Design of Low Residue Packs by Smoothed Particle Hydrodynamics

A thesis is submitted to The University of Manchester for the degree of Doctor of Philosophy in Mechanical Engineering to the Faculty of Science and Engineering

2020

Aaron R. English

School of Engineering
Department of Mechanical, Aerospace and Civil Engineering

Contents

Word count: 33134

List of Figures

List of Tables

Nomenclature

Roman Symbols

$\mathcal{C}$	Whalley Flooding Coefficient	
c_0	Numerical Speed of Sound	m/s
D	Bottle Diameter	m
$\mathbf{f}$	Acceleration Due To Body Forces	m/s
dp	Particle Spacing	m
$\mathbf{g}$	Acceleration Due To Gravity	m/s
h	Smoothing Length	m
K	Consistency Index	Pas^n
m	Mass	kg
M	Papanastiou Exponential Growth Coefficient	
n	Power Law Index	
$\mathbf{n}$	Unit Normal Vector	m/s
Q	Flow Rate	m^3/s
$\mathbf{t}$	Unit Tangent Vector	m/s
t	Time	s
Δt	Timestep	s
$\mathbf{u}$	Particle Velocity	m/s
$\mathbf{u}_{slip}$	Partial Slip Velocity	m/s

V	Volume	m^3
W	Kernel Function	
$\mathbf{x}$	Particle Position	m

Particle Subscripts

b	Boundary Particle
f	Fluid Particle
g	Ghost node
i	Central Particle
j	Neighbour Particle
w	Wall

Dimensionless Numbers

Ca	Capillary Number
Fr	Froude Number
Re	Reynolds Number
We	Weber Number

Greek Symbols

α	Artificial Viscosity Coefficient	
$\delta(\mathbf{x})$	Dirac δ Function	
δ	Density Diffusion Parameter	
γ	Tait Equation Index	
$\dot{\gamma}$	Shear Rate	$1/s$
$\dot{\boldsymbol{\gamma}}$	Shear Rate Tensor	$1/s$
λ_s	Slip Length	m
κ	Friction Coefficient	

λ_s	Slip Length	m
μ	Dynamic Viscosity	Pas
ν	Kinematic Viscosity	m^2/s
ρ	Density	kg/m^3
ρ_0	Reference Density	kg/m^3
τ	Shear Stress	N/m^2
τ_y	Yield Stress	N/m^2
$\boldsymbol{\tau}$	Shear Stress Tensor	N/m^2

Acronyms

BC	Boundary Conditions
DBC	Dynamic Boundary Condition
mDBC	Modified Dynamic Boundary Condition
MLS	Moving Least Squares Method
MPS	Moving Particles Semi-Implicit Method
PIV	Particle Image Velocimetry
SPH	Smoothed Particle Hydrodynamics
WCSPH	Weakly Compressible Smoothed Particle Hydrodynamics

List of Published Works

The work presented in this thesis has led to the following publications:

In peer-reviewed journals:
English A, Dominguez JM, Vacondio R, Crespo AJC, Stansby PK, Lind SJ, Chiapponi L, Gomez-Gesteira M. 2020. Modified dynamic boundary conditions (mDBC) for general purpose smoothed particle hydrodynamics (SPH): application to tank sloshing, dam break and fish pass problems. To be submitted to the special issue on "Latest Developments and Application of SPH using DualSPHysics" in Computational Particle Mechanics.

Conference Papers:
English A, Stansby PK, Lind SJ. 2018. Introduction of partial slip boundary conditions to Smoothed Particle Hydrodynamics. In: Proceedings of the 13th International SPHERIC Workshop, Galway, Ireland

English A, Domínguez JM, Vacondio R, Crespo AJC, Stansby PK, Lind SJ, Gómez-Gesteira M. 2019. Correction for Dynamic Boundary Conditions. In: Proceedings of the 14th International SPHERIC Workshop, Exeter, United Kingdom

English A, Stansby PK, Lind SJ. 2020. Partial Slip Boundary Conditions in SPH. In: Proceedings of the 2020 SPHERIC Harbin International Workshop, Harbin, China.

Abstract

Design of Low Residue Packs by Smoothed Particle Hydrodynamics

A thesis submitted for the degree of Doctor of Philosophy at the University of Manchester
By Aaron English
2020

Many Unilever products, principally gels and sauces, leave a residue in the container when ‘empty’ due to the interaction of the product with the interior packaging surface. This is a free surface and boundary problem. Smoothed particle hydrodynamics (SPH) is well suited for free surface flows in complex geometries but accurate representation of solid surfaces requires fundamental development. A robust but approximate boundary condition for general complex shapes, known as the dynamic boundary condition, is widely used but has underlying physical and theoretical weaknesses, limiting accuracy. This has been improved in this project by extrapolating properties from the flow domain, to fixed boundary particles to maintain the robustness of the dynamic boundary condition while removing its physical and theoretical limitations; this has become known as the modified dynamic boundary condition. However conventional boundary conditions are no slip but many flow problems in the process industries incur partial slip, with a small flow velocity on the solid surface. Formulations in SPH have been developed for the first time which has been possible with the ‘modified’ dynamic boundary condition. This has been progressed with application to parallel flows, Couette and Poiseuille, for which new analytical solutions have been developed to enable validation. The important rheometer case, used for quantifying particularly non-Newtonian fluid properties, has also been studied, comparing with limited available experimental data. The groundwork for generalisation to complex shapes, including no slip and partial slip, has thus been laid. The aim of a versatile simulation tool with free surfaces has been demonstrated for diverse applications, the standard dam break with violent impact on a column and the new fish pass problem comprising a complex channel flow with transverse weirs and gates. New experimental data for pouring from a beaker has been simulated but have shown that limitations for small flow depths of the order of particle size need to be addressed when the beaker is almost empty. The modified dynamic boundary condition with density extrapolation and zero velocity solid boundary particles (or with velocity of the body if moving) is now available in the open source DualSPHysics code which can run efficiently on CPUs and GPUs.

Declaration

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

Copyright Statement

Acknowledgement

Firstly, thanks to my supervisors Prof Peter Stansby and Dr Steven Lind for all you support and advice during this PhD, especially through the final months during the Covid lock down.

Secondly, I would like to thank my parents and brother and sister for your support and encouragement, even if you didn't always understand what I was talking about and just liked the pretty pictures.

To all the PhD students, postdocs and academics of the Manchester SPH group for helping to create a welcoming and friendly environment to develop ideas and solve current problems. And for coming to the pub every Thursday.

Special thanks to Dr Annelie Baines for all you helpful discussions and sanity checks over countless cups of coffee, and for putting up with all my incredible puns. And to Dr George Fourtakas for all you help and never being too busy to sit down and go through any code or idea I was struggling with right from the start.

To Jose, Renato, Alex, Moncho and Luca for your help with developing MDBC and putting up with all my terribly written code.

Thanks also to Adam, Waldo, Haus, Olga and Saarah from Unilever R&D for you help getting set up with equipment and test fluids for the experiments at Port Sunlight.

This project was funded by a studentship from the EPSRC in partnership with Unilever. Thanks also to Unilever for producing PG Tips that kept me going.

Chapter 1

Introduction

1.1 Motivation

Unilever (2020a) is a large British-Dutch company that produces over 400 brands of products including: personal care products (Lynx, Dove, Tresemmé); detergents (Comfort, Surf); ice creams (Ben & Jerry's, Wall's), and sauces (Coleman's, Hellmann's). It is estimated that seven in every ten homes worldwide contains a product produced by Unilever, and globally their products are used by over two billion people each day. Closer to home it is estimated that 98% of British homes use a Unilever product.

With such a large range of popular products available, used by so many consumers, comes a lot of packaging and potential for waste product to be left inside the packaging after it has been thrown away. This is a problem as consumers will be less likely to buy a product if they think a large amount of what they have bought will be throw away. To tackle this Unilever have been coming up with ways to reduce the waste left inside packaging. For example Hellmann's Mayonnaise came up with a new squeezable bottle design that gets more Mayo out with a solution that "doesn't involve whacking it, slicing the top off, or scraping the remnants out with a spoon" (Unilever (2015)). It's estimated that this new design reduced the amount of leftover mayo in a bottle from 13% to 3%.

By 2025 Unilever aim to ensure that all their plastic packaging will be reusable, recyclable or decomposable and that 25% of the plastic used will come from recycled plastic content. They recently added a further aim to halve the amount of plastic used in their packaging. Unilever consumer research has found that a third of consumers are now buying brands based on their social and environmental impact, and over 50% of consumers are more likely to buy a product that is sustainably produced (Unilever (2020b)). To this aim they have started testing reusable packaging for selected brands here in the UK. It is important therefore for Unilever to know how much product will be left inside this packaging, and packaging in general, and if there are way to reduce the amount.

What would be useful is a tool that can be used to investigate if a new pack design will be better than the previous in terms of filling rates and through the use of the product to reduce residue. Residue left inside packs is the amount of product left inside when the pack is thrown away. In order to reduce the amount of residue three options were identified as possible solutions: sacrificial layers; surface treatments, and the redesign of the fluid in the pack. Computational tools can assist in assessing the efficiency of each of these approaches.

- The idea of a sacrificial layer is that a small amount of product will be left inside the pack when it is thrown away, this is what currently happens. This could be improved upon by finding exactly how much fluid is left inside a pack at the end of the packs life and trying to minimise it by modifying the shapes of the packs. Ng et al. (2013) showed how this could be done using a moving particle semi-implicit method. This approach however could prove difficult for non-Newtonian fluids that may stick to a surface no matter the shape.

- Redesigning the fluids inside would have the hope of allowing the fluid to more freely flow out of the pack and therefore reduce the amount left behind. A computational tool could predict flows of new products in advance if they are well characterised and make suggestions to viscosity for example to improve flow rate. However the design of the products is more complex than just changing the viscosity. Consumer opinion on attributes such as taste or smell or how well it creates bubbles also effects the design of a fluid. These are a beyond the scope of this thesis and more suited to a chemical science project.

- Surface treatments would involve changing the material of the packs, or treating them in some way, in order to allow the fluid to more easily flow over them. This option would involve building a working model for the current packs and then using different boundary conditions to try and reduce the amount of residue. At a macro-level this requires the use of a general slip boundary condition, known as partial slip. An example of a technique that already works is that of LiquiGlide (2020) in which the inside of bottles are given a special slip coating that allows fluid to flow out the bottle with ease.

The outcomes of this research will be valuable for diverse applications, e.g. industrial strainers in the chemical, petroleum and pharmaceutical industries, filtering or straining out unwanted impurities from fluids present in products like medical syrups, and in blood flow and nano-fluid applications.

1.2 Project Aim

The aim of this project is to deliver a toolbox for the virtual design and evaluation of product-pack combinations of Newtonian and Non-Newtonian fluids. This toolbox should be able to recreate current packaging designs incorporating the physical parameters of the fluid of interest. The designs may be modified in order to increase the flow out of the packaging in ways that would not be possible in real world manufacture due to time and cost restraints. The best way to deliver this toolbox will be with a computational fluid dynamics (CFD) method. The CFD method chosen should be capable of modelling all of the different physical elements of the problem. Classical grid based CFD methods have been used in the past to simulate the flow out of bottles and containers, for example Geiger et al. (2012), Mer et al. (2018), Chihara et al. (2009) and Nishio et al. (2019). In each of these examples a two phase model as required by the mixing of liquid and gas. However, there are some examples of flows where the air and water phases do not mix and a single phase approach can be uses instead. Examples of this are found in Sun et al. (2011) and Ng et al. (2013) who both use the particle based moving particle semi-implicit method. In order to simulate these kinds of problems with a grid based method would require complex and expensive re-meshing or the volume of fluid (VOF) approach with a fixed mesh and an air phase which would increase the computational cost. A Lagrangian based approach avoid these problems.

The smoothed particle hydrodynamics (SPH) method would be a good candidate due to its ability to model free surface flows such as in Gomez-Gesteira et al. (2009) or Violeau and Rogers (2016). The method has also be used in the past to model different physics that could be useful such as deformable boundaries in Nasar et al. (2016), and non-Newtonian flows in Fourtakas and Rogers (2016) and Xenakis et al. (2015). SPH is a Lagrangian particle based method which models a fluid flow by following particles that are free to move around the entire domain. This ability to track particle movement is very attractive as it would give good visualization of fluid flow out of packaging. SPH also conserves mass exactly as each particle carries a constant mass. This conservation of mass would give a good description of the amount of product left after use, simply by counting particles. SPH will be the method used in the toolbox and will be based on the open source SPH solver

DualSPHysics (Crespo et al. (2015)) due to its parallel written code and ability to run on GPUs allowing for fast calculations. Further justification for the use of SPH will be provided in Chapter 3.

1.2.1 Objectives

The main objective of this work is to present a SPH tool based on DualSPHysics that is capable of modelling the emptying of packaging and containers. This tool should then be able to test new and modified designs to investigate the effect of the residue left behind.

Of the three avenues identified above, surface treatments are the most suitable option as it aims to reduce the amount of residue without incorporating a sacrificial layer that would be wasteful and adds cost. It also allows for changes to the design of packaging to aid in the flow of the contained product. This approach would involve exploring partial slip boundary conditions and their implementation in the SPH method in order to improve current packaging designs.

The objectives of this thesis are:

- Assess current approaches to bottle and package emptying
- Review present partial slip boundary conditions and their application
- Assess available boundary conditions for SPH
- Develop boundary conditions for SPH to enable accurate implementation of no slip, slip and partial slip boundary conditions
- Apply new boundary conditions to various applications including standard SPH applications (dam break, sloshing tank) and pouring flows
- Make the new formulations available for general use with complex geometries through the open source code DualSPHysics

1.3 Layout of this Thesis

The thesis will be organised as follows:

Chapter 2 will review the literature on: the experimental and numerical work on emptying bottles; partial slip boundary conditions and the slip length, and Non-

Newtonian fluid models and their uses for fluids such as food stuffs and body care essentials.

Chapter 3 covers the methodology of the SPH method, how it is applied to fluid dynamics and how it will be used in this project.

Chapter 4 will introduce two new boundary conditions. These new conditions will be validated using analytical solutions.

Chapter 5 includes results of pouring experiments conducted to provide data to help validate the SPH model.

Chapter 6 will present the use of the new boundary conditions for complex test cases including pouring with comparisons with available experimental results.

Chapter 7 concludes the findings and results of the thesis.

The thesis also includes some useful results as appendices and a collection of references used to support this work.

Chapter 2

Literature Review

2.1 Introduction

The focus of this project is to look at modelling packaging and containers and the processes of emptying and filling of said containers in order to reduce the amount of waste produced. Of the three suggested courses of action: sacrificial layers; surface treatments, and redesign of fluids, applying a surface treatment to the inside of the containers to increase the flow rate is the approach of choice for this thesis. Before a surface treatment can be applied what first needs to be understood is the current state of play in the area of packaging and numerical simulation. The current advances in trying to reduce the amount of waste in packaging also needs to be investigated.

In order to do this, this Chapter will review the literature in these areas. First experimental and numerical studies of the emptying and filling of bottles will be investigated, this will then lead to the use of partial slip boundary conditions to increase flow rate. Finally a brief review of Non-Newtonian fluid models will be presented along with previous uses to model fluids of interest to this project.

2.2 Emptying of Containers

Much of the early research looking into the emptying of containers looked at measuring the emptying time and what changes could be made to reduce this. Many authors also looked at the bubbles that often appear when emptying closed containers and the effects these have on the flow. Many of the earlier papers were experimental in approach with more numerical and CFD related papers appearing more recently.

2.2.1 Experimental methods

Whalley (Whalley (1987), Whalley (1991)) conducted experiments measuring the filling and emptying times of bottles of different shapes and sizes. The experiments consisted of submerging bottle in tanks of water and measuring the time take to fill, and emptying the same bottle by holding them upside down and letting the water drain into the tank and again measuring the time taken. These times were used to calculate a dimensionless number $\mathcal{C}$ called the flooding coefficient. The flooding coefficient is given by

$$\mathcal{C} = \frac{\left(\rho_g^{1/4} + \rho_f^{1/4}\right)}{\left[(\rho_g - \rho_f)gD\right]^{1/4}} \left(\frac{4V}{\pi D^2 t}\right)^{1/2}, \tag{2.1}$$

where ρ_g and ρ_f are the gas and fluid densities, D is the internal diameter of the bottle neck, g is the acceleration due to gravity, V is the bottle volume and t is the emptying or filling time. The flooding coefficient was used to find a link between ”flooding” and ”slugging” as the bottles were filled and emptied. It is interesting to note that the flooding coefficient can be re-arranged to contain the Froude number Fr by writing it as

$$\mathcal{C} = \frac{\left(\rho_g^{1/4} + \rho_f^{1/4}\right)}{\left[(\rho_g - \rho_f)\right]^{1/4}} \left(\frac{4Fr}{\pi D^2 t}\right)^{1/2} \tag{2.2}$$

The Froude number defined as

$$Fr = \frac{U}{\sqrt{gL}} \tag{2.3}$$

for a length scale L and velocity U is the ratio of inertial to gravitational forces. Larger Froude numbers, $Fr > 1$ indicates fast moving or super-critical flow, and small Froude numbers, $Fr < 1$ slower moving or sub-critical flows. A larger Froude number leads to a larger flooding coefficient.

When a bottle is emptied or filled it can experience three states. The first is counter-current flow where there is a continuous exchange of gas and fluid and the two do not mix, this would occur when pouring a bottle of water and there is a gap between the water surface and the rim of the neck to allow air in. Flooding is the limit of counter-current flow (Whalley (1987)), this occurs when the bottle is tipped further and there is no longer a gap between the water surface and the bottle rim. Air bubbles into the bottle but there is a continuous flow of water out of the bottle. Slugging occurs when the bottle is tipped further still, close to upside down. At this point large air bubbles enter the bottle and as they do the water flow out is stopped until the bubble passes the neck. These bubbles give the familiar glugging sound as they enter the bottle (Clanet and Searby (2004)).

When emptying a bottle, counter-current flow and flooding will be much faster than slugging as in these cases the fluid in the bottle is always exiting the bottle. However, when a bottle starts slugging, the flow out of the bottle is stopped when the air bubble enters stopping the overall emptying process. This can be useful to know when washing milk bottles or drinks bottles for example, as emptying bottles will be faster if they do not enter the slugging phase. It is likely possible to transfer from slugging to counter-current flows in some situations by shaking and spinning the bottle to form a small whirlpool type effect inside the bottle. When the bottle is held upside down and the bottom of the whirlpool reaches the bottleneck air is sucked up the middle and counter-current flow is achieved.

Whalley (1991) found that bottle with larger values of $\mathcal{C}$ are less likely to flood and therefore have shorter emptying times, this can be seen in Equation (2.1) where a smaller value of t would lead to a larger value of $\mathcal{C}$. This is consistent with respect to the Froude number, as the larger flooding coefficient and therefore Froude number indicates that the fluid is flowing faster out of the bottle. Whalley looked at the effects of bottle inclination and extensions to the neck of the bottles on the emptying and filling times, and found that inclining the bottle decreased the filling times of the bottles. Whalley concluded that the empting and filling of bottles was controlled by flooding when done in the manner of these experiments.

Tehrani et al. (1992), Schmidt and Kubie (1995) and Clanet and Searby (2004) all looked at the multiphase flow of an idealised bottle being emptied. An idealised bottle is considered to be a closed cylinder of liquid with a hole at the bottom for the fluid to drain through. Schmidt and Kubie (1995) looked at the emptying times of bottles of different diameters and heights as well as different diameter drainage holes. They noted that glugging happened but only when the water level in the tank was a sufficient distance from the top and bottom of the bottle. When the water level was close to the top of the cylinder glugging did not occur as the space was too small for it to develop. Once the water level got close enough to the bottom of the cylinder the outlet became "self-venting". They found that the average discharge velocity was independent of the water depth and height of cylinder but increased with the diameters of both the cylinder or the outlet.

Tehrani et al. (1992) measured pressure oscillations in the air inside a partially filled sealed tank as water was drained through a vertical tube from the centre of the bottom of the tank. They described the flow as a repeated cycle of air and water exchange in four steps

1. liquid from the tank falls down the tube lowering the pressure inside the tank

2. once the pressure in the tank drops low enough the water flow stops and a bubble forms at the bottom of the tube

3. the bubble rises up the tube and enters the tank which re-pressurizes back to atmospheric pressure

4. the water in the tank falls back into and down the tube

The experimental results were compared to a theoretical model developed by solving the governing PDEs of the above four stages using a fourth order Runge-Kutta scheme. In the first step the PDEs were solved for the velocity of the fluid and the pressure in the tank, once the fluid velocity reached a critical value the theoretical models solves for the velocity of the rising bubble. Once the bubble reached the top of the tube the method changed to solve for the change in pressure in the tank followed by the velocity of the liquid falling back into the tube. The process then repeats itself. A comparison of the experimental to the theoretical results showed good agreement for the pressure vs time inside the tank. All the changes in pressure are captured in the experimental and theoretical results. The authors found that the period of the cycle of air and water exchange increased during the emptying process, as the water level in the tank dropped, due to the larger air volume in the tank allowing for more water to drain before the pressure drop formed a new bubble. The number of cycles required to drain the tank did not change when different diameter tubes were used for the drainage. The long bubbles that formed in these experiments are examples of a slug flow.

Clanet and Searby (2004) looked at both the drainage time and period of pressure oscillations in idealized bottles of different heights, diameters and with different diameter openings at the bottom. The results found for the drainage time and oscillations agreed with the results of the previous two works. They also tested fluid with different viscosities and found that the period of oscillations was not sensitive to the fluids viscosity. However, when testing a fluid other than air for the surrounding fluid they found that if the second phase is incompressible that glugging did not occur. This was done by creating a liquid egg timer with an upper cylinder filled with water and a bottom cylinder filled with a less dense oil with a hole allowing flow between the two cylinders. When the water was allowed to flow through the hole a continuous exchange of oil and water occurred without any pressure oscillations, a counter-current flow. When the experiment was repeated with the same set up but with air in place of the oil the glugging again occurred. They concluded that the glugging was due to the compressibility of the air.

2.2.2 Numerical methods

Geiger et al. (2012) used a volume of fluid method using the code OpenFOAM to model the drainage of bottle and measure the time to fully empty. They conducted some experiments measuring the emptying time of bottle similar to those of Whalley (1987) and compared these results to the numerical model for validation. Then simulation results were then also compared to the results fo Whalley (1991) and Clanet and Searby (2004). The emptying times of the numerical bottle matches the experimental times very well for both of the bottle shapes considered. This was true for both bottle held upside down and inclined at a 45^o angle. The pressures measured in the bottles also matched the experiment reasonably well. The numerical model was used to look at the effects of increasing the diameter of the bottle neck and different angles of inclination on the emptying times. It was found that increasing the bottle neck diameter helps to decrease the bottle filling time. When the neck diameter was in the region of $10 - 20mm$ the glugging oscillations were observed, but increasing the diameter towards $30mm$ leads to a counter-current flow, further decreasing the emptying time. It was also found that inclining the bottles decreased the emptying times of the smaller diameter bottles as the glugging flow transferred to a counter-current flow, however the emptying time of the larger diameter bottle increased when inclined due to the flow starting to glug. They concluded that CFD can be a useful optimisation tool for the bottle emptying process. This can lead to reduced emptying times for applications where this is of importance such as bottle washing.

Mer et al. (2018) used the emptying of ideal bottles as a test case to compared three bubbly flow models in the finite volume code NEPTUNE_CFD. The three bubble models investigated were the Large Interface Model (LIM), the Generalized Large Interface Model (GLIM) and the Generalized Two-Phase Flow Model (GENTOP) referred to in the paper as the Large Bubble Model. The results of the computational models were compared to experimental results of the drainage of idealised bottles using a similar design of that of Clanet and Searby (2004) and Schmidt and Kubie (1995). When the air phase was modelled as a compressible fluid they found that pressure oscillations measured in the bottle matched the experimental data well. This was true of all three models compared. When comparing the shape of the bubble formed in the simulation to photos of the experiment they found that the LIM approach failed to predict the breakup of the bubbles. The other two methods predicted this much closer and performed comparably well to each other. Either of these two methods would then be good to use for these types of bubbly flow.

While designing a new aluminium bottle to be used as a plastic alternative in Japan, Chihara et al. (2009) looked at consumer satisfaction levels when drinking from bottles with different neck diameters. They surveyed 120 adults on the ease fo drinking from bottles with three different neck diameters. After identifying the preferred neck diameter of $33mm$ they used bottle inclination angles to develop a CFD model of the bottles. The inclination angle was measured as the angle a consumer hold the bottle when drinking for each the bottle sizes. Using a finite element based volume of fluid they measured the flow rates out of bottles for a range of inclination angles including those measured from consumers. This was repeated for each size of bottle. The average flow rates were then compared to the survey data with an aim of finding a correlation between average flow rate and consumer satisfaction. They found that when drinking from a bottle with a less favourable neck diameter the consumer will drink using a different angle so to better match the flow rate experienced from the more favourable. When drinking from a bottle with a $38mm$ neck the consumer would tilt the bottle less so the average flow rate was reduced to a value closer to that of the $33mm$ at a larger angle. It was concluded that average flow rate out of a bottle can be used to indicate consumer satisfaction when drinking from the bottle.

Sun et al. (2011) simulated pouring from a $2D$ cup using a single phase moving particle semi-implicit (MPS) method. They started by modifying the boundary condition to a SPH style approach to improve their no-slip boundary velocity. The new method involved treating the first layer of boundary particles as fluid in the particle interactions, the velocity and position however was controlled by the movement of the boundary. The MPS code using then new boundary condition was validated using a dam break simulation which showed good agreement to experimental data for the motion of the leading edge of the flow. They noted that the new boundary condition helped give a closer result than standard MPS. The model was then used to simulate the pouring of liquid from a $2D$ cup with width $120mm$ and height $190mm$ filled to a depth of $160mm$ tilted to an angle of 90^o. Taking advantage of the particle nature of the MPS method, fluid particles were given number in the order they left the cup and flowed over the rim. This order was then used to colour the particles. Returning the particles to their original position but retaining the colouring bands of colour can be seen that indicate the order that fluid leaves the cup. The colour bands showed that the fluid close to the lip of the cup flowed in U shaped bands and that the last fluid particles to leave the cup were those in the corner of the cup underneath the pouring lip. The mass flow rate out of the cup was measured using $\eta = N_{out}/N_{all}$ where N is the number particles total (all) or that have been poured out of the cup (out). Parameters such as the position of the centre of rotation, viscosity and rotation speed were changed to see any difference in the mass flow rate. They found that moving the centre of rotation up or down did not change the mass flow rate but moving the centre of rotation closer to the pouring lip increases the mass flow rate. Also more fluid is able to pour out of the cup at any angle when a lower rotation speed is used.

Ng et al. (2013) also used the MPS method, this time to simulate the draining of open topped tanks. Since the top of the tanks were open to the air, the second air phase did not need to be modelled and single phase model could be used instead. They measured the draining time of $2D$ tanks of water with different shaped bottoms, the aim to see which if these shapes allowed for faster drainage. The shapes of the tanks were defined using the equation

$$S_1(h) = \pm 0.5(\alpha h^n + S_2) \tag{2.4}$$

for $n = 0, 0.5, 1, 2$. S_1 is the width of the tank for given height h for $0 < h < 0.1$ and $S_2 = 0.2$ is the width of the outlet. The α parameter is a scaling term that ensures each tank holds the same volume. They found that the tanks with the larger value of n drain much faster than the tanks with the lower values. Plotting the velocity vectors on the particles, after a short time they found that for larger value of n all the fluid particles close to the outlet were moving downwards with the bulk of the flow. However, for smaller values of n the fluid particles close to the outlet, but not directly above it, were pointing more towards the centre of the tank and the particles had a lower magnitude velocity. Taking advantage of the particle nature of the method, and colouring the particles by the order they left the outlet, it was shown that for lower values of n U shaped bands of particle appeared spreading out from the outlet. For $n = 0$ this meant that particles close to the bottom corners of the tank were some of the last particles that would drain out. As the values of n increases the U shaped bands flatten out showing that for these types of tanks the fluid drained in layers from the bottom of the tank to the top. The authors concluded that this result could be useful for industry applications where drainage times are big factors to production and cost, such as drainage of chemical tanks.

Nishio et al. (2019) used a volume of fluid approach in STAR-CCM+ to simulate fluid pouring out a drinks can. They simulated a can filled with water rotating at a constant angular velocity. Numerical results for the free surface can then be visually compared to photos of experiments of the same set up. The comparison showed good agreement between the CFD model and the experiment. Aluminium powder was suspended in the water of the experiment to show particle movement, with longer paths of aluminium particles indicating faster moving regions of fluid. This was shown to be similar to the velocity field of the fluid in the computational result. The paper then moves on to look at the effect of condensation forming on the outside of the can on the contact angle and wettability of the surface of the can. They found that condensation increases the contact angle and therefore the wettability allowing fluid to spread across the surface easier. Repeating the numerical simulation using the larger contact angle for the condensation covered can they found the flow would dribble down the side of the can when poured. The flow out of the dry can would detach from the can as it was being poured with no dribble present. This was again compared to photos the experimental pour of a condensation covered can showing a good visual agreement.

2.2.3 Increasing flow rate out of bottles

All of the work so far has looked at studying how fluid leaves a container and how changes to the boundary shape effects the flow. However there is a way to leave the container shape unaltered and instead apply a slippery coating to the inside that allows for more fluid to flow out. An example is the method of LiquiGlide (2020) which aims to get more fluid out of a bottle. The inside of bottles are coated in a textured hydrophobic (water hating) material which is then covered in a hydrophobic fluid which stick together. This combined surface repels the product placed in side the bottle allowing it to flow out more easily. This method has been tested for a number of different fluids including ketchup, mayonnaise, laundry detergent and paint with each showing a large decrease in the amount of residue left inside the bottle. The process is based on the work of Smith et al. (2013) and Anand et al. (2012) who created a method of coating a boundary in a hydrophobic material that increases surface slip.

For a flow over this kind of surface the no-slip boundary condition no longer applies due to the small amount of fluid slip at the boundary surface. A partial slip boundary condition is needed instead.

2.3 Partial Slip Boundary Conditions

The classic boundary condition in CFD is the no-slip and no-penetration Dirichlet type boundary condition. It states that the fluid next to a solid boundary cannot slip along the boundary and cannot enter the boundary. Fluid at the boundary must therefore satisfy the condition

$$\mathbf{u} \cdot \mathbf{t} = \mathbf{0} \qquad \mathbf{u} \cdot \mathbf{n} = 0, \tag{2.5}$$

where $\mathbf{n}$ is the unit normal to the boundary pointing into the fluid, and $\mathbf{t}$ is a unit vector tangent to the boundary.

In a flow with partial slip boundary conditions, the fluid near the boundary also satisfies a Dirichlet condition where the fluid is allowed to move with a small velocity called the partial slip velocity $\mathbf{u}_{slip}$. The fluid also obeys the no-penetration condition so no fluid can enter the boundary. Fluid near a partial slip boundary therefore satisfies

$$\mathbf{u} \cdot \mathbf{t} = \mathbf{u}_{slip} \qquad \mathbf{u} \cdot \mathbf{n} = 0, \tag{2.6}$$

where the partial slip velocity is given by the equation

$$\mathbf{u}_{slip} = \lambda_s [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{n}, \tag{2.7}$$

where the quantity λ_s is called the slip length (Lauga et al. (2005)). Equation (2.7) comes from Navier (1823), and later Maxwell (1879), as 'the velocity tangent to the surface is proportional to the rate of strain at the surface' (Lauga et al. (2005)). Lauga et al also suggest that 'Alternatively, the right-hand side of this boundary condition, when multiplied by the shear viscosity of the liquid, states that the tangential component of the surface velocity is proportional to the surface shear stress'.The slip length can be thought of as the depth into the boundary at which no-slip is attained if the velocity profile is linearly extrapolated into the boundary. A graphical representation of the slip length can be seen in Figure 2.1. Setting the slip length to zero result in the partial slip velocity becoming zero and the no-slip condition is recovered.

2.3.1 Effects of Partial Slip

An easy way to see the effect of partial slip is to look at the Poiseuille flow. Ferrás et al. (2012) derived analytical solutions for Poiseuille and Couette flow for Newtonian and non-Newtonian fluids with partial slip boundary conditions. For a Poiseuille flow driven by a body force F in the x direction defined between two plates at $z = \pm l$ and kinematic viscosity ν the no-slip velocity profile is given by

$$u(z) = \frac{F}{2\nu}\left(l^2 - z^2\right). \tag{2.8}$$

The partial slip version is given by

$$u(z) = \frac{F}{2\nu}\left(l^2 + 2\lambda_s l - z^2\right), \tag{2.9}$$

where λ_s is the slip length. Clearly the addition of partial slip at the boundaries increases the velocity of the flow, this can be seen in Figure 2.2 where examples of different sized slip lengths can be seen. It can also be seen that setting the slip length to zero recovers the no-slip result.

2.3.2 Examples of Partial Slip Flows

There have been a number of studies of partial slip flows involving hydrophobic surfaces, for example Lauga and Stone (2003). A surface is defined to be hydrophobic if the contact angle, θ, between the surface and the leading edge of a water droplet on the surface is greater than 90^o. The contact angle can be described as a function of the surface tensions γ between the solid (S), liquid (L) and gas (G) phases present through Young's equation (Bonn et al. (2009))

$$\cos(\theta) = \frac{\gamma_{SG} - \gamma_{SL}}{\gamma_{LG}}. \tag{2.10}$$

The larger the contact angle between the solid and fluid phases, the more hydrophobic the surface is. This then lowers the friction between the solid and liquid phases (Martines et al. (2005)). If the contact angle becomes large enough, the water droplets form balls that roll across the solid surface when tilted through even small inclinations. This phenomenon can be seen in nature on the leaves of the Lotus plant for example. This is known, aptly, as the Lotus effect and allows the plant to self clean. As the balls of water roll over the surface of the leaf they pick up dirt and dust and remove them along their way (Ma and Hill (2006)).

Anand et al. (2012) and Smith et al. (2013) developed a hydrophobic surface that was shown to be very effective at allowing fluid to flow across the surface. The surface was formed of a rough textured hydrophobic surface that was submerged in a hydrophobic lubricating fluid layer. The lubricating layer makes the overall surface hydrophobic, and as both the surface material and lubricant are hydrophobic they stick to each other. The surface is also said be self-healing as any scratches formed in the surface are filled in by the lubricating layer. This technique of constructing

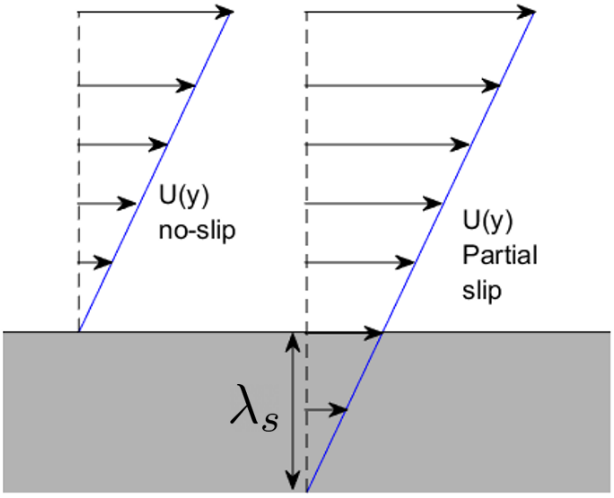


Figure 2.1: Graphical representation of the slip length λ_s for a two dimensional linear flow field.

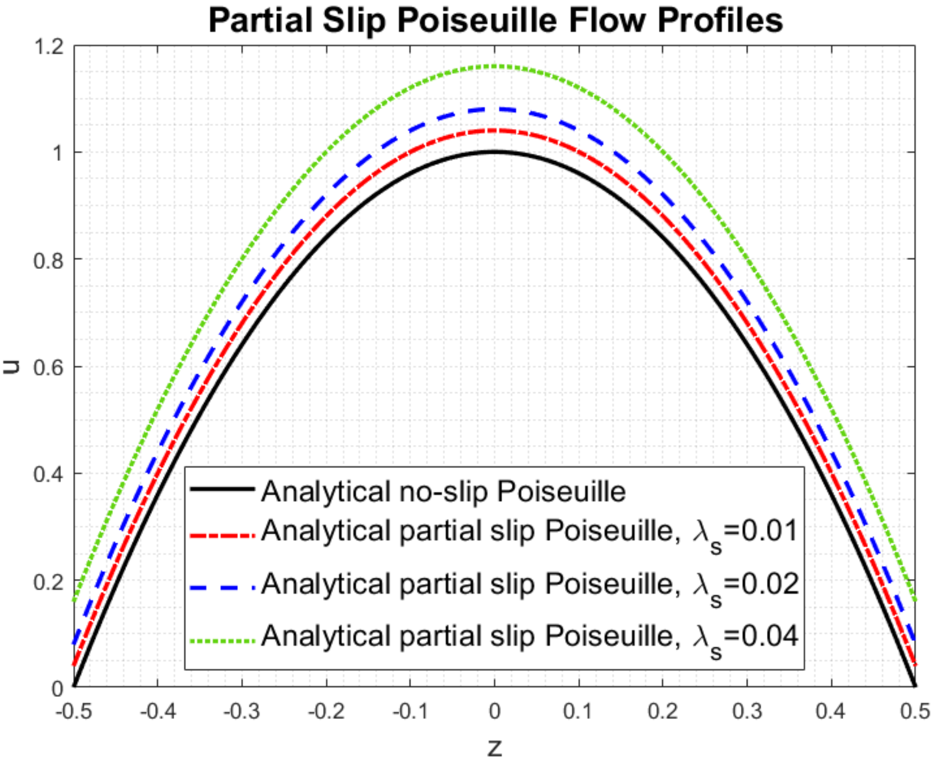


Figure 2.2: Plot of Poiseuille velocity profiles with no-slip and partial slip boundary conditions with increasing sized slip length.

a hydrophobic surface was then used in the company LiquiGlide (2020) to coat the inside of bottles and increase the amount of liquid that can be discharged . The main advantage for the LiquiGlide coating is its durability which is an area of concern for hydrophobic surfaces. Any scratches of imperfections in the surface can cause it to loose its hydrophobicity. This issue was tackled by Brown and Bhushan (2015) and Brown and Bhushan (2016) who developed a different method to create a durable hydrophobic surface. The surface was created by layering hydrophobic polymer on a smooth plastic surface in a textured manner. When tested, water flowed over the surface with ease and the surface even showed signs of being "Shampoo-phobic", allowing shampoo to slip over the surface. However the surface developed lost its hydrophobic nature after a while.

Partial slip has also been seen to play a role in flows of polymer melts. Both Black (2000) and Denn (2001) discuss the instabilities present in the extrusion processes of polymer melts. As the polymer is extruded and begins to cool, it will occasionally stick to the sides of the extrusion die and fold over its self. This instability is know as shark skin due to its rough scaly appearance. It was found that by using a surface over which the polymer can more easily slip the magnitude of the instability was reduced and even eliminated completely in some cases. This shows that partial slip can play an important roll in the extrusion of polymer melts.

2.3.3 Slip Length

In the partial slip condition, Equation (2.7) the slip length λ_s is a quantity with dimensions of length. In most Newtonian cases the slip length is a few nanometers to a few micrometers in length (Lauga et al. (2005)), hence it is normally only considered in flows of that length scale and the standard no-slip condition is used for macro-scale flows. The review of Denn (2001) on the extrusions of polymer melts references flows with slip lengths of several hundred to over a thousand micrometers. Though the process of LiquiGlide (2020) shows large amount of fluid slip near the boundaries, the slip lengths were not stated.

The slip length controls the magnitude of the partial slip condition, but due to its size it can be difficult to measure. Experiments have found that it changes with may different material properties and so could depend on many things. One simple approach for a Newtonian fluid is the observation that the tangential force per unit area exerted on the solid by the fluid is proportional to the partial slip velocity u_{slip}, that is

$$\sigma_{xy} = \kappa u_{slip}, \tag{2.11}$$

where κ is the friction coefficient of the surface. This is then combined with the bulk Newtonian fluid constitutive equation

$$\sigma_{xy} = \mu \frac{\partial u}{\partial y}, \tag{2.12}$$

where μ is the dynamic viscosity of the liquid, giving the relation

$$u_{slip} = \frac{\sigma_{xy}}{\kappa} = \frac{\mu}{\kappa} \frac{\partial u}{\partial y}. \tag{2.13}$$

Then using the partial slip condition it is found that (De Gennes (2002))

$$\lambda_s = \frac{\mu}{\kappa}, \tag{2.14}$$

For simple fluids the slip length is very small (De Gennes (2002)) however there are a number of cases where the slip length can be large. For example flow of polymer melts and flows over hydrophobic surfaces.

2.3.4 Slip Length Measurements

Due to the size of a slip length it can be difficult to measure. However there does exist a number methods to do this with varying degrees of accuracy. Methods can be split into two groups of direct and indirect techniques, Lauga et al. (2005). Direct techniques include micro-particle image velocimetry on total internal reflection fluorescence (Lauga et al. (2005)), the accuracy of these techniques has been gradually improving with the most accurate in the range of a $10nm$. Other direct methods include near field laser velocimetry, nuclear magnetic resonance (NMR) imaging and different forms of particle tracking, Sochi (2011). The more indirect methods include measuring hydrodynamic forces with a Colloid probe atomic force microscope or with surface force apparatus, torque measurements with a rheometer and pressure drop analysis where pressure and flow rate are measured in a channel. The most accurate of these methods can achieve measurements with accuracies of $1 - 2nm$.

Malm and Waigh have had a lot of success in measuring the slip lengths in a number of different fluids including toothpaste, shampoo and soap using a technique called optical coherence tomography velocimetry (OCTV) (investigated in Malm (2015), Malm and Waigh (2014), Malm et al. (2014), Malm et al. (2015) and Harvey and Waigh (2011)). With this method a sample of fluid is place in a rheometer comprising of two flat circular disks, the top disk is then rotated at a set velocity and the bottom disk held stationary. A laser beam is aimed through the sample into a detector on the other side. Doppler shifting in the light spectrum then allows accurate velocity measurements through the sample. Using the velocity data, slip lengths are then found. Malm et al. (2015) noted that the same liquid can exhibit different slip lengths on different materials. A sample of soap was placed into a rheometer with a glass lower plate and a perspex upper plate, with a larger amount of slip occurring on the perspex plate. This shows that partial slip not only depend on the fluid but also on the boundary.

Slip lengths for non-Newtonian flows can be much larger than those of Newtonian flows. In the review by Denn (2001) observations of slip lengths of up to hundreds or in some cases thousands of micrometers are possible for polymer melts. However these slip lengths only appear after a yield stress has been exceeded, until this point lower slip lengths are reported and even sometimes no slip at all. In all cases the boundary surface had been made up of either fluoropolymers or polished silica, each experiment being with a different polymer melt.

2.3.5 Partial Slip used in Computational Methods

In CFD a number of different techniques have been employed to model flows with a partial slip condition. Due to the small size of the slip length a lot of work has been done using Molecular Dynamics (MD). MD is a particle based method used to model flows on a molecular scale where the intermolecular forces play a role. For a particle i surrounded by particles j Newton's second law is written as

$$m_i \frac{d^2 \mathbf{r}_i}{dt^2} = \sum_j \mathbf{F}_{ij}, \tag{2.15}$$

where $\mathbf{F}_{ij}$ are the intermolecular forces between particle i and particle j. $\mathbf{F}_{ij}$ takes the form Lauga et al. (2005)

$$\mathbf{F}_{ij} = -\nabla_i V_{ij}. \tag{2.16}$$

V_{ij} is the interaction potential and takes the form of a Lennard-Jones two-body potential Lauga et al. (2005)

$$V_{ij} = \epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - c_{ij} \left(\frac{\sigma}{r_{ij}} \right)^{6} \right], \tag{2.17}$$

where ϵ is an energy scale, σ is the atomic size of the particles, r_{ij} is the distance between particles and c_{ij} are constants that describe the intermolecular attraction. The c_{ij} are tuned to approximate the interactions between two fluid molecules (c_{FF}) and between fluid and solid molecules (c_{FS}).

Barrat and Bocquet (1999) used MD to simulate the formation of a water droplet on a smooth surface that was initially wetted. The initial conditions for the simulation were a “fluid slab” held between two plates. The simulation is left to form a fluid droplet and the contact angle was measured via the approximation to Young’s equation

$$\cos(\theta) = -1\frac{2\rho_S c_{FS}}{\rho_F c_{FF}}, \tag{2.18}$$

where ρ_S and ρ_F are the densities of the solid and fluid respectively (Barrat and Bocquet (1999), Lauga et al. (2005)). It was found that changing the value of c_{FS} results in a change in the contact angle, with $c_{FS} = 1.0$ resulting in $\theta = 90^o$ and $c_{FS} = 0.5$ giving a contact angle of $\theta = 140^o$, resulting in a hydrophobic surface. The same set up was then used to model Couette and Poiseuille flows with the different values of c_{FS}. Comparing their results to analytical solutions they found that large slip effects occur at surface with higher contact angles.

Cottin-Bizonne et al. (2004) also looked at flows passed hydrophobic surfaces using MD, but this time grooved roughened surfaces were considered. The rough surface was filled with fluid and the effects of pressure were investigated on the wetting behaviour of the surface for different values of c_{FS}. For high pressures and high values of c_{FS}, for example $c_{FS} = 0.8$, for a “normal surface” the surface stays wetted and so the fluid stays in the grooves where it started. For lower pressures and lower values of c_{FS}, for example $c_{FS} = 0.5$ corresponding to a contact angle of $\theta = 137^o$, the surface de-wets and the fluid in the groove starts to rise up out of the groove. In a simulation of a Couette flow they found that for the “normal” surfaces with a lower contact angle the presence of the surface roughness decreased the slip length of the flow, but for the higher contact angle hydrophobic surface the slip length of the flow increased.

The Lattice Boltzmann Method (LBM) is mesoscopic particle based method that has been used to simulate flows involving partial slip. LBM, like MD, also uses forces between particles to calculate flow properties. However the particles used in LBM are much larger than their MD counterparts allowing for larger length and times scale to be simulate. Harting et al. (2006) presented an approach of using a multi-phase LBM to investigate slip in hydrophobic micro-channels. By linking the fluid-wall interaction strength to the constant angle that form between a fluid droplet and a solid surface, they were able to hydrophobic interactions. They found that a thin layer of fluid close to the boundary would develop a lower density leading to the bulk fluid slipping over the top. This mesoscopic layer then lead to a macroscopic partial slip flow with the size of the slip length linked to the thickness of this layer through. The slip length was then given by

$$\lambda_s = \left(\frac{\rho_{bulk}}{\rho_l} - 1 \right) \delta,$$

where δ is the thickness of the layer and ρ_{bulk} and ρ_l are the densities of the bulk fluid and the layer respectively. Their investigation reproduced results from MD showing that the slip length is independent of flow velocity. They conclude that their method was an improvement over the existing MD simulations as the LBM is able to simulate the larger length and time scales that MD could not.

Yan-Yan et al. (2008) also used the LBM to model Couette flows with partial slip at the boundary. The also found that a thin layer of fluid near the boundary becomes less dense and the bulk of the fluid slips of glides over this layer. They then investigated how the slip length changes when the strength of the fluid-fluid and fluid-solid particle interactions are varied. The results showed that the slip length increased with the strength of attraction for both fluid-fluid and fluid-solid interactions. The effect of viscosity on the slip length was also investigated revealing that the slip length increase with the fluid viscosity.

In both of these LBM examples the slip length is not directly used by the method, instead analytical partial slip solutions are fitted to the results and the slip length found using this. The bulk of the fluid slips over a thin layer of less dense fluid, the fluid in this thin layer experiences no-slip at the boundary however the overall effect is of apparent partial slip.

Renardy et al. (2001) used a finite difference volume of fluid method to model the spreading of a fluid droplet on a smooth surface. They used a partial slip condition in the region near the contact angle between the liquid and air phases. This is a natural application in which to find partial slip, as if the no-slip condition is strictly applied then the droplet would not spread. The contact line including the partial slip condition moves faster than the same line on the no-slip surface. This is to be expected as there is a force helping to spread the droplet when partial slip is present. However the no-slip condition applied was not perfect as the algorithm used introduced a slip on the scale of the mesh.

When investigating instabilities in the extrusion of polymer melts, Black (2000) found that fluid slip can sometimes have a stabilizing effect. However sometimes depending of the velocity of the extrusion, slip can have a destabilizing effect. The instability investigated was called “shark skin” or waves forming on the surface of the polymer melt that occur if the extrusion process exceeds a critical flow rate.

The instability looks like the corrugations on a straw and reduces the usefulness and marketability of the extruded polymer. Through a method called Chebyshev Collocation, he found that at low velocities an increased amount of slip destabilizes the flow. But for large slip coefficients the critical flow rate becomes constant and increasing the amount of slip stabilizes the flow.

2.3.6 SPH

Looking at the above range of methods it is clear that partial slip is a technique used by a number of CFD methods. The majority of the studies that use partial slip are micro or meso-scopic particle methods like MD and LBM, this likely to be due to the small scales at which partial slip is normally observed at. However it has been shown that it can be used for macroscopic flows by Cottin-Bizonne et al. (2004), and surfaces such as those of LiquiGlide give insight into how effective it can be in real world applications. MD excels in simulating flows at the micro-scale due to each particle representing individual atoms or molecules. LBM can simulate flows at larger length and times scales while still incorporating the molecular interactions between fluids and hydrophobic surfaces seen by MD, as by Harting et al. (2006). However, as the size of the lattices increases to cover more macro-scale flows, LBM struggles to fly resolve the correct flow profile close to the wall when this region is thinner than a single lattice spacing, Harting et al. (2010). At this length scale the molecular details used by the smaller scale LBM can no longer be taken into account. It could however bridge the gap between MD and a more macro-scale method such as SPH that cannot take into account any of the molecular forces.

The SPH method was previously identified as a good candidate for this project due to its ability to model non-Newtonian flows, free-surface flows and deformable boundaries. Compared to MD, SPH is more suited to this project as MD is limited by its small length scales. SPH would need far fewer particles to model bottle pouring than MD. As for the LBM, it is based in the discrete Boltzmann equation and not the Navier-Stokes equations like SPH. The Navier-Stokes equations give a better description of fluid mechanics at the macro-scale than the Boltzmann equation. As will be shown in the following, true partial slip has not yet been implemented in the SPH method, opening a new avenue of research. Hence SPH is the preferred method and will be used for this project.

State-of-the-art no-slip SPH boundary conditions SPH boundary conditions in general are an interesting and complex subject, and have been highlighted as one of the grand challenges of SPH by SPHERIC (Vacondio et al. (2020)), the international SPH community. Boundary conditions in SPH can be split into three main groups: repulsive forces; boundary integral methods, and fictitious particle methods. The first type of boundary conditions are some of the earliest adopted SPH boundary conditions, for example Monaghan (1994) and Monaghan and Kajtar (2009). Using this approach the boundary is formed of a single layer of fixed particles, these boundary particles then produce a repulsive force through a Lennard-Jones type interaction with fluid particles. Apart from this force, the boundary particles have no other effect on the fluid particles. The force felt by the fluid particle from the boundary is dependent on the distance between the fluid particle and the surrounding boundary particles. The force is set to zero is the distance is greater than the initial inter-particle distance and grows as the fluid particle gets closer to the boundary. This repulsive force stop the fluid particles going through the boundary, thus creating a no-penetration condition. This approach alone is not enough however to give a no-slip boundary condition at the boundary. When no-slip is required, the boundary particles are then included in the calculation of the viscous terms of the fluid particles as suggested by Monaghan (1994). While this method does create a no-penetration condition, the kernel supports of fluid particles near the boundary become truncated leading to errors in calculation.

The second type are the semi-analytical boundary or boundary integral conditions that have been used by Ferrand et al. (2013) and Mayrhofer et al. (2015). Using this process the conservation of momentum and mass equations evaluated for fluid particles can be slip up into contributions from fluid and from boundary. The method uses re-normalisation of the discrete SPH operators to compensate for the incomplete kernels of fluid particles near boundaries. By splitting the boundary up into line segments, the boundary contribution can be calculated by corrected boundary integrals using the normals to the boundary segments. The fluid particles close to the boundary feel an extra repulsive force from the boundary segments, thus ensuring no-penetration. The re-normalisation precess helps to over come the truncated kernel issue of the previous method. This method does however struggle with complex shaped geometries, especially in 3D. The method is also computationally expensive due to the added calculation of boundary integrals required.

The third type uses fictitious particles to fill the solid region beyond the fluid domain thus completing the kernel of fluid particles with boundary particles when close to a boundary. This type of method can be further split into dummy particle methods and mirroring methods. Examples of dummy particle methods include the dynamic boundary condition of Dalrymple and Knio (2001), and the methods of Adami et al. (2012) and Marrone et al. (2011). These methods fill the boundary region with fixed particles that complete the kernel of fluid particles close to the boundary.

The no penetration and no-slip conditions are provided by the properties of the boundary particles including density and velocity that are updated along side the fluid particles. Complex shaped geometries can be constructed simply by forming the boundaries out of multiple layers of boundary particles. A big advantage of these methods is their simplicity while still giving good approximations to the no-slip and no penetration conditions.

The mirroring method also fill the boundary region with particles, but unlike the dummy particle methods these particles are not fixed and change each time step. The method of Colagrossi and Landrini (2003), often called mirror boundaries, reflect fluid particle close to a boundary across the boundary, forming mirror images of the fluid particles. The kernel of fluid particles close to the boundary are completed by these mirror images that have the same density and pressure as the original. No-slip and no penetration can be defined by reversing the velocity direction of the mirror particles. This method however does struggle with complex shaped boundaries. This was solved by the multiple boundary tangent method of Yildiz et al. (2009). This method creates a number of tangent lines to the boundary and uses these as the planes through which to mirror. This however requires knowledge of the boundary normals and tangents which can be difficult in 3D.

For this thesis the dummy particle type methods are the most appropriate as they are able to create complex shaped geometrics and are simple to use compared to the other methods. As far as has been seen in the literature the partial slip boundary condition described above has not yet been used in SPH.

Partial slip in SPH There have not been many references found in the SPH literature referring to partial slip at the boundary. Shen et al. (2000) mentioned a 'partial slip' condition for the modelling of river ice dynamics. This boundary condition is used when river ice is grounded and so is dragging along the bottom of the river. The SPH boundary technique used was that of mirror particles, with the tangential velocity of the mirror particles set equal to that of the fluid. This boundary condition is a free slip boundary condition and not a partial slip condition as suggested.

Cao and Li (2017) modelled the resistance to slippage of fresh concrete flowing over a solid boundary. The test geometry was an L-shaped channel with the vertical section filled with concrete and contained at the bottom with a gate. Like for a dam break simulation the gate is opened and the concrete is allowed to flow along the horizontal section of the channel. After the gate was opened, they measured the distance that the concrete flowed down the channel and compared it to experimental results, for three different concrete samples, with good agreement between the numerical and experimental results. Each of the different concrete mixes had different levels of water content which was show to effect the amount of slippage, with more slippage occurring for higher water content mixes. The boundary conditions used for the SPH model were the Lennard-Jones like repulsive forces used by Monaghan (1994) with a slippage resistance at the boundary described by

$$\tau_s = \eta_s v + \tau_{sy}, \tag{2.19}$$

where τ_s is the slippage resistance stress at the boundary, η_s is the plastic viscosity of slippage, τ_{sy} is the yield stress of slippage and v is the velocity in the direction of flow down the channel. To highlight the effect of slippage at the boundary they ran the SPH simulations again without the slippage resistance. The results showed that the flows with lower water content gave unphysical results in the absence of slippage resistance. For these simulations the simulated concrete flowed much further than was seen in the experiments. While this condition does involve slip at the boundary, it is not of the form that was described previously, and it is not clear if the particles at the boundary are given a velocity or if the slip is controlled though the Lennard-Jones potential like in Molecular Dynamics.

2.4 Non-Newtonian fluids

The Newtonian fluid is any fluid in which the shear stress and shear rate of the fluid share a linear relationship given by

$$\boldsymbol{\tau} = \mu \dot{\boldsymbol{\gamma}}, \tag{2.20}$$

where $\boldsymbol{\tau}$ is the shear stress, $\dot{\boldsymbol{\gamma}}$ is the shear rate and μ is the dynamic viscosity of the fluid. The dynamic viscosity is constant and can be found at the gradient of the line found when plotting the shear stress against the shear rate. Examples of fluids that can be described in this way include water, sugar syrups, carbonated drinks, oils, milk and most honeys (Ojediran and Raji (2010)).

If a fluid does not have this linear relationship between the shear stress and the shear rate then the fluid can be said to be non-Newtonian. Some examples of different non-Newtonian fluid models will be described below and plots of their curves included in Figure 2.3. A large number of Unilever type fluids have been described using non-Newtonian models in the literature such as Mayonnaises, sauces and shampoos.

2.4.1 Examples of non-Newtonian fluids

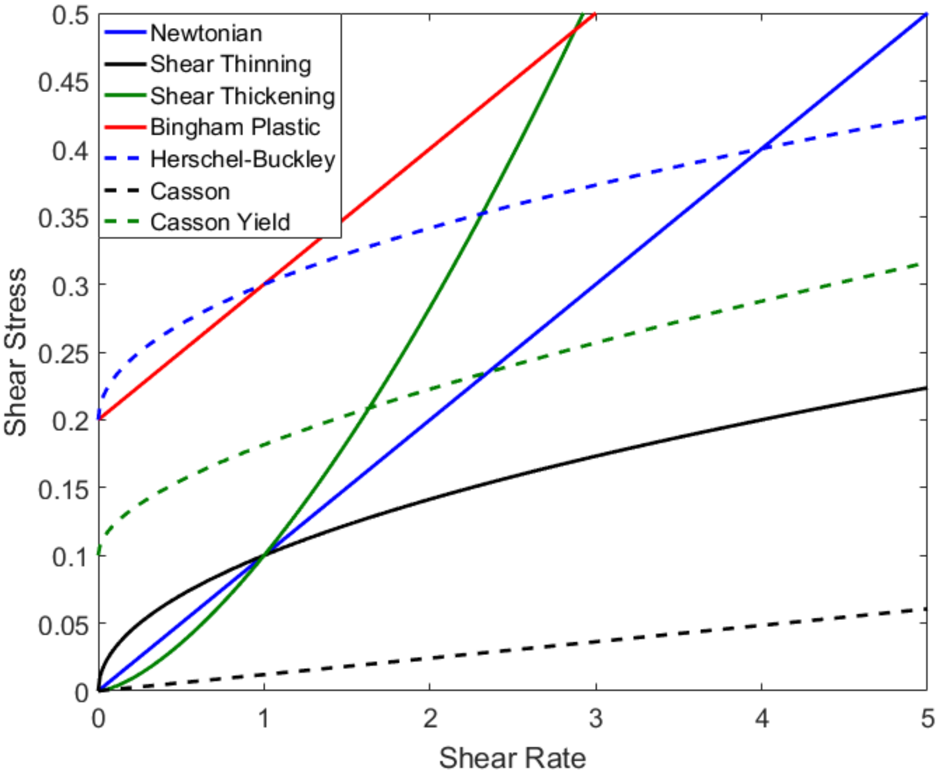


Figure 2.3: Plots of shear stress versus shear rate for a number of different non-Newtonian fluid models.

Bingham plastic model If a fluid does not flow when not under external stresses then there exists a yield stress under which it will flow. If the fluid exhibits a yield stress but otherwise acts as Newtonian fluid, the fluid is called a Bingham plastic fluid (Bingham (1922)). This type of fluid is described by the equation

$$\tau = \tau_y + \mu\dot{\gamma}, \tag{2.21}$$

where τ_y is the yield stress of the fluid. If the applied shear stress is lower than the yield stress then the fluid acts as a solid and does not flow, however once the yield stress is reached then the fluid flows in the same was a Newtonian fluid. The Bingham model is shown in Figure 2.3 as the solid red line.

Power law model The viscosity of some fluids change as the applied shear rate increases, either increasing or decreasing. These types of fluids are called shear thickening and shear thinning respectively. The simplest approach to model these types of is through a model called the power law model which is described by the equation

$$\tau = K\dot{\gamma}^n, \tag{2.22}$$

where K is the consistency index with units Pas^n and n is dimensionless the power law index. The power law index controls how close the fluid is to Newtonian, when $n > 1$ the fluid is shear thickening and when $n < 1$ the fluid is shear thinning. $n = 1$ recovers the Newtonian fluid. The values of K and n for a fluid can be found by plotting shear stress against shear rate on a $log - log$ scale. A straight line is formed whose intercept with the y axis is $log(K)$ and whose gradient is n. The shear thickening and shear thinning fluids are shown in Figure 2.3 by the solid green and black lines respectively. Examples of shear thickening fluids are rare but include solutions of corn starch or very thick custard. Examples of shear thinning fluids are more common and most non-Newtonian foods exhibit shear thinning behaviour such as salad dressings or ice creams (Joyner (2019)).

Hershel-Buckley model If a power law fluid also exhibits a yield stress then the fluid can be described as a Herschel-Buckley fluid (Herschel (1924)). The equation to describe the relationship between shear stress and shear rate is given by

$$\tau = \tau_y + K\dot{\gamma}^n, \tag{2.23}$$

where τ_y is the yield stress. For this type of fluid, after the yield stress has been reached, the fluid can shear thin or thicken in the same way as a power law fluid. A shear thinning Herschel-Buckley fluid is shown in Figure 2.3 by the dashed blue line. Mayonnaises and mustard are examples of Hershel-Buckley fluids (Joyner (2019)).

Casson model Another example of a fluid model that can be used to model shear thinning or shear thickening fluids is the Casson model (Casson (1959)). For a fluid with a yield stress the Casson model is given by

$$\sqrt{\tau} = K_0 + K\sqrt{\dot{\gamma}}, \tag{2.24}$$

where K_0 is the square root of the yield stress and K is the square root of the Casson viscosity. When the square root of shear stress is plotted against the square root of shear rate a straight line is formed which has gradient K and intersects the y axis at K_0. If the fluid does not have a yield stress then K_0 is equal to zero. Plots for the unyielded and yielded Casson models are shown in Figure 2.3 by the dashed black and green lines respectively. The Casson model has been adopted as the official method of the International Office of Cocoa and Chocolate for modelling the flow behaviour of chocolate (Ojediran and Raji (2010)).

Cross and Carreau models The cross and Carreau models are two similar models for shear thinning fluids. The Carreau model has been used by King and Lind (2019) to match to rheometer data of a sample of shampoo for example. The Cross model for the apparent dynamic viscosity is given by

$$\mu_{app} = \mu_{\text{inf}} + \frac{\mu_0 - \mu_{\text{inf}}}{1 + (\alpha_c \dot{\gamma})^m}, \tag{2.25}$$

and the Carreau model by

$$\mu_{app} = \mu_{\text{inf}} + \frac{\mu_0 - \mu_{\text{inf}}}{\left[1 + (\lambda_c \dot{\gamma})^2\right]^N}, \tag{2.26}$$

where μ_0 is the zero shear viscosity, μ_{inf} infinite shear viscosity, α_c and λ_c are time constants related to the relaxation time of the fluids and m and N are dimensionless constants. Both the Cross and Carreau models have been used to describe the shear dependence of aqueous solutions of pectins or gums such as locust gum (Ojediran and Raji (2010)).

2.4.2 Non-Newtonian Flows in SPH

Non-Newtonian flows have previously been modelled by a number of authors using SPH including Shao and Lo (2003), Ellero and Tanner (2005), Fang et al. (2006), Fan et al. (2010), Xenakis et al. (2015) and Fourtakas and Rogers (2016) have modelled non-Newtonian flows for example. Shao and Lo (2003) used an incompressible SPH model to simulate Newtonian and non-Newtonian dam breaks using both Bingham and Cross models for the non-Newtonian behaviour. The Newtonian results were compared with experimental data and to results of a VOF simulation with good agreement to the experiment for the location of the leading edge of the toe of the dam break. A comparison of the shape of the column of water at different time instants showed good agreement with the shape of the free surface produced by the VOF model. The non-Newtonian models were tested using a dam break type mud flow and also compared to experimental results showing good agreement for the surface profiles at different times.

Ellero and Tanner (2005) and Fang et al. (2006) used weakly compressible SPH to simulate transient viscoelastic flows. Ellero and Tanner (2005) focussed on Poiseuille flow using first a Newtonian fluid and then Oldroyd-B and Upper convective Maxwell models for the viscoelastic properties. Comparison of the SPH result to the transient analytical solution showed good agreement for both the shape of the velocity profiles ans the evolution of the maximum velocity for both the Newtonian an Oldroyd-B fluids. The agreement for the Maxwell fluid was not as good for small times, but as the flow progressed in time the agreement got closer. Fang et al. (2006) focussed on transient free surface flows, also using a Oldroyd-B model for the viscoelastic fluid. The authors included an artificial stress term in the SPH momentum equation to overcome a tensile stress instability. The test case used was of a fluid drop falling on a rigid flat plate with attention given to the change in width of the droplet during the impact. Without the artificial stress term the water droplet displays non-physical fracturing when impacting the plate. Comparing the change in width of SPH droplet to results from a finite difference model showed good agreement, for both the Newtonian and Oldroyd-B droplets, when the artificial stress term is included. By changing the parameter of the viscoelastic model the SPH was able to capture the bouncing effect seen in viscoelastic free surface flows.

Fan et al. (2010) used weakly compressible SPH to simulate moulding flows of non-Newtonian fluids with viscosity described using a power law. The model was first validated using Poiseuille flows for Newtonian and power law fluids showing good agreement to the analytical solutions. The model was then used to simulate the moulding flow fo a power law fluid pressed through a die, comparing three equations of state for the pressure in the fluid. The three methods used were the weakly compressible fluid (WCF) method, often used in SPH for Newtonian flow, the Tait model, and the model of Sun, Song and Yan (SSY). The shape of the fluid as it is pressed through the die shows agreement between the Tait and SSY models but not for the WCF model. The authors concluded that the Tait and SSY models were more realistic for these types of flows but did require more computational time to use. Xenakis et al. (2015) developed a non-Newtonian incompressible SPH formulation that was validated through a number of bench mark test cases. The first test case considered was a Poiseuille flow using a bilinear model, an exponential model and a cross model for the viscous terms with close to first order convergence to the analytical solution. The model was then used to simulate a dam break of a water and clay mixture using both Herschel-Buckley and power law models for the viscous terms. The results showed good visual agreement with results from previous studies modelled using the weakly compressible SPH formulation, highlighting the ability of the method to reproduce results but with a smoothed pressure field characteristic of incompressible SPH. Later moulding flows were modelled using two geometries, including that of Fan et al. (2010), again highlighting the smooth pressure field compared to weakly compressible SPH and discussing the importance of particle shifting to achieve a good particle distribution.

Fourtakas and Rogers (2016) used the Herschel-Buckley-Papanastiou (HBP) (Papanastasiou (1987)) model to simulate scouring of sediment beds subject to dam breaks. This was done using the weakly compressible formulation SPH and accelerated using a GPU. First a water droplet was dropped on a flat plate to see the effect of shifting, it was seen that the use of shifting greatly improved the particle distribution avoiding unphysical voids. Next the sediment phase of the simulation was tested to validate the yield stress approach for the sediment by modelling a sediment damn break. The HBP model was used to smooth out the stress discontinuity in the standard Herschel-Buckley model. The modification due to Papanastasiou (1987) is given by

$$\tau = \tau_y \left[1 - \exp(-m\dot{\gamma})\right] + K\dot{\gamma}^n, \tag{2.27}$$

where the parameter m controls the stress growth below the yield stress. The article then moved on to model a 2-D and 3-D two-phase erodible dam break. The dam breaks showed good agreement to previous numerical and experimental results capturing the scouring effect of the water phase on the sediment phase.

2.5 Summary

The aim of this project is to develop a tool that can be used to reduce the amount of waste left inside packaging through a modelling method. Design of the shape of the packaging and the effects of this on the flow using CFD have been explored and well covered by grid based methods for multiphase flows. The particle based studies highlighted their advantage of being able to model just the fluid phases when air and fluid exchange is not important.

Coating the inside of the packaging with a partial slip coating provides an alternative approach that has not been as well investigated for macro-scale flows. The current modelling methods of MD and LBM focus on the meso-scopic details of the flows. SPH can bridge the gap applying partial slip flows to macro-scale and complex geometries. Partial slip is unexplored in SPH and gives the potential for range of different applications in the longer term, that would be beyond reach of MD and LBM.

In order to develop a partial slip boundary condition in SPH a new no-slip boundary condition is required. These new boundary conditions can build upon existing dummy particle type methods that are well suited to creating complex geometries, are inexpensive to use and have been used in a wide range of applications.

Chapter 3

Smoothed Particle Hydrodynamics

3.1 Introduction

Smoothed particle hydrodynamics (SPH) is a particle based Lagrangian interpolation method, first developed by Gingold and Monaghan (1977) and Lucy (1977), that can be used for modelling fluid flow. SPH was first developed to simulate astrophysics applications such as star formation. Since then a number of additions have been made including viscosity models and various boundary conditions to allow it to simulate fluid flows. In SPH the fluid domain is split up into small packets of fluid of constant mass, each with its own physical properties such as pressure and velocity. Unlike a traditional grid based CFD methods, these fluid packets or particles are able to move around the fluid domain according to the governing equations. Due to its particle nature it is well suited to problems involving free surfaces and large deformations. Grid based method require a complex re-meshing for flows with large deformations or inclusion of a second air phase for flows involving a free surface. These abilities have made SPH very attractive for many engineering applications including coastal defences (Zhang et al. (2018)), fish pass (Novak et al. (2019)), and fibre manufacture processes (Huntley et al. (2020)). The aim of the following chapter is to introduce the reader to the weakly compressible (WC) SPH method that will be used in this project.

3.2 Interpolation process

The SPH method uses a two step interpolation process. The first step involves the continuous approximation of a function by an integral. This starts with the convolution of a function $\phi(\mathbf{x}, t)$ and the Dirac δ function defined on the domain

Ω:

$$\phi(\mathbf{x}, t) = (\phi * \delta)(\mathbf{x}, t) = \int_\Omega \phi(\mathbf{x}', t)\delta(\mathbf{x} - \mathbf{x}')d\Omega, \tag{3.1}$$

where $\mathbf{x}'$ is any other point in the domain. The Dirac δ function can then be approximated through a smoothing kernel $W(\mathbf{x} - \mathbf{x}')$ defined on a domain Ω_h of radius $2h$ where h is called the smoothing length

$$\phi(\mathbf{x}, t) \approx \int_{\Omega_h} \phi(\mathbf{x}', t)W(\mathbf{x} - \mathbf{x}')d\Omega_h. \tag{3.2}$$

Violeau (2012) recommend that the interpolation possess the properties

$$\int_{\Omega_h} W(\mathbf{x} - \mathbf{x}')d\Omega_h = 1, \tag{3.3}$$

known as the zeroth order moment, and

$$\int_{\Omega_h} (\mathbf{x} - \mathbf{x}')W(\mathbf{x} - \mathbf{x}')d\Omega_h = \mathbf{0}, \tag{3.4}$$

called the first order moment.

Using a discrete approximation the integral in equation (3.2) can be split up into Riemann sums evaluated at a central particle with position $\mathbf{x}_i$, approximating the function further:

$$\phi(\mathbf{x}_i, t) \approx \sum_j \phi(\mathbf{x}_j, t)W(\mathbf{x}_i - \mathbf{x}_j)V_j, \tag{3.5}$$

where V_j is the volume of the associated particle j. Setting $V_j = m_j/\rho_j$ the expression becomes

$$\phi(\mathbf{x}_i, t) \approx \sum_j \frac{m_j}{\rho_j}\phi(\mathbf{x}_j, t)W(\mathbf{x}_i - \mathbf{x}_j). \tag{3.6}$$

The zeroth order moment, equation (3.3), in discrete form gives the condition

$$\sum_j \frac{m_j}{\rho_j}W(\mathbf{x}_i - \mathbf{x}_j) = 1, \tag{3.7}$$

that the kernel should also satisfy (Gomez-Gesteira et al. (2009))

3.2.1 The SPH Kernel

As well as being an approximation for the Dirac δ function and have the property described in equation (3.7), the kernel function should ideally have a few other features. The kernel should have compact support such that it is non zero on a small domain and zero everywhere else, the compactness is needed for the sparsity of the discrete equations (Belytschko et al. (1998)). The kernel should also be positive and symmetric inside the support domain and monotonically decreasing towards the edge of the support (Gomez-Gesteira et al. (2009)). The kernel should also be at least once differentiable on Ω_h, so that gradients can later be evaluated. An example of the shape of a kernel can be found in Figure 3.1.

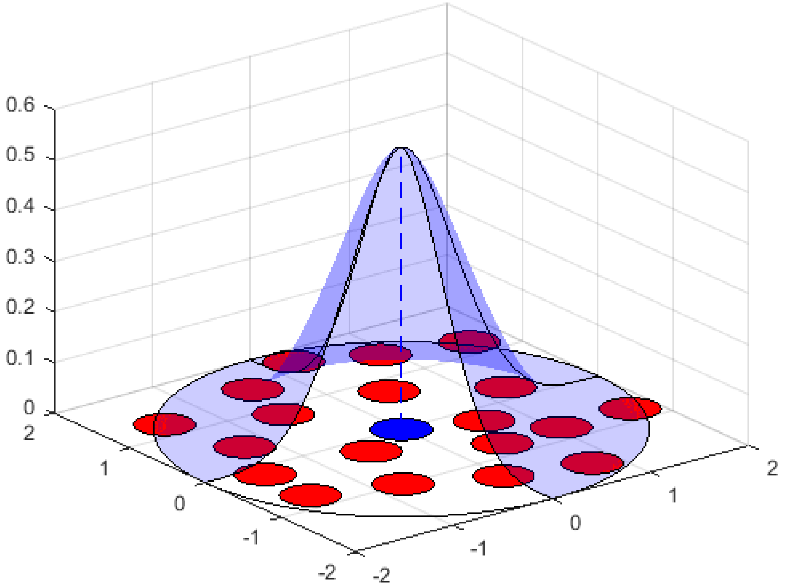


Figure 3.1: Example of a smoothing kernel. The kernel should have its largest value in the centre and monotonically decrease as the radius increases.

Examples of Kernels A kernel function can be written in the form

$$W(\mathbf{x}_i - \mathbf{x}_j) = \frac{\alpha}{h^n} f\left(\frac{\mathbf{x}_i - \mathbf{x}_j}{h}\right), \tag{3.8}$$

where f is the function that describes the kernel and α is a normalisation constant that depends on the kernel and the number of dimensions which ensures that the area under the curve f is equal to 1. Using the substitution

$$q = \frac{\mathbf{x} - \mathbf{x}'}{h} = \frac{\mathbf{r}}{h}, \tag{3.9}$$

α can be found for each kernel through

$$\alpha = \frac{h^n}{\int_{\Omega_h} f(q) d\Omega_h}. \tag{3.10}$$

There are a number of different kernels in use in the literature, a few examples of which will be given here in non-dimensional form along with their normalisation constants.

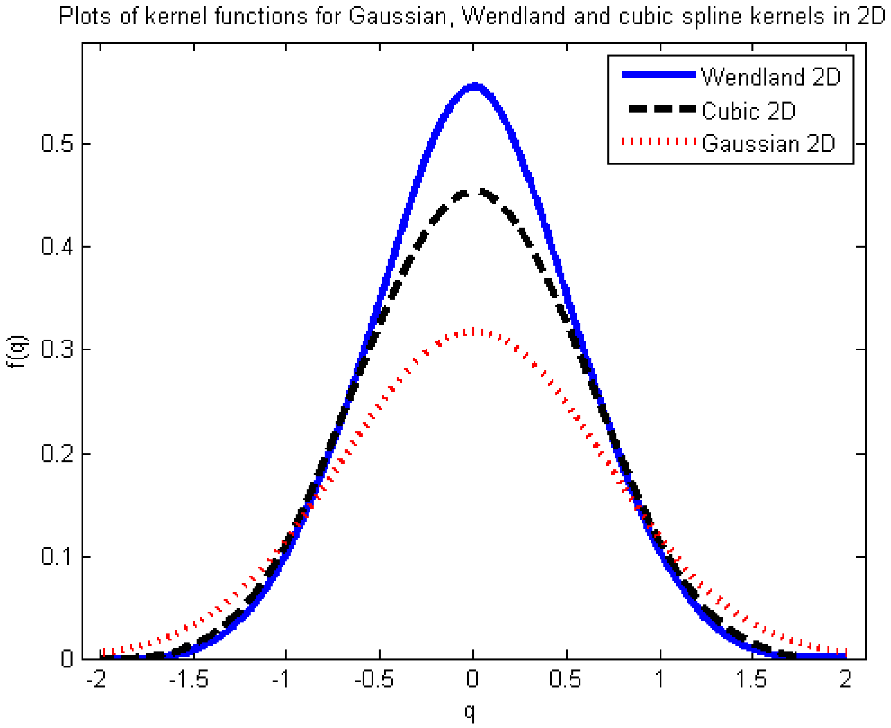


Figure 3.2: Plots of $f(q)$ vs q for Wendland, Cubic spline and Gaussian kernels.

Gaussian Kernel One of the earliest kernels used in SPH is the Gaussian kernel

$$f_G = e^{-q^2}, \tag{3.11}$$

and has normalization constants

$$\alpha_G = \frac{1}{\pi^{n/2} h^n}, \tag{3.12}$$

given by Goffin (2013), where n is the number of dimensions. The Gaussian kernel is the original kernel that was used by Monaghan. It has compact support on an infinite radius so in theory the computational domain around a particle can be extended without increasing the smoothing length, however in practice the contribution of a particle more that $4h$ away from centre is negligible. Since the Gaussian kernel is based on an exponential, it only tends to zero outside of the domain and never actually reaches it. The plot of the Gaussian kernel is found in Figure 3.2 as the red dotted line.

Polynomial Splines A group of kernels that follow the criteria set are the polynomial splines. These are piecewise polynomial functions that are constructed to be positive, decreasing and piecewise smooth in the domain and zero everywhere else. Examples include the Cubic spline which is given by the equation

$$f_3 = \begin{cases} 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3 & 0 \leq q \leq 1 \\ \frac{1}{4}(2-q)^3 & 1 < q \leq 2 \\ 0 & 2 < q, \end{cases} \tag{3.13}$$

with normalization constants for 1-D, 2-D and 3-D respectively Goffin (2013)

$$\alpha = \frac{1}{h}\,,\ \alpha = \frac{15}{7\pi h^2}\,,\ \alpha = \frac{3}{2\pi h^3}. \tag{3.14}$$

The plot of the cubic kernel is found in Figure 3.2 as the black dashed line.

Wendland kernel Wendland (1995) developed a family of polynomial splines which can be used as SPH kernels. The most popular of these is the quintic version that is often simply referred to as the Wendland kernel. The kernel is defined by the function

$$f_W(q) = \begin{cases} (1 - \frac{q}{2})^4(1+2q) & 0 \leq q \leq 2 \\ 0 & 2 < q, \end{cases} \tag{3.15}$$

with normalization constants in 1-D, 2-D and 3-D respectively Goffin (2013)

$$\alpha = \frac{3}{4h}\,,\ \alpha = \frac{7}{4\pi h^2}\,,\ \alpha = \frac{21}{16\pi h^3}. \tag{3.16}$$

The plot of the Wendland kernel is found in Figure 3.2 as the solid blue line.

Using the kernel function f, the derivative or gradient of the kernel can be found through

$$\nabla W(\mathbf{x}_i - \mathbf{x}_j) = \frac{\alpha}{h^{n+1}} f'(q), \tag{3.17}$$

The gradients of the above three kernels are given by the following equations:
Gaussian

$$f_G = -2qe^{-q^2}. \tag{3.18}$$

Cubic spline

$$f_3 = \begin{cases} -3q + \frac{9}{4}q^2 & 0 \leq q \leq 1 \\ -\frac{3}{4}(2-q)^2 & 1 < q \leq 2 \\ 0 & 2 < q \end{cases} \tag{3.19}$$

Wendland

$$f_W(q) = \begin{cases} -5q(1 - \frac{q}{2})^3 & 0 \leq q \leq 2 \\ 0 & 2 < q \end{cases} \tag{3.20}$$

For the remainder of the work in this thesis the Wendland kernel will be the kernel of choice unless otherwise stated.

3.2.2 Gradients and Divergences

Using the gradient functions above, operators such as the gradient and divergence can also be expressed in discrete SPH form. Through a similar process as for the function ϕ, the gradient of the scalar function g evaluated at $\mathbf{x}_i$ can be written as

$$\nabla g(\mathbf{x}_i, t) \approx \sum_j \frac{m_j}{\rho_j} g_j \nabla W_{ij}, \tag{3.21}$$

where $W_{ij} = W(\mathbf{x}_i - \mathbf{x}_j)$ and $g_j = f(\mathbf{x}_j, t)$.
The divergence of a vector quantity $\mathbf{d}$ can be written as

$$\nabla \cdot \mathbf{d}(\mathbf{x}_i, t) \approx \sum_j \frac{m_j}{\rho_j} \mathbf{d}_j \cdot \nabla W_{ij}. \tag{3.22}$$

However, these do not correctly approximate constant functions when particles are irregularly spaced as is often the case during a simulation. This can be seen by using the function $g = 1$ in Equation (3.21)

$$\nabla 1 = \sum_j \frac{m_j}{\rho_j} 1 \nabla W_{ij} \neq 0, \tag{3.23}$$

since the mass, density and kernel gradient are all non zero at points in the kernel radius.

In order to obtain accurate gradient and divergence operators other approximations need to be used. Violeau (2012) showed that this is accomplished by taking advantage of the identities

$$\begin{aligned} \nabla g &= \rho^k \nabla \left(\frac{g}{\rho^k} \right) + \frac{g}{\rho^k} \nabla \rho^k \\ \nabla \cdot \mathbf{d} &= \rho^k \nabla \cdot \left(\frac{\mathbf{d}}{\rho^k} \right) + \frac{\mathbf{d}}{\rho^k} \cdot \nabla \left(\rho^k \right), \end{aligned} \tag{3.24}$$

and

$$\begin{aligned} \nabla g &= \frac{1}{\rho^k} \nabla (\rho^k g) - \frac{g}{\rho^k} \nabla \rho^k \\ \nabla \cdot \mathbf{d} &= \frac{1}{\rho^k} \nabla \cdot \left(\rho^k \mathbf{d} \right) - \frac{\mathbf{d}}{\rho^k} \cdot \nabla \left(\rho^k \right), \end{aligned} \tag{3.25}$$

for any real k. Applying the definitions of the SPH gradient and divergence from equations (3.21) and (3.22) to the right hand side of the above identities gives a number of SPH gradient and divergence operators. To help identify these operators in the following, operators derived using Equation (3.24) will be referred to as operators of type (I) and operators derived from Equation (3.25) as type (II). However, in many cases consistency is preferred with type (II) operators used everywhere. This is the case with DualSPHysics and will be the case for gradient and divergence approximations used in this thesis.

Setting $k = 0$ in Equation (3.24) gives

$$\begin{aligned}\nabla g &= \sum_j \frac{m_j}{\rho_j}\left(g_i + g_j\right)\nabla W_{ij} \\ \nabla\cdot\mathbf{d} &= \sum_j \frac{m_j}{\rho_j}\left(\mathbf{d}_i + \mathbf{d}_j\right)\cdot\nabla W_{ij},\end{aligned} \tag{3.26}$$

and setting $k = 1$ gives

$$\begin{aligned}\nabla g &= \sum_j m_j\left(\frac{g_i}{\rho_i^2} + \frac{g_j}{\rho_j^2}\right)\nabla W_{ij} \\ \nabla\cdot\mathbf{d} &= \sum_j m_j\left(\frac{\mathbf{d}_i}{\rho_i^2} + \frac{\mathbf{d}_j}{\rho_j^2}\right)\cdot\nabla W_{ij}.\end{aligned} \tag{3.27}$$

Similarly setting $k = 0$ in Equation (3.25) gives

$$\begin{aligned}\nabla g &= -\sum_j \frac{m_j}{\rho_j}\left(g_i - g_j\right)\nabla W_{ij} \\ \nabla\cdot\mathbf{d} &= -\sum_j \frac{m_j}{\rho_j}\left(\mathbf{d}_i - \mathbf{d}_j\right)\cdot\nabla W_{ij},\end{aligned} \tag{3.28}$$

and setting $k = 1$ gives

$$\begin{aligned}\nabla g &= -\frac{1}{\rho_i}\sum_j m_j\left(g_i - g_j\right)\nabla W_{ij} \\ \nabla\cdot\mathbf{d} &= -\frac{1}{\rho_i}\sum_j m_j\left(\mathbf{d}_i - \mathbf{d}_j\right)\cdot\nabla W_{ij}.\end{aligned} \tag{3.29}$$

It can be noted that the operators of type (II) correctly approximate the gradient and divergence of constant functions due to the presence of the minus sign. The operators of type (I) do not. These operators are still important and useful however, as due to their asymmetry they the conserve total momentum of the system (Violeau (2012)). It is preferable for some applications to use gradient operators of type (I) and divergence operators of type (II). Mayrhofer (2014) showed that the gradient operator of type (I) with $k = 1$ and the divergence operator of type (II) with $k = 1$ are skew-adjoint and therefore conserve energy exactly when used together. The book by Violeau (Violeau (2012)) also shows that for any k that gradients of type (I) and divergences of type (II) and visa versa are skew-adjoint.

3.3 Errors in SPH

When a function is approximated by the SPH interpolation a small amount of error is introduced. Considering again Equations (3.2) (3.3) (3.4) which are written again in 1D

$$\phi(x) \approx \phi(x)_{SPH} = \int_{\Omega_h} \phi(x')W(x-x')d\Omega_h \tag{3.30}$$

$$\int_{\Omega_h} W(x-x')d\Omega_h = 1 \tag{3.31}$$

$$\int_{\Omega_h} (x-x')W(x-x')d\Omega_h = 0. \tag{3.32}$$

The size of the error can be found by evaluating a Taylor series expansion of the function $\phi(x)$ around x gives

$$\phi(x') = \phi(x) + (x'-x)\frac{d\phi(x)}{dx} + \mathcal{O}\left((x'-x)^2\right). \tag{3.33}$$

Combining this with the SPH approximation it can be seen that

$$\begin{aligned}\phi(x)_{SPH} = \phi(x) &\int_{\Omega_h} W(x-x')d\Omega_h \\ &+ \frac{d\phi(x)}{dx}\int_{\Omega_h} (x'-x)W(x-x')d\Omega_h + \mathcal{O}\left((x'-x)^2\right).\end{aligned} \tag{3.34}$$

Taking advantage of the symmetry of the kernel functions, Equations (3.31) and (3.32) can be used giving

$$\phi(x)_{SPH} = \phi(x) + \mathcal{O}\left((x'-x)^2\right). \tag{3.35}$$

The smoothing length h is generally of the same order $(x'-x)$ meaning that

$$\phi(x)_{SPH} = \phi(x) + \mathcal{O}\left(h^2\right). \tag{3.36}$$

Thus the leading order error in the SPH approximation is h^2. SPH is therefore often referred to as second order accurate in space.

However Quinlan et al. (2006) found that this is not quite true, and that the error in SPH also depends on the ratio of particle spacing dp to smoothing length, called the discretisation error, as well as the particle disorder. For a uniform particle distribution Quinlan et al. found that the total error is second order in h and order $\beta + 2$ in the ratio of particle spacing to smoothing length dp/h. The integer β is the boundary smoothness of the kernel, defined as the largest integer such that all derivatives of the kernel up to and including the β^{th} are zero at the edge of the kernel support. This means that by reducing the ratio dp/h while holding h constant the overall error will eventually be controlled by the smoothing error of order h^2. This makes sense as more particles will included in the approximation as this happens and the discretisation tends to the continuous approximation. On the other hand, if holding the ratio constant and decrease the smoothing length will lead to the error becoming dominated by the discretisation error of order $(dp/h)^{\beta+2}$. Thus to get true idea of the rate of convergence a SPH simulation both the ratio of particle spacing to smoothing length and the smoothing length need to decreased together.

In order to find the order of convergence of an SPH simulation a convergence study of error versus particle spacing needs to be conducted. For this thesis the error of a property u compared to the analytical or theoretical solution U will be found using the L_2 norm given by

$$L_2 = \frac{1}{N}\sqrt{\sum_i^N \frac{(U-u)^2}{U^2}}. \tag{3.37}$$

3.4 Fluid Dynamics equations

Using the above knowledge of gradient and divergence operators, the SPH method can now applied to the equations of motion for a fluid flow, the Navier-Stokes equations. For a weakly-compressible flow the Navier-Stokes equations are given by the conservation of mass

$$\frac{d\rho}{dt} + \rho\nabla\cdot\mathbf{u} = 0, \tag{3.38}$$

and the conservation of momentum

$$\frac{d\mathbf{u}}{dt} = -\frac{1}{\rho}\nabla P + \frac{1}{\rho}\nabla\cdot\boldsymbol{\tau} + \mathbf{f}, \tag{3.39}$$

where ρ is the density of the fluid, $\mathbf{u}$ is the fluid velocity, P is the pressure, $\boldsymbol{\tau}$ is the stress tensor and $\mathbf{f}$ is the acceleration due to body forces such as gravity. Pressure is found using an equation of state, such as the Tait equation of state

$$P = \frac{c_0^2 \rho_0}{\gamma} \left[\left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right], \tag{3.40}$$

where ρ_0 is the reference density in the fluid, c_0 is the speed of sound and γ is a fluid specific parameter normally taken for water to be $\gamma = 7$.

3.4.1 SPH Conservation of mass

Rearranging the conservation of mass equation (3.38) for the material derivative gives

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{u}. \tag{3.41}$$

Applying the divergence operator in Equation (3.29) gives

$$\frac{d\rho_i}{dt} = \sum_j m_j \mathbf{u}_{ij} \cdot \nabla W_{ij}, \tag{3.42}$$

where $\mathbf{u}_{ij} = \mathbf{u}_i - \mathbf{u}_j$. This equation is then used to update the density gradient of the fluid particles in the flow domain.

An alternative approximation for the density can be found using the SPH discretisation for a scalar quantity, Equation (3.6)

$$\rho_i = \sum_j \frac{m_j}{\rho_j} \rho_j W_{ij} = \sum_j m_j W_{ij}. \tag{3.43}$$

This approach however is not widely used as near free-surfaces or boundaries the kernel support will be truncated leading to poor results.

These equations for the conservation of momentum are for a weakly-compressible fluid, and assumed in this thesis. The density of the particles is allowed to change by up to 1%, this is done by using a numerical speed of sound chosen to be ten times the maximum flow velocity, $c_0 = 10 \times u_{max}$ (Violeau and Rogers (2016))

3.4.2 SPH Conservation of Momentum

Inviscid Flow Looking first at the inviscid case, the conservation of momentum is given by the equation

$$\frac{d\mathbf{u}}{dt} = -\frac{1}{\rho} \nabla P + \mathbf{f}. \tag{3.44}$$

Using the gradient from Equation (3.26) the pressure gradient is given by

$$\nabla P_i = \sum_j \frac{m_j}{\rho_j} \left(P_i + P_j\right) \nabla W_{ij}, \tag{3.45}$$

this then gives the conservation of momentum equation as

$$\left(\frac{d\mathbf{u}}{dt}\right)_i = -\sum_j m_j \left(\frac{P_i + P_j}{\rho_j \rho_i}\right) \nabla W_{ij} + \mathbf{f}_i. \tag{3.46}$$

Once an appropriate viscosity model has been chosen, these equations can then be used to update the acceleration of the fluid particles.

3.4.3 Viscous terms

The shear stress tensor in the conservation of momentum equation controls the viscous dissipation forces. For a Newtonian incompressible fluid it simplifies to

$$\frac{1}{\rho} \nabla \cdot \boldsymbol{\tau} = \nu \nabla^2 \mathbf{u}. \tag{3.47}$$

The Laplacian term $\nabla^2 = \nabla \cdot \nabla$ is evaluated using the gradient and divergence operators described above would involve using the second derivative of the kernel function W''_{ij}. While the kernel functions are chosen so that the sign of the first derivative is guaranteed Violeau (2012) found that this cannot be said for the the second derivative. Monaghan, J. (1992) also states that ”The second derivative of the kernel is also sensitive to particle disorder”. Another option would be to find the divergence of the gradient using only the first derivatives, but this would require a second summation that would be computationally expensive. Luckily two methods have been developed for Newtonian flows that overcome these issues.

Artificial Viscosity The first approach is the artificial viscosity model of Monaghan, J. (1992). The viscous terms are given by

$$\Pi_{ij} = \begin{cases} \frac{-\alpha \overline{c_{ij}} \mu_{ij}}{\overline{\rho_{ij}}} & \mathbf{u}_{ij} \cdot \mathbf{r}_{ij} < 0 \\ 0 & \mathbf{u}_{ij} \cdot \mathbf{r}_{ij} > 0 \end{cases}, \tag{3.48}$$

where

$$\mu_{ij} = \frac{h \mathbf{u}_{ij} \cdot \mathbf{r}_{ij}}{|r_{ij}|^2 + \eta^2}. \tag{3.49}$$

η is a small number used to avoid singularities and is usually of the form $\eta^2 = 0.01h^2$, $\overline{c_{ij}} = (c_i + c_j)/2$ is the average speed of sound, and $\overline{\rho_{ij}} = (\rho_i + \rho_j)/2$ is the average density. α is the artificial viscous term used to introduce the correct amount of dissipation, it normally take values in the range $[0,1]$.

According to Morris and Monaghan (1997) the Reynolds number associated with a flow using this viscosity treatment is given by

$$Re = \frac{8VL}{\alpha c h}. \tag{3.50}$$

Using the definition for the standard Reynolds number $Re = VL/\nu$ Morris and Monaghan (1997) showed that the artificial viscosity is related to the kinematic viscosity through

$$\nu = \frac{\alpha c h}{8}. \tag{3.51}$$

Including the artificial viscosity terms, the momentum equation (3.46) becomes

$$\left(\frac{d\mathbf{u}}{dt}\right)_i = -\sum_j m_j \left(\frac{P_i + P_j}{\rho_j \rho_i} + \Pi_{ij}\right) \nabla W_{ij} + \mathbf{f}_i. \tag{3.52}$$

The artificial viscosity method was first used to stabilize the computation of flows, especially near shocks, and was not meant to be used to model a physical viscosity Basa et al. (2008).

Laminar Viscosity The second approach is the so called laminar viscosity method developed as an analogue of physical viscosity. Laminar viscosity was first developed by Morris et al. (1997) as an improvement over the artificial viscosity treatment. Laminar viscosity approximates the Laplacian terms in the momentum equation through a first order SPH gradient and then a first order finite difference. This method avoids the troublesome second derivative of the kernel and the costly double sum that would otherwise occur. In this work the formulation of Lo and Shao (2002) is used, it is written as

$$\nu \nabla^2 \mathbf{u}_i = \sum_j m_j \left(\frac{4\nu \mathbf{r}_{ij} \cdot \nabla W_{ij}}{(\rho_i + \rho_j)(r_{ij}^2 + \eta^2)}\right) \mathbf{u}_{ij}. \tag{3.53}$$

In this case the momentum equation becomes

$$\frac{d\mathbf{u}_i}{dt} = -\sum_j m_j \left(\frac{P_i + P_j}{\rho_j \rho_i}\right) \nabla W_{ij} + \sum_j m_j \left(\frac{4\nu \mathbf{r}_{ij} \cdot \nabla W_{ij}}{(\rho_i + \rho_j)(r_{ij}^2 + \eta^2)}\right) \mathbf{u}_{ij} + \mathbf{f}_i. \tag{3.54}$$

The laminar viscosity method is an improvement over artificial viscosity as it can be used to directly represent physical viscosity.

Non-Newtonian SPH For a non-Newtonian fluid there are two approaches to evaluate the viscous forces. The first is to use the laminar viscosity operator of Morris et al. (1997)

$$\left(\frac{1}{\rho}\nabla\cdot\boldsymbol{\tau}\right)_i = -\sum_j \left(\frac{m_j(\nu_i+\nu_j)\mathbf{r}_{ij}\cdot\nabla W_{ij}}{\rho_j(r_{ij}^2+\eta^2)}\right)\mathbf{u}_{ij}. \tag{3.55}$$

This approach uses the viscosities of each of the particles which need to be calculated in advance using a non-Newtonian viscosity model.

The second approach uses a SPH gradient operator directly on the shear stresses (for example Fourtakas and Rogers (2016))

$$\left(\frac{1}{\rho}\nabla\cdot\boldsymbol{\tau}\right)_i = \sum_j m_j\left(\frac{\boldsymbol{\tau}_i+\boldsymbol{\tau}_j}{\rho_i\rho_j}\right)\cdot\nabla W_{ij}. \tag{3.56}$$

This method requires the knowledge of the shear stresses at each particle. The shear stress is found through

$$\boldsymbol{\tau} = \mu_{app}\dot{\boldsymbol{\gamma}}, \tag{3.57}$$

where μ_{app} is the apparent dynamic viscosity found using a non-Newtonian viscosity model, and $\dot{\boldsymbol{\gamma}}$ is the shear rate given by

$$\dot{\boldsymbol{\gamma}} = \left[\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right]. \tag{3.58}$$

This then requires the velocity gradients to be evaluated at each particle, this can be done through either an SPH sum

$$(\nabla\mathbf{u})_i = \sum_j \frac{m_j}{\rho_j}(\mathbf{u}_i-\mathbf{u}_j)\nabla W_{ij} \tag{3.59}$$

or a finite difference approach

$$(\nabla\mathbf{u})_i = \frac{\mathbf{u}_{ij}\mathbf{r}_{ij}}{|r_{ij}|^2} \tag{3.60}$$

the latter of which being the faster option.

3.5 SPH Improvements

Weakly compressible SPH can suffer from noisy pressure fields and non-uniform particle distributions after a simulation has progressed. The methods developed to combat these issues are described below.

3.5.1 Kernel corrections and density filters

In some cases for example near free surfaces or boundaries the kernel support of a particle can be incomplete. This incomplete kernel can cause errors in the solution. Considering Equation (3.34), an incomplete kernel would lead to integrals that do not satisfy the zeroth and first order moments in Equations (3.31) and (3.32). These integrals would then introduce zeroth and first order errors to the continuous interpolation. This can be corrected using a kernel correction that acts to renormalise the kernel and give a better interpolation.

Shepard filter One example is the Shepard filter (described in Gomez-Gesteira et al. (2009) for example) which corrects the kernel through

$$W_{ij}^{Shep} = \frac{W_{ij}}{\sum_j \frac{m_j}{\rho_j} W_{ij}}. \tag{3.61}$$

This method is quick and simple to use as it only requires summing over the surrounding particles. It is however only zeroth order consistent meaning it can only accurately reproduce constant functions. Bilotta et al. (2011) noted that while the Shepard filter can give a lower minimum error than standard SPH, it can also give a higher maximum error for non-uniform fields.

The Shepard filter can be used to help smooth out a noisy density field. It is used after an interaction loop to re-assign density to particles through

$$\rho_i^{Shep} = \sum_j m_j W_{ij}^{Shep}. \tag{3.62}$$

Gomez-Gesteira et al. (2009) recommends that this correction should only be used every m(=20-50) time steps in order to avoid non physical smoothing of the flow.

MLS kernel Another kernel correction method corrects the kernel using a Moving least squares (MLS) approach and has been use by Colagrossi and Landrini (2003) for example. The corrected kernel takes the form in 2D

$$W_{ij}^{MLS} = \left[\beta_0 + \beta_1 x_{ij} + \beta_2 y_{ij}\right] W_{ij}, \tag{3.63}$$

where

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} = \mathbf{A}^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \tag{3.64}$$

$$\mathbf{A} = \begin{bmatrix} \sum_j W_{ij} & \sum_j x_{ij} W_{ij} & \sum_j y_{ij} W_{ij} \\ \sum_j x_{ij} W_{ij} & \sum_j x_{ij}^2 W_{ij} & \sum_j x_{ij} y_{ij} W_{ij} \\ \sum_j y_{ij} W_{ij} & \sum_j x_{ij} y_{ij} W_{ij} & \sum_j y_{ij}^2 W_{ij} \end{bmatrix}. \tag{3.65}$$

The MLS kernel correction is more complicated and expensive than the Shepard kernel as it involves inverting the matrix **A** for each particle. It is however first order consistent meaning it can exactly reproduce linear functions.

The MLS kernel can also be used to smooth out the density field in the same way as the Shepard filter

$$\rho_i^{MLS} = \sum_j m_j W_{ij}^{MLS}. \tag{3.66}$$

This correction should also only be used every m(=20-50) time steps as recommended by Gomez-Gesteira et al. (2009).

Both of these kernel corrections can be used in ways other than as density filter, for example in boundary conditions as will be seen later.

3.5.2 Density diffusion

As the density of the particles is updated and allowed to change at each time step, oscillations of the fluid density and therefore pressure are know to occur in weakly compressible SPH. One method used to combat the noise in the pressure field is to include a density diffusion term in the continuity equation. In the original approach of Molteni and Colagrossi (2009) the diffusion term takes the form

$$D_i = \delta h c_0 \sum_j \frac{m_j}{\rho_j} \psi_{ij} \cdot \nabla W_i j, \tag{3.67}$$

where ψ_{ij} is given by

$$\psi_{ij}^{MC} = 2 \left(\rho_j - \rho_i\right) \frac{\mathbf{x}_{ij}}{||\mathbf{x}_{ij}||}. \tag{3.68}$$

This diffusion term is added onto the end of the continuity equation in a similar way to artificial viscosity in the momentum equation. The δ term controls the magnitude of the diffusion applied, for the paper of Molteni and Colagrossi (2009) a value of less than $\delta = 0.1$ is used. Using this density diffusion term helps to reduce spurious oscillations in the density and therefore pressure field of a flow without the use of a filtering method.

Antuono et al. (2010) developed a higher order correction for the density diffusion term following on from the work of Molteni and Colagrossi. Their higher order method replaces the ψ function of Molteni and Colagrossi with the following

$$\psi_{ij}^{A} = \left(\rho_j - \rho_i - \frac{1}{2} \left(\langle \nabla \rho \rangle_j^{MLS} + \langle \nabla \rho \rangle_i^{MLS} \right) \cdot \mathbf{x}_{ij} \right) \frac{\mathbf{x}_{ij}}{||\mathbf{x}_{ij}||}, \tag{3.69}$$

where $\langle \nabla \rho \rangle^{MLS}$ is a renormalised density gradient evaluated using an MLS correction. This approach has benefits over the original method of Molteni and Colagrossi as the MLS corrected terms ensure consistency near the free surface. This advantage is highlighted in cases such still water where a method without the MLS correction the fluid particles near the free surface tend to diffuse upwards in an un-physical manner. With the MLS correction this does not happen.

Fourtakas et al. (2019) developed a new version of the density diffusion term, this time using the same approach as Molteni and Colagrossi but replacing the particle density used in the original with the dynamic density. The dynamic density is given by

$$\rho^D = \rho^T - \rho^H, \tag{3.70}$$

where the superscripts D, T and H denote the dynamic, total and hydrostatic densities respectively. In Fourtakas et al. the ψ function takes the form

$$\psi_{ij}^{F} = 2 \left(\rho_{ji}^{D} \right) \frac{\mathbf{x}_{ij}}{||\mathbf{x}_{ij}||} = 2 \left(\rho_{ji}^{T} - \rho_{ij}^{H} \right) \frac{\mathbf{x}_{ij}}{||\mathbf{x}_{ij}||}. \tag{3.71}$$

The hydrostatic density is found using the hydrostatic pressure as

$$\rho_{ij}^{H} = \rho_0 \left(\sqrt[\gamma]{\frac{P_{ij}^{H} + 1}{C_B}} - 1 \right), \tag{3.72}$$

where $P_{ij}^{H} = \rho_0 g z_{ij}$ is the difference in hydrostatic pressure between the two particles, $C_B = c_0^2 \rho_0 / \gamma$ and $\gamma = 7$. The method of Fourtakas et al. also ensures consistency near free surfaces and stops un-physical diffusion of the fluid, but avoids the use of the MLS correction and therefore the matrix inversion needed for every particle. However, as the method includes acceleration due to gravity it can only be used for gravity driven flows. When gravity is not present the methods reverts back to the original approach of Molteni and Colagrossi (2009).

In this thesis the approaches of Molteni and Colagrossi (2009) and Fourtakas et al. (2019) will be used

3.5.3 Shifting

As the SPH particles are Lagrangian and a free to move they can sometimes clump together. This then causes errors in the simulation and in some cases can cause the simulation to crash. To combat this Lind et al. (2012) developed a technique of moving particles a small distance to avoid this clumping, called shifting. Once fluid particles have been advanced in time they are shifted so to create a more uniform particle distribution. The magnitude and direction in which a particle is shifted is controlled by Fick's law and shifts the particles from areas of high concentration to areas of low concentration.

$$\mathbf{J} = -\mathcal{D}'\nabla C, \tag{3.73}$$

where $\mathbf{J}$ is the flux, C is the particle concentration and $\mathcal{D}'$ is the diffusion coefficient. This flux is proportional to a fluid particle's velocity meaning a shifting velocity, and therefore a shifting distance, can be found for the fluid particle. The shifting distance for a particle i is defined as

$$\delta\mathbf{r}_s = -\mathcal{D}\nabla C, \tag{3.74}$$

where $\mathcal{D}$ is the shifting coefficient which has absorbed the diffusion coefficient $\mathcal{D}'$ and the constant of proportionality between (3.73) and (3.74). Lind et al. (2012) suggest that the diffusion coefficient takes a value of

$$\mathcal{D} = \frac{0.5h^2}{\Delta t}. \tag{3.75}$$

In SPH form the concentration gradient at each particle can be found using

$$\nabla C_i = \sum_j \frac{m_j}{\rho_j}\nabla W_{ij}. \tag{3.76}$$

Lind et al. (2012) found that in some cases clumping still occurred, and so they incorporated a repulsive force from Monaghan to further reduce the clumping

$$\nabla C_i = \sum_j \frac{m_j}{\rho_j}\left(1 + f_{ij}\right)\nabla W_{ij}, \tag{3.77}$$

where the repulsive force is given by

$$f_{ij} = R\left(\frac{W_{ij}}{W(dp)}\right)^n, \tag{3.78}$$

taking suggested values of $R = 0.2$ and $n = 4$. Skillen et al. (2013) suggest a alternative equation for the shifting coefficient

$$\mathcal{D} = Ah||\mathbf{u}||_i \Delta t. \tag{3.79}$$

In this equation A is a dimensionless constant that is "independent of the problem set-up and discretisation" (Skillen et al. (2013)), and a value of $A = 2$ is suggested. This Skillen form for the shifting coefficient will be used in this thesis. Care needs to be taken when using shifting when simulating violent free surface flow that the shifting distance does not exceed the smoothing length. This would shift the particles too far causing errors. A limit on the shifting distance is therefore set as

$$|\delta \mathbf{r}_s| \leq 0.2h. \tag{3.80}$$

3.6 DualSPHysics

DualSPHysics Crespo et al. (2015) is an accurate, highly optimized open-source SPH code developed jointly by the University of Vigo, the University of Manchester, the University of Parma, the University of Lisbon, the Polytechnic University of Barcelona and the New Jersey Institute of Technology. The code is written in C++ and CUDA to allow simulations to be run on both CPUs and NVIDIA GPUs. The CPU code is written in parallel using OpenMP allowing for simulations to be run quickly. The GPU version contains all the same features as the CPU version, and allows for simulations to be run much quicker than the CPU version on single desktops fitted with GPUs.

The work in this thesis was completed using v5.0 of the code with a few modifications. In the following some of the features of the code will be described and summarized.

3.6.1 Formulation

Version v5.0 of DualSPHysics uses a combination of Equation (3.42) for the conservation of mass and Equation (3.46) for the conservation of momentum.

$$\frac{d\rho_i}{dt} = \sum_j m_j \mathbf{u}_{ij} \cdot \nabla W_{ij}, \tag{3.81}$$

$$\left(\frac{d\mathbf{u}}{dt}\right)_i = -\sum_j m_j \left(\frac{P_i + P_j}{\rho_j \rho_i}\right) \nabla W_{ij} + \mathbf{f}_i. \tag{3.82}$$

3.6.2 Time Stepping

The time stepping scheme used in this work is the symplectic predictor corrector scheme Leimkuhler and Matthews (2015) as it is included in the DualSPHysics code. During the predictor step the acceleration $\mathbf{F}_i$ and density gradient D_i are found at each particle. The predicted particle position, velocity and density are then found at the middle of the time step

$$\mathbf{r}_i^{n+1/2} = \mathbf{r}_i^n + \frac{\Delta t}{2}\mathbf{u}_i^n \tag{3.83}$$

$$\mathbf{u}_i^{n+1/2} = \mathbf{u}_i^n + \frac{\Delta t}{2}\mathbf{F}_i^n \tag{3.84}$$

$$\rho_i^{n+1/2} = \rho_i^n + \frac{\Delta t}{2}D_i^n. \tag{3.85}$$

During the corrector step the intermediate acceleration is used to find the corrected velocity and position

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \Delta t\mathbf{F}_i^{n+1/2} \tag{3.86}$$

$$\mathbf{r}_i^{n+1} = \mathbf{r}_i^{n+1/2} + \Delta t\frac{\left(\mathbf{u}_i^n + \mathbf{u}_i^{n+1}\right)}{2}. \tag{3.87}$$

The corrected density is then found according to Parshikov et al. (2000) as

$$\rho_i^n = \rho_i^n + \frac{2-\varepsilon_i^{n+1/2}}{2+\varepsilon_i^{n+1/2}} \tag{3.88}$$

$$\varepsilon_i^{n+1/2} = -\left(\frac{D_i^{n+1/2}}{\rho_i^{n+1/2}}\right)\Delta t. \tag{3.89}$$

3.6.3 Time Step Size

In the DualSPHysics code a variable time step is used. The time step used is defined by

$$\Delta t = CFL \min\left(\Delta t_1, \Delta t_2\right), \tag{3.90}$$

where

$$\Delta t_1 = \min_i\left(\left[\frac{h}{|f_i|}\right]^{1/2}\right), \tag{3.91}$$

$$\Delta t_2 = \min_i\left(h/\left[c_s + \max_j\left|\frac{h\mathbf{u}_{ij}\cdot\mathbf{r}_{ij}}{r_{ij}^2}\right|\right]\right), \tag{3.92}$$

where f_i is the force per unit mass of the particle i. The smallest of these time steps is found and then multiplied by the Courant-Friedrich-Levy (CFL) number to give the final time step, with a typical range of $0.1-0.5$. If a non-Newtonian model is used a third time step is introduced involving the non-Newtonian dynamic viscosity μ

$$\Delta t_3 = \frac{h^2}{\max_i(\mu_i)}. \tag{3.93}$$

In this case the time step is found as the minimum of the three calculated time steps

$$\Delta t = CFL \min\left(\Delta t_1, \Delta t_2, \Delta t_3\right). \tag{3.94}$$

3.7 Boundary Conditions

3.7.1 Solid Boundaries

There are three main approaches to implementing solid boundary conditions in SPH. The first is through repulsive forces such as those used in Monaghan (1994) and Monaghan and Kajtar (2009) in which fluid particles feel a repulsive force from boundary particles when they get close to a boundary. The second type are the semi-analytical boundary conditions by Ferrand et al. (2013). The third type uses particles to fill the solid region beyond the fluid domain thus completing the kernel of fluid particles with boundary particles when close to a boundary. Three widely used boundary conditions of the third types will be the focus of the following section. As justified in the literature review these are more directly relevant to the types of flows of interest in this thesis, as well as the development of the new boundary conditions that will be presented in the next Chapter. These boundary conditions are the Dynamic Boundary Conditions of Dalrymple and Knio (2001), and the methods of Adami et al. (2012) and Marrone et al. (2011).

3.7.1.1 Dynamic Boundary Particles

The first approach the Dynamic Boundary Condition (DBC) was first introduced by Dalrymple and Knio (2001) and later studied by Crespo et al. (2007). To implement the boundary condition layers of boundary particles are created in the solid outside the fluid domain, these particles then complete the kernel of fluid particles close to the boundary. The density, and therefore pressure, of the boundary particles is updated in the same way as the surrounding fluid particles using the density gradient, Equation (3.42). The velocity and position however are updated according to boundary movement, so for a stationary boundary the particles have zero velocity. This results in a zeroth order consistent boundary condition. The approach makes it very easy to create complex shaped boundaries by creating extra layers of boundary particles to fill in gaps of irregular solid shapes. DBC is also the standard boundary condition used in the DualSPHysics code (Crespo et al. (2015)).

3.7.1.2 Adami et al.

The second approach is that of Adami et al. (2012). Again layers of boundary particles are placed outside the fluid domain to provide the kernel support of a nearby fluid particle. However in the approach of Adami the boundary particles are given pressure, density and velocity data extrapolated from the fluid. During each time step a sum of the surrounding fluid particle pressure and velocities is calculated and corrected using a Shepard filter. Due to the use of the Shepard filter, the Adami boundary condition is zeroth order consistent.

The velocity of a boundary particle b is found by summing around the boundary particle using

$$\tilde{\mathbf{u}}_b = \frac{\sum_{j \in fluid} \mathbf{u}_j W_{bj}}{\sum_{j \in fluid} W_{bj}}, \tag{3.95}$$

and the pressure of the boundary particle is found using

$$p_b = \frac{\sum_{j \in fluid} p_j W_{bj} + (\mathbf{g} - \mathbf{a}_w) \cdot \sum_{j \in fluid} \rho_j \mathbf{r}_{bj} W_{ij}}{\sum_{j \in fluid} W_{bj}}, \tag{3.96}$$

where $\mathbf{a}_w$ is the acceleration of the wall. The boundary particle is then assigned the calculated pressure and the velocity is given by

$$\mathbf{u}_b = 2\mathbf{U}_w - \tilde{\mathbf{u}}_b, \tag{3.97}$$

where $\mathbf{U}_w$ is the physical velocity of the boundary, if the boundary is not moving then this is zero. The density of the boundary particle is found using the pressure and rearranging the equation of state

$$\rho_b = \rho_0 \left(p_b \frac{\gamma}{\rho_0 c_0^2} + 1 \right)^{1/\gamma}. \tag{3.98}$$

The velocities, pressures and densities found are then used in the conservation of momentum equation for fluid particles close to the boundary.

3.7.1.3 Marrone et al.

The third method is that of Marrone et al. (2011). In this method the boundary interface is represented by particles, and the normals to the boundary calculated at these particle locations. Boundary particles are then created along these normals pointing away from the fluid in layers with the first layer half a particle spacing from the boundary interface. These layers of particles are the used as the boundary particles. Each boundary particle b is then given a unique ghost node g that is mirrored into the fluid across the boundary interface in the direction of the normal. Fluid properties such as pressure and velocity are found at these ghost nodes and extrapolated back to boundary particle.

The velocity and pressure are found at the ghost node by summing over the surrounding fluid particles, this summation is corrected using an MLS kernel, ensuring a first order consistent boundary condition .

$$\mathbf{u}_g = \sum_j \frac{m_j}{\rho_j} \mathbf{u}_j W_{gj}^{MLS}, \tag{3.99}$$

and the pressure is given by

$$p_g = \sum_j \frac{m_j}{\rho_j} p_j W_{gj}^{MLS}. \tag{3.100}$$

The boundary particles are then given the velocities and pressures given by

$$\mathbf{u}_b = 2\mathbf{U}_w - \mathbf{u}_g, \tag{3.101}$$

where $\mathbf{U}_w$ is the physical velocity of the boundary, and

$$p_b = p_g + 2d\rho\mathbf{g} \cdot \mathbf{n}_{out}, \tag{3.102}$$

where d is the distance between the ghost node and the physical boundary, and $\mathbf{n}_{out}$ is the normal to the physical boundary pointing out of the fluid. The densities of the boundary particles are again found by using the equation of state. The velocities and pressures found are then used in both the conservation of mass and momentum equations of the fluid particles close to the boundary.

3.7.2 Open Boundaries

Another type of boundary condition is the open boundary condition. These occur where fluid may flow out of or into a region of interest. For many flow such are Poiseuille flow or Couette flow a periodic boundary condition will work very well. In this case as a fluid particle leaves one side of a fluid domain, it is transported so that it re-enters on the other side of the domain. Ghost versions of particles close to the open boundaries are generated and moved to the other end of the domain in to form a buffer zone outside the open boundary and complete the kernel support of the fluid particles close to the open boundary. These ghost particles have the same properties as the real fluid particles.

However for some more complex cases such as flow past a cylinder periodic conditions are not appropriate as the flow into one side of the domain will be different to the flow leaving the other end. For these kinds of flows inlet-outlet boundaries need to be used.

3.7.2.1 Inlet-Outlet boundaries

Tafuni et al. (2018) developed an inlet-outlet boundary condition in DualSPHysics. A buffer zone is defined next to an open boundary and filled with buffer particles. These buffer particles help to complete the kernel support of fluid particles close to the open boundary. If a buffer particle flows into the fluid domain it becomes a fluid particle and is treated as such, and if a fluid particle enters a buffer zone it becomes a buffer particle and follows the laws of the buffer zone. New buffer particles are created as particles enter the fluid domain and can be deleted if they leave the computational domain through the other side of the buffer zone.

Physical properties in the buffer zone such as density and velocity can either be assigned values or extrapolated from the fluid domain. When extrapolation is used each buffer particle is assigned a ghost node that has been mirrored across the open boundary into the fluid domain in a similar fashion to Marrone et al. (2011). At this ghost node corrected kernel and kernel gradient sums are calculated for the density and velocity using a method by Liu and Liu (2006) that ensures first order kernel and particle consistency. The corrected sum and gradient for a function f at the ghost node g are given by

$$f_g = \frac{\begin{vmatrix} \sum_j V_j f_j W_{gj} & \sum_j V_j(\mathbf{x}_j - \mathbf{x}_g)W_{gj} \\ \sum_j V_j f_j \nabla_\beta W_{gj} & \sum_j V_j(\mathbf{x}_j - \mathbf{x}_g)\nabla_\beta W_{gj} \end{vmatrix}}{\begin{vmatrix} \sum_j V_j f(\mathbf{x}) W_{gj} & \sum_j V_j(\mathbf{x}_j - \mathbf{x}_g)W_{gj} \\ \sum_j V_j f(\mathbf{x}) \nabla_\beta W_{gj} & \sum_j V_j(\mathbf{x}_j - \mathbf{x}_g)\nabla_\beta W_{gj} \end{vmatrix}}, \tag{3.103}$$

and

$$f_{g,\beta} = \frac{\begin{vmatrix} \sum_j V_j W_{gj} & \sum_j V_j f_j W_{gj} \\ \sum_j V_j \nabla_\beta W_{gj} & \sum_j V_j f_j \nabla_\beta W_{gj} \end{vmatrix}}{\begin{vmatrix} \sum_j V_j W_{gj} & \sum_j V_j(\mathbf{x}_j - \mathbf{x}_g)W_{gj} \\ \sum_j V_j \nabla_\beta W_{gj} & \sum_j V_j(\mathbf{x}_j - \mathbf{x}_g)\nabla_\beta W_{gj} \end{vmatrix}}, \tag{3.104}$$

where β represents the chose derivative (eg. x, y or z). The value given to the buffer particle b is then found through

$$f_b = f_k + (\mathbf{r}_b - \mathbf{r}_g) \cdot \tilde{\nabla} f_k, \tag{3.105}$$

where $\tilde{\nabla} f_k$ is the corrected gradient of f calculated at the ghost node.

3.8 Summary

In this Chapter the Weakly Compressible SPH method has been introduced. The formulation used in the SPH solver DualSPHysics that will be used in this thesis have also be presented along with the state-of-the-art improvements of the SPH method. The details of three widely used and effective SPH boundary conditions, and of an inlet-outlet boundary condition have been described. Using this knowledge two new boundary conditions will be presented in Chapter 4.

Chapter 4

New Boundary Conditions

In this Chapter two new boundary conditions developed during this PhD are presented. The first is the Modified Dynamic Boundary Condition (MDBC) which is an improvement over the existing DBC method used in DualSPHysics. The second is a partial slip boundary condition that can be used for small fluid slip near solid boundaries. In each section the new methods of the boundary conditions are presented and tested and validated using benchmark and analytical solutions.

4.1 Modified Dynamic Boundary Condition

The current method included in the DualSPHysics code is the DBC method. Using the DBC method it is very easy to create complex shaped geometries, for example the work of Altomare et al. (2014) who recreated sea breakwaters made of a large number of individual blocks. DBC are very robust and has been used for a wide range of applications. However there are a number of drawbacks that can limit application of the method. The density and pressure of boundary particles can be noisy resulting in noisy density in the fluid close to the boundary. Another drawback is a non-physical gap that appears between the fluid and boundary when previously unsubmerged boundary becomes submerged as shown by Domínguez et al. (2015). In order to obtain surface pressure measurements numerical pressure probes must be placed a short distance of the order of h inside the fluid domain rather than at the real location on the boundary. These drawbacks are shown in Figure 4.1. In order to combat these issues the new approach MDBC has been developed.

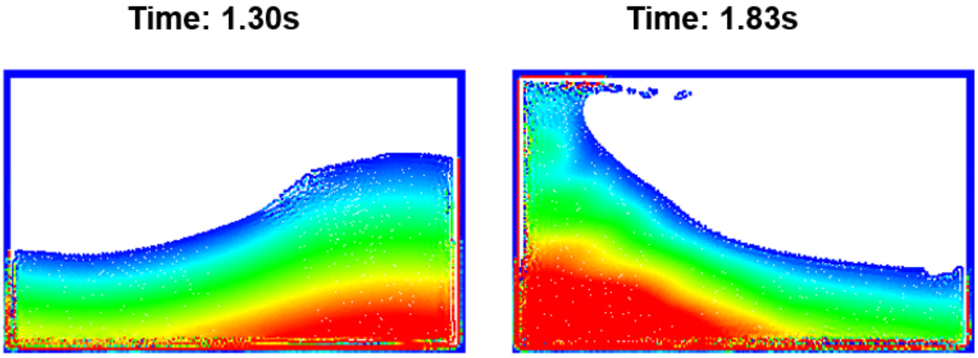


Figure 4.1: Example fo a sloshing tank coloured by pressure highlighting the drawbacks of the DBC method including a noisy pressure field in the boundary and fluid close to the boundary (left) and the presence of a gap between the fluid and boundary particles (right).

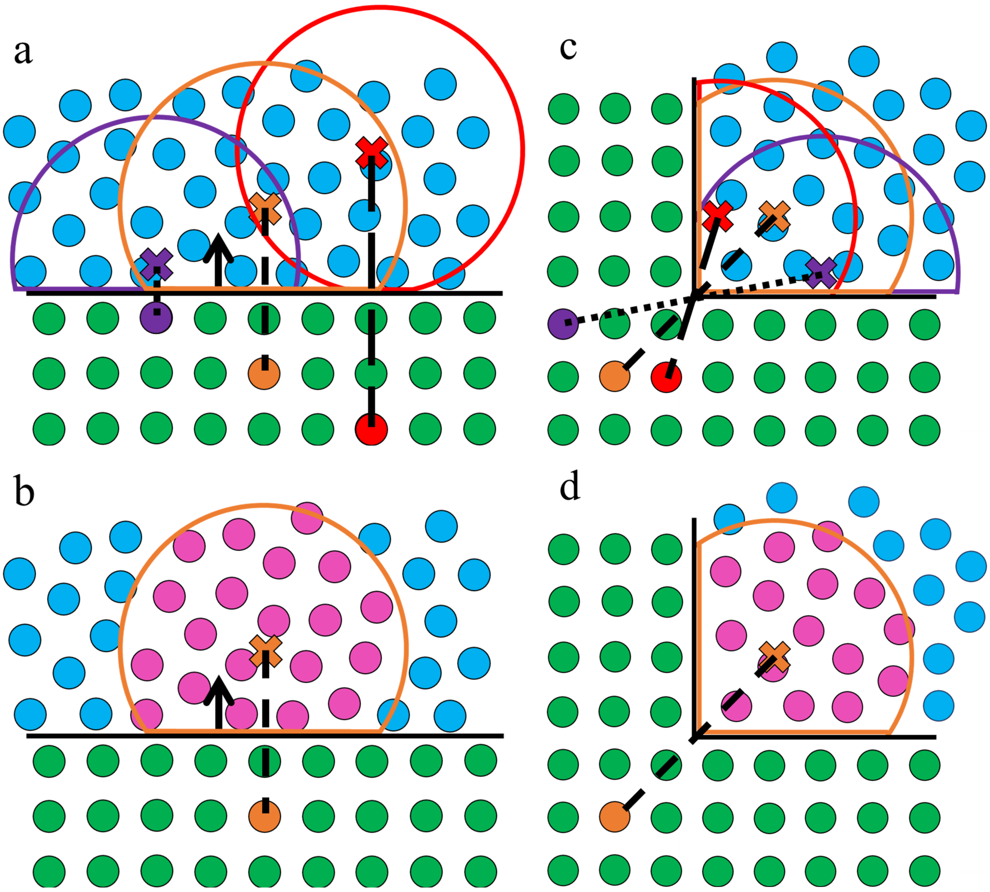


Figure 4.2: Examples of ghost node locations and support kernels around them. (a) Ghost node locations for boundary particles in a flat boundary at three depths into the boundary, the ghost nodes positions are found by mirroring the boundary particle across the boundary surface (b) the fluid particles included in the summation around the ghost node (c) Ghost node locations for boundary particles in boundary corner, the ghost node positions are found by mirroring the boundary particle through the corner of the boundary surface (d) the fluid particles included in the summation around the ghost node.

4.1.1 Method

The method of MDBC is similar to that of Tafuni et al. (2018) for the open boundaries, but applied to solid boundaries. The first step is to create ghost nodes for each boundary particle that are mirrored across the boundary surface into the fluid in the same way as Marrone et al. (2011), as shown in Figure 4.2. Fluid properties are then found at the ghost node and extrapolated back to the boundary particle.

4.1.1.1 Density

In $2D$ the density of the boundary particle is found using a Taylor series between the boundary particle b and the ghost node g

$$\rho_b = \rho_g + (x_b - x_g, y_b - y_g) \cdot (\partial\rho_g/\partial x, \partial\rho_g/\partial y). \tag{4.1}$$

This requires the use of corrected kernel sum and kernel gradient functions at the ghost node due to the truncated kernel near the boundary. To find these functions the method of Liu and Liu (2006) is applied. This requires the use of a second Taylor series expansion around the ghost node. For a general function f the Taylor series takes the form

$$f(\mathbf{x}) = f_g + (x - x_g)\frac{\partial f_g}{\partial x} + (y - y_g)\frac{\partial f_g}{\partial y} + ... \tag{4.2}$$

Ignoring the higher order derivatives and multiplying by the kernel and its derivatives and then integrating over the fluid domain three equations are formed

$$\int f(\mathbf{x})W d\mathbf{x} = f_g \int W d\mathbf{x} + \frac{\partial f_g}{\partial x}\int (x - x_g)W d\mathbf{x} + \frac{\partial f_g}{\partial y}\int (y - y_g)W d\mathbf{x} \tag{4.3}$$

$$\int f(\mathbf{x})\frac{\partial W}{\partial x} d\mathbf{x} = f_g \int \frac{\partial W}{\partial x} d\mathbf{x} + \frac{\partial f_g}{\partial x}\int (x - x_g)\frac{\partial W}{\partial x} d\mathbf{x} + \frac{\partial f_g}{\partial y}\int (y - y_g)\frac{\partial W}{\partial x} d\mathbf{x} \tag{4.4}$$

$$\int f(\mathbf{x})\frac{\partial W}{\partial y} d\mathbf{x} = f_g \int \frac{\partial W}{\partial y} d\mathbf{x} + \frac{\partial f_g}{\partial x}\int (x - x_g)\frac{\partial W}{\partial y} d\mathbf{x} + \frac{\partial f_g}{\partial y}\int (y - y_g)\frac{\partial W}{\partial y} d\mathbf{x}. \tag{4.5}$$

In SPH form these equations become

$$\sum_j V_j f_j W_{gj} = f_g \sum_j V_j W_{gj} + \frac{\partial f_g}{\partial x}\sum_j (x_j - x_g) V_j W_{gj} + \frac{\partial f_g}{\partial y}\sum_j V_j (y_j - y_g) W_{gj} \tag{4.6}$$

$$\sum_j V_j f_j \frac{\partial W_{gj}}{\partial x} = f_g \sum_j V_j \frac{\partial W_{gj}}{\partial x} + \frac{\partial f_g}{\partial x}\sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial x} + \frac{\partial f_g}{\partial y}\sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial x} \tag{4.7}$$

$$\sum_j V_j f_j \frac{\partial W_{gj}}{\partial y} = f_g \sum_j V_j \frac{\partial W_{gj}}{\partial y} + \frac{\partial f_g}{\partial x}\sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial y} + \frac{\partial f_g}{\partial y}\sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial y}, \tag{4.8}$$

where the j particles are the fluid particles surrounding the ghost node. This then forms a linear system of 3 equations in the 3 unknowns f_g, $\partial f_g/\partial x$ and $\partial f_g/\partial y$. Writing this as a linear system of the form $\mathbf{Ax} = \mathbf{b}$ the following is obtained

$$\begin{bmatrix} \sum_j V_j W_{gj} & \sum_j (x_j - x_g) V_j W_{gj} & \sum_j V_j (y_j - y_g) W_{gj} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial x} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial x} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial x} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial y} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial y} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial y} \end{bmatrix} \begin{bmatrix} f_g \\ \partial f_g/\partial x \\ \partial f_g/\partial y \end{bmatrix} = \begin{bmatrix} \sum_j V_j f_j W_{gj} \\ \sum_j V_j f_j \frac{\partial W_{gj}}{\partial x} \\ \sum_j V_j f_j \frac{\partial W_{gj}}{\partial y} \end{bmatrix}. \tag{4.9}$$

This can be solved by inverting the matrix. The values of f_g, $\partial f_g/\partial x$ and $\partial f_g/\partial y$ are then given by

$$f_g = \frac{det\left(\begin{bmatrix} \sum_j V_j f_j W_{gj} & \sum_j (x_j - x_g) V_j W_{gj} & \sum_j V_j (y_j - y_g) W_{gj} \\ \sum_j V_j f_j \frac{\partial W_{gj}}{\partial x} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial x} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial x} \\ \sum_j V_j f_j \frac{\partial W_{gj}}{\partial y} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial y} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial y} \end{bmatrix}\right)}{det\left(\begin{bmatrix} \sum_j V_j W_{gj} & \sum_j (x_j - x_g) V_j W_{gj} & \sum_j V_j (y_j - y_g) W_{gj} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial x} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial x} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial x} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial y} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial y} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial y} \end{bmatrix}\right)}, \tag{4.10}$$

$$\partial f_g/\partial x = \frac{det\left(\begin{bmatrix} \sum_j V_j W_{gj} & \sum_j V_j f_j W_{gj} & \sum_j V_j (y_j - y_g) W_{gj} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial x} & \sum_j V_j f_j \frac{\partial W_{gj}}{\partial x} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial x} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial y} & \sum_j V_j f_j \frac{\partial W_{gj}}{\partial y} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial y} \end{bmatrix}\right)}{det\left(\begin{bmatrix} \sum_j V_j W_{gj} & \sum_j (x_j - x_g) V_j W_{gj} & \sum_j V_j (y_j - y_g) W_{gj} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial x} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial x} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial x} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial y} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial y} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial y} \end{bmatrix}\right)}, \tag{4.11}$$

$$\partial f_g/\partial y = \frac{det\left(\begin{bmatrix} \sum_j V_j W_{gj} & \sum_j (x_j - x_g) V_j W_{gj} & \sum_j V_j f_j W_{gj} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial x} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial x} & \sum_j V_j f_j \frac{\partial W_{gj}}{\partial x} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial y} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial y} & \sum_j V_j f_j \frac{\partial W_{gj}}{\partial y} \end{bmatrix}\right)}{det\left(\begin{bmatrix} \sum_j V_j W_{gj} & \sum_j (x_j - x_g) V_j W_{gj} & \sum_j V_j (y_j - y_g) W_{gj} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial x} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial x} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial x} \\ \sum_j V_j \frac{\partial W_{gj}}{\partial y} & \sum_j (x_j - x_g) V_j \frac{\partial W_{gj}}{\partial y} & \sum_j V_j (y_j - y_g) \frac{\partial W_{gj}}{\partial y} \end{bmatrix}\right)}, \quad (4.12)$$

where the denominator in each of the equations is the determinant of the matrix in Equation (4.9). Substituting the density ρ for the function f, these corrected sums and gradients can be used in the Taylor series in Equation (4.1). Similar expressions have been extended to $3D$.

Using this approach ensures both first order kernel consistency and particle consistency (Liu and Liu (2006)). However, if the matrix in Equation (4.9) is singular, which may happen when there are few fluid particles close to the ghost node, then the denominators in the above equations would become zero. In this case the density at the ghost node is calculated through

$$\rho_g = \frac{\sum_j V_j \rho_j W_{gj}}{\sum_j V_j W_{gj}}, \quad (4.13)$$

and the boundary particle is given this density. The pressure of the boundary particles is then found using the equation of state.

This density extrapolation method can be used while setting the velocity to zero as was done for the original DBC, however some cases may need a more accurate no-slip condition. For these cases a further velocity extrapolation can also be used.

4.1.1.2 Velocity

The velocity extrapolation process for no-slip is much simpler. The velocity at the ghost node is found through

$$\mathbf{u}_g = \frac{\sum_j V_j \mathbf{u}_j W_{gj}}{\sum_j V_j W_{gj}}. \quad (4.14)$$

The boundary particle velocity is updated via

$$\mathbf{u}_b = 2\mathbf{u}_w - \mathbf{u}_g, \quad (4.15)$$

where $\mathbf{u}_w$ is the velocity of the boundary if it is moving. This velocity extrapolation is zeroth order consistent, similar to the method of Adami et al. (2012).

4.1.1.3 Implementation

The normals are generated for each of the boundary particles at the beginning of a simulation. During the creation of the geometry in DualSPHysics both the boundary particles and the boundary interface (the black lines in Figure 4.2) are created. The normals are then defined as the vector pointing from the boundary particle to the nearest point on the boundary interface line, the normal distance is then doubled to define the ghost node. Alternatively, the normals for blocks of boundary particles can be defined when the geometry is created, such as for flat boundaries seen in Poiseuille and Couette flows. If the boundary is set to move during the simulation the new normals are computed at the start of each time step by applying the same movement to the normals as is applied to the boundary particle (e.g. rotation or translation). This ensures the normal still points from the boundary particle to its ghost node.

In each time step the MDBC method should be used at the start of the step before any fluid interactions. When used with the predictor corrector scheme currently implemented in DualSPHysics, MDBC is calculated before the predictor step only. This avoids needing to redefine the location of ghost nodes twice during each time step as well as the associated matrix inversions. The error with respect to time step will however be reduced to first order compared to the second order time step scheme used for the fluid.

4.1.2 Results

In this section a number of test cases will be used for validation of the MDBC method.

4.1.2.1 Still Water

The first test case used is the still water test case with a wedge. A 2D still water tank with dimensions of 2.4 × 1.2m encloses a triangle shaped wedge of height 0.24m in the bottom centre of the tank. The initial water height is H=0.5m. The simulation is run for 4 seconds of physical time for three particle spacings, the numerical parameters of the simulations are displayed in Table 4.1. This test case while looking simple can be challenging due to the corners created by the wedge. The MDBC approach used was for the density extrapolation only, since the velocity in the fluid is zero. Shifting is not used for this test case as the movement of particles in still water would increase errors in the kinetic energy.

The final instant is shown in Figure 4.3 for the two finer resolutions, dp = 0.02m

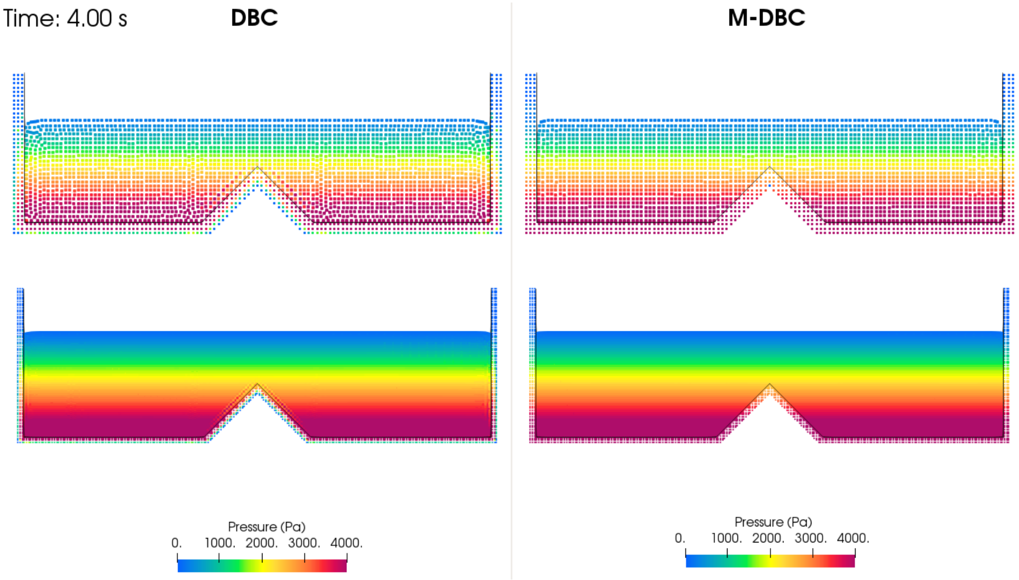


Figure 4.3: Pressure field in still water tank comparing DBC (left) and MDBC (right) for two particle spacings $dp = 0.02m$ (top) and $dp = 0.01m$ (bottom).

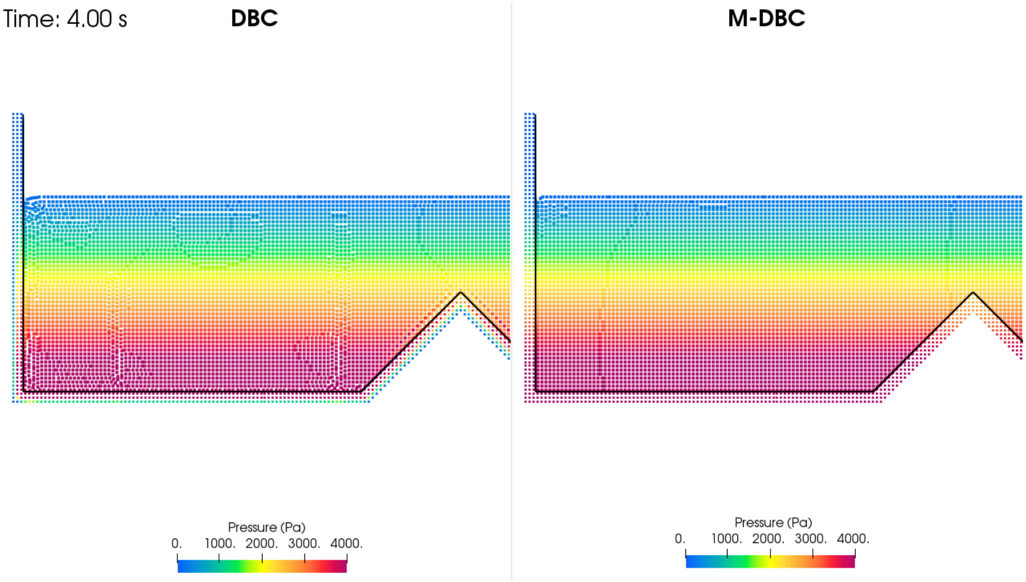


Figure 4.4: Pressure field in still water tank comparing DBC (left) and MDBC (right) for particle spacing $dp = 0.01m$ zooming in on the corner of the tank to better see the pressure in the boundary.

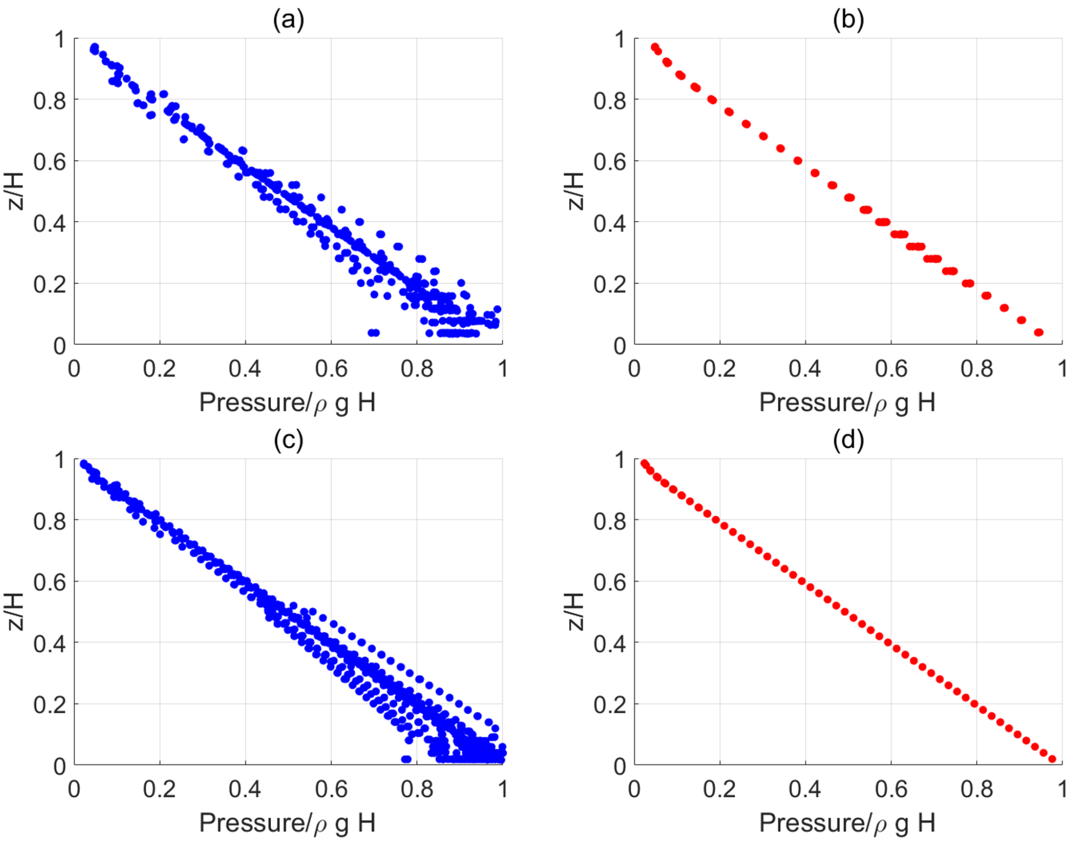


Figure 4.5: Plots of normalised pressure versus normalised depth in still water tank comparing DBC (left) and MDBC (right) for two particle spacings $dp = 0.02m$ (top) and $dp = 0.01m$ (bottom).

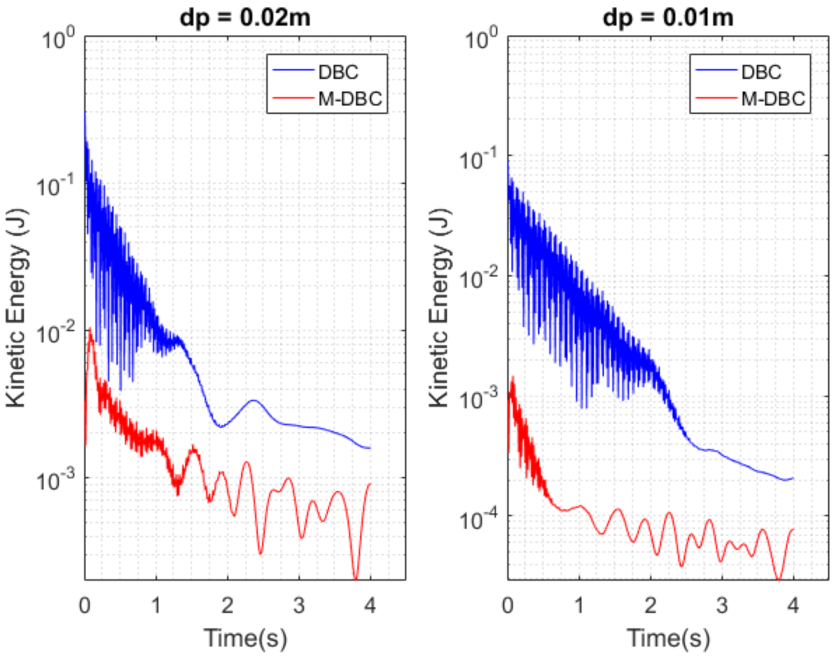


Figure 4.6: Plots of kinetic energy of fluid particles in still water tank versus time for two particle spacings $dp = 0.02m$ (left) and $dp = 0.01m$ (right). The energy of the fluid with DBC boundaries is shown using the blue line and with MDBC boundaries using the red.

Table 4.1: Table of numerical parameter for still water simulations

dp (m)	h/dp	α (Artificial)	δ (Molteni)	Shifting	CFL
0.04	2	0.01	0.1	OFF	0.2
0.02	2	0.01	0.1	OFF	0.2
0.01	2	0.01	0.1	OFF	0.2

(top) and $dp = 0.01$m (bottom), showing the fluid and boundary particles coloured by pressure with DBC on the left and MDBC on the right. Both methods show generally good results for the pressure field in the fluid, but closer inspection shows that the cases using MDBC have much smoother pressure field in the boundary for both the flat surfaces and the corners. Figure 4.4 provides a close-up into the corners of the case with $dp = 0.01$m, where the improvement is more noticeable. This shows that mirroring technique helps to better represent the boundary conditions for support of fluid particles.

Figure 4.5 shows the pressure of each fluid particle normalised using the hydrostatic pressure (ρgH) versus the height of the particle normalised by the initial water height (H) at the final time instant (4s). The left hand side shows the result using DBC and the right using MDBC, the top row has particle spacing $dp = 0.02m$ and the bottom $dp = 0.01m$. In both resolutions the DBC result shows noise in the pressure field near the bottom of the tank, this noise is minimised in the MDBC results. Therefore MDBC shows better hydrostatic pressure behaviour than DBC.

As well as the hydrostatic pressure behaviour, the kinetic energy of the fluid particles, which should be zero, can also be investigated. Figure 4.6 shows a time series of the total kinetic energy of all the fluid particles (note that a log scale is used). The left hand plot shows the time series with a particle spacing of $dp = 0.02m$ and the right $dp = 0.01m$ with the DBC result shown in blue and the MDBC result in red . As the kinetic energy of the particle with the MDBC boundary is smaller in both resolutions, it is clear that when using MDBC the movement of the particles is much lower than when using the original DBC.

4.1.2.2 Sloshing Tank

The second test case is the SPHERIC Benchmark Test Case #10, consisting of a sloshing tank of 0.9×0.508m with an initial water level H=0.093m as shown in Figure 4.7. This is a challenging example with a moving boundary and free surface. The numerical pressures are obtained using DBC and M-DBC and compared with the experimental values detected at Sensor 1.

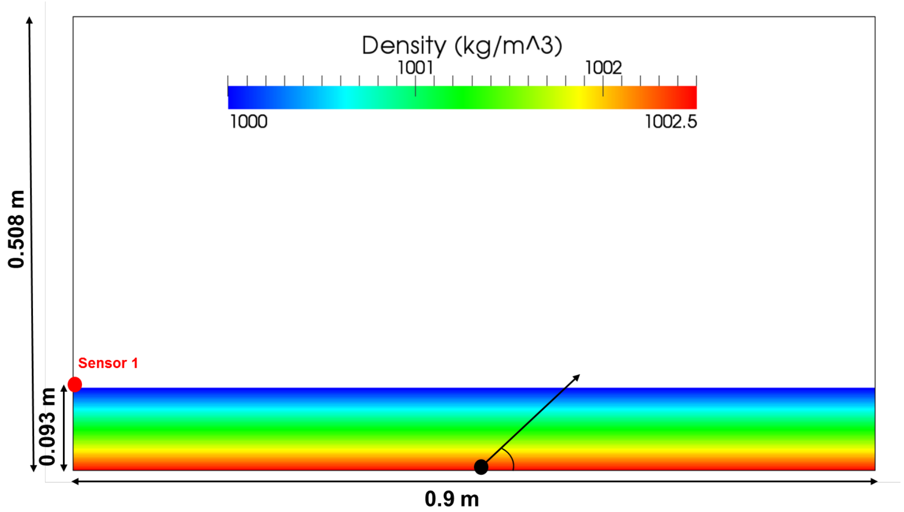


Figure 4.7: Set up of sloshing tank test case according to SPHERIC benchmark test case #10 including the location of sensor 1 and the initial water depth.

Two simulations with different resolutions are simulated and the numerical parameters used are displayed in Table 4.2. For these simulations the MDBC method used is again extrapolating the density only. Shifting is not used for this highly transient flow case.

Table 4.2: Table of numerical parameter for sloshing tank simulations

dp (m)	h/dp	α (Artificial)	δ (Molteni)	Shifting	CFL
0.004	2	0.01	0.1	OFF	0.2
0.002	2	0.01	0.1	OFF	0.2

The instant t=2.47s of the simulation is shown in Figure 4.8, this is the time when the fluid first impacts the left wall and Sensor 1. The left column shows the result with DBC and the right MDBC, with the top row showing the coarser resolution and the bottom row the finer. The colour of the particles corresponds to their pressure values. Figure 4.9 shows more in more detail the pressure field of the particles with DBC (left) and MDBC (right) with $dp = 0.002$m. Two improvements can be easily seen using MDBC: i) values of pressure of the boundary particles in the walls using MDBC are smoother than the ones shown in DBC and ii) the gap between the fluid and the boundary is much smaller, so that the spurious gap is avoided with MDBC. This is a very important result since the computation of pressure values at Sensor 1 with DBC provides representative results only if the numerical pressure gauge is moved to take into account the size of the gap between the fluid and the boundary. The work of Domínguez et al. (2015) estimated the size of the gap as the value of the smoothing length h.

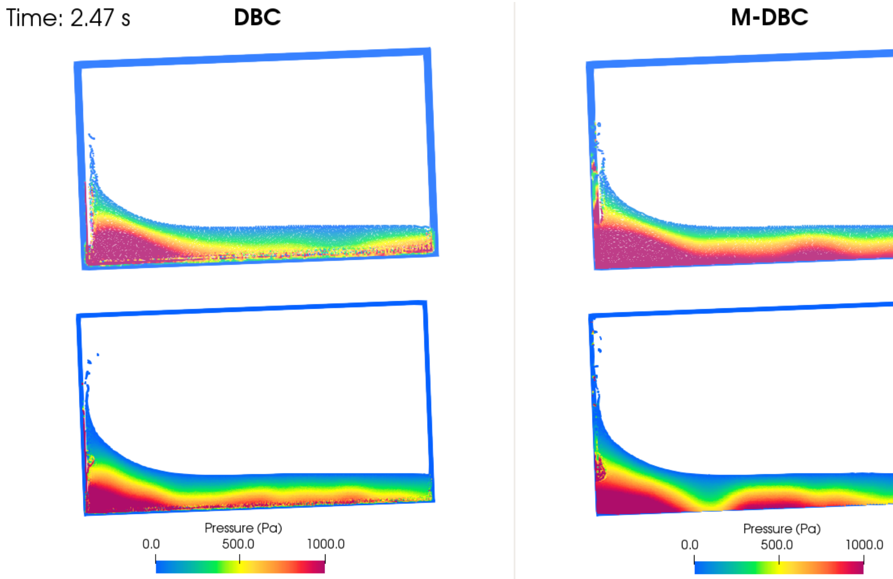


Figure 4.8: Time instant of sloshing tank at 2.47s showing the first wave impacting the wall with sensor 1 with particles coloured by pressure. Comparing DBC (left) and MDBC (right) for two particle spacings $dp = 0.004m$ (top) and $dp = 0.002m$ (bottom).

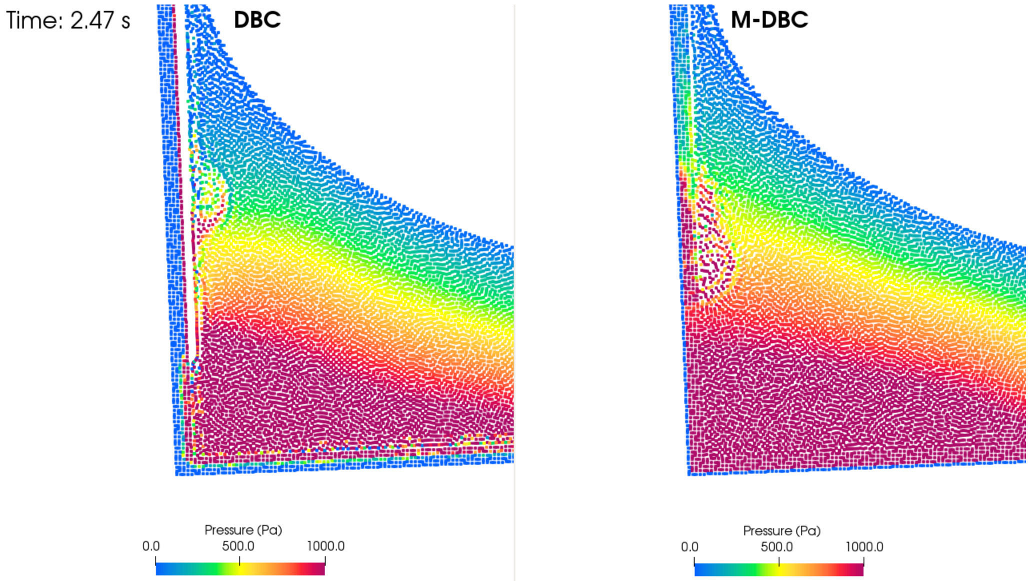


Figure 4.9: Time instant of sloshing tank at 2.47s showing the first wave impacting the wall with sensor 1 with particles coloured by pressure, close up on the side wall being impacted. Comparing DBC (left) and MDBC (right) for particle spacing $dp = 0.002m$.

The experimental pressure (black line) is compared with numerical results in Figure 4.10. The SPH results are from the simulations using the finer resolution. In the first row the blue line shows the SPH result using DBC with the pressure probe on the boundary surface. The results are invalid since the large gap keeps fluid away from the wall. The second row shows the numerical pressure (green line) using DBC with pressure gauge moved $+h$ into the fluid, so the results are much better and the agreement with experimental data. Using MDBC reduces the size of the gap, allowing pressure to be computed at the exact location of the experimental gauge. With MDBC a very good result is obtained, shown by the red line in the bottom panel with a good agreement with the experimental data.

Computational Cost The run times for the sloshing tank test case on a GPU for both DBC and MDBC with particle spacing dp=0.002m are shown in Table 4.3, along with a ratio of the run times showing the increased computational cost of MDBC on selection of hardware.

Table 4.3: Table of run times of sloshing tank with particle spacing dp=0.002m using different hardware.

Hardware	DBC(s)	MDBC(s)	MDBC/DBC
CPU i7-6700K	7350	8191	1.11
Tesla K40	811	967	1.19
Ge Force RTX 2080 Ti	625	681	1.09
Tesla V100	376	430	1.14

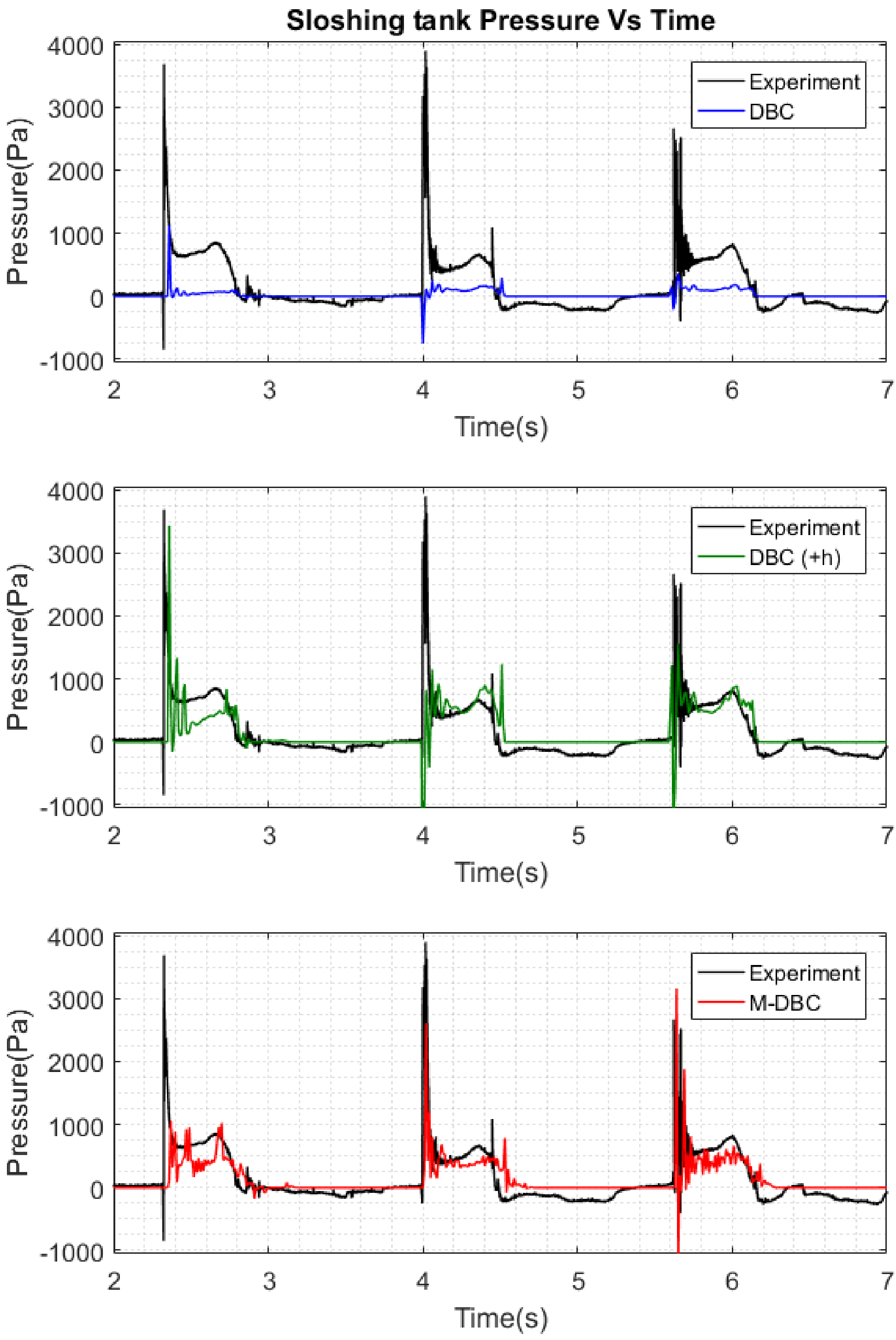


Figure 4.10: Plots of SPH pressure at sensor 1 versus experimental data (black line). Top plot (blue) shows pressure using DBC with pressure probe located on boundary surface. Middle plot (green) shows pressure with DBC with pressure probe placed one smoothing length distance into the fluid. Bottom (red) shows pressure with MDBC with pressure probe located on the boundary surface.

4.1.2.3 Couette and Poiseuille Flow

So far the test cases investigated have used the density extrapolation only. To test the improved no-slip condition flows that are more sensitive to boundary velocity need to be investigated. To this end Couette and Poiseuille flows will be simulated. These are $2D$ channel flows with analytical solutions for the velocity at different times.

The channel used for the Couette flow is $1m$ in width with the bottom boundary at $z = 0m$ and the top boundary at $z = 1m$. The flow is set up without gravity and the bottom boundary is fixed, the flow is driven by the top boundary moving with velocity $1ms^{-1}$. The numerical parameters of the simulations are displayed in Table 4.4. Periodic boundary conditions are used at the ends of the channel. Shifting is included for these test cases to avoid particle clumping and instability leading to simulation failure, as highlighted by Basa et al. (2008).

Table 4.4: Table of numerical parameters for Couette flow simulations

dp (m)	h/dp	ν (Laminar)	δ (Molteni)	Shifting	CFL
0.05	1.25	0.1	0.1	ON	0.1
0.04	1.75	0.1	0.1	ON	0.1
0.02	2.25	0.1	0.1	ON	0.1
0.01	2.75	0.1	0.1	ON	0.1
0.005	3.25	0.1	0.1	ON	0.1

The analytical solution for this Couette flow is given by

$$u(z) = U_{max} z = z. \tag{4.16}$$

The Reynolds number of the this flow is $Re = 10$. The fluid starts initially at rest and the boundary starts at its maximum velocity, the total simulation time is 5 seconds in which time the fluid accelerates to steady state. It is worth therefore looking at the comparison of the SPH result to the time dependent analytical solution given by

$$u(z,t) = z + 2\sum_{n=1}^{\infty} \frac{(-1)^n}{n\pi} e^{-\lambda_n^2 \nu t} \sin(\lambda_n z), \tag{4.17}$$

where $\lambda_n = n\pi$ for $n = 1, 2, ...$ given by Emin Erdoğan (2002).

Figure 4.11 shows the comparison of SPH to analytical solution for the flow velocity. The DBC result is shown in the left column and the MDBC result on the right both

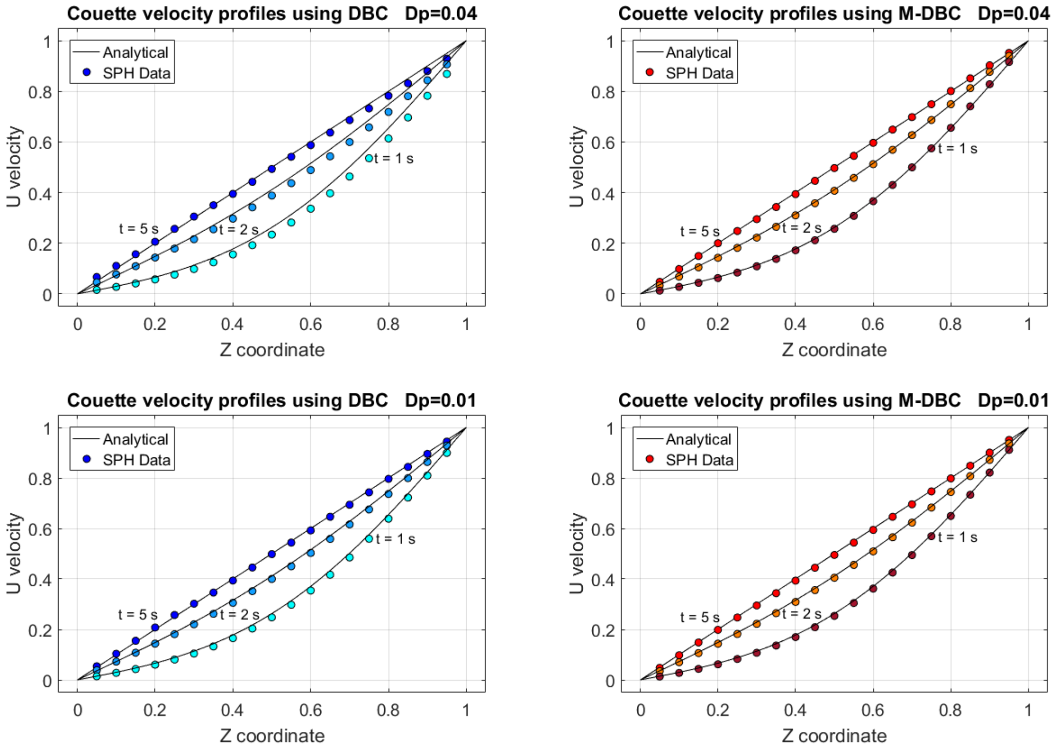


Figure 4.11: Plots of SPH data versus analytical solutions for time dependent Couette flow for two particle spacing $dp = 0.04m$ (top) and $dp = 0.01m$ (bottom) and boundary methods DBC (left) and MDBC (right), all h/dp ratios are 1.25.

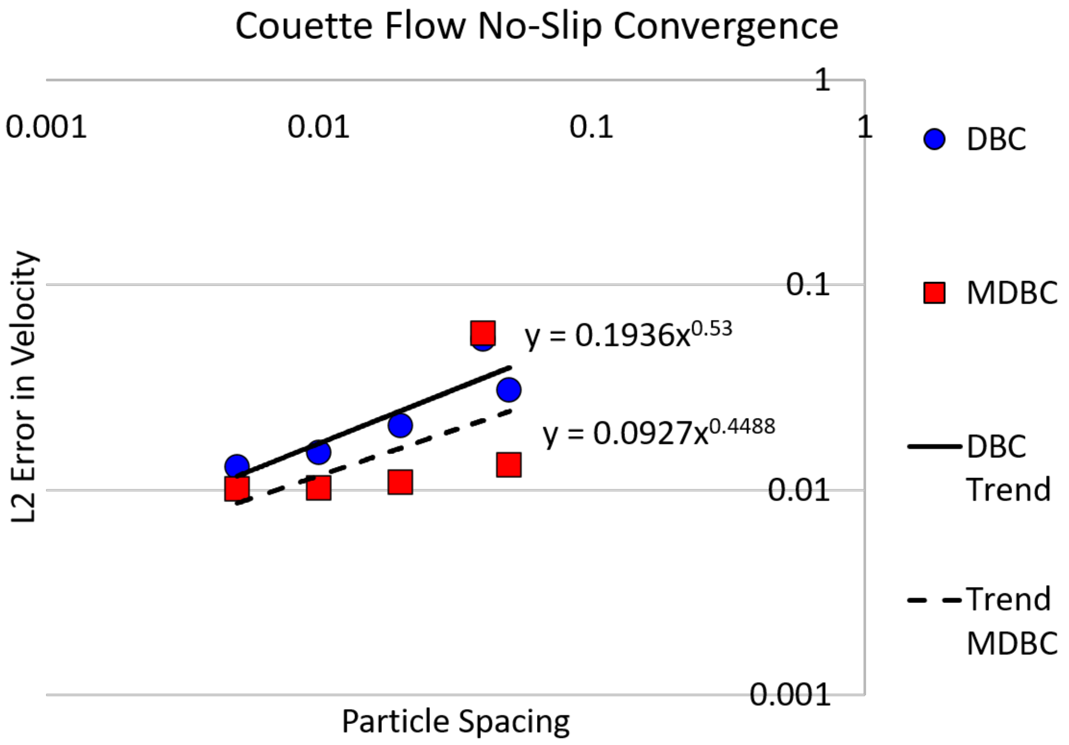


Figure 4.12: Convergence study of Couette flow.

for two particle spacings, $dp = 0.04m$ on the top row and $dp = 0.01m$ on the bottom row all with a h/dp ratio of 1.25. Comparing the DBC and MDBC results, MDBC shows a small improvement in agreement to the analytical solution for early times, at 1 second the $dp = 0.04m$ MDBC velocity is closer to the analytical result than DBC. Figure 4.12 shows a convergence study for the Couette flow plotting L2 error in velocity versus particle spacing. It can be seen that both the DBC (blue) and MDBC (red) cases converge with rate of convergence of 0.53 and 0.45 respectively. The error in the MDBC velocity is overall lower for each of the particle spacings chosen, which is to be expected.

The channel used for the Poiseuille flow has walls placed at $z = \pm 0.5m$ giving a total channel width of $1m$. There is no gravity and both walls are fixed, flow is driven by a body force with acceleration $F = 0.8ms^{-2}$. The numerical parameters of the simulations are displayed in Table 4.5. To allow fluid to enter on one side after leaving the other periodic boundary conditions are again used at the ends of the channel.

Table 4.5: Table of numerical parameters for Poiseuille flow simulations

dp (m)	h/dp	ν (Laminar)	δ (Molteni)	Shifting	CFL
0.05	1.25	0.1	0.1	ON	0.1
0.04	1.75	0.1	0.1	ON	0.1
0.02	2.25	0.1	0.1	ON	0.1
0.01	2.75	0.1	0.1	ON	0.1
0.005	3.25	0.1	0.1	ON	0.1

The analytical velocity for the Poiseuille flow is given by

$$u(z) = \frac{F}{2\nu}\left(0.5^2 - z^2\right). \tag{4.18}$$

With the values for kinematic viscosity and the body force, the maximum velocity of the flow is $U_{max} = 1ms^{-1}$. The Reynolds number for this flow is therefore $Re = 10$. The fluid initially starts at rest and accelerates up to steady state in 5 seconds of physical time, so it is interesting to look at how the SPH result matches to the time dependent Poiseuille velocity given by

$$u(z,t) = \frac{F}{2\nu}\left(0.5^2 - z^2\right) + \sum_{n=0}^{\infty} F\left(\frac{16(-1)^n 0.5^2}{(2n+1)^3\pi^3\nu}\right) e^{-\lambda_n^2 \nu t}\cos(\lambda_n z), \tag{4.19}$$

where $\lambda_n = (2n+1)\pi$ for $n = 0, 1, 2, ...$ given by Emin Erdoğan (2002).

A comparison of the SPH particle velocities and the analytical solutions are shown

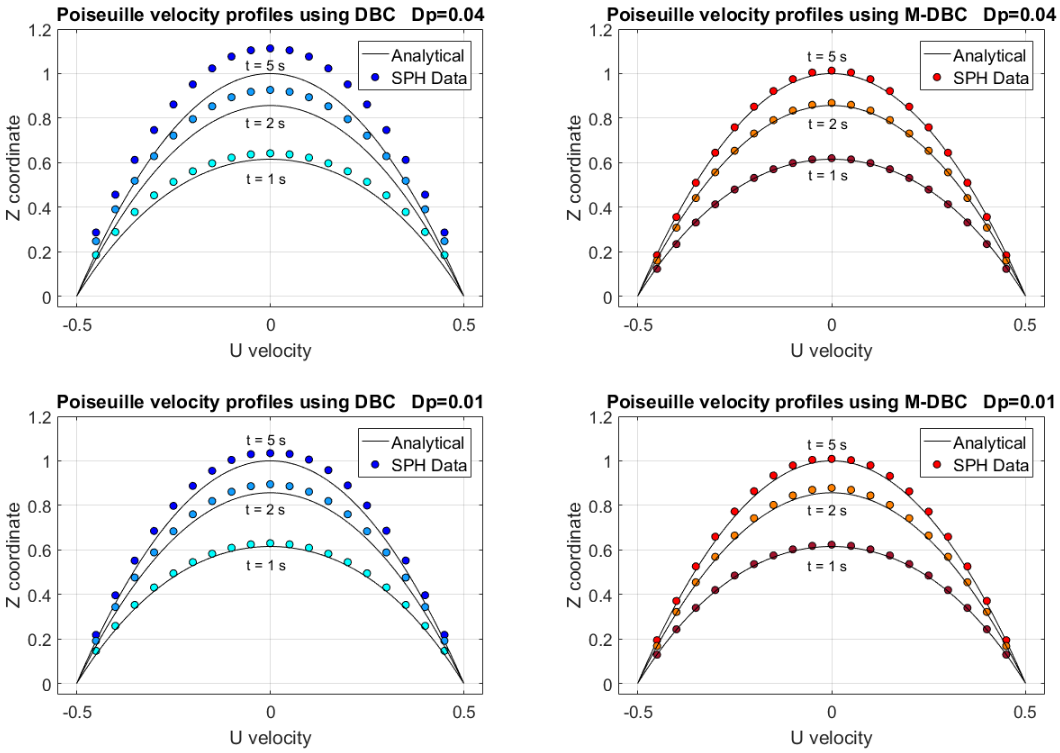


Figure 4.13: Plots of SPH data versus analytical solutions for time dependent Poiseuille flow for two particle spacing $dp = 0.04m$ (top) and $dp = 0.01m$ (bottom) and boundary methods DBC (left) and MDBC (right),all with h/dp ratio of 1.25.

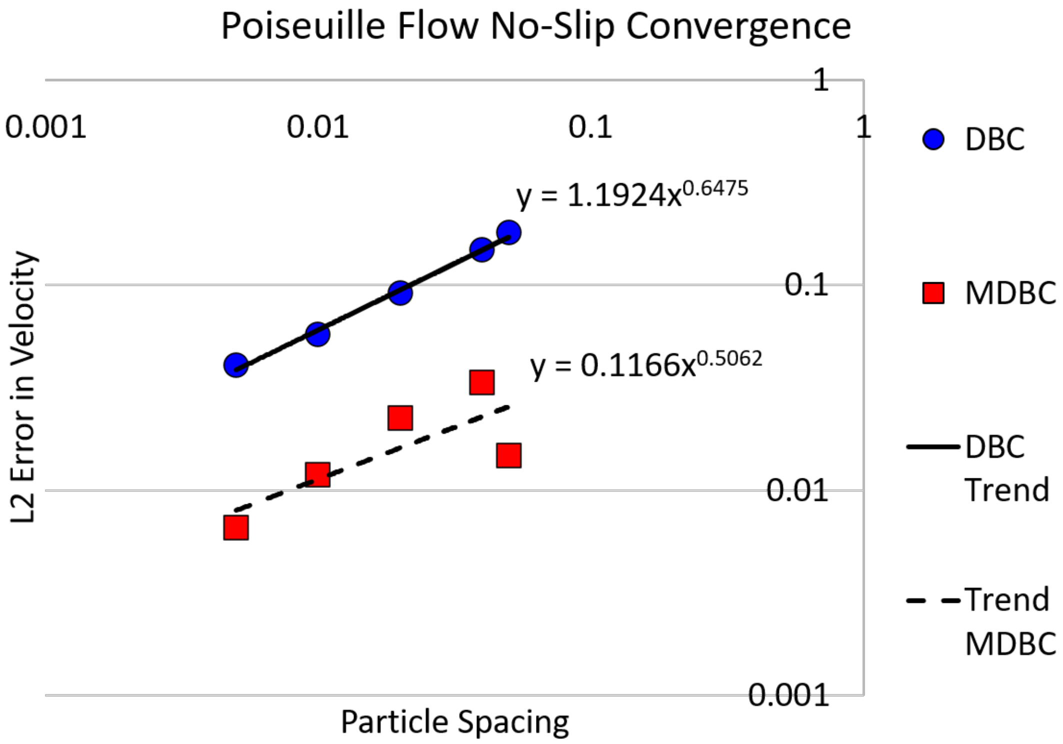


Figure 4.14: Convergence study of Poiseuille flow.

in Figure 4.13 for two particle spacings, $dp = 0.04m$ on the top row and $dp = 0.01m$ on the bottom row, and the two boundary conditions, DBC on the left and MDBC on the right all with a h/dp ratio of 1.25. Comparing the DBC and MDBC results it is clear to see that the MDBC result agrees much closer to the analytical solution for all times. The DBC result actually moves away from the analytical solution as the time increases due to its less accurate no-slip condition. The DBC result does get better as the resolution is increased but it should be noted that the coarse resolution MDBC result gives closer agreement than the finer resolution DBC result. A convergence study is shown in Figure 4.14 plotting the L2 error in velocity against particle spacing. As can be seen from the plot both the DBC and MDBC cases converge with convergence rates of 0.64 and 0.51 respectively. The error in the MDBC is much lower than the error in DBC, which is to be expected, and close to an order of magnitude lower for the coarser particle spacings.

4.1.3 Summary

The new boundary condition, MDBC, presented shows a big improvement over the original DBC method. The still water test case shows that the new approach of density extrapolation gives a much smoother density and pressure fields in the boundary, even for more complex shaped geometries such as the wedge at the bottom of the tank. This smoother boundary pressure leads to a reduced amount of kinetic energy in the fluid particles and a closer approximation to the hydrostatic pressure in the fluid. The sloshing tank case shows that the method works well with moving boundaries and free-surfaces. The reduction in the gap between boundary and fluid particles allows for more realistic pressure result on solid structures with out needing to place pressure probes a distance away from the boundary. The velocity extrapolation shows a big improvement in accuracy in creating no-slip boundary conditions. The agreement with analytical solutions for Poiseuille and Couette flow is improved and the error in velocity is much reduced.

4.2 Partial Slip Boundary Conditions

As was shown in the review of the literature, partial slip boundary conditions have not yet been used in SPH. In this section the formulation and validation of a SPH partial slip boundary condition are presented.

4.2.1 Method and implementation

In their paper on boundary conditions, Macia et al. (2011) studied the theory of SPH boundary treatments including antisymmetric extensions or mirroring method of which MDBC is an example. The MDBC interpolated boundary velocity is a mirror image of the velocity at the respective ghost nodes located in the fluid. They suggest that the velocity in the fluid close to a boundary located at $z = 0$ with fluid above can be described by the equation

$$u(z) = U_B + a_1 z + a_2 z^2 + \cdots \tag{4.20}$$

for some a_i and boundary velocity U_B. The velocity in the boundary would then be given by

$$u(z) = 2U_B - u(z^+) \tag{4.21}$$

where $u(z^+)$ is the velocity at the positive version of the z coordinate. This results in a net velocity at the boundary surface of U_B as

$$u(0) = 2U_B - (U_B + 0) = U_B$$

. A graphical representation of this can be seen in Figure 4.15 (c) for a moving boundary with velocity U_B and (a) for a fixed boundary. For a partial slip flow, while the boundary is fixed, the overall effect is for the fluid to move over the boundary at a small velocity u_{slip}. Following the ideas of Macia et al. (2011), the velocity in the fluid close to a partial slipping boundary would then be given by

$$u(z) = u_{slip} + a_1 z + a_2 z^2 + \cdots, \tag{4.22}$$

and the velocity in the boundary would then be given by

$$u(z) = 2u_{slip} - u(z^+). \tag{4.23}$$

Figure 4.15 (b) shows a graphical representation of this, with an overall velocity of u_{slip} at the boundary surface.

More generally, if a partial slip boundary is moving with velocity U_B then the overall

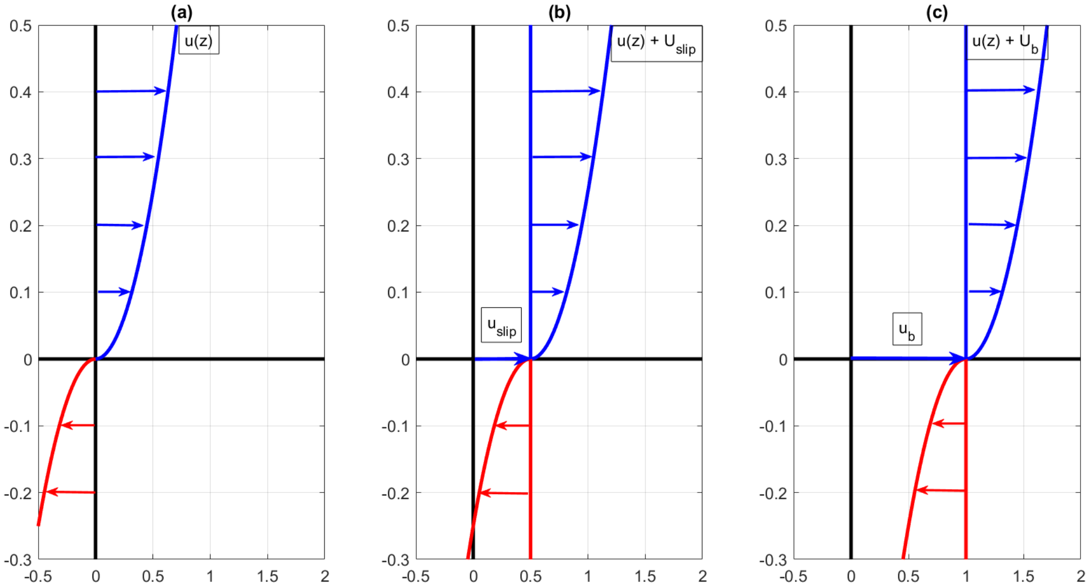


Figure 4.15: Flow profile close to a boundary (blue), and the extrapolated flow profile from a mirroring type boundary treatment (red) for: (a) no-slip flow with fixed boundary; (b) partial slip flow with partial slip velocity U_{slip}; (c) no-slip flow with moving boundary with velocity U_B

effect can be combined into one velocity $U_{B+} = U_B + u_{slip}$ which can then be used as in Equations (4.20) & (4.21). Writing this out in full vector notation, the velocity of a particle b moving partial slip boundary is given by

$$\mathbf{u}_b = 2\mathbf{U}_B + 2\mathbf{u}_{slip} - \mathbf{U}_{BC}, \tag{4.24}$$

where $\mathbf{u}_{BC}$ is the no-slip velocity from the boundary condition used.

From the theory, the partial slip velocity is given by

$$\mathbf{u}_{slip} = \lambda_s \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right] \cdot \mathbf{n}, \tag{4.25}$$

where λ_s is the slip length, $\mathbf{u} = [u, v, w]^T$ is the velocity of the fluid and $\mathbf{n}$ is the unit normal to the boundary pointing into the fluid. Expanding out the gradient operators, the partial slip velocity can be seen to be given by

$$\begin{aligned}\mathbf{u}_{slip} &= \lambda_s \begin{bmatrix} 2\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & 2\frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} & 2\frac{\partial w}{\partial z} \end{bmatrix} \cdot \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}, \\ &= \lambda_s \begin{bmatrix} 2n_x\frac{\partial u}{\partial x} + n_y\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + n_z\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\ n_x\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + 2n_y\frac{\partial v}{\partial y} + n_z\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \\ n_x\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) + n_y\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) + 2n_z\frac{\partial w}{\partial z} \end{bmatrix}. \end{aligned} \tag{4.26}$$

It is clear from the entries in the vector of (4.26) that the velocity gradient needs to be found at the boundary surface. Defining the boundary surface to be located half a particle spacing away from the first layer of boundary particles, as shown in Figure 4.16 the velocity gradients are evaluated at a ghost node X on the boundary surface. The location of the ghost node is found using the boundary normal and the distance to the boundary surface. The velocity gradient $\partial u/\partial z$ for example is found through the equation

$$\left(\frac{\partial u}{\partial z}\right)_X = -\sum_j \frac{m_j}{\rho_j}\frac{(U_B - u_j)(z_X - z_j)}{r_{Xj}}\frac{dW_{Xj}}{dr_{Xj}}, \tag{4.27}$$

where U_B is the velocity of the solid boundary and the j particles included in the summation include both the surrounding fluid and boundary particles. Since both the surrounding fluid and boundary particles are included in the summation the kernel is complete and no kernel corrections are used. If the boundary is not moving then $U_B = 0$. Once the partial slip velocity has been evaluated it can be used in Equation (4.24) for the boundary particle velocity.

If the boundary particles are arranged in a Cartesian grid as shown in Figure 4.16, it is possible for more than one boundary particle to share a ghost node X. For example, the three boundary particles in the dashed box behind the ghost node on the boundary surface would be updated with the same partial slip velocity as they each would all share the same ghost node.

For the work that follows the base no-slip boundary condition is the MDBC method described above. During each time step, the partial slip velocity should be calculated and the boundary particles updated after the no-slip boundary condition has been calculated but before any fluid particle interactions.

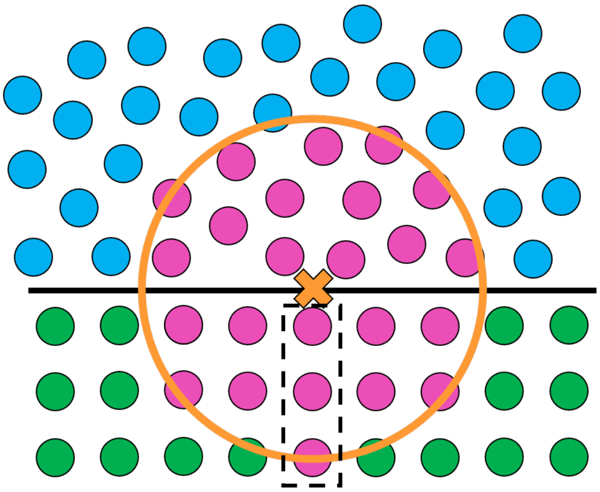


Figure 4.16: Particles included in partial slip velocity gradient summation around the point X. Each of the boundary particles in the dashed box will mirror to this point and receive the same partial slip velocity.

4.2.2 Results

To test the partial slip condition a number of test cases will be simulated.

4.2.2.1 Poiseuille Flow

Newtonian Poiseuille Flow First a Poiseuille flow with a Newtonian fluid will be used to test a number of slip lengths. The Poiseuille flow is an ideal test case to test partial slip as for even a small slip length the overall velocity of the flow is increased and known. The analytical solution for a partial slip Poiseuille flow driven by a body force with acceleration F with boundaries located at $z = \pm l$ is given by

$$u(z) = \frac{F}{2\nu}\left(l^2 + 2\lambda_s l - z^2\right), \tag{4.28}$$

where ν is the kinematic viscosity and λ_s is the slip length . For the cases considered here $l = 0.5m$. The flows are started from rest and the fluid accelerated using the body force over 5 seconds of physical time. The time dependent solution can then also be looked at, this is given by

$$\begin{aligned} u(z,t) = &\frac{F}{2\nu}\left(l^2 + 2\lambda_s l - z^2\right) \\ &+ \sum_{n=0}^{\infty}\left(\frac{16(-1)^n F l^2}{(2n+1)^3\pi^3\nu}\right) e^{-\lambda_n^2 \nu t}\cos(\lambda_n y) \\ &+ \sum_{n=0}^{\infty}\left(\frac{8\lambda_s F l}{(2n+1)^2\pi^2\nu}\right) e^{-\lambda_n^2 \nu t}, \end{aligned} \tag{4.29}$$

where

$$\lambda_n = \frac{(2n+1)\pi}{2l} \quad n = 0, 1, 2, ...\infty. \tag{4.30}$$

A derivation of this equation is included in the appendices. Setting the slip length in these equations to zero recovers the no-slip versions

Four slip lengths will be tested: $\lambda_s = 0$ (no-slip); $\lambda_s = 0.005m$; $\lambda_s = 0.01m$, and $\lambda_s = 0.02m$. Five particle spacings will be used and the numerical parameters used for each particle spacing are displayed in Table 4.6. The maximum velocity of the no-slip flow is $U_{max} = 1ms^{-1}$ resulting in a Reynolds number of $Re = 10$. The three non zero partial slip lengths result in maximum velocities of $U_{max} = 1.02ms^{-1}$, $U_{max} = 1.04ms^{-1}$ and $U_{max} = 1.08ms^{-1}$ with corresponding increases in Reynolds number. Shifting is also included for these test cases to avoid particle clumping and instability leading to simulation failure, as highlighted by Basa et al. (2008).

Table 4.6: Table of numerical parameter for Newtonian Poiseuille flow simulations with no-slip and partial slip boundary conditions

dp (m)	h/dp	ν (Laminar)	δ (Molteni)	Shifting	CFL
0.05	1.25	0.1	0.1	ON	0.1
0.04	1.75	0.1	0.1	ON	0.1
0.02	2.25	0.1	0.1	ON	0.1
0.01	2.75	0.1	0.1	ON	0.1
0.005	3.25	0.1	0.1	ON	0.1

The first slip length used is $\lambda_s = 0$, which corresponds to no-slip. Plots of the SPH velocity profiles compared to the analytical solution for different times are shown for two particle spacings in Figure 4.17. The plot shows that there is good agreement between the SPH data and the analytical result for both particle spacings at the three times shown, these times are 1 and 2 seconds from the start and steady state at 5 seconds. The finer resolution result shows a closer agreement as it does not over predict the steady state velocity as much as the coarser resolution. This is to be expected as this is the same result as seen in the previous section for no-slip MDBC.

Figures 4.18, 4.19 and 4.20 shows the SPH velocity profiles compared to the analytical solution for different times for slip lengths $\lambda_s = 0.005$, $\lambda_s = 0.01$ and $\lambda_s = 0.02$ respectively. Each plot shows two particle spacings of $dp = 0.04$ with a h/dp ratio of 1.25 and $dp = 0.01$ with a h/dp ratio of 2.25. Overall each of the results shows good agreement with the analytical result. Looking at the two smaller slip lengths, the coarser resolution tends to over-predict the velocity while the finer resolution gives closer agreement. As the slip length increases the velocity the agreement between

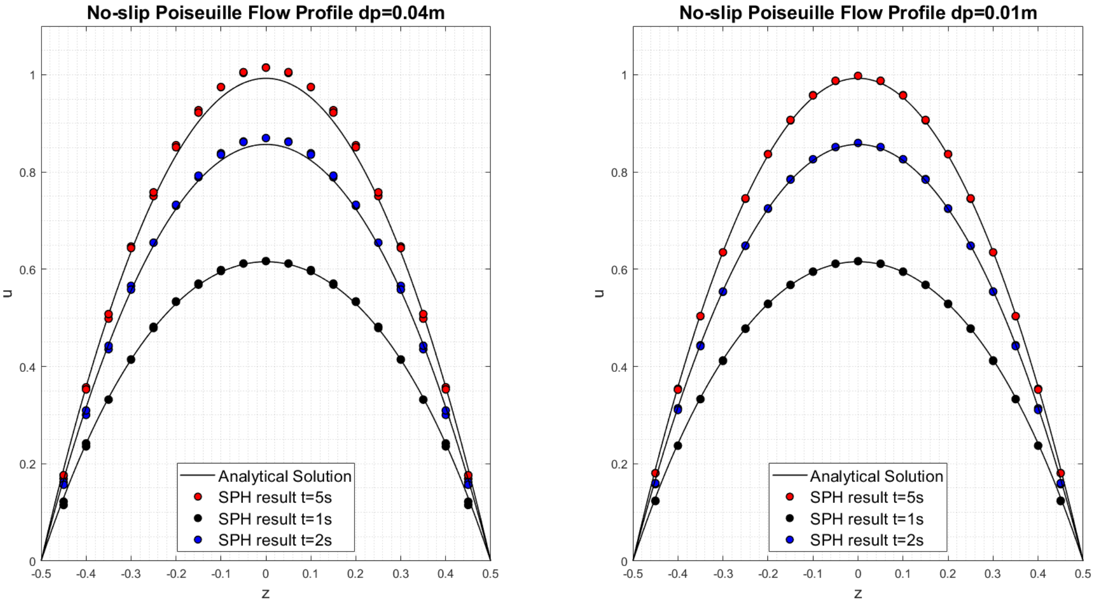


Figure 4.17: Poiseuille flow profile with no-slip boundary conditions compared to analytical solution for particle spacings $dp = 0.04$ (left) and $dp = 0.01$ (right).

the SPH result and the analytical solution gets worse. This can be seen clearly in the plots for $\lambda_s = 0.02$ where the coarse resolution under-predicts the velocity for the smaller times and the finer resolution under-predicts the velocity for all the times shown. A likely cause is that the partial slip velocity is being under predicted at the boundary, leading to the overall velocity of the flow also being under predicted. The affects the finer resolution more since the no-slip result for this particle spacing was much closer to the analytical result. The only difference between the no-slip flow and a partial slip flow is the increased velocity at the boundary. Adding an under predicted partial slip velocity to a good no-slip condition results in an under prediction of the flow velocity. The coarser resolutions are less affected as they already over predict the no-slip velocity, and so this combined with an under predicted partial slip velocity by luck results in a closer result. This under prediction is more obvious for larger slip lengths as the velocity increase, controlled by the slip length, is larger. To get a closer approximation of the partial slip velocity a more accurate approximation of the velocity gradients will be needed.

This can be further seen when looking at convergence plots for the flows. Figure 4.21 shows a convergence study for the partial slip Poiseuille flows plotting L2 error in velocity against particle spacing. For this convergence study the ratio of smoothing length to particle spacing, h/dp, was increased as the particle spacing was decreased according to Quinlan et al. (2006). In Figure 4.21 the red squares and line correspond to the no-slip Poiseuille flow from the previous section, and

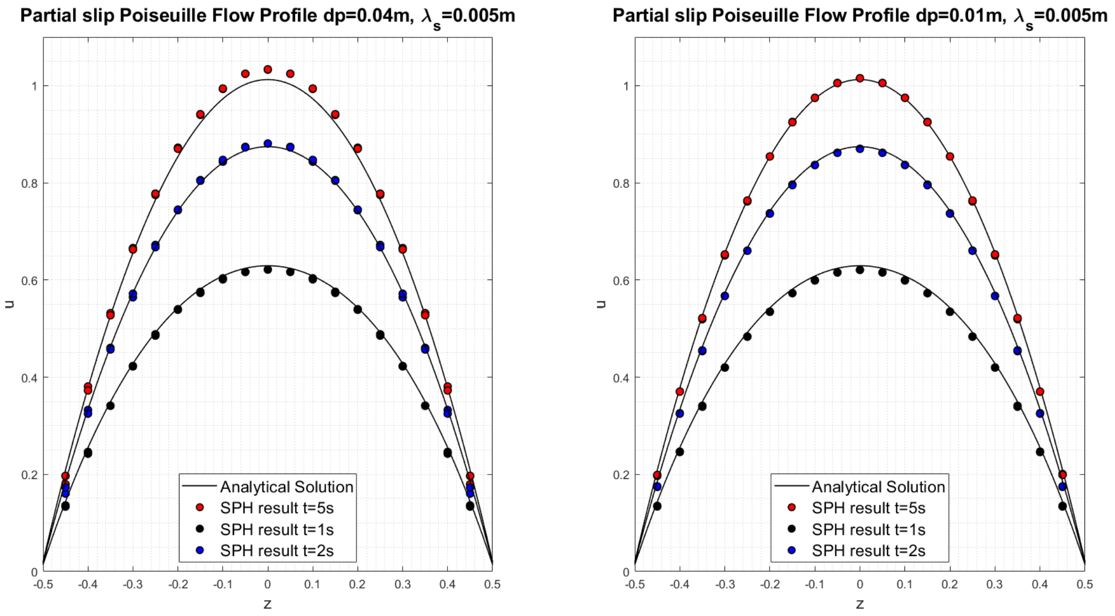


Figure 4.18: Poiseuille flow profile with partial slip boundary conditions $\lambda_s = 0.005$ compared to analytical solution for particle spacings $dp = 0.04$ (left) and $dp = 0.01$ (right).

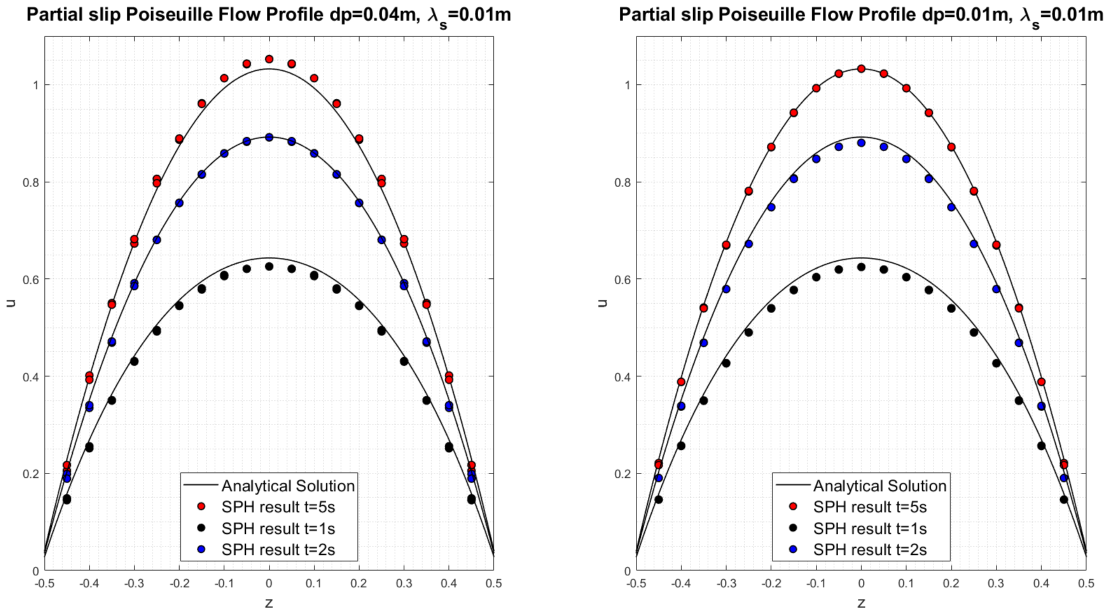
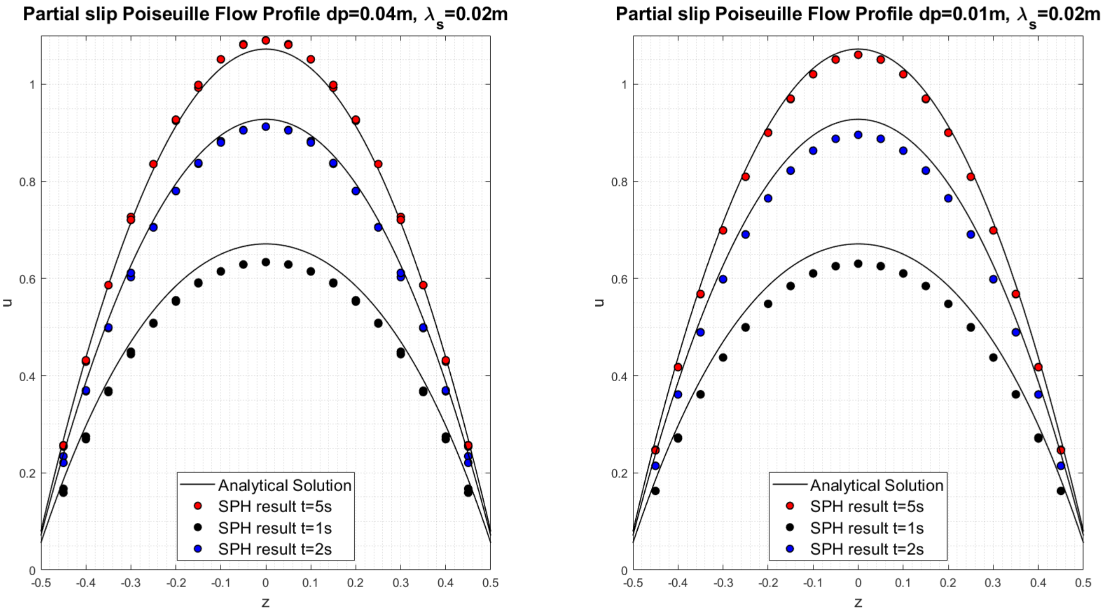


Figure 4.19: Poiseuille flow profile with partial slip boundary conditions $\lambda_s = 0.01$ compared to analytical solution for particle spacings $dp = 0.04$ (left) and $dp = 0.01$ (right).


Partial slip Poiseuille Flow Profile dp=0.04m, λ_s=0.02m

Partial slip Poiseuille Flow Profile dp=0.01m, λ_s=0.02m

Figure 4.20: Poiseuille flow profile with partial slip boundary conditions $\lambda_s = 0.02$ compared to analytical solution for particle spacings $dp = 0.04$ (left) and $dp = 0.01$ (right).

converges with rate 0.51. The blue triangles, green square and black X's then show the error for the partial slip flows with slip lengths of $\lambda_s = 0.005$, $\lambda_s = 0.01$, and $\lambda_s = 0.02$ with the matching coloured lines showing the trend of convergence. The $\lambda_s = 0.005$ and $\lambda_s = 0.01$ converge with rates 0.35 and 0.24 respectively. The $\lambda_s = 0.02$ case however does not converge, and the two finer resolutions failed with the h/dp ratios stated in Table 4.6, these were rerun with larger ratios finally running with ratios of 3.25 and 6.25 respectively.

To test if the error introduced is from the velocity gradient calculation method or an error introduced from the zeroth order MDBC velocity interpolation, the cases were rerun with the first order consistent Marrone et al. (2011) boundary treatment based on an MLS interpolation. The errors for the Marrone boundary runs are shown in 4.22 with the same colours and shapes used as for MDBC. As can be seen the same issue occurs with the first order consistent method, and the finer resolution with the larger slip length also fail unless the much larger h/dp ratios were used. It is clear therefore that a more accurate approach for the velocity gradient calculation needs to be used to ensure accurate results.

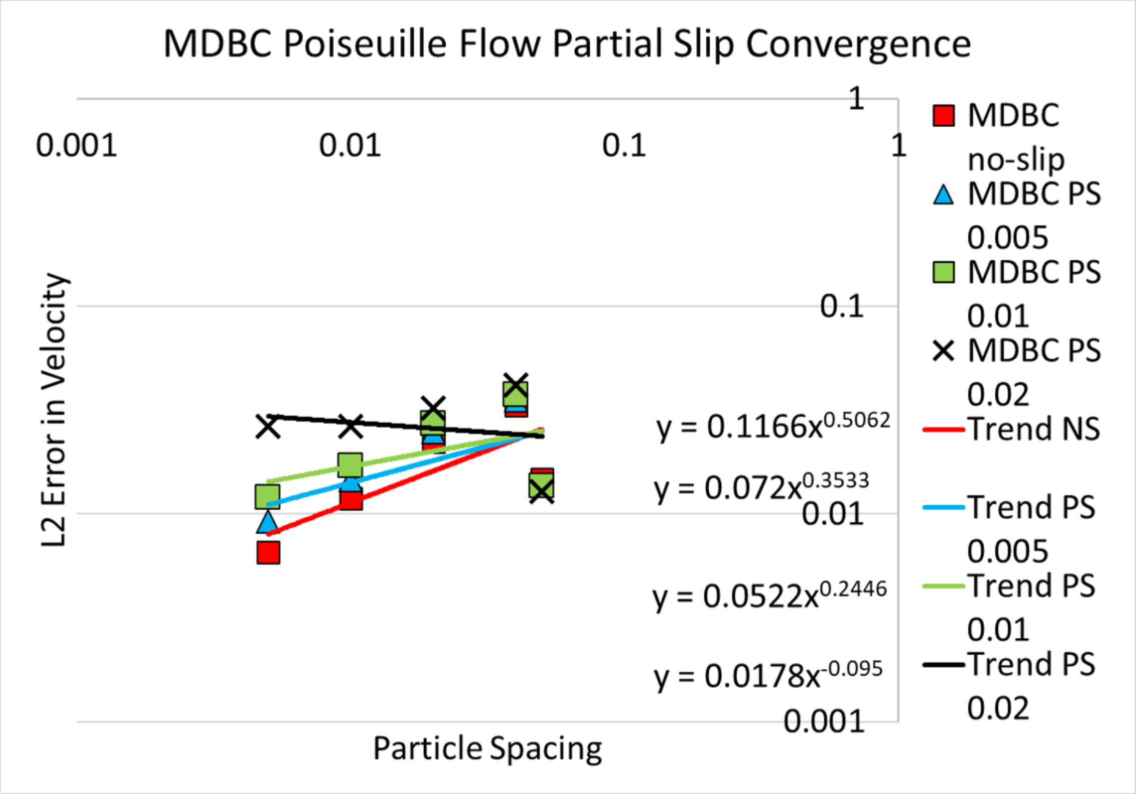


Figure 4.21: Convergence studies for partial slip Poiseuille flows

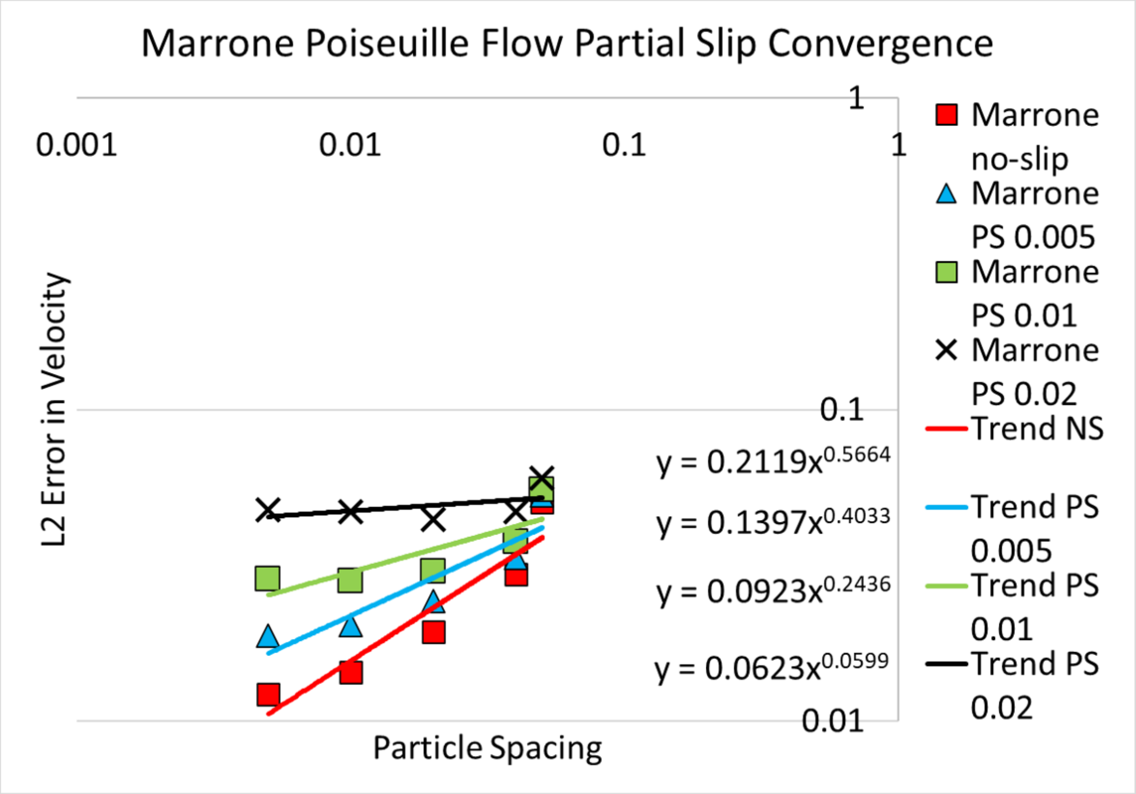


Figure 4.22: Convergence studies for partial slip Poiseuille flows

Non-Newtonian Poiseuille Flow The second test case will also be a Poiseuille flow but with a non-Newtonian fluid. Since many of Unilever type fluids show non-Newtonian behaviour it is important to see how both MDBC and the partial slip boundary conditions work with non-Newtonian flows. Here a shear thinning power law Poiseuille flow with walls at $z = \pm l (= 0.5m)$ is simulated with no-slip and partial slip boundaries. The fluid stars at rest and is accelerated by a body force with acceleration of $F = 0.1ms^{-2}$ in the x direction. Table 4.7 displays the numerical parameter used for this flow including the power law index n and the consistency index K.

Table 4.7: Table of numerical parameter for non-Newtonian Poiseuille flow simulations with no-slip and partial slip boundary conditions

dp (m)	h/dp	K	n	δ (Molteni)	Shifting	λ_s(m)	CFL
0.04	2.25	100	0.9	0.1	ON	NS 0	0.1
0.02	2.25	100	0.9	0.1	ON	NS 0	0.1
0.01	2.25	100	0.9	0.1	ON	NS 0	0.1
0.04	2.25	100	0.9	0.1	ON	PS 0.01	0.1
0.02	2.25	100	0.9	0.1	ON	PS 0.01	0.1
0.01	2.25	100	0.9	0.1	ON	PS 0.01	0.1

The analytical solution for a non-Newtonian Poiseuille flow with partial slip boundary conditions is given by

$$u(z) = \frac{n}{n+1}\left(\frac{\rho F}{K}\right)^{1/n}\left(l^{(n+1)/n} - \frac{2(n+1)\lambda_s l^{1/n}}{n} - z^{(n+1)/n}\right), \qquad (4.31)$$

where setting the slip length to zero recovers the no-slip solution, as found in Ferrás et al. (2012).

Figures 4.23 and 4.24 show the flow profiles of the SPH data compared to the analytical solutions for the no-slip and partial slip flows respectively. The plots show that the SPH data tend to under predict the analytical solution and the agreement between the getting closer as the partial spacing reduces. The under prediction of the flow velocity is much larger in the partial slip flow. Looking at the partial slip flow, focussing on the boundaries and the partial slip velocity, it can be seen that none of the resolutions accurately predict the partial slip velocity. The two larger particle spacings under predict and the finest one over predicts. This shows that to accurately calculate the partial slip velocity a more accurate velocity gradient calculation is needed, especially for non-Newtonian flows. However the current method still gives approximate agreement. Figure 4.25 shows a convergence study for each of the flows, plotting L2 error in velocity against particle spacing. The

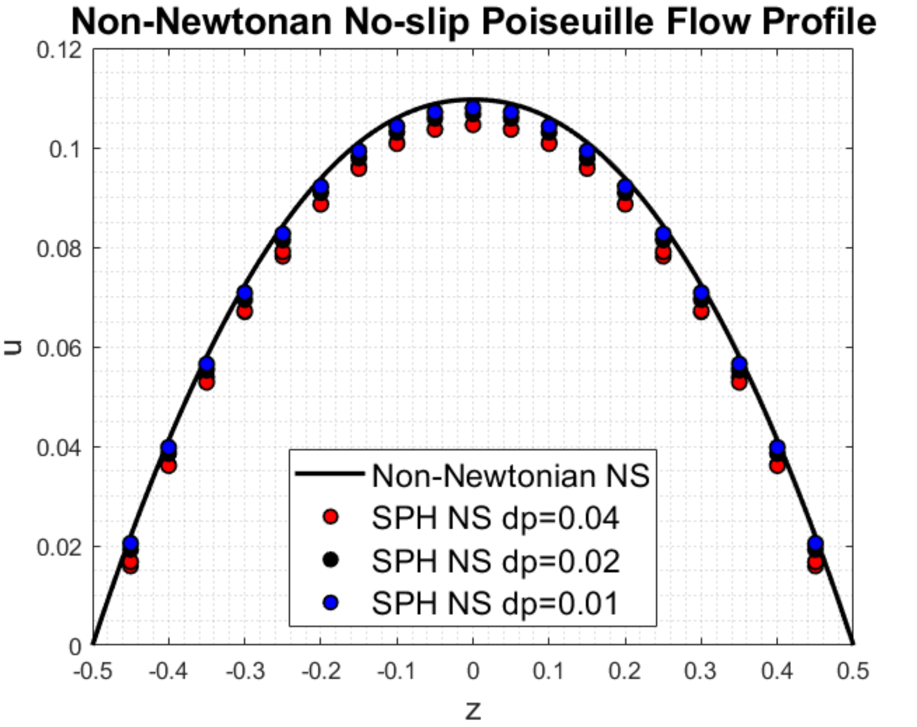


Figure 4.23: Non-Newtonian Poiseuille profile with no-slip boundary conditions showing comparison between analytical solution and SPH data for three resolutions, units in m and ms^{-1}.

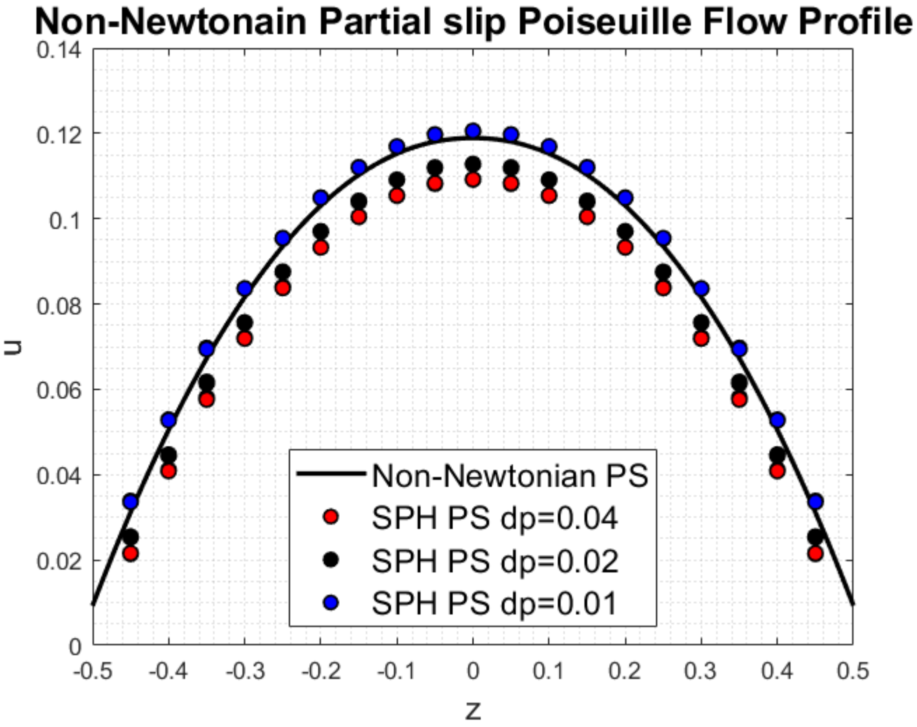


Figure 4.24: Non-Newtonian Poiseuille profile with partial slip boundary conditions showing comparison between analytical solution and SPH data for three resolutions, units in m and ms^{-1}.

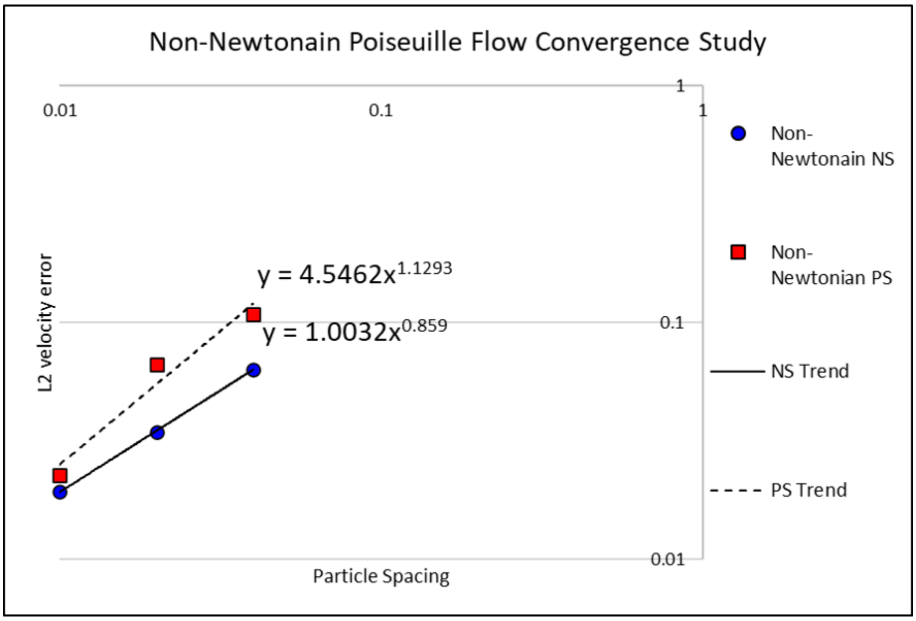


Figure 4.25: Non-Newtonian Poiseuille flow convergence study where NS indicate no-slip flow and PS partial slip with slip length $\lambda_s = 0.01m$.

error in both flows gets smaller as the particle spacing is reduced, which is to be expected from the flow profiles. The larger rate of convergence of the partial slip flow is due to the larger under prediction of the flow velocity for the larger particle spacing flows. While only one slip length and one non-Newtonian fluid have been investigated, this test case is encouraging showing that both MDBC and partial slip can work with non-Newtonian flows.

4.2.2.2 3D Rheometer

The final test case is a numerical flat plate rheometer that has the same design as the one used in Malm (2015). This test case is considered as it is real example of a partial slip flow, the rheometer having been used to experimentally measure partial slip in a number of fluids. The real rheometer consists of two flat plates with fluid sandwiched in the middle, the plates are 3cm in radius and held 500 μm apart, a sketch of the rheometer is shown in Figure 4.26. The bottom plate remains stationary while the top plate spins applying a shear rate to the fluid.

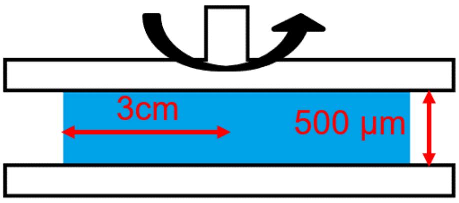


Figure 4.26: Sketch of the rheometer used by Malm (2015).

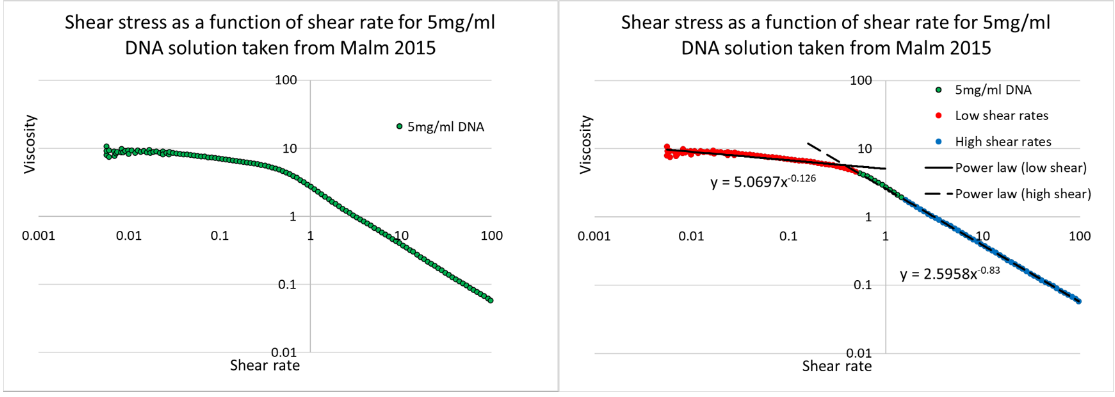


Figure 4.27: Dynamic viscosity of DNA solution (left) and two power law curves used to match the viscosity overlaid on the same plot (right), produced using data extracted from plots in Malm (2015).

Of the many fluid tested in the rheometer, the one with the most complete data that could be reproduced numerically, and displaying slip at the boundaries, was of a DNA solution with concentration of 5mg/mol. With an applied shear rate of $100s^{-1}$ the viscosity was measured to be $0.056Pas$ and the density of the fluid was approximately $1000kgm^{-3}$. The average speed of the top plate was $25mms^{-1}$ resulting in a Reynolds number of around $Re = 0.223$ (Malm (2015)). Malm measured the rheology of the DNA sample, from which the above viscosity was found. Using this data a non-Newtonian model could be fitted to the data, as in Figure 4.27. The DNA solution is shear thinning and the viscosity can be fitted with two power law models depending on the shear rate, as shown in the right of Figure 4.27. However due to very restrictive non-Newtonian viscosity time step and the very large number of particles required to simulate the rheometer in 3 dimensions, this simulation was not possible.

Instead a scaled down Newtonian approximation of the rheometer was simulated instead with comparisons made at steady state when behaviour is predominately Newtonian. The numerical rheometer had a radius of 1mm, a depth of 500 μm and the top plate spins at a speed of $100rads^{-1}$. The particle spacing used was 5×10^{-5} resulting in 26000 particles and the simulation was run for $0.3s$ of physical time. The rest of the numerical parameters used are displayed in Table 4.8. Shifting is not used for this test case as it was found not to be needed due to the short time of simulation required to reach steady state. The simulation was run twice, once with a no-slip bottom plate and again with a partial slip bottom plate with

Table 4.8: Table of numerical parameters for Rheometer flow

dp (m)	h/dp	ν (Laminar)	δ (Molteni)	Shifting	CFL
5×10^{-5}	1.75	0.000056	0.1	OFF	0.1

slip length $\lambda_s = 1.5 \times 10^{-5}m$ calculated using data extracted from Malm (2015). Since the section of rheometer being simulated would have had fluid outside this domain, this was represented by boundary particles to prevent ejection outside the computational domain as the top plate spins.

To ensure that changing the size of the rheometer and measuring the velocity at a different radius would still give a comparable result, the velocity was measured at three radii in the flow. For each radius measured the velocity was normalised by the top plate velocity at that radius and the depth normalised by the total rheometer depth. This same normalisation is used by Malm (2015). The results are plotted in Figure 4.28 and they show good agreement with each other. This shows that the numerical result should be comparable to the experimental one at the chosen radius.

The normalised velocity profile for SPH rheometer with a partial slip plate is plotted alongside the Malm data in Figure 4.29. The SPH data shows good agreement with the Malm data, especially near the bottom partial slip boundary. The two sets of data diverge slightly towards the middle of the flow which could be use to the simplifications made for the numerical model not fully capturing some non-Newtonian effects. The SPH data again shows good agreement near the top rotating boundary where the no-slip condition is known to be well captured by MDBC.

4.2.3 Summary

In this section a new SPH partial slip boundary condition has been presented and validated in two and three dimensions using analytical solutions and experimental data. While the method shows good agreement to analytical solutions for small slip lengths, errors in the flow velocity tend to increase with the size of the slip length. This was caused by an under prediction of the partial slip velocity coming from an under prediction of the velocity gradient. This result could be improved with a higher order velocity gradient calculation. The partial slip condition, and the base MDBC no-slip condition, were shown to be accurate for a non-Newtonian flow showing good agreement with analytical solutions. When compared to a real example of a 3D partial slip flow, a rheometer used to measure slip lengths, the method showed good agreement with experimental readings despite constraining the model to be Newtonian.

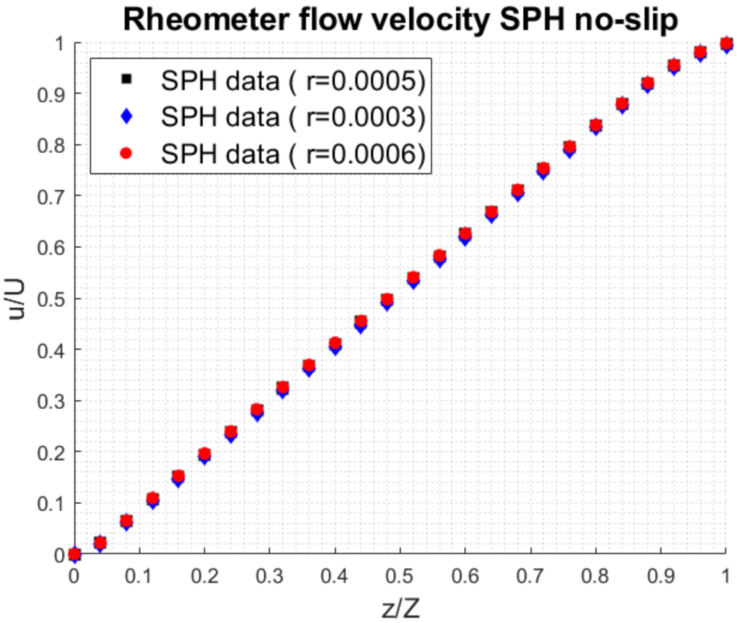


Figure 4.28: Normalised velocity profiles for SPH numerical rheometer with no-slip bottom plate measured at three radii, units are in meters.

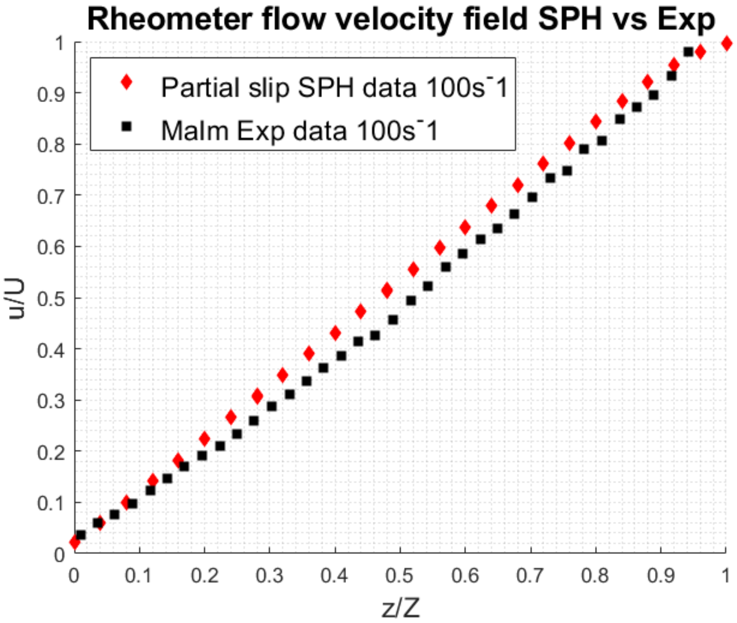


Figure 4.29: Normalised Rheometer velocity profile comparing SPH results with partial slip bottom plate (Red) to data extracted from Malm (2015)(Black).

Chapter 5

Pouring Experiments

All CFD methods need some form of validation to show that they capture the physics of different types of problems. To do this for problems involving the pouring of fluids from containers experimental data is required. The aim of this chapter therefore is to describe and present the results of the experiments completed to validate numerical results for this project. The outcome of the experiments will be the measurement of the mass of fluid poured from two types of container for two different fluids. A range of pouring angles will be used and the mass of fluid left in the container after the pouring process will be found.

5.1 Experimental set up

The experiments took place at Unilever R & D in Port Sunlight with the experimental set up described as follows, with diagrams of the set up included in Figures 5.1 and 5.2. A test vessel filled with fluid is placed on the rotating arm of a pouring rig and held in place by a brace screwed down on top of the vessel. The vessels tested include a plastic beaker filled with 500ml of fluid; and a square based plastic bottle filled with 600ml of fluid, a photo of the vessels is shown in Figure 5.3. A volume of 500ml was chosen for the beaker as it did not completely fill the beaker, and when the beaker was poured it was done so over the back edge so not to use the spout. The bottle was filled to 600ml as is was close to the top of the bottle but not completely full. This avoids the issue of the surrounding air causing the bottle to glug when it is poured. The bottle itself was chosen for it's wide neck, that aids in the prevention of glugging, and it's square base that should be easier to replicate in the CFD model.

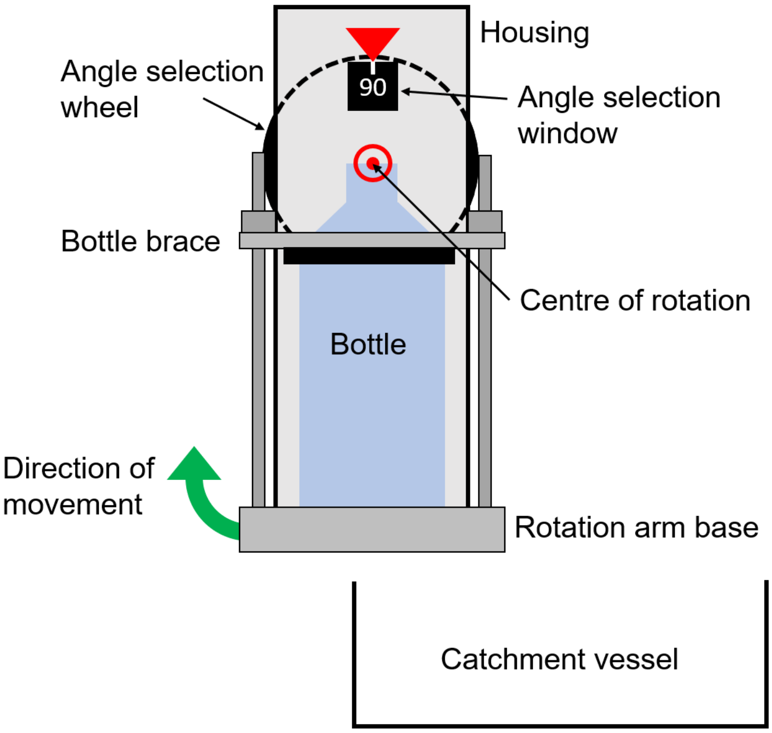


Figure 5.1: Diagram of the experiment set up. The bottle is held in place between the rotation arm base and bottle brace. The whole arm is rotated clockwise around the centre of rotation to the angle set using the angle selection wheel.

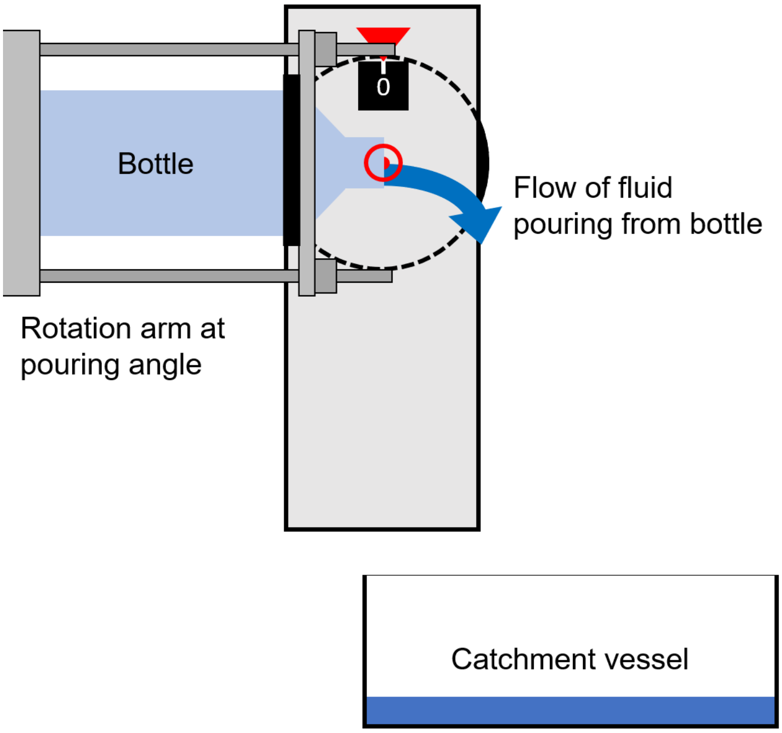

0
Bottle
Flow of fluid pouring from bottle
Rotation arm at pouring angle
Catchment vessel

Figure 5.2: Diagram of the experiment set up after the bottle has been rotated and the fluid poured out of the bottle into the catchment vessel. After a pre-set time the bottle returns to its start position.

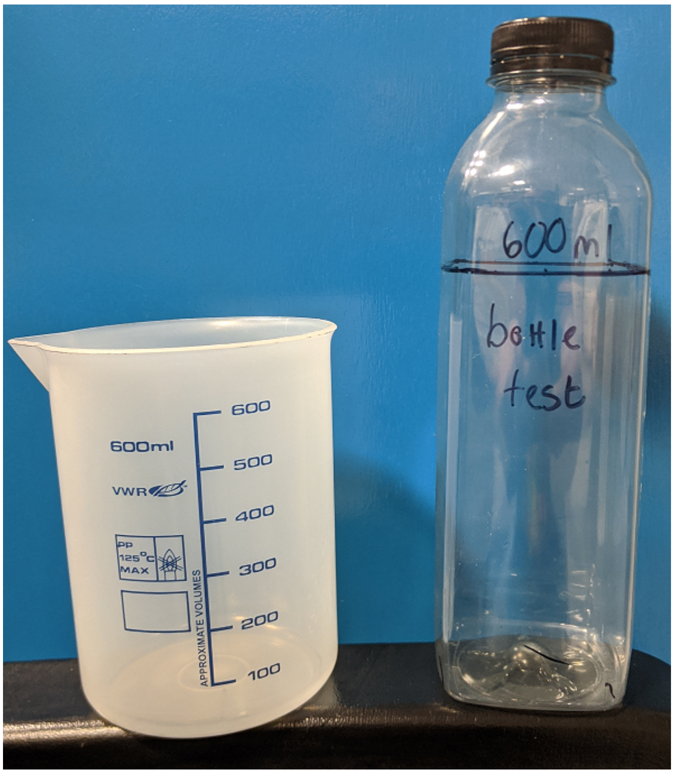


Figure 5.3: The vessels to be tested in the pouring rig. A plastic beaker to be filled to the 500ml line, and a square based plastic bottle to be filled to the 600ml line.

The angle of rotation is set using a selection wheel with increments of 2 degrees, and the rotational speed and hold time are programmed into the rig's control box. For all test cases a speed setting or 2.5 is used, this corresponds to a rotational speed of 2.91 RPM according to the user manual, or 17.46 deg/s. The hold time is the time that the test vessel is held at the desired angle after initial rotation before returning to the starting position, the hold times used for these experiments are stated blow for each case.

Once all the settings have been selected the pouring rig rotates the arm clockwise, and the fluid is poured out of the test vessel into an awaiting catchment vessel to the side of the rig. After the required hold time has been reached the arm rotates anti-clockwise back to its starting position.

5.1.1 Fluids Tested

Two fluids were tested using this set up and using both containers. The fluids tested were tap water taken from taps in the lab and a 0.4% solution of Carbopol, an example of a shear thinning fluid. Carbopol is a water soluble polymer that is often used as a thickener in industry. Two 10l solutions were mixed by diluting a 4% solution to the required concentration followed by adding a basic solution of sodium

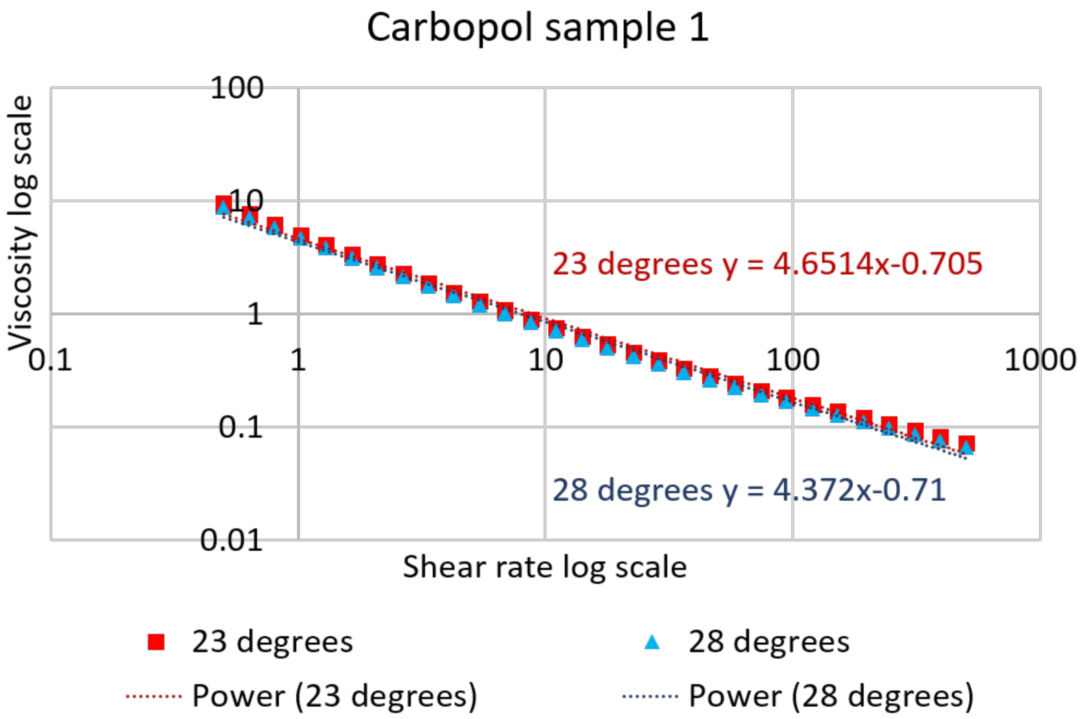


Figure 5.4: Carbopol sample 1: Plot of shear rate vs dynamic viscosity on a log log scale

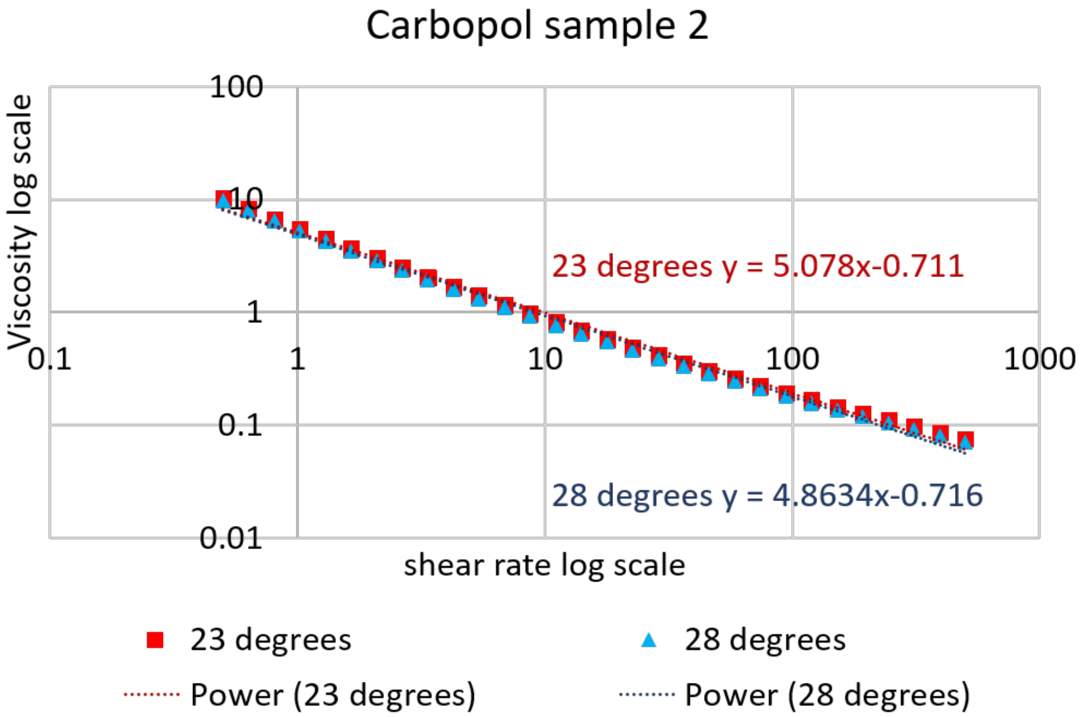


Figure 5.5: Carbopol sample 2: Plot of shear rate vs dynamic viscosity on a log log scale

hydroxide to raise the pH to a reading of 5.3. Increasing the pH of the solution acts to thicken the solution to an acceptable level. Samples of the two formulations were taken and the shear rate, shear stress and dynamic viscosity of the samples were measured using a pot and bob rheometer. The measurements were taken at two temperatures of 23^oC and 28^oC as these cover the temperature range measured in the lab used for the pouring. The measurements for shear rate vs viscosity are plotted in Figures 5.4 and 5.5. As Carbopol is a shear thinning fluid, the effective dynamic viscosity μ_{eff} takes the form

$$\mu_{eff} = K\dot{\gamma}^{n-1}, \tag{5.1}$$

where K is the flow consistency index and n is the flow behaviour index. From the data plotted in Figures 5.4 and 5.4 the values for K and n can be found for each of the samples, these are found in Table 5.1.

Table 5.1: Table of temperatures and flow consistency and flow behaviour indexes from Carbopol rheometer measurements

Carbopol sample 1				Carbopol sample 2			
Temp	K	$n-1$	n	Temp	K	$n-1$	n
23	4.6514	-0.705	0.295	23	5.0780	-0.711	0.289
28	4.3720	-0.710	0.290	28	4.8630	-0.716	0.284

The lab in which the pours took place was not temperature controlled but during the Carbopol pouring had a temperature ranging between 25.5^oC and 26.8^oC measured at 1 hour intervals, which is within the range of the rheometer measurements. Since the values of K and n do not change a large amount across the temperature range, a numerical model can then use the median of the two rheometer readings for each of the samples. For Carbopol sample 1 the values will then be taken as $K = 4.5117$ and $n = 0.2925$ and for Carbopol sample 2 $K = 4.9705$ and $n = 0.2865$. All of the pours using the bottle were done using the Carbopol from the sample 1 and the pours using the beaker from sample 2. After each pour the leftover Carbopol was emptied into a third bucket and not reused due to air bubbles forming in the fluid. Due to the high viscosity of the Carbopol when sat in the bucket these air bubbles took a long time to rise to the surface and so every effort was made to avoid aeration of the Carbopol. After letting the Carbopol sit in the lab overnight the majority of the bubbles would have disappeared allowing for leftover Carbopol to be re-used the following day.

5.1.2 Method

For each individual pour a number of masses were measured in order to find out how much fluid is poured from a container and how much is left over. The following measurements were taken:

- TVBD (Test vessel before decant) this is the mass of the test vessel when it is empty

- TVAD (Test vessel after decant) this is the mass of the test vessel when it has been filled with the test fluid, before pouring

- MJBD (Measuring jug before decant) the mass of a measuring jug used to decant the test fluid before decanting

- MJAD (Measuring jug after decant) the mass of a measuring jug used to decant the test fluid after decanting

- CVBP (Catchment vessel before pouring) the mass of the tub used to catch the poured fluid before pouring

- CVAP (Catchment vessel after pouring) the mass of the tub used to catch the poured fluid after pouring

- TVAP (Test vessel after pouring) mass off the test vessel after the test fluid has been poured

Using the masses TVBD, TVAD, MJBD and MJAD the mass of fluid inside the test vessel can be found, and similarly the mass poured can be found using the masses CVBP, CVAP, TVAD and TVAP. the mass of fluid left in the bottle is then found as the difference between the start mass and the end mass.

For the water cases both the bottle and beaker were filled using a measuring jug before pouring. After the pour and final mass measurements the remaining fluid was poured away and the containers dried where possible before refilling for the next pour. For the Carbopol cases both the beaker and the bottle were filled using 60ml syringes to reduce spillage between the 10l buckets and the containers to be tested and to reduce aeration of the Carbopol in the bottle and beaker. After each pour the remaining Carbopol was poured into a separate bucket and the containers washed with water and dried as much as possible. The beaker was dried with paper towels, the bottle was left upside down to drain before the inside of the neck and shoulder of the bottle dried with paper towel.

5.2 Results

In the following sections the masses found are recorded in tables for each of the fluids and test vessels.

5.2.1 Experimental Error

Due to a issue with the pouring rig, during the pouring involving water the pouring rig would sometimes over rotate the pouring arm beyond the desired angle, however when this happened the angle selection wheel would also continue to rotate and give the reading of the final angle attained when it returned to its starting position. The magnitude of error varied from pour to pour with the final angle being between 2 and 22 degrees above the desired angle. In some cases the pouring arm would rotate 180 degrees thus holding the test vessel upside down and remain there until the rig was turned off and the arm manually rotated back to its start position. This explains the random order or angles used in the tables for water. Before the experiments using the Carbopol were done, the angle selection wheel was tightened fixing this issue. The issue was found to be a loose nut on the angle selection wheel which after being tightened allowed for the selected angle to be reached as required.

The measurement error in the experiments were $\pm 1^o$ for the angle selection and $\pm 0.01g$ for the mass. The large error in the angle was due to the angle selection wheel having increments of 2^o. The window used to select the angle also did not have an accurate selection indicator that may have introduced further error in the angle selected, this can be seen in Figure 5.6.

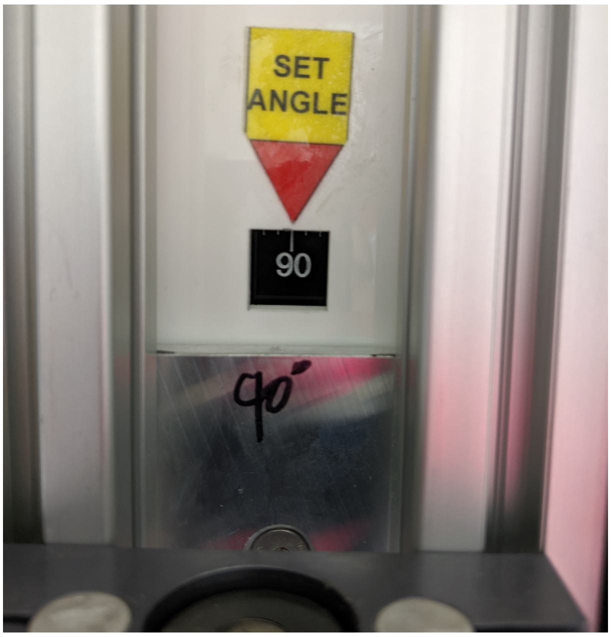


Figure 5.6: Close up image of the angle selection wheel and window.

5.2.2 Water results

The results for pouring using water are organised as follows: Table 5.2 shows the mass and angle measurements for water poured from a plastic beaker filled to the 500ml line and the data is plotted in Figure 5.7; Table 5.3 shows the mass and angle measurements for water poured from a plastic bottle filled to the 600ml line and the data is plotted in Figures 5.8 and 5.9. The data for the bottle is split between two plots so that all the data can be clearly shown, the first plot shows angles below 100 degrees and the second above 100 degrees.

Table 5.2: Table of mass measurements, percentage mass left, pouring angles and hold times (T) for pouring of water from a plastic beaker

#	Start mass(g)	Poured mass(g)	Remainder	% left	Angle(deg)	T(s)
1	499.95	489.75	10.21	2.04	90	1
2	500.13	480.34	19.75	3.95	90	1
3	500.87	477.81	23.07	4.60	88	1
4	500.39	464.21	36.18	7.23	86	1
5	500.59	500.20	0.40	0.08	96	1
6	497.82	497.41	0.41	0.08	92	1
7	499.07	476.48	22.59	4.53	88	1
8	494.77	485.70	9.07	1.83	90	1
9	500.50	498.86	1.64	0.33	92	1
10	500.00	491.47	8.53	1.71	90	1
11	495.77	495.48	0.29	0.06	96	1
12	503.84	503.41	0.43	0.09	96	1
13	502.77	502.47	0.30	0.06	100	1
14	494.78	443.67	51.11	10.33	84	1
15	499.57	490.03	9.54	1.91	90	1
16	498.63	490.75	7.88	1.58	90	1
17	498.96	463.04	35.92	7.20	86	1
18	496.92	487.71	9.21	1.85	90	1

Plotting the data highlights a number of outlying data points: one point for the beaker pours with 90 degree pour angle, and two points for the bottle pours with pour angles or 94 and 112 degrees respectively. These irregularities are likely down to the over rotation issue seen with the pouring rig. After selecting the angle and setting the arm to rotate the angle selection wheel would stick during the rotation and causing the arm to over rotate somewhere between 2 and 22 degrees. The arm would then stop and hold for the prescribed time before rotating back to its start position. The angle selection wheel would then indicate the final angle reached, however there could have been an error in this value resulting in the outlying data points. For example the mass of the beaker outlier matches more closely with the

Table 5.3: Table of mass measurements, percentage mass left, pouring angles and hold times (T) for pouring of water from a plastic bottle

#	Start mass(g)	Poured mass(g)	Remainder	% left	Angle(deg)	T(s)
1	603.13	340.51	262.63	43.54	92	3
2	600.67	388.81	211.86	35.27	94	3
3	599.90	287.80	312.10	52.03	90	3
4	604.15	186.63	417.52	69.11	86	3
5	603.70	238.03	365.67	60.57	88	3
6	601.49	289.67	311.82	51.84	90	3
7	603.23	257.16	346.07	57.37	94	3
8	602.05	438.27	163.78	27.20	96	3
9	601.83	438.65	163.19	27.11	96	3
10	601.62	289.36	312.26	51.90	90	3
11	600.00	287.39	312.08	52.01	90	3
12	602.76	188.92	413.84	68.66	86	3
13	604.64	136.66	467.98	77.40	84	3
14	604.85	239.96	364.89	60.33	88	3
15	602.92	387.05	215.87	35.80	94	3
16	597.86	593.40	4.46	0.75	118	10
17	600.60	593.33	7.26	1.21	116	10
18	600.77	584.44	16.33	2.72	112	10
19	603.36	599.93	3.44	0.57	120	10
20	603.62	598.36	5.26	0.87	116	10
21	601.03	592.91	8.12	1.35	114	10
22	603.70	598.31	5.39	0.89	116	10
23	605.85	596.20	9.66	4.59	110	10
24	601.67	594.45	7.23	1.20	114	10
25	602.64	590.98	11.66	1.93	108	10
26	602.16	597.55	4.61	0.77	118	10
27	605.12	599.45	5.66	0.94	116	10
28	604.05	599.84	4.21	0.70	118	10
29	601.53	595.23	6.29	1.05	116	10
30	602.71	595.96	6.75	1.12	114	10
31	604.21	600.50	3.71	0.61	118	10
32	605.72	601.68	4.04	0.67	120	10
33	600.90	597.21	3.68	0.61	120	10
34	599.12	595.76	3.36	0.56	122	10
35	602.25	599.60	2.65	0.44	122	10

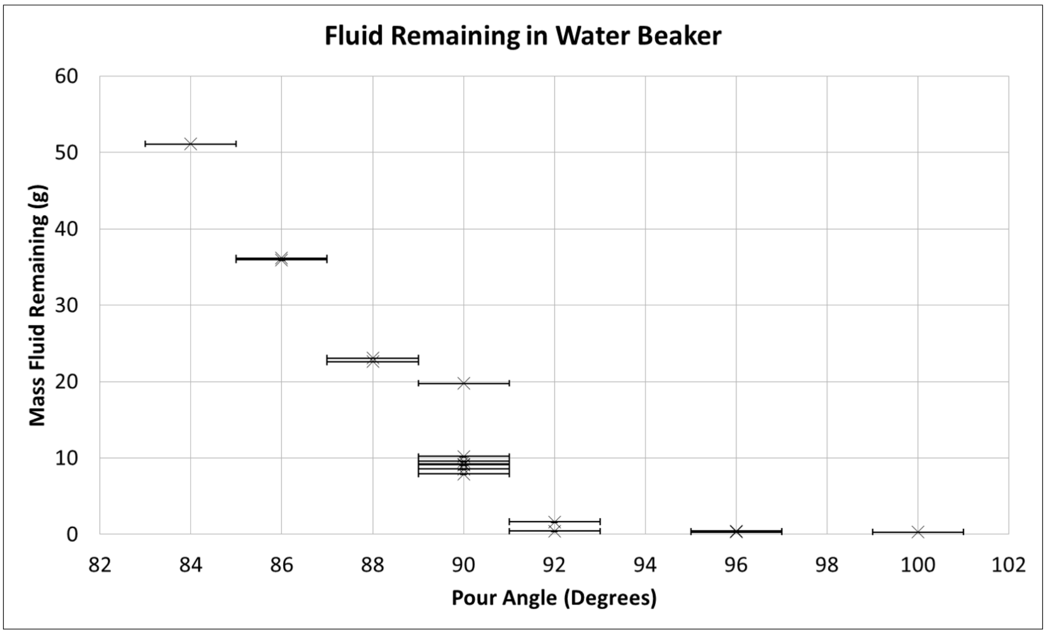


Figure 5.7: Plot showing the mass of water remaining in the beaker after pouring with error bars

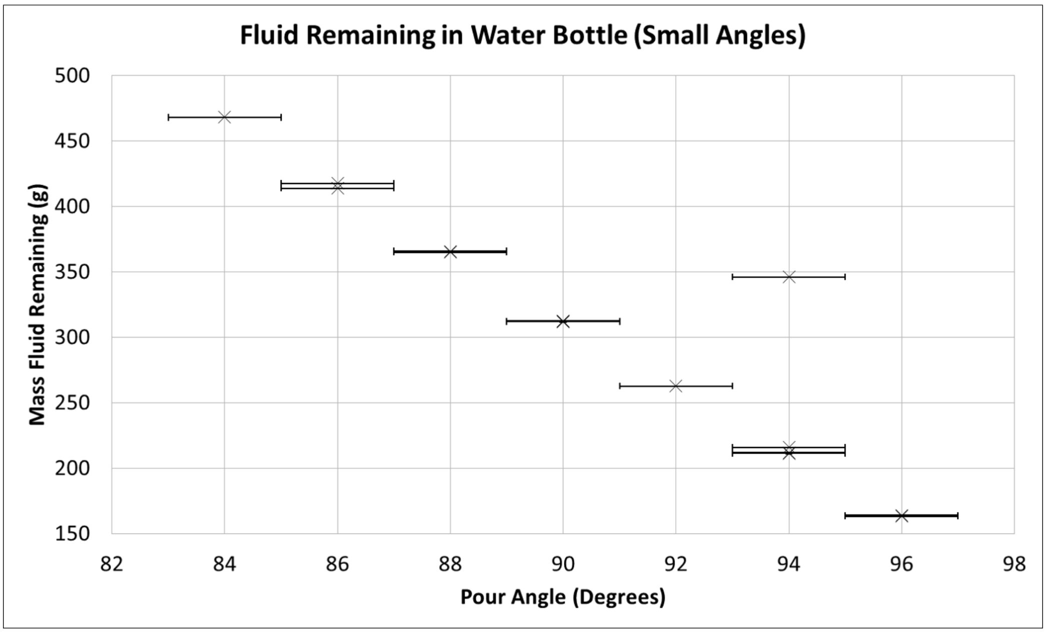


Figure 5.8: Plot showing the mass of water remaining in the bottle after pouring for small angles less than 100 degrees with error bars

pouring angle of 88 degrees than 90, so the final pouring was likely closer to 88 than to 90.

From the results it can be seen that a larger pouring angle results in less fluid remaining in the container after the pouring has finished. This is to be expected. For beaker angles over 90^o result in the beaker being close to empty after the pouring has finished with only a few droplets of water remaining. The bottle however never fully empties, this is due to a small amount of liquid getting trapped in the shoulder of the bottle towards the end of the pouring. A larger pouring angle could over come this but could result in the bottle glugging which is not desired.

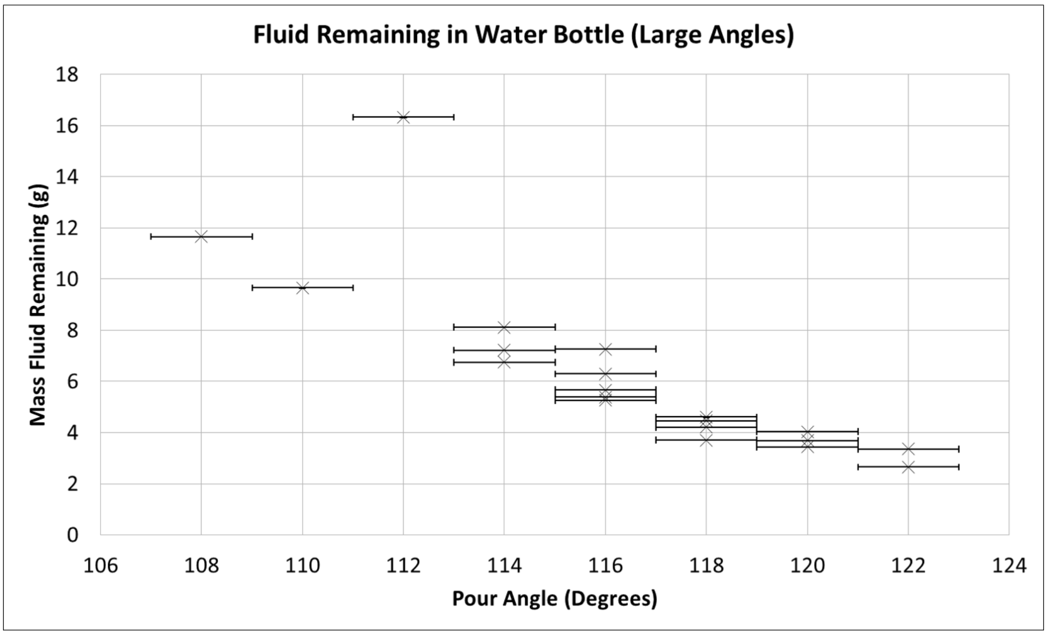


Figure 5.9: Plot showing the mass of water remaining in the bottle after pouring for large angles over 100 degrees with error bars

5.2.3 Carbopol results

The results for pouring using the Carbopol are organised as follows: Table 5.4 shows the mass and angle measurements for Carbopol poured from a plastic beaker filled to the 500ml line and the data is plotted in Figure 5.10; Table 5.5 shows the mass and angle measurements for Carbopol poured from a plastic bottle filled to the 600ml line and the data is plotted in Figure 5.11. Plotting the data shows no outlying data points, an improvement over the pours involving water. This was achieved after finding out that the angle selection wheel could be tightened and thus avoiding the over rotation issue. The wheel was re-tightened each time the angle was changed to ensure the correct pouring angle was attained.

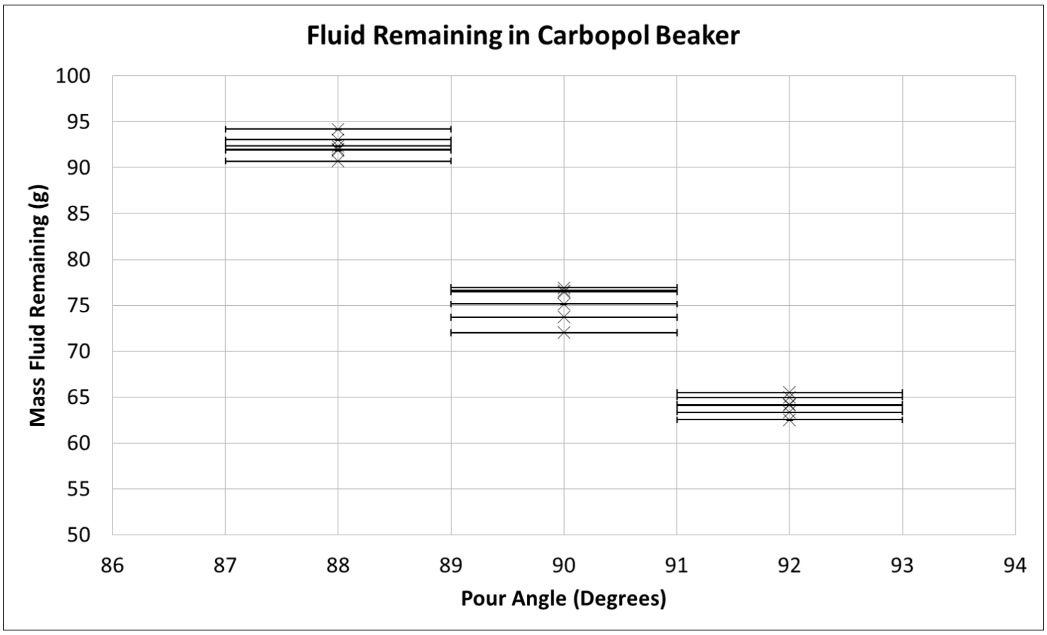


Figure 5.10: Plot showing the mass of Carbopol remaining in the beaker after pouring with error bars

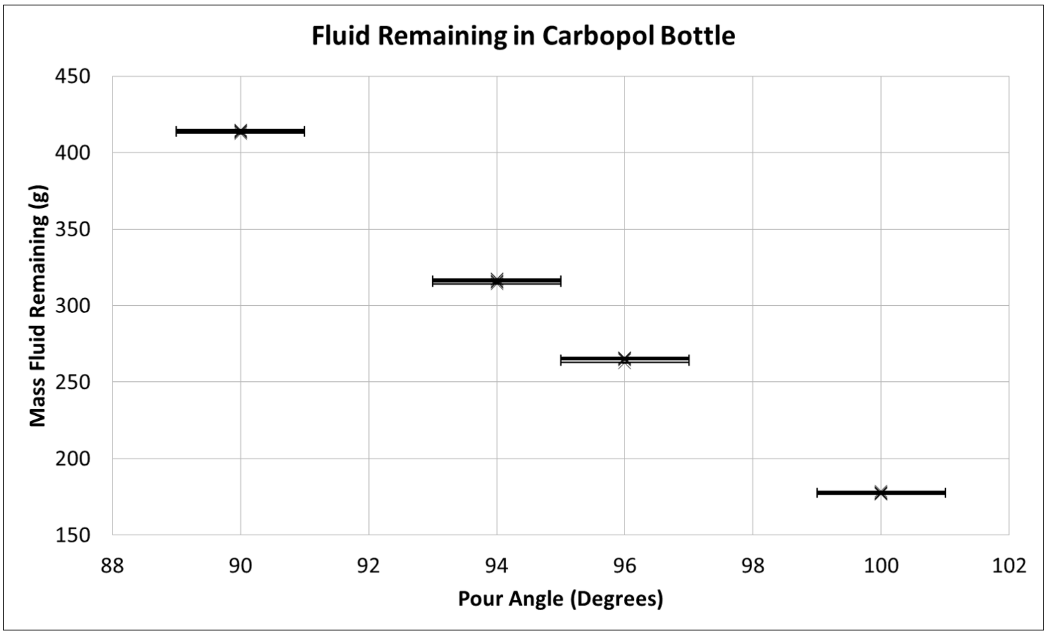


Figure 5.11: Plot showing the mass of Carbopol remaining in the bottle after pouring with error bars

Table 5.4: Table of mass measurements, percentage mass left, pouring angles and hold times (T) for pouring of Carbopol from a plastic beaker

#	Start mass(g)	Poured mass(g)	Remainder	% left	Angle(deg)	T(s)
1	502.10	410.23	91.88	18.30	88	2
2	504.41	412.42	92.00	18.24	88	2
3	505.71	414.84	90.72	17.94	88	2
4	503.60	410.47	93.04	18.47	88	2
5	507.17	412.99	94.19	18.57	88	2
6	498.45	421.84	76.61	15.37	90	2
7	503.71	426.79	76.92	15.27	90	2
8	501.86	425.41	76.46	15.23	90	2
9	500.32	428.31	72.02	14.39	90	2
10	498.76	425.06	73.71	14.78	90	2
11	501.16	436.21	64.96	12.96	92	2
12	501.91	438.57	63.35	12.62	92	2
13	499.26	433.76	65.50	13.12	92	2
14	500.23	436.04	64.20	12.83	92	2
15	500.22	437.69	62.54	12.50	92	2

The results follow the trend of larger pouring angle resulting in less fluid remaining in the container after the pouring is finished, which is to be expected. Neither the beaker nor the bottle fully emptied for the pouring angle chosen, even when rotated beyond the horizontal. The beaker could be emptied further if a larger pouring angle was used or the hold time increased above the 2 seconds used. However due to the high viscosity at low shear rates this will likely a long time. Close to the start of the pouring a large mass of the fluid will be moving together causing high shear near the container surface and thus a lowering of the viscosity. But when the fluid layer gets thinner as the test vessel is held in place, the viscosity will increase again and the amount of fluid that can flow out of the vessel will decrease. The bottle could also have been emptied further by using larger angles or hold times. However the bottle will also suffer from some fluid being frapped in the shoulder of the bottle. Unfortunately due to the limited supply of Carbopol further tests could not be performed to investigate larger angles and hold times.

Table 5.5: Table of mass measurements, percentage mass left, pouring angles and hold times (T) for pouring of Carbopol from a plastic bottle

#	Start mass(g)	Poured mass(g)	Remainder	% left	Angle(deg)	T(s)
1	602.58	189.90	412.69	68.49	90	6
2	600.09	186.44	413.66	68.93	90	6
3	607.77	193.53	414.25	68.16	90	6
4	610.17	196.85	413.32	67.74	90	6
5	604.48	189.47	415.02	68.66	90	6
6	600.57	186.55	414.03	68.94	90	6
7	606.75	292.52	314.23	51.79	94	6
8	606.65	289.74	316.92	52.24	94	6
9	605.23	289.69	315.34	52.12	94	6
10	603.23	287.53	315.70	52.33	94	6
11	602.42	286.42	316.01	52.46	94	6
12	603.86	286.56	317.31	52.55	94	6
13	607.84	341.92	265.92	43.75	96	6
14	605.63	339.99	265.65	43.86	96	6
15	605.28	339.79	265.49	43.86	96	6
16	607.06	344.31	262.75	43.28	96	6
17	614.64	349.81	264.83	43.09	96	6
18	609.48	344.93	264.55	43.31	96	6
19	606.41	429.56	176.85	29.16	100	6
20	607.62	429.98	177.65	29.24	100	6
21	604.64	426.79	177.86	29.42	100	6
22	607.99	429.36	178.64	29.38	100	6
23	608.70	431.26	177.44	29.15	100	6
24	604.01	427.35	176.67	29.25	100	6
25	616.63	503.14	113.49	18.40	104	6
26	617.88	506.48	111.41	18.03	104	6
27	613.63	499.99	113.65	18.52	104	6
28	612.14	498.60	113.55	18.55	104	6
29	622.67	509.07	113.60	18.24	104	6
30	615.59	502.05	113.54	18.44	104	6

5.3 Summary

In this Chapter the experimental method and results of pouring flows involving two fluids and two test container shapes have been presented. The results of remaining mass vs pouring angle have been plotted and any anomalous results highlighted, and the reasons for them explained. As expected the results show that larger pouring angles results in less fluid remaining in the test container. For cases involving water neither the beaker nor the bottle could be fully emptied with only a small number of droplets remaining in the beaker. The bottle could not be fully emptied due to fluid being trapped in the shoulder section of the bottle, and larger angles that would have been needed to overcome this possibly resulting in undesired glugging. The cases involving Carbopol also did not fully empty, possibly due to pouring angles and hold times not being large enough to do so. However a limited supply of Carbopol restricted further pouring. The rheology of two samples of Carbopol have been analysed and the data required to model the non-Newtonian behaviour extracted. Using this data a non-Newtonian SPH model can be tested and validated for the use of pouring flows, however due to the computational expensive of Non-Newtonian models in SPH this can not be presented in this thesis.

Chapter 6

Complex Applications

In this Chapter the MDBC method will be further tested using three more complex applications than was included in Chapter 4, with comparisons to experimental results.

6.1 3D Dam Break

The first test case is a 3D dam break over a cuboid obstacle, this is SPHERIC Benchmark test case #2 with comparison to the experimental data of Kleefsman et al. (2005). The simulation set-up is shown in Figure 6.1 showing a column of fluid that falls under gravity before impacting the obstacle at the opposite end of the tank. The fluid depths are measured at the points $H1-4$. When the fluid impacts the obstacle the pressure is measured at the pressure gauges $P1-8$ shown in Figure 6.2. This is an interesting test case as it allows testing of the model to predict accurate free surface elevations and pressure measurements of fluid impacting on a structure. The simulation is run for 6 seconds of physical time with the numerical parameters displayed in Table 6.1. The MDBC approach used for this example is for density extrapolation only. Shifting is not used for this test case of a highly transient flow.

Table 6.1: Table of numerical parameter for 3D dam break simulations.

dp (m)	h/dp	α (Artificial)	δ (Fourtakas)	Shifting	CFL
0.02	2	0.05	0.1	OFF	0.2
0.01	2	0.05	0.1	OFF	0.2

A time instant at $t = 0.8s$ is shown in Figure 6.3 comparing the DBC result on the top and MDBC result on the bottom. There is a clear difference in the flow over

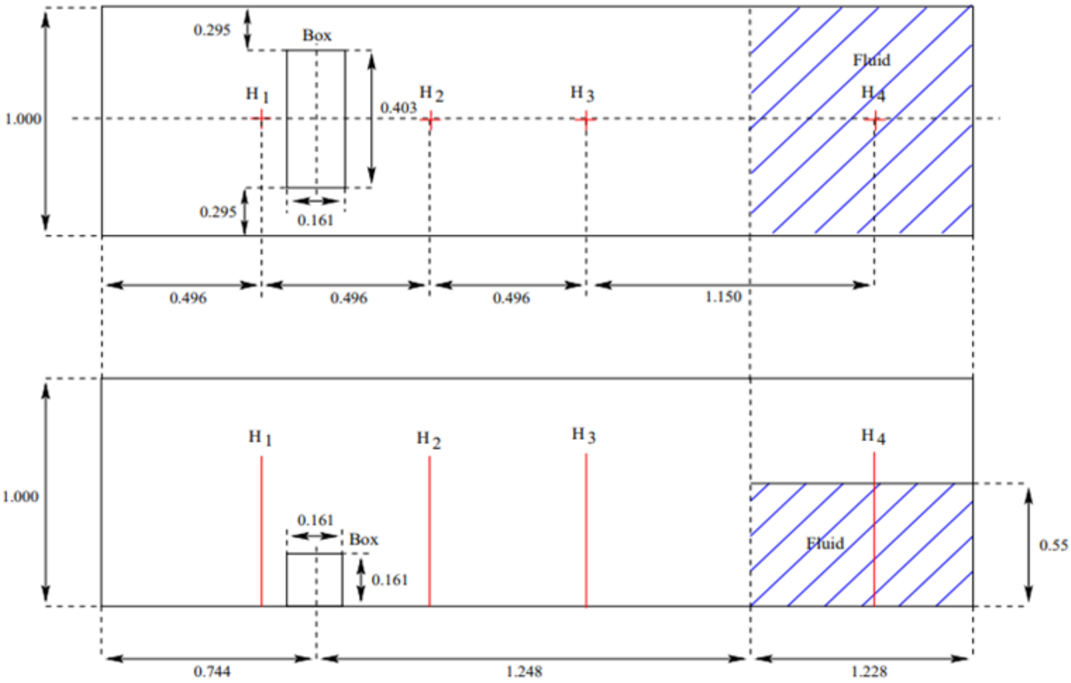


Figure 6.1: Initial set up of 3D damn break with obstacle showing locations of surface elevation measurements $H1-4$.

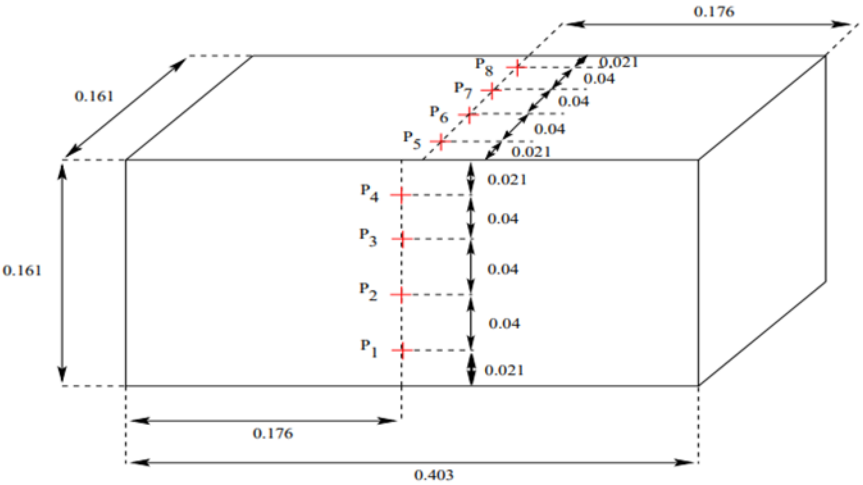


Figure 6.2: Location of the pressure measurement points $P1-8$ on the surface of the obstacle.

the top of the obstacle between the boundary conditions. The gap created by DBC can also be seen along the sides of the top image in front of the original location of the water column.

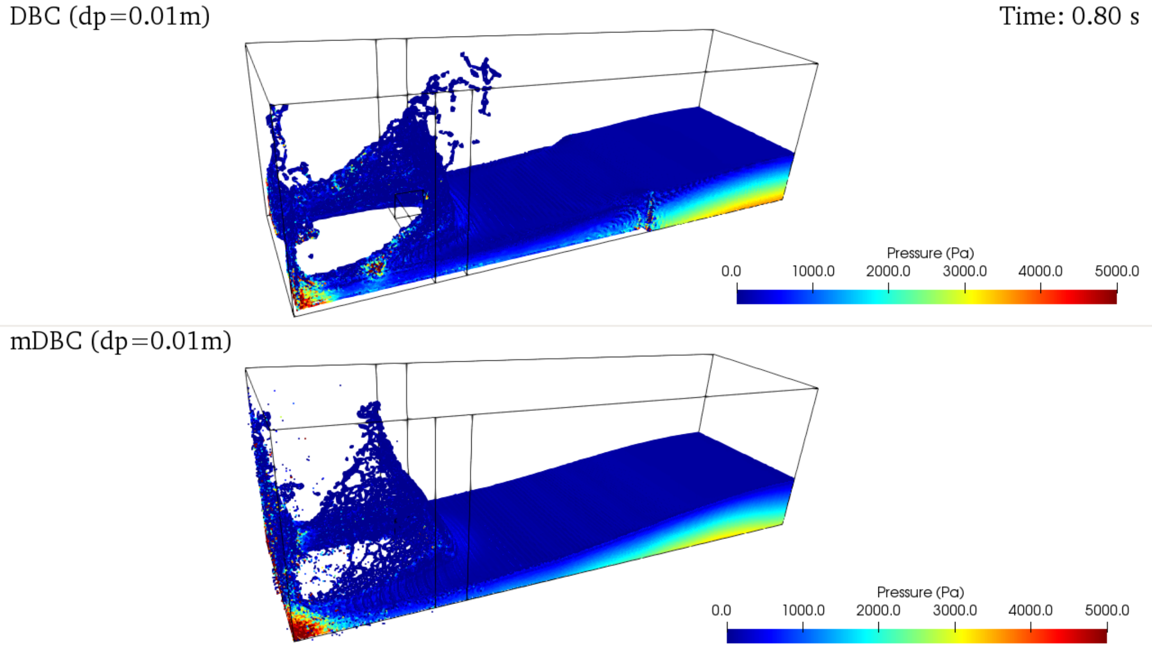


Figure 6.3: Instant of dam break after impact with obstacle at $t = 0.8s$ with DBC on the top and MDBC on the bottom. Particles coloured by pressure.

Figure 6.4 shows plots of the surface elevation versus times at three of the measurement points: H2; H3 and H4. The result for both resolutions using MDBC are shown along with the finer resolution DBC result, these are both compared to the experimental data. Measurement point H4 is located in the initial column of fluid, both the DBC and MDBC surface elevations show good agreement with the experimental results with MDBC giving a closer agreement. This is also true for the measurements at location H3 which is around half way between the initial water column and the obstacle. The MDBC measurements also show good agreement with experiment at location H2, in front of the obstacle, the DBC result however shows a large spike in surface elevation at around the time of impact before returning down and following the experimental result again.

Figure 6.5 shows plots of pressure versus time measured at pressure gauges P2, located on the front of the obstacle, and P5, located on the top face close to the corner. The MDBC measurement at P2 is again close to the experiment and generally closer than the DBC result. A closer look at the peak pressures measured at P2, shown in the middle plot, show that DBC greatly under predicts the peak pressure, where as MDBC gives good agreement. It should be noted that the numerical

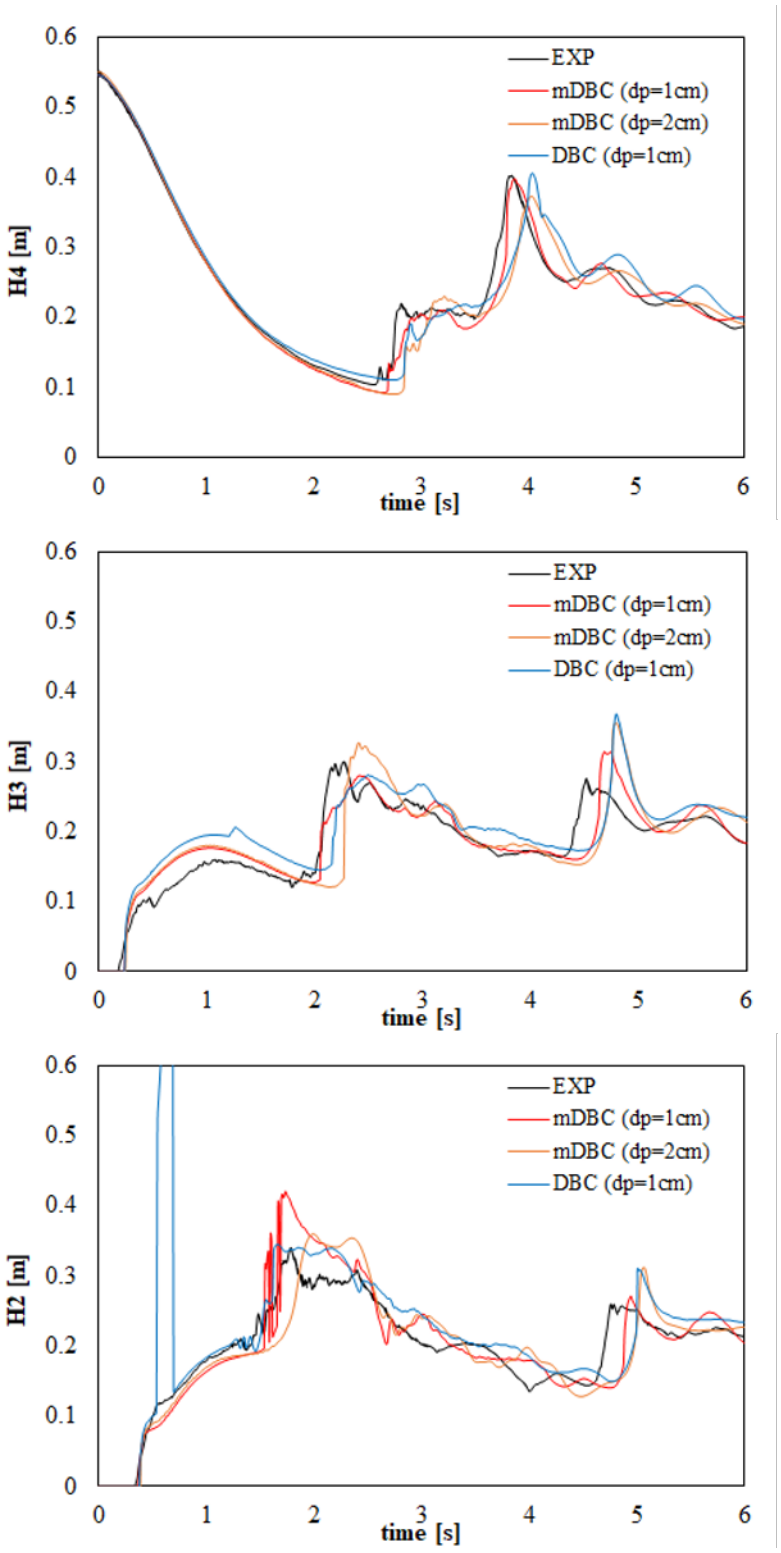


Figure 6.4: Plots of surface elevation versus time measured at locations H4 (top), H3 (middle) and H2 (bottom) for the two MDBC resolutions and the finer DBC resolution compared to experimental result.

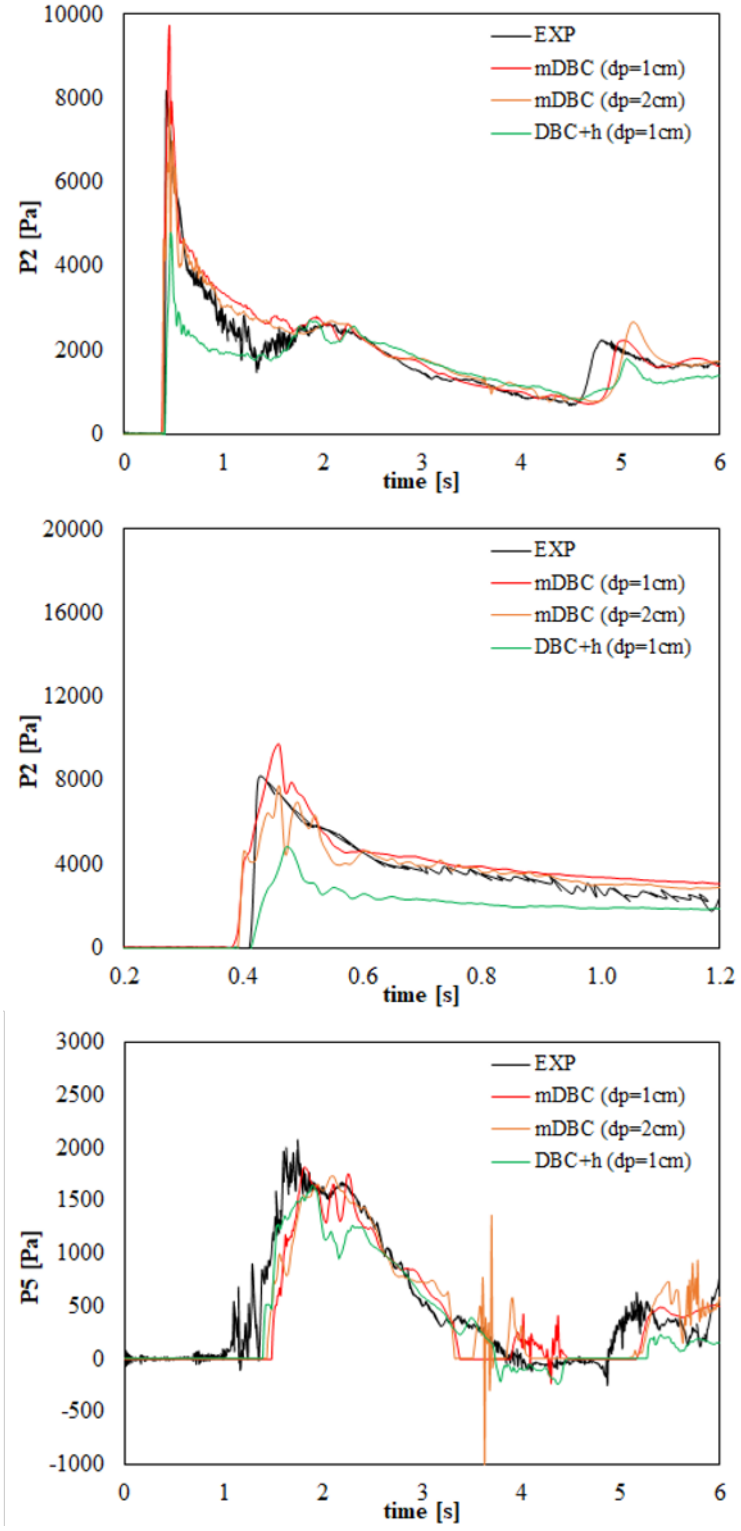


Figure 6.5: Plots of surface elevation versus time measured at locations P2 (top and middle) and P5 (bottom) for the two MDBC resolutions and the finer DBC resolution compared to experimental result. The middle plot shows a close up view of the pressure at P2 between 0.2s and 1.2s during the impact of the water on the obstacle.

pressure probe used for the DBC measurements was placed a distance h away from the boundary to allow for this measurement to be taken. At P5 the finer resolution MDBC and DBC results show good agreement with the experiment. The coarser resolution MDBC however shows some large fluctuations in the pressure that are not seen for the finer resolution.

Computational Cost The run times for the sloshing tank test case on a GPU for both DBC and MDBC with particle spacing dp=0.002m are shown in Table 6.2, along with a ratio of the run times showing the increased computational cost of MDBC on selection of hardware.

Table 6.2: Table of run times of 3D Dam Break using different hardware.

Hardware	dp(m)	DBC(s)	MDBC(s)	MDBC/DBC
CPU i7-6700K	0.02	9998	12442	1.24
Tesla K40	0.02	624	768	1.23
Ge Force RTX 2080 Ti	0.02	224	252	1.12
Tesla V100	0.02	171	190	1.11
Tesla K40	0.01	7366	8700	1.18
Ge Force RTX 2080 Ti	0.01	1406	1586	1.13
Tesla V100	0.01	1050	1180	1.12

This test case demonstrates that SPH with MDBC can predict highly transient free surface flow and pressures quite accurately.

6.2 Fish Pass

The second test case is a numerical fish pass with comparisons to data from experiments run at the University of Parma. A diagram of the numerical fish pass used in shown in Figure 6.6.

The fish pass comprises of a 300mm wide channel tilted at a 4.43^o angle split into four 625mm long pools separated by 20mm thick walls. Each wall includes a gate of height 75mm and width 80mm in one of the bottom corners and a weir of height 177mm (from the bottom of the wall) and width 80mm in the opposite top corner. The orientation of the gates and weirs swaps between each wall as shown in Figure 6.6. The walls can be changed so that they include only gates, only weirs or both gates and weirs. By changing the set up of the walls and the flow rate into the fish pass, the flow through the gates and the weirs can be studied. During the experiments the water depth in each pool was measured in the centre of each pool, measuring from the top of the weir in wall no.3. Table 6.3 shows the flow rates,

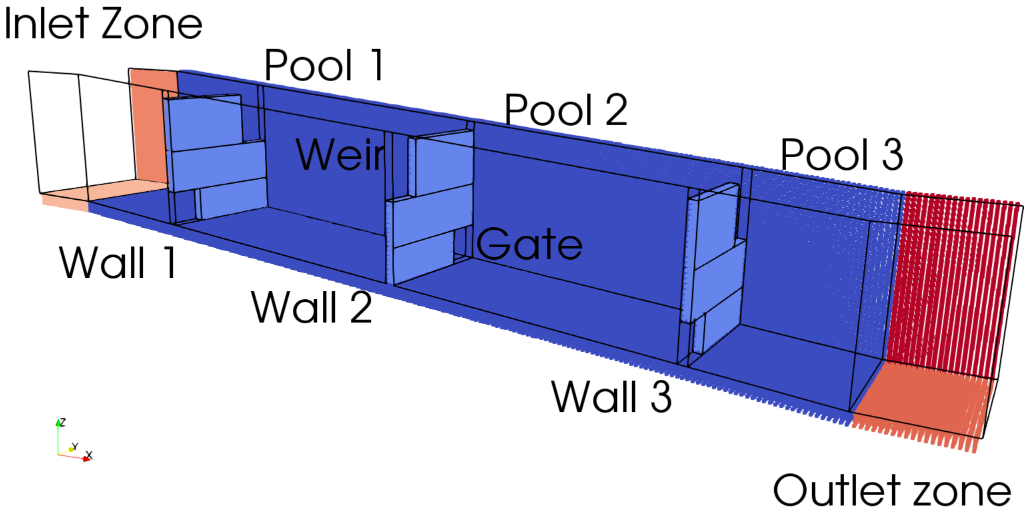


Figure 6.6: Diagram of the numerical fish pass showing the locations of the walls and the gates and weirs. The numbering system used for the pools and walls is also shown.

pool depths and wall set ups of interest to the current work, where $H(in)$ is the water depth in the most upstream pool.

Table 6.3: Table of flow rate, pool depths and baffle set ups for fish pass simulations. Water heights measured from the top of weir no.3.

Q#	Q (l/s)	Set up	$H(in)$(mm)	$H(p_1)$(mm)	$H(p_2)$(mm)	$H(p_3)$(mm)
Q1	1.7	Gate Only	3.9	-10.4	-24.9	-37.2
Q2	2.45	Gate Only	46.2	15.9	-15.7	-44.9
Q3	2.267	Weir Only	166	115	63	13
Q4	1.963	Weir Only	160	109	57	7
Q5	1.634	Weir Only	154	103	51	1
Q6	6.0	Gate & Weir	169.9	117.9	78.0	38.4

The SPH fish pass uses the inlet-outlet boundary condition in the upstream and downstream pools as shown in Figure 6.6 to deal with the flow into and out of the fish pass. The water depths in each pool were set to the experimental values from Table 6.3, with $H(in)$ used for the inlet and $H(p_3)$ for the outlet. The inlet and outlet zones were prescribed an accelerated linear velocity across the depth of the pool, according to the water depth and flow rate in the inlet and outlet pools. Each simulation was run for 10 seconds of physical time reaching a steady state, and the numerical parameters used are displayed in Table 6.4. For most of the flow rates only the two coarsest resolutions were used with the exception of Q2 in which a third particle spacing was run. In the results the DBC method is compared to the density only extrapolation MDBC method. Shifting is included for these test cases to aid the particle distribution and avoid clumping, although the overall effect was minimal .

Table 6.4: Table of numerical parameter for fish pass simulations

dp (m)	h/dp	α (Artificial)	δ (Fourtakas)	Shifting	CFL
0.01	1.3	0.001	0.1	ON	0.2
0.005	1.3	0.001	0.1	ON	0.2
0.0025 (Q2 only)	1.3	0.001	0.1	ON	0.2

Gate only flows - For the gate only flows, denoted by Q1 and Q2 in Table 6.3, the water levels are very low and so the water flows through only the gates. An example of the flow showing particle velocity at the end of the 10 seconds is shown in Figure 6.7 for flow rate Q2 with a particle spacing of $dp = 0.005m$. The depths in pools 1 and 2 and the flows rates through gates 2 and 3 were measured during the simulation and compared to the experimental result, theses results are shown in Figures 6.8 and 6.9 for flow rate Q2, all plots averaged using a moving mean over an interval of 15 values to smooth out the result.

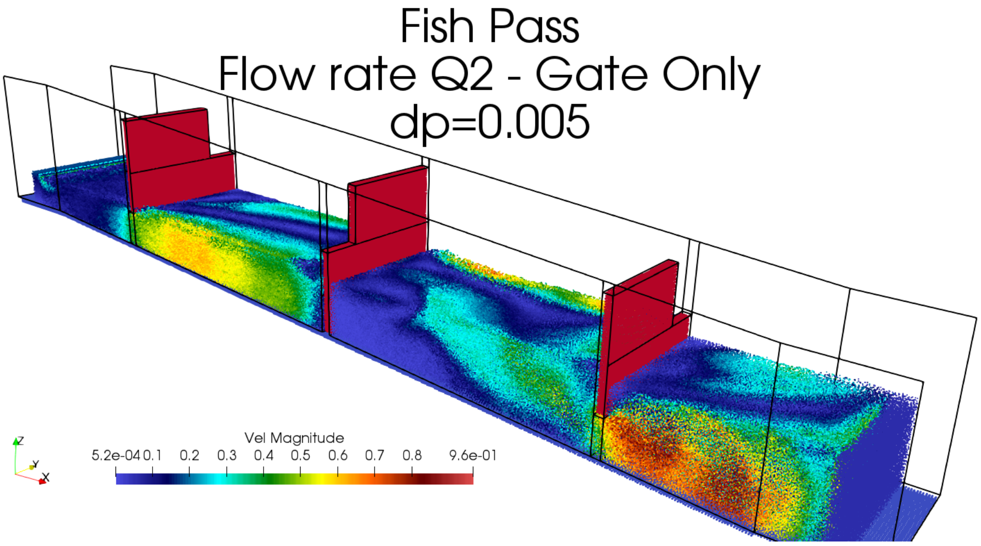


Figure 6.7: Fish pass flow through gates only for flow rate Q2 and particle spacing $dp = 0.01m$, fluid particles coloured by velocity magnitude.

For flow rate Q2, three particle spacings were simulated comparing the DBC and MDBC boundary conditions. Figure 6.8 shows the SPH water depths measured in pools 1 and 2 compared to the experimental data. The blue lines correspond to a particle spacing of $dp = 0.01m$, red to $dp = 0.005m$ and green to $dp = 0.0025m$, for each particle spacing the darker coloured line is using DBC and the lighter MDBC.

The water levels in pool 1 get closer to the experimental depth as the particle spacing is refined, once the flow has reached steady state there is little difference between the DBC and MDBC results for each resolutions. The water levels in pool 2 get larger as the particle spacing is decreased with the closest result coming from the $dp = 0.01m$ MDBC simulation. It should be noted however that the difference in depth between the $dp = 0.005m$ result and the experiment is around $1dp$ and the difference for $dp = 0.0025m$ around $2dp$, so overall still a reasonable result.

The flow rates measured through gates 2 and 3 are plotted in Figure 6.9. These plots show that both the DBC and MDBC flows give good agreement to the experimental result for all three resolutions once the flow reaches steady state. The results for flow rate Q1 are similar to those shown for Q2 with flow rates giving overall good agreement and the pool depths having a difference of around $1 - 2dp$ from the experimental readings. The plots produced for flow rate Q1 are included in Appendix B.

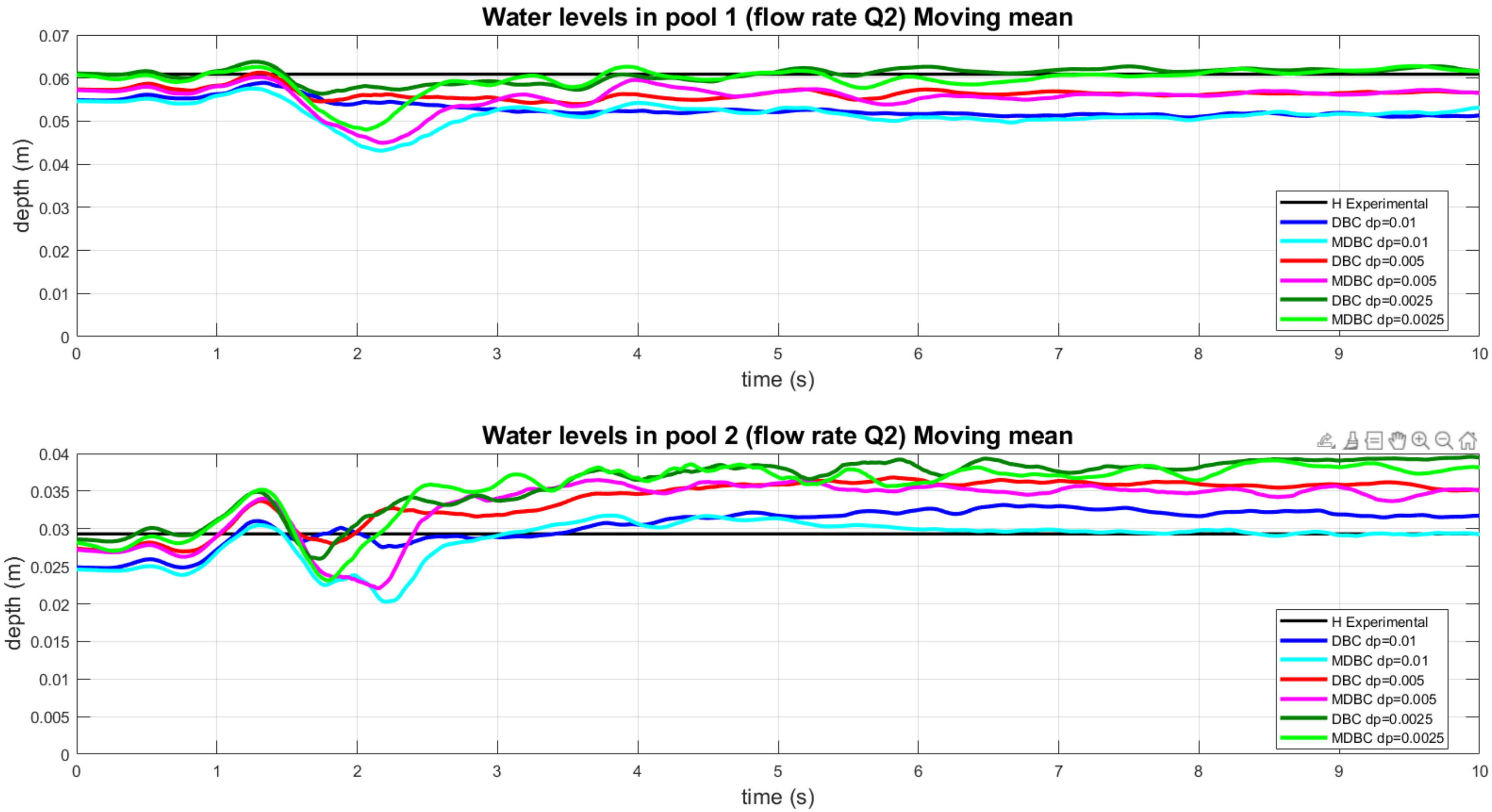
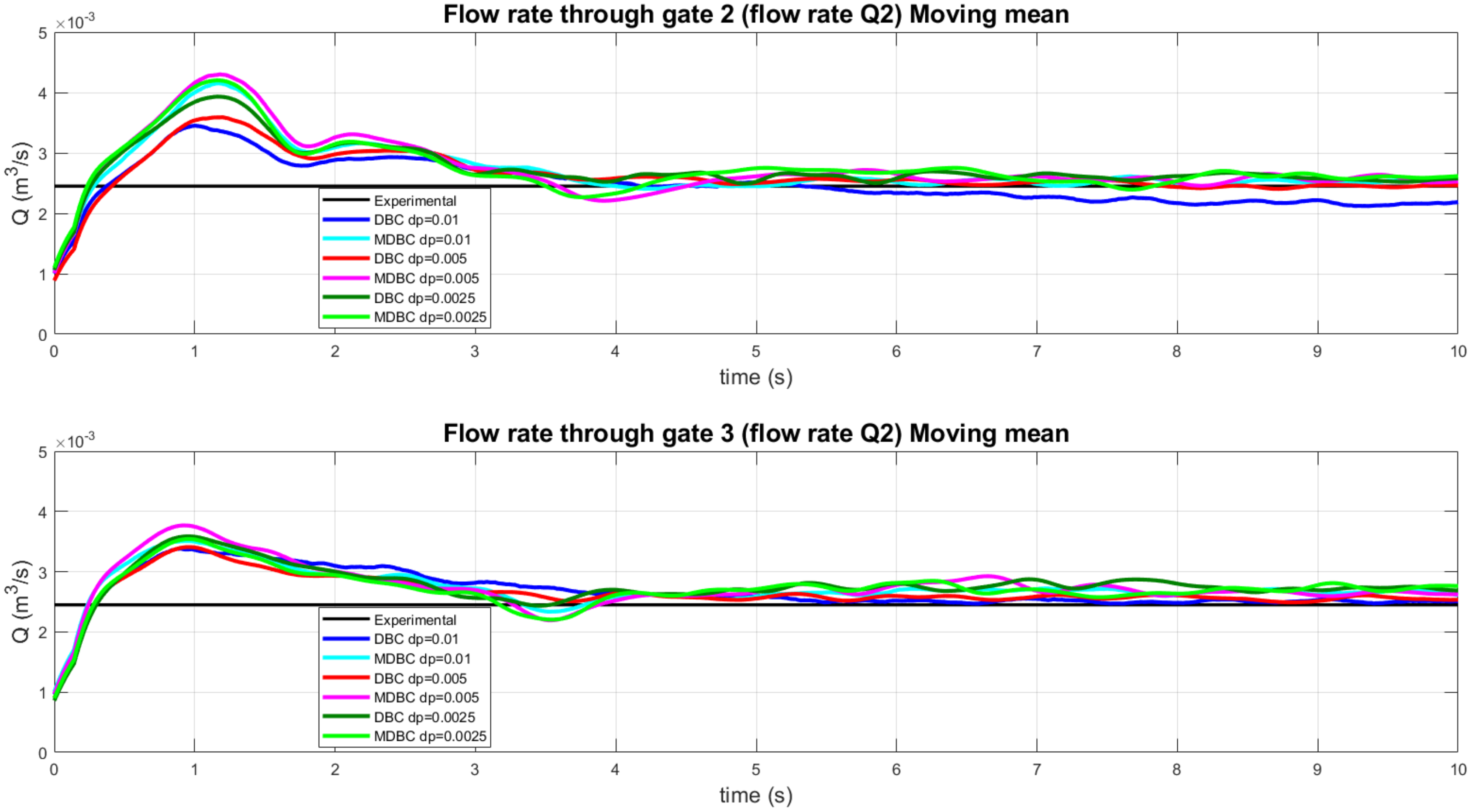


Figure 6.8: Plot showing water depths in pools 1 and 2 for flow rate Q2 for three particle spacings: $dp = 0.01m$ (blue); $dp = 0.005m$ (red), and $dp = 0.0025m$ (green). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.



Flow rate through gate 2 (flow rate Q2) Moving mean

Flow rate through gate 3 (flow rate Q2) Moving mean

Figure 6.9: Plot showing flow rates through gates 2 and 3 for flow rate Q2 for three particle spacings: $dp = 0.01m$ (blue); $dp = 0.005m$ (red), and $dp = 0.0025m$ (green). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

Weir only flows - For the weir only flows, denoted by Q3, Q4 and Q5 in Table 6.3, the gates in the walls are blocked and so the water can flow over the weirs only. An example of the flow showing particle velocity at the end of the 10 seconds is shown in Figure 6.10 for flow rate Q3 with a particle spacing of $dp = 0.005m$. The depths in pools 1 and 2 and the flows rates over weirs 2 and 3 were measured during the simulation and compared to the experimental result, theses results are shown in Figures 6.11 and 6.12 for flow rate Q3, all plots averaged using a moving mean over an interval of 15 values to smooth out the result.

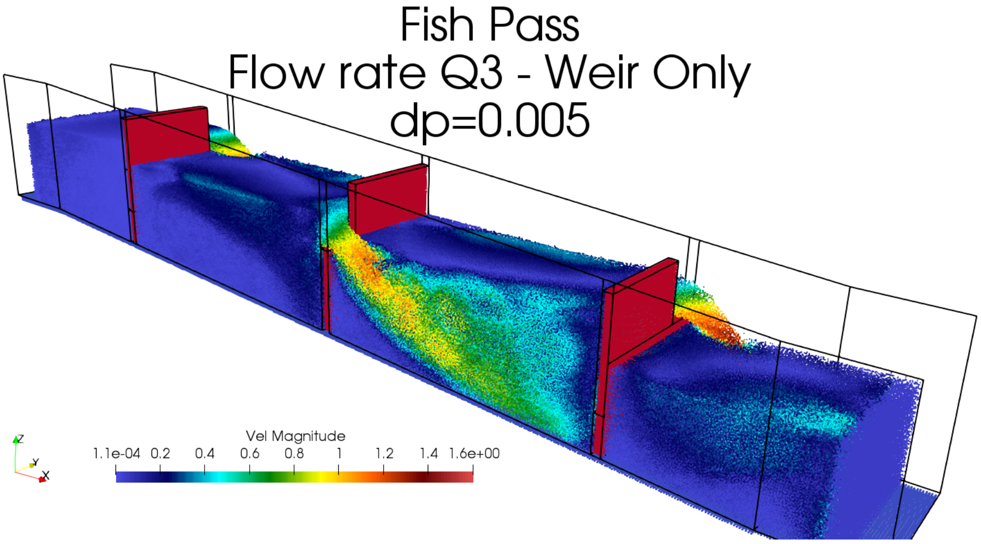


Figure 6.10: Fish pass flow over weirs only for flow rate Q3 and particle spacing $dp = 0.01m$, fluid particles coloured by velocity magnitude.

For flow rate Q3 two particle spacings were used. The water depths measured in pools 1 and 2 are plotted in Figure 6.11 for the two particle spacings and boundary conditions. The blue lines show results with $dp = 0.01m$ and red lines $dp = 0.005m$ with darker colours for DBC and lighter colours for MDBC. The water depths show good agreement within 1dp of the experimental measurement in both pools for both particle spacings. For each particle spacing the MDBC result shows a closer agreement to the experiment, and the coarser resolution shows a closer agreement than the finer resolution.

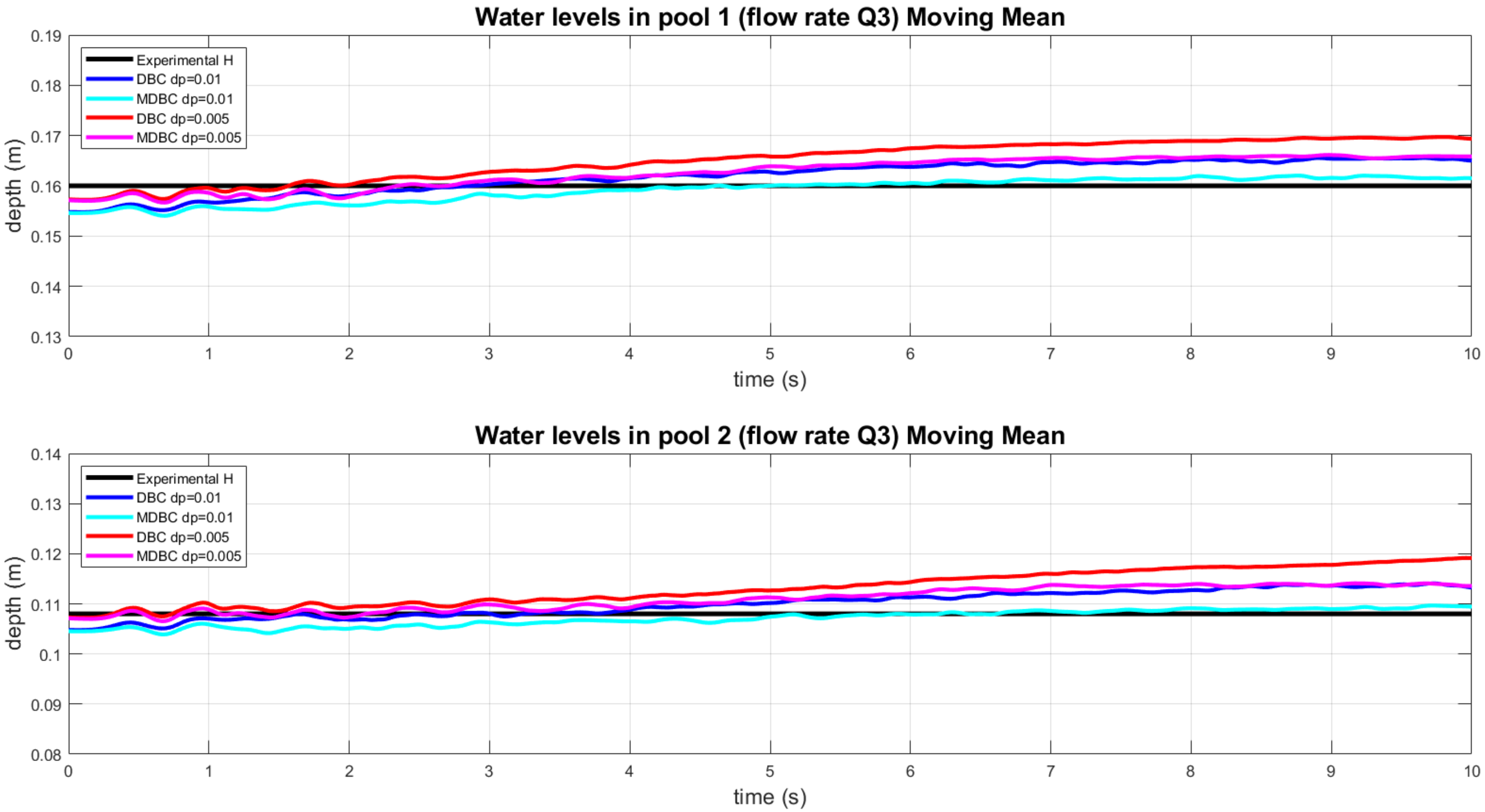


Figure 6.11: Plot showing water depths in pools 1 and 2 for flow rate Q3 for three particle spacings: $dp = 0.01m$ (blue), and $dp = 0.005m$ (red). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

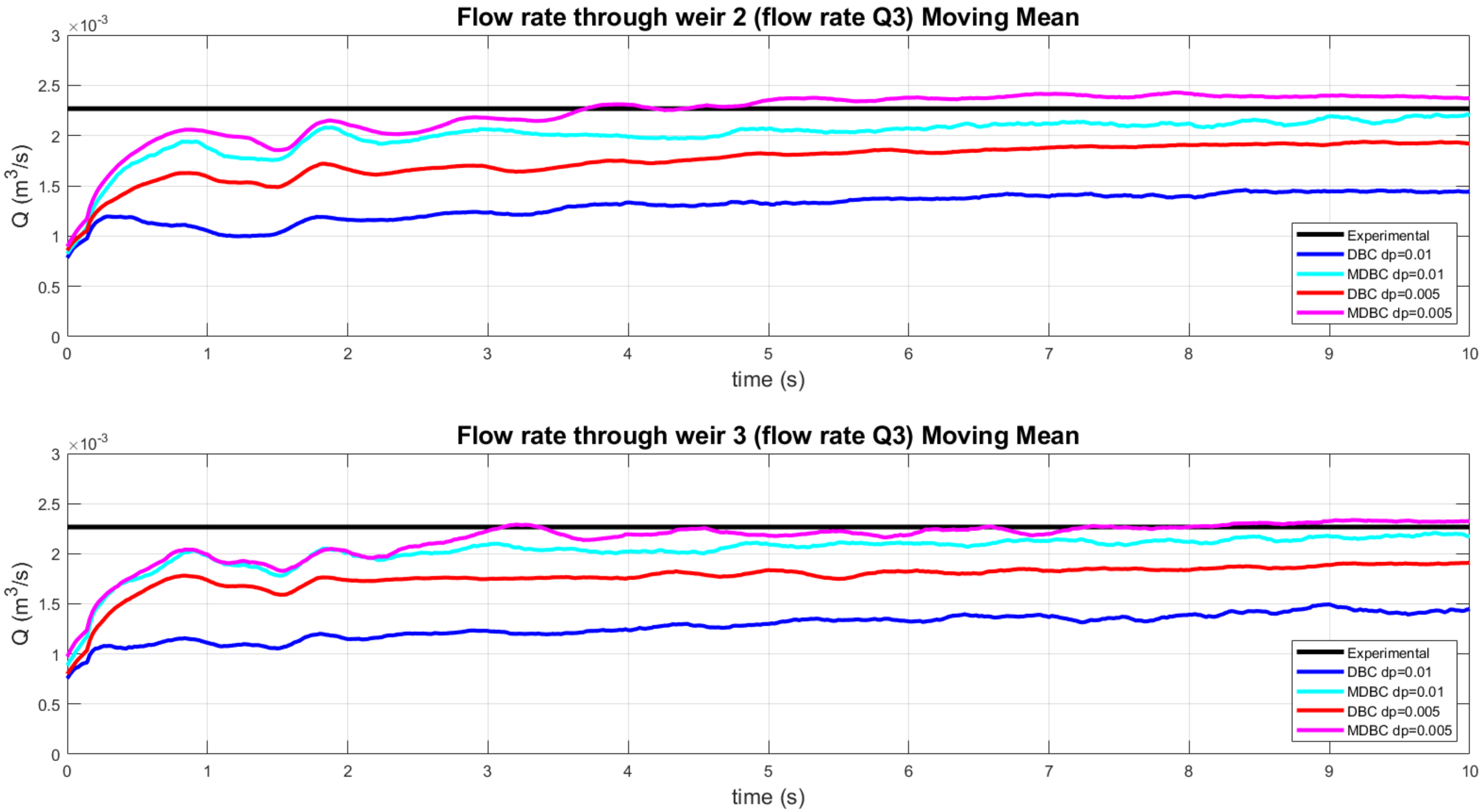


Figure 6.12: Plot showing flow rates over weirs 2 and 3 for flow rate Q3 for three particle spacings: $dp = 0.01m$ (blue), and $dp = 0.005m$ (red). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

Figure 6.12 shows the flow rates over weirs 2 and 3 for the two particle spacings and boundary conditions used. The result show good agreement at steady state when MDBC is used but worse agreement with DBC. A finer resolution also gives a better result. The reason for the larger under predictions when using DBC despite a good prediction for the water depth in the pools immediately upstream from the weirs is due to the gap that form close to the boundary when using DBC. This can be seen in Figure 6.13, which shows the fluid particles flowing over weir 2 at the end of the simulation and the initial particle arrangement for a particle spacing dp=0.01. The middle panel shows the initial particle arrangement in green with the boundary particle in black and the black lines show the boundary surface. Looking at the blue DBC particle after 10 seconds (left) a gap can be seen between the fluid and the top of the weir that is not present with MDBC (right). When using the DualSPHysics measure tool to measure the flow rate over the weir this gap causes issues since some of the points used for the measure tool will be located in this gap and giving a false lower flows rate reading. The plots produced for flow rates Q4 and Q5 are included in Appendix B.

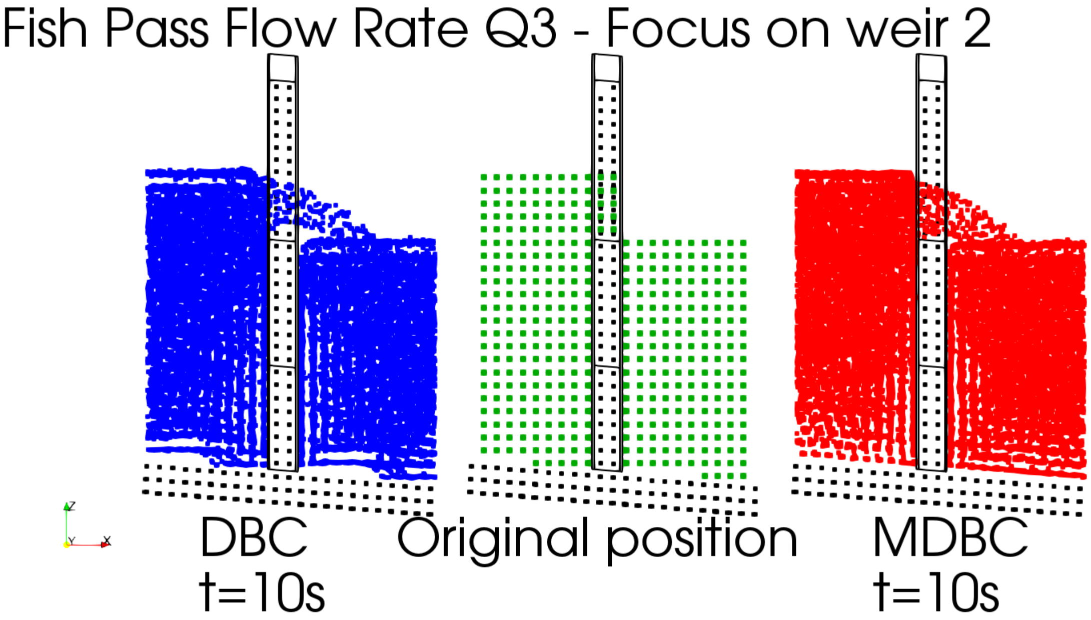


Figure 6.13: Fluid particles flowing over weir for $dp = 0.01m$. Initial particle arrangement (green, centre), DBC after 10 seconds (blue, left), MDBC after 10 seconds (red, right) and the boundary particle forming the wall and weir (black). The black lines show the real boundary surface.

PIV measurements - The final fish pass flow is flow rate Q6, for this case the water level is quite high and the water is able to flow through both the gates and the weirs. An example of the flow showing particle velocity at the end of the 10 seconds is shown in Figure 6.14 for flow rate Q2 with a particle spacing of $dp = 0.005m$. For this flow rate PIV measurements were taken of the flow velocity along 5 planes in pool 2, the locations of the PIV planes are shown in 6.15. The more interesting measurements are the readings taken in planes 1 and 2 that run the length of the pool, these show the velocity of the water as it travels over a weir and then down and through the following gate and vice versa. Figures 6.16 and 6.17 show the comparison between the PIV measurements and the SPH velocities in the same plane with $dp = 0.01m$.

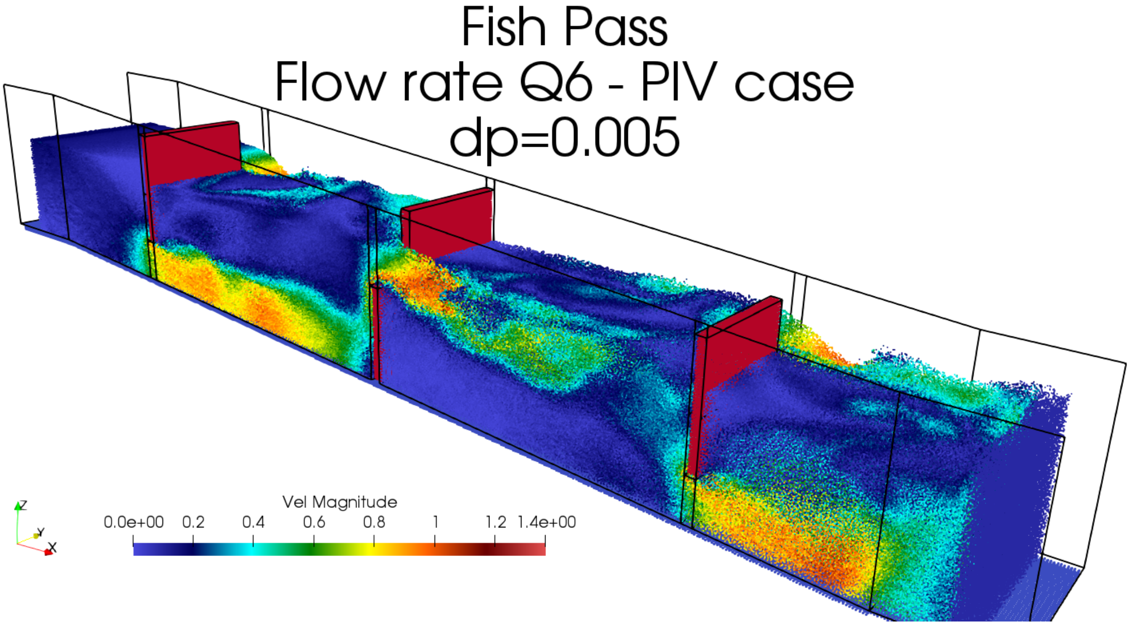


Figure 6.14: Fish pass flow through gates and weirs for flow rate Q6 and particle spacing $dp = 0.01m$, fluid particles coloured by velocity magnitude.

PIV plane 1 runs up the right hand side of pool 2 when viewed from above, the flow enters the pool through a gate and exits over a weir. The PIV plot in Figure 6.16 shows that as the fluid enters through the gate the flow spreads out slightly and follows the bottom of the pool before curving upwards close to the next wall and finally flowing over the weir. The thick red lines show more clearly the overall flow directions through the pool. Both the DBC and MDBC velocity fields show the fluid flowing through the gate and following the path of the bottom of the pool before curving upwards near the wall and over the weir. They do not however capture the spread near the gate. These flow paths are also seen in the velocity fields of the finer resolution fish pass.

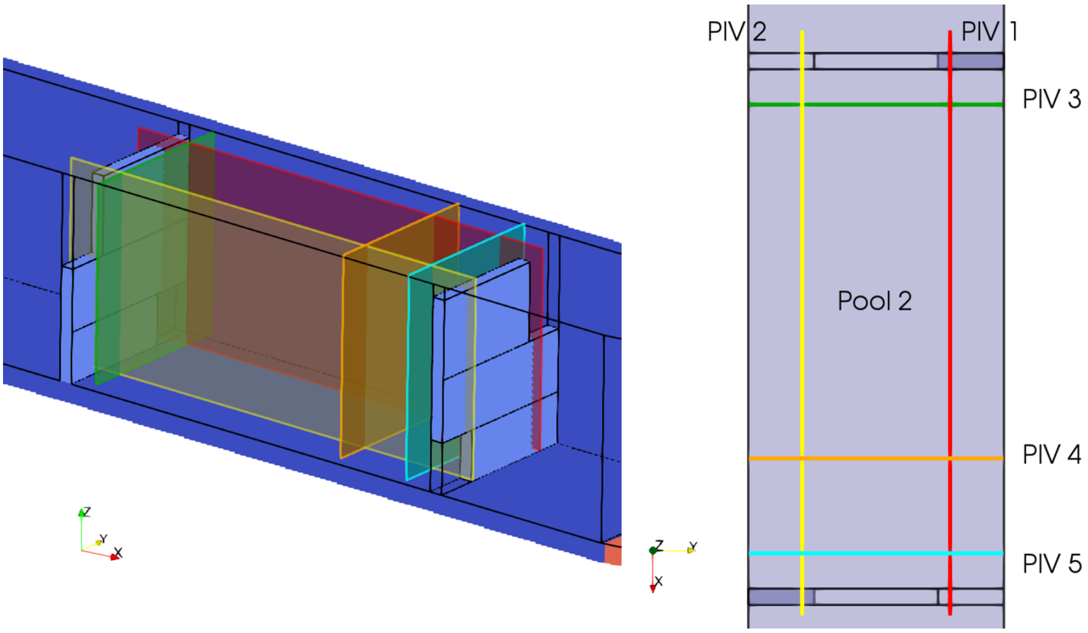


Figure 6.15: Planes showing where the PIV data measurements were taken in pool 2. Left shows the heights of the planes and their location in the pool, right shows a birds eye view of pool 2 and the PIV planes.

On the other side of the pool, PIV plane 2 shows the flow over the upstream weir downwards towards the bottom of the pool and out through the gate. Again thick red arrows have been added to Figure 6.17 to better show the overall flow direction. In this case the MDBC results follows the PIV path much closer with the flow travelling down from the weir towards the bottom of the pool before exiting through the down-stream gate. The DBC velocity field however spreads out more after the water flowing over the weir enter the pool. The flow exiting through the gate also comes from more directions around the gate as can be seen by the red arrows in the top of Figure 6.17. The closer agreement for MDBC is not seen when the resolution is increased however. In this case the flow, which can be seen in Figure 6.14, does not flow downwards towards the bottom of the pool, but instead staying close to the surface. The DBC result for this finer resolution also follows this pattern. The PIV plots for these planes are included in Appendix B.

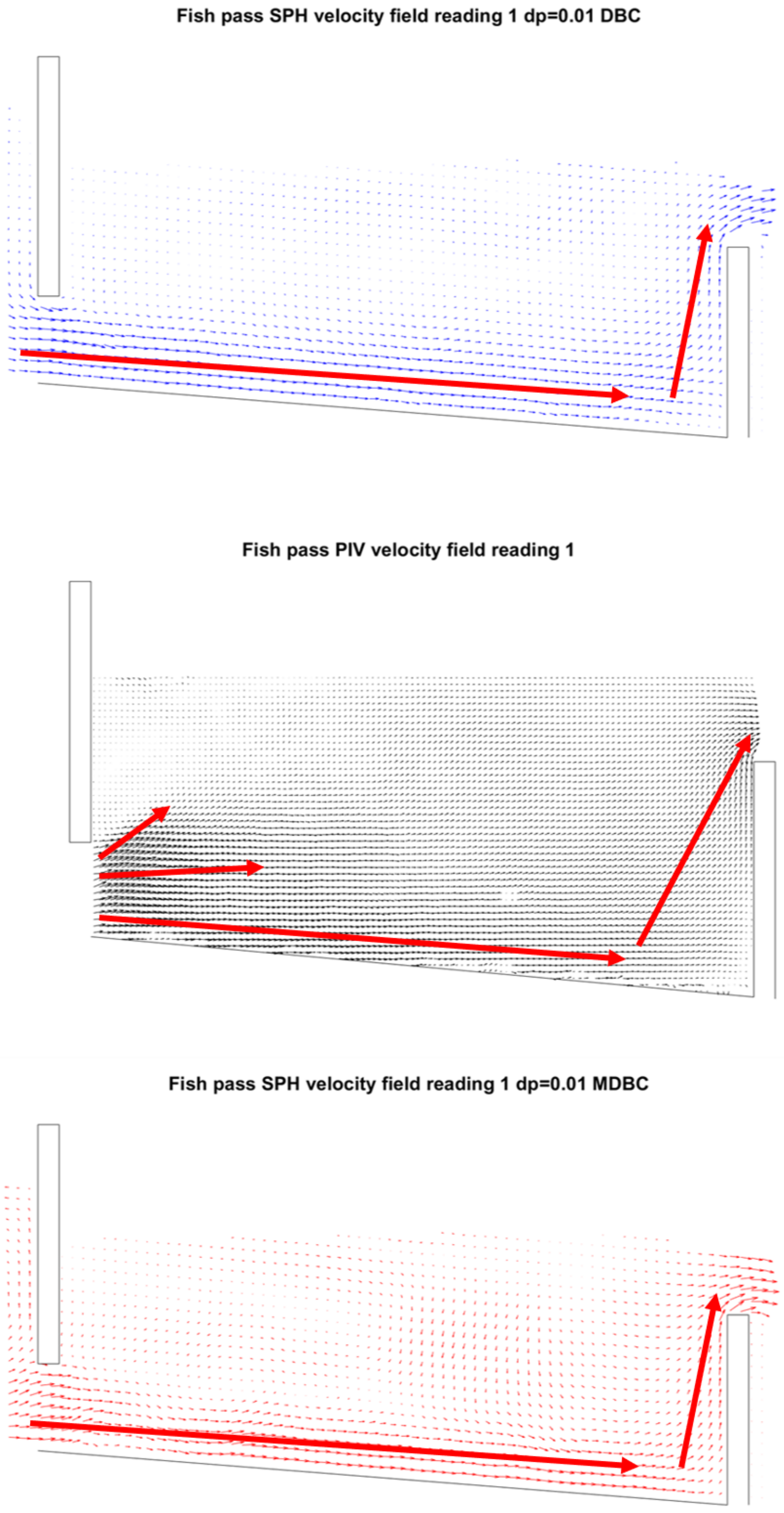


Figure 6.16: Comparison of velocity vectors in pool 2 plane 1 for DBC (top), PIV (middle) and MDBC (bottom) with particle spacing $dp = 0.01m$. The thick red lines show the overall directions of the flow

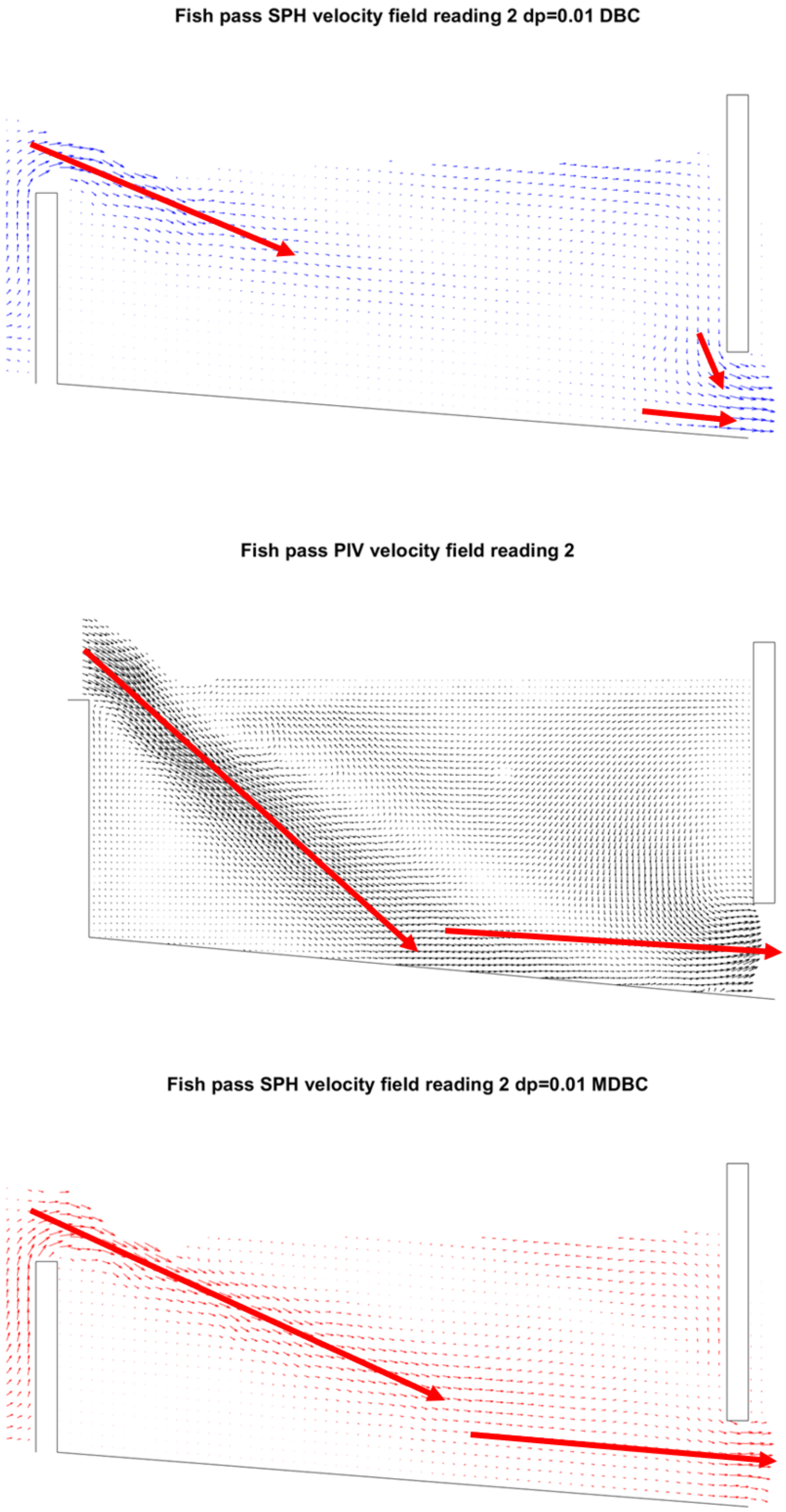


Figure 6.17: Comparison of velocity vectors in pool 2 plane 2 for DBC (top), PIV (middle) and MDBC (bottom) with particle spacing $dp = 0.01m$. The thick red lines show the overall directions of the flow

The Reynolds numbers for these fish pass flows are in the region $Re = \mathcal{O}(10000 - 25000)$ depending on the flow set up, with the set up for the PIV simulations having a Reynolds number $Re = 22471$. A Reynolds numbers of this magnitude indicate that the flow is turbulent. Since the SPH formulation used in this thesis does not include a turbulence model, much of the turbulent physics in the real flow will not be being captured in these simulations.

The MDBC approach used for these flows involved the density interpolation only. The velocities of the boundary particles were set to zero at all times, in the same way as for the original DBC. Including the velocity interpolation would give the boundary particles a negative velocity, with respect to the overall flow, slowing down the fluid close to the boundaries. For the gate and weir only cases it is expected that this would reduce the flow rates through the gates and weirs due the layers of slower moving fluid in these regions. For the PIV flow which includes both the gates and weirs, it is possible that the slower moving fluid going over the weir could help to better capture the flow plunging down into the pool that currently is not captured by the finer resolution simulations. It is possible that the speed of the layer of the fluid going over the weir close to the boundary is preventing the expected downward flow.

This test case shows that MDBC can be used for flows with complex shaped boundaries and work well with inlet outlet boundaries in DualSPHysics. Overall the results showed good agreement with the experimental data with MDBC giving better agreement for flows over the weirs due to the removal of the DBC boundary gap.

6.3 Pouring Flows

The final test case is pouring water from rotating cylinders, the aim being to create numerical models of the experiments presented in Chapter 5. To begin with the density only extrapolation MDBC method will be used to simulate water being poured from the measuring jug using DualSPHysics. Two particle spacings are used with the numerical parameters displayed in Table 6.5. For each resolution seven final pouring angles were simulated each rotating at the same speed for longer and longer times. The numerical jug is then held stationary for 1 second before rotating back towards its initial position. The pouring angles and rotation times modelled are shown in Table 6.6. Each simulation was run for 8 seconds of physical time, long enough for the jug to rotate to the desired angle, stop there and rotate back enough so that the fluid has stopped flowing out of the jug. Shifting is included for these test cases to aid the particle distribution and avoid clumping.

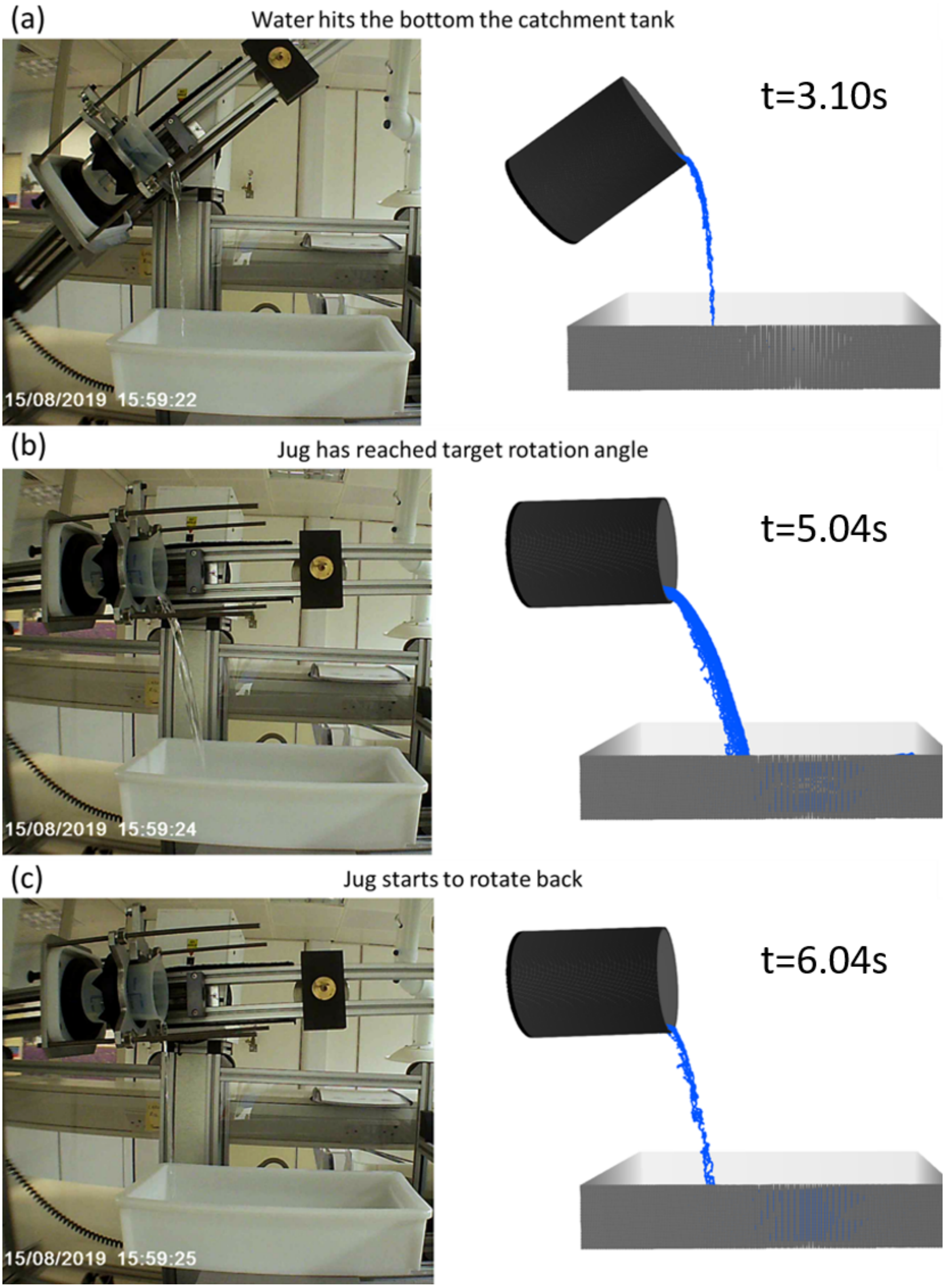
Table 6.5: Table of numerical parameter for Pouring flow simulations

dp (m)	h/dp	ν (Laminar)	δ (Fourtakas)	Shifting	CFL
0.0025	1.25	1×10^{-6}	0.1	ON	0.1
0.001	1.75	1×10^{-6}	0.1	ON	0.1

Table 6.6: Table of pouring angles and times and speeds of rotation

Angle (degrees)	Angular Speed (degrees/s)	Rotation time (s)	Hold time (s)
84	17.46	4.811	1
86	17.46	4.926	1
88	17.46	5.040	1
90	17.46	5.155	1
92	17.46	5.269	1
94	17.46	5.384	1
96	17.46	5.498	1

During the experiments videos of the pouring process were captured using a GoPro style camera and instants of the pouring process are compared to the SPH pour with dp=0.001 in Figure 6.18. The SPH shows a good visual agreement with the photos for the instants in time shown. The free surface of the SPH pour shows more break up than the smooth stream seen in the experiment, however there is no surface tension included in the model to keep free surface particle together. This could also be improved by using a third smaller particle spacing.



(a) Water hits the bottom the catchment tank

(b) Jug has reached target rotation angle

(c) Jug starts to rotate back

Figure 6.18: Comparison of experiment and SPH pouring at three points during the pouring process using particle spacing of $dp = 0.001m$. The final pouring angle was 88^o. Experiment pictures taken from a video captured using a GoPro style camera.

The jug was filled with enough particles for the total mass of fluid in the jug to be $500g$, the same mass of water used in the experiments. For $dp = 0.0025m$ this resulted in 32000 fluid particles and 500000 fluid particles for $dp = 0.001m$. At the end of the simulation the particles left inside the jug were counted and multiplied by mass of a fluid particle to find the total mass remaining.

The remaining mass of fluid is shown in Table 6.7 and plotted in Figure 6.19 comparing to the masses from the experiment. The two SPH result show good agreement with each other with less than $2g$ difference between the two particle spacings in the mass left in the jug at the end of the pour. The results however do not show agreement with the experimental result except by following a similar shaped curve, as can be seen in Figure 6.19.

For smaller pouring angles ($84^o - 88^o$) the jug has not rotated far enough to be held horizontal, so it is not expected that the jug will fully empty at these angles. However the SPH results greatly over predict the mass of the fluid left in the jug. As the angle of rotation increases the magnitude of the over prediction is reduced. For the larger angles, where the jug has rotated beyond the horizontal, the SPH still over predicts and never gets close to fully empty until the 94^o and 96^o pours. After checking the manufacturer of the jug, VWR (2020), it was found that the jug was made of a plastic called Polypropylene (PP). PP has been shown to be hydrophobic, for example Brown and Bhushan (2016), so it is possible that there is an amount of slip at the boundary surface. This however would not account for the over predictions seen for the smaller pouring angles, and the amount of slip would likely be very small. Another possibility could be that the angles in the experiment are wrong and the true pouring angles are 2^o larger than what was measured by the pouring rig used. Looking at the data plotted in Figure 6.19, the SPH points match reasonably well with the experimental data of the pouring angles 2^o smaller then prescribed. This is possible given the over rotation issue and angle selection

Table 6.7: Table of masses and number of particles remaining in the SPH jugs.

Angle (degrees)	Parts left (dp=0.0025)	Mass remaining (g)	Parts left (dp=0.001)	Mass remaining (g)
84	4554	71.16	69791	69.79
86	3265	51.02	49824	49.82
88	2108	32.94	31930	31.93
90	1111	17.36	16475	16.48
92	434	6.78	5201	5.20
94	155	2.42	1233	1.23
96	62	0.97	631	0.63

wheel issues that were experienced during the experiments. In order to test this, the experiment would need to be repeated and the final pouring angle checked in a second way not relying on the angle selection wheel.

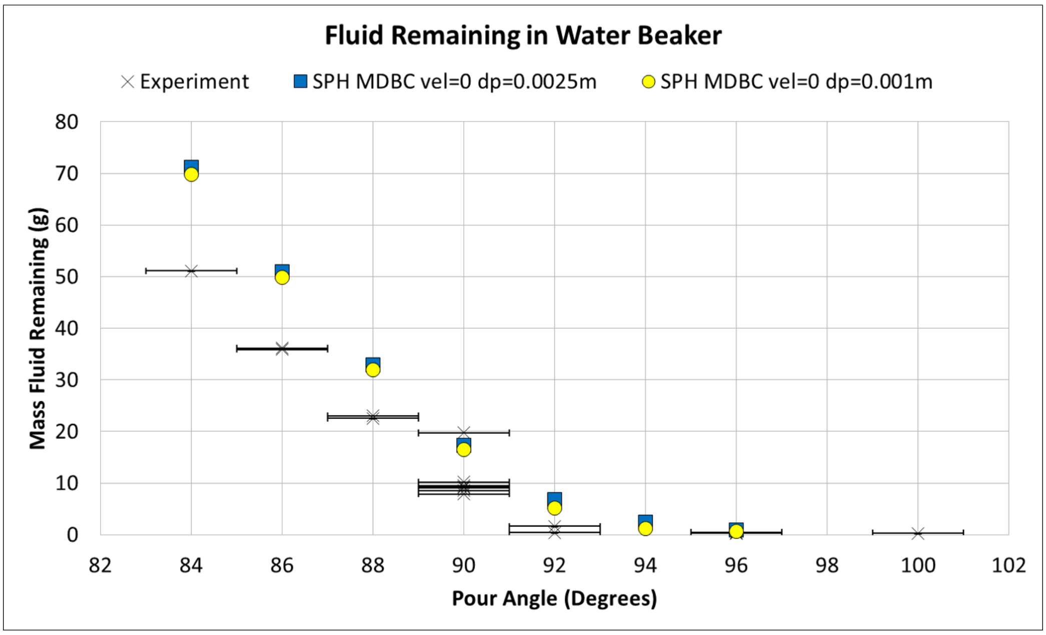


Figure 6.19: Comparison of SPH and experimental data for pouring angle versus mass of fluid remaining in the measuring jug.

Despite the discrepancy between the experimental measurements and the SPH result shown in Figure 6.19 the results show that the SPH method is capable of modelling the pouring process of Newtonian fluids. The snapshots of the SPH flow shown in Figure 6.19 look realistic, resembling the the flows in the experiments. The close agreement between the two particle spacing used is also encouraging. SPH is therefore capable of simulating these kinds of pouring flows, and suitable for their study.

Current Problems In order to test the possible partial slip idea for the jug, it would be useful to rerun the pouring simulations using the MDBC method with the velocity extrapolation. However there is an issue here of fluid particles leaking through the boundaries. This can be seen on the left of Figure 6.20 where some fluid particles can be seen to have leaked out of the boundaries of the jug as it is being rotated. For reference, the density only approach is shown on the right for the same flow. The current view is that by changing the boundary velocity, energy has been introduced into the divergence of velocity used for the density gradient calculations. Similar issues have previously been discussed by De Leffe et al. (2011), where a solution was proposed involving not using the no-slip velocity in the conservation of mass to avoid this energy introduction. The approach to De

Leffe et al. has been attempted in an effort to resolve the issue with some success for sloshing tanks, however the issue has still not been remedied for these pouring type flows.

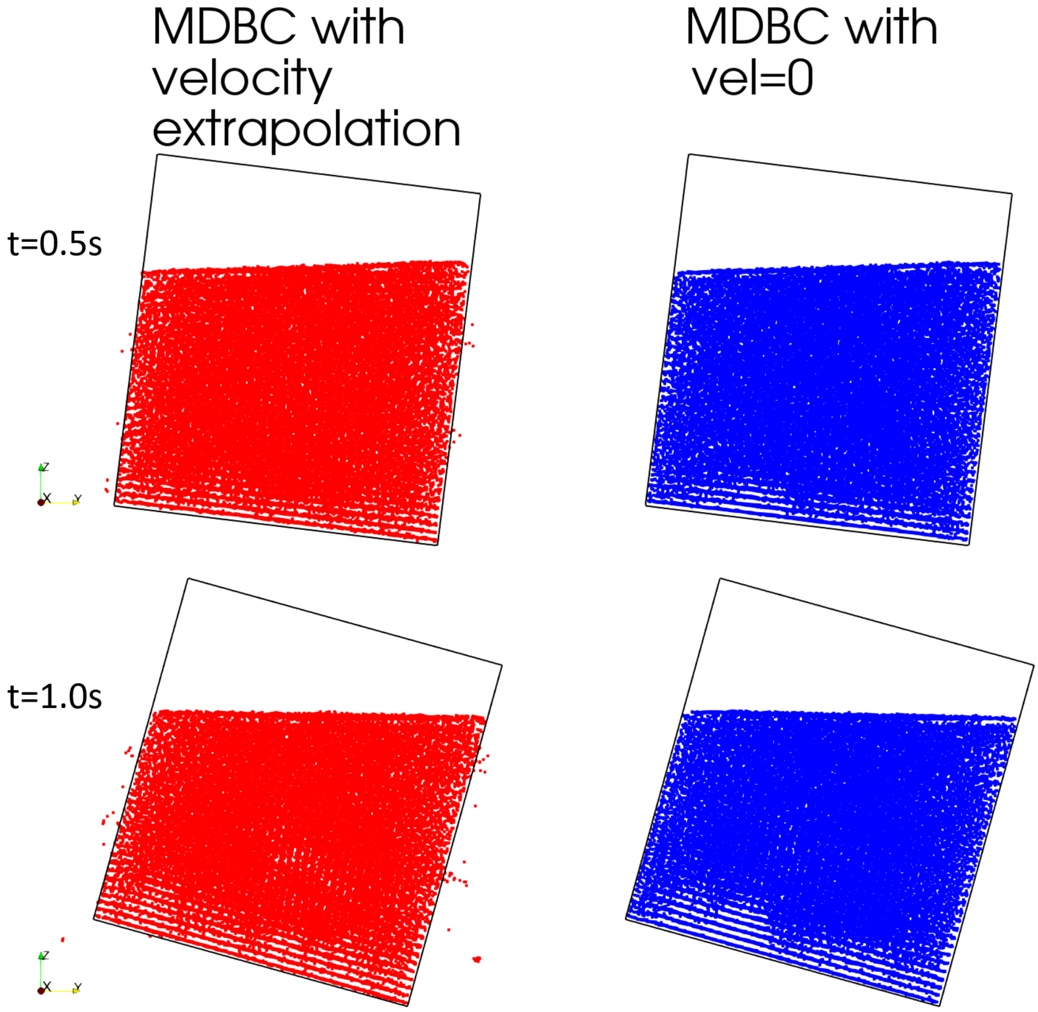


Figure 6.20: Instants of the pouring simulation using MDBC with velocity extrapolation (left) and density extrapolation only (right) showing the issues of fluid particle leaking through the boundaries when the velocity extrapolation is used.

This means that the partial slip boundary condition can not yet be used for this test case as it requires the use of the MDBC velocity extrapolation. Also flows of Carbopol out of the jug, that would have involved non-Newtonian models, could not be simulated due to the high viscosity at low shear and the strict non-Newtonian viscous time step resulting in extremely long run times. The results however do show that SPH with MDBC based on density extrapolation only are robust and can be used to qualitatively model pouring types flows. If the full velocity extrapolation had worked for these flows, it is expected that the velocity of the fluid particles close to the boundaries would be reduced. It would be possible for more particles to stay stuck to the boundary as the no-slip condition would slow them down. The number of particles left in the vessels at the end of the pouring sequence could be higher than presented for the density only extrapolation. Including the partial slip boundaries would be expected to reduce the number of particles left inside as the fluid close to the boundary would be able to move with the slip velocity.

6.4 Summary

In this chapter three complex test cases have been simulated using MDBC. The 3D dam break and fish pass test cases show that MDBC can be used to model free surface flows involving complex shaped geometries with good levels of accuracy. Using MDBC the gap near boundaries is reduced allowing impact pressures of the dam break hitting an obstacle can be measured at the true probe location on the boundary, rather than projecting the pressure probe into the fluid. The reduction in the gap also allows for more accurate simulations of flows over obstacles such as the weirs in the fish pass.

The final test case showed, despite current problems with velocity extrapolation, that SPH with MDBC can be used to qualitatively model pouring type flows of interest to Unilever.

Chapter 7

Conclusions & Future Work

In this chapter the conclusions of this thesis are presented and some recommendations for futures work suggested.

7.1 Conclusions

Boundary conditions at solid surfaces have been a major limitation in SPH, and as such are regarded as one the grand challenges of the method. Accurate and high order formulations are available but are difficult to apply for complex shapes which is an intended strength of SPH. The robust boundary condition for general complex shapes, known as the dynamic boundary condition, has underlying physical and theoretical weaknesses, limiting accuracy. A new method has been developed based on the existing dynamic boundary condition philosophy with an improved density extrapolation technique combining the ghost node mirroring of Marrone et al. (2011) and the extrapolation method of Tafuni et al. (2018) to maintain the robustness of the dynamic boundary condition while removing physical and theoretical limitations; this has become known as the modified dynamic boundary condition. This has been implemented in the widely used open source code DualSPHysics with small additional computational cost.

Traditional boundary conditions are the no slip or free slip boundary conditions, but many physical problems exhibit partial slip an intermediate between the two with a small flow velocity on the solid surface. Formulations in SPH have been developed for the first time which has been possible with the modified dynamic boundary condition. This has been progressed with application to parallel flows with analytical solutions, Couette and Poiseuille flows, for which new analytical solutions have been developed to enable validation. The important rheometer case, used for quantifying particularly non-Newtonian fluid properties, has also been studied, comparing with limited available experimental data. The groundwork for generalisation to complex shapes, including no-slip and partial slip, has been laid.

The key aim of a versatile simulation tool with free surfaces should be demonstrated. This has been achieved for diverse applications, the standard dam break with violent impact on a column and the new fish pass problem comprising a complex channel flow with transverse weirs and gates. New experimental data for pouring from a beaker has been simulated.

The modified dynamic boundary condition with density extrapolation and zero velocity solid boundary particles (or with velocity of the body if moving) is now available in the open source DualSPHysics code which can run efficiently on both CPUs and GPUs.

7.2 Future Work

Currently the MDBC method with density extrapolation works for a wide range of test cases, and the velocity extrapolation is working for parallel flows. The partial slip boundary condition implemented using MDBC also works for parallel flows, including an example of a power law Poiseuille flow. Future avenues of study following on from this PhD can be summarised as:

1. Correct the no-slip velocity extrapolation of MDBC for non parallel flows

2. Generalise this condition to partial slip flows

3. Include general non-Newtonian formulations with partial slip boundaries

4. Generalise the above to work with two-phase flows

5. Generalise for use with deformable boundary

Appendix A

Poiseuille Flow Transient Solution with Partial Slip

Consider a Poiseuille flow in a channel of length L with walls at $y = l$ and $y = -l$. The Navier-Stokes equations for this flow can be reduced to the PDE

$$\frac{\partial u}{\partial t} = -\frac{1}{\rho}\frac{dp}{dx} + \nu\frac{\partial^2 u}{\partial y^2} + F_x \tag{A.1}$$

subject to the initial and boundary conditions

$$u\,(\pm\, l, t) = u_{slip}(\pm l, t) \tag{A.2}$$
$$u(y, 0) = 0 \tag{A.3}$$

using the partial slip boundary Lauga et al. (2005),

$$\mathbf{u}_{slip} = b\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)\cdot\mathbf{n} \tag{A.4}$$

where $\mathbf{n}$ is the normal facing into the liquid and b is the slip length at the walls.

As $t \to \infty$ the Poiseuille flow tends to the steady state, which is the solution of the PDE

$$\frac{\partial^2 u}{\partial y^2} = \frac{1}{\mu}\left(\frac{dp}{dx} - \rho F_x\right) = \frac{1}{\mu}G(x) \tag{A.5}$$

subject to boundary condition (A.2).

Integrating with respect to y the velocity and velocity gradient are found as

$$\frac{\partial u}{\partial y} = \frac{1}{\mu}G(x)y + A \tag{A.6}$$
$$u = \frac{1}{2\mu}G(x)y^2 + Ay + B. \tag{A.7}$$

The boundary condition (A.2) is applies in order to find the values of the constants

A and B

$$u(l) = \frac{1}{2\mu}G(x)l^2 + Al + B = u_{slip}(l) \tag{A.8}$$

$$u(-l) = \frac{1}{2\mu}G(x)l^2 - Al + B = u_{slip}(-l). \tag{A.9}$$

Adding these together gives

$$u_{slip}(l) + u_{slip}(-l) = \frac{1}{\mu}G(x)l^2 + 2B \tag{A.10}$$

which yields

$$B = -\frac{1}{2\mu}G(x)l^2 + \frac{u_{slip}(l) + u_{slip}(-l)}{2}. \tag{A.11}$$

Subtracting (A.9) from (A.8) and noting that $l \neq 0$ it can be seen that

$$A = 0. \tag{A.12}$$

Hence the velocity profile takes the form

$$u(y) = -\frac{1}{2\mu}G(x)\left(l^2 - y^2\right) + \frac{u_{slip}(l) + u_{slip}(-l)}{2}. \tag{A.13}$$

Using the partial slip velocity given by (A.4) and evaluating the velocity gradient terms, $\mathbf{u}_{slip}$ is given by

$$\mathbf{u}_{slip} = b\begin{bmatrix} 2\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & 2\frac{\partial u}{\partial x} \end{bmatrix} \cdot \mathbf{n}. \tag{A.14}$$

For the upper boundary $\mathbf{n} = \begin{bmatrix} 0 & -1 \end{bmatrix}^T$, and so

$$\mathbf{u}_{slip} = \begin{bmatrix} -b\frac{\partial u}{\partial y} \\ 0 \end{bmatrix} \tag{A.15}$$

thus the velocity in the x direction is then given by

$$u_{slip}(l) = -b\frac{\partial u}{\partial y} = -b\frac{1}{\mu}G(x)l. \tag{A.16}$$

Similarly, for the lower boundary$\mathbf{n} = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$ and so

$$\mathbf{u}_{slip} = \begin{bmatrix} b\frac{\partial u}{\partial y} \\ 0 \end{bmatrix} \tag{A.17}$$

and the velocity in the x direction is then given by

$$u_{slip}(-l) = b\frac{\partial u}{\partial y} = b\frac{1}{\mu}G(x)(-l). \tag{A.18}$$

Hence the value of the slip velocity on both boundaries is equal to

$$u_{slip}(\pm l) = u_{slip} = -bl\frac{1}{\mu}G(x). \tag{A.19}$$

Substituting this into (A.13) the steady state equation is found as

$$u(y) = -\frac{1}{2\mu}G(x)\left(l^2 + 2bl - y^2\right) \tag{A.20}$$

or again using the quantity u_{slip}

$$u(y) = -\frac{1}{2\mu}G(x)\left(l^2 - y^2\right) + u_{slip} \tag{A.21}$$

which can also be written as

$$\frac{u(y)}{-\left(1/2\mu\right)G(x)l^2} = 1 - \frac{y^2}{l^2} + \frac{u_{slip}}{-\left(1/2\mu\right)G(x)l^2}. \tag{A.22}$$

The unsteady form of the equation can be written in the form

$$\frac{u(y,t)}{-\left(1/2\mu\right)G(x)l^2} = 1 - \frac{y^2}{l^2} + \frac{u_{slip}(t)}{-\left(1/2\mu\right)G(x)l^2} - f(y,t) \tag{A.23}$$

where $f(y,t)$ is a function satisfying the equation

$$\frac{\partial f}{\partial t} = \nu\frac{\partial^2 u}{\partial y^2} \tag{A.24}$$

subject to the initial and boundary conditions

$$f\left(\pm\, l, t\right) = 0 \tag{A.25}$$

$$f(y,0) = 1 - \frac{y^2}{l^2} \tag{A.26}$$

In order to use the separation of variable method the substitution

$$f(y,t) = Y(y)T(t) \tag{A.27}$$

is used in equation (A.24) to get

$$Y(y)T'(t) = \nu Y''(y)T(t) \tag{A.28}$$

which rearranges to give

$$\frac{T'(t)}{\nu T(t)} = \frac{Y''(y)}{Y(y)}. \tag{A.29}$$

Since the left hand side of equation (A.29) is a function in t only and the right hand side is a function in y only, they can only be equal is they are also equal to a constant function, $-\lambda^2$, say. The equation is then written as

$$\frac{T'(t)}{\nu T(t)} = -\lambda^2 = \frac{Y''(y)}{Y(y)} \tag{A.30}$$

which leads to the two equations

$$T'(t) + \lambda^2 \nu T(t) = 0 \tag{A.31}$$
$$Y''(y) + \lambda^2 Y(y) = 0. \tag{A.32}$$

From (A.31) it is seen that $T(t)$ has the form

$$T(t) = \alpha e^{-\lambda^2 \nu t} \tag{A.33}$$

and from (A.32) that $Y(y)$ has the form

$$Y(y) = \beta \sin(\lambda y) + \gamma \cos(\lambda y). \tag{A.34}$$

From the boundary condition, (A.25), on f we see that

$$f(\pm l, t) = Y(\pm l)T(t) = 0 \tag{A.35}$$

$T(t) \neq 0$ therefore

$$Y(l) = \beta \sin(\lambda l) + \gamma \cos(\lambda l) = 0 \tag{A.36}$$
$$\begin{aligned} Y(-l) &= \beta \sin(-\lambda l) + \gamma \cos(-\lambda l) \\ &= -\beta \sin(\lambda l) + \gamma \cos(\lambda l) = 0. \end{aligned} \tag{A.37}$$

Subtracting (A.36) from (A.37) gives

$$2\beta \sin(\lambda l) = 0 \tag{A.38}$$

and so $\beta = 0$, which means

$$\gamma \cos(\lambda l) = 0 \tag{A.39}$$

Since a non zero solution is sought

$$\cos(\lambda l) = 0 \tag{A.40}$$

which means that

$$\lambda_n = \frac{(2n+1)\pi}{2l} \qquad n = 0, 1, 2, \ldots\infty. \tag{A.41}$$

Combining this with (A.27), (A.33) and (A.34) it is seen that

$$f(y,t) = \sum_{n=0}^{\infty} \alpha_n e^{-\lambda_n^2 \nu t} \cos(\lambda_n y) \tag{A.42}$$

where α_n is the combination of the α in (A.33) and the γ in (A.39) and the λ_n is given by (A.41).

The values of the α_n can be found using the second condition on $f(y,t)$, that is

$$f(y,0) = \sum_{n=0}^{\infty} \alpha_n e^0 \cos(\lambda_n y) = 1 - \frac{y^2}{l^2}. \tag{A.43}$$

In order to take advantage of the orthogonality properties of the cos function, both sides are multiplied by $\cos(\lambda_n l)$ and integrated with respect to y across the depth of the channel to find

$$\sum_{n=0}^{\infty} \alpha_n \int_{-l}^{l} \cos^2(\lambda_n y) dy = \int_{-l}^{l} \left(1 - \frac{y^2}{l^2}\right) \cos(\lambda_n y) dy. \tag{A.44}$$

The orthogonality of cos gives the expression

$$\alpha_n = \frac{1}{l} \int_{-l}^{l} \left(1 - \frac{y^2}{l^2}\right) \cos(\lambda_n y) dy \tag{A.45}$$

All of the other terms in the summation become 0 due to the other orthogonality property of cos.

In order to get the value of the α_n integration by parts is used twice, first by setting the reduction formulae as

$$v = 1 - \frac{y^2}{l^2} \qquad w' = \cos(\lambda_n y) \tag{A.46}$$

$$v' = -\frac{2y}{l^2} \qquad w = \frac{1}{\lambda_n} \sin(\lambda_n y) \tag{A.47}$$

which transforms (A.45) into

$$\alpha_n = \frac{1}{l} \left\{ \left[\left(1 - \frac{y^2}{l^2}\right) \frac{1}{\lambda_n} \sin(\lambda_n y) \right]_{-l}^{l} + \int_{-l}^{l} \frac{2y}{l^2 \lambda_n} \sin(\lambda_n y) dy \right\}. \tag{A.48}$$

For the second integration the reduction formulae are

$$v = \frac{2y}{l^2} \qquad w' = \frac{1}{\lambda_n} \sin(\lambda_n y) \tag{A.49}$$

$$v' = \frac{2}{l^2} \qquad w = -\frac{1}{\lambda_n^2} \cos(\lambda_n y) \tag{A.50}$$

(A.48) then gives

$$\alpha_n = \frac{1}{l} \left\{ \left[\left(1 - \frac{y^2}{l^2}\right) \frac{1}{\lambda_n} \sin(\lambda_n y) \right]_{-l}^{l} + \left[\left(-\frac{2y}{l^2 \lambda_n^2}\right) \cos(\lambda_n y) \right]_{-l}^{l} + \int_{-l}^{l} \frac{2}{l^2 \lambda_n^2} \cos(\lambda_n y) dy \right\}. \tag{A.51}$$

Evaluating the final integral gives

$$\alpha_n = \frac{1}{l}\left\{ \left[\left(1-\frac{y^2}{l^2}\right)\frac{1}{\lambda_n}\sin(\lambda_n y)\right]_{-l}^{l} + \left[\left(-\frac{2y}{l^2\lambda_n^2}\right)\cos(\lambda_n y)\right]_{-l}^{l} + \left[\left(\frac{2}{l^2\lambda_n^3}\right)\cos(\lambda_n y)\right]_{-l}^{l} \right\}. \tag{A.52}$$

This then reduces down to

$$\alpha_n = \frac{32(-1)^n}{(2n+1)^3\pi^3}. \tag{A.53}$$

Substituting this back into the unsteady state equation (A.23) and using the summation form for $f(y,t)$ we get

$$\frac{u(y,t)}{-(1/2\mu)\,G(x)l^2} = 1 - \frac{y^2}{l^2} - \sum_{n=0}^{\infty}\left(\frac{32(-1)^n}{(2n+1)^3\pi^3}\right)e^{-\lambda_n^2\nu t}\cos(\lambda_n y) + \frac{u_{slip}(t)}{-(1/2\mu)\,G(x)l^2} \tag{A.54}$$

which rearranges to give

$$u(y,t) = -\frac{1}{2\mu}G(x)\left(l^2-y^2\right) + \sum_{n=0}^{\infty}G(x)\left(\frac{16(-1)^n l^2}{(2n+1)^3\pi^3\mu}\right)e^{-\lambda_n^2\nu t}\cos(\lambda_n y) + u_{slip}(t) \tag{A.55}$$

with

$$\lambda_n = \frac{(2n+1)\pi}{2l} \qquad n = 0, 1, 2, ...\infty \tag{A.56}$$

All that is left to do know is find the time-dependent partial slip velocity. Previously it was shown that

$$u_{slip}(l,t) = u_{slip}(-l,t) = -b\frac{\partial u}{\partial y}(l,t). \tag{A.57}$$

First finding the velocity gradient of Equation (A.55) it is seen that

$$\frac{\partial u}{\partial y}(y,t) = \frac{G(x)y}{\mu} - \sum_{n=0}^{\infty}G(x)\frac{(2n+1)\pi}{2l}\left(\frac{16(-1)^n l^2}{(2n+1)^3\pi^3\mu}\right)e^{-\lambda_n^2\nu t}\sin(\lambda_n y) \tag{A.58}$$

$$= \frac{G(x)y}{\mu} - \sum_{n=0}^{\infty}G(x)\left(\frac{8(-1)^n l}{(2n+1)^2\pi^2\mu}\right)e^{-\lambda_n^2\nu t}\sin(\lambda_n y) \tag{A.59}$$

Then substituting $y = \pm l$ the partial slip velocity is given by

$$u_{slip}(t) = -\frac{G(x)bl}{\mu} + \sum_{n=0}^{\infty}G(x)\left(\frac{8bl}{(2n+1)^2\pi^2\mu}\right)e^{-\lambda_n^2\nu t} \tag{A.60}$$

And so finally the equation for the unsteady Poiseuille flow with partial slip can be written as

$$\begin{aligned} u(y,t) = &-\frac{1}{2\mu}G(x)\left(l^2+2bl-y^2\right) \\ &+\sum_{n=0}^{\infty} G(x)\left(\frac{16(-1)^n l^2}{(2n+1)^3\pi^3\mu}\right)e^{-\lambda_n^2\nu t}\cos(\lambda_n y) \\ &+\sum_{n=0}^{\infty} G(x)\left(\frac{8bl}{(2n+1)^2\pi^2\mu}\right)e^{-\lambda_n^2\nu t} \end{aligned} \tag{A.61}$$

Note that if the standard no-slip condition was imposed on the boundary then simply setting $b=0$ would recover the unsteady plane Poiseuille flow equation

$$u(y,t) = -\frac{1}{2\mu}G(x)\left(l^2-y^2\right)+\sum_{n=0}^{\infty} G(x)\left(\frac{16(-1)^n l^2}{(2n+1)^3\pi^3\mu}\right)e^{-\lambda_n^2\nu t}\cos(\lambda_n y) \tag{A.62}$$

Appendix B

Further Fish Pass Results

In this Appendix further results from the fish pass test case are presented.

Figure B.1 shows the depths in pools 1 and 2 for flow rate Q1 for two particle spacings. dp=0.01m are shown in blue and dp=0.005m in red, the darker shade represent DBC results and the lighter shades MDBC.

Figure B.2 shows the flow rates through gates 2 and 3 for flow rate Q1 for two particle spacings. dp=0.01m are shown in blue and dp=0.005m in red, the darker shade represent DBC results and the lighter shades MDBC.

Figure B.3 shows the depths in pools 1 and 2 for flow rate Q4 for two particle spacings. dp=0.01m are shown in blue and dp=0.005m in red, the darker shade represent DBC results and the lighter shades MDBC.

Figure B.4 shows the flow rates through gates 2 and 3 for flow rate Q4 for two particle spacings. dp=0.01m are shown in blue and dp=0.005m in red, the darker shade represent DBC results and the lighter shades MDBC.

Figure B.5 shows the depths in pools 1 and 2 for flow rate Q5 for two particle spacings. dp=0.01m are shown in blue and dp=0.005m in red, the darker shade represent DBC results and the lighter shades MDBC.

Figure B.6 shows the flow rates through gates 2 and 3 for flow rate Q5 for two particle spacings. dp=0.01m are shown in blue and dp=0.005m in red, the darker shade represent DBC results and the lighter shades MDBC.

Figures B.7 and B.8 show the comparison of the SPH result to PIV velocity data measured in planes 1 and 2 respectively. The top blue plots show the DBC result, bottom red shows MDBC and the middle black shows the PIV measurement.

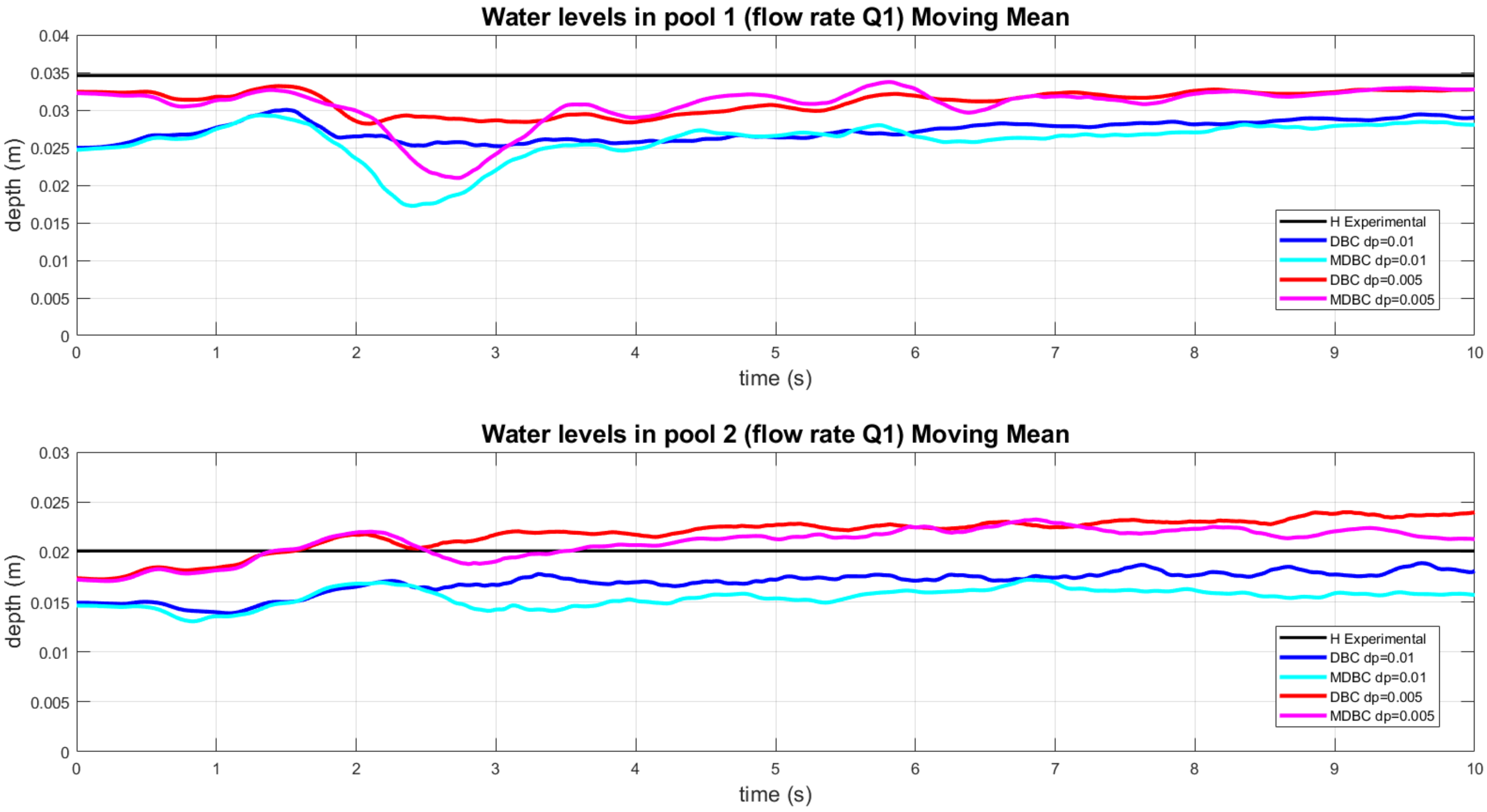


Figure B.1: Plot showing water depths in pools 1 and 2 for flow rate Q1 for two particle spacings: dp=0.01m (blue), and dp=0.005m (red). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

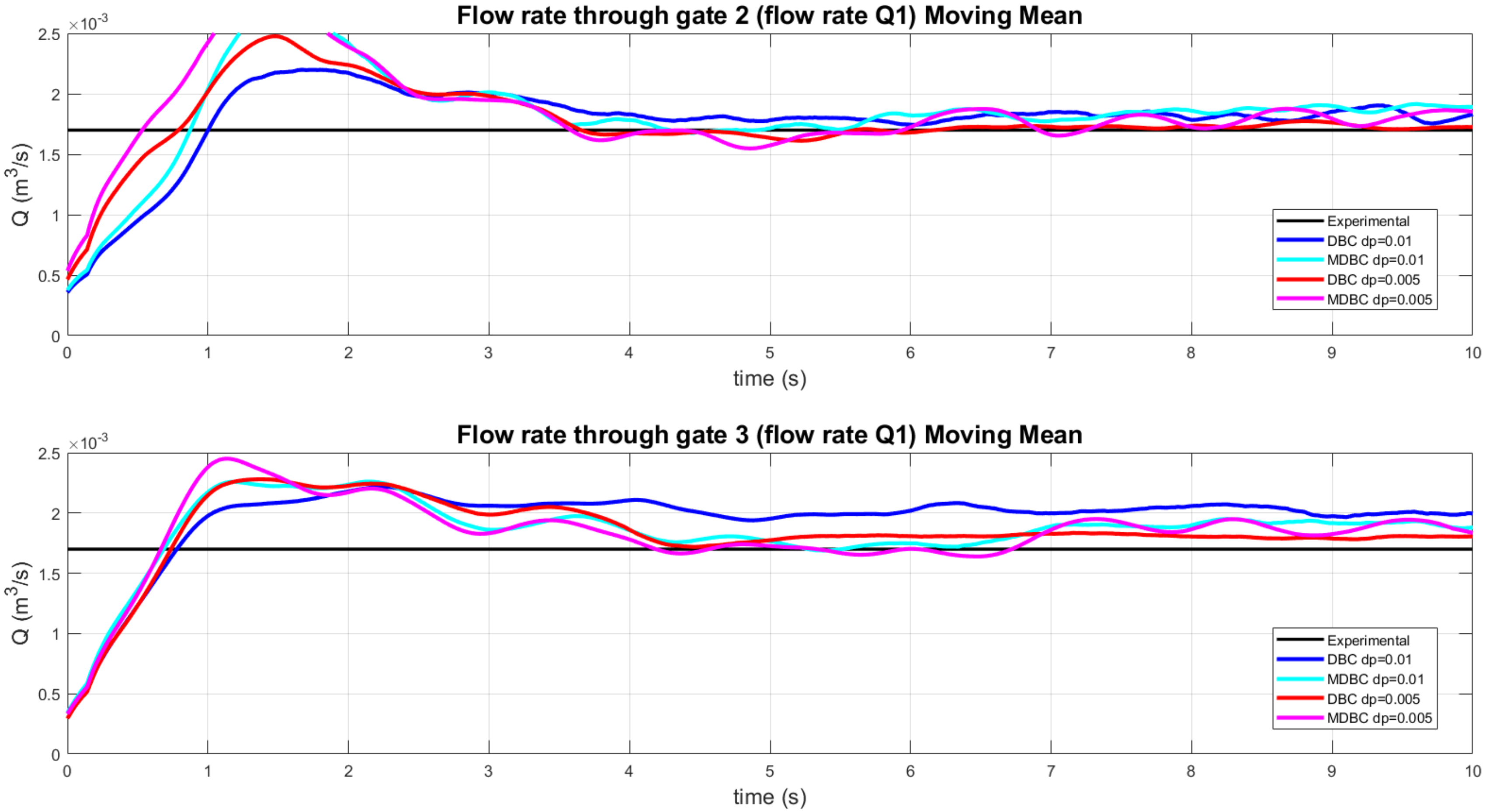


Figure B.2: Plot showing flow rates through gates 2 and 3 for flow rate Q1 for two particle spacings: dp=0.01m (blue)and dp=0.005m (red). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

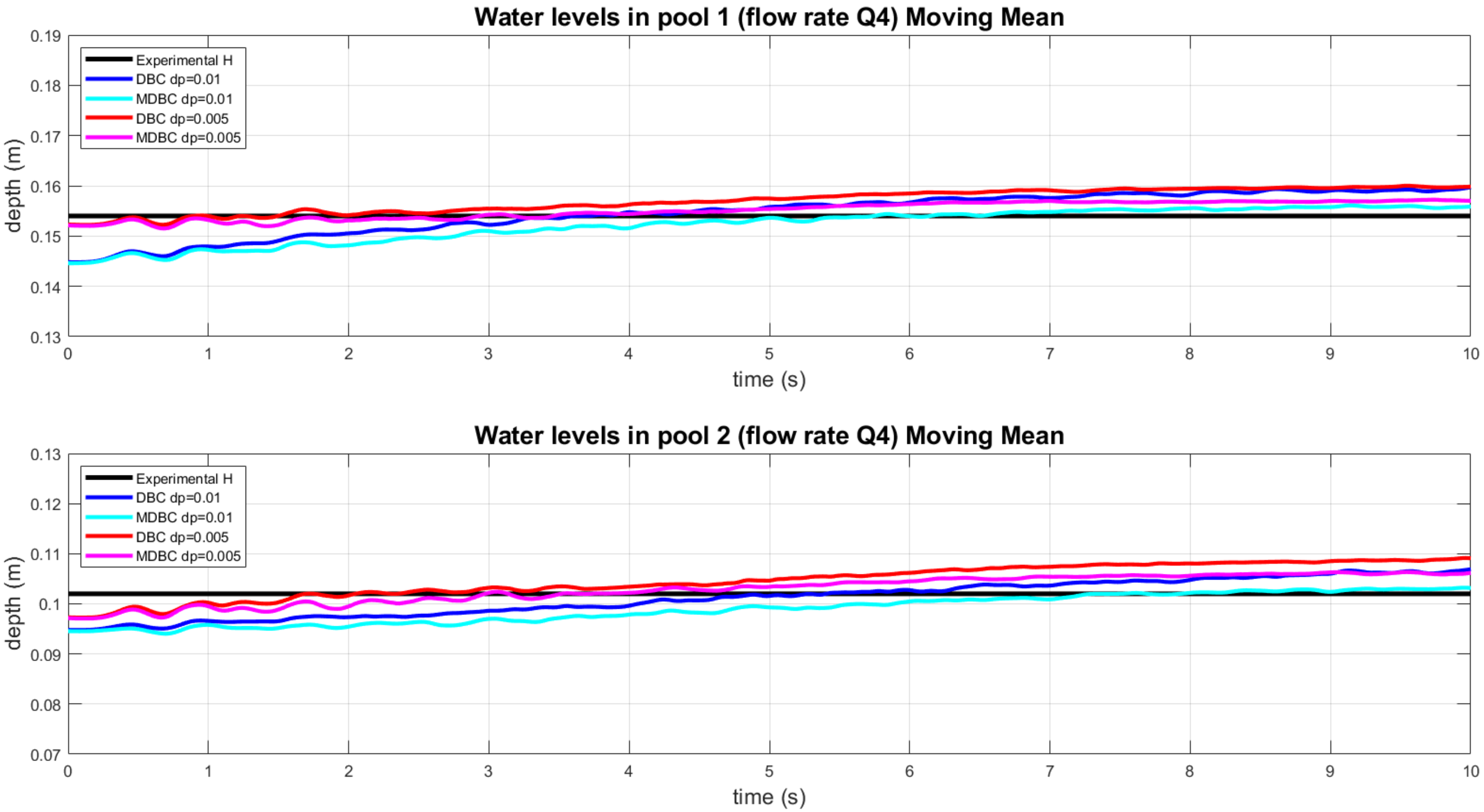


Figure B.3: Plot showing water depths in pools 1 and 2 for flow rate Q4 for two particle spacings: dp=0.01m (blue), and dp=0.005m (red). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

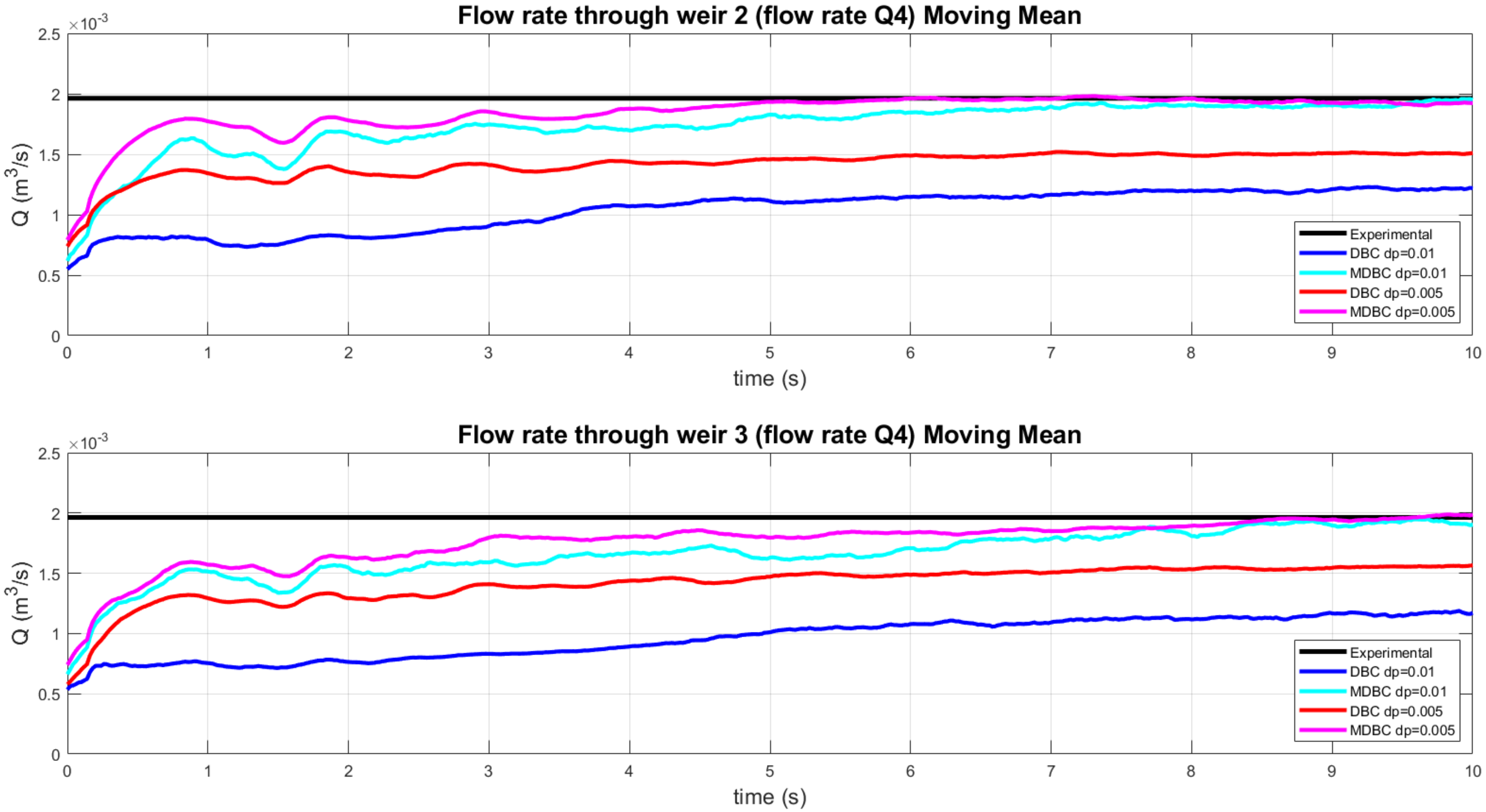


Figure B.4: Plot showing flow rates over weirs 2 and 3 for flow rate Q4 for two particle spacings: dp=0.01m (blue), and dp=0.005m (red). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

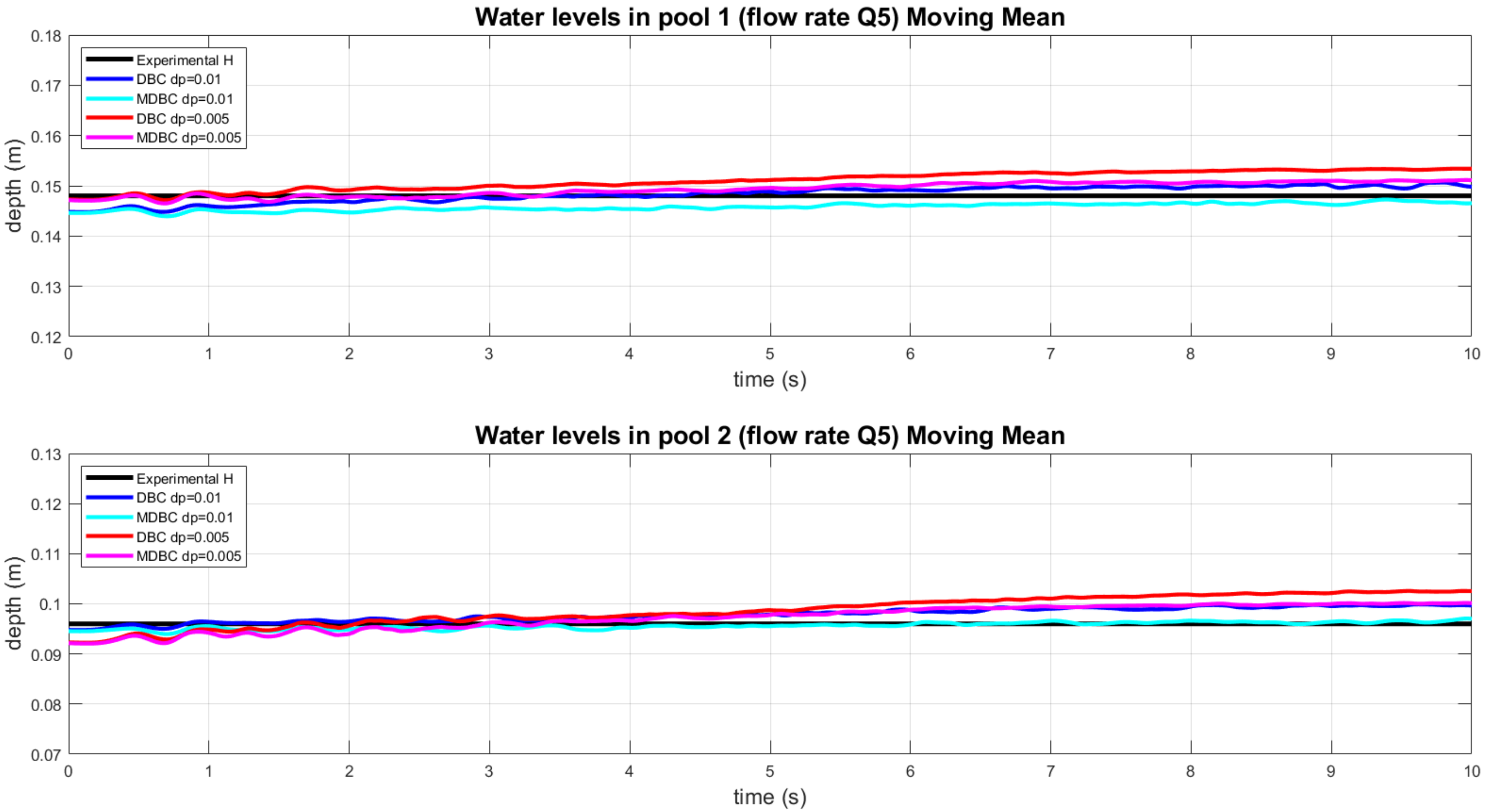


Figure B.5: Plot showing water depths in pools 1 and 2 for flow rate Q5 for two particle spacings: dp=0.01m (blue), and dp=0.005m (red). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

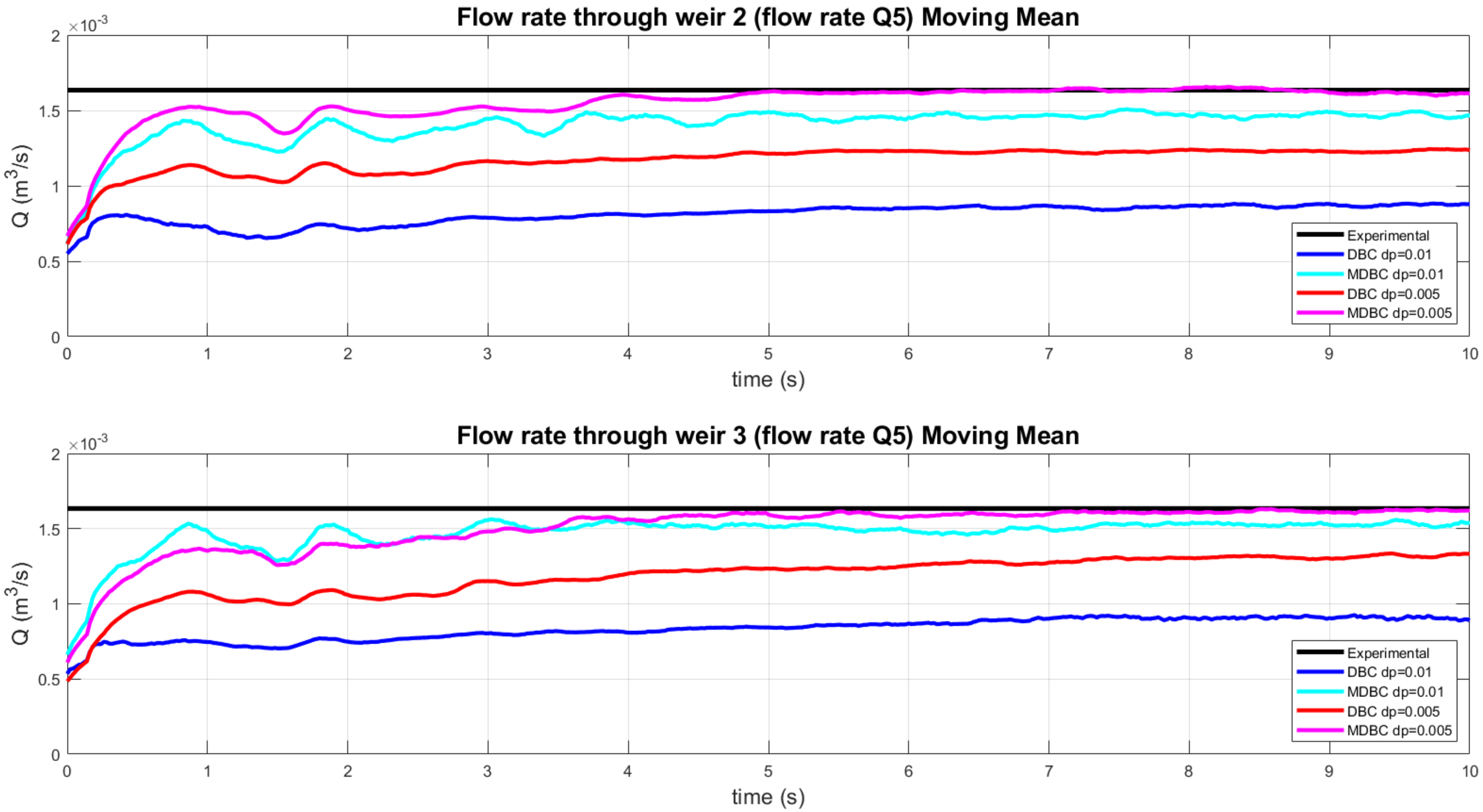
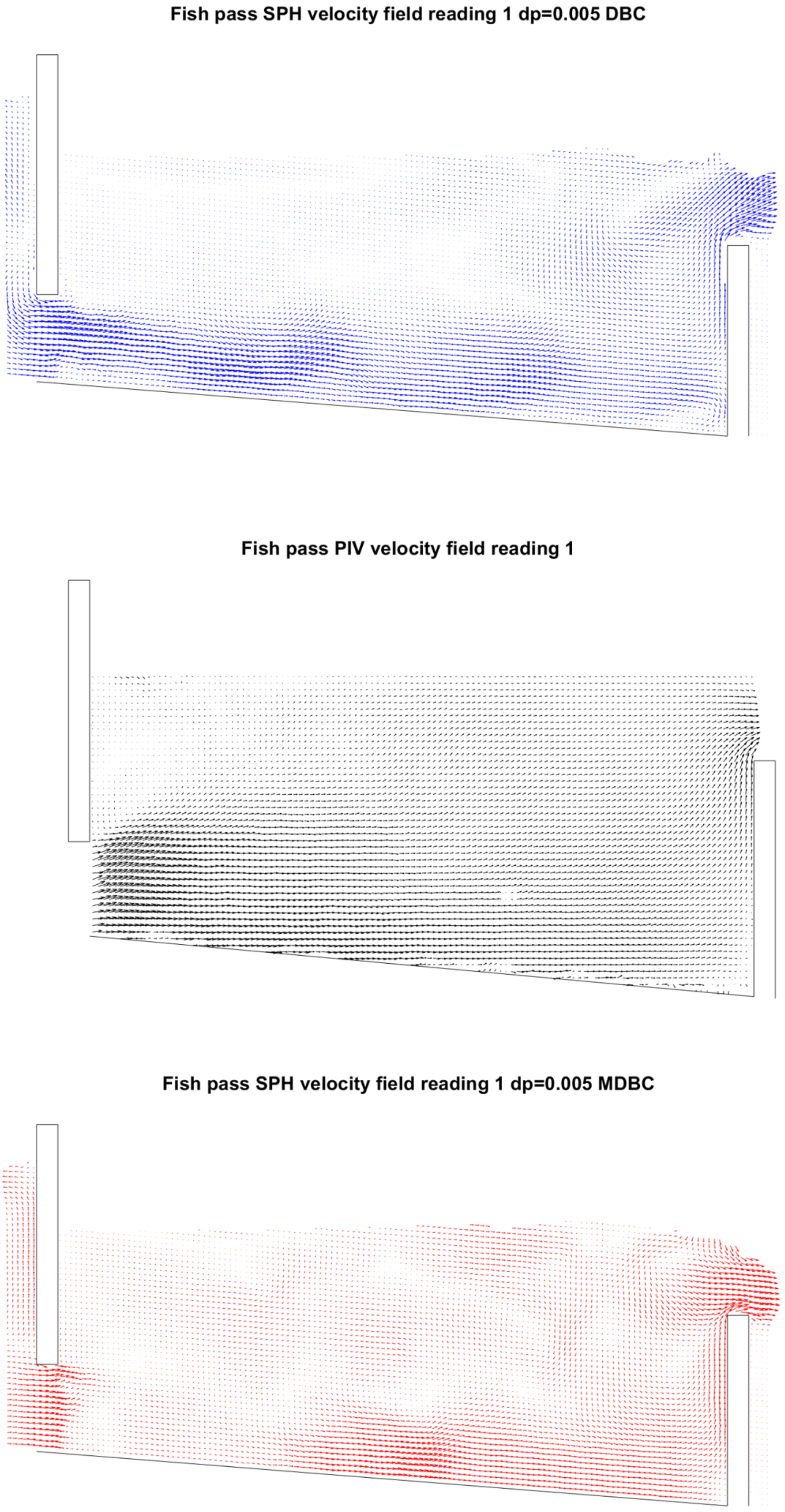


Figure B.6: Plot showing flow rates over weirs 2 and 3 for flow rate Q5 for two particle spacings: dp=0.01m (blue), and dp=0.005m (red). DBC (darker shades) and MDBC (lighter shades), results are averaged using a moving mean.

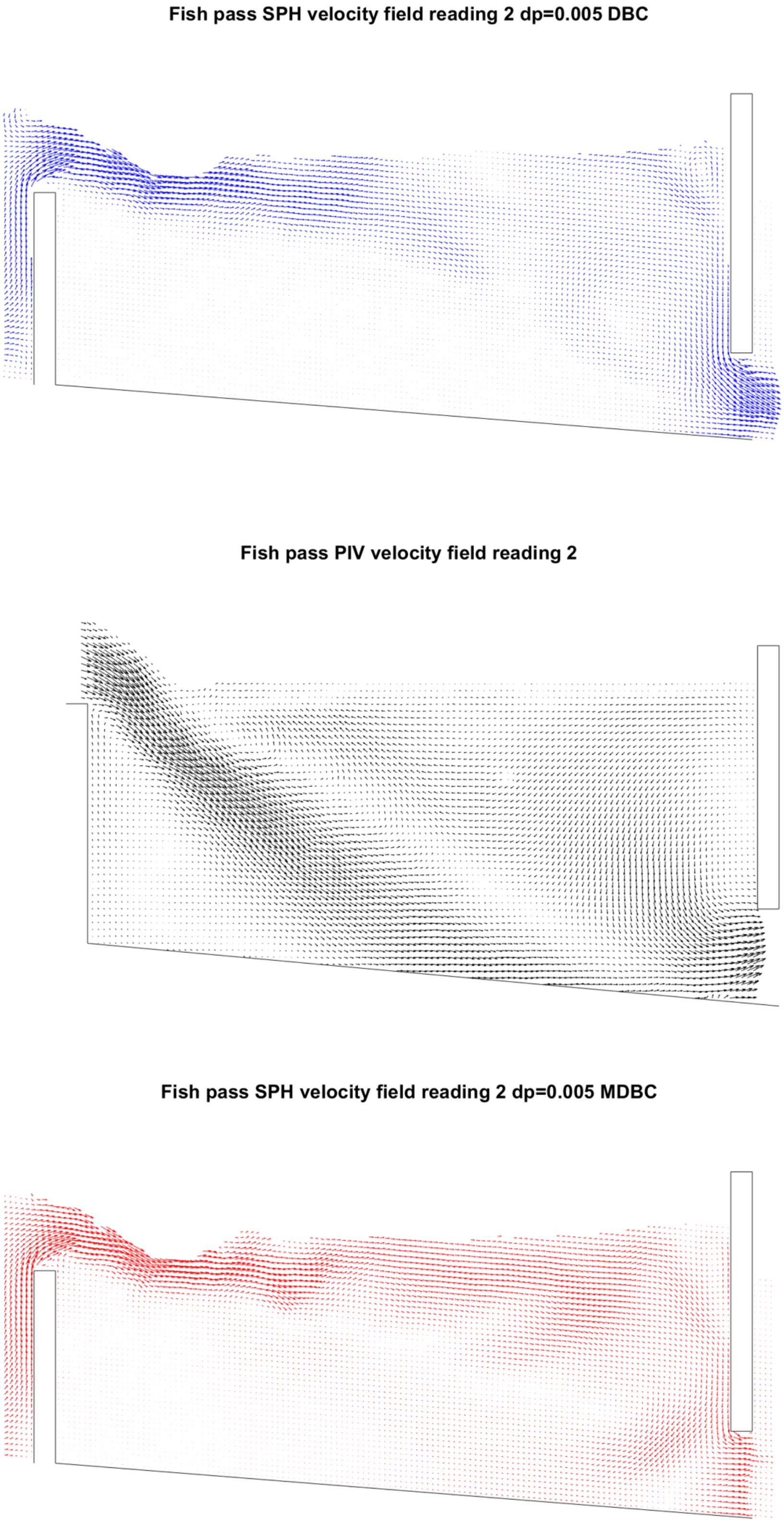


Fish pass SPH velocity field reading 1 dp=0.005 DBC

Fish pass PIV velocity field reading 1

Fish pass SPH velocity field reading 1 dp=0.005 MDBC

Figure B.7: Comparison of velocity vectors in pool 2 plane 1 for DBC (top), PIV (middle) and MDBC (bottom) with particle spacing $dp = 0.005m$



Fish pass SPH velocity field reading 2 dp=0.005 DBC

Fish pass PIV velocity field reading 2

Fish pass SPH velocity field reading 2 dp=0.005 MDBC

Figure B.8: Comparison of velocity vectors in pool 2 plane 2 for DBC (top), PIV (middle) and MDBC (bottom) with particle spacing $dp = 0.005m$

References

S. Adami, X. Y. Hu, and N. A. Adams. A generalized wall boundary condition for smoothed particle hydrodynamics. *Journal of Computational Physics*, 231(21): 7057–7075, 2012. ISSN 00219991. doi: 10.1016/j.jcp.2012.05.005.

C. Altomare, A. J. Crespo, B. D. Rogers, J. M. Dominguez, X. Gironella, and M. Gómez-Gesteira. Numerical modelling of armour block sea breakwater with smoothed particle hydrodynamics. *Computers and Structures*, 130:34–45, 2014. ISSN 00457949. doi: 10.1016/j.compstruc.2013.10.011. URL `http://dx.doi.org/10.1016/j.compstruc.2013.10.011`.

S. Anand, A. T. Paxson, R. Dhiman, J. D. Smith, and K. K. Varanasi. Enhanced condensation on lubricant-impregnated nanotextured surfaces. *ACS Nano*, 6(11):10122–10129, 2012. ISSN 19360851. doi: 10.1021/nn303867y. URL `www.acsnano.org`.

M. Antuono, A. Colagrossi, S. Marrone, and D. Molteni. Free-surface flows solved by means of SPH schemes with numerical diffusive terms. *Computer Physics Communications*, 181(3):532–549, 2010. ISSN 00104655. doi: 10.1016/j.cpc.2009.11.002.

J. L. Barrat and L. Bocquet. Large slip effect at a non-wetting fluid-solid interface. *Physical Review Letters*, 82(23):4671–4674, 1999. ISSN 0031-9007. doi: 10.1103/PhysRevLett.82.4671.

M. Basa, N. J. Quinlan, and M. Lastiwka. Robustness and accuracy of SPH formulations for viscous flow. *International Journal for Numerical Methods in Fluids*, (October 2007):601–629, 2008. ISSN 02712091. doi: 10.1002/fld.

T. Belytschko, Y. Krongauz, J. Dolbow, and C. Gerlach. On the Completeness of Meshfree Particle Methods. 43(5):785–819, 1998.

G. Bilotta, G. Russo, A. Hérault, and C. del Negro. Moving least-squares corrections for smoothed particle hydrodynamics. *Annals of Geophysics*, 54(5):622–633, 2011. ISSN 15935213. doi: 10.4401/ag-5344.

E. C. Bingham. Fluidity and plasticity. *Engineering*, page 440, 1922. ISSN 1098-6596. doi: 10.1017/CBO9781107415324.004.

W. B. Black. Wall Slip and Boundary Effects in Polymer Shear Flows. page 142, 2000.

D. Bonn, J. Eggers, J. Indekeu, and J. Meunier. Wetting and spreading. *Reviews of Modern Physics*, 81(2):739–805, 2009. ISSN 00346861. doi: 10.1103/RevModPhys.81.739.

P. Brown and B. Bhushan. Durable superoleophobic polypropylene surfaces. *Philosophical Transactions A*, 2016. ISSN 1364503X. doi: 10.1098/rsta.2016.0193.

P. S. Brown and B. Bhushan. Mechanically durable, superomniphobic coatings prepared by layer-by-layer technique for self-cleaning and anti-smudge. *Journal of Colloid and Interface Science*, 456:210–218, 2015. ISSN 10957103. doi: 10.1016/j.jcis.2015.06.030.

G. Cao and Z. Li. Numerical flow simulation of fresh concrete with viscous granular material model and smoothed particle hydrodynamics. *Cement and Concrete Research*, 100(July):263–274, 2017. ISSN 00088846. doi: 10.1016/j.cemconres.2017.07.005. URL `http://linkinghub.elsevier.com/retrieve/pii/S0008884617302235`.

N. Casson. A flow equation for pigment-oil dispersions of the printing-ink type. In *Rheol. Disperse Systems, Proc. Conf. Univ. Coll. Swansea*, pages 84–104, 1959.

T. Chihara, K. Yamazaki, R. Itoh, and J. Han. Evaluation of drinking ease relative to the opening diameter and beverage type of aluminum beverage bottles. *Journal of Food Engineering*, 95(2):264–271, 2009. ISSN 02608774. doi: 10.1016/j.jfoodeng.2009.05.006. URL `http://dx.doi.org/10.1016/j.jfoodeng.2009.05.006`.

C. Clanet and G. Searby. On the glug-glug of ideal bottles. *Journal of Fluid Mechanics*, 510:145–168, 2004. ISSN 0022-1120. doi: 10.1017/S002211200400936X. URL `http://www.journals.cambridge.org/abstract{\_}S002211200400936X`.

A. Colagrossi and M. Landrini. Numerical simulation of interfacial flows by smoothed particle hydrodynamics. *Journal of Computational Physics*, 191(2):448–475, 2003. ISSN 00219991. doi: 10.1016/S0021-9991(03)00324-3.

C. Cottin-Bizonne, C. Barentin, E. . Charlaix, L. Bocquet, and J. L. Barrat. Dynamics of simple liquids at heterogeneous surfaces: Molecular-dynamics simulations and hydrodynamic description. *European Physical Journal E*, 15(4):427–438, 2004. ISSN 12928941. doi: 10.1140/epje/i2004-10061-9.

A. J. Crespo, J. M. Domínguez, B. D. Rogers, M. Gómez-Gesteira, S. Longshaw, R. Canelas, R. Vacondio, A. Barreiro, and O. García-Feal. DualSPHysics: Open-source parallel CFD solver based on Smoothed Particle Hydrodynamics (SPH). *Computer Physics Communications*, 187:204–216, 2015. ISSN 00104655. doi: 10.1016/j.cpc.2014.10.004.

A. J. C. Crespo, M. Gomez-Gesteira, and R. A. Dalrymple. Boundary conditions generated by dynamic particles in SPH methods. *Computers, Materials and Continua*, 2007. ISSN 15462218. doi: 10.3970/cmc.2007.005.173.

R. A. Dalrymple and O. Knio. SPH modelling of water waves. In *Coastal Dynamics 2001*, pages 779–787, 2001. ISBN 0784405662. doi: 10.1061/40566(260)80.

P. G. De Gennes. On fluid/wall slippage. *Langmuir*, 18(9):3413–3414, 2002.

M. De Leffe, D. Le Touzé, and B. Alessandrini. A modified no-slip condition in weakly-compressible SPH. In *6th International SPHERIC Workshop*, pages 291–297, 2011.

M. M. Denn. Extrusion instabilities and wall slip. *Annual Review of Fluid Mechanics*, 33:265–287, 2001. ISSN 00664189. doi: 10.1146/annurev.fluid.33.1.265.

J. M. Domínguez, A. J. C. Crespo, G. Fourtakas, B. D. Rogers, J. L. Cercos-Pita, and R. Vacondio. Evaluation of reliability and efficiency of different boundary conditions in a SPH code. In *Proceedings of the 10th SPHERIC Workshop, Parma, Italy*, pages 341–348, 2015.

M. Ellero and R. I. Tanner. SPH simulations of transient viscoelastic flows at low Reynolds number. *Journal of Non-Newtonian Fluid Mechanics*, 132(1-3):61–72, 2005. ISSN 03770257. doi: 10.1016/j.jnnfm.2005.08.012.

M. Emin Erdoğan. On the unsteady unidirectional flows generated by impulsive motion of a boundary or sudden application of a pressure gradient. *International Journal of Non-Linear Mechanics*, 37(6):1091–1106, 2002. ISSN 00207462. doi: 10.1016/S0020-7462(01)00035-X.

X. J. Fan, R. I. Tanner, and R. Zheng. Smoothed particle hydrodynamics simulation of non-Newtonian moulding flow. *Journal of Non-Newtonian Fluid Mechanics*, 165(5-6):219–226, 2010. ISSN 03770257. doi: 10.1016/j.jnnfm.2009.12.004. URL `http://dx.doi.org/10.1016/j.jnnfm.2009.12.004`.

J. Fang, R. G. Owens, L. Tacher, and A. Parriaux. A numerical study of the SPH method for simulating transient viscoelastic free surface flows. *Journal of Non-Newtonian Fluid Mechanics*, 139(1-2):68–84, 2006. ISSN 03770257. doi: 10.1016/j.jnnfm.2006.07.004.

M. Ferrand, D. R. Laurence, B. D. Rogers, D. Violeau, and C. Kassiotis. Unified semi-analytical wall boundary conditions for inviscid, laminar or turbulent flows in the meshless SPH method. *International Journal for Numerical Methods in Fluids*, 71(4):446–472, 2013. ISSN 02712091. doi: 10.1002/fld.3666.

L. L. Ferrás, J. M. Nóbrega, and F. T. Pinho. Analytical solutions for Newtonian and inelastic non-Newtonian flows with wall slip. *Journal of Non-Newtonian Fluid Mechanics*, 175-176(May):76–88, 2012. ISSN 03770257. doi: 10.1016/j.jnnfm.2012.03.004.

G. Fourtakas and B. D. Rogers. Modelling multi-phase liquid-sediment scour and resuspension induced by rapid flows using Smoothed Particle Hydrodynamics (SPH) accelerated with a Graphics Processing Unit (GPU). *Advances in Water Resources*, 92:186–199, 2016. ISSN 03091708. doi: 10.1016/j.advwatres.2016.04.009. URL http://dx.doi.org/10.1016/j.advwatres.2016.04.009.

G. Fourtakas, J. M. Dominguez, R. Vacondio, and B. D. Rogers. Local uniform stencil (LUST) boundary condition for arbitrary 3-D boundaries in parallel smoothed particle hydrodynamics (SPH) models. *Computers and Fluids*, 190:346–361, 2019. ISSN 00457930. doi: 10.1016/j.compfluid.2019.06.009. URL https://doi.org/10.1016/j.compfluid.2019.06.009.

F. Geiger, K. Velten, and F. J. Methner. 3D CFD simulation of bottle emptying processes. *Journal of Food Engineering*, 109(3):609–618, 2012. ISSN 02608774. doi: 10.1016/j.jfoodeng.2011.10.008. URL http://dx.doi.org/10.1016/j.jfoodeng.2011.10.008.

R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181(3):375–389, 1977. ISSN 0035-8711. doi: 10.1093/mnras/181.3.375.

L. Goffin. *Development of a didactic SPH model*. Master, Universite de Liege, 2013.

M. Gomez-Gesteira, B. D. Rogers, R. A. Dalrymple, and A. J. Crespo. State-of-the-art of classical SPH for free-surface flows. *Journal of Hydraulic Research*, 48 (extra):000, 2009. ISSN 0022-1686. doi: 10.3826/jhr.2010.0012.

J. Harting, C. Kunert, and H. J. Herrmann. Lattice Boltzmann simulations of apparent slip in hydrophobic microchannels. *Interactions*, 328(Md):7, 2006. ISSN 0295-5075. doi: 10.1209/epl/i2006-10107-8. URL http://arxiv.org/abs/physics/0509035.

J. Harting, C. Kunert, and J. Hyväluoma. Lattice Boltzmann simulations in microfluidics: Probing the no-slip boundary condition in hydrophobic, rough, and surface nanobubble laden microchannels. *Microfluidics and Nanofluidics*, 8(1): 1–10, 2010. ISSN 16134982. doi: 10.1007/s10404-009-0506-6.

M. Harvey and T. A. Waigh. Optical coherence tomography velocimetry in controlled shear flow. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 83(3):1–5, 2011. ISSN 15393755. doi: 10.1103/PhysRevE.83.031502.

W. H. Herschel. Consistency of Rubber-Benzene Solutions. *Industrial and Engineering Chemistry*, 16(9):927, 1924. ISSN 00197866. doi: 10.1021/ie50177a019.

S. Huntley, T. Rendall, M. Longana, T. Pozegic, K. Potter, and I. Hamerton. Validation of a smoothed particle hydrodynamics model for a highly aligned discontinuous fibre composites manufacturing process. *Composites Science and Technology*, 196, 2020. ISSN 02663538. doi: 10.1016/j.compscitech.2020.108152.

H. S. Joyner. *Rheology of Semisolid Foods*. Food Engineering series. Springer, Cham, 2019. ISBN 9783030271336.

J. R. King and S. J. Lind. The Kaye effect: New experiments and a mechanistic explanation. *Journal of Non-Newtonian Fluid Mechanics*, 273, 2019. ISSN 03770257. doi: 10.1016/j.jnnfm.2019.104165.

K. M. Kleefsman, G. Fekken, A. E. Veldman, B. Iwanowski, and B. Buchner. A Volume-of-Fluid based simulation method for wave impact problems. *Journal of Computational Physics*, 206(1):363–393, 2005. ISSN 10902716. doi: 10.1016/j.jcp.2004.12.007.

E. Lauga and H. Stone. Effective slip in pressure-driven Stokes flow. *J. Fluid Mech*, 489:55–77, 2003. ISSN 00221120. doi: 10.1017/S0022112003004695.

E. Lauga, M. P. Brenner, and H. A. Stone. Microfluidics: The no-slip boundary condition. *Springer Handbook of Experimental Fluid Mechanics*, pages 1219–1240, 2005. ISSN 15264564. doi: 10.1007/978-3-540-30299-5_19. URL `http://arxiv.org/abs/cond-mat/0501557`.

B. J. Leimkuhler and C. Matthews. *Numerical Integrators. In: Molecular Dynamics. Interdisciplinary Applied Mathematics, vol 39.* Springer, Cham., 2015. URL `https://doi.org/10.1007/978-3-319-16375-8{\_}2`.

S. J. Lind, R. Xu, P. K. Stansby, and 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):1499–1523, 2012. ISSN 10902716. doi: 10.1016/j.jcp.2011.10.027.

LiquiGlide. LiquiGlide Website, accessed 16-08-2020, 2020. URL `https://liquiglide.com/`.

M. B. Liu and G. R. Liu. Restoring particle consistency in smoothed particle hydrodynamics. *Applied Numerical Mathematics*, 56(1):19–36, 2006. ISSN 01689274. doi: 10.1016/j.apnum.2005.02.012.

E. Y. Lo and S. Shao. Simulation of near-shore solitary wave mechanics by an incompressible SPH method. *Applied Ocean Research*, 24(5):275–286, 2002. ISSN 01411187. doi: 10.1016/S0141-1187(03)00002-6.

L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *The Astronomical Journal*, 82(12):1013, 1977. ISSN 00046256. doi: 10.1086/112164.

M. Ma and R. M. Hill. Superhydrophobic surfaces. *Current Opinion in Colloid and Interface Science*, 11(4):193–202, 2006. ISSN 13590294. doi: 10.1016/j.cocis.2006.06.002.

F. Macia, M. Antuono, L. M. Gonzalez, and A. Colagrossi. Theoretical Analysis of the No-Slip Boundary Condition Enforcement in SPH Methods. *Progress of Theoretical Physics*, 125(6):1091–1121, 2011. ISSN 0033-068X. doi: 10.1143/PTP.125.1091.

A. Malm, T. A. Waigh, S. Jaradat, and R. Tomlin. Optical Coherence Tomography Velocimetry with complex fluids. In *Journal of Physics: Conference Series*, volume 602, 2015. doi: 10.1088/1742-6596/602/1/012039. URL `http://iopscience.iop.org/1742-6596/602/1/012039`.

A. V. Malm. *OCT Velocimetry and X-ray Scattering Rheology of Complex Fluids*. PhD thesis, University of Manchester, 2015.

A. V. Malm and T. Waigh. Elastic turbulence in DNA solutions measured with optical coherence tomography velocimetry. *Journal of Materials Chemistry*, 2(2): 303, 2014. ISSN 2050-7488. doi: 10.1039/x0xx00000x.

A. V. Malm, A. W. Harrison, and T. A. Waigh. Optical coherence tomography velocimetry of colloidal suspensions. *Soft matter*, 10(41):8210–5, 2014. ISSN 1744-6848. doi: 10.1039/c4sm01111b.

S. Marrone, M. Antuono, A. Colagrossi, G. Colicchio, D. Le Touzé, and G. Graziani. δ-SPH model for simulating violent impact flows. *Computer Methods in Applied Mechanics and Engineering*, 200(13-16):1526–1542, 2011. ISSN 00457825. doi: 10.1016/j.cma.2010.12.016. URL `http://dx.doi.org/10.1016/j.cma.2010.12.016`.

E. Martines, K. Seunarine, H. Morgan, N. Gadegaard, C. D. Wilkinson, and M. O. Riehle. Superhydrophobicity and superhydrophilicity of regular nanopatterns. *Nano Letters*, 5(10):2097–2103, 2005. ISSN 15306984. doi: 10.1021/nl051435t.

J. C. Maxwell. On stresses in rarified gases arising from inequalities of temperature. *Philos. Trans. R. Soc.*, 11(January):231–256, 1879. ISSN 0261-0523. doi: 10.1098/rstl.1879.0067.

A. Mayrhofer. *An investigation into wall boundary conditions and three-dimensional turbulent flows using smoothed particle hydrodynamics*. Phd, University of Manchester, 2014.

A. Mayrhofer, M. Ferrand, C. Kassiotis, D. Violeau, and F. X. Morel. Unified semi-analytical wall boundary conditions in SPH: analytical extension to 3-D. *Numerical Algorithms*, 68(1):15–34, 2015. ISSN 15729265. doi: 10.1007/s11075-014-9835-y.

S. Mer, O. Praud, H. Neau, N. Merigoux, J. Magnaudet, and V. Roig. The emptying of a bottle as a test case for assessing interfacial momentum exchange models for Euler–Euler simulations of multi-scale gas-liquid flows. *International Journal of Multiphase Flow*, 106:109–124, 2018. ISSN 03019322.

doi: 10.1016/j.ijmultiphaseflow.2018.05.002. URL `https://doi.org/10.1016/j.ijmultiphaseflow.2018.05.002`.

D. Molteni and A. Colagrossi. A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH. *Computer Physics Communications*, 180(6):861–872, 2009. ISSN 00104655. doi: 10.1016/j.cpc.2008.12.004.

J. J. Monaghan. Simulating free surface flows with SPH. *Journal of Computational Physics*, 110(2):399–406, 1994. ISSN 00219991. doi: 10.1006/jcph.1994.1034.

J. J. Monaghan and J. B. Kajtar. SPH particle boundary forces for arbitrary boundaries. *Computer Physics Communications*, 180(10):1811–1820, 2009. ISSN 00104655. doi: 10.1016/j.cpc.2009.05.008. URL `http://dx.doi.org/10.1016/j.cpc.2009.05.008`.

J. Monaghan, J. Smoothed Particle Hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30:543–574, 1992. ISSN 0066-4146. doi: 10.1146/annurev.aa.30.090192.002551. URL `http://adsabs.harvard.edu/full/1992ARA{\&}A..30..543M`.

J. P. Morris and J. J. Monaghan. A switch to reduce SPH viscosity. *Journal of Computational Physics*, 136(1):41–50, 1997. ISSN 00219991. doi: 10.1006/jcph.1997.5690.

J. P. Morris, P. J. Fox, and Y. Zhu. Modeling Low Reynolds Number Incompressible Flows Using SPH. *Journal of Computational Physics*, 136:214–226, 1997. ISSN 00219991. doi: 10.1006/jcph.1997.5776.

A. M. Nasar, B. D. Rogers, A. Revell, and P. K. Stansby. An Eulerian Weakly Compressible SPH Immersed Boundary Method for Fluid-Structure Interaction. In *Proceedings of the 11th international SHPERIC Workshop*, 2016.

C. Navier. M´emoire sur les lois du mouvement des fluides. *M´emoires de l'Acad´emie Royale des Sciences de l'Institut de France*, VI:389–440, 1823.

K. C. Ng, Y. L. Ng, and W. H. Lam. Particle simulation and flow sequence on drainage of liquid particles. *Computers and Mathematics with Applications*, 66 (8):1437–1451, 2013. ISSN 08981221. doi: 10.1016/j.camwa.2013.08.018.

Y. Nishio, K. Niwa, and T. Ogawa. Numerical simulation of a pouring flow from a beverage can. In *ASME-JSME-KSME 2019 8th Joint Fluids Engineering Conference, AJKFluids 2019*, volume 3A-2019, pages 1–5, 2019. ISBN 9780791859049. doi: 10.1115/AJKFluids2019-5308.

G. Novak, A. Tafuni, J. M. Domínguez, M. Četina, and D. Žagar. A numerical study of fluid flow in a vertical slot fishway with the smoothed particle hydrodynamics method. *Water (Switzerland)*, 11(9), 2019. ISSN 20734441. doi: 10.3390/w11091928.

J. O. Ojediran and A. O. Raji. *Flow and functional models for rheological properties of fluid foods*, volume 17 of *Food Engineering Series*. Springer US, Boston, MA, 3rd ed. 20 edition, 2010. ISBN 1461492300. doi: 10.1007/978-1-4614-9230-6.

T. C. Papanastasiou. Flows of Materials with Yield. *Journal of Rheology*, 31(5): 385–404, 1987. ISSN 0148-6055. doi: 10.1122/1.549926.

A. N. Parshikov, S. A. Medin, I. I. Loukashenko, and V. A. Milekhin. Improvements in SPH method by means of interparticle contact algorithm and analysis of perforation tests at moderate projectile velocities. *International Journal of Impact Engineering*, 24(8):779–796, 2000. ISSN 0734743X. doi: 10.1016/S0734-743X(99)00168-2.

N. J. Quinlan, M. Basa, and M. Lastiwka. Truncation error in mesh-free particle methods. *International Journal for Numerical Methods in Engineering*, 66(13): 2064–2085, 2006. ISSN 00295981. doi: 10.1002/nme.1617.

M. Renardy, Y. Renardy, and J. Li. Numerical Simulation of Moving Contact Line Problems Using a Volume-of-Fluid Method. *Journal of Computational Physics*, 171(1):243–263, 2001. ISSN 00219991. doi: 10.1006/jcph.2001.6785. URL `http://www.scopus.com/inward/record.url?eid=2-s2.0-0001532298{\&}partnerID=tZOtx3y1`.

O. Schmidt and J. Kubie. An experimental investigation of outflow of liquids from single-outlet vessels. *International Journal of Multiphase Flow*, 21(6):1163–1168, 1995. ISSN 03019322. doi: 10.1016/0301-9322(95)00051-X.

S. Shao and E. Y. M. Lo. Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface. *Advances in Water Resources*, 26 (7):787–800, 2003. ISSN 03091708. doi: 10.1016/S0309-1708(03)00030-7.

H. T. Shen, J. Su, and L. Liu. SPH Simulation of River Ice Dynamics. *Journal of Computational Physics*, 165(2):752–770, 2000. ISSN 00219991. doi: 10.1006/jcph.2000.6639. URL `http://www.idealibrary.com`.

A. Skillen, S. Lind, P. K. Stansby, and B. D. Rogers. Incompressible smoothed particle hydrodynamics (SPH) with reduced temporal noise and generalised Fickian smoothing applied to body-water slam and efficient wave-body interaction. *Computer Methods in Applied Mechanics and Engineering*, 265, 2013. ISSN 00457825. doi: 10.1016/j.cma.2013.05.017.

J. D. Smith, R. Dhiman, S. Anand, E. Reza-Garduno, R. E. Cohen, G. H. McKinley, and K. K. Varanasi. Droplet mobility on lubricant-impregnated surfaces. *Soft Matter*, 9(6):1772–1780, 2013. ISSN 1744-683X. doi: 10.1039/c2sm27032c. URL `http://dx.doi.org/10.1039/C2SM27032C`.

T. Sochi. Slip at Fluid-Solid Interface. *Polymer Reviews*, 51(4):309–340, 2011. ISSN 1558-3724. doi: 10.1080/15583724.2011.615961. URL `http://arxiv.org/abs/1101.4421`.

Z. G. Sun, Y. Y. Liang, and G. Xi. Numerical study of the flowing sequence of a pouring liquid. *Science China: Physics, Mechanics and Astronomy*, 54(8): 1514–1519, 2011. ISSN 16747348. doi: 10.1007/s11433-011-4387-9.

A. Tafuni, J. M. Domínguez, R. Vacondio, and A. J. Crespo. A versatile algorithm for the treatment of open boundary conditions in Smoothed particle hydrodynamics GPU models. *Computer Methods in Applied Mechanics and Engineering*, 342:604–624, 2018. ISSN 00457825. doi: 10.1016/j.cma.2018.08.004.

A. A. Tehrani, M. A. Patrick, and A. A. Wragg. Dynamic fluid flow behaviour of a tank draining through a vertical tube. *International Journal of Multiphase Flow*, 18(6):977–988, 1992. ISSN 03019322. doi: 10.1016/0301-9322(92)90071-N.

Unilever. Unilever: Hellman's bottle design, accessed 16-08-2020, 2015. URL `https://www.unilever.co.uk/about/innovation/latest-news/squeezing-more-hellmanns-from-every-bottle.html`.

Unilever. Unilever Website, accessed 16-08-2020, 2020a. URL `https://www.unilever.co.uk/`.

Unilever. Unilever Sustainability, accessed 16-08-2020, 2020b. URL `https://www.unilever.co.uk/news/news-and-features/2020/were-testing-reusable-refillable-packaging-to-help-cut-waste.html`.

R. Vacondio, C. Altomare, M. De Leffe, X. Hu, D. Le Touzé, S. Lind, J. C. Marongiu, S. Marrone, B. D. Rogers, and A. Souto-Iglesias. Grand challenges for Smoothed Particle Hydrodynamics numerical schemes, 2020. ISSN 21964386.

D. Violeau. *Fluid Mechanics and the SPH Method: Theory and Applications*. 2012. ISBN 9780191741227. doi: 10.1093/acprof:oso/9780199655526.001.0001.

D. Violeau and B. D. Rogers. Smoothed particle hydrodynamics (SPH) for free-surface flows: past , present and future. *Journal of Hydraulic Research*, 1686 (January):1–26, 2016. ISSN 00221686. doi: 10.1080/00221686.2015.1119209. URL `http://dx.doi.org/10.1080/00221686.2015.1119209`.

VWR. VWR Website, accessed 16-08-2020, 2020. URL `https://uk.vwr.com/store/`.

H. Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics*, 4 (1):389–396, 1995. ISSN 10197168. doi: 10.1007/BF02123482.

P. B. Whalley. Flooding, slugging and bottle emptying. *International Journal of Multiphase Flow*, 13(5):723–728, 1987. ISSN 03019322. doi: 10.1016/0301-9322(87)90048-6.

P. B. Whalley. Two-phase flow during filling and emptying of bottles. *International Journal of Multiphase Flow*, 17(1):145–152, 1991. ISSN 03019322. doi: 10.1016/0301-9322(91)90076-F.

A. M. Xenakis, S. J. Lind, P. K. Stansby, and B. D. Rogers. An incompressible SPH scheme with improved pressure predictions for free-surface generalised Newtonian flows. *Journal of Non-Newtonian Fluid Mechanics*, 218:1–15, 2015. ISSN 03770257. doi: 10.1016/j.jnnfm.2015.01.006. URL `http://dx.doi.org/10.1016/j.jnnfm.2015.01.006`.

C. Yan-Yan, Y. Hou-Hui, and L. Hua-Bing. Boundary slip and surface interaction: A Lattice Boltzmann simulation. *Chinese Physics Letters*, 25(1):184–187, 2008. ISSN 0256-307X. doi: 10.1088/0256-307X/25/1/050. URL `http://stacks.iop.org/0256-307X/25/i=1/a=050?key=crossref.996f9bb5eede762974630d147cb9a68b`.

M. Yildiz, R. A. Rook, and A. Suleman. SPH with the multiple boundary tangent method. *International Journal for Numerical Methods in Engineering*, 77(10):1416–1438, 2009. ISSN 00295981. doi: 10.1002/nme.2458.

F. Zhang, A. Crespo, C. Altomare, J. Domínguez, A. Marzeddu, S. P. Shang, and M. Gómez-Gesteira. Dualsphysics: A numerical tool to simulate real breakwaters. *Journal of Hydrodynamics*, 30(1), 2018. ISSN 18780342. doi: 10.1007/s42241-018-0010-0.