

**TECHNISCHE UNIVERSITÄT MÜNCHEN**  
Lehrstuhl für Bioverfahrenstechnik

**Computational Fluid Dynamics Assisted Design of Thin-Layer  
Cascade Photobioreactor Components**

**Timm Steffen Severin**

Vollständiger Abdruck der von der Fakultät für Maschinenwesen der Technischen Universität München zur  
Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender: Prof. Dr.-Ing. Hartmut Spliethoff  
Prüfer der Dissertation: 1. Prof. Dr.-Ing. Dirk Weuster-Botz  
2. Prof. Dr.-Ing. Harald Klein

Die Dissertation wurde am 26.06.2017 bei der Technischen Universität München  
eingereicht und durch die Fakultät für Maschinenwesen am 17.11.2017 angenommen



It's still magic even if you know how it's done

SIR TERRY PRATCHETT



# Contents

1	Introduction .....	8
2	Objectives .....	9
3	Cultivation of Microalgae .....	12
3.1	Microalgae .....	12
3.2	Photobioreactor Systems .....	13
3.2.1	Closed Systems .....	15
3.2.2	Open Systems .....	17
3.3	Computational Fluid Dynamics in Algal Photobioreactors .....	19
4	Computational Fluid Dynamics .....	25
4.1	Governing Equations .....	25
4.1.1	Conservation of Mass .....	26
4.1.2	Conservation of Momentum .....	26
4.1.3	Conservation of Energy .....	26
4.1.4	General Scalar Transport Equation .....	26
4.2	Methods .....	27
4.2.1	Finite Differences (FD) .....	27
4.2.2	Finite Volume (FV) .....	29
4.3	Discretisation and Iteration Algorithms .....	31
4.3.1	Temporal Discretisation Algorithms .....	31
4.3.2	Spatial Interpolation Algorithms .....	32
4.3.3	Iterative Solution Algorithms .....	34
4.4	Turbulence .....	38
4.4.1	Reynolds-Averaged Navier-Stokes .....	38
4.4.2	Low Reynolds Turbulence Models .....	40
4.5	Boundary Conditions .....	41
4.5.1	The Inlet Boundary Condition .....	41
4.5.2	The Outlet Boundary Condition .....	42
4.5.3	Walls .....	42
4.5.4	Pressure at Boundaries .....	43
4.5.5	Symmetry .....	43
4.6	Multiphase Systems .....	44
4.6.1	Volume of Fluid Method (VoF) .....	45
4.6.2	Passive Scalar Transport .....	45
5	Experimental and Simulative Environment .....	47
5.1	Thin-Layer Cascade Photobioreactor .....	47
5.1.1	Inlet Tank .....	49
5.1.2	Channel .....	50
5.1.3	Flow Reversal Tank .....	51

5.2	Thin-Layer Cascade Operation .....	52
5.2.1	Microalgae.....	52
5.2.2	Cultivation Conditions .....	53
5.2.3	Operational Constraints.....	54
5.3	Simulation Approach .....	54
5.3.1	Computational Environment.....	54
5.3.2	Pre-processing.....	56
5.3.3	Post-processing .....	56
6	Simulation Setup.....	57
6.1	Mesh Generation .....	57
6.1.1	Inlet .....	57
6.1.2	Channel .....	59
6.1.3	Flow Reversal Tank.....	61
6.2	Solution Procedure .....	64
6.2.1	Two-Phase Flow Simulation .....	64
6.2.2	Two-Phase Flow Simulation with Passive Species Transport .....	64
6.3	Comparison of Turbulence Models .....	68
6.3.1	Cases .....	68
6.3.2	Comparison of Turbulence Effects .....	69
6.3.3	Evaluation of Turbulence Models .....	69
6.4	General Simulation Setup .....	72
6.4.1	Boundary Conditions .....	72
6.4.2	Transport Properties .....	72
6.4.3	Schemes and Algorithm Options .....	72
6.4.4	Runtime Settings.....	76
7	Measurement and Validation Techniques .....	77
7.1	Rapid Salinity Measurement .....	77
7.1.1	Working Principle .....	78
7.1.2	Sensor Design .....	79
7.1.3	Evaluation Method.....	80
7.1.4	Sensor Performance .....	80
7.1.5	Comparison with the CFD Simulation.....	82
7.2	Fluid Height .....	82
7.2.1	Experimental .....	82
7.2.2	Simulative.....	83
7.3	Fluid Velocity.....	85
7.3.1	Experimental .....	85
7.3.2	Simulative.....	86
7.4	Empirical Correlation for Open Channels .....	86
7.5	Vertical Mixing.....	87
7.5.1	Reynolds Approach .....	87
7.5.2	Mixing Index .....	88
7.5.3	Data Export .....	91

8	Reactor Simulations .....	92
8.1	Inlet Tank.....	92
8.1.1	Mesh Independence .....	92
8.1.2	Symmetry Assumption .....	97
8.1.3	Validation .....	99
8.1.4	Analyses of Geometry and Operating Conditions.....	99
8.1.5	Estimation of Power Requirements .....	109
8.1.6	Conclusion and Discussion.....	112
8.2	Channel .....	113
8.2.1	Mesh Independence .....	113
8.2.2	Validation .....	117
8.2.3	Layer Thickness .....	117
8.2.4	Vertical Mixing.....	118
8.2.5	Conclusion and Discussion.....	122
8.3	Flow Reversal Tank.....	123
8.3.1	Initial Configuration .....	124
8.3.2	Geometric Modifications for Improved Flow Patterns .....	126
8.3.3	Additional Baffles .....	129
8.3.4	Conclusion and Discussion.....	131
9	Conclusion and Outlook.....	132
9.1	Setup and Validation of Simulation Models.....	133
9.2	Guidelines for the Design of Thin-Layer Cascade Photobioreactors.....	136
10	Acknowledgements .....	140
	Appendix .....	141
A	Bibliography .....	142
B	Symbols .....	151
C	Abbreviations .....	154
D	Glossary .....	156
E	OpenFOAM Libraries.....	160
E.1	patchPartToFace Source for the topoSet Utility .....	160
E.2	cloudSphere Source for the sample Utility .....	160
F	Salinity Measurements .....	163
F.1	Carrier Board .....	163
F.2	Sliding Maximum Algorithm.....	164

# 1. Introduction

It is no news that in recent time the global temperature has increased due to the accumulation of greenhouse gasses in the atmosphere. Among these, carbon dioxide (CO<sub>2</sub>) contributes more than 50 % of the global warming potential [Lashof and Ahuja 1990]. One major source for CO<sub>2</sub> emissions is the transportation sector, whose share of the total emissions has risen from 15 to 20% in Germany over the last 15 years [Umweltbundesamt 2017]. This very transportation sector also depends on the availability of crude oil which can be converted to the required fuels (petrol, jet fuel, . . .). However, the available reserves of crude oil are decreasing and thus the production is predicted to decline in the near future [Owen et al. 2010]. These results have led to a reconsideration of fossil fuels during the last decades.

Alternative energy sources include the usage of electricity or biofuels. While both have already been established in the automotive industry, neither electricity nor biofuels are viable in the aerospace industry yet [Saynor et al. 2003]. Here the focus lies on the development of biofuels chemically similar to jet fuel, since their energy density is significantly higher than what can be achieved with electrical storage [Dresselhaus and Thomas 2001].

To state, land based biomass is processed (bio-)chemically to yield biofuels. In first generation biofuels, sugars and oils are extracted from arable crops, directly competing with food production. Second generation fuels can be manufactured from agricultural residues, however require a more intensive processing. One promising alternative is the use of microalgae that convert carbon dioxide and sunlight photosynthetically to energy-rich lipids. Using highly selective catalysed chemical processes, these lipids can in subsequent steps be converted to hydrocarbons that are e.g. identical to those found in jet fuel. This approach is generally referred to as third generation biofuels.

Microalgal cultivation plants have already been established for sectors of health food and speciality chemicals (pigments, food supplement, pharmaceutical, . . .) [Becker 1994; Oncel 2013]. Unfortunately, still a number of bottlenecks exist for the commercial cultivation of microalgae for biofuel production (list non-exhaustive): A high level of technology is required for the successful growth [Becker 1994]; investment and operation costs are still too high to compete with conventional, fossil resources [Lundquist et al. 2010; Sheehan et al. 1998; Wigmosta et al. 2011]; current algae ponds are subject to efficiency constraints that restrict productivities well below it's maximum potential [Grobelaar 2012].

This list indicates that further research is required to make algal fuels economically viable. While experimental investigation is inevitable, it is a cost and time demanding procedure. Here numerical models can help mitigate these expenses: Computational Fluid Dynamics (CFD) provides a powerful tool to simulate the flow conditions inside reactor systems. Operational parameters can be analysed for the effect on characteristics of algal cultivation processes like mixing or light availability. Geometric modifications can be made that improve the fluid flow or reduce dead volume inside the reactor. Using the insights generated this way, more focused experimental studies can be performed, accelerating the design process and reducing the need for a thorough, empirical exploration of reactor configurations.



## 2. Objectives

The development of photobioreactors, suitable for the cultivation of microalgae, is a topic of industrial and academic research since the beginning of last century [Becker 1994; Preisig and Andersen 2005]. These interests resulted in commercial production plants for food or speciality chemicals (e.g. fatty acids or carotenoids) [Becker 1994; Spolaore et al. 2006]. The application of microalgal biofuel production was amongst others already investigated during World War II and in the “Algae Species Programme” [Benemann and Oswald 1996; Borowitzka and Moheimani 2013a; Sheehan et al. 1998]. Yet, despite these attempts, no feasible process could be developed in the area of biofuel production [Sheehan et al. 1998].

The primary energy source for photoautotrophic microalgal growth, i.e. without a carbon source except CO<sub>2</sub>, is light [Brennan and Owende 2010]. Thus, the efficient utilisation of light is essential for high productivities [Borowitzka 2013]. The effect of light on the algal suspension can be distinguished by three categories [Pfaffinger 2017b]: On the one side, if insufficient light is supplied in the microalgal suspension, the process is bounded by photolimitation resulting in low growth rates. On the other side, photoinhibition due to excessive irradiation inhibits or even inactivates microalgal cells, reducing the biomass growth and yield [Ragni et al. 2008]. Fitting a kinetic model to data obtained for microalgal growth, between both states an optimal region of photosaturation can be identified. Here, sufficient light is supplied for each cell to achieve a maximal growth rate [Pfaffinger et al. 2016]. However, this is only true under the assumption that thorough mixing in the direction of the light source is provided in the suspension.

Up to now, due to low mixing [Terry and Raymond 1985] and high layer thicknesses (usually 10–30 cm) [Pulz 2001], in current open photobioreactor systems only low productivities and biomass concentrations can be achieved. To generate a product with a low value like biofuel, the production costs need to be monitored. Due to the very simple design of common open photobioreactors, low investment and operational costs occur. Nevertheless, the low biomass concentrations require the processing of huge amounts of algal suspension during the harvesting process at poor yields of dry mass. In contrast to that, Benemann and Oswald [1996] postulate the need to reduce the harvesting cost in algal mass culturing systems as a necessary goal. Currently productivities are in the region of 7.5–15 g m<sup>-2</sup> d<sup>-1</sup> at concentrations of less than 1.5 g l<sup>-1</sup> [Grobbelaar 2009b; Pulz 2001].

To overcome these limitations, Šetlík et al. [1970] developed the Thin-Layer Cascade (TLC) photobioreactor. It is constructed from an inlet tank, a cultivation area in form of a channel with very low layer thicknesses, and a retention tank through which the microalgal suspension circulates sequentially. In the TLC reactor type, concentrations up to 50 g l<sup>-1</sup> could be measured [Apel, Pfaffinger et al. 2017; Zittelli, Biondi et al. 2013]. This is accomplished by very low layer thicknesses in the order of 6 mm and increased mixing of the algal suspension [Doucha and Lívanský 1995]. To this end, the TLC reactor itself consists of several distinct units through which the suspension flows with significantly higher volume flows than in other photobioreactor systems. The reduction of harvesting costs can thereby be achieved due

to the high biomass concentrations. Unfortunately, this advantage comes with increased expenditure for investment and operation. It is thus necessary to ensure a cost and energy efficient reactor design to obtain an economically feasible process.

With their investigation of the TLC reactor, Doucha and Lívanský [1995] as well as Masojidek, Kopecky et al. [2011] provided a good starting point for further development. Nevertheless, a nearly infinite amount of possible operational modes, and even more geometric modifications render an optimisation experimentally challenging, if not impossible. Apart from that, due to the complex geometry, no heuristics or analytical solutions are available to guide the design process of this reactor type. A solution is the application of Computational Fluid Dynamics (CFD) software which can be used to analyse the different units for their performance. Therefore the distinct reactor units were simulated independently, and characterised according to their purposes:

The main cultivation unit, the channel, gives the reactor its name. It consists of a flat, sloped channel over which the algal suspension flows with a velocity of 0.4–0.5 m s<sup>-1</sup> [Masojidek, Kopecky et al. 2011]. It can mitigate the problem of unfavourable, not photosaturated light conditions inside currently used photobioreactor systems. This is done by adjusting the layer thickness in the reactor for given operational conditions (e.g. incident irradiation, biomass concentration). Therefore operational parameters like the volume flow or the channel's slope can be changed to achieve growth under photosaturated conditions, if a perfectly mixed layer is assumed. However, the flow conditions inside the layer cannot be characterised and sufficient mixing can thus not be guaranteed. The Reynolds number for the open channel is in the order of  $Re = 3000$  (calculated for an open channel geometry with a suspension height of 6 mm and a velocity of 0.5 m s<sup>-1</sup>), while the transition from laminar to turbulent flows occurs in the Reynolds number range of  $Re_{trans} = 500–12\,500$  [Masojidek, Kopecky et al. 2011]. This means no conclusion regarding the flow characteristics can be drawn this way. The mixing can further not be determined analytically and can hardly be measured. Thus a method is required which characterises the vertical mixing (in direction of the light source) inside the layer and allows predictions on the light conditions experienced by the cells. Here CFD simulations allow to gain insights. Exploiting the simulated velocity profile of the suspension, virtual particles can be tracked in the flow. From these the mixing properties of the reactor unit can be inferred that are otherwise unavailable to measurements.

Upstream of the channel, the reactor features an inlet tank. The algal suspension is pumped into this tank and exits it by an overflow weir onto the channel. To support the spread of the suspension in the tank, a baffle is installed directly above the inlet. In order to obtain a uniform layer thickness on the channel, it is necessary that an even distribution of the suspension is already achieved on the inlet tank's weir. CFD analysis allows to vary the geometric configuration of the inlet tank as well as the volume flow to find an operation which optimises the resulting fluid layer height over the width of the weir. Apart from that, the inlet tank is effectively the unit where energy is added to the system. This can be a significant part of the operational costs. As the unit is directly connected to the pump, it immediately influences the pressure loss due to hydrostatics (weir height), kinematics (fluid velocity), and friction (baffle, piping between tank and pump). To obtain an estimate of these effects, numerical estimations

allow the calculation of the pressure gradient inside the unit. Comparing the magnitude of the different contributions to the energy input (proportional to the pressure loss), it is possible to identify the aspects with the highest potential for improvement.

In reactors of smaller scale, in which it can not be replaced by multiple inlet tanks, an additional flow reversal tank is applied. This tank gathers the fluid from the channel and redirects it onto another channel, which leads back into a retention tank from which the pump circulates the suspension into the inlet tank. Although this unit can be replaced by another inlet tank for large reactor systems, it is required in technical preparation. Since the channel following the tank is (again) a cultivation region, the flow reversal tank has to provide an even fluid layer similarly to the inlet tank. This should be achieved by simulatively comparing the effects of modifications on the geometry as well as the use of different baffle configurations. Extracting the height of the water layer on the following channel from the simulation, the configurations investigated can be assessed.

In order to perform the CFD simulations for the reactor units, models of the different units are required. To this end, an approximation of the physical geometry is required that is discretised sufficiently fine to allow the investigation of all the effects above. Furthermore, to allow predictions of the units' behaviour, a validation of the virtual geometries is required. Thus, in this thesis a procedure to create these models should be established that allows the robust simulation of varying geometric parameters and operational modes. To validate the results, a reactor in the technical scale with a surface area of  $8 \text{ m}^2$  exists at the TUM AlgaeTec Center (Ottobrunn). This reactor should be exploited to obtain measurements that can be compared to the simulation results, thereby validating the simulation model and quantifying the expected deviation for further simulations.

## 3. Cultivation of Microalgae

The first deliberate upkeep of microalgae was reported in 1850: “[...] Ferdinand Cohn [...] succeeded in keeping the unicellular flagellate *Haematococcus* in his laboratory [...] and called this procedure ‘cultivation’.” [Preisig and Andersen 2005]. Although this cultivation was very rudimentary, many improved cultivation procedures were developed during the next century. As a direct result of World War II, in the 1940s microalgal mass culturing was booming the first time in history [Preisig and Andersen 2005], and the large-scale cultivation of the biomass was “probably first considered seriously in Germany” [Becker 1994]. The next significant step in research started in the 1970s, when the U.S. Department of Energy’s Office of Fuels Development funded the development of algal-based, renewable fuel [Sheehan et al. 1998].

Current commercial plants mainly focus on high-value products such as health food and pigments [Becker 1994; Spolaore et al. 2006]. In research, however, the ambition of finding a cost-effective cultivation method for low-value products like biofuels continues. This led to the development of new technologies and solutions, expanding the list of possible applications. The next sections will thus introduce microalgae and their potential, followed by an overview over cultivation systems that are common at present. Finally, a compact overview over the usage of Computational Fluid Dynamics (CFD) for the analysis of photobioreactors will be given.

### 3.1. Microalgae

In general terms, algae are photosynthetic active organisms. They do, however, differentiate from plants in growing neither roots, stem, nor leaves [Gualtieri and Barsanti 2006]. This allows microalgae to have a five times higher areal productivity than land-based biomass [Posten and Walter 2012]. While these microalgae usually grow unicellular with a characteristic length of 3–10  $\mu\text{m}$ , their counterpart, the macroalgae, can grow up to 60 m in length [Sastre and Posten 2010].

The focus on microalgae is due to a number of qualities favourable for biotechnological production:

- Microalgae can contain up to 60 % of their mass in form of energy-rich lipids [Rodolfi et al. 2009].
- Albeit many algae grow heterotrophic [Alkhamis and Qin 2013; Gualtieri and Barsanti 2006], most algae are able to grow photoautotrophic [Gualtieri and Barsanti 2006]. Hence they primarily require light,  $\text{CO}_2$ , and only low amounts of other nutrients (nitrogen, phosphor) [Posten and Walter 2012].
- With a high tolerance towards temperature, pH, and salt concentration microalgae can be cultivated under extreme conditions [Gualtieri and Barsanti 2006]. This makes cultivation less sensitive to contamination and reduces the requirements of the process (e.g. controlling, fresh water) [Amaro et al. 2011].

Among the effects influencing microalgal growth the most important is the light that is available to the cells [Molina Grima et al. 1999]. For maximal growth, sufficient light must be provided to every cell in the suspension. However, due to the light attenuation in the suspension by the microalgae, for unmixed systems, this cannot be guaranteed for regions farther away from the surface. Thus, depending on the layer thickness and the incident irradiation, up to three regions can be found in an unmixed system: If too much light is provided at the surface, the high number of incident photons can damage the microalgal photosynthesis process temporarily or permanently [Masojidek, Torzillo et al. 2013; Ragni et al. 2008; Torzillo et al. 2010]. This effect is called photoinhibition or photoinactivation respectively. After a certain suspension depth, depending among others on cell density, the “excess” light has been absorbed by the cells to prevent the destructive effects. In this region of photosaturation sufficient photons are available to allow for a near maximal growth of the cells, provided the supply with CO<sub>2</sub> and other nutrients can be guaranteed. Further into the suspension, hardly any photosynthesis can take place, since here the shading effect of the upper layers has absorbed most of the incoming irradiance. Due to this effect of photolimitation little to no algal growth can be observed [Pfaffinger et al. 2016].

The development of these different layers can, however, be avoided by thorough mixing of the suspension [Torzillo et al. 2010]. If the algae pass through the regions sufficiently fast, an average or integral irradiance can be assumed, that is obtained by each cell in the suspension (for details see Pfaffinger et al. [2016]). Furthermore, for frequencies higher than 10 Hz [Terry 1986; Torzillo et al. 2010], improved photosynthesis rates have been reported. This phenomenon is called “flashing light effect”, as it requires a continuous fluctuation of the light environment an algal cell is exposed to.

## 3.2. Photobioreactor Systems

In literature, the precise meaning of the term “photobioreactor” is regularly debated [Grobbeelaar 2009a]. For the purpose of this thesis the term refers to “any container — be it open or closed — in which a bioreaction using photosynthesis takes places” [Apel and Weuster-Botz 2015]. With this definition, photobioreactors can primarily be distinguished between open and closed systems. A rough comparison of both system types is shown in Table 3.1.

As a general rule, closed photobioreactors allow for a significantly higher degree of process control. This is often accompanied by better mixing and surface-to-volume ratios ( $S/V$ ). It is hence no surprise that increased productivities and biomass concentrations can be achieved here. However, this also means increased investment and operation costs. It is thus common ground that the production of low value products by mass cultivation of microalgae is only feasible in open systems [Apel and Weuster-Botz 2015]. A comparison of typical photobioreactor parameters for open and closed systems can be seen in Table 3.2.

In the next sections the most common photobioreactors — closed and open — will be introduced.

**Table 3.1** Direct comparison of selected properties of open photobioreactors and closed systems [Apel and Weuster-Botz 2015; Becker 1994; Pulz 2001].

Parameter	Open Systems	Closed Systems
Contamination	Only the chance of contamination can be reduced by extreme conditions. “Macrocontamination” by animals likely <sup>1</sup> .	Low risk of unwanted organisms, however in large systems this is not completely avoidable, either.
Water loss	High evaporation occurs due to large, open surface.	Controlled loss of water, e.g. by sampling and gassing.
CO <sub>2</sub> loss	Loss of CO <sub>2</sub> to atmosphere when gassing is applied. Atmospheric CO <sub>2</sub> is rarely sufficient.	Loss depends on the reactor design and can thus be minimised.
Illumination	Here always natural sunlight is used, thus the illumination strongly depends on the local climate.	Often artificial light sources are applied, but there are few systems which use natural sunlight, too.
Economics	Low-value products can be produced only this way for an economic balance.	In closed systems high-value products are the primary target, due to the added investment and operation costs.
Reproducibility	The cultivation process strongly depends on weather and environmental conditions and can hardly be reproduced.	Using defined media and operation modes the process can often be reproduced very closely.
Mixing	Mixing is often only applied to distribute nutrients and keep microalgae in suspension.	Distinct mixing is achieved, mostly by the airlift principle. In smaller scales, systems can be approximated as ideally mixed.

<sup>1</sup> “it is not unusual for frogs to be found swimming happily in the pond. . . until they reach the paddle wheel. . .” [Becker 1994].

**Table 3.2** Typical photobioreactor operation parameters distinguished by open and closed systems. Data summarised from Pulz [2001].

Parameter	Unit	Open	Closed
Productivity	g l <sup>-1</sup> d <sup>-1</sup>	0.05–1.2	0.8–1.3
Biomass concentration	g l <sup>-1</sup>	0.3–6.5	5–8
Flow rate	m s <sup>-1</sup>	0.3–0.6	0.5–1.2
Layer Thickness	cm	4–55	50–120
S/V ratio	m <sup>2</sup> m <sup>-3</sup>	6–85	≈ 85

### **3.2.1. Closed Systems**

The most common closed photobioreactor systems are based on the airlift principle. This includes the bubble column and the flat plate photobioreactor [Koller 2017]. However, also tubular systems have been developed and are used in several facilities.

These closed systems are most appropriate for microalgae strains which grow in environmental conditions that are vulnerable to contamination, or when the products are susceptible to biodegradation [Carvalho et al. 2006]. For their design, light penetration in the suspension can be considered as primary parameter [Carvalho et al. 2006].

Apart from the above mentioned systems, many more cultivation techniques can be considered. For extremely simple processes, e.g. inoculum preparation, (polyethylene) bags, hung or standing, can be used. These bags are gassed from the bottom to supply CO<sub>2</sub> and provide a minimal mixing. A further alternative is the use of the common stirred-tank reactors. These offer the advantage of having been characterised well in “normal” biotechnological processes. Additional, internal lighting can be supplied to increase their productivity [Becker 1994; Carvalho et al. 2006]. Nonetheless, these play a minor role, and the next paragraphs will focus on the bubble column, flat plate, and tubular photobioreactors. It should be noted that the information — if not marked otherwise — has been summarised from the reviews by Carvalho et al. [2006] and Zittelli, Rodolfi et al. [2013]. These also include more details for the interested reader.

#### **Bubble Column**

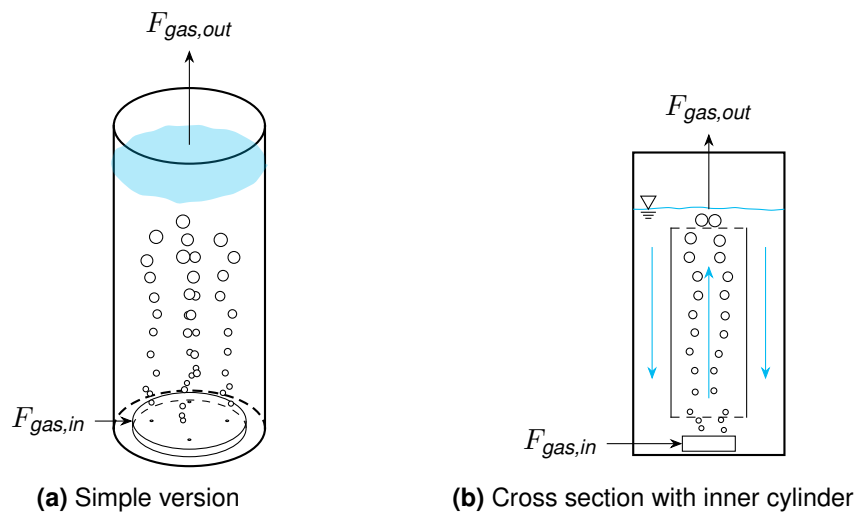
Bubble columns are cheap photobioreactors. They provide mixing by a gas sparger at the bottom, which induces sufficient mixing for CO<sub>2</sub> supply and nutrient distribution. An example for this system can be seen in Figure 3.1. Some alternatives are also equipped with a second, inner cylinder, in which the gassing takes place, thus inducing a circular flow of the suspension through the inner and outer volume.

#### **Flat Plate Photobioreactors**

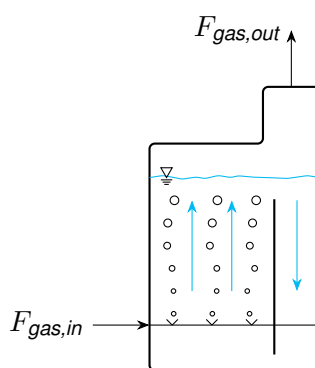
Flat plate photobioreactors have the advantage of offering a uniform light distribution everywhere in the culture medium. This, and the fact that they are designed and placed to make efficient use of supplied (sun)light, results in high productivities for these types of reactor. Mixing can be achieved by a pump, yet airlift systems as shown in Figure 3.2 are also possible. Here a baffle inside the reactor, combined with one-sided gassing induces a circular flow of the suspension distributing nutrients and CO<sub>2</sub>. Different arrangements of gas sparger and baffles are reported in literature (e.g. Huang, Kang et al. [2015]) and can be used to adjust the process to the specific application.

#### **Tubular Reactors**

Tubular reactors mainly exist in vertical and horizontal versions (Figure 3.3). While horizontal units achieve a very efficient usage of the available light, they suffer from increased temperatures if no control system is installed, due to being extremely exposed to the light [Carvalho



**Figure 3.1** Schematic drawing of a bubble column reactor operated as batch process: The algal suspension is mixed by a gas flow  $F_{gas}$ , which is added at the bottom of the reactor (a) via a gas sparger and leaves over the surface. A second, inner cylinder can be added to induce a directed, circulating flow, since gas is only added in the central compartment (b). The light source (not shown) can be placed above the columns (for small volumes) or circular around the column.



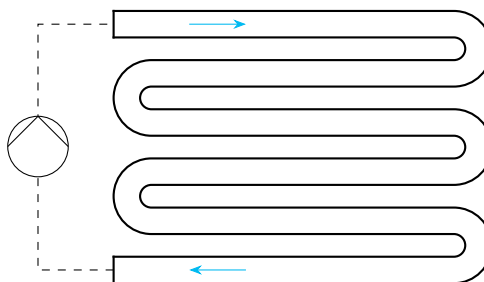
**Figure 3.2** Schematic drawing of a flat plate photobioreactor operated as batch process: The flow of the algal suspension is ensured by the airlift principle induced by the gas flow  $F_{gas}$ . The added gas can simultaneously be used to supply the suspension with  $CO_2$ . Light is usually supplied by a (LED) panel parallel to the cultivation chamber, e.g. behind.



et al. 2006]. A further problem in long tubular reactors of any design is the accumulation of toxic concentrations of  $O_2$  [Molina et al. 2001]. This also means that a scale-up is not easily possible, since these inhibitory effects, combined with possibly limiting concentrations of nutrients decrease the volumetric productivity.

Apart from the normal designs, helical tubular reactors have been developed. These allow a better usage of light (artificially lit from two sides or oriented towards the sun). However, this concept also increases the system's complexity and thus investment and development costs.

Circulation in tubular reactors is often induced by centrifugal pumps [Becker 1994].



**Figure 3.3** An example for a tubular, closed photobioreactor operated as batch process. The (transparent) pipes can be arranged vertically, horizontally or circular in various ways. A pump is used to circulate the suspension. The light location of the light source depends on the orientation of the reactor system (horizontal, vertically) and can be arbitrary for artificially lit systems.

### 3.2.2. Open Systems

Among the open photobioreactors different types can be distinguished, too. Following a classification by Terry and Raymond [1985], these are:

*Ponds* Ponds are the simplest algal mass cultivation system. They are characterised as unmixed, artificial or natural pools of water in which algae are grown. Issues occurring in this cultivation method are amongst others settling of the algal cells and low yields. As a direct result of this low productivity and an unstable performance, their application is usually limited to small-scale applications in rural areas.

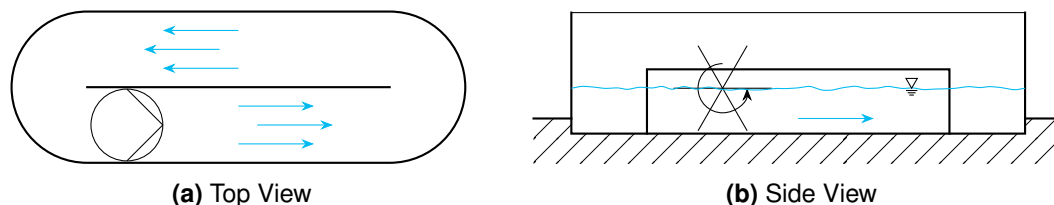
*Channelled systems* These systems are built as relatively deep (in the order of 20–30 cm), flowing channels, in which the suspension slowly circulates. Turbulence in the flow is induced with the goal of suppressing settling and mixing nutrients, yet not to increase light usage. Due to moderate costs combined with a decent productivity most commercial plants fall within this category.

*Shallow circulating systems* To achieve very high microalgae concentrations, shallow systems with fast circulating, highly mixed suspensions can be employed. This requires significantly increased investment costs and power input in the system (pumping). Still, due to the substantially higher biomass concentrations, it greatly favours the expenditure needed to harvest the algae cells. Influence of growth due to the flashing light effect is discussed.

The following paragraphs will introduce the Raceway Pond (RWP) as a deep channelled system, followed by a sloped channel, which can either be operated as deep channel or as a

shallow, highly mixed reactor. Finally, as a typical shallow circulating system, the Thin-Layer Cascade (TLC) will be described.

### Raceway Pond

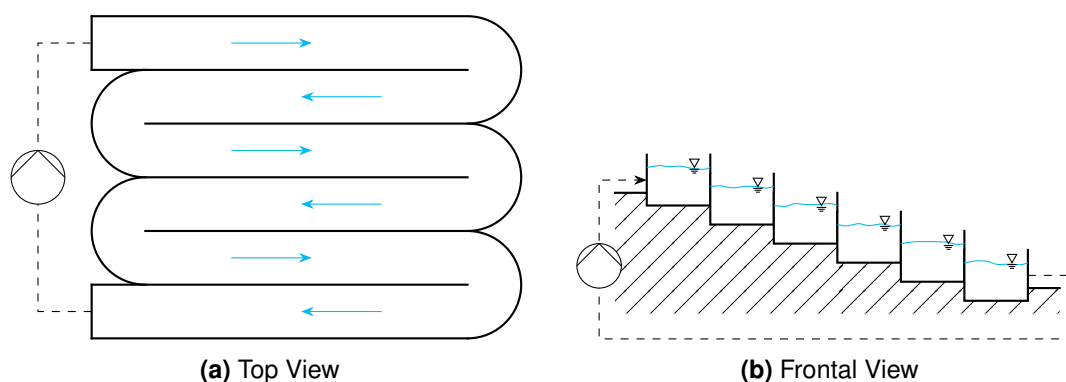


**Figure 3.4** Sketch of a typical raceway pond operated as batch process: The algal suspension is moved along an oval channel **(a)** by a pump, or — most common — a paddle wheel **(b)**. Light is supplied over the surface, usually by natural, solar irradiation

The Raceway Pond is the most widely used commercial system for algae cultivation [Borowitzka and Moheimani 2013b]. A sketch of a typical RWP can be seen in Figure 3.4: The algal suspension circulates in an oblong pond, in the majority of all designs driven by a paddle wheel (Figure 3.4b). Typically the channel depth is between 10 and 30 cm [James and Boriah 2010] with a flow velocity of 20–30 cm s<sup>-1</sup> [Borowitzka and Moheimani 2013b].

The typical areal productivity of RWP is in the order of 10 g m<sup>-2</sup> d<sup>-1</sup> and achieved biomass concentrations range from 0.3 to 0.5 g l<sup>-1</sup> [Pulz 2001].

### Sloped Channel



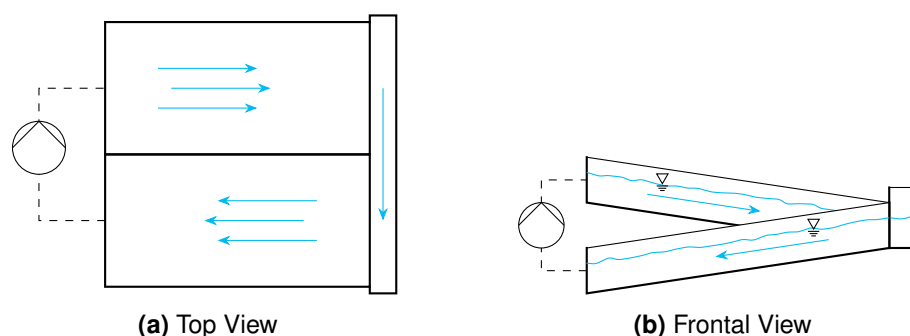
**Figure 3.5** Sketch of a sloped channel photobioreactor operated as batch process: The algal suspension flows along a long, sloped, meandering channel. A pump is used to circulate the suspension. Light is usually supplied naturally by solar irradiation over the fluid surface. Image based on Apel and Weuster-Botz [2015].

Sloped channels, as shown in Figure 3.5, can be regarded as the continued version of a RWP. Depending on the region, these reactors can be widely adapted to the local terrain [Heussler 1985] and allow slopes between 1 and 10 % [Apel and Weuster-Botz 2015]. Although a pump is required to circulate the suspension (Figure 3.5a), the main driving force for the fluid flow is gravity. Thus, theoretically, this reactor system can be designed as deep or shallow channel, depending on the slope and volume flow chosen.

A further advantage of this system was suggested by Märkl and Mather [1985], who investigated the aeration of shallow ponds and suggested an operation mode, in which segments of a sloped channel can be operated individually (e.g. for harvesting or cleaning), allowing a (semi-)continuous operation.

Due to an improved mixing and often low layer thickness, sloped channels achieve productivities around  $20 \text{ g l}^{-1} \text{ m}^{-2}$  [Borowitzka and Moheimani 2013b] with biomass concentrations of approximately  $5 \text{ g l}^{-1}$  [Pulz 2001].

### Thin-Layer Cascade



**Figure 3.6** Sketch of a Thin-Layer Cascade photobioreactor operated as batch process: The algal suspension flows through an upper channel, is collected in a flow reversal tank and redirected into a second channel, from which it is pumped up again. Light is usually supplied by natural, solar irradiation over the fluid surface.

The Thin-Layer Cascade reactor type was initially introduced by Šetlík et al. [1970], and development of the original system continued by Doucha and Lívanský [1995] and Masojidek, Kopecky et al. [2011]. Its main cultivation area are two very shallow channels (Figure 3.6a) with a layer thickness in the order of 1 cm or below. It is further operated at inclinations between 1 and 3% [Masojidek, Kopecky et al. 2011]. Given these conditions, the flow velocities achieved are commonly  $50\text{--}60 \text{ cm s}^{-1}$  [Pulz 2001].

The drawback of these “extreme” design choices are high operational costs (the suspension needs to be circulated continuously by a pump at comparable high fluid velocities) as well as high investment costs. These are, however, counterbalanced by the performance of this setup: Borowitzka and Moheimani [2013b] reports areal productivities of up to  $25 \text{ g m}^{-2} \text{ d}^{-1}$ , while Masojidek, Kopecky et al. [2011] and Pulz [2001] even present results of  $50\text{--}75 \text{ g m}^{-2} \text{ d}^{-1}$ . Furthermore, biomass concentrations of up to  $50 \text{ g l}^{-1}$  can be achieved [Zittelli, Biondi et al. 2013] as maximum, and Masojidek, Kopecky et al. [2011] use a pilot production system with high biomass densities of  $25\text{--}35 \text{ g l}^{-1}$ .

## 3.3. Computational Fluid Dynamics in Algal Photobioreactors

The use of CFD tools for the area of photobioreactors is a relative recent development. Still, many different aspects have been investigated exploiting numerical simulations. These include:

mixing, light, dead zones, growth, shear rate, and mass transfer. A summary of methods used to analyse these aspects in current literature will be described in the following paragraphs.

One of the main influencing parameters in algal growth is the availability of light. A simple approach to incorporate radiation in photobioreactors is the use of 1D models for light absorption and scattering, while resolving the flow behaviour in the reactor using CFD. Sato et al. [2010] use the Lambert-Beer law (describing the light attenuation through a material by absorption, see Glossary for details) for that purpose and tracked the algal cells as following the fluid flow with an additional random walk. Similarly, Huang, Kang et al. [2015] use the Two-Flux approximation (also including the effect light scattered at particles [Glossary]) to incorporate absorption and scattering of light and a Lagrangian particle model (balancing the forces on individual particles, see Glossary) to trace the algal cells. A more advanced approach that can be used in CFD simulations is to also solve the Radiative Transfer Equation (RTE) numerically. This equation describes the distribution of radiation inside a spacial (and possibly temporal) discretised geometry (see Glossary). Huang, Liu et al. [2011] were the first to apply this equation to a photobioreactor and showed that it can be used to predict the light attenuation inside a photobioreactor, including the scattering at cells as well as the reflection at walls. To allow more complex reactor geometries, Kong and Vigil [2014] used the same method extended to respect inner boundaries, such as liquid/glass interfaces, where refraction and reflection take place. While in both publications the reactor geometry was reduced to a 2D representation, the possible need for a 3D model was pointed out. This has been investigated by Wheaton and Krishnamoorthy [2012], who used a fully coupled hydrodynamic/light model, which also respects bubbles of different size (1–100  $\mu\text{m}$ ). They investigated different influences such as gas volume flow or bubble size on the light availability inside the reactor.

Above mentioned simulations are all for closed reactors, where the boundaries of the fluid are well known. James and Boriah [2010] thus used a far simpler approach to simulate a RWP, in which light (and temperature) were assumed to only depend on time, not on culture depths. In combination with a very rough simulation, this allowed them to simulate microalgal growth during long term cultivations (approximately 15 d).

The mixing behaviour is usually investigated to improve the performance of the reactor. Liffman et al. [2013] analysed the flow around the bends of a RWP for power consumption, while at the same time removing stagnant zones after the bends by modifying the curve profile of the inner wall, thus improving the overall mixing. A similar approach has been described by Perner et al. [2003], simulating and improving the bends in a tubular photobioreactor by using larger inner radii. Focusing on the mixing, Pruvost, Pottier et al. [2006] simulated a tracer substance in their reactor to describe the lateral dispersion in the reactor. This allowed the direct comparison between different impellers. For RWPs with less power input this mixing is even more interesting: Ali et al. [2015] and Prussi et al. [2014] analysed the different mixing effects of algal cells in these reactors. Ali et al. [2015] primarily analysed the horizontal effects of different aspect ratios (length/width) of the RWP. They further used the residence time of the cells and a mixing length calculation (path length each cell travels) to compare the different results. Contrary to that, Prussi et al. [2014] focused on vertical mixing of sedimenting algae cells, and exploited the simulation to predict the vertical distribution of the microalgae in the

(relatively unmixed) straight sections of the RWP, where it could be shown that in a large scale pond (500 m<sup>2</sup>) insufficient mixing leads to strongly sedimenting cells.

As a further use of CFD simulations the work by Bannari et al. [2011] should be mentioned. They used a two-phase simulation to compute the mass transfer of O<sub>2</sub> between the phases in a bubble column reactor. Furthermore, a modified reactor geometry, in which the impeller was removed and the gas sparger has been redesigned to ensure a consistent circulation of the fluid. An evaluation of shear stress in both reactor geometries allowed to find suitable conditions for the cultivation of shear-sensitive microalgae.

As of now, the only publication discussing a CFD simulation of a TLC-like photobioreactor is shown by Larrucea [2010]. The author analysed the recirculation of fluid in a reactor with a waved ground. However, the simulation has neither been validated nor could any results for a flat TLC (as used in this work) be extracted.

A summary of these literature sources can be found in Table 3.3 for closed systems and Table 3.4 for open photobioreactors. While for the former a review by Bitog et al. [2011] also gives an exhaustive overview over current and past applications of CFD simulations, reactors of the latter, open type have been investigated less often and are usually of the RWP-type. There are currently no simulations of TLC reactor systems available in literature.

**Table 3.3** Non-exhaustive list of recent CFD simulations of closed photobioreactor systems. Further simulations of closed photobioreactors can also be found in the review by Bitog et al. [2011]. Volume given corresponds to the simulated part of the reactor. Number of cells may vary in the given publication between studies.

Author	Reactor	Software	Details
<b>Light</b>			
Huang, Liu et al. [2011]	Tubular 45 l 9000 cells (2D)	n/A	Validated with previously published experimental data. Solve the RTE in a 2D model (axis-symmetric). Reflection at walls and scattering through algal cells is included. No gas phase.
Huang, Kang et al. [2015]	Flat-Plate 15 l 200 000 cells	ANSYS® ICEM®	Verified by comparing biomass growth to predicted reactor performance. Two-Phase model for gas and liquid interaction. Light is assumed to behave according to the Two-Flux approximation. Microalgae are traced using a Lagrangian particle tracking.

Kong and Vigil [2014]	Tubular 780 l 200 000 cells	OpenFOAM®	Validated using measurements of photon flux density. Solve the RTE in a 2D model. Also consider interfaces between glass and fluid as inner boundaries with reflection and refraction. No gas phase.
Sato et al. [2010]	Pipe Reactor 70 l cells n/A	n/A	Validated by comparison with measured and calculated growth rate. Two-Phase model for bubble and fluid. Light is introduced according to the Lambert-Beer law. A photosynthesis model calculates O <sub>2</sub> production. Microalgae cells are assumed as massless and without volume. They are tracked using the fluid flow vectors extended by a random walk, based on turbulence.
Wheaton and Krishnamoorthy [2012]	Bubble Column 7 l 280 000 cells	ANSYS® Fluent®	Validated with previously published experimental data. Solve a 3D, axis-symmetric RTE. Use a fully coupled hydrodynamic/light simulation. Include the absorption due to algal cells. Bubbles (1–100 µm) are simulated and scatter the light.

### Mixing and Dead Zones

Pruvost, Pottier et al. [2006]	Torus Shape 1.3 l 740 000 cells	ANSYS Fluent	Validated using Particle Image Velocimetry (PIV) measurements. Use a “virtual” photobioreactor. Compare two impellers to determine mixing time and lateral dispersion.
Perner et al. [2003]	Tubular 0.22 l 880 000 cells	ANSYS Fluent	Validated using Laser Doppler Anemometry (LDA) measurements. Evaluate the simulation for dead zones where cells attach to the wall (lysis, shading) in a section of the complete reactor. Also estimate the power requirements with a changed geometry and reduced turbulence. The effects of those changes on growth rate (including dark/light frequency) have been considered.

### Shear and Mass Transfer

Bannari et al. [2011]	Bubble Column 35 l 500 000 cells	OpenFOAM	Validated using measurements of a previous publication. Calculate O <sub>2</sub> mass transfer ( $k_L a$ ) and shear stress. Present reactor modifications based on the findings to improve cultivation of shear sensitive microalgae.
-----------------------	--	----------	---

**Table 3.4** Non-exhaustive list of recent CFD simulations of open photobioreactors. Volume given corresponds to the simulated part of the reactor. Number of cells may vary in the given publication between studies.

Author	Reactor	Software	Details
<b>Light</b>			
James and Boriah [2010]	RWP $38 \cdot 10^3$ l 7440	EFDC/CE-QUAL	Validated using published experimental data. Compute microalgal growth based on nutrients, light and temperature as time dependent functions. Usage of a very rough, geometric representation allows the simulation of long term cultivations.
<b>Mixing</b>			
Ali et al. [2015]	RWP 12.5– $50 \cdot 10^3$ l 55000 cells	COMSOL®	Validated by a comparison with theoretical predictions by empirical Manning equation and Chèzy equation. Lagrangian particle tracking for horizontal mixing using mixing length measure. Residence time of simulated particles gives distribution of mixing lengths.
Liffman et al. [2013]	RWP $2.26 \cdot 10^6$ l 300 000 cells	ANSYS® CFX®	Not validated. Alternative bend geometries are simulated and compared for power consumption as well as mixing.
Prussi et al. [2014]	RWP $1/27.5 \cdot 10^3$ l 500 000 cells	OpenFOAM	Validated with experimental data from 20 m <sup>2</sup> pond; simulation then also used for 500 m <sup>2</sup> pond; Lagrangian particle tracking for vertical mixing, including sedimentation; $Re$ only limited useful

**Other**

Larrucea [2010]	TLC <i>unknown</i>	ANSYS Flu- ent	Not validated. Waved channel ground, investigated for re- circulation in the valleys; reactor also de- scribed in Torzillo et al. [2010]
--------------------	-----------------------	-------------------	---

---



## 4. Computational Fluid Dynamics

The flow of fluids in arbitrary geometries is generally described by very complex flow patterns. For a number of well investigated cases — e.g. standard heat transfer problem [VDI e.V. 2013], the stirred-tank reactor in biochemical engineering [Chmiel 2011] — empirical approximations, describing the overall units behaviour, have been developed over the last decades. However, once geometries deviate from the given norm, or detailed information is necessary, more advanced methods are required [Schwarze 2013].

One of these methods is Computational Fluid Dynamics (CFD). This technique allows the numerical, simulation based investigation of fluid flow, heat transfer, (bio-)chemical reactions, and many more [Versteeg and Malalasekera 2007]. Advantages of this method are [Schwarze 2013]:

- Empirical relations (e.g. turbulence as function of Reynolds number) are usually valid only for a limited range of similar problems which have been analysed in detail before. Analytical solutions are furthermore hard to obtain if the problem is not simplified to its very basic properties (e.g. 3D to 1D). CFD simulations allow a numerical estimation in these situations. It thus also allows to justify approximations made during simplification for analytical solutions or models.
- Simulative studies of alternative setups are usually significantly faster and cheaper than their experimental counterpart.
- Some flow properties can not, or only with a lot of effort be analysed experimentally. Flow behaviour in inaccessible units or where sensors would influence the flow substantially are examples for these problems.

However, as a numerical method, CFD also suffers from limitations [Ferziger and Perić 2002]:

- Numerically solving (differential) equations always introduces **numerical errors**, which can be reduced by iterative solutions, but in practice not completely eliminated.
- Oftentimes models are used to describe otherwise too complex behaviour (e.g. turbulence). These introduce **model errors**, which are subject to the model's assumptions and limitations.
- To solve differential equations, a discretisation is necessary. Since differential equations assume infinitesimal small differences, these produce **discretisation errors**.

The next sections will describe the methods and model approaches used in the CFD simulations in more detail.

### 4.1. Governing Equations

The governing equations in CFD simulations are defined by the conservation laws of physics, and thus include the conservation of **mass**, **momentum** and **energy** [Versteeg and Malalasekera 2007]. These will be introduced in the following sections, followed by a more general form of a transport equation.

If not noted otherwise, all information has been taken from Versteeg and Malalasekera [2007] and Schwarze [2013]. For the interested reader, the former authors also include the derivation of the differential equations, which are not mentioned here for the sake of simplicity.

#### 4.1.1. Conservation of Mass

The first equation that has to be respected in CFD simulations is the conservation of mass. From an infinitesimal balance around a control volume containing a fluid with density  $\rho$  moving with velocity  $\mathbf{u}$ , Equation (4.1) can be derived.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (4.1)$$

This equation is also known as the continuity equation, in which  $\nabla \cdot (\dots)$  represents the divergence. For the special case of an incompressible fluid (e.g. water) the density  $\rho$  does not change and the problem simplifies to Equation (4.2).

$$\nabla \cdot \mathbf{u} = 0 \quad (4.2)$$

#### 4.1.2. Conservation of Momentum

The second equation of conservation is the conservation of momentum. This equation balances the forces around the control volume, and thus considers the stress tensor  $\boldsymbol{\tau}$  (based on the fluids viscosity) and the pressure  $p$ . From its derivation follows Equation (4.3).

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \nabla \cdot \boldsymbol{\tau} - \nabla p + S \quad (4.3)$$

The last term ( $S$ ) is a source term for the momentum equation. It can e.g. include effects of gravitation  $\mathbf{g}$  ( $S = \rho \mathbf{g}$ ) or turbulence models.

#### 4.1.3. Conservation of Energy

The third and last equation of conservation regards energy. It can be expressed in terms of enthalpy  $h$  that is transported.

$$\frac{\partial \rho h}{\partial t} + \nabla \cdot (\rho \mathbf{u} h) = -\nabla \cdot \dot{q}'' + \frac{\partial p}{\partial t} + \nabla \cdot (\boldsymbol{\tau} \mathbf{u}) + S \quad (4.4)$$

This equations also includes the effects of heat transfer  $\dot{q}''$  over the boundaries of the control volume and the changes of kinetic energy ( $\nabla \cdot (\boldsymbol{\tau} \mathbf{u})$ ). The pressure term  $\frac{\partial p}{\partial t}$  occurs due to the definition of the enthalpy related to inner energy.

#### 4.1.4. General Scalar Transport Equation

Comparing the conservation equations above, it can easily be seen that they all share common terms. It is thus possible to write one equation for the transport of scalar quantities. This general form of the transport equations can be seen in Equation (4.5). Therein  $\rho$  is the density of the fluid,  $\phi$  any arbitrary fluid property that is transported through the domain,  $\mathbf{u}$  the flow

velocity,  $\Gamma$  the (mass based) binary diffusion coefficient, and  $S_\phi$  adds local source or sink terms for  $\phi$ .

$$\underbrace{\frac{\partial \rho \phi}{\partial t}}_{\text{transient}} + \underbrace{\nabla \cdot (\rho \phi \mathbf{u})}_{\text{convective}} = \underbrace{\nabla \cdot (\Gamma \nabla \phi)}_{\text{diffusive}} + \underbrace{S_\phi}_{\text{source/sink}} \quad (4.5)$$

This equation is often called “convection-diffusion equation” or “(general) scalar transport equation” [Baukal et al. 2001] and shows the different transport processes involved: The **transient** term shows the local rate of change. The **convective** term describes the transport of  $\phi$  due to the fluid flow. Diffusive effects, usually due to a gradient, are represented in the **diffusive** part of the equation. Finally, a **source/sink** term is added, to allow for additional influences (e.g. reactions, absorption, ...). Depending on the problem at hand, the separate terms of the equation can be simplified or extended to reflect different simplifications or model assumptions, but the general form will stay consistent.

Some examples for modifications of the general transport equation:

*Stationary* The steady-state behaviour can be reflected by defining the transient part of the equation to zero  $\frac{\partial \rho \phi}{\partial t} \equiv 0$ .

*Convection dominated* In situations where the convection is significantly higher than the diffusion, it can be reasonable to neglect the diffusion term ( $\Gamma = 0$ ).

*Turbulence* Turbulent effects are often incorporated into the transport equation by an effective viscosity, which behaves similar to a diffusive term and is thus incorporated in this part of the equation (see Section 4.4).

## 4.2. Methods

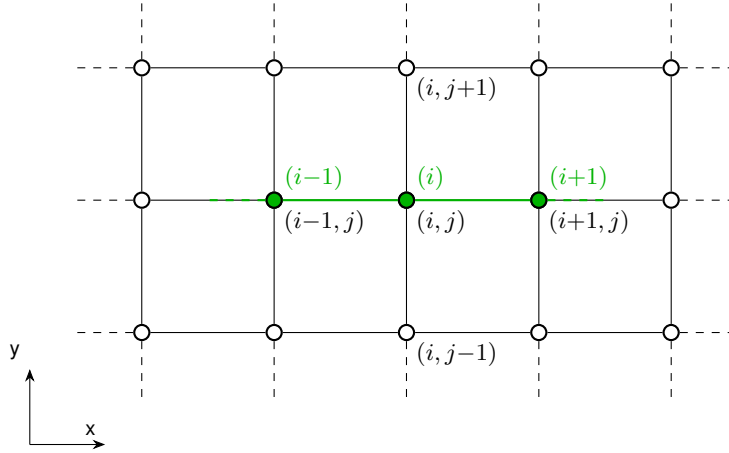
In order to solve the partial differential equations for a given geometric domain different methods exist. The most common methods are the Finite Volume (FV) approach and the use of Finite Differences (FD). Less common methods include Finite Elements (FE) [Bathe 2002; Burnett 1987] and the Lattice Boltzmann Method (LBM) [Succi 2013; Wolf-Gladrow 2004].

The next sections will give a brief overview about the FD method, followed by a slightly more detailed summary of the FV method. FE and LBM use a different approach, which will not be discussed here.

*If not noted otherwise, the information has been taken from Ferziger and Perić [2002].*

### 4.2.1. Finite Differences (FD)

Finite Differences methods are usually based on structured grids, where all computational coordinates (nodes) can be numbered by a set of indices  $(i, j, k)$  in three dimensions. This structure also means that each node’s neighbours can easily be found by addressing them with an decreased/increased index in the corresponding dimension. An exemplary grid of this type can be seen in Figure 4.1 for two- and one-dimensional simulations.



**Figure 4.1** An exemplary Cartesian grid for 1D (●) /2D (○) FD simulations. Since the grid is structured nodes can be labelled consecutively by  $i$  along the  $x$ -axis and  $j$  in  $y$  direction. Image based on Ferziger and Perić [2002].

Each node stores the value of a simulated property  $\phi$  and supplies one algebraic equation that depends on the values stored in the neighbouring nodes. This equation is — matching the methods name — a finite difference approximation of the differential equation for the given property. The finite difference equations are derived from a Taylor Series expansion around each node (Equation (4.6) for a one dimensional example).

$$\begin{aligned} \phi(x) = & \phi(x_i) + (x - x_i) \left( \frac{\partial \phi}{\partial x} \right)_i + \frac{(x - x_i)^2}{2!} \left( \frac{\partial^2 \phi}{\partial x^2} \right)_i \\ & + \frac{(x - x_i)^3}{3!} \left( \frac{\partial^3 \phi}{\partial x^3} \right)_i + \dots + \frac{(x - x_i)^n}{n!} \left( \frac{\partial^n \phi}{\partial x^n} \right)_i + \mathcal{O}(\Delta x^{n+1}) \end{aligned} \quad (4.6)$$

This expression can be rewritten to get an equation for the derivatives of the property  $\phi$  at each node. When all terms on the right hand side of the Equation (4.6) would be used, this derivative would be exact. However, for practical reasons, it is usually truncated after the first term. The power  $n$  of the truncated element is then taken as a measure of the truncation error. In one dimensional problems different approximations can be generated, depending on the nodes used in the series expansion: When the current node  $i$  and the next one  $i + 1$  are considered, the resulting approximation is the so called Forward Differencing Scheme (FDS). Using the previous node  $i - 1$  and the current results in the Backward Differencing Scheme (BDS), and using the previous node and the next one (skipping the current) yields the Central Differencing Scheme (CDS) (Equation (4.7)–(4.9)).

$$\frac{\partial \phi}{\partial x_i} \approx \frac{\phi_{i+1} - \phi_i}{x_{i+1} - x_i} \quad \text{for nodes } i \text{ and } i + 1, \text{ FDS} \quad (4.7)$$

$$\frac{\partial \phi}{\partial x_i} \approx \frac{\phi_i - \phi_{i-1}}{x_i - x_{i-1}} \quad \text{for nodes } i - 1 \text{ and } i, \text{ BDS} \quad (4.8)$$

$$\frac{\partial \phi}{\partial x_i} \approx \frac{\phi_{i+1} - \phi_{i-1}}{x_{i+1} - x_{i-1}} \quad \text{for nodes } i - 1 \text{ and } i + 1, \text{ CDS} \quad (4.9)$$

Analysing these equations for their truncation error, it can be found that the FDS and BDS are first-order approximations ( $\mathcal{O}(\Delta x)$ ), while the CDS is second-order accurate ( $\mathcal{O}(\Delta x^2)$ ). A special case of FDS and BDS is the Upwind Differencing Scheme (UDS), which can be used with directional flow in a way that the current node's value is determined by the properties of the node upwind (against the flow direction) of it. These schemes can also be implemented dynamically, e.g. depending on the transient, local velocity field.

To obtain more precise approximations, multiple options can be found in literature: More terms of the Taylor Series expansion can be included, or — as an alternative approach — arbitrary functions can be fit to the stored values and the known form of the function can be used for interpolation with its exact derivative.

#### 4.2.2. Finite Volume (FV)

Instead of using the differential forms of the conservation equations, FV methods use the integral form, thus integrating all properties over a control volume (CV)  $V$ . Based on the general scalar transport equation (4.5) we obtain Equation (4.10) [Schwarze 2013; Versteeg and Malalasekera 2007].

$$\int_{CV} \frac{\partial (\rho\phi)}{\partial t} dV + \int_{CV} \nabla \cdot (\rho\phi\mathbf{u}) dV = \int_{CV} \nabla \cdot (\Gamma\nabla\phi) dV + \int_{CV} S_\phi dV \quad (4.10)$$

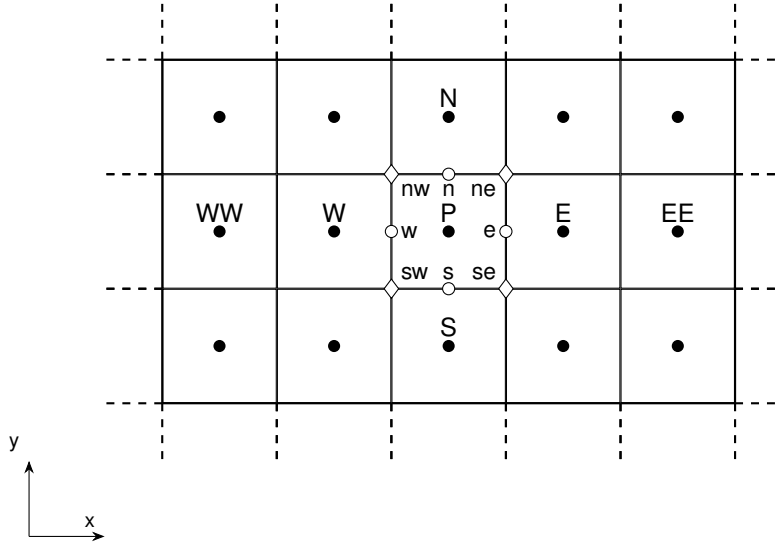
According to Gauss's divergence theorem the convective and diffusive term can be rewritten as surface integrals [Schwarze 2013; Versteeg and Malalasekera 2007]. This theorem states that the volume integral of the divergence of an arbitrary vector  $\mathbf{a}$  is equal to the integral of the face-normal (vector  $\mathbf{n}$ ) component of  $\mathbf{a}$  over the volumes surface (Equation (4.11)).

$$\int_{CV} \nabla \cdot \mathbf{a} dV = \int_A \mathbf{n} \cdot \mathbf{a} dA \quad (4.11)$$

This results in Equation (4.12) as starting point for the FV method.

$$\frac{\partial}{\partial t} \int_{CV} \rho\phi dV + \int_A \mathbf{n} \cdot (\rho\phi\mathbf{u}) dA = \int_A \mathbf{n} \cdot (\Gamma\nabla\phi) dA + \int_{CV} S_\phi dV \quad (4.12)$$

In contrast to Finite Differences, FV stores the value inside the control volume and uses the grid as boundary. This is visualised in Figure 4.2 along with the common notation of cell names in structured grids. Although advantageous for the solving process, FV grids do not have to be structured and can (theoretically) consist of volumes of arbitrary shape. To solve this equation, we need to approximate surface integrals as well as volume integrals.



**Figure 4.2** An exemplary Cartesian grid used in a 2D FV simulation. Marked coordinates: cell centred point  $\bullet$ , face centred point  $\circ$ , mesh point  $\diamond$ . Cells are named according to their position relative to the central, current one (P), as east (E), west (W), north (N), and south (S) with double letters for cells further away. Image based on Ferziger and Perić [2002, p. 73].

## Surface Integrals

Surface integrals can be computed as sum over the separate cell face areas  $A_f$  (Equation (4.13)) for a surface function  $f(\phi)$  of the property  $\phi$ .

$$\int_A f(\phi) \, dA = \sum_{A_f} \int_{A_f} f(\phi) \, dA \quad (4.13)$$

$$\int_{A_P} f(\phi) \, dA_P = \int_{A_e} f(\phi) \, dA_e + \int_{A_s} f(\phi) \, dA_s + \int_{A_w} f(\phi) \, dA_w + \int_{A_n} f(\phi) \, dA_n \quad (4.14)$$

Based on the two dimensional example, this equation would be solved for the eastern side by the following Equation (4.15). However, this requires knowledge of the surface function  $f(\phi)$  of the property  $\phi$ . The most simple approximation is to assume the face centred function value  $\phi_e$  to be constant over the surface, Equation (4.15).

$$\int_{A_e} f(\phi) \, dA = \phi_e A_e \quad (4.15)$$

Still, since FV methods store values inside the cell, the face values need to be interpolated. Similar to Finite Differences different approximations are possible, essentially resulting in equations identical to the Forward Differencing Scheme, Backward Differencing Scheme and Central Differencing Scheme, Equations (4.7)–(4.9), derived in this method.

If higher order approximations are required, more surface points have to be considered, e.g. the north and south corner values,  $\phi_{ne}$  and  $\phi_{se}$ , can be used to obtain a second-order approximation assuming a linear function behaviour. Using even more points the error of the surface calculation can be reduced further, however — since every surface value needs to be

interpolated — significantly increases computation and implementation effort. Furthermore, these interpolation schemes must also be of at least the desired order to retain the accuracy throughout the process.

### Volume Integrals

Volume integrals behave analogous to the surface integrals, with the only difference that the property's value is known inside the volume and assumed to be exact. Then the simplest volume integral reduces to Equation (4.16).

$$\int_{CV} f(\phi) dV = \phi_P V_P \quad (4.16)$$

Again, more points on the volume's faces can be used to obtain a more accurate value of the integral, with the same benefits and problems as described above.

## 4.3. Discretisation and Iteration Algorithms

To solve the Partial Differential Equations (PDEs) numerically they have to be discretised in temporal and spacial dimensions. Some spacial discretisation schemes have already been introduced in Section 4.2. This section now aims at giving a complete overview over the — for this thesis — relevant schemes.

First methods for the temporal discretisation will be described, before the used spacial schemes are explained. Afterwards the algorithms that are used to solve the nonlinear equations iteratively will be introduced.

### 4.3.1. Temporal Discretisation Algorithms

Temporal discretisation is required for all simulations that solve unsteady, transient problems. The general term for the temporal derivative of a property  $\phi$  can be written as a function of itself (Equation (4.17)).

$$\frac{d\phi(t)}{dt} = f(t, \phi(t)) \quad (4.17)$$

The simplest schemes for temporal discretisation are called the Euler schemes [Ferziger and Perić 2002]. Based on these the Crank-Nicolson scheme can be constructed. These will be introduced in the following paragraphs.

#### Euler

To approximate the derivative in Equation (4.17), the direct approach is the calculation of the value of  $\phi$  at the next point in time  $t_{n+1}$  based on its current value and a chosen time step  $\Delta t$  (Equation (4.18)).

$$\phi(t_{n+1}) = \phi(t_n) + f(t_n, \phi(t_n)) \cdot \Delta t \quad (4.18)$$

This method is called explicit or forward Euler. Since all new values can directly be calculated from the former ones, this method is very fast and efficient [Jasak 2006]. However, it requires a stability criterion to be fulfilled which limits the size of the time step  $\Delta t$  [Schwarze 2013]. This criterion was first formulated by Courant et al. [1928] and is thus called Courant number (Equation (4.19)). For the explicit Euler scheme a Courant number  $Co \leq 1$  is required for stability, thus limiting the progress in time.

$$Co = \frac{u\Delta t}{\Delta x} \Rightarrow \Delta t \leq Co \frac{\Delta x}{u} \quad (4.19)$$

A solution to this problem is the use of the implicit or backward Euler scheme, which is defined in Equation (4.20).

$$\phi(t_{n+1}) = \phi(t_n) + f(t_{n+1}, \phi(t_{n+1})) \cdot \Delta t \quad (4.20)$$

In this equation, the desired property value  $\phi(t_{n+1})$  is found on both sides of the equation. Therefore an iterative solution is necessary. It can be found that this method is unconditionally stable for all time steps  $\Delta t$ . However, the number of iterations until convergence thus depends (among others) on the initial guess for the new  $\phi$ . This can require substantially more computational power. Further, implicit, iterative methods are not guaranteed to converge to the correct solution [Jasak 2006], especially if the initial guess is too far from the correct solution. Thus, although theoretically not necessary, limiting the Courant number still is appropriate to ensure convergence to the true result.

Both Euler schemes are first order accurate, i.e. introduce a second order truncation error.

## Crank-Nicolson

The Crank-Nicolson scheme combines both approaches into one equation with a weighting factor of  $\frac{1}{2}$  (Equation (4.21)).

$$\phi(t_{n+1}) = \phi(t_n) + \frac{1}{2} [f(t_n, \phi(t_n)) + f(t_{n+1}, \phi(t_{n+1}))] \cdot \Delta t \quad (4.21)$$

It uses a trapezoidal rule to calculate  $\phi(t_{n+1})$  implicitly and thus is second order accurate in time for arbitrary  $\Delta t$  [Ferziger and Perić 2002; Schwarze 2013]. However, to ensure boundedness of the result the time step has to be limited in a similar manner as above, too [Versteeg and Malalasekera 2007].

### 4.3.2. Spatial Interpolation Algorithms

As described in the above Section 4.2.2 about Finite Volume methods, in the discretisation process of the transport equation values at cell faces have to be computed. To obtain these values, different spatial interpolation algorithms are available. In this section the Central Differencing Scheme (CDS) and the Upwind Differencing Scheme (UDS) will be described as schemes used in many default cases or as base for more advanced schemes.

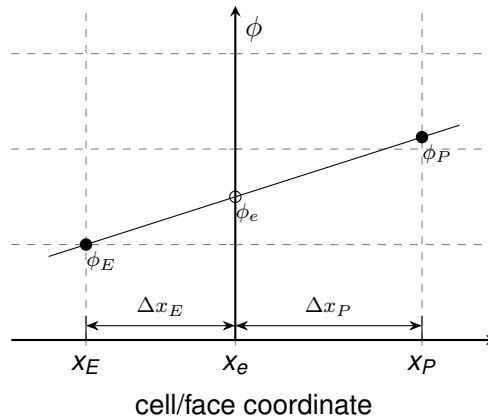


### Linear Interpolation (CDS)

A straightforward approach is the linear interpolation between two cells as shown in Figure 4.3. For the face value  $\phi_e$  between the two cells  $P$  and  $E$  Equation (4.22) reflects the interpolation [Ferziger and Perić 2002].

$$\phi_e = \phi_E \lambda_e + \phi_P (1 - \lambda_e) \quad \text{where} \quad \lambda_e = \frac{\Delta x_P}{\Delta x_E + \Delta x_P} \quad (4.22)$$

This scheme is numerically identical to the CDS scheme introduced in the section on Finite Differences methods. However, although it is a second order scheme and can provide fast convergence, it also produces unphysical oscillations and violates the boundedness of the solution, i.e. it can result in values outside of a physically achievable range [Jasak 1996].



**Figure 4.3** Linear interpolation between two cells. The property  $\phi$  is linearly interpolated between the two non-uniformly distributed cells  $P$  and  $E$  to obtain its face value  $\phi_e$ . Image extended from Jasak [1996].

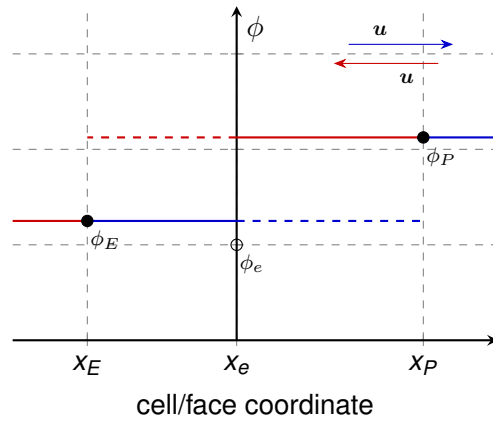
### Upwind Interpolation (UDS)

The upwind interpolation scheme assumes that the property  $\phi$  is primarily transported by the convective flow  $\mathbf{u}$ . It thus “interpolates” the face value between two cells based on the upstream cell value (Figure 4.4).

If the surface normal  $\mathbf{n}$  is known, e.g. for the eastern face  $e$ , this can be expressed by Equation (4.23) [Ferziger and Perić 2002].

$$\phi_e = \begin{cases} \phi_E & \text{if } (\mathbf{u} \cdot \mathbf{n})_e > 0 \\ \phi_P & \text{if } (\mathbf{u} \cdot \mathbf{n})_e < 0 \end{cases} \quad (4.23)$$

This scheme has the advantage that it is unconditionally bounded. Unfortunately, it reduces the accuracy, as it smears the distribution of the transported property similar to an additional diffusion (numerical diffusion) [Versteeg and Malalasekera 2007].



**Figure 4.4** Upwind interpolation between two cells  $P$  and  $E$ . The property  $\phi$  is assumed to have a constant value in the direction of the flow  $u$  (positive flow direction  $\rightarrow$ , negative  $\leftarrow$ ). The dashed lines would be used when strictly extending this scheme *into* the neighbouring cell.

### Further Schemes

The above described schemes are only the simplest ones, and many more can be constructed that will not be described here in detail. A combination of the linear and upwind scheme depending on the local flow conditions can be used as hybrid scheme and combines the advantages of both [Versteeg and Malalasekera 2007]. Higher order schemes can be employed for increased accuracy, and limiter can be introduced to restrict the solution to be bound in a defined range. Detailed information can be found in Ferziger and Perić [2002] and Versteeg and Malalasekera [2007].

### Summary of Schemes

A short summary of interpolation schemes used in this thesis can be found in Table 4.1. For some schemes detailed information can be found in Ferziger and Perić [2002] and Versteeg and Malalasekera [2007], for the remaining the corresponding OpenFOAM source code has to be excavated.

**Table 4.1** A short summary of interpolation schemes and their behaviour in FV simulations [Green-shields 2017a; Holzmann 2017b].

Scheme	Accuracy	Properties
linear	second order	unbounded
upwind	first order	bounded, introduces numerical diffusion
linear/upwind	second order	unbounded, upwind-biases
linear (limited)	first/second order	linear, limited by upwind scheme values
van Leer	second order	bounded, limited

### 4.3.3. Iterative Solution Algorithms

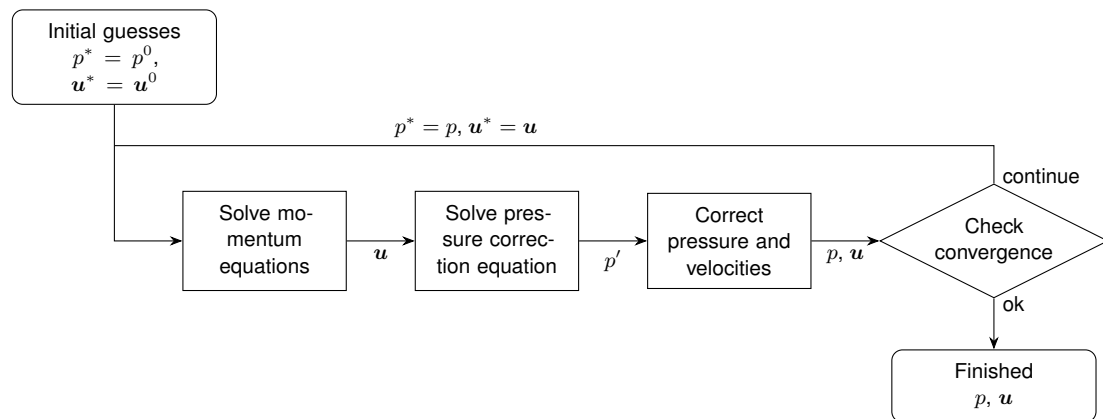
When solving the continuity equation (4.1) and the momentum equation (4.3), a coupling between the pressure and the velocity in both equations can be found. This so-called “pressure-

velocity coupling” has to be considered carefully. Due to memory restrictions the solution of these equations has historically been developed as a segregated algorithm, solving one equation at a time [Clifford and Jasak 2009]. However, due to the interdependence, additional errors are introduced, which have to be reduced by suitable algorithms. Three algorithms of that class will be presented in this section.

Apart from the segregated solvers, the increase in computational power has made so called block-coupled solvers feasible, in which all equations are solved simultaneously. Unfortunately, these algorithms are still under development and were — at the time of this thesis — not yet implemented.

## SIMPLE

The Semi-Implicit Method for Pressure Linked Equations (SIMPLE) algorithm is used to iteratively compute the solution of a steady-state problem. It was first developed by Patankar and Spalding [1972]. The algorithmic steps are shown in Figure 4.5 and will be explained below.



**Figure 4.5** Visualisation of the iterative procedure used in the SIMPLE algorithm. Image based on Versteeg and Malalasekera [2007].

The SIMPLE algorithm starts with an initial guess. This is usually the (temporal) boundary condition, i.e. the initialised field values  $p^0$  and  $u^0$  in the solution domain. The detailed steps are:

1. Obtain guessed values for the pressure and velocity field  $p^*$  and  $u^*$ .
2. The momentum equation is solved with these guessed values to compute a new velocity field. This step is also called the momentum predictor.
3. The new velocities have to respect the continuity equation. Based on this, a pressure correction  $p'$  is calculated.
4. The calculated fields are updated by the correction term. Therein under-relaxation terms  $\alpha_p$  and  $\alpha_u$  are employed. Here the pressure is calculated implicitly with the computed pressure correction term, while the velocity fields are updated explicitly based on the

previous iteration ( $\mathbf{u}^{n-1}$ ) [Versteeg and Malalasekera 2007], giving the algorithm its name:

$$p^{new} = p^* + \alpha_p * p'$$

$$\mathbf{u}^{new} = \alpha_u \mathbf{u} + (1 - \alpha_u) \mathbf{u}^{n-1}$$

5. Check for convergence: If a given tolerance is not yet reached, use the newly calculated fields as new guesses ( $p^* = p$ ,  $\mathbf{u}^* = \mathbf{u}$ ) and continue with step 2.

In step 4 of the algorithm, under-relaxation is applied, limiting the correction to the corresponding field by the factor  $\alpha$ . Although this slows the convergence down, it is usually necessary to prevent the algorithm from diverging [Jasak 2006].

## PISO

The Pressure Implicit with Splitting Operator (PISO) algorithm, developed by Issa [1986], is an algorithm to solve transient flows. It works under the assumption that linear effects due to the pressure-velocity coupling are much stronger than nonlinear effects of the  $\mathbf{u}-\mathbf{u}$  coupling in the momentum equation [Jasak 1996]. Based on this, it is possible to only solve the momentum predictor once per time step and use the discretised momentum equation for multiple pressure correction iterations [Jasak 2006].

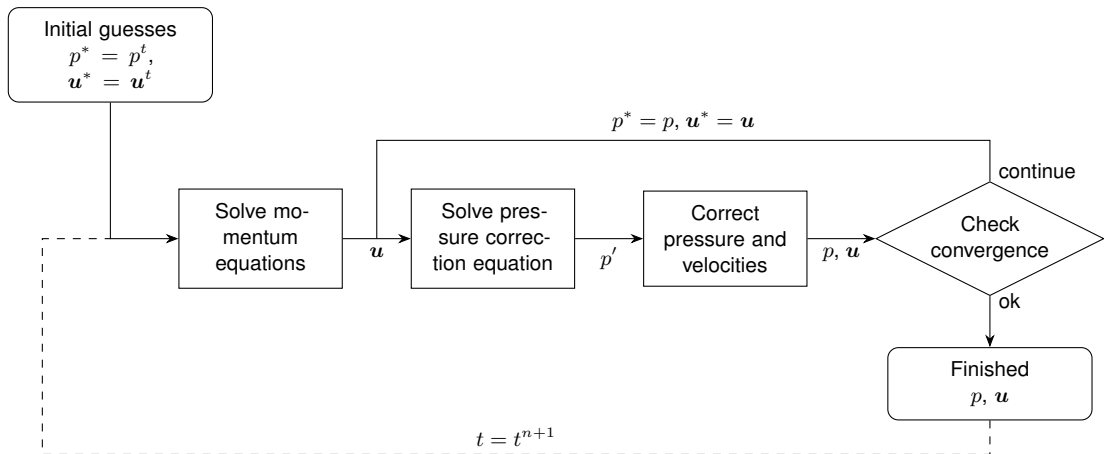
The detailed steps of the PISO algorithm, also visualised in Figure 4.6:

1. Obtain guessed values for the pressure and velocity field  $p^*$  and  $\mathbf{u}^*$ .
2. Discretise and solve the momentum equation to obtain a new velocity field (momentum predictor).
3. Based on the current velocity field, a pressure correction can be computed.
4. The velocity is corrected to reflect the new pressure field.
5. If convergence is not yet reached the algorithm continues with step 3.
6. Continue with next time step.

Although the assumption of having to solve the momentum predictor just once can speed up the solution, it requires small time steps (i.e. small Courant number). This is important, since for larger time steps the velocity field changes too much to justify “freezing” its discretisation [Jasak 1996].

## PIMPLE

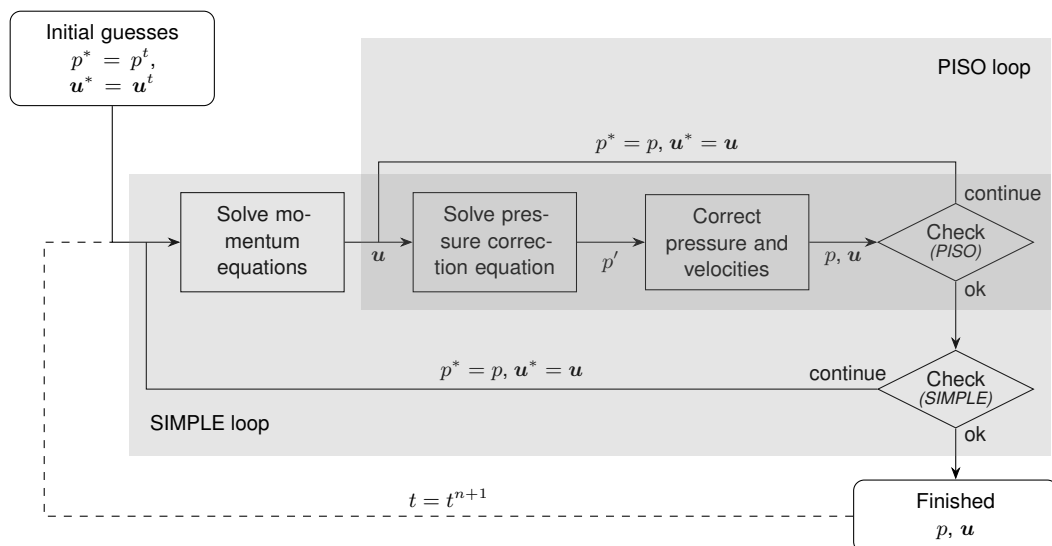
The PIMPLE algorithm has been developed to combine both the SIMPLE and PISO algorithm [Holzmann 2017a]. It thus consists essentially of an inner PISO loop, surrounded by an outer SIMPLE loop (compare Figure 4.7):



**Figure 4.6** Visualisation of the PISO algorithm. Compared to the SIMPLE algorithm, Figure 4.5, the momentum equation is only discretised once per time step.

1. Obtain guessed values for the pressure and velocity field  $p^*$  and  $u^*$ .
2. Discretise and solve the momentum equation as done in the PISO algorithm. Loop around the pressure correction from the continuity equation to update the pressure and velocity field while checking for an inner convergence.
3. Check for outer convergence, and if not satisfied, continue with step 2.

The biggest advantage of this algorithm is that it removes the time step limitation found in the PISO algorithm, thus allowing higher Courant numbers ( $Co \gg 1$ ). This way a significant speed up of the simulation can be achieved [Holzmann 2017a].



**Figure 4.7** Visualisation of the PIMPLE algorithm. It combines the method of the PISO algorithm (Figure 4.6) with an outer loop of the SIMPLE algorithm (Figure 4.5) to allow for larger time steps.

## 4.4. Turbulence

The flow regime of a fluid is generally distinguished in the two flow regimes of laminar and turbulent flow. A typically used measure to classify a flow is the Reynolds number  $Re$  (Equation (4.24)). For a fluid with given density  $\rho$  and viscosity  $\mu$ , flowing with a velocity  $v$  in a domain with characteristic length scale  $L$ , this number gives the ratio of inertial forces to viscous forces.

$$Re = \frac{\rho v L}{\mu} \quad (4.24)$$

For low Reynolds numbers, a fluid behaves in a completely laminar fashion and can be described by the equations stated in Section 4.1 [Versteeg and Malalasekera 2007]. However, in most engineering solutions for reactors turbulence is a desired effect. Generally it increases mixing and homogeneity of the fluid. Be that as it may, in CFD simulations this adds a level of complexity: The random, complex, and dynamic behaviour of turbulence requires — except for very detailed simulations (Direct Numerical Simulation (DNS)) — further approximations. This led to the development of several approaches of modelling turbulence, which can be grouped in three categories [Versteeg and Malalasekera 2007]:

*Reynolds-Averaged Navier-Stokes* In these sets of models only mean flow properties are considered relevant. The Navier-Stokes equations are averaged, thus effectively removing all fluctuations that are typical for turbulent flows from the solution. This approximation allows to estimate the overall flow behaviour for many (standard) cases at a relatively modest cost. It is thus the most commonly used class of models for practical, non-research applications.

*Large Eddy Simulation* In Large Eddy Simulations (LESs) the largest occurring eddies are assumed to be the primary influencing element. Thus, their behaviour is solved dynamically, while smaller eddies are averaged by so called sub-grid scale models.

*Direct Numerical Simulation* This method actually is no modelling approach, but the theoretically rather simple idea of refining the grid in the solution domain up to a point where all turbulence effects can be resolved by the Navier-Stokes equations. This method is very costly and is thus mostly used in research practice for very small domains.

In the following sections the methods relevant for this thesis will be described more detailed. First, the Reynolds-Averaged Navier-Stokes (RANS) method will be described. Afterwards, a special case of the RANS models, the Low Reynolds turbulence models will be introduced.

### 4.4.1. Reynolds-Averaged Navier-Stokes

As mentioned above, RANS models only describe the average flow properties (averaged property  $\phi \equiv \bar{\phi}$ ). Applying this averaging to the Navier-Stokes momentum Equation (4.3), an additional term arises, which describes the random, turbulent fluctuations of the fluid [Versteeg and Malalasekera 2007]. This term comes in the representation of a stress tensor, generally referred to as Reynolds stresses  $\underline{\tau}'$ . The modified momentum equation can be seen

in Equation (4.25).

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \nabla \cdot (\boldsymbol{\tau} + \boldsymbol{\tau}') - \nabla p + S \quad (4.25)$$

The corresponding stress tensor is computed using the so called Boussinesq approximation (Equation (4.26)). It describes the Reynolds stresses as being proportional to the rate of deformation of the mean fluid velocity  $\bar{\mathbf{u}}$ .

$$\boldsymbol{\tau}' = \mu_t \left( \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij} \quad (4.26)$$

The proportionally factor therein is the turbulent or eddy viscosity  $\mu_t$ . This causes the term to behave similar to the viscous stress term of the momentum Equation (4.3). The second term of the approximation is a correction factor, which ensures the correct diagonal entries of the Reynolds stress tensor [Versteeg and Malalasekera 2007]. It includes the specific turbulent kinetic energy  $k$  and the Kronecker Delta function  $\delta_{ij}$ . This equation still contains two unknowns, the turbulent viscosity  $\mu_t$  and the turbulent kinetic energy  $k$ . To calculate these, turbulence models can be employed. A number of different RANS turbulence models is available, out of which two will be described in the next paragraphs.

### The $k - \varepsilon$ Turbulence Model

The  $k - \varepsilon$  turbulence model has been developed by Launder and Spalding [1974] and is the most widely validated turbulence model in engineering practice [Versteeg and Malalasekera 2007]. The turbulent viscosity is modelled as a function of the turbulent kinetic energy  $k$  and the rate of viscous dissipation  $\varepsilon$  (Equation (4.27)).

$$\mu_{\text{eff}} = \rho C_\mu \frac{k^2}{\varepsilon} \quad (4.27)$$

To describe the transport of the turbulent properties  $k$  and  $\varepsilon$ , two additional transport equations are added (Equations (4.28) and (4.29)).

$$\underbrace{\frac{\partial \rho k}{\partial t}}_{\text{transient}} + \underbrace{\nabla \cdot (\rho k \bar{\mathbf{u}})}_{\text{convective}} = \underbrace{\nabla \cdot \left[ \frac{\mu_{\text{eff}}}{\sigma_k} \nabla k \right]}_{\text{diffusive}} + \underbrace{2 \mu_{\text{eff}} \bar{s}_{ij} \bar{s}_{ij}}_{\text{production}} - \rho \varepsilon \quad (4.28)$$

$$\underbrace{\frac{\partial \rho \varepsilon}{\partial t}}_{\text{transient}} + \underbrace{\nabla \cdot (\rho \varepsilon \bar{\mathbf{u}})}_{\text{convective}} = \underbrace{\nabla \cdot \left[ \frac{\mu_{\text{eff}}}{\sigma_\varepsilon} \nabla \varepsilon \right]}_{\text{diffusive}} + \underbrace{C_{1\varepsilon} \frac{\varepsilon}{k} \cdot 2 \mu_{\text{eff}} \bar{s}_{ij} \bar{s}_{ij}}_{\text{production}} - \underbrace{C_{2\varepsilon} \rho \frac{\varepsilon^2}{k}}_{\text{destruction}} \quad (4.29)$$

Apart from the stress tensor, for which a relation to the fluid's velocity can be given ( $\bar{s} = \frac{1}{2} (\nabla \bar{\mathbf{u}} + (\nabla \bar{\mathbf{u}})^T)$ ), five empirical parameters are used in these equations. They have been determined by fitting the model to a wide range of turbulent flows and are commonly used [Versteeg and Malalasekera 2007]. For special cases, e.g. increased wall roughness, these parameters can be adapted:

$$C_\mu = 0.09 \quad \sigma_k = 1.00 \quad \sigma_\varepsilon = 1.30 \quad C_{1\varepsilon} = 1.44 \quad C_{2\varepsilon} = 1.92$$

## $k - \omega$ Turbulence Model

The  $k - \omega$  model has been developed by Wilcox [1988, 2008]. Instead of the dissipation rate  $\varepsilon$  it balances the turbulent frequency  $\omega = \frac{\varepsilon}{k}$  [Versteeg and Malalasekera 2007]. The turbulent viscosity is then calculated by Equation (4.30).

$$\mu_{eff} = \rho \frac{k}{\omega} \quad (4.30)$$

Furthermore the transport equations for turbulent flow are given by Equation (4.31) and (4.32):

$$\underbrace{\frac{\partial \rho k}{\partial t}}_{transient} + \underbrace{\nabla \cdot (\rho k \bar{\mathbf{u}})}_{convective} = \underbrace{\nabla \cdot \left[ \left( \mu + \frac{\mu_{eff}}{\sigma_k} \right) \nabla k \right]}_{diffusive} + \underbrace{P_k}_{production} - \underbrace{\beta^* \rho k \omega}_{destruction} \quad (4.31)$$

$$\underbrace{\frac{\partial \rho \omega}{\partial t}}_{transient} + \underbrace{\nabla \cdot (\rho \omega \bar{\mathbf{u}})}_{convective} = \underbrace{\nabla \cdot \left[ \left( \mu + \frac{\mu_{eff}}{\sigma_\omega} \right) \nabla \omega \right]}_{diffusive} + \underbrace{\gamma_1 \cdot P_\omega}_{production} - \underbrace{\beta_1 \rho \omega^2}_{destruction} \quad (4.32)$$

where the production terms are defined as

$$P_k = 2\mu_{eff} \bar{s}_{ij} \bar{s}_{ij} - \frac{2}{3} \rho k \frac{\partial \bar{u}_i}{\partial x_j} \delta_{ij}$$

$$P_\omega = 2\rho \bar{s}_{ij} \bar{s}_{ij} - \frac{2}{3} \rho \omega \frac{\partial \bar{u}_i}{\partial x_j} \delta_{ij}$$

The commonly used parameters in this model are:

$$\sigma_k = 2.0 \quad \sigma_\omega = 2.0 \quad \gamma_1 = 0.553 \quad \beta_1 = 0.075 \quad \beta^* = 0.09$$

The advantage of this model is — contrary to the standard  $k - \varepsilon$  model — its applicability in the near wall region, where it can handle laminar effects much better. However, it could be shown to depend strongly on free stream values of  $\omega$  [Versteeg and Malalasekera 2007]. This means that either the values at these boundaries need to be determined precisely, or the boundaries have to be sufficiently far from the region of interest.

### 4.4.2. Low Reynolds Turbulence Models

Comparing the two models above, it could be demonstrated that the  $k - \varepsilon$  model showed better behaviour regarding its free stream properties, while the  $k - \omega$  model performed better at near wall regions [Versteeg and Malalasekera 2007]. Thus, the hybrid  $k - \omega$  *SST* (shear stress transport) model described in Menter et al. [2003] was developed. This model uses an approach similar to the  $k - \omega$  model near the wall and the standard  $k - \varepsilon$  model in the fully turbulent region.

This makes the model a so called “Low Reynolds turbulence model”. It should be noted, that “Low Reynolds” does — in the context of turbulence model — not refer to the global Reynolds number, but to a local turbulent Reynolds number, which is defined based on the turbulent length scale  $l$  as characteristic length and behaves proportional to the ratio of turbulent and



molecular viscosity [Davidson 2016]. This group of models has the general advantage that they can also be used near the wall, where the molecular viscosity dominates and we can not expect a fully turbulent flow (compare turbulence boundaries, Section 4.5.3).

## 4.5. Boundary Conditions

In CFD simulations Boundary Conditions (BCs) are the main influencing factor on the solution obtained [Versteeg and Malalasekera 2007]. Thus caution should be taken when defining the BCs of a solution domain. This section aims to introduce the primary, most relevant conditions met in CFD applications. Details on their effect on the solution of conservation and momentum equation can be found in Ferziger and Perić [2002] and Versteeg and Malalasekera [2007].

*If not noted otherwise, all information in this section has been taken from Versteeg and Malalasekera [2007].*

### 4.5.1. The Inlet Boundary Condition

Most properties at an inlet boundary should be known. Fluid velocity can be specified directly, while the pressure can be given (if known) or calculated relative to a known reference (which has to be defined elsewhere in the solution domain). For turbulent properties  $k$  and  $\varepsilon$  (or  $\omega$ ), however, usually no data is available. This data then has to be approximated with empirical equations [Flórez-Orrego et al. 2012]:

$$k = \frac{3}{2} (\bar{u} \cdot I)^2 \quad (4.33)$$

$$\varepsilon = C_\mu^{3/4} \frac{k^{3/2}}{l} \quad (4.34)$$

$$\omega = \frac{\sqrt{k}}{l} \quad (4.35)$$

Therein (obviously) some information about the existing turbulence at the inlet is required. In Equation (4.33) the turbulent intensity  $I$  of the flow needs to be found. It is defined as the ratio of turbulent fluctuations of the flow  $u'$  to the mean flow  $u$ . For duct flows, this can be estimated based on the Reynolds number (Equation (4.36)). Usually, a turbulent intensity of 1 % is considered low turbulence, while 10 % show a high degree of turbulence [Flórez-Orrego et al. 2012].

$$I = \frac{u'}{u} \approx 0.16 \cdot Re^{-\frac{1}{8}} \quad (4.36)$$

For the estimation of  $\varepsilon$  and  $\omega$  furthermore the turbulent length scale  $l$  is required. It can be approximated by Equation (4.37), where  $L$  is the characteristic length of the flow.

$$l = 0.07L \quad (4.37)$$

### 4.5.2. The Outlet Boundary Condition

At the outlet usually very little is known about the flow. It should thus be far away from the region of interest to prevent errors propagating through the domain [Ferziger and Perić 2002]. If the outlet BC is sufficiently far away from disturbances, the fluid flow can be assumed to behave fully developed and a zero gradient approximation can be made.

### 4.5.3. Walls

Walls are the most common boundaries found in CFD simulations. For the flow velocity the no-slip condition defines that the fluid velocity  $\mathbf{u}$  equals the velocity of the  $\mathbf{u}_{wall}$  (usually  $\equiv 0$ ). For most other variables, reasonable information on their near-wall behaviour can mostly be found easily or defined (e.g. defined heat flux or adiabatic wall, zero gradient concentration profile, ...). However, for turbulence properties a more complicated approach is necessary.

As stated above in Section 4.4, the flow behaviour near the wall is dominated by viscous forces, and thus no fully developed turbulence is found. Generally, the fluid flow close to the wall is described by multiple layers, which are distinguished by a dimensionless wall distance  $y^+$  (Equation 4.38).

$$y^+ \equiv \frac{\rho u_\tau y}{\mu} \quad (4.38)$$

$$u_\tau \equiv \sqrt{\frac{\tau_w}{\rho}} \quad (4.39)$$

$$\tau_w = \mu_{eff} \left( \frac{\partial \mathbf{u}}{\partial y} \right)_{y=0} \quad (4.40)$$

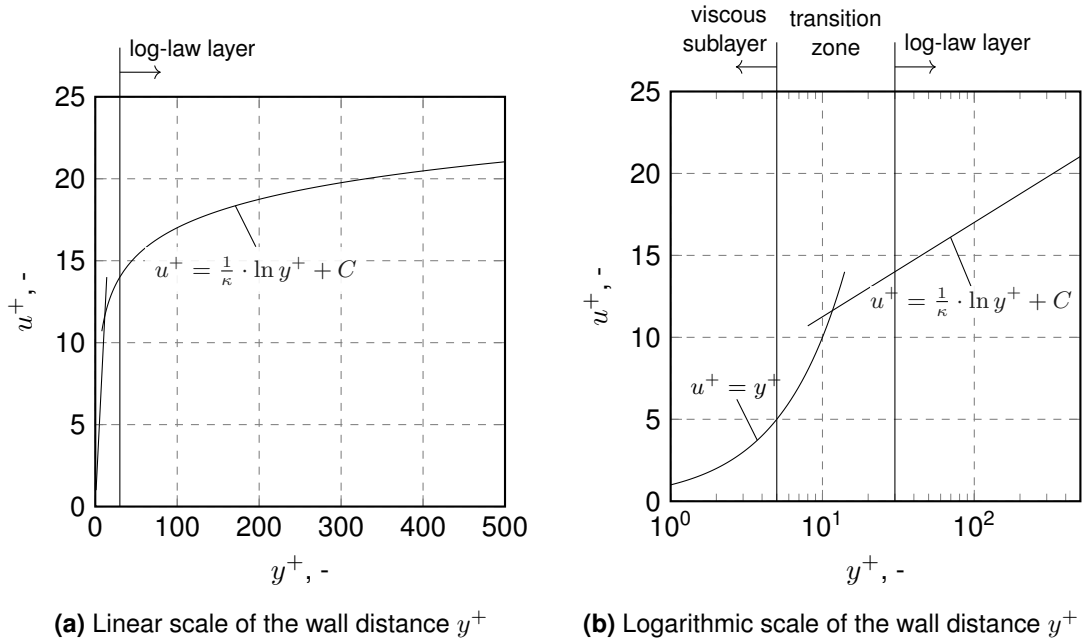
The value of  $y^+$  depends on the friction velocity  $u_\tau$ , the distance to the wall  $y$ , and the fluid's viscosity  $\mu$  and density  $\rho$ . Therein  $u_\tau$  can be calculated from the wall shear stress  $\tau_w$  (Equations (4.39) and (4.40)). Since the position of the first cells location is not known before the calculation, in Equation (4.40) the effective viscosity ( $\mu_{eff} = \mu + \mu_t$ ) is used [Wei 2012b]. Defining furthermore the dimensionless velocity  $u^+$  (Equation (4.41)), the following layered behaviour can be observed (compare also Figure 4.8):

$$u^+ = \frac{v}{u_\tau} \quad (4.41)$$

*Viscous sublayer* The first layer at the wall, in which the viscous effects of the fluid dominate. Here the fluid velocity  $u^+$  is proportional to the wall distance  $y^+$ . This layer is usually in the region of  $0 < y^+ < 5$ .

*Transition zone* As the name indicates, in this zone the flow behaviour transitions from the laminar flow in the first layer into the fully turbulent profile. This layer is in the range of  $5 < y^+ < 30$ . Here viscous and turbulent stresses are equally relevant.

*Log-Layer* The third layer is named after the log-law, which describes the development of the fluid velocity  $u^+$ . It starts at  $y^+ \geq 30$  and evolves the flow into the fully developed turbulent regime.



**Figure 4.8** Visualisation of the different zones of fluid velocity close to the wall: In the viscous sublayer  $u^+$  is proportional to  $y^+$ , while in the log-law layer the dependency follows a logarithmic law. The equation contains the Von Kármán constant  $\kappa$  and a constant  $C$ .

To reflect these layers in the turbulence model, most commonly wall functions are used. These wall functions exploit the log-law to correct the turbulence model. However, since the log-law is not valid close to the wall, it requires the first cell of the grid to be sufficiently far away from the wall ( $30 < y^+ < 300$ ,  $y^+ \approx 30$  desirable [ANSYS Inc. 2009]). An alternative is the usage of Low Reynolds turbulence models (Section 4.4.2). If the mesh is resolved sufficiently fine, the viscous sublayer can be resolved by these models computationally and thus generate a more detailed flow profile. This allows to define the values of  $k$ ,  $\varepsilon$ , or  $\omega$  at the wall directly [henry 2010].

#### 4.5.4. Pressure at Boundaries

In situations where the exact flow properties are not known, a constant pressure can be defined to have the algorithm compute the velocity. This can be used e.g. in surface flows or in flows, where only the pressure difference between inlet and outlet is known. The inverse situation can also occur, where the flow is defined (e.g. at inlets or walls) and the pressure must be calculated accordingly.

Both cases must be handled by specific boundary conditions, which take care of the velocity-pressure coupling in the solution algorithm.

#### 4.5.5. Symmetry

Symmetry BCs can be used, when the simulated domain shows a steady-state symmetric behaviour. For this BC it is assumed that no flow crosses the boundary (normal velocity is zero). It should however be noted that a geometrically symmetric domain does not necessarily result in symmetric flow conditions, and whether it can be used must be evaluated for each case.

## 4.6. Multiphase Systems

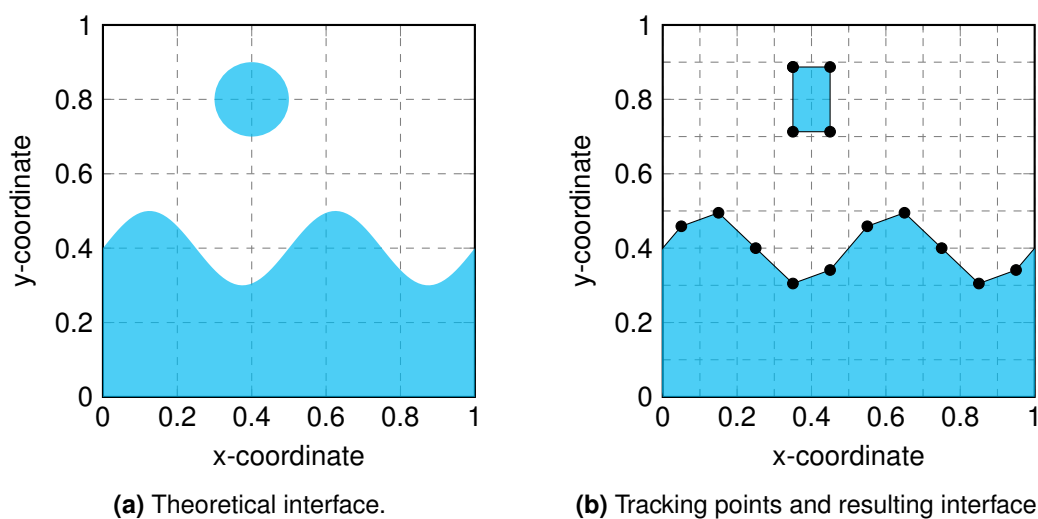
For many simulation cases the simulation of a single, continuous phase does not reflect the reality. Thus multiphase models that are able to describe the interaction of two or more phases have to be employed. In this section, an overview over multiphase models for two-phase system will be given. After this overview, the method applied in this thesis will be explained more detailed. Finally, equations for the transport of a passive scalar quantity will be described.

Different approaches exist to model two separate phases in CFD simulations (all information taken from Rusche [2002] if not marked otherwise):

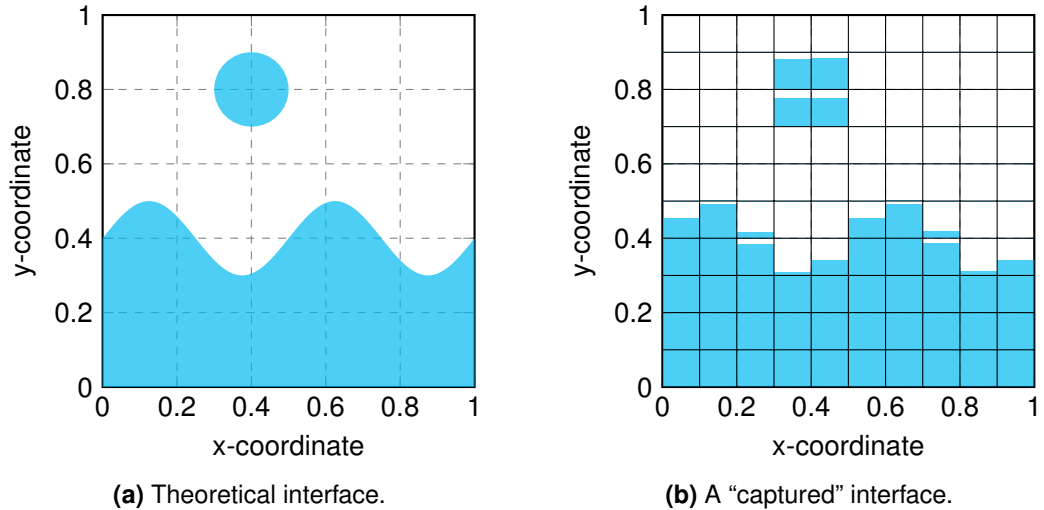
*Surface Tracking* Surface tracking methods use the normal computational grid and add an additional interface marker function (compare Figure 4.9) or a height function. Although additional handling of a marker function is required, this is also an advantage of these methods: The resolution used for the computation can be chosen independently for fluid and interface. However, surface tracking methods do not strictly conserve the volume of each fluid.

*Dynamic Meshes* This technique modifies the mesh to align with the interface between the phases. Although this always gives a sharp interface, it can require a lot of complex re-meshing of the domain and is usually only feasible for simple surface geometries.

*Interface Capturing* Here the phases can be distinguished e.g. by massless particles or a scalar volume fraction. No sharp interface is present, since it is not tracked inside a cell (compare Figure 4.10). This method is the most commonly used, since implementation of a volume fraction only requires one additional transport equation. However, as with regard to the interpolation schemes shown in Section 4.3.2, this has to be handled with care to avoid unbounded solutions. This problem has — among others — been addressed by a group of algorithms termed Volume of Fluid (VoF) methods.



**Figure 4.9** Example for an interface tracking method: The interface is here tracked by markers • in a Cartesian mesh and assumed to behave linear between the tracking points.



**Figure 4.10** Example for an interface capturing method: For each cell a volume fraction of fluid is calculated. The exact course of the surface in the cell is unknown, and in this graph just assumed to align in y-direction. Image adapted from Noh and Woodward [1976].

#### 4.6.1. Volume of Fluid Method (VoF)

Volume of Fluid methods were first introduced by Hirt and Nichols [1981]. Apart from the continuity and momentum equation a further transport equation is solved, Equation (4.42).

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \alpha \mathbf{u} = 0 \quad (4.42)$$

In this equation  $\alpha$  is the (volumetric) phase fraction of one phase. To account for the surface as well as varying fluid properties, some further modifications have to be made:

- Fluid properties are interpolated between phase 1 and 2 as function of the phase fraction.

$$\phi = \alpha \phi_1 + (1 - \alpha) \phi_2 \quad (4.43)$$

- The momentum equation is extended by a term representing surface tension [Rusche 2002].

Since for this method an indicator function is used, it must be ensured that the values obtained in the computation are bounded ( $0 \leq \alpha \leq 1$ ). This can be achieved by selecting a suitable interpolation scheme or limited schemes. Still, oftentimes the interface is smeared over few cells [Ferziger and Perić 2002]. To obtain a good idea of the surface, it is necessary to have a sufficiently refined grid in the vicinity of the interface.

#### 4.6.2. Passive Scalar Transport

Another type of multiphase is the transport of arbitrary scalar properties. If these properties take the form of particles, this can either be solved by a Lagrangian approach, in which the particles trajectory is solved additionally to the fluid's equations, or by an Eulerian approach similar to the Volume of Fluid method. For properties that do not even influence the fluid properties, an even more simplified method can be applied: passive scalar transport.

This method can e.g. be used for dissolved substances that do not influence the fluid's

physical properties substantially. For a single phase, this scalar transport of a concentration  $c$  can be described by the transport Equation (4.44).

$$\frac{\partial c}{\partial t} + \nabla \cdot (c\mathbf{u}) - \nabla \cdot (D\nabla c) = 0 \quad (4.44)$$

For cases with more phases, however, the equation becomes more difficult. This is due to the fact that a concentration jump can be expected at the phase boundary and mass transfer over the interface must be calculated correctly. Haroun et al. [2010] developed a model for two-phase systems that employs the Henry coefficient  $He$  of a fluid/gas equilibrium. It adds an additional source term  $S_c$ , which corresponds to the mass transfer over the interface (Equation (4.45)).

$$\frac{\partial c}{\partial t} + \nabla \cdot (c\mathbf{u}) - \nabla \cdot (D\nabla c) = S_c \equiv -\nabla \cdot (\phi_{He}c) \quad (4.45)$$

where  $\phi_{He} = D \frac{1 - He}{\alpha + (1 - \alpha) \cdot He} \nabla \alpha$

Since the binary diffusion coefficient may differ between the phases, it also has to be interpolated. In this case Nieves-Remacha et al. [2015] found that a harmonic interpolation of the diffusion coefficient  $D$  (Equation (4.46)) behaves better than the linear interpolation as given in Equation (4.43).

$$D = \frac{D_1 D_2}{\alpha D_1 + (1 - \alpha) D_2} \quad (4.46)$$

A special case occurs, when the transported species cannot pass through the interface. Following the same procedure as Haroun et al. [2010], one can develop a source term, which prescribes an equilibrium concentration  $c_{eq}$  at the interface (Equation (4.47), [Cyprien 2012]) or prohibits flow over the interface, but allows the species to be existent in the second phase (Equation (4.48)).

$$S_c \equiv c_{eq} \nabla \cdot (D\nabla \alpha) \quad (4.47)$$

$$S_c \equiv -\nabla \cdot (D\nabla \alpha \cdot c) \quad (4.48)$$

## 5. Experimental and Simulative Environment

This chapter will introduce the experimental and simulative environment used: First, the reactor employed in the experiments will be described. Therein details to the separate reactor units as they have been modelled in the simulations will be provided. The second section will give some details to the operational conditions of the reactor, the derived parameters, and limits for simulative parameter studies. Last, the tools employed for the simulations as well as the hardware environment will be listed.

### 5.1. Thin-Layer Cascade Photobioreactor

The Thin-Layer Cascade (TLC) photobioreactor has originally been developed by Šetlík et al. [1970]. In the thesis at hand a modified version of the reactor has been used [Apel, Pfaffinger et al. 2017]. It consists of five individual, modular units (Figure 5.1) that are passed by the microalgae suspension in their order: The suspension is pumped into the inlet tank (1), where it is distributed onto the following upper channel (2). At the channel's end it is redirected onto the second, lower channel (4) by the flow reversal tank (3) before it is collected in the retention tank (5).

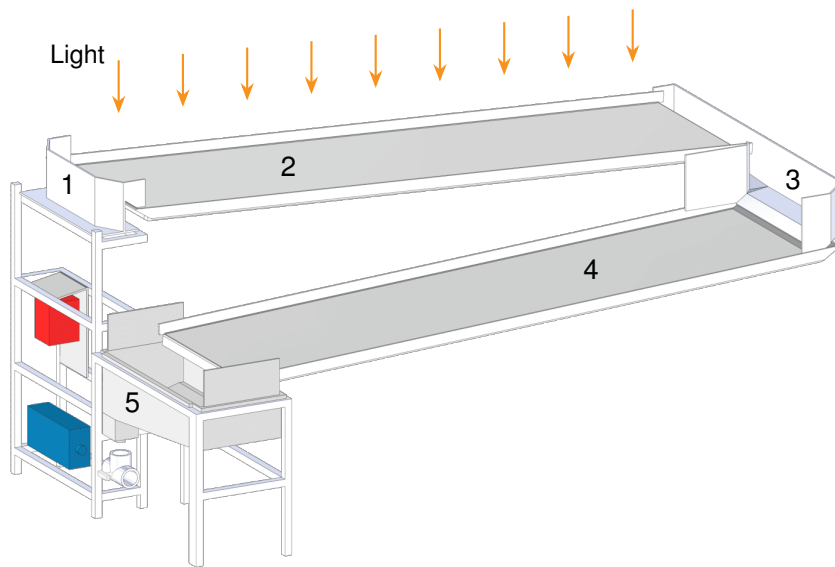
The complete reactor is made of 10 mm thick, high-density polyethylene (Rauch, Feldkirchen, Germany). From the retention tank a PVC hose (Rauspiraflex Liquitec, Rehau, Rehau, Germany) connects to the inlet tank via a magnetically coupled centrifugal pump (MKPG, Ventaix, Monschau, Germany) operated by a frequency inverter (Movitrac, SEW Eurodrive, Bruchsal, Germany). In order to determine the volume flow rate a magnetic-inductive sensor (MIK, Kobold, Hofheim, Germany) was added between pump and inlet tank [Apel, Pfaffinger et al. 2017].

The reference reactor used for validation measurements has been operated at a slope of  $1^\circ$  and a volume flow of  $2.4 \text{ l s}^{-1}$ . Both channels had a length of approximately 4 m and a width of 1 m. A summary of the reactors nominal parameters can be found in Table 5.1.

**Table 5.1** Summary of the relevant parameters of the Thin-Layer Cascade (TLC) [Apel, Pfaffinger et al. 2017].

Parameter	Value	Parameter	Value
Reactor Area	$8 \text{ m}^2$	Cultivation volume	50 l
Surface-to-volume ratio	$160 \text{ m}^{-1}$	Thereof in	
Slope	$1^\circ$	Inlet	12 %
Channel Width	1 m	Channels	64 %
Volume Flow	$2.4 \text{ l s}^{-1}$	Flow reversal tank	8 %

In the next sections the geometries of the inlet tank, the channel, and the flow reversal tank will be detailed. Geometric and operational parameters, some of which can be adjusted to change the units flow behaviour, are highlighted, and their reference values as well as ranges in the simulation are noted.



(a) Schematic drawing of reactor's design.



(b) Photo of the pilot plant reactor.

**Figure 5.1** Visualisation of the complete Thin-Layer Cascade used in this thesis as 3D representation and with a photo of the experimental system. The separate reactor units are marked in the schematic drawing: inlet tank (1); upper channel (2); flow reversal tank (3); lower channel (4); retention tank (5). The red box marks the control unit for the pump (blue box). Next to the pump a three-way valve is shown, which connects the retention tank to the pump. The drawing of the piping has been omitted in favour of clarity. Original images by Andreas Apel, modified for this thesis.



All parameters can be assigned a category, which defines how they have to be treated in the simulation and parameter studies:

*Geometric* These parameters are limited by geometric constraints, e.g. the width at the transition from the inlet to the channel. They can, however, be changed for the complete reactor. This would be necessary in case of a scale-up, where wider channels are used and thus the inlet has to be adjusted.

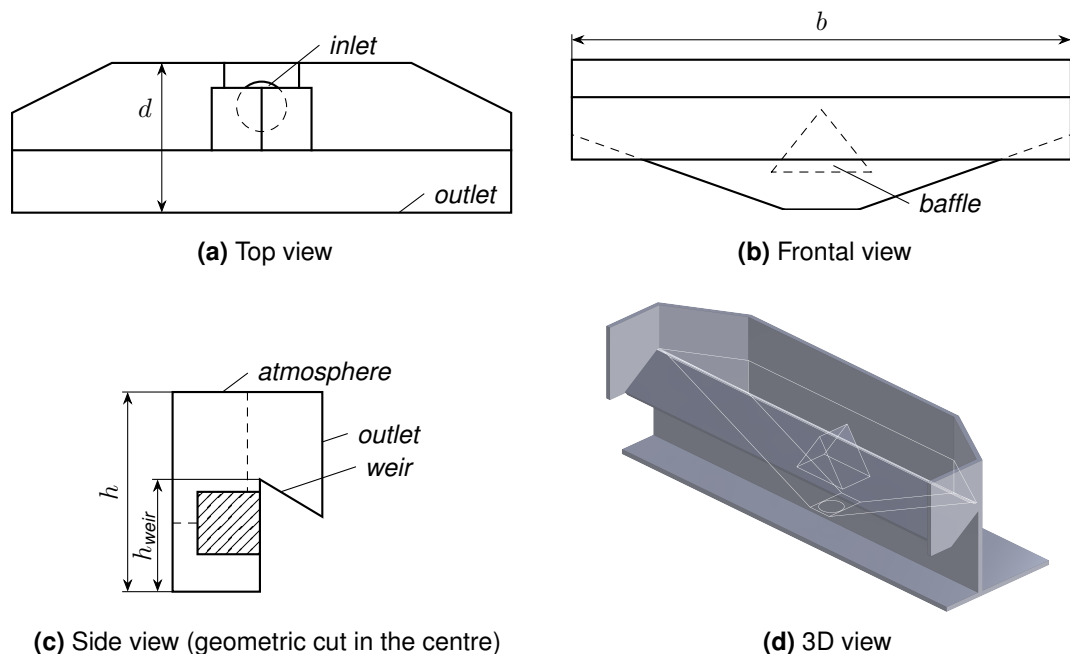
*Simulative* Some (geometric) parameters are to be set in the simulation, they do not necessarily have to reflect reality. This is often used to simplify the solution, e.g. by using symmetric conditions or reducing length/width of the channel in quasi-stationary regions.

*Free* Apart from common sense, these parameters are not limited by any constraint and can be chosen freely to investigate their influence on the reactor performance or achieve desired effects.

*Measured* As a result of a distinct setup, these quantities can be measured and used for comparison of the simulation with the experimental setup. They usually cannot be manipulated directly, but may influence the reactor's performance significantly, and have thus to be considered (e.g. layer thickness).

Furthermore, the boundaries used in the simulation will be shown at the corresponding geometrical feature of each unit.

### 5.1.1. Inlet Tank



**Figure 5.2** Sketch of the inlet tank with the primary geometry features and measures highlighted. The inlet tank has the boundaries *inlet*, *outlet*, *atmosphere*, *baffle*, and (not marked) *walls*.

The inlet tank can be seen in Figure 5.2. The algal suspension enters the inlet through a central pipe at the ground and is distributed by a baffle. After the water level reaches a height

$h_{water} > h_{weir}$ , the fluid leaves the inlet tank and immediately flows onto the upper channel. Thus the requirements on this unit can be derived: The algal suspension must be distributed evenly over the complete width of the weir in order to ensure a uniform layer thickness on the following channel. Dead zones or non-mixed areas in the volume must be avoided to allow the fast mixing of added substances and algal cells.

To achieve these goals a number of parameters can be adjusted (compare Table 5.2): The height of the weir can be modified to change the hydrostatic pressure that will build up inside the tank. Contrasting, the volume flow defines the dynamic pressure with which the suspension enters the tank, and thus balances with the hydrostatic pressure. Both can be adapted to reduce the fountain-like overshooting of the fluid. Furthermore, the baffle itself could be changed from its default, triangular shape, to reduce pressure loss after the inlet while maintaining a sufficient distribution of the fluid to both sides of the reactor. The further parameters depth and width are defined by the overall reactor design: The width is given by the scale of the system and has to match the width of the following channel. The depth is limited due to the size of the inlet adapter and the connected hose. Apart from these restrictions, the overall volume of the tank has to be considered, since — in regard to the microalgae — it does not take part in the cultivation and reduces the surface-to-volume ratio  $S/V$  and thus the efficiency of the reactor.

**Table 5.2** Parameters and their (used) range for the inlet tank unit. Parameter without range are just listed for completeness and were not changeable.

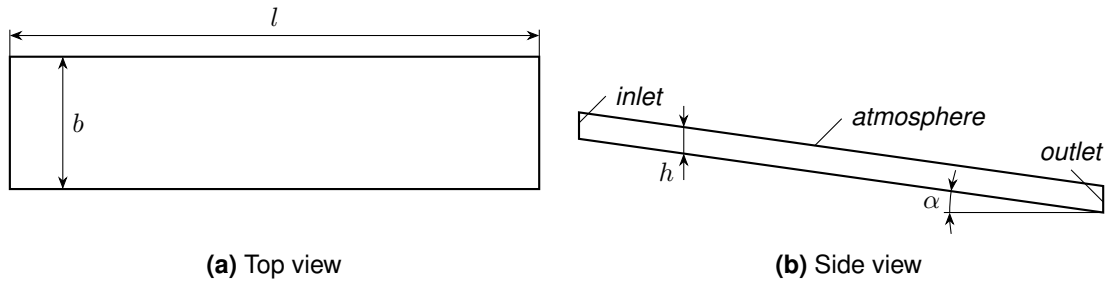
Parameter	Type	Reference	(Typical) Range		Unit
			Lower	Upper	
width $b$	geometric	1	1	10	m
depth $d$	geometric	0.16	–	–	m
height $h$	simulative	0.27	–	–	m
weir height $h_{weir}$	free	0.18	0.16	0.25	m
volume flow $\dot{V}_b^a$	free	2.4	1.6	3.6	$l s^{-1} m^{-1}$
baffle type	geometric/free	triangular	–	–	

<sup>a</sup> The volume flow is normalised to the inlet width (which in most cases corresponds to the channel width) to allow easier comparability.

### 5.1.2. Channel

The channel (Figure 5.3), being the only reactor unit where light is available throughout the suspension, is the main cultivation area. Viewed as a separate unit, it is assumed that the suspension has been distributed evenly over the complete width and leaves the unit without any downstream obstacles affecting the fluid layer inside the channel.

The channel's parameters are shown in Table 5.3: The width, length and height are specifications that are given by the simulation. The height must be sufficient to allow the complete fluid layer to develop without the free surface being influenced by the atmospheric boundary. The width and length of the channel can be set to show a quasi-stationary behaviour, while limiting the amount of computational power required. The main factors influencing the layer thickness and fluid velocity as parameters measured on the channel are the volume flow and



**Figure 5.3** Sketch of the channel including adjustable parameters. The channels Boundary Conditions (BCs) are the *inlet*, *outlet*, *atmosphere*, the *walls*, and the *channel ground* as part of the walls.

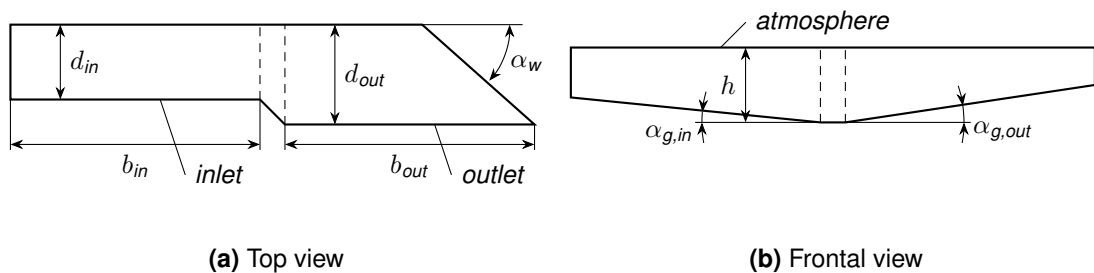
the slope of the channel, these can be chosen freely, although, of course, the volume flow has to match the ones defined by the inlet tank.

**Table 5.3** Parameters and their (used) range for the channel.

Parameter	Type	Reference	(Typical) Range		Unit
			Lower	Upper	
width $b$	simulative	0.2	0.2	1	m
length $l$	simulative	1.2	1	2	m
slope $\alpha$	free	1	0.5	3	°
height $h$	simulative	1	1	5	cm
volume flow $\dot{V}_b^a$	free	2.4	1.6	3.6	$\text{l s}^{-1} \text{m}^{-1}$
layer thickness $L$	measured	5.6	4	7	mm
flow velocity	measured	0.48	0.4	0.6	$\text{m s}^{-1}$

<sup>a</sup> The volume flow is normalised to the channel width to allow easier comparability.

### 5.1.3. Flow Reversal Tank



**Figure 5.4** Sketch of the flow reversal tank with adjustable geometric parameters. Boundaries shown are the *inlet*, the *outlet* and the *atmospheric* top. Remaining boundaries are *walls*. A *baffle* can be added in the simulation. The ground can be sloped as shown in the frontal view.

The flow reversal tank, shown in Figure 5.4, is the unit with the most parameters (Table 5.4). Its role is the collection of the water from the upper channel and the even redistribution on the lower channel. As with the inlet tank, the volume stored here does not contribute to the microalgal growth and should thus be minimised.

The width of the reversal tank's inlet  $b_{in}$  is (again) dictated by the channel's geometry. The depth  $d_{in/out}$  is a constructive measure, which is limited by the available space and restricted to keep the volume low. The width at the outlet  $b_{out}$  can be modified along with the outlet angle  $\alpha_w$  to redirect the flow on the lower channel with a remaining horizontal velocity for better distribution. The height  $h$  is again chosen to not influence the free surface flow in the simulation. The angles  $\alpha_{in/out}$  of the ground can be adjusted; this ensures that the unit can be emptied completely, while at the same time enforcing the flow away from the upper channel and slowing it again towards the lower channel.

Different from the previous reactor units, the flow reversal tank can also be fitted with further internal baffles that help to redistribute the water flow.

**Table 5.4** Parameters and their (used) range for the flow reversal tank.

Parameter	Type	Reference	(Typical) Range		Unit
			Lower	Upper	
inlet width $b_{in}$	geometric	1	1	10	m
inlet depth $d_{in}$	geometric	8	–	–	cm
outlet width $b_{out}$	geometric/free	1	1	10	m
outlet depth $d_{out}$	geometric	15	–	–	cm
outlet angle $\alpha_w$	free	45	40	70	°
height $h$	simulative	0.2	–	–	m
ground angle inlet $\alpha_{g,in}$	free	0	0	3	°
ground angle outlet $\alpha_{g,out}$	free	0	0	3	°
volume flow $\dot{V}_b^a$	free	2.4	1.6	3.6	l s <sup>-1</sup> m <sup>-1</sup>
internals	free	none	–	–	

<sup>a</sup> The volume flow is normalised to the channel width to allow easier comparability.

## 5.2. Thin-Layer Cascade Operation

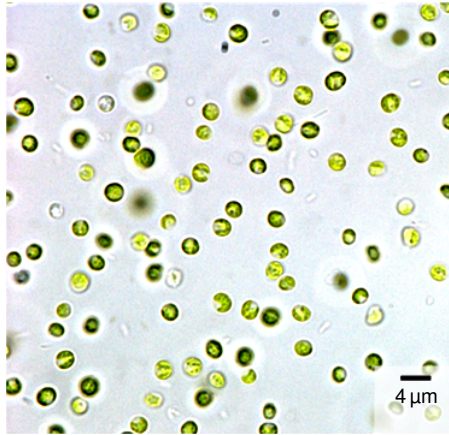
In the next sections details on the experimental conditions will be given, starting with the organism and continuing with the cultivation conditions. From these data limitations on the operational conditions of the reactor are then derived.

### 5.2.1. Microalgae

The microalgae strain used in this project was *Nannochloropsis salina* (SAG 40.85, Figure 5.5). Pfaffinger et al. [2016] published a model describing the growth kinetics of *N. salina*, exploiting a mean integral photon flux density  $I^*$  and the layer thickness  $L$  of the system (Equation (5.1)). Therein they described the light attenuation inside the fluid layer by the Lambert-Beer law with the absorption coefficient  $\varepsilon$ .

$$I^* = \frac{1}{L} \int_0^L I(c_X, l) dl \quad \text{where} \quad I(c_X, l) = I_0 \cdot e^{-\varepsilon \cdot c_X \cdot l} \quad (5.1)$$

This mean integral photon flux density can be used as an average irradiance each cell in

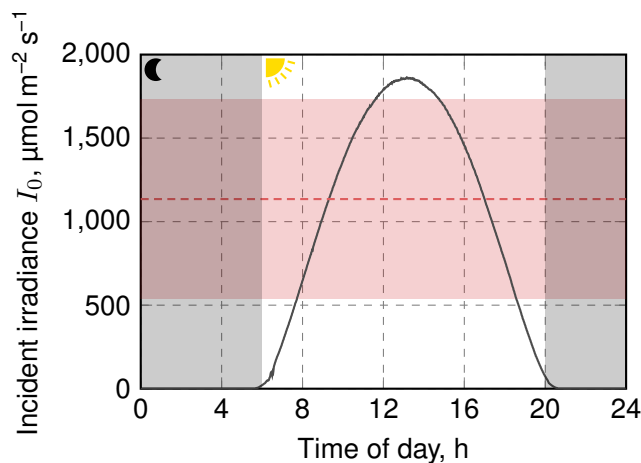


**Figure 5.5** Microscopic image of *Nannochloropsis salina* magnified by a factor of 100.

the fluid layer receives and can thus be used to compare the conditions in different scenarios with varying light conditions and fluid layers (e.g. during scale-up).

The absorption coefficient was calculated to  $\varepsilon = 0.67 \pm 0.14 \text{ l g}^{-1} \text{ cm}^{-1}$  [Pfaffinger 2017a]. The publication by Pfaffinger et al. [2016] also determined the optimal mean integral irradiance of *N. salina* at which the maximum growth can be observed to  $I^*_{opt}$  to  $551 \pm 166 \mu\text{mol m}^{-2} \text{ s}^{-1}$ .

### 5.2.2. Cultivation Conditions



**Figure 5.6** Irradiation profile used during the cultivation, as measured in Almería, Spain, 15 June 2012 ( $I_0$ , —). Also marked is the average incident irradiation received by the algae over the day ( $\bar{I}_0$ , - - -) and its standard deviation (□). The time of the day that is considered as day (6am–8pm) and night are also highlighted.

Cultivation of the microalgae was performed under irradiation conditions measured in Almería, Spain, on 15 June 2012 (Figure 5.6), provided by Stefan Wilbert (DLR, Plataforma Solar de Almería). From the course of the incident irradiation  $I_0$  the mean irradiance could be determined to  $\bar{I}_0 = 1135 \pm 596 \mu\text{mol m}^{-2} \text{ s}^{-1}$ .

To maximise productivity, the optimal concentration at which the microalgae are cultivated was found to be approximately  $10 \text{ g l}^{-1}$  in previous analyses. However, to allow for initial

growth and in case higher biomass fractions prove to be very advantageous for harvesting, the concentration range from 3–20 g l<sup>-1</sup> was considered.

### 5.2.3. Operational Constraints

**Table 5.5** Parameter range investigated for the layer-thickness of the Thin-Layer Cascade photobioreactor. Nominal and corresponding upper and lower limits were obtained from average and standard deviation of an experimental evaluation ( $\epsilon$ ), as a result of the kinetic model and its parameter uncertainty ( $I^*_{opt}$ ), as average of the naturally occurring conditions during the day, and the standard deviation ( $I_0$ ), or defined as reasonable target (range) for continuous cultivation ( $c_X$ ).

Parameter	Symbol	Min	Nominal	Max	Unit
Absorption coefficient	$\epsilon$	0.53	0.67	0.81	l g <sup>-1</sup> cm <sup>-1</sup>
Optimal Integral Photon Flux Density	$I^*_{opt}$	385	551	717	$\mu\text{mol m}^{-2} \text{s}^{-1}$
Incident Photon Flux Density	$I_0$	539	1135	1731	$\mu\text{mol m}^{-2} \text{s}^{-1}$
Biomass Concentration	$c_X$	3	11.5	20	g l <sup>-1</sup>

The summary of all constraints derived above can be found in Table 5.5. Using these values in a transformed Equation (5.1), the layer thickness  $L$  for the optimal cultivation conditions can be estimated iteratively from Equation (5.2).

$$L = I_0 \cdot \frac{1}{I^*_{opt} \cdot \epsilon \cdot c_X} \cdot (1 - e^{-\epsilon \cdot c_X \cdot L}) \quad (5.2)$$

Solving this equation for the parameter ranges in Table 5.5, the range of viable layer thicknesses can be plotted for varying sets of parameters (Figure 5.7). Based on these data, the operational parameters should be chosen to have a layer thickness in the computed range. The image shows that the optimal layer thickness is in the range of 2 to 8 mm. Higher levels of suspension in the cultivation are only desirable at very strong incident radiation or low biomass concentrations, e.g. at the beginning of a batch cultivation. On the lower side the layer thickness is limited by technical aspects, since the ground of the cultivation channel cannot be considered perfectly flat (e.g. due to using pool liner, which is often used in industrial-scale open photobioreactors).

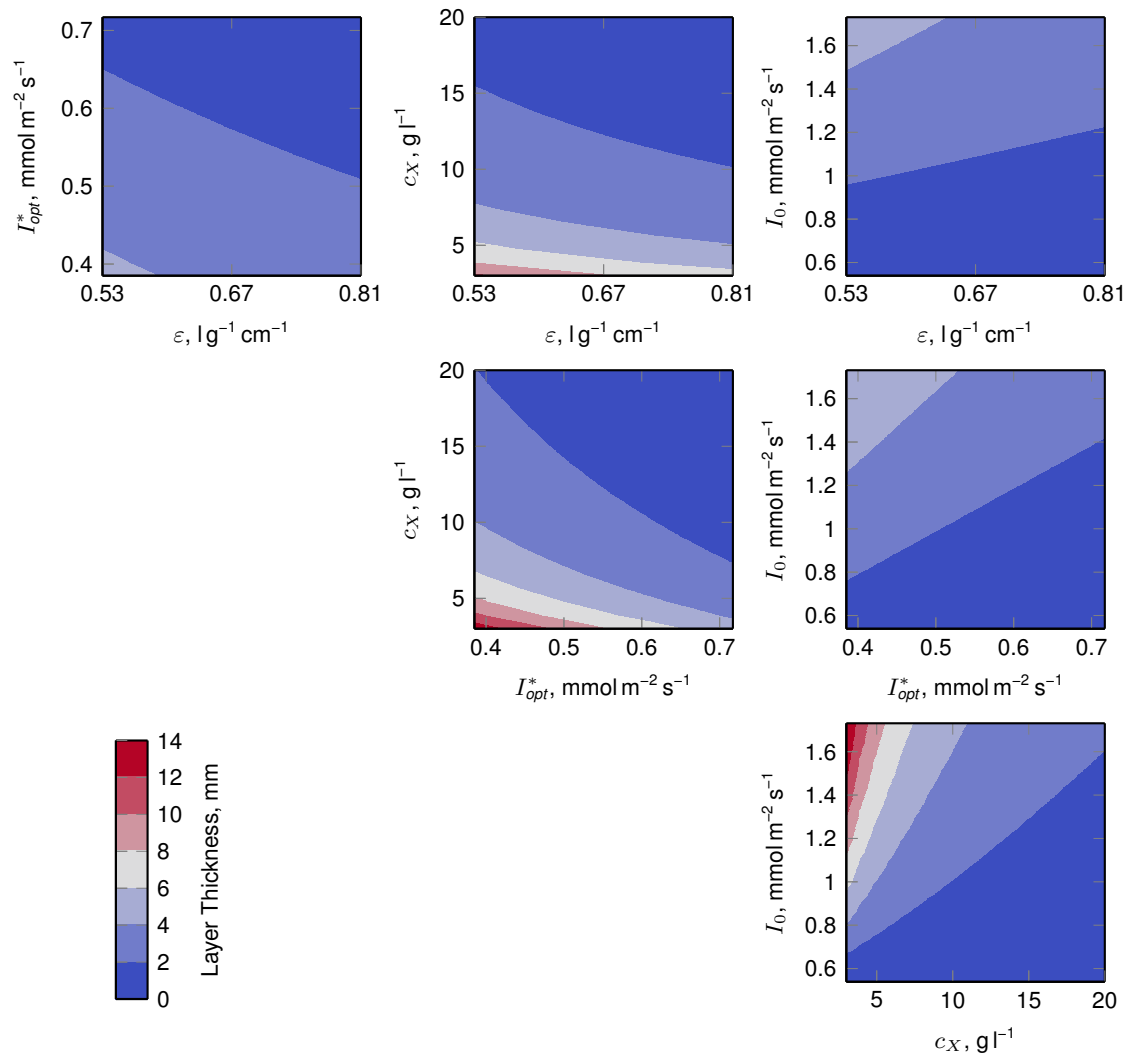
## 5.3. Simulation Approach

### 5.3.1. Computational Environment OpenFOAM

All simulations were performed with OpenFOAM 2.3.x (The OpenFOAM Foundation Ltd, London, UK; [Weller et al. 1998]). The installation was extended by the swak4Foam toolset in version 0.3.2.

### Hardware and Operating System

The simulations were performed on three different computers listed in Table 5.6. For graphical post-processing and data evaluation only the local workstation was used.



**Figure 5.7** The “optimal” layer thickness on the TLC photobioreactor as a function of four parameters when using the light growth/inhibition kinetics [Pfaffinger et al. 2016]. Each plot shows the variation of two parameters, while the remaining two were held at their nominal values according to Table 5.5.

**Table 5.6** A list of the hardware used for the simulation and post-processing of the simulations.

	Workstation	BVTCluster	Athena
Operating System	Linux Mint 17.2, 64bit	Linux Mint 17.3, 64bit	Ubuntu Linux 16.04, 64bit
Kernel	Linux 3.16.0-38	Linux 3.19.0-32	Linux 4.4.0-66
Processor	Intel Xeon E5- 2609 v2 @2.5 GHz 8 cores (2 sockets á 4 cores)	Intel Xeon E5- 2699 v3 @2.3 GHz 36 cores (2 sockets á 18 cores)	Intel Xeon E5- 2680 v4 @2.4 GHz 28 cores (2 sockets á 14 cores)
RAM	4 × 8 GB Hynix DDR3 1866 MHz ECC	8 × 8 GB Hynix DDR4 2133 MHz ECC	8 × 8 GB Samsung DDR4 2400 MHz ECC
Graphics	Nvidia Quadro K4000, Driver: nvidia	<i>unused</i>	<i>unused</i>

### 5.3.2. Pre-processing

To generate the mesh for the simulations some pre-processing steps were necessary: Complex 3D geometries were created using the Computer Aided Design (CAD) Software OpenSCAD 2015.03-1 (<http://www.openscad.org>) and exported in the Stereolithography (STL) format, which is readable by OpenFOAM. For some purposes, these geometries were further parsed using MATLAB<sup>®</sup> R2015b 64bit (The MathWorks, Inc., Natick, Massachusetts, United Stated). Meshes for simple geometries consisting of cuboid blocks could directly be calculated in OpenFOAM.

### 5.3.3. Post-processing

Graphical post-processing was done using ParaView 4.4.0, 64bit (Kitware Inc., New York, United States) [Ayachit 2015]. It was compiled with GCC 4.8.4. MPI was not supported. Python was included in version 2.7.6 with Numpy 1.8.2.

For the evaluation of the data MATLAB R2015b 64bit (The MathWorks, Inc., Natick, Massachusetts, United Stated) was used.



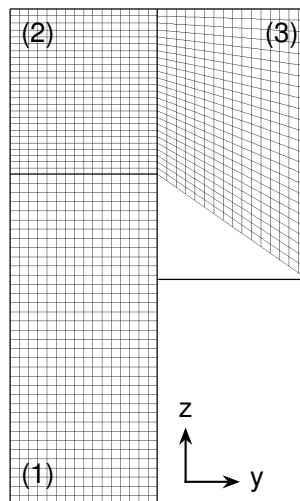
## 6. Simulation Setup

### 6.1. Mesh Generation

#### 6.1.1. Inlet

The inlet possesses a complex geometry that can not be constructed from pure hexagonal cells. It is thus constructed using a basic mesh, in which a Computer Aided Design (CAD) geometry is inserted using `snappyHexMesh`. To reduce computational effort, the (symmetric) inlet tank is assumed to behave identical around its symmetry axes in the centre. The exact procedure to generate the mesh is described below:

1. A coarse mesh is produced as bounding box using the OpenFOAM utility `blockMesh` (Figure 6.1). This mesh is created in three blocks, two describing the main tank ((1) and (2)), while one block is especially designed for the region above the weir (3). All blocks consist of purely hexahedral, cubic cells with a side length of 5 mm or less.



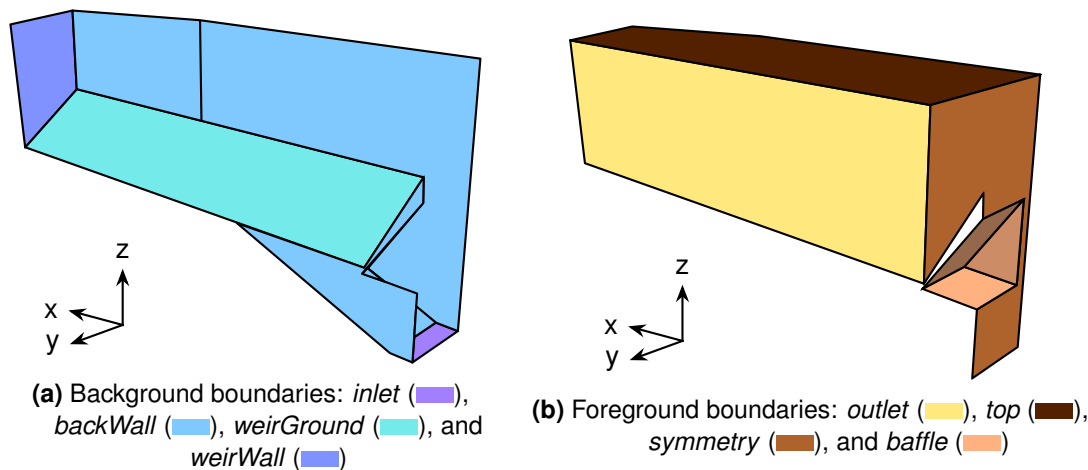
**Figure 6.1** The basic mesh for the inlet tank before the running `snappyHexMesh`. Block (1) and (2) will be modified by `snappyHexMesh` to give the main tank geometry, while block (3) already represents the primary features of the weir. Compare Figure 5.2c for the geometry.

In this approach, the cells directly above the weir are aligned with the flow direction, which is advantageous for the solution procedure. They are further refined in  $z$  direction with an expansion ratio of 1.15, which results in smaller cells in the region where the strongest flow is expected and thus a better resolution of the relevant flow behaviour and the interface is needed.

2. To speed up mesh creation, the following steps will be executed in parallel. Therefore the mesh is decomposed with the `decomposePar` utility using the `scotch` algorithm.
3. The complete geometry of the inlet tank is split into multiple surfaces, one for each

simulation boundary, and supplied to snappyHexMesh as separate geometric objects (Stereolithography (STL) files). The different surfaces are shown in Figure 6.2. Therein the *inlet*, *outlet* and *symmetry* surfaces correspond to their identically named boundaries. The *backWall*, *baffle*, *weirWall*, and *weirGround* surface are of the wall type and are separated to allow independent adjustments of the respective boundary mesh. The *top* boundary reflects the atmospheric boundary condition. The following refinements are then applied:

- a. All surfaces are refined twice in the adaptation of the basic mesh to the geometry. Since thinner fluid layers are expected at the *outlet* and on the *weirGround*, a three-time refinement is applied when necessary.
  - b. In the next step refinement regions are defined: Near the *backWall*, the *baffle* and the *weirGround* the inner region of the mesh is refined up to two times in a distance of 5 cm for the former two and 7.5 cm for the latter. This is done to account for wall layer effects in the fluid.
  - c. Since the mesh along the *weirGround* already is refined in the base mesh, no further layer addition is applied and the complete step is deactivated in snappyHexMesh.
4. The mesh generated in parallel by this procedure is reconstructed using the reconstruct-ParMesh utility. Figure 6.3 shows the final mesh generated by this procedure.



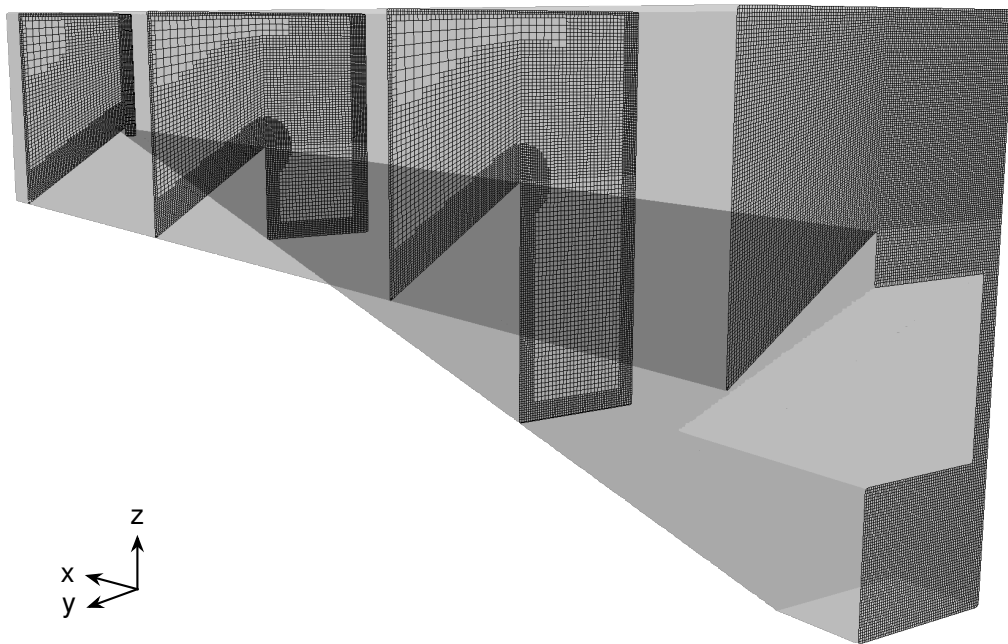
**Figure 6.2** The separate geometry elements from which the complete geometry of the inlet tank is constructed and which define the boundaries that are available in the simulation and snappyHexMesh. The left image 6.2a shows the surfaces that define the background of the volume from the given point of view, while Figure 6.2b shows the foreground boundaries.

In this procedure the CAD geometry of the inlet tank is required. This was generated using OpenSCAD, which allowed to create 3D geometries based on scripted input using simple shapes. The compiled geometry was then exported as STL file.

Since the exported STL geometry only contained one object, a MATLAB script was used to split it into multiple files: Therefore all faces were read and evaluated for their neighbouring

surfaces' normals. Surfaces for which the normal deviated less than  $1^\circ$  were considered to belong together. However, this operation still produced a high number of individual faces, which had to be merged manually to result in the above mentioned boundaries. The script (`splitSTLintoAreas.m`) can be found in the attachment to this thesis.

In order to be able to generate inlet tanks of different widths, a bash script was used, which re-calculated the base mesh (coordinates, number of cells) for the desired width and modified the dictionary for the basic `blockMesh` accordingly. Furthermore, the individual STL files were modified to account for the increased width. The corresponding script (`rescaleGeometry`) can also be found in the case folder of the inlet's base case in the attachment.



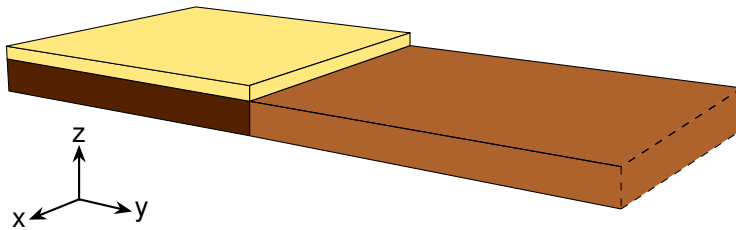
**Figure 6.3** Slices of the final mesh for the inlet tank. Shown are the symmetry boundary and the internal mesh at three positions inside the tank.

### 6.1.2. Channel

In contrast to the inlet, the channel is a geometrically very simple element. It can be represented by purely hexagonal cells, which is advantageous for the simulation (e.g. no non-orthogonal faces). Thus the procedure reduces to a few steps:

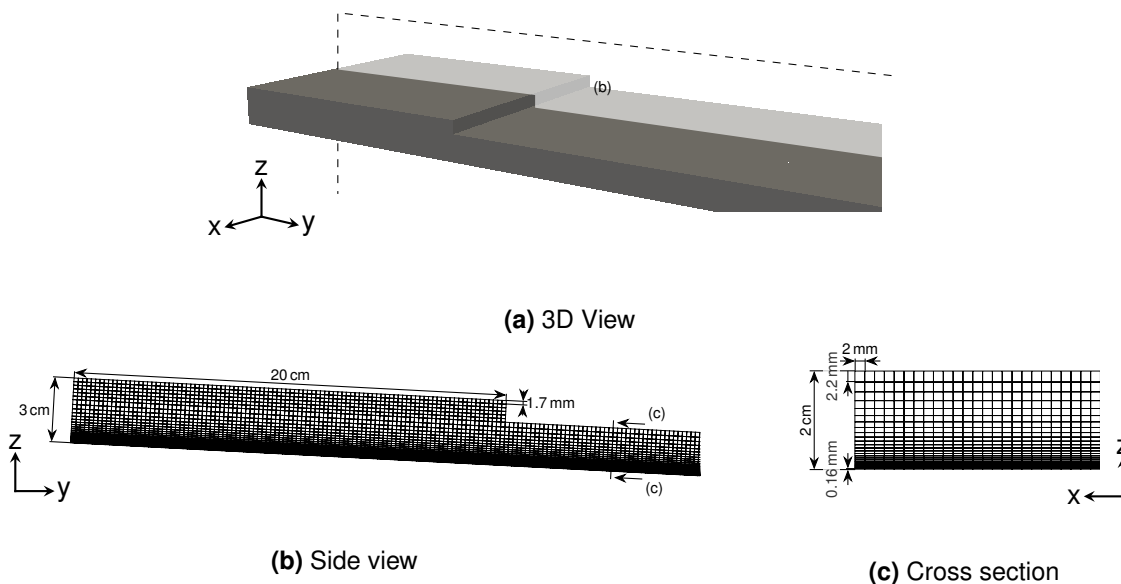
1. The complete channel geometry is defined in a dictionary for the `blockMesh` utility. It is defined by three blocks (see Figure 6.4): Block #1 (■) and #3 (■) define the inlet region, which has a length of 20 cm and — compared to the remaining channel with a height of 2 cm — an increased height of 3 cm. For block #3 (■) the mesh resolution is  $2\text{ mm} \times 2\text{ mm} \times 1.67\text{ mm}$  per cell. The  $x$  and  $y$  resolution for both lower blocks (■ and ■) is consistent to that ( $2\text{ mm} \times 2\text{ mm}$ ). In  $z$ -direction an expansion ratio of 1.12 leads to a cell height between 0.16–2.2 mm from ground to top.

2. Since the *inlet* (boundary) does not span the complete height of 3 cm, the upper 2 cm of the complete inlet are converted to a wall boundary. The area is selected using the `topoSet` tool with the custom `patchPartToFace` source (compare Appendix E.1). The conversion is done by the `createPatch` utility.
3. To adjust for different slopes of the channel unit, the complete mesh is finally rotated around the *x*-axis using `transformPoints`.



**Figure 6.4** The three blocks from which the channel unit is meshed as defined in the `blockMeshDict`. The inlet boundary for this geometry is in the background on the left side, the outlet far off the right (not shown). To account for a less defined surface at the inlet region, the channel has been given an increased height by adding a third block (yellow). Below this, the first (dark brown) is positioned and connects to the second (light brown) block, so that at each interface the mesh resolution is identical.

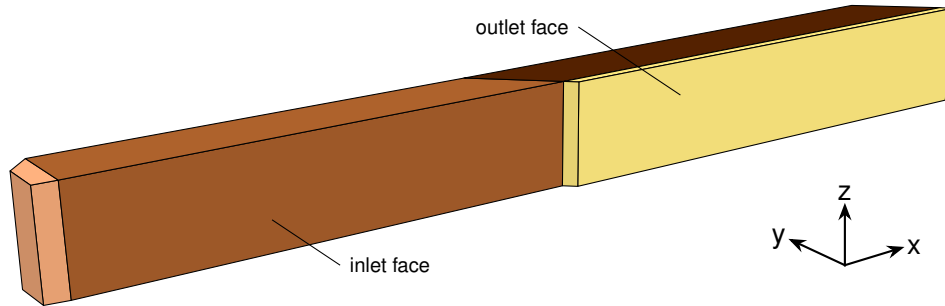
This procedure generates a mesh which is highly refined to the channel ground to allow detailed analyses of the thin layer fluid flow. Details of the mesh as described above can be seen in Figure 6.5.



**Figure 6.5** Mesh of the channel unit: (a) 3D representation of the channel inlet region; (b) side view of the mesh in the inlet region for a  $3^\circ$  inclined channel; (c) detail of the mesh perpendicular to the flow direction highlighting the cell refinement towards the channel ground.

### 6.1.3. Flow Reversal Tank

Although defined by several geometric parameters, the flow reversal tank can be meshed using purely hexagonal cells. To achieve this, the geometry has to be divided in four blocks as shown in Figure 6.6.

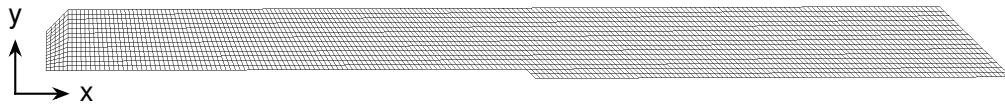


**Figure 6.6** The (hexahedral) blocks by which the flow reversal tank is meshed. The first (light orange) and second block (dark orange) define the inlet region, where the fluid coming from the upper channel is collected. Block #3 (dark brown) then takes the suspension and redirects it onto the lower channel with the outlet formed by the fourth block (yellow).

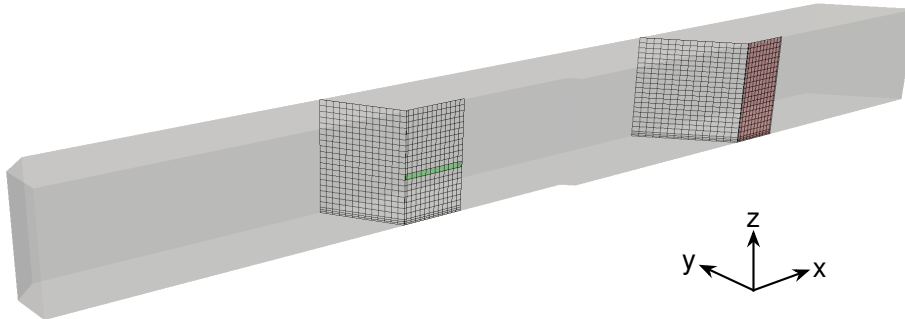
As stated above, the flow reversal tank can be fitted with baffles that help redistribute the suspension in the following channel. To add these inside the geometry, snappyHexMesh can be used again. Thus, the complete meshing procedure consists of the following steps:

1. The four blocks shown in Figure 6.6 are created using the blockMesh utility. The resolution is chosen to have a maximum cell lengths of 1 cm in all coordinate directions  $x$ ,  $y$ , and  $z$ . However, due to the skewing of the blocks up to an angle of  $45^\circ$ , the created cells have side length of up to 1.4 cm diagonal to the  $x$  and  $y$  axis (Figure 6.7a).
2. The upper channel ends approximately 5 cm above the ground of the flow reversal tank. To account for that, the actual *inlet* boundary of the flow reversal tank is set 5 cm above the ground with a height of 1 cm. This is done using the topoSet utility on the complete inlet face and selecting only the faces in the defined area. The patch was then created using the createPatch utility, which splits the inlet face into an *inlet* and *inletWall* boundary (Figure 6.7b).
3. If any further constructs were to be placed in the geometry as baffles, these were defined in a snappyHexMeshDict. If this dictionary was available, snappyHexMesh was executed automatically to mesh around the additional geometry.
4. In a final step, the refineWallLayerSmart utility [Wei 2012a] was used to refine the ground of the flow reversal tank by two levels with a stretch ratio of 1.2 for adjacent cells, supporting improved turbulence modelling and allowing thin fluid layers.

The resulting mesh of this procedure without any internal baffles can be seen in Figure 6.7. The top view, Figure 6.7a, shows the skewing of the cells due to the strict usage of hexagonal blocks as described above. In the lower image, Figure 6.7b, the inner mesh, including wall refinement at the ground, and the *inlet/outlet* boundaries can be seen.



(a) Top view on the mesh.



(b) 3D view showing the inner mesh and the *inlet/outlet* boundaries.

**Figure 6.7** The final mesh of the flow reversal tank assuming no internal baffles have been added. The top view shows the skewness of the cells due to the purely hexahedral approach. The mesh has been refined in proximity of the ground. The inlet (left, green) is only a small fraction of the surface, the remaining part is modelled as wall. The outlet (right, red) spans the complete height of the tank.

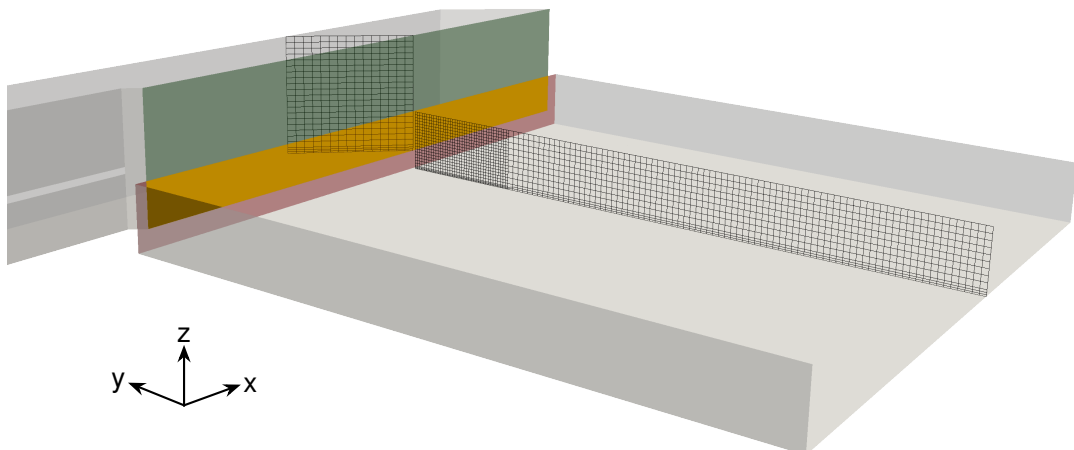
Using purely hexahedral cells comes at the cost of producing a non-orthogonal, skewed mesh. These properties can be checked using the `checkMesh` utility. It reports a maximum skewness of 1.369 and a maximum non-orthogonality of  $45^\circ$  with an average of  $30^\circ$ . Both values are below the thresholds defined by OpenFOAM (skewness  $\leq 4$ , non-orthogonality  $\leq 65^\circ$ ) and are thus considered acceptable.

Since the flow reversal tank redistributes the suspension on the lower channel, it should also be created to achieve an even surface over the width of the following lower channel. With the simulation geometry given above, this can not be evaluated, since the effects on the channel are not visible. Thus, a channel can be added at the outlet of the flow reversal tank, which allows to compare the effects of different tank geometries on the channel's fluid layer. To achieve this, a few additional steps are necessary:

5. The channel's mesh is generated with a length and width of 1 m, separately from the flow reversal tank:
  - a. The basic mesh is created by `blockMesh`. It creates equally spaced cells with a side length of 1 cm.
  - b. The first 20 cm of the channel are refined once in all directions to account for the transition zone between flow reversal tank and channel. The refinement is done executing `refineMesh` on cells selected via the `topoSet` tool and results in a side length of 0.5 cm per cell in the refined region.

- c. Along the ground refineWallLayerSmart with an adjacent stretch ratio of 1.2 is executed to further refine the mesh similar to the mesh in the tank by two levels to improve turbulence modelling and thin-layer resolution.
6. With both meshes generated, mergeMeshes is run to merge both separate grids into one. In this process the individual domains are only stored in one location, but are not connected and thus form two separate regions.
7. To connect the meshes, the outlet of the flow reversal tank and the inlet of the channel are selected using topoSet, and connected using the createBaffles utility. This tool sets the corresponding boundary patches to the `cyclicACMI` type, which allows the dynamic coupling of not matching mesh interfaces.
8. Finally, the initial and boundary field have to be restored, since they got overwritten by createBaffles.

The resulting geometry and mesh can be seen in Figure 6.8. It shows the combination of the flow reversal tank and the channel as well as the resulting mesh in the transition zone.



**Figure 6.8** The merged meshes of the flow reversal tank and a 1 m long part of the channel. At the boundary, the outlet of the flow reversal tank (green) and the inlet of the channel (red) are connected using a `cyclicACMI` Boundary Condition. This Boundary Condition (BC) allows the interpolation of the flow in the intersection region (yellow), while the remaining areas are considered as walls.

The `cyclicACMI` BC used at the interface gives a simple method to combine both meshes: It defines an Arbitrarily Coupled Mesh Interface (ACMI), which can detect an overlap with its defined neighbouring patch. For the *outlet* of the flow reversal tank (green) and the *inlet* of the following channel (red) the overlap is highlighted (yellow). The BC now distinguishes between the overlapping and non-overlapping area by applying two different BC: For the overlapping area an interpolation between the two meshes is used to couple them. For the remaining area, another BC can be applied, in this case a wall.

**Listing 6.1** Implementation of the general transport equation for a single passive species  $Y$  that does not transfer over the phase boundary (compare Equation (4.44)).

---

```
1 // calculate the mixture diffusion coefficient and interpolate it to the
  cell surfaces from the diffusion coefficient D_ and DPhase2_ for the
  different phases
2 tmp<surfaceScalarField> DS = fvc::interpolate
3 (
4   D_ * alpha_ + DPhase2_ * (1 - alpha_)
5 );
6
7 // get the phiAlpha field (compare NievesRemacha2015)
8 surfaceScalarField phiAlpha =
9 (
10  DS() * fvc::snGrad(alpha_) * alpha_.mesh().magSf()
11 );
12
13 // create the complete species transport matrix
14 // define the solution schemes manually
15 fvScalarMatrix YEqn
16 (
17   fvm::ddt(Y_)
18   + fvm::div(phi, Y_, "div(phi,Y)")
19   - fvm::laplacian(DS(), Y_, "laplacian(Y)")
20   + fvm::div(phiAlpha, Y_, "div(phi,Y)")
21 );
22
23 // solve the equation
24 YEqn.solve(mesh.solver("Y"));
```

---

The advantage of this boundary is that it does not require the interface meshes to be identical, thus allowing modification (e.g. by `snappyHexMesh`) on any side of the interface without the need to recalculate the other mesh. However, the interpolation only works successfully if both meshes are sufficiently similar, e.g. the cell volume does not change too much. Otherwise it will result in significant interpolation errors. This condition was matched by the mesh created as described above

## 6.2. Solution Procedure

### 6.2.1. Two-Phase Flow Simulation

Two-phase simulations were performed using the solver `interFoam`, which comes with OpenFOAM. `interFoam` implements a Volume of Fluid (VoF) method to capture the free surface of the fluid and solves the pressure-velocity coupling iteratively using the PIMPLE algorithm [Rusche 2002].

### 6.2.2. Two-Phase Flow Simulation with Passive Species Transport

In order to simulate a passive species transport equation (Section 4.6.2), the new solver `interSpeciesFoam` was derived from `interFoam`. It implements the general transport equation similar to Nieves-Remacha et al. [2015], but with a source term that suppresses the phase change (Equation (4.48), Listing 6.1).



**Listing 6.2** Example for a speciesProperties dictionary. It contains a list of species that should be considered in the solution process as well as a set of default species properties which are used in the calculation.

---

```

1 /*----- C++ -----*\
2 | ===== |
3 | \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
4 | \\ / O p e r a t i o n | Version: 2.3.0 |
5 | \\ / A n d | Web: www.OpenFOAM.org |
6 | \\ / M a n i p u l a t i o n | |
7 \*-----*/
8 FoamFile
9 {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     location      "constant";
14     object        speciesProperties;
15 }
16 // * * * * * //
17
18 // List the names of all species to solve for, for each species a
19 // boundary condition must with the same name must be supplied species
20 (
21     NaCl
22 );
23
24 // Default properties if no specific data is entered
25 defaultProperties {
26     // Diffusion constant, m^2 s^-1
27     D      D [0 2 -1 0 0 0 0] 0;
28
29     // Molar mass, kg * mole^-1
30     M      M [1 0 0 0 -1 0 0] 0e-3;
31
32     // Default dictionary for all phase boundaries
33     phaseBoundary
34     {
35         // Type of the phase transition modelling
36         phaseModel      noTransition;
37     }
38 }

```

---

Based on this implementation, a library was developed which allowed the simulation of multiple passive species with different phase transition schemata. This library implements all phase transition schemes introduced in Section 4.6.2 and reads the settings for the solution from the constant/speciesProperties dictionary in the case folder. An example for this dictionary can be seen in Listing 6.2.

This dictionary can be extended by entries which are specific to a given species, e.g. NaCl, by supplying an additional sub-dictionary. Using the molar mass of  $58.4 \text{ g mol}^{-1}$  and a diffusion coefficient of  $1.99 \cdot 10^{-9} \text{ m}^2 \text{ s}^{-1}$  in the water phase the used dictionary can be found in Listing 6.3.

To add this library to the interFoam solver, two primary code fragments have to be added (Listings 6.4 and 6.5): An object for the library has to be created, which reads the properties defined for all species. In the time iteration of the solver, a loop over all species solves the

**Listing 6.3** The NaCl specific sub-dictionary of the speciesProperties dictionary defining the diffusion and phase change behaviour of this species.

---

```
1 // Species specific dictionary
2 // Sources: http://oto2.wustl.edu/cochlea/model/diffcoef.htm
3 NaCl
4 {
5     // Molar mass
6     M          M [1 0 0 0 -1 0 0] 58.4e3;
7
8     // Properties of NaCl that are only valid in the phase "water"
9     water
10    {
11        D          D [0 2 -1 0 0 0 0] 1.99e-9;
12    }
13
14    // Properties of NaCl that are only valid in the phase "air"
15    // We want to prevent any diffusion here!
16    air
17    {
18        D          D [0 2 -1 0 0 0 0] 0.0;
19    }
20
21    /*
22     * Phase transition for water/air phase boundary
23     * To prevent transition use noTransition or define the equilibrium
24     * concentration if a steady state at this concentration is expected
25     * to be achieved at some point.
26     * Both are assumptions that might produce errors!
27     */
28    phaseBoundary_water_air
29    {
30        phaseModel    noTransition;
31        //            phaseModel    constantEquilibrium;
32
33        Yeq            Yeq [0 0 0 0 0 0 0] 0;
34    }
35 }
```

---

**Listing 6.4** Code used in the initialisation of the solver to create the library required to solve for passive species transport.

---

```
1 // create a list of phases manually, could be automated
2 wordList phaseNames;
3 phaseNames.append("water");
4 phaseNames.append("air");
5
6 // create the species transport library for two phase flows
7 twoPhaseMultiComponentMixture speciesMixture
8 (
9     speciesProperties,      // the speciesProperties dictionary
10    speciesNames,          // the species names extracted from the
11                            // dictionary
12    phaseNames,            // the names of the phases
13    mesh                    // reference to the mesh
14 );
```

---

**Listing 6.5** Code used in the runtime loop of the solver for each time iteration after the flow field has been solved. Since no interaction of the species with the fluid is assumed, they can be solved independently from the velocity and pressure field.

---

```
1 // loop over all species defined
2 forAll(speciesMixture.species(), specieI)
3 {
4     // create the matrix for each species and obtain the transport
5     // equation from the library, which respects the phase change term
6     // defined in the corresponding dictionary
7     fvScalarMatrix YEqn
8     (
9         fvm::ddt(speciesMixture.Y(specieI))
10        + speciesMixture.transport(phi, specieI)
11    );
12    YEqn.solve(mesh.solver("Y"));
13    speciesMixture.calculateFlux(specieI);
14 }
```

---

transport equations. The solution can be generated after and independent from the velocity and pressure calculations, since for passive species (by definition) no interaction with the fluid occur.

The library assumes that the divergence scheme used for the species transport is the same for all species ( $\text{div}(\phi, Y)$ ). Generally a bounded scheme has the advantage of not violating the minimum and maximum concentration. However, a pure upwind scheme can not be recommended, and in exemplary evaluations best results could be obtained using a linearUpwind scheme ( $\text{div}(\phi, Y)$  Gauss linearUpwind grad(U);).

The complete library (twoPhaseMultiComponentMixture) and the modified solver implementing passive species (interSpeciesFoam) can be found in the attachment to this thesis.

## 6.3. Comparison of Turbulence Models

Since turbulence plays a major role in the mixing of fluids, its effects should be accounted for with sufficient precision. However, due to computational limits, from the introduced turbulence modelling techniques in Section 4.4 only Reynolds-Averaged Navier-Stokes (RANS) models are feasible. Thus the two most commonly applied models,  $k - \varepsilon$  and  $k - \omega SST$ , were compared. As geometry for the comparison the channel was chosen, since it is influenced the most by wall layer effects due to its thin fluid layer. Furthermore, this thin layer is a limiting measure for the computation of turbulent effects, and can also be found in the other two geometries: The inlet tank has a thin fluid layer on the weir, while the flow reversal tank requires thin-layer modelling if it is combined with the following channel.

For the comparison different cases were analysed, out of which the four most relevant are presented in the next section. This will be followed by a comparison of the effects the choices have on the simulation results and concluded with an evaluation for the choice of turbulence model.

### 6.3.1. Cases

The cases shown in this section for the comparison of the turbulence models use three different meshes for the  $k - \varepsilon$  model, one of which is the reference mesh shown above. Additionally, this reference case has been evaluated using the  $k - \omega SST$  model.

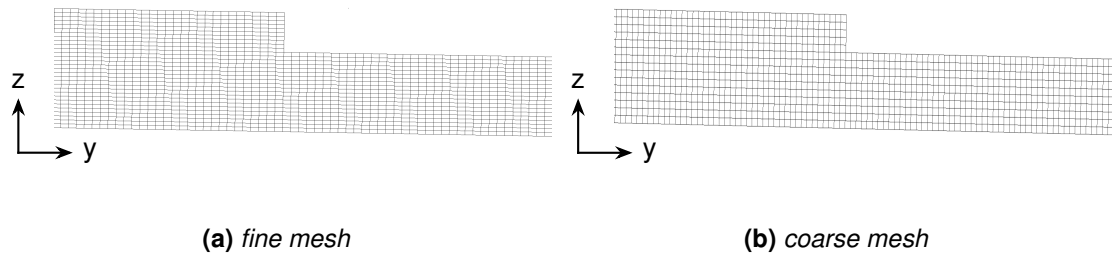
All cases have been run for 10 s with an adaptive time step ( $Co \leq 6$ ). A quasi-steady state in the flow behaviour 0.5 m after the inlet and 5 s after start of the simulation was assumed. After running the cases, the  $y^+$  value for the complete channel ground in the steady-state region was determined. The number of cells in the fluid layer was counted manually in ParaView. Flow profiles and water height were sampled 1 m after the inlet at the centre of the channel in vertical ( $z$ -axis) direction. This gave the following four different cases and basic properties:

*reference* The reference case uses the mesh shown in Section 6.1.2 and the  $k - \omega SST$  turbulence model. In the simulation a  $y^+$  value of  $2.40 \pm 0.01$  can be calculated. In this configuration 14–15 cells are in the fluid layer.

*same mesh* As the name implies, the mesh is identical to the one in the *reference* case, however here the  $k - \varepsilon$  model is applied. It results in a  $y^+$  value of  $2.90 \pm 0.05$  with the fluid layer being resolved by 15–17 cells.

*fine mesh* The *fine mesh* is a case with an equally spaced mesh with a cell height of 1 mm ( $x/y$  direction unchanged from reference, Figure 6.9a). The  $k - \varepsilon$  turbulence model is used. This results in 4–5 fluid layer cells and  $y^+ = 14.0 \pm 0.5$ .

*coarse mesh* To increase the  $y^+$  value to  $40.00 \pm 0.15$  the *fine mesh* has been coarsened to a cell height of 2 mm (Figure 6.9b). Although the  $y^+$  is now in the appropriate range for the  $k - \varepsilon$  model, this leaves only 2–3 cells that describe the fluid layer.



**Figure 6.9** The inlet region of the meshes used for the *fine mesh* and *coarse mesh* case. In the fine mesh, the cell height in the vertical direction is 1 mm, for the coarse mesh 2 mm.

### 6.3.2. Comparison of Turbulence Effects

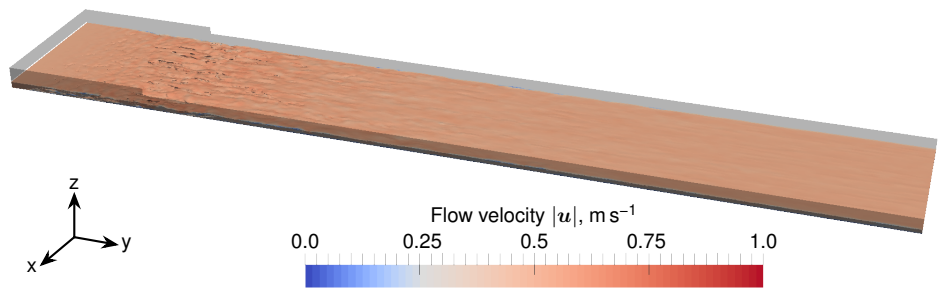
The fluid surface in all simulations can be seen in Figure 6.10. These already give a first indication of the performance of the used settings: For the *fine mesh* case, the water builds up at the outlet. This indicates that the flow at the outlet boundary is not yet fully developed (requirement for `zeroGradient` BC). The coarser mesh does not show this behaviour, however, its surface shows ripples all along the channel. These ripples can be an effect of the interpolation. Due to the — compared to the fluid layer — large cells, the water volume fraction in each cell varies strongly, resulting in fluctuating results of the interpolation algorithm.

For both cases using the reference mesh, a deviation in velocity can be seen. However, no further differences are apparent. Figure 6.11 thus shows the flow profiles of the fluid in  $y$ -direction for the three quasi-steady state cases. It can be seen that all three result in significantly different flow profiles (approximately 30% deviation to the reference case) over the water height (solid lines). Here it can also be seen that for the *coarse mesh* ( $\Delta$ ) only very few points in the fluid layer can be found.

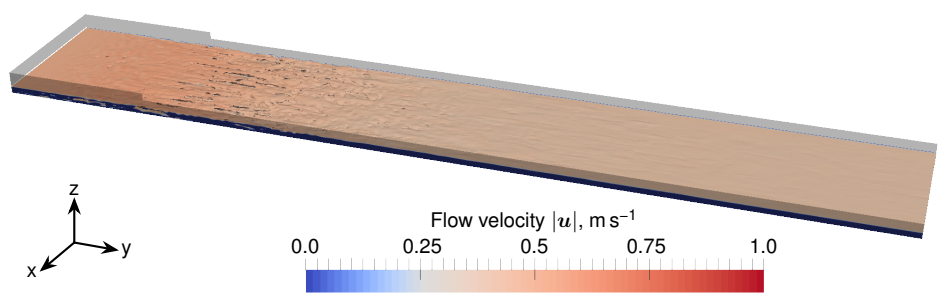
Another criteria is the water volume balance on the channel. This is displayed in Figure 6.12. For the *coarse mesh* this measure reflects the surface ripples seen above as oscillating. However, in average the mass (here: volume) balance is satisfied for all cases.

### 6.3.3. Evaluation of Turbulence Models

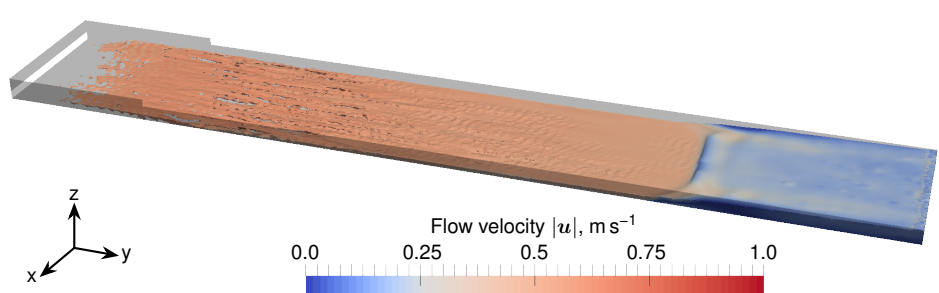
In the direct comparison it could be shown that the reference case given above in combination with the  $k - \omega$  *SST* turbulence model is the most promising, since it gives a sufficiently fine resolution of the fluid layer while obeying the theoretical restrictions of the  $y^+$  value for turbulence wall functions. The simulation using the *same mesh* with the  $k - \epsilon$  model gives reasonable results, however violates the condition of  $y^+ \leq 30$  at the channel ground. This can be changed by using a much coarser mesh, however, this also results in a resolution of the fluid layer with only 2–3 cells, which is insufficient to investigate the fluid behaviour. Finally, the case using a *fine mesh* did not converge to a physical solution, and can thus be excluded right away. The results are summarised in Table 6.1.



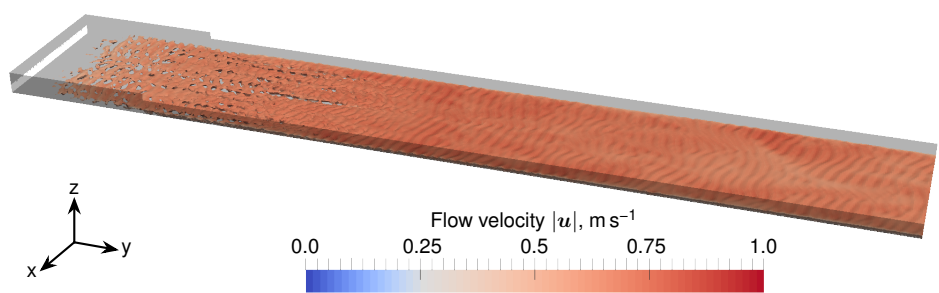
(a) reference case



(b) same mesh case

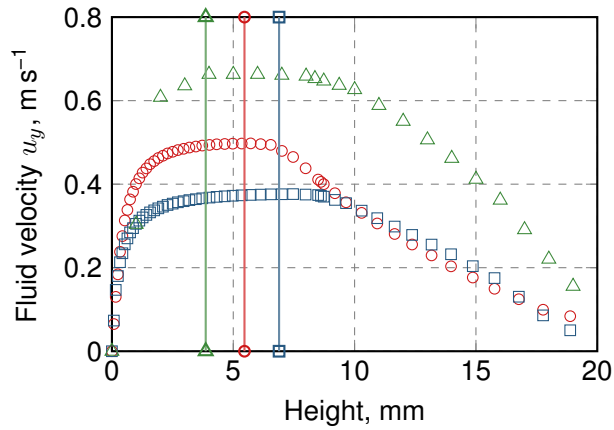


(c) fine mesh case

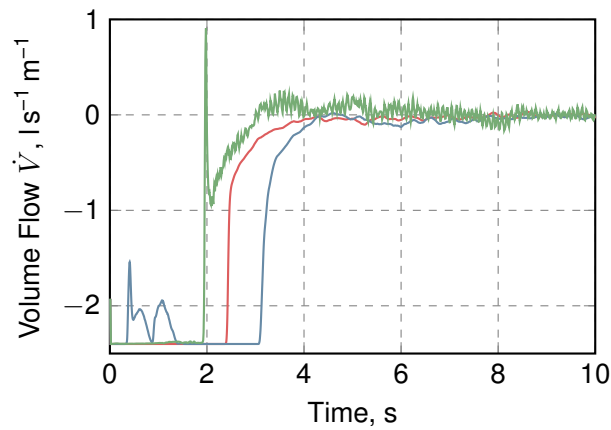


(d) coarse mesh case

**Figure 6.10** The fluid surfaces and the magnitude of the flow velocities for all four cases at  $t = 10$  s. Note that the colour bar is biased towards the velocities above  $0.25 \text{ m s}^{-1}$ .



**Figure 6.11** Velocity on the channel in  $y$ -direction for the *reference* case ( $\circ$ ), the *same mesh* case ( $\square$ ) and the *coarse mesh* ( $\triangle$ ). The data has been extracted using the sample utility at all cell centres and faces. Also shown is the water height for each case ( $-$ ,  $-$ ,  $-$ ).



**Figure 6.12** Water balance on the channel for the simulated cases *reference* ( $-$ ), *same mesh* ( $-$ ), and *coarse mesh* ( $-$ ). Shown is the sum of all flows through the *inlet* and *outlet* boundary, a value of 0 is required to satisfy the numerical conservation of mass and/or check if any fluid leaves the domain via the *atmospheric* boundary. The volume flows have been normalised to the channel width.

**Table 6.1** Summary of the properties of the difference cases.

property	<i>reference</i>	<i>same mesh</i>	<i>fine mesh</i>	<i>coarse mesh</i>
turbulence model	$k - \omega$ SST	$k - \varepsilon$	$k - \varepsilon$	$k - \varepsilon$
$y^+$	$2.400 \pm 0.009$	$2.90 \pm 0.05$	$14.0 \pm 0.5$	$40.00 \pm 0.15$
cells in fluid layer	14–15	15–17	4–5	2–3
fluid velocity, $\text{m s}^{-1}$	0.49	0.37	–	0.67
water balance	OK	OK	effluent through <i>top</i>	OK, oscillating

## 6.4. General Simulation Setup

The simulations for each unit described above were conducted separately. The next sections will provide the detailed setup of boundaries, transport properties, and solution settings that have been used.

### 6.4.1. Boundary Conditions

The BCs used in the simulations for the different reactor units are listed in Tables 6.2 (inlet), 6.3 (channel), and 6.4 (flow reversal tank). Their main difference is the choice of `inlet` BC and turbulence wall computation.

Some values of the BC shown contain special data as described below:

- estimate* An estimate for the turbulence variables based on the equations introduced in Section 4.5.1.
- internal* The value used for this boundary condition is derived from the initial value of the internal field.
- setPoint* A set point that is defined to match the desired volume flow, usually either by setting the volume fraction of water at the inlet or by defining the flow velocity at fixed water fraction.
- manning* The value has been calculated by empirical correlations (Manning equation) to obtain a good initial guess.
- timed* This is a special case, where the value of a relative tracer concentration  $y$  at the inlet is switched based on the simulation time:

$$\begin{aligned} t < 5 \text{ s} & \quad y_{NaCl} = 0 \\ 5 \text{ s} \leq t < 20 \text{ s} & \quad y_{NaCl} = 1 \end{aligned}$$

### 6.4.2. Transport Properties

The transport properties for the simulation can be found in Table 6.5. For all simulations a gravity of  $g = 9.81 \text{ m s}^{-2}$  was assumed in vertical direction.

### 6.4.3. Interpolation Schemes, Discretisation Schemes, and Solution Algorithm Options

The chosen interpolation and discretisation schemes for the simulations are listed in Table 6.6. It is followed by Table 6.7, which describes the primary settings of solution algorithms and convergence requirements for the different fields. Both lists have been adapted from the tutorial case `multiphase/interFoam/ras/waterChannel` describing a similar free surface flow scenario.



**Table 6.2** BC and initial conditions used in the simulations of the inlet unit. The *setPoint* in these simulations defines the inlet velocity of the fluid based on the inlet area to obtain the requested volume flow.

Field	Boundary				
	<i>internal</i>	inlet	outlet	atmosphere	wall/baffle
$\alpha$	0	fixedValue value 1	inletOutlet inletValue 0	inletOutlet inletValue 0	zeroGradient
$\varepsilon$	$6.41 \cdot 10^{-3}$	groovyBC value <i>estimate</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	epsilonWall- Function
$\omega$	14.7	groovyBC value <i>estimate</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	omegaWall- Function
$y_{NaCl}$	0	groovyBC value <i>timed</i>	zeroGradient	inletOutlet inletValue 0	zeroGradient
$k$	$2.65 \cdot 10^{-3}$	groovyBC value <i>estimate</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	kqRWall- Function
$p_{pgh}$	0	fixedFlux- Pressure	fixedFlux- Pressure	total- Pressure value 0	fixedFlux- Pressure
$u$	(0 0 0)	groovyBC value <i>setPoint</i>	inletOutlet inletValue (0 0 0)	pressure- InletOutlet- Velocity	fixedValue value (0 0 0)
$\nu_t$	0	calculated	calculated	calculated	nutkWall- Function

**Table 6.3** BC and initial conditions used in the simulations of the channel unit. In these simulations the *setPoint* is defined by calculating the water volume fraction that enters the channel over the inlet area for the estimated velocity to give the required volume flow.

Field	Boundary				
	<i>internal</i>	inlet	outlet	atmosphere	wall/baffle
$\alpha$	0	groovyBC value <i>setPoint</i>	zeroGradient	inletOutlet inletValue 0	zeroGradient
$\varepsilon$	$2.02 \cdot 10^{-5}$	groovyBC value <i>estimate</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	epsilonLowRe- WallFunction
$\omega$	0.259	groovyBC value <i>estimate</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	omegaWall- Function
$k$	$4.75 \cdot 10^{-4}$	groovyBC value <i>estimate</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	kLowReWall- Function
$p_{pgh}$	0	fixedFlux- Pressure	fixedFlux- Pressure	totalPressure value 0	fixedFlux- Pressure
$u$	(0 0 0)	groovyBC value <i>manning</i>	inletOutlet inletValue (0 0 0)	pressure- InletOutlet- Velocity	fixedValue value (0 0 0)
$\nu_t$	0	calculated	calculated	calculated	nutUSpalding- WallFunction

**Table 6.4** BC and initial conditions used in the simulations of the flow reversal unit. Here the *setPoint* defines the water volume fraction for an assumed water velocity of  $0.6 \text{ m s}^{-1}$  at a 1–2 cm high strip of the inlet.

Field	Boundary				
	<i>internal</i>	inlet	outlet	atmosphere	wall/baffle
$\alpha$	0	groovyBC value <i>setPoint</i>	zeroGradient	inletOutlet inletValue 0	zeroGradient
$\varepsilon$	$3.86 \cdot 10^{-6}$	fixedValue value <i>internal</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	epsilonWall- Function
$\omega$	0.122	fixedValue value <i>internal</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	omegaWall- Function
$k$	$3.51 \cdot 10^{-4}$	fixedValue value <i>internal</i>	inletOutlet inletValue <i>internal</i>	inletOutlet inletValue <i>internal</i>	kqRWall- Function
$p_{\rho gh}$	0	fixedFlux- Pressure	fixedFlux- Pressure	totalPressure value 0	fixedFlux- Pressure
$\mathbf{u}$	(0 0 0)	groovyBC value (0 0.6 0)	inletOutlet inletValue (0 0 0)	pressure- InletOutlet- Velocity	fixedValue value (0 0 0)
$\nu_t$	0	calculated	calculated	calculated	nutkWall- Function

**Table 6.5** Default transport properties used in the multiphase simulation. The values correspond to pure water and air at  $20^\circ\text{C}$ . Data is stored in the `constant/transportProperties` dictionary.

Property	Symbol	Unit	Water	Air
Transport model			Newtonian	
Viscosity	$\nu$	$\text{m}^2 \text{ s}^{-1}$	$1 \cdot 10^{-6}$	$1.48 \cdot 10^{-5}$
Density	$\rho$	$\text{kg m}^{-3}$	1000	1
Surface tension	$\sigma$	$\text{kg s}^{-2}$	0.07	

**Table 6.6** Solution schemes defined in the `system/fvSchemes` dictionary (compare Section 4.3.1 and 4.3.2).

Operation	Solution Scheme	Notes
<code>ddt</code>	Euler	Implicit Euler
<code>grad</code>	Gauss linear	–
<code>div(rhoPhi,U)</code>	Gauss linearUpwind grad(U)	–
<code>div(phi,alpha)</code>	Gauss vanLeer	Transport of volume fraction indicator, bounded scheme
<code>div(phirb,alpha)</code>	Gauss linear	Compression velocity [Rusche 2002], bounded by MULES algorithm [Greenshields 2017b]
<code>div(phi,k)</code>	Gauss upwind	Turbulence, bounded
<code>div(phi,epsilon)</code>	Gauss upwind	Turbulence, bounded
<code>div(phi,omega)</code>	Gauss upwind	Turbulence, bounded
<code>div(phi,Y)</code>	Gauss linearUpwind grad(U)	Tracer concentration ( $y_{NaCl}$ ), bounded
<code>div((muEff*dev(T(grad(U)))))</code>	Gauss linear	–
<code>laplacian</code>	Gauss linear corrected	–
<code>interpolation</code>	linear	–
<code>snGrad</code>	corrected	Surface normal gradients, non-orthogonal correction

**Table 6.7** Solution algorithms defined in the `system/fvSolution` dictionary. More information can be found in Greenshields [2017c].

Field	solver (smoother/preconditioner)	tolerance	relTol
$\alpha$	smoothSolver (symGaussSeidel)	$1 \cdot 10^{-8}$	0
$p_{\rho gh}$	GAMG <sup>1</sup> (GaussSeidel)	$5 \cdot 10^{-9}$	0
$p'$	PCG <sup>2</sup> (GAMG using GaussSeidel)	$1 \cdot 10^{-5}$	0
$y$	smoothSolver (symGaussSeidel)	$1 \cdot 10^{-8}$	0
$\mathbf{u}$	smoothSolver (symGaussSeidel)	$1 \cdot 10^{-6}$	0.1
$k, \varepsilon, \omega$	smoothSolver (symGaussSeidel)	$1 \cdot 10^{-6}$	0.1

<sup>1</sup> Geometric-Algebraic Multi-Grid (GAMG) algorithm using a smoother

<sup>2</sup> Preconditioned Conjugate Gradient (PCG) algorithm using preconditioner

#### 6.4.4. Runtime Settings

The runtime settings control the simulated time and the written outputs. The relevant options chosen can be found in Table 6.8.

**Table 6.8** Simulation runtime options set in the `system/controlDict` dictionary.

Property	Setting	Description
<code>startTime</code>	0	Starting at 0 s, continuation using <code>startFrom latestTime</code> setting. Some tools (exec-FlowFunctionObjects) require <code>startFrom startTime</code> .
<code>endTime</code>	10-30	Depending on the simulation until a quasi steady-state has been achieved.
<code>adjustTimeStep</code>	yes	Initial time step 0.1 s, adjusted by $Co$ number, with the upper limit <code>maxDeltaT 1</code> .
<code>maxCo</code>	6	Maximum $Co$ allowed
<code>maxAlphaCo</code>	6	Maximum $Co$ allowed for the surface transport indicator
<code>writeInterval</code>	0.05-1	Modified as required for the post-processing analyses.

## 7. Measurement and Validation Techniques<sup>1</sup>

This chapter will describe the methods used to analyse the flow properties in the reactor. In the first section, a novel sensor for the rapid measurement of salinity in the reactor will be introduced, which can be used to obtain detailed information on the flow properties. Afterwards, the measurement and (for comparison) simulative extraction of the fluid height and velocity will be described, followed by an empirical, theoretical approach to generate this data as a straightforward validation technique. The last section will define a measure for the mixing in the channel and outline the data export and evaluation methods.

### 7.1. Rapid Salinity Measurement

For the hydrodynamic characterisation of reactors handling aqueous liquids, especially in conjunction with the validation of numerical simulations, data with high temporal resolution is essential. The measurement of the residence time is a simple parameter to characterise the reactor's hydrodynamic distribution. This can be accomplished by applying a very short impulse (approaching the ideal of a dirac delta function) or by conducting step change experiments in which a tracer concentration at the inlet changes abruptly for the remaining measurement. The resulting concentration change is then monitored at the outlet to calculate the residence time distribution. These measurements are effortless, as only in- and outgoing streams are determined [Baerns 2006]. This information can then be used to validate reactor models based on Computational Fluid Dynamics (CFD) simulations or for simple reactor characterisation.

The determination of the flow distribution demands the addition of a soluble and nonreactive tracer into the fluid, which does not influence its physical properties and does not interact with the surfaces [Fogler 2016]. Moreover, the applied sensor has to be minimal invasive for the fluid flow as well as fast enough to capture all characteristics. Moreover, the applied sensor has to be minimal invasive for the fluid flow as well as fast enough to capture all characteristics. Commonly used tracers include dyes for optical or fluorescent measurements or NaCl solutions for conductivity measurements [Giannelli et al. 2015; Jimenez et al. 1988]. Although optical measurements can be arbitrarily fast, the analysis of the fluid's conductivity has the advantage that it does not require an additional light source. It can thus easily be used for a three dimensional evaluation without additional, flow diverting obstacles.

Commercial conductivity sensors are designed for the measurement of impurities or determination of salt concentration. However, these sensors do not meet the demands for a reactor, as their measurement rate is too low and their dimensions affect the fluid flow (rate  $\approx 1 \text{ s}^{-1}$ , diameter  $\approx 1 \text{ cm}$ ). Contrary, for the characterisation of the hydrodynamics of reactors, no absolute values are necessary, and relative measurements suffice.

---

<sup>1</sup> Parts of this chapter have already been published in: Timm Steffen Severin, Sebastian Plamauer, Andreas Christoph Apel, Thomas Brück and Dirk Weuster-Botz [2017]. 'Rapid salinity measurements for fluid flow characterisation using minimal invasive sensors'. In: *Chemical Engineering Science* 166 [2017], pp. 161–167. ISSN: 00092509. DOI: 10.1016/j.ces.2017.03.014

Following these specifications, the next sections describe the design and construction of a rapid, low-cost, and minimal invasive conductivity sensor array that can be used to track the movement of a change salinity change through the reactor, thus allowing to extract detailed information on the reactor's performance as well as providing data for the validation of CFD models.

### 7.1.1. Working Principle

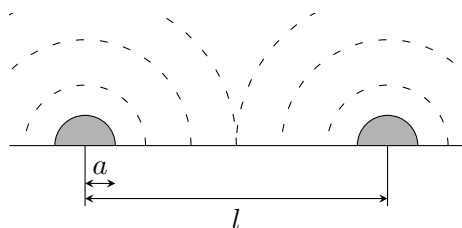
The basic principle of conductivity measurement in a solution is the measurement of the resistance  $R$  between two electrodes (two-electrode-cell) [Cummings et al. 2003]. Combined with a geometry-dependent cell constant  $K$ , the specific conductivity  $\kappa$  of a solution can be written as

$$\kappa = K \cdot \frac{1}{R} = K \cdot G \quad (7.1)$$

where  $G$ , the conductance, is the inverse of the resistance.

As an approximation for the cell constant  $K$  Hyldgård et al. [2008] showed that for a flat system (compare Figure 7.1) the approximation in Equation (7.2) can be used.

$$K = \frac{1}{\pi a} \frac{l - 2a}{l - a} \quad (7.2)$$



**Figure 7.1** Schematic drawing of a sensor spot cross section and the dimensions. The electrodes (⊙) are assumed to be infinitely long half cylinders with constant radius  $a$  and distance  $l$ . Image based on Hyldgård et al. [2008].

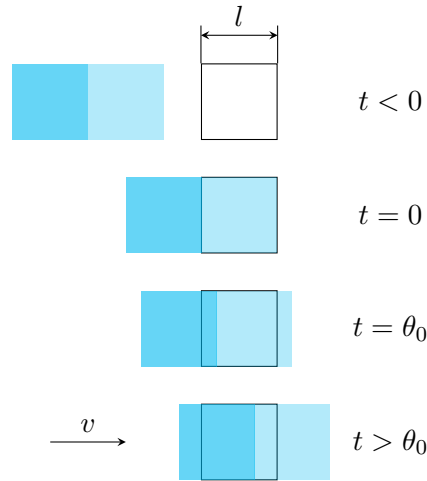
Apart from the cell geometry, the measured conductivity is influenced by different effects, which have to be considered for the sensor design and application:

**Temperature** With rising temperature, the salt ions in the fluid become more mobile, thus increasing the conductivity. This can easily be avoided by having a temperature controlled environment for the measurement.

**Polarisation** Using Direct Current (DC) the electrodes will polarise the fluid and increase the measured resistance. To avoid this, Alternating Current (AC) can be used. A further modification would be the use of a four-electrode-cell, in which the current is applied in two outer electrodes, while the resistance is measured in an inner, non-polarised region.

**Concentration** While the effect of the concentration on the conductivity is desired for this measurement, a linear dependence can only be expected in dilute solutions [Gray 2005].

Another aspect to be considered is the measurement frequency. Figure 7.2 shows how a control volume moves over the sensor spot. While the sensor measures for a time  $\theta_0$ , the flow moves the control volume further over the sensor spot.



**Figure 7.2** The one dimensional movement of two control volumes (■, ■) over a sensor spot (□). With the start of the measurement ( $t = 0$ ) the control volume that should be measured (■) completely covers the sensor spot. After the duration of the measurement  $\theta_0$  the control volumes have moved, and the overlapping area of the second control volume (■) has influenced the resulting conductivity.

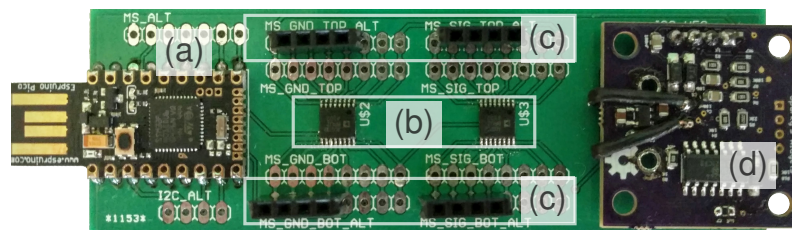
Treating the fluid velocity  $v$  as infinitesimal displacement (Equation (7.3)) one can define two dimensionless factors  $n$  and  $k$  that correlate to geometric and temporal constraints (Equation (7.4)).

$$v = \frac{ds}{dt} \quad (7.3)$$

$$n = \frac{l}{ds}, k = \frac{\theta_0}{dt} \Rightarrow v = \frac{k}{n} \cdot \frac{l}{\theta_0} \quad (7.4)$$

The ratio  $k/n$  then defines the time it takes for a control volume to move over the sensor to the time the measurement takes and is thus a measure for the spacial resolution of one sensor spot. It can therefore be used for the design process giving upper limits for the temporal resolution of the sensor for a desired spatial resolution.

### 7.1.2. Sensor Design



**Figure 7.3** The complete board: Marked are the microcontroller with USB connector (a), the matrix switches (b), the connectors for the sensor arrays (c), and the MinieC eC chip (d).

The design of the device can be seen in Figure 7.3. A simple carrier board, designed to accommodate the chips and manufactured by Where Labs, LLC (for schematics see Appendix F), has been used as basis. The measurement is controlled by an Espruino Pico

microcontroller with USB interface, Pur3 Ltd. The microchip has been programmed using the MicroPython support of the Espruino Pico. The measurement is performed using a MinieC eC Interface, Sparky's Widgets, which produces an AC voltage signal, measures the resistance of the liquid and uses an Analogue to Digital Converter (ADC) to communicate the signal to the microcontroller as a value between 0 and 4096 (inversely proportional to the voltage drop over the fluid between 0–3.3 V). To increase the measurement speed the filter capacitor placed before the ADC has been removed. In order to control multiple sensors, two Matrix Switches ADG738, Analog Devices are used to toggle between the connectors.



**Figure 7.4** The sensor strip array used for the measurement. The four sensor spots are placed every 5 cm. The distance between the electrodes of one sensor spot is 1 cm.

The sensor array strip (Figure 7.4) has been custom made with a dimension of 220 mm × 25 mm on one layer of flexible Printed Circuit Board (PCB), Leiton GmbH (Electroless Nickel Immersion Gold (ENIG) surface, 65  $\mu\text{m}$  adhesive-based polyimide, 28  $\mu\text{m}$  copper thickness, 26  $\mu\text{m}$  cover-lay yellow solder mask, drawing see Supplemental). Each sensor spot is a two-electrode-cell on which AC with a frequency of 1666 Hz is applied. The distance between the electrodes on the sensor is 1 cm, which results in a cell constant  $K \approx 6.0 \text{ cm}^{-1}$  (Equation (7.2)), thus lying in a measurement range of approximately 10–20 000  $\mu\text{S cm}^{-1}$  [Boyes 2003] or a concentration of 0–1 % (w/w) NaCl [Haynes and Lide 2010].

The connection of the sensor array stripes and the carrier board is achieved via 70 cm 4-pin Dupont cables with female connectors. The sensor stripes themselves were fixed to the reactor ground using water resistant, adhesive tape.

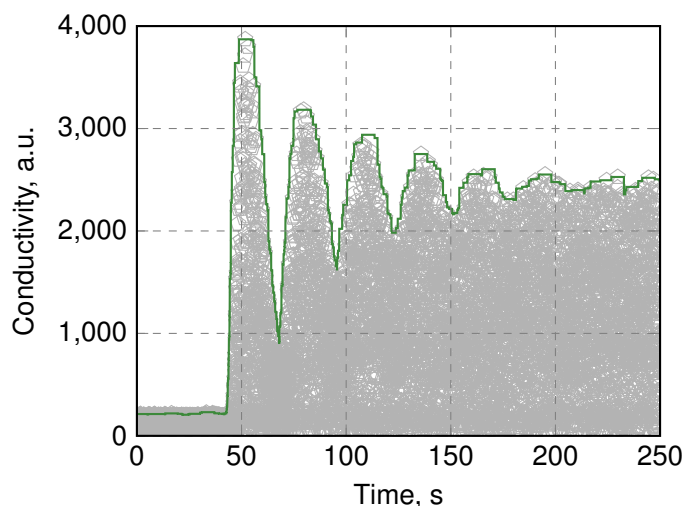
### 7.1.3. Evaluation Method

The raw data read from the microprocessor was evaluated using MATLAB. In a first reduction step, the raw data was filtered: Values exceeding a given threshold of 4096 were discarded as outliers (they exceed the theoretical maximum of the device), while measurements close to zero were discarded to decrease the amount of data. The remaining data, due to the removal of the capacitor on the sensor board, scatters over the complete range between 0 and the actual value due to the overlaying effect of the device's intrinsic AC frequency. Using a sliding maximum algorithm a hull curve was extracted from the data (Figure 7.5, see Supplemental for algorithm details).

### 7.1.4. Sensor Performance Effect on Fluid Flow

With a thickness of 125  $\mu\text{m}$  the sensor array has less than 4.5 % of the fluid layer thickness. During the experiments, no difference on the surface flow over the sensor could be seen when compared to the surrounding region.





**Figure 7.5** The filtered raw data ( $\diamond$ ) and the generated hull curve (—) for an exemplary data set (data reduced to only show every second point for visualisation).

### Sensor Speed

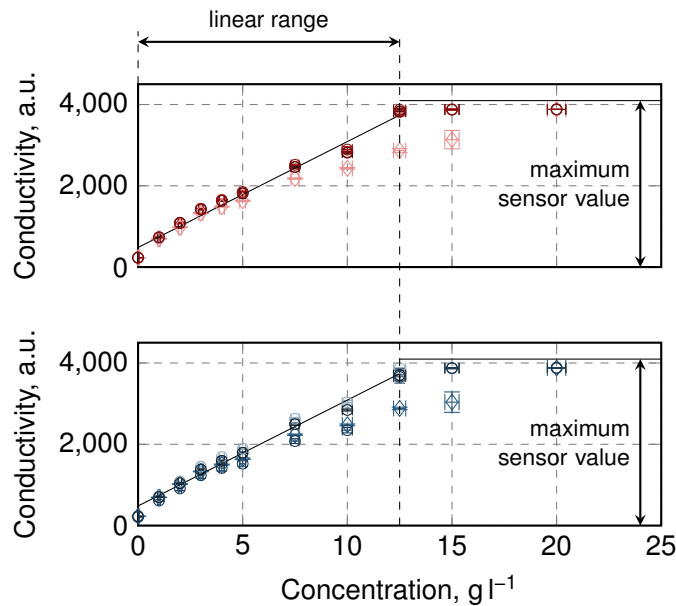
Evaluating the raw measurement data gives an idea of the time requirement for one measurement. The average time difference between the data entries of two consecutive sensors is  $722.6 \pm 9.6 \mu\text{s}$ . This includes a time of  $520 \mu\text{s}$  to switch from one electrode pair to the next. Thus, one measurement takes  $202.6 \mu\text{s}$ . With an estimated velocity of  $v \approx 1 \text{ m s}^{-1}$  this gives us a ratio  $k/n = 0.02$  (Equation (7.4)), i.e. the measured volume is 2% larger than the sensor, which provides a very high temporal resolution. However, due to the duration of the write process, the average time between the data points for one sensor is  $6.78 \pm 0.61 \text{ ms}$ . Although this results in a measurement frequency of 147 Hz per sensor, the current evaluation method of using the hull curve further reduces the data density.

### Sensor Accuracy

To evaluate the accuracy of the sensors, calibration measurements were performed. The sensor arrays were submerged in 500 ml glass cylinders containing tap water ( $c_{\text{NaCl}} \approx 0 \text{ g l}^{-1}$ ) enriched with defined amounts of NaCl. The results of these measurements can be seen in Figure 7.6.

The results indicate, that at least three sensors on each array behave linear in a range between  $0\text{--}12.5 \text{ g l}^{-1}$  with a coefficient of determination of  $R^2 \approx 0.98$ . This is in accordance with the estimation of the cell constant  $K$  (compare Section 7.1.2), which predicted a valid linear measurement range of  $0\text{--}10 \text{ g l}^{-1}$ . While visible, the minor trend of a saturation curve is negligible for our evaluation.

Sensors #2 ( $\diamond$ ) and #5 ( $\diamond$ ) however deviate strongly from the linear behaviour of the other sensor due to a defective wire as revealed by post-experimental unit evaluation. Nonetheless, relative measurements, in which no absolute information is required, are still possible within the given range.



**Figure 7.6** Calibration measurements for the sensors at different concentrations: Upper graph sensor 1–4 (○, ◇, ◻, ◻), lower graph sensor 5–8 (◇, ◻, ◻, ◻). Conductivity errors are standard deviations of the measurement over time (< 2%), concentration errors have been estimated by error propagation from the precision of the balance and the beaker. Linear range determined by regression fit with  $R^2 \approx 0.98$ .

### 7.1.5. Comparison with the CFD Simulation

In order to generate comparison data, the extended solver implementing the passive species transport (Section 6.2.2) was used. The Boundary Conditions (BCs) of the solution were defined to reflect the experimental conditions. Listing 7.1 shows the `sampleDict` entry which is used to obtain the simulated concentrations at the position of the sensor spots. Since the sensor spots have a width of 1 cm, a custom sampling library is used (`cloudSphere`, see Appendix E.2). This library reads a set of coordinates and a radius, and extracts the data of all cells which centre point is inside a sphere at the given point and radius (here 0.005). In a post-processing step, this data could then be averaged to generate a mean predicted value as well as a standard deviation.

Since the simulated concentration is modelled to be bound ( $0 < y < 1$ ), the experimentally measured data is normalised: Using a baseline (usually the first second of the undisturbed measurement), the initial conditions are defined as  $y = 0$  a.u.. The maximum value, or — depending on the measurement — the average of the stationary data is set to  $y = 1$  a.u. as upper bound.

## 7.2. Fluid Height

### 7.2.1. Experimental

The fluid height has been measured using a very thin (less than 0.5 mm) metal ruler. It was placed flatly on the ground along the fluid direction (compare Figure 7.7). The height was measured at the upstream facing side of the ruler, to reduce the influence of the added obstacle

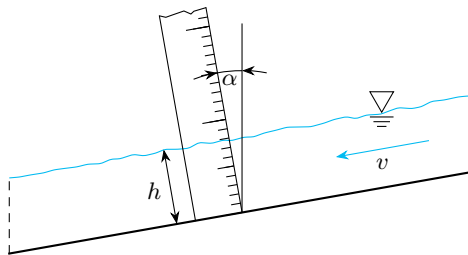
**Listing 7.1** The sampleDict entry to export the extract the simulated concentration profile at the sensor position.

```

1   measuredPointsStd
2   {
3       type          cloudSphere;
4       axis          xyz;
5       radius        0.005;
6
7       points
8       (
9           // inner sensor array (in inlet tank), 1-4
10          (-0.236 0.04 0.103)
11          (-0.1907 0.04 0.082)
12          (-0.1453 0.04 0.061)
13          (-0.1 0.04 0.04)
14          // outer sensor array (on weir), 5-8
15          (-0.25 0.122 0.14)
16          (-0.2 0.122 0.14)
17          (-0.15 0.122 0.14)
18          (-0.1 0.122 0.14)
19      );
20  }

```

on the fluid's surface.



**Figure 7.7** Visualisation of the fluid height measurement: A thin metal ruler is inserted perpendicular to the channel ground into the fluid. The fluid height is measured at the upper edge of the ruler to minimise the influence the thickness of the metal has on the flow.

The fluid height determined that way was thus perpendicular to the channel ground. Assuming a quasi-steady state flow, i.e. no change in the fluid height, the vertical height of the fluid layer can be calculated by relation 7.5.

$$h' = h \cdot \cos(\alpha) \quad (7.5)$$

Unfortunately, in many literature sources it is not clearly stated which height is used. However, for the channel the differences in height can be neglected, since the error induced by this measurement is lower than 0.4% for angles up to 5°.

### 7.2.2. Simulative

In the simulation the fluid height was determined by exporting the water volume field  $\alpha$  and post-processing it. This could be done with a cut through the geometry, as defined in Listing 7.2.

**Listing 7.2** The `sampleDict` used to extract the data of the water phase fraction in a plane (here for the channel) perpendicular to the ground. The `basePoint` is fixed on the ground at the exact position of the measurement, while the `normalVector` is adjusted for the slope.

---

```

1 surfaces
2 (
3     measurePlane
4     {
5         // use triangulated plane
6         type          plane;
7
8         // select basePoint on channel ground
9         // and normal vector perpendicular to it
10        basePoint     (0.0 1.15 -0.02);
11        normalVector  (0.0 0.9998 -0.01745);
12    }
13 );

```

---

**Listing 7.3** The `sampleDict` used to extract the water phase fraction at one single point in the geometry.

---

```

1 sets
2 (
3     steadyEnd
4     {
5         // export cell centre and face values for higher resolution
6         type          midPointAndFace;
7         axis          xyz;
8
9         // for simply things ignore slope of the channel, use only
10        // the z coordinate. Select points far below and above the mesh
11        // and let OpenFOAM select the correct coordinates inside the mesh
12        start          (0 1.0 -0.05);
13        end             (0 1.0 0.05);
14    }
15 );

```

---

Using a plane to sample the water height gives the water distribution over the complete width of the selected geometry. If less data is required, it can also be determined by using a straight line, as shown in Listing 7.3.

The extracted data was then post-processed using MATLAB. The water height was determined by integrating the water phase fraction over the height coordinate  $z \cos(\alpha)$  (measurements perpendicular to the ground) according to Equation (7.6).

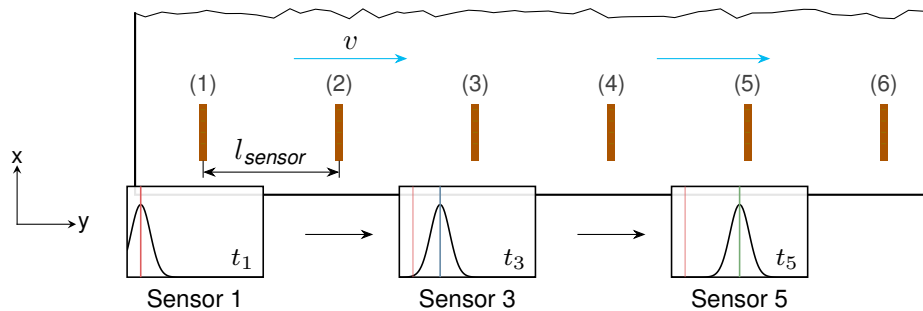
$$h = \int_{z_{min}}^{z_{max}} \alpha \, dz \quad (7.6)$$

The integration was done using a trapezoidal numerical integration algorithm (`trapz`). For planar data, this was interpolated on a regular grid using the MATLAB function `meshgrid` and `griddata` (script added in attachment as part of the `sampleToolbox`). If data for multiple time steps was available, it was used for averaging the fluid height over time.

## 7.3. Fluid Velocity

### 7.3.1. Experimental

The fluid velocity on the channel was measured using the salinity sensors described above. Therefore six sensor arrays were placed along the channel, with a distance  $l_{sensor}$  each (compare Figure 7.8). An impulse of highly concentrated salt water was added shortly before sensor (1), its effect on the fluid's conductivity could then be measured at each sensor array.



**Figure 7.8** Fluid velocity measurement: A pulse with a high salt concentration is added to the fluid flow immediately before sensor array (1). Its occurrence can then be tracked at each sensor array (—) and the time passed  $t_i$  can be determined. From the known distance between the sensor arrays  $l_{sensor}$  the local velocity  $v$  can be calculated.

Measuring the time at which the peak concentration arrived at each sensor could thus be used to calculate the average fluid velocity between any two sensor arrays. The sensor arrays were positioned approximately every 60 cm with the first sensor being 30 cm away from the channel start. To find the peak, an automatic processing of the raw sensor data using MATLAB was implemented with the following steps

1. From data that was recorded before the salinity pulse was added a baseline was measured and evaluated for each sensor spot at the given flow velocity by calculating the average. This baseline was defined to represent a salinity of zero.
2. The data of the salinity pulse experiment for each sensor was read and reduced according to the procedure described above (Section 7.1). To create the hull curve, the sliding maximum algorithm was used with a window size of 2 s and a central window mode. Since the reading of the sensors was started sequentially, the stored time steps were corrected for the time difference between the begin of the writing procedure.
3. The approximate position of salinity impulses' peaks was detected using a modified version of the `findpeaks` algorithm included in MATLAB: Peaks were assumed to be at least 5 s apart, while the threshold above zero for the detection of a peak was iteratively increased, until only a defined number of  $N$  peaks was left,  $N$  representing the number of times a salinity pulse was added to the system during one experiment.
4. Separately for each peak a standard distribution curve corresponding to Equation (7.7) was fitted, in which  $t_{peak}$  represents the position of the maximum,  $y_{amp}$  is the height of

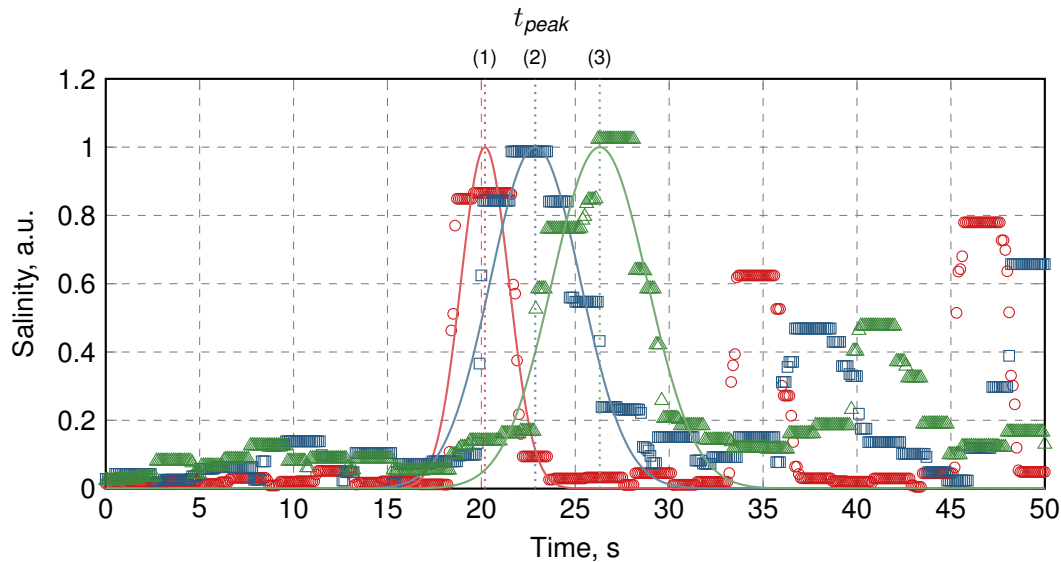
the peak (amplitude) and  $\sigma$  is a measure for the width of the peak.

$$y = y_{amp} \cdot e^{-\frac{(t-t_{peak})^2}{2 \cdot \sigma^2}} \quad (7.7)$$

The data and the resulting fit for the first peak in one set of data can exemplarily be seen in Figure 7.9.

- From this information, using the distance between each sensor, the velocity of the fluid between two sensor  $i$  and  $j$  could be calculated according to Equation (7.8).

$$v_{ij} = \frac{l_{ij}}{t_{peak,j} - t_{peak,i}} \quad (7.8)$$



**Figure 7.9** Measurement of the fluid velocity on the channel by tracking salinity pulses. Shown is the hull curve data (markers) and fits (solid lines) at position #1 ( $\circ$ ), #3 ( $\square$ ), and #5 ( $\triangle$ ). The data points also include further peaks (e.g. between  $t = 35\text{--}42$  s). These are due to further pulses that were neglected for evaluation in this example. Experimental setup: Volume flow  $\dot{V} = 1.6 \text{ l s}^{-1}$ ,  $\alpha = 1^\circ$ .

### 7.3.2. Simulative

The fluid velocity was exported from the simulation using the same sample commands as for the export of the water volume fraction in for the determination of the water height (Section 7.2.2).

## 7.4. Empirical Correlation for Open Channels

For the validation of this channel geometry, the water height and fluid velocity in the channel could be measured as described above. Furthermore, theoretical predictions were used as a more versatile channel validation method: To describe the flow in open channels, Manning

et al. [1890] (re-)developed an empirical equation for open channels (Equation (7.9)). It gives an estimate for the water velocity in a channel with given hydraulic radius  $R_h$  and slope  $S$  (in %). The surface of the channel is incorporated in the Gauckler-Manning coefficient  $n$  (for a list of coefficients see Oswald [1988]).

$$v = \frac{1}{n} \cdot R_h^{\frac{2}{3}} S^{\frac{1}{2}} \quad (7.9)$$

The hydraulic radius is defined as the cross section of the flow  $A$  over the wetted perimeter  $P$ , resulting in Equation (7.10) for shallow channel flows which are much wider than high.

$$R_h = \frac{A}{P} = \frac{h \cdot b}{b + 2h} \stackrel{b \gg h}{\approx} h \quad (7.10)$$

Based on Equation (7.9), the water height  $h$  on the channel, as an easier measurable quantity, can be derived from the volume flow  $\dot{V}$  and the channel width  $b$  (7.11):

$$h = \frac{\dot{V}}{v \cdot b} \stackrel{Eq.(7.9)}{=} \left( \frac{n \dot{V}}{S^{\frac{1}{2}} b} \right)^{\frac{3}{5}} \quad (7.11)$$

## 7.5. Vertical Mixing<sup>2</sup>

The vertical mixing on the channel is a major influence on the cultivation, since it is essential for the availability of light to the microalgal cells. The following sections will introduce the method used to determine the mixing properties.

### 7.5.1. Reynolds Approach

Commonly, mixing efficiencies are described using the Reynolds number of the flow, assuming a turbulent regime provides sufficient mixing. However, in literature no common ground can be found for the definition of the critical Reynolds number  $Re_{crit}$  in shallow channels: While Savat [1977] mention a transition area of  $Re = 250-1000$ , French [1987] and Masojidek, Kopecky et al. [2011] consider  $Re_{crit} = 12500$ , with the most popular choice being around  $Re_{crit} = 2000$  [Buhr and Miller 1983; Te Chow 1959]. Yet, with the definition of the Reynolds number (Equation (7.12), velocity  $v$ , hydraulic radius  $R_h$ , kinematic viscosity  $\nu$ ), () (typically operated at velocities around  $0.5 \text{ m s}^{-1}$  and layer thickness of 5 mm) have a Reynolds number of 2500. This cannot with certainty considered to be fully turbulent, but is in the transition or low turbulent regime.

$$Re = \frac{v \cdot R_h}{\nu} \quad (7.12)$$

Furthermore, the Reynolds number employs the overall flow velocity, while for the light exposure only vertical mixing is relevant. This was already used by Pruvost, Pottier et al.

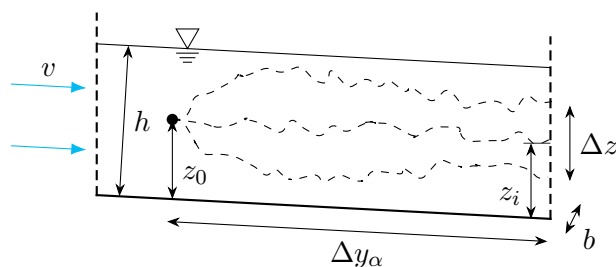
<sup>2</sup> Parts of this chapter have already been published in: Timm Steffen Severin, Andreas Apel, Thomas Brück and Dirk Weuster-Botz [2018]. 'Investigation of Vertical Mixing in Thin-Layer Cascade Reactors using Computational Fluid Dynamics'. In: *Chemical Engineering Research and Design* [2018]. ISSN: 02638762. DOI: 10.1016/j.cherd.2018.01.036

[2006] for their closed torus photobioreactor, considering only the fluid velocity in direction of the light as a function of mean (bulk) velocity. In another work, they use CFD particle trajectories to visualise a radial displacement of algal cells as an indicator for improved mixing [Pruvost, Legrand et al. 2002]. For open raceway ponds Prussi et al. [2014] used a similar approach, in which the vertical displacement of a CFD particle injected at defined height is evaluated after a bend.

The drawbacks of these solutions are that they are only valid for the respective system, and variations in operational parameters must not influence e.g. water height or curvature to allow a comparison of the results, since these changes would also influence the evaluated variable independent of the mixing effects. This prevents the usage of this method for thin-layer cascade reactors, where the water height and velocity are operational and geometric parameters that can be modified in order to improve the efficiency of the reactor.

In the following section a more general analysis of vertical mixing will be described, in which the conditions present in the reactors are reduced to one characteristic, dimensionless number that can be used to compare different systems.

### 7.5.2. Mixing Index



**Figure 7.10** Visualisation of the stream traces used for the mixing evaluation: A virtual particle is injected at a defined height  $z_0$ . From there it can travel different stream lines (dashed lines) depending on the momentary flow conditions. After a certain distance  $\Delta y_\alpha$  the span  $\Delta z$  of a number of these traces can be evaluated as the spread that the algal cell experience.

Imagining a number of particles starting at a height  $z_0$  above the ground in the channel, travelling a distance  $\Delta y_\alpha$  with the flow. After this distance, the particles will have spread in vertical direction over a range  $\Delta z$  in the water layer of height  $h$ . A simple dimensional analysis of these flow properties on the channel yields — among others — the following dimensionless numbers:

$$\Pi_{\Delta z} = \frac{\Delta z}{y_\alpha} \approx \frac{\Delta z}{v \cdot t} \quad (7.13)$$

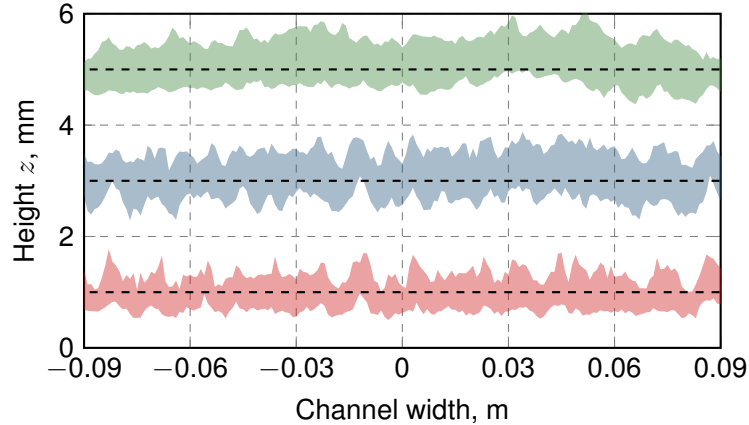
$$\Pi_h = \frac{\Delta z}{h} \quad (7.14)$$

Therein Equation (7.13) is a relation of the spread to the distance the particle travelled, while the second Equation (7.14) relates the spread achieved to the maximum available water height  $h$  and as such has a limit of  $0 \leq \Pi_h \leq 1$ . Based on these equations a few adaptations



can be made to create a simplified, universal mixing index:

1. For angles  $\alpha \leq 5^\circ$  the distance travelled  $\Delta y_\alpha$  can be approximated by the pure  $y$ -distance of start and end coordinate with an error of  $\varepsilon_y \leq 0.5\%$ .
2. So far the analysis only considered one slice along the length of the reactor. Due to having a 3D simulation, this data can be generated for multiple slices, replacing the spread  $\Delta z$  with an areal spread  $\Delta A$  (compare Figure 7.11).



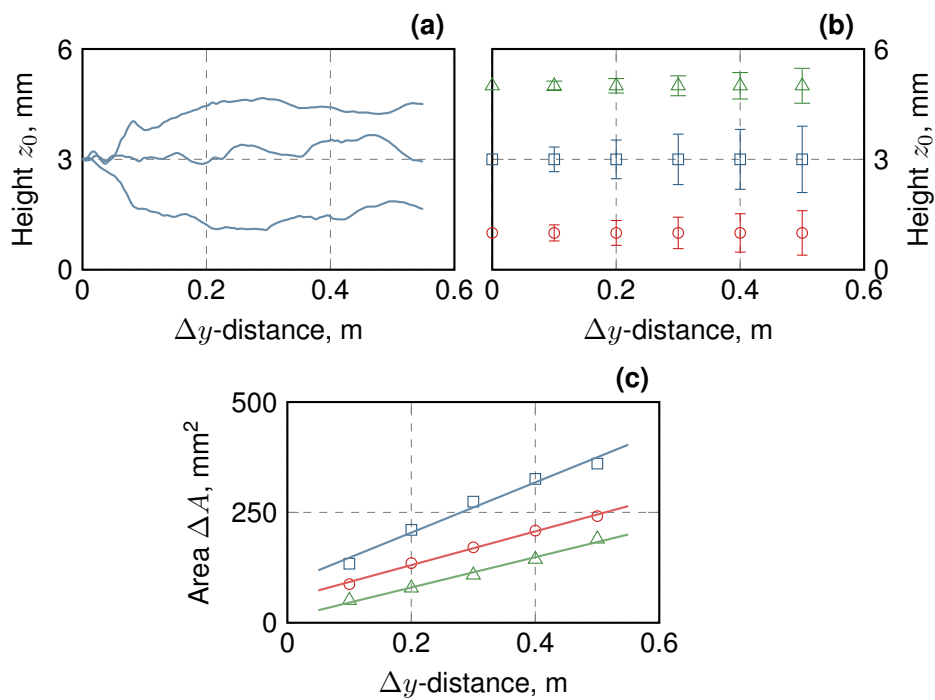
**Figure 7.11** Exemplary standard deviations  $\Delta z$  for three different starting heights  $z_0 = 1$  mm (red), 3 mm (blue), and 5 mm (green) (dashed lines) as used in the evaluation.

3.  $\Pi_{\Delta z}$  is the spread measured over distance travelled. This is essentially the slope of the function  $\Delta z = f(y)$ . Since we can export stream traces for different starting positions, we can use regression analysis to obtain a more robust  $\frac{d\Delta z}{dy}$  (Figure 7.12).
4. Both dimensionless numbers can be combined into one index, calculating the achieved spread while respecting of the limiting factor of the water height. This solves the above mentioned issue of comparability between channels with different layer thickness.

Including these adaptations in the dimensional analysis, we obtain the following Equation (7.15) for the mixing index  $MI$ , split in a mixing and a scaling term.

$$MI = \underbrace{\frac{1}{b} \frac{d\Delta A}{dy}}_{\text{mixing}} \cdot \underbrace{\frac{\Delta A}{b \cdot h}}_{\text{scale to } h} \quad (7.15)$$

Therein the mixing term directly describes the speed of the vertical mixing and has a range from zero to infinity (0 = completely laminar,  $\infty$  = instant mixing). It can also be interpreted as the tangent of the angle that defines the cone of end positions reachable by the (virtual) particle. The scaling term adjusts the mixing to the height of the water surface, to render different systems comparable. It effectively is the fraction of the channels cross section which can be reached by a particle starting at the original, initial height, and thus has a range from zero to one.



**Figure 7.12** Calculation of the areas for the reference case ( $\alpha = 1^\circ$ ,  $\dot{V} = 2.4 \text{ l s}^{-1}$ ,  $\mu = 1 \text{ mPa s}$ ): The course of three exemplary stream tracers with a start height of  $z_0 = 3 \text{ mm}$  are shown (a). Calculating the standard deviation of the computed traces' height at the measurement position, one can obtain the spread in the height, visualised in (b) for the start heights  $z_0 = 1 \text{ mm}$  (○),  $3 \text{ mm}$  (□), and  $5 \text{ mm}$  (△). This height can be converted to an area by considering multiple stream traces over the width a the channel. A linear fit ( $R^2 = 0.996$  [—],  $0.980$  [—],  $0.989$  [—]) can then be applied to the data (c).

### 7.5.3. Data Export

Post-processing of the simulation data was done using ParaView 4.4.0. The stream tracer filter was used to approximate particle traces. Figure 7.10 showed an example for the evaluation: Stream tracers were started using a line source at a defined height  $z_0$ . Evaluating the traces at different times resulted in a number of possible path, whose end height  $z_i$  can be measured after a travelled distance  $\Delta y_\alpha$ .

The export of this data was automated using a python script coupled with ParaView (script in OpenFOAM data export supplemental): The spread of the stream tracers was evaluated for all simulated time steps at 1.15 m (0.05 m before the end to avoid boundary effects) starting from five different positions every 10 cm between 0.65 and 1.05 m. The start heights  $z_0$  were evaluated between 1 and 5.5 mm every 0.5 mm. Data where the start height exceeded the water height  $h$  were discarded.

All following evaluations were done using MATLAB. To obtain the spread  $\Delta z$  of all particles for the given start height, the 25 and 75 % percentiles (`prctile` function) were calculated as time average for each tracer position  $i$  (Equation 7.16).

$$\Delta z_{25/75} = p_{75}(z_i) - p_{25}(z_i) \quad (7.16)$$

These values were converted to an areal spread  $\Delta A$  by integrating over the width  $b$  of the channel (Equation (7.17)).

$$\Delta A = \int_{-b/2}^{b/2} \Delta z_{25/75} db \quad (7.17)$$

With these data,  $MI$  were calculated according to Equation (7.15).

## 8. Reactor Simulations

### 8.1. Inlet Tank

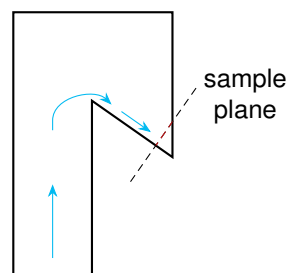
The first unit of the reactor is the inlet tank. Its primary purpose in the reactor is the distribution of the fluid over the whole width of the following channel. Therefore it must be designed in a way that gives a sufficient even fluid surface on its weir. In the following section the mesh independence for the used mesh will be shown. Next, the symmetry assumption for the inlet tank, splitting it in the centre in two halves, will be verified by a comparison of a complete geometry with only one half of it. Having evaluated the effects of the symmetry assumption, the simulation will be validated using experimental data. With the mesh's applicability confirmed, the influence of geometric and operational parameters on the resulting water surface as well as on the required pump capacity will be analysed. A list of the investigated parameters can be found in Table 8.1.

**Table 8.1** Parameters that were modified for the investigation of the inlet. The range has been defined by common operational parameters from literature. For the purpose of the analyses the volume flow has been normalised to the width of the inlet unit. Compare Table 5.2 for all of the reactor unit's parameters.

Parameter		Range
Weir height	$h_{weir}$	0.16–0.25 m
Width	$b$	0.83–6.00 m
Volume flow	$\dot{V}_b$	1.6–3.6 l s <sup>-1</sup> m <sup>-1</sup>

#### 8.1.1. Mesh Independence

The reference mesh used for the independence study is described in Section 6.1.1. For the comparison a volume flow of 2.0 l s<sup>-1</sup> at a weir width of 0.83 m was applied. To analyse the effect of the cell size on the results, different refinements of this mesh in all axis directions were run and evaluated. An overview over these cases can be seen in Table 8.2. All data for comparison were calculated as an average over the last 5 s of the simulation, where the simulation is in a quasi steady-state.



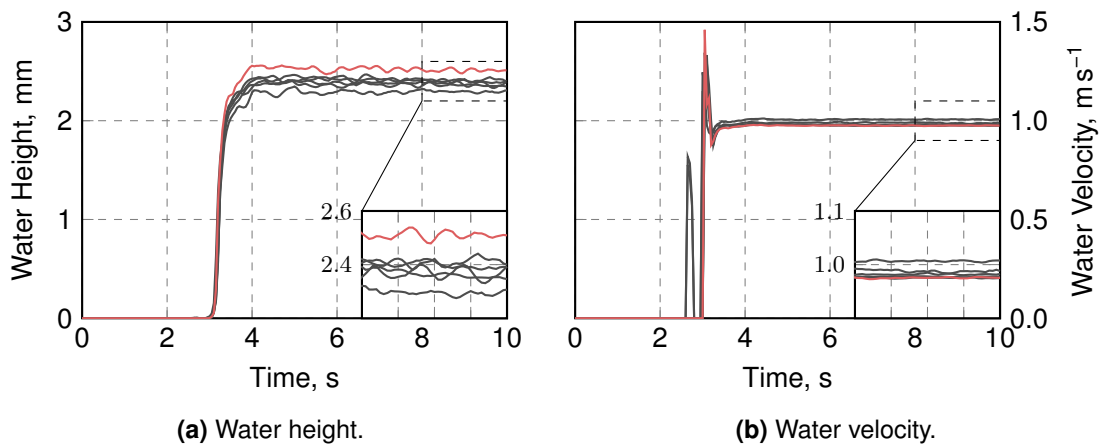
**Figure 8.1** Sample plane where water height and velocity on the weir were determined for all simulations shown.

**Table 8.2** The cases evaluated for the mesh independence study. For each case  $\langle factor \rangle \times$ , the basic mesh has been refined in all axis directions by  $factor$  compared to the chosen reference case ( $\ast$ ). Also shown is the average resolution of the basic mesh (before running snappyHexMesh) and the maximum cell volume in the final mesh.

Case	Cells $10^6$	Resolution mm	Max. Volume $\text{mm}^3$
0.25x	0.53	8.5	630
0.5x	1.04	6.7	310
1x $\ast$	2.01	5.3	162
2x	3.88	4.3	83
3x	5.90	3.7	55
4x	7.64	3.4	41

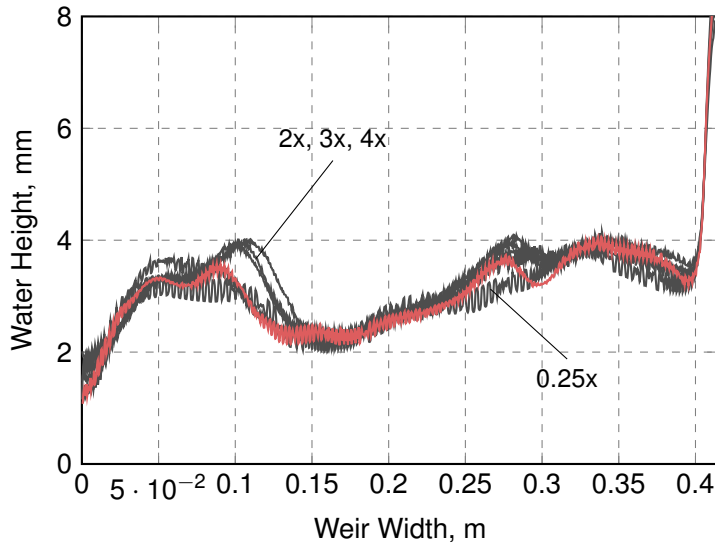
For all cases the water height on the weir, the water velocity and the water volume in the tank were evaluated at the sample plane shown in Figure 8.1. Additionally, a concentration of a tracer at eight different positions was compared.

Figure 8.2 shows the water height and velocity on the weir. Initially, the reactor unit is empty. After approximately 3 s the water reaches the weir height, and a few surface waves of water flow over it, resulting in the peaks that can be observed in Figure 8.2b. For all simulations the results are very close, and only small differences can be seen in the detail. Calculating the maximum error relative to the reference case, one obtains a deviation of less than 10 % for the water height and approximately 3 % for the velocity (Table 8.2).

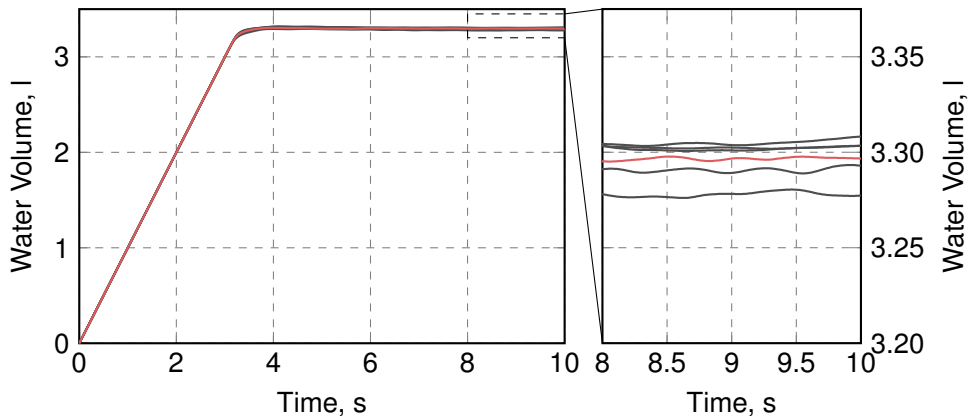


**Figure 8.2** Water height and velocity simulated in different meshes of the inlet geometry at the sample plane (Figure 8.1). The chosen mesh resolution is marked (—). The inlet tank starts of empty at  $t = 0$ , and the first water passes the weir after approximately 3 s. For the water height, the meshes at  $t = 10$  s are from top to bottom: 1x (—), 0.25x, 0.5x, 4x, 3x, 2x. For the water velocity, the order of the meshes is: 0.5x, 0.25x, 2x, 3x, 1x (—), 4x.

Directly comparing the water profile over the weir width (Figure 8.3), two positions of a slightly increased deviation can be found: At around 0.1 m relative to the symmetry plane the meshes with a higher resolution show a significantly (approximately 1 mm) increased peak. Farther to the side, at a width of 0.3 m a second peak occurs, which is not completely reproduced with the reference mesh, yet far better than with coarser meshes. Still, given the increased computational power required for the refined meshes, the precision of the reference



**Figure 8.3** Profile of the water height over the weir for different meshes averaged over the time  $t$  5–10 s (after the tank is filled and behaves quasi-stationary). The chosen mesh (—) slightly underpredicts the peak in water height that occurs at around 0.1 m weir width. Apart from that, however, it reflects the dynamic surface predicted by the meshes with higher resolution closely.

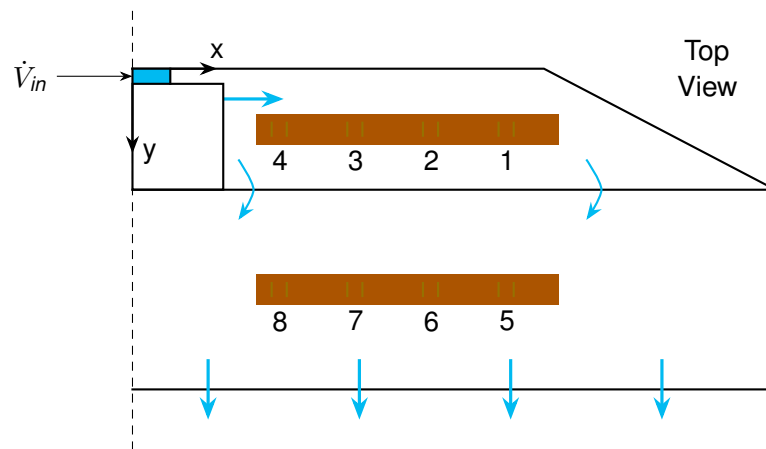


**Figure 8.4** The results for the accumulated water volume inside the inlet tank's simulation domain. At the start of the simulation,  $t = 0$  s, the inlet tank is empty. It then fills continuously over time. For all meshes, the results are very similar, with a maximum deviation of 0.5% from the reference case (—). The detail shows the volume in the last two seconds, the meshes associated with the lines (top to bottom): 4x, 3x/2x, 1x (—), 0.5x, 0.25x.

mesh is considered sufficient.

The water volume (Figure 8.4) behaves similar to the water height and velocity on the weir. Again, no substantial difference between the simulations can be seen, and it can rather be assumed that the difference occurs due to a finer resolution of the interface and thus a minimally increased number of “filled” cells.

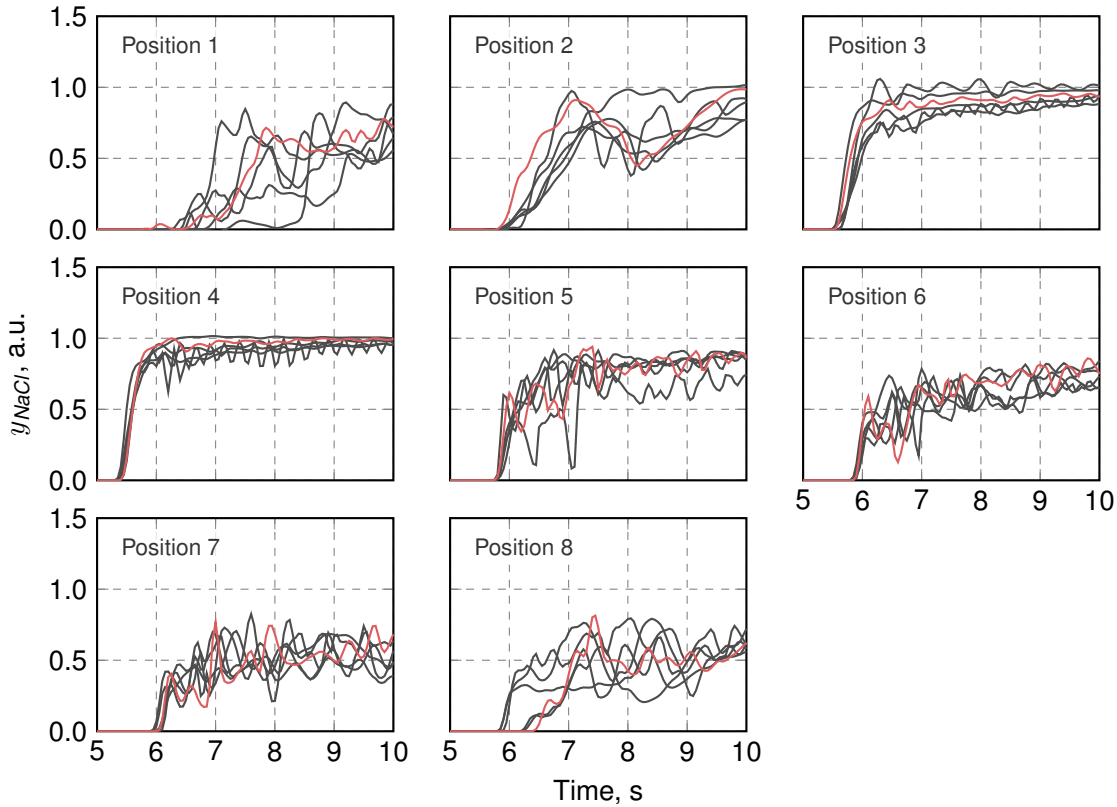
The final comparison of the meshes is given by evaluating the passive species transport of a *NaCl* tracer at eight positions. Therefore two sensor stripes, as described in Section 7.1 were fixed to the ground of the tank and on the weir (Figure 8.5). At the tank’s inlet a step function of in salt concentration was applied at  $t = 5$  s. This allowed to compare the results of the simulations with regard to the distribution inside the tank for different meshes (Figure 8.6).



**Figure 8.5** Positions of the sensors for the validation of the inlet tank. Position 1–4 are in the main tank, spanning from  $x = 0.1$  m (4) to  $x = 0.24$  m (1). The sensor spots on the weir, 5–8, reach from  $x = 0.1$  m (8) to  $x = 0.25$  m (5). The approximate flow of the water through the tank is marked as reference.

For position 1 a chaotic behaviour can be observed, where no clear “correct” solution can be found. This sensor is placed at the shallow end in the inner part of the inlet tank, the fluid flow might suffer from wall effects that are not completely resolved and strongly depend on the near-wall mesh resolution. For positions 2–7, however, a very good agreement between all simulations can be found. At position 8 again a deviation can be found. This can correspond to the unresolved peak in the water profile (remember Figure 8.3), which is at the same coordinate.

In summary, for nearly all measures, the chosen reference case shows very good agreement with the refined simulations. An overview over the deviations of all cases compared to the reference simulation can be found in Table 8.3. The use of a coarser mesh would also be possible without significant loss of precision, although this would reduce the resolution of flow effects such as velocity and smear the fluid interface that can be observed on the weir. Therefore the mesh introduced earlier was regarded as the optimal case for the purpose of this work.



**Figure 8.6** The simulated, relative tracer concentrations  $y_{NaCl}$  at all sensor positions for the different meshes. Position 1–4 are the sensors inside the main tank, position 5–8 are on the weir (Figure 8.5). Shown is the data starting at  $t = 5$  s, when the  $NaCl$  pulse is introduced to the system. Therein position 1 and 5 represent the outermost sensor spot ( $x \approx 0.25$  m) and position 4 and 8 the innermost ( $x \approx 0.1$  m). The reference case is again highlighted for all positions (—).

**Table 8.3** The cases evaluated for the mesh independence study. For each case  $\langle factor \rangle \times$ , the basic mesh has been refined in all axis directions by  $factor$  compared to the chosen reference case ( $\hat{\cdot}$ ). The error was calculated as deviation relative to the reference case. Errors are shown for water height, velocity and volume.

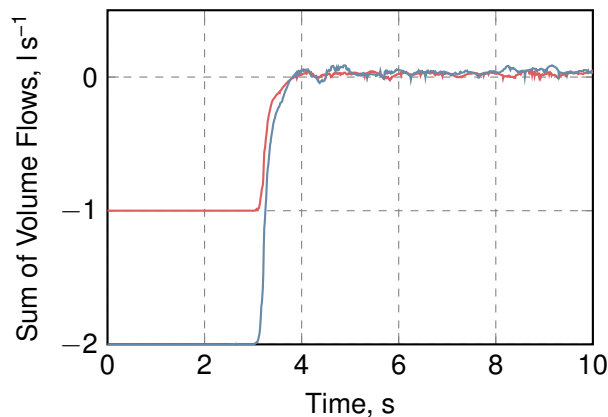
Case	$\varepsilon_h, \%$	$\varepsilon_v, \%$	$\varepsilon_V, \%$
0.25x	-3.89	1.46	-0.49
0.5x	-6.00	3.15	-0.18
1x $\hat{\cdot}$	-	-	-
2x	-9.02	-2.56	0.18
3x	-5.72	0.22	0.18
4x	-4.33	0.06	0.27



### 8.1.2. Symmetry Assumption

To reduce the computational requirements for the simulation, only half of the complete inlet tank should be simulated. To prove that this approach does not influence the solution, the reference simulation with an inlet width of 0.83 m and a volume flow  $\dot{V}$  of  $2.0 \text{ l s}^{-1}$  was calculated for the symmetric case and a mirrored mesh. This extended mesh is an exact replica of the other half and has been generated using the OpenFOAM utility `mirrorMesh`.

Figure 8.7 shows a comparison of the water balance in the tank for both cases: Initially the tank volume is being filled with the defined inlet volume flow, which, naturally, is twice as high for the complete tank. After approximately 3 s the water begins to exit the tank over the weir. The final balance, after the system has reached a quasi steady-state, is off by 2 % of the input volume flow for both cases. This minor error can result from numerical inaccuracies and can be considered negligible.

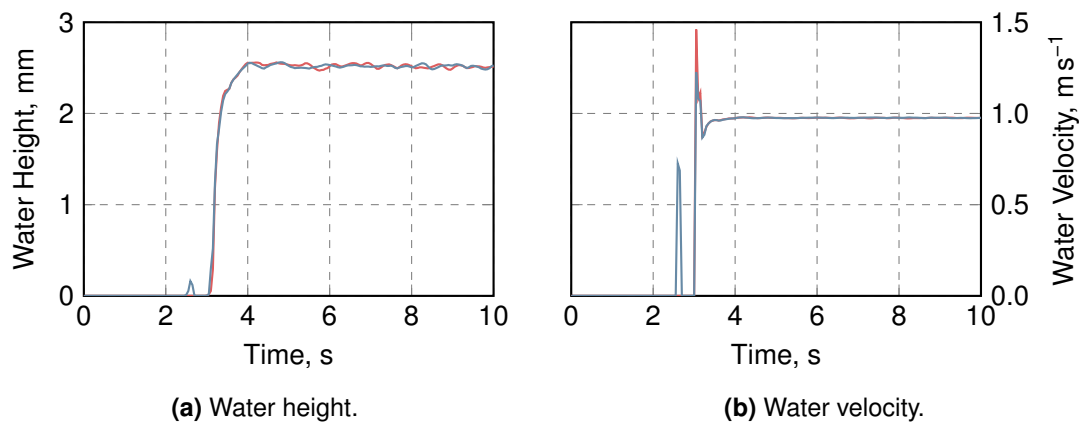


**Figure 8.7** Comparison of the water balance in the complete case (—) and the symmetric one (—): For both simulations the sum of flows balances after the inlet is filled ( $t \geq 5 \text{ s}$ ), with an average sum of  $2.1 \cdot 10^{-2} \text{ l s}^{-1}$  for the symmetric case and  $3.7 \cdot 10^{-2} \text{ l s}^{-1}$  for the complete inlet.

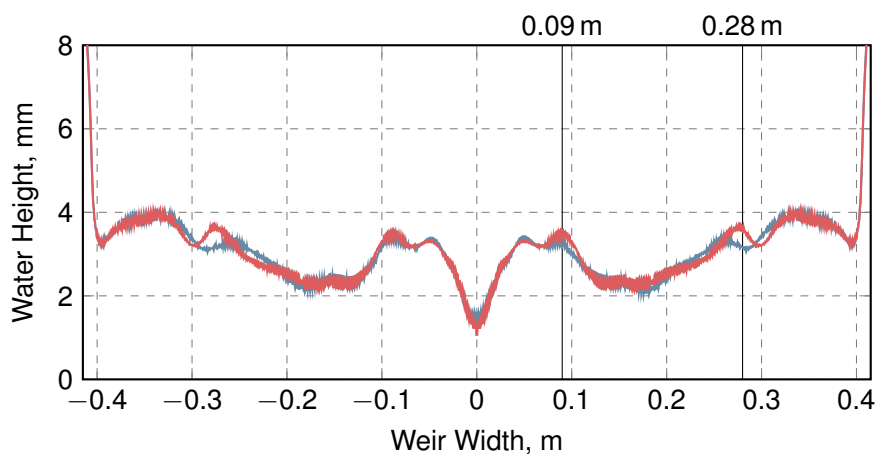
As the water height distribution over the weir width is another important aspect for the inlet tank, both simulations should give similar results regarding the average height and velocity as well as the profile over the width. The comparison of these attributes can be seen in Figures 8.8 and 8.9.

While for the average water height and velocity no significant difference can be found (deviation 2–3 orders lower than actual values), the water profile over the weir width shows some irregularities: At a weir width of around 0.09 m the symmetric case shows a peak, which does not occur in the simulation of the complete tank. This is the same peak that could also be observed in the mesh independence study (Figure 8.3). It could be discussed, whether the occurrence of this observed peak is a feature of the symmetric case, or an indication for a not sufficiently refined mirrored mesh. A further discrepancy can be found at 0.28 m, again a position where the mesh independence showed differences.

Nevertheless, the overall profile can be closely reproduced with only half the geometry. Moreover, any difference found has already been observed in the mesh independence study, thus can not be assigned unambiguously to the introduction of a symmetry plane. Summarised,



**Figure 8.8** Comparison of the water height and velocity averaged over the weir width in the complete case (—) and the symmetric one (—) at the end of the weir: In steady state ( $t \geq 5$  s), the average velocity and water height are  $0.975 \text{ m s}^{-1}$  and  $2.518 \text{ mm}$  for the symmetric case and  $0.976 \text{ m s}^{-1}$  and  $2.515 \text{ mm}$  for the complete.



**Figure 8.9** Comparison of the water profile averaged over time (in steady state,  $t \geq 5$  s,  $\Delta t = 0.5$  s) over the weir width in the complete case (—) and the symmetric one (—) at the end of the weir. The results of the symmetric simulation were mirrored at  $x = 0$  m to be able to compare both sides.

it could thus be shown that — although some minor deviations between the mirrored case and the symmetric exist — no substantial variance of the results can be observed.

### 8.1.3. Validation

To validate the inlet geometry, measurements with the rapid salinity sensors (Section 7.1) were made for a tank with a width of 0.83 m. The same configuration as described earlier (Figure 8.5) was used, with two sensor stripes attached to the ground of the inlet tank and the weir. To compare simulation and actual reactor system a step function in salt concentration was applied and the resulting response at the sensor positions was measured.

The reactor system was modified to allow switching the influent to the recirculation pump from the retention tank to a secondary tank filled with salt-water. In the experiments, the reactor was allowed to develop a quasi-stationary behaviour with the default configuration (cyclic pumping of water from the retention tank). At a time  $t = 0$  the measurement was started. After 5 s the feed for the pump was switched to the secondary tank, initiating the step function. The measured conductivities were then smoothed with a sliding maximum algorithm with a window width of 0.5 s.

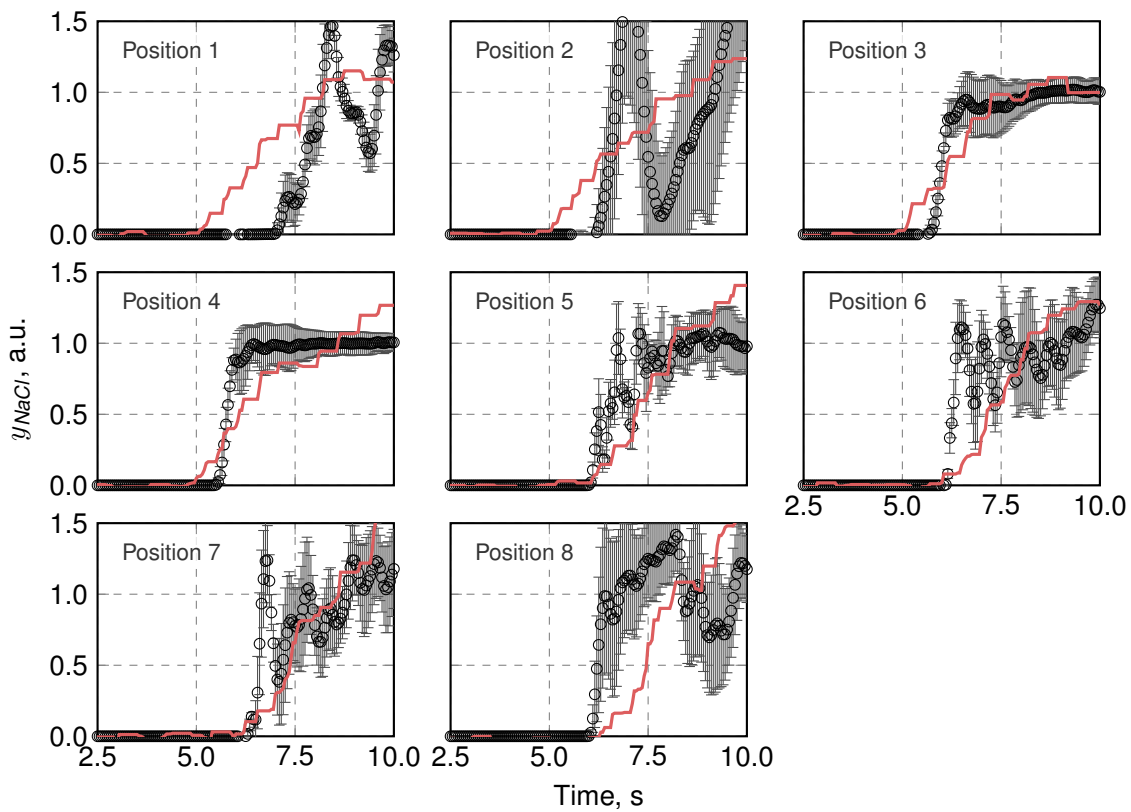
The results of this evaluation can be seen in Figures 8.10 and 8.11 for volume flows of 1.6 and 2.0 l s<sup>-1</sup>. In both simulations the results agree quite well for the sensor positions 2–7. The sensor at position 1 however shows larger deviations. The simulation substantially underpredicts the timing of the initial pulse of tracer concentration, and — at least for  $\dot{V} = 1.6$  l s<sup>-1</sup> — the increase itself. Looking more closely at the simulation (Figure 8.12) one can see that an eddy occurs in the outermost region of the inlet tank (Figure 8.12b). This “redirects” the incoming tracer to the surface, effectively delaying the detection of the increased tracer concentration by the simulation at position 1. The same effect leads to the oscillation at position 2. The first pulse of tracer reaches the measurement position, before the eddy temporarily redirects the flow and the concentration decreases again.

The last sensor spot at position 8 also shows a larger difference between simulation and measurements for 1.6 l s<sup>-1</sup>. Figure 8.13 shows this sensor at  $t = 0.77$  s. Here a peak in the water height can be observed, which was also described in the previous sections as feature that differs with the used meshes.

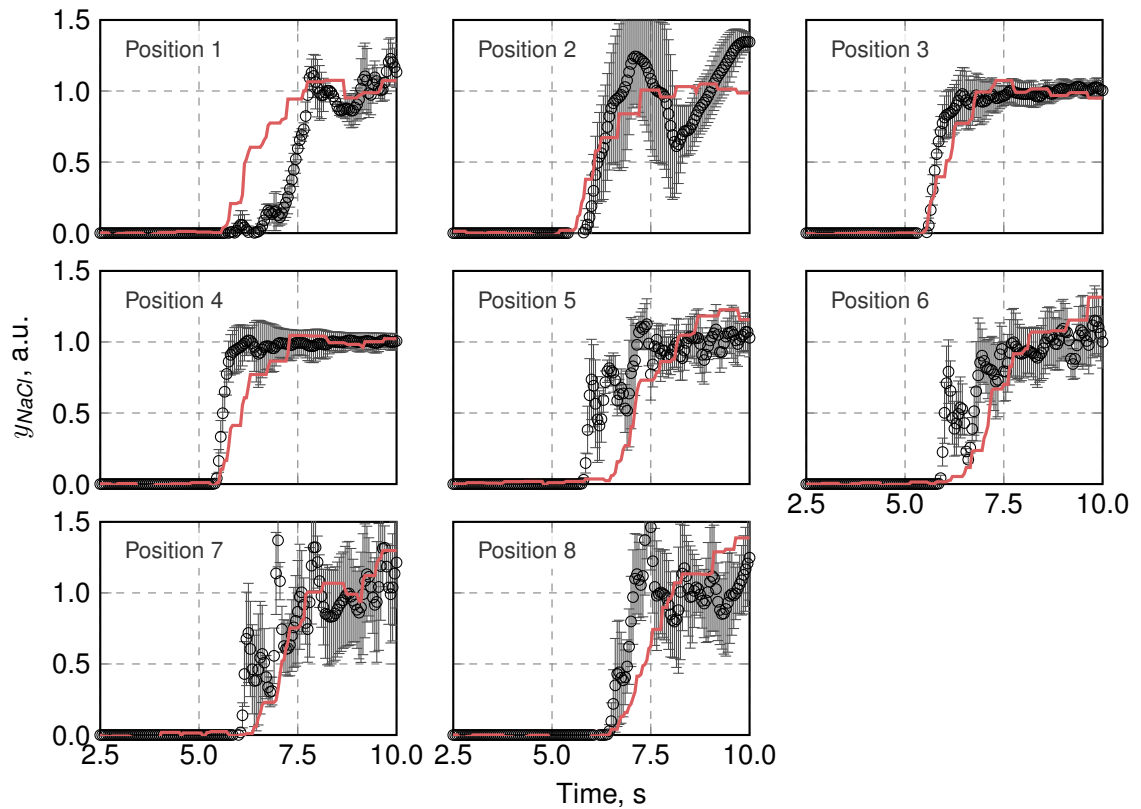
Overall, it could be shown that the simulation can reproduce the measurements for the main part of the weir. With given delays of less than 1 s the tracer concentration can be reproduced at every sensor position. Comparing the different volume flows it appears from Figure 8.11 that the higher volume flow can be described better by the simulation. A reason for that can be the increased average velocity: For higher velocities, the thickness of the laminar wall layer decreases, possibly improving the results of the wall functions used for the turbulence models.

### 8.1.4. Analyses of Geometry and Operating Conditions

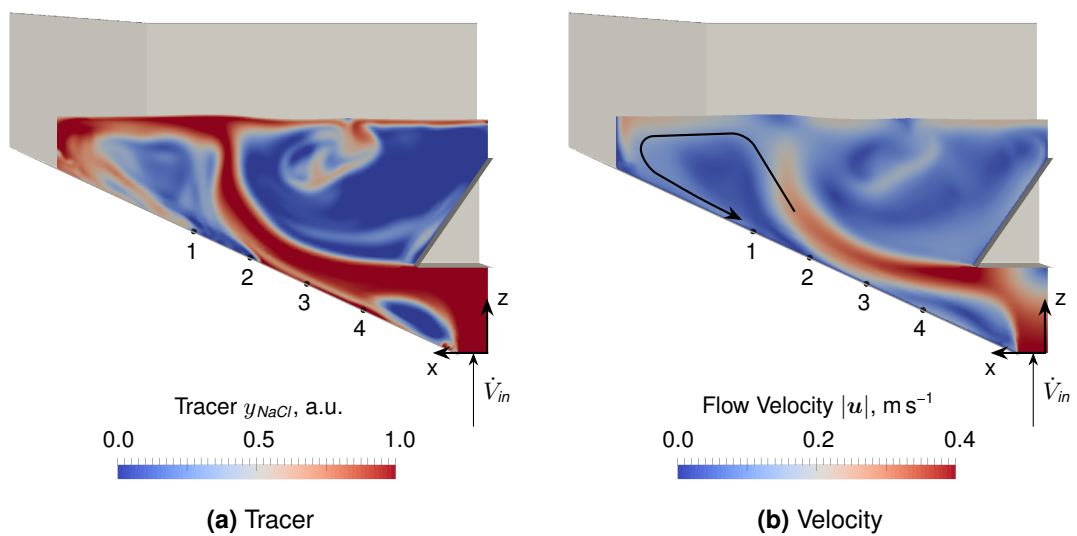
With the given mesh, it was now possible to investigate the influence of modified geometry or operational conditions on the behaviour of the inlet. In the following paragraphs the results for these modifications will be presented. The parameters changed are listed in Table 8.1. The simulations were evaluated in terms of water height on the weir, inlet pressure, and overall



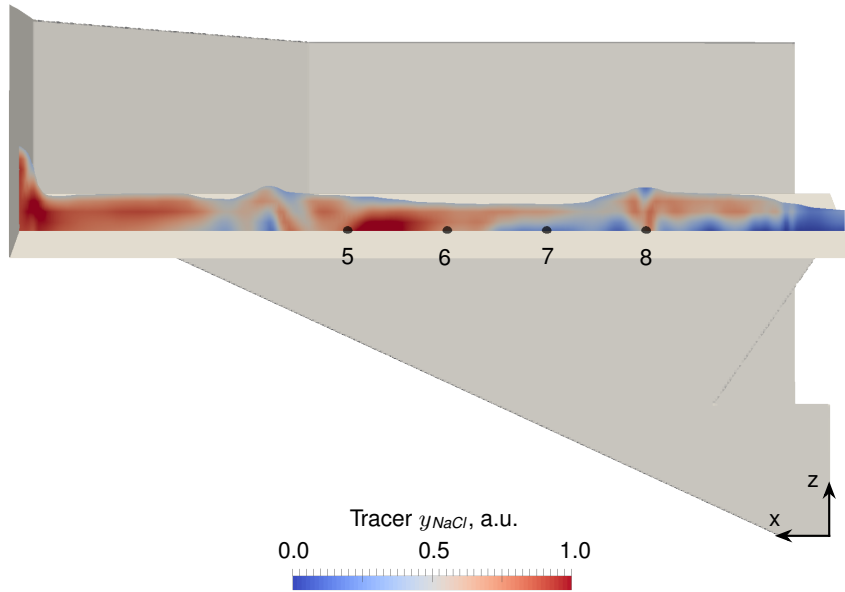
**Figure 8.10** Results of the tracer experiments at a volume flow of  $1.6 \text{ l s}^{-1}$ . Measurements (—) were smoothed using the sliding maximum algorithm in central window mode with a window of 1 s. The simulation results (○) were sampled over all cells in a radius of 0.5 cm around the sensor position. The positions correspond to Figure 8.5. Concentration data was normalised to the range  $0 < y_{\text{NaCl}} < 1$ .



**Figure 8.11** Results of the tracer experiments at a volume flow of  $2.0 \text{ l s}^{-1}$ . Measurements (—) were smoothed using the sliding maximum algorithm in central window mode with a window of 1 s. The simulation results (○) were sampled over all cells in a radius of 0.5 cm around the sensor position. The positions correspond to Figure 8.5. Concentration data was normalised to the range  $0 < y_{\text{NaCl}} < 1$ .



**Figure 8.12** Frontal view on the inlet tank, showing relative tracer concentration  $y_{\text{NaCl}}$  and velocity  $u$  in the inner region at  $t = 7.7 \text{ s}$ . The velocity graph shows the developed eddy in the outer region of the inlet tank. Sensor positions are marked (●).



**Figure 8.13** Frontal view on the inlet tank, showing relative tracer concentration  $y_{NaCl}$  on the weir at  $t = 7.7$  s. Fluid layer is scaled to five times the actual height for visualisation. Sensor positions are marked (●).

flow behaviour.

All simulations in this section were computed with an inlet width of 1 m, a normalised volume flow of  $2.4 \text{ l s}^{-1} \text{ m}^{-1}$ , and a weir height of 18 cm if not noted otherwise.

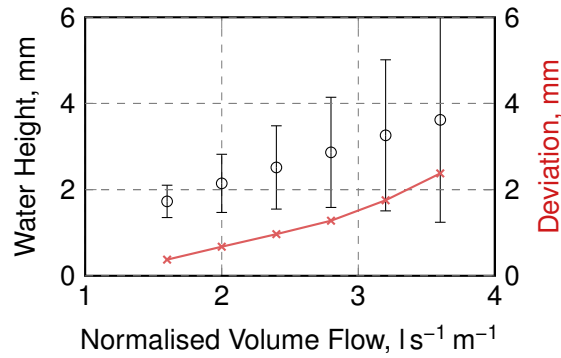
### Volume Flow

The volume flow through the reactor directly affects the cultivation conditions on the channel. It furthermore is a parameter, which can be changed easily for or even during a process. Thus it plays a major role in the operation of a Thin-Layer Cascade (TLC).

The results of modified volume flows can be seen in Figure 8.14. The graph shows the average water height developing on the weir in steady state after  $t = 5$  s. As dictated by conservation of mass (volume) the average water height on the weir increases proportionally to the volume flow. Likewise, an increase in the differences in water height over the weir width can be observed. Looking closely at the results, an increase of the slope at  $\dot{V}_b = 2.8 \text{ l s}^{-1} \text{ m}^{-1}$  can be found. The corresponding values are listed in Table 8.4.

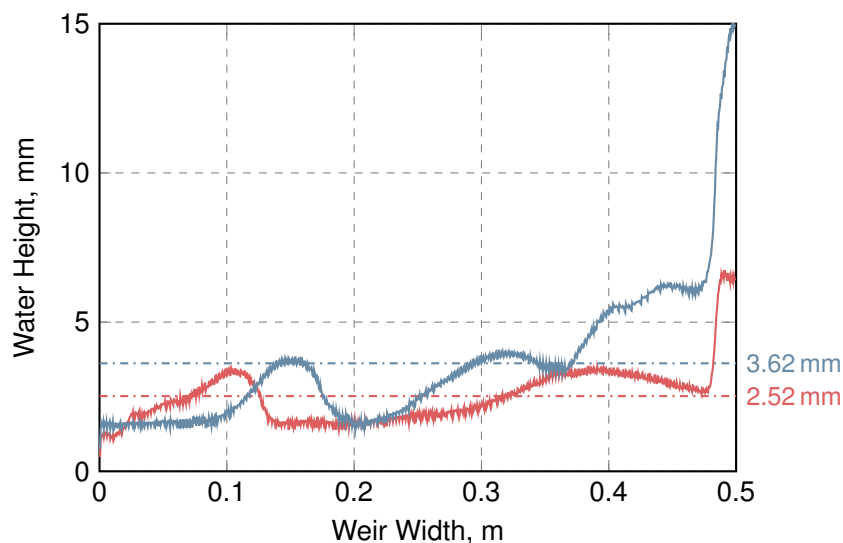
**Table 8.4** Mean water height and standard deviation over the width of the weir depending on the normalised volume flow (compare Figure 8.14).

Volume flow $\dot{V}_b$ , $\text{l s}^{-1} \text{ m}^{-1}$	Water height $h$ , mm	Deviation $\sigma(h)$ , mm
1.6	1.73	0.38
2.0	2.15	0.68
2.4	2.52	0.97
2.8	2.87	1.28
3.2	3.26	1.75
3.6	3.62	2.38



**Figure 8.14** Height of the water on the weir depending on the volume flow. Shown is the absolute water height on the weir ( $\circ$ ) averaged over the width including the standard deviation. As a measure for the height variance over the width the deviation has also been plotted separately ( $-x-$ ). Connecting lines between data for visualisation purpose only.

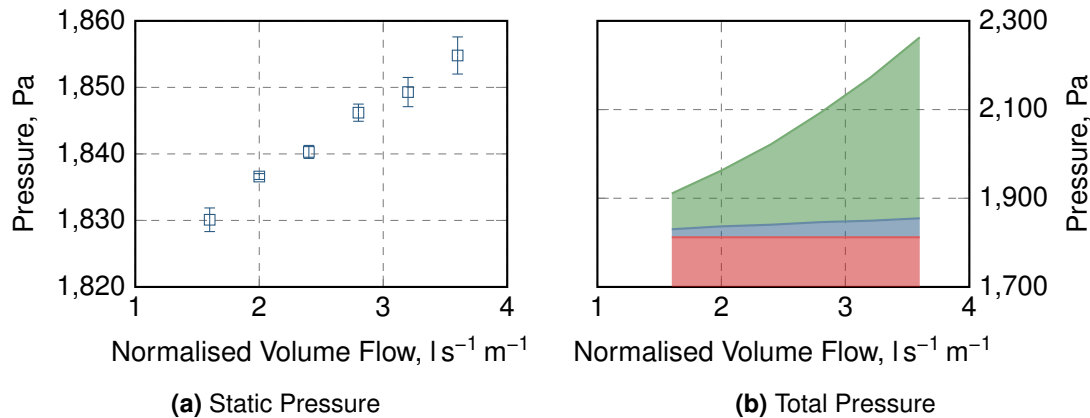
An increase in standard deviation for the first four volume flows of  $\Delta\sigma(h) = 0.3 \text{ mm}$  can be seen, while starting at  $2.8 \text{ l s}^{-1}$  it increases to  $0.5 \text{ mm}$ . This indicates that at high volume flows effects occur that significantly change the internal flow pattern. Figure 8.15 compares the water profile that develops for the case with  $\dot{V}_b = 2.4 \text{ l s}^{-1} \text{ m}^{-1}$  and  $3.6 \text{ l s}^{-1} \text{ m}^{-1}$ . This shows that for the higher volume flow ( $\text{—}$ ), starting at the side wall going up to  $0.1 \text{ m}$  into the inlet tank, the water piles up significantly more than for the lower volume flows.



**Figure 8.15** Comparison of water profile on the weir for normalised volume flows of  $\dot{V}_b = 2.4 \text{ l s}^{-1} \text{ m}^{-1}$  ( $\text{—}$ ) and  $3.6 \text{ l s}^{-1} \text{ m}^{-1}$  ( $\text{—}$ ). Also labelled is the average water height for both cases. For the higher volume flow, a piling up of water on the right side over a width of approximately  $10 \text{ cm}$  can be observed.

Another aspect of the inlet tank is the pressure required to pump the water. Figure 8.16a shows the pressure computed by the simulation that is required at the inlet boundary to allow for the defined fluid velocity. This pressure includes the pressure loss due to the geometric hindrance (e.g. the baffle) as well as hydrostatic pressure. To divide them, a linear fit can be created through the data points, with the pure hydrostatic pressure at the y-intercept at

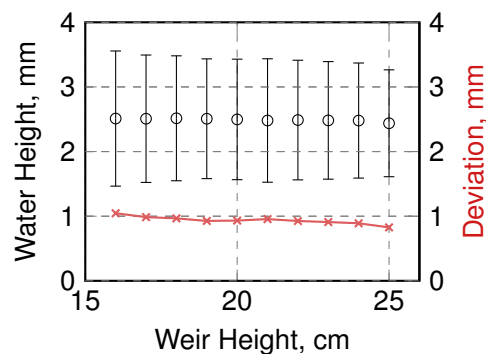
$\dot{V} = 0 \text{ l s}^{-1} \text{ m}^{-1}$ . Additionally, the dynamic pressure can be calculated from the inlet velocity ( $p_{dyn} = \frac{1}{2} \rho v^2$ ). The results of this separation can be found in Figure 8.16b. It is obvious that the influence of the pressure loss on the total energy requirements of the pump is negligible. Furthermore, in the inlet tank itself, even the fluid velocity plays a minor role relative to the hydrostatic pressure.



**Figure 8.16** Influence of the volume flow on the pressure required at the inlet. (a) shows the static pressure calculated in the simulation for the inlet averaged over time. The total pressure in (b) is the sum of hydrostatic (red) and dynamic pressure (green) as well as pressure loss (blue).

### Weir Height

The same analyses as for the volume flow can be done for the height of the weir. This height influences the hydrostatic pressure, and thus the counter pressure on the incoming fluid. However, it also strongly influences the volume in the tank, which should generally be minimised. Figure 8.17 shows the water height on the weir at different weir heights. Although the deviation shows a slight decrease with increasing weir height, no significant changes can be observed.



**Figure 8.17** Height of the water on the weir depending on the weir height. Shown are the absolute water height on the weir (o) averaged over the weir width. The corresponding standard deviations (—x—) are again plotted separately as a measure for the height variance over the width.

This can also be seen in Table 8.5. Increasing the weir height by 1 cm leads to a decrease

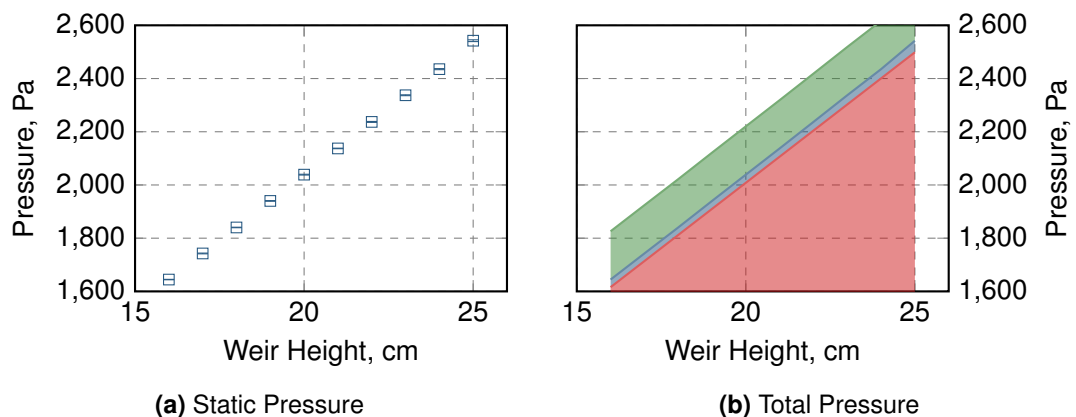


in the unevenness of approximately 0.03 mm while the volume in the tank increased by around  $\Delta V = 1 \text{ cm} \times 1 \text{ m} \times 0.08 \text{ cm} = 0.8 \text{ l}$  ( $h \times b \times d$ ). Compared to the reference case ( $h_{weir} = 18 \text{ cm}$ ) this changes the volume in the tank by nearly 20 %, and thus influences the surface-to-volume ratio  $S/V$  negatively. Interestingly, although minimal, the water height appears to decrease with an increased weir height, which can only be achieved by a faster fluid velocity on the weir. It can thus be assumed, that with higher weirs the dominance of the  $y$ -component of the velocity (in direction of the channel) increases compared to the other components and the flow is more aligned.

**Table 8.5** Mean water height and standard deviation over the width of the weir depending on the weir height (compare Figure 8.17).

Weir height, cm	Mean water height $h$ , mm	Deviation $\sigma(h)$ , mm
16	2.51	1.05
17	2.51	0.99
18	2.52	0.97
19	2.51	0.93
20	2.50	0.93
21	2.48	0.96
22	2.49	0.93
23	2.48	0.91
24	2.48	0.89
25	2.44	0.83

The results for the pressure are shown in Figure 8.18. As could be expected, the pressure required at the inlet (Figure 8.18a) increases with weir height. The accumulated curve, Figure 8.18b, differentiates the occurring effects. Again, it can be seen that the influence of the hydrostatic pressure is the main source for pump energy required.



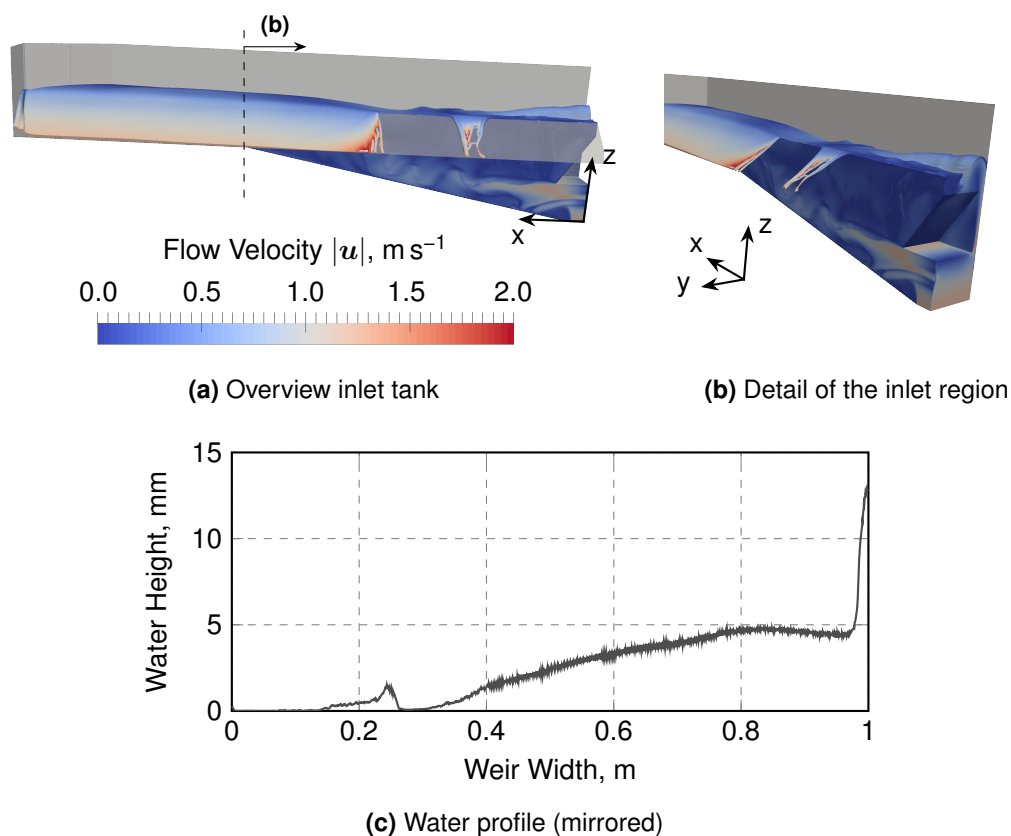
**Figure 8.18** Influence of the weir height on the pressure required at the inlet. **(a)** shows the static pressure calculated in the simulation for the inlet. The total pressure in **(b)** is the sum of hydrostatic (red) and dynamic pressure (green) as well as the pressure loss (blue).

## Inlet Width

The width of the inlet is an important aspect for upscaling of the reactor to industrial systems. The units presented so far have all been designed with a width of 1 m or less, while for the known production systems a width in the order of 10 m are common [Viera 2015].

To increase the width, the same method as in an experimental unit was used for the computational approach: The main tank's walls are widened to reflect the defined width, while the baffle and the inlet patch are kept identically. All measures in  $y$  and  $z$  direction remain unchanged. The inlet volume flow is set based on the specific volume flow  $\dot{V}_b = 2.4 \text{ l s}^{-1} \text{ m}^{-1}$  and thus increases proportional to the weir width.

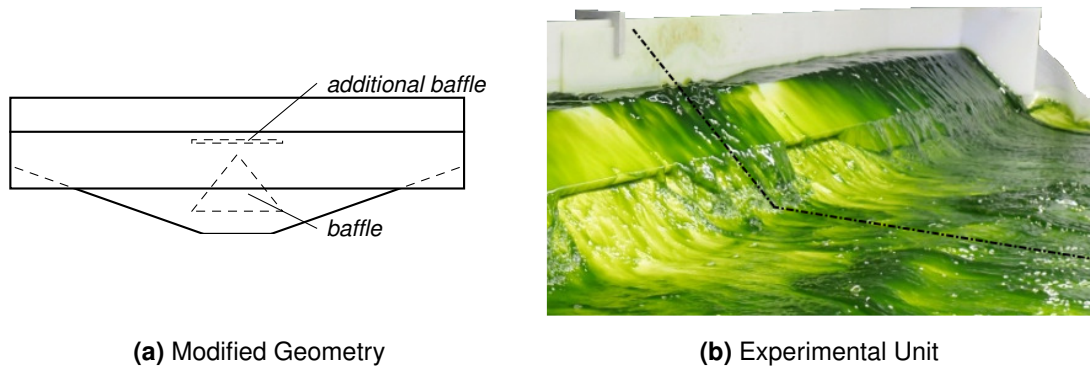
The results for a tank with a width of 2 m can be seen in Figure 8.19. This tank corresponds geometrically to a real, physical unit that has been employed in a pilot plant experimental reactor system at TUM AlgaeTec Center. The detail of the inlet, Figure 8.19b reveals the water flow is redirected at the baffle towards the sides ( $x$  direction) with hardly any  $z$  component.



**Figure 8.19** The inlet tank with a width of 2 m. An overview over the complete tank, view from the front, is shown in (a), while (b) focuses on the inlet region. Finally, (c) shows the water profile extracted from the simulation starting at the symmetry axes. The images show nearly no water flow over the weir in the inner region, while for the remaining tank the distribution is relatively even, as can also be seen in the profile of the water height.

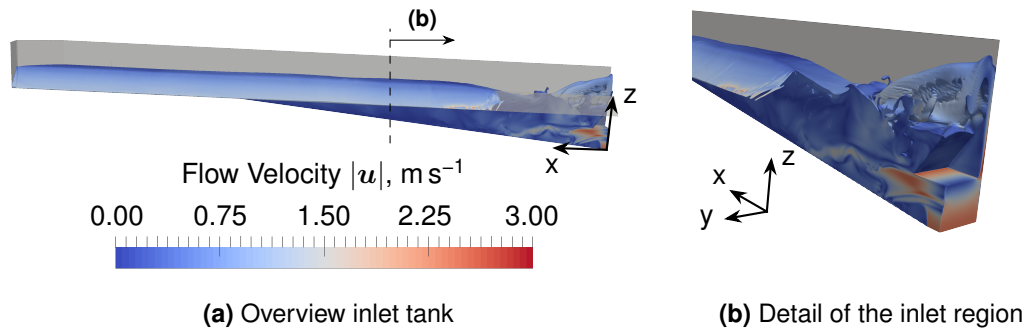
The result of this is an approximately 30 cm wide area in the centre of the inlet, where nearly no water flows over the weir (Figure 8.19c). Farther to the side, however, the surface is relatively even. This observation corresponds to the real unit's behaviour, Figure 8.20. Here, however, an additional flat-plate baffle has been installed directly above the inlet over the

existing baffle (Figure 8.20a), to help redirecting the flow and prevent the vertical peak that can be observed in the simulation (Figure 8.19b). This leads to an increased volume flow along the symmetry axis, which cannot be observed in the simulation.



**Figure 8.20** The modified geometry and experimental observation of the used inlet tank. An additional baffle has been added above the existing one (a). The photograph of the physical unit shows a similar behaviour to the simulative predictions in Figure 8.19 in reality. The different baffle configuration led to an increase flow of the suspension along the middle (---), which cannot be observed in the simulation.

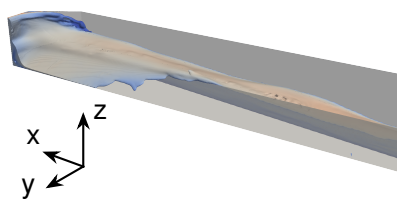
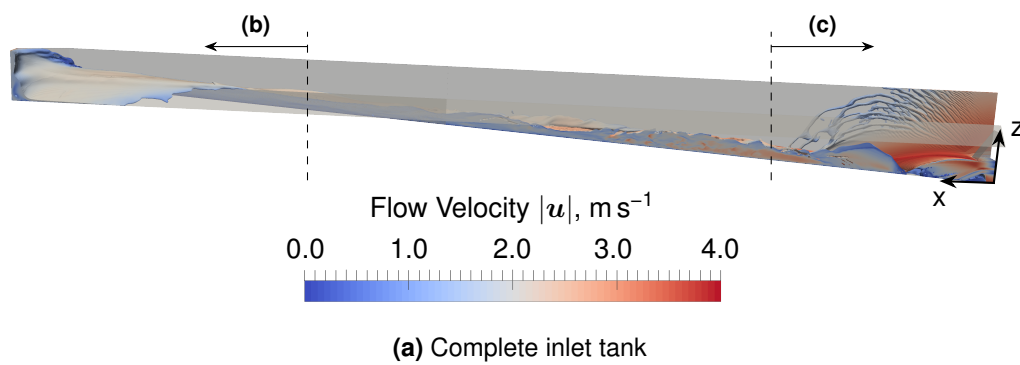
Using a width of 4 m (Figure 8.21) results in a similar behaviour as for the 2 m tank. At the inlet, the water is pushed to the sides, leaving the weir dry. Turbulence around the inlet has increased significantly, and intense instabilities in the water surface directly above the inlet can be found.



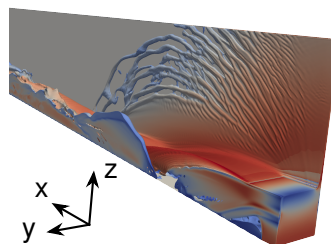
**Figure 8.21** Inlet tank with a width of 4 m: (a) shows an overview over the complete tank, while (b) only shows the detail from the inlet region. The turbulence at the inlet has increased observably, and the inner region is still not wetted. However, for the remaining weir again an even surface can be achieved.

Increasing the width further, while keeping the inlet geometry identical, results in flow velocities that exceed the manageable range: Figure 8.22 shows the corresponding simulation for a width of 6 m. Around the inlet region, the fluid is accelerated to high speeds and forms a beautiful, albeit useless, fountain. The fluid redirected by the baffle is accelerated to velocities, where leaving the inlet tank via the weir does not seem a considerable option. Instead, the wall at the side of the inlet serves as an emergency brake (Figure 8.22b).

It is quite obvious that this configuration is not a reasonable, physical unit. However, it actively demonstrates the limits of purely computational simulations: The assumption of an



(b) Detail of the side wall



(c) Detail of the inlet region

**Figure 8.22** Details of the inlet tank with a width of 6 m. While (a) shows the complete tank, (b) and (c) shows details from the side wall and the inlet region without the weir. At this width, the flow velocity required at the inlet is that extreme, that the water flows through the tank uncontrollably. The gap between baffle and wall leads to a “fountain-like” effect, and the redirection of the baffle accelerates the fluid to the sides, where it leaves the tank. It should be noted that for this simulation all fluid touching the atmospheric boundary (top) also leaves the simulation and is not considered for further computations.

unchanging inlet patch and baffle relates to the idea that it is beneficial to re-use existing geometries and materials (pipes, pumps). However, while these ideas are certainly given for actual units, too, it also is restricted by physical problems, which would barely allow these high input velocities.

### 8.1.5. Estimation of Power Requirements

The power requirements of a reactor are one major characteristic that — especially for the production of biofuel — decide over the feasibility of the system. As the inlet tank is the unit directly after the pump, it defines the power input needed to circulate the suspension through the reactor. Therefore the pressure loss of the fluid between the retention tank and the inlet was calculated. This pressure loss has to be compensated by the centrifugal pump and directly corresponds to the power  $P$  required as described by Equation (8.1).

$$P = \Delta p \cdot \dot{V} \quad (8.1)$$

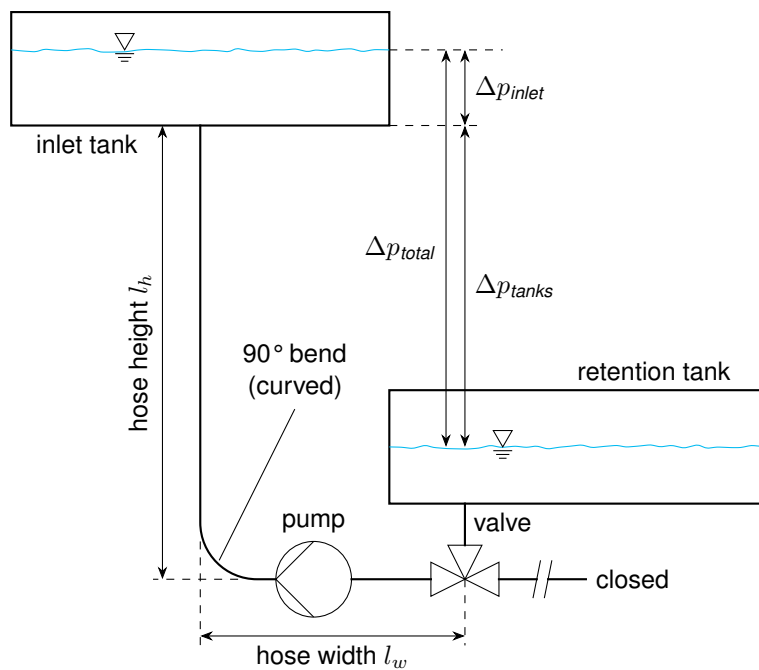
Above, for the variation of the volume flow and the weir height, the pressure loss has been calculated. However, further aspects have to be considered to accurately predict the required power (compare Figure 8.23):

- The surface of the hose connecting the tanks induces friction on the fluid, which adds to the pressure loss proportional to the length of the hose and inverse proportional to the hose's diameter. The hose used in this setup has an inner diameter of 50 mm. The length  $l$  of the pipe can be estimated by the height difference between the retention tank and the inlet tank ( $l_h = 0.9$  m) as well as half the width of the channel ( $l_w = 0.5$  m) [Apel 2017].
- Curvature of the hose as well as fittings (e.g. a valve) increase the friction effects on the fluid. The reactor design presented in this thesis redirects the flow from the retention tank via a 90° valve and a curved hose with a large radius.
- The hydrostatic pressure between the inlet tank and the retention tank  $\Delta p_{tanks}$  has to be added, since the above evaluation only considered the hydrostatic pressure inside the tank ( $\Delta p_{inlet}$ ). The total height difference between the surfaces of the retention tank and the inlet tank's inlet is approximately 0.9 m, resulting in a hydrostatic pressure of 8.8 kPa.
- The pumps efficiency differentiates between the power actually added to the system and the power needed (and thus measured) to operate the pump. For laboratory scale devices this efficiency is usually between 0.6 and 0.75 [Baerns 2006, p. 234].

From the assumptions listed, the pressure loss  $\Delta p$  can be calculated following Equation (8.2) [Baerns 2006] (also called Darcy-Weisbach equation).

$$\Delta p = \left( \lambda \frac{l}{d} + \sum \xi \right) \frac{\rho v^2}{2} \quad (8.2)$$

In this equation the hose's friction is included in the first term with the Darcy friction factor  $\lambda$ . The effect of fittings can be included by resistance coefficients  $\xi$ , which are added for every considered element. The total pressure loss is further proportional to the fluids density  $\rho$  and



**Figure 8.23** Schematic representation showing the sources for pressure loss in the system: The hydrostatic pressure difference between both water surface ( $\Delta p_{total}$ ) has to be overcome by the pump. This can be split in the contribution between the tanks ( $\Delta p_{tanks}$ ) and inside the inlet tank ( $\Delta p_{inlet}$ ). The friction in the hose is defined by its overall length  $l = l_w + l_h$ . Further influences on the pressure are induced due to the 90° valve and the curvature of the hose. The power required to overcome all the pressure losses is provided by the pump.

the squared velocity  $v$ .

To calculate the Darcy-Weisbach friction coefficient in turbulent regimes ( $Re > 4000$ ) the Colebrook Equation (8.3) can be used [Avallone et al. 2007].

$$\frac{1}{\sqrt{\lambda}} = -2 \log_{10} \left( \frac{\varepsilon}{3.7 \cdot D_h} + \frac{2.51}{Re \sqrt{\lambda}} \right) \quad (8.3)$$

This equation includes the absolute surface roughness ( $\varepsilon = 0.0015$  mm for Polyvinyl chloride (PVC) tubes [Larock et al. 2000, Table 2.1]), the hydraulic diameter  $D_h$ , which corresponds to the tubes diameter, and the Reynolds number  $Re$  of the flow. Resistance coefficients for the bend and the valve can be determined to  $\xi_{bend} = 20\lambda$  and  $\xi_{valve} = 200\lambda$  respectively [Avallone et al. 2007, Table 3.3.11]. With this data, the separate pressure losses as well as the power required can be calculated for different volume flows (Table 8.6). All other parameters were set to the reference with a weir width of 1 m and a weir height of 18 cm .

**Table 8.6** Summary of the flow conditions, surface friction, pressure losses, and power required in the system. The friction coefficient has been calculated from the Colebrook Equation (8.3). The total pressure loss includes a hydrostatic pressure of 8.8 kPa (not listed in table) between the tanks and a pressure of approximately 1.8 kPa inside the inlet tank, found by the simulation (Figure 8.16). For the pump's power a "worst case" efficiency of 0.65 was assumed.

Volume Flow $\dot{V}$ l s <sup>-1</sup>	Reynolds $Re$ -	Friction $\lambda$ -	Pressure Loss				Power	
			Pipe kPa	Valve kPa	Bend $\Delta p_i$ kPa	Total kPa	Input W	Pump $P$ W
1.6	40 700	0.0220	3.3	23.4	2.3	39.6	63	98
2.0	50 900	0.0208	4.9	34.7	3.5	53.7	107	165
2.4	61 100	0.0201	6.7	48.0	4.8	70.2	169	259
2.8	71 300	0.0195	8.9	63.4	6.3	89.3	250	385
3.2	81 500	0.0189	11.2	80.3	8.0	110.2	353	543
3.6	91 700	0.0185	13.9	99.5	10.0	134.1	483	743

Experimentally, the power required for a volume flow of 2.4 l s<sup>-1</sup> was determined to 270 W, which agrees well with the estimation of 260 W. It thus can be concluded, that the major influences on the pressure loss of the flow have been detected.

From the summary of the results it can be found that the design of the inlet tank only has minor influences on the power required (less than 5%). Primary improvements can be achieved by reducing the losses in the piping system: Pipes with a larger diameter reduce the effect of all influences. The usage of the current valve should be reconsidered, as it has significant influence on losses. Furthermore an industrial size pump could operate at efficiencies of up to 0.95, further decreasing the power required for the reactor operation. This is also shown in Table 8.7. The areal productivities for the current setup as well as for a "best case", in which the valve was replaced by a bend and the pumps efficiency was increased, were compared. It can be seen, that with these changes the required power can be reduced by up to 77%. Still, the contribution of the tank is less than 10%. Overall, this calculation shows that significant potential is given to improve the energy balance of the reactor system, however, an extensive investigation of modified tank geometries with the goal of optimising the

power requirements is — at this point — not recommended.

**Table 8.7** Current areal power requirement of the reactor as well as possible energy reduction for a “Best Case” scenario, where the valve is replaced by another bend and the pump is operated at an efficiency of 0.95. This allows a reduction in power requirement of around 75%. Furthermore, for each volume flow the contribution of the tank itself to the power requirements can be seen.

Volume Flow $\dot{V}$ l s <sup>-1</sup>	Areal Power Requirement			Contribution Tank -%
	Current $P/A$ W m <sup>-2</sup>	Best Case $P/A$ W m <sup>-2</sup>	Reduction - %	
1.6	12.2	3.9	68	9.8
2.0	20.6	5.9	71	8.2
2.4	32.4	8.5	74	6.8
2.8	48.1	11.9	75	5.7
3.2	67.9	16.0	76	4.9
3.6	92.8	21.1	77	4.2

### 8.1.6. Conclusion and Discussion

In this chapter the validity and applicability of the inlet’s mesh was demonstrated: Based on a mesh independence study, the influence of a refinement of the mesh could be shown to be minimal. Using only half the geometry for the simulation, presuming a symmetry plane in the centre, allowed to decrease to computational complexity. No significant differences between the symmetric and a complete simulation were evident. Finally, the simulation was validated using salinity measurements that were compared to the results of a simulation with passive species transport. Here few deviations could be found at the boundaries of the measurement area, however, the primary zone on the weir could be described sufficiently well by the simulation. All deviations found during validation and symmetry check were in the same region, where differences for the measured properties could be found in the mesh independence study, too. This can be interpreted as a further indication that the mesh is not sufficiently refined in this region. However, in the overall profile the variation emerges in a small section of the weir and has no substantial influence. Thus a refinement of the mesh, which would approximately double the computational effort is considered unfeasible for a parameter study as presented in this chapter. Still, it can be used in analyses of inlet tanks for which detailed information should be obtained.

After the validation of the simulation, some analyses of the inlet tank’s behaviour could be simulated: The influence of different volume flows and weir heights on the water height and pressure loss was evaluated. It could be shown that increased volume flows lead to a more unstable water profile, while changing the weir height has no serious influence. The pressure required at the inlet due to geometric hindrance was negligible for all volume flows compared to hydrostatic and dynamic pressure. It should however be noted, that especially at higher volume flows the friction in the connecting hose increases the pressure loss with the square of the fluid velocity, which is not included in the calculation shown here. The mass of the fluid for increased weir heights does increase the pressure (compared to the reference case) by



approximately 30 %, however, as stated above, these increased weir heights are not beneficial for the operation anyway.

Investigating an upscaling of the inlet tank, the tank width was modified. In this modification only one dimension of the tank was modified, while the connection to the pump's hose was assumed to behave unchanged. Although the results for scales up to 4 m showed the expected behaviour, it resulted in highly distorted velocity profiles for tank widths above 6 m. Despite the fact that modifying the inlet boundary and the baffle would easily fix the problem, this shows the limits of Computational Fluid Dynamics (CFD) simulations: Both elements in need of change are restricted to physical boundaries, e.g. pipe diameters and accessibility for construction. This gives a finite number of reasonable options. In simulative approaches, however, an arbitrary number of possible configurations exist, but cannot — due to the high computational requirements — be evaluated. It is thus necessary to restrict the changes applied to physical examples. Since these were not yet available for this thesis, no further upscale cases could be compared. However, given the valid simulation combined with the promising results of the 2 m inlet tank simulation, it should be possible to predict the behaviour of a modified tank closely to reality once the range of constructive options has been found.

Regarding the power input required to circulate the suspension, it could be found that the design of the inlet tank itself plays a minor role compared to the pressure losses due to the piping of the reactor. It was thus concluded that an extensive investigation of power-optimised inlet tank designs is not reasonable at this point. However, along with the upscale of the reactor, significant optimisation of the piping system could be achieved: Removing the valve (e.g. replacing it with a second outlet from the retention tank) will have a major influence. Furthermore, due to larger volume flows, the pipe's diameter has to be increased, which can be chosen to further reduce effects of friction. Finally it should be considered, that while the efficiency factor for laboratory centrifugal pumps is between 0.6–0.75, industrial size pumps can achieve efficiencies of 0.95 [Baerns 2006], further reducing the power requirements.

## 8.2. Channel

Considering the sequence of units in the TLC reactor, the channel comes in second place. When cultivating microalgae, this is the only area where light is highly available to the organisms and it must thus be considered as the main unit of the reactor.

In the following section the mesh used for the channel simulations will be analysed and its ability to describe the real unit will be shown. Afterwards, the validated model will be used to perform a parameter study on the channel, where geometric parameters and the fluid properties are changed (Table 8.8). The influence of these parameters on the fluid layer and its mixing behaviour will be shown and discussed in details. A final summary will review the results for the channel and examine further options for a simulative model.

### 8.2.1. Mesh Independence

The reference grid for the mesh independence study had a spacial resolution of  $100 \times 600 \times 20$  cells for the region of interest. The channel was sloped at an angle of  $\alpha = 1^\circ$  and a volume

**Table 8.8** Parameters that were modified in the investigation of this reactor unit. The range has been defined by common operational parameters from literature as well as expected/observed algae viscosity for concentrations up to 50 g l<sup>-1</sup> dry cell mass. Compare Table 5.3 for all the unit's parameters.

Parameter		Range
Angle	$\alpha$	0.5–3 °
Volume flow	$\dot{V}_b$	1.6–3.6 l s <sup>-1</sup> m <sup>-1</sup>
Viscosity	$\mu$	1–10 mPa s

flow of  $\dot{V}_b = 2.4 \text{ l s}^{-1} \text{ m}^{-1}$  was applied. To analyse the effect of modifications of the mesh topology, a number of simulations were performed in which the cells were refined in multiple directions. Given the refinement factor  $f_{total}$  of the total mesh, each single axis direction was refined by  $f_{x/y/z} = f_{total}^{\frac{1}{3}}$ . This factor was also applied, when not all mesh directions were refined. A list of simulated cases can be found in Table 8.9.

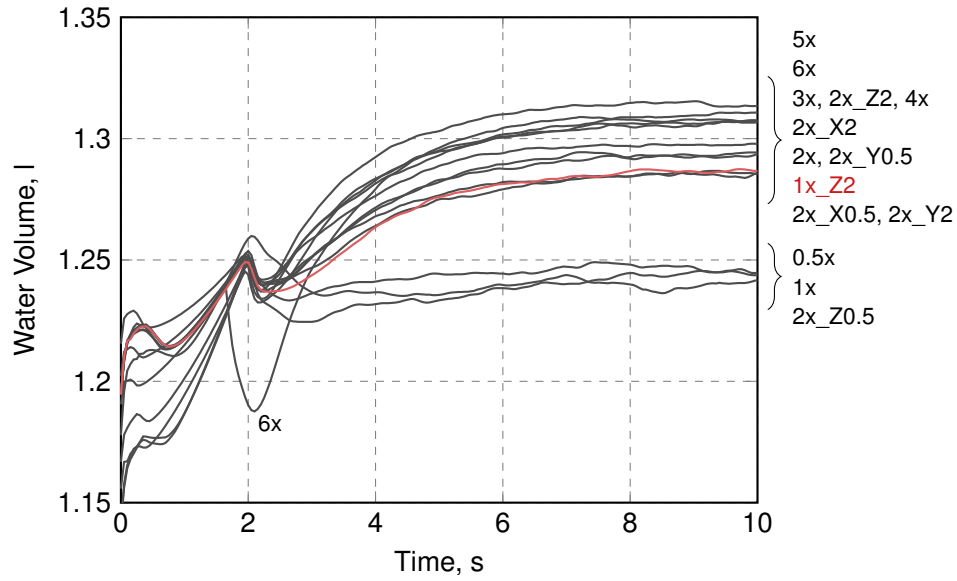
**Table 8.9** The cases evaluated for the mesh independence study. Naming conventions: <factor>x signals the number of cells used in this case relative to the reference case (1x). The appendix \_(X/Y/Z)<factor> signals a refinement of cells in the named dimension(s) only. Also shown is the mesh resolution in all directions. Due to the grading in z-direction, the resolution of the cells increases in z direction. The case marked (ˆ) is the option chosen for all further simulations.

Case	Cells, - 10 <sup>6</sup>	Resolution, mm			
		x	y	z (ground)	z (top)
0.5x	0.63	5.25	2.52	0.24	3.45
1x	1.25	4.15	2.00	0.20	2.78
1x_Z2 <sup>ˆ</sup>	1.56	4.15	2.00	0.16	2.24
2x	2.48	3.29	1.59	0.16	2.24
2x_X0.5	1.97	4.15	1.59	0.16	2.24
2x_X2	3.13	2.61	1.59	0.16	2.24
2x_Y0.5	1.97	3.29	2.00	0.16	2.24
2x_Y2	3.12	3.29	1.26	0.16	2.24
2x_Z0.5	1.98	3.29	1.59	0.20	2.78
2x_Z2	3.08	3.29	1.59	0.13	1.81
3x	3.76	2.88	1.39	0.14	1.93
4x	5.05	2.61	1.26	0.12	1.76
5x	6.23	2.43	1.17	0.12	1.65
6x	7.45	2.28	1.10	0.11	1.56

All cases were analysed with regard to the relevant parameters water height  $h$ , water velocity  $v$ , and additionally the water volume  $V$  on the channel.

Concerning the water volume (Figure 8.24), it can be seen that the reference case (1x) does not yield the same result as the cases where the mesh has been refined in all directions (2x–6x).

However, a detailed study of refinement in individual axis directions revealed that the main influence on the result is due to the  $z$  direction: The simulation where every direction was refined except for  $z$  (2x\_Z0.5) yields no improvement, while for a refinement in  $z$  direction only (1x\_Z2) a volume very similar to those of higher refinements can be achieved. A summary of



**Figure 8.24** Overall water volume on the channel over time for different meshes. The channel was pre-filled to height  $h$  and set to velocity  $v$  as predicted by the Manning equation (Equation (7.11) and (7.9)) to speed up convergence. The selected case is marked (1x\_Z2, —), all other cases listed in order of final volume.

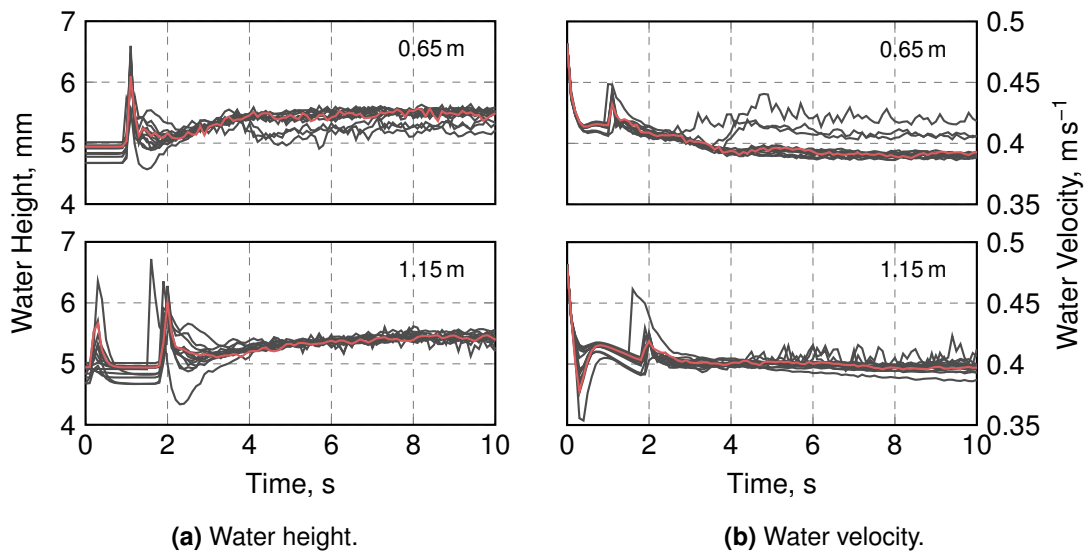
the deviations of the simulations can be found in Table 8.10. The relative error was calculated by averaging the data of the last two seconds of the simulation ( $t = 8-10$  s) and subtracting it from the min/max value found among all simulations (except for water volume, where 0.5x, 1x and 2x\_Z0.5 were treated separately). This is represented by the following Equation (8.4) for a given property  $p$  (averaged over time:  $\bar{p}$ ) and a case  $i$ :

$$\varepsilon_{p,i} = \frac{|\bar{p}_i - \max(\min \bar{p}, \max \bar{p})|}{\bar{p}_i} \quad (8.4)$$

Since the case with a refinement only in  $z$  direction still retains a low number of cells in the mesh, while at the same time maintaining errors in the order of the other cases, it was assumed to represent a sufficiently refined mesh. This could also be found when evaluating the water height and velocity (Figure 8.25). Since the channel was pre-filled to speed up convergence, in the initial state the flow profile has not yet developed. As a result of this, two peaks can be observed in the simulation. The first peak, at  $t \approx 0.5$  s at the end of the channel ( $y = 1.15$  m) corresponds to the outlet zeroGradient Boundary Condition (BC). Since it requires a developed flow profile, a temporary accumulation of water can be found until the computation converges to a physical behaviour. The second peak represents a wave of water that probably occurs due to the water at the surface accelerating faster than below, due to the decreased friction. It wanders along the channel and can thus be observed at  $t \approx 1$  s at the first sampling position and  $t \approx 2$  s at the channel's end. However, for water height as well as water velocity the selected mesh yields results very similar to the ones that could be achieved with highly refined meshes in quasi-steady state, and was thus considered sufficient. It was consequently used for all simulations described in the following sections. Details on the mesh can be found in Section 6.1.2.

**Table 8.10** The cases evaluated for the mesh independence study. Naming conventions:  $\langle factor \rangle \times$  signals the number of cells used in this case relative to the reference case (1x). The appendix  $\_ (X/Y/Z) \langle factor \rangle$  signals a refinement of cells in the named dimension(s) only. Errors were determined by the maximum deviation for all cases that showed similar behaviour. The case marked (\*) is the option chosen for all further simulations. Highlighted are the highest and lowest errors estimated for each column.

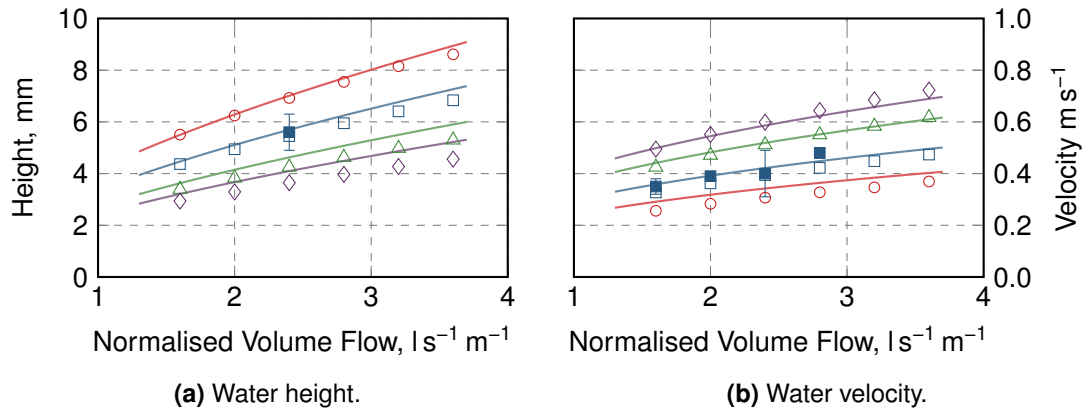
Case	$\max \varepsilon_h, \%$	$\max \varepsilon_v, \%$	$\max \varepsilon_V, \%$
0.5x	7.14	7.46	6.07
1x	4.62	4.76	5.62
1x_Z2*	5.43	7.06	2.14
2x	5.72	7.16	1.64
2x_X0.5	5.37	6.85	2.22
2x_X2	5.27	6.68	1.31
2x_Y0.5	5.39	6.98	1.67
2x_Y2	5.06	6.75	2.28
2x_Z0.5	4.86	4.75	5.45
2x_Z2	6.66	7.33	1.72
3x	6.25	7.86	1.65
4x	5.87	7.23	1.59
5x	6.64	7.94	2.23
6x	6.47	7.68	1.88



**Figure 8.25** Water height and velocity on the channel over time for different meshes. The channel was pre-filled to height  $h$  and set to velocity  $v$  as predicted by the Manning equation (Equation (7.11) and (7.9)) to speed up convergence. After the start of the simulation, the fluid height and velocity adjust according to the balance in the Navier-Stokes equations. Upper graphs show the data at  $y = 0.65$  m, lower graph at  $y = 1.15$  m. The selected case is marked (—).

### 8.2.2. Validation

To validate the channel simulations, the results of different operational parameters were compared to the Manning equation. To ensure the validity of this rather theoretical prediction, furthermore measurements of the fluid height and velocity for a channel sloped at  $1^\circ$  were performed. The results of these measurements, the empirical, and the simulative predictions can be seen in Figure 8.26.



**Figure 8.26** Water height and velocity in the channel as predicted by the Manning equation (Equations (7.11) and (7.9), solid lines) in comparison with the data determined in the CFD simulation (empty symbols) and measurements (filled). Shown are the results for channel angles of  $0.5^\circ$  (—, ○),  $1^\circ$  (—, □, ■),  $2^\circ$  (—, △), and  $3^\circ$  (—, ◇)

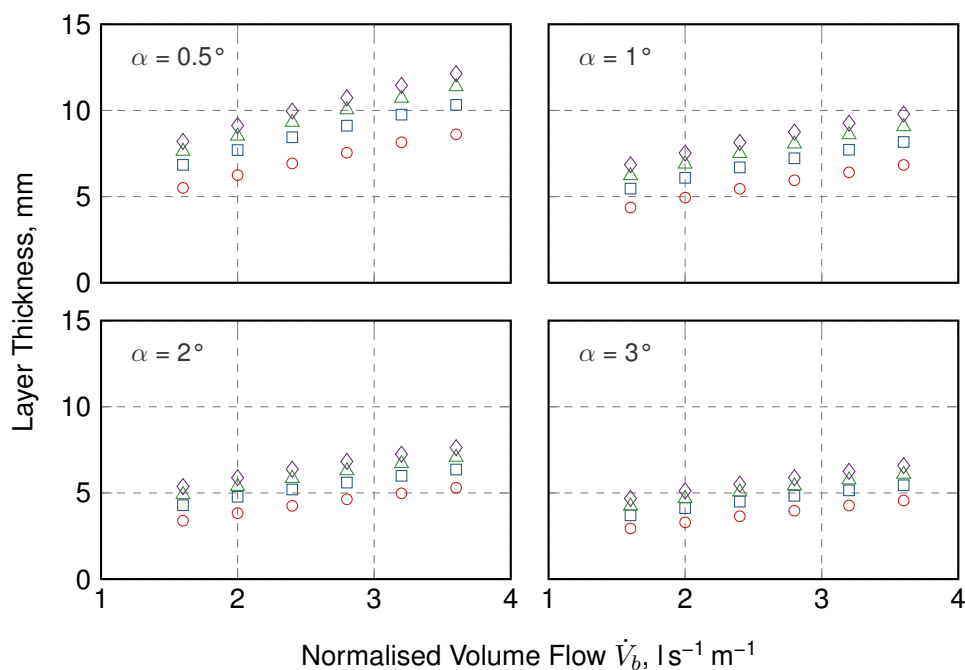
Although Oswald [1988] determined a Manning coefficient of 0.008 for a smooth plastic surface, in this study a coefficient of 0.010 was found to describe the experimental (and simulative) data best and is used in the graphs.

With the plotted data, the validity of the Manning equation for the channel used can be shown for an angle of  $1^\circ$ . Since the Manning equation has been compared with empirical data for a high number of observed open channels, it can thus be inferred that the prediction at different channel slopes closely matches the reality, too. As a result of this, the validation of further channels by this empirical correlation can be presumed acceptable. It can be shown that the simulation deviates at most by 12.7% in the height (at high angles and volume flows) and by 12.0% in velocity (at low volume flows and angles) from the values predicted by the Manning equation.

### 8.2.3. Layer Thickness

To optimise the cultivation in the channel, the layer thickness should be restricted to given limits, depending on the optical properties and light growth kinetics of the strain (compare Figure 5.7). Furthermore, due to high density growth or extracellular accumulation of products, the viscosity of the algal suspension can change during the cultivation or depending on the operational conditions.

This effect already has to be considered during the construction of the reactor, since the slope of the channel has a major influence on the resulting flow properties. Figure 8.27 shows the layer thickness computed for different angles and volume flows: The angle of the channel restricts the height of the suspension to a spread of approximately 5 mm for the common



**Figure 8.27** Layer thickness in the channel for different viscosities and channel angles: For sloped channels with an angle  $\alpha = 0.5\text{--}3.0^\circ$  the resulting layer thickness is plotted depending on the volume flow  $\dot{V}_b$  and the viscosity of the fluid ( $\mu = 1 \text{ mPa s}$ :  $\circ$ ,  $\mu = 4 \text{ mPa s}$ :  $\square$ ,  $\mu = 7 \text{ mPa s}$ :  $\triangle$ ,  $\mu = 10 \text{ mPa s}$ :  $\diamond$ ).

specific volume flows  $\dot{V}_b$ . It can also be seen that during a cultivation in which the viscosity of the suspension changes from  $1 \text{ mPa s}$  (water) to  $10 \text{ mPa s}$ , the layer thickness increases by  $2\text{--}4 \text{ mm}$  depending on the channel's slope. This change can only partially be compensated by an adjustment of the volume flow.

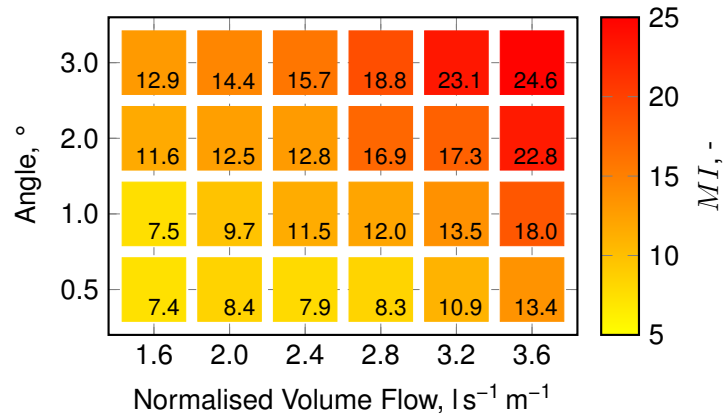
Regarding the layer thickness, another aspect apart from the viscosity has to be considered: Even for algae where the viscosity is not influenced significantly it can be assumed that the optical density increases notably with concentration. To keep the light available to all cells constant, again the height of the suspension on the channel has to be adjusted. The diagrams shown in Figure 8.27 can thus give an orientation which angle of the channel allows to adjust the height to the desired irradiance in the fluid, i.e. layer thicknesses.

#### 8.2.4. Vertical Mixing<sup>1</sup>

Evaluating the mixing index introduced in Section 7.5.2 for the different channel parameters listed in Table 8.8, matrices can be generated to compare the results. The computed mixing indices  $MI$  for cases with a viscosity of  $1 \text{ mPa s}$  can be seen in Figure 8.28. As could be expected, mixing directly improves with an increased slope of the channel or a higher volume flow: Between the lowest settings ( $0.5^\circ$ ,  $1.6 \text{ l s}^{-1} \text{ m}^{-1}$ ) and the maximum ( $3.0^\circ$ ,  $3.6 \text{ l s}^{-1} \text{ m}^{-1}$ ) a three-fold difference in the mixing index can be observed.

This would suggest that the optimal channel configuration is comparable to a waterfall, with

<sup>1</sup> Parts of this chapter have already been published in: Timm Steffen Severin, Andreas Apel, Thomas Brück and Dirk Weuster-Botz [2018]. 'Investigation of Vertical Mixing in Thin-Layer Cascade Reactors using Computational Fluid Dynamics'. In: *Chemical Engineering Research and Design* [2018]. ISSN: 02638762. DOI: 10.1016/j.cherd.2018.01.036



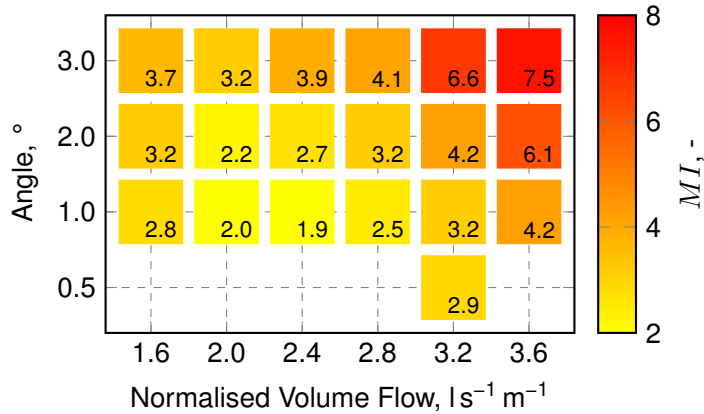
**Figure 8.28** Matrix of the mixing indices  $MI$  achieved with varying angles and volume flows of the fluid in the channel for a viscosity of  $\mu = 1$  mPa s. Corresponding values are shown, too. Diagonal “bands” of equal mixing occur which indicate that e.g. an decreased slope can be compensated by increasing the volume flow.

very high volume flows and steep slopes. However, for production systems, further effects have to be considered. These include —among others— power consumption, construction costs and light availability. Especially for low cost products like biofuel, both former restrictions tighten the bounds of feasible designs substantially.

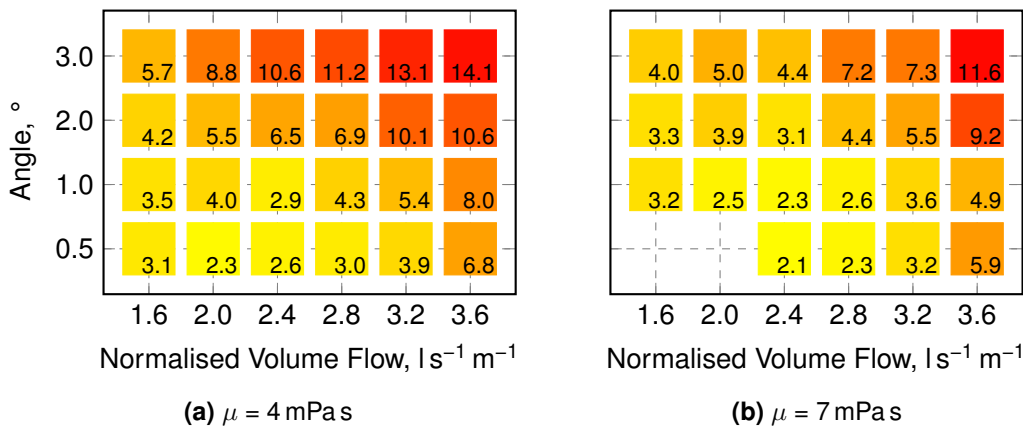
Still, the analysis might prove helpful when designing photobioreactors. Parameters like the reactors slope will often be dictated by terrain properties. Within the limits of photosynthetic activity (layer thickness) the mixing index gives a criterion which allows to compare different channels operating at identical mixing behaviour. To scale pilot plants or transfer existing systems between sites, this can give a useful estimation of reactor performance under these changes. It also allows to change operating conditions along the diagonal “bands” of equal mixing, which can be seen in Figure 8.28.

This can further be extended for different viscosities, as will occur in higher density cultivations. As thin-layer cascade reactors support concentrations of up to  $50 \text{ g l}^{-1}$  and are subject to temperature changes of the environment, the viscosity can increase significantly. The evaluation of the channel for a viscosity of  $\mu = 10$  mPa s can be seen in Figure 8.29. To achieve the same mixing effects as with dilute suspensions, increased volume flows are necessary. This evaluation can therefore be used to adjust the operating conditions of an algal reactor with increasing biomass concentration in batch cultivations. As mixing is correlated to turbulence and thus sedimentation [Doucha and Lívanský 1995], ensuring the mixing index to not decrease would also be advantageous to decrease the loss of microalgal cells in a layer on the channel ground.

The mixing properties for viscosities between these two extrema can be seen in Figure 8.30 ( $\mu = 4$  mPa s and  $7$  mPa s). Here it can be seen that the major decrease in mixing already occurs for very low changes in viscosity, as the decrease of the  $MI$  from  $1$  to  $4$  mPa s is roughly twice as big as the decrease from  $4$  to  $7$  mPa s. It can thus be concluded that an increase in viscosity due to algal growths or secretion of extracellular substances significantly reduces mixing when starting at very low cell concentrations (water-like fluid). However, if a high density cultivation is desired, further changes in viscosity will not affect the mixing



**Figure 8.29** Matrix of the mixing indices  $MI$  achieved with varying angles and volume flows of the fluid in the channel for a viscosity of  $\mu = 10$  mPa s. Points not drawn were found not to behave quasi-stationary in the simulation.



**Figure 8.30** Matrices of the mixing indices  $MI$  for  $\mu = 4$  mPa s and  $\mu = 7$  mPa s. Points not drawn were found not to behave quasi-stationary in the simulation.

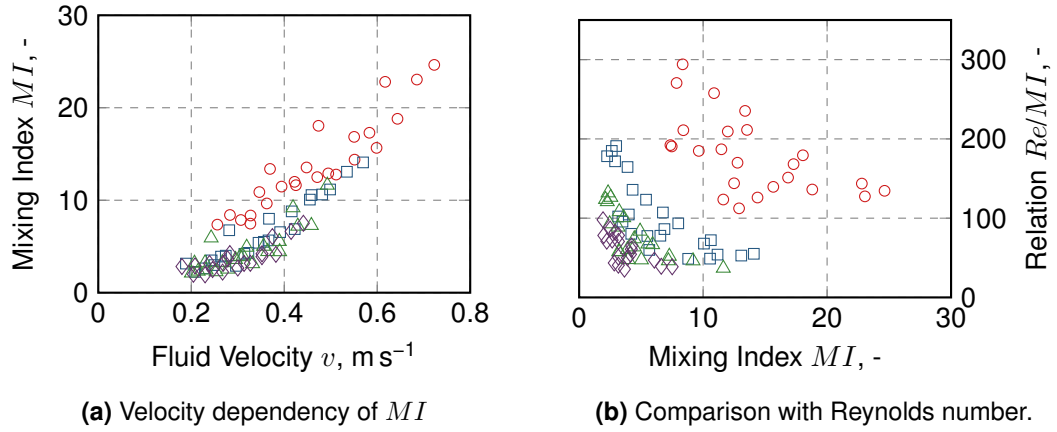
significantly.

The result shown above indicate that with increased fluid velocity (either due to slope or volume flow) better mixing can be achieved. This relation is plotted in Figure 8.31a, where the mixing index  $MI$  is displayed versus the fluid velocity: While at low velocities ( $v \approx 0.2$  m s<sup>-1</sup>) no significant effect on the mixing index can be observed, increasing velocities lead to an approximately proportional dependency ( $MI \propto v$ ).

Since the flow velocity is also one parameter of the Reynolds number, one might assume that  $Re$  already reflects the flow properties, and thus should be compared to the mixing index. This can be seen in Figure 8.31b. It shows the Reynolds number for all channel simulations in relation to the calculated mixing indices. The grouping by viscosity indicates that the  $Re/MI$  relation can be arranged in bands decreasing with increasing mixing indices.

It can thus be followed that the Reynolds number is not sufficient to describe the vertical mixing in a thin fluid layer as described in this thesis. Considering the data of the Reynolds number over the mixing indices more closely (not shown), it can further be seen that higher mixing indices can also be achieved without a substantial change in the Reynolds number, but





**Figure 8.31** Velocity dependency of and relation of the Reynolds number (Equation (7.12)) to the mixing index  $MI$  grouped by viscosity ( $\mu = 1 \text{ mPa s}$   $\circ$ ,  $4 \text{ mPa s}$   $\square$ ,  $7 \text{ mPa s}$   $\triangle$ ,  $10 \text{ mPa s}$   $\diamond$ ).

with an increased slope of the channel. This is in accordance with the Manning equation: If the height and velocity predictions by the Manning equation are substituted in the Reynolds number,  $Re$  becomes independent of the channel's slope (Equation (8.5)) and only depends on the volume flow  $\dot{V}$ , the channel width  $b$ , and the fluid's viscosity  $\mu$ .

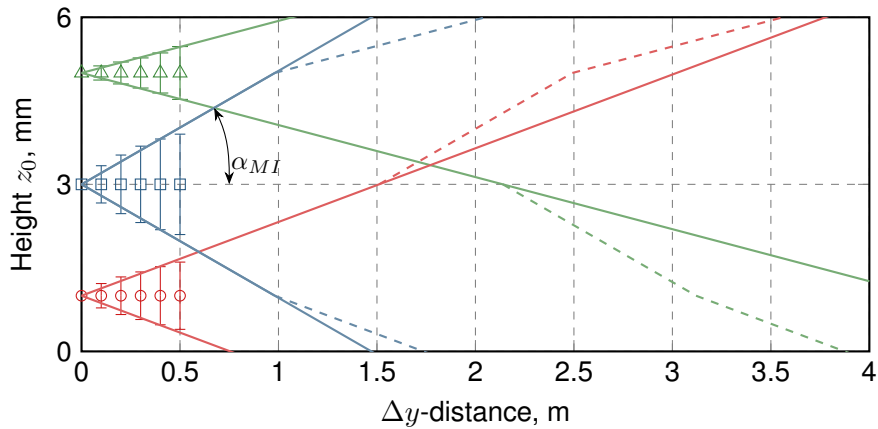
$$Re_{Manning} = \frac{\dot{V}}{b\mu} \quad (8.5)$$

An interpretation of the mixing index can be seen in Figure 8.32. If the scaling term for the mixing index is set to unity, the  $MI$  describes the angle  $\alpha_{MI}$  with the relation  $\alpha_{MI} = \arctan(MI)$ . It thus gives an estimate of the distance a particle has to travel before it has crossed the complete height of the fluid layer, e.g. the transition from photoinhibition (top layer) to photolimitation (ground). Assuming a constant angle for each starting height  $z_0$  (solid lines) would require a channel length of more than 4 m for a particle starting in the photoinhibited zone to reach the photolimited. However, as can be seen, the mixing index varies depending on the start height  $z_0$ . If this is taken into consideration for the data shown (dashed lines), a channel length of 4 m is sufficient for one transition.

If sufficient data can be extracted from the simulation to obtain a function  $MI = f(z)$ , the length  $\Delta y_{0 \rightarrow h}$  for one crossing of the fluid height  $h$  can be estimated by Equation (8.6).

$$d\Delta y = \frac{dz}{MI(z)} \Rightarrow \Delta y_{0 \rightarrow h} = \int_0^h MI(z) dz \quad (8.6)$$

It is, however, advisable to set the integral's limits away from the boundaries, since here the stream tracers are limited by physical effects, i.e. are the deviation possible is bounded by the wall and the water surface, and the  $MI$  is very small. Using a lower limit of 0.5 mm (photolimitation) and an upper limit of 5.5 mm (photoinhibition) for the data shown here, this results in a length of  $\Delta y_{0 \rightarrow h} = 4.6 \text{ m}$ .



**Figure 8.32** Different scenarios for the mixing of virtual particles (compare Figure 7.12). Shown are the spread of particles starting at three different heights (1 mm, 3 mm, and 5 mm). The spread is represented by the standard deviation of all computed stream tracers. Using a linear extrapolation of the spread (solid lines), it can be estimated how long a particle takes to cross the complete fluid height. A slightly different scenario evolves when the change of vertical velocity is considered depending on the vertical position in the channel. Assuming the change only to occur at the measurement positions shown (dashed lines), the  $\Delta y$  distance required for the fluid to reach the surface or the ground can be read more realistically. Data shown for a case with a volume flow of  $\dot{V}_b = 2.4 \text{ l s}^{-1} \text{ m}^{-1}$  and an angle of  $\alpha = 1^\circ$ .

### 8.2.5. Conclusion and Discussion

In this chapter the channel simulation was analysed. The mesh independence for the selected mesh could be proven, and the results were validated with experimental data. It could further be shown that the simulation results agree very well with the predictions of the Manning equation for a Manning coefficient of  $n = 0.010$ . Based on these results, the simulation could be presumed valid in a comparison with the theoretical results from the Manning equation, even for parameters where no measurements were available. However, although this empirical correlation for open channels is well established, further measurements could help confirm the Manning coefficient of 0.010, and thus also allow simple but fast predictions on the flow behaviour without numerical simulations.

Following the validation, the influence of different operational and fluid parameters on the layer thickness was investigated. This can and should be used for the construction as well as the operation of the reactor: During the design process, pre-operational, the parameter of the channel's slope has to be defined. This parameter directly limits the range of achievable heights of the microalgal suspension on the channel. Since the height immediately influences the photosynthetic activity of the microalgae, it should be chosen according to a prior measured kinetic for the selected strain. During the operation of the reactor, this height will change depending on the viscosity of the fluid, but can also be adjusted by changing the volume flow. To complete these analyses, a coupling of these results with economical (e.g. investment costs for correction of the channels slope) and energetic aspects (e.g. pumping to achieve the desired volume flow) would be useful.

The mixing index  $MI$  can also be used as a design tool to transfer insights of a pilot plant to a real system. However, its applicability is restricted to the range of operational parameters

dictated by the microalgae's kinetics, i.e. the layer thickness. Further measurements should be made to compare different reactor systems. To obtain these results would — due to the multitude of influencing parameters on algal growth — require significant experimental effort. An alternative that could be investigated during a process could be to develop a correlation with sedimentation of particles (cells). Should the vertical mixing be characterised further, the integration of an irradiance model in the CFD simulation should be considered. This would allow to precisely compute the incident irradiation on simulated particles, which could be validated by light-sensitive chemicals.

Exploiting the mixing index further, the length of the reactor that is required for a particle to cross the complete fluid height could be estimated. For a typical reactor setup ( $\alpha = 1^\circ$  and  $\dot{V}_b = 2.4 \text{ l s}^{-1} \text{ m}^{-1}$ ) this length is longer than 4 m (approximately 4.6 m). It could thus be discussed, whether a reactor with a channel length of “only” 4 m is sufficient to investigate the effects of light intensity on the microalgal suspension. In contrast to this, it can be assumed that for economical reasons the surface of the channel is significantly rougher than assumed here and can even have bumps. This would increase the vertical mixing significantly and might mitigate the problems that would arise with too little vertical mixing. If this raised vertical mixing proves to be the case, a reactor length of less than 4.6 m for the technical scale would give more realistic results, since it “simulates” an increased mixing on the channel by redistributing algal cells in the inlet or flow reversal tank in short intervals.

Based on the previous evaluation, combined with fluid velocities, the mixing index could further be used to characterise the light availability with regard to light-dark cycles. However, Powell et al. [1965] concluded that turbulence will only produce “favourable light-dark cycles” for a small number of cases, and an adaptation of the mixing would fall back to a problem of design complexity and power consumption. Thus, this aspect has not been investigated further in this project.

### 8.3. Flow Reversal Tank

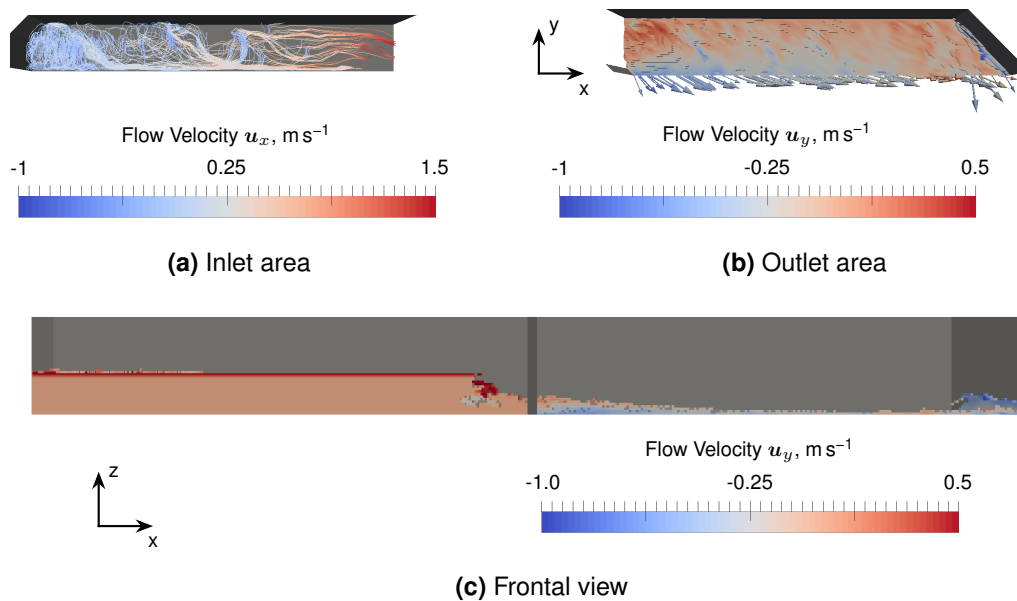
The flow reversal tank has the purpose of redirecting the fluid flow from the upper channel to the lower channel. It thus has to be designed to achieve an even fluid distribution on the following channel. For long channels, however, it can be replaced with just another retention tank and inlet unit. Thus its importance is limited to few cases, where only short channel lengths are available (e.g. experimental units) or investment costs for an inlet tank are too high. Since these cases are considered of less importance, this chapter will only give a basic overview over the optimisation strategies for the flow reversal tank.

First the initial configuration that has been built for the reactor will be introduced, and its shortcomings shown. Following this, simple geometric modifications will be shown that can be achieved without much effort. For cases where the equal distribution of the suspension on the channel is crucial, further baffles that can be added to the tank will be introduced and an optimised solution will be presented.

All simulations shown were computed with a volume flow of  $2.5 \text{ l s}^{-1}$  if not noted otherwise.

### 8.3.1. Initial Configuration

The initial configuration of the flow reversal tank can be seen in Figure 8.33. The tank is not sloped and has a depth of 13 cm. The outlet area has a width of 1 m.



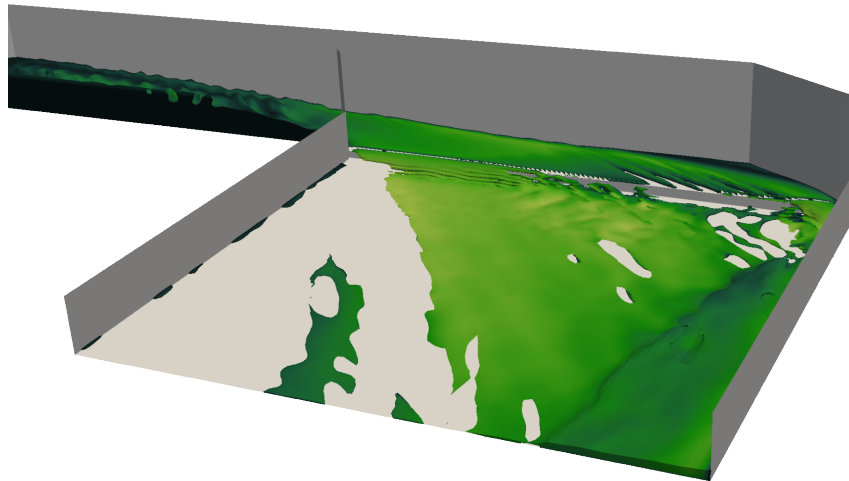
**Figure 8.33** Flow pattern in the initial configuration of the flow reversal tank with a volume flow of  $3.6 \text{ l s}^{-1}$ . The stream tracer in the inlet area (a) visualise the circulation on the left side. The velocity's  $x$ -component at the transition to the channel can clearly be seen by the arrows in the outlet region (b). The frontal view (c) shows the vertical water distribution in the tank.

This visualisation clarifies a number of problems that occur in this setup:

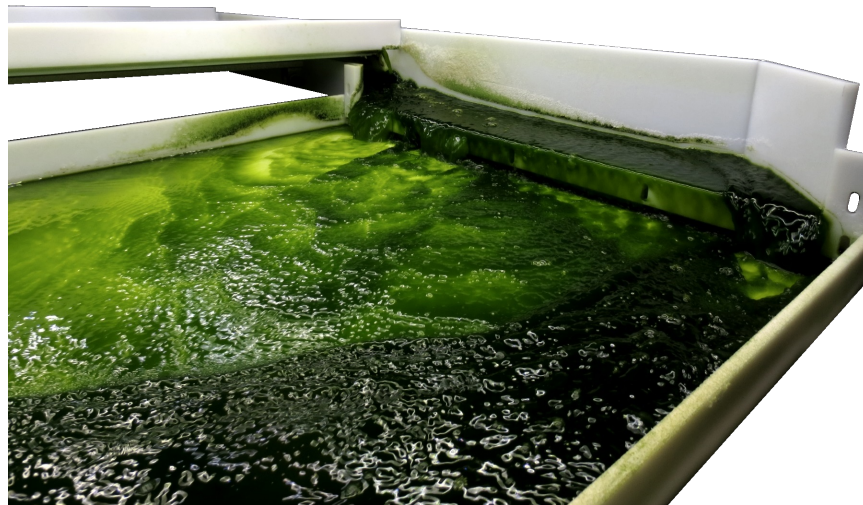
- Figure 8.33a shows the streamlines for the fluid in the inlet area. It can be seen that in the left half vortices occur, and the water barely flows in direction of the outlet. In the right half, the fluid is accelerated to approximately  $1.5 \text{ m s}^{-1}$  in  $x$ -direction before entering the outlet region.
- In the outlet region, Figure 8.33b, the fluid is coloured by its velocity towards the outlet. Arrows show the direction of the flow directly at the boundary. This highlights the dominating  $x$  component of the velocity vector at the interface to the channel. Furthermore, some fluid piles up on the right side of the tank, where it hits the wall, which also indicates that the  $x$ -velocity inside the tank is higher than necessary.
- The frontal view gives an overview over the water volume in the tank. Significant amounts of water accumulate in the inlet region. This should be minimised for an optimal  $S/V$  ratio of the reactor.

The results of these observations can also be seen in Figure 8.34a, where the channel after the tank has been added to the simulation. Compared to Figure 8.34b this results also gives a qualitative comparison of the simulation and the experimental unit. All effects mentioned above can be found in the photograph. The water level in the inlet area is increased, and barely any suspension leaves the tank in the centre of the channel, instead piling up on the right side, leading to a very uneven distribution on the channel.

Evaluating the water height on the channel quantitatively, the increased water height on the



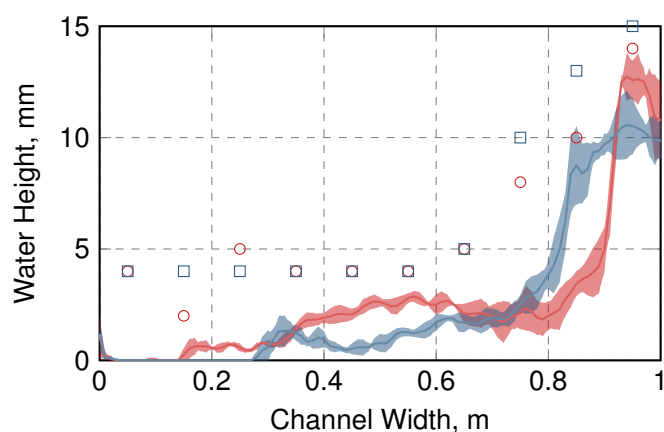
(a) Simulation



(b) Experimental

**Figure 8.34** Comparison of the simulation and an experimental unit of the flow reversal tank's initial configuration. The gathering of water on the right side can be seen in both cases, while the left part of the channel contains nearly no algal suspension. Simulated surface extracted with ParaView at  $\alpha = 0.5$ ; colours arbitrary.

outermost side can be seen very clearly (Figure 8.35). The simulation furthermore shows that the water in the middle of the channel still has a significant  $x$ -component in its velocity, since the water height in the centre of the channel decreased the further from the flow reversal tank the data is sampled, while the area under the curve at the right side increases. The measurements supplied can again be used as qualitative comparison. They show the same profile, although all measurements indicate a higher water level than simulated. A reason for this might be inaccuracies during the measurement. Furthermore, the exact volume flow could not be measured experimentally, thus adding to the uncertainty. However, this could also be caused by an overestimation of the fluid velocity (which was not measured) in the simulation.



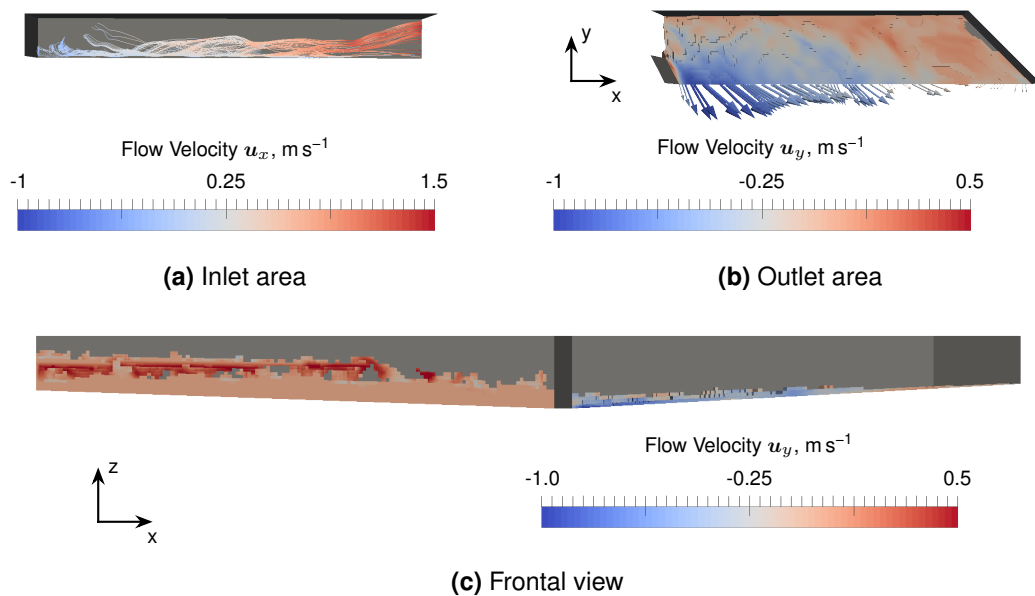
**Figure 8.35** Water height on the channel for the flow reversal tank in its initial configuration. The increased water height on the right side of the channel can clearly be seen for measurements (marks) as well as simulation (solid line) at both evaluated positions, 37 cm (○, —) and 67 cm (□, —) after the flow reversal tank. Simulation results averaged over time and standard deviation around mean highlighted (■, ■).

After the shortcomings of the flow reversal tank have been pointed out, and with the agreement of the simulation and the experimental data shown, modifications of the tank should be suggested. The following section will thus introduce simple modifications that have been compared for their effect on the fluid flow.

### 8.3.2. Geometric Modifications for Improved Flow Patterns

A few simple, geometric modifications of the flow reversal tank were simulated to study their effect on the flow. Figure 8.36 shows the tank with sloped grounds in the inlet as well as the outlet region. Increasing the angle of the ground in both areas can have multiple advantages. Firstly, it decreases the volume that is stored in the tank. Secondly, the slope increases the tendency of the water to flow to the lower regions, i.e. leave the inlet region faster. It possibly also improves the distribution of water on the subsequent channel. This can be seen in Figure 8.36b: The flow at the outermost wall of the outlet is directed in positive  $y$ -direction, and thus flows away from the outlet's edge. This indicates a large stagnant area with little fluid exchange and is thus undesirable, although the goal of having an increased flow on the inner side of the channel can probably be achieved. It can be discussed if a more detailed analysis of the effect of an angle could help to find a balanced solution, yet this would also require a

sensitivity analysis of the setup depending on a change of the flow properties (physical, e.g. viscosity, or technical, e.g. volume flow).



**Figure 8.36** Flow pattern in the flow reversal tank with the ground sloped at  $3^\circ$  with a volume flow of  $3.6 \text{ l s}^{-1}$ . The stream tracer in the inlet area **(a)** have a stronger tendency to flow into the outlet region **(b)**, where the water is slowed by the slope. This can also be seen in the frontal view **(c)**, where barely any water reaches the right side. As a result of this, most of the water leaves the tank on the inner side of the channel.

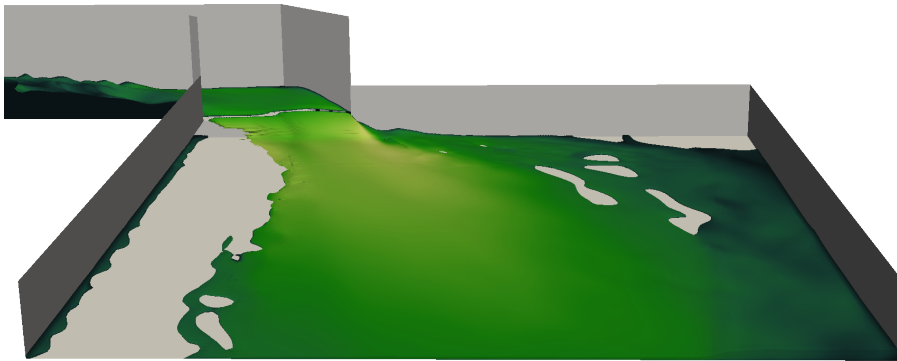
Further modifications included a central wall with a length of 1 m (data not shown). It was positioned to be half in inlet and outlet region, separating the flow reversal tank in  $y$ -direction in two regions. While the idea was to allow the left half of the inlet area to flow a separate path, it could be shown the the additional obstacle increased vorticity in the left half of the inlet region. Furthermore this increases the velocity of the water reaching the outermost, right wall, resulting in higher water levels here. Still, a slightly improved velocity distribution at the interface to the channel could be achieved.

As an opposite approach the tanks depth was increased by 8 cm (data not shown), to decrease the interaction of the fluid with itself and thus undesired vorticity generation. Unfortunately, due to the high flow velocities, this effect could not be observed.

A final approach to force more fluid on the left side of the channel was a shortened outlet of the tank. As it showed promising results, this case was compare to a shortened outlet in the real unit (Figure 8.38). It can be seen that a significantly improvement to the initial configuration can be achieved.

This can also be seen in the water profile (Figure 8.38). The highest peaks, in the initial configuration 15 mm, could be reduced by more than 5 mm, with an increased water level in the central part of the channel. Although still a flow to the right side can be observed from 37 cm (—) to 67 cm (—), the resulting surface also flattens over the complete width.

It could thus be shown that simple geometric modifications can help to improve the flow behaviour of the flow reversal tank. Still, high variations in water height on the channel can



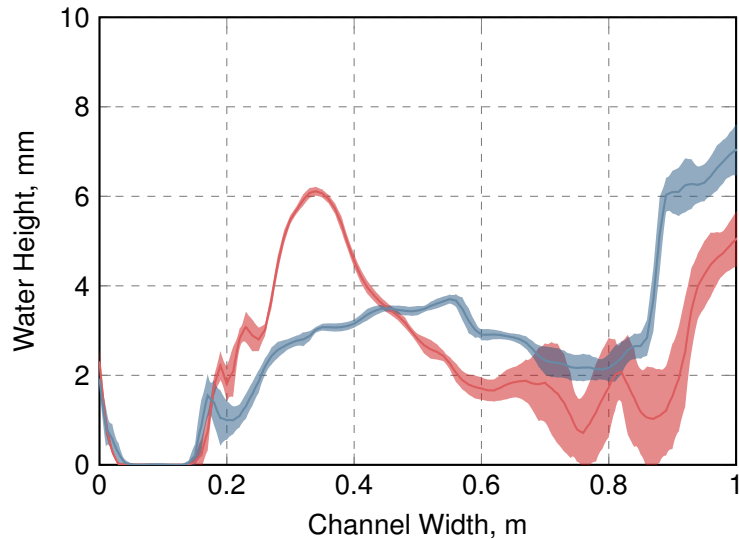
(a) Simulation



(b) Experimental

**Figure 8.37** Comparison of the simulation and an experimental unit with a short outlet. The suspension still has a tendency to flow outwards, however a significantly improved coverage of the channel ground can now be achieved. Simulated surface extracted with ParaView at  $\alpha = 0.5$ ; colours arbitrary.



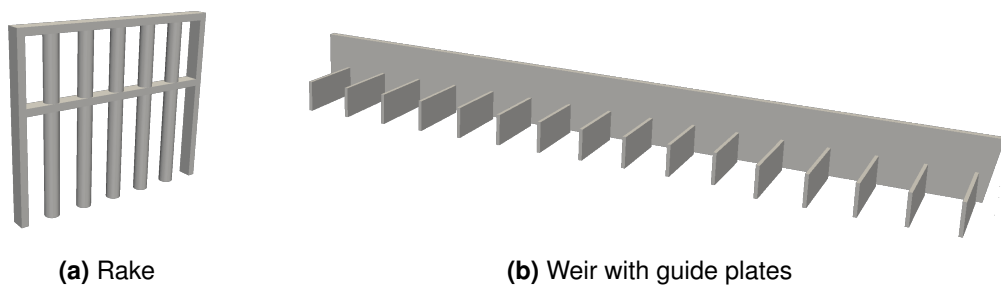


**Figure 8.38** Water height on the channel for the flow reversal tank with a short outlet. The water height on the right side is significantly lower (more than 5 mm) for both evaluated positions 37 cm (—) and 67 cm (—) after the flow reversal tank. Simulation results averaged over time and standard deviation around mean highlighted (■, ■).

be expected. For cases, where this is not desired, more complex constructions are needed. Examples for this will be presented in the next section.

### 8.3.3. Additional Baffles

Figure 8.39 shows two types of baffles that could be introduced to the flow reversal tank/channel combination. The rake, Figure 8.39a, can be used to slow the suspension in the flow reversal tank itself. In contrast the weir (Figure 8.39b) is positioned at the transition to the channel and aims at an even distribution of water on the channel. It can be fitted with guide plates that restrict the lateral velocity component.



**Figure 8.39** Additional elements used to modify the flow behaviour. The rake (a) can be placed inside the flow reversal tank in order to slow the fluid down. The weir (b) is positioned directly after the tank's outlet on the channel. It leaves a small gap to the channel ground to allow draining onto the channel. The number of elements (cylinders or guide planes) as well as length or diameter could be changed for both baffles.

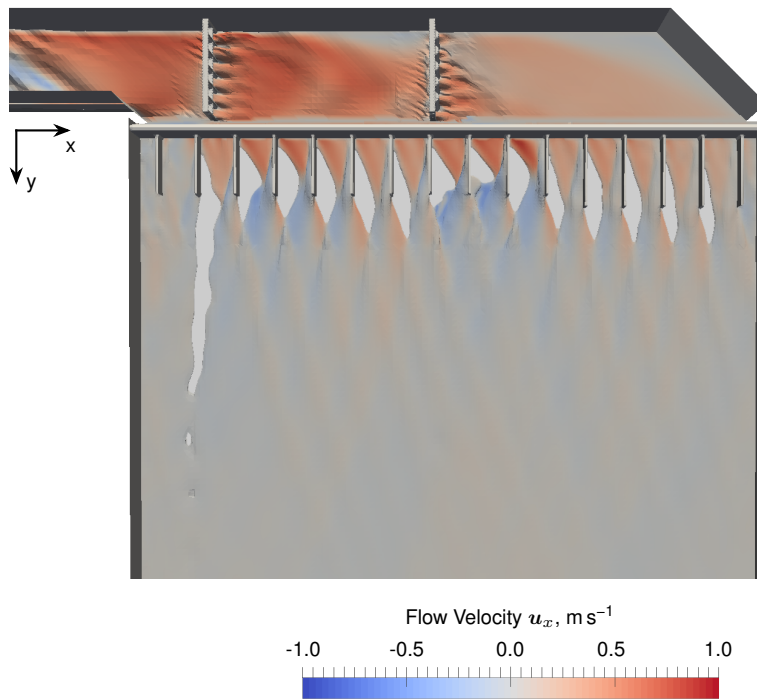
Both elements were parametrised and different options compared simulatively. Results were compared based on the standard deviation of the water height on the channel  $\sigma(h)$ :

*Rake* For the rake, the diameter of the cylinders could be modified. In a parameter study

employing a rake in the centre of the tank and a weir with 16 guide plates, the cylinders diameter was varied between 3 and 6 mm. The optimum diameter was found to be 5 mm with a standard deviation of  $\sigma(h) = 0.34$  mm (maximum 0.7 mm).

*Weir* The weir could be constructed with varying number of guide plates (10–16) of different length (6–16 cm). Using 14 guide plates resulted in a deviation of  $\sigma(h) = 0.48$  mm (maximum 0.8 mm). In a separate comparison using 16 guide plates, with the rake developed before, a length of 10 cm was found to be optimal ( $\sigma(h) = 0.34$  mm, maximum 1.85 mm).

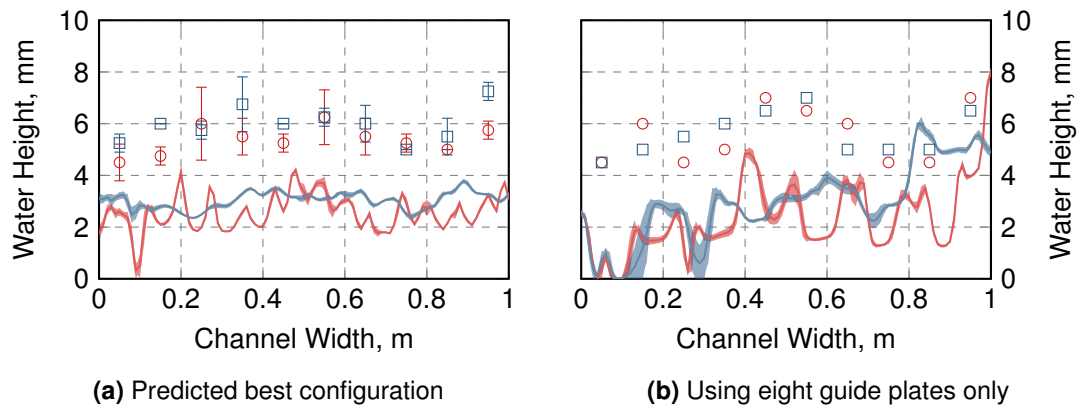
Using further evaluations (not described), an optimal configuration could be found employing two rakes on the flow reversal tank (in the centre of the complete tank as well as in the centre of the outlet region). The weir was designed with 16 guide plates, with a length of 10 cm for the innermost 10 and 12 cm for the remaining 6. The  $x$ -component of the velocity for this configuration is shown in Figure 8.40. While the slowing effect due to the rakes is not obvious in this view, the redirection of the fluid via the guide plates can be seen clearly.



**Figure 8.40** Top view on the predicted best configuration of the baffled flow reversal tank. The fluid surface was exported with ParaView at  $\alpha = 0.5$  and is coloured by the  $x$ -component of the fluid's velocity.

This very complex setup yielded a water surface on the channel as shown in Figure 8.41a. It can be seen that the predicted water surface is very even over the complete channel width. Again, a qualitative agreement with the measurements can be observed.

During the experimental work, an alternative setup using only the weir and eight guide plates was found to give similarly good results. These can be found in Figure 8.41b. While the measurements show a relatively even surface, the simulated water height over the width varies strongly. As the measurements are off for every evaluation, this graph indicates the simulation used might lack precision in more complex flow patterns.



**Figure 8.41** Comparison of the water height measurements (markers) and results (solid lines) for the best prediction and an experimentally determined optimum baffled tank. Shown are the water height 37 cm (○, —) and 67 cm (□, —) after the outlet of the tank. The simulations results were averaged over time, standard deviation has been highlighted (■, ■).

### 8.3.4. Conclusion and Discussion

This chapter presented some alternative designs for the flow reversal tank. It could be shown that simple modifications already allow a measurable improvement of its flow behaviour. If a very even distribution of the suspension on the channel is desired, more complex modifications like the introduction of baffles and a weir are required. However, the modifications also emphasise the complex nature of the tank: A high number of parameters is available that can be investigated for their influence on the fluid flow. Thus, the simulative investigation of the tank is not yet exhausted.

Yet, for the purpose of this thesis, the options shown are considered sufficient. Their application can easily be made in pilot size reactor systems. For large, industrial-size reactors the use of straight channels only connected via inlet tanks can assumed to be the more economic variant, since it reduces the number of different units required. Furthermore, if designed accordingly, no increase in investment costs are expected, since the same height difference can be achieved with two channels as well as with one.

## 9. Conclusion and Outlook

The current state of technology is the production of biofuels of the first and second generation: Crops or residues are converted in biotechnological processes to fuel. Thereby, they either compete with the harvest of foods or require intensive processing. Third generation biofuels, created from energy rich lipids stored in microalgae, offer a promising alternative. For an economically feasible process, however, operational and investment costs must be reduced.

Current open photobioreactor systems show productivities of at most  $15 \text{ g m}^{-2} \text{ d}^{-1}$  and biomass concentrations of less than  $1.5 \text{ g l}^{-1}$  [Grobbelaar 2009b; Pulz 2001]. These low biomass yields and dilute suspensions require significant effort to be harvested. To increase the efficiency of reactors, the utilisation of light must be optimised. A photobioreactor concept that takes especially the light availability in a microalgal suspension into account is the Thin-Layer Cascade (TLC). It is constructed from an inlet tank, a cultivation area in form of a channel with very low layer thicknesses, and a retention tank through which the microalgal suspension circulates sequentially. Although here biomass concentrations of up to  $50 \text{ g l}^{-1}$  at productivities of up to  $50 \text{ g m}^{-2} \text{ d}^{-1}$  [Apel, Pfaffinger et al. 2017; Masojidek, Kopecky et al. 2011] can be achieved and harvesting costs can be greatly reduced, its construction from several independent units and high volume flows leads to an increase in operational and investment costs. Thus, Computational Fluid Dynamics (CFD) models were employed in this thesis to assist in finding a cost and energy efficient design for the various photobioreactor components.

The main cultivation area among the reactor units is a channel. On this channel the layer thickness of the suspension is in the order of a few millimetres. As a result microalgae can grow under photosaturation, i.e. under optimal growth conditions. However, this requires sufficient vertical mixing (in the direction of the light source). If this can not be ensured, photoinhibition and -inactivation can occur at the surface, i.e. a temporary or permanent destruction of the microalgal ability to harvest light. At the channel's ground, a region of photolimitation may develop. Here insufficient light is available, prohibiting maximum growth rates. Since the vertical mixing could not be measured directly, CFD simulations were employed to characterise the fluid behaviour in the thin layer.

The fluid enters the channel by an inlet tank. The algal suspension is pumped into the tank from a retention tank, is distributed by a baffle, and leaves it via an overflow weir. This inlet tank has to ensure an even distribution of the fluid on the channel. To achieve this, operational and geometric parameters should be varied and their effect on the fluid layer height on the weir should be investigated. Furthermore, the inlet tank requires the major part of the energy consumed by the system, since it requires the pump to overcome hydrostatic and kinematic pressure as well as pressure loss due to friction. An estimation of the significance of the different contributions on the pressure should thus be generated to highlight optimisation potential.

To redirect the flow for a circulation, a flow reversal tank can be placed between two channels. It gathers the fluid from an upper channel and redirects it onto a lower channel. This unit is

only relevant in smaller scales, where it can not be replaced by multiple inlet tanks. Again, an even distribution of the algal suspension over the complete width of the lower channel is desired. Therefore geometric modifications should be compared, and the installation of baffles to redirect the flow onto the channel should be evaluated.

To perform the simulations, CFD representations of the different geometries were required. Thus, models should be developed, that allowed the robust analyses of different aspects of the corresponding unit. Using a technical scale TLC photobioreactor in the TUM AlgaeTec Center, the results of the simulations should be validated.

The next section will present the results for the development of the CFD models, followed by a summary of the simulation results for the reactor units. For each section, a short outlook will be given that suggests research opportunities in this area.

## 9.1. Setup and Validation of Simulation Models

In order to show the usability of the CFD models generated, the mesh development process included a mesh independence study, in which the mesh geometry was refined until the solution did not depend on the cell size applied. Furthermore a validation of the simulative data with experiments was conducted to prove the applicability of the generated CFD models. To obtain robust meshes for the various use cases, the computational domain was discretised using mostly hexahedral cells (hex-dominant mesh) that are aligned with the flow direction. Since all three reactor units, inlet tank, channel, and flow reversal tank, were treated separately, the above analyses were conducted according to their specific requirements:

In the inlet tank's mesh independence study the water height and velocity on the overflow weir as well as the overall fluid volume were evaluated. The chosen mesh has  $2.01 \cdot 10^6$  cells, with a basic mesh resolution of 5.3 mm, down to approximately 1.3 mm on the weir. It could be shown that the solution varied less than 10 % in height and 3 % in fluid velocity from further refined meshes. The water volume in the tank was consistent over all simulations with an error of 0.5 %. Additionally, tracer simulations were conducted and samples taken at eight different positions inside the tank and on the weir. An agreement of the obtained values in the inner region could be shown, however deviations in the outer, shallow region of the tank were found that did not converge to an independent solution in this study. To decrease the computational effort, the inlet tank was assumed to behave symmetrically along the central axis. In a comparison of a complete tank and one, for which a symmetric Boundary Condition (BC) was applied, no substantial variances could be found, thus approving this simplification: Deviations for the height and the velocity of the fluid on the weir were evaluated to be 2–3 orders of magnitudes lower than the actual values. In the subsequent validation of the CFD model, tracer experiments were conducted. At the unit's inlet a step function in *NaCl* concentration was applied, which was measured over the width of the tank on the weir and inside the tank. For the measurements, newly developed salinity sensors were used, which allowed a temporal resolution of less than 1 ms. Comparing the experimental measurements for two different volume flows ( $\dot{V} = 1.6 \text{ l s}^{-1}$  and  $2.0 \text{ l s}^{-1}$ ) with the simulation results, for both cases a good agreement could be found for most positions. Only for the same position as in

the mesh independence study a deviation could be found. An analysis of the flow field yielded the development of an eddy in this region, which appeared to be highly sensitive to wall effects. However, overall, all tracer positions could be resolved with a delay of less than 1 s.

The channel was discretised by  $1.56 \cdot 10^6$  purely hexahedral cells with a side length of 4.2 mm over the width and 2 mm along the length. In the vertical direction, a refinement of the ground was applied to improve the resolution of the fluid's interface. Here a side length between 0.16 and 2.24 mm was used. Comparing different mesh refinements, for this mesh a maximum error of 5.4 % in water height, 7 % in velocity and 2.1 % in water volume was found, evaluated in the middle of the channel and at the end. To validate the channel, simulation results were compared to measurements as well as theoretical predictions by the empirical Manning equation. The latter correlates the slope of an open channel and the volume flow with its velocity and height (for water) using a coefficient for the roughness of the channel's surface. It thus allows to provide validation in operational states where no measurements were available. For the channel described in this thesis, a Manning coefficient of  $n = 0.010$  was found to match the measurements, which also agreed with the CFD simulation. Comparing the empirical data, over the complete simulated range of volume flows ( $\dot{V} = 1.6\text{--}3.6 \text{ l s}^{-1}$ ) and channel slopes ( $\alpha = 0.5\text{--}3^\circ$ ) a maximum simulation error of 12.7 % for the water height and 12 % for the flow velocity were found.

Finally, the flow reversal tank was only validated qualitatively. In a comparison, the fluids behaviour of both simulation and experimental unit showed an increased water velocity towards the outer wall, resulting in a pile-up of water there, while nearly no water could be observed in the inner region for neither the simulation nor the technical-scale plant. Measurements of the fluid height for this configuration as well as modifications also showed a qualitative agreement of the simulated and measured surface height.

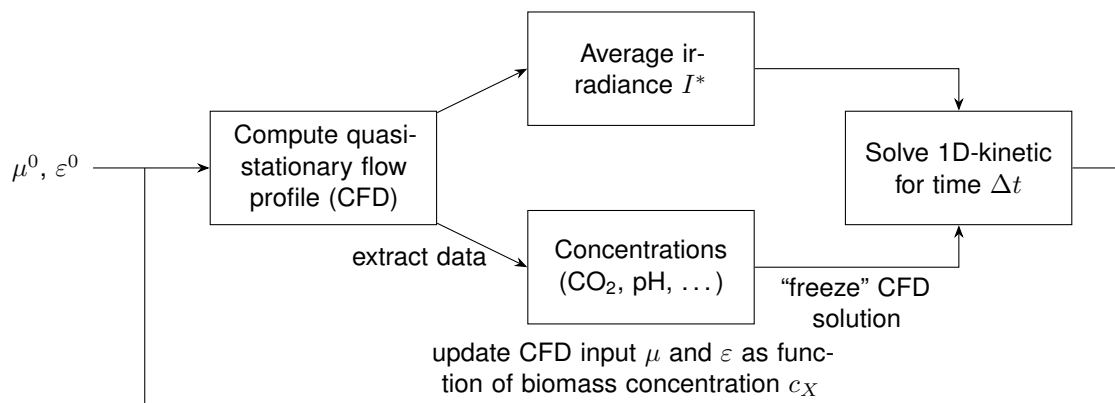
For the simulations, turbulence was considered using Reynolds-Averaged Navier-Stokes (RANS) turbulence models. These describe the turbulence based on averaged quantities for the fluctuation of the velocity. Exploiting the channel geometry, it could be shown that, in order to correctly describe turbulence effects, the  $k - \omega$  shear stress transport (SST) turbulence model had to be applied. In contrast to the most commonly applied  $k - \varepsilon$  model it is of the group of so-called low-Reynolds turbulence models. These models are able to resolve the flow in the vicinity of the wall. Among the different turbulence models and mesh combinations analysed, the  $k - \omega$  SST was the only model that could accurately describe the flow behaviour while providing a resolution of more than 10 cells over the height of the fluid layer.

In further research, the provided models could be extended to give more insights into the effects influencing the cultivation behaviour:

*CO<sub>2</sub> concentration* The implemented passive species transport library already provides the means to simulate liquid-gas interfaces for passive species using Henry's Law. It could thus easily be extended to allow the simulation of dissolved CO<sub>2</sub>. With little effort, it could also be extended to support reactions which describe the equilibrium of the dissolved CO<sub>2</sub> with its acids HCO<sub>3</sub><sup>-</sup> and CO<sub>3</sub><sup>2-</sup>. This would allow to predict the availability of substrate for the microalgae in large-scale reactors. It could further be used to predict positions, where an additional gassing is required.

*Light* Kong and Vigil [2014] already published a simulation for the light distribution inside a photobioreactor using OpenFOAM. Building on their work, it would be advantageous to dynamically simulate the average irradiance available in the suspension, including internal interfaces. This would be of special interest if an uneven underground is considered in the channels and the fluid layer's thickness varies. Here, precise information on the light conditions in the fluid layer could be generated which would allow the estimation of local growth rates.

With these additions, it would be possible to develop an iterative process (Figure 9.1), in which a one-dimensional growth model [Pfaffinger et al. 2016] is solved based on the flow predictions generated by the CFD model. While the CFD simulation would provide average (local) irradiances on the channel and the availability of substrate ( $\text{CO}_2$ ), the growth model would provide input on viscosity and absorption coefficients, which in turn influences the flow behaviour. Due to the significantly slower growth compared to the development of a quasi-stationary fluid flow, only few iterations of the CFD simulations would be needed and could be assumed "frozen" for a certain time during which only the simple growth kinetic is solved.



**Figure 9.1** Iterative process to couple the growth model with predictions by the CFD simulation.

Finally, up to now, the reactor units have been viewed as separate, individual modules. In order to see the complete interaction, a combination of all units might be of interest. This would require to couple the different meshes, which could be achieved using Arbitrarily Coupled Mesh Interface (ACMI) patches, which allow the coupling of arbitrary geometries, independent of their mesh. Yet, for a correct simulation, a lot of effort is required to minimise the interpolation errors between the meshes.

## 9.2. Guidelines for the Design of Thin-Layer Cascade Photobioreactors

For the design of a TLC photobioreactor the following results were obtained for the different units:

*Inlet Tank* The inlet tank greatly influences the distribution of the algal suspension on the following channel. It must thus ensure an even distribution. This was evaluated with respect to volume flow, weir height and width of the inlet. Regarding the volume flow, it was found that for volume flows of less than  $2.8 \text{ l s}^{-1} \text{ m}^{-1}$ , normalised to the width of the following channel, the standard deviation of the fluid's height on the inlet tank's weir is less than 1 mm. For volume flows higher than that a pile-up of water on the outer wall was obtained, resulting in increased disturbances and unequal distribution of the fluid on the following channel. It was assumed that due to the high volume flows the redirection of the flow to the sides increases, leading to the observed behaviour. It was thus concluded that volume flows of  $2.8 \text{ l s}^{-1} \text{ m}^{-1}$  should not be exceeded for the given design to achieve an optimal performance. Contrary to that, the height of the weir itself has negligible influence on the distribution. The deviation of the fluid height over the width is in the range of 0.8–1 mm for weir heights between 16 and 25 cm, with a reference height of 18 cm. It could, however, be calculated, that 1 cm weir height changes the water volume in the tank by approximately 20 %, which is undesirable due to the tanks volume being considered as not taking part in the cultivation process, since here light is hardly available. Thus, the weir should be kept as low as possible. Finally the width of the inlet tank was analysed. A (virtual) scale-up of the tank to a width of 2 m showed again an uneven distribution, similar to the one observed for high volume flows: The water piles up on the sides, while in the central area of the weir no water exits the tank. Although this problem could be mitigated with an additional baffle, it still persisted at a width of 4 m. The simulation of higher width (6 m and above) showed that the operation at these widths is not possible with the currently used construction, and modifications would be necessary.

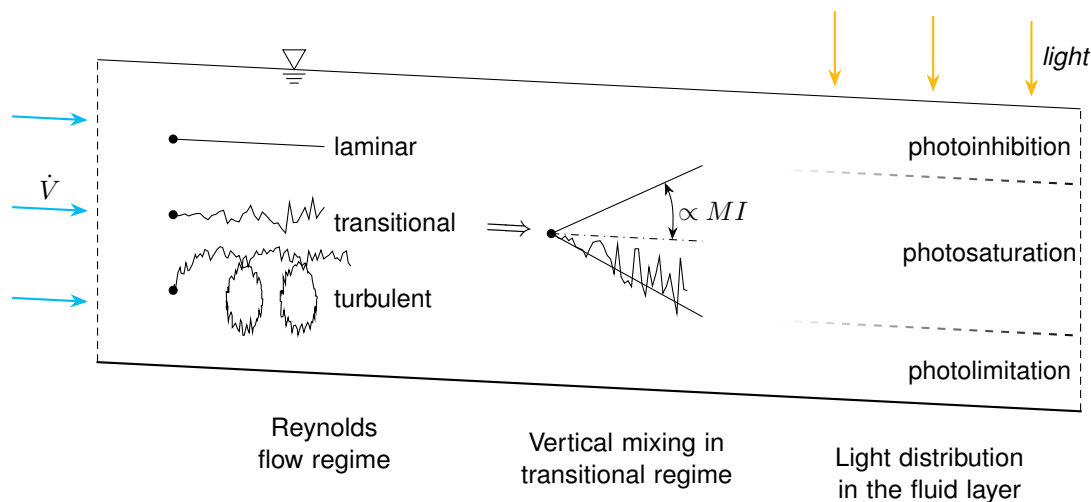
Apart from these influences, the pressure loss for the tank and the hydrostatic pressure were extracted from the CFD simulation. It showed that the pressure loss due to the baffle is relatively small (approximately 50 Pa) compared to the hydrostatic pressure in the tank (1800 Pa), indicating little room for improvement. Based on the pressure loss the system's power input could be estimated. Here, also the effects of the piping (surface and curvature of the hose, valve), the hydrostatic pressure between retention tank and inlet tank, and the pumps efficiency were considered. For a volume flow of  $2.4 \text{ l s}^{-1}$  with a 1 m wide reactor an areal power input of  $32 \text{ W m}^{-2}$  was estimated for the  $8 \text{ m}^2$  technical-scale reactor, which agreed well with measurements of  $33.8 \text{ W m}^{-2}$ . It could, however, be shown that with an improved piping (without the valve) and an industrial sized pump with higher efficiency the power requirements could be reduced by roughly 75 %, depending on the volume flow. Nevertheless, the tanks influence was estimated as 5 % in the current solution and less than 10 % in the redesign. It was thus concluded that an energetic optimisation of the tank is not reasonable.



*Channel* The CFD simulations gave predictions on the height and velocity of the suspension on the channel. Operational conditions with volume flows of  $1.6\text{--}3.6\text{ l s}^{-1}\text{ m}^{-1}$ , normalised to the channel width, for viscosities between 1 and 10 mPa s were evaluated. It could be shown that the layer thickness can vary in the range of 5 to 13 mm for a channel sloped at  $0.5^\circ$ . For channels sloped at  $3^\circ$ , this range was reduced to 2.5 to 7 mm. Since the layer thickness for optimal light availability depends on the microalgae strain used as well as the target biomass concentration, it could be concluded that the reactor already has to be designed to provide the required range in an early stage of the construction.

A further result of the channel simulations was the development of a mixing index *MI*. It allows the quantification and comparison of the vertical mixing in different channels and reactor systems if a turbulent regime cannot be guaranteed. Imagining a virtual particle, the mixing index is proportional to the cone of the possible spreads of this particle (Figure 9.2) and is further scaled to the water height for comparability. In an analysis of the index itself, it could be shown to behave proportionally to the fluid's velocity, however distinctively different than the Reynolds number. With this index, estimates of the mixing for a channel with a volume flow of  $3.6\text{ l s}^{-1}\text{ m}^{-1}$  and a slope of  $3^\circ$  were three times as high as for a channel with a volume flow of  $1.6\text{ l s}^{-1}\text{ m}^{-1}$  and a slope of  $0.5^\circ$ . Matrices of mixing indices were computed for varying volume flows, viscosities, and angles that allow to adjust the operational conditions of a process to maintain mixing if the viscosity increases during a cultivation. These matrices also showed that initial changes of the viscosity in dilute suspensions have a significantly higher influence than further increases: Mixing is reduced twice as much between 1 and 4 mPa s than between 4 and 7 mPa s. Further, the mixing index could be exploited to calculate the channel length it takes a particle to cross the fluid layer from top to bottom (photoinhibited to photolimited zone or vice versa). For the reference system with a volume flow of  $2.4\text{ l s}^{-1}\text{ m}^{-1}$ , sloped at  $1^\circ$ , it was calculated that 68.2% ( $= 1\sigma$ ) of all particles require a length of at least 4.6 m. Since the channel is only 4 m long, it could be discussed, whether this technical-scale system is sufficient to predict the behaviour in an industrial plant. Should, however, in this plant an increased mixing occur, e.g. due to a rougher channel ground, the length of the technical-scale will be sufficient, since it artificially mixes the suspension on each pass through the flow reversal or inlet tank.

*Flow Reversal Tank* In the initial configuration of the flow reversal tank the velocity towards its outer wall dominated. The simulation showed a piling up of water approximately 10 mm above the water level in the centre, while in the inner region barely any fluid could be found. In order to create a completely even surface on the channel following the flow reversal tank, the usage of different baffles in the tank itself and a weir with guide plates was simulated. A parameter study revealed an optimal configuration with 16 guide plates and two baffles inside the tank, which produces a very even surface in the channel. It could further be shown, that a setup consisting of only a weir with eight guide plates is sufficient to reduce the piling up of water to hardly measurable ranges. However, the utilisation of additional elements increases costs and maintenance, thus an alternative, more simple geometry was also desired. It could be found that a shortened outlet of the flow reversal tank already reduces the peaks in water height by approximately 5 mm and visibly more suspension can



**Figure 9.2** Visualisation of vertical mixing in an open channel: In a completely laminar regime, no mixing takes place, while in a highly turbulent regime complete mixing can be assumed. For the transitional regime further measures are required. Here the  $MI$  helps to interpret how thorough a particle is mixed in the fluid layer. This is especially important in conjunction with the exemplary drawn zones of photoinhibition, -saturation and -limitation, where sufficient mixing must be achieved to allow maximum growth of the microalgal cells.

be found in the inner region of the channel. Nevertheless, it was concluded that it is more advantageous to replace the flow reversal tank with a second inlet tank. Increasing the channel length, a similar setup as before can be achieved here. Since the  $\text{CO}_2$  supply is mounted in the inlet tank, this would also allow more process control.

To continue on these results, further options for the reactor units could be compared: With a list of constructive restrictions, the width study of the inlet tank should be repeated. Here it could also be beneficial to redesign the connections between the retention tank and the inlet tank in order to improve the energetic efficiency of the system. With improved piping and upscaled reactor system, the energy evaluation could be repeated. If, due to higher volume flows, the pressure loss in the tank has significant influence, a redesign of the geometry and the baffle could be considered.

Regarding the channel, the vertical velocity of particles should be verified in order to validate whether a channel length of 4 m is sufficient for the scale-up. Furthermore, simulations with increased surface roughness of the channel or added bumps should be performed, to investigate their influence on the vertical mixing. If the evaluation shows improvements, the — for common photobioreactors typical — pool liner as channel ground could help mitigate problems that might occur in a nearly unmixed suspension. Apart from the reactor simulations, a further comparison of the mixing index in different geometries could be of interest, e.g. a comparative study with literature data from Raceway Ponds [Prussi et al. 2014]. This could help to further identify different regimes in which reactors operate and help understand the light availability in such systems.

Although the flow reversal tank, as third investigated unit, should be avoided, it might be necessary in some processes. If it is required, a more thorough parameter study should be performed, which quantifies the influence of gradual changes of angles and lengths.

Having gathered trends of the reactor's performance under different geometric and operational conditions, various measures (e.g. fluid height, mixing) can be evaluated. Using modelling techniques such as High-Dimensional Model Representation (HDMR) [Sobol 1993], a simplified model could be constructed which could then easily be the base for optimisation or design choices in industrial plants. Analysing the results with respect to energy costs (pressure losses) and investment costs (size, complexity of baffles), this model could also provide estimates for economic variables.

## 10. Acknowledgements

There are dozens of people who supported me while I worked on this thesis. The list includes bosses, colleagues, family, friends, and students. And probably also people I just met and don't even know, that just give moral support ("Wow. I don't understand this, but it sounds very interesting. This must be important work."). Normally this would be the part where I'd name them. I certainly could try, but chances are that I forget at least one person. And, quite frankly, those who helped me should already know that I'm grateful for whatever they did. And those who don't might conclude that I'm not, though that's probably not true either.

So the only person left to thank is you. The person who (surprisingly) has, does or will read this document. What may have brought you here, you obviously kept going until the acknowledgements. Therefore I kind of like you. Also, let me tell you that you have too much time at hand.

Summarised, at the very end of a thesis, one realises that a document like this always has many contributors. Half of the contents are actual input by the author. Another half are the people that helped. And the final half are the ones who read this, and used the input. Skip any of those, and a thesis won't work anymore. For those who know math, you might have realised that those were three halves. That is, because a thesis is much more than what's written here. But that is a story for another time. . .

Over and out, Timm

# Appendix

## A. Bibliography

- Ali, Haider, Taqi A. Cheema, Ho-Sung Yoon, Younghae Do and Cheol W. Park (2015). 'Numerical prediction of algae cell mixing feature in raceway ponds using particle tracing methods'. eng. In: *Biotechnology and bioengineering* 112 (2 2015), pp. 297–307. ISSN: 0006-3592. DOI: 10.1002/bit.25443.
- Alkhamis, Yousef and Jian G. Qin (2013). 'Cultivation of *Isochrysis galbana* in phototrophic, heterotrophic, and mixotrophic conditions'. eng. In: *BioMed research international* 2013 (2013), p. 983465. ISSN: 2314-6141. DOI: 10.1155/2013/983465.
- Amaro, Helena M., A. Catarina Guedes and F. Xavier Malcata (2011). 'Advances and perspectives in using microalgae to produce biodiesel'. In: *Applied Energy* 88 (10 2011), pp. 3402–3410. ISSN: 03062619. DOI: 10.1016/j.apenergy.2010.12.014.
- ANSYS Inc. (2009). *ANSYS Fluent 12.0. Users's Guide*. ANSYS Inc., 2009.
- Apel, A. C., C. E. Pfaffinger, N. Basedahl, N. Mittwollen, J. Göbel, J. Sauter, T. Brück and D. Weuster-Botz (2017). 'Open thin-layer cascade reactors for saline microalgae production evaluated in a physically simulated Mediterranean summer climate'. In: *Algal Research* 25 (2017), pp. 381–390. ISSN: 22119264. DOI: 10.1016/j.algal.2017.06.004.
- Apel, Andreas Christoph (2017). *Length and diameter of the hose used in the reactor setup*. oral. 2017.
- Apel, Andreas Christoph and Dirk Weuster-Botz (2015). 'Engineering solutions for open microalgae mass cultivation and realistic indoor simulation of outdoor environments'. eng. In: *Bioprocess and biosystems engineering* 38 (6 2015), pp. 995–1008. ISSN: 1615-7591. DOI: 10.1007/s00449-015-1363-1.
- Avallone, Eugene A., Theodore Baumeister, Ali M. Sadegh and Lionel S. Marks (2007). *Marks' standard handbook for mechanical engineers*. 11th ed. New York and London: McGraw-Hill, 2007. ISBN: 978-0071428675.
- Ayachit, Utkarsh (2015). *The Paraview Guide. A Parallel Visualization Application*. Kitware, 2015. 276 pp. ISBN: 978-1930934306.
- Baerns, Manfred (2006). *Technische Chemie*. Weinheim: Wiley-VCH, 2006. ISBN: 978-3-527-31000-5.
- Bannari, R., A. Bannari, B. Selma and P. Proulx (2011). 'Mass transfer and shear in an airlift bioreactor. Using a mathematical model to improve reactor design and performance'. In: *Chemical Engineering Science* 66 (10 2011), pp. 2057–2067. ISSN: 00092509. DOI: 10.1016/j.ces.2011.01.038.
- Bathe, Klaus-Jürgen (2002). *Finite-Elemente-Methoden*. 2., vollst. neu bearb. und erw. Aufl. Berlin [u.a.]: Springer, 2002. XVI, 1253 S. ISBN: 978-354066806-0.
- Baukal, Charles E., Vladimir Y. Gershtein and Li Xianming (2001). *Computational fluid dynamics in industrial combustion*. Boca Raton (FL) [etc.]: CRC Press, 2001. 630 pp. ISBN: 978-08493-2000-2.
- Becker, E. W. (1994). *Microalgae. Biotechnology and microbiology / E.W. Becker*. Cambridge: Cambridge University Press, 1994. ISBN: 0521350204.

- Benemann, J. R. and W. J. Oswald (1996). *Systems and economic analysis of microalgae ponds for conversion of CO<sub>2</sub> to biomass. Final report*. 1996. DOI: 10.2172/493389.
- Bitog, J. P., I.-B. Lee, C.-G. Lee, K.-S. Kim, H.-S. Hwang, S.-W. Hong, I.-H. Seo, K.-S. Kwon and E. Mostafa (2011). 'Application of computational fluid dynamics for modeling and designing photobioreactors for microalgae production. A review'. In: *Computers and Electronics in Agriculture* 76 (2 2011), pp. 131–147. ISSN: 01681699. DOI: 10.1016/j.compag.2011.01.015.
- Borowitzka, Michael A. (2013). 'Energy from Microalgae. A Short History'. In: *Algae for Biofuels and Energy*. Developments in applied phycology 5. Dordrecht, 2013, pp. 1–15. ISBN: 978-94-007-5479-9. DOI: 10.1007/978-94-007-5479-9\_1.
- Algae for Biofuels and Energy* (2013a). Vol. 5. Developments in applied phycology. Dordrecht, 2013. ISBN: 978-94-007-5479-9.
- Borowitzka, Michael A. and Navid Reza Moheimani (2013b). 'Open Pond Culture Systems'. In: *Algae for Biofuels and Energy*. Developments in applied phycology 5. Dordrecht, 2013, pp. 133–152. ISBN: 978-94-007-5479-9. DOI: 10.1007/978-94-007-5479-9\_8.
- Instrumentation Reference Book* (2003). 3rd Ed. Boston and Oxford, 2003. ISBN: 978-0-7506-7123-1.
- Brennan, Liam and Philip Owende (2010). 'Biofuels from microalgae? A review of technologies for production, processing, and extractions of biofuels and co-products'. In: *Renewable and Sustainable Energy Reviews* 14 (2 2010), pp. 557–577. ISSN: 13640321. DOI: 10.1016/j.rser.2009.10.009.
- Buhr, H. O. and S. B. Miller (1983). 'A dynamic model of the high-rate algal-bacterial wastewater treatment pond'. In: *Water Research* 17 (1 1983), pp. 29–37. ISSN: 00431354. DOI: 10.1016/0043-1354(83)90283-X.
- Burnett, David S. (1987). *Finite element analysis. From concepts to applications / David S. Burnett*. Reading, Mass.: Addison-Wesley Pub. Co, 1987. ISBN: 9780201108064.
- Carvalho, Ana P., Luis A. Meireles and F. Xavier Malcata (2006). 'Microalgal reactors: a review of enclosed system designs and performances'. eng. In: *Biotechnology progress* 22 (6 2006), pp. 1490–1506. ISSN: 1520-6033. DOI: 10.1021/bp060065r.
- Chmiel, Horst (2011). 'Bioreaktoren'. In: *Bioprozesstechnik*. Heidelberg, 2011, pp. 197–236. ISBN: 978-3-8274-2476-1. DOI: 10.1007/978-3-8274-2477-8\_7.
- The application of a multi-physics toolkit to spatial reactor dynamics* (2009). 2009. 3-7.
- Cornet, J. F., C. G. Dussap, P. Cluzel and G. Dubertret (1992). 'A structured model for simulation of cultures of the cyanobacterium *Spirulina platensis* in photobioreactors. II. Identification of kinetic parameters under light and mineral limitations'. eng. In: *Biotechnology and bioengineering* 40 (7 1992), pp. 826–834. ISSN: 0006-3592. DOI: 10.1002/bit.260400710.
- Courant, R., K. Friedrichs and H. Lewy (1928). 'Über die partiellen Differenzgleichungen der mathematischen Physik'. In: *Mathematische Annalen* 100 (1928), pp. 32–74.
- Cummings, W. G., K. Torrance and I. Verhappen (2003). 'Chaper 17: Chemical Analysis: Electrochemical Techniques'. In: *Instrumentation Reference Book*. 3rd Ed. Boston and Oxford, 2003, pp. 343–381. ISBN: 978-0-7506-7123-1. DOI: 10.1016/B978-075067123-1/50021-1.

- Cyprien (2012). *InterFoam: Add an equation to only one phase*. 2012. URL: <https://www.cfd-online.com/Forums/openfoam-programming-development/110512-interfoam-add-equation-only-one-phase.html#post397752> (visited on 10/03/2017).
- Davidson, Lars (2016). *An Introduction to Turbulence Models*. Göteborg, Sweden: Department of Thermo and Fluid Dynamics, Chalmers University of Technology, 2016. URL: [http://www.tfd.chalmers.se/~lada/postscript\\_files/kompendium\\_turb.pdf](http://www.tfd.chalmers.se/~lada/postscript_files/kompendium_turb.pdf) (visited on 12/03/2017).
- Doucha, Jiří and Karel Lívanský (1995). 'Novel outdoor thin-layer high density microalgal culture system: Productivity and operational parameters'. In: *Algolog. Stud., Stuttgart* 76 (1995), pp. 129–147.
- Dresselhaus, M. S. and I. L. Thomas (2001). 'Alternative energy technologies'. eng. In: *Nature* 414 (6861 2001), pp. 332–337. ISSN: 0028-0836. DOI: 10.1038/35104599.
- Ferziger, Joel H. and M. Perić (2002). *Computational methods for fluid dynamics*. 3rd, rev. ed. Berlin and New York: Springer, 2002. xiv, 423. ISBN: 9783540420743.
- Experimental and CFD study of a single phase cone-shaped helical coiled heat exchanger* (2012). *An empirical correlation*. 2012. 375-394.
- Fogler, H. Scott (2016). 'Chapter 13: Distributions of Residence Times for Chemical Reactors'. In: *Elements of Chemical Reaction Engineering*. Fifth edition. Prentice Hall International Series in the physical and chemical engineering sciences. 2016, pp. 867–944. ISBN: 978-0-13-388751-8.
- French, Richard H. (1987). *Open-channel hydraulics*. Internat. ed. New York [u.a.]: McGraw-Hill, 1987. XI, 739 S. ISBN: 0070221340.
- Giannelli, Luca, Hideki Yamaji and Tomohisa Katsuda (2015). 'A Numerical Model for the Quantification of Light/Dark Cycles in Microalgal Cultures. Air-Lift and Bubble-Column Photobioreactor Analysis by Means of Computational Fluid Dynamics'. In: *Journal of Chemical Engineering of Japan* 48 (1 2015), pp. 61–71. ISSN: 0021-9592. DOI: 10.1252/jcej.14we093.
- Gray, James R. (2005). 'Chapter 23: Conductivity Analyzers and Their Application'. In: *Environmental Instrumentation and Analysis Handbook*. New York and Chichester, 2005, pp. 491–510. ISBN: 978-0-471-46354-2. DOI: 10.1002/0471473332.ch23.
- Greenshields, Chris (2017a). *OpenFOAM User Guide. Numerical Schemes*. 2017. URL: <https://cfd.direct/openfoam/user-guide/fvschemes/> (visited on 09/03/2017).
- Greenshields, Chris (2017b). *OpenFOAM User Guide. Breaking of a dam*. CFD Direct Ltd. 2017. URL: <https://cfd.direct/openfoam/user-guide/dambreak/> (visited on 05/04/2017).
- Greenshields, Chris (2017c). *OpenFOAM User Guide. Solution and algorithm control*. CFD Direct Ltd. 2017. URL: <https://cfd.direct/openfoam/user-guide/fvsolution/> (visited on 05/04/2017).
- Grobbelaar, Johan U. (2009a). 'Factors governing algal growth in photobioreactors. The “open” versus “closed” debate'. In: *Journal of Applied Phycology* 21 (5 2009), pp. 489–492. ISSN: 0921-8971. DOI: 10.1007/s10811-008-9365-x.
- Grobbelaar, Johan U. (2009b). 'From laboratory to commercial production. A case study of a Spirulina (Arthrospira) facility in Musina, South Africa'. In: *Journal of Applied Phycology* 21 (5 2009), pp. 523–527. ISSN: 0921-8971. DOI: 10.1007/s10811-008-9378-5.



- Grobbelaar, Johan U. (2012). 'Microalgae mass culture. The constraints of scaling-up'. In: *Journal of Applied Phycology* 24 (3 2012), pp. 315–318. ISSN: 0921-8971. DOI: 10.1007/s10811-011-9728-6.
- Gualtieri, Paolo and L. Barsanti (2006). *Algae. Anatomy, biochemistry, and biotechnology / by Paolo Gualtieri and Laura Barsanti*. Boca Raton, Fla. and London: CRC, 2006. ISBN: 978-0-8493-1467-4.
- Haroun, Y., D. Legendre and L. Raynal (2010). 'Volume of fluid method for interfacial reactive mass transfer. Application to stable liquid film'. In: *Chemical Engineering Science* 65 (10 2010), pp. 2896–2909. ISSN: 00092509. DOI: 10.1016/j.ces.2010.01.012.
- Haynes, W. M. and David R. Lide (2010). 'Electrical Conductivity of Aqueous Solutions'. In: *CRC Handbook of Chemistry and Physics. A ready-reference book of chemical and physical data: 2010-2011*. 91st ed. Vol. 91st Edition. Boca Raton, Fla. and London, 2010, pp. 5–71. ISBN: 978-1-4398-2077-3.
- henry (2010). *Compressible kOmegaSST*. 2010. URL: <http://www.cfd-online.com/Forums/openfoam-bugs/62339-compressible-komegasst-3.html#post265756> (visited on 30/03/2016).
- Heussler, P. (1985). 'Aspects of sloped algae pond engineering'. In: *Production and use of Microalgae. 2nd International conference: Selected papers*. Vol. 20. Advances in Limnology. 1985, pp. 71–83. ISBN: 978-3-510-47018-1.
- Hirt, C.W and B.D Nichols (1981). 'Volume of fluid (VOF) method for the dynamics of free boundaries'. In: *Journal of Computational Physics* 39 (1 1981), pp. 201–225. ISSN: 00219991. DOI: 10.1016/0021-9991(81)90145-5.
- Holzmann, Tobias (2017a). *Mathematics, Numerics, Derivations and OpenFOAM®. The Basics for Numerical Simulations*. 4th Edition. Leoben, 2017. URL: <http://www.holzmann-cfd.de>.
- Holzmann, Tobias (2017b). *Numerical Schemes*. 2017. URL: <http://www.holzmann-cfd.de/index.php/en/numerical-schemes> (visited on 09/03/2017).
- Huang, Jianke, Shaofeng Kang, Minxi Wan, Yuanguang Li, Xiaoxing Qu, Fei Feng, Jun Wang, Weiliang Wang, Guomin Shen and Wei Li (2015). 'Numerical and experimental study on the performance of flat-plate photobioreactors with different inner structures for microalgae cultivation'. In: *Journal of Applied Phycology* 27 (1 2015), pp. 49–58. ISSN: 0921-8971. DOI: 10.1007/s10811-014-0281-y.
- Huang, Qingshan, Tianzhong Liu, Jing Yang, Lishan Yao and Lili Gao (2011). 'Evaluation of radiative transfer using the finite volume method in cylindrical photoreactors'. In: *Chemical Engineering Science* 66 (17 2011), pp. 3930–3940. ISSN: 00092509. DOI: 10.1016/j.ces.2011.05.032.
- Hyldegård, A., D. Mortensen, K. Birkelund, O. Hansen and E. V. Thomsen (2008). 'Autonomous multi-sensor micro-system for measurement of ocean water salinity'. In: *Sensors and Actuators A: Physical* 147 (2 2008), pp. 474–484. ISSN: 09244247. DOI: 10.1016/j.sna.2008.06.004.
- Issa, R.I (1986). 'Solution of the implicitly discretised fluid flow equations by operator-splitting'. In: *Journal of Computational Physics* 62 (1 1986), pp. 40–65. ISSN: 00219991. DOI: 10.1016/0021-9991(86)90099-9.

- James, Scott C. and Varun Boriah (2010). 'Modeling algae growth in an open-channel raceway'. eng. In: *Journal of computational biology : a journal of computational molecular cell biology* 17 (7 2010), pp. 895–906. ISSN: 1066-5277. DOI: 10.1089/cmb.2009.0078.
- Jasak, Hrvoje (1996). 'Error Analysis and Estimation for Finite Volume Method with Applications to Fluid Flow'. Department of Mechanical Engineering, Imperial College of Science, Technology and Medicine. PhD-Thesis. London: University of London, 1996.
- Jasak, Hrvoje (2006). *Numerical Solution Algorithms for Compressible Flows*. Zagreb: University of Zagreb, Croatia, 2006.
- Jimenez, B., A. Noyola and B. Capdeville (1988). 'Selected dyes for residence time distribution evaluation in bioreactors'. In: *Biotechnology Techniques* 2 (2 1988), pp. 77–82. ISSN: 0951-208X. DOI: 10.1007/BF01876154.
- Koller, Anja Pia (2017). *Reaktionskinetik des lichtabhängigen Wachstums von Scenedesmus spec. in Flachplattenphotobioreaktoren*. ger. München: Universitätsbibliothek der TU München, 2017. Online-Ressource.
- Kong, Bo and R. Dennis Vigil (2014). 'Simulation of photosynthetically active radiation distribution in algal photobioreactors using a multidimensional spectral radiation model'. eng. In: *Bioresource technology* 158 (2014), pp. 141–148. ISSN: 0960-8524. DOI: 10.1016/j.biortech.2014.01.052.
- Larock, Bruce E., Roland W. Jeppson and Gary Z. Watters (2000). *Hydraulics of pipeline systems*. Boca Raton and London: CRC, 2000. ISBN: 9780849318061.
- Larrucea, Silvia (2010). 'Computational fluid dynamic model of a thin layer cascading photobioreactor'. Master Thesis. Università di Roma "La Sapienza", 2010. URL: <http://hdl.handle.net/2099.1/14675> (visited on 17/03/2017).
- Lashof, Daniel A. and Dilip R. Ahuja (1990). 'Relative contributions of greenhouse gas emissions to global warming'. In: *Nature* 344 (6266 1990), pp. 529–531. ISSN: 0028-0836. DOI: 10.1038/344529a0.
- Lauder, B. E. and D. B. Spalding (1974). 'The numerical computation of turbulent flows'. In: *Computer Methods in Applied Mechanics and Engineering* 3 (2 1974), pp. 269–289. ISSN: 00457825. DOI: 10.1016/0045-7825(74)90029-2.
- Liffman, Kurt, David A. Paterson, Petar Liovic and Pratih Bandopadhyay (2013). 'Comparing the energy efficiency of different high rate algal raceway pond designs using computational fluid dynamics'. In: *Chemical Engineering Research and Design* 91 (2 2013), pp. 221–226. ISSN: 02638762. DOI: 10.1016/j.cherd.2012.08.007.
- Lundquist, Tryg J., Ian C. Woertz, N. W.T. Quinn and John R. Benemann (2010). 'A realistic technology and engineering assessment of algae biofuel production'. In: *Energy Biosciences Institute* (2010), pp. 1–178.
- Manning, Robert, John Purser Griffith, T. F. Pigot and Leveson Francis Vernon-Harcourt (1890). *On the flow of water in open channels and pipes*. 1890.
- Märkl, H. and M. Mather (1985). 'Mixing and aeration of shallow open ponds'. In: *Production and use of Microalgae. 2nd International conference: Selected papers*. Vol. 20. Advances in Limnology. 1985, pp. 85–93. ISBN: 978-3-510-47018-1.
- Masojidek, Jiri, Jiri Kopecky, Luca Giannelli and Giuseppe Torzillo (2011). 'Productivity correlated to photobiochemical performance of Chlorella mass cultures grown outdoors in

- thin-layer cascades'. eng. In: *Journal of industrial microbiology & biotechnology* 38 (2 2011), pp. 307–317. ISSN: 1367-5435. DOI: 10.1007/s10295-010-0774-x.
- Masojidek, Jiri, Giuseppe Torzillo and Michal Koblížek (2013). 'Photosynthesis in Microalgae'. In: *Handbook of Microalgal Culture. Applied Phycology and Biotechnology*. Second Edition. 2013, pp. 21–36. ISBN: 978-1-118-56718-0. DOI: 10.1002/9781118567166.ch2.
- Menter, F. R., M. Kuntz and R. Langtry (2003). 'Ten years of industrial experience with the SST turbulence model'. In: *Turbulence, heat and mass transfer* 4 (1 2003), pp. 625–632.
- Molina Grima, E., F.G.Acién Fernández, F. García Camacho and Yusuf Chisti (1999). 'Photobioreactors. Light regime, mass transfer, and scaleup'. In: *Journal of Biotechnology* 70 (1-3 1999), pp. 231–247. ISSN: 01681656. DOI: 10.1016/S0168-1656(99)00078-4.
- Molina, E., J. Fernández, F. G. Acién and Y. Chisti (2001). 'Tubular photobioreactor design for algal cultures'. In: *Journal of Biotechnology* 92 (2 2001), pp. 113–131. ISSN: 01681656. DOI: 10.1016/S0168-1656(01)00353-4.
- Nieves-Remacha, María José, Lu Yang and Klavs F. Jensen (2015). 'OpenFOAM Computational Fluid Dynamic Simulations of Two-Phase Flow and Mass Transfer in an Advanced-Flow Reactor'. In: *Industrial & Engineering Chemistry Research* 54 (26 2015), pp. 6649–6659. ISSN: 0888-5885. DOI: 10.1021/acs.iecr.5b00480.
- Noh, W. F. and Paul Woodward (1976). 'SLIC (Simple Line Interface Calculation)'. In: *Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics June 28 – July 2, 1976 Twente University, Enschede*. Vol. 59. Lecture Notes in Physics. Berlin, Heidelberg, 1976, pp. 330–340. ISBN: 978-3-540-08004-6. DOI: 10.1007/3-540-08004-X\_336.
- Oncel, Suphi S. (2013). 'Microalgae for a macroenergy world'. In: *Renewable and Sustainable Energy Reviews* 26 (2013), pp. 241–264. ISSN: 13640321. DOI: 10.1016/j.rser.2013.05.059.
- Oswald, W. J. (1988). 'Large-scale algal culture systems (engineering aspects)'. In: *Micro-algal biotechnology*. Cambridge, 1988, pp. 357–394. ISBN: 9780521323499.
- Owen, Nick A., Oliver R. Inderwildi and David A. King (2010). 'The status of conventional world oil reserves—Hype or cause for concern?' In: *Energy Policy* 38 (8 2010), pp. 4743–4749. ISSN: 03014215. DOI: 10.1016/j.enpol.2010.02.026.
- Patankar, S.V and D.B Spalding (1972). 'A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows'. In: *International Journal of Heat and Mass Transfer* 15 (10 1972), pp. 1787–1806. ISSN: 00179310. DOI: 10.1016/0017-9310(72)90054-3.
- Perner, Iris, Clemens Posten and Jürgen Broneske (2003). 'CFD Optimization of a Plate Photobioreactor Used for Cultivation of Microalgae'. In: *Engineering in Life Sciences* 3 (7 2003), pp. 287–291. ISSN: 16180240. DOI: 10.1002/e1sc.200302964.
- Pfaffinger, Christina Evi (2017a). *Absorption coefficient of N. salina*. 2017.
- Pfaffinger, Christina Evi (2017b). 'Reaktionstechnische Untersuchung zur Lipidherstellung mit *Nannochloropsis* sp. in verschiedenen Photobioreaktoren'. Institute of Biochemical Engineering. PhD Thesis. Technical University of Munich, 2017.
- Pfaffinger, Christina Evi, Dennis Schöne, Sascha Trunz, Hannes Löwe and Dirk Weuster-Botz (2016). 'Model-based optimization of microalgae areal productivity in flat-plate gas-

- lift photobioreactors'. In: *Algal Research* 20 (2016), pp. 153–163. ISSN: 22119264. DOI: 10.1016/j.algal.2016.10.002.
- Posten, Clemens and Christian Walter (2012). *Microalgal biotechnology. Potential and production*. Berlin and Boston: De Gruyter, 2012. xix, 266. ISBN: 9783110225013.
- Powell, C. K., J. B. Chaddock and J. R. Dixon (1965). 'The motion of algae in turbulent flow'. In: *Biotechnology and bioengineering* 7 (2 1965), pp. 295–308. ISSN: 0006-3592. DOI: 10.1002/bit.260070209.
- Preisig, Hans R. and Robert A. Andersen (2005). 'Historical Review of Algal Culturing Techniques'. In: *Algal culturing techniques*. Burlington, MA, 2005, pp. 1–12. ISBN: 978-008045650-8.
- Prussi, Matteo, Marco Buffi, David Casini, David Chiaramonti, Francesco Martelli, Mauro Carnevale, Mario R. Tredici and Liliana Rodolfi (2014). 'Experimental and numerical investigations of mixing in raceway ponds for algae cultivation'. In: *Biomass and Bioenergy* 67 (2014), pp. 390–400. ISSN: 09619534. DOI: 10.1016/j.biombioe.2014.05.024.
- Pruvost, J., J. Legrand, P. Legentilhomme and A. Muller-Feuga (2002). 'Lagrangian trajectory model for turbulent swirling flow in an annular cell. Comparison with residence time distribution measurements'. In: *Chemical Engineering Science* 57 (7 2002), pp. 1205–1215. ISSN: 00092509. DOI: 10.1016/S0009-2509(02)00009-X.
- Pruvost, J., L. Pottier and J. Legrand (2006). 'Numerical investigation of hydrodynamic and mixing conditions in a torus photobioreactor'. In: *Chemical Engineering Science* 61 (14 2006), pp. 4476–4489. ISSN: 00092509. DOI: 10.1016/j.ces.2006.02.027.
- Pulz, O (2001). 'Photobioreactors. Production systems for phototrophic microorganisms'. In: *Applied Microbiology and Biotechnology* 57 (3 2001), pp. 287–293. ISSN: 0175-7598. DOI: 10.1007/s002530100702.
- Ragni, Maria, Ruth L. Airs, Nikos Leonardos and Richard J. Geider (2008). 'Photoinhibition of PSII in *Emiliania huxleyi* (Haptophyta) Under High Light Stress: The Roles of Photoacclimation, Photoprotection, and Photorepair'. In: *Journal of Phycology* 44 (3 2008), pp. 670–683. ISSN: 00223646. DOI: 10.1111/j.1529-8817.2008.00524.x.
- Rodolfi, Liliana, Graziella Chini Zittelli, Niccolò Bassi, Giulia Padovani, Natascia Biondi, Gimena Bonini and Mario R. Tredici (2009). 'Microalgae for oil. Strain selection, induction of lipid synthesis and outdoor mass cultivation in a low-cost photobioreactor'. eng. In: *Biotechnology and bioengineering* 102 (1 2009), pp. 100–112. ISSN: 0006-3592. DOI: 10.1002/bit.22033.
- Rusche, Henrik (2002). 'Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions'. PhD. London: University of London, 2002.
- Sastre, Rosa Rosello and Clemens Posten (2010). 'Die vielfältige Anwendung von Mikroalgen als nachwachsende Rohstoffe'. In: *Chemie Ingenieur Technik* 82 (11 2010), pp. 1925–1939. DOI: 10.1002/cite.201000124.
- Sato, Toru, Daiki Yamada and Shinichiro Hirabayashi (2010). 'Development of virtual photobioreactor for microalgae culture considering turbulent flow and flashing light effect'. In: *Energy Conversion and Management* 51 (6 2010), pp. 1196–1201. ISSN: 01968904. DOI: 10.1016/j.enconman.2009.12.030.

- Savat, Jan (1977). 'The hydraulics of sheet flow on a smooth surface and the effect of simulated rainfall'. In: *Earth Surface Processes 2* (2-3 1977), pp. 125–140. ISSN: 03601269. DOI: 10.1002/esp.3290020205.
- Saynor, Bob, Ausilio Bauen and Matthew Leach (2003). 'The potential for renewable energy sources in aviation'. In: *report commissioned for the DTI, Imperial College Centre for Energy Policy and Technology* (2003).
- Schuster, Arthur (1905). 'Radiation through a foggy atmosphere'. In: *The astrophysical journal* 21 (1905), p. 1.
- Schwarze, Rüdiger (2013). *CFD-Modellierung. Grundlagen und Anwendungen bei Strömungsprozessen*. Berlin: Springer Vieweg, 2013. 199 pp. ISBN: 978-3-642-24377-6.
- Šetlík, Ivan, Vladimír Šust and Ivan Málek (1970). 'Dual purpose open circulation units for large scale culture of algae in temperate zones. I. Basic design considerations and scheme of a pilot plant'. In: *Algological Studies/Archiv für Hydrobiologie, Supplement Volumes* (1970), pp. 111–164.
- Severin, Timm Steffen, Andreas Apel, Thomas Brück and Dirk Weuster-Botz (2018). 'Investigation of Vertical Mixing in Thin-Layer Cascade Reactors using Computational Fluid Dynamics'. In: *Chemical Engineering Research and Design* (2018). ISSN: 02638762. DOI: 10.1016/j.cherd.2018.01.036.
- Severin, Timm Steffen, Sebastian Plamauer, Andreas Christoph Apel, Thomas Brück and Dirk Weuster-Botz (2017). 'Rapid salinity measurements for fluid flow characterisation using minimal invasive sensors'. In: *Chemical Engineering Science* 166 (2017), pp. 161–167. ISSN: 00092509. DOI: 10.1016/j.ces.2017.03.014.
- Sheehan, J., T. Dunahay, J. Benemann and P. Roessler (1998). *Look Back at the U.S. Department of Energy's Aquatic Species Program. Biodiesel from Algae; Close-Out Report*. 1998. DOI: 10.2172/15003040.
- Sobol, Ilya M. (1993). 'Sensitivity estimates for nonlinear mathematical models'. In: *Mathematical Modelling and Computational Experiments* 1.4 (1993), pp. 407–414.
- Spolaore, Pauline, Claire Joannis-Cassan, Elie Duran and Arsene Isambert (2006). 'Commercial applications of microalgae'. eng. In: *Journal of bioscience and bioengineering* 101 (2 2006), pp. 87–96. ISSN: 1347-4421. DOI: 10.1263/jbb.101.87.
- Succi, Sauro (2013). *The lattice Boltzmann equation for fluid dynamics and beyond*. [1st paperback ed.] Oxford Science Publications. Oxford: Oxford University Press, 2013. XVI, 288 s. ISBN: 978-0-19-967924-9.
- Te Chow, Ven (1959). *Open-channel hydraulics*. McGraw-Hill, 1959.
- Terry, K. L. (1986). 'Photosynthesis in modulated light. Quantitative dependence of photosynthetic enhancement on flashing rate'. eng. In: *Biotechnology and bioengineering* 28 (7 1986), pp. 988–995. ISSN: 0006-3592. DOI: 10.1002/bit.260280709.
- Terry, Kenneth L. and Lawrence P. Raymond (1985). 'System design for the autotrophic production of microalgae'. In: *Enzyme and Microbial Technology* 7 (10 1985), pp. 474–487. ISSN: 01410229. DOI: 10.1016/0141-0229(85)90148-6.
- Torzillo, G., L. Giannelli, A. J. Martinez-Roldan, N. Verdone and P. de Filippis (2010). 'Microalgae culturing in thin-layer photobioreactors'. In: *CHEMICAL ENGINEERING* 20 (2010).

- Umweltbundesamt (2017). *Emissionen von Kohlendioxid, Stickstoffoxiden, Schwefeldioxid, Kohlenmonoxid und Gesamtstaub nach Quekkgkategorien. Tabelle 9 der Gesamtausgabe Energiedaten*. Bundesministerium für Wirtschaft und Energie. 2017. (Visited on 16/03/2017).
- VDI-Wärmeatlas (2013). 11. bearb. und erw. Aufl. VDI-Buch. Berlin [u.a.], 2013. 1650 pp. ISBN: 978-3-642-19981-3.
- Versteeg, H. K. and W. Malalasekera (2007). *An introduction to computational fluid dynamics. The finite volume method*. 2nd ed. Harlow, England and New York: Pearson Education Ltd., 2007. xii, 503. ISBN: 9780131274983.
- Viera, Vítor Verdelho (2015). *Microalgae production systems. The cascade raceways*. European Algae Biomass Conference: A4F, 2015.
- Wei, Daniel (2012a). *Contribution a new utility: refine wall layer mesh based on yPlus field*. 2012. URL: <https://www.cfd-online.com/Forums/openfoam-meshing-utilities/109885-contribution-new-utility-refine-wall-layer-mesh-based-yplus-field.html> (visited on 16/04/2017).
- Wei, Daniel (2012b). *y+ and u+ values with low-Re RANS turbulence models: utility + testcase. Proposing a more general and less-confusing yPlus and yStar utilities*. 2012. URL: <https://www.cfd-online.com/Forums/openfoam/86562-y-u-values-low-re-rans-turbulence-models-utility-testcase-3.html#post392408> (visited on 12/03/2017).
- Weller, H. G., G. Tabor, H. Jasak and C. Fureby (1998). 'A tensorial approach to computational continuum mechanics using object-oriented techniques'. In: *Computers in Physics* 12 (6 1998), p. 620. ISSN: 08941866. DOI: 10.1063/1.168744.
- Wheaton, Zachary C. and Gautham Krishnamoorthy (2012). 'Modeling radiative transfer in photobioreactors for algal growth'. In: *Computers and Electronics in Agriculture* 87 (2012), pp. 64–73. ISSN: 01681699. DOI: 10.1016/j.compag.2012.05.002.
- Wigmosta, Mark S., André M. Coleman, Richard J. Skaggs, Michael H. Huesemann and Leonard J. Lane (2011). 'National microalgae biofuel production potential and resource demand'. In: *Water Resources Research* 47 (3 2011), n/a–n/a. ISSN: 00431397. DOI: 10.1029/2010WR009966.
- Wilcox, D. C. (1988). 'Reassessment of the scale-determining equation for advanced turbulence models'. In: *AIAA Journal* 26 (11 1988), pp. 1299–1310. ISSN: 0001-1452. DOI: 10.2514/3.10041.
- Wilcox, D. C. (2008). 'Formulation of the k- $\omega$  Turbulence Model Revisited'. In: *AIAA Journal* 46 (11 2008), pp. 2823–2838. ISSN: 0001-1452. DOI: 10.2514/1.36541.
- Wolf-Gladrow, Dieter A. (2004). *Lattice-gas cellular automata and lattice Boltzmann models. An introduction*. Berlin [etc.]: Springer, 2004. IX, 308. ISBN: 978-3-540-46586-7.
- Zittelli, Graziella Chini, Natascia Biondi, Liliana Rodolfi and Mario R. Tredici (2013). 'Photobioreactors for Mass Production of Microalgae'. In: *Handbook of Microalgal Culture. Applied Phycology and Biotechnology*. Second Edition. 2013, pp. 225–266. ISBN: 978-1-118-56718-0. DOI: 10.1002/9781118567166.ch13.
- Zittelli, Graziella Chini, Liliana Rodolfi, Niccoló Bassi, Natascia Biondi and Mario R. Tredici (2013). 'Photobioreactors for Microalgal Biofuel Production'. In: *Algae for Biofuels and Energy*. Developments in applied phycology 5. Dordrecht, 2013, pp. 115–131. ISBN: 978-94-007-5479-9. DOI: 10.1007/978-94-007-5479-9\_7.

## B. Symbols

Symbol	Description	Unit
<b>Roman Symbols</b>		
$A$	Surface area	$\text{m}^2$
$b$	Width of a reactor unit	$\text{m}$
$c$	Concentration of a substance	$\text{g m}^{-3}$
$D$	Diffusion coefficient	$\text{m}^2 \text{s}^{-1}$
$d$	Depth of a reactor unit	$\text{m}$
$g$	Gravity, usually $g = (0, 0, -9.81)$	$\text{m s}^{-2}$
$h$	Specific enthalpy ( <i>in thermodynamic context</i> )	$\text{J kg}^{-1}$
$h$	Height ( <i>in geometric context</i> )	$\text{m}$
$I^*$	Mean integral photon flux density	$\mu\text{mol m}^{-2} \text{s}^{-1}$
$I$	Irradiance ( <i>for light attenuation</i> )	$\mu\text{mol m}^{-2} \text{s}^{-1}$
$I_0$	Incident irradiance	$\mu\text{mol m}^{-2} \text{s}^{-1}$
$I$	Turbulent intensity ( <i>for turbulence estimations</i> )	%
$k$	Turbulent kinetic energy in $k - \varepsilon$ and $k - \omega$ turbulence models	$\text{J kg}^{-1}$
$L$	Characteristic length, used e.g. for $Re$ or the turbulent length scale	$\text{m}$
$L$	Layer thickness of the algal suspension in a reactor unit ( <i>for water levels</i> )	$\text{cm}$
$l$	Turbulent length scale ( <i>for turbulence properties</i> )	$\text{m}$
$l$	Depth/length of a reactor unit ( <i>in geometric context</i> )	$\text{m}$
$n$	Manning coefficient	$\text{m}^{-1/3} \text{s}$
$p$	Pressure	$\text{Pa}$
$P$	Wetted perimeter ( <i>for flow properties</i> )	$\text{m}$
$P$	Power ( <i>for power calculation</i> )	$\text{W}$
$p_{\rho gh}$	Pressure field without the hydrostatic pressure	$\text{Pa}$
$\dot{q}''$	Areal based heat transfer (conduction, radiation, ...)	$\text{W m}^{-2}$
$R_h$	Hydraulic radius	$\text{m}$
$S$	Slope of a unit, usually the channel	%
$S$	Source or sink term (in general, $S$ , or for a defined property $\phi$ as $S_\phi$ )	a.u.
$S/V$	Surface-to-Volume ratio of an algal bioreactor	$\text{m}^2 \text{m}^{-3}$
$t$	Time	$\text{s}$
$\Delta t$	Time step	$\text{s}$
$u$	Velocity	$\text{m s}^{-1}$
$u^+$	Dimensionless velocity	-
$u_\tau$	Friction velocity of a fluid close to the wall	$\text{m s}^{-1}$

Symbol	Description	Unit
$v$	Fluid velocity	$\text{m s}^{-1}$
$V$	Volume	$\text{m}^3$
$\dot{V}$	Volume flow	$\text{l s}^{-1}$
$\dot{V}_b$	Volume flow normalised to a specific width $b$ of a reactor unit, usually the channel, to account for different geometries	$\text{l s}^{-1} \text{m}^{-1}$
$y$	Relative fraction of a substance normalised to an arbitrary maximum	$\text{g g}^{-1}$
$y^+$	Dimensionless wall distance	-
<b>Greek Symbols</b>		
$\alpha$	Angle of a reactor part ( <i>in geometric context</i> )	$^\circ$
$\alpha$	Under-relaxation factors ( <i>in iterative algorithms</i> )	-
$\alpha$	Phase fraction of one phase in a two-phase system ( <i>in two-phase system</i> )	% (v/v)
$\varepsilon$	Absorption coefficient ( <i>in Lambert-Beer law</i> )	$\text{m}^{-1}$
$\varepsilon$	Viscous dissipation rate used in $k - \varepsilon$ turbulence models ( <i>in turbulence models</i> )	$\text{J kg}^{-1} \text{s}^{-1}$
$\varepsilon$	Error ( <i>for deviations</i> )	a.u.
$\varepsilon$	Absolute roughness of a surface ( <i>for friction calculation</i> )	mm
$\Gamma$	Mass diffusion coefficient, $\Gamma = D \cdot \rho$	$\text{kg s}^{-1} \text{m}^{-1}$
$\lambda$	Darcy friction factor for pipe surfaces	-
$\mu$	Dynamic viscosity	$\text{kg m}^{-1} \text{s}^{-1}$
$\mu_{\text{eff}}$	Effective viscosity, $\mu_{\text{eff}} = \mu + \mu_t$	$\text{kg s}^{-1} \text{m}^{-1}$
$\mu_t$	Eddy viscosity used in turbulence models	$\text{kg m}^{-1} \text{s}^{-1}$
$\nu$	Kinematic viscosity	$\text{m}^2 \text{s}^{-1}$
$\nu_t$	(Kinematic) Eddy viscosity used in turbulence models	$\text{m}^2 \text{s}^{-1}$
$\omega$	Specific turbulent dissipation rate used in $k - \omega$ turbulence models	$\text{s}^{-1}$
$\phi$	General variable for fluid properties in a CFD simulation, can e.g. be temperature, concentration or velocity.	a.u.
$\rho$	Density	$\text{kg m}^{-3}$
$\sigma$	Surface tension	$\text{kg s}^{-2}$
$\underline{\tau}$	Stress Tensor	Pa
$\underline{\tau}'$	Reynolds Stress Tensor	Pa
$\tau_w$	Wall shear stress	$\text{kg s}^{-2} \text{m}^{-1}$
<b>Dimensionless numbers</b>		
$Co$	Courant number, $Co = \frac{v\Delta t}{\Delta x}$	-
$He$	Henry coefficient for fluid/gas equilibrium	-
$MI$	Mixing index for vertical mixing in channels, compare Section 7.5.2	-



Symbol	Description	Unit
$Re$	Reynolds number, $Re = \frac{\rho v L}{\mu}$ for a characteristic length $L$	-
<b>Subscripts</b>		
0	Value at $t = 0$	
$eq$	Given property is in equilibrium	
$f$	Face of a cell	
$i, j, k$	(node) indices correlating to the x, y, z coordinate system	
$n$	Time step $n$ for a transient solver	
$n, e, s, w$	north, east, south, and west face of the central node P in a Cartesian coordinate system	
$nw, sw, ne, se$	north west, south west, north east, south east corner of the central node P in a Cartesian coordinate system	
$P, N, E, S, W$	Node names in a Cartesian coordinate system, Central node P, surrounded by North, East, South, and West node	
$sp$	specific	
$X$	Property is biomass related (e.g. $c_X$ )	
<b>Superscripts</b>		
*	Guessed value of a field	
0	Initial value	
$n$	Iteration $n$ of an iterative algorithm	
'	Correction term for a field	
<b>Math Operators and Definitions</b>		
$\bar{\phi}$	Averaged property $\phi$ (only used for RANS turbulence models)	
$\nabla \cdot \phi$	Divergence of $\phi$ , e.g. for a vector $\phi$ it can be computed as $\nabla \cdot (\phi) = \frac{\partial \phi_x}{\partial x} + \frac{\partial \phi_y}{\partial y} + \frac{\partial \phi_z}{\partial z}$	
$\nabla \phi$	Gradient $\phi$ , $\nabla \phi = \frac{\partial \phi}{\partial x} + \frac{\partial \phi}{\partial y} + \frac{\partial \phi}{\partial z}$	
$\mathcal{O}$	Landau symbol, usually order of the error	
$\mathbf{n}$	Normal vector for a surface of a cell	
$\frac{d\phi}{dt}, \frac{\partial \phi}{\partial t}$	(Partial) derivative of $\phi$ with respect to $t$	
$\underline{\phi}$	$\phi$ is a tensor variable	
$\phi^T$	Transposed version of the matrix $\phi$	
$\phi$	$\phi$ is a vector variable	
$x, y, z$	Coordinate system directions of the Cartesian system	

## C. Abbreviations

### A

AC - Alternating Current

ACMI - Arbitrarily Coupled Mesh Interface

ADC - Analogue to Digital Converter

### B

BC - Boundary Condition

BDS - Backward Differencing Scheme

### C

CAD - Computer Aided Design

CDS - Central Differencing Scheme

CE-QUAL - U.S. Army Corp of Engineers' water-quality code

CFD - Computational Fluid Dynamics

CV - control volume

### D

DC - Direct Current

DNS - Direct Numerical Simulation

### E

EFDC - Environmental Fluid Dynamics Code

ENIG - Electroless Nickel Immersion Gold

### F

FD - Finite Differences

FDS - Forward Differencing Scheme

FE - Finite Elements

FV - Finite Volume

### G

GAMG - Geometric-Algebraic Multi-Grid, (*Glossary*)

### H

HDRM - High-Dimensional Model Representation

### L

LBM - Lattice Boltzmann Method, (*Glossary*)

LDA - Laser Doppler Anemometry

LES - Large Eddy Simulation

## **M**

MULES - Multidimensional Universal Limiter for Explicit Solution

## **P**

PCB - Printed Circuit Board

PCG - Preconditioned Conjugate Gradient

PDE - Partial Differential Equation

PISO - Pressure Implicit with Splitting Operator

PIV - Particle Image Velocimetry

PVC - Polyvinyl chloride

## **R**

RANS - Reynolds-Averaged Navier-Stokes

RTE - Radiative Transfer Equation, (*Glossary*)

RWP - Raceway Pond

## **S**

SIMPLE - Semi-Implicit Method for Pressure Linked Equations

SST - shear stress transport

STL - Stereolithography

## **T**

TLC - Thin-Layer Cascade, (*Glossary*)

## **U**

UDS - Upwind Differencing Scheme

## **V**

VoF - Volume of Fluid

## D. Glossary

*Airlift Principle* An airlift pump is often used in aquacultures. It works on the principle of submersed bubbles decreasing the overall density and thus inducing an upward flow of the bubbles and the surrounding liquid. They have the advantage that, parallel to the pumping, they allow the aeration of the suspension with O<sub>2</sub> or — in case of microalgae — CO<sub>2</sub>.

*Areal Productivity* Usually the productivity of a biotechnological process is defined as the amount of product produced per volume of the broth used in a given time. However, since for microalgae the surface area exposed to the light source is a relevant measure, here the areal productivity is used instead. It is defined as the amount of product produced in a given time per area required by the reactor setup.

*Autotrophic Growth* (also: *phototrophic*, *photoautotrophic*) The growth of organisms — mainly plants and algae — based on photosynthesis using CO<sub>2</sub> as carbon source and light as energy source.

See also: Heterotrophic growth.

*Boundedness* Boundedness describes the ability of an interpolation scheme to restrict the solution inside given physical boundaries.

An example is the simulation of a temperature gradient between two plates: If one plate has a temperature of 200 K, and the other of 100 K, all computed temperatures must be in the range of 100–200 K. This is guaranteed only with bounded schemes.

See also: Versteeg and Malalasekera [2007].

*Cartesian Grid* A Cartesian grid is a special case of a structured grid, in which the distance between the elements are equal over all dimensions.

*Continuity Equation* Equation describing the conservation of mass, i.e. in the most general form  $\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0$ .

*Courant Number* (also: *Courant–Friedrichs–Lewy condition*) The Courant number is a stability criterion developed from the explicit Euler scheme. It is defined as

$$Co = \frac{\mathbf{u} \Delta t}{\Delta x}$$

For the explicit Euler scheme the condition  $Co \leq 1$  has to be satisfied to allow a stable numerical solution. For implicit schemes this requirement is not as strict, however, limiting the Courant number is generally beneficial for the stability and accuracy of the simulation [Jasak 2006].

*Crank-Nicolson Scheme* The Crank-Nicolson scheme is a blended version of the explicit and implicit Euler scheme. It calculates the property value at the new time step as a 50/50 % mixture of both schemes, and is thus also an explicit scheme.

See also: Versteeg and Malalasekera [2007].

*Euler Scheme(s)* The Euler schemes are the simplest time interpolation schemes. They can be divided in the explicit, forward method and the implicit, backward Euler. While the former is easier to use and can compute the new time step faster, it can behave numerically unstable for large time steps. The latter method is unconditionally stable for the time steps, but requires an iterative, computational more expansive solution procedure.

See also: Versteeg and Malalasekera [2007].

*Flashing Light Effect* The “flashing light effect” is a technique used to enhance the growth of microalgae at oversaturated irradiance conditions by creating a light-dark cycle for the microalgal cells of 1 Hz or more [Terry 1986].

*Geometric-Algebraic Multi-Grid (Abbr. GAMG)* “The generalised method of Geometric-Algebraic Multi-Grid (GAMG) uses the principle of: generating a quick solution on a mesh with a small number of cells; mapping this solution onto a finer mesh; using it as an initial guess to obtain an accurate solution on the fine mesh.” [Greenshields 2017c].

*Heterotrophic Growth* The growth of organisms by usage of organic source, in biotechnology dissolved carbon source, e.g. glucose or glycerine.

See also: Autotrophic growth.

*Kronecker delta* The Kronecker delta  $\delta_{ij}$  is a mathematical function of two indices  $i$  and  $j$ , which is 1 when  $i = j$  and 0 otherwise:

$$\delta_{ij} = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases}$$

*Lagrangian particle tracking* Lagrangian particle tracking is a method in which a particle is tracked through a continuous (fluid) field by balancing the forces that affect it. A particles velocity  $\mathbf{u}_P$  and position  $\mathbf{x}_P$  can thereby be described by the following two differential equations:

$$\begin{aligned} \frac{d\mathbf{u}_P}{dt} &= \mathbf{F}_P(\mathbf{u} - \mathbf{u}_P) \\ \frac{d\mathbf{x}_P}{dt} &= \mathbf{u}_P \end{aligned}$$

Therein  $\mathbf{u}$  is the velocity of the surrounding fluid and  $\mathbf{F}_P$  represents the drag forces on the particle.

*Lambert-Beer Law* The Lambert-Beer law describes the light attenuation of an initial irradiance  $I_0$  in a material based on the length of the light path  $s$  and a given absorption coefficient  $\varepsilon$ . In applications with suspended particles the absorption coefficient can further be resolved to a specific absorption  $\varepsilon_{sp}$  and the particle (biomass) concentration  $c$ .

$$I(s) = I_0 e^{-s\varepsilon} = I_0 e^{-s\varepsilon_{sp}c}$$

Contrary to the Two-Flux approximation this equation does not consider scattering by particles.

*Lattice Boltzmann Method (Abbr. LBM)* The Lattice Boltzmann Method solves the Boltzmann Equation and thus models the fluid as virtual, interacting particles instead of solving for the macroscopic properties mass, momentum and energy.

For details see Succi [2013].

*Manning equation* The Manning equation gives an empirical correlation for the average flow velocity of water in an open channel with a given roughness:

$$v = \frac{1}{n} \cdot R_h^{\frac{2}{3}} S^{\frac{1}{2}}$$

Therein  $n$  is a coefficient depending on the surface roughness,  $R_h$  is the hydraulic radius and  $S$  the slope of the channel.

*Numerical Diffusion (also: false diffusion)* Numerical diffusion is the occurrence of a “smearing” effect in the solution of transported property due to the interpolation scheme. The error in the interpolation can be rewritten in the form of a diffusion term, thus the name.

See also: Ferziger and Perić [2002] and Versteeg and Malalasekera [2007].

*PIMPLE* The PIMPLE algorithm is a combination of the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) and Pressure Implicit with Splitting Operator (PISO) algorithm.

*Radiative Transfer Equation (Abbr. RTE)* The Radiative Transfer Equation describes the propagation of radiation  $I$  through a domain by the following equation:

$$\underbrace{\frac{1}{c} \frac{\partial I}{\partial t}}_{\text{temporal}} + \underbrace{\frac{\partial I}{\partial \mathbf{s}}}_{\text{spacial}} = - \underbrace{(\alpha + \sigma_s) I}_{\text{interaction sink}} + \underbrace{\frac{\sigma_s}{4\pi} \cdot \int_0^{4\pi} I \cdot \Phi \, d\Omega}_{\text{interaction source}} + \underbrace{\frac{\alpha n^2 \sigma T^4}{\pi}}_{\text{emission}}$$

The equation consists of several parts: The first, temporal term on the left describes the time dependent behaviour. For most applications it can be neglected due to nearly instantaneous propagation of light with the speed of light  $c$ . The spacial discretisation gives the irradiance along the vector  $\mathbf{s}$ . The interaction terms respect the loss of radiation energy (sink) due to absorption  $\alpha$  and out-scattering  $\sigma_s$  as well as the radiation “gain” due to in-scattering from the surrounding regions (source). The emission term allows to add a radiation source by black body radiation for the fluid with temperature  $T$ , refractive index  $n$  and the Stefan-Boltzmann constant  $\sigma$ .

*Structured Grid* A structured grid is a grid in which the nodes in each dimension can be indexed consecutively in a way that the neighbours of a node can easily be found by decreasing/increasing the nodes index.

*Thin-Layer Cascade (Abbr. TLC)* The Thin-Layer Cascade reactor is an open photobioreactor originally developed by Šetlík et al. [1970]. It consists of several sequential units: An inlet tank, the upper channel, a flow reversal tank, the lower channel and a retention tank.

The inlet tank distributes the algal suspension over the upper channel, along which the fluid flows gravity driven. At the end of the channel it is collected by the flow reversal tank which redirects it onto the lower channel. After this unit the suspension flows into the retention tank from where it is circulated back into the inlet tank by a pump.

More details can be found in Chapter 5.1.

*Truncation Error* The error that is introduced into an approximation by a truncation of a series expansion, usually the Taylor Series expansion. It is assumed to be of the order of  $\Delta x$  to the power of the first truncated term of the series, where  $\Delta x$  is the grid spacing.

*Two-Flux Approximation* The Two-Flux approximation, originally developed by Schuster [1905], describes the light attenuation of an initial irradiance  $I_0$  inside a suspension considering absorption and scattering at particles. Solved for a flat-plate geometry, the following equation can be obtained [Cornet et al. 1992]:

$$I(s) = I_0 \cdot \frac{4\alpha}{(1 + \alpha)^2 \cdot e^{\delta c_X s} - (1 - \alpha)^2 \cdot e^{-\delta c_X s}}$$

where

$$\alpha = \sqrt{\frac{E_a}{E_a + E_s}} \quad \text{and} \quad \delta = \sqrt{E_a \cdot (E_a + E_s)}$$

Therein  $I_0$  is the incident radiation,  $s$  the light path,  $c$  the particle (biomass) concentration. Absorption and scattering are included by the mass specific absorption and scattering coefficients  $E_a$  and  $E_s$ .

*Under-relaxation* Under-relaxation is a technique used in iterative solvers. It limits the change of the iteratively solved variable  $\phi$  from one iteration  $n$  to the next  $n + 1$  (with under-relaxation  $\alpha_{relax}$ ):

$$\phi_{n+1} = \phi_n + \alpha_{relax} \cdot \phi_{correction}$$

Although this slows convergence down, it usually is necessary to prevent the result from diverging (e.g. in the SIMPLE algorithm).

## E. OpenFOAM Libraries

This chapter gives a detailed description of the custom made OpenFOAM libraries that have been used in this thesis. The source code for them can be found in the attached data.

### E.1. patchPartToFace Source for the topoSet Utility

This library extends the capabilities of the topoSet tool, by adding a new command to select faces on a patch. As input it expects the name of a patch, a radius and an direction. Optional keywords are `inside` and `mode`. A corresponding dictionary entry can be seen in Listing E.1.

The method by which this source selects the cell faces of the given patch is the following:

1. Find the patch with the given `name` and retrieve all its faces.
2. Compute the (signed) distance  $l$  of all the patch's face centres  $\mathbf{f}_c$  to the input vector `direction`  $\mathbf{d}$  based on the face normal  $\mathbf{f}_n$  according to the following equation:

$$\begin{aligned}t &= \mathbf{f}_c \cdot \frac{\mathbf{d}}{\text{mag}(\mathbf{d})} \\s &= (\mathbf{f}_c \times \mathbf{f}_n) \cdot \mathbf{d} \\l &= \text{sign}(s) \cdot \text{mag}(\mathbf{f}_c - \mathbf{d} \cdot t)\end{aligned}$$

3. Reduce the either the maximum or minimum distance value found to zero, depending on which `mode` was selected. This effectively defines through which point of the the direction vector passes through.

$$l = \begin{cases} l - \min(l) & \text{if mode min;} \\ \max l - l & \text{if mode max;} \end{cases}$$

4. Depending on the value of `inside`, select all faces whose distance is now less or equal to the given radius (`inside true`);, or larger than the given radius (`inside false`);).

The source code to this library can be found in the Attachment to this thesis under `OpenFOAM/libraries/topoSet`.

### E.2. cloudSphere Source for the sample Utility

This library allows the sampling of data around a given set of coordinate. It basically extends the included sample source `cloud`, which accepts a number of coordinates and writes the cell



**Listing E.1** An exemplary dictionary entry for the `patchPartToFace` Source of the extended `topoSet` library.

```
1 {  
2     name      inletWallSet;  
3     type      faceSet;  
4     action    delete;  
5     source    patchPartToFace;  
6     sourceInfo  
7     {  
8         name      inlet;  
9         radius    0.01;  
10        direction (1 0 0);  
11  
12        inside    true;          // "true", "false"  
13        mode      min;          // "min" or "max"  
14    }  
15 }
```

value at that point. Thus, it accepts the same input for the sampled `points` plus an additional `radius`, which defines the range in which the data is sampled. For a sample dictionary entry see Listing E.2.

The implemented algorithm works recursively and performs the following steps:

1. For each coordinate that is to be sampled find the corresponding primary cell. If it is found, start the recursive method and pass it the index of the primary cell as base cell and as current cell.
  - a. Check if the current cell exists in the list of stored cells. If it is found, return without changes, otherwise continue.
  - b. Calculate the distance of the current cell to the base cell. If the distance is smaller than or equal to the radius add it to the list of sampled cells, otherwise return without changes.
  - c. Find all neighbouring cells to the current one. For each neighbouring cell, restart at a with the neighbour cell as current cell, while passing on the base cells index as is.
2. If requested (`debug` flag), write the found set of cells to the `constant` directory of the case to make it accessible externally.

The above recursion will end once no new cells are found that are inside the radius. However, for very fine meshes or large radii, the recursion might go through several levels. This can make the process very slow, especially since every new found cell has to be compared to the previously stored list, and cells that are outside the radius might be checked multiple times. However, for the cases at hand, no significant performance problems could be found.

A very useful application of the sample set is to not only extract a value, but also generate a standard deviation, e.g. if it is to be compared to measurements whose position might not be known exactly or where the measurement area is larger than a single CFD cell.

**Listing E.2** Exemplary dictionary entry for the `cloudSphere` sample source, here taken from the salinity measurements on the weir, where a sensor spot has an area of 1 cm<sup>2</sup>, and thus multiple cells have to be considered during the sampling of the data.

---

```
1 measuredPointsStd
2 {
3     type          cloudSphere;
4     axis          xyz;
5     radius        0.005;
6
7     points
8     (
9         // inner points, 1-4
10        (-0.236 0.04 0.103)
11        (-0.1907 0.04 0.082)
12        (-0.1453 0.04 0.061)
13        (-0.1 0.04 0.04)
14        // weir points, 5-8
15        (-0.25 0.122 0.14)
16        (-0.2 0.122 0.14)
17        (-0.15 0.122 0.14)
18        (-0.1 0.122 0.14)
19    );
20 }
```

---

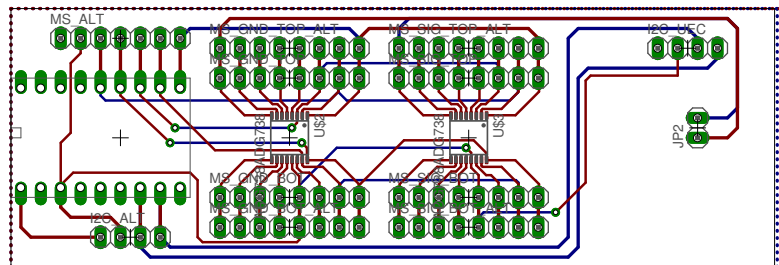
The source code to this library can be found in the Attachment to this thesis under `OpenFOAM/libraries/sampling`.

## F. Salinity Measurements

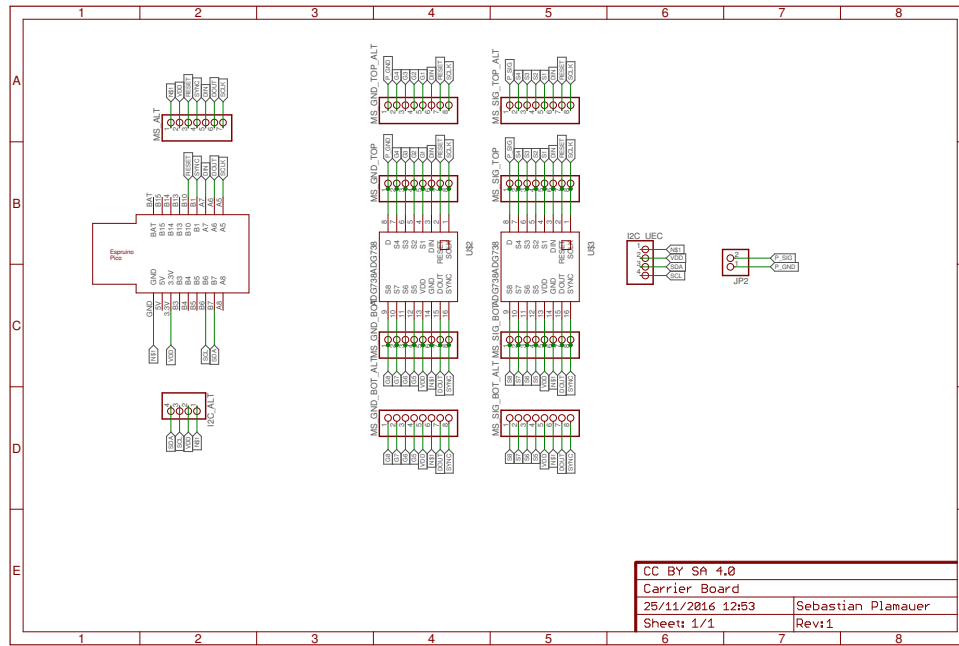
This chapter shows the detailed design of the board used for the salinity sensor. Afterwards the algorithm to smooth the data will be displayed.

### F.1. Carrier Board

Images F.1 and F.2 show the design and schematic of the carrier board. The board itself was manufactured by Where Labs, LCC (<http://dirtypcbs.com>). The layouts have been drawn using the software EAGLE 7.7.0 by Cadsoft. The corresponding .brd and .sch files can be found in the supplemental.



**Figure F.1** Drawing of the carrier board, exported from EAGLE 7.7.0.



**Figure F.2** Schematic of the carrier board, exported from EAGLE 7.7.0.

## F.2. Sliding Maximum Algorithm

The sliding maximum algorithm used in this thesis is shown in Listing F.1. For the evaluation in this theses, the maximum function was used to obtain the hull curve (`max()`). The `windowSize` parameter was varied based on the experiment.

**Listing F.1** The MATLAB `slideFun` function used to generate the hull curves over the raw data using the sliding maximum.

```

1 function result = slideFun( functionHandle, xData, yData, windowSize,
    varargin )
2 %SLIDEFUN applies a user defined function to a sliding window approach
    with
3 %given x and y data
4 %
5 %   SLIDEFUN(FUNCTIONHANDLE, XDATA, YDATA, WINDOWSIZE) applies the
    function
6 %   given as FUNCTIONHANDLE on the YDATA, with a sliding window of
7 %   WINDOWSIZE length selected based on the XDATA.
8 %
9 %   SLIDEFUN(..., 'WindowMode', <value>) allows to select between central
10 %   (default), backward and forward window mode.
11 %
12 %   Details:
13 %       The function will return a vector of the same length as YDATA or
14 %       XDATA, in which each entry contains a value obtained by
15 %       FUNTIONHANDLE for the given window.
16 %

```

```

17 %         The FUNCTIONHANDLE given must be a function, which accepts one,
and
18 %         only one input. If another function is to be used, an anonymous
19 %         function should be used as a wrapper. The result of the function
is
20 %         stored at the evaluated place in the result vector.
21 %
22 %         The window used is calculated both on XDATA and on the WINDOWSIZE,
23 %         where WINDOWSIZE is used to select all data, which is in the range
24 %         of the current value, e.g. for a central window
25 %         currentX - windowSize / 2 < XDATA < currentX + windowSize / 2
26
27 %% Read the input
28 input = inputParser;
29 input.addRequired('functionHandle', @(x)(isa(x,'function_handle')));
30 input.addRequired('xData', @(x)( ...
31     (isnumeric(x) && isvector(x)) ...      xData given as
        numerical vector
32     || (isempty(x)) ...                    empty, so just 1:
        length(yData) implied
33     ));
34 input.addRequired('yData', @(x)( ...
35     isnumeric(x) && isvector(x) ...      must be a vector of
        numerical data
36     ));
37 input.addRequired('windowSize', @(x)( ...
38     isnumeric(x) && isscalar(x) ...      just a single value
39     && (x > 0) ...                        which should be
        positive
40     ));
41 input.addParameter('WindowMode', 'central', @(x)( ...
42     ismember(x, {'central', 'backward', 'forward'}) ...
43     ));
44
45 input.parse(functionHandle, xData, yData, windowSize, varargin{:});
46
47 %% Evaluate inputs given
48 % store the yData
49 yData = input.Results.yData(:);
50 % figure out wheter we got x inputs or not
51 if isempty(input.Results.xData)
52     % default, handle all points equally
53     xData = 1:length(yData);
54 else
55     % check validity
56     % length
57     xData = input.Results.xData(:);
58     if (length(xData) ~= length(yData))
59         error('The vectors for x and y-data must have the same length'
60             );
61     end
62     % sorted in ascending order

```

```

62     if ~all(diff(xData) > 0)
63         error('The x vector must be sorted in ascending order');
64     end
65 end
66 % check window size
67 windowSize = input.Results.windowSize;
68 % restore the function handle
69 functionHandle = input.Results.functionHandle;
70
71 % options
72 % . window size mode: central, forward, backward
73 switch input.Results.WindowMode
74     case 'central'
75         windowLimLower = windowSize/2;
76         windowLimUpper = windowSize/2;
77     case 'forward'
78         windowLimLower = 0;
79         windowLimUpper = windowSize;
80     case 'backward'
81         windowLimLower = windowSize;
82         windowLimUpper = 0;
83 end
84
85 %% Apply the function
86 % initialise result array
87 result = nan(length(yData), 1);
88
89 for iIndex = 1:length(yData)
90     % get current windowSize
91     currentX = xData(iIndex);
92     selectY = (xData >= currentX-windowLimLower) & (xData <= currentX+
93         windowLimUpper);
94     result(iIndex) = functionHandle(yData(selectY));
95 end
96 end

```

---