fluctuation at the channel top appears due to its free surface state. Overall, the results obtained using the proposed technique are in agreement with inflow/outflow algorithm by Federico [29] throughout the flow domain.

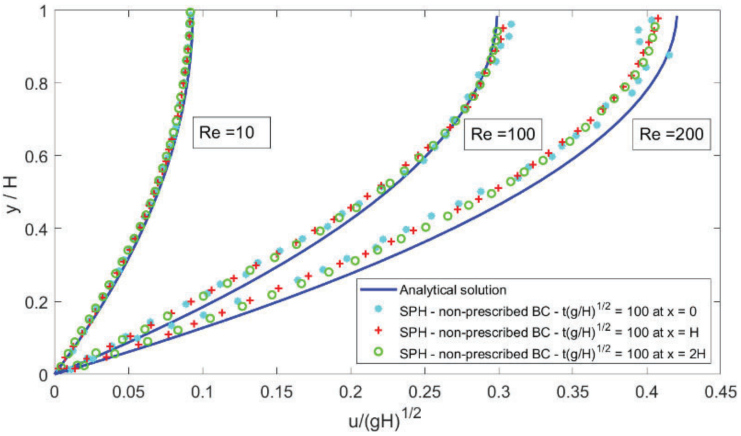


Figure 3.8 Comparisons between analytical solution and numerical results at $\mathrm{t(g/H)}^{1/2} = 100$ for $Re = 10$, 100 and 200 at at $x = 0,\ x = H$, and $x = 2H$.

Figure 3.8 shows a comparison between analytical and numerical profiles. The performance at $Re = 10$ is approximately at three different x-positions with the highest error around 0.6 % as shown in **Figure 3.9** The result of the MSEP obtained significantly lower to compare with inflow/outflow in I. Federico [29] at the same spatial resolution. At the higher Reynold number $Re = 100$ and $Re = 200$, the numerical results have a slightly different with a difference of about 1.2% and 2 % as shown in **Figure 3.10**, respectively

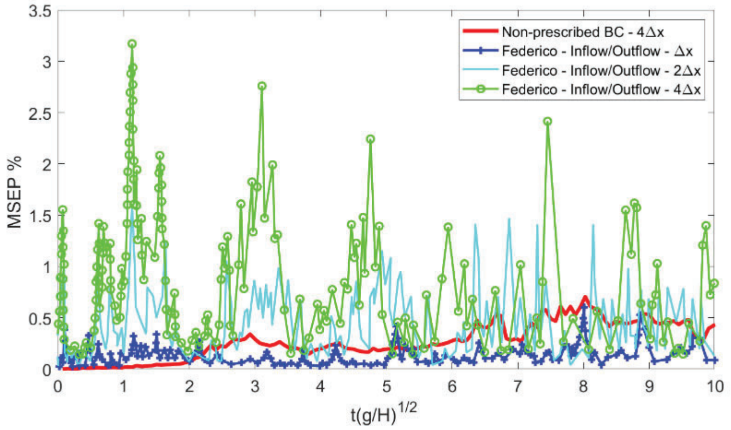


Figure 3.9 The Mean Square Error Percent (MSEP) for $Re = 10$ with the proposed technique and inflow/outflow by I. Federico [29]

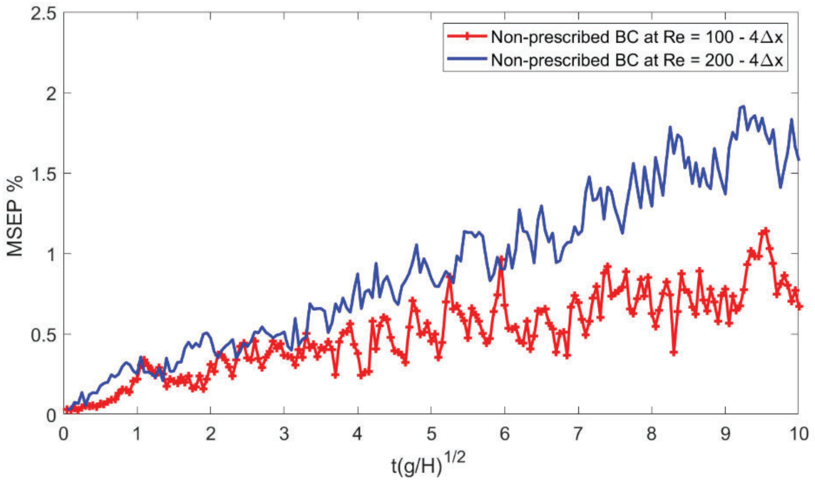


Figure 3.10 The Mean Square Error Percent (MSEP) for $Re = 100$ and $Re = 200$ with the proposed technique

3.2 A simplified approach of open boundary conditions

The NROBC is effective in simulation with the same sizes of the inlet and outlet. To widely apply for various simulation in different fields, we introduce a simplified approach for open boundary conditions, which is without an extrapolation process for calculating information on inflow/outflow particles.

3.2.1 The proposed techniques

In general, particle-based simulation with open boundary conditions requires four sets of particles that correspond to fluid, wall, inflow and outflow particles as shown in **Figure 3.11**. In the methods by Tafuni et al. [28] and Negi et al. [37], the particle motions at inflow/outflow zones are computed using quantities like pressure and velocity of the particles. The particle quantities should be estimated using some kind of extrapolation techniques which requires complicated process like normalized weighted sum of spatial distribution [37] or using ghost nodes for inlet/outlet particles combining with a higher-order interpolation scheme [28]. In addition, such extrapolation is required for both of inflow/outflow cases.

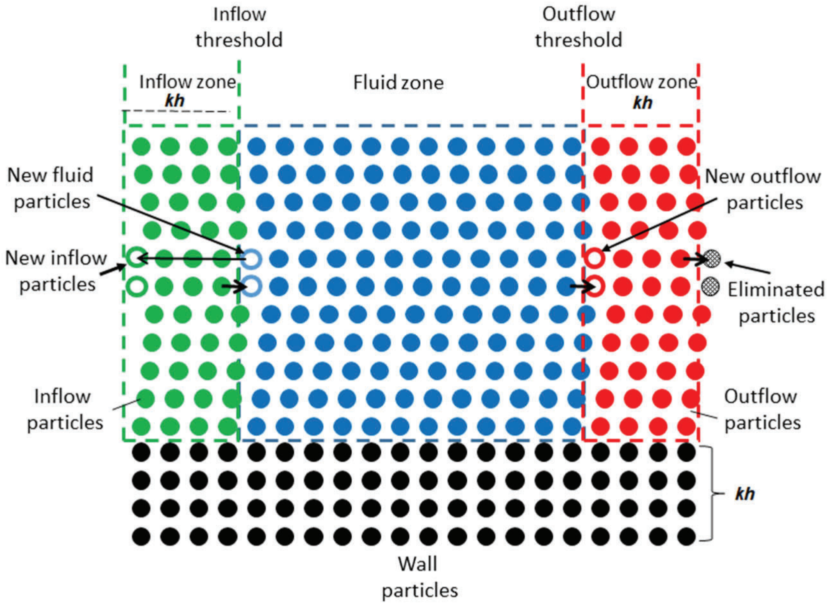


Figure 3.11 Sketch of the simplified approach of open boundary conditions, where kh is the kernel support

In order to avoid the extrapolation process and perform the computation with a simple procedure, we modify the algorithm so that the SPH can be applied not only at the fluid zone but also at inflow/outflow zones. Similar to other inflow/outflow algorithms, we assume that the inflow zone is placed in front of the fluid region so that the attached zone covers a region as wide as the radius of the kernel support kh, where h is the smoothing length and k is a parameter varying with the selected kernel function. In our algorithm, the particle motion at the inflow zone is determined as follows.

- The inflow particles move according to their velocity until they cross the border between the inflow and fluid zones.
- The inflow particles that cross the border turn into fluid particles and behave as fluid particles.
- At the same time, the particles that cross the border produce new inflow particles periodically at the front end of the inflow zone and we copy all the information of the particle to the new inflow particle.

Assuming x- and y-axes that are perpendicular and parallel to the border, respectively, the positions of the new inflow particles, $(x^{\text{new}}, y^{\text{new}})$, are determined by

$(x^{\mathrm{new}}, y^{\mathrm{new}}) = (x^{\mathrm{inflow}} - kh, y^{inflow})$. In this approach, all the quantities of inflow particles can be determined in the same way as the particles at fluid zone except for the generated particles at the end of the inflow zone.

In terms of the particles at the outflow zone, we apply the SPH algorithm in the same way as the fluid zone and we simply eliminate particles that go out of the outflow zone as shown in **Figure 3.11**. Although the total number of particles is not constant because the generation of new inflow particles is not synchronized with the elimination of outflow particles, managing generated/eliminated particles is still simple because the total number of particles does not change so much as far as the domain is fixed. Furthermore, this approach is free from extrapolation process because the quantities of outflow particles is obtained by the SPH algorithm which contributes to simple and consistent computation of fluid particles with open boundary conditions.

In some cases of the extrapolation-based open boundary conditions, prescribed particle quantities given a priori are applied to some of the inflow particles during the entire simulation [28,37]. The prescribed values can also be applied in our method with a small modification. In the case where prescribed values are applied to some areas of inflow zone, we simply use the prescribed values instead of copying the quantities of the particles that cross the inflow threshold or applying the SPH algorithm.

3.2.2 Simulation test cases

To demonstrate the effectiveness and applicability of the proposed technique, in this section, we perform several numerical test cases of flow over a backward-facing step with prescribed and non-prescribed inflow boundary conditions.

3.2.2.1 Backward-facing step with Prescribed Inflow Boundary Condition

A 2D backward-facing step problem was simulated at $Re = 389$ and compared to the experimental results of Negi [37] and Armaly et al. [62]. Following Negi [37], the inflow particles were set to a constant velocity for the entire computational time. However, no-slip boundary conditions were enforced at the walls, causing non-physical pressure fluctuations. To address this issue, a small part of the initial wall next to the inlet zone can include an imposed slip, as in Ref. [37], or a parabolic profile of the velocity can be used at the inlet zone, as in Refs. [30,63]. These approaches can cause several inconveniences during test cases. Taking advantage of the proposed technique, we only imposed a prescribed velocity value at the initial time step and then the velocity automatically adjusted to be well suited to the near-by boundary conditions. The velocity in the inflow zone was less than the maximum velocity and ensured a Mach number of the flow of $Ma < 0.1$.

The step height was set to $H = 4.9$ mm with an inlet width of $H1 = 5.2$ mm. The density was $\rho = 1.0\ \mathrm{kg/m^3}$, the dynamic viscosity was $\mu = 0.017\ \mathrm{kg/ms}$, and the length of the domain was $L = 130.6$ mm. The initial inlet velocity was 0.67 m/s with a maximum of 1 m/s. We compared the velocity profiles at four different marked positions P1–P4 of x/H = 2.55, 3.57, 4.80, and 7.14 (where x is the distance downstream of the step). A schematic of the simulation model is shown in **Figure 3.12**.

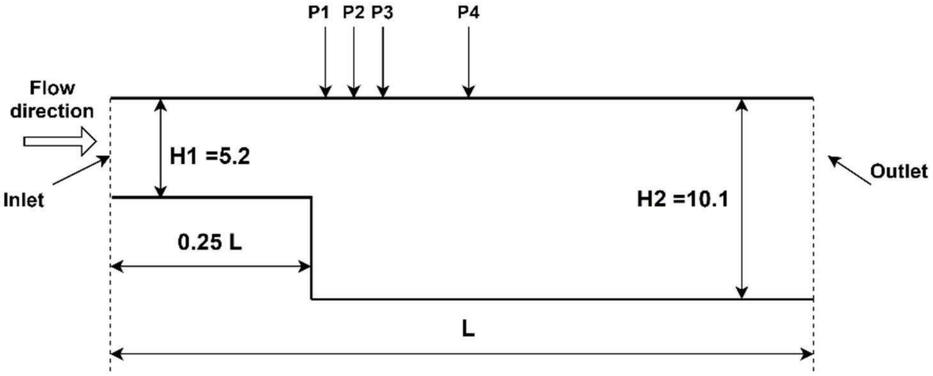


Figure 3.12 Sketch of the domain used for the backward-facing step simulations

Figure 3.13 illustrates the axial velocity of the simulation at $Re = 389$. The maximum axial velocity is 1.0 m/s, and the minimum value is -0.11 m/s. The PST helps remove particle clumping near the step. Good performance for the axial velocity is shown, where the velocity at the inlet automatically adjusts corresponding to the wall boundary with the velocity profile shown in **Figure 3.14a**. The velocity tends to low near the wall and to reach maximum values at the center of the domain. This is likely equivalent to using a special treatment at the inlet, as in Ref. [37].

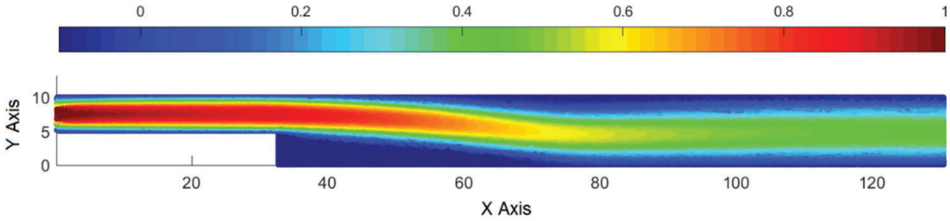
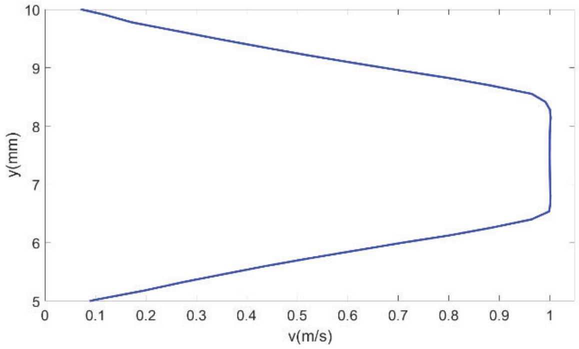
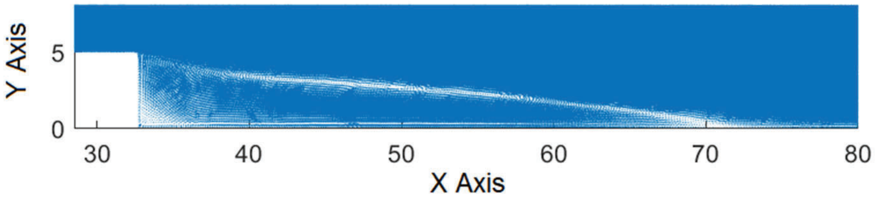


Figure 3.13 Axial velocity for Re = 389

The reattachment length was determined to be $x/H = 7.84$, as shown in **Figure 3.14b**. This result is close to the result obtained by Negi [37] of 7.9 and the experiment value [62] of 7.94.



(a)



(b)

Figure 3.14 (a) Velocity profile at the inlet and (b) reattachment length for Re = 389

The velocities at the four different marked positions, P1–P4, were considered over the channel height and compared to the profiles of the reference results from Refs. [37], see

Figure 3.15. The results obtained are in good agreement with the reference solution at each marked position.

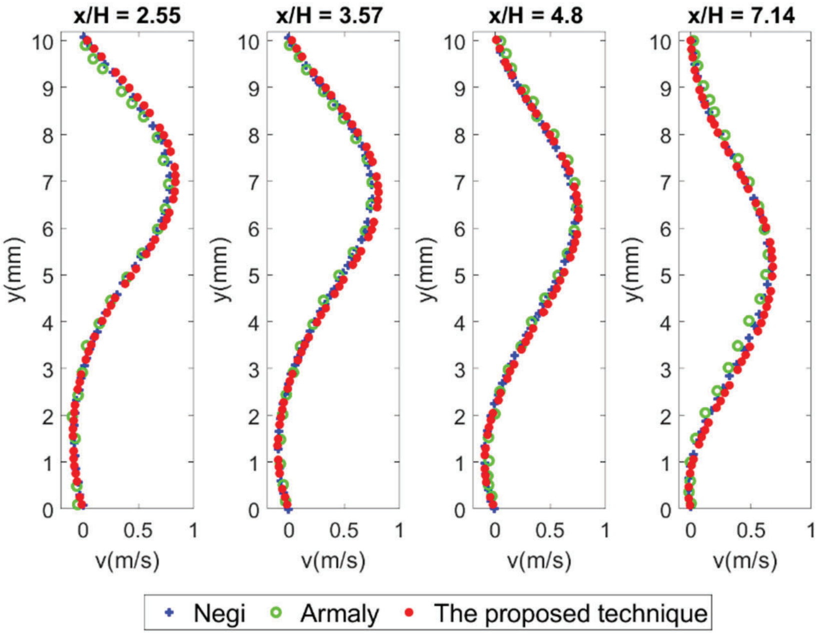


Figure 3.15 Comparison of velocities at four different locations for Re $= 389$.

3.2.2.2 Backward-facing step with non-prescribed inflow boundary condition

In this case, a 2D backward-facing step problem was simulated at $Re = 100$ and compared to the experimental results of Adami [48] and Issa et al. [64]. In these studies, another step was added at the end of the channel with the same length to apply periodic conditions. Using the proposed technique, we can easily simulate this case without a second step.

In this simulation, the density was $\rho = 1.0$ kg/m^3, the viscosity was $\nu = 1.456 \times 10^{-2}$, and the length of the domain was $L = 122.4$ m. We applied a constant body force of $F = 3.0 \times 10^{-4}$ m/s and a sound speed of $c = 2.1$. The step height was set to $H = 4.9$ m with an inlet width of $H1 = 5.2$ m. No-slip boundary conditions were enforced on the walls [48]. A schematic of the simulation model is shown in **Figure 3.16**. We compared the velocity profiles at four different marked positions, P1–P4, as shown in **Figure 3.16**.

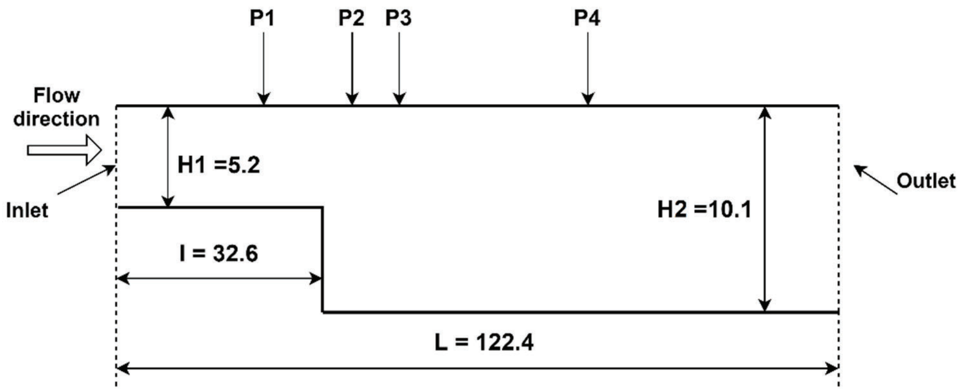


Figure 3.16 Sketch of the domain used for the backward-facing step simulations.

The fluid flow was driven by a constant body force F. The fluid particles propagated and evolved during the simulation until reaching a corresponding stable state at $Re = 100$. This led to unknown values in the inflow region. The non-prescribed inflow test case demonstrates the effectiveness and applicability of the proposed technique.

Figure 3.17 illustrates the axial velocity of the simulation at $Re = 100$. It can be seen that the flow field develops smoothly over the entire computational domain. The fluid particles are colored with the maximum axial velocity being 0.21 m/s and the minimum value being -0.015 m/s.

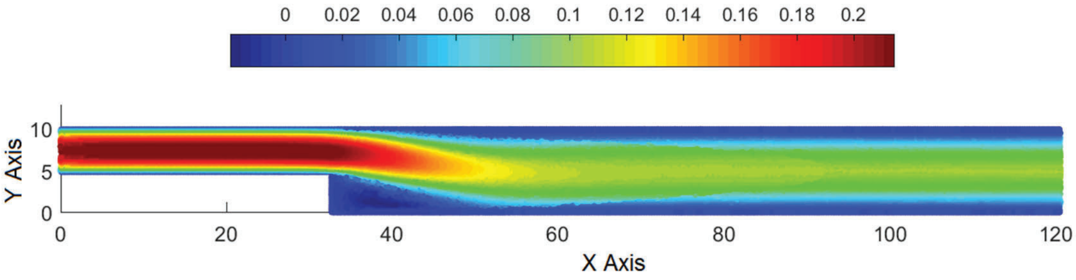


Figure 3.17 Axial velocity for Re = 100.

Assessing the quality of simulation, the axial velocities at four different marked positions, P1–P4, were considered over the channel height and compared to the reference profiles from Refs. [48,64], as shown in **Figure 3.18**. The results obtained at each marked position are in fairly good agreement with the reference solution.

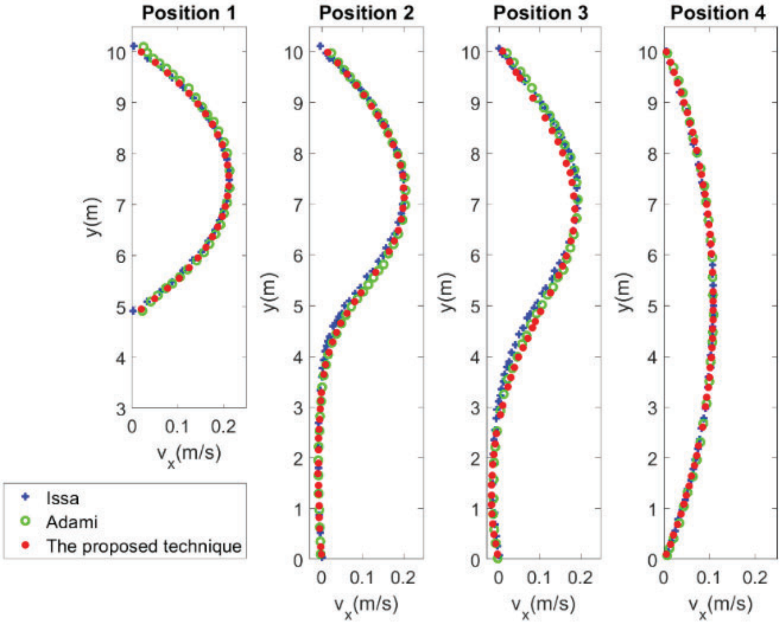


Figure 3.18 Comparison of the axial velocities at four different marked positions, P1–P4, for $\mathrm{Re} = 100$.

Chapter 4 Improved boundary treatment in solid wall boundary conditions

4.1 Boundary treatment techniques using implicit surface

Presently, applying a simulation with an implicit surface is quite new and promising. Generating boundary particles with implicitly defined walls is the novelty of this proposed technique. To remedy the deficiency in the support domain of fluid particles close to the wall, we propose a boundary treatment (BT) technique for generating wall particles appropriately to comply with the shape. Here, several uniform layers of wall particles are generated.

4.1.1 The proposed techniques

In this technique, we propose schemes for constructing wall particles as follows: Firstly, we initialize a computation domain and keep particles as presented in **Figure 4.1**. We assume that blue particles, black circle particles and green circle particles sequentially are fluid particles, fluid particles close the wall and grid points particles, respectively. A black curve defines as an implicit curve $f(x, y) = 0$. Wall particles are generated only using black circle particles or green circle particles whose distance to the wall is smaller than 3Δx. We divide these particles into three groups by their distances, i.e., the distance of the first, second, and third groups are less than Δx , 2Δx, and 3Δx, respectively. Here Δx is an initial spacing of fluid particles.

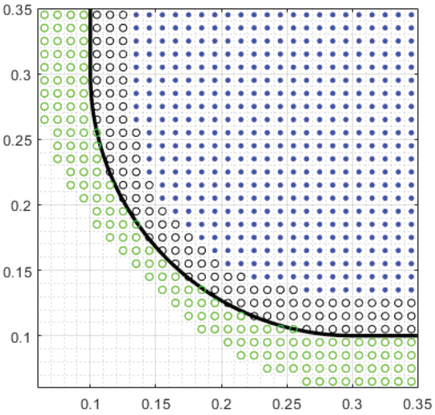


Figure 4.1 The distribution of fluid particles(blue), fluid particles close the wall (black circle) and grid points particles (green circle)

To generate the first layer, the first group of these particles are copied and moved onto the boundary. The wall particles are formed by the movement of these particles following the normal vector to the implicit surface with the approximate distances to the surface.

$$\boldsymbol{r}_{wall} = \boldsymbol{r}_{\substack{selected\\particles}} - d \cdot \boldsymbol{n}_{\substack{seclected\\particles}} \tag{4.1}$$

where $\boldsymbol{r}_{wall}$ is the position of the wall particles after being copied and moved from $\boldsymbol{r}_{\substack{selected\\particles}}$, d is the distance of the fluid particles from the wall as calculated in Eq. (2.17) and $\boldsymbol{n}_{\substack{selected\\particles}}$ is its normal vector obtained by Eq. (2.18). After generating the wall particles on the first layer, the subsequent layers are similarly formed using the particles in the second, and third groups. To generate uniformly distributed boundary particles, the distance between neighbouring layers of the wall particles should be equal to the initial spacing of the fluid particles. Therefore, the positions of the boundary particles in subsequent layers are generated as follows:

$$\boldsymbol{r}_{wall} = \boldsymbol{r}_{\substack{selected\\particles}} - (d + (i_{layer} - 1) \cdot \Delta x) \cdot \boldsymbol{n}_{\substack{selected\\particles}} \tag{4.2}$$

where i_{layer} is the index of each layers. After they have been moved, the wall particles for the implicit surfaces are generated as indicated with red dots shown in **Figure 4.2**

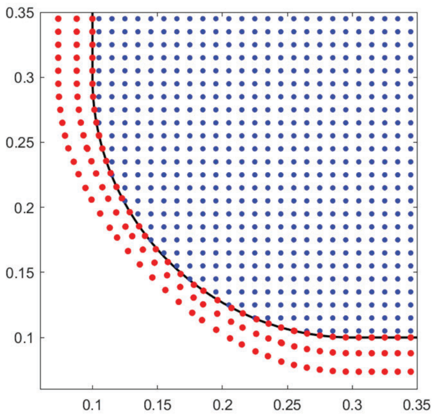


Figure 4.2 Generation of wall particles using proposed technique

4.1.2 Simulation test cases

4.1.2.1 Flow past a circular at Re = 20 and 200

The test case is a flow past a circular cylinder where the circular cylinder is represented by the function.

$$f(x, y) = x^2 + y^2 - r^2 \tag{4.3}$$

where r is the radius of the circular cylinder. The implicit surface of a circular cylinder is defined as the zero-level set of the function: $f(x, y) = 0$. The boundary conditions were implemented as follows.

(1) No-slip boundary conditions were enforced on the body surfaces, and the information concerning these solid particles was extrapolated from fluid particles following the method in Ref. [48].

(2) For the upper and lower walls, symmetry boundary conditions for the velocity were imposed as described in Ref. [47], i.e., $v_y = 0$ and $\partial v_x / \partial y = 0$.

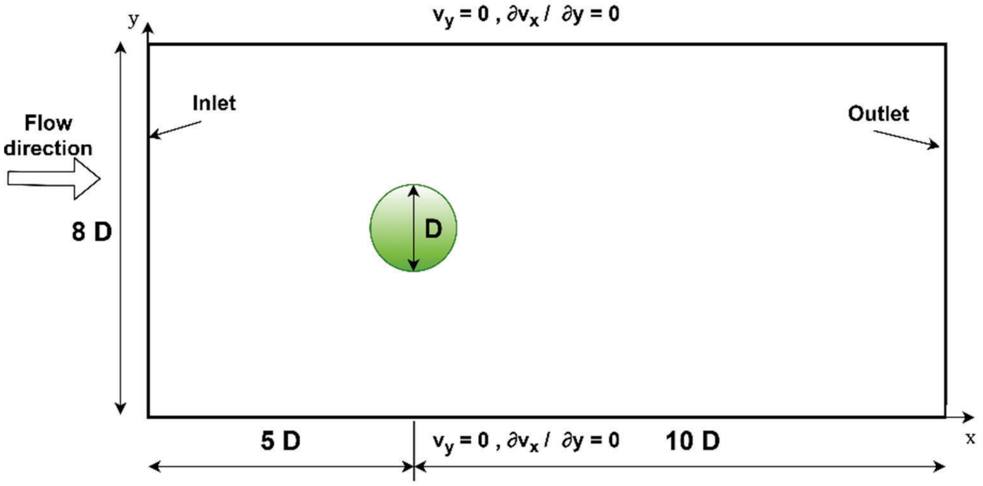

y
v_y = 0 , ∂v_x / ∂y = 0
Inlet
Outlet
Flow direction
8 D
D
5 D
v_y = 0 , ∂v_x / ∂y = 0
10 D
x

Figure 4.3 Computational domain for 2-D flow past a circular cylinder

Figure 4.3 depicts the initial setup of the simulation according to Lastiwka [25]. The computational domain is $[0\ ,15D] \times [0\ ,8D]$ with a cylinder of diameter $D = 0.1$ m and a cylinder is located at the position $[5D\ ,4D]$. A constant density of $\rho_\infty = 1000$ kg m^{-3}, a constant x - velocity of $U = 1$ m s^{-1}, and a null y - velocity are initialized for the fluid particles.

Furthermore, fluid phase is discretized with initial particles placed at a resolution of $D/\Delta x = 40$. The Quintic kernel function used with the smoothing length is set equal to $h = 1.5\,\Delta x$. The Reynolds number of the flow is $Re = UL/\nu$, where ν is the kinematic viscosity in m^{2}s^{-1}. The viscosity varies to simulate flow past at $Re = 20$ and 200

The validation for this case was performed by comparing the time histories of the drag and lift coefficients using the proposed technique in A.Tafuni [28] in **Figure 4.4**. A good agreement is obtained between the two solvers. The magnitude of the drag force increases to the peak value and then decreases to a steady state during the time interval $tU/D \in [0\,,10]$ at $Re = 20$ and $tU/D \in [0\,,40]$ at $Re = 200$, respectively. The drag and lift coefficients converge to a steady value of $C_D = 2.4$, $C_L$is null lift coefficient for $Re = 20$, and $C_D = 1.43$, $C_L = \pm 0.75$ for $Re = 200$. Also, these results are related with those obtained by other authors in Marrone et al.[65], Vacondio et al. [66] , Braza et al.[67].

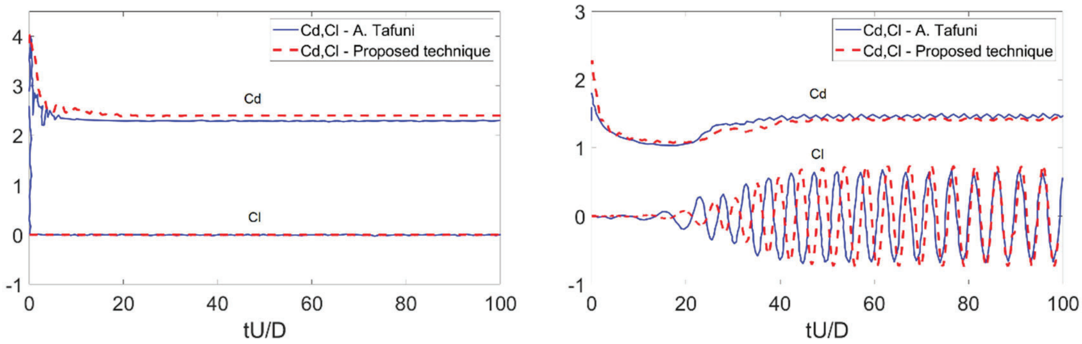


Figure 4.4 Time history of drag and lift coefficients for flow past a circular cylinder Re = 20 and 200

Figure 4.5 and **Figure 4.6** illustrate the velocity and pressure contours at $Re = 20$ and 200. Two dimensionless of velocity magnitude $U^* = u(x/D, y/D)U^{-1}$ and $P^* = 2P(x/D, y/D)\rho^{-1}U^{-2}$ are used.. The velocity and pressure fields are smooth through the whole computational domain. The tensile instability which causes the generation of numerical cavitation is eliminated by particle shifting technique. **Figure 4.5** shows a steady-state with a steady smooth wake behind the cylinder. Whereas, **Figure 4.6** depicts unsteady von Karman street with oscillatory wake appeared behind the cylinder at $Re = 200$.

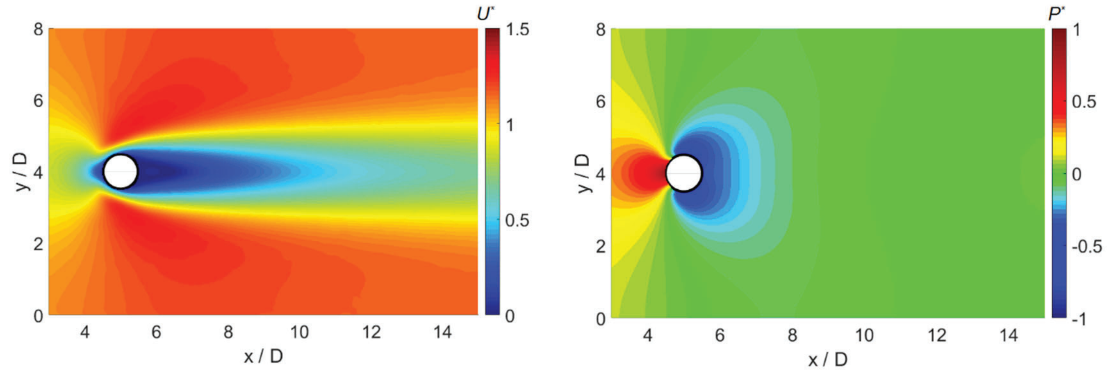


Figure 4.5 2-D flow past a cylinder: velocity field and pressure field at Re = 20

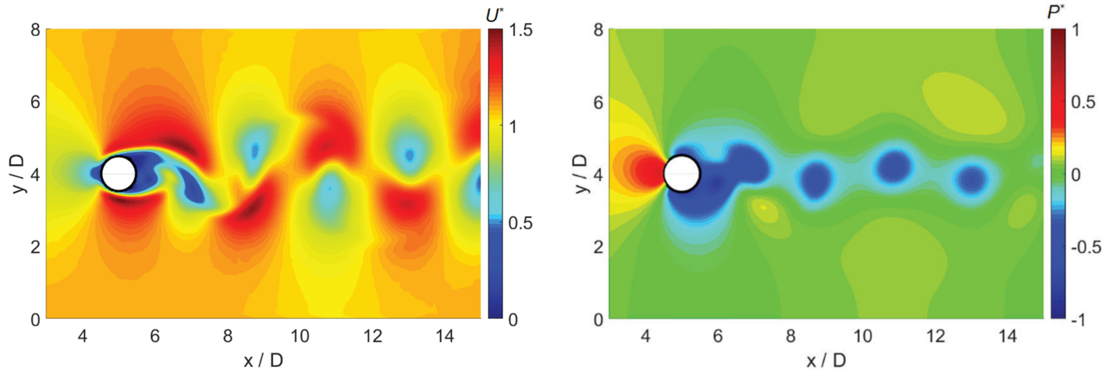


Figure 4.6 2-D flow past a cylinder: velocity field and pressure field at Re = 200

Figure 4.7 shows the velocity vector field at $Re = 200$. It can be seen that the velocity vector field at $[0\,,3.5D]$ on the horizontal axis developed with parallel layers and exhibits stability without numerical noise. This result was obtained as a result of the upstream stability, where the proposed scheme was applied.

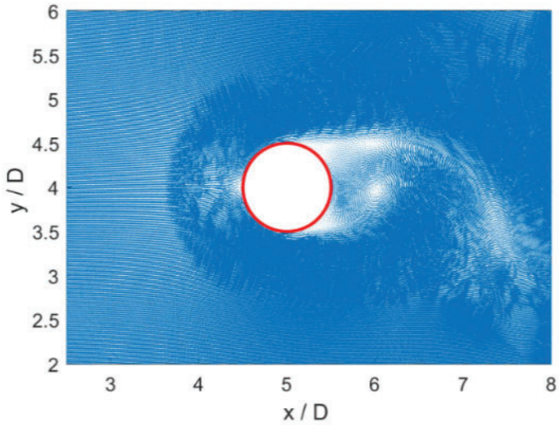


Figure 4.7 2D flow past a cylinder: the velocity vector field at Re $= 200$.

4.1.2.2 Flow past a circular cylinder without a sector at Re = 100

In the second case, the flow past a more complex implicit surface is simulated. The circular cylinder section without a sector as **Figure 4.8** can be performed by the function in Eq. (4.4).

$$f(x,y) = \begin{cases} d(x,y)\ \big((y > -ax \text{ or } y < 0) \text{ and } x^2 + y^2 < 1\big) \\ -d(x,y)\ (\text{otherwise}) \end{cases} \tag{4.4}$$

Where:

$$d(x,y) = \begin{cases} \min\big(d_1(x,y)\,,d_2(x,y)\big)\ (\,y < -ax \text{ and } y > 0\,) \\ \min\big(d_1(x,y)\,,d_2(x,y), d_3(x,y)\big)\ (\text{otherwise}) \end{cases}$$

$$d_1(x,y) = \begin{cases} \|(\mathrm{x},\mathrm{y})\|\ (\,y < \frac{x}{a}) \\ \left|(ax+y)/\sqrt{(1+a^2)}\right|\ \left(\frac{x}{a} \le y < \frac{x-x_1}{a} + y_1\right) \\ \|(\mathrm{x},\mathrm{y}) - (x_1,y_1)\|\ (\,y \ge \frac{x-x_1}{a} + y_1) \end{cases}$$

$$x_1 = -\frac{1}{\sqrt{(1+a^2)}}\ ,y_1 = -\frac{a}{\sqrt{(1+a^2)}}\ and\ \|(x,y)\| = \sqrt{x^2+y^2} \tag{4.5}$$

$$d_2(x,y) = \begin{cases} \|(x+1,y)\|\ (\,x < -1\,) \\ |y|\ (-1 \le x < 0) \\ \|(x,y)\|\ (\,x \ge 0) \end{cases}$$

$$d_3(x,y) = |1 - \|(x,y)\||$$

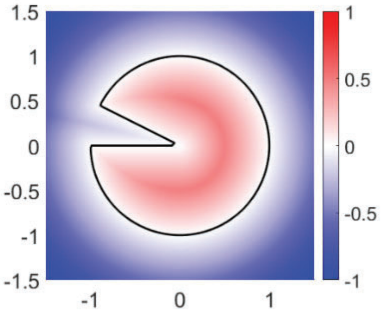


Figure 4.8 A circular cylinder without a sector

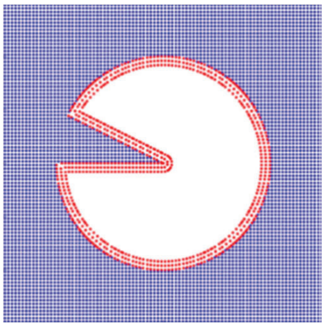


Figure 4.9 Generation of wall particles.

Figure 4.9 shows the distribution of wall particles well-suited along the implicit curve. It is quite challenging to treat wall particles at a sharp edge position. The BT technique is successfully applied and distributed in compliance with the curvature of objects, therefore taking into consideration the effect of bounsdary for the simulation.

The computational domain is $[0\,,15D] \times [0\,,8D]$ with the body located at the position $[5D\,,4D]$, where diameter $D = 2$m. The spatial resolution is $D/\Delta x = 40$. The fluid flow past the obstacle at $Re = 100$.

The results for this case were compared to the results in P.N.Sun [53]. **Figure 4.10** depicts the time histories of the drag and lift force coefficients that are compared with the FVM solver in [53]. Thus, a good agreement is observed between the two solvers. The values obtained in the proposed technique is slightly higher than the FVM.

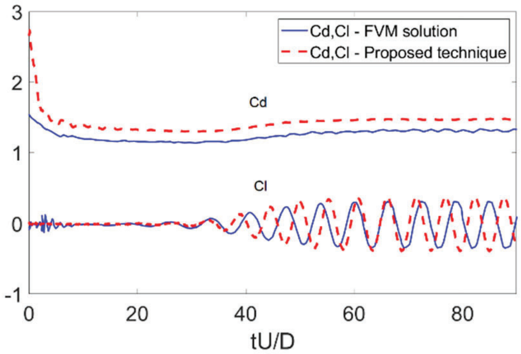


Figure 4.10 Time history of drag and lift coefficients for flow past the body at Re = 100

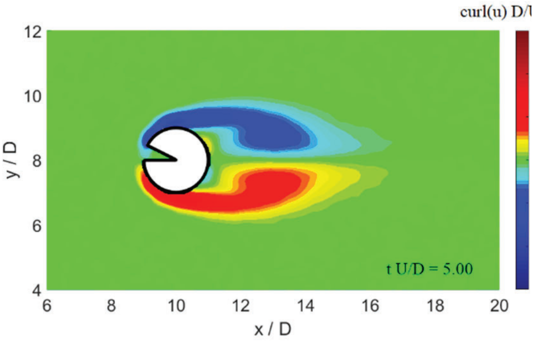


Figure 4.11 Vorticity field at $tU/D = 5$

Figure 4.11 shows the vorticity field around the body at $tU/D = 5$. The result is quite smooth and symmetric with no effect by the un-physical cavitation that appears at sharp corner positions.

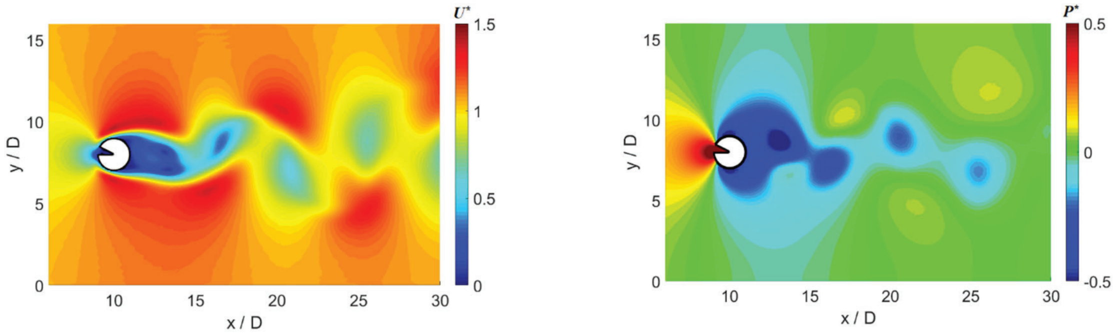


Figure 4.12 2-D flow past the body: velocity field and pressure field at Re $= 100$

Figure 4.12 illustrates the velocity and pressure contours at $Re = 100$. The velocity and pressure fields are smooth and no spurious oscillation through the whole computational domain. The unsteady von Karman street appeared behind the cylinder at $Re = 100$.

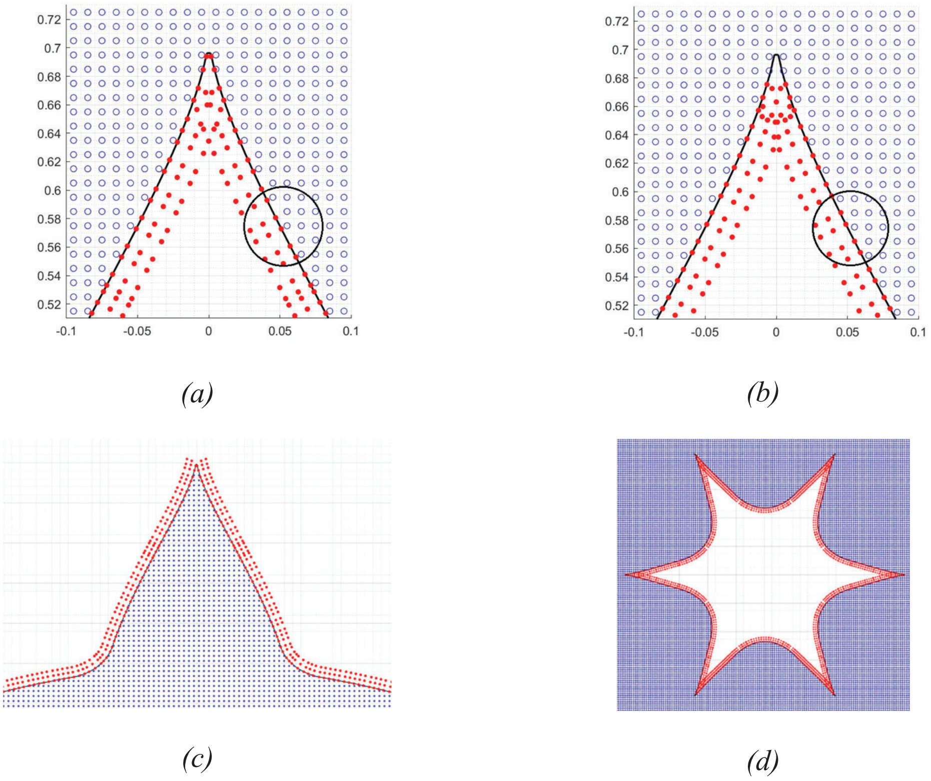
4.2 Improved boundary treatment technique for implicit surfaces

The boundary treatment (BT) technique in section 4.1 successfully handled for implicit surface with plenty of shapes. However, these schemes are limited to complex geometries, especially at sharp corners. In order to remedy this issue, we propose an improved boundary treatment (IBT) technique which can consider as a better technique to compare with the first approach. The techniques provide complete treatment for complicated geometries even though at sharp corners without any additional treatment. It is a highlight to compare with several other previous works.

4.2.1 The proposed techniques

In this technique, we propose two different schemes for reconstructing wall particles as follows; (1) Improved wall particles from fluid particles (IWFF), (2) Improved wall particles from ghost particles (IWFG), respectively. The former one, IWFF, is suitable for fluid simulation of external flow, in which case internal wall particles are required. This approach is also suitable in fluid simulation deforming obstacles because the wall particles can be generated on time depending on the locations of fluid particles. On the other hand, the latter one, IWFG, is suitable for fluid simulation of internal flow, in which case external wall particles are required. In this approach, the wall particles can be computed as pre-processing for static obstacles.

Figure 4.13 depicts the wall particles generated by applying our techniques to implicit curves. **Figure 4.13(a)** and **Figure 4.13(b)** show the result of wall particles generated along sharp edges using IWFF and IWFG, respectively, where blue circles are fluid particles and red dots are wall particles. In these figures, the particles are distributed along layers so that the distance of neighbouring layers are uniform. In addition, the sharp corner is covered with the wall particles appropriately.



(a)

(b)

(c)

(d)

Figure 4.13 Improved boundary treatment technique with implicit surface

Figure 4.13(c) illustrates an exterior wall boundary with well-distributed along a complicated curve even at the sharp corner position. **Figure 4.13(d)** depicts the robustness of the IBT technique for handling complex shape models.

First, let I, W, and O represent the regions inside, on, and outside the implicit curves, as shown in **Figure 4.14(a)**. We focus on a portion of the shape, namely the red square region in **Figure 4.14(b)**, to illustrate the proposed technique.

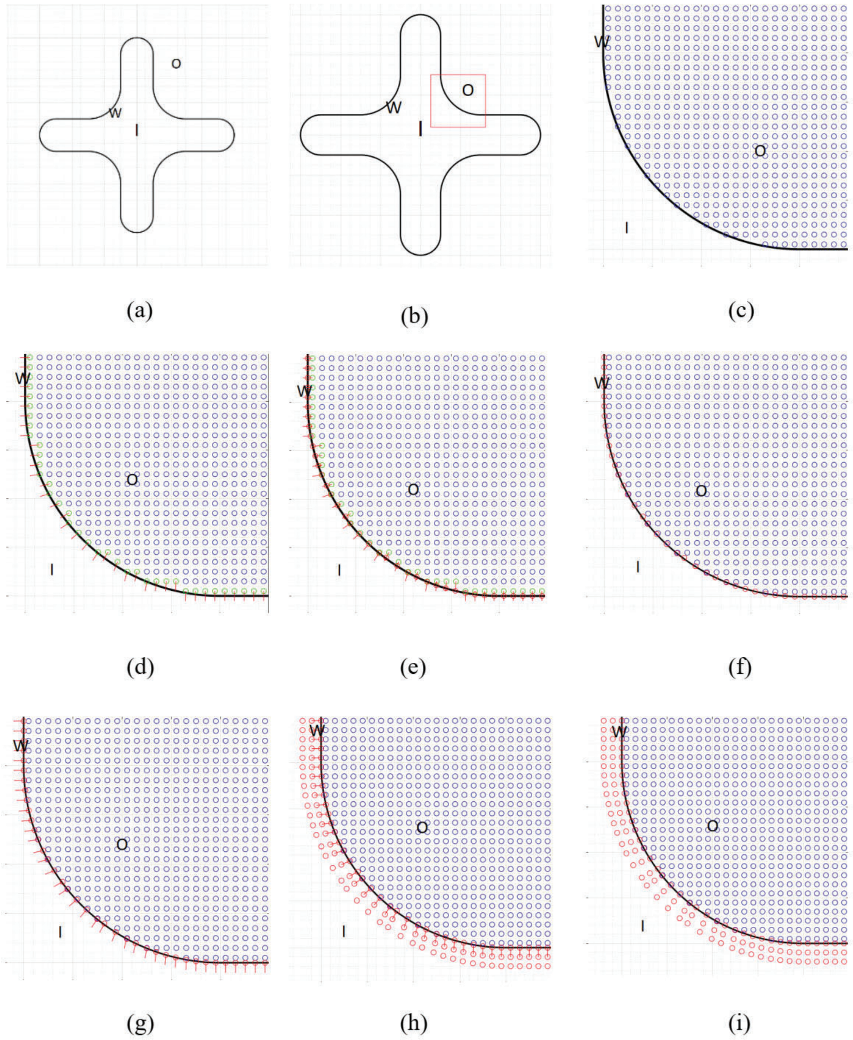
4.2.1.1 Improved wall particles from fluid particles

In this scheme, wall particles are generated using fluid particles whose distance to the boundary are smaller than the support radius of the kernel.

We assume that the outside region is filled with fluid particles as shown in **Figure 4.14(c)**. The green fluid particles near the boundary in **Figure 4.14(d)** are separated by a distance that is less than the smoothing length h and are copied and moved right onto the boundary. The wall particles are formed by the movement of the fluid particles following the normal vector to the implicit surface with the approximate distances to the surface so that the selected fluid particles are projected onto the boundary as shown in **Figure 4.14(e)**.

$$\boldsymbol{r}_{wall} = \boldsymbol{r}_{fluid} - d \cdot \boldsymbol{n}_{fluid} \tag{4.6}$$

where $\boldsymbol{r}_{wall}$ is the position of the wall particles after being copied and moved from $\boldsymbol{r}_{fluid}$ and d is the distance of the fluid particles from the wall as calculated in Eq. (2.17) and $\boldsymbol{n}_{fluid}$ is its normal vector are calculated as Eq. (2.18). After they have been moved, the boundary particles on the first layer are located as indicated in **Figure 4.14(f)**.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 4.14 Process of constructing ghost particles using IWFF: (a)–(i) sequential steps in the process

After generating the ghost particles on the first layer, the subsequent layer is formed by duplicating the first layer to the region **I** following the normal vector as in **Figure 4.14(g)**. To generate uniformly distributed boundary particles, the distance between each ghost particle layer should be equal to the initial spacing of the fluid particles. Therefore, the positions of the boundary particles in subsequent layers are generated as **Figure 4.14(h)** following as:

$$\boldsymbol{r}_{wall} = \boldsymbol{r}_{ghost} - i_{layer} \cdot \Delta x \cdot \boldsymbol{n}_{ghost} \tag{4.7}$$

Finally, the wall particles for the implicit surfaces are generated as in **Figure 4.14(i).**

4.2.1.2 Improved boundary particles from ghost particles

As opposed to the generation of the boundary particles from the fluid particles, in this technique, wall particles are generated from ghost particles that are near the wall.

The green ghost particles created near the boundary in **Figure 4.15(c)** are separated by a distance that is approximate the support radius of the kernel. The black circles in **Figure 4.15(d)** illustrate the selected ghost particles. These ghost particles are moved toward the boundary along the normal vector by Eq. (2.17) with the distance equal to the approximated distance obtained by Eq. (2.18) as shown in **Figure 4.15(e)**. The wall particles generated along the boundary are illustrated in **Figure 4.15(f)**.

$$\boldsymbol{r}_{wall} = \boldsymbol{r}_{ghost} - d \cdot \boldsymbol{n}_{ghost} \tag{4.8}$$

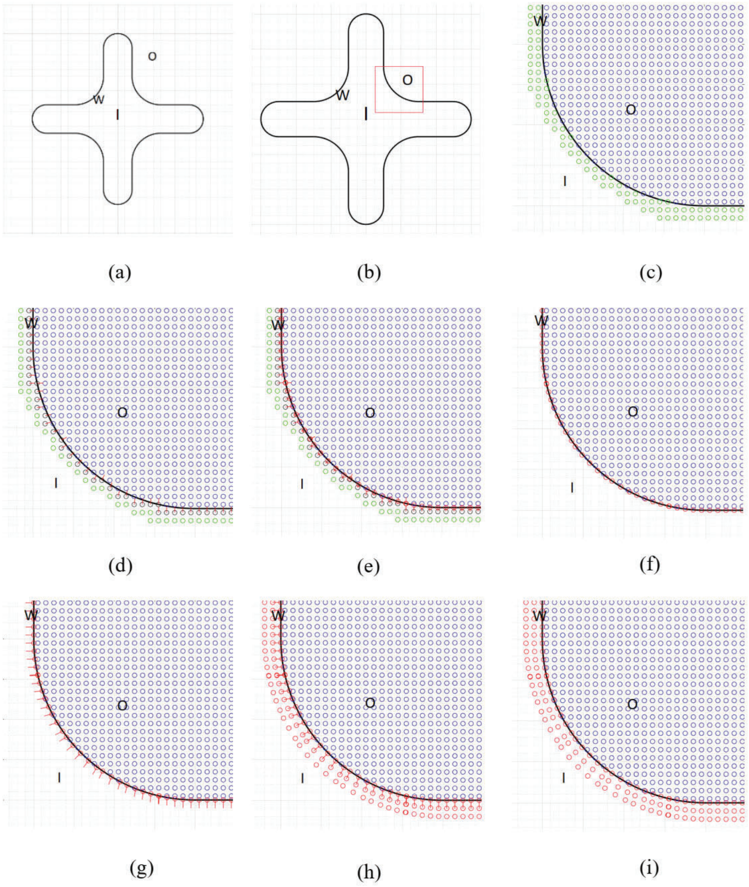


Figure 4.15 Process of constructing ghost particles using IWFG: (a)–(i) sequential steps in the process.

After generating the ghost particles on the first layer, the subsequent layer is formed by copying the first layer to the region **I** following its normal vector, as shown in **Figure 4.15(g)**. Therefore, the positions of the boundary particles in the subsequent layer are generated as **Figure 4.15(h)** following as:

$$\boldsymbol{r}_{wall} = \boldsymbol{r}_{ghost} - i_{layer} \cdot \Delta x \cdot \boldsymbol{n}_{ghost} \tag{4.9}$$

Finally, the wall particles for implicit surfaces are generated as in **Figure 4.15(i). Figure 4.16** shows the final result of particle generation along the implicit curve for external flow.

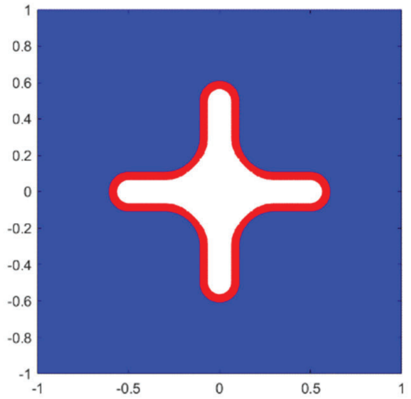


Figure 4.16 Wall particles generated by IWFG along implicit curve

4.2.2 Simulation test cases

In this section, the IBT technique is compared to the BT technique in section 4.1 and other numerical results to demonstrate the effectiveness of the proposed technique. The no-slip boundary condition is implemented for the solid boundary and the information of solid particles is only interpolated from fluid particles as shown in [48].

4.2.2.1 Flow over a cylinder

a, Flow over a cylinder Re = 1

The test case is a periodic flow past a cylinder where the cylinder is represented by the implicit function.

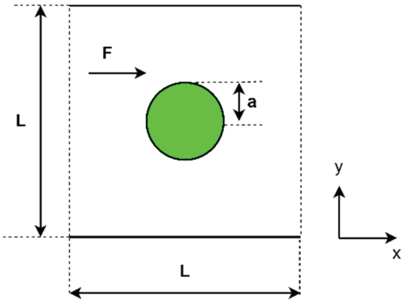


Figure 4.17 Cylinder model within a periodic lattice.

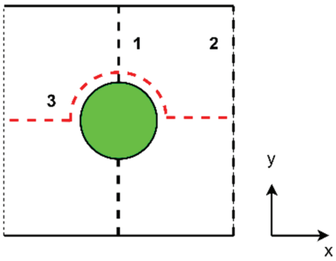


Figure 4.18 Paths for comparing the SPH and FEM solutions

Following Morris [23], the initial setup of the simulation is illustrated in **Figure 4.17**. A cylinder with a radius of a = 0.02 m is centered in a domain with L = 0.1 m, a kinematic viscosity of v = 10^{-6} m^2 s^{-1}, a constant body force of F = 1.5 × 10^{-7} m s^{-2}, and a sound speed of c = 5.77 × 10^{-4} m s^{-1}.

The fluid phase is discretized with particles initially placed at a resolution of Δx = 0.002 m with 2184 fluid particles and approximately 200 wall particles for IBT and BT technique. The simulation was initialized with zero velocity, and the velocity scale was taken to be V_0 = 5 × 10^{-5} m s^{-1} to give a Reynolds number of Re = 1.

The validation for this case was performed by comparing the velocity and pressure distributions along the paths indicated in **Figure 4.18** with a steady incompressible viscous flow using the finite element method (FEM) [23].

Figure 4.19 illustrates the velocity magnitude around the cylinder and the contour plots of the velocity magnitude for the IBT and BT technique using SPH for Re = 1. The results obtained using SPH are in good agreement with those of FEM [23] throughout the flow domain. The performances of the IBT and BT technique are approximately the same over the entire computational domain.

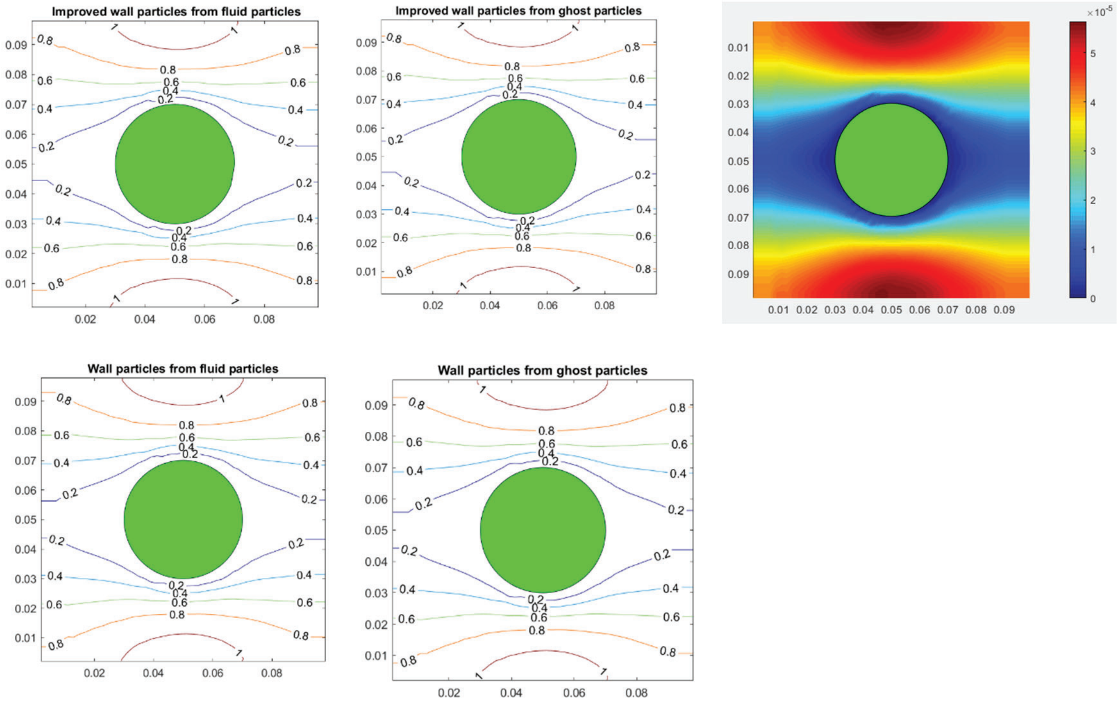
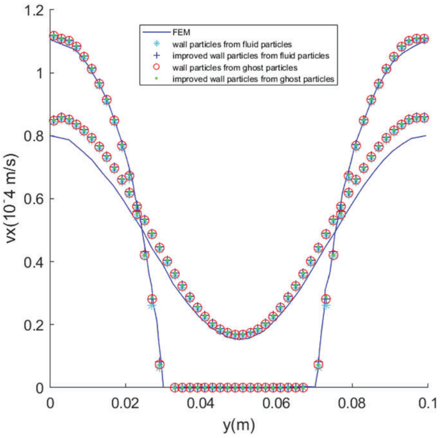


Figure 4.19 Velocity magnitude around a cylinder and contour plots of the velocity magnitude for Re =1.

A comparison of the velocity profiles along paths 1 and 2 appears in **Figure 4.20**. The velocity profiles are in fairly good agreement along path 1, while SPH has slightly higher velocities than FEM along path 2 in the far field.



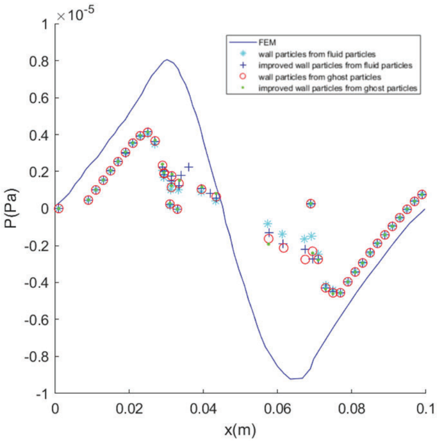


Figure 4.20 Comparison of the SPH and FEM velocity profiles along paths 1 and 2 for Re = 1.

Figure 4.21 Comparison of the SPH and FEM pressure profiles along path 3 for Re = 1.

Figure 4.21 depicts the pressure profiles along path 3 for SPH and FEM. It can be seen that the pressure distribution in SPH is better than that in FEM with smaller fluctuations in the pressure. The peak pressures obtained in SPH were less than 40% of those obtained in FEM, as shown in **Table 4.1**. **Table 4.1** also shows that the difference in the pressures for the IBT and BT technique is nearly insignificant.

Table 4.1 Comparison of the SPH and FEM pressures for Re = 1.

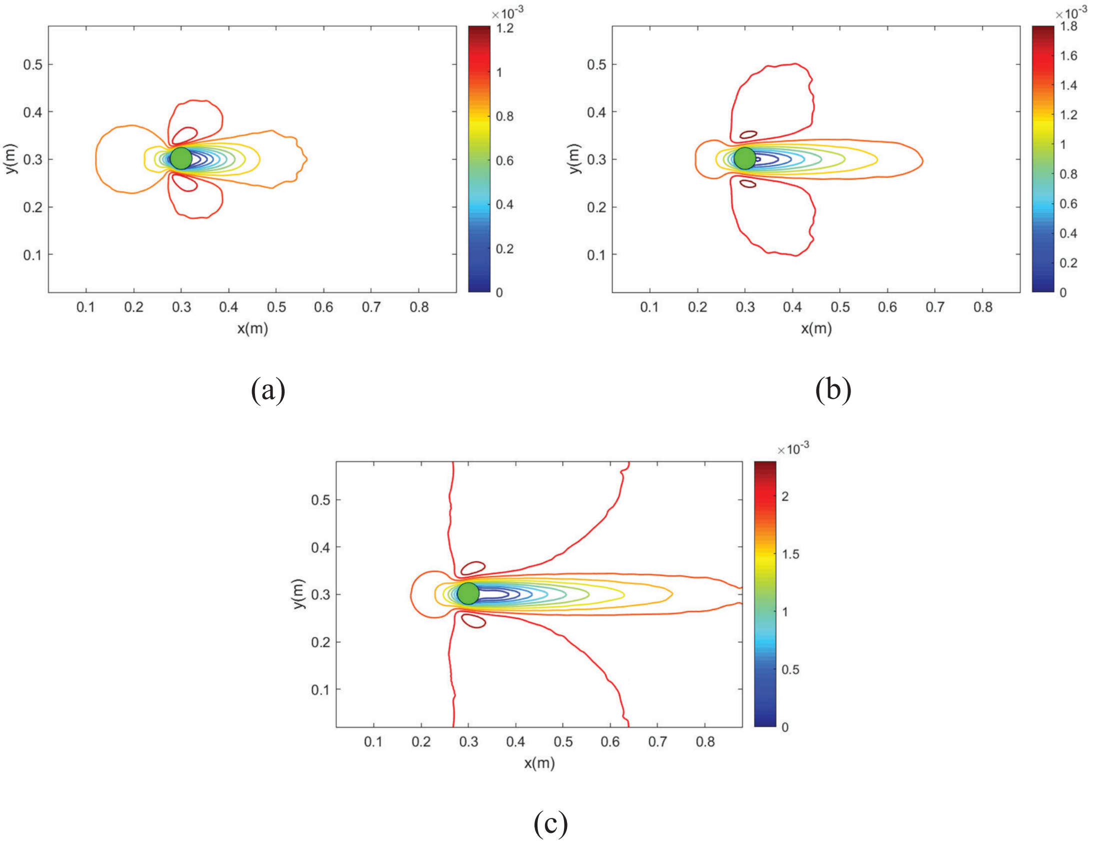
	Maximum Pressure(Pa)	Minimum Pressure (Pa)
FEM	8.0477×10^{-6}	-9.2191×10^{-6}
Improved wall particle from fluid particles	4.1819×10^{-6}	-6.1881×10^{-6}
Improved wall particle from ghost particles	4.2166×10^{-6}	-5.6038×10^{-6}
Wall particle from fluid particles	4.0953×10^{-6}	-5.8941×10^{-6}
Wall particle from ghost particles	4.1947×10^{-6}	-5.6129×10^{-6}

b, Flow Past a Circular Cylinder Re = 20,30 and 40

The flow past a cylinder with a Reynolds number from 20 to 40 is simulated using the multiple boundary tangent (MBT) technique in the incompressible smoothed particle hydrodynamic method studied by Yildiz [47]. In this case, the computational domain with a cylindrical obstacle is located at Cartesian coordinates of (L/3, H/2), where the length is L = 0.9 m and the height is H = 0.6 m. The simulation parameters were diameter = 0.04 m, density = 1000 kg m^{-3}, viscosity = 10^{-3} kg m^{-1} s^{-1}, and a body force of $F = 3 \times 10^{-6}$ N kg^{-1}.

The fluid phase was discretized with particles placed initially in a 200 × 134 array in a rectangular domain. The total number of fluid particles was 19,726, and approximately 150 wall particles were used for the IBT and BT technique. The simulation was initialized with zero velocity, and the adaptive time stepping satisfied the Courant–Friedrichs–Lewy condition.

Figure 4.22 illustrates the velocity magnitude contours of the IBT techniques for Re = 20, 30, and 40. The result of the IBT technique is in good agreement with that of the MBT technique [47] throughout the flow domain.



(a)

(b)

(c)

Figure 4.22 The velocity magnitude contours of the IBT for (a) Re =20, (b) Re = 30, and (c) Re = 40.

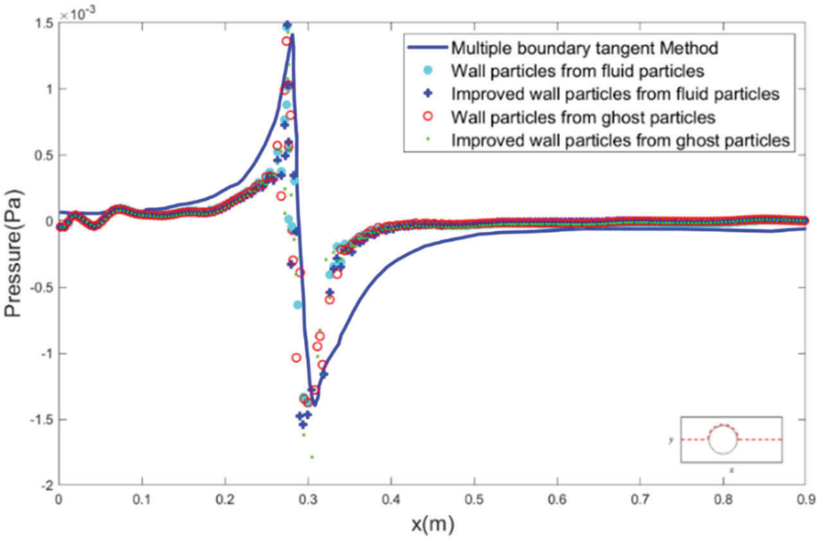


Figure 4.23 Pressure curves along the path indicated in the inset for Re = 20.

Figure 4.23 shows the pressure curves along the path indicated in the inset for the IBT, BT, and MBT techniques. It can clearly be seen that the pressure distributions in the IBT and BT techniques have smaller oscillations than the MBT technique over the entire computational domain. In **Table 4.2**, the maximum pressure values for the two IBT techniques are listed as 0.00125580 Pa and 0.00148058 Pa. These pressures are lower than the maximum pressure in the MBT technique and approximately the same as those found with the BT techniques. Meanwhile, the minimum pressure values of the two IBT techniques are higher than those of the MBT and BT techniques.

Table 4.2 Comparison of the SPH and MBT pressures for Re = 20.

	Maximum pressure (Pa)	Minimum pressure (Pa)
MBT Technique	0.0015657	−0.0014772
Improved wall particle from fluid particles	0.00148058	−0.0016236
Improved wall particle from ghost particles	0.00125580	−0.00174866
Wall particle from fluid particles	0.00146979	−0.00153648
Wall particle from ghost particles	0.00136027	−0.00137676

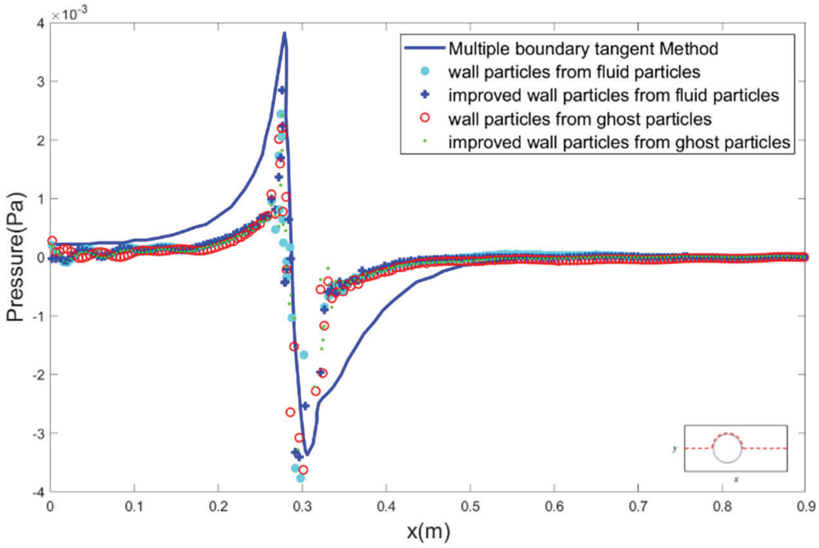


Figure 4.24 Pressure curves along the path indicated in the inset for Re = 30.

Figure 4.24 depicts the pressure curves along the path indicated in the inset for the IBT, BT, and MBT techniques. In **Figure 4.24**, the pressure distributions in the IBT and BT techniques have smaller oscillations than the MBT technique over the entire computational domain. **Table 4.3** shows that the maximum and minimum pressure values in the two IBT techniques are smaller than the maximum pressure values in the MBT and BT techniques.

Table 4.3 Comparison of the pressure values of SPH and the MBT technique for Re = 30.

	Maximum Pressure (Pa)	Minimum Pressure (Pa)
MBT Technique	0.00391918	−0.00354169
Improved wall particle from fluid particles	0.002850860	−0.003464650
Improved wall particle from ghost particles	0.002419953	−0.003299559
Wall particle from fluid particles	0.002440592	−0.003766334
Wall particle from ghost particles	0.002692417	−0.003725582

Figure 4.25 illustrates the pressure curves along the indicated path for the IBT and BT techniques. It can be seen that the difference in the pressure from the highest point to the lowest point in the IBT technique is lower than that in the BT technique. Therefore, the pressure distribution in the IBT technique has a lower fluctuation than that in the BT technique over the entire computational domain.

Table 4.4 shows that the maximum and minimum pressure values for the IBT and BT techniques are lower than those for the MBT technique.

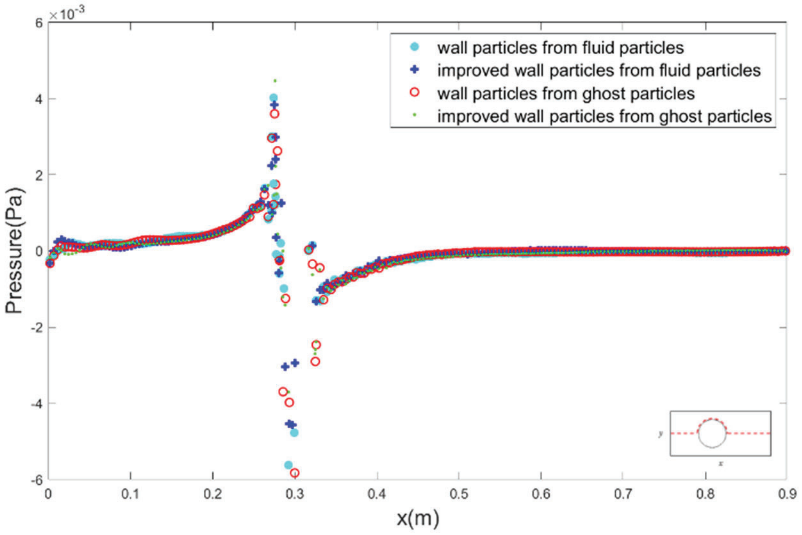


Figure 4.25 Pressure curves along the path indicated in the inset for Re = 40.

Table 4.4 Comparison of the pressure values of SPH and the MBT technique for Re = 40.

	Maximum Pressure (Pa)	Minimum Pressure (Pa)
MBT Technique	0.007052	−0.006162
Improved wall particle from fluid particles	0.003837	−0.004573
Improved wall particle from ghost particles	0.004467	−0.003701
Wall particle from fluid particles	0.004017	−0.005614
Wall particle from ghost particles	0.003601	−0.005822

In simulation for a cylinder shape, the IBT and BT techniques are obtained well-performed results. The difference is negligible with low Reynold number. It seems that IBT will be better in simulation with higher Reynold number.

4.2.2.2 Flow over a complicated shape

a, Flow over a complicated shape with very low Re

This test case is a periodic flow past a complicated shape represented by the implicit surface:

$$f(x,\ y) = \frac{(\max\{1 - 5f_1(x,y), 0\ \})^2 + (\max\{1 - 5f_2(x,y), 0\})^2 - 1}{10} \tag{4.10}$$

Where:

$$f_1(x,\ y) = x^2 + 3 \cdot \sqrt[3]{y^2} - 1 \tag{4.11}$$

$$f_2(x,\ y) = 2.5 \cdot \sqrt[3]{x^2} + 2y^2 - 1 \tag{4.12}$$

The initial setup of the simulation is illustrated in **Figure 4.26.** The implicit function with d = 2 m is centered in a domain with L = 3.0 m, a viscosity of v = 5.0 × 10^{-2} m^2 s^{-1}, a constant body force of F = 5 × 10^{-9} m s^{-2}, and a sound speed of c = 6.0 × 10^{-4} m s^{-1}. The fluid phase was discretized with particles placed initially at a resolution of Δx = 0.02 m with 20,764 fluid particles and approximately 500 wall particles for the IBT and BT technique. The simulation was initialized with zero velocity, and the velocity scale was taken to be V_0 = 2.5 × 10^{-5} m s^{-1} to give a Reynolds number of Re = 1.

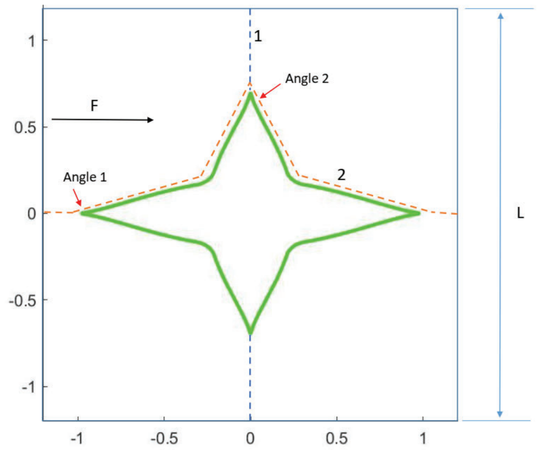


Figure 4.26 Computational domain for the flow past an implicit surface.

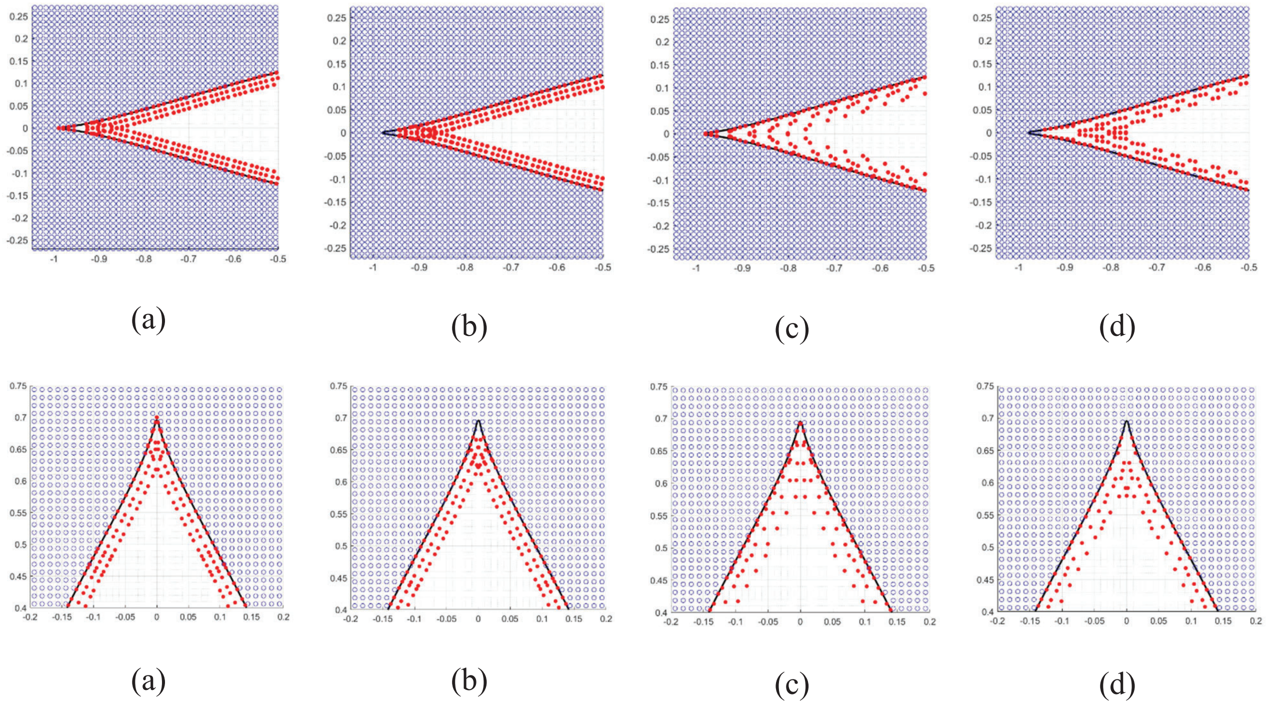


Figure 4.27 Boundary treatment for the implicit surface at angles 1 and 2 (Fig. 4.26) using the IBT and BT techniques as follows: (a) improved wall particles from fluid particles, (b) improved wall particles from ghost particles, (c) wall particles from fluid particles, and (d) wall particles from ghost particles..

Figure 4.27 shows the boundary treatment for the implicit surface at angles 1 and 2. As the figure shows, at angles 1 and 2, the wall particles from the ghost particle technique include a small gap in the red particles close to the edges. This leads to a lack of particles in the support domain of the fluid particles near the wall. Conversely, the IBT technique displays a good distribution of boundary particles and the blank space close to the edges is filled in. Therefore, the pressure obtained in the area near the sharp angle is more stable.

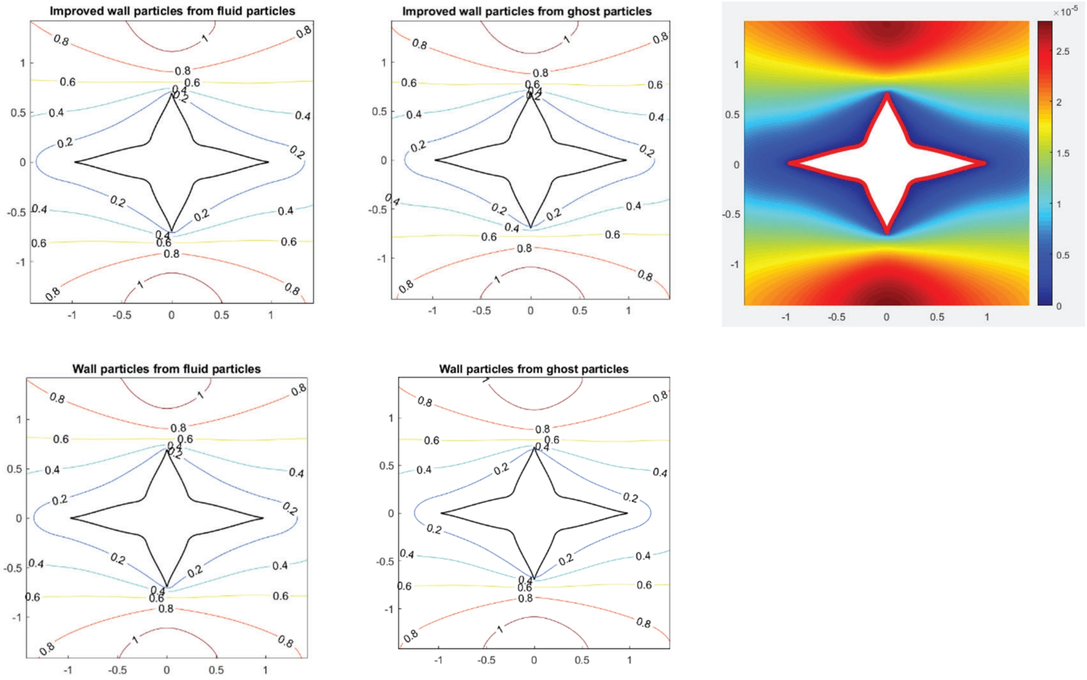


Figure 4.28 Velocity magnitude around an implicit surface and contour plots of the velocity magnitude using the IBT and BT techniques for Re = 1.

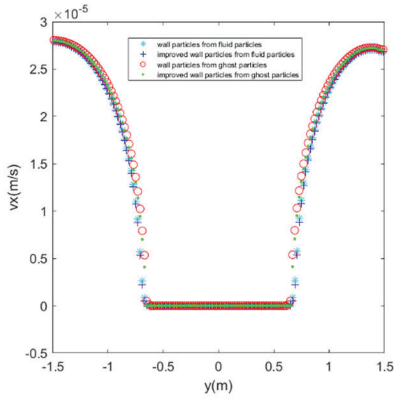


Figure 4.29 Comparison of the SPH velocity profiles along path 1 (**Figure 4.26**) using the IBT and BT techniques for Re = 1.

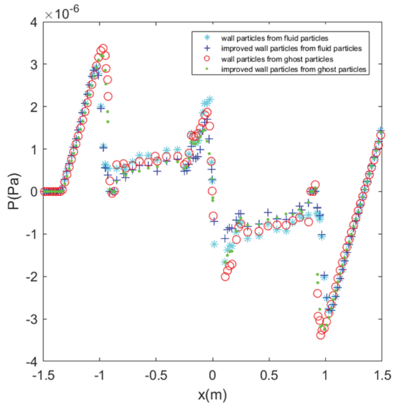


Figure 4.30 Comparison of the SPH pressure profiles along path 2 (**Figure 4.26**) using the IBT and BT techniques for Re = 1.

Figure 4.28 shows the velocity magnitude around the implicit surface and the contour plots of the velocity magnitude using the IBT and BT techniques for Re = 1. The results indicate the same behaviour as in the case of flow past a cylinder throughout the flow domain. The performance of the IBT and BT techniques are approximately the same over the entire computational domain.

A comparison of the velocity profiles along path 1 (indicated in **Figure 4.26**) appears in **Figure 4.29**. The velocity profiles are good agreement along path 1 with nearly insignificant differences between the IBT and BT techniques. However, the pressure in **Figure 4.30** shows higher fluctuations for the BT technique compared to the IBT technique, especially at positions close to the sharp edges as shown in **Table 4.5**.

Table 4.5 Comparison of the SPH pressure for Re = 1.

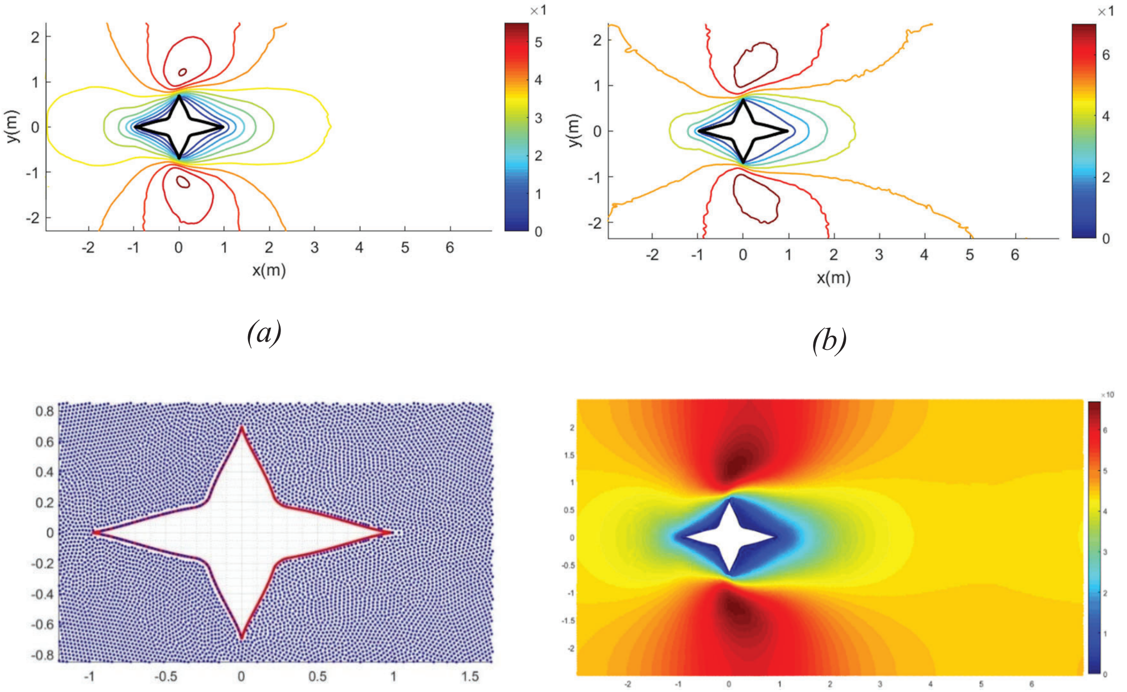
	Maximum Pressure (Pa)		Minimum Pressure (Pa)	
	Angle 1	Angle 2	Angle 2	Angle 3
Improved wall particle from fluid particles	2.905×10^{-6}	1.702×10^{-6}	-1.106×10^{-6}	-2.795×10^{-6}
Improved wall particle from ghost particles	3.213×10^{-6}	1.453×10^{-6}	-1.504×10^{-6}	-3.185×10^{-6}
Wall particle from fluid particles	2.935×10^{-6}	2.164×10^{-6}	-1.666×10^{-6}	-2.814×10^{-6}
Wall particle from ghost particles	3.375×10^{-6}	1.868×10^{-6}	-2.006×10^{-6}	-3.383×10^{-6}

b, Flow over a complicated shape with low Reynolds number

In this case, a computational domain with the implicit function from the previous case with d = 2 m is located at (0, H/2) in a domain with L = 10.0 m, a height of H = 5 m, a viscosity of v = 3.0×10^{-2} m^2 s^{-1}, a constant body force of F = 1×10^{-7} m s^{-2}, and a sound speed of c = 7.5×10^{-3} m s^{-1}. The fluid phase was discretized with particles placed initially at a resolution of Δx = 0.025 m with 78,892 fluid particles and approximately 600 wall particles.

The tensile instability in SPH is always challenging, especially at sharp corners. In this simulation, we use a particle shifting technique in order to remove for this issue of simulation.

Figure 4.31(a) and **(b)** illustrate the contour plots of velocity magnitude using the IBT technique for Re = 30 and 40, respectively. The performance is about the same as the second test case with a cylinder. **Figure 4.31(c)** shows a distribution of fluid particles around the implicit function without a numerical cavitation which violate the accuracy of simulation. Thus, a good results of velocity magnitude obtained as shown in **Figure 4.31(d).**



(a) *(b)*

(c) *(d)*

Figure 4.31 The contour plots of velocity magnitude, particle positions and velocity magnitude using the IBT technique for Re=30 and Re=40

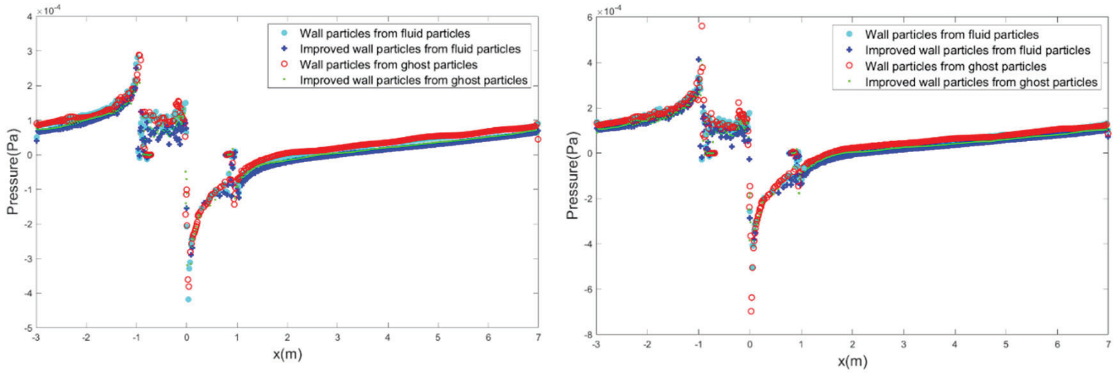


Figure 4.32 Comparison of the pressure profiles along the path 2 in Fig. 4.26 using the IBT and BT techniques for Re = 30 and Re = 40.

Figure 4.32 illustrates the pressure curves along indicated path 2 as in **Figure 4.26** of the IBT and BT techniques. Following the figure, the difference of pressure at the highest point to lowest point in the IBT technique is lower than the BT technique. Thus, the pressure distribution in the IBT technique is lower fluctuation than the BT technique in whole the computational domain. **Table 4.6** and

Table 4.7 show the maximum and minimum value of pressures in the IBT and BT techniques at the position close the sharp edges. It is clearly seen that, the value of pressure with the IBT technique is better than the BT technique with a lower oscillation of pressure performance.

Table 4.6 Comparison of pressure for Re=30

	Maximum Pressure(Pa)		Minimum Pressure(Pa)	
	Angle 1	Angle 2	Angle 2	Angle 3
Improved wall particle from fluid particles	2.51x10^{-4}	1.002x10^{-4}	-2.892x10^{-4}	-1.232x10^{-4}
Improved wall particle from ghost particles	2.459x10^{-4}	1.324x10^{-4}	-3.191x10^{-4}	-1.263x10^{-4}
Wall particle from fluid particles	2.808x10^{-4}	1.497x10^{-4}	-4.178x10^{-4}	-1.260x10^{-4}
Wall particle from ghost particles	2.888x10^{-4}	1.552x10^{-4}	-3.805x10^{-4}	-1.434x10^{-4}

Table 4.7 Comparison of pressure for Re=40

	Maximum Pressure(Pa)		Minimum Pressure(Pa)	
	Angle 1	Angle 2	Angle 2	Angle 3
Improved wall particle from fluid particles	4.13x10^{-4}	1.182x10^{-4}	-3.812x10^{-4}	-1.387x10^{-4}
Improved wall particle from ghost particles	4.085x10^{-4}	1.421x10^{-4}	-3.866x10^{-4}	-1.769x10^{-4}
Wall particle from fluid particles	3.34x10^{-4}	1.776x10^{-4}	-5.066x10^{-4}	-1.162x10^{-4}
Wall particle from ghost particles	5.605x10^{-4}	2.23x10^{-4}	-6.973x10^{-4}	-1.458x10^{-4}

Chapter 5 A flexible boundary treatment for implicit surfaces

In Chapter 4, we have successfully overcome the challenge in generating wall particles along implicit surfaces. The deficiency of particles close to the wall in the support domain was compensated with several layers of boundary particles. However, these boundary particles take high memory consumption and increase the number of particles in whole simulations. Several simulations require number of boundary particles approximate to fluid particles, as in Xu [73]. The construction of boundary particles can consider a precomputing process; however, it is not straightforward for complex simulations. To be well-balanced of computation costs and accuracy, we propose a flexible boundary treatment for implicit surfaces. This technique focuses on addressing the difficulty of efficiently determining appropriate particle motion around free surface and implicit surfaces. A modification of the PST helps to reduce computation costs in the free surface treatment. A new formulation of the PST and repulsive force for implicit surfaces are proposed to achieve stability at the interaction of fluid and obstacles. This approach can be a significant breakthrough to minimize boundary particles to compare with the conventional techniques.

5.1 The proposed techniques

The PST has demonstrated robustness and effectiveness, especially for dealing with highly dynamic fluid flows. Generally, the PST prevents tensile instability and avoids numerical cavitation in the fluid domain [53,56-59]. The PST is based on Fick's law of diffusion to disposition particles toward regions with low particle distribution in order to avoid cavitation. Although the PST has achieved significant results in surmounting particle clustering and has become indispensable in SPH simulation, it still suffers from inaccuracy and instability problems for free surfaces. Particularly, the shifting process of particles located at the free-surface and vicinity region tends to move toward the free surface. It causes small gaps in the particle distribution, as shown in **Figure 5.1**

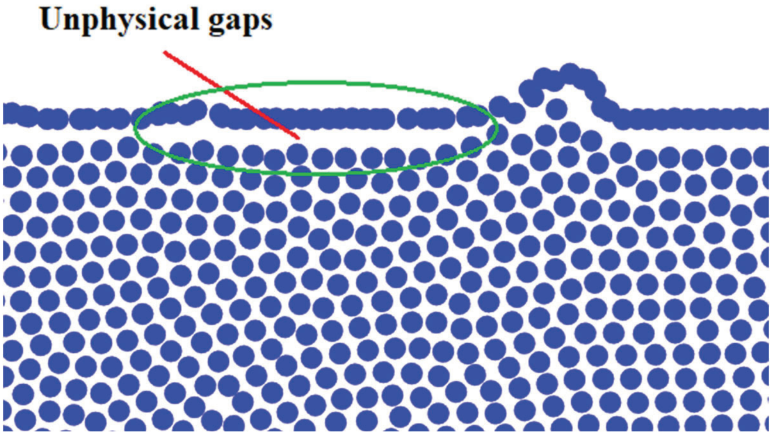


Figure 5.1 Simulation with the conventional PST

To overcome this issue, several studies have been proposed [56-59] to improve the PST.

A free surface detection algorithm was developed by Marrone et al. [68], where free surface particles and their normal vectors are obtained using the properties of the renormalization tensor:

$$\boldsymbol{L}(r_i) := \left[\sum_j (r_j - r_i) \otimes \nabla_i W_{ij} V_j\right]^{-1}. \tag{5.1}$$

Doring [69] showed that the value of the minimum eigenvalue λ of the tensor $\boldsymbol{L}^{-1}$ inside the interval [0,1]. The value tends to 0 when particles approach the free surface and to 1 for particles sufficiently far from the free surface. Marrone et al [68] and Sun [53] pointed out that particles with an eigenvalue less than 0.75 belong to the free surface. The normal vector field is computed as in Sun [53] and has a direction perpendicular to the free surface, as shown in **Figure 5.2**.

$$\langle \nabla \lambda_i \rangle := \sum_j (\lambda_j - \lambda_i) \otimes \boldsymbol{L}_i \nabla_i W_{ij} V_j. \tag{5.2}$$

$$\boldsymbol{n}(r_i) = \frac{\langle \nabla \lambda_i \rangle}{|\langle \nabla \lambda_i \rangle|}.$$

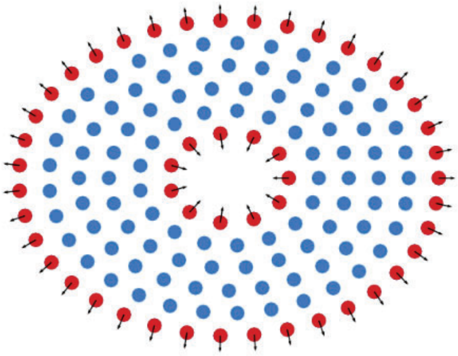


Figure 5.2 Normal vector of free surface particles

Many studies have been proposed to modify the conventional particle shifting for free surfaces [53,70,71,**Error! Reference source not found.**]. An effective approach was proposed by Krimi [70] for WCSPH to address the particles' instability. Different to Eq. (2.12), a modification of formulation is expressed as follows:

$$\delta \boldsymbol{r}_i = -\phi CFL\,.\,Ma\,.2h_i^2 \sum_j \left[\xi + 0.2\left(\frac{W_{ij}}{W_{\Delta x}}\right)^4\right] \nabla_i W_{ij} V_j \tag{5.3}$$

where ξ is equal to 1 when particles belong to the inner region of the domain and 0 for particles in the free surface region. The ϕ is defined by

$$\phi = \begin{cases} 0 & r_i \in free\ surface \\ \dfrac{r_{ij_{min}}.n_{j_{min}}}{kh} & r_i \in inner\ free\ surface \\ 1 & r_i \in inner\ region \end{cases} \tag{5.4}$$

where $r_{ij_{min}}.n_{j_{min}}$ is the nearest distance of particle i to the free surface. In order to determine these distances, particles need to look up all free surface particles over their corresponding kernel support and find the minimum distance. This process is not complicated but increases computation cost and may lead to some inconvenience process. To simplify this process, we proposed a modification of the PST to avoid the computation.

5.1.1 Modification of the particle shifting technique

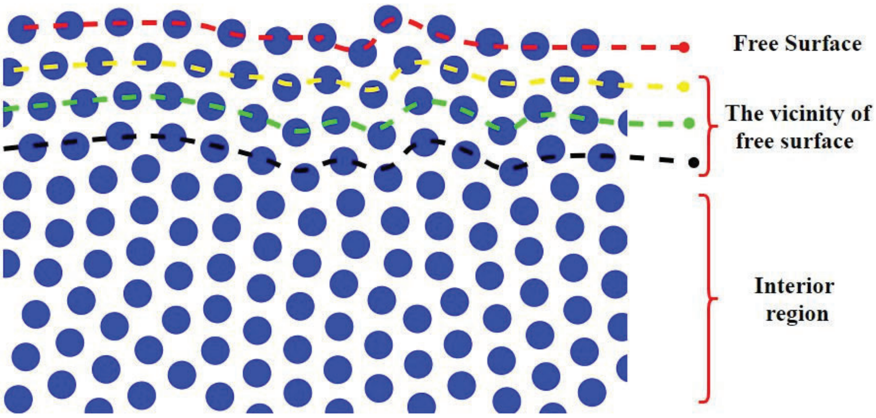


Figure 5.3 Particle distribution during the simulation

To determine the position of particles in the vicinity of a free surface, we observe on several kernel functions as Quintic spline, Gaussian kernel function with a support radius of $3h$, where h is the smoothing length and equals $1.33\,\Delta x$. Δx is the initial spacing of particles. The PST helped to preserve the spacing of particles during simulation approximately the initial spacing Δx, as shown in **Figure 5.3**. We assume that red dashed line is free surface particles with the first layer. Yellow, green, and black dashed line is corresponding with the second, third, and fourth layer, respectively. They are layers of the vicinity of free surface. It can be seen that the distance of each layer in range $\left(\Delta x, \sqrt{2}\Delta x\right)$. In addition, we discovered that the eigenvalue values for 2D and 3D cases are mostly similar in the same layers. Therefore, we can divide the particles in the vicinity of the free surface based on the eigenvalue values.

$$\boldsymbol{r}_i \in \begin{cases} \text{First layer} & \text{i } \in \text{free surface} \\ \text{Second layer} & 0.5 < \lambda_i \leq 0.875 \\ \text{Third layer} & 0.875 < \lambda_i \leq 0.985 \\ \text{Fourth layer} & 0.985 < \lambda_i < 1.0 \end{cases} \tag{5.5}$$

Following the classification of the eigenvalue λ , we can rewrite Eq (5.4) as follows:

$$\phi = \begin{cases} 0 & \lambda \leq 0.5 \\ \dfrac{\Delta x}{kh} & 0.5 < \lambda \leq 0.875 \\ \dfrac{2\Delta x}{kh} & 0.875 < \lambda \leq 0.985 \\ \dfrac{3\Delta x}{kh} & 0.985 < \lambda < 1.0 \\ 1 & \lambda \geq 1.0 \end{cases} \tag{5.6}$$

Based on the value of eigenvalue that computed beforehand, this division help us remove a computation process of distance to free surface and help the implementation of ϕ become faster and simpler computation.

5.1.2 The particle shifting technique with implicit surface

In SPH simulations, boundary particles are generated to compensate for the deficiency of particles close to the walls. However, these boundary particles consume a lot of computational time and increase the number of particles. The number of boundary particles used in simulations are approximate to fluid particles, as in Xu [73]. Dealing with this challenge, we propose an approach without boundary particles at the beginning of the simulation. We assume that boundary particles are not used for a solid body; thus, particles near the wall can be considered free-surface particles. Consequently, the truncated kernel function causes instability and inaccuracy values for these particles. We still need the PST to correct the particle information to minimize errors caused by the cutoff kernel function. ϕ is expressed as follows:

$$\phi = \begin{cases} 0 & \in free\ surface\ and\ \lambda \leq 0.5 \\ \dfrac{distance\ to\ surface}{kh} & distance\ to\ surface < kh\ and\ \lambda \leq 0.5 \end{cases} \tag{5.7}$$

To prevent particle penetration through an implicit surface, we use a formulation presented in [19]. The activate condition of this formulation is the distance between a particle and an implicit surface to be less than *d*, which is normally chosen to be equal to Δx.

$$\boldsymbol{f}_i = m_i \frac{d - distance\ to\ surface}{\Delta t^2} . \boldsymbol{n_i} \qquad if\ \ distance\ to\ surface < d \tag{5.8}$$

where $\boldsymbol{n_i}$ is the normal vector of a particle to an implicit surface calculated by Eq. (2.18). The calculation of the distance and normal vector uses all particles in the computational domain. Particles that are located only at a free surface may penetrate the surface. On the basis of the eigenvalue, we select free-surface particles, with $\lambda \leq 0.5,$ instead of using all particles. Moreover, the normal vector of free-surface particles tends to move toward the free surface in **Figure 5.2**. It is similar to the normal vector to the implicit surface. Thus, we can substitute the normal vector of free-surface particles in Eq. (5.2) into Eq. (5.8). Therefore, we can rewrite Eq. (5.8) as follows:

$$\boldsymbol{f}_i = \alpha . m_i \frac{d - distance\ to\ surface}{\Delta t^2} . \boldsymbol{n}_i \qquad \begin{cases} if\ \ \lambda_i \leq 0.5 \\ distance\ to\ surface < d \end{cases} \tag{5.9}$$

Here, we introduce a new parameter named "α" to reduce the effect of repulsive force. The repulsive force $\boldsymbol{f}_i$ in Eq. (5.8) prevented particles from penetrating the solid wall by providing a force opposite to its movement. However, this force may cause unphysical influence and instability at the region near the implicit surface. Therefore, α should be choose in interval $[0,1]$.

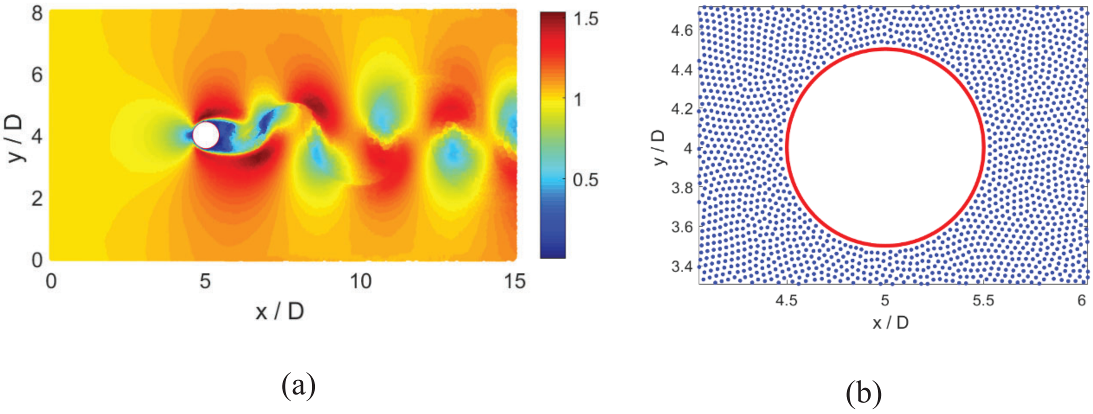
Generally, the no-slip or free-slip boundary condition is imposed on the solid walls in the WCSPH. However, this is complex for implicitly defined walls. To make this process easier, temporary wall particle are generated at each time step of the simulation. Particles with an eigenvalue $\lambda \leq 0.5$ and a distance to less than Δx the implicit surface will be reflected right on the wall.

5.2 The simulation test cases

5.2.1 Flow past a circular cylinder

To evaluate the accuracy of the proposed technique, a test case, as described in Section 4.1.2.1 is conducted at $Re\ = 200$. In contrast to the previous simulation, no boundary particles were generated at the start of this simulation. Fluid particles are distributed in the entire computational domain. The simplified open boundary conditions in Section 3.2 are imposed on the inlet/outlet. During the simulation, temporary boundary particles are created corresponding to fluid particles surrounding the circular cylinder using the technique in Section 4.2 and the no-slip boundary condition is enforced.

Figure 5.4 (a) illustrates the velocity contours at $Re = 200$. The velocity fields are smooth with the unsteady von Karman street behind the cylinder at $Re = 200$. **Figure 5.4 (b)** shows the distribution of fluid particles around the implicit function without numerical cavitation. Hence, the performance of the proposed technique is reliable.



(a)

(b)

Figure 5.4 Two-dimensional flow past a cylinder: (a) velocity field and (b) particle distribution surrounding the implicit surface at $Re = 200$

Figure 5.5 shows the time history of lift and drag coefficient around the circular cylinder. The drag and lift coefficients converge to a steady value of $C_D = 1.55$, $C_L = \pm 0.83$ for $Re = 200$. The obtained results with a single layer are higher than those of Tafuni [28] and agree with those obtained using multiple layers.

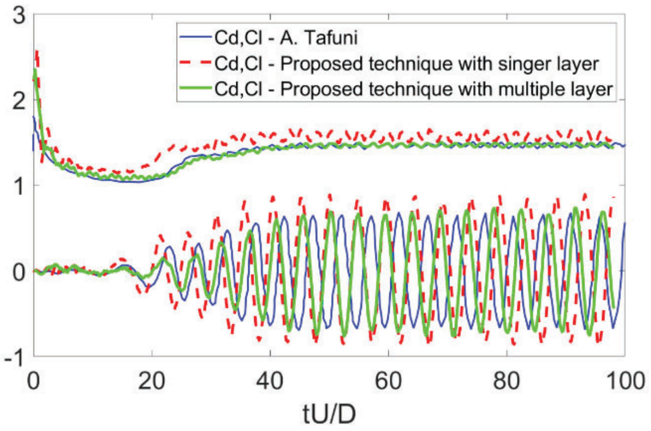


Figure 5.5 Time history of lift and drag coefficient

Table 5.1 Comparison of C_D and C_L values using different method for $Re = 200$

	C_D	C_L
Proposed technique – single layer	1.55 ± 0.05	± 0.830
Proposed technique–multiple layers	1.46 ± 0.05	± 0.701
Negi et al. [37]	1.54 ± 0.05	± 0.729
Guerrero [102]	1.409 ± 0.048	± 0.725
Marrone et al. [49]	1.38 ± 0.05	± 0.680
Tafuni et al.[28]	1.46 ± 0.0	± 0.693

A comparison of methods is shown in **Table 5.1**. The proposed technique with a single layer obtained results in good agreement with those obtained using the other techniques, which used multiple layers of boundary particles. The drag coefficients are approximate to those in Negi et al. [37], and the lift coefficients are slightly higher than those of the other techniques. Based on the performance of the proposed technique, it is an appropriate approach for a well-balanced accuracy and computation cost.

Better results can be obtained using multiple layers, but it increases the number of particles and the computational time. Thus, the choice of accuracy and computation cost depends on the expected outcome of each simulation. Better accuracy and computational cost can be achieved with the proposed technique by replacing the single layer with multiple layers, ensuring that the full kernel support is at a position close to the implicit surface. This proves that the proposed technique is flexible and effective.

5.2.2 Dam break simulation

The proposed technique is applied to study the challenging 3D dam-break flow against a vertical wall. The implicit surface of tank is defined as the zero-level set of the function: $f(x, y, z) = 0$.

$$f(x, y, z) = \{x_{min} - x, x - x_{max}, y_{min} - y, y - y_{max}, z_{min} - z, z - z_{max}\} \tag{5.10}$$

The initial setup of the simulation is illustrated in **Figure 5.6** . The computational domain is $[0, L] \times [0\,, W] \times [0\,, 3.4H]$ where $H = 0.6$ m is the height of the water column, $L = 5.366H$ and $W = 2H$. The width and length of water column are $2H$. We observe the water height evolution at Points A and B at the centerline of the dry horizontal bed with marked positions of $3.721H$ and $4.542H$, respectively.

The density of fluid $\rho_0 = 1000$ kg m^3 and the viscosity $\eta = 10^3$ Pa s. The gravity acts in the downward direction with $g = -9.81$ m s^2. The initial particle spacing is equal $H/\Delta x = 25$ with 62500 fluid particles. A time step $\Delta t = 0.0001$ is used in the entire computation process to ensure numerical stability.

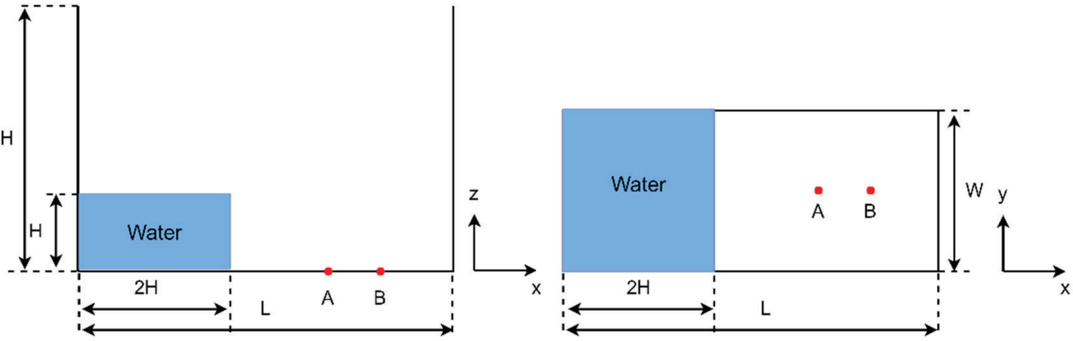
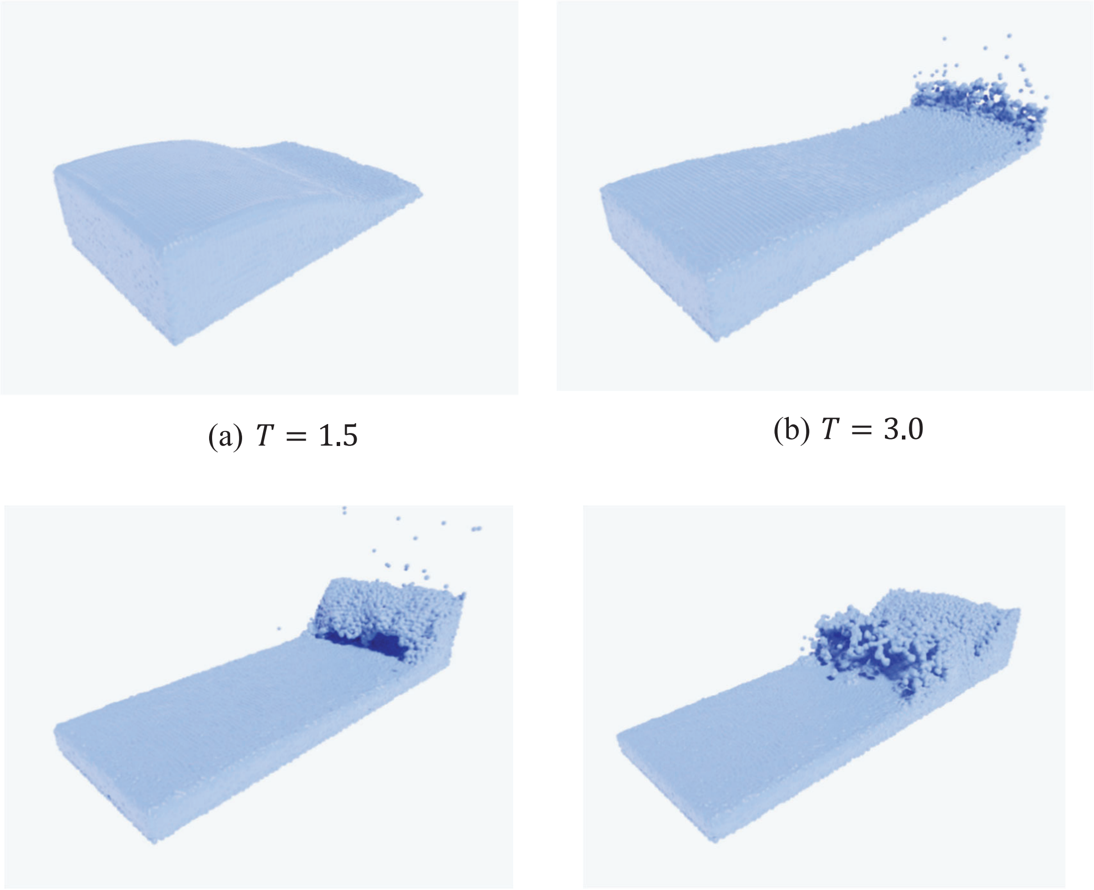


Figure 5.6 Sketch of 3D dam-break simulation

Figure 5.7 shows the results of the 3D dam-break simulation obtained using the proposed technique at four different times $T = 1.5, 3.0, 5.5$ and 7.5, where the time is $T = t\sqrt{g/H}$. The water column collapses under the influence of gravity force and propagates along the dry horizontal bed. The water column makes contact with the front vertical wall, and a vertical jet is formed along the front vertical wall. The retroflection phenomenon is observed when the flow overturns backward.



(a) $T = 1.5$

(b) $T = 3.0$

(c) $T = 5.5$

(d) $T = 7.5$

Figure 5.7 Numerical simulation of 3D dam-break simulation at different times T

Figure 5.8 and **Figure 5.9** depict a comparison of the time histories of the water height evolution at Points A and B obtained using the proposed technique with those obtained by Xu et al. [73], Zhou et al. [74], and Ferrari et al. [75]. A good agreement is observed between the solvers. Specifically, in **Figure 5.8**, the time interval $T \in [1.4, 2.1]$, the water height at point A obtained using the proposed technique is slightly lower than that obtained by Zhou et al. [74] but nearly comparable with that obtained by Xu et al. [73] and Ferrari et al. [75]. The proposed technique is consistent with Ref [73,75] but outperforms Ref [74] at $T \in [2.1, 6.4]$. There is a difference at $T \in [6.4, 8.0]$, where the proposed technique outperforms other solvers

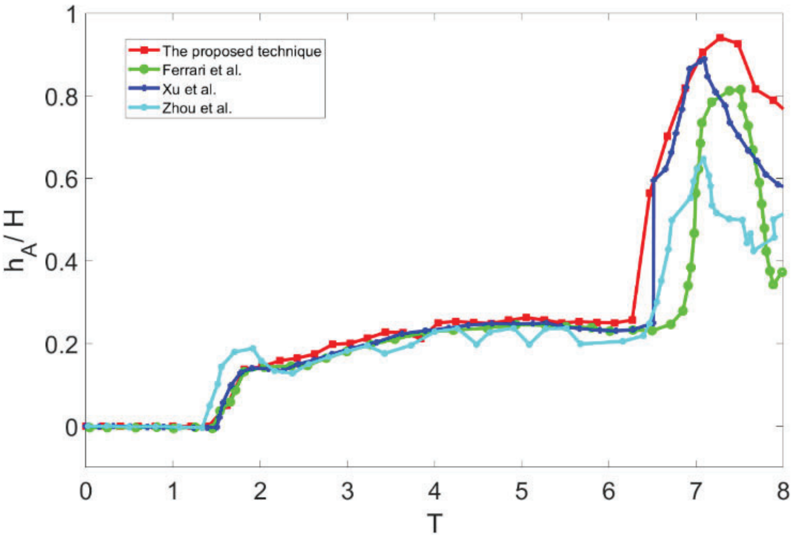


Figure 5.8 Comparison of the time histories of water height evolution at point A

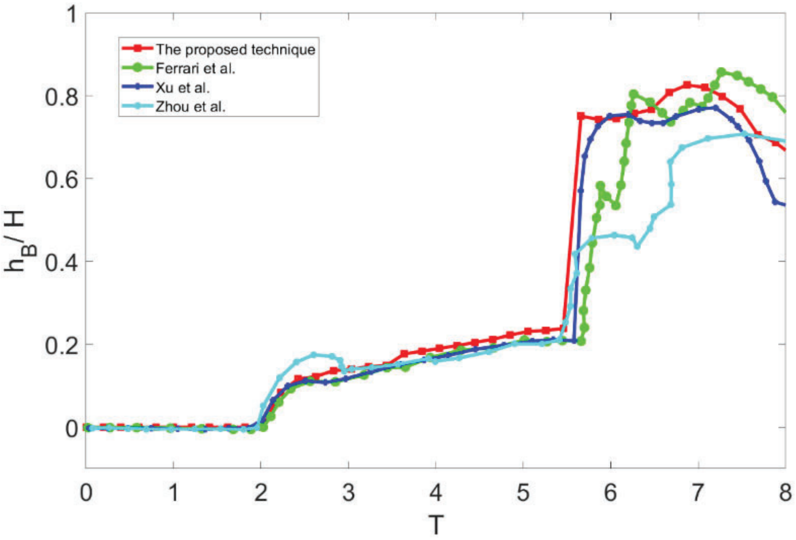


Figure 5.9 Comparison of the time histories of water height evolution at point B

Similar to the water height at Point B in **Figure 5.9**, there is a consistent performance between the solvers at $T \in [0.0, 5.5]$. At $T \in [5.5, 7.0]$, the proposed technique agrees well with Ref [73,75], whereas the water height in Ferrari et al. [75] is higher than the remaining results at $T \in [7.0, 8.0]$.

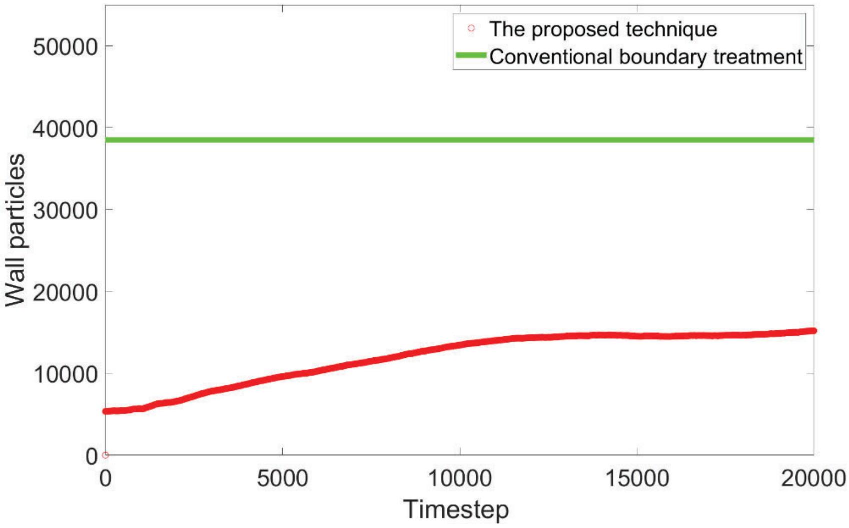


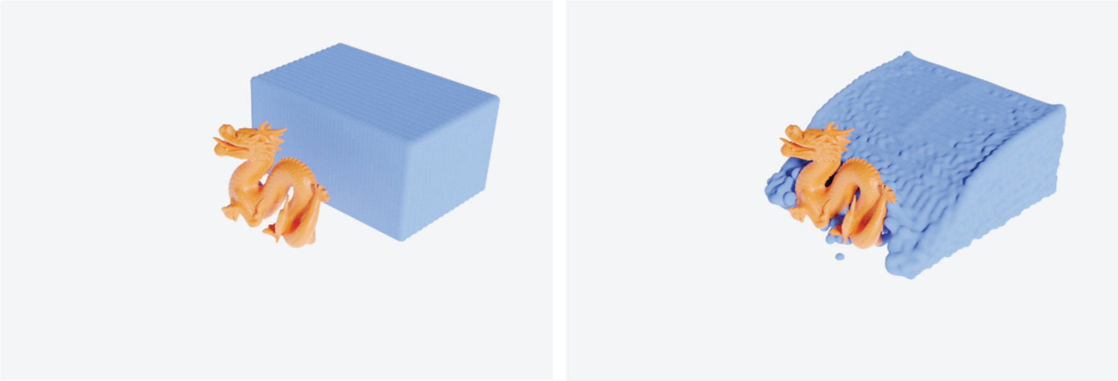
Figure 5.10 Comparison of boundary particles in the proposed technique and conventional technique in dam-break simulation.

Figure 5.10 illustrates a comparison of boundary particles in simulation. It is observed that the simulation starts with no boundary particle. After checking the distance from fluid particles to the tank, the temporary boundary particles with 5500 particles are created from the proposed technique in the first time step. These particles have variation with respect to time corresponding to fluid particles around the implicit surface. The maximum boundary particles used equal 15000 particles to compare with 38500 boundary particles used in conventional boundary treatment. In the conventional technique, boundary particles are generated at the initial time of simulation for the entire solid wall and are at a fixed position and the number of particles in the entire computation. By contrast, the proposed technique only generates temporary boundary particles corresponding to fluid particles close to the surface. It helps reduce the number of boundary particles at unnecessary positions on solid walls. The disadvantage of the proposed technique is that boundary particles need to be updated at each time step. Therefore, the distance between fluid particles and implicit surfaces must be computed in each time step. However, the use of free-surface particles with eigenvalue $\lambda \leq 0.5$ instead of all particles is the breakthrough improvement of the proposed technique. Generally, particles with eigenvalue $\lambda \leq 0.5$ are approximately 7% percentage of the total number of particles. Hence, this computation cost is much lower than when using boundary particles and finding their nearest neighboring particles in conventional boundary treatment.

5.2.3 Fluid simulation in computer graphics

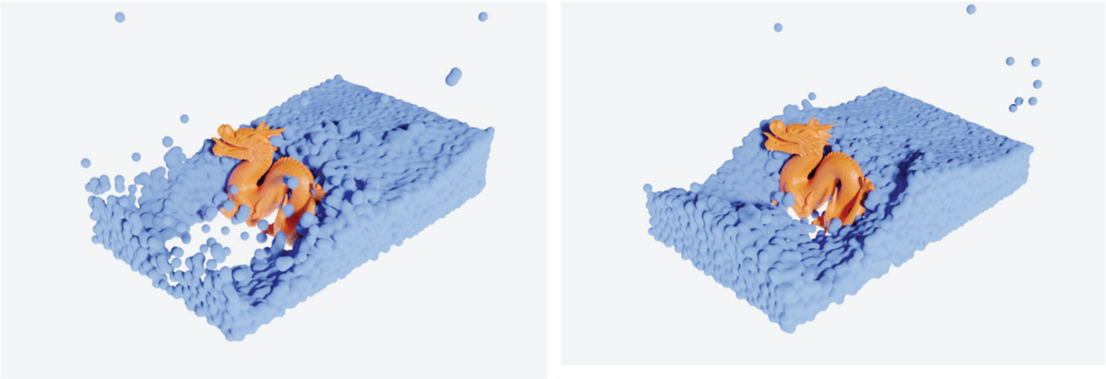
In this section, several simulations in computer graphics are performed. Implicit surfaces such as a dragon or bunny shape are obtained from the RBF function [13]. The results are rendered by applying metaballs [40,41] to fluid particles.

In a 3D dam break past a solid dragon, approximately 9600 fluid particles are used. **Figure 5.11** shows the simulation results in 8000 time steps with $\Delta t = 0.0002$. The water column was propagated along the dry horizontal bed and partially blocked by the solid dragon. Eq. (5.9) was activated when fluid flow approaches the solid dragon to prevent fluid particles from penetrating the solid wall. Temporary boundary particles are generated during the computation shown in **Figure 5.12**. It is observed that no boundary particle is generated using the proposed technique in the first 1000 time steps. Temporary boundary particles are created when fluid particles approach close to the solid dragon. These particles varied with respect to the time corresponding to fluid particles around the dragon surface. A maximum of 355 boundary particles is used in the proposed technique, whereas approximately 3000 boundary particles are used in conventional boundary treatment.



(a) Initial state

(b) Step 3000



(c) Step 5000

(d) Step 8000

Figure 5.11 Dam break simulation past a solid dragon

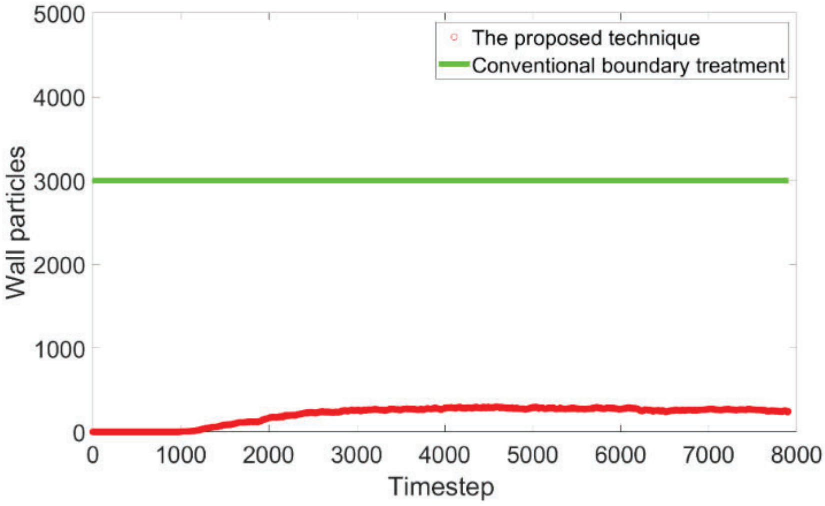
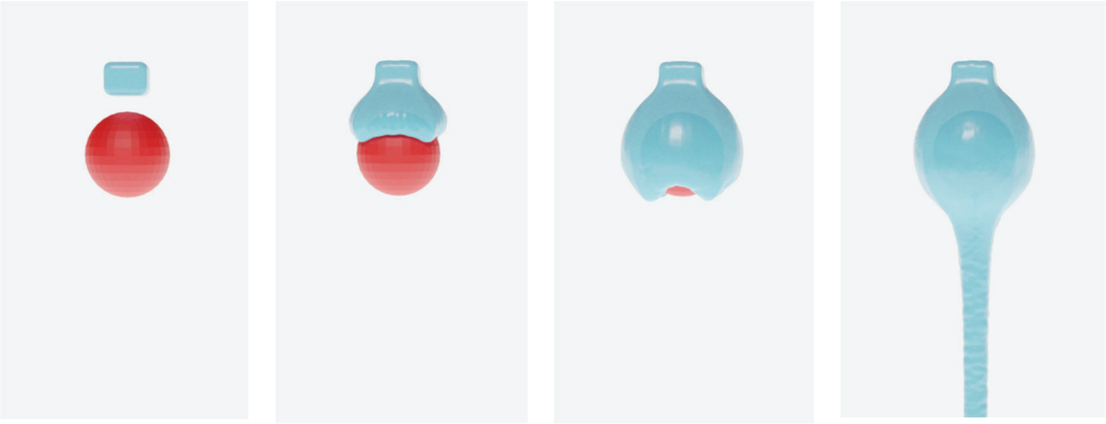


Figure 5.12 Comparison of boundary particles in the proposed technique and conventional technique on a solid dragon

In the simulation of water pouring on the sphere, the source of water is located at the overhead of the sphere. At the beginning of the simulation, 400 inflow particles and 700 fluid particles are used. A simplified open boundary condition in Section 3.2 is easily used to create a stability source. **Figure 5.13** shows the results of the simulation after 15000 steps with $\Delta t = 0.0001$. The fluid flow falls on the sphere under the impact of gravity force and sticks along the sphere before dropping on the ground. During the simulation, the number of particles increased from 1100 particles to 25000 particles because of the evolution of the fluid flow. The fluid particles stick together, causing adhesion effects, at the fluid–solid boundary. The fluid particles behave adequately in the vicinity of the sphere and free surface, which are controlled by improving the PST.



(a) Initial state (b) Step 2000 (c) Step 5000 (d) Step 15000

Figure 5.13 Fluid drops on sphere

Figure 5.14 shows a comparison of boundary particles in the proposed technique and conventional technique during the computation. The number of wall particles gradually increased and reached a number comparable to conventional boundary treatment after 15000 time steps. It works well when fluid flow fully sticks on a solid sphere.

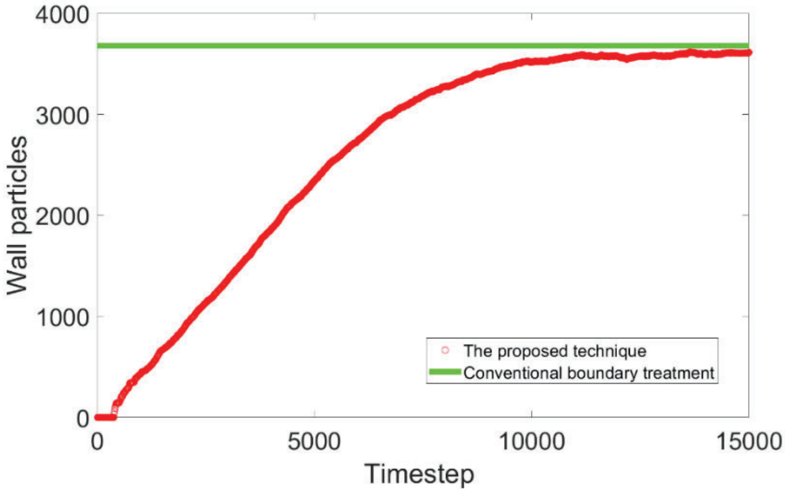


Figure 5.14 Comparison of boundary particles in the proposed technique and conventional technique on solid sphere

In the simulation of a fluid sphere dropping on the bunny, the fluid sphere is also located at the overhead of the solid bunny. Approximately 6000 fluid particles are used in this simulation. A maximum of approximately 1000 wall particles are used at time step 2500, as shown in **Figure 5.15**. Compared with the conventional technique used for the entire solid bunny, it is less than five times.

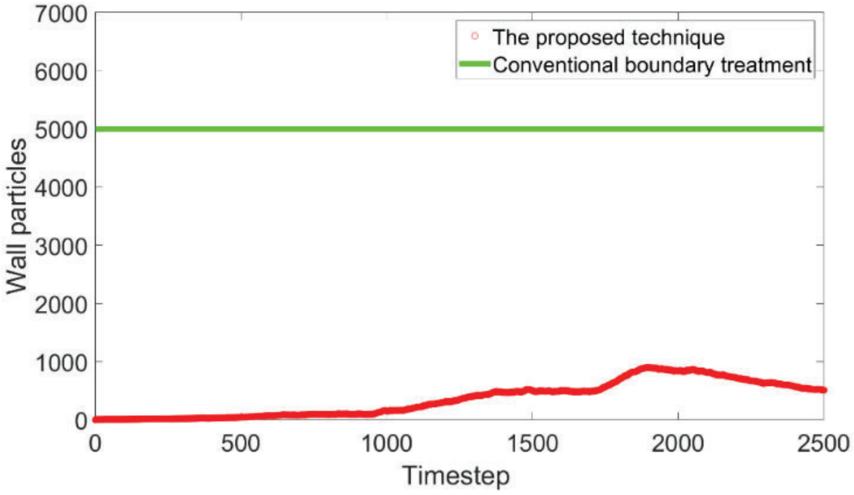
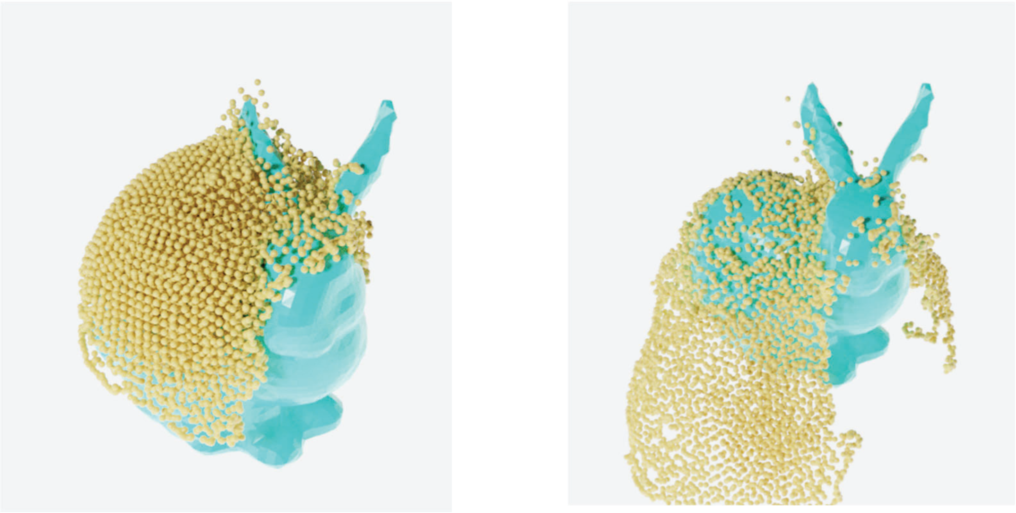


Figure 5.15 Comparison of boundary particles in the proposed technique and conventional technique on solid bunny

Figure 5.16 shows the simulation results in 2000 time steps with $\Delta t = 0.00025$. The sphere water drops on the solid bunny's head and back, separating both sides of objects. The result has a smooth performance across the entire computational domain.

(a), Initial state

(b), Step 1000



(c), Step 1500

(d), Step 2000

Figure 5.16 Fluid sphere drops on solid bunny

Chapter 6 Conclusions and future work

6.1 Conclusions

The thesis objectives have successfully addressed in boundary treatment for the SPH method. The following are some results we obtained:

- The complication in implementing open BC have handled by NROBC and a simplified technique. NROBC based on the hybrid of in/outflow algorithm and periodic boundary condition is presented. The technique can be considered an improvement of periodic boundary conditions. Thus, it eliminated the violation of the periodic boundary conditions while ensuring the conservation of mass. A simplified approach is developed to expand the simulation with different length sizes in open boundary conditions. Benefiting from the δ^{+}-SPH scheme, the proposed technique is simple to implement and is effective in test cases. Inflow/outflow information is treated to ensure appropriate evolution to the flow field instead of using a renormalization process referencing the fluid particles. The test case examined the flexibility of inflow boundary conditions to prescribed or nonprescribed values. The performance demonstrated the versatility of the proposed technique.
- The challenge in generating wall particles along implicit surface have overcome successfully with boundary treatment techniques. We proposed approaches for generating boundary particles using implicit surfaces. In these approaches, wall particles can be computed as preprocessing for static obstacles. The boundary treatment technique provides a full treatment for complicated geometries and can handle sharp corners without any special treatment. The deficiency of particles close to the wall in the support domain was compensated for. The numerical tests demonstrate that the proposed technique obtains good results with a high agreement with the reference solutions.
- Finally, we proposed a flexible boundary treatment for implicit surfaces to deal with the difficulty of efficiently determining appropriate particle motion around free surface and implicit surfaces. This technique can use a few boundary particles

as temporary particles instead of fixed particles in conventional techniques. This approach is a significant breakthrough that can eliminate costly ghost particles in the computational process, paving a way for real-time simulation with high accuracy. A modification of the PST also proposed to reduce computation costs in the free surface treatment. A new formulation of the PST and repulsive force for implicit surfaces achieved stability at the interaction of fluid and obstacles. thereby ensuring ensures high accuracy and fast computation. The obtained results demonstrate the robustness and versatility for various simulations in computer graphics.

6.2 Future works

- A simplified approach in Sec 3.2 is addressed various simulations in open boundary conditions. However, the flow reversion is a limitation of this technique. We expect to improve this issue in future work.

- The idea of a flexible boundary treatment for implicit surfaces in Sec 5 is well-suited for simulations with deformable objects. The temporary boundary particles are straightforward to generate at deformable positions which big issues in conventional techniques.

- The combination of numerical methods and machine learning is cutting-edge technology and becoming a trend. Recently, several approaches such as data-driven method, physics-driven method, or combined data-driven and physics-driven method have obtained impressive results and have a lot of potentials to exploit. These approaches help broaden the scope of simulation issues and acquire fast computation and more accurate solutions. We expect to pursue this approach in future work.

References

1. Andre Brock, Theodore Lim, James M. Ritchie, and Nick Weston. “Generative and discriminative voxel modeling with convolutional neural networks”. arXiv.org, 1608.04236, 2016.

2. Matheus Gadelha, Subhransu Maji, and Rui Wang. “3d shape induction from 2d views of multiple objects”. In Proc. of the International Conf. on 3D Vision (3DV), 2017.

3. Haoqiang Fan, Hao Su, and Leonidas J. Guibas. “A point set generation network for 3d object reconstruction from a single image”. Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2017.

4. Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas J. Guibas. “Learning representations and generative models for 3D point clouds”. In Proc. of the International Conf. on Machine learning (ICML), 2018.

5. Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. “Learning category-specific mesh reconstruction from image collections”. In Proc. of the European Conf. on Computer Vision (ECCV), 2018.

6. G. Lavoué, F. Dupont. “Technical Section: Semi-Sharp Subdivision Surface Fitting Based on Feature Lines Approximation", Computer and Graphics, vol. 33, no. 2, pp. 151-161. 2009

7. Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. “AtlasNet: A papiermache approach to learning 3d surface generation”. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2018.

8. Z. Chen and H. Zhang. “Learning implicit fields for generative shape modeling”. Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2019.

9. Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. “Occupancy networks: Learning 3d reconstruction in function space”. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2019.

10. L. Cheng, S. Chen, X. Liu, H. Xu, Y. Wu, M. Li, Y.Chen. “Registration of Laser Scanning Point Clouds: A Review”. Sensors, 18, 1641–1666, 2018.

11. Berger M, Levine JA, Nonato LG, Taubin G, Silva CT. “A benchmark for surface reconstruction”. ACM Trans Graph (2013) 32: Article 20.

12. Manson J, Petrova G, Schaefer S. “Streaming surface reconstruction using wavelets”. Comput Graph Forum (2008) 27:1469–1476.

13. Süßmuth J, Meyer Q, Greiner G. “Surface reconstruction based on hierarchical floating radial basis functions”. Comput Graph Forum (2010) 29:1854–1864.

14. Kazhdan M, Hoppe H. “Screened Poisson surface reconstruction”. ACM Trans Graph (2013) 32:1–13.

15. T. Itoh and S. Nakata. “Fast Generation of Smooth Implicit Surface Based on Piecewise Polynomial”, Computer Modeling In Engineering & Sciences, Vol. 107, No.3, 187-199, 2015.

16. Akinci N., Akinci G., Teschner M.: “Versatile surface tension and adhesion for SPH fluids”. ACM Transactions on Graphics (Proc. SIGGRAPH Asia) 32, 6 (2013), 182:1–182:8

17. Akinci N., Cornelis J., Akinci G., Teschner M.: “Coupling elastic solids with smoothed particle hydrodynamics fluids”. Comp. Anim. Virtual Worlds 24 (2013), 195–203.

18. Müller M.,Heidelberger B., Hennix M., Ratcliff J. “Position based dynamics”. J. Vis. Commun. 18, 2 (2007), 109–118

19. S. Nakata, Y. Sakamoto. “Particle-based parallel fluid simulation in three-dimensional scene with implicit surfaces,” The Journal of Supercomputing. 2015;71(5):1766-1775

20. Kanetsuki Y., Nakata S. “Acceleration of Particle Based Fluid Simulation with Adhesion Boundary Conditions Using GPU”. In: Mohamed Ali M., Wahid H., Mohd Subha N., Sahlan S., Md. Yunus M., Wahap A. (eds) Modeling, Design and Simulation of Systems. AsiaSim 2017. Communications in Computer and Information Science, vol 752. Springer, Singapore. https://doi.org/10.1007/978-981-10-6502-6_30

21. Lucy L. B,” Numerical approach to testing the fission hypothesis”, Astronomical Journal, 82:1013-1024, 1977.

22. Gingold R. A. and Monaghan J. J, “Smoothed Particle Hydrodynamics: Theory and Application to Non-spherical stars”, Monthly Notices of the Royal Astronomical Society, 181:375-389, 1977

23. P.M. Joseph, J.F. Patrick, Y. Zhu, “Modeling low Reynolds number incompressible flows using SPH”, Journal of Computational Physics 136(1) (1997) 214–226.

24. E.-S. Lee, C. Moulinec, R. Xu, D. Violeau, D. Laurence, P. Stansby, “Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method”, Journal of Computational Physics 227(18) (2008) 8417–8436.

25. M. Lastiwka, M. Basa, N.J. Quinlan, “Permeable and non-reflecting boundary conditions in SPH”, International Journal for Numerical Methods in Fluids 61(7) (2009) 709–724.

26. M. Ferrand, A. Joly, C. Kassiotis, D. Violeau, A. Leroy, F.X. Morel, B.D. Rogers, “Unsteady open boundaries for SPH using semi-analytical conditions and Riemann solver in 2D”, Computer Physics Communications 210 (2017) 29–44.

27. A. Leroy, D. Violeau, M. Ferrand, L. Fratter, A. Joly, "A new open boundary formulation for incompressible SPH", Computers and Mathematics with Applications 72(9) (2016) 2417–2432.

28. A. Tafuni, J.M. Domínguez, R. Vacondio, A.J.C. Crespo, "A versatile algorithm for the treatment of open boundary conditions in smoothed particle hydrodynamics GPU models", Computer Methods in Applied Mechanics and Engineering 342 (2018) 604–624.

29. I. Federico, S. Marrone, A. Colagrossi, F. Aristodemo, M. Antuono, "Simulating 2D open-channel flows through an SPH model", European Journal of Mechanics B/Fluids 34 (2012) 35–46

30. C.E. Alvarado-Rodríguez, J. Klapp, L.D.G. Sigalotti, J.M. Domínguez, Edl Cruz Sánchez, "Nonreflecting outlet boundary conditions for incompressible flows using SPH", Computers and Fluids 159 (2017) 177–188.

31. R. Vacondio, B.D. Rogers, P.K. Stansby, P. Mignosa, "Sph modeling of shallow flow with open boundaries for practical flood simulation", Journal of Hydraulic Engineering 138(6) (2012) 530–541.

32. V.W. Azizi Tarksalooyeh, G. Závodszky, B.J.M. van Rooij, A.G. Hoekstra, "Inflow and outflow boundary conditions for 2D suspension simulations with the immersed boundary lattice Boltzmann method", Computers and Fluids 172 (2018) 312–317.

33. D. Molteni, R. Grammauta, E. Vitanza, "Simple absorbing layer conditions for shallow wave simulations with smoothed particle hydrodynamics", Ocean Engineering 62 (2013) 78–90.

34. C. Altomare, J.M. Domínguez, A.J.C. Crespo, J. González-Cao, T. Suzuki, M. Gómez-Gesteira, P. Troch, "Long-crested wave generation and absorption for SPH-based DualSPHysics model", Coastal Engineering 127 (2017) 37–54.

35. M.B. Liu, G.R. Liu, "Restoring particle consistency in smoothed particle hydrodynamics", Applied Numerical Mathematics 56(1) (2006) 19–36.

36. P. Wang, A. Zhang, F. Ming, P. Sun, H. Cheng. “A novel non-reflecting boundary condition for fluid dynamics solved by smoothed particle hydrodynamics”, Journal of Fluid Mechanics 860 (2019) 81–114.

37. P. Negi, P. Ramachandran, A. Haftu, “An improved non-reflecting outlet boundary condition for weakly compressible SPH”, Computer Methods in Applied Mechanics and Engineering 367 (2020), http://www.ncbi.nlm.nih.gov/pubmed/113119

38. Monaghan JJ, “Simulating free surface flows with sph”. J Comp Phys (1994) 110:399–406.

39. T. Harada, S. Koshizuka, Y. Kawaguchi, “Smoothed Particle Hydrodynamics in Complex Shapes,” Proceedings of the 23rd Spring Conference on Computer Graphics (pp. 191-197). ACM.

40. Blinn JF, “A generalization of algebraic surface drawing”. ACM Trans Graph (1982) 1:235–256.

41. Nishimura H, Hirai M, Kawai T et al. “Object modeling by distribution function and a method of image generation”. Trans Inst Electron Commun Eng Jpn. (1985) J68- D:718–725.

42. Bloomenthal J, Bajaj C, Blinn J et al. “Introduction to implicit surfaces”. Morgan Kaufmann Publishers. San Francíco (1997) 1997.

43. Wyvill B, Guy A, Galin E, “Extending the CSG tree. Warping, blending and boolean operations in an implicit surface modeling system”. Comput Graph (1999) 18:149–158.

44. G. Yngve, G. Turk, “Creating Smooth Implicit Surfaces from Polygonal Meshes”. Georgia Institute of Technology; 1999.

45. G. Taubin. “Distance Approximations for Rasterizing Implicit Curves”. ACM Transactions on Graphics. 1994; 13:3–42.

46. S. Park, G. Jeun. “Coupling of rigid body dynamics and moving particle semi-implicit method for simulating isothermal multi-phase fluid interactions,” Computer Methods in Applied Mechanics and Engineering. 2011; 200:130–140.

47. M. Yildiz, R. A. Rook and A. Suleman, “SPH with the Multiple Boundary Tangent Method,” International journal for numerical methods in engineering. 2008;77(10):1416–1438.

48. Adami S, Hu XY, Adams NA A generalized wall boundary condition for smoothed particle hydrodynamics. J Comp Phys (2012) 231:7057–7075.

49. S. Marrone, M. Antuono, A. Colagrossi, G. Colicchio, D. Le Touzé, G. Graziani. “δ-SPH Model for Simulating Violent Impact Flows,” Computer Methods in Applied Mechanics and Engineering. 2011;200(13-16):1526–1542.

50. T. Zhang, S. Koshizuka, K. Murotani, K. Shibata, E. Ishii, M. Ishikawa. “Improvement of Boundary Conditions for Nonplanar Boundaries Represented by Polygons with an Initial Particle Arrangement Technique,” International Journal of Computational Fluid Dynamics. 2016; 30(2):155–175.

51. G.R. Liu and M.B. Liu, “Smoothed Particle Hydrodynamics a Meshfree Particle Methods”. World Scientific Publishing Co. Pte. Ltd. : Singapore, 2003.

52. M.B. Liu, G.R. Liu, K.Y. Lam, “Constructing smoothing functions in smoothed particle hydrodynamics with applications”, Journal of Computational and Applied Mathematics, Volume 155, Issue 2, 2003, Pages 263-284, ISSN 0377-0427, https://doi.org/10.1016/S0377-0427(02)00869-5.

53. P.N. Sun, A. Colagrossi, S. Marrone, A.M. Zhang, “The δ+ -SPH model: Simple procedures for a further improvement of the SPH scheme”, Computer Methods in Applied Mechanics and Engineering 315 (2017) 25–49.

54. M. Antuono, A. Colagrossi, S. Marrone, D. Molteni, “Free-surface flows solved by means of SPH schemes with numerical diffusive terms,” Computer Physics Communications. 2010;181(3):532–549.

55. M. Antuono, S. Marrone, A. Colagrossi, B. Bouscasse, “Energy balance in the δ-SPH scheme”, Computer Methods in Applied Mechanics and Engineering 289 (2015) 209–226.

56. S.J. Lind, R. Xu, P.K. Stansby, B.D. Rogers, “Incompressible smoothed particle hydrodynamics for free-surface flows: A generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves”, Journal of Computational Physics 231(4) (2012) 1499–1523.

57. Sun, P.; Colagrossi, A.; Marrone, S.; Antuono, M.; Zhang, A.M. “A consistent approach to particle shifting in the δ^{+}-SPH model”. Comput. Methods Appl. Mech. Eng. 2019, 348, 912–934.

58. Jandaghian, M.; Shakibaeinia, A. “An enhanced weakly-compressible MPS method for free-surface flows”. Comput. Methods Appl. Mech. Eng. 2020, 360, 112771.

59. Abbas Khayyer , Hitoshi Gotoh, Yuma Shimizu, “Comparative study on accuracy and conservation properties of two particle regularization schemes and proposal of an optimized particle shifting scheme in ISPH context”. Journal of Computational Physics, Volume 332, 2017, Pages 236-256.

60. E.-S. Lee, C. Moulinec, R. Xu, D. Violeau, D. Laurence, P. Stansby, “Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method”, Journal of Computational Physics 227(18) (2008) 8417–8436.

61. E Eun-Sug Lee,“Truly incompressible approach for computing incompressible flow in SPH and comparison with traditional weakly compressible approach”. PhD thesis, University of Manchester, 2007

62. B. Armaly, J. Durst, F. Pereira, J., Schonung, B. “Experimental and theoretical investigation of backward-facing step flow”, Journal of Fluid Mechanics 127 (1983) 473–496

63. F.F. Wang, S.-Q. Wu, S.-L. Zhu, “Numerical simulation of flow separation over a backward-facing step with high Reynolds number”, Water Science and Engineering 12(2) (2019) 145–154.

64. R. Issa, E.S. Lee, D. Violeau, D.R. Laurence, “Incompressible separated flows simulations with the smoothed particle hydrodynamics gridless method”, International Journal for Numerical Methods in Fluids 47(10–11) (2005) 1101–1106.

65. Marrone S, Colagrossi A, Antuono M, Colicchio G, Graziani G, “An accurate SPH modeling of viscous flows around bodies at low and moderate Reynolds numbers”. J Comp Phys (2013) 245:456–475

66. Vacondio R, Rogers BD, Stansby PK, Mignosa P, Feldman J, “Variable resolution for SPH: a dynamic particle coalescing and splitting scheme”. Comput Methods Appl Mech Engrg. (2013) 256:132–148.

67. Braza M, Chassaing P, Ha Minh HH, “Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder”. J Fluid Mech (1986) 165:79.

68. S. Marrone, A. Colagrossi , D. Le Touzé , G. Graziani , “Fast free-surface detection and level-set function definition in SPH solvers”. Journal of Computational Physics, Volume 229, Issue 10, 2010, Pages 3652-3663.

69. M. Doring, Développement d’une méthode SPH pour les applications à surface libre en hydrodynamique, Ph.D. Thesis, Ecole Centrale Nantes, 2005

70. Krimi, A.; Jandaghian, M.; Shakibaeinia, A. “A WCSPH Particle Shifting Strategy for Simulating Violent Free Surface Flows”. Water 2020, 12, 3189.

71. Akbari, H. “An improved particle shifting technique for incompressible smoothed particle hydrodynamics methods.” International Journal for Numerical Methods in Fluids 90 (2019): 603-631.

72. Ping-Ping Wang, Zi-Fei Meng, A-Man Zhang, Fu-Ren Ming, Peng-Nan Sun, “Improved particle shifting technology and optimized free-surface detection

method for free-surface flows in smoothed particle hydrodynamics", Computer Methods in Applied Mechanics and Engineering, Volume 357, 2019, 112580.

73. Xiaoyang Xu, "An improved SPH approach for simulating 3D dam-break flows with breaking waves", Comput. Methods Appl. Mech. Engrg. 311 (2016) 723–742.

74. Z.Q. Zhou, J.O. De Kat, B. Buchner, "A nonlinear 3-D approach to simulate green water dynamics on deck", in: J. Piquet (Ed.), Proc. 7th Int. Conf. Num. Ship Hydrod., Nantes, 1999, pp. 5.1–1, 15.

75. A. Ferrari, M. Dumbser, E.F. Toro, A. Armanini, "A new 3D parallel SPH scheme for free-surface flows", Comput. & Fluids 38 (2009) 1203–1217.

76. A. Colagrossi, M. Landrini, "Numerical simulation of interfacial flows by smoothed particle hydrodynamics", J. Comput. Phys. 191 (2003) 448–475.

77. Muller M., Charypar D., Gross M. "Particle-based fluid simulation for interactive applications". In Proceedings of Eurographics/SIGGRAPH Symposium on Computer Animation (2003), pp. 154–159.

78. Premoze S., Tasdizen T., Bigler J., Lefohn A., Whitaker R. T. "Particle-based simulation of fluids". In Proceedings of EUROGRAPHICS 2003 (2003), vol. 22.

79. Gamito M. N., Lopes P. F., Gomes M. R. "Two-dimensional simulation of gaseous phenomena using vortex particles". In the 6th Eurographics Workshop on Computer Animation and Simulation (1995), vol. 28, Springer-Verlag, pp. 3–15.

80. Sang Il Park and Myoung Jun Kim. "Vortex Fluid for Gaseous Phenomena". In ACM SIGGRAPH/Eurographics Symposium on Computer Animation, July 2005.

81. L. Chiron, M. de Leffe, G. Oger, D. Le Touzé, "Fast and accurate SPH modelling of 3D complex wall boundaries in viscous and non-viscous flows", Computer Physics Communications, Volume 234, 2019, Pages 93-111.

82. Ihmsen, M, Orthmann, J., Solenthaler, B, Kolb, A., & Teschner, M. “SPH Fluids in Computer Graphics”. Eurographics. 2014

83. Ihmsen M., Cornelis J., Solenthaler B., Horvath C., Teschner M. “Implicit incompressible SPH”. IEEE Transactions on Visualization and Computer Graphics (2013). doi:10.1109/TVCG.2013.105.

84. Macklin M., Mueller M. “Position based fluids”. ACM Transactions on Graphics (Proceedings SIGGRAPH) 32 (2013).

85. Solenthaler B., Gross M. “Two-scale particle simulation”. ACM Transactions on Graphics (Proceedings SIGGRAPH) 30, 4 (2011), 72:1–72:8.

86. Solenthaler B., Schläfli J., Pajarola R. “A unified particle model for fluid-solid interactions”. Computer Animation and Virtual Worlds 18, 1 (2007), 69–82.

87. Vacondio, R., Altomare, C., De Leffe, M. et al. “Grand challenges for Smoothed Particle Hydrodynamics numerical schemes”. Comp. Part. Mech. 8, 575–588 (2021). https://doi.org/10.1007/s40571-020-00354-1.

88. Lind Steven J., Rogers Benedict D. and Stansby Peter K. “Review of smoothed particle hydrodynamics: towards converged Lagrangian flow modelling”,2020, Proc. R. Soc. A.476 20190801 20190801 http://doi.org/10.1098/rspa.2019.0801.

89. Crespo A, Gómez-Gesteira M, Dalrymple R (2007), “Boundary conditions generated by dynamic particles in SPH methods”. Comput Mater Continua 5:173–184.

90. Gotoh H, Khayyer A. 2016, “Current achievements and future perspectives for projectionbased particle methods with applications in ocean engineering”. J. Ocean Eng. Mar. Energy 2, 251–278. (doi:10.1007/s40722-016-0049-3).

91. Ye T, Pan D, Huang C, Liu MB ,”Smoothed particle hydrodynamics (SPH) for complex fluid flows: recent developments in methodology and applications”. Phys. Fluids 31, 011301. 2019. (doi:10.1063/1.5068697).

92. Shadloo MS, Oger G, Le Touze DL. “Smoothed particle hydrodynamics method for fluid flows, towards industrial applications: motivations, current state, and challenges”. Comput. Fluids 136, 11–34. 2016

93. Schoenberg, I. J., “Contributions to the problem of approximation of equidistant data by analytic functions”, Q. Appl. Math. 4, 45 (1946).

94. Monaghan, J. J. and Lattanzio, J. C., “A refined particle method for astrophysical problems”, Astron. Astrophys. 149, 135 (1985).

95. R.A. Gingold, J.J. Monaghan, “Smoothed particle hydrodynamics: theory and application to non-spherical stars Mon”. Not. R. Astron. Soc., 181 (1977), pp. 375-389

96. J.J. Monaghan, “Smoothed particle hydrodynamics”, Ann. Rev. Astron. Astrophys., 30 (1992), pp. 543-574.

97. Daniel J. Price, “Smoothed particle hydrodynamics and magnetohydrodynamics”, Journal of Computational Physics, Volume 231, Issue 3, 2012, Pages 759-794.

98. Wendland, H. “Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree”. Adv Comput Math 4, 389–396 (1995).

99. Wendland H., 2005, “Scattered Data Approximation”. Cambridge University Press, Cambridge, UK.

100. Walter Dehnen, Hossam Aly, “Improving convergence in smoothed particle hydrodynamics simulations without pairing instability”, Monthly Notices of the Royal Astronomical Society, Volume 425, Issue 2, 9/2012, Pages 1068–1082.

101. Thomas Douillet-Grellier, Florian De Vuyst, Henri Calandra, Philippe Ricoux, “Simulations of intermittent two-phase flows in pipes using smoothed particle hydrodynamics”, Computers & Fluids, Volume 177, 2018, Pages 101-122, ISSN 0045-7930, https://doi.org/10.1016/j.compfluid.2018.10.004.

102. Guerrero, J.. “Numerical simulation of the unsteady aerodynamics of flapping flight”. Department of Civil, Environmental, Architectural Engineering 2009.

Research Achievements

Journal Papers

1. **Bui Tien Thanh**., Nakata Susumu. (2021), “Nonreflecting Outlet Boundary Conditions for Smoothed Particle Hydrodynamics Simulation of Small-Scale Open-Channel Flow”. In: Atluri S.N., Vušanović I. (eds) Computational and Experimental Simulations in Engineering. ICCES 2021. Mechanisms and Machine Science, vol 97. Springer, Cham. https://doi.org/10.1007/978-3-030-64690-5_7

2. **Thanh Tien Bui**, Susumu Nakata. “Improved boundary treatment for the smoothed particle hydrodynamics method using implicit surfaces’. Mathematical Methods in the Applied Sciences Math. Mech. Appl. Sci. 2020, 1-21, 02/2020, 10.1002/mma.6314.

3. **Thanh T. Bui**, Yoshihisa Fujita, Susumu Nakata, "A simplified approach of open boundary conditions for the smoothed particle hydrodynamics method", submitted to Computer Modeling in Engineering & Sciences (2021) (***under review***)

International Conferences

1. **Thanh T. Bui**, Susumu Nakata, “Non-reflecting outlet boundary conditions for smoothed particle hydrodynamics simulation of small-scale open-channel flow”, 26th International Conference on Computational and Experimental Engineering and Sciences (ICCES), Thailand, 2020/2021. (***The best student paper award***)

2. **Bui Tien Thanh** and Susumu Nakata, "Boundary treatment for smoothed particle hydrodynamics method with implicit surfaces", The 2019 International Conference on Computational & Mathematical Methods in Science & Engineering (CMMSE), Cádiz, Spain (2019).