D.A.L. Schouten

Simulation of the interception of plastic fluxes in rivers

Simulation of the interception of plastic fluxes in rivers

by

D.A.L. (David) Schouten

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Tuesday October 5 2021, at 15:00.

Student number :	4373650	
Project duration:	September 1, 2020 – October 5, 2021	
Thesis committee:	Prof. dr. ir. W.S.J. Uijttewaal	TU Delft, chair
	Dr. ir. J. D. Bricker	TU Delft & University of Michigan
	Dr. ir. E. Mosselman	TU Delft
	Dr. ir. G. Duró	Witteveen+Bos

Preface

This master's thesis is performed in cooperation with Witteveen+Bos, an engineering and consultancy firm based in the Netherlands with a sustainable and innovative profile. The bureau is internationally active and operates in 11 countries. This project concludes my study Hydraulic Engineering at Delft University of Technology, with the track River Engineering.

Firstly, I would like to express my gratitude to my committee. Thank you all for giving me the opportunity in exploring the possibilities into the small scale plastic and fluid dynamics and in CFD modelling. Many thanks to Erik Mosselman, you always reminded me of the relevance and context of my project. You made sure I did not forget the bigger picture. Jeremy Bricker, thank you very much for guiding me through the first stages of my thesis as my main supervisor and for providing me a great deal in understanding the different model processes.

A word of gratitude to Wim Uijttewaal for aiding me in working on the final product of my thesis. You always provided me with valuable knowledge and reassurance in the unknown field of plastic dynamics. The weekly meetings always helped me to move one step at a time to the final product.

Great thanks to Gonzalo Duró. The weekly meetings gave me better insight into giving my research a better academic and more practical approach. Thank you for always making enough time free for me. I also liked the conversations with you on other work and lots of non-work related aspects.

I would like to convey special thanks to everyone I met at Witteveen+Bos, especially for making me feel part of the company and the group. The inclusion in the group meetings and different events made my experience unforgettable. The student CFD community set up was useful in providing me in better understanding this modelling environment and offered me another moment to reflect on my choices.

Lastly, I would like to thank all my friends and family for helping me through this rough time of a world-wide pandemic. Your unconditional support and focus on the things that matter in life helped me to reach the final goal.

David A.L. Schouten
Rotterdam, October 2021

"*Essentially, all models are wrong, but some are useful*"

George E. P. Box

Summary

One of the preferred features of plastic is its high durability. This merit is its main disadvantage in the natural environment. When poorly disposed of, plastic waste can enter rivers from surface waters and subsequently end up in the ocean. Here, plastic spreads out over the entire ocean and ultimately ends in ocean gyres. Plastic material in the ocean causes environmental damage. Marine animals can get entangled in plastic debris, plastic accumulation occurs when consuming seafood and the material spreads toxins in the sea water. It has also caused economic damage: marine plastic accumulation led to \$6 to 19 billion to key economic sectors for 87 coastal countries in 2020 (Deloitte, 2019). Attempts have been made to capture buoyant plastics from the ocean with interception devices, with mixed success.

The placement of an interception device in a river is thought to be a better approach to the problem. A standard interception design is that of a floating boom. Plastic debris is transported by the river flow and these objects guide buoyant plastics into a collection tray. Currently, these interception designs are still in their infancy. A difficulty in the optimisation of these interception devices lies in the limited knowledge of the trajectories of plastic waste. Computational fluid dynamics (CFD) modelling can help to create a better understanding of these aspects. This tool can provide a rapid assessment of different flow conditions with different types of plastic debris modelled. Further knowledge into this field can ultimately help into better retention of plastics in rivers.

Plastic modelling for buoyant particles has been performed in the marine environment (Van Utenhove, 2019) and in the river (Van Welsenes, 2019) with a Lagrangian study. However, both works did not perform a validation study. The combination of a CFD model and the results of physical model tests can be valuable in removing uncertainties related to plastic trajectories and the modelling of these. One physical model test performed on plastics is the release of plastic particles in a flume (Zaat, 2020).. Small films were released in a 2DV physical model, representing plastic bags. It was found that these particles followed an inverted Rouse profile for spherical particles times a shape factor $\alpha_{film} = 0.64$.

The study by Zaat (2020) was replicated in the CFD modelling environment of OpenFOAM in this thesis. Subsequently the retention of plastic particles was investigated. Three hydrodynamic conditions were investigated, with uniform inlet flow velocities of 0.10, 0.55 and 0.90 m/s. The $k - \epsilon$ turbulence model was used. An Euler-Euler approach was applied with a discrete and continuous phase. This modelling approach is computationally less expensive compared to a Euler-Lagrangian framework and can be easily established for different cases. The drag models Gibilaro, Wen-Yu and Syamlal-O'Brien were tested to investigate which concentration profile best fits an inverted Rouse profile for rising particles. The interception device was modelled as a square obstruction of 10% of the water depth and the model was assumed as a rigid-lid. The retention of particles by the system was explored for neutrally buoyant particles (ρ_p=750 kg/m^3), light material (D_p=0.001 mm ρ_p=750 kg/m^3) and high-density polyethylene (HDPE) and polypropylene (PP) films.

This study found relative differences in near-surface flow and the analytical approach of 3 to 4%. The Gibilaro model was best applicable and model coefficients for the virtual mass, lift and turbulent dispersion of 0.5, 1.6 and 1.0 were found. The concentration profiles followed the inverted Rouse profile and that for films closely for the medium and high flow cases, but not for the low flow case. A lower retention of plastics was observed for increasing flow velocities. In the extreme case of low flow and light particles a build-up of particles near-surface can be observed and runs against the streamwise direction.

Several limitations were present in this study. Only two parameters of the plastic particles were adjustable, which were the particle diameter and density. In reality, plastic debris has different shapes will experience drag differently, especially the friction drag. The rigid-lid model is valid only if the velocity head is smaller than that of 10% of the water depth, which is not the case for the high flow velocity case.

The results show that on the basis of a rise velocity and a density a representative diameter can be determined from Stokes' law. The Euler-Euler modelling approach provides an accurate assessment and gives a proper representation of the particles for the Rouse distributions. This is done without handling the particles individually and thus results can be found more quickly. Both for HDPE and PP inverted Rouse profiles were found that followed the theoretical profile with an underestimation of 10%-20% near-surface. The retention was influenced mainly by the size of the wake. It is evident that the interception system can work for buoyant particles. The object can be placed during low flow conditions for the highest efficiency. For high discharge in rivers more particles will be in suspension and thus further away from the water surface, which makes this design less preferred during high flow conditions.

Recommendations for further research are to perform more physical model tests, implement a non-spherical drag model applicable for plastic material and to apply a three-phase model. This removes uncertainties related to the model and to the current knowledge in particle dynamics. The addition of physical model tests would benefit the current knowledge the greatest. It is only possible to expand the complexity of the CFD model with a larger availability of these physical model tests.

Contents

List of Figures

List of Tables

Nomenclature

Physics constants

g	Gravitational acceleration	9.8 m/s^2

Latin symbols

α_{phase}	Phase fraction of the given phase	[-]
a	Certain location in depth	[m]
A_p	Area particle	[m^2]
C	Chézy coefficient	[m$^{1/2}$/s]
c	Particle concentration	[kg/m^3]
C_1	Model coefficient, by default equal to 1.44	[s]
C_2	Model coefficient, by default equal to 1.92	[-]
C_μ	Model coefficient, by default equal to 0.09	[-]
c_a	Concentration of particles at given point a	[kg/m^3]
C_D	Drag coefficient	[-]
c_f	Friction coefficient	[-]
$C_{3,RDT}$	Rapid-distortion theory compression term coefficient, by default equal to 0.0	[-]
D	Particle diameter	[m]
D_ϵ	Effective diffusivity for ϵ	[m^2/s]
d_{eff}	Effective particle diameter	[m]
f_i	Driving force	[N/m^3]
G	Turbulent kinetic energy production rate due to the anisotropic part of the Reynolds-stress tensor	[m^2/s^3]
h	Water depth	[m]
I	Turbulence intensity	[%]
K	Drag parameter	[-]
k	Turbulent kinetic energy	[m^2/s^2]

k_s	Nikuradse roughness height	[m]
L	Reference length scale	[m]
ℓ	Turbulence length scale, $0.07 \cdot L$	[m]
Le	Entrance length	[m]
M_a	Interphase momentum force	[N/m^3]
M_a^d	Drag force	[N/m^3]
M_a^{td}	Turbulent dispersion force	[N/m^3]
M_a^{vm}	Virtual mass force	[N/m^3]
M_a^{wl}	Wall lubrication force	[N/m^3]
P	Turbulent kinetic energy production rate	[m^2/s^3]
p	Fluid pressure	[m^2/s^2]
$\overline{p}$	Average pressure	[m^2/s^2]
p'	Deviations in the pressure	[m^2/s^2]
p_{rgh}	Hydrostatic pressure	[m^2/s^2]
R_p	Radius particle	[m]
Re	Reynolds number	[-]
Re_p	Particle Reynolds number	[-]
S_ϵ	Internal source term for ϵ	[kg/s^3]
S_k	Internal source term for k	[kg/s^3]
S_{bij}	Deviatoric part of the fluid-phase strain rate tensor	[kg/s^3]
$S_{fvOptions,\epsilon}$	Source term introduced by fvOptions dictionary for k	[kg/s^3]
$S_{fvOptions,k}$	Source term introduced by fvOptions dictionary for ϵ	[kg/s^3]
t_l	Characteristic timescale of energetic eddies	[s]
t_p	Particle response time	[s]
$\overline{u}$	Average velocity	[m/s]
u'	Deviations in the velocity	[m/s]
u^+	Dimensionless velocity	[-]
u_i^f	Fluid velocity in different directions, with $i = 1, 2$ used for the streamwise and vertical component, respectively	[m/s]
u^P	Velocity particle	[m/s]
u_i^p	Particle velocity in different directions, with $i = 1, 2$ used for the streamwise and vertical component, respectively	[m/s]

u_*	Friction velocity	[m/s]
V_p	Volume particle	[m^3]
v_{stokes}	Stokes velocity	[m/s]
w_r	Rise velocity	[m/s]
w_s	Settling velocity	[m/s]
x	Point in streamwise direction	[m]
y	Point in water depth	[m]
y^+	Dimensionless wall distance	[-]
y_0	Roughness height	[m]

Greek symbols

α	Solid fraction	[-]
α_{film}	Shape factor of particles to films, from experiment by Zaat (2020): 0.64	[-]
β	Fluid fraction, $1-\alpha$	[-]
β	Rouse number	[-]
Δt	Time step	[s]
Δu	Velocity difference per time step	[m/s]
ϵ	Turbulent energy dissipation	[m^2/s^3]
ϵ_T	Turbulent eddy diffusivity	[m/s^2]
η	Dynamic viscosity, taken as 0.001 for water	[kg/m/s]
κ	Von Kármán constant	[-]
ν	Kinematic viscosity, taken as 1E-6 for water	[m/s^2]
ν^{mix}	Mixture viscosity	[m/s^2]
ν_t	Turbulent eddy viscosity	[m/s^2]
ϕ	Particle volumetric concentration	[-]
ψ	Shape factor	[-]
ρ	Density	[kg/m^3]
ρ_l	Density water, taken as 1000 in this study	[kg/m^3]
ρ_p	Density particle	[kg/m^3]
σ_T	Schmidt number	[-]
τ_{ij}^a	Shear stress	[N/m^2]
τ_{ij}^b	Fluid stress	[N/m^2]
τ_w	Wall shear stress	[m/s^2]

1
Introduction

This chapter provides the underlying problem and the relevance of this study in the first section. Then the connected objectives and research questions are elaborated on. Ultimately the reader's guide is presented in the third section.

1.1. Problem statement and research gap

Plastics have been incorporated worldwide and designed for multiple functions, such as food packaging, security and manufacturing of goods. The material has been increasingly used over time and ranges in size, volume and composition. Since 2014, global production of plastic exceeds 300 million tonnes per year (PlasticsEurope, 2016). This number is expected to rise over the years, as due to its material properties plastic is more favourable in situations than for example glass, steel and aluminum due to its durability and low price. Their high stability and durability can be favourable for their designed use, but unfavourable in degradation of the item in the natural environment (Lebreton et al., 2017).

Plastic collection in oceanic environments has proven to be a challenge in the last years (Fairs, 2019). Furthermore, the amount of debris has been found to be increasing in the Great Pacific Garbage Patch, because of its low biodegradability and plastics only tend to break down into smaller pieces (National Geographic, 2021). The source of the accumulation of plastic material comes mostly from riverine areas. An approach where plastic can be collected from rivers can be more successful than that of plastic in the ocean due to the scale of this aquatic body. Different designs have been developed, but these have been tested on a low number of real locations and do not always achieve results that were expected.

Concentrations of plastics in Dutch rivers have been measured and efforts have been made to quantify this number in plastic fluxes (Mani et al., 2015; Van Emmerik and Schwarz, 2020). In the Netherlands it is important to retrieve plastics from rivers as the debris ends up in waste water treatment plants, in the sea and in the river banks, of which the latter is harmful for ecosystems (Michielssen et al., 2016; Kooi et al., 2018b).

In developing countries these concentrations were measured to be even higher, where areas near rivers can get flooded because of plastic debris getting stuck in the river (see Fig. A.3) (Honingh et al., 2020). The clogging of urban drainage infrastructure due to the accumulation of plastic consumer goods in waterways has produced lethal floods in the past (Hinshaw, 2015). Both in the Netherlands and in other countries the urgency in removal of debris in rivers arises on an economic level as well: a study by Deloitte (2019) found that the economic damage of the marine plastic accumulation led to $6 to 19 billion.

Plastics are the main small-scale polluters in rivers and represented more than 97 % of all small and micro particles caught in the Rhine, Po and Danube river (Van der Wal et al., 2015). The mostly applied classification of plastics is a subdivision into three classes (González et al., 2016):

- Microplastics, with a grain size of at least 1 and smaller than 5 mm;
- Mesoplastics, with a grain size larger than or equal to 5 mm and smaller than 25 mm;
- Macroplastics, with a grain size larger than or equal to 25 mm.

Microplastic concentrations in the Waal and Rhine have been assessed to be almost 900,000 microplastic particles per km^2 on average, with a peak 3.9 million particles per km^2 in a single sample by Mani et al. (2015) in the Ruhr-area in Germany in 2014. These particles spread out from Germany and travel further downstream towards the Netherlands. Macroplastics are found of different resins: assessments of bulk plastics by Vriend et al. (2020) mentions that 1.3–9.7 kg floating litter was found in the River Rhine per day for floating macrotransport. Removing residual plastics in the river is essential, because in the river itself these particles can be captured and consequently be recycled. When macroplastics are transported into the ocean, it is going to need increased effort to capture these particles due to changing currents due to the tide and wind, varying water depths in the ocean and further degradation.

More knowledge in the transportation and removal of these plastics is preferred (Conchubhair et al., 2019), with the source of plastics in the river being difficult to mitigate, as numerous sources are responsible for the plastic pollution (see Appx. A.1.2 for an overview of these sources). Research has been undertaken in determining the settling or rise velocity in laboratory conditions (Waldschläger and Schüttrumpf, 2019b) for small particles and on the pathway of films in a horizontal flume Zaat (2020) for HDPE and LDPE films.

It is unknown what the trajectories of other plastics in the water column are and more importantly, how these trajectories can be modelled for plastic particles. Modelling these trajectories enables rapid assessments for different river discharges, water depths and river beds. Ultimately , knowledge in the trajectories of plastics with different properties in the horizontal and vertical direction can help engineers in the design of plastic removal systems.

1.2. Objective and research question

This report aims to bring more knowledge into the research field on how certain plastic resins can be effectively modelled in the water column and how this material can be collected by an interception system.

The main research questions therefore reads:

How can plastics be computationally modelled and be captured by an interception system in a riverine reach for different flow velocities?

It is important to first replicate a known case before investigating trajectories of these particles with an interception system. This report views the replication and the investigation of the trajectories with an obstruction as two different objectives.

The plastic path in the water column among other things is dependent on the flow velocity and the turbulence. This path can be predicted with computational fluid dynamics (CFD) modelling. There is much to choose from the different packages and software available. The model can be tailored to find the correct solution in this way. It is therefore necessary to take enough time to come up with the appropriate tools of the CFD environment and gather knowledge of the different tools. This research simulates the characteristics of turbulent flow and is a simplification of the reality. Modelling and calibrating a flume-like system with the data of Zaat (2020) can aid the predictability of the pathway of plastics in the water column.

Equilibrium conditions in straight reaches can give a solid basis to compare different cases. This is difficult for varying flow conditions and thus only a short straight reach is defined. The desired outcome is a model that has the capability of reproducing the interception device and the pathway of plastic material, while also giving a methodology on the establishment of such a model.

Sub-questions have been set up in support of the aforementioned main research question.

a. *How can a channel with plastic debris and an interception system be modelled in a CFD programme?*

b. *What is the required distance for both turbulent flow and that of particles to reach an equilibrium profile?*

c. *How can a concentration profile similar to the results of Zaat (2020) for the plastic flux be replicated?*

d. *How can the relationship between the flow in front of an interception device and that of plastic properties be defined in terms of retention of a plastic flux?*

The answers to these questions give a prediction for large-scale systems in expected trajectories of plastic particles in flume flows, such as riverine areas. In the future this basis can aid engineers in the mitigation of plastic accumulation in oceans from rivers.

1.3. Report structure

The outline of this projects can be schematised as in Fig. 1.1. Three main pillars in this study can be described, which are the representation of flow, of plastics and the interception system of plastic debris. It is done step by step to indicate clear differences between model and theoretical results, as the representation of plastic and the interception is a niche with physical model tests in its emergence.

In this framework four different items per pillar are indicated. These are the analogies with earlier knowledge per field in the upper row, followed by the assumptions made per step. Then the distinction between steady and transient solver has been made per column in the third row and finally the phases of flow are given. Releases of very light material, neutrally buoyant material and HDPE and PP resins are simulated for the phases of the representation of plastic and the interception of it.

The context for this study is given in Chapter 2. This chapter gives information into turbulent flow, approaches made to approximate this flow behaviour with time averaged models. Consequently particle transport and the application of this for plastic transport is discussed. Available physical model tests and measurements are described here as well. Then flow near an interception system is given in the last section. The methodology is presented in Chapter 3. The results for this study are given in Chapter 4. This study is then reflected upon in Chapter 5, wherein the applicability of this project in further cases is especially described, as a simplification as presented in this study has its implications. Ultimately the research questions are answered and recommendations are given in Chapter 6.

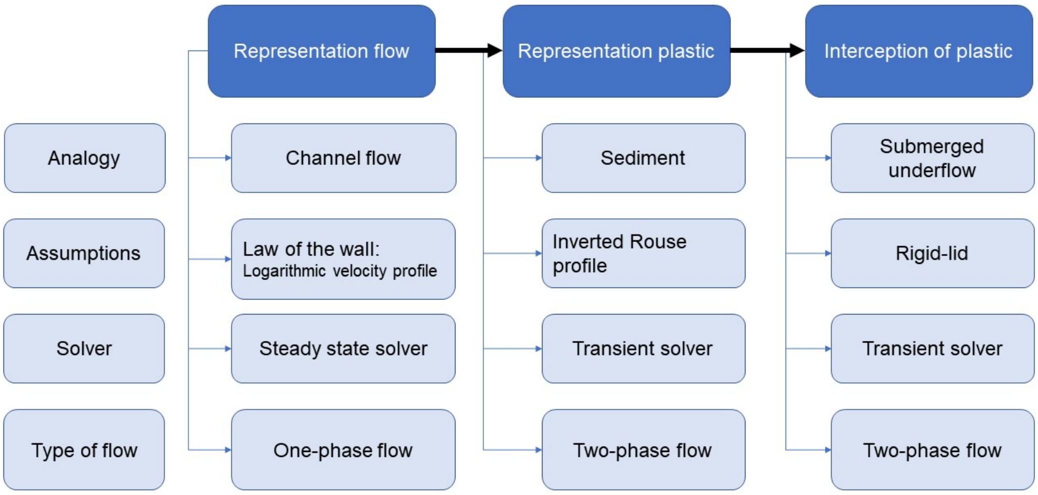


Figure 1.1: Thesis framework.

2

Background

This chapter handles the technical information necessary for this study. Literature related to the different topics is investigated and background is given on how these aspects can be modelled. Firstly, turbulent flow is discussed. Then processes related to particle and plastic transport and modelling are given. Ultimately, the state of the art of interception systems and their analogies with theory are treated.

2.1. Turbulent flow

Turbulent flow is characterised by disturbances in the horizontal and vertical and is the opposite of laminar flow, which is flow with no disruption. In this section the physical processes behind turbulent flow are described and following that the numerical interpretation of CFD modelling of turbulence is explained.

2.1.1. Physical processes

Turbulent flow is described by irregularity in flow, mixing of particles and vorticity fluctuations. Flow can be classified as turbulent with a Reynolds number larger than 4000, see Eq. A.1. Typically the velocity and pressure can be regarded as a mean and an additional component, which describes the randomness of the parameters in turbulent flow:

$$u = \overline{u} + u' \qquad\qquad p = \overline{p} + p' \tag{2.1}$$

When time averaged, the mean of the fluctuations is equal to zero. However, these fluctuations need to be considered when working with turbulent flow as these contribute to a turbulent shear stress. The mean flow is driven by a pressure gradient in the streamwise direction.

The flow velocity can be described as a profile with an increasing velocity gradient over the depth. This velocity profile is the turbulent boundary layer. The turbulent boundary layer can be divided into three different zones, the viscous sub-layer, a region that can be described by a logarithmic function and the velocity defect region, that follows a defect law because of a velocity deficit with respect to the free-stream velocity (Ting, 2016). This is further elaborated on in the next sections.

2.1.2. Bed roughness

The shape of the velocity profile at the bottom is dependent on the bed roughness. For hydrodynamically rough beds, the lower layer of this bed is defined by y_0. It is defined as the height where the extrapolated log profile is zero (Van Prooijen, 2020). The bed roughness is approximately 30 times the Nikuradse roughness height for rough beds. This is defined as the diameter of the grains for a completely flat bed and uniform spheres or a representative grain diameter.

In CFD models the value for the Nikuradse roughness height can be manually set and is related to the evaluations of the velocity distribution, shear stress and erosion depth. Nikuradse (1933) found an expression for this from measuring the velocity profile and pressure drop in smooth and rough pipes, see Eq. 2.2:

$$\frac{1}{\sqrt{c_f}} = \frac{1}{\kappa}\left(\ln(\frac{h}{y_0}) - 1\right), \text{ with } k_s = 30 \cdot y_0 \tag{2.2}$$

Another expression for the roughness height is written from the Chézy number:

$$C = 18\log_{10}(\frac{12h}{k_s}) \quad , \quad C = \sqrt{g/c_f} \tag{2.3}$$

The relationship between the non dimensionless velocity and the Nikuradse bed roughness for fully rough beds ($u_* k_s/\nu > 70$) can be described as in Eq. 2.4. In the next sections the non-dimensional velocity is explained with the use of defining the logarithmic velocity distribution.

$$U^+ = \frac{1}{\kappa}\ln\frac{y}{k_s} + B_k \tag{2.4}$$

2.1.3. Logarithmic velocity distribution

For the layer above the flow can be approximated by a logarithmic velocity distribution. The law of the wall has been developed for this region, which describes the flow over the depth sufficiently (Jackson, 1981), see Eq. 2.5. This flow velocity profile is commonly solved with turbulence models as the flow field is complex to solve.

$$u = \frac{u_*}{\kappa}\log(\frac{y}{y_0}) \tag{2.5}$$

2.1.4. Turbulent flow modelling

A common approach to model incompressible turbulent flow is by applying Reynolds-averaged Navier–Stokes (RANS) equations. The fluctuating component of the velocity is the Reynolds stress, which accounts for the effects of the turbulent fluctuations in the fluid momentum balance.

One of the mostly applied turbulence models is the $k-\epsilon$ model (Lee and Wahab, 2019). The transport variables k represents the turbulent kinetic energy and ϵ the turbulent dissipation rate. The Boussinesq approximation can be applied for these models, which represents the momentum transfer by an eddy viscosity. The $k-\epsilon$ turbulence model is a two equation model, as two extra transport equations to represent the turbulent properties of the flow.

The two equation $k-\omega$ turbulence model is frequently used as well. This is best used for near-wall treatment (Menter, 1992). A combination of both models is the $k-\omega$ (Shear Stress Transport) SST turbulence model. This model applies the $k-\omega$ turbulence near-wall and the $k-\epsilon$ in the field.

The largest scales can also be solved directly with the Large Eddy Simulation (LES). This ignores the small field perturbations as these are computationally the most expensive to solve. The flow is not regarded as steady anymore and needs smaller time steps. It offers increased fidelity, but also needs increased mesh resolution to capture all these details directly (Ideal Simulations, n.d.).

Direct numerical simulation (DNS) is computationally the most expensive and solves both the large and small scale process in the field, until the smallest turbulence scales, the Kolmogorov length scales. The necessary mesh resolution and time steps for these simulations are estimated to be approximately the cube of the Reynolds number (Ideal Simulations, n.d.).

2.1.5. Wall functions

At the area close to the wall, around 20% of the grid, the near-wall flow is typically approached with a wall function (Liu, 2016b). Relevant parameters for the near-wall region are defined in the equations given in Eq. 2.6.The value for y^+ defines the layer for the different flow regimes.

$$y^+ = \frac{y \cdot u_*}{\nu} \qquad\qquad u_* = \sqrt{\frac{\tau_w}{\rho}} \tag{2.6}$$

The dimensionless velocity connected to the dimensionless distance to the wall is given by:

$$u^+ = \frac{u}{u_T} \tag{2.7}$$

The near-wall region can be subdivided into four regions, as can be seen in Fig. 2.1:

- Viscous sub-layer, where viscous stress dominates the flow, with $y^+ < 5$ and the dimensionless velocity given by:
$$u^+ = y^+ \tag{2.8}$$
- Buffer layer, where viscous and turbulent stresses are of similar magnitude. In the buffer layer the model approaches the flow with one of the two aforementioned relations, with the first cell centre located in the buffer layer (Liu, 2016a; OpenCFD Ltd, 2018).
- Log-law layer, where turbulent stress dominates the flow. This is for $30 < y^+ < 200$, given by:
$$u^+ = \frac{1}{\kappa} \cdot \ln y^+ + B \tag{2.9}$$
- Defect layer, the layer close to the water surface the velocity profile deviates slightly from the log law. This is defined by the law of the wake (Coles, 1956). The name comes from the observed deviation which has a wake-like shape relative to the free stream .This is especially seen in flow with a pressure profile and is constrained primarily by inertia rather than by viscosity (Apsley, 2009).

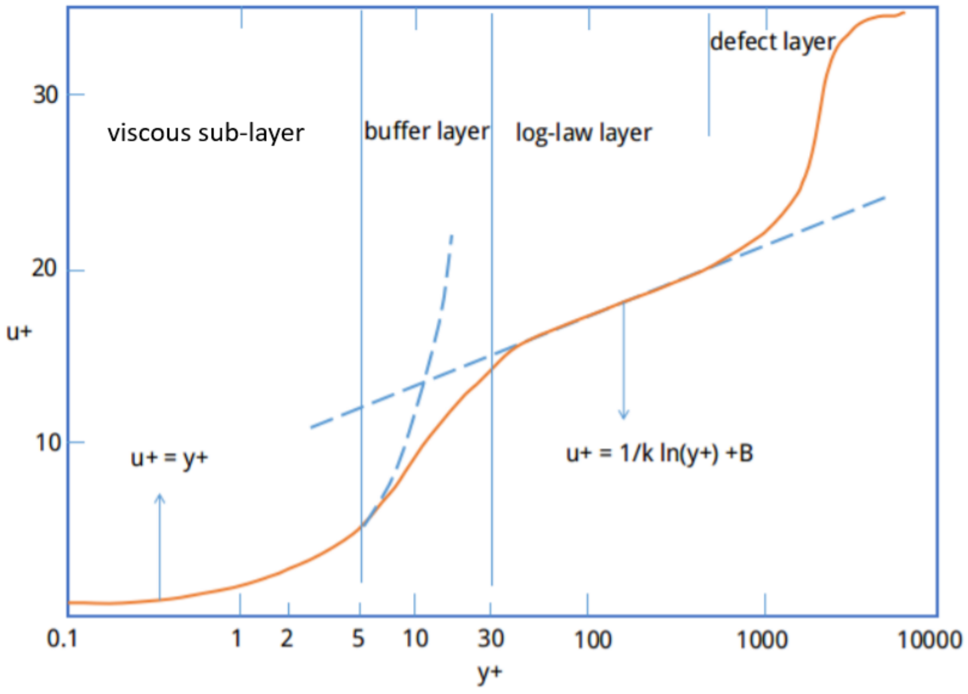


Figure 2.1: Regions with different flow regimes, edited from Liu (2016b).

2.1.6. Entrance length

When modelling turbulent flow, boundary conditions need to be imposed. The flow velocity usually needs to develop to an equilibrium profile from this imposed velocity profile at the inlet. This distance is called the entrance length Le, see Fig. 2.2. Typically the velocity gradients in the entrance and near the channel bed are high. This is because of the fact that the boundary layer increases over a certain length (Bonakdari et al., 2014).

No consensus has been reached what this exact entrance length needs to be. The developing turbulent flow in an open channel is influenced by the secondary currents and free surface effects and it is a big challenge to come to an analytical solution. Previous research was mostly based on the water depth. A collection was made by Bonakdari et al. (2014) to see what the advice was for these entrance length in previous research, see Tab. 2.1.

Bonakdari et al. (2014) also discovered that other parameters also influenced the entrance, or establishment length. This length decreases for increasing channel roughness and is independent on the Froude number. When the boundary layer is still growing, no equilibrium conditions will be encountered, which can also be seen in Fig. 2.2.

Table 2.1: Advice of entrance length for model. Edited from Bonakdari et al. (2014).

Reference	**Type of channel**	**Entrance length over depth** Le/h [-]
Grass (1971)	Smooth and rough	130
Dean (1978)	Smooth	55
Nezu and Rodi (1985)	Smooth	60-90
Cordaso et al. (1989)	Smooth	69-108
Graf (1991)	Rough	140
Ranga Raju et al. (2000)	Smooth and rough	50-100
Zanoun et al. (2003)	Smooth	115
Lien et al. (2004)	Smooth	130

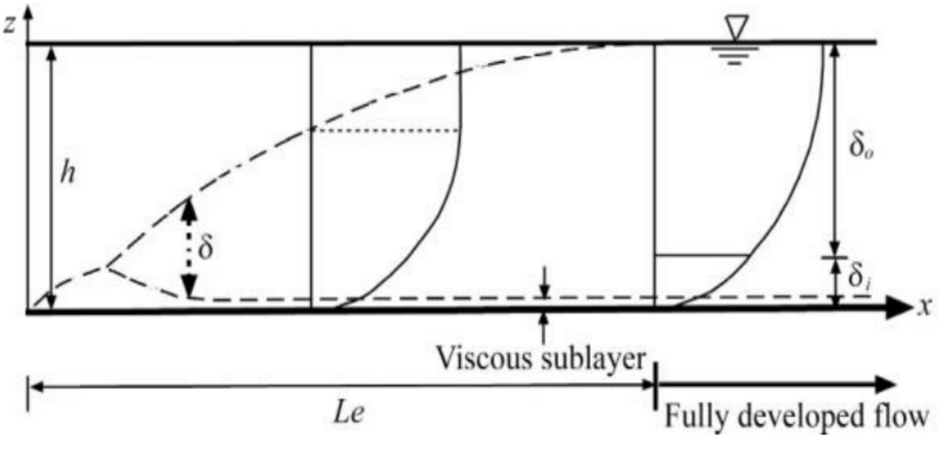


Figure 2.2: Development of the boundary layer (Bonakdari et al., 2014).

2.2. Particle transport

In this section the processes for particles in suspension are described for spherical particles. This is then compared with current knowledge into plastic transport from physical tests and computational models.

2.2.1. Bed load and suspended load

River sediment can be distinguished in two categories, as bed load and suspended load. Bed load consists of the larger sediment and is situated at the river bed.

Suspended load is sediment that is transported in the water column. Typically this is finer material with less energy needed to transport these particles. It can be said that the flow speed of the river at the given water depth is the same velocity of the travelling particle.

2.2.2. Settling

Three forces act on a particle for very small Reynolds numbers, which are the drag, gravity and buoyancy force. Expressions for the forces for a spherical object are found in Appx. A.4.1. For spherical particles, which is equal to a shape factor of 1, the rise and settling velocity can be derived from the balance between these forces. Settling occurs when the force induced by gravity is larger than the upward forcing.

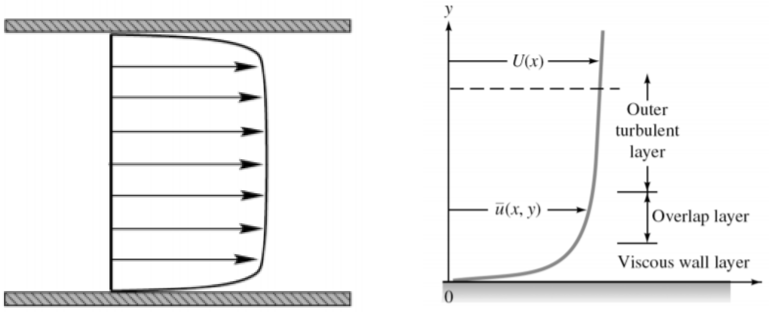
When assuming a perfectly spherical plastic particle, then the particle behaves as a Stokes particle. The particle settling velocity is then equal to the Stokes settling velocity and is given as Stokes' law. This finds application in several areas, particularly with regard to the settling of sediment in fresh water and to measurements of the viscosity of fluids. The equation for the Stokes' settling velocity is given in Eq. 2.10:

$$v_{Stokes} = \frac{2}{9} R_s^2 \frac{\rho_p - \rho_w}{\eta \cdot g} \tag{2.10}$$

Drag consists of form drag and friction drag. The form drag relates to 1/3 of the drag for form drag and 2/3 for the friction drag for Stokes particles and for low Reynolds numbers. Form drag comes from the separation of the boundary layer from a surface and the wake created by this separation. (Skybrary, 2017) It is primarily dependent upon the shape of the object. The friction drag comes from the fact that the fluid experiences friction against the surface of a particle.

2.2.3. Turbulent dispersion

Movement of particles through the water column is enhanced by turbulence. The transportation of a particle is dependent on the velocity profile, which mostly is a turbulent velocity profile in Dutch rivers throughout the year (Uijttewaal, 2014). The flow velocity profiles for a straight channel are given in Fig. 2.3, over the horizontal Section 2.2.3 and vertical profile Section 2.2.3. The spreading of the particles over the depth occurs randomly because of these turbulent motions.



(a) Horizontal profile.

(b) Vertical profile.

Figure 2.3: Velocity profiles of a straight channel with turbulent flow. Edited from Kudela (n.d)

The turbulent diffusion of particles is dependent on the turbulent eddy viscosity. This can be defined as Eq. 2.11 from the law of the wall. The particle distribution can be expressed from this equation under certain conditions, which is the Rouse profile.

$$\nu_t = \kappa \rho u_* y \left(1 - \frac{y}{h}\right) \tag{2.11}$$

2.2.4. Rouse profile

The sediment distribution over the water column can be approximated by an empirical formulation. This is the and is dependent on several assumptions:

- There is a steady flow in a uniform channel;
- The particles have a constant and uniform size and density;
- The resulting shape of the eddy viscosity profile is parabolic;

The Rouse number is dependent on the Schmidt number, which relates the turbulent eddy viscosity number over the eddy diffusivity number, Eq. 2.12:

$$\sigma_T = \frac{\nu_t}{\epsilon_T} \tag{2.12}$$

The eddy diffusivity can be expressed as the combination of Eq. 2.11 and Eq. 2.12:

$$\epsilon_T = \frac{\kappa \rho u_*}{\sigma_T} y \left(1 - \frac{y}{h}\right) \tag{2.13}$$

The sediment balance under the aforementioned assumptions is given in Eq. 2.14. From this the expression for the Rouse profile can be found.

$$-\frac{\partial}{\partial y}(w_s c) - \frac{\partial}{\partial y}\left(\varepsilon_{T,y} \frac{\partial c}{\partial y}\right) = 0 \tag{2.14}$$

Integration and a boundary at the height y=h and constant=0 yields the expression as in Eq. 2.15:

$$w_s c = -\epsilon_{T,y} \frac{dc}{dy} \tag{2.15}$$

The expression for the Rouse profile can be found by expressing the eddy diffusivity according to Eq. 2.13 and integration to c and y at the left hand side (LHS) and right hand side (RHS). The sediment

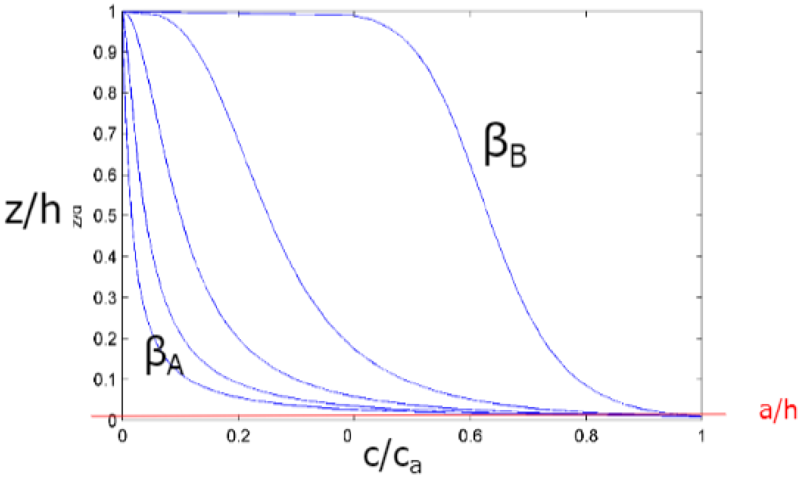


Figure 2.4: Typical concentrations for varying values of the Rouse number. Note that $\beta_a > \beta_b$. From Van Prooijen (2020)

concentration c_a at point a is derived from an integration constant and the resulting equation for the Rouse profile for sediments is given in Eq. 2.16 and the Rouse number β in Eq. 2.17:

$$c = c_a \left(\frac{a}{h-a} \frac{h-y}{y} \right)^{\beta} \tag{2.16}$$

$$\beta = \frac{\sigma_T w_s}{\kappa u_*} \tag{2.17}$$

The Rouse number indicates whether the particle distribution is close to the bed for high values of β and distributed nearer to the surface for lower values of β. This variability in concentration with respect to the Rouse number is given in Fig. 2.4 and from the variables of Eq. 2.17 it can be quickly said when the Rouse profile shows a trend downward or upward.

From these concentration profiles it is possible to determine the settling velocity. The eddy diffusivity profile actually is not completely parabolic, but runs uniform near the surface and thus the mixing is underestimated with the Rouse profile and thus the near surface concentration (Van Prooijen, 2020). The Rouse profile is still considered to be a very accurate approach to the problem.

2.2.5. Plastic found in water bodies

Studies on rapid plastic waste assessment in rivers covered mostly the resins PET, $\mathrm{PO_{soft}}$, PP and HDPE (Van der Wal et al., 2015; Vriend et al., 2020). Rise and settling velocities for these particles have not been studied extensively, but studies on the rise or settling velocity have been undertaken by Reisser et al. (2015) and Zaat (2020). Reisser et al. (2015) found that the median rise velocity near the water surface was 0.0125 m/s for plastics of a few centimeters.

Van Emmerik et al. (2019) stated that in the upper part of the water column, to 0.4 m depth at a total depth of 1.0 m, registered a concentration of 5 times as high as in the lower layer in Indonesia. Mani et al. (2015) stated that in the Rhine measured data at different water levels in the Rhine and stated that the focus towards the water surface when calculating daily freight discharges was justified, with no measurable plastic at a higher water depth. A focus to retain particles near surface therefore is desired.

2.2.6. Concentration profiles of plastic material

While the Rouse profile in Section 2.2.4 is for sediment, this profile can be inverted for rising particles. The concentration profiles over the depth can be approximated by a line similar to an inverted Rouse profile that for positively buoyant plastics (Reisser et al., 2015; Sundby, 1983; Zaat, 2020). Similarly to the settling velocity, the rise velocity can be defined as in Eq. 2.18:

$$w_r = \sqrt{\frac{2g \cdot (\rho_f - \rho_p) \cdot V_p}{C_D \cdot \rho_f \cdot A_p}} \tag{2.18}$$

The spreading of plastic films in a flume at Delft University of Technology was assessed by Zaat (2020). The dimensions of this flume are a depth x width x length of 0.25 m x 0.40 m x 15 m. The release was performed for 100 plastic films at 7 different depths, with a setup of 6.7 m between the release and measurement of the plastic film. The dimensions of the plastic films were 30x40 mm^2 and a few mm thick. These were scaled from plastic bags, typically found in riverine areas.

Tests were performed for different propagation velocities in fully developed, uniform flow. The flow was turbulent and the flow velocity was assumed as a depth averaged velocity profile. The results for the different tests performed by Zaat (2020) are visualised in Fig. 2.5. The particles are mostly abundant near the water surface, as is expected for positively buoyant particles and the results can be roughly described with a Rouse profile times a form factor $\alpha_{film} = 0.64$.

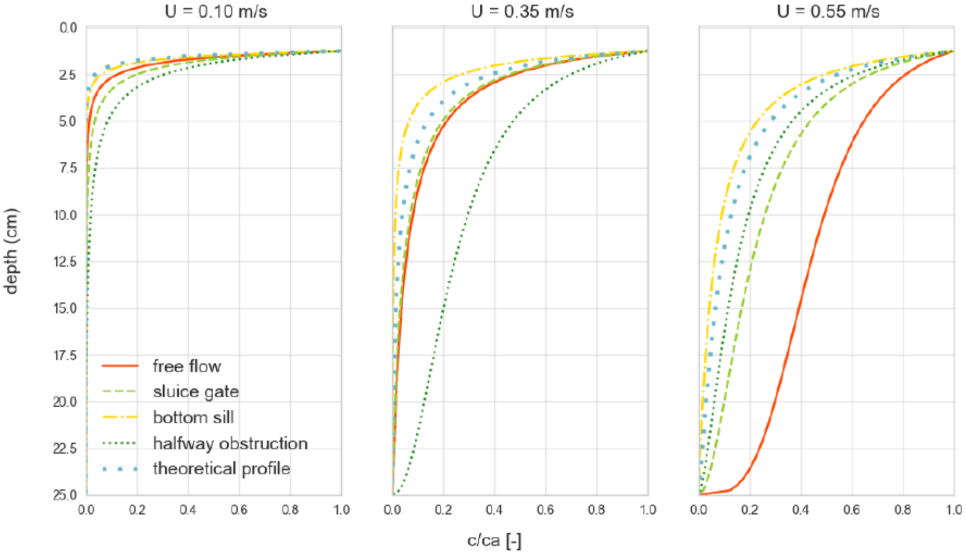


Figure 2.5: Concentration distribution profiles normalised to the near surface measurement C_a for three velocities. The three obstructions are compared with the undisturbed distribution, from Zaat (2020).

Obstructions in the water column were also tested. A solid gate of 1/5 of the water depth was tested just below the water surface, as well as a bottom sill and a obstruction half way the depth of the same dimension. It showed that plastic films were captured, but eventually that the plastic passed underneath the gate when these where in abundance adjacent to the gate. Large uncertainties are connected to this study, as only a rapid assessment for these particles was performed.

2.2.7. Particle modelling in liquid flow

Particles in turbulent flow can be modelled with a two-phase model. There are different modelling approaches for two-phase modelling, which are DNS, Lagrangian particle tracking and Euler-Euler modelling. The non-applicability of DNS was discussed in Section 2.1.4. The Lagrangian-Eulerian and Euler-Euler model are the most commonly applied modelling approaches on macroscale (Rusche, 2003).

Lagrangian particle tracking consists of an Eulerian description of the fluid phase and solves the trajectories of the particles in a Lagrangian framework. An advantage of this is that for each particle the movement is accounted for in the equation of motion explicitly for the motion of each particle, meaning that explicit particle properties and flow vectors can be specified. A disadvantage is that each particle needs to be solved individually for the dispersed phase, which can be of large computational effort for high concentrations.

In the Euler-Euler model both the particle and the fluid phase are described using Eulerian conservation equations. A separate continuity and mass conservation equation is solved for both phases. Per cell these equations are solved and the phases share the same pressure field. The momentum equations for phase l and p, for no mass transfer is given in Eq. B.18. Here, l represents the liquid or fluid phase and p the particle or solid phase.

Both phases are described as interpenetrating continua and because of this share the momentum and continuity equations (Busch, 2015). Particle-fluid interactions can be describes as dispersed-continuous phase interactions, meaning that every cell can take any value of the volume fraction between 0 and 1 for both phases, as long as the total of both phases is 1. The other option available is continuous-continuous, meaning that a cell is restricted to a volume fraction of 0 or 1. The plastic particles can be seen as dispersed flow, as this consists of the smallest concentration and therefore spreads in the larger continuum, which is the fluid phase.

2.2.8. Interphase momentum exchange

The Reynolds averaging process introduces the phase fraction a into the equation set, which is defined as the probability that a certain phase is present at a certain point in space and time. The averaging process causes loss of information with additional terms in the averaged momentum equation for each phase. Besides the Reynolds stresses another term arises, which is the interphase momentum transfer term. Because the momentum equation for both phases is solved in every cell, the velocity of both particles is different due to drag and interphase forces, which creates an interphase momentum transfer (Wimshurst, 2019). The interphase drag is considered as the primary mechanism in the momentum transfer, because this reflects the two-phase flow effects for both the dispersed and continuous phase in determining the flow fields (Akimi, 2010).

The interphase momentum transfer term is given by the sum of different forces on the particles, which is given by the momentum exchange terms due to the drag force and the virtual mass force, the lift force, the wall lubrication force and the turbulent dispersion force respectively, see Eq. 2.19 (Kumar et al., 2018). The drag force and the virtual mass force are considered to have the largest influence on liquid-solid flow (Syamlal et al., 1993).

$$\mathbf{M}_a = \mathbf{M}_a^d + \mathbf{M}_a^{vm} + \mathbf{M}_a^l + \mathbf{M}_a^{wl} + \mathbf{M}_a^{td} \tag{2.19}$$

Where:

- M_a^d is the drag force;
- M_a^{vm} is the virtual mass force;
- M_a^l is the lift force;
- M_a^{wl} is the wall lubrication force;
- M_a^{td} is the turbulent dispersion force.

The drag force specifies the relative velocity between the phases and acts opposite to the relative motion of the particles moving with respect to the surrounding fluid. The magnitude of the drag force exerted on a body moving in a fluid depends on the velocity of the body relative to the medium, the viscosity and density of the medium, the shape and cross-sectional area of the body, and the roughness of its surface (Peker et al., 2008).

If a particle or bubble is transported in a fluid field, the near field also needs to be accelerated. This can be seen as the particle having added or virtual mass. Research has been undertaken into the effect of virtual mass coefficient and when the virtual mass coefficient is set as 0, this effect is not simulated and the removal of the virtual mass force generated instabilities at the inlet (Ghione, 2012). The effect due to virtual mass appears only in unsteady flows. Cheng et al. (2017) advised a virtual mass coefficient of 0.5.

The lift force or Magnus force is a force that acts upon a rotating particle when it is transported through the fluid. Due to the spinning of the particle a deflection in perpendicular direction to the spinning axis and the velocity vector is generated.

The wall lubrication force can be included for liquid-gas bubbly flows. The wall lubrication force pushes the dispersed phases away from walls. The effect is that the dispersed phase is concentrated in a region near, but not immediately adjacent to, the wall (Ansys, 2009).

Turbulent dispersion is the effect where a constant flux of particles moving through a turbulent fluid spreads out. Two mechanisms govern this particle dispersion, related to the turbulent carrier flow and to its particle properties (Huilier, 2021). It should not be confused with molecular diffusion, which is the phenomenon where particles move from a high to low concentration caused by Brownian motion on microscopic scale. The turbulent dispersion force should be considered when turbulent eddies are larger than the particle size simulated (Kasat et al., 2008). With additional turbulent dispersion forces added, this effect can be resolved. This effect is usually added as a flux of dispersed material in the continuity equation. The Burns drag model replicates this effect well and is shown to yield superior predictions compared to other turbulent dispersion models (Burns et al., 2004).

2.2.9. Plastic modelling

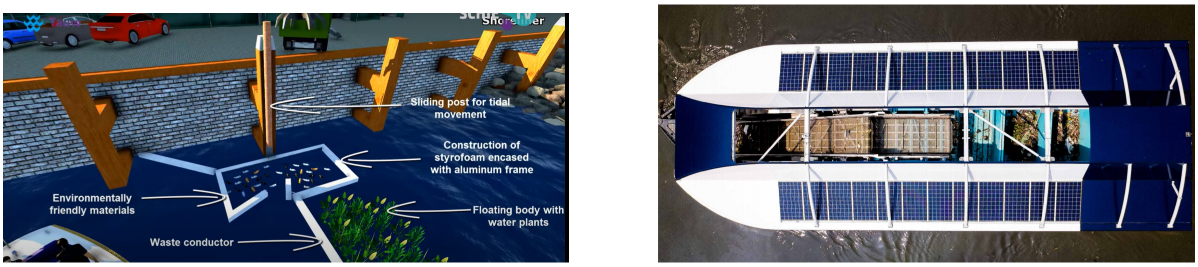
Efforts have been made to describe the plastic trajectories in rivers and channels (Van Welsenes, 2019) and in oceanic environments in Delft3D (Van Utenhove, 2019).

Van Welsenes (2019) quantified the effects that drive the horizontal transport of plastics at the water surface. He found that the wind, water current and gravitational forces are the most significant factors that influence transportation in the horizontal and possibly towards the river bank. Further findings were related to river structures. A river confluence increases the concentration along the tributary side of the channel, whereas a groyne increases the concentration at the side of the river where there is no groyne.

In coastal waters, wind effects can even have a larger influence on the distribution of plastics (Van Utenhove, 2019) and further increase the difficulty to capture plastic debris. An interception device in the river is therefore deemed as a more suitable approach to this problem.

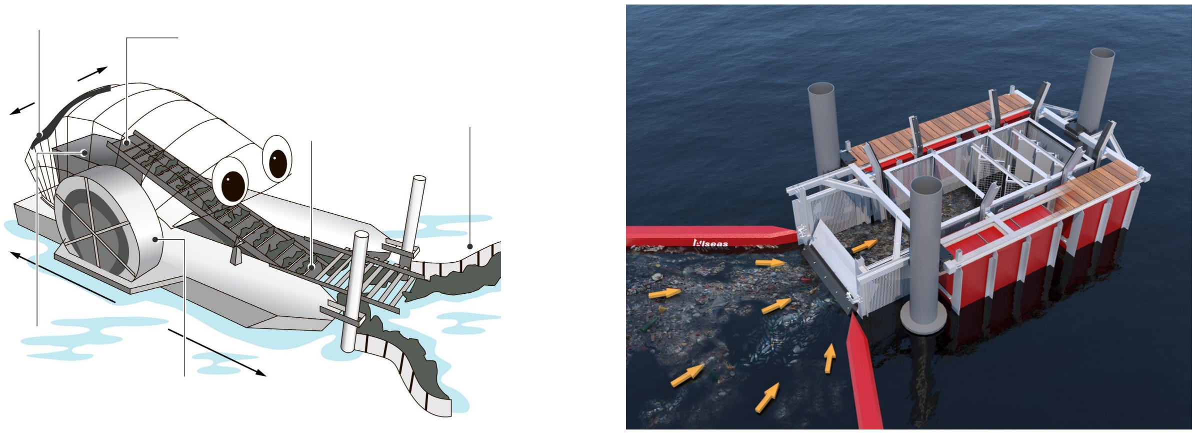
2.3. Interception system

Interception systems can be divided into different categories that follow the same mechanisms. There are active systems that follow or move themselves to capture plastics and passive systems, meaning that the systems are put at a certain location and guide the debris to an extraction point. The latter category is investigated in this research. Examples are the Interceptor developed by the Ocean Cleanup, the Shoreliner by Tauw and the Litterboom project by Obscape, see Fig. 2.6. These systems have been tested at several locations and are scalable for different situations (Tauw, 2019; The Ocean Cleanup, 2020; Obscape, n.d.). Placing clean-up infrastructure in areas with a constant influx could significantly decrease the amount of plastic litter entering the open ocean (Van Emmerik and Schwarz, 2020).



(a) The Shoreliner, from Tauw (2017).

(b) The Interceptor, from The Ocean Cleanup (2020).



(c) Mr. Trash Wheel, from Calvert (2019).

(d) Catchy, from Allseas (2021).

Figure 2.6: Different interception systems.

2.3.1. Principle of operation

The systems in Fig. 2.6 can be schematised as passive systems, meaning that debris is guided without the input of energy by the system. The systems float on the water and consist of an arm or barrier that over time guide the plastic material in a retention system. The floating barriers are usually placed diagonally with respect to the flow.

The Shoreliner can be placed in such a way that the wind and current transports the plastic debris towards the system and needs no energy itself to work. The debris in the Shoreliner needs to be removed by a mechanical arm periodically. The waste gets sorted and the plastic recycled when removed (Tauw, 2019). The Interceptor guides debris in rivers with two different guidance arms. Then a conveyor transports the garbage into garbage bins. These need to be periodically emptied. All electronics on the system are solar powered.

The Trash wheel and Catchy consist of a two booms as well, with a collection tray behind these. The Trash Wheel was designed for the city of Baltimore, USA. It has collected more than five hundred thousand kg of trash (Campbell, 2016). No news has been published on the device designed by Allseas, Catchy.

2.3.2. Analogies with theory

Current insights into the effect of the flow of the placement of a plastic interception system is limited. However, the flow under objects or submerged sluice gates can serve as reference cases. The effect of these components on the flow is well known, with physical and computational tests applied. Tests with an opening of roughly halfway the depth ($h/a \approx 1.9$), with a being the gate depth under the gate, similar to the physical and computational tests applied by Finnie and Jeppson (1991) and computational efforts by Shammaa et al. (2005) and Montes (1997) can serve as a reference to verify the hydraulics around the system.

The measuring techniques of the flow in that time were not the most sophisticated meters, but no more accurate meters were available and that the placement of the probes proved to be a challenge. A finite element method was applied with a $q - r$ variable substitution by Finnie and Jeppson (1991). This substitution uses the same assumptions as the $k - \epsilon$ model, which assumes isotropic flow. The results can also be verified with similar $k - \epsilon$ model setups applied in later studies by Montes (1997) and Shammaa et al. (2005). The different results can be seen in Fig. 2.7. It is clear that contraction occurs for this experiment, with an increase of horizontal flow under the sluice gate. The vertical flow velocity increases by a factor 10 halfway the water depth and could cause plastic debris to move underneath an interception device.

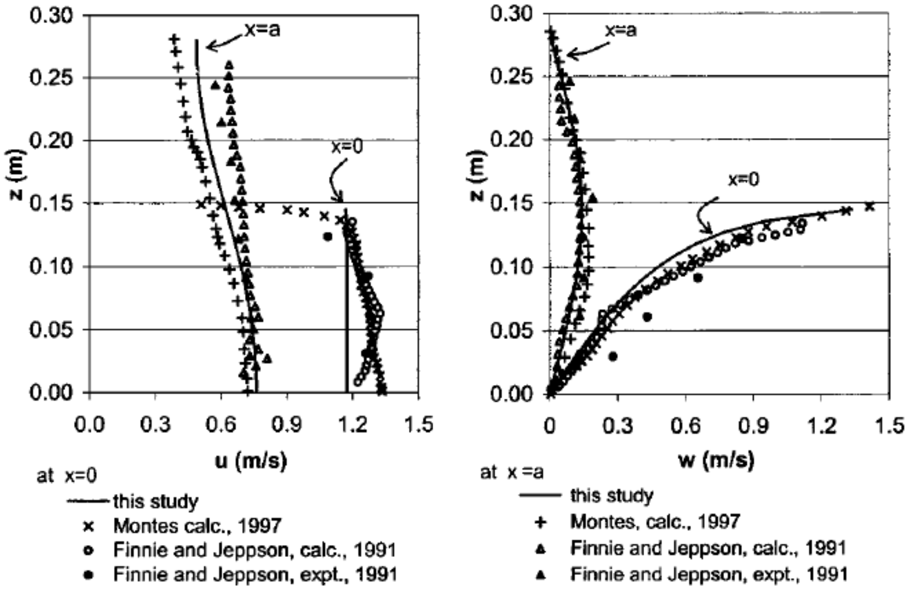


Figure 2.7: Flow velocity under and upstream of a sluice gate with h= 0.295 m and h/a = 1.9, from Shammaa et al. (2005). x=0 describes the profiles just under the gate, with a=0.15 m upstream of the gate. Note that the positive velocities in the right figure are orientated downward.

2.3.3. Research on plastic retention

The threshold whether a plastic particle is retrieved by the system is approximated with the observed flow fields with the tests of Hoogevorst (2019). He proposed that the specific streamline where the plastic debris is situated determines whether the plastic is retained by the system. Flow in front of the structure contracts and a recirculation zone is generated just in front of the system. Dependent on the opening and on the flow velocity, a downward velocity is generated in front of the structure.

Carleton and Nielsen (1990) compared the use of trash racks and that of floating arms, with booms being similar to the systems in Fig. 2.6. The trash rack was placed over the full depth. It was found that the booms were more effective in the retention of smaller plastic debris, which would flow through the trash racks. The rack was better in capturing partially or fully submerged litter. These were drawn under booms with the flow and escaped the interception mechanism.

During dry weather flow the floating boom showed a larger effectivity in retention of plastic debris than that for higher flow. For high flow, the debris would tend to flow underneath the debris.

3

Methodology

The methodology is linked to the literature review and context in the previous chapter and to the three segments in Fig. 1.1. These three segments are further elaborated in this chapter. These stages are described in the next section. Then, the data available to verify this study is briefly explained in Section 3.2. Section 3.3 describes the processes handled by the model and the computational solvers set.

Next, the general model setup for the single-phase flow is described in Section 3.4. The single-phase flow provides a validation on the hydrodynamics for the flow velocity profile and the turbulent eddy viscosity profile. The particle dynamics needed to be verified as well. The process in finding the accurate solver and model coefficients to ultimately replicate plastic particles and film dynamics is given in Section 3.5.

In Section 3.6 a set of boundary conditions needed to be added to represent the interception device. The flow pattern around the interception system was expected to be crucial for the retention of the particles and was analysed. Then neutrally buoyant particles, particles representing very light debris and HDPE and PP films were released in the model with the interception system placed. Ultimately, the retention of the plastics by this device was assessed.

3.1. Stages

Three different cases were tested with steady state flow per modelling stage. These cases are a low turbulent flow with an inlet flow velocity of 0.10 m/s, an inlet flow roughly similar to mean river flow conditions of 0.55 m/s in the Rhine (Van der Wal et al., 2015) and a high flow velocity case of 0.90 m/s. The high flow velocity case can indicate how much debris is retained with a higher value for the turbulent kinetic energy and thus a higher turbulent eddy viscosity profile. In reality plastic debris that is at the bottom or at the river bank can get stirred up into suspension for conditions with higher flow velocities, which enables the possibility of retaining these particles.

The simulations for the three different stage of Tab. 3.1. Per stage the model results were used in later phases to compare the single-phase flow results with the two-phase flow with and without the interception device.

The two-phase flow resembles the plastic flow in a flume. The tools applied, the release of the different particles and the outcome of these steps are summarised in Tab. 3.1. These stages are further explained in the next sections.

Table 3.1: Different stages in this study.

Stage	Single-phase flow	Two-phase flow	Retention of plastic
Tools	Law of the wall Parabolic eddy viscosity	Inverted Rouse profile Flume experiment (Zaat, 2020)	Spin-up time Concentration profiles
Release of particles	-	Light particles Neutrally buoyant particles HDPE films PP films	Light particles Neutrally buoyant particles HDPE films PP films
Outcome	Entrance length Spin-up time	Entrance length Spin-up time Model coefficients Concentration profiles	Retention of plastic material

3.1.1. Single-phase flow

The outcome of this step monitors whether the boundary conditions and the initial conditions were applied in such a way that the results are in line with theory on turbulent flow. The flow field was compared with the log-law development of the velocity. The resulting turbulent eddy viscosity profile from the turbulent kinetic energy and turbulent energy dissipation were compared with parabolic profile over the depth from analytical results.

3.1.2. Two-phase flow

The resulting concentration profiles over the vertical were compared with the flume test results found by Zaat (2020) and the inverted Rouse profile for the flow conditions. This step served as a calibration of the particle concentration profile and indicated where discrepancies lie. The model was calibrated with the use of interphase momentum transfer coefficients. Besides the concentration profiles from these tests, the spin-up time and the entrance length to come to the equilibrium concentration profiles for the different cases were assessed.

3.1.3. Interception of plastic

The interception system was added as a square obstruction in the grid that blocks the flow of both the water and the plastic particles. An analogy can be made with flow under a weir or a partially opened submerged gate, which contracts flow in a similar way.

Both single-phase and two-phase simulations were performed. The single-phase results give insight how the flow propagates just in front of the structure and consequently how the vortex in front of the structure impacts the retention of plastic. The two-phase simulations were performed in order to find concentrations near the interception device for the different particles assessed.

3.2. Collection of data

The flume test set up by Zaat (2020) assumes an average flow velocity over the depth and imposes a constant flow velocity. In the work, a bottom friction coefficient of 0.003 was assumed, resulting in a Nikuradse roughness height of 0.004576, see Eq. 2.4.

The inverted Rouse profiles for the different rise velocities were compared with the concentration profiles at different points along the streamwise direction. Also, the profiles found by Zaat (2020) were compared with the results found from these simulations. The goal was to find results similar to the inverted Rouse profile and the results of Zaat (2020) and served as a validation step for the model.

The release of films resulted in an inverted Rouse concentration profile times a constant, α_{film}.

3.3. Application of CFD model

This section comprises the different tools necessary to run the simulation in OpenFOAM. The choice for OpenFOAM lies in the versatility of the application: there are numerous single and multi-phase solvers available for specific problems. First the model assumptions were handled, which defines the input for the model and which forcing effects have been excluded. Then the type of solver applied is discussed. The turbulence model for the different stages is described after that. The type of multigrid solver and numerical schemes are discussed in the last subsection.

3.3.1. Model assumptions

Waves from shipping or wind input were not regarded in this study. This study applies the principles in flow only and how the flow and particles can be modelled and subsequently be retained by an interception system. These particles were impacted by the principles of the flow only.

The model is viewed as a rigid-lid model for all the different cases, meaning that an equal water depth was imposed over the domain. This generally provides a good representation of the flow velocity (Kheirkhah Gildeh et al., 2016) and is less complex compared to a free surface model. For a free surface model both air and water flows are considered and thus water level differences can be modelled (Kheirkhah Gildeh et al., 2016).

A no slip boundary condition at the bottom was assumed, which means that the velocity at the boundary condition is the same at the boundary layer. There is no relative movement between the boundary and this fluid layer, which means that there is no slip. This is usually applied in fluid parcel modelling, especially because no erosion of the bed was considered (Rapp, 2017).

3.3.2. Transient and steady state solvers

Both transient and steady state solvers can be implemented and a distinction has been made between the most applied incompressible solvers, which are the steady state SIMPLE solver, the transient PISO solver and the combined PIMPLE (PISO-SIMPLE) solver. The solvers originate from the problem that for incompressible flows, there is a pressure-velocity coupling problem, with no equation set up for the pressure. The continuity equation acts as a restriction on the computed momentum field and iterations are performed to find a solution that satisfies both the momentum and continuity equation. The entire loop for the different algorithms is indicated in Fig. B.1. More background information into these solvers is given in Appx. B.2. The single-phase modelling has been done with `simpleFoam`.

During the multi-phase stages in this study, the PIMPLE algorithm was applied. The combination of a steady state solver together with a multi-phase solver is not commonly applied: two-phase flows are intrinsically unsteady. Running transient simulations for a long time approaches a steady state solution and the PIMPLE solver provides both a stable and accurate solution for that.

The two-phase modelling is performed with `reactingTwoPhaseEulerFoam`. The solver follows a certain numerical procedure, which is specified as (Chauchat et al., 2017):

1. The particle concentration is solved;
2. The fluid concentration is updated;
3. The drag parameter is updated in the drag term;
4. The k and ϵ parameters are solved and the turbulent eddy viscosity is then calculated;
5. It is solved for the particle interphase drag and the particle stress;
6. The PIMPLE loop is solved, see Fig. B.1;
7. These steps are iterated for the iterations set or when the relative tolerance is reached and then this procedure is undertaken for the next time step.

The spin up time for the different simulations was determined with the placement of probes, see Appx. F.5.1. These can be set at different locations and per time step write the desired parameter. This can then be plotted over time to see when the final solution has been reached. At x=25 m, and near-surface, at y=0.25 m, a probe was placed to visualise the results for the pressure, flow velocity, turbulent eddy viscosity and the plastic particle concentration.

3.3.3. Commands to run simulation

In OpenFOAM the mesh was constructed with the `blockMesh` command, which makes a mesh of a predefined dictionary for the command, which is the blockMeshDict file, see Appx. F.5.2 and Appx. F.5.3. In here lies information about the vertices of the grid, as well as which blocks in the mesh to be defined, the inlet, outlet and wall locations. The number of cells need to be specified and can be graded at the location of a structure. For the two-phase model the command `setFields` needs to be performed as well. This generates the initial condition of a plastic particle concentration in a specified region. Then the simulation is run with either of the commands `simpleFoam` or `reactingTwoPhaseEulerFoam`. The shell file for the simulations is given in Appx. F.2. The time step was programmed to satisfy the Courant–Friedrichs–Lewy (CFL) condition (Eq. 3.1) for both the Interface Courant Number and the Courant Number in the field.

$$CFL = \frac{u\Delta t}{\Delta x} \tag{3.1}$$

3.3.4. Turbulence model

The $k - \epsilon$ turbulence model was applied, as it offers good convergence and is not memory-intensive, being a good general-purpose model. The $k-\epsilon$ turbulence model was deemed as to solve the behaviour in front of and behind the interception system sufficiently. The turbulence intensity is set at 10%, which is the peak value measured by Hanmaiahgari et al. (2017) for immobile beds. A turbulence dissipation length of $0.07h$ is assumed (COMSOL, 2020; Versteeg and Malalasekera, 1995).

The equations for the k and ϵ values are stated in the User Manual (OpenCFD Ltd, 2018). These equations are listed in Eq. B.1 for the turbulent kinetic energy dissipation rate and for the turbulent kinetic energy in Eq. B.2, respectively. The model constants for the $k - \epsilon$ model are listed in Tab. B.1.

The inlet flow parameters for this turbulence model are listed in Tab. 3.2. The corresponding values for the flow parameters for the different cases were determined from Eq. 3.2 (OpenCFD Ltd, 2018):

$$k = \frac{3}{2}\left(I|u|\right)^2, \qquad \epsilon = \frac{C_\mu^{0.75} k^{1.5}}{L}, \qquad \nu_t = C_\mu \frac{k^2}{\epsilon} \tag{3.2}$$

Table 3.2: Flow parameters for the k-epsilon turbulence models, for a turbulence intensity $I = 10\%$

Case	**Low flow**	**Medium flow**	**High flow**
Inlet flow velocity	U_x **= 0.10 m/s**	U_x **= 0.55 m/s**	U_x **= 0.90 m/s**
k [m^2/s^2]	1.500e-5	4.538e-4	1.215e-3
ϵ [m^2/s^3]	1.725e-5	2.870e-3	1.258e-2
ν_t [m^2/s^1]	1.174e-4	6.457e-4	1.057e-3

3.3.5. Multigrid solver

OpenFOAM applies a collocated grid: fluid dynamic quantities are stored at a single point within a control volume, which differs from a staggered grid, which defines volume based quantities at a single point in the centre, and flux based quantities on the faces. Most modern codes apply the collocated storage (CFD Online, 2012).

The equation solvers, tolerances and algorithms are given in fvSolution, see Appx. F.5.6. The multigrid solver applied is the GAMG solver for the hydrostatic pressure. This coarsens the grid in steps and guesses an accurate solution on this mesh. Then this guess is performed on the original mesh Cristo (2021). The other parameters were solved with the smoothSolver. According to (OpenFOAM, 2006) this solver is: “an iterative solver for symmetric and asymmetric matrices which uses a run-time selected smoother”.

3.3.6. Numerical schemes

OpenFOAM handles a Finite Volume Method (FVM) to solve the governing equations in OpenFOAM. This method subdivides the flow domain into a finite number of control volumes. The method uses the integral forms of the transport equations set as a starting point. The integration of these equations is then performed and this results in a finite number of linear equations which can be solved when translated into matrices (OpenCFD Ltd, 2018).

The different numerical schemes are given in Appx. F.5.5. The time scheme applied is the Euler scheme. This scheme provides a good compromise between accuracy and robustness.

The divergence terms were solved with the Gauss upwind, linear and limited linear schemes. The First Order Upwind scheme can be applied in any case and has a somewhat low accuracy (OpenFOAM, 2006). The focus lies on the applicability of a stable system and thus the scheme was decided to be acceptable in these cases assessed. The Gauss entry applies the standard finite volume discretisation of Gaussian integration. An application of a linear entry means that an interpolation of values from cell centres to face centres is performed. The limited linear scheme applies a limiting coefficient of 1, meaning that boundedness is guaranteed.

The Laplacian terms were solved with the Gauss linear scheme. The extra setting is that it is corrected, which is unbounded, second order and conservative. This means that it yields a solution that obeys physical conservation laws. The interpolation scheme is set as linear (OpenCFD Ltd, 2018)[C4.4, 6.2].

3.4. Single-phase model setup

The single-phase model tests the hydrodynamics of the modelled segment. The boundary conditions grid size are first given in this section.

Two different sensitivity analyses were performed. One study is tested for the domain of 15 m and 30 m. These simulations indicate the entrance length necessary to come to equilibrium conditions for the flow velocity and turbulent eddy viscosity. Another sensitivity analysis was performed applying a fine and coarse mesh. The impact of the mesh size on the flow velocity was assessed with this study.

In Section 4.1 the resulting hydrodynamics of these simulations are given. These were checked with the analytical results from theory.

3.4.1. Boundary conditions

The boundary conditions are given in Tab. 3.3 and the lines of OpenFOAM code in Appx. F. A uniform and constant bed was assumed, which is indicated by $C_s = 0.5$ in Tab. 3.3. The pressure boundaries set in OpenFOAM are specified after dividing the pressure by the density of the fluid. This means that for single-phase flow the results for the flow velocity field are the same in incompressible flow, as the fluid properties are independent on the pressure values set. This does not hold for the two-phase modelling.

Table 3.3: Boundary conditions for the different cases.

Parameter	**Inlet**	**Outlet**	**Surface**	**Bottom**
u [m/s]	fixedValue uniform (u_x 0 0)	zeroGradient	slip	noSlip
k [m^2/s^2]	fixedValue k	zeroGradient	kqRWallFunction	kqRWallFunction
ϵ [m^2/s^3]	fixedValue ϵ	zeroGradient	epsilonWallFunction	epsilonWallFunction
ν_t [m^2/s]	calculated	calculated	nutkWallFunction	nutkRoughWallFunction k_s 0.004576 m C_s 0.5
p [m^2/s^2]	zeroGradient	fixedValue 0	zeroGradient	zeroGradient

3.4.2. Sensitivity analysis to domain size

Different simulations on the entrance length of the model were performed. These were simulations with a domain of 15 m and that of a twice as long model, thus of 30 m. The entrance length needs to be well away from the area of interest. The grid cells are equidistant in the horizontal and vertical direction and the grid cells are 600x40 in xy-direction (see Fig. 3.1, where x represents the stream-wise direction from inlet to outlet and y is the upward direction). The model with the dimensions of 15 m x 0.25 m consists of 300x40 cells.

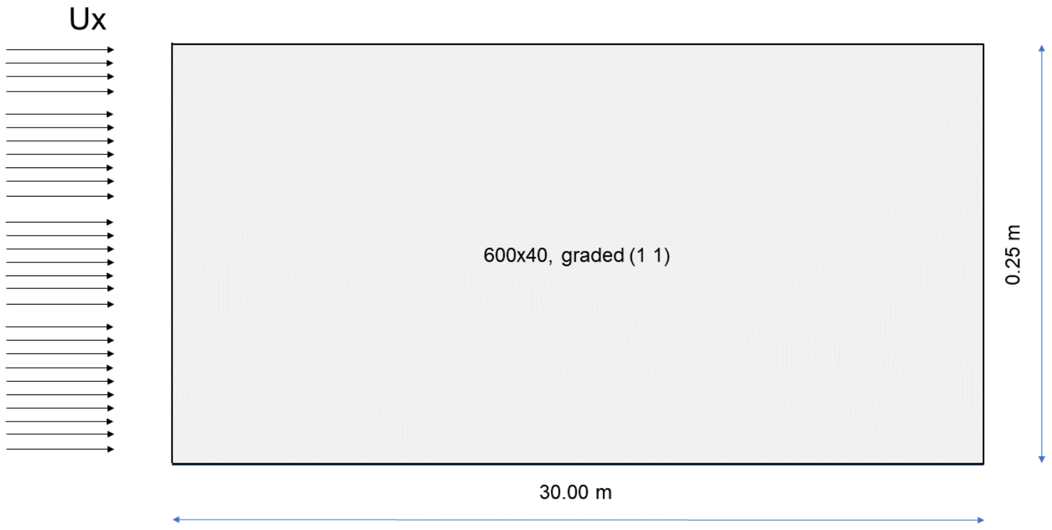


Figure 3.1: The modelled segment, which is twice as long as the flume wherein the results of Zaat (2020) were produced.

3.4.3. Sensitivity analysis to number of grid cells

The sensitivity of the model to the number of cells in the vertical was also assessed. A y^+ value larger than 30 is desired to properly resolve near-wall flow. However, a fine resolution is desired for the release of plastic material and near the interception system, as the particle concentration can vary near the system considerably. The simulations were performed for the different flow conditions for 40 cells in the vertical and a lower number of cells that ensures a $y^+ > 30$, both at the water surface and at the bottom.

3.5. Two-phase model setup

This section handles the boundary conditions necessary for the setup of the two-phase model first. Then the sensitivity study on model coefficients is handled. Ultimately the modelled particles and necessary translation from the experiment results by Zaat (2020) to the OpenFOAM model are explained.

3.5.1. Boundary conditions

The parameters for the two-phase model are defined in Tab. 3.4. Parameters set on the drag and the phases and on the (thermo)physics in Appx. F.4.

It should be noted that the temperature is also solved, but that this does not have influence on the results, as both phases have an initial temperature of 300K and thus no interaction in temperature is considered. The solver inherently is a compressible solver, but a constant density can be set and thus can be viewed as an incompressible solver in this way.

In OpenFOAM an alternative hydrostatic pressure is defined, which is convenient from a numerical point of view. This is defined by p_{rgh}, see Eq. 3.3. The pressure gradient at the inlet boundary is the resulting value such that the flux on the boundary is specified by the velocity boundary condition (OpenCFD Ltd, 2018)[A.4].

$$p_{rgh} = p - \rho g \left(h - h_{ref}\right) \tag{3.3}$$

This equation for the pressure is then substituted in the momentum equation, yielding Eq. 3.4 (OpenFOAM, 2006).

$$\frac{\partial}{\partial t}(\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) - \nabla \cdot (\mu_{\mathrm{eff}} \nabla \mathbf{u}) = -\nabla p_{rgh} - \mathbf{g} \cdot \mathbf{h} \nabla \rho \tag{3.4}$$

At the inlet two fractions of the total volume were provided as a boundary condition. 0.01 out of 1 was set as plastic, whereas the remainder of 0.99 was set as the fluid phase.

Table 3.4: Two-phase initial and boundary conditions. The ”walls” parameter was used in the modelling phase with the device placed.

Parameter	Inlet	Outlet	Surface	Bottom	Walls
u_w	fixedValue (u_x 0 0)	pressureInletOutletVelocity	slip	noSlip	noSlip
u_p	fixedValue (u_x 0 0)	pressureInletOutletVelocity	slip	noSlip	noSlip
k	fixedValue k	inletOutlet inletValue $internalField	kqRWallFunction	kqRWallFunction	kqRWallFunction
ϵ	fixedValue ϵ	inletOutlet inletValue $internalField	epsilonWallFunction	epsilonWallFunction	epsilonWallFunction
$\nu_{t,w}$	calculated	calculated	nutkWallFunction	nutkRoughWallFunction k_s 0.004576 C_s 0.5	nutkWallFunction
$\nu_{t,p}$	calculated	calculated	nutkWallFunction	nutkRoughWallFunction k_s 0.004576 C_s 0.5	nutkWallFunction
p	calculated 1e5	calculated 1e5	calculated 1e5	calculated 1e5	calculated 1e5
p_{rgh}	fixedFluxPressure 1e5	prghTotalPressure p0 uniform 1e5	fixedFluxPressure 1e5	fixedFluxPressure 1e5	fixedFluxPressure 1e5
α_w	fixedValue 0.99	inletOutlet inletValue uniform 0	zeroGradient	zeroGradient	zeroGradient
α_p	fixedValue 0.010	inletOutlet inletValue uniform 0	zeroGradient	zeroGradient	zeroGradient
$\alpha_{T,w}$	calculated	calculated	calculated	calculated	calculated
$\alpha_{T,p}$	calculated	calculated	calculated	calculated	calculated
T_w	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
T_p	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
Θ_p	1e-4	zeroGradient	zeroGradient	zeroGradient	zeroGradient

3.5.2. Interphase momentum exchange model coefficients

The forces considered are the drag force, virtual mass force, lift force and the turbulent dispersion force, conform Section 2.2.8. The drag together with the turbulent dispersion force were expected to yield the largest influence towards the final concentration profile. Solid particles in fluids experience motion between different streamlines due to turbulent flow and thus it is very important that this is correctly resolved. Without added dispersion the particle were projected to rise to the water surface as a consequence of the rise velocity only.

A sensitivity study was performed with different drag models, typically in the basic installation of OpenFOAM. An overview of the available drag models in `reactingTwoPhaseEulerFoam` and their applications is given in Appx. B.4. From this it was clear that the drag models that could be applied were the Syamlal O'Brien, Gibilaro and Wen-Yu drag models.

The drag model of Syamlal O'Brien is derived for a single spherical particle in a fluid and the assumptions lie that the Archimedes number is the same in for a single particle and for a system with more particles (Lundberg and Halvorsen, 2008). The application of Gibilaro is for dilute particle systems and relates the energy dissipation in the fluidised bed with the pressure loss to ultimately obtain the particle drag force (Luo et al., 2019). The Wen and Yu drag model has been derived from settling experiments of particles and the correlation is based on the fact that the flow between the two-phases is homogeneous (Prabhansu et al., 2015).

These drag forces were compared for the different flow velocity cases, which are the cases c01-c03 in Tab. 3.5. Following these results, the drag model that resulted in the least deviations in the concentration profile for the different flow velocities was chosen. Next eight cases (c04-c12) with an inlet flow velocity of 0.55 m/s were differentiated that apply different interphase momentum exchange coefficients, see Tab. 3.5. The results of this sensitivity study were used to find the right combination of interphase momentum exchange coefficients.

Table 3.5: Cases for sensitivity study coefficients. The text in bold describes the deviations from the model of c01.

case #	**Drag model**	ν_p	c_{vm}	c_{vm} **water in plastic**	c_L	C_{td}
c01	**Syamlal O'Brien**	0.01	0.5	0.5	1.6	1
c02	**WenYu**	0.01	0.5	0.5	1.6	1
c03	**Gibilaro**	0.01	0.5	0.5	1.6	1
c04	Gibilaro	**0**	0.5	0.5	1.6	1
c05	Gibilaro	0.01	**1**	0.5	1.6	1
c06	Gibilaro	0.01	0.5	**0**	1.6	1
c07	Gibilaro	0.01	0.5	0.5	**0**	1
c08	Gibilaro	0.01	0.5	0.5	1.6	**0**
c09	Gibilaro	0.01	**1**	**0**	1.6	1
c10	Gibilaro	0.01	**1**	**1**	1.6	1
c11	Gibilaro	0.01	**0**	**0**	1.6	1

3.5.3. From flume experiment to model

In the model environment of `reactingTwoPhaseEulerFoam` it is possible to adjust the diameter and the density of the particle, but not the rise velocity itself.

The concentration for plastic HDPE films can be approximated with a modified Rouse number from spherical particles $\beta_1 = \alpha_{film}\beta_0$ (Zaat, 2020). From a modelling point of view, the modified diameter to match this Rouse profile can be approximated by multiplying the old diameter with $\sqrt{\alpha}$, see Eq. 3.5. The other parameters in this equation were not changed, as this would change the hydraulic conditions.

$$\begin{aligned} \beta_1 &= \alpha_{film}\beta_0 \\ \frac{w_{r,1}\cdot\sigma}{\kappa u_*} &= \alpha_{film}\frac{w_{r,0}\cdot\sigma}{\kappa u_*} \\ -\frac{2}{9}\left(R_{p,1}\right)^2\cdot\frac{\rho_p-\rho_w}{\eta g} &= \alpha_{film}\cdot\left(-\frac{2}{9}\left(R_{p,0}\right)^2\cdot\frac{\rho_p-\rho_w}{\eta g}\right) \\ D_{p,1} &= \sqrt{\alpha_{film}}D_{p,0} \end{aligned} \tag{3.5}$$

3.5.4. Released plastics

The different plastics that were tested are HDPE plastic films both resins typically found in the ocean (Reisser et al., 2015) and because these particles are positively buoyant, interception of this particles could be possible.

The particles were selected on rise velocity and are similar to Zaat (2020) and PP plastic particles. These types have been retrieved using Manta nets (Van der Wal et al., 2015) and the Shoreliner (Vriend et al., 2020) at the water surface.

The particles assessed are defined in Tab. 3.6. The density of the different plastic resins is taken as the average density of these resins and the nominal diameter was calculated from the rise velocity with Eq. A.9. The following Rouse numbers are given in Tab. E.1

Very light particles and neutrally buoyant particles are modelled besides the HDPE and PP particles. This shows the retention of the particles that are in suspension and floating at the water surface. The very light particles have a much higher rise velocity than the downward components of the flow velocity in front of the interception wall and therefore were expected to be retained until a certain volume of particles was reached. With a constant flux it is not possible to retain all the particles. It was expected that for particles with a similar density of that of water that only the upper layer retains some of the particles, but that the downward pointed flow in front of the structure would transport most of the debris downstream.

Table 3.6: Assessed particles

Specified particle	**Diameter [mm]**	**Density [kg/m^3]**	**Rise velocity [m/s]**
Neutrally buoyant particles	0.100	1000	0
Light particles	1.00	750	0.136
HDPE spheres (Rouse)	0.782	970	0.01
HDPE films	0.626	970	0.0064
PP spheres (Rouse)	0.536	920	0.0125
PP films	0.429	920	0.008

3.6. Interception of plastic

The model with the device placed needs additional boundary conditions and a refined mesh closed to the device to properly represent the interception device and resolve plastic dynamics. A single-phase model simulation with the earlier applied boundary conditions was performed, providing insights into the dynamics around the interception system. The two-phase model needed another set of boundaries as well for the same reason. Then the different plastic particles were released and the retention of these was determined.

3.6.1. Adapted single-phase model setup

The meshing is different from the block with equidistant cells in Fig. 3.1 due to refinement close to the system. The number of cells is roughly equal, being: 22120 compared to 24000 cells. The way that the mesh is set up is visible in Fig. 3.2. The exact mesh around the system is given in Appx. C.

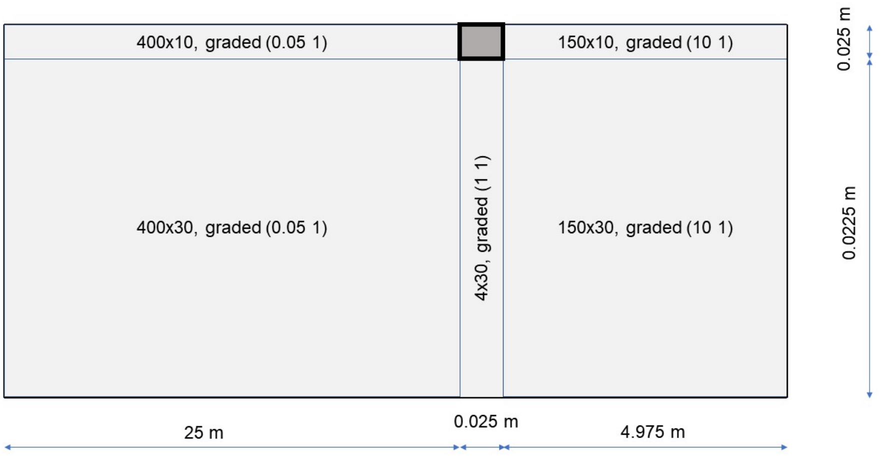


Figure 3.2: Schematised interception system and its mesh in blocks.

Table 3.7: Boundary conditions for the different cases.

Parameter	**Inlet**	**Outlet**	**Surface**	**Bottom**	**Walls**
u [m/s^1]	fixedValue uniform (u_x 0 0)	zeroGradient	slip	noSlip	noSlip
k [m^2/s^2]	fixedValue k	zeroGradient	kqRWallFunction	kqRWallFunction	kqRWallFunction
ϵ [m^2/s^3]	fixedValue ϵ	zeroGradient	epsilonWallFunction	epsilonWallFunction	epsilonWallFunction
ν_t [m^2/s^1]	calculated	calculated	nutkWallFunction	nutkRoughWallFunction k_s 0.004576 m C_s 0.5	nutkWallFunction
p [m^2/s^2]	zeroGradient	fixedValue 0	zeroGradient	zeroGradient	zeroGradient

3.6.2. Adapted two-phase model setup

The setup consists of the same properties as in Tab. 3.4, with again the ”walls” boundary that resembles the interception system. The particles released are given in Tab. 3.6.

The interception tests of the release of neutrally buoyant and light particles indicate two extreme values. These simulations were performed in order to indicate where discrepancies in the simulation lie and how effective the interception system can be for debris in suspension and for mostly floating debris. Both particles are usually encountered in the natural environment.

The results of this step provided clarity into the propagation and entrapment of the plastic fluxes viewed at in this project for the interception system. The different flow velocities were expected to deliver different results, as the wake in front of the structure will be shorter for lower velocities and the downward velocity just in front of the structure will be higher for the cases with a higher velocity compared with the low flow case.

3.7. Projected outcome

The purpose of this study yields a number of desired outcomes, which can aid in better understanding plastic particle dynamics and ultimately the retention of plastic material.

- A model setup that validates the hydrodynamics for a given straight and rectangular flow segment;
- A methodology to replicate the concentration profiles for positively buoyant plastic particles and films with an Euler-Euler modelling approach;
- Concentrations near the interception device: this shows how the interception can work for certain situations.

4

Results

This chapter delivers the results from the tests stated in Chapter 3. First the replication of the flume is handled on the basis of the establishment length, number of grid cells in the vertical and the flow velocity. Then the results of the two-phase flow models are given, with first given the results from the sensitivity analyses. Then the equilibrium concentration profiles are given for the HDPE and PP resins. Hereafter, the hydrodynamics surrounding and specifically in front of the interception mechanism are described. Ultimately, the interception of the different particles are simulated.

In some cases only one of the low, medium or high flow case is discussed. This is because the other figures showed much similarity with the case discussed. This is also mentioned in the different sections. The figures of the cases that have been excluded from the main comments in this chapter are given in Appx. E.

4.1. Single-phase flow

The results of the flow replication were considered first and the plots of the different flow velocities were compared with the theorem on flow.

The entrance length necessary in the flow conditions given are handled first. Then the sensitivity of a change of the number of grid cells is assessed, with respect to the flow velocity. Following the sensitivity analyses, the spin-up time is given. Then a comparison is made between the resulting turbulent eddy viscosity profile from the model and the analytical profile.

4.1.1. Sensitivity on domain size: entrance length required

The development of the flow velocity over the length for high flow velocity is given in Fig. 4.1. Similar figures were found for low flow (Fig. E.1) and medium flow (Fig. E.6). This figure illustrates that at x=15 m, the flow velocity has developed sufficiently over the domain. The flow velocity development is compared with the Law of the wall flow for $c_f = 0.005$, rather than the $c_f = 0.003$ value applied in Chapter 3. This is further explained in Section 4.1.2.

Table 4.1: Entrance length results.

Case	**Entrance length [m]**	**Similar to the findings of:**
u_x= 0.10 m/s	$60 \cdot h$	Nezu and Rodi (1985) , Dean (1978)
u_x= 0.55 m/s	$60 \cdot h$	Nezu and Rodi (1985) , Dean (1978)
u_x= 0.90 m/s	$60 \cdot h$	Nezu and Rodi (1985) , Dean (1978)

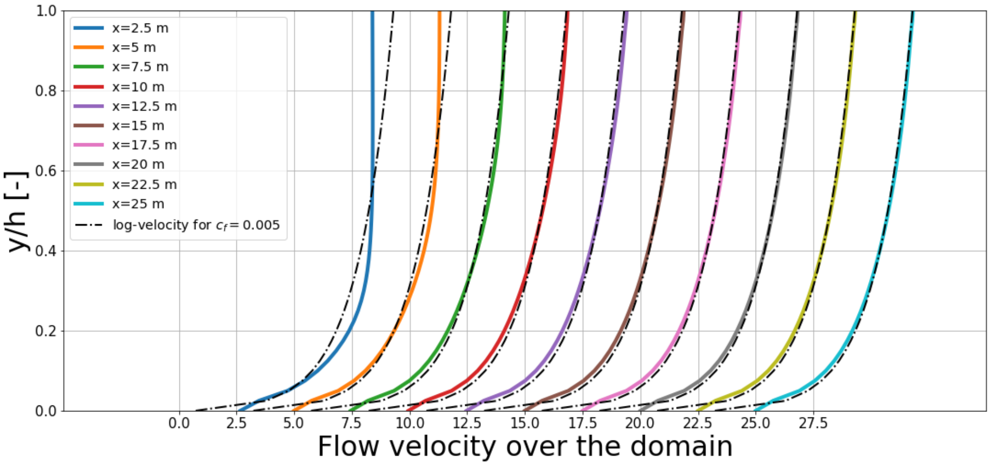


Figure 4.1: Development of the horizontal flow velocity profile in the streamwise direction for high flow. The lines have been scaled and the dashed line illustrates the log-flow velocity for a c_f value of 0.005.

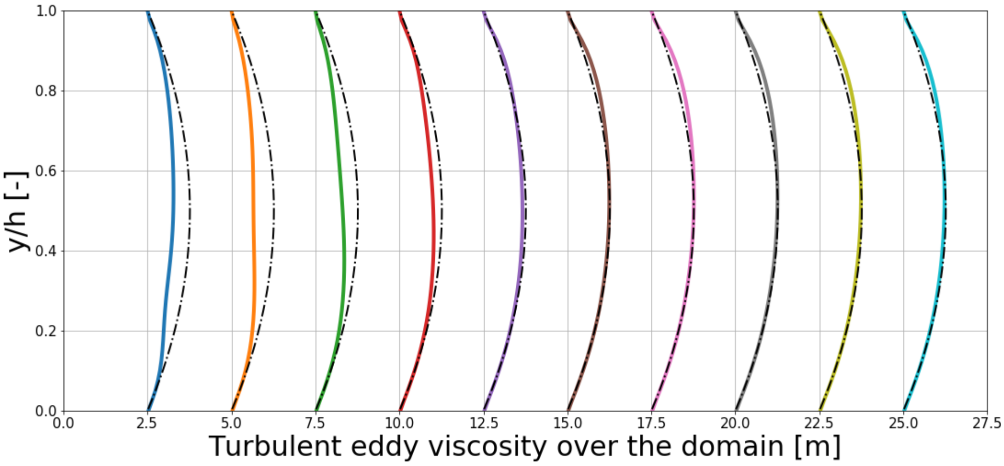


Figure 4.2: Development of the turbulent eddy viscosity profile in the streamwise direction for high flow. The lines have been scaled and compared with the analytical results of Chapter 3. The numerical results from the simulation can be found in Fig. 4.7.

The results of the setup of the flume experiment with similar dimensions as Zaat (2020) and that of twice the domain are visualised in Appx. E.1. Here the flow velocity, turbulent kinetic energy, energy dissipation and the turbulent eddy viscosity are given for low flow (Appx. E.1.1), medium flow (Appx. E.1.2) and high flow (Appx. E.1.3).

4.1.2. Sensitivity on number of grid cells: flow velocity profile

The results of coarse grids were compared with the results from Section 4.1, which are simulations with 40 grid cells in the vertical. The coarse grids for the different simulations ensured a y^+-value above 30 in every cell, so that these cells near the wall are out of the buffer layer zone. The y^+ values for the different simulations are visible in Tab. 4.2. The results of the different simulations can be found in Fig. 4.3 and do not show clear differences in the gradient of the velocity profile.

What is striking is that both flow velocities over depth do not follow the flow velocity line for law of the wall, but do this for that of an increased friction coefficient, of that of 0.005. The differences near the surface are small and fulfil a logarithmic development over the depth. It can be seen from the figure with low flow that a non-logarithmic flow velocity is present. This can be explained from the law of the wake.

Table 4.2: Average results y⁺.

Case		n_{cells} **vertical**	y^+ **bottom**	y^+ **surface**
u_x = 0.10 m/s	coarse	8	66.6	38.9
	fine	40	20.8	4.88
u_x = 0.55 m/s	coarse	20	54.1	217
	fine	40	77.5	35.2
u_x = 0.90 m/s	coarse	15	599.2	178.9
	fine	40	182.6	38.2

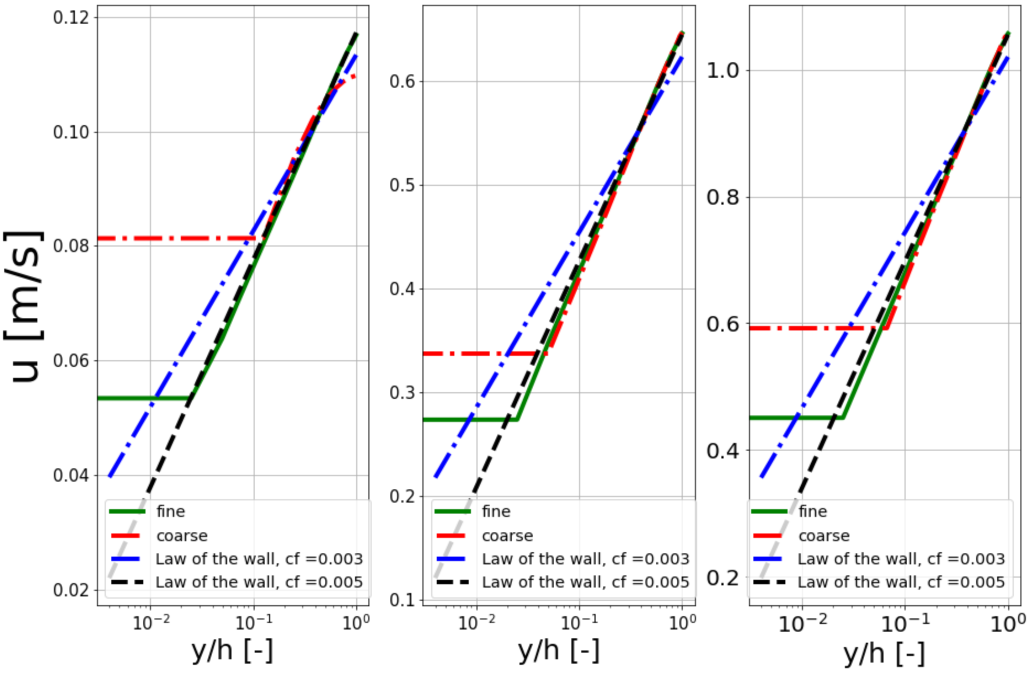


Figure 4.3: Simulations for the relative depth over flow velocity for u=0.1 m/s, u=0.55 m/s, u=0.1 m/s, with a different number of grid cells in the vertical, see Tab. 4.2.

At the bottom a horizontal line can be observed for the flow velocity over the depth in Fig. 4.3. This can be explained from the level of detail in the model compared to the analytical solution. The level of detail at the bottom cannot be resolved for the turbulence model applied. This near-bottom flow is not in the scope of this project, as the focus lies on near-surface flow. The deviations there are given in Tab. 4.3. The difference in near-surface flow results seem large from the flow velocity figures, especially with a logarithmic plotted axis, but are in the order of maximally 4% and thus deemed acceptable differences.

Table 4.3: Maximum and mean velocity of the different flow velocity cases

Case		**CFD Model**	**Analytical result c_f=0.003**	**Analytical result c_f=0.005**
Low flow	maximum velocity [m/s]	0.117	0.113	0.117
	mean velocity [m/s]	0.0995	0.0996	0.0995
Medium flow	maximum velocity [m/s]	0.646	0.623	0.645
	mean velocity [m/s]	0.547	0.548	0.547
High flow	maximum velocity [m/s]	1.06	1.02	1.06
	mean velocity [m/s]	0.896	0.896	0.895

4.1.3. Spin-up time

The simulation time towards the steady state simulation can be seen for low flow (Fig. 4.4), medium flow (Fig. 4.5) and high flow (Fig. 4.6). The cases can be described as steady after t=4000 s for low flow and for the two other cases after t=3000 s.

These simulation times found have been applied for the assessment of finding a steady case for the flow around the structure in Section 4.6.

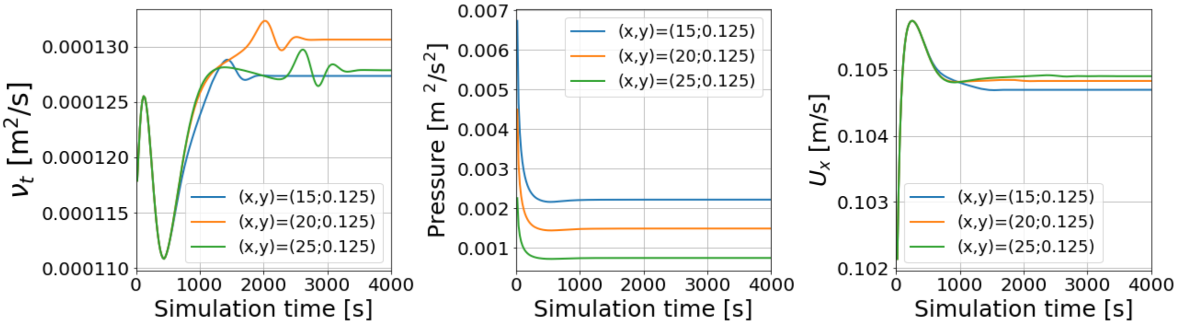


Figure 4.4: Simulation time (0 s < t < 4000 s) to steady simulation for different parameters for low flow conditions.

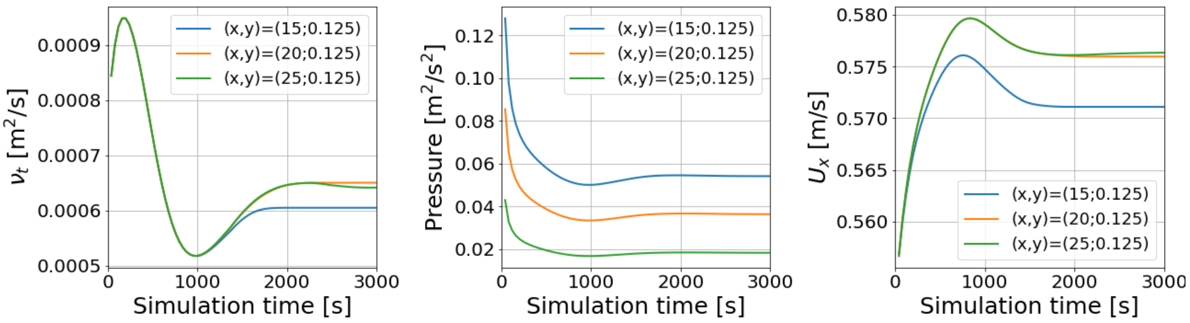


Figure 4.5: Simulation time (0 s < t < 3000 s) to steady simulation for different parameters for medium flow conditions.

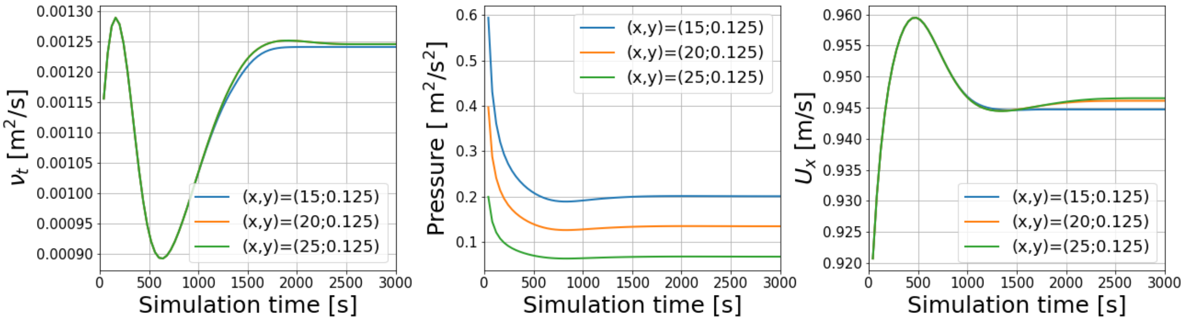


Figure 4.6: Simulation time (0 s < t < 3000 s) to steady simulation for different parameters for high flow conditions.

4.1.4. Estimation turbulent eddy viscosity

The figure for high flow and the turbulent eddy viscosity is given in Fig. 4.7. Similar figures were found for low flow (Fig. E.5) and for medium flow (Fig. E.7). It is clear from the results that the different functions for the turbulent eddy viscosity do not deviate much. However, the simulation of the turbulent eddy viscosity for low flow gives an underestimation in the results, which could cause less turbulent diffusion of the particles than expected. This yields a maximum difference of approximately 9% between the turbulent eddy viscosity maxima.

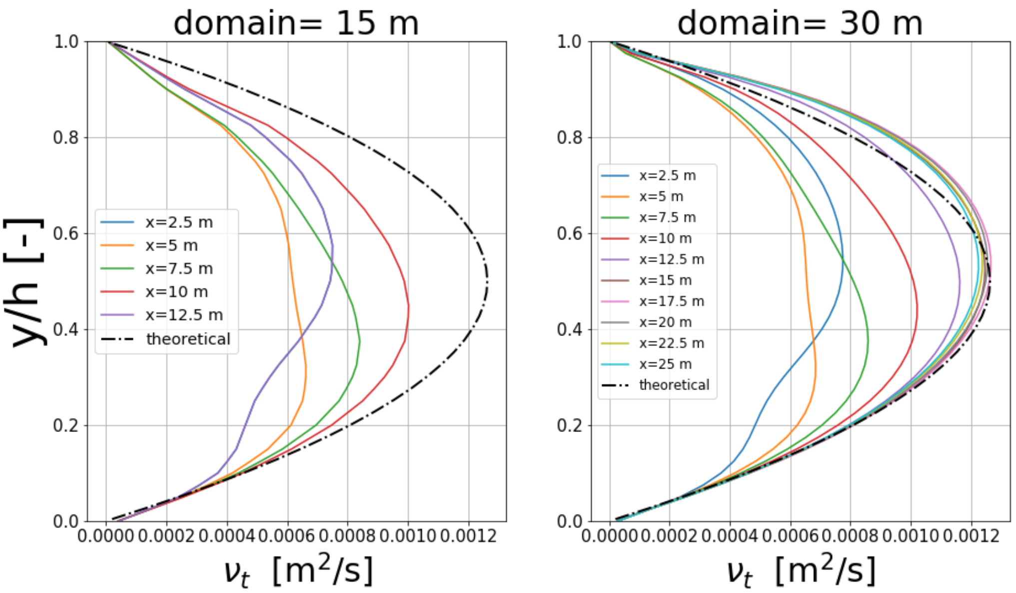


Figure 4.7: Turbulent viscosity over the domain and water depth for 15 m and 30 m. The flow velocity at the inlet is 0.90 m/s.

The maximum of the turbulent eddy viscosity is roughly equal to the value computed analytically ($\approx$ 3%), see Fig. E.7. However, the maximum of the turbulent eddy viscosity profile is found at 0.135 m, rather than in the middle of the water column, at 0.125 m as would be expected. A deviation of 1 cm is minor and especially small compared to the flow conditions of the shorter domain size.

Similar conclusions can be made for the higher flow velocity case: the maximum value is roughly equal to the maximum found from analytical results ($\approx$ 3%) and the profile is overestimated in the upper half of the water column, with its maximum also at h=0.135 m.

4.1.5. Remarks single phase model

The further simulations have been executed with a domain size of 30 m because these results come near the analytical results and an equilibrium for the flow conditions is reached. The assessment of different y^+ values show that the difference in number of cells in the vertical does not have an impact on the final results for these assessments. The wall functions are designed for a correct value of the turbulent quantities at the first node away from the wall, not at the wall itself and further simulations were executed with 40 grid cells in the vertical. This level of detail is preferred for further simulations near the interception system.

The entrance length is 15 m for the flow parameters of the single-phase model, with the area of interest needed to be set for x larger than 15 m, or 60 times the depth in these simulations. The addition of a second phase influences the entrance length of an equilibrium particle concentration. Particles of different size and density could experience different entrance lengths. An investigation is performed if these lengths are in the same order in the next stages.

The flow velocity over the depth deviates from the analytical results, which could be caused by a source term in the model. The solution corresponds with the turbulent eddy viscosity profile, which is of more importance than the solution of the flow velocity, as this largely influences the diffusion of particles in the water column.

4.2. Sensitivity analysis two-phase model

This section consists of tests that investigate how the model reacts to certain differences in the simulations. First, simulations with neutrally buoyant particles were performed. Then the impact of different drag models was investigated. Ultimately, the different coefficients for the interphase momentum exchange were simulated with the most appropriate drag model simulation found.

4.2.1. Release of neutrally buoyant particles

The distribution of neutrally buoyant particles over the depth is given in Fig. 4.8, at x= 25 m. It can be seen that the concentration profile follows a near uniform distribution for the different cases and a very minor decrease of concentration is visible at the water surface. This can be explained from the fact that the flow velocity at the water surface is highest and flushes away the particles at this location.

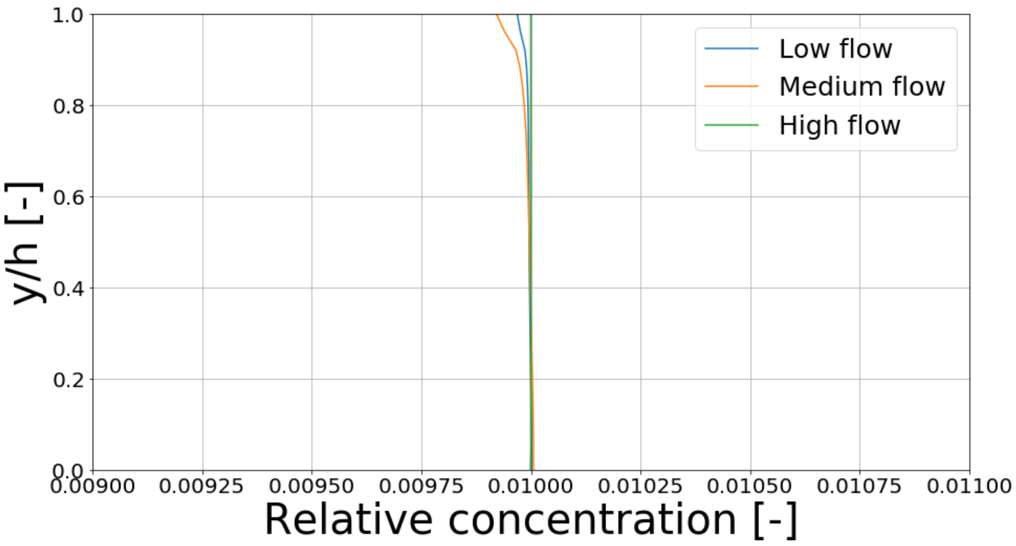


Figure 4.8: Concentration profiles of particles with $\rho = 1000$ kg/m^3 for low, medium and high flow at x =25 m.

4.2.2. Application of different drag models

The results of the simulations c01-c03 with the different drag models for HDPE material are given in Fig. 4.9. It shows that an overestimation for the Wen-Yu and Syamlal-O'Brien drag models are given in suspension. This is also true for the Gibilaro model until around 0.85 times the depth, but deviates very slightly. These deviations are further elaborated on in Tab. 4.4. The Gibilaro model together with a buoyancy and virtual mass force agrees well (Rasul, 1999) and is specifically developed for high Re flow (Visuri et al., 2012). For this reason the results with the Gibilaro drag model fit the profiles better.

Near-surface an underestimation in the particle concentration is visible. This is against the expectations from Section 4.1.4, where an overestimation of the turbulent eddy viscosity profile was found. It would be more logical to observe an underestimation at this region: this increased value of the turbulent eddy viscosity usually indicates more mixing and thus fewer particles found near-surface. This overestimation is in the order of 10%, see Tab. 4.4.

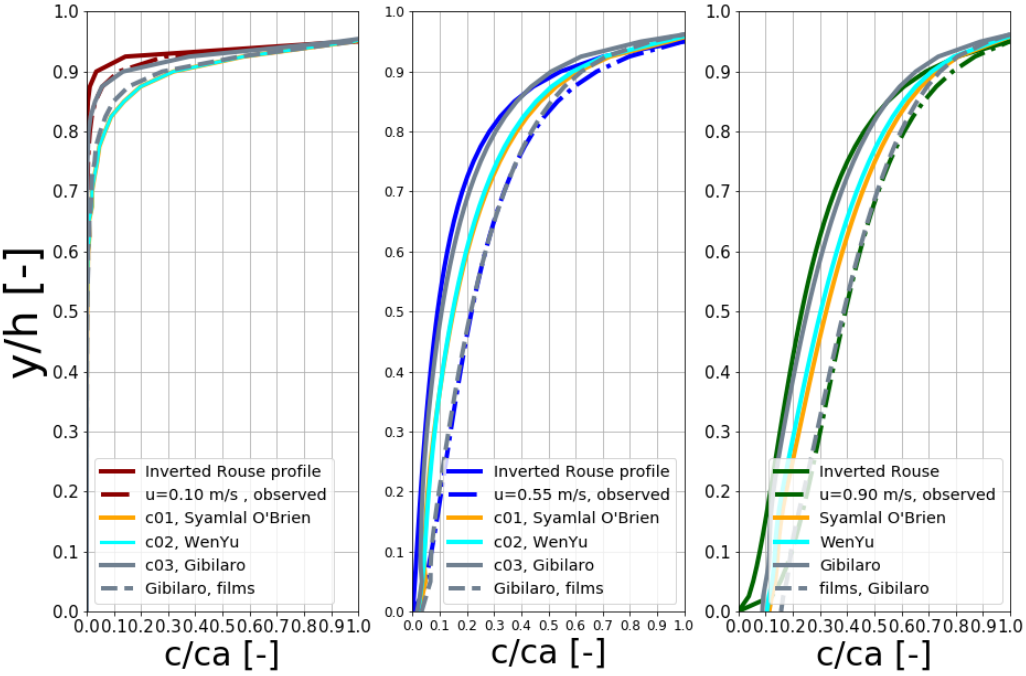
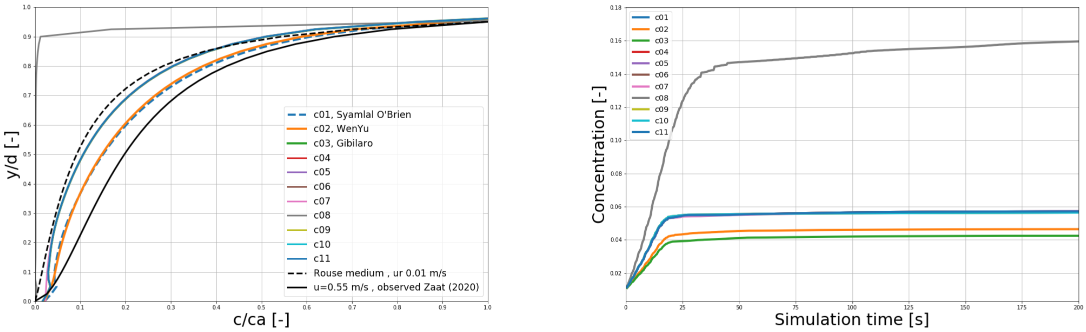


Figure 4.9: The relative plastic concentration for the different drag models for HDPE material over the depth, at x=25 m. The left figure illustrates low flow, the middle figure medium flow and the right figure high flow.

4.2.3. Response to model coefficients

The results from the cases with different model coefficients are indicated in Fig. 4.10a. These simulations have been performed with the medium inlet velocity of 0.55 m/s. The simulations that applied the same drag model showed no noticeable differences between all but one simulation, case 8, which does not consider a turbulent dispersion coefficient. It can be said that the influence of the turbulent dispersion is largest, which is logical as the transport of the particles is largely influenced by the spreading of these particles.



(a) Concentration profiles at x=25 m.

(b) Concentration over time at (x, y) = (25; 0.25).

Figure 4.10: Simulation of the plastic concentration for the different coefficients from Tab. 3.5.

4.3. Two-phase flow: equilibria HDPE resin

The simulation time to reach steady state, entrance length of the turbulence viscosity and the development of the concentration profile over the domain are checked in this section for high density poly ethylene, similarly to the single-phase case.

4.3.1. Spin-up time

In Fig. 4.11, Fig. 4.12, Fig. 4.13 the simulation time relative to other flow parameters are plotted. For low flow conditions it is clear that the simulation time is much longer than that of medium and high flow cases. This increase in simulation time could be explained from the fact that the turbulence governs the development of the parameters less for low flow than that for the medium and high flow cases. It could also be the case that the friction governs the flow more for low flow, implicating longer simulation times. The concentration profile shows little differences compared to the simulation for a low flow velocity. This can be explained from the fact that for a lower degree of turbulence the particles do not spread in the horizontal and vertical and are mainly driven towards the surface due to the buoyancy force.

The time to reach equilibrium conditions is much shorter than that for single-phase flow. This can be explained from the fact that the model coefficients influence the simulation time in a way. The virtual mass coefficient has been tested to decrease the necessary simulation time to reach equilibrium conditions (Davidson, 1990; Drew et al., 1979; Drew, 1983).

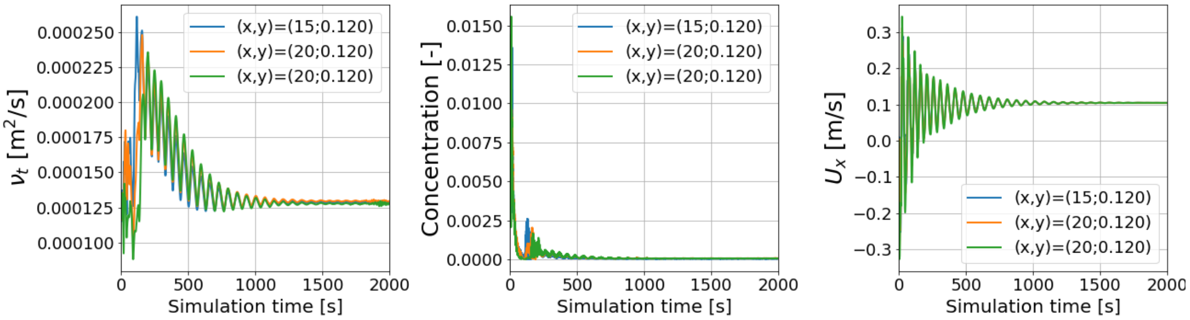


Figure 4.11: Simulation time (0 s < t < 2000 s) to steady simulation for different parameters for low flow conditions.

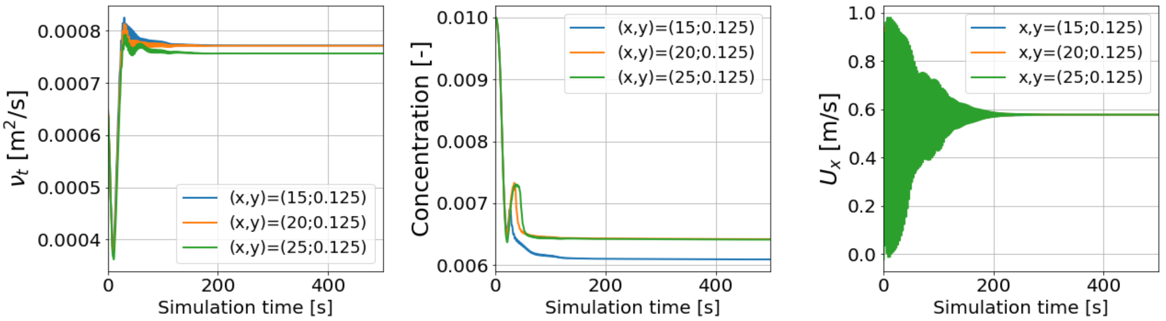


Figure 4.12: Simulation time (0 s < t < 500 s) to steady simulation for different parameters for medium flow conditions.

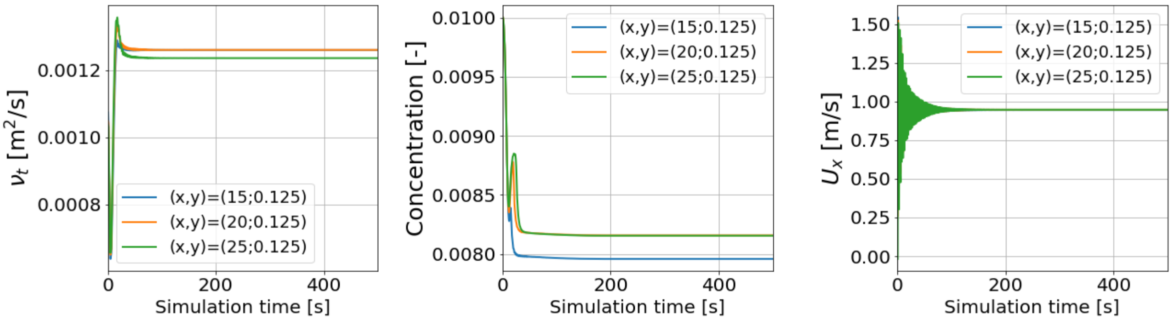


Figure 4.13: Simulation time (0 s < t < 500 s) to steady simulation for different parameters for high flow conditions.

4.3.2. Comparison of flow and turbulent viscosity with single-phase runs

The entrance length for the turbulent eddy viscosity is 15 m for the HDPE material, similarly to the entrance length for the single-phase flow in Section 4.1.4. This can be seen for medium flow in Fig. 4.14 and in Appx. E for low Fig. E.14 and high flow Fig. E.15. A very similar profile is also found, with the maximum of the profile just above the middle of the water column again at y =0.135 m. A slight overestimation of the turbulent eddy viscosity is visible as well, upward from the same y-coordinate.

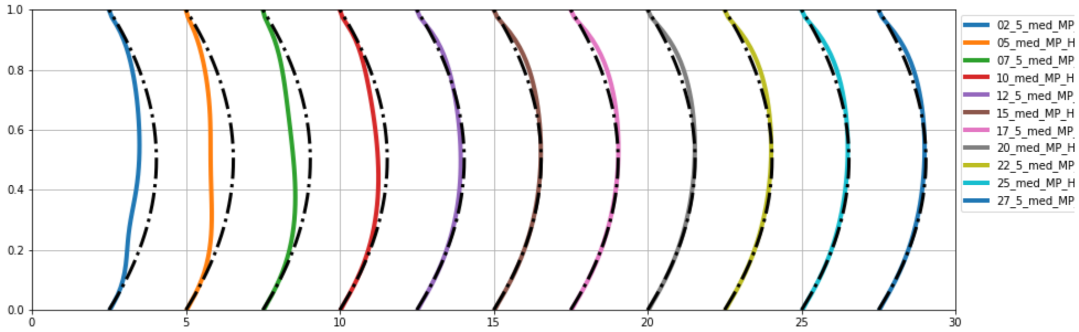
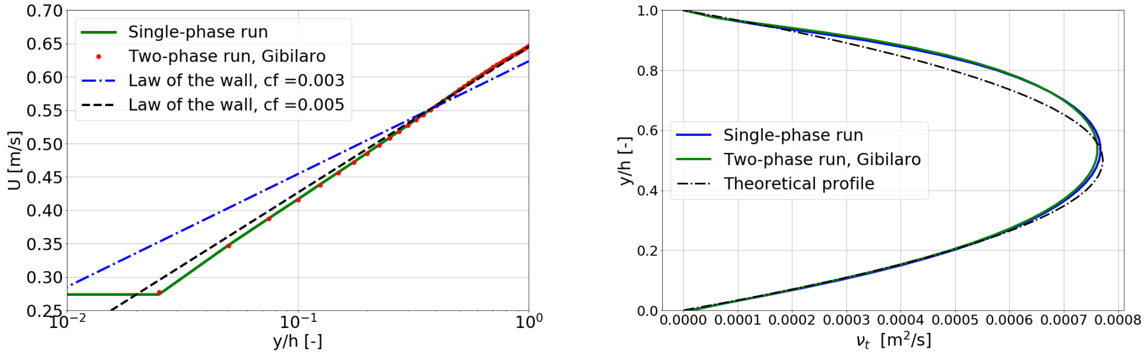


Figure 4.14: Turbulent eddy viscosity over length and depth over 30 m for medium flow conditions. The eddy viscosity is scaled x2000 for viewing purposes.

This is clearly illustrated in Fig. 4.15: the flow over the depth is exactly similar. The turbulent eddy viscosity value differs less than a percent.



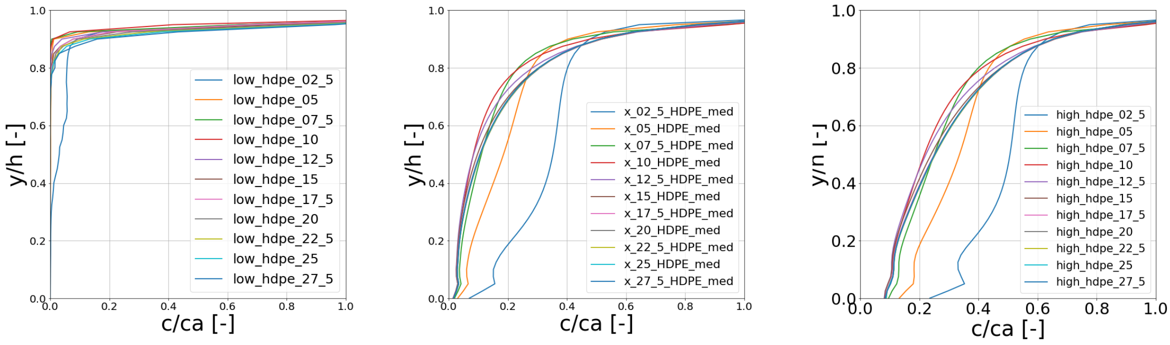
(a) Comparison flow velocity.

(b) Comparison turbulent eddy viscosity.

Figure 4.15: Single and two-phase simulations at x=25 m, compared with theorteical profiles.

4.3.3. Entrance length particle concentration

Where the entrance length for the different flow parameters is not influenced by the value of the inlet flow velocity, this does matter for the development of the particle concentration profile. It can be clearly seen in Fig. 4.16 that the development of the concentration profile for low inlet flow (Fig. 4.16a) the concentration profile over the depth is changing and more or less comes to an equilibrium profile after 15 m, but still moves downward, even until the end of the domain of the model. For medium and high flow the equilibrium concentration profile it can be reached that the equilibria for the concentration are reached after 15 m as well, with less noticeable differences after this distance.



(a) Low flow.

(b) Medium flow.

(c) High flow.

Figure 4.16: Relative HDPE concentrations along the water depth at different slices in x-direction (a) for low flow; (b) for medium flow; (c) for high flow.

The reason that the entrance length for low flow is higher than for high flow can be explained by the increase in flow vortices with an increase of flow. A comparison between laminar and turbulent flow can be made. If a particle experiences motion in laminar flow, this motion will be mainly influenced by the gravity for vertical motion and horizontally with respect to the flow. In a turbulent flow field the particles experience another random motion due to the vortices and will come faster to the expected equilibrium profile. These differences between low and high flow are not as apparent as for laminar and turbulent flow. However, the difference in entrance length comes from a less energetic flow field and smaller vortices for low flow is plausible.

4.3.4. Replication concentration profiles

Similar over- and underestimations are found from the simulations compared to the results of Zaat (2020). The results yield a conservative output for the interception of plastics in Section 4.6, as the material near-surface is estimated to be higher than in reality. The figures that give a one-on-one comparison for the different flow velocities are given in Appx. E.3. The results left in Fig. 4.17 contain the theoretical profiles from Rouse and that for films (Zaat, 2020), where the simulation results are shown in the right figure.

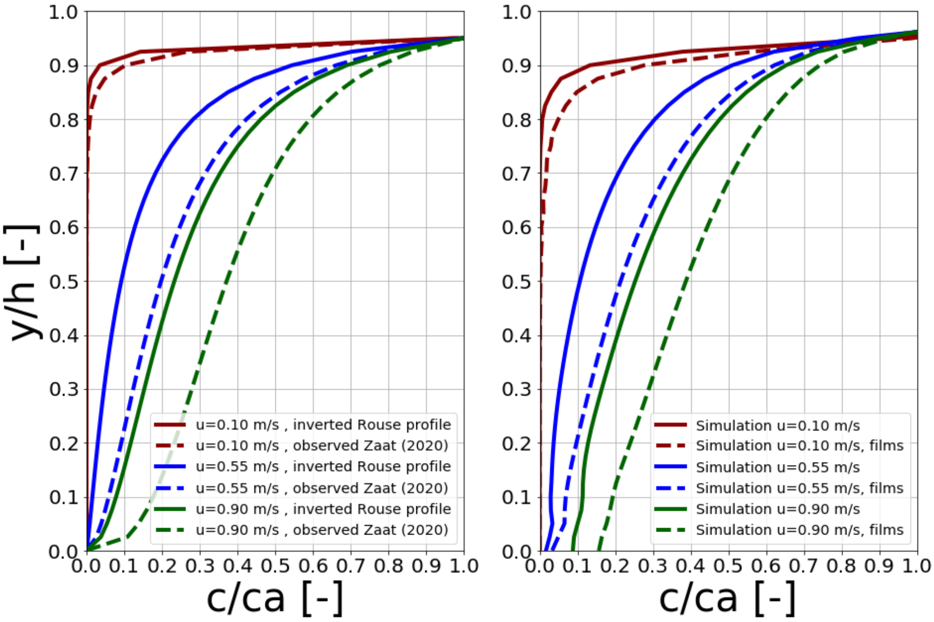


Figure 4.17: Plastic concentrations for the different inlet flow velocities.

The largest differences in the concentration profiles left and right are given in Tab. 4.4. This excludes the first six cells (0.031 m) near the bottom, as the results near the bottom do not go towards zero, whereas the empirical function of the Rouse profile is designed to go towards zero. It can be said that the shape factor for the diameter can be used to reach the concentration profile for HDPE films found by Zaat (2020). However, this is only the case for higher flow velocities, with very low flow and thus a low Reynolds number this replication does not work with the inverted Rouse profile and plastic films. The replication of a concentration profile similar to the Rouse profile is tested for the PP material in the next section.

Table 4.4: Maximum and minimum underestimations in percentages.

	Maximum underestimation [%]	**Maximum overestimation [%]**
Low flow Rouse	39	4
Low flow films	69	-
Medium flow Rouse	18	15
Medium flow films	39	-
High flow Rouse	9	11
High flow films	4	8

The differences that are striking the most is that the results for the high flow velocity for plastic films show approximately the smallest differences, which agrees with earlier findings that this drag model shows a significant improvement for higher Reynolds numbers and thus more particles in suspension (Gibilaro et al., 1985). From this viewpoint it is also explicable that this mismatch is largest for low flow.

4.4. Two-phase flow: equilibria PP resin

The same procedure has been applied for the polypropylene resin as in Section 4.3. First the spin-up times are discussed, then the entrance length for this lighter material and ultimately the found equilibrium concentration profiles.

4.4.1. Spin-up time

Similar figures were found to ultimately get the steady state solutions. The difference with the HDPE material is that the solution converges faster. The spin-up time for the flow velocity was longest for every simulation. These are given in Tab. 4.5 for both the HDPE and PP resin.

It should be noted that small wiggles were still seen after t=2000 seconds for the low flow conditions, but that these were considered negligibly small, the difference being less than a percent compared to the steady solution for the single-phase flow.

Table 4.5: Spin-up times for the flow velocity HDPE and PP resins.

	Spin-up time for HDPE [s]	Spin-up time for PP [s]
Low flow	1400	1250
Medium flow	250	225
High flow	120	110

4.4.2. Comparison of flow velocity and turbulent viscosity with single-phase runs

Turbulent eddy viscosity profiles were found similar to the profiles found in Section 4.3.2. The turbulent eddy viscosity was found to be smaller than at most 2% compared to the turbulent eddy viscosity of the single-phase flow results.

The flow velocity and turbulent eddy viscosity profiles are given in Fig. 4.19a and Fig. 4.19b, respectively. Results of the low flow and high flow cases showed similarity with the single-phase model results as well.

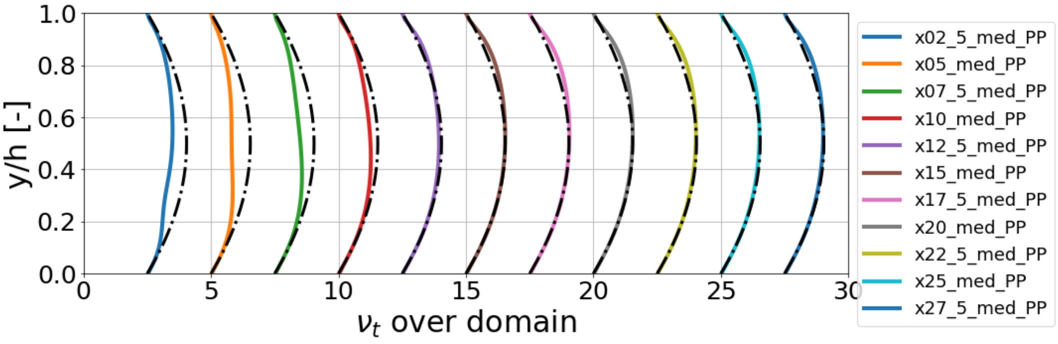
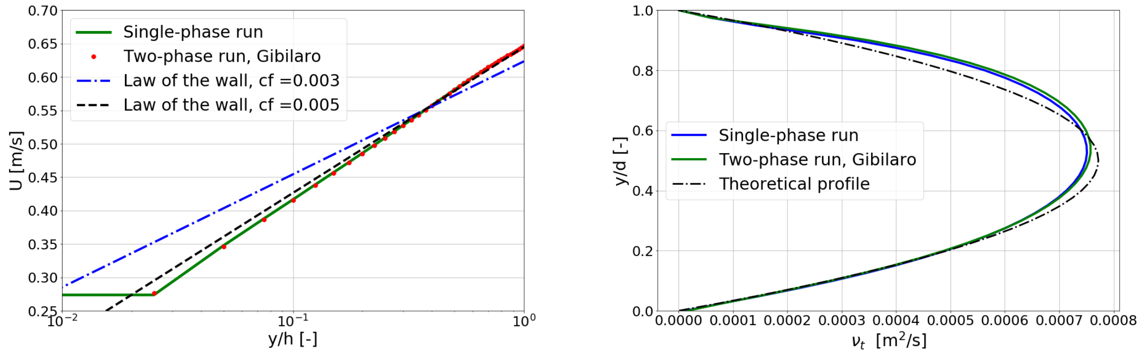


Figure 4.18: Turbulent eddy viscosity over length and depth over 30 m for medium flow conditions. The eddy viscosity is scaled x2000 for viewing purposes.



(a) Comparison flow velocity.

(b) Comparison turbulent eddy viscosity.

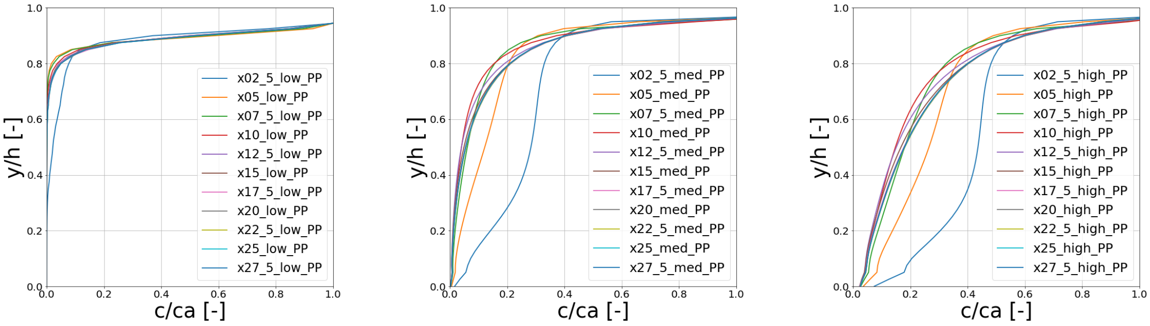
Figure 4.19: Single and two-phase simulations at x=25 m.

4.4.3. Entrance length particle concentration

The simulations of the PP particles resulted in similar profiles with roughly the same entrance lengths, as can be seen in Section 4.4.3. The entrance lengths found from the figures are given in Tab. 4.6. It can be said from these results that such a small difference in rise velocity does not affect the entrance length.

Table 4.6: Entrance length for the different flow velocity cases for HDPE and PP material

	Entrance length for HDPE [m]	Entrance length for PP [m]
Low flow	12.5	12.5
Medium flow	15	15
High flow	17.5	17.5



(a) Low flow

(b) Medium flow

(c) High flow

Figure 4.20: Relative PP sphere concentrations along the water depth at different slices in x-direction (a) for low flow; (b) for medium flow; (c) for high flow.

4.4.4. Concentration profiles

Similarly to Section 4.3 the concentrations for the low flow conditions are underestimated near the water surface. The medium and high flow velocity cases give similar underestimations in the lower half of the water column and overestimates the relative concentration when comparing to the concentration profiles of Zaat (2020) and Rouse.

The particle dynamics for low flow for both the larger diameter and the smaller diameter to represent films show much deviation from theory. The larger diameter, thus the brown line are brought more in suspension than for the smaller diameter. Here the Gibilaro drag model does not solve the particle fields correctly.

The concentration profiles for the medium and high flow again confirm that a inverted Rouse profile and a diameter of $\sqrt{\alpha}$ is reached with the current model set-up. The flow around the interception system and subsequently the interception of the different particles is investigated in the next sections.

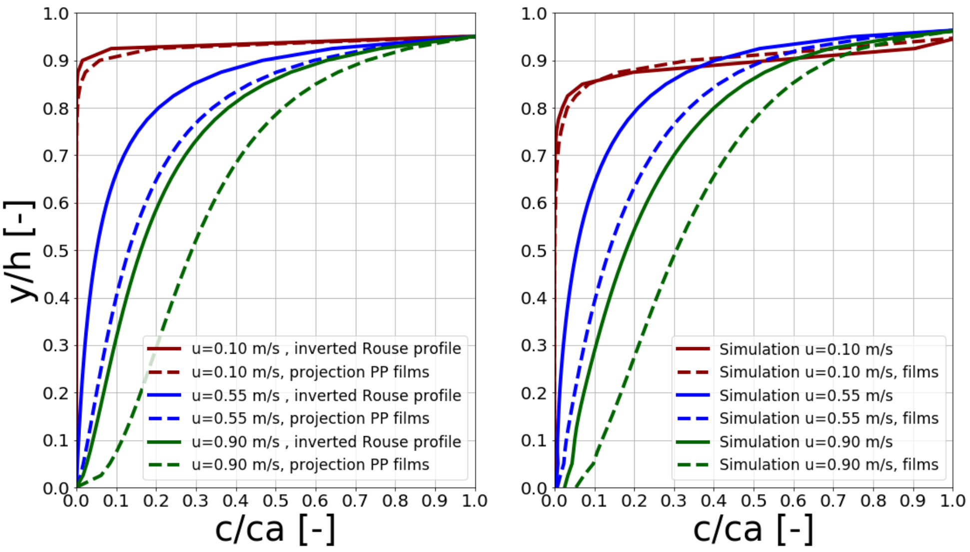


Figure 4.21: PP resin concentrations for the different inlet flow velocities.

4.5. Hydrodynamics interception system: underflow

The results of the interception system are given in this section for the flow around the structure and the turbulent eddy viscosity around the system. The downward velocity gives an indication on the threshold on the retention of the particles.

4.5.1. Development of horizontal flow velocity and turbulent eddy viscosity

The contraction of flow for the medium flow case is described in this section for the horizontal flow velocity and the resulting turbulent eddy viscosity. The results for the low and high flow cases followed the same essence and are given in Appx. E.5.1 and Appx. E.5.2 for the low and high flow respectively.

In Fig. 4.22 the horizontal velocity field between x=24.5 m and x=25.5 m is visualised for the horizontal flow velocity. A wake upstream and downstream of the interception device can be observed from this figure. This was further analysed and distinguished into different regions.

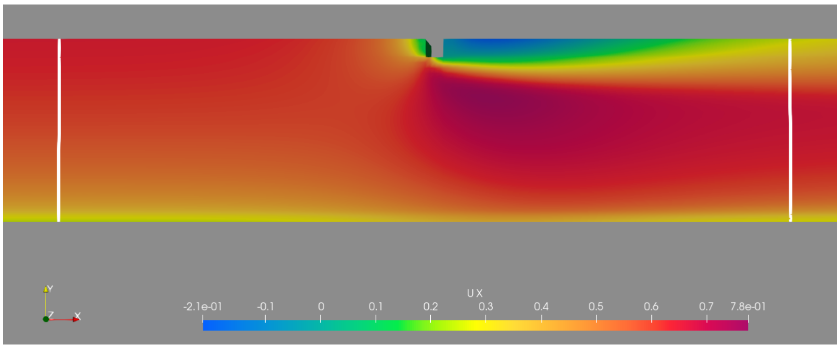


Figure 4.22: The horizontal velocity field for $u_x = 0.55$ m/s. The vertical lines plotted mark x=24.5 m to 25.5 m.

Different slices of the flow around the interception device were taken and subdivided into four categories: upstream of the device, underneath the device, just behind the device and further downstream. These locations are visualised for the flow velocity in Fig. 4.23 and for the turbulent eddy viscosity in Fig. 4.24.

- Upstream of the device 24.4 m $\leq$ x $<$ 25 m:

The wake is present from x= 24.8 m until the location of the interception system. The deviations are visible in the upper left of the four figures. At x=24.4 m the flow velocity and the turbulent eddy viscosity follow the lines of the simulations in Section 4.1, but feel the placement of the structure after this point. The maximum flow velocity increases by 8% compared to the maximum flow velocity for the equilibrium conditions. This deviation is 8% as well for the low and high flow conditions. The decrease in flow velocity occurs mainly in the top 10% of the water column, which is the draught of the interception system.

The turbulent eddy viscosity decreases at the water surface as a result of the decreasing flow velocity near-surface. This decrease indicates that the zone for retention of particles lies between x=24.8 m and x=25 m.

- Underneath the device 25.005 m $\leq$ x $\leq$ 25.020 m:

Under the system the flow is contracted and an increase in flow velocity is visible. This is logical from the perspective that the area is smaller and the discharge is equal. The increase in flow and turbulent eddy viscosity are approximately 18% larger than for equilibrium flow.

Particles were expected to be in concentrated in the middle of the water column here, as the turbulent mixing occurs the most there, apparent from the steep profile in the figure for the turbulent eddy viscosity.

- Just behind the device 25.0275 m $\leq$ x $\leq$ 25.5 m:

Flow separation occurs in this region, visible by the two-directional flow in two layers and two peaks in the turbulent eddy viscosity. Very near the interception device the near-surface flow velocity is negative and the vector increasing until 0.075 m behind the system. This wake continues until the later mentioned region.

- Further downstream of device 26 m $\leq$ x $\leq$ 29 m:

Further downstream of the interception system the flow does not develop into equilibrium conditions and does not follow a logarithmic profile. The peak flow is situated approximately halfway of the water depth.

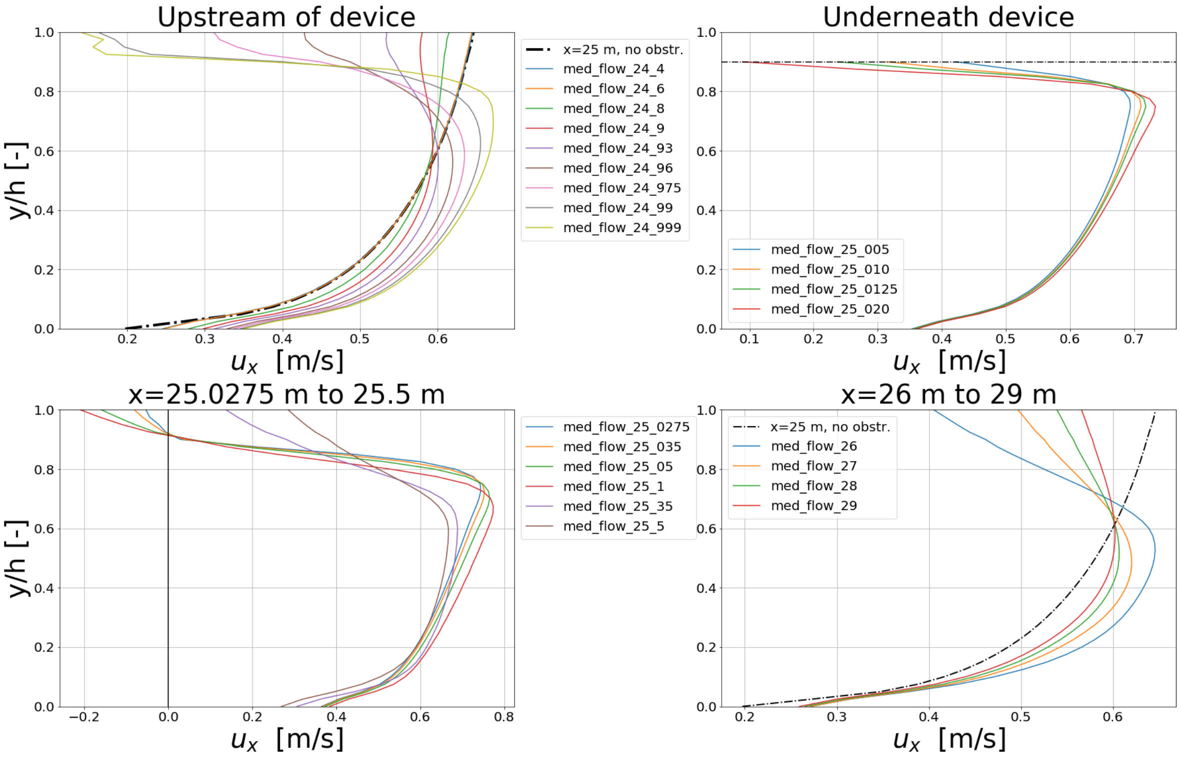


Figure 4.23: Flow velocity profile over the depth at different locations around the interception device for medium flow. The dotted line in the upper right figure indicates the draft of the interception device. The vertical line in the lower left figure indicates u_x =0 m/s.

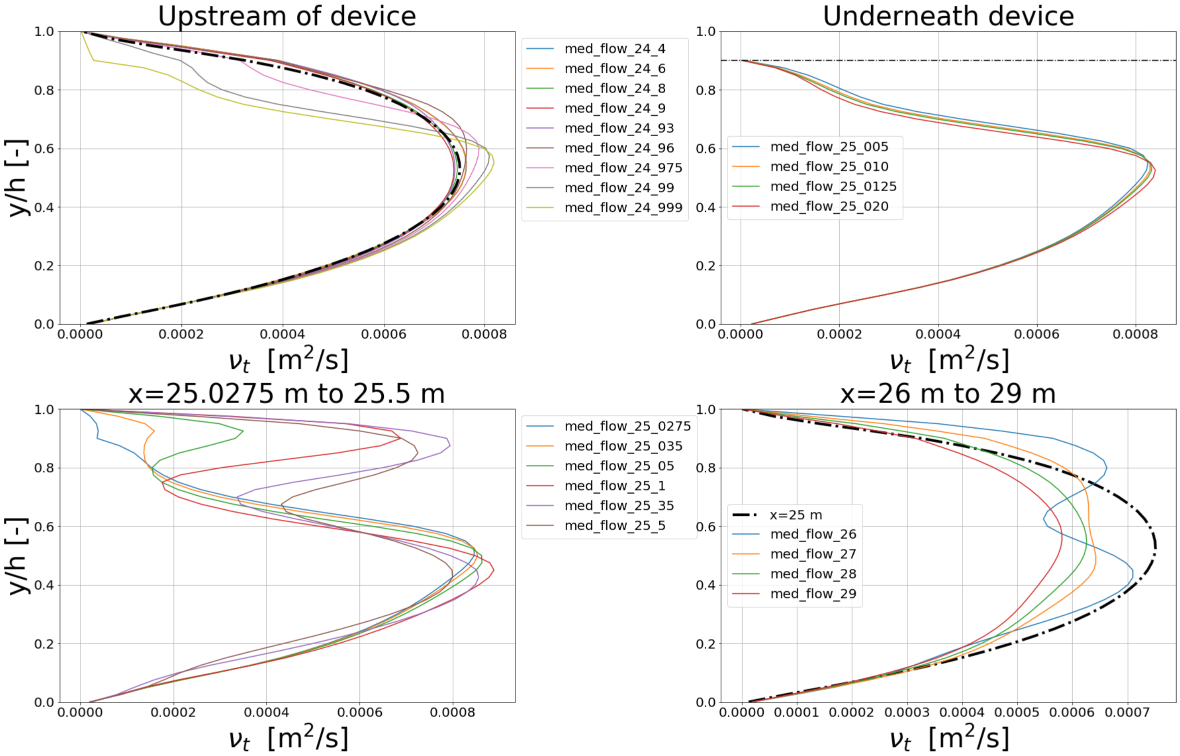


Figure 4.24: Turbulent eddy viscosity over the depth at different locations around the interception device for medium flow. The dotted line in the upper right figure indicates the draft of the interception device.

4.5.2. Vertical flow velocity around interception device

In Fig. 4.25 the vertical velocity field is visualised between x=24.5 m and x=25.5 m. A downward flow field can be observed just upstream of the interception device. Results in the same line were found for low and high flow, see Appx. E.5.

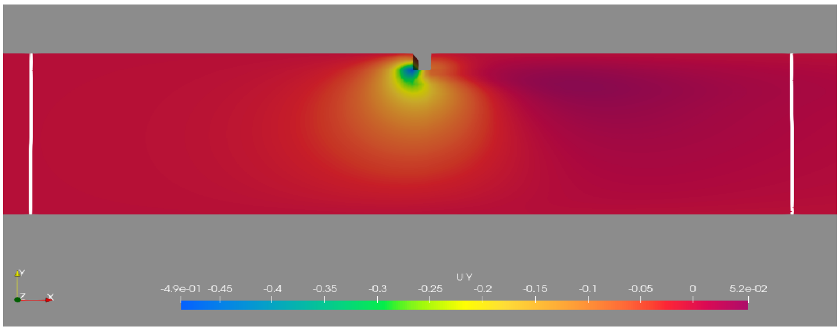


Figure 4.25: The vertical velocity field for $u_x = 0.55$ m/s. The vertical lines plotted mark x=24.5 m to 25.5 m.

- Upstream of the device 24.4 m ≤ x < 25 m:

Upstream of the system a clear division between the top 10% and of that of the other 90% can be made. In the upper section a steep gradient in downward flow can be observed, which could influence the particles moving downward of the interception device. The other region does not show this behaviour and in the lower half of the water column not as much downward flow is observed.

- Underneath the device 25.005 m ≤ x ≤ 25.020 m:

Underneath the system there is downward flow only. Particles here will not be encountered to be retained when having passed the interception device.

- Just behind the device 25.0275 m ≤ x ≤ 25.5 m:

The wake behind the system consists of mainly downward flow only and develops into a profile from x=25.35 m to a velocity profile with only upward flow. This value is small compared to the downward velocities encountered.

- Further downstream of device 26 m ≤ x ≤ 29 m:

In this region there is upward flow only, albeit much smaller than the downward flow upstream of the device, $O \approx 100$.

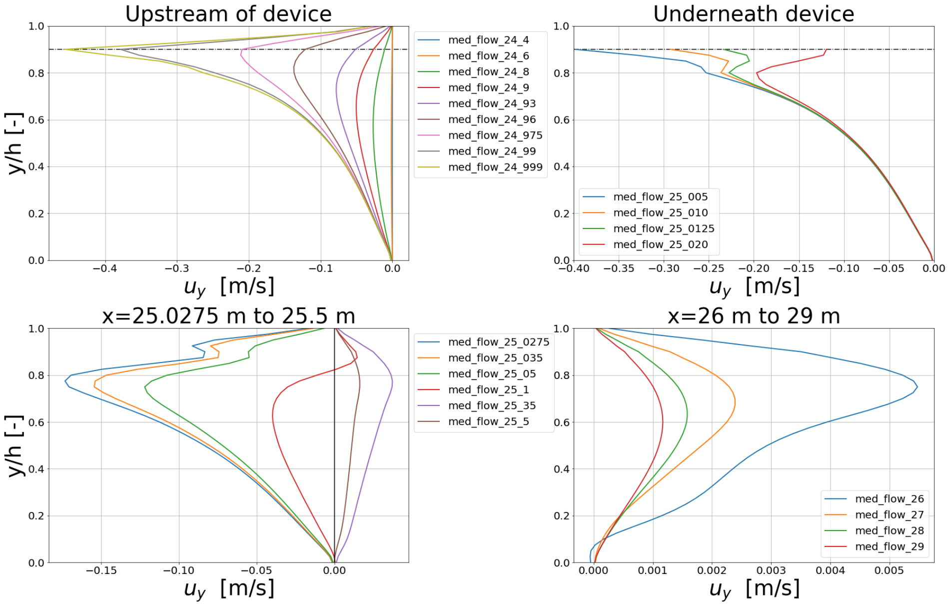


Figure 4.26: Vertical flow velocity profile over the depth at different locations around the interception device for medium flow. The dotted line in the upper right figure indicates the draft of the interception device. This line is also indicated to show the impact of the device w.r.t. flow in the upper left figure. The vertical line in the lower left figure indicates u_y =0 m/s.

4.5.3. Conclusions from hydrodynamics

The contraction upstream of the interception device was expected to be too large to retain a large portion of the particles. The area of the wake in front of the design is limited to 0.20 meters for the different flow cases.

Downstream of the flow a flow separation can be observed, which does not develop back into equilibrium conditions. This length is longer than what is generally Because of this, concentrations downstream of the interception device have been excluded in the further steps.

The encountered maximum downward velocities just upstream of the interception system are visible in Tab. 4.7. The downward velocities for the medium and high flow are of much higher order compared to low flow, which was thought to be of the largest influence in retention of a fraction of the flux near the water surface. The next section explores this finding whether the downward velocity is decisive in the ultimate removal of plastics.

Table 4.7: Maximum downward flow velocity just in front of the system, at x=24.999.

	Low flow	**Medium flow**	**High flow**
Maximum downward velocity [m/s]	0.0827	0.456	0.744

4.6. Interception of particles

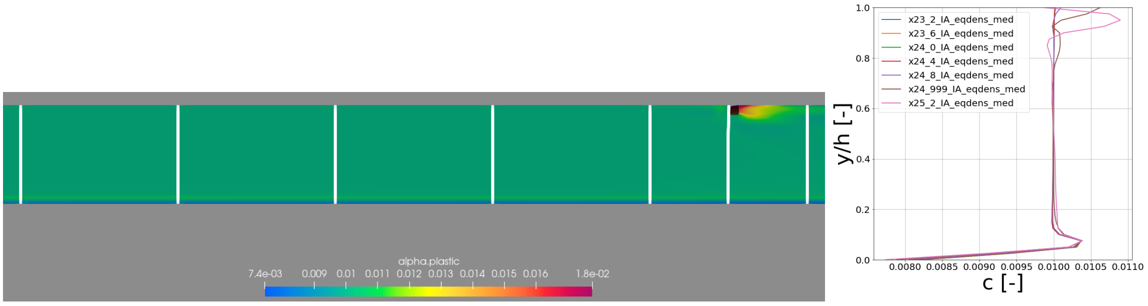
In this section the interception of the plastic particle flux of different particles is described. Firstly the results of the retention of neutrally buoyant particles is discussed. Next, the ability to intercept light particles is handled. Ultimately, fluxes with particles similar to the modified particles of HDPE and PP plastics to represent films were modelled.

4.6.1. Neutrally buoyant particles

Particles with a similar density to the density of water did show much retention by the device. The particles do not experience upward motion and thus can only move downward in front of the device, underneath the interception system. In Fig. E.35, Fig. 4.27 and Fig. E.36 the figures for the interception of these particles for low, medium and high flow are given.

The concentration results for the low and medium flow cases are similar: an increased concentration near the system is encountered, however the retention of these particles is not as high and only occurs very close to the system. The increase of the particles is just 2.5% higher than the imposed entrance concentration of 0.01.

The fraction of high flow near the system is the highest, which is similar to the particle concentration over the depth found from Section 4.2.1. Here the concentration profiles followed the flow velocity profiles as well, with an increased fraction near the water surface. The retention here is 6% larger than the imposed concentration. The concentration over time is plotted in Fig. 4.28. In this figure the spin-up time is shortest for high flow, which is increasing for decreasing flow velocity.



(a) Model results.

(b) Relative concentration over the depth.

Figure 4.27: The concentration of equally buoyant plastic material with profiles taken between x=23.2 m and x=25.2 m for medium flow conditions.

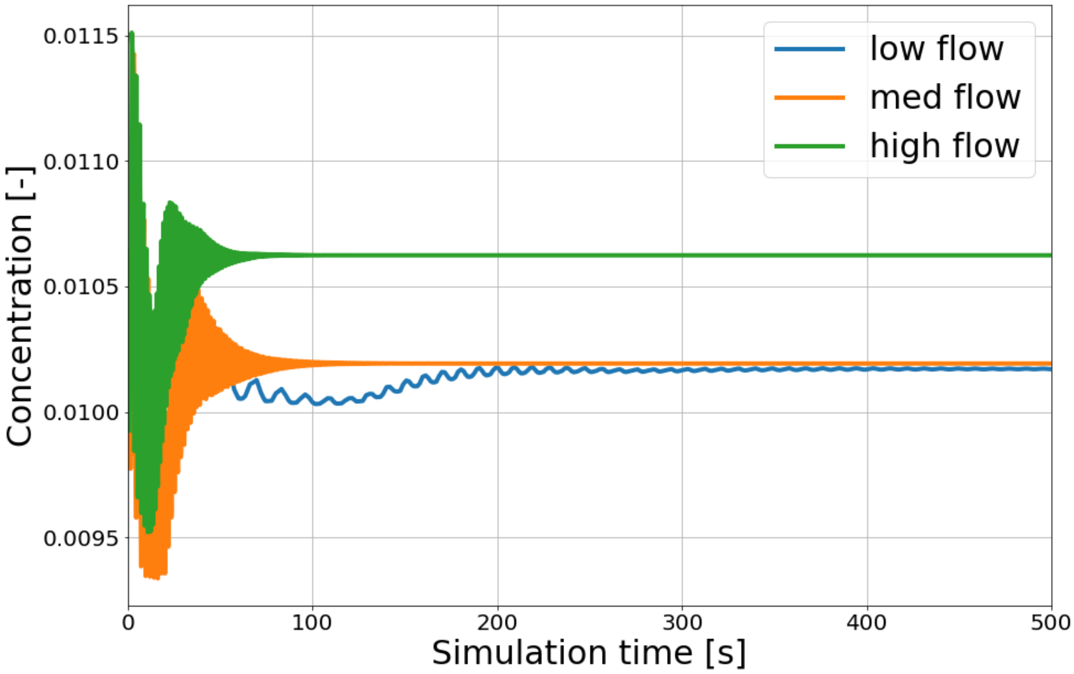
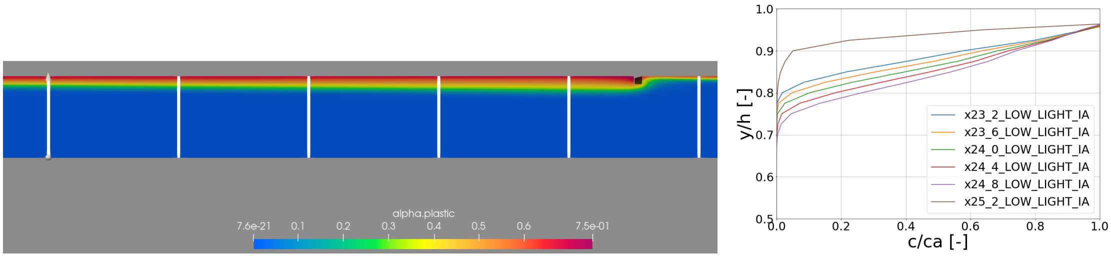


Figure 4.28: Simulation for the concentration just in front of the interception mechanism (x=25 m) over time for neutrally buoyant particles.

4.6.2. Light particles

It can be seen in Fig. 4.29 that for low flow velocities the downward velocity is not high enough to push these particles underneath the interception system. The particles that do get underneath the system, seem to be an end product of the filling up of the area behind the system. Hence, the area is the limiting factor in retention of a plastic flux of light particles.

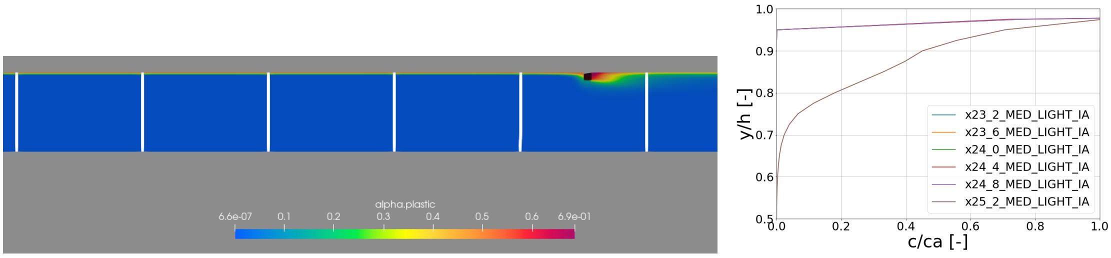


(a) Model results.

(b) Relative concentration over the depth

Figure 4.29: The concentration of the plastic material with profiles taken between x=23.2 m and x=25.2 m for low flow conditions.

This does not occur for the medium and high velocity case, as can be seen in Fig. 4.30 and Fig. E.34. The final concentration in front of the system is very similar and it can be said that the downward flow velocity is the limiting factor in retention of the particles. It is clear that upstream of the system an equilibrium is present, whereas in Fig. 4.29 the concentration near the system is much higher than that of approximately 2 meters upstream of the system. This means that for the retention of other plastic resins with high rise velocities this limit will also be reached.



(a) Model results.

(b) Relative concentration over the depth.

Figure 4.30: The concentration of the plastic material with profiles taken between x=23.2 m and x=25.2 m for medium flow conditions.

Over time this increase of the concentration of the particles is visible in Fig. 4.31. After $t \approx 250$ s, the concentration just in front of the interception device is not increasing and has reached a steady state, with the area filled up with these particles.

The retention of particles for the medium and high velocity case is roughly equal. What is unexpected is that the concentration for light particles near the system is higher near the system for high flow than for medium flow. This could be due to the fact that the particles experience a higher flow velocity for the high flow case and thus reach this point quickly.

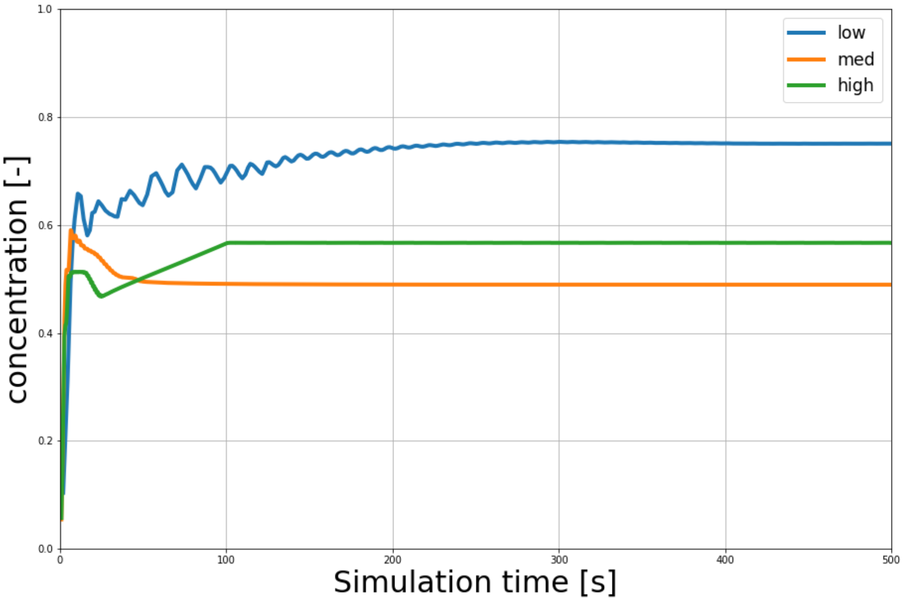
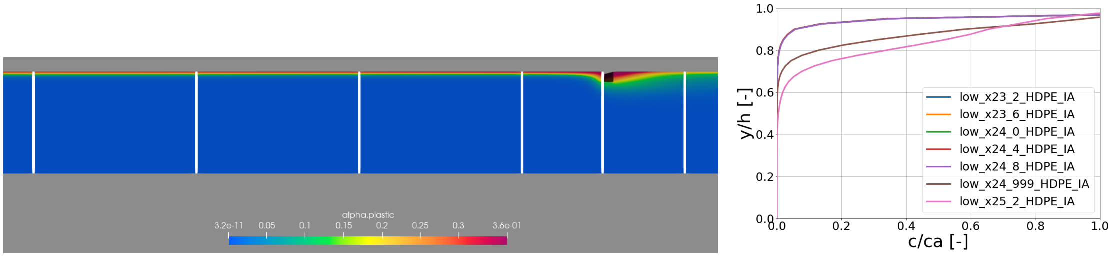


Figure 4.31: Concentration of the particles near the system at x=25 m for light material.

4.6.3. HDPE films

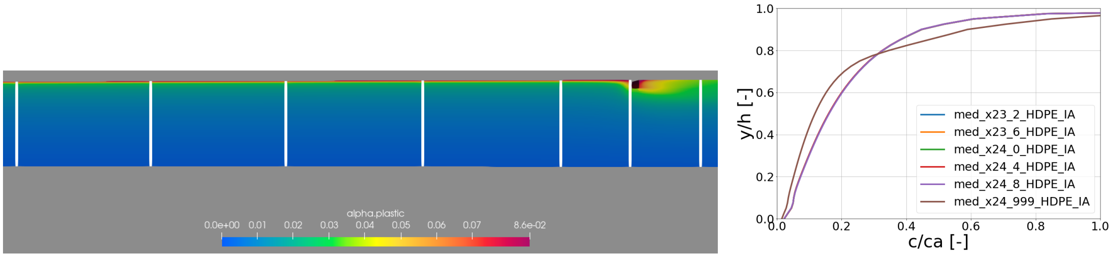
The model results for low flow and the relative concentration over the depth are given in Fig. 4.32. The figure shifts to the right near the interception system, meaning that particles are retained. Similar figures were found for medium and high flow, see Fig. 4.33 and Fig. 4.34 respectively.



(a) Model results.

(b) Relative concentration over the depth.

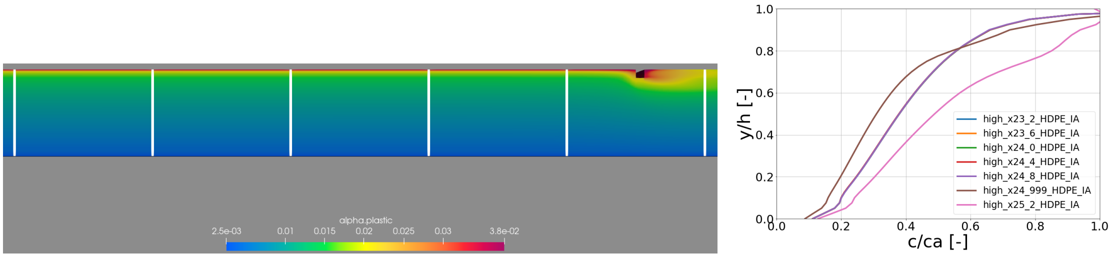
Figure 4.32: The concentration of the HDPE films with profiles taken between x=23.2 m and x=25.2 m for low flow conditions.



(a) Model results.

(b) Relative concentration over the depth.

Figure 4.33: The concentration of the HDPE films with profiles taken between x=23.2 m and x=25.2 m for medium flow conditions.



(a) Model results.

(b) Relative concentration over the depth.

Figure 4.34: The concentration of the HDPE films with profiles taken between x=23.2 m and x=25.2 m for high flow conditions.

Naturally, the highest concentrations in front of the system can be observed from the results for low flow. This further confirms the expectation that for the low flow conditions the retention of the particles is highest. The particle concentration over time is plotted in Fig. 4.35. This difference in retention for the simulations is evident and for the combination of low flow and the same flux of plastics, this system is the most efficient.

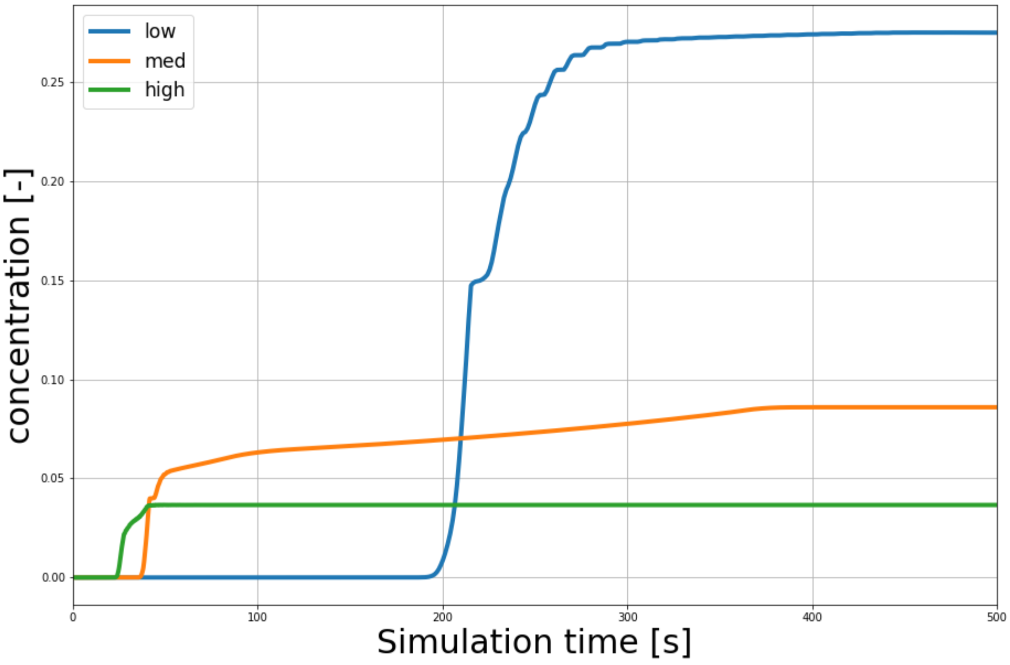
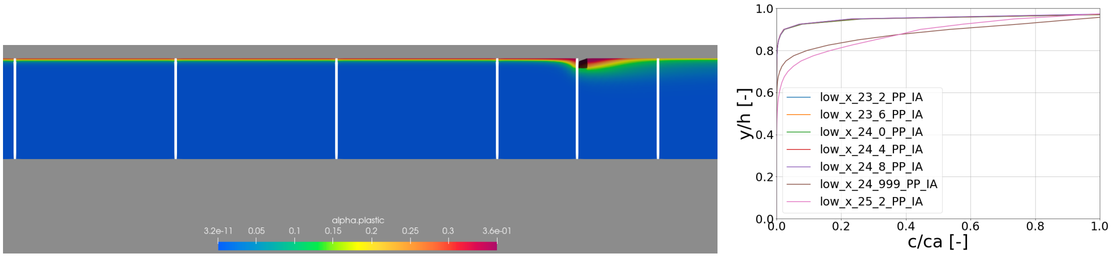


Figure 4.35: Simulation for the concentration just in front of the interception mechanism (x=25 m) over time for HDPE films.

4.6.4. PP films

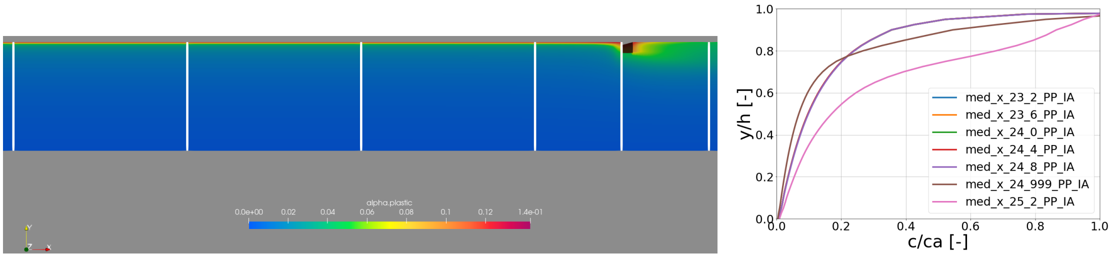
The results for low flow velocity are given in Fig. 4.36, for medium flow in Fig. 4.37 and for high flow in Fig. 4.38.



(a) Model results.

(b) Relative concentration over the depth.

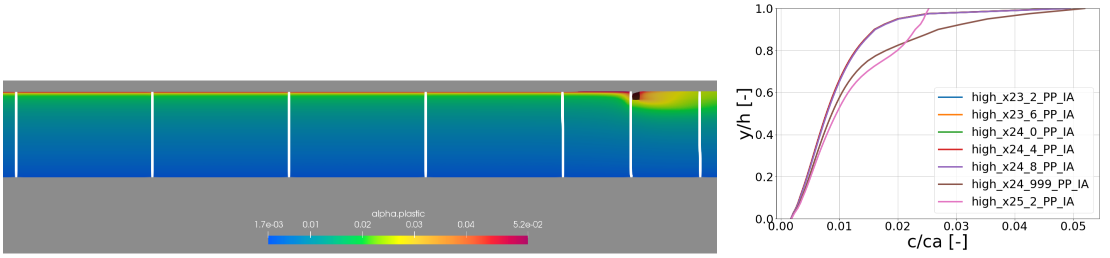
Figure 4.36: The concentration of the PP material with profiles taken between x=23.2 m and x=25.2 m for low flow conditions.



(a) Model results.

(b) Relative concentration over the depth.

Figure 4.37: The concentration of the PP material with profiles taken between x=23.2 m and x=25.2 m for medium flow conditions.



(a) Model results.

(b) Relative concentration over the depth.

Figure 4.38: The concentration of the PP material with profiles taken between x=23.2 and x=25.2 m for high flow conditions.

The particle concentration over time for the different flow velocity cases of PP films are given in Fig. 4.39. The results for the PP retention of films produce roughly similar results as that of HDPE: the retention of that of PP for low flow conditions is highest. Then in decreasing order follow medium and high flow conditions.

The concentrations of the PP resin just in front of the wall are higher than that of the previous simulations of HDPE and thus the rise velocity does influence a certain threshold in the interception of these particles.

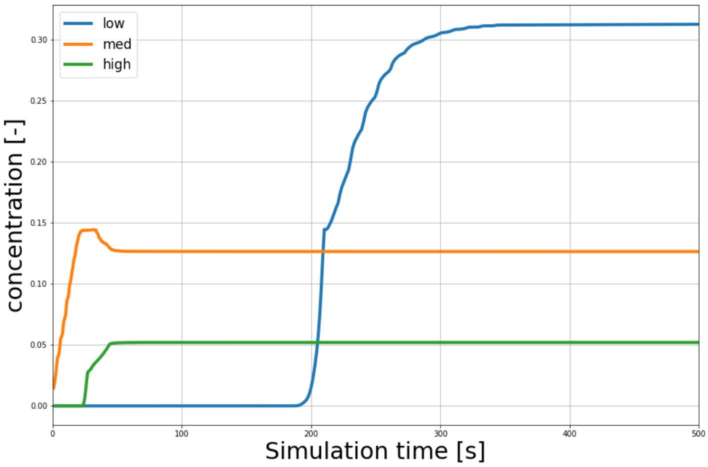


Figure 4.39: Simulation for the concentration just in front of the interception mechanism (x=25 m) over time for polypropylene films.

4.6.5. Final results near the system

The concentrations encountered near the system at the steady state solution are given in Tab. 4.8. From this it is clear that the most particles are retained with lower flow velocities and that the rise velocity is an important factor in the retention of particles, as just more particles are retained that were classified as the PP films, over the HDPE films. In Tab. 4.7 the maximum downward velocities just in front of the interception system are listed and the relation with the encountered downward velocities and the retention of the light particles best tell the capacity of the retention of these particles.

The system is filled up for particles that have a much higher rise velocity than the downward velocity, whereas for the medium flow, that has a downward velocity that is just lower than the rise velocities, the system does not tend to fill up. For the high flow velocity, the downward flow is 1.5 times as high as the rise velocity of the light particles. Thus, these particles flow underneath the system, and only a small portion that is at the wake is retained by the system.

Table 4.8: Final concentrations near the interception system at (x,y) = (25; 0.25).

	Rise velocity [m/s]	**Low flow**	**Medium flow**	**High flow**
Neutrally buoyant particles	0	0.0102	0.0102	0.0106
Light particles	0.136	0.751	0.491	0.520
HDPE films	0.0064	0.275	0.0610	0.0366
PP films	0.008	0.313	0.127	0.0517

The according spin-up times per simulation are given in Tab. 4.9. The spin-up times for high flow are much shorter than low flow. This is because of the fact that the initial conditions have a smaller influence with respect to the increase in flow. The residence time is much shorter. With an increase in flow there also is more exchange of impulse. The flow is inhibited due to this and the friction increases quadratically with a linear increase of the flow, see Eq. 2.2.

Table 4.9: Spin-up time near the interception system at (x,y) = (25; 0.25).

	Rise velocity [m/s]	**Low flow**	**Medium flow**	**High flow**
Neutrally buoyant particles	0	220	100	70
Light particles	0.136	270	60	100
HDPE films	0.0064	400	370	50
PP films	0.008	350	50	50

5

Discussion

Modelling assumptions have been made to represent the flow, plastic particles and the interception device. This study handles a simulation with a distinct profile and clear boundary conditions, propagating into equilibrium flow conditions. Rivers cannot always be described with these imposed boundary conditions.

5.1. Representation of flow

The flow velocity was imposed with the boundary conditions in Tab. 3.3 and a $k-\epsilon$ turbulence model was applied. The flow velocity profiles are uncertain: no physical sensitivity tests have been performed on the interaction between the system and the flow.

5.1.1. Comparison model results with analytical results

The flow velocity profile over the depth deviates from the analytical results for a friction coefficient of 0.003, but is quite similar to that for a friction coefficient of 0.005. This study did not perform a sensitivity analysis on the bed roughness, which can be advisable for future research in order to replicate the flow velocity better. The deviations in flow resulted only in maximum flow velocity differences of 4% and were deemed acceptable in this study. It is uncertain if this deviation will be larger for higher depths or for higher flows velocities.

The results for the turbulent kinetic energy and the turbulent energy dissipation profiles yielded a parabolic profile for the turbulent eddy viscosity. This profile comes close to the analytical profile of this parameter for these flow parameters. The differences from the bottom until halfway the water depth were maximally 5%. However, after the maximum of the turbulent eddy viscosity profile an overestimation was found of maximally 15%.

5.1.2. Near-wall flow

The applied turbulence model was the $k-\epsilon$ turbulence model. For near-wall flow this turbulence model is not commonly applied. The application of the $k-\omega$, $k-\omega$ SST or a LES model could prove to be a better model to resolve the turbulence fields.

5.1.3. Constant plastic flux

A constant flux of plastic particles was assumed for varying flow velocities, whereas in rivers an increasing flux of plastic debris is observed (Van Emmerik et al., 2019; Van Emmerik and Schwarz, 2020) for increasing flow velocities.

When the flow increased it was also found that the plastic material differed in shape. It is plausible that this is due to larger plastic on floodplains got transported with higher discharge. This means that the comparison of the interception of particles with different fluxes could give more ground to compare than the current comparison with an equal flux of particles.

5.2. Representation of plastic particles

The modelling approach has been done with a two-phase Euler model. The plastic phase was modelled as a continuum of particles. The particles were modelled as spheres and an imposed shape factor. The drag for films is expected to differ, compared to how it was implemented in this study.

5.2.1. Euler-Euler approach

The study applied a two-phase Eulerian flow model. In this modelling framework particles are not handled individually, but as a flux of particles. Particle-particle interactions were not considered. However, it is expected that this in fact could be a process with an impact on the final solution, especially when particles are packed (Padding et al., 2015). In laboratory conditions for HDPE films, the trajectories of plastic films were largely influenced by the attachment of plastic films (Zaat, 2020). This behaviour cannot be modelled in the current modelling environment of a two-phase Euler model.

5.2.2. Limitations in particle properties

The only adjustable parameters of the particles in this study were their diameter and density. The density was set following the plastic resin density in tests and the natural environment and the diameter was calculated from the Stokes particle velocity. An inclusion of a particle shape factor or the possibility to include particle dimensions is more preferred for further research: this feature makes a replication of research on rise velocities (Waldschläger and Schüttrumpf, 2019b; Zaat, 2020) easier and more reliable than the modelling of the particles in this way.

This study views four types of particles and two type of films. Uniform distributions of these particles will not be encountered in a river and particles of different shape are expected to enhance the distribution profile considerably. It is necessary to study the different particles such as films first, before generalising these plastic categories to a final distribution.

A Schmidt number of 1 was assumed. In reality the turbulent eddy diffusivity profile is non-parabolic and thus the Schmidt number not equal to 1 (Liu, 2016c). Concerning the distribution of the particles this means that the debris will be more in suspension than originally assumed.

5.2.3. Applied drag model

The concentration profiles showed deviations both near-surface and in suspension from the Rouse profile. The distribution along the vertical was mainly influenced by choosing a specific drag model and thus the dimensions of the plastic body influences the plastic distribution considerably.

The general drag models available in OpenFOAM for the Euler-Euler approach do not include a drag force for a non-spherical flux of particles. From a physical perspective it would better to be able to adjust more particle properties and impose a shape factor.

As mentioned in Section 2.2.2 the drag comprises form drag and friction drag. The difference between spherical bodies and films is that the orientation of the body matters for films, whereas it does not matter for a spherical body. This means that a larger drag force is exerted on the plastic film compared to a spherical particle. It is especially important for the friction drag to include a shape factor or add the different dimensions of the plastic debris, as the orientation of the particle has a large influence on the amount of friction drag that the body experiences.

Films have a lower rise velocity because of this difference in friction drag. This in turn means that particles are kept more in suspension and that the Rouse profile will move to the right, as per observations by Zaat (2020).

5.3. Representation of interception system

The interception system was modelled as an wall with a specific draught and a width. The wall is not floating but is set at a fixed position in the water and has sharp edges. The rigid-lid assumption for contraction of flow is not valid for high flow conditions in this specific setup.

5.3.1. Contraction of flow

The system is modelled as a block with perpendicular edges. These sharp edges cause a strong wake and flow separation because of the curved streamlines that occur in the flow, see Section 4.5. When these walls would be more slanted or curved, the flow separation is expected to be less (Filippini et al., 2005) and in reality the arms of the designs are shaped with more circular tubes, see Fig. 2.6.

As the system is not floating, but fixed for a specific water depth and draught, the contraction is larger for this design than for a floating system, impacting the dynamics of the particles around the structure as well. A floating object furthermore causes a displacement of water when deployed, which can be directly determined from Archimedes' principle. An implementation of a floating object into a model has been performed in Delft3D (Deltares, 2021) and tested in OpenFOAM as well (Hatari Labs, 2018). The complexity of the model rises much compared to the current model, with the necessity to introduce a dynamic mesh when approximating the system as a floating object. The floating object also minimally requires parameters for geometry, density, centre of inertia, degrees of freedom, which need to be evaluated for the specific design in Fig. 2.6.

A contraction coefficient for vertical sluice gates is considered to be a value between 0.598 and 0.611 (Yen et al., 2001). For practical purposes this value is usually selected as $C_c = 0.61$ (Henderson, 1966). A more slanted or oblique interception mechanism in its turn could suppress the degree of contraction and ultimately the downward velocities near the system. However, this would also decrease the area at which particles can get retained. It is unsure if there is an ideal design of obliqueness or that the sharp edge performs the best in terms of retention.

5.3.2. Rigid-lid assumption

The rigid-lid assumption for underflow of an object has its implications and regarding the water level this assumption does not hold. Due to the fact that a water level is set, a pressure variation in front of and behind the wake is generated. This can be seen in Fig. 5.1. This problem arises for the different cases, but is only illustrated for the medium flow case. In reality a water level difference would occur just in front of the system, where the water level is of the same height as that of the energy head. From the velocity head a water level difference in front of the system can be determined:

$$H_1 = \frac{u_1^2}{2g} + h_1 \tag{5.1}$$

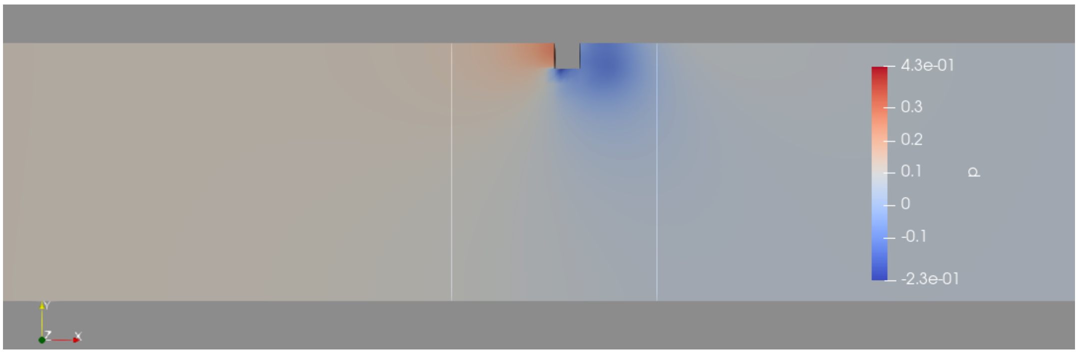


Figure 5.1: Pressure around plastic interception system. Indicated are x=24.9 m and x=25.1 m.

For the different flow velocities, the water level just in front of the system can be said to be equal to H_1, which are listed in Tab. 5.1. The rigid-lid model is only valid if the velocity head is smaller than that of 10% of the water depth, which is not the case for the high flow velocity case.

Table 5.1: Approximate water level just in front of the system for submerged flow and a free-surface model, calculated from the energy head, situated at x=25 m in the model.

Inlet flow velocity [m/s]	**u = 0.10**	**u = 0.55**	**u = 0.90**
Water level at the system [m]	0.251	0.265	0.291

Also, it is expected that behind the system a lower water level would be observed with a small hydraulic jump also present. This is due to the fact that there is a head loss due to flow, similar to a sluice gate or underflow for a structure (Swamee, 1992; Oskuyi and Salmasi, 2012).

5.3.3. Reference study submerged gate analogy

A reference study for submerged gate flow was performed whether the analogy for an submerged gate flow holds. The sensitivity study is performed on the horizontal and vertical velocities generated by the model simulation and if these are in line with earlier hydrodynamic results.

The results of the reference study agree very well with the experimental results of Finnie and Jeppson (1991) and the computational methods by Montes (1997) and Shammaa et al. (2005) The horizontal and vertical velocity values just underneath the gate structure and in front of the structure (x=a) are given in Fig. 5.2. The $k - \epsilon$ provides an accurate enough prediction of the flow near the wall for underflow situations in OpenFOAM.

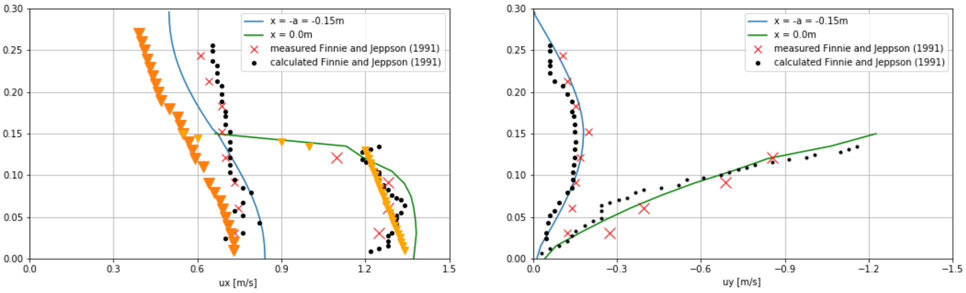


Figure 5.2: Horizontal and vertical velocity results from the reference case of Finnie and Jeppson (1991).

5.3.4. Absence of plastic collection system behind the structure

In this study 3D effects have not been considered. The system was assumed to be perpendicular to the flow without a location of collection. In reality this interception component is tilted with respect to the streamwise component and guides debris towards the river bank, where a collection cage or container is placed. In the model a final value of retention is reached and the model ”clogs” and rolls out a carpet of debris against the streamwise direction.

5.4. Application of model results to field conditions

The model was set to be long enough and simulated for a certain time to come to a steady state simulation, with a constant flux and geometry. In rivers, input of external factors, a variance in plastic material properties and more complex geometries are present. This section mentions to what extent this model could have been extended on and the relevance of this project. The flow velocity profiles in rivers cannot always be described with equilibrium conditions due to constant changes in the river profile.

5.4.1. Absence of external forcing effects

External effects were not taken into account in this study. More complications rise up with the addition of an extra phase or forcing effects. This behaviour has been analysed in other studies by Kooi et al. (2018b) and Van Welsenes (2019) and these results indicate that the effect of wind increases cross-sectional displacement and moves particles towards the river banks, resulting in beaching of the material. However, Kooi et al. (2018b) is applied in oceanic environments and Van Welsenes (2019) applies no validation study.

5.4.2. Variety of plastics in the natural environment

The plastic particle phase in this study considers four different types of particles. In reality a large variation is present, with a plastic distribution on the basis of shape, size and density constantly changing over the river length (Waldschläger and Schüttrumpf, 2019a). There lie many uncertainties in the current motions of plastics, with numerous processes influencing the physical plastic properties. This in its turn impacts the distribution of plastic material.

It is unclear what the interactions are with the plastic debris of different size and shape and if these form clusters of material, which is difficult to predict in a model. Further exploration in the behaviour of plastics is necessary to come up with an system to represent a batch of different plastics. In the current Eulerian-Eulerian model it was not possible to model particle-particle interactions. This can be incorporated by adding lines of code in the solver applied, or another two-phase solver needs to be used which has these processes incorporated already.

5.4.3. Influence of river geometry on concentration profile

The river shape, bends and structures largely influence the distribution of sediments in the vertical and in the horizontal direction and probably that of plastics as well. Van Welsenes (2019) found that the cylindrical flow in river bends and the difference in flow velocity between the inner and outer bend caused plastics to move outward. Furthermore, the flow of a river changes in time and does not follow a long period with a constant discharge. The Rouse profile assumes that the flow segment has a uniform shape and a constant discharge. It is probable that the flow velocity profile always shows differences with the theory, especially in the horizontal, as described in the previous section.

The available bed roughness parameters were the Nikuradse roughness height and a roughness constant. The latter is a number that can vary between 0.5 and 1.0. When this constant is taken as 0.5 the bed is uniform and constant and for 1.0 this means that the bottom is very rough. In reality the bed is non-uniform and shows ripples and dunes in the streamwise component, whereas river cross sections typically have a U or V-shaped form rather than a rectangular shape. The 2DV results represent a slice of the flow conditions in the width. The flow varies in cross-directional component with respect to the streamwise component, due to friction and a lower depth at the river banks. What this means for the particle distribution is that the concentration of the plastic flux near the river bank is lower than in the middle of the river, due to the decrease in river flow.

The bottom friction coefficient of the simulation is taken as a constant 0.003, whereas in riverine areas the bed is rougher and non-uniform. The results of the flume experiment by Zaat (2020) used a scaling effect of $c_f = 0.005$, with a uniform bed. A higher bed friction coefficient causes an increase in the Nikuradse roughness height, which causes an increase in the bed friction velocity. Ultimately, this means that the turbulent eddy viscosity results in a larger maximum, with more turbulent diffusion as a result. This can be seen from Eq. 2.17 as well: the Rouse number decreases with increasing bed friction velocity and thus leads to an equilibrium profile that shows more uniformity halfway the water column.

6

Conclusions and recommendations

A main research question together with the objective was established in Chapter 1. In this chapter the research question and the sub-questions from Section 1.2 are answered with the results of Chapter 4. Following the conclusions, recommendations for further research are given to further explore into the unexplored territories of research into plastics in riverine areas. These recommendations are important to understand plastic particle trajectories better and ultimately intercept plastic debris near the water surface.

6.1. Conclusions

This project applies the numerical software tool of OpenFOAM. The first stages of this study consisted of a verification and validation study for the flow and plastic concentration. A Eulerian-Eulerian framework for the two-phase model was applied for the flow of plastic particles in a straight reach. The obtained validation results agreed well with the inverted Rouse profiles for higher Reynolds numbers. The aim of the project was to investigate the interception of these particles. With the conditions and design in this project a build-up in particle concentration near the system was observed, indicating interception of material. The main research question reads:

How can plastics be computationally modelled and captured by an interception system in a riverine reach for different flow velocities?

The sub-questions are answered one by one first. Together, these sub-questions answer the main question. Then, an item is set that further expresses the applicability of the final result.

a. ***How can a channel with a steady flow and flux of plastic debris and an interception system be modelled in a CFD programme?***

The channel was modelled with a 2DV $k - \epsilon$ turbulence model. A rectangular grid was set up with rectangular grid cells. In the x-direction there was an equidistant grid of 0.05 m wide and over the depth the grid size was 0.00625 m. The flow was described as steady, with a uniform inlet flow velocity imposed. This flow velocity profile developed to a logarithmic velocity profile over the vertical. The entrance length was assessed for the different flow velocities and plastic particles. This development to a log-flow velocity profile was induced by a no slip boundary at the bottom and slip boundary at the water surface. A rigid-lid model was applied.

The results of the flow without plastic debris was a log-flow velocity profile that gives an accurate representation of the flow, deviating little from the theoretical profile that was expected for uniform flow conditions, with a mean flow velocity difference of less than 1% for the different flow cases. Compared to the maximum flow velocities, thus the velocity near-surface, a 3-4% difference was found. These numbers are considered to be too small to have a large impact on the results on the concentration profiles, with its impact on the resulting turbulent eddy viscosity profile being of parabolic shape.

The turbulent eddy viscosity showed a slight deviation from the analytical profile, with its maximum just higher than half of the depth for the different flow cases at 0.135 m, see Fig. E.5, Fig. E.7, Fig. 4.7 for the different flow cases. The maximum value itself showed agreements with theory desired from a point of view for the diffusion of particles. Furthermore, the spin-up time needs to be high enough. Spin-up times were found to be in the order of regarded and can take much longer for lower flow velocities than for higher flow velocities, when comparing the same geometry and time steps.

A constant flux of particles was assumed, which was modelled with a two-phase Eulerian approach. A fraction of the total volume (1.00) in the model is designated as particles (0.01), whereas the other fraction is water (0.99) at the inlet. These fractions both shared the same pressure field. The particles were modelled as spheres, following kinetic theory. The particle parameters that could be modelled were the diameter, density and the interphase momentum exchange forces between the two different phases. A sensitivity study on the coefficients between the phases was performed. The appropriate drag model for medium and high flow was found to be the Gibilaro drag model. For the low flow the particle concentrations differed around 39%, which is a considerable difference compared to what the medium flow and high flow deviated. For the medium flow a maximum under- and overestimation in the order of 10-20% was found, whereas the high flow concentration profiles differed 10%. The resulting values were for the virtual mass, turbulent dispersion and the lift force were found to be $C_{vm} = 0.5$, $C_{td} = 1.0$ and $C_l = 1.6$ and a Gibilaro drag model.

The interception device is modelled as a square with perpendicular edges of 10% of the depth (0.025 x 0.025 m^2). The system was modelled with no slip walls.

b. ***What is the required distance for both turbulent flow and that of particles to reach an equilibrium profile?***

In this study the necessary length to reach an equilibrium flow velocity profile was 60 times the depth, $60 \cdot h$, and this length was similar for the low, medium and high flow velocity cases. The entrance length for for the plastic concentration was dependent on the inlet flow velocity and the rise velocity of the particles. It was seen that for PP particles the entrance length was considered shorter than that for HDPE particles. A plausible reason is that is because of their difference rise velocity: the lighter material had a shorter entrance length.

When the flow increased, the entrance length for the plastic particles decreased. More mixing is present for higher flow velocity and the travel distance for the particles to reach the equilibrium is mainly influenced by their rise velocity. When modelling one should keep in mind that this travel distance can change considerably, indicated by the collection of numerous studies on the entrance length for flow (Bonakdari et al., 2014).

c. ***How can a concentration profile similar to the results of Zaat (2020) for the plastic flux be replicated?***

Two different plastic resins were tested with a specific set of model coefficients and a representative diameter and density that agrees with spherical particles that follow the inverted Rouse profile. It was seen that for the HDPE resin the model returns a distribution that agrees with the inverted Rouse profile for these parameters. Furthermore, the applicability of the diameter times the coefficient of $\sqrt{\alpha} = 0.8$ delivers a concentration profile similarly as a shift of the theoretical profile.

The simulations of the polypropylene material and the release of films also agreed with the Gibilaro drag model, but not for the lower flow velocity. It shows that the application of this drag model is not always valid to replicate particle flow in equilibrium conditions. The application of PP films needs to be verified from physical tests, as the outcome of these tests could not be verified.

In the model the transformation from spherical items to films has been performed by only changing the diameter, which in its turn changed the rise velocity towards this modified profile. A Gibilaro drag model has been applied, which has been established for sediment-liquid flow, especially for fluidised beds.

It moreover shows that these differences come from the fact that drag models have been developed for a limited number of experiments and that some aspects of CFD modelling cannot be taken as universally applicable.

d. ***How can the relationship between the flow in front of an interception system and that of plastic properties be defined in terms of retention of a plastic flux?***

The particles that were retained the most, were the rise velocities of the light particles and show that debris of this density is the easiest to remove from a river. This is confirmed by the distributions encountered in natural environments from tests in rivers with Manta nets or samplers (Sadri and Thompson, 2014). When the particles go up faster than the downward flow velocity near the system, these particles cannot be intercepted. The interception device showed retention values, of which the numbers can be found in Tab. 4.8. Particles that had the highest rise velocity for the lowest downward conditions did not flow underneath the system, until a certain moment in time. This moment occurred when the constant flux of particles filled up the area of particles where the ”carpet” had filled up and consequently these particles were transported downstream. This phenomenon did not occur for other flow conditions and for other particles released.

When the flow velocity increased, the downward velocities in front of the system also increased. This affected the retention in a negative way, when comparing this retention of a constant flux. It is however expected that the particle concentration increases for increasing flow velocity and that in this way more debris could be captured, as in reality for an interception system the flow velocities can be come so high that the particles do not have time to flow underneath the arms or guidance wall but are collected in their catchment area.

The interception of neutrally buoyant particles was negligible, with on average a 4% increase in the field near the approach compared to the average concentration value. It can be seen from the retention for neutrally buoyant particles that the concentration for high flow is highest near the system. This can be explained from the velocity profile, with its maximum flow being much higher of that of medium and low flow.

- **Applicability of final result**

The final Euler-Euler modelling approach provided a rapid assessment to replicate the inverted Rouse profile and also replicate HDPE films with an imposed shape factor. It can provide results quickly for engineers interested in testing or other designs or other plastic particles with the current design tested.

The model results can help understand the trajectories of particle fluxes better. It explains how the particle dynamics between a wall and the flow around it occurs. This problem is scaled as a small flume but the essence is similar of that of a project with a larger scale.

Currently, there are exploratory studies available on better understanding the plastic dynamics and the bringing together of more studies on this topic will help to ultimately remove debris in rivers of small and larger size.

The study by Carleton and Nielsen (1990) is further confirmed with this thesis that the interception design works for very light particles and especially particles near-surface. Particles in suspension however do not show retention, as these particles will not get stirred up towards the interception device.

6.2. Recommendations

Recommendations for further research are to perform more physical tests, implement a non-spherical drag model applicable for plastic material and to expand the complexity of the current model. The addition of physical tests would benefit the knowledge into the plastic behaviour the greatest: the expansion of complexity of the model is only possible with a larger availability of these physical model tests.

6.2.1. Physical model tests

Considering the different behaviour of the different plastic materials such as films, closed or open bottles, bags, fragments, it is difficult to represent these materials into a model. The release of different plastic materials similar to Zaat (2020) can bring additional understanding into plastic paths and make it easier to validate this model for other material as well. Now the only validation study performed is done with one physical model. Better understanding into plastic particle trajectories will remove uncertainties related to the modelling results. It can give more insight if a modified and inverted Rouse profile for other buoyant plastic particles holds or whether the distribution should be approached from another curve, such as the Gaussian curve (Sundby, 1983).

6.2.2. Application of three-phase model

This study considers the flow in a rectangular segment and an interception system with the rigid-lid model. It is expected that a solver that does not negate water level differences may better represent the plastic dynamics in a next stage, as the water level is expected to increase with more than 10% near the interception device. The shape of the wake could change considerably and thus the outcome of the retention of plastic particles by this system.

This solver could be a three-phase solver without a rigid-lid. These three-phases would then be: water, plastic and air. In OpenFOAM these solvers can be either three-phase Euler or a partially Euler and Lagrangian solver, with the plastic material being described from either a Lagrangian or Eulerian perspective. In OpenFOAM these possibilities lie in `multiphaseEulerFoam`, which is an expansion of the current model, or `interMixingFoam`, which derives its basics from a volume of fluid approach with two of the phases being miscible. Another option is the use of `multiphaseInterFoam`, which also is based upon the volume of fluid method and a mixture model.

6.2.3. 3D model

The project did not consider friction in the direction perpendicular to the streamwise component, while this is expected to influence the concentration of plastic material over the width. From the results a fill-up of plastic particles in front of the interception system was observed and particles went underneath the system for higher flow velocities. If 3D effects are considered and a collection system is placed behind the device, then a better assessment can be made of the trapping of particles. A clear view on the trapping efficiency can help engineers in making a better interception system design.

6.2.4. Application of non-spherical drag model

It is advisable to apply a non-spherical drag model that not only time averaged agrees with the motion of films in the water column and to further explore the applicability of different drag models for the motion of different plastic materials.

A non-spherical drag model can be added into the CFD toolbox of OpenFOAM in order to capture the form and friction drag of irregularly shaped particles. A non-spherical drag force is available in the Lagrangian module of OpenFOAM and is based on the equations found by Haider and Levenspiel (1989). An average error for this drag model is found to be 21.5% (Chhabra et al., 1999). Research was undertaken to find a coherent drag coefficient and model for plastic fragments and other types of material (Waldschläger and Schüttrumpf, 2019b). Average errors here were also found to be in the same order, but can give an indication in the trajectories of a plastic flux for steady flow conditions.

Exploration into the implementation of a non-spherical drag model has been performed by Andric (n.d.), wherein a step by step procedure in the application of such a model is done. A replication of this study applied for plastic material, together with a physical test can show the validity of such a drag model, hereafter this could be tested on larger scale models. The drag exerted on these particles can be better resolved with such a drag model. Ultimately specific systems can be designed for areas where specific plastic debris found.

6.2.5. Sensitivity analysis on turbulence model with respect to flow

The $k-\epsilon$ turbulence model is one of the more basic turbulence models to resolve the wakes and slurs of the flow. The $k-\omega$ model might resolve the flow better near the wall, thus a sensitivity study for this turbulence model or a $k - \omega$ Shear Stress Transport ($k - \omega$ SST) model should be performed to see if this is the case. It is unsure if the differences found in flow velocity between the analytical results and the computational model impact the concentration profile and subsequently the retention of particles by the system. This needs to be further assessed with other turbulence models to improved the predictability of plastic dynamics and ultimately the removal of plastic debris.

Bibliography

S. Akimi, 2010. Dispersed Flow. `https://thermopedia.com/content/5/`.

Allseas, 2021. Catchy waste collection system. `https://allseas.com/project/catchy/`.

A. Ammala, S. Bateman, K. Dean, E. Petinakis, P. Sangwan, S. Wong, Q. Yuan, L. Yu, C. Patrick, and K. Leong. An overview of degradable and biodegradable polyolefins. *Progress in Polymer Science*, 36(8):1015 – 1049, 2011. ISSN 0079-6700. doi: https://doi.org/10.1016/j.progpolymsci.2010.12.002. URL `http://www.sciencedirect.com/science/article/pii/S0079670010001267`. Special Issue on Biomaterials.

A. L. Andrady. Microplastics in the marine environment. *Marine Pollution Bulletin*, 62(8):1596 – 1605, 2011. ISSN 0025-326X. doi: https://doi.org/10.1016/j.marpolbul.2011.05.030. URL `http://www.sciencedirect.com/science/article/pii/S0025326X11003055`.

J. Andric. CFD with OpenFOAM software , Lagrangian particle tracking. Technical report, Chalmers University of Technology, n.d.

Ansys, 2009. FLUENT User Guide. `https://www.afs.enea.it/project/neptunius/docs/fluent/html/ug/node1.htm`.

D. Apsley, 2009. Structure of a turbulent boundary layer. `https://personalpages.manchester.ac.uk/staff/david.d.apsley/lectures/turbbl/regions.pdf`.

N. Badano, M. Sabarots Gerbec, M. Re, and A. Menéndez. A coupled hydro-sedimentologic model to assess the advance of the parana river delta front. volume 1, 09 2012.

A. Ballent, S. Pando, A. Purser, M. F. Juliano, and L. Thomsen. Modelled transport of benthic marine microplastic pollution in the nazaré canyon. *Biogeosciences*, 10(12):7957–7970, 2013. doi: 10.5194/bg-10-7957-2013. URL `https://bg.copernicus.org/articles/10/7957/2013/`.

H. Bonakdari, G. Lipeme-Kouyi, and G. Lal Asawa. Developing turbulent flows in rectangular channels: A parametric study. *Journal of Applied Research in Water and Wastewater*, 1(2):51–56, 2014. ISSN 2476-6283. URL `https://arww.razi.ac.ir/article_52.html`.

H. Burak, 2015. Modification of a computational fluid dynamics model (ansys-fluent) for the purpose of river flow and sediment transport modeling. Master's thesis, the Graduate School of Engineering and Sciences of İzmir Institute of Technology.

A. Burns, T. Frank, I. Hamill, and J.-M. Shi. The Favre Averaged Drag Model for Turbulent Dispersion in Eulerian Multi-Phase Flows. volume 392, 01 2004.

E. Burns and A. Boxall. Microplastics in the aquatic environment : Evidence for or against adverse impacts and major knowledge gaps. *Environmental Toxicology and Chemistry*, 37:2776–2796, 09 2018. doi: 10.1002/etc.4268.

S. Busch. A twophaseEulerFoam tutorial. Technical report, Chalmers University of Technology, 2015.

S. Calvert. Baltimore harbor's garbage-gobbling tool gains speed. *Wall Street Journal*, 2019. URL `https://www.wsj.com/articles/baltimore-harbors-garbage-gobbling-tool-gains-speed-11574611200`.

C. Campbell, 2016. Rank record: Mr. trash wheel gathers 1 millionth pound of trash from jones falls.

L. S. Caretto, A. D. Gosman, S. V. Patankar, and D. B. Spalding. Two calculation procedures for steady, three-dimensional flows with recirculation. pages 60–68, Berlin, Heidelberg, 1973. Springer Berlin Heidelberg. ISBN 978-3-540-38392-5.

M. G. Carleton and J. S. Nielsen. A Study of Trash and Trash Interception Devices. *Water Science and Technology*, 22(10-11):287–290, 10 1990. ISSN 0273-1223. doi: 10.2166/wst.1990.0318. URL `https://doi.org/10.2166/wst.1990.0318`.

CFD Online, 2012. Staggered grid. `https://www.cfd-online.com/Wiki/Staggered_grid`.

S.-M. Chang, H. Kim, J. H. Shin, and Y. Kim. The feasibility of multidimensional cfd applied to calandria system in the moderator of candu-6 phwr using commercial and open-source codes. *Science and Technology of Nuclear Installations*, 2016:1–10, 01 2016. doi: 10.1155/2016/3194839.

J. Chauchat, Z. Cheng, T. Nagel, C. Bonamy, and T.-J. Hsu. Sedfoam-2.0: a 3-d two-phase flow numerical model for sediment transport. *Geoscientific Model Development*, 10(12):4367–4392, 2017. doi: 10.5194/gmd-10-4367-2017. URL `https://gmd.copernicus.org/articles/10/4367/2017/`.

Z. Cheng, T.-J. Hsu, and J. Calantoni. Sedfoam: A multi-dimensional eulerian two-phase model for sediment transport and its application to momentary bed failure. *Coastal Engineering*, 119:32–50, 2017. ISSN 0378-3839. doi: https://doi.org/10.1016/j.coastaleng.2016.08.007. URL `https://www.sciencedirect.com/science/article/pii/S0378383916301958`.

R. Chhabra, L. Agarwal, and N. Sinha. Drag on non-spherical particles: an evaluation of available methods. *Powder Technology*, 101(3):288–295, 1999. ISSN 0032-5910. doi: https://doi.org/10.1016/S0032-5910(98)00178-8. URL `https://www.sciencedirect.com/science/article/pii/S0032591098001788`.

D. Coles. The law of the wake in the turbulent boundary layer. *Journal of Fluid Mechanics*, 1:191–226, 1956.

COMSOL, 2020. Inlet values for the turbulence length scale and turbulent intensity. `https://doc.comsol.com/5.5/doc/com.comsol.help.cfd/cfd_ug_fluidflow_single.06.095.html`.

D. O. Conchubhair, D. Fitzhenry, A. Lusher, A. L. King, T. van Emmerik, L. Lebreton, C. Ricaurte-Villota, L. Espinosa, and E. O'Rourke. Joint effort among research infrastructures to quantify the impact of plastic debris in the ocean. *Environmental Research Letters*, 14(6):065001, 2019. doi: 10.1088/1748-9326/ab17ed. URL `https://doi.org/10.1088%2F1748-9326%2Fab17ed`.

S. Cristo, 2021. Error: Mesh Too Coarse for the GAMG Solver . `https://www.cfd-online.com/Forums/openfoam/78273-gamg-hexa-vs-tetrahedron-meshes.html`.

Dalberg Advisors, W. de Wit, and N. Bigaud. No plastic in nature: assessing plastic ingestion from nature to people. Technical report, WWF, 2019.

M. Davidson. Numerical calculations of two-phase flow in a liquid bath with bottom gas injection: The central plume. *Applied Mathematic Modelling*, 14, 1990.

R. B. Dean. Reynolds Number Dependence of Skin Friction and Other Bulk Flow Variables in Two-Dimensional Rectangular Duct Flow. *Journal of Fluids Engineering*, 100(2):215–223, 06 1978. ISSN 0098-2202. doi: 10.1115/1.3448633. URL `https://doi.org/10.1115/1.3448633`.

Deloitte. The price tag of plastic pollution. `https://www2.deloitte.com/nl/nl/pages/strategy-analytics-and-ma/articles/the-price-tag-of-plastic-pollution.html`, 2019.

Deltares. Delft3D-FLOW User Manual. 3D/2D modelling suite for integral water solutions. Technical report, Deltares, 2021.

D. Drew, L. Cheng, and R. Lahey. The analysis of virtual mass effects in two-phase flow. *International Journal of Multiphase Flow*, 5:233–242, 1979.

D. A. Drew. Mathematical modeling of two-phase flow. *Annual Review of Fluid Mechanics*, 15(1):261–291, 1983. doi: 10.1146/annurev.fl.15.010183.001401. URL `https://doi.org/10.1146/annurev.fl.15.010183.001401`.

H. Enwald, E. Peirano, and A.-E. Almstedt. Eulerian two-phase flow theory applied to fluidization. *International Journal of Multiphase Flow*, 22:21–66, 1996a. ISSN 0301-9322. doi: https://doi.org/10.1016/S0301-9322(96)90004-X. URL `https://www.sciencedirect.com/science/article/pii/S030193229690004X`.

H. Enwald, E. Peirano, and A.-E. Almstedt. Eulerian two-phase flow theory applied to fluidization. *International Journal of Multiphase Flow*, 22:21–66, 1996b. ISSN 0301-9322. doi: https://doi.org/10.1016/S0301-9322(96)90004-X. URL `https://www.sciencedirect.com/science/article/pii/S030193229690004X`.

M. Fairs. The Ocean Cleanup labelled ”a dream that seduced many people”. *Dezeen*, 2019.

G. Filippini, G. Franck, N. Nigro, M. Storti, and J. D’Elía. Large eddy simulations of the flow around a square cylinder. 11 2005.

J. Finnie and R. Jeppson. Solving turbulent flows using finite elements. *Journal of Hydraulic Engineering-asce - J HYDRAUL ENG-ASCE*, 1991. doi: 117:1513-1530.

A. Ghione, 12 2012. Development and validation of a two-phase cfd model using openfoam. Master’s thesis, KTH Royal Institute of Technology.

L. Gibilaro, R. Di Felice, S. Waldram, and P. Foscolo. Generalized friction factor and drag coefficient correlations for fluid-particle interactions. *Chemical Engineering Science*, 40(10):1817–1823, 1985. ISSN 0009-2509. doi: https://doi.org/10.1016/0009-2509(85)80116-0. URL `https://www.sciencedirect.com/science/article/pii/0009250985801160`.

D. González, G. Hanke, G. Tweehuysen, B. Bellert, M. Holzhauer, A. Palatinus, P. Hohenblum, and L. Oosterbaan. Riverine Litter Monitoring - Options and Recommendations. Technical report, Joint Research Centre, 2016.

V. Gumus, O. Simsek, N. G. Soydan, M. S. Akoz, and M. S. Kirkgoz. Numerical modeling of submerged hydraulic jump from a sluice gate. *Journal of Irrigation and Drainage Engineering*, 142(1):04015037, 2016. doi: 10.1061/(ASCE)IR.1943-4774.0000948. URL `https://ascelibrary.org/doi/abs/10.1061/%28ASCE%29IR.1943-4774.0000948`.

A. Haider and O. Levenspiel. Drag coefficient and terminal velocity of spherical and nonspherical particles. *Powder Technology*, 58(1):63–70, 1989. ISSN 0032-5910. doi: https://doi.org/10.1016/0032-5910(89)80008-7. URL `https://www.sciencedirect.com/science/article/pii/0032591089800087`.

P. Hanmaiahgari, V. Roussinova, and R. Balachandar. Turbulence characteristics of flow in an open channel with temporally varying mobile bedforms. *Journal of Hydrology and Hydromechanics*, 65, 03 2017. doi: 10.1515/johh-2016-0044.

Hatari Labs, 2018. Floating object stability modeling with openfoam - tutorial. URL `https://hatarilabs.com/ih-en/floating-object-stability-modeling-with-openfoam`.

F. M. Henderson. *Open Channel Flow*. MacMillan Company, New York, 1966.

F. Hernández-Jiménez, S. Sánchez-Delgado, A. Gómez-García, and A. Acosta-Iborra. Comparison between two-fluid model simulations and particle image analysis & velocimetry (piv) results for a two-dimensional gas–solid fluidized bed. *Chemical Engineering Science*, 66(17):3753–3772, 2011. ISSN 0009-2509. doi: https://doi.org/10.1016/j.ces.2011.04.026. URL `https://www.sciencedirect.com/science/article/pii/S0009250911002685`.

D. Hinshaw. Ghana's Growth Spurs Uncontrollable Trash. *The Wall Street Journal*, 2015.

D. Honingh, T. van Emmerik, W. Uijttewaal, H. Kardhana, O. Hoes, and N. van de Giesen. Urban river water level increase through plastic waste accumulation at a rack structure. 2020.

C. Hoogevorst, 2019. Plastic onderscheppen aan het wateroppervlak. Master's thesis, Delft University of Technology.

D. G. F. Huilier. An Overview of the Lagrangian Dispersion Modeling of Heavy Particles in Homogeneous Isotropic Turbulence and Considerations on Related LES Simulations. *Fluids*, 6(4), 2021. ISSN 2311-5521. doi: 10.3390/fluids6040145. URL `https://www.mdpi.com/2311-5521/6/4/145`.

Ideal Simulations, n.d. Turbulence Models In CFD. URL `https://www.idealsimulations.com/resources/turbulence-models-in-cfd/`.

R. Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of Computational Physics*, 62(1):40–65, 1986. ISSN 0021-9991. doi: https://doi.org/10.1016/0021-9991(86)90099-9. URL `https://www.sciencedirect.com/science/article/pii/0021999186900999`.

P. S. Jackson. On the displacement height in the logarithmic velocity profile. *Journal of Fluid Mechanics*, 111:15–25, 1981. doi: 10.1017/S0022112081002279.

G. Kasat, A. Khopkar, V. Ranade, and A. Pandit. CFD simulation of liquid-phase mixing in solid–liquid stirred reactor. *Chemical Engineering Science*, 63(15):3877–3885, 2008. ISSN 0009-2509. doi: https://doi.org/10.1016/j.ces.2008.04.018. URL `https://www.sciencedirect.com/science/article/pii/S0009250908001929`.

H. Kheirkhah Gildeh, A. Mohammadian, and R. Shaheed. 3D Numerical Modelling of Secondary Current in the Shallow River Bends and Confluences. 05 2016.

M. Kooi, E. Besseling, C. Kroeze, A. P. van Wezel, A. A. Koelmans, M. Wagner, and S. Lambert. *Modeling the Fate and Transport of Plastic Debris in Freshwaters: Review and Guidance*, pages 125–152. Springer International Publishing, Cham, 2018a. doi: 10.1007/978-3-319-61615-5_7. URL `https://doi.org/10.1007/978-3-319-61615-5_7`.

M. Kooi, E. Besseling, C. Kroeze, A. P. van Wezel, A. A. Koelmans, M. Wagner, and S. Lambert. *Modeling the Fate and Transport of Plastic Debris in Freshwaters: Review and Guidance*, pages 125–152. Springer International Publishing, Cham, 2018b. doi: 10.1007/978-3-319-61615-5_7. URL `https://doi.org/10.1007/978-3-319-61615-5_7`.

S. Kubowicz and A. M. Booth. Biodegradability of Plastics: Challenges and Misconceptions. *Environmental Science & Technology*, 51(21):12058–12060, 2017.

H. Kudela, n.d. Turbulent flow. `http://www.itcmp.pwr.wroc.pl/~znmp/dydaktyka/fundam_FM/Lecture_no3_Turbulent_flow_Modelling.pdf`.

S. Kumar, A. Saha, and P. Munshi. Euler-euler two-fluid model based code development for two-phase flow systems. 01 2018. doi: 10.1615/IHMTC-2017.2880.

L. C. M. Lebreton, J. van der Zwet, J. Damsteeg, B. Slat, A. Andrady, and J. Reisser. River plastic emissions to the world's oceans. *Nature Communications*, 8, 2017. ISSN 1.

H. C. Lee and A. K. A. Wahab. Performance of different turbulence models in predicting flow kinematics around an open offshore intake. *SN Applied Sciences*, 2019.

F. Liu. A Thorough Description Of How Wall Functions Are Implemented In OpenFOAM. 2016a.

S. Liu. Implementation of a Complete Wall Function for the Standard $k - \epsilon$ Turbulence Model in OpenFOAM 4.0. 2016b.

X. Liu. Analytical solutions for steady two-dimensional suspended sediment transport in channels with arbitrary advection velocity and eddy diffusivity distributions. *Journal of Hydraulic Research*, 54(4): 389–398, 2016c. doi: 10.1080/00221686.2016.1168880. URL `https://doi.org/10.1080/00221686.2016.1168880`.

J. Lundberg and B. Halvorsen. A review of some existing drag models describing the interaction between phases in a bubbling fluidized bed. *Proc. Proc. 49th Scand. Conf. Simulation and Modeling*, 01 2008.

H. Luo, C. Zhang, and J. J. Zhu. Development of a numerical model for the hydrodynamics simulation of liquid-solid circulating fluidized beds. *Powder Technology*, 348, 03 2019. doi: 10.1016/j.powtec.2019.03.018.

T. Mani, A. Hauk, U. Walter, and P. Burkhardt-Holm. Microplastics profile along the Rhine River. *Scientific Reports*, 2015. ISSN 1.

F. R. Menter. Influence of freestream values on k-omega turbulence model predictions. *AIAA Journal*, 30(6):1657–1659, 1992. doi: 10.2514/3.11115. URL `https://doi.org/10.2514/3.11115`.

M. R. Michielssen, E. R. Michielssen, J. Ni, and M. B. Duhaime. Fate of microplastics and other small anthropogenic litter (sal) in wastewater treatment plants depends on unit processes employed. *Environ. Sci.: Water Res. Technol.*, 2:1064–1073, 2016. doi: 10.1039/C6EW00207B. URL `http://dx.doi.org/10.1039/C6EW00207B`.

J. Montes. Irrotational flow and real fluid effects under planar sluice gates. *Journal of Hydraulic Engineering-asce - J HYDRAUL ENG-ASCE*, 123:219–232, 1997. ISSN 2.

A. Nash. Impacts of marine debris on subsistence fishermen an exploratory study. *Marine Pollution Bulletin - MAR POLLUT BULL*, 24:150–156, 03 1992. doi: 10.1016/0025-326X(92)90243-Y.

National Geographic, 2021. Great pacific garbage patch. URL `https://www.nationalgeographic.org/encyclopedia/great-pacific-garbage-patch/`.

I. Nezu and W. Rodi. Experimental study on secondary currents in open channel flow. pages 19–23, Melbourne, Australia, 1985.

J. Nikuradse. . *Forschungsh. Ver. dtsch. Ing.*, 1933. ISSN 361.

H. Norouzi and R. Khodabandehlou, n.d. Mastering twoPhaseEulerFoam, One: Fluidized bed. `https://www.cemf.ir/PDFs/OpenFOAM/Fluidizedbed.pdf`.

NOS, 2019. Zo vol met plastic zitten de rivieren bij Jakarta.

Obscape, n.d. Obscape overview. `https://www.linkedin.com/company/obscape?originalSubdomain=nl`.

T. Ojeda. *Polymer and the Environment*. InTech, 2013.

OpenCFD Ltd, 2018. OpenFOAM User Guide. `https://www.openfoam.com/documentation/user-guide/index.php`.

OpenFOAM, 2006. OpenFOAM: User Guide. `https://www.openfoam.com/documentation/guides/latest/doc/index.html`.

OpenFOAM Wiki, 2014. Piso. `https://openfoamwiki.net/index.php/PISO#cite_note-1`.

N. Oskuyi and F. Salmasi. Vertical sluice gate discharge coefficient. *Journal of Civil Engineering and Urbanism*, 2:108–114, 2012. ISSN 3. URL `http://www.ojceu.ir/main/attachments/article/17/JCEU-B20,%20108-114,%202012.pdf`.

J. T. Padding, N. G. Deen, E. (Frank) Peters, and J. (Hans) Kuipers. Chapter three - euler–lagrange modeling of the hydrodynamics of dense multiphase flows. volume 46 of *Advances in Chemical Engineering*, pages 137–191. Academic Press, 2015. doi: https://doi.org/10.1016/bs.ache.2015.10.005. URL `https://www.sciencedirect.com/science/article/pii/S006523771500006X`.

S. M. Peker, . Helvacı, H. B. Yener, B. İkizler, and A. Alparslan. *4 - Motion of Particles in Fluids*. Elsevier, Amsterdam, 2008. doi: https://doi.org/10.1016/B978-044452237-5.50006-0.

PlasticsEurope, 2016. Plastics - the Facts 2016. `https://www.plasticseurope.org/en/resources/publications/3-plastics-facts-2016`.

Prabhansu, S. Dey, M. Karmakar, P. Chandra, and P. K. Chatterjee. Studies on various drag models in fluidized bed forabatement of environmental pollution. *INTERNATIONAL JOURNAL OF ENVIRONMENTAL SCIENCES*, 5:1011–1021, 01 2015. doi: 10.6088/ijes.2014050100095.

B. Rapp. *Chapter 9 - Fluids*. Micro and Nano Technologies. Elsevier, Oxford, 2017. ISBN 978-1-4557-3141-1. URL `http://www.sciencedirect.com/science/article/pii/B9781455731411500095`.

M. G. Rasul. Buoyancy force in liquid fluidized beds of mixed particles. *Particle & Particle Systems Characterization*, 16(6):284–289, 1999. doi: https://doi.org/10.1002/(SICI)1521-4117(199912)16:6<284::AID-PPSC284>3.0.CO;2-3. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291521-4117%28199912%2916%3A6%3C284%3A%3AAID-PPSC284%3E3.0.CO%3B2-3`.

J. Reisser, B. Slat, K. Noble, K. du Plessis, M. Epp, M. Proietti, J. de Sonneville, T. Becker, and C. Pattiaratchi. The vertical distribution of buoyant plastics at sea: an observational study in the North Atlantic Gyre. *Biogeosciences*, 12(4):1249–1256, 2015. doi: 10.5194/bg-12-1249-2015. URL `https://bg.copernicus.org/articles/12/1249/2015/`.

F. Ries, P. Obando, S. I., J. J., and A. Sadiki. Numerical analysis of turbulent flow dynamics and heat transport in a round jet at supercritical conditions. *International Journal of Heat and Fluid Flow*, 66:172–184, 08 2017. doi: 10.1016/j.ijheatfluidflow.2017.06.007.

H. Rusche. *Computational fluid dynamics of dispersed two-phase flows at high phase fractions* . PhD thesis, Imperial College London (University of London), 2003.

S. S. Sadri and R. C. Thompson. On the quantity and composition of floating plastic debris entering and leaving the tamar estuary, southwest england. *Marine Pollution Bulletin*, 81(1):55–60, 2014. ISSN 0025-326X. doi: https://doi.org/10.1016/j.marpolbul.2014.02.020. URL `https://www.sciencedirect.com/science/article/pii/S0025326X14001167`.

Y. Shammaa, Z. David, and N. Rajaratnam. Flow Upstream of Orifices and Sluice Gates. *Journal of Hydraulic Engineering-asce - J HYDRAUL ENG-ASCE*, 131, 02 2005. doi: 10.1061/(ASCE)0733-9429(2005)131:2(127).

Skybrary, 2017. Form drag. URL `https://www.skybrary.aero/index.php/Form_Drag`.

S. Sundby. A one-dimensional model for the vertical distribution of pelagic fish eggs in the mixed layer. *Deep Sea Research Part A. Oceanographic Research Papers*, 30(6):645–661, 1983. ISSN 0198-0149. doi: https://doi.org/10.1016/0198-0149(83)90042-0. URL `https://www.sciencedirect.com/science/article/pii/0198014983900420`.

P. K. Swamee. Sluice-gate discharge equations. *Journal of Irrigation and Drainage Engineering-asce*, 118:56–60, 1992.

M. Syamlal, W. Rogers, and T. O'Brien. Mfix documentation: Theory guide. *Other Information: PBD: Dec 1993*, 1, 01 1993. doi: 10.2172/10145548.

Tauw, 2017. Shoreliner vangt plastic soup in havens. `https://www.youtube.com/watch?v=9Xmimy3yUC4&ab_channel=TauwNL`, Accessed 10 Sep 2020.

Tauw, 2019. Shoreliner - vermindering aanvoer plastic naar de oceaan. `https://www.tauw.nl/op-welk-gebied/duurzaamheid/shoreliner.html?sqr=shoreliner&`, Accessed 10 Sep 2020.

The Ocean Cleanup, 2020. How it works - The Interceptor™. `https://theoceancleanup.com/rivers/` , Accessed 11 Sep 2020.

D. Ting. Basics of engineering turbulence. page v. Academic Press, 2016. ISBN 978-0-12-803970-0. URL `https://www.sciencedirect.com/science/article/pii/B9780128039700000131`.

M. Tritthart, M. Liedermann, and H. Habersack. Modelling spatio-temporal flow characteristics in groyne fields. *River Research and Applications*, 25(1):62–81, 2009. doi: https://doi.org/10.1002/rra.1169. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/rra.1169`.

W. Uijttewaal. Turbulente ondiepe stromingen. *NEDERLANDS TIJDSCHRIFT VOOR NATUURKUNDE*, 2014.

E. Van der Wal, M. Festa-Bianchet, D. Réale, D. Coltman, and F. Pelletier. Sfra0025: Identification and assessment of riverine input of (marine) litter. Technical report, DG Environment, 03 2015.

T. Van Emmerik and A. Schwarz. Plastic debris in rivers. *WIREs Water*, 7(1):e1398, 2020.

T. Van Emmerik, M. Loozen, K. van Oeveren, F. Buschman, and G. Prinsen. Riverine plastic emission from Jakarta into the ocean. *IOP Publishing Ltd*, 2019.

T. Van Emmerik, R. Tramoy, C. van Calcar, S. Alligant, R. Treilles, B. Tassin, and J. Gasperi. Seine plastic debris transport tenfolded during increased river discharge. *Frontiers in Marine Science*, 6, 2019. ISSN 2296-7745. doi: 10.3389/fmars.2019.00642.

B. Van Prooijen, June 2020. Lecture slides of Sediment Dynamics, CIE 4308, Delft University of Technology, Faculty CEG.

E. Van Utenhove, 2019. Modelling the transport and fate of buoyant macroplastics in coastal waters. Master's thesis, Delft University of Technology.

T. Van Welsenes, 2019. Modelling forces on buoyant macro plastics and their cross-sectional distribution in rivers. Master's thesis, Delft University of Technology.

H. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics* . Prentice Hall, 1995.

O. Visuri, G. A. Wierink, and V. Alopaeus. Investigation of drag models in cfd modeling and comparison to experiments of liquid–solid fluidized systems. *International Journal of Mineral Processing*, 104-105:58–70, 2012. ISSN 0301-7516. doi: https://doi.org/10.1016/j.minpro.2011.12.006. URL `https://www.sciencedirect.com/science/article/pii/S0301751611002109`.

P. Vriend, C. van Calcar, M. Kooi, H. Landman, R. Pikaar, and T. van Emmerik. Rapid assessment of floating macroplastic transport in the rhine. *Frontiers in Marine Science*, 7:10, 2020. ISSN 2296-7745. doi: 10.3389/fmars.2020.00010. URL `https://www.frontiersin.org/article/10.3389/fmars.2020.00010`.

K. Waldschläger, M. Born, W. Cowger, A. Gray, and H. Schüttrumpf. Settling and rising velocities of environmentally weathered micro- and macroplastic particles. *Environmental Research*, 191:110192, 2020. ISSN 0013-9351. doi: https://doi.org/10.1016/j.envres.2020.110192.

K. Waldschläger and H. Schüttrumpf. Erosion behavior of different microplastic particles in comparison to natural sediments. *Environmental Science & Technology*, 53(22):13219–13227, 2019a. URL `https://doi.org/10.1021/acs.est.9b05394`.

K. Waldschläger and H. Schüttrumpf. Effects of particle properties on the settling and rise velocities of microplastics in freshwater under laboratory conditions. *Environmental Science & Technology*, 53 (4):1958–1966, 2019b. doi: 10.1021/acs.est.8b06794.

A. Wimshurst, 2019. (cfd) eulerian multi-phase modelling. `https://www.youtube.com/watch?v=6BJauDTpCmo&ab_channel=FluidMechanics101`.

J. Yen, C. Lin, and C. Tsai. Hydraulic characteristics and discharge control of sluice gates. *Journal of the Chinese Institute of Engineers*, 24(3):301–310, 2001. doi: 10.1080/02533839.2001.9670628. URL `https://doi.org/10.1080/02533839.2001.9670628`.

B. Yilmaz, S. Ozdogan, and D. Raheem. Cold Flow Simulation of a 30 kWth CFB Riser with CPFD. *Journal of Applied Fluid Mechanics*, 13:603–614, 09 2019. doi: 10.29252/jafm.13.02.30534.

B. Yuill, Y. Wang, M. Allison, E. Meselhe, and C. Esposito. Sand settling through bedform-generated turbulence in rivers. *Earth Surface Processes and Landforms*, 45(13):3231–3249, 2020. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/esp.4962`.

L. Zaat, 2020. Below the surface. Master's thesis, Delft University of Technology.

A

Literature review

A.1. River plastics

Estimated plastic releases from Asian rivers represented 86% of the total global input. This is due to the high population density in combination with high plastic production rates. The estimated annual input is approximately 1.21 million tonnes per year. Of the total fraction worldwide, 0.28% comes from Europe with an estimated annual input 3,900 (range: 2,310–9,320) tonnes per year. (Lebreton et al., 2017) It is estimated from Van der Wal et al. (2015) that 20–31 tonnes flows into the North Sea every year from the Rhine River, originating from different sources.

The plastic concentration rises with a higher flow in water. This is due to the fact that higher discharge delivers a higher bed shear stress, transporting debris that has been laying in the river bed or in river floodplains. Thus, with the occurrence of rain and high fluctuations of discharge come high fluctuations of plastic concentration in the river flow. (Van Emmerik and Schwarz, 2020)

A.1.1. Types and applications

Originally plastics are produced from fossil fuels, with certain plastics developed with biomass also. Plastic has been incorporated in the daily life of people. Plastics can be manufactured in a multitude of ways and shapes as the material properties are diverse. The material has been engineered to meet its various demands for its given purpose, with the general properties of the plastics are their strength, resistance and durability.

The downside of the durability is that it hardly degrades in the environment. The fragmentation to micron-sized particle can take centuries, while this material can be harmful for organisms and environment (Kubowicz and Booth, 2017). Efforts have been made to make plastics biodegradable with shorter residence times in the environment. (Ojeda, 2013). However, plastics labelled as ”biodegradable” are still conventional plastics with additives accelerating the first breakdown process and still end up as micro particles in the environment. (Ammala et al., 2011) Plastics are regarded as particles that do not degrade throughout the river reach because of these findings.

A.1.2. Harms of plastics

When plastics end up in the environment, they can have damaging effects. This can occur to nature, wildlife and human interventions in multiple ways. This section discusses certain damaging effects of plastics.

- Ingestion and entanglement

Figure A.1: A manta net sampler. Photo by Julia Farkas, SINTEF Ocean.

Figure A.2: A WFW-sampler.

The consumption of plastic particles can occurs for species of different sizes (e.g. whales,birds and fish), causing death if this accumulates in the body of animals. Entanglement of these species can also be lethal. Humans can consume these plastics when consuming other animals: on average people ingest 5g of plastic every week (Dalberg Advisors et al., 2019). Different types of plastic have varying toxic properties and thus varying consequences for plastic consumption for humans. A buildup of these toxins could impact the immune system, however long term effects remain unclear.

- Navigation and fisheries

Plastic pollution in rivers can result in hindrance to navigation when plastics are accumulated in river or canal segments. An extreme case of plastic buildup in a river is indicated in Fig. A.3, of the Pisang Batu river in Indonesia.

Figure A.3: Plastic buildup in the Pisang Batu river, which runs through a village in the Bekasi Regency of West Java, Indonesia, from (NOS, 2019).

Fisheries can be affected by plastic debris, similar to navigation problems, but also economically. Propellers have been noted to get entangled, damage to fishing gear and injuries have been reported by Nash (1992) in Indonesia.

A.1.3. Transportation

Disposal of plastics is poorly managed. The material can be found in various locations after being disposed of: on the streets, in the rivers and on landfill locations. When plastics are littered, wind or rainfall can carry a plastic toward a drain into a river or directly into the river. After the plastic debris enters a river, it is deposited at a location in the river sediment, river obstructions, floodplains or transported downstream.

A.1.4. Plastic sinks

Deposition can occur due to sinks and retention of particles, turbulences, stillwaters and drift ashore towards the river bank (Mani et al., 2015). Furthermore, plastics can be be deposited into the soil of the river, when the particle settles. Particle settling is further elaborated on in Appx. A.4.2.

If the particle lies in the river bed for a period of time, the particle can get buried by other sediment on top of this, resulting in an armouring effect. These plastics can then only be resuspended when the shear stress is higher than the critical shear stress for this particles, thus with high energy events. This happens during storm events.

When debris is transported downstream towards the sea, it ends up at either the beach or further into the ocean. Ultimately this ends up in one of the ocean gyres, which are locations where ocean currents form a circling pattern. Buoyant litter is concentrated at these locations.

A.1.5. Plastic concentration variation

High seasonality in concentration of plastic debris in rivers are found throughout the year. These differences are related to arid and wet weather: in the Italian Po River, sampled concentrations differed by one order of magnitude between Winter and Spring, due to the larger amount of precipitation in the Winter time (Lebreton et al., 2017). The report by Mani et al. (2015) found substantial temporal variations in plastic contamination levels.

Differences in plastic concentrations in rivers are found throughout the year. These differences are related to arid and wet weather, runoff can be directly related to plastic contamination levels and the transport of plastics into freshwater systems. In the Po River, sampled concentrations differed by one order of magnitude between winter and spring (Lebreton et al., 2017), emphasising seasonality of freshwater contamination in rivers. High seasonality in concentration of plastic debris in the river was found corresponding with the high rainfall peaks throughout the year.

Van Emmerik et al. (2019) found that plastics are abundant in the top layer, with specifications in two different layers in Jakarta, see Fig. A.4. The level of detail is relatively low, with only two layers specified in the vertical.

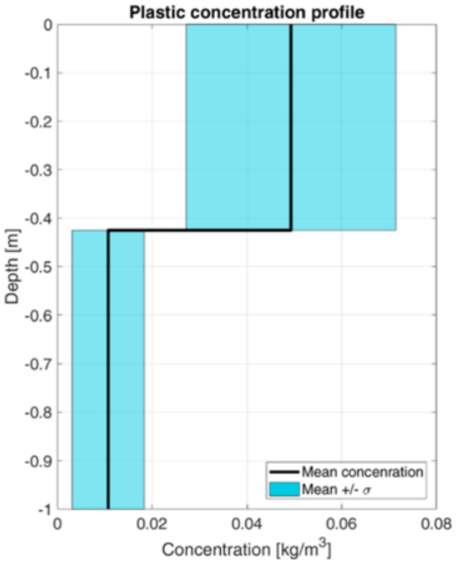


Figure A.4: 2 layer measurements in Jakarta.

A.2. Schematisations systems

The Ocean Cleanup started with a system to catch plastic debris floating in the ocean around the major rotating gyres. The system has shown to be working, but has not had much success Fairs (2019). The Ocean Cleanup has developed a version of a plastic interception system that can be applied in rivers.

The first prototype of the Interceptor has been tested in Zuidland and by 2018 the second design had been made. In 2019 the first tests started and currently the system is active in Malaysia. The first prototype is currently deployed in a river in Indonesia.

A.3. Turbulent flow

A.3.1. Reynolds number

From the Reynolds number the flow patterns in different situations can be determined. This is for laminar, turbulent and intermediate flow fields. The number is expressed as the inertial forces divided by the viscous forces. Low Re-numbers indicate laminar flow (< 2300) , where high numbers (> 4000) indicate turbulent flow. For laminar flow the viscous forces are dominant, where turbulent flow is dominated by inertial forces, with eddies, vortices and other flow instabilities as features of this flow field. The equation for the Re-number is given in Eq. A.1:

$$\mathrm{Re} = \frac{\rho u L}{\mu} = \frac{uL}{\nu} \tag{A.1}$$

The equation given for the flow velocity in the vertical is given by Eq. A.2.

$$\frac{u}{u_*} = \frac{1}{\kappa} \ln\left\{\frac{y}{y_0}\right\} \tag{A.2}$$

Where:

κ is the Von Karman constant, which is 0.41;

z is the position in the vertical;

u^* is the friction velocity

A.4. Particle transport

A.4.1. Equilibrium conditions spherical particles

$$C(y) = C_a \left(\frac{h-y}{y} \frac{a}{h-a}\right)^{\beta} \tag{A.3}$$

Rouse profile relates the sediment fall velocity with the shear velocity.

Where

R_0 is the particle diameter

ρ_p is the particle density

ρ_w is the water density

μ is the dynamic viscosity of water

g is the gravitational acceleration of 9.81 m/s^2

A.4.2. Nonspherical particles

For nonspherical particles the drag coefficient is determined to be higher than for spherical particles, as a decrease in the Stokes velocity was measured. In the past multiple expressions have been set up and these have been compared by Chhabra et al. (1999). This included 1900 data points covering a range of Reynolds numbers (10−4 < $Renp$ <105), geometrical shapes of particles, and sphericity values of 0.09 < ψ < 1.

Expressions derived by Haider and Levenspiel (1989) and Ganser (1993) give more accurate results with an average error of 16.3%. This included a Reynolds number specifically for nonspherical particles Re_{np} related to the drag coefficient. The sphericity is a nondimensional number indicating whether the particle is spherical (ψ=1) and smaller than 0.67 for disks (Chhabra et al., 1999).

A.5. Plastic modelling

There is a great level of uncertainty related to both measurements and models of plastic transport in riverine systems. Degradation processes, the origin and deposition are poorly understood, with potentially an important role in the size and spatial distributions of plastic in freshwater ecosystems. With the equations described at Eq. A.8 the settling and rise velocities can be determined, while this still yields errors of around 30%. A more accurate manner is yet to be found and in turbulence this number is expected to increase more. (Burns and Boxall, 2018)

A.5.1. Settling of plastics

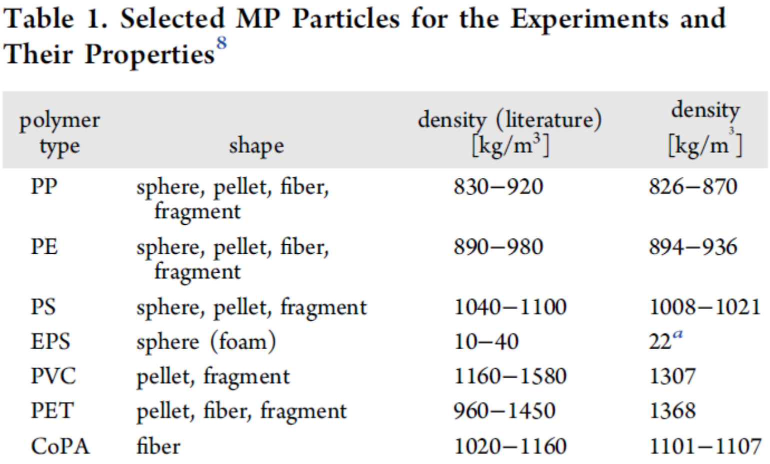


Table 1. Selected MP Particles for the Experiments and Their Properties[8]

polymer type	shape	density (literature) [kg/m^3]	density [kg/m^3]
PP	sphere, pellet, fiber, fragment	830−920	826−870
PE	sphere, pellet, fiber, fragment	890−980	894−936
PS	sphere, pellet, fragment	1040−1100	1008−1021
EPS	sphere (foam)	10−40	22[a]
PVC	pellet, fragment	1160−1580	1307
PET	pellet, fiber, fragment	960−1450	1368
CoPA	fiber	1020−1160	1101−1107

Figure A.5: Measured and found densities from different plastic materials. (Waldschläger and Schüttrumpf, 2019b)

For settling pellets and fragments, the drag coefficient is:

$$C_{D,s,p+f} = \frac{3}{CSF \cdot \sqrt[3]{Re}} \tag{A.4}$$

For settling fibres this is:

$$C_{D,s,fibre} = \frac{4.7}{\sqrt{Re}} + \sqrt{CSF} \tag{A.5}$$

For rising pellets and fragments, the drag coefficient becomes:

$$C_{D,r,p+f} = \left(\frac{20}{Re} + \frac{10}{\sqrt{Re}} + \sqrt{1.195 - CSF}\right) \cdot \left(\frac{6}{P}\right)^{1-CSF} \tag{A.6}$$

For fibres, this coefficient reads:

$$C_{D,s,fibre} = \frac{10}{\sqrt{Re}} + \sqrt{CSF} \tag{A.7}$$

The subsequent settling or rise velocity reads:

$$w_s = \sqrt{\frac{4}{3}\frac{d_{\text{equi}}}{C_D}\left(\frac{\rho_s - \rho}{\rho}\right) g} \tag{A.8}$$

Waldschläger et al. (2020) compared the results of settling MP particles for the above equations with the equations for Stokes particles and saw that the accuracy is higher for the upper equations, with errors $\approx 30\%$. The Stokes equations for different Re numbers are given below in Eq. A.9

$$\begin{aligned} &\mathrm{Re} < 0.5 : C_D = \frac{24}{\mathrm{Re}} (3 \text{ of } 100 \text{ particles }) \\ &0.5 < \mathrm{Re} < 10^3 : C_D = \frac{24}{\mathrm{Re}} + \frac{4}{\sqrt{Re}} + 0.4 (92 \text{ of } 100 \text{ particles }) \\ &10^3 < \mathrm{Re} < 2^* 10^5 : C_D = 0.44 (5 \text{ of } 100 \text{ particles }) \end{aligned} \tag{A.9}$$

A.5.2. Efforts in simplifying plastics with respect to shape, density, size

Non-buoyant plastic debris is subject to the advective, dispersive, and sedimentation processes. A unique feature here is that a high proportion of the plastic will have a density not that different from that of water, in contrast to natural suspended (mineral) solid particles of the same size. The variety of plastic sizes and densities, however, still varies enormously, leading to a wide variety of transport patterns for individual particles in the mixture.

Table A.1: lower and upper limit of the microparticles assessed

	lower limit	mean	upper limit
Size [mm]	0.2	0.2	5
CSF [$-$]	0	0.4	1
Density [kg/m^3]	955	1000	1099

Table A.2: Plastic types, commonality and density limits

Plastic type	Percentage found [%]	Lower density limit	Upper density limit
PE	25	890	980
PET	16.5	960	1450
PA	12	1020	1160
PP	14	830	920
PS	8.5	1040	1100
PVA	6	1190	1310
PVC	2	1100	1580

A.5.3. Fate of riverine and marine plastics

Kooi et al. (2018a) used a NanoDUFLOW model, parameterized for the investigation of plastic particle behaviour in riverine systems. The scale of this paper was 40km , thus with another scale than this project. The input values for the parameterisation of i.e. the density can be useful. Microplastic was assigned a density of 1040 kg/m3, which is the average of polymer densities found in the marine environment (Andrady, 2011). Furthermore, polymer density was varied from 1000 to 1500 kg/m3, representing a wide range of polymer types and with 1000 kg/m3 also representing non-settling plastic. For the scenarios with different polymer densities, the upstream plastic input concentration was kept constant on a volume basis. The model showed retention of plastics in the river, meaning that plastics are not only transported in the river to the ocean but also settle in the bed.

When the shear stress is higher than the critical shear stress, settled plastics in the bed can be resuspended if flow is high enough. This is possible for high energy events, such as storm events. After long sedimentation the particle can be buried into the sediment bed, only resuspended when the upper bed is eroded away. (Kooi et al., 2018a)

(Waldschläger and Schüttrumpf, 2019a) looks into the erosion of MP on and in the bed. 620 runs with 51 different plastic types were investigated in a physical model, with 4 different types of river bed (3 uniform, 1 mixed). Armouring effects are taken into consideration, with a final assessment for the applicability of the Shields diagram and the impact of the hiding-exposure effect on MP transport. The results are that the combination of the MP particle properties density, diameter, and shape with the sediment grain size influences the critical shear stress of MP particles severely. The critical shear stress increases with higher particle densities.

Natural sediment has an average density of 2650 kg/m3, the density of MP ranges between 20 and 1400 kg/m3. Approximately half of all manufactured plastics are heavier than water and can therefore sink to the bottom without any further influence necessary. MP particles have very diverse shapes such as pellets, fibres and fragments, whereas sediment consists mainly of almost round grains.

MP particles with a spherical shape move earliest, even before pellets with smaller diameters and lower particle densities. For example, the four points in Figure 5, which are well below the other points, belong to the PS spheres. Of all particle shapes, spheres have the smallest surface contact with the underlying ground, so there is less shear resistance and therefore less drag force is required for particle movement

The shape of these different type of materials was assessed with the Corey Shape Factor (CSF), where a, b, and c are the three main particle side lengths:

$$CSF = \frac{c}{\sqrt{ab}}$$

Because of those strongly varying particle properties and insufficient knowledge of the prevailing transport mechanisms, **a transferability of the basic principles from sediment transport is at least questionable.**

”Findings where that a comparison with the Shields diagram has shown that half of the MP particles move earlier than expected (thus earlier than natural sediments would), and therefore a higher MP transport rate can be assumed than would be determined with the theory from sediment transport. This was explained by the impact of the hiding-exposure effect and based on the nonuniform sediment transport an adapted equation was determined, which can be used to determine the critical shear stress of MP particles on natural sediment.”

”When considering the erosion behavior in numerical simulations, the dependence of the critical shear stress on the sediment bed has to be paid attention to and a fundamental transferability of the parameters from sediment transport to MP transport has to be rethought. ”

The erosion behaviour of MP is looked into in Ballent et al. (2013) as well, although this is looked into for a marine and not fluvial environment. The project looked at MP particles on a bed of MP in a uniform composition, and determined critical shear stresses of MP pellets in saltwater. According to Waldschläger and Schüttrumpf (2019a) this investigation is suitable as an introductory consideration, but the significance for processes appearing in nature is low.

The definition of critical shear stress is also calculated in a different order, as it was calculated back from the prevailing flow velocities to the critical shear stresses, where in Waldschläger and Schüttrumpf

(2019a) the critical shear stresses were determined directly. Furthermore only pellets were investigated.

Van Welsenes (2019) has looked into the propagation of floating macroplastics in riverine sections. However, these particles are too large and wind coefficients have not been tested in a sensitivity case study for this project. The model applied is a D3D model, which differs from CFD modelling.

A.5.4. Physical modelling

Zaat (2020) tested the spreading of plastic films in a flume with a depth of 0.25m with a setup of 6.7m between the release and measurement of the plastic films. The release was for 100 plastic films at 7 different depths.

The dimensions of the plastic films were assessed and different propagation velocities were measured. The vertical profile of the plastics followed the Rouse profile, with a shape factor. From this flume test it can be said what the average rise velocities are for this film of HDPE and LDPE material. The discussion for this thesis is the amount of air bubbles attached to the film largely influence the rise velocity.

Obstructions in the upper water column were also tested. A solid gate of 1/5 of the water depth was tested at the water surface, as well as a bottom sill and a obstruction half way the depth of the same dimension. The gate at the water surface is of specific interest. It showed that plastic films were captured, but eventually that the plastic passed underneath the gate when these where in abundance adjacent to the gate. No noticeable difference in shear velocity was found for a varying approach angle: in an additional experiment, the orientation of the sluice gate was altered to 70° and 110° with respect to the waterline.

A.6. CFD modelling

According to (Badano et al., 2012) OpenFOAM is a modelling toolbox based on the finite-volume method that has become widely used in the research community because of the open-source code, easy customisation, and wide array of available solvers and utilities.

Application to open-flow hydrodynamics consists of studies focusing among others on horizontal recirculation in canyon rivers (Alvarez et al., 2017), river channel evolution by landslides (Zhao et al., 2017), and flow through vegetation (Chakrabarti et al., 2016).

A.6.1. Particle modelling

(Andric, n.d.) looked at various two phase flow regimes and the associated packages to it. It describes the different packages necessary for modelling particles in OpenFOAM, which are the classes Solidparticles and Solidparticlecloud with their according specifications in this case.

The paper by (Tritthart et al., 2009) has applied a CFD model for groyne fields, together with a particle tracking tool to measure residence times in eddies next to groynes.

A 2nd order upwind model was applied, as well as 3D RANS averaged equations in the modelling tool RSIm 3D. A particle tracing model was applied so that the preturbations from turbulence that are averaged with RANS are incorporated. The project applied a $k - \epsilon$ model as turbulence model. Where OpenFOAM is an implicit model, this project followed an explicit model, that had to meet the Courant definition.

The grid length was 4 km (test reach of 3 km + 0.5 km up- and downstream) with a 1x1 m^2 grid

The system was validated with observed tracer data in the river and model data. Most accuracy was reached for simulation time with $t < 45\ min.$ Velocities were measured at different depths compared with random walk model: limited errors in some cases.

The random walk particle tracing model applied is of the order:

$$\begin{aligned} x(t+\Delta t) &= x(t)+\bar{u}(t)\Delta t+u'(t)\Delta t \\ y(t+\Delta t) &= y(t)+\bar{v}(t)\Delta t+v'(t)\Delta t \\ z(t+\Delta t) &= z(t)+\bar{w}(t)\Delta t+w'(t)\Delta t \end{aligned} \tag{A.10}$$

And the kinetic energy:

$$k=\frac{1}{2}(\overline{u}'^2+\overline{v}'^2+\overline{w}'^2) \tag{A.11}$$

The turbulent fluctuations follow:

$$\begin{aligned} u'(t) &= 2Z_1\sqrt{k} \\ v'(t) &= 2Z_2\sqrt{k} \\ w'(t) &= 0 \end{aligned} \tag{A.12}$$

Where Z_1 , Z_2 are independent random numbers between -1 and 1.

This study by Yuill et al. (2020) uses a three-dimensional OpenFOAM hydrodynamic and particle-tracking model to investigate how turbulence generated from bedforms and the channel bed influences medium sand-sized particle settling, in terms of the distribution of suspended particles within the flow field and particle-settling velocities.

Turbulence was resolved using detached eddy simulation (DES) and particle settling using a Lagrangian particle-tracking solver. ”non-hydrostatic Navier–Stokes equations for single-phase flow with constant density and viscosity. Turbulence, within the flow field, was simulated using a DES approach which employs a one-equation mixing-length RANS (Reynolds-averaged Navier–Stokes) model to approximate boundary-layer flow near walls and a large-eddy simulation (LES) scheme away from walls. The LES scheme fully resolves turbulent flow dynamics at and above a length scale approximate to the cell dimensions of the computational grid or mesh.”

As with other computational modelling papers read, this study has not been verified, while it has adjusted underlying transport mechanisms.

Burak (2015) compared different turbulence models at a river location simplified in the model. The release of 11,31 kg particles injected from inlet surface at the beginning of simulation for k-ε and k-ω turbulence models to understand particle motions was performed. Furthermore for 22,62 kg and 11,31 kg sediment particles were released under the same conditions for this purpose and results showed that particles settled in a similar way at both cases. These simulations were conducted for uniform particle distributions of D=0.001 mm, D= 0.003 mm and D = 0.005 mm as median diameters with an added particle tracking tool.

The grid resolution was LWD = 90x40x3.7 with a grid size of = 2.25x2.5x0.53. Thus it is possible in CFD modelling to model the release of very small particles in a relatively large domain, here the ratio between particle : cell is an order of magnitude of ≈ 1 :100 - 1000.

B

OpenFOAM details

B.1. Implementation of $k-\epsilon$ turbulence model in OpenFOAM

Table B.1: Model constants k-epsilon in OpenFOAM. (OpenFOAM, 2006)

Constant	C_μ	C_1	C_2	C_3	σ_k	σ_ϵ
Value	0.09	1.44	1.92	0.0	1.0	1.3

The turbulent kinetic energy dissipation rate is given by :

$$\frac{\partial}{\partial t}(\alpha\rho\epsilon) + \nabla\cdot(\alpha\rho\mathbf{u}\epsilon) - \nabla^2(\alpha\rho D_\epsilon\epsilon) = C_1\alpha\rho G\frac{\epsilon}{k} - \left(\left(\frac{2}{3}C_1 - C_{3,RDT}\right)\alpha\rho\nabla\cdot\mathbf{u}\epsilon\right) - \left(C_2\alpha\rho\frac{\epsilon}{k}\epsilon\right) + S_\epsilon \quad \text{(B.1)}$$

The turbulent kinetic energy equation is given by :

$$\frac{\partial}{\partial t}(\alpha\rho k) + \nabla\cdot(\alpha\rho\mathbf{u}k) - \nabla^2(\alpha\rho D_k k) = \alpha\rho G - \left(\frac{2}{3}\alpha\rho\nabla\cdot\mathbf{u}k\right) - \left(\alpha\rho\frac{\epsilon}{k}k\right) + S_k \quad \text{(B.2)}$$

B.2. Pressure coupling algorithms

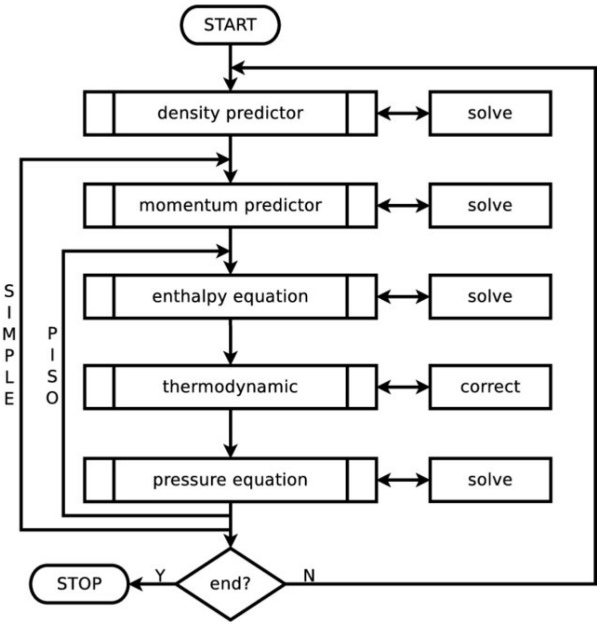


Figure B.1: PISO-SIMPLE loop. From Ries et al. (2017).

$$\nabla \cdot U = 0 \tag{B.3}$$

$$\frac{\partial U}{\partial t} + \nabla \cdot (UU) = -\frac{1}{\rho}\nabla p + \mu \nabla^2 U + g \tag{B.4}$$

PISO and SIMPLE deals with this problem.

The momentum equations are written in general matrix form:

$$MU = -\nabla p \tag{B.5}$$

In OpenFOAM a discretisation of the terms is performed, with the FVM.

OF : function:

```
solve(UEqn == -fvc::grad(p))
```

$$\mathcal{M} = \begin{pmatrix} M_{1,1} & M_{1,2} & M_{1,3} & \dots & M_{1,n} \\ M_{2,1} & M_{2,2} & M_{2,3} & \dots & M_{2,n} \\ M_{3,1} & M_{3,2} & M_{3,3} & \dots & M_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ M_{n,1} & M_{n,2} & M_{n,3} & \dots & M_{n,n} \end{pmatrix} \tag{B.6}$$

It is a momentum predictor stage, as this computed velocity field U does not satisfy the continuity equation.

From the computed matrix M a diagonal matrix is computed, matrix A.

$$\mathcal{A} = \begin{pmatrix} M_{1,1} & 0 & 0 & \dots & 0 \\ 0 & M_{2,2} & 0 & \dots & 0 \\ 0 & 0 & M_{3,3} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & M_{n,n} \end{pmatrix} \tag{B.7}$$

The inverse of this matrix is then computed:

$$\mathcal{A}^{-1} = \begin{pmatrix} 1/A_{1,1} & 0 & 0 & \dots & 0 \\ 0 & 1/A_{2,2} & 0 & \dots & 0 \\ 0 & 0 & 1/A_{3,3} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1/A_{n,n} \end{pmatrix} \tag{B.8}$$

This is written in OpenFOAM as

```
volScalarField rUA = 1.0 / UEqn().A()
```

Because of the extraction from the diagonal from the $\mathcal{M}$ matrix, Appx. B.2 can be written:

$$\mathcal{M}U = \mathcal{A}U - \mathcal{H} \tag{B.9}$$

Rearrangement gives:

$$\mathcal{H} = \mathcal{A}U - \mathcal{M}U$$

Where H is the residual left after extraction of the diagonal of the matrix. This enables the calculation of the source term for the pressure equation.

From the momentum equation similar as before, see Eq. B.10, both sides are multiplied by $\mathcal{A}^{-}$ (Eq. B.11 and substituted in the continuity equation, see Eq. B.12:

$$\mathcal{A}U - \mathcal{H} = -\nabla p \tag{B.10}$$

$$\mathcal{A}^{-1}\mathcal{A}U = \mathcal{A}^{-1}\mathcal{H} - \mathcal{A}^{-1}\nabla p \quad , U = \mathcal{A}^{-1}\mathcal{H} - \mathcal{A}^{-1}\nabla p \tag{B.11}$$

$$\nabla U = 0; \quad \nabla \cdot \left(\mathcal{A}^{-1}\nabla p\right) = \nabla \cdot \left(\mathcal{A}^{-1}\mathcal{H}\right) \tag{B.12}$$

In OpenFOAM this is written as:

$$\nabla \cdot \left(\frac{1}{a_p}\nabla p\right) = \nabla \cdot \left(\frac{\mathcal{H}(U)}{a_p}\right) \tag{B.13}$$

When the pressure field is solved, this solution can be used to correct the velocity field, satisfying the continuity equation, Eq. B.14:

$$U = \mathcal{A}^{-1}\mathcal{H} - \mathcal{A}^{-1}\nabla p \tag{B.14}$$

This is an explicit pressure-corrector step. However, after this correction, the pressure equation is no longer satisfied, and the source term H, which is dependent on U, needs to be updated.

There are two ways to proceed here, which are the PISO and SIMPLE algorithms.

1. Solve momentum predictor
2. Extract A matrix from M matrix
3. Compute H matrix using the computed velocity field
4. Assembly of pressure equation

B.2.1. PISO algorithm

The abbreviation of PISO is Pressure-Implicit with Splitting of Operators, which splits the operators into an implicit predictor and multiple explicit corrector steps. It is not an iterative process. The momentum equation for RANS, one dimensional, inviscid flow along the x-direction, with a gravity component in the equation as well. (OpenFOAM Wiki, 2014)(Issa, 1986)

$$\frac{\partial \bar{u}}{\partial t} + \frac{\partial}{\partial x}(\bar{u}\bar{u}) = -\frac{\partial p}{\partial x} + \rho_k g \tag{B.15}$$

The momentum predictor is again solved and used to update H, which is called an outer corrector loop, as it loops the entire process, see Eq. B.16

$$\begin{gathered} \mathcal{M}U = -\nabla p \\ \mathcal{H} = \mathcal{A}U - \mathcal{M}U \\ \nabla \cdot \left(\mathcal{A}^{-1}\nabla p\right) = \nabla \cdot \left(\mathcal{A}^{-1}\mathcal{H}\right) \\ U = \mathcal{A}^{-1}\mathcal{H} - \mathcal{A}^{-1}\nabla p \end{gathered} \tag{B.16}$$

For the PISO algorithm, the momentum predictor is once solved, and then proceeds to inner loops, which are the remaining three steps, until a certain amount of iterations or a convergence is reached:

$$\begin{gathered} \mathcal{H} = \mathcal{A}U - \mathcal{M}U \\ \nabla \cdot \left(\mathcal{A}^{-1}\nabla p\right) = \nabla \cdot \left(\mathcal{A}^{-1}\mathcal{H}\right) \\ U = \mathcal{A}^{-1}\mathcal{H} - \mathcal{A}^{-1}\nabla p \end{gathered} \tag{B.17}$$

B.2.2. SIMPLE algorithm

The abbreviation SIMPLE stands for Semi-Implicit Method for Pressure Linked Equations and consists of the following method for each iteration (Caretto et al., 1973) (OpenFOAM, 2006):

1. Advance to the next iteration t=tn+1;
2. Initialise u_{n+1} and p_{n+1} using latest available values of u and p;
3. Construct the momentum equations;
4. Under-relax the momentum matrix;
5. Solve the momentum equations to obtain a prediction for u_{n+1};
6. Construct the pressure equation;
7. Solve the pressure equation for p_{n+1};
8. Correct the flux for ϕ_{n+1};
9. Under-relax p_{n+1};
10. Correct the velocity for u_{n+1};
11. If not converged, go back to step 2.

B.2.3. PIMPLE algorithm

The PIMPLE algorithm is the methods of PISO and SIMPLE combined.

B.3. Two phase Eulerian modelling

$$\frac{\partial(1-\phi)}{\partial t} + \frac{\partial(1-\phi)u_i^l}{\partial x_i} = 0 \qquad \frac{\partial \phi}{\partial t} + \frac{\partial \phi u_i^p}{\partial x_i} = 0 \tag{B.18}$$

The momentum equations for the fluid (Eq. B.19) and particle (Eq. B.20) phases are written as:

$$\frac{\partial \rho^l (1-\phi) u_i^l}{\partial t} + \frac{\partial \rho^l (1-\phi) u_i^l u_j^l}{\partial x_j} = -\frac{\partial (1-\phi) p^l}{\partial x_i} + \frac{\partial (1-\phi)\tau_{ij}^l}{\partial x_j} + \rho^l (1-\phi) g \delta_{i3} + M_i^{fs} \tag{B.19}$$

$$\frac{\partial \rho^p \phi u_i^p}{\partial t} + \frac{\partial \rho^p \phi u_i^p u_j^p}{\partial x_j} = -\frac{\partial \phi p^l}{\partial x_i} - \frac{\partial p^p}{\partial x_i} + \frac{\partial \tau_{ij}^p}{\partial x_j} + \rho^p \phi g \delta_{i3} + M_i^{pl} \tag{B.20}$$

B.4. Drag models in reactingTwoPhaseEulerFoam

Table B.2: Drag models in reactingTwoPhaseEulerFoam, from Norouzi and Khodabandehlou (n.d.), Enwald et al. (1996a) and Yilmaz et al. (2019).

Drag model	**Application**	**Applicability**
AttouFerschneider	Gas-liquid flow	✗
Beetstra	Solid-gas flow	✗
Ergun	Fluidised bed	✗
Gibilaro	**Liquid-solid flow**	✓
GidaspowErgunWenYu	Combination Gidaspow-Ergun-WenYu	✗
GidaspowSchillerNaumann	Combination Gidaspow Schiller Naumann	✗
IshiiZuber	Gas-liquid flow	✗
Lain	Gas-liquid flow	✗
SchillerNaumann	Dispersed bubble flow	✗
Syamlal O'Brien	**Liquid-solid flow**	✓
Tenneti	Drag model for monodisperse gas-particle flows	✗
TomiyamaAnalytic	Bubble flow	✗
TomiyamaCorrelated	Bubble flow	✗
TomiyamaKataokaZunSakaguchi	Bubble flow	✗
WenYu	**Liquid-solid flow**	✓
segregated	Application for when there is no dispersed phase	✗

B.4.1. Gibilaro drag model

The drag model of Gibilaro (Enwald et al., 1996b):

$$\mathrm{K_{sg}} = \frac{\left(\frac{18}{Re} + 0.33\right)\left(\rho_g \left|v_\mathrm{p} - v_l\right|\right)}{d_p}\alpha_\mathrm{p}\alpha_l^{-1.8} \tag{B.21}$$

B.4.2. Syamlal O'Brien drag model

The Syamlal O'Brien drag model reads (Syamlal et al., 1993)(Hernández-Jiménez et al., 2011):

$$K_{ls} = \frac{3\alpha_p\alpha_l\rho_l}{4v_{r,s}2d_p}C_D\left(\frac{Re_p}{v_{r,s}}\right)\left|\overrightarrow{v_p} - \overrightarrow{v_l}\right| \tag{B.22}$$

$$C_D = \left(0.63 + \frac{4.8}{\sqrt{\frac{Re_p}{v_{r,s}}}}\right)^2 \tag{B.23}$$

$$Re_p = \frac{\rho_l d_p\left|\overrightarrow{v_p} - \overrightarrow{v_l}\right|}{\mu_l} \tag{B.24}$$

The terminal velocity is given as:

$$w_r = 0.5\left(A - 0.06Re_p + \sqrt{\left(0.06Re_p\right)^2 + 0.12\,\mathrm{Re}_p(2B - A) + A^2}\right) \tag{B.25}$$

$$\text{with}\quad A = \alpha_l^{4.14} \quad\text{and}\quad B = \begin{cases} 0.8\varepsilon_l^{1.28} & \text{if } \varepsilon_l \leq 0.85 \\ \varepsilon_l^{2.65} & \text{if } \varepsilon_l > 0.85 \end{cases} \tag{B.26}$$

B.4.3. Wen-Yu drag model

The Wen-Yu drag model reads:

$$K_{sl} = \frac{3\rho_l \alpha_l \left(1-\alpha_l\right)}{4d_p} C_D \left|\overrightarrow{u_p} - \overrightarrow{u_l}\right| \alpha_l^{-2.65} \tag{B.27}$$

With the drag coefficient for spherical particles given as:

$$C_D = \frac{24}{\alpha_l \,\mathrm{Re}_p}\left[1 + 0.15\left(\alpha_l \,\mathrm{Re}_p\right)^{0.687}\right] \tag{B.28}$$

And the Reynolds number for the solids

$$\mathrm{Re}_p = \frac{\rho_g d_p \left|\vec{V}_p - \vec{V}_g\right|}{\mu_g} \tag{B.29}$$

C

Mesh around interception system

C.1. Mesh between x=24.8 m to x=25.2 m over the depth

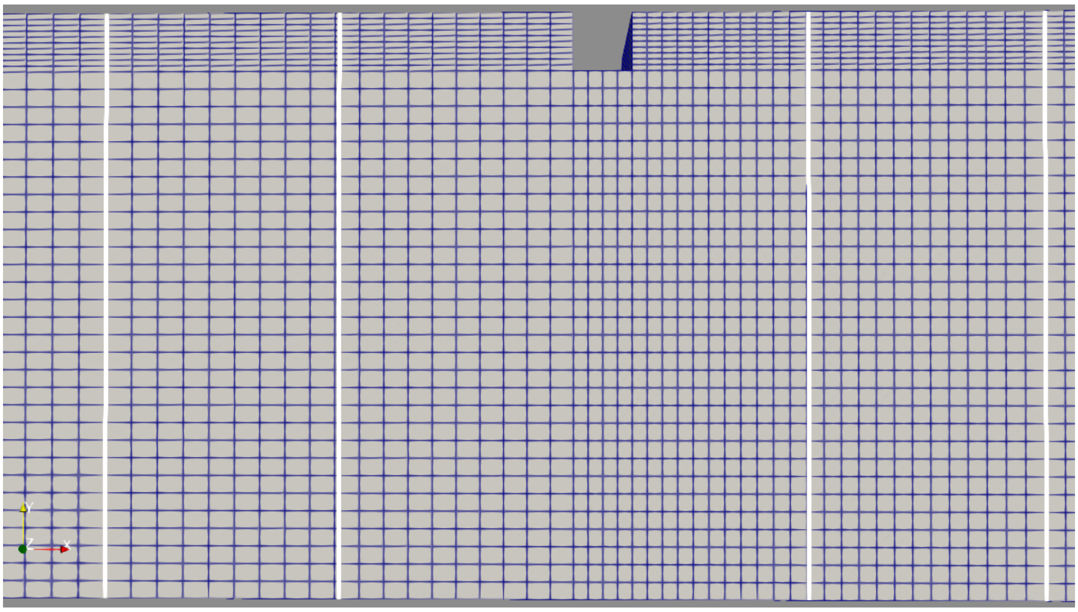


Figure C.1: Mesh around the interception system. The white vertical lines are coordinated at x=24.8 m, x=24.9 m,x=25.1 m and x=25.2 m.

C.2. Mesh between x=24.9 m to x=25.1 m and y=0.20 m to y=0.25 m

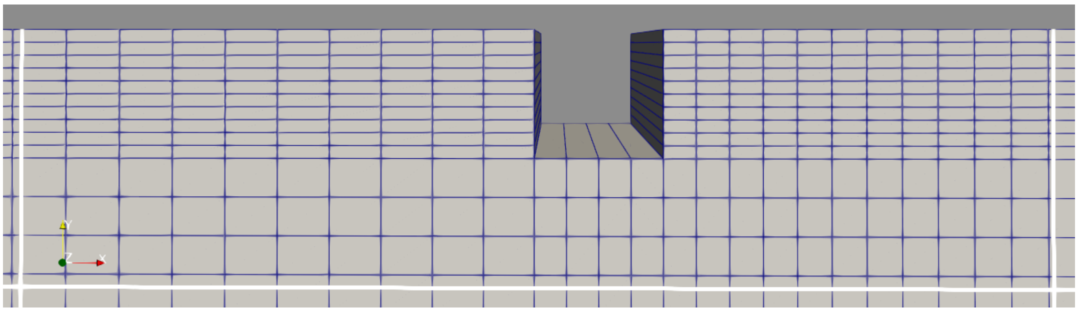


Figure C.2: Mesh around the interception system. The white vertical lines are coordinated at x=24.9 m and x=25.1 m. The horizontal line is situated at y=0.20 m.

D

Submerged gate study

A one-on-one validation of the horizontal and vertical velocity validation is not possible, as the opening is 90% of the water depth, whereas submerged gate studies mostly only go as far as 50% in their studies (Shammaa et al., 2005; Gumus et al., 2016).

A study was performed with similar inlet flow compared to earlier applied physical and computational model tests on this problem. This setup indicated what differences between the modelled flow velocity and the actual flow velocity in the horizontal and vertical were encountered.

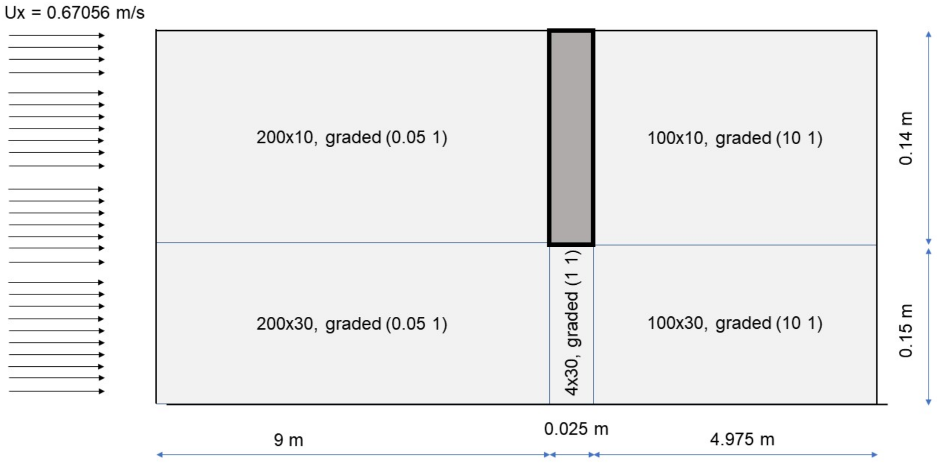


Figure D.1: Setup reference study, similar to produced results by Finnie and Jeppson (1991).

E

Supplementary results

E.1. Single-phase flow

E.1.1. Results for low flow

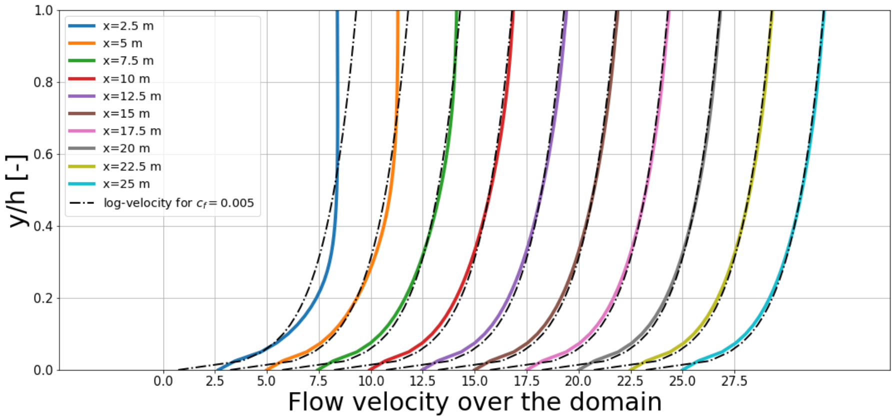


Figure E.1: Development of the horizontal flow velocity profile in the streamwise direction. The lines have been scaled and the dashed line illustrates the results from the simulation at x=25 m.

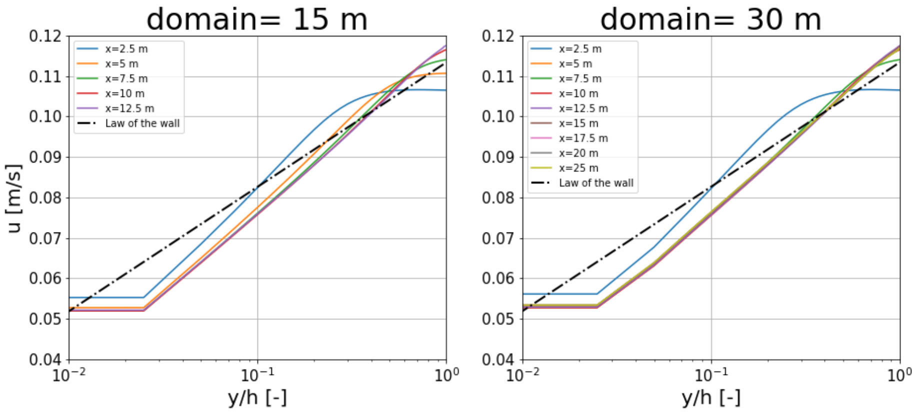


Figure E.2: Flow velocity over the depth and the length for 15 m and 30 m. The flow velocity at the inlet is 0.10 m/s.

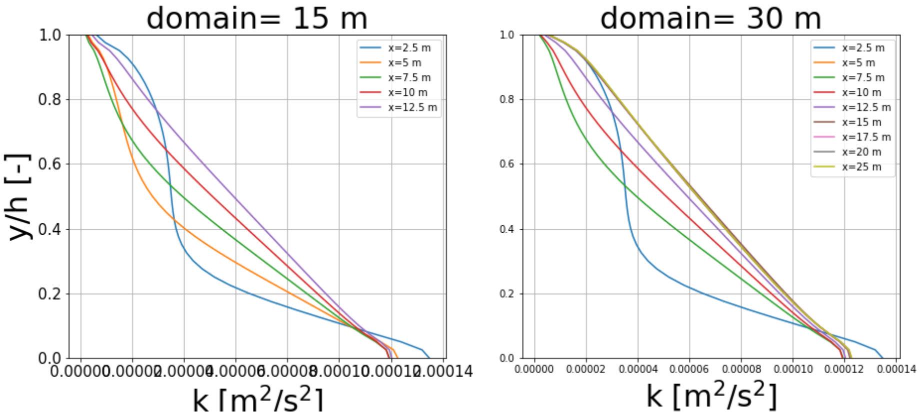


Figure E.3: Turbulent kinetic energy over the depth and length for 15 m and 30 m. The flow velocity at the inlet is 0.10 m/s.

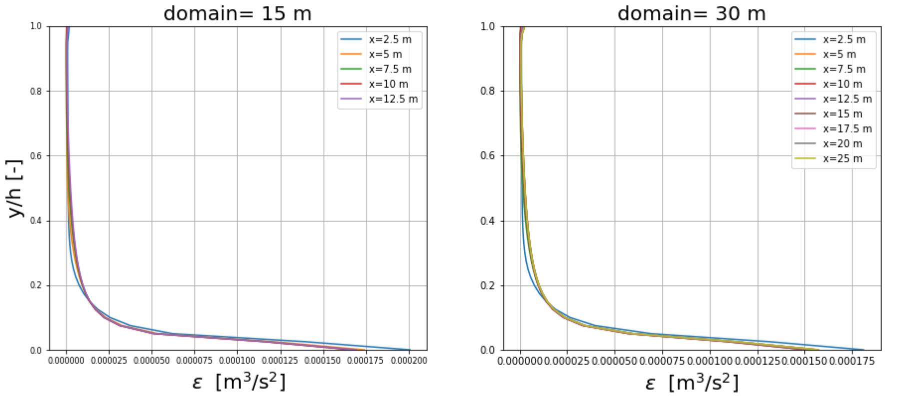


Figure E.4: Turbulent energy dissipation for low flow.

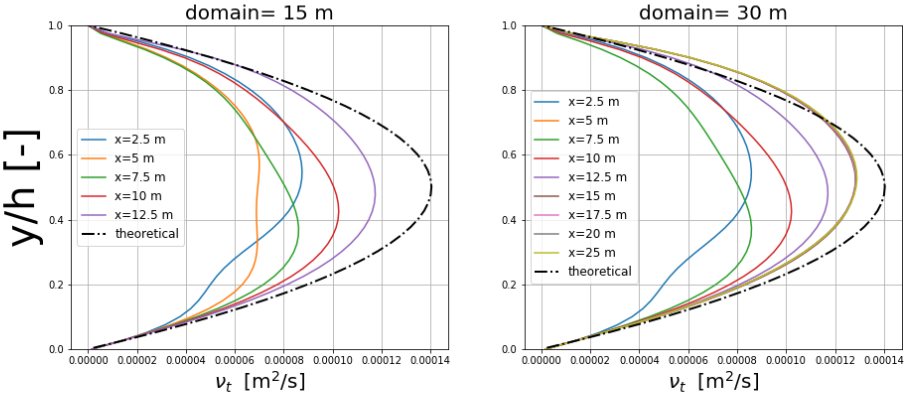


Figure E.5: Turbulent viscosity over the depth and length for 15 m and 30 m. The flow velocity at the inlet is 0.10 m/s.

E.1.2. Results for medium flow

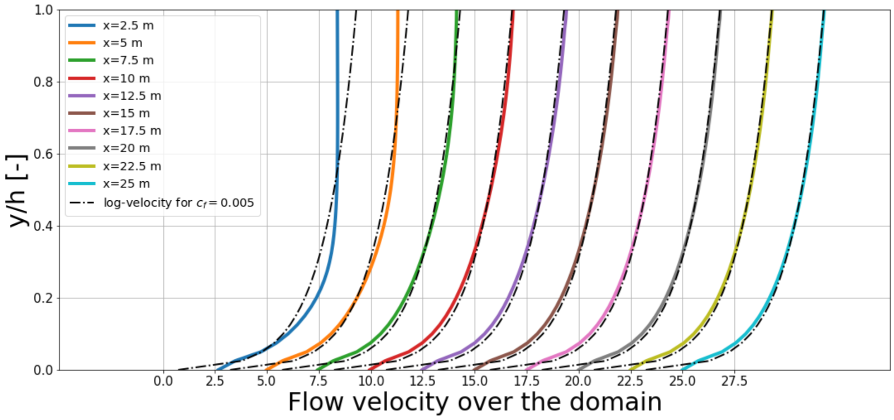


Figure E.6: Development of the horizontal flow velocity profile in the streamwise direction. The lines have been scaled and the dashed line illustrates the results from the simulation at x=25 m.

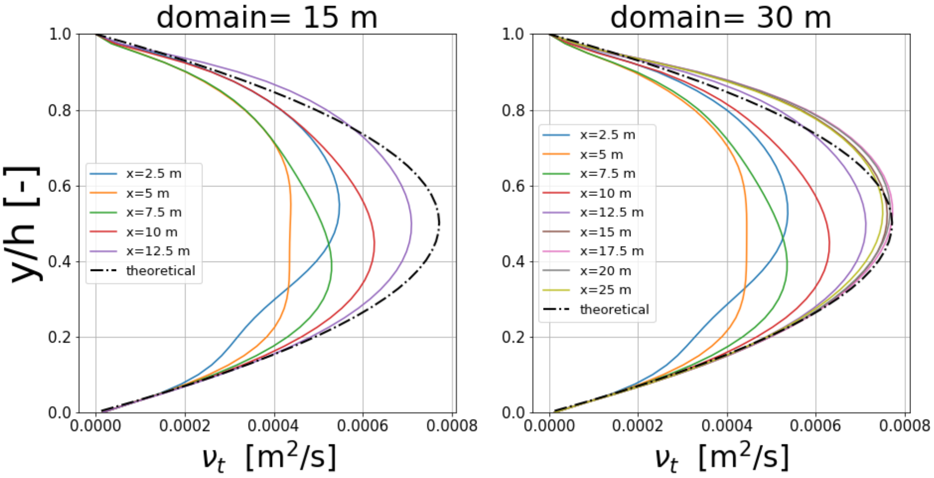


Figure E.7: Turbulent viscosity over the depth and the length for 15 m and 30 m. The flow velocity at the inlet is 0.55 m/s.

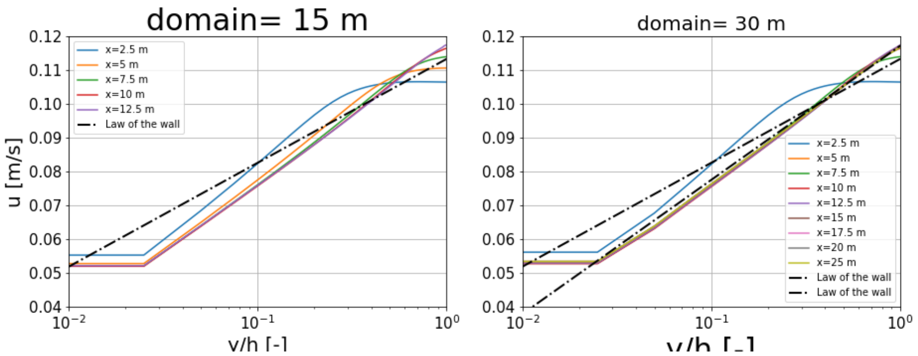


Figure E.8: Flow velocity over the depth and the length for 15 m and 30 m. The flow velocity at the inlet is 0.55 m/s.

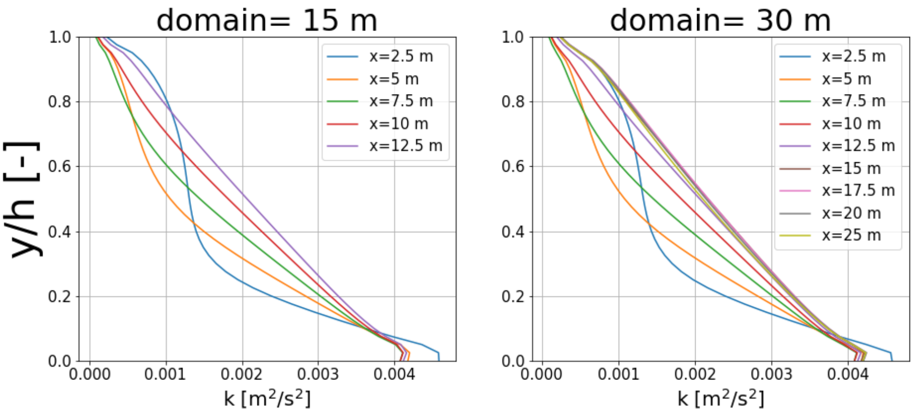


Figure E.9: Turbulent kinetic energy over the depth and length for 15 m and 30 m. The flow velocity at the inlet is 0.55 m/s.

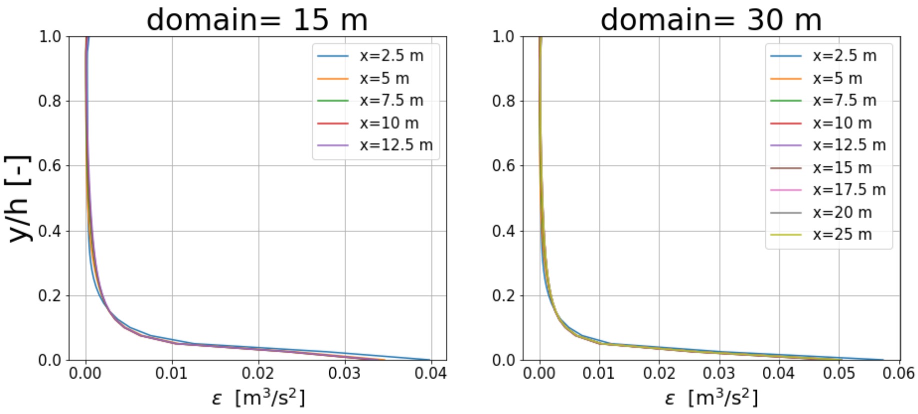


Figure E.10: Turbulent energy dissipation over the depth and length for 15 m and 30 m. The flow velocity at the inlet is 0.55 m/s.

E.1.3. Flow parameters high flow

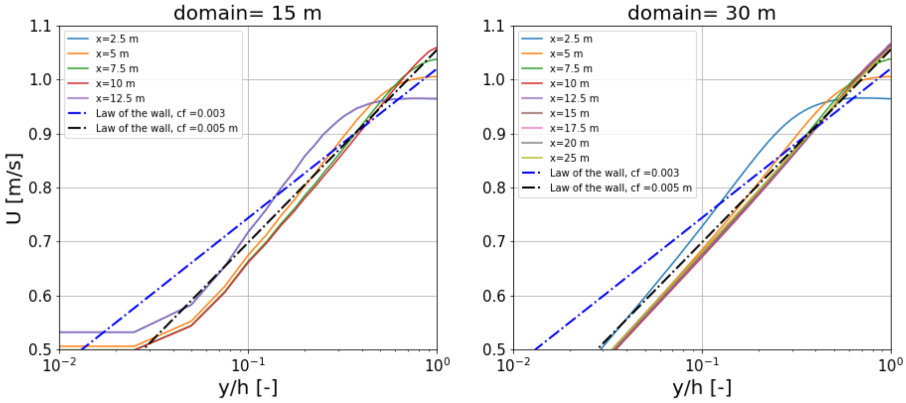


Figure E.11: Flow velocity over the depth and the length for 15 m and 30 m. The flow velocity at the inlet is 0.90 m/s.

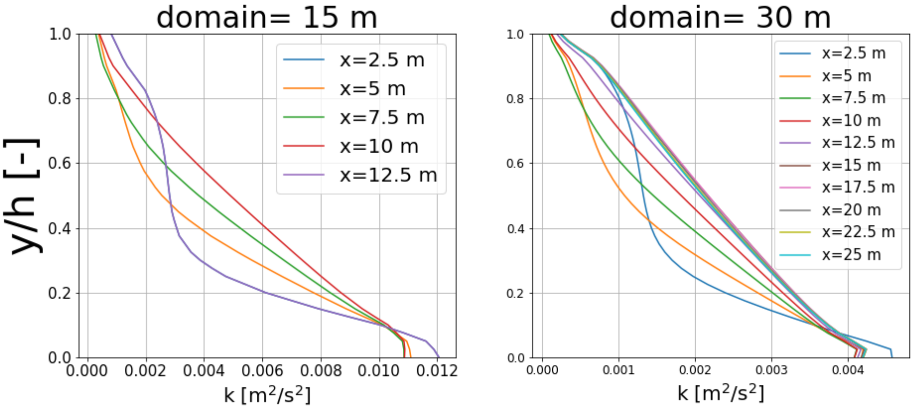


Figure E.12: Turbulent kinetic energy over the depth and length for 15 m and 30 m. The flow velocity at the inlet is 0.90 m/s.

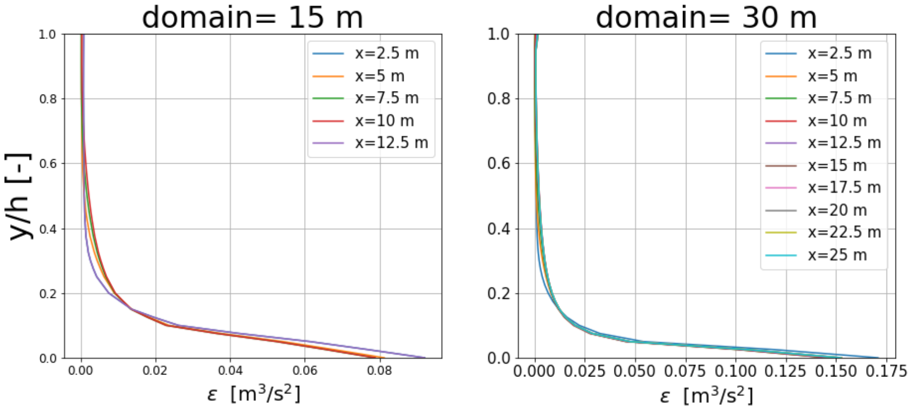


Figure E.13: Turbulent energy dissipation over the depth and length for 15 m and 30 m. The flow velocity at the inlet is 0.90 m/s.

E.2. Rouse numbers simulations

Table E.1: Rouse numbers simulations

Specified particle	**Diameter [mm]**	**Density [kg/m^3]**	**Rise velocity [m/s]**
Neutrally buoyant particles	0	0	0
Light particles	61	11	6.7
HDPE spheres (Rouse)	4.45	0.81	0.49
HDPE films	2.85	0.52	0.32
PP spheres (Rouse)	5.57	1.01	0.62
PP films	3.56	0.65	0.40

E.3. Two-phase flow: equilibria HDPE resin

E.3.1. Entrance length

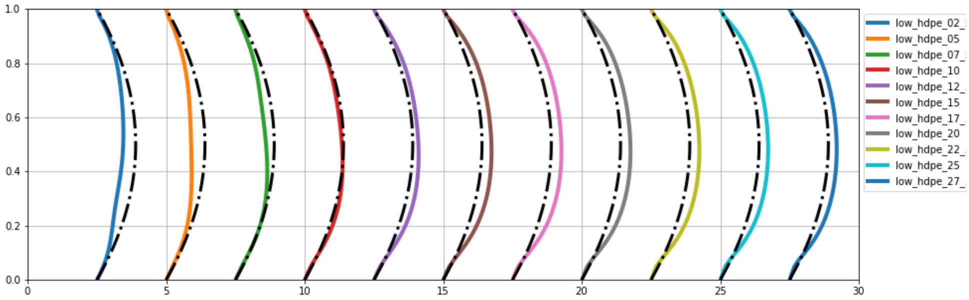


Figure E.14: Turbulent eddy viscosity over length and depth over the domain for low flow conditions. The eddy viscosity is scaled x10000 for viewing purposes.

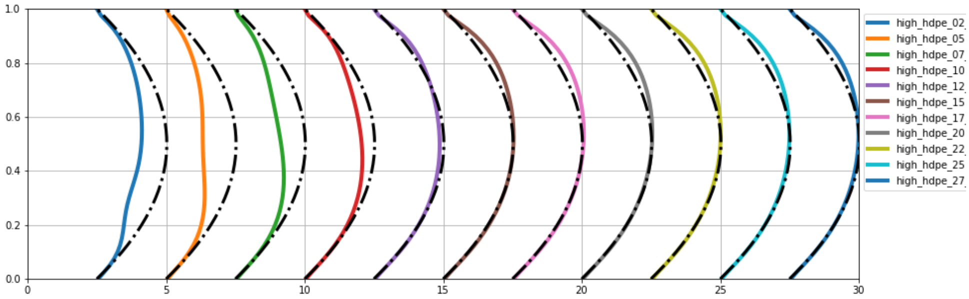


Figure E.15: Turbulent eddy viscosity over length and depth over the domain for high flow conditions. The eddy viscosity is scaled x2000 for viewing purposes.

E.3.2. Concentration profiles

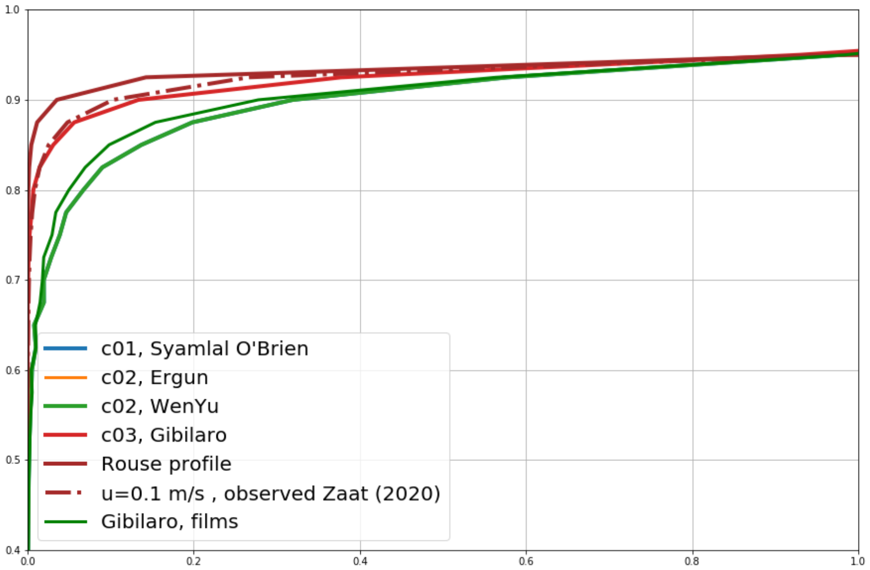


Figure E.16: Relative HDPE particle concentration over the depth for low flow conditions.

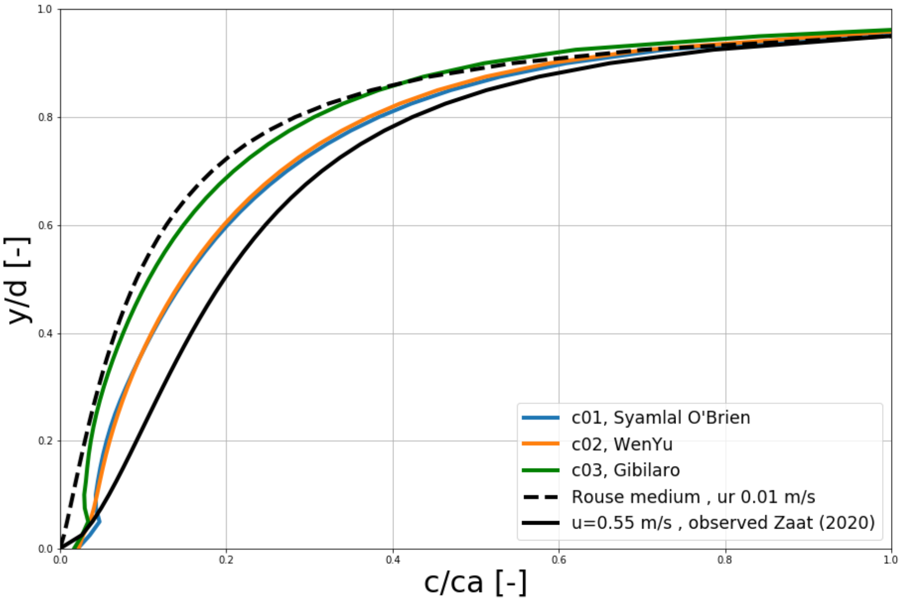


Figure E.17: Relative HDPE particle concentration over the depth for medium flow conditions.

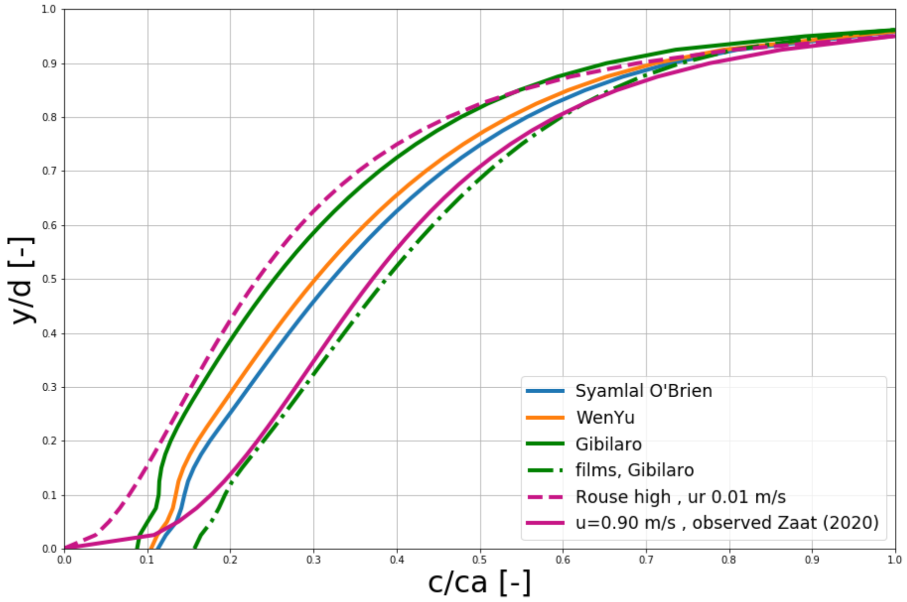


Figure E.18: Relative HDPE particle concentration over the depth for high flow conditions.

E.4. Two-phase flow: equilibria PP resin

E.4.1. Spin-up time to steady state solution

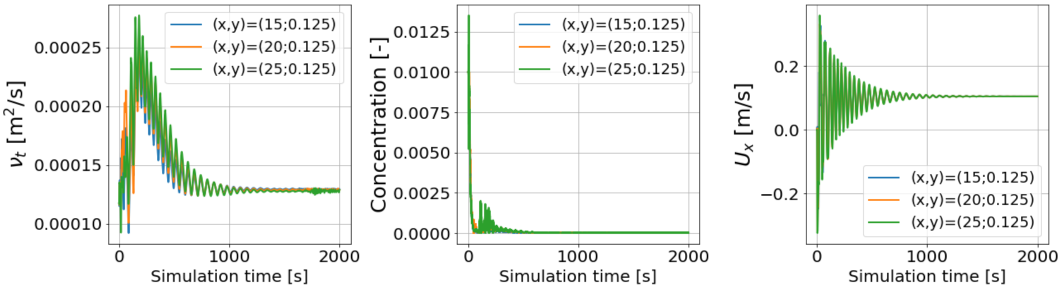


Figure E.19: Simulation time (0 s < t < 2000 s) to steady simulation for different parameters for low flow conditions.

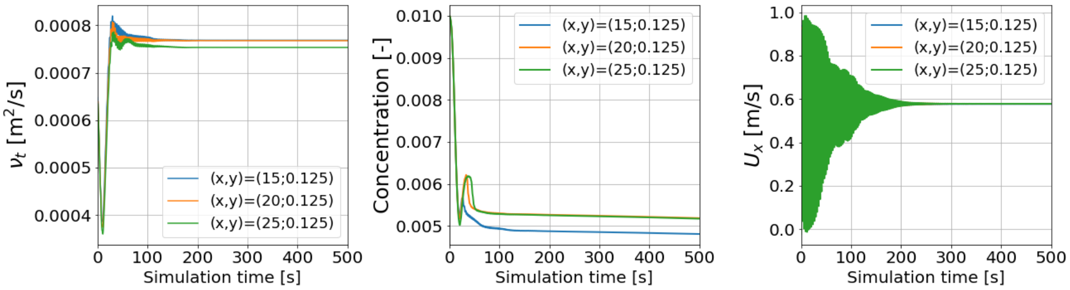


Figure E.20: Simulation time (0 s < t < 500 s) to steady simulation for different parameters for medium flow conditions.

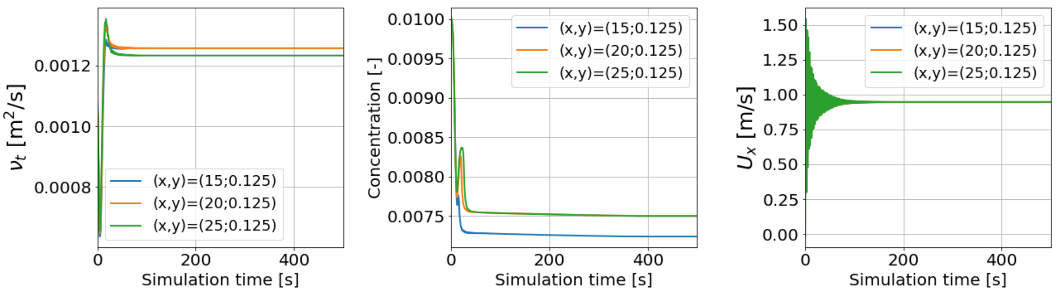


Figure E.21: Simulation time (0 s < t < 500 s) to steady simulation for different parameters for high flow conditions.

E.4.2. Turbulent eddy viscosity

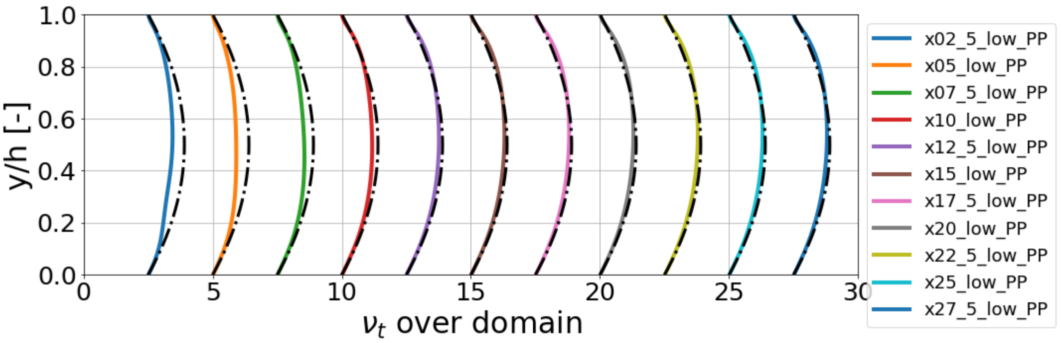


Figure E.22: Turbulent eddy viscosity over length and depth over the domain for low flow conditions. The eddy viscosity is scaled x10000 for viewing purposes.

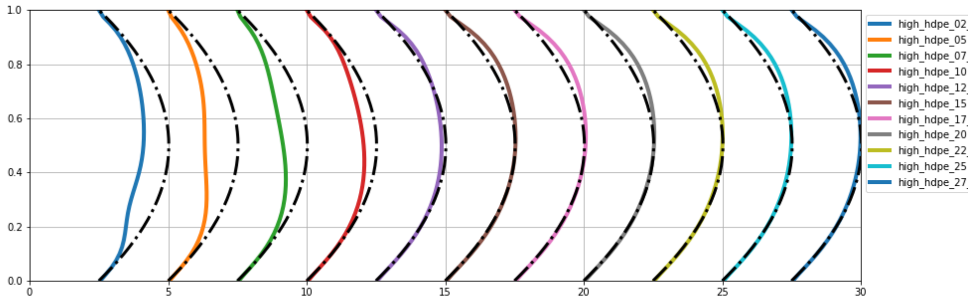


Figure E.23: Turbulent eddy viscosity over length and depth over the domain for high flow conditions. The eddy viscosity is scaled x2000 for viewing purposes.

E.5. Hydrodynamics interception device

E.5.1. Hydrodynamics low flow velocity

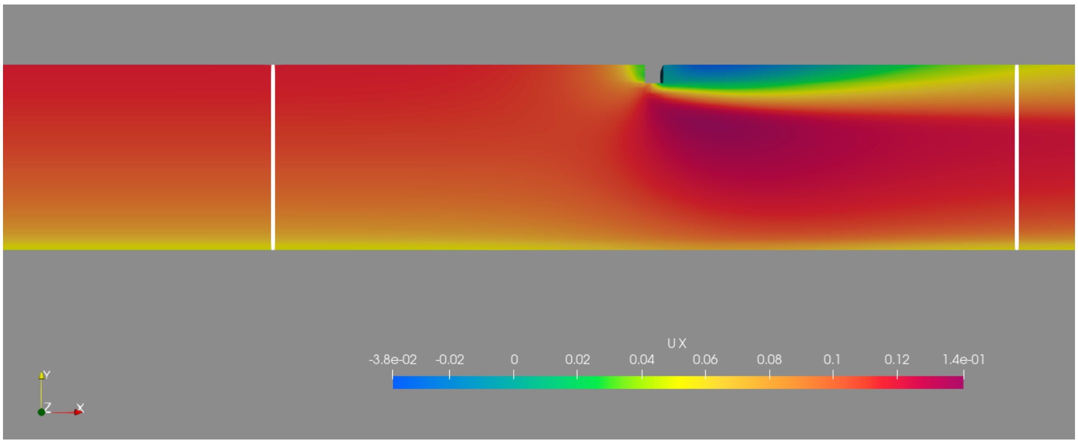


Figure E.24: Horizontal velocity for $u_x = 0.10$ m/s. The vertical lines plotted mark x=24.5 m to x=25.5 m.

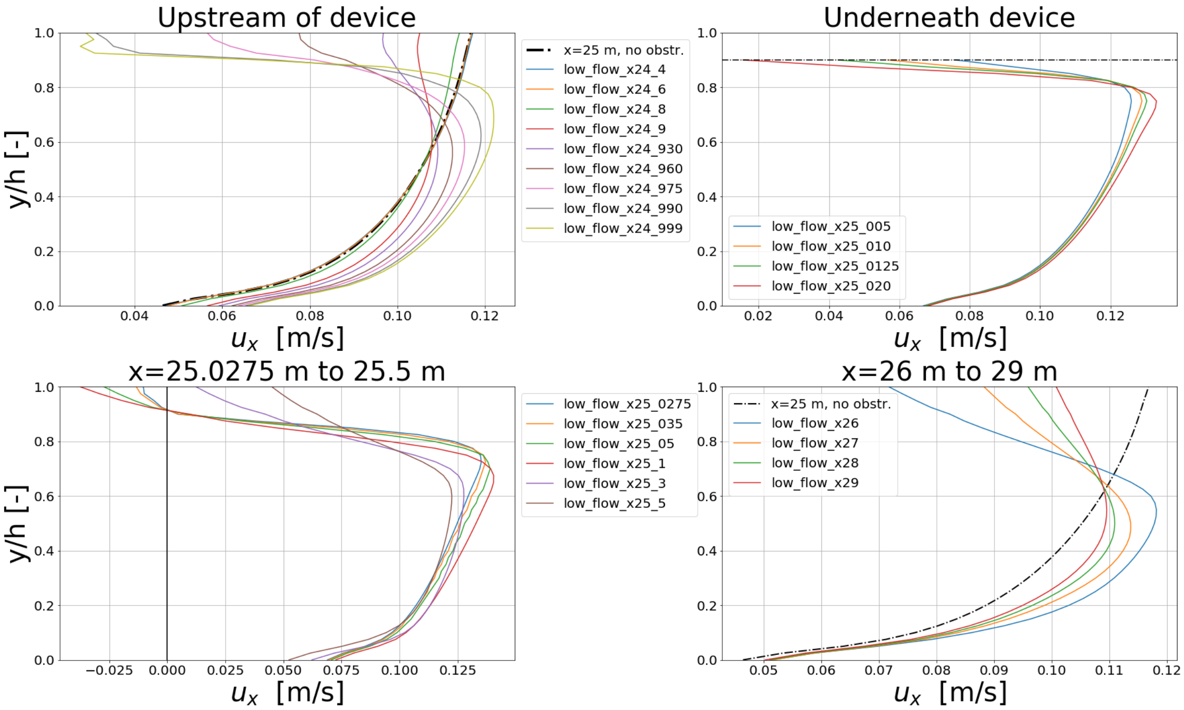


Figure E.25: Horizontal flow velocity over the depth at different locations around the interception device. The dotted line in the upper right figure indicates the draft of the interception device. The vertical line in the lower left figure indicates u_x =0 m/s.

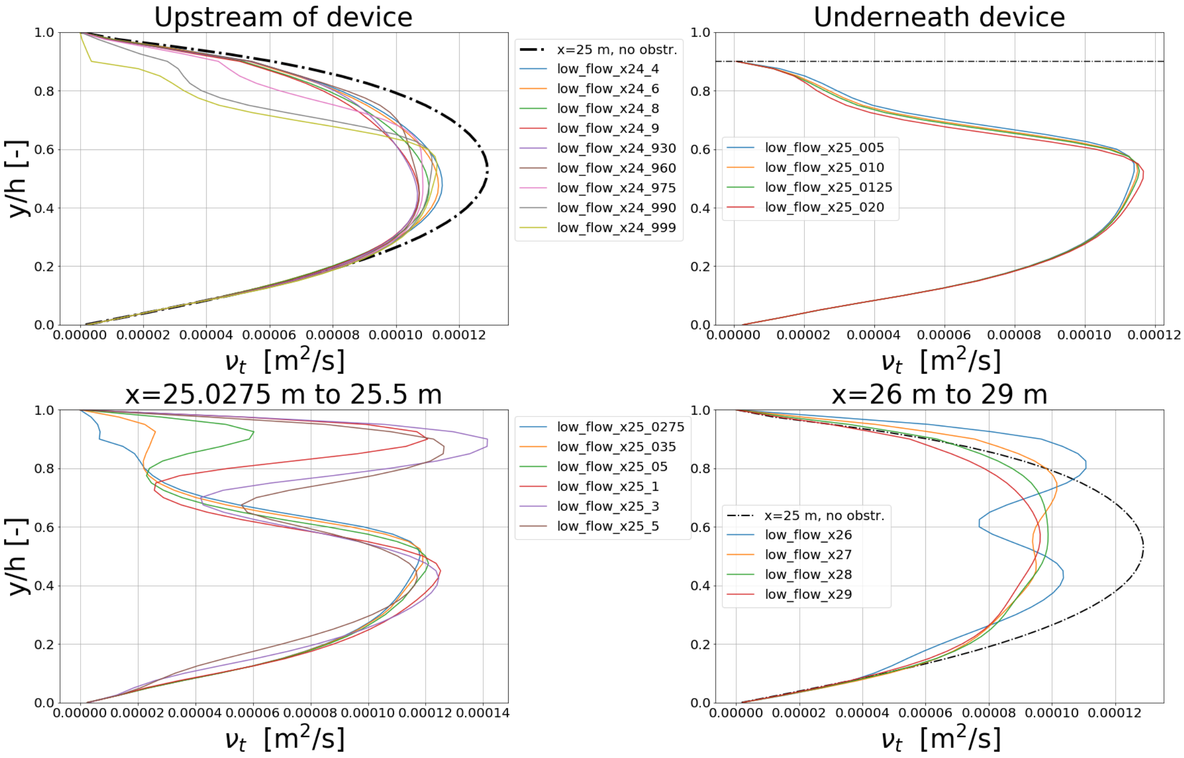


Figure E.26: Turbulent eddy viscosity over the depth at different locations around the interception device.

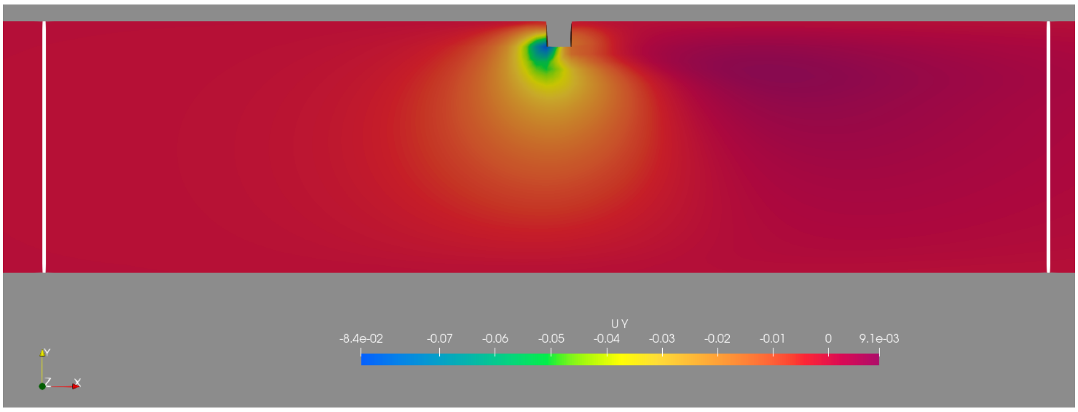
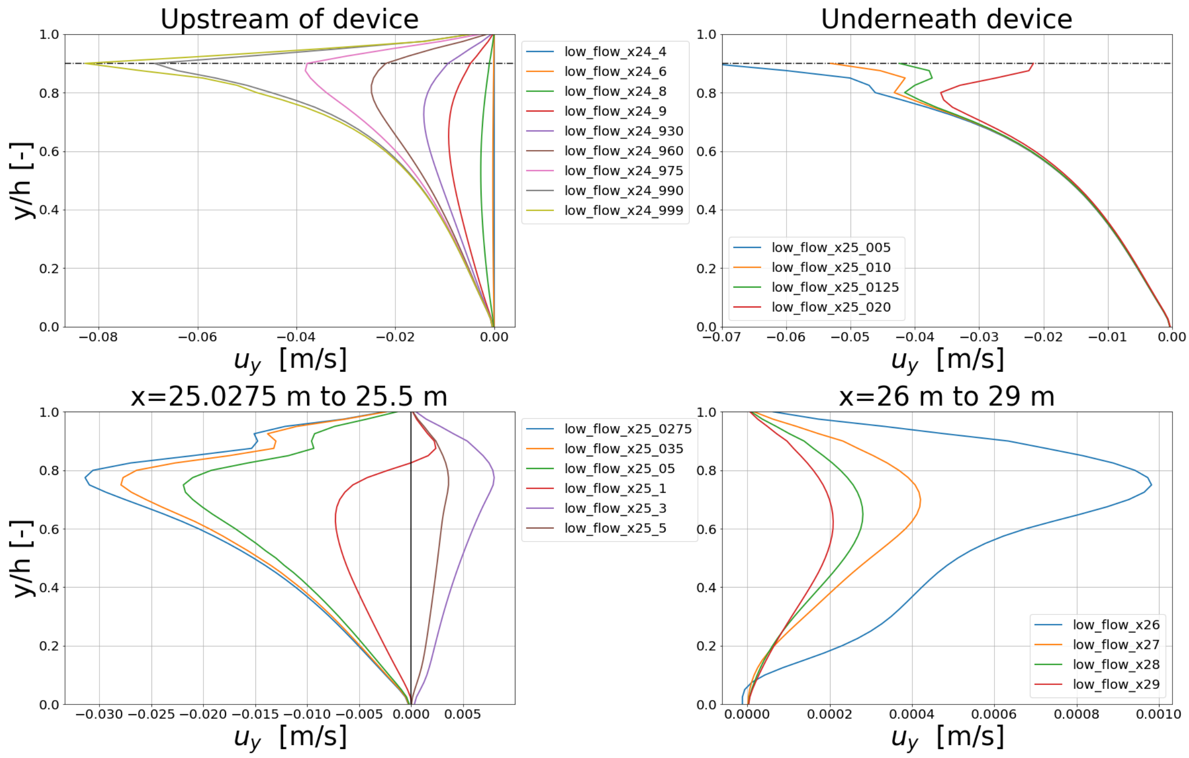


Figure E.27: The vertical velocity for $u_x = 0.10$ m/s. The vertical lines plotted mark x=24.5 m to x=25.5 m. The dotted line in the upper right figure indicates the draft of the interception device. This line is also indicated to illustrate the impact of the device for the flow in front of the device in the upper left figure. The vertical line in the lower left figure indicates u_y =0 m/s.



Upstream of device

Underneath device

x=25.0275 m to 25.5 m

x=26 m to 29 m

Figure E.28: Vertical flow velocity profile over the depth at different locations around the interception device for low flow.

E.5.2. Hydrodynamics high flow velocity

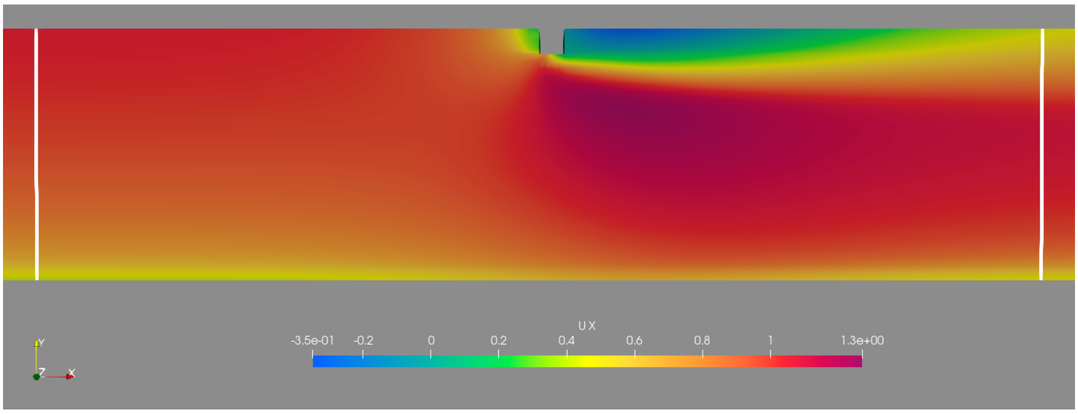


Figure E.29: The horizontal velocity for u_x = 0.90 m/s. The vertical lines plotted mark x=24.5 m to x=25.5 m.

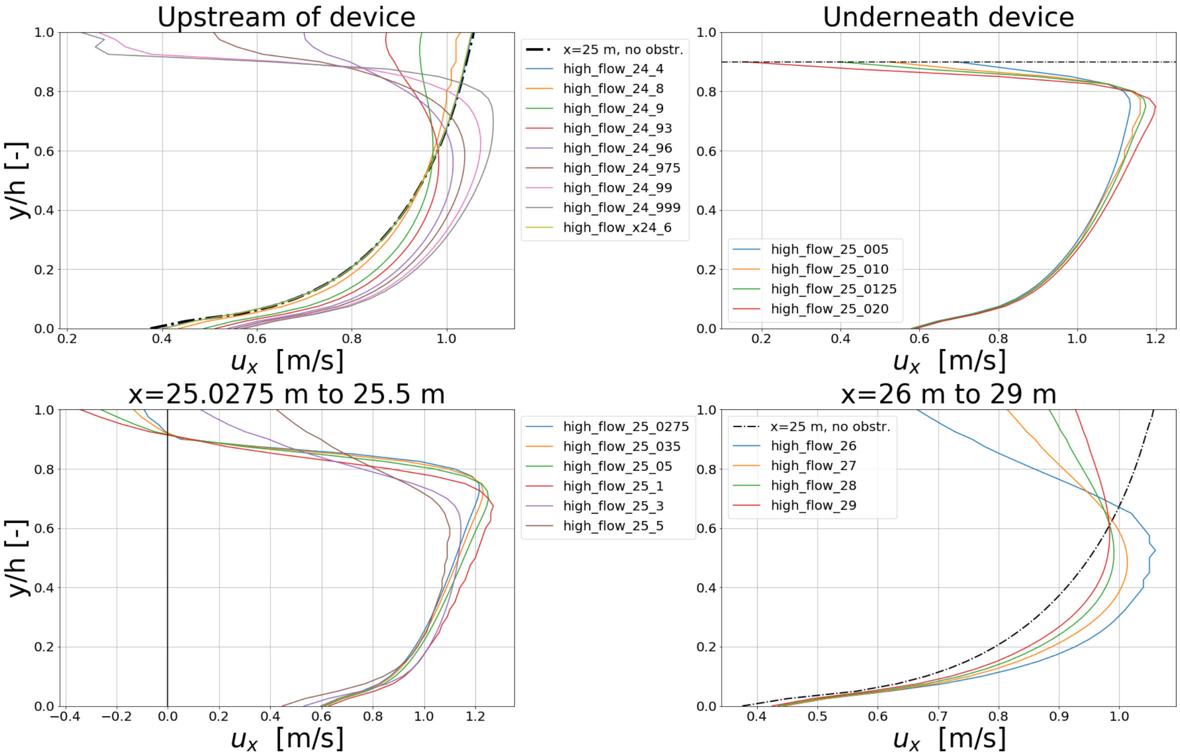


Figure E.30: Horizontal flow velocity over the depth at different locations around the interception device. The dotted line in the upper right figure indicates the draft of the interception device. The vertical line in the lower left figure indicates u_x =0 m/s.

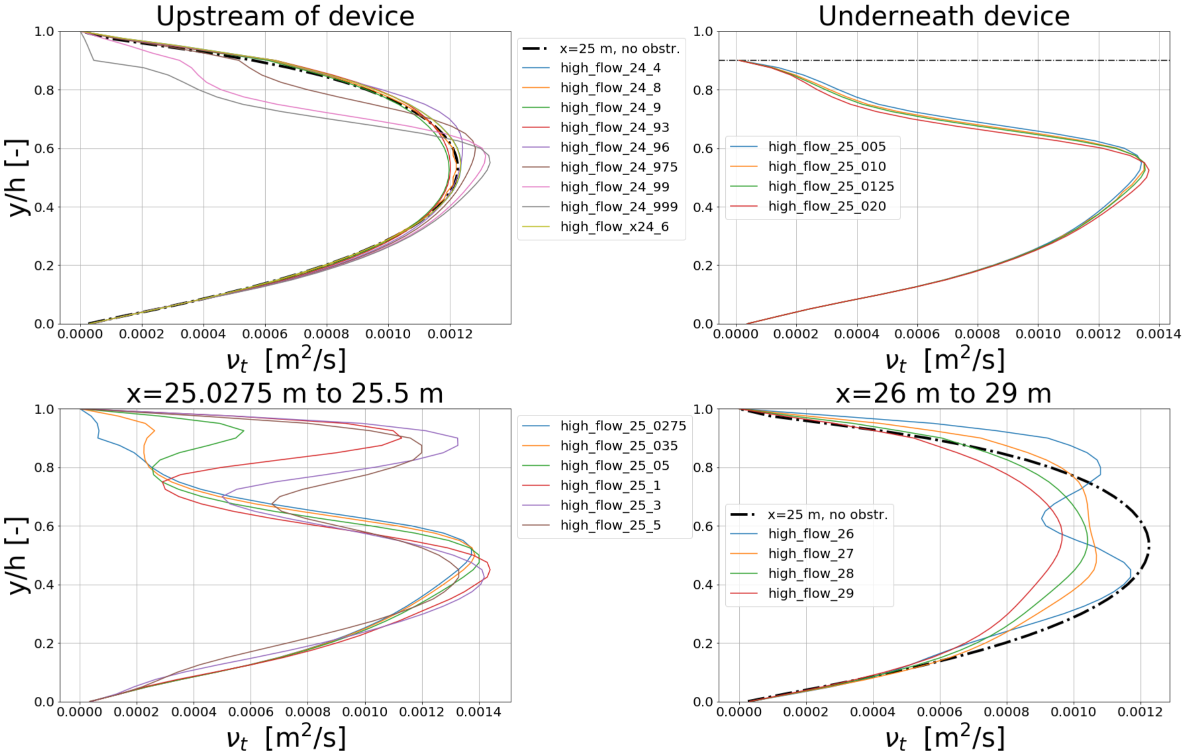


Figure E.31: Turbulent eddy viscosity over the depth at different locations around the interception device.

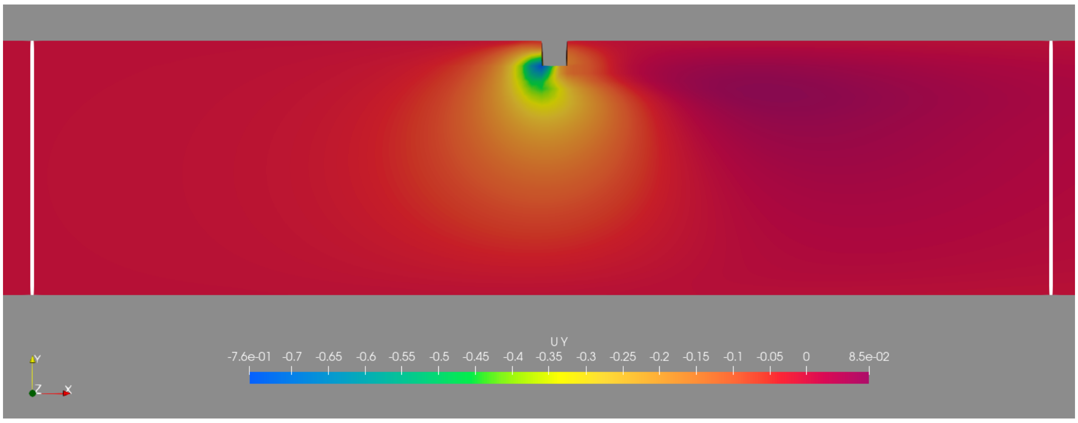


Figure E.32: The vertical velocity for $u_x = 0.90$ m/s. The vertical lines plotted mark x=24.5 m to x=25.5 m.

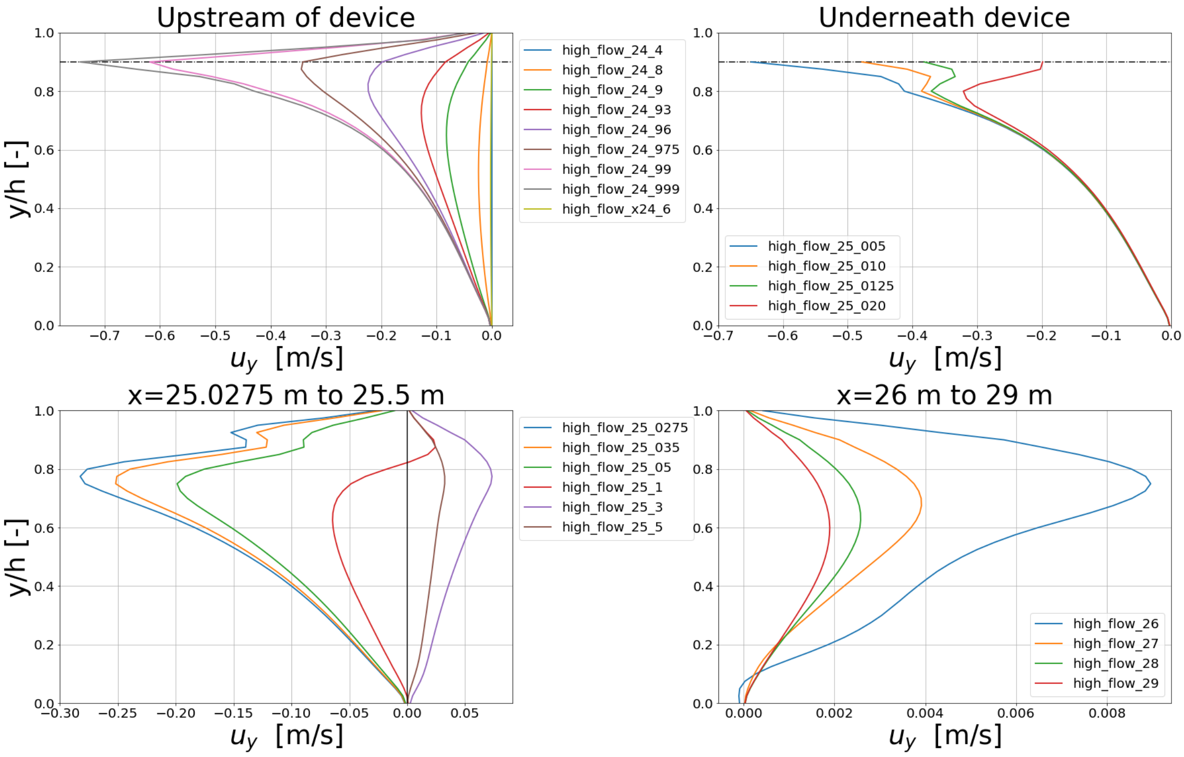
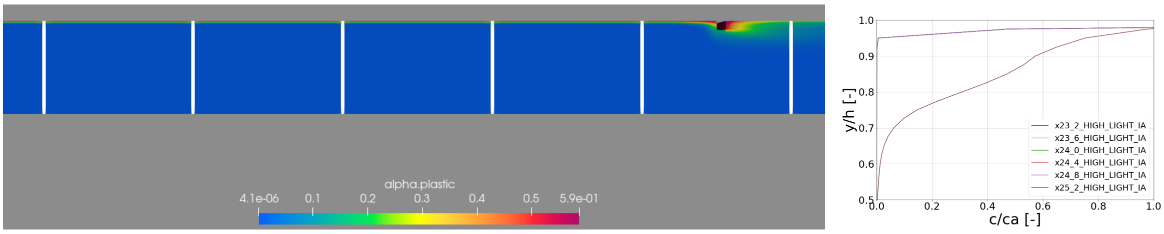


Figure E.33: Vertical flow velocity profile over the depth at different locations around the interception device for high flow. The dotted line in the upper right figure indicates the draft of the interception device. This line is also indicated to illustrate the impact of the device for the flow in front of the device in the upper left figure. The vertical line in the lower left figure indicates u_y =0 m/s.

E.6. Interception of particles

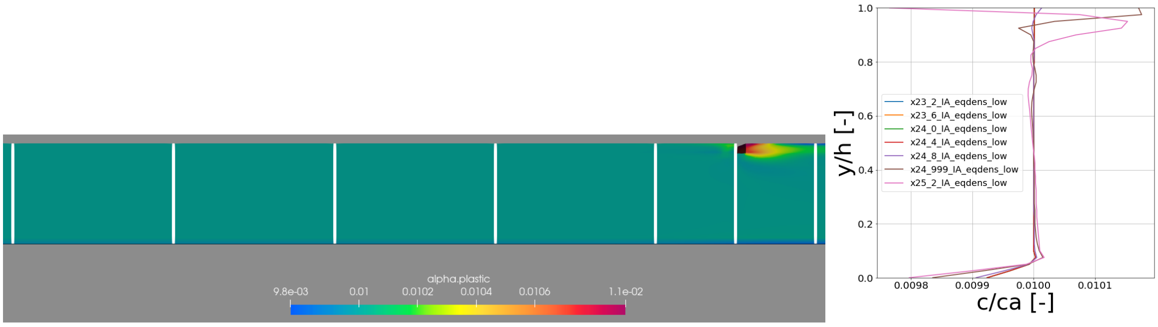
E.6.1. Light particles



(a) Model results. (b) Relative concentration over the depth.

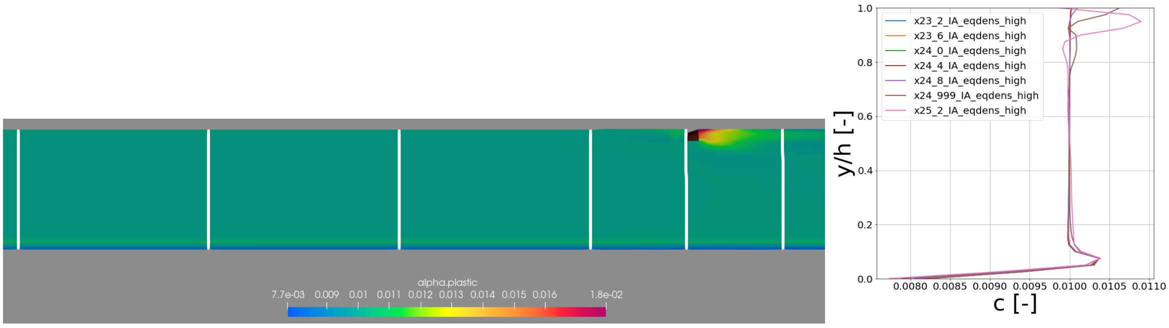
Figure E.34: The concentration of the plastic material with profiles taken between x=23.2 m and x=25.2 m for high flow conditions.

E.6.2. Neutrally buoyant particles



(a) Model results. (b) Relative concentration over the depth.

Figure E.35: The concentration of the plastic material with profiles taken between x=23.2 m and x=25.2 m for low flow conditions.



(a) Model results. (b) Relative concentration over the depth.

Figure E.36: The concentration of the plastic material with profiles taken between x=23.2 m and x=25.2 m for high flow conditions.

E.6.3. HDPE films

E.6.4. PP films

F

Code applied

F.1. File structure of OpenFOAM

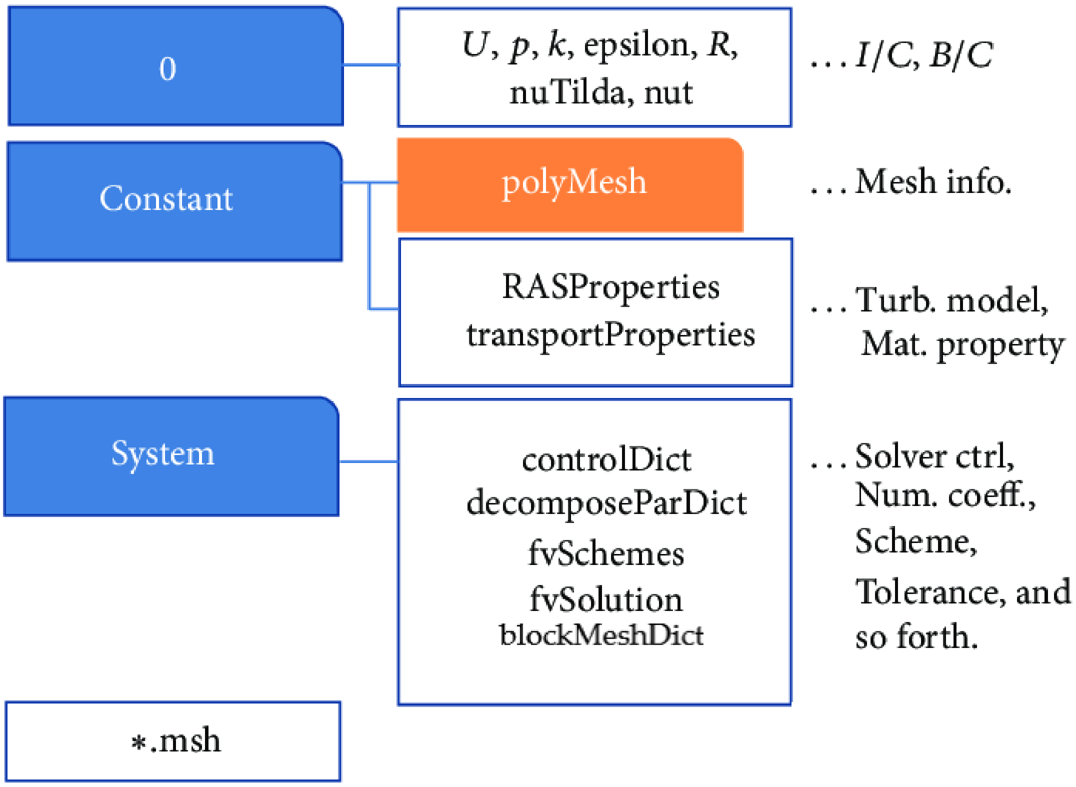


Figure F.1: Typical file structure in OpenFOAM. From Chang et al. (2016)

F.2. Shell file: file to run simulation

```

#PBS !/bin/bash
#PBS -l nodes=1:ppn=20
```

```
#PBS -q guest
#PBS-N OF_MP_HIGH_hdpe_c01
#PBS -m bea


# set environment variables for OpenFOAM
source ~/../../opt/ud/OpenFOAM-v2006/OpenFOAM-v2006/etc/bashrc

#change to current directory

cd ~/multiphase/HDPE/high/c01

ulimit -s unlimited


# start of command list, where <NPROC> will be changed to no. of cores

cp -r 0.orig 0 #copy initial conditions

blockMesh #builds mesh

setFields #applies initial conditions to field

decomposePar  #decomposes simulation into slices

#Run simulation: 20 nodes defined, logs the simulation into txt file
mpirun -np 20 reactingTwoPhaseEulerFoam -parallel > log.twophase

reconstructPar

rm -rf ./processor* #removes folders with partitions


#Create FOAM file for ParaView Post-processing
touch results.foam

# end of command list

 // * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

F.3. Initial conditions

These files are listed in an 0.orig folder and copied to a 0 folder in the command file.

The single-phase applications used the following files:

- U.water
- k.water
- epsilon.water
- nut.water
- p

The “.water” file extension was removed for the application in the `simpleFoam` model. The “walls” initial and boundary conditions were added when the interception system was added in the mesh, with the boundaries as given as in Tab. 3.4.

F.3.1. alpha.plastic

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    ”0”;
    object      alpha.plastic;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 0 0 0 0 0 0];


internalField   uniform 0.01;

boundaryField
{
    inlet
    {
        type            fixedValue;
        value           uniform 0.01;
    }
    outlet
    {
        type            inletOutlet;
        inletValue      uniform 0;
        value           uniform 0;
    }
    bottom
    {
        type            zeroGradient;
    }
    surface
    {
        type            zeroGradient;
    }
    frontAndBackPlanes
    {
        type            empty;
    }
}


// ************************************************************************* //
```

F.3.2. alpha.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    ”0”;
    object      alpha.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 0 0 0 0 0 0];


internalField   uniform 0.99;

boundaryField
{
    inlet
    {
        type            fixedValue;
        value           uniform 0.99;
    }
    outlet
    {
        type            inletOutlet;
        inletValue      uniform 0;
        value           uniform 0;
    }
    bottom
    {
        type            zeroGradient;
    }
    surface
    {
        type            zeroGradient;
    }
    frontAndBackPlanes
    {
        type            empty;
    }
}


// ************************************************************************* //
```

F.3.3. alphat.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      alphat.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [1 -1 -1 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    inlet
    {
        type            calculated;
        value           $internalField;
    }

    outlet
    {
        type            calculated;
        value           $internalField;
    }

    bottom
    {
        type            compressible::alphatWallFunction;
        Prt             0.85;
        value           $internalField;
    }

        surface
    {
        type            compressible::alphatWallFunction;
        Prt             0.85;
        value           $internalField;
    }

    defaultFaces
    {
        type            empty;
    }
}

// ************************************************************************* //
```

F.3.4. alphat.plastic

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      alphat.plastic;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [1 -1 -1 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    inlet
    {
        type            calculated;
        value           $internalField;
    }

    outlet
    {
        type            calculated;
        value           $internalField;
    }

    surface
    {
        type            calculated;
        value           $internalField;
    }

        bottom
    {
        type            calculated;
        value           $internalField;
    }

    defaultFaces
    {
        type            empty;
    }
}

// ************************************************************************* //
```

F.3.5. epsilon.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      epsilon.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [0 2 -3 0 0 0 0];

internalField       uniform 1.25750074e-02;

boundaryField
{
    inlet
    {
            type    turbulentMixingLengthDissipationRateInlet;
                mixingLength    0.0175;
                value      uniform 200;

                k  k.water; #remove line for single-phase model
    }

    outlet
    {
        type               zeroGradient;
    }

    bottom
    {
        type               epsilonWallFunction;
        value              $internalField;
    }

        surface
    {
        type               epsilonWallFunction;
        value              $internalField;
    }

    defaultFaces
    {
        type               empty;
    }
}

// ************************************************************************* //
```

F.3.6. k.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      k.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [0 2 -2 0 0 0 0];

internalField       uniform 0.01215;

boundaryField
{
    inlet
    {
    type                turbulentIntensityKineticEnergyInlet;
        intensity       0.10; // 10% turbulence intensity
        value           uniform 1; // placeholder

        U                       U.water; #remove line for single-phase model
}
    outlet
    {
        type                   zeroGradient;
    }

    bottom
    {
        type                   kqRWallFunction;
        value                  $internalField;
    }

        surface
    {
        type                   kqRWallFunction;
        value                  $internalField;
    }


    defaultFaces
    {
        type                   empty;
    }
}

// ************************************************************************* //
```

F.3.7. nut.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      nut.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [0 2 -1 0 0 0 0];

internalField       uniform 0;

boundaryField
{
    inlet
    {
        type               calculated;
        value              $internalField;
    }

    outlet
    {
        type               calculated;
        value              $internalField;
    }

    bottom
    {
        type                    nutkRoughWallFunction;
                Ks                          uniform 0.004576;
                Cs              uniform 0.5;
                value           $internalField;
    }

        surface
    {
        type               nutkWallFunction;
        value              $internalField;
    }


    defaultFaces
    {
        type               empty;
    }
}


// ************************************************************************* //
```

F.3.8. nut.plastic

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      nut.plastic;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [0 2 -1 0 0 0 0];

internalField       uniform 0;

boundaryField
{
    inlet
    {
        type               calculated;
        value              $internalField;
    }

    outlet
    {
        type               calculated;
        value              $internalField;
    }

    surface
    {
        type               nutkWallFunction;
        value              $internalField;
    }

        bottom
    {
        type                    nutkRoughWallFunction;
                Ks                          uniform 0.004576;
                Cs              uniform 0.5;
                value           $internalField;
    }

    defaultFaces
    {
        type               empty;
    }
}

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

F.3.9. p

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      p;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [1 -1 -2 0 0 0 0];

internalField       uniform 1e5;

boundaryField
{
    inlet
    {
        type            calculated;
        value           $internalField;
    }

    outlet
    {
        type            calculated;
        value           $internalField;
    }

    surface
    {
        type               calculated;
        value              $internalField;
    }

        bottom
    {
        type               calculated;
        value              $internalField;
    }

    frontAndBackPlanes
    {
        type               empty;
    }
}

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

F.3.10. p_rgh

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      p_rgh;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [1 -1 -2 0 0 0 0];

internalField       uniform 1e5;

boundaryField
{
    inlet
    {
        type            fixedFluxPressure;
        value           $internalField;


    }

    outlet
    {
        type              prghTotalPressure;
                p0                   uniform 1.0e5;
                value                   uniform 1.0e5;
                U       U.water;

    }

    surface
    {
        type               fixedFluxPressure;
        value                          $internalField;
    }

        bottom
    {
        type               fixedFluxPressure;
        value              $internalField;

    }

    frontAndBackPlanes
    {
        type               empty;
    }
}
```

```
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

F.3.11. T.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      T.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [0 0 0 1 0 0 0];

internalField       uniform 300;

boundaryField
{
    inlet
    {
        type                   fixedValue;
        value                  uniform 300;
    }

    outlet
    {
        type                   zeroGradient;
        phi                    phi.water;
        inletValue             uniform 300;
        value                  uniform 300;
    }

    bottom
    {
        type                   zeroGradient;
    }

        surface
    {
        type                   zeroGradient;
    }

    defaultFaces
    {
        type                   empty;
    }
}
```

```
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

F.3.12. T.plastic

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      T.plastic;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [0 0 0 1 0 0 0];

internalField       uniform 300;

boundaryField
{
    inlet
    {
        type                zeroGradient;
    }

    outlet
    {
        type                zeroGradient;
        phi                 phi.plastic;
        inletValue          uniform 300;
        value               uniform 300;
    }

    surface
    {
        type                zeroGradient;
    }

        bottom
    {
        type                zeroGradient;
    }

    defaultFaces
    {
        type                empty;
    }
}
```

```
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

F.3.13. Theta.plastic

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      Theta.plastic;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions          [0 2 -2 0 0 0 0];

internalField       uniform 0;

referenceLevel      1e-4;

boundaryField
{
    inlet
    {
        type               fixedValue;
        value              uniform 1e-4;
    }

    outlet
    {
        type               zeroGradient;
    }

    surface
    {
        type               zeroGradient;
    }

        bottom
    {
        type               zeroGradient;
    }

    defaultFaces
    {
        type               empty;
    }
}

// ************************************************************************* //
```

F.3.14. U.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    object      U.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (0 0 0);

boundaryField
{
    inlet
    {
        type               fixedValue;
                value                      uniform (0.9 0 0);

    }

    outlet
    {
        type                pressureInletOutletVelocity;
                phi                             phi.water;
                inletValue      uniform (0 0 0);
                value                           uniform (0 0 0 );
    }

    surface
    {
        type               slip;
    }

        bottom
    {
        type               noSlip;
    }


    defaultFaces
    {
        type               empty;
    }
}
```

```
// ************************************************************************* //
```

F.3.15. U.plastic

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    object      U.plastic;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (0.0 0 0);

boundaryField
{
    inlet
    {
        type               fixedValue;
                value                      uniform (0.9 0 0);

    }

    outlet
    {
        type            pressureInletOutletVelocity;
                inletValue      uniform (0 0 0);
        value           uniform (0 0 0);
    }

    surface
    {
        type               slip;
    }

        bottom
    {
        type               noSlip;
    }

    defaultFaces
    {
        type               empty;
    }
}
```

```
// ************************************************************************* //
```

F.4. Constants

F.4.1. g

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       uniformDimensionedVectorField;
    location    ”constant”;
    object      g;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 1 -2 0 0 0 0];
value           (0 -9.8 0);


// ************************************************************************* //
```

F.4.2. hRef

F.4.3. phaseProperties

F.4.4. thermophysicalProperties.plastic

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
```

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    ”constant”;
    object      thermophysicalProperties.plastic;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

thermoType
{
    type            heRhoThermo;
    mixture         pureMixture;
    transport       const;
    thermo          hConst;
    equationOfState rhoConst; // rhoconst = incompressible , perfectfluid = nno density
    specie          specie;
    energy          sensibleInternalEnergy;
}


mixture
{
    specie
    {
        molWeight   100;
    }
    equationOfState
    {
        rho         970;


    }
    thermodynamics
    {
        Cp          50000;
        Hf          0;
    }
    transport
    {
        mu          0.01;
        Pr         1;
    }
}


// ************************************************************************* //
```

F.4.5. thermophysicalProperties.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
```

```
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      thermophysicalProperties.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

thermoType
{
    type            heRhoThermo;
    mixture         pureMixture;
    transport       const;
    thermo          hConst;
    equationOfState rhoConst; // rhoconst = incompressible , perfectfluid = nno density
    specie          specie;
    energy          sensibleEnthalpy;
}

mixture
{
    specie
    {
        molWeight   100;
    }

        equationOfState
    {
        rho             1000;

    }

    thermodynamics
    {
        Cp          6000;
        Hf          0;
    }
    transport
    {
        mu           1.00e-3;
        Pr           2.289;
    }
}


// ************************************************************************* //
```

F.4.6. turbulenceProperties.water

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
```

```
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    ”constant”;
    object      turbulenceProperties.water;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //


simulationType  RAS;

RAS
{
    RASModel kEpsilon;

    turbulence      on;
    printCoeffs     on;
}




// ************************************************************************* //
```

F.4.7. turbulenceProperties.plastic

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    ”constant”;
    object      turbulenceProperties.particles;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

simulationType  RAS;

RAS
{
 RASModel  kineticTheory;

    turbulence      on;
```

```
    printCoeffs     on;

    kineticTheoryCoeffs
    {
        equilibrium             off;

        e                       0.8;
        alphaMax                0.99;
        alphaMinFriction        0.5;
        residualAlpha           1e-4;

        viscosityModel          Gidaspow;
        conductivityModel       Gidaspow;
        granularPressureModel   Lun;
        frictionalStressModel   JohnsonJackson;
        radialModel             SinclairJackson;

        JohnsonJacksonCoeffs
        {
            Fr                      0.05;
            eta                     2;
            p                       5;
            phi                     28.5;
            alphaDeltaMin           0.05;
        }
    }

    phasePressureCoeffs
    {
        preAlphaExp     500;
        expMax          1000;
        alphaMax        0.99;
        g0              1000;
    }
}


// ************************************************************************* //
```

F.5. System files

F.5.1. controlDict

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
```

```
    location    ”system”;
    object      controlDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

application     reactingTwoPhaseEulerFoam;

startFrom       latestTime;

startTime       0;

stopAt          endTime;

endTime         500;

deltaT          0.05;

writeControl    runTime;

writeInterval   50;

purgeWrite      0;

writeFormat     ascii;

writePrecision  6;

writeCompression off;

timeFormat      general;

timePrecision   6;

runTimeModifiable yes;

adjustTimeStep  yes;

maxCo           0.99;

maxAlphaCo      0.99;


// maxDeltaT       1e-05;

yPlus
    {
        type            yPlus;
        libs            (fieldFunctionObjects);
        writeControl    writeTime;
    }
functions {
probes
    {
        type probes;

                // Where to load it from
        libs (”libsampling.so”);

                // Write at same frequency as fields
```

```
                writeControl    timeStep;
                writeInterval   50;

                // Fields to be probed

                fields  (p nut.water U.water alpha.plastic);

            probeLocations
                (
                        ( 15.0 0.125 0 )         // halfway x=15 m

                        ( 20.0 0.125 0 )         // halfway x=20 m

                        ( 25.0 0.125 0 )         // halfway x=25 m

                        ( 15.0 0.250 0 )         // near-surface x =15 m
                        ( 20.0 0.250 0 )         // near-surface x =20 m

                        ( 22.0 0.250 0 )         // near-surface x =22 m

                        ( 25.0 0.250 0 )         // near-surface x =25 m
                );
}

}


// ************************************************************************* //
```

F.5.2. blockMeshDict for mesh without interception system

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      blockMeshDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

scale   1;

vertices
(
    (0 0 -0.1)
    (30 0 -0.1)
    (30 0.25 -0.1)
    (0 0.25 -0.1)
    (0 0 0.1)
```

```
    (30 0 0.1)
    (30 0.25 0.1)
    (0 0.25 0.1)
);

blocks
(
    hex (0 1 2 3 4 5 6 7) (600 40 1) simpleGrading (1 1 1)
);

edges
(
);

patches
(
    patch inlet
    (
        (0 4 7 3)
    )
    patch outlet
    (
        (2 6 5 1)
    )

        wall bottom
        (
                        (1 5 4 0)
        )

        wall surface
        (
                (3 7 6 2)
    )



        empty frontAndBackPlanes
    (
        (0 3 2 1)
        (4 5 6 7)
    )
);

mergePatchPairs
(
);

// ************************************************************************* //
```

F.5.3. blockMeshDict for mesh with interception system

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
```

```
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      blockMeshDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

scale   1;

vertices
(
    (0 0 0)                             // 0
    (25 0 0)                            // 1
    (25.025 0 0)                // 2
    (30 0 0)                            // 3
    (0 0.225 0)         // 4
    (25 0.225 0)                // 5
    (25.025 0.225 0)    // 6
    (30 0.225 0)                // 7
    (0 0.25 0)                          // 8
    (25 0.25 0)                         // 9
    (25.025 0.25 0)             // 10
    (30 0.25 0)                                 // 11

    (0 0 0.1)                           // 12
    (25 0 0.1)                          // 13
    (25.025 0 0.1)                      // 14
    (30 0 0.1)                          // 15
    (0 0.225 0.1)                       // 16
    (25 0.225 0.1)                      // 17
    (25.025 0.225 0.1)          // 18
    (30 0.225 0.1)                      // 19
    (0 0.25 0.1)                                // 20
    (25 0.25 0.1)                               // 21
    (25.025 0.25 0.1)                   // 22
    (30 0.25 0.1)                       // 23
);

blocks
(
hex (0 1 5 4 12 13 17 16) (400 30 1) simpleGrading (0.05 1 1) //lower left
hex (2 3 7 6 14 15 19 18) (200 30 1) simpleGrading (10 1 1)  //lower right
hex (4 5 9 8 16 17 21 20) (400 10 1) simpleGrading (0.05 1 1) //top left
hex (1 2 6 5 13 14 18 17) (4 30 1) simpleGrading (1 1 1) //under the system
hex (6 7 11 10 18 19 23 22) (200 10 1) simpleGrading (10 1 1) //top right
);

edges
(
);

boundary
(
    inlet
```

```
    {
        type wall;
        faces
        (
            (0 12 16 4)
            (4 16 20 8)
        );
    }
    outlet
    {
        type wall;
        faces
        (
            (7 19 15 3)
            (11 23 19 7)
        );
    }
    bottom
    {
        type wall;
        faces
        (
            (0 1 13 12)
                        (1 2 14 13)
            (2 3 15 14)
        );
    }
    surface
    {
        type wall;
        faces
        (
            (8 20 21 9)
                        (5 9 21 17)
            (5 6 18 17)
                        (6 18 22 10)
            (10 22 23 11)
        );
    }
);


mergePatchPairs
(
);

// ************************************************************************* //
```

F.5.4. decomposeParDict

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  1.5                                   |
|   \\  /    A nd           | Web:      http://www.OpenFOAM.org               |
|    \\/     M anipulation  |                                                 |
```

```
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      decomposeParDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

numberOfSubdomains 20;

method          simple;


simpleCoeffs
{
    n               (1 20 1); #20 slices over y
    delta           0.001;
}

hierarchicalCoeffs
{
    n               (1 1 1);
    delta           0.001;
    order           xyz;
}

distributed     no;

roots
(
);



// ************************************************************************* //
```

F.5.5. fvSchemes

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    ”system”;
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

```
ddtSchemes
{
    default     Euler;
}

gradSchemes
{
    default     Gauss linear;
}

divSchemes
{
    default                             none;

    "div\(phi,alpha.*\)"                Gauss upwind;
    "div\(phir,alpha.*\)"               Gauss upwind;

    "div\(alphaRhoPhi.*,U.*\)"          Gauss limitedLinearV 1;
    "div\(phi.*,U.*\)"                  Gauss limitedLinearV 1;

    "div\(alphaRhoPhi.*,(h|e).*\)"  Gauss limitedLinear 1;
    "div\(alphaRhoPhi.*,K.*\)"          Gauss limitedLinear 1;
    "div\(alphaPhi.*,p\)"               Gauss limitedLinear 1;

    div(alphaRhoPhi.plastic,Theta.plastic) Gauss limitedLinear 1;

    "div\(alphaRhoPhi.*,(k|epsilon).*\)"  Gauss limitedLinear 1;

div((((alpha.water*thermo:rho.water)*nuEff.water)*dev2(T(grad(U.water)))))

Gauss linear;

div((((alpha.plastic*thermo:rho.plastic)*nuEff.plastic)*dev2(T(grad(U.plastic)))))

Gauss linear;

div((((thermo:rho.plastic*nut.plastic)*dev2(T(grad(U.plastic))))+(((thermo:rho.plastic*lambda.p
Gauss linear;

        div(phim,epsilonm)                                              Gauss linear;
        div(phim,km)                                            Gauss linear;

}

laplacianSchemes
{
    default     Gauss linear corrected;
    bounded     Gauss linear corrected;
}

interpolationSchemes
{
    default     linear;
        interpolate(U.water) linear;
        interpolate(U.plastic) linear;
}

snGradSchemes
```

```
{
    default     corrected;
    bounded     corrected;
}

wallDist
{
    method      meshWave;
}


// ************************************************************************* //
```

F.5.6. fvSolution

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    ”system”;
    object      fvSolution;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

solvers
{
    ”alpha.*”
    {
        nAlphaCorr      1;
        nAlphaSubCycles 2;
        implicitPhasePressure yes;

        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-9;
        relTol          0.001;
        minIter         1;
    }

    p_rgh
    {
        solver          GAMG;
        smoother        DIC;
        tolerance       1e-8;
        relTol          0.01;
    }
```

```
        p_rghFinal
    {
        $p;
        tolerance       1e-06;
        relTol          0;
                solver          smoothSolver;
                smoother        symGaussSeidel;
    }

        alpha.plasticFinal
    {
        $alpha.plastic;
        tolerance       1e-04;
        relTol          0;
                solver          smoothSolver;
                smoother        symGaussSeidel;
    }



    ”U.*”
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-5;
        relTol          0;
        minIter         1;
    }

    ”(h|e).*”
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-6;
        relTol          0;
        minIter         1;
    }

    ”Theta.*”
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-6;
        relTol          0;
        minIter         1;
    }

    ”(k|epsilon).*”
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-5;
        relTol          0;
        minIter         1;
    }
}

PIMPLE
```

```
{
    nOuterCorrectors 3;
    nCorrectors      1;
    nNonOrthogonalCorrectors 0;
    faceMomentum     yes;

residualControl
{
    ”(U.water|k.water|epsilon.water)”
    {
        relTol          0;
        tolerance       1e-4;
    }
}
}


relaxationFactors
{
    equations
    {
        ”U.*”                0.7;
        ”k.*”                0.3;
        ”epsilon.*”          0.3;

                ”.*”                1;
    }
}


// ************************************************************************* //
```

F.5.7. setFieldsdict

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2006                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    ”system”;
    object      setFieldsDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

defaultFieldValues
(
);

regions
```

```
(
    boxToCell
    {
        box (0 0 -0.1) (30 0.25 0.1);
        fieldValues
        (
            volScalarFieldValue alpha.water 0.99
            volScalarFieldValue alpha.plastic 0.01
        );
    }
);






// ************************************************************************* //
```