Modeling and Simulation of Drop Size Distributions in Stirred Liquid-Liquid Systems

vorgelegt von
Diplom-Mathematikerin
Sonja Schlauch
aus Karlsruhe

Von der Fakultät II - Mathematik und Naturwissenschaften
der Technischen Universität Berlin
zur Erlangung des akademischen Grades
Doktorin der Naturwissenschaften
Dr. rer. nat.

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr. John M. Sullivan
Berichter:	Prof. Dr. math. Volker Mehrmann
Berichter:	Prof. Dr.-Ing. Matthias Kraume
Gutachter:	Prof. Dr. rer. nat. Stefan Turek

Tag der wissenschaftlichen Aussprache: 7.3.2007

Berlin 2007

D 83

Acknowledgment

First of all, I would like to express my gratitude to my advisor Prof. Dr. Volker Mehrmann for his support and guidance throughout my PhD project. He has been actively interested in my work and has always been available to advise me. I am very grateful for his helpful suggestions, valuable discussions, and encouragement.

Furthermore, I would like to thank Prof. Dr. Matthias Kraume and his working group for the good collaboration as well as for many valuable hints and discussions concerning the field of process engineering. Special thanks go to Ansor Gäbler, PD Dr. Anja Paschedag, and Sebastian Maaß.

I would like to thank Prof. Dr. Stefan Turek and his working group in Dortmund for trying to solve any problem I had with FeatFlow, in particular Dominik Göddeke and Andriy Sokolov.

I would like to thank all my colleagues from the working group "Numerical Analysis" for many fruitful discussions and for always lending me an open ear. Special thanks go to Ulrike Baur for proofreading this thesis, to Michael Schmidt for the collaboration, first in the "GRK" and later in the "SFB", and to Dr. Andreas Steinbrecher for answering my questions concerning the strangeness-index and for telling me how to hand in my dissertation.
Special thanks also go to the system administrators for solving all the smaller and larger problems "platon", "pythagoras", "tarski", and "legendre" were able to produce. Especially Jörn Beutner helped me so much with the simulations on the super computer "platon" that he really deserved to win **one** ticket for the world cup.
Another "thanks" goes to Prof. Dr. Fredi Tröltzsch, who warmly welcomed me to his working group.

Since my PhD project was part of an interdisciplinary "Graduiertenkolleg", I had the luck to work together with the nicest engineers of the world (including one physicist). I have really enjoyed our common lectures, seminars, lunch meetings, parties, and BTTs.

Of course, I would also like to thank my friends for a life besides work, but also Dr. Markus Middendorf for proof-reading and finding even the smallest typo. I thank Tobias Dussa for solving any TeX-nical problem.
I would like to express my gratitude to my parents for their love, encouragement, and support over the years.
Extraordinary thanks go to Thomas Schmelter, who did not only help me permanently in so many fields (calculating probabilities, answering mathematical questions, proof-reading, problems with LaTeX, converting graphics, ..., computers in general), but also gave me so much love and support and always had an open ear for my problems.

I would like to take this opportunity to thank the German Research Foundation (DFG) for the financial support through the GK 827 “Transport Phenomena with Moving Boundaries” and the SFB 557 “Control of Complex Turbulent Shear Flows”.

Abstract

In this thesis, the modeling and numerical simulation of drop size distributions in stirred liquid-liquid systems is considered. Therefore, the turbulent flow field in the stirred tank as well as the behavior of the drops is investigated.

The turbulent flow field is modeled by means of the Reynolds-averaged Navier-Stokes equations in combination with a k-ε turbulence model. The population dynamical processes, i. e., the behavior of the drops, are described by an averaged population balance equation, where the coalescence and breakage phenomena are modeled via integral terms.

In order to investigate the dynamical behavior of the coupled system, the index of the underlying differential algebraic equations, which are obtained after space-discretization, is determined. This analysis shows that the index of the semi-discretized Navier-Stokes equations is not increased by the coupling with the population balance equation, which means that a solver that is suitable for the solution of the Navier-Stokes equations can be extended so that it can also be used for the solution of the coupled system.

However, since the effort to build such a solver is immense, in this thesis, the coupling is realized by means of a simulator coupling approach. In this approach, the CFD code FeatFlow is used for the flow simulation, whereas the population balance solver Parsival is applied for the calculation of the drop size distributions.

Since it is justified for the considered application to neglect the influence of the drops on the flow field, only a one-way coupling is considered, regarding the modeling as well as the numerical simulation. As far as the numerical simulation is concerned, this means that only the results of the flow simulation are used for the calculation of the drop size distributions and not vice versa. According to these results, the stirred tank is then subdivided into compartments, and the data from the flow simulation are averaged with respect to these compartments so that they can be used for the simulation with Parsival.

Both, the results of the flow simulation as well as the drop size distributions derived by the simulation with Parsival, are validated by comparison with experimental data, which are taken from the literature or obtained by the experimental part of a joint project, respectively.

Besides the realization of the coupling, in this thesis, we also investigate the occurring difficulties in order to demonstrate the possibilities, but also the restrictions, of the presented approach and to discuss alternatives, like the use of the (direct) quadrature method of moments as another way to solve the population balance equation.

Zusammenfassung

Die hier vorliegende Arbeit beschäftigt sich mit der Modellierung und numerischen Simulation von Tropfengrößenverteilungen in gerührten Flüssig-/flüssig-Systemen. Hierfür wird einerseits die turbulente Strömung im Rührbehälter und andererseits das Verhalten der Tropfen betrachtet.

Zur Modellierung der turbulenten Strömung werden die Reynolds-gemittelten Navier-Stokes-Gleichungen zusammen mit einem k-ε-Turbulenzmodell verwendet. Die populationsdynamischen Phänomene (d. h. das Verhalten der Tropfen) werden mit Hilfe einer gemittelten Populationsbilanzgleichung modelliert, in der die Koaleszenz- und Bruchvorgänge über Integralterme beschrieben werden.

Um das dynamische Verhalten des gekoppelten Systems zu untersuchen, wird eine Indexanalyse der differentiell-algebraischen Gleichungen, die man nach der räumlichen Diskretisierung erhält, durchgeführt. Diese Analyse zeigt, dass sich der Index der semidiskretisierten Navier-Stokes-Gleichungen durch die Kopplung mit der Populationsbilanzgleichung nicht erhöht, d. h., dass ein zur Lösung der Navier-Stokes-Gleichungen geeigneter Löser (zumindest theoretisch) so erweitert werden kann, dass er zur Lösung des gesamten gekoppelten Systems verwendet werden kann.

Aufgrund des hohen Aufwands, der für die Implementierung eines solchen Lösers erforderlich ist, wird dieser Ansatz hier jedoch nicht weiter verfolgt. Stattdessen wird die Lösung des gekoppelten Systems mit Hilfe einer Simulator-Kopplung realisiert, bei der der CFD-Code FeatFlow zur Strömungssimulation und der Populationslöser Parsival zur Berechnung der Tropfengrößenverteilung verwendet wird.

Da es für die in dieser Arbeit betrachtete Anwendung gerechtfertigt ist, den Einfluss der Tropfen auf die Strömung zu vernachlässigen, wird sowohl bei der Modellierung als auch bei der numerischen Simulation nur eine Kopplung in eine Richtung („one-way coupling“) betrachtet. Für die numerische Simulation bedeutet das, dass nur die Ergebnisse der Strömungssimulation zur Berechnung der Tropfengrößenverteilung verwendet werden und nicht umgekehrt. Entsprechend dieser Ergebnisse wird der Rührbehälter dann in Kompartimente unterteilt, und die Strömungsdaten werden bezüglich dieser Kompartimente gemittelt, so dass sie für die Berechnung der Tropfengrößenverteilung in Parsival verwendet werden können.

Sowohl die Ergebnisse der Strömungssimulation als auch die in Parsival simulierten Tropfengrößenverteilungen werden mit Hilfe von experimentellen Daten, teils aus der Literatur, teils aus dem experimentellen Teil eines gemeinsamen Forschungsprojekts, validiert.

Neben der Umsetzung der Kopplung werden in dieser Arbeit auch die auftretenden Schwierigkeiten genauer beleuchtet, um die Möglichkeiten, aber auch die Beschränkungen des verwendeten Ansatzes aufzuzeigen und Alternativen, wie beispielsweise die Verwendung der (direkten) Quadraturmomentenmethode als alternative Lösung der Populationsbilanzgleichung, darzustellen.

Contents

1 Introduction

Stirred liquid-liquid systems are encountered in a large variety of technical processes, with major importance for chemical, pharmaceutical, mining, petroleum, and food industries. In these systems, two immiscible liquids are stirred so that one liquid disperses into the other one by building drops. The size distribution of these drops, resulting from the opposed phenomena of turbulent drop breakage and coalescence, plays an important role in the overall performance of many technical processes. Thus, the aim is to control the mean drop size and drop size distribution. In the production of polystyrene foam, for example, all drops in the system should have a certain, pre-defined drop size. For liquid-liquid systems with chemical reactions taking place at the interface between the two phases, the ratio between the surface area and the volume of the drops should be maximal, which would be achieved by the smallest possible drop size. However, one usually has the restriction that the drops still have to be so large that the phases can be separated after the reaction.
In order to control the drop size distribution in a stirred tank reactor, first a good modeling and numerical simulation of the processes is required so that the drop size distributions can be predicted for the considered application.

Dispersed systems are usually modeled by means of population balance equations, which describe the rate of change of drop size distributions due to inflow, outflow, convective transport, and population dynamical phenomena, see [HK64], [Ger99]. In this approach, the drops are described by a so-called number density function, which is, besides the usual dependency on space and time, also dependent on so-called internal coordinates, like the size or other significant characteristics of the drops. These additional coordinates are needed in order to distinguish the drops from each other.
Since each additional coordinate increases the dimension of the problem by one, in many applications the assumption of an ideally mixed tank is used, which means that the dependency on space is neglected. For this case, a lot of modeling approaches can be found in the literature, see, e. g., [CT77], [BS91], [TT94], [LS96], [AKK99], [MML99a], [MML99b], [DO02a], [DO02b], [LEMM02], [Rit02], or [WWJ03].

Since the flow field, which highly influences the coalescence and breakage rates, is usually very inhomogeneous throughout the tank, such a simplification is often too restrictive. For example, near the impeller, where turbulence is several orders of magnitude higher than in the bulk of the tank (see, e. g., [AKK99]), mainly drop breakage occurs, whereas coalescence only takes place in the less turbulent regions. The reason for this is that colliding drops are only able to coalesce if they stay in contact long enough, which is not possible if the flow is too turbulent. Thus, for an appropriate description of the processes in a stirred tank, these spatial inhomogeneities have to be taken into account.

However, the solution of the "full" coupled system, consisting of the Navier-Stokes equations, the population balance equation, and, if applicable, additional equations for modeling turbulence, is too extensive for most applications. Thus, it is reasonable to find a compromise between detailed modeling of the different phenomena and savings in the computational time.
Therefore, a compartment model approach is used in this thesis. While the flow simulation is done on a relatively fine grid, for the calculation of the drop size distributions, several cells are combined to one compartment. Within each compartment, the drop size distribution is assumed to be space-independent, and the compartments are chosen in such a way that this is (at least approximately) satisfied. The inhomogeneity of the flow field is taken into account by using different mean energy dissipation rates for the calculation of the drop size distributions in the different compartments. Furthermore, the exchange between the zones is modeled via volume flows, which are calculated from averaged velocities derived from the flow simulation.
The use of such compartment model approaches in the context of multi-phase systems can already be found in the literature for different applications, see, e. g., [AKK99], [AKKM02], [LAA02], [PB], [VDV02], [WR05], or [KBM06]. The idea of these approaches is to take into account how the turbulent flow field in the different regions of the tank influences the population dynamical (or other) processes, but, at the same time, to avoid the enormous computing times needed for solving the full coupled system.

In this thesis, the coupled system consisting of fluid flow and population balance equations is solved by applying a simulator coupling approach using the CFD (short for computational fluid dynamics) code FeatFlow for the flow simulation and the population balance solver Parsival for the calculation of the drop size distributions. The coupling is realized as a one-way coupling, which means that the CFD results are used for the calculation of the drop size distributions, but the influence of the drops on the flow field is neglected.
However, in contrary to most approaches in the literature, a non-commercial code is used for the flow simulation in this thesis. The reason for the choice of a non-commercial CFD code is that the source code is freely available and can be adapted, which is, on one hand, already advantageous for the coupling and, on the other hand, essential for further applications, like control or optimization tasks, where additional equations need to be implemented.
Furthermore, besides the realization of the one-way coupling approach, in this thesis, the coupling is also investigated from a more theoretical point of view. The system of equations, describing the turbulent flow field in the stirred tank and the behavior of the drops, is derived and its dynamics are analyzed by means of determining an index of the underlying differential-algebraic system. Such an analysis gives information about how strong the coupling is and what problems may arise when the equations are solved numerically.

Since this thesis was originated in the context of a joint project together with engineers from the Institute for Process Engineering at the TU Berlin, a validation of the simulation results was possible by comparing the simulated drop size distributions with

experimental data. Such a combination of experimental investigations, modeling, and numerical simulation, is essential for understanding the processes in a stirred tank reactor, and it forms the basis for an appropriate description and an enhancement of the mathematical models.

The thesis is organized as follows. In the following chapter, the system of equations modeling the processes in the stirred tank reactor is derived. The turbulent flow field in the stirred tank is described by the Reynolds-averaged Navier-Stokes equations with k-ε turbulence model, whereas the behavior of the drops is modeled by means of an averaged population balance equation. The coalescence and breakage phenomena are taken into account by means of integral terms on the right-hand side of the population balance equation. For these terms, several models from the literature are presented, and it is discussed how they can be adapted so that they fit to the considered framework. Even though the modeling is mainly considered for the investigated application, also extensions concerning two-way coupling on one hand and control on the other hand are discussed.

In Chapter 3, the dynamical behavior of the coupled system is analyzed. The index of the underlying differential-algebraic equation, which is derived after space discretization, is determined in order to get information about numerical problems that may arise. Such an analysis is particularly important in consideration of control and optimization tasks, which are the long-term objective for the considered application. It is shown that the index of the semi-discretized Navier-Stokes equations is not increased by the coupling, and thus existing algorithms that are suitable for the solution of the Navier-Stokes equations can, at least theoretically, be extended so that they are able to treat the coupled system.
However, since the effort to build such a solver is immense, this approach is currently not feasible. Thus, in this thesis, the already mentioned simulator coupling approach between FeatFlow and Parsival is applied.

In Chapter 4, the simulation of the flow field in the stirred tank using the non-commercial code FeatFlow is discussed. First, besides a brief introduction to the mathematical background of FeatFlow, we give a detailed description of the preparations that need to be done before a simulation of the flow field in a stirred tank can be started. Afterwards, numerical results are presented and compared with experimental data from the literature. In addition, the implementation of a k-ε turbulence model in FeatFlow is described. The chapter closes with a discussion of the pros and cons concerning the use of a turbulence model for the considered application.

In Chapter 5, the coupling between FeatFlow and Parsival is investigated. After a short description of the mathematical algorithm implemented in Parsival and a summary of some previous simulation results obtained with Parsival, the realization of the coupling approach is described in detail. Afterwards, simulation results for a variety of different test cases as well as for the compartment model representing the considered stirred tank reactor are presented. For the latter one, the simulated drop size distributions are validated by comparison with experimental data obtained within the joint

project. At the end of the chapter, after a discussion of the presented approach, an introduction to the (direct) quadrature method of moments is given. This approach represents a very promising alternative for solving the population balance equation, in particular, if it is used within a coupling approach.

Finally, Chapter 6 summarizes the opportunities and drawbacks of the presented coupling approach and gives an outlook to future investigations.

In the two appendices of this thesis, the derivation of the Navier-Stokes and Reynolds-averaged Navier-Stokes equations, on one hand, and of a general population balance equation, on the other hand, is described.
Furthermore, all used variables and abbreviations are explained in the nomenclature at the end of this thesis.

2 Modeling

In this chapter, we will consider the modeling of stirred liquid-liquid systems, i. e., of systems, where two immiscible fluids are stirred so that one of the phases disperses into the other one by building droplets. All droplets together are called the dispersed phase, whereas the surrounding fluid is called the continuous phase.

However, before we will start with the derivation of the model equations, we will briefly describe an application, which will be considered as practical example throughout this thesis.

Considered application. *The following application originates from a joint project together with engineers from the Institute for Process Engineering at the TU Berlin. In this project, a liquid-liquid system inside a cylindric tank with dished bottom (so-called torispherical head) stirred by a six-bladed Rushton turbine was investigated. The tank was equipped with four baffles, which were installed in a small distance to the wall, see Fig. 2.1.*
The vessel had an (inner) diameter of $D_{tank} = 0.15\,\mathrm{m}$ *and a height of* $H_{tank} = D_{tank} = 0.15\,\mathrm{m}$. *The Rushton turbine with diameter* $D_{stirrer} = {}^{D_{tank}}/_{3} = 0.05\,\mathrm{m}$ *was mounted at* $^1/_3$ *of the tank height. The baffles had a width of* $D_{baffle} = 0.013\,\mathrm{m}$. *The exact mathematical description (including the dimensions of all the other parts of the system) can be found in Section 4.2.*
The vessel was filled with water as continuous and toluene as dispersed phase, whereof the latter one had a volume fraction of $\chi = 0.1$. *The two fuids were stirred with constant rotational speeds, varying from* 400 *to* 700 *rounds per minute.*
The whole system was kept at the constant temperature of $20\,^{\circ}\mathrm{C}$.

Generally, in two-phase systems the interaction between the continuous and the dispersed phase leads to different population phenomena, like, on one hand, formation, dissolution, growth, and shrinkage (which are induced by the heat and mass transfer between the two phases) and, on the other hand, coalescence and breakage (which are caused by fluid dynamic processes and momentum transfer between the two phases). Here, coalescence describes the process, where a larger drop is generated because of the collision and confluence of two smaller drops. Breakage, on the other hand, is the process, where a drop collides with an eddy and, hence, breaks up into some smaller drops. For information about the other population phenomena, see Appendix B.
Note that in the liquid-liquid system considered here, only coalescence and breakage occur, since there is no heat and mass transfer between the two phases.

For an appropriate description of the considered stirred liquid-liquid system, not only

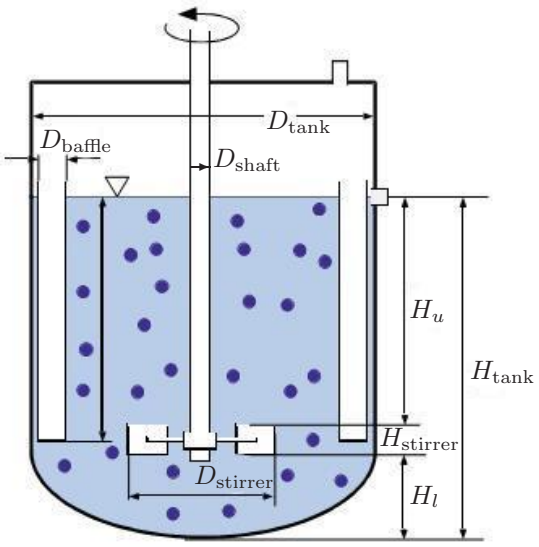


Figure 2.1: Schematic representation of the considered stirred liquid-liquid system.

the population dynamical processes but also the flow field in the stirred tank has to be modeled, since it has a strong influence on the coalescence and breakage phenomena. For example, breakage mainly occurs in the turbulent region close to the impeller, whereas coalescence only takes place in the less turbulent parts of the tank, where the drops are able to stay in contact as long as coalescence needs to take place (see, e. g., [AKK99]).

In the following sections, we will derive a mathematical model describing the above mentioned processes occurring in the considered stirred liquid-liquid system. First, in Section 2.1, we will introduce so-called internal coordinates describing some important characteristics of the drops. Then, in Section 2.2, the flow field in the stirred tank will be modeled by the Navier-Stokes equations for incompressible fluids. Furthermore, turbulence will be accounted for by Reynolds-averaging and then solving the arising closure problem by using a k-ε turbulence model. Finally, in Section 2.3, the behavior of the drops is modeled by a population balance equation, where coalescence and breakage appear as source and sink terms on the right-hand side.

2.1 Internal and External Coordinates

For modeling the processes occurring in a stirred tank appropriately, we do not only need space coordinates $x = [x_1, x_2, x_3]^T \in \Omega \subset \mathbb{R}^3$ and a time coordinate $t \in [0, t_{\text{end}}]$, but also so-called internal coordinates $e = [e_1, \ldots, e_a]^T \in \mathbb{R}^a$, which describe a different properties of the dispersed phase. We need these additional coordinates to distinguish

the individuals of the dispersed phase (i. e., the drops) from each other with respect to their significant characteristics.

Note that, in this context, when internal coordinates are used, the space coordinates are called external coordinates accordingly. Both internal and external coordinates together form a space, the so-called particle phase space, see [HK64]. Any individual can be represented by a point in this particle phase space. The coordinates of this point identify the phase of the particle – i. e., location and characteristics – uniquely.

In general, the internal coordinates describe properties related to geometry (e. g., size, volume), material properties (e. g., density, chemical composition, color), or characteristics in the interaction with the continuous phase (e. g., surface charge). These coordinates may depend on each other (e. g., mass and volume), some of them are discrete (e. g., number of smaller particles / atoms the considered particle consists of), others are continuous (e. g., mass).

In the discussed application, it is sufficient to distinguish the drops due to their size. Therefore, in the following, we only use one internal coordinate, namely the volume V. Note that, in the literature, often the diameter d of a drop is used as internal coordinate. That is also why most of the formulas in Subsection 2.3.1 are also given in terms of the diameter. But since in the considered application the drops are assumed to be spherical, which means that the diameter d of a drop is directly related to its volume V by the relation $V = c_V d^3$ with $c_V = \frac{\pi}{6}$, the formulas from Subsection 2.3.1 can easily be transformed such that they depend on the volume V.

Remark 2.1. *In the simulation package* Parsival *[Wul], which is used for the solution of the population balance equation within the joint project, the volume as well as the diameter can be chosen as internal coordinate. However, some processes are harder to implement if the volume is used. The coordinate transformations, which describe how to switch between volume and diameter representation in the different terms of the population balance equation, can be found in the* Parsival *tutorial [GB04].*

2.2 Modeling the Flow

In this section, we will model the flow in the stirred tank. At this stage of modeling we assume that the drops do not have any influence on the flow field. (They are just moving with the fluid with the same velocity.) This implies, in particular, that the description of the flow is independent of the internal coordinates. However, we will comment on this later.

The flow field in the stirred tank is described by the Navier-Stokes equations for incompressible fluids (cf. Eqs. (A.2) and (A.6) in Appendix A)

$$\left.\begin{aligned} \tfrac{\partial}{\partial t}u + \nabla\cdot(u\otimes u) + \tfrac{1}{\rho_f}\nabla p &= \nu\Delta u + g \\ \nabla\cdot u &= 0 \end{aligned}\right\} \quad \text{in } \Omega\times[0,t_{\text{end}}], \tag{2.1}$$

where u is the velocity of the fluid, p is the pressure, and g are the external forces. In the discussed application, the only external force that has to be considered is gravity. Furthermore, ρ_f is the density of the fluid, $\nu = \frac{\mu}{\rho_f}$ is the kinematic viscosity, and μ is the dynamic viscosity. The first equation in (2.1) accounts for the conservation of momentum and is therefore called momentum equation. Here, $\frac{\partial}{\partial t}u + \nabla \cdot (u \otimes u)$ describes the acceleration of the particles of the fluid, ∇p is the pressure gradient, and $\nu \Delta u$ accounts for the friction between the particles of the fluid. The second equation in (2.1) is called the continuity equation and models the conservation of mass.

Additionally, we prescribe the following boundary conditions on $\partial\Omega =: \Gamma = \Gamma_{\text{stirrer}} \cup \Gamma_{\text{wall}} \cup \Gamma_{\text{surface}}$:

$$u = u_{\text{stirrer}} \text{ on } \Gamma_{\text{stirrer}}, \quad u = 0 \text{ on } \Gamma_{\text{wall}}, \quad \frac{\partial u_1}{\partial \mathfrak{n}_1} = \frac{\partial u_2}{\partial \mathfrak{n}_2} = 0,\ u_3 = 0 \text{ on } \Gamma_{\text{surface}},$$

where Γ_{stirrer} describes the stirrer, Γ_{wall} the fixed walls, and Γ_{surface} the surface of the liquid in the tank. Furthermore, u_1, u_2, and u_3 are the velocities in x_1-, x_2-, and x_3-direction, and $\mathfrak{n}_1$ and $\mathfrak{n}_2$ are the outer normal vectors in x_1- and x_2-direction, respectively. The prescribed velocity on the stirrer u_{stirrer} is given by

$$u_{\text{stirrer}} = [-r \sin(\varphi)\omega, r \cos(\varphi)\omega, 0]^T,$$

where the radius r and the angle φ are defined by

$$r = \sqrt{x_1^2 + x_2^2}, \quad \varphi = \arccos\left(\frac{x_1}{r}\right),$$

and ω is the constant angular velocity. This angular velocity is determined by the adjusted rotational speed N^*, i. e., the number of revolutions per time unit.
The initial condition is chosen such that it is consistent with the boundary conditions.

Remark 2.2. *Note that the parameter N^* can be used as a control input in order to influence the drop size distribution. In this case, the assumption that ω is constant should be dropped. Further information about how the control of the drop size distribution can be realized will be given in Remark 2.9 at the end of this chapter.*

Let us now consider how turbulence can be modeled. Theoretically, the flow field of every incompressible, viscous fluid can be described by the Navier-Stokes equations, regardless if it is laminar or turbulent. However, if this approach is used to compute a turbulent flow field, the Navier-Stokes equations must be discretized and solved on such a fine grid that even the smallest eddies are resolved. Since the size of the smallest eddies is much smaller than the size of the computational domain, many grid points are needed. In fact, in 3D the number of required grid points is proportional to $\text{Re}^{9/4}$, see [GDN98], where Re denotes the Reynolds number. Thus, this approach is only feasible for simple geometries or problems with low (or medium) Reynolds numbers.
Since these assumptions are not satisfied for the considered application, we will in the following apply the so-called "Reynolds-averaging" to the Navier-Stokes equations in

order to avoid the problems described above. In this approach, the Navier-Stokes equations are only solved on a relatively coarse grid (i. e., with a manageable number of grid points), by which the main flow field can be calculated, whereas the effects of the eddies that are not resolved by this grid are modeled separately. The idea behind this is that in most practical investigations of turbulent flow, one is not interested in every microscopic detail, but only in macroscopically observed mean values, see [GDN98]. In addition, all the details of turbulence cannot be measured in experiments, too. In engineering science, when turbulent flows are considered, one is mostly interested in the mean flow rate or the flow resistance of an obstacle. Therefore, only averaged quantities need to be considered, but not every small variation. As already mentioned before, these small variations will be "summarized" in the turbulence model.

Thus, we split each quantity – velocity u, pressure p, and external forces g – into a mean part $\bar{u}$, $\bar{p}$, and $\bar{g}$, respectively, and into the so-called fluctuations u', p', and g', which model the smallest variations of each quantity:

$$u = \bar{u} + u', \quad p = \bar{p} + p', \quad g = \bar{g} + g'.$$

The mean part is usually chosen as a component-wise temporal averaged value. However, in general, also other averages can be used. This can be expressed by applying a filter $\langle\cdot\rangle$, i. e.,

$$\bar{u} := \langle u \rangle, \quad \bar{p} := \langle p \rangle, \quad \bar{g} := \langle g \rangle.$$

After introducing these relations into the Navier-Stokes equations and averaging them (i. e., applying the filter operator $\langle\cdot\rangle$ to them), we get the Reynolds-averaged Navier-Stokes equations

$$\left.\begin{aligned} \tfrac{\partial}{\partial t}\bar{u} + \nabla\cdot(\bar{u}\otimes\bar{u}) + \tfrac{1}{\rho_f}\nabla\bar{p} - \nu\Delta\bar{u} + \nabla\cdot\langle u'\otimes u'\rangle &= \bar{g} \\ \nabla\cdot\bar{u} &= 0 \end{aligned}\right\} \quad \text{in } \Omega\times[0,t_{\text{end}}], \tag{2.2}$$

where the tensor

$$R(u') := -\langle u'\otimes u'\rangle$$

is known as Reynolds stress tensor.

Remark 2.3. *Note that the term "Reynolds-averaged Navier-Stokes equations" usually denotes the temporal average of the Navier-Stokes equations, at least in engineering science. However, another kind of averaging could be applied analogously and would lead to the same form of equations.*

Comparing the equations given in (2.2) with the Navier-Stokes equations (2.1) shows that the averaging has led to almost the same equations, just with the additional term $-\nabla\cdot R(u')$. Since R depends on the fluctuations u' of the velocities, the system consists of one more unknown variable u' so that the Reynolds-averaged Navier-Stokes equations do not form a closed system any more, i. e., there are more unknowns than equations. This difficulty is known as the closure problem in turbulence modeling, see [GDN98].
In order to solve this problem, one has to introduce new equations, which are usually

based on hypotheses and approximations given by empirical information or experimental data. These equations form the so-called turbulence model.

The probably most widely used turbulence model is the k-ε model introduced in [LS74]. It is a so-called two-equations model, which means that two additional partial differential equations are introduced in order to get a closed system. In the k-ε model, the following two additional variables are used to model the Reynolds stresses: the turbulent kinetic energy k and its dissipation rate ε, given by

$$k := \frac{1}{2}\langle \|u'\|_F \rangle, \quad \varepsilon := \frac{\nu}{2}\langle \|\nabla u' + \left(\nabla u'\right)^T\|_F^2 \rangle,$$

where $\|\cdot\|_F$ denotes the Frobenius norm. (The definition of the Frobenius norm can be found in Appendix A.)
In the k-ε model the Reynolds tensor $R(u')$ is approximated by

$$R(u') \approx \mathcal{R}(\nabla \bar{u}, k, \varepsilon) := -\frac{2}{3}kI + c_\mu \frac{k^2}{\varepsilon}(\nabla \bar{u} + (\nabla \bar{u})^T),$$

where I denotes the identity matrix and c_μ is a constant. If we insert this into the averaged momentum equation, i. e., into the first equation in (2.2), and set $\nu^* := \nu + \nu_t$ with

$$\nu_t = c_\mu \frac{k^2}{\varepsilon}, \tag{2.3}$$

this leads to

$$\frac{\partial}{\partial t}\bar{u} + \nabla \cdot (\bar{u} \otimes \bar{u}) + \frac{1}{\rho_f}\nabla \bar{p} + \frac{2}{3}\nabla k - \nabla \cdot (\nu^*(\nabla \bar{u} + (\nabla \bar{u})^T)) = \bar{g}. \tag{2.4}$$

Together with the two additional transport equations for k and ε, which close the system of equations, the following system is derived:

$$\left.\begin{array}{rcl} \frac{\partial}{\partial t}\bar{u} + \nabla \cdot (\bar{u} \otimes \bar{u}) + \frac{1}{\rho_f}\nabla \bar{p} + \frac{2}{3}\nabla k - \nabla \cdot (\nu^*(\nabla \bar{u} + (\nabla \bar{u})^T)) & = & \bar{g} \\ \nabla \cdot \bar{u} & = & 0 \\ \frac{\partial}{\partial t}k + \bar{u} \cdot \nabla k - \frac{\nu_t}{2}\|\nabla \bar{u} + (\nabla \bar{u})^T\|_F^2 - \nabla \cdot (\nu_t \nabla k) + \varepsilon & = & 0 \\ \frac{\partial}{\partial t}\varepsilon + \bar{u} \cdot \nabla \varepsilon - \frac{c_1 c_\mu}{2}k\|\nabla \bar{u} + (\nabla \bar{u})^T\|_F^2 - \nabla \cdot \left(\frac{1}{c_\varepsilon}\nu_t \nabla \varepsilon\right) + c_2 \frac{\varepsilon^2}{k} & = & 0 \end{array}\right\} \quad \text{in } \Omega \times [0, t_{\text{end}}], \tag{2.5}$$

where the constants c_μ, c_ε, c_1, and c_2 are chosen from experimental investigations. In [LS74], for example, the following values are proposed:

$$c_\mu = 0.09, \quad c_\varepsilon = 1.3, \quad c_1 = 1.44, \quad c_2 = 1.92.$$

For the averaged velocity $\bar{u}$ we can use the same boundary conditions as prescribed for the velocity u in the Navier-Stokes equations (2.1) before. Therefore, we set:

$$\bar{u} = u_{\text{stirrer}} \text{ on } \Gamma_{\text{stirrer}}, \quad \bar{u} = 0 \text{ on } \Gamma_{\text{wall}}, \quad \frac{\partial \bar{u}_1}{\partial \mathfrak{n}_1} = \frac{\partial \bar{u}_2}{\partial \mathfrak{n}_2} = 0, \ \bar{u}_3 = 0 \text{ on } \Gamma_{\text{surface}} \tag{2.6}$$

with the same notation as above.
For k and ε we additionally prescribe the following boundary conditions:

$$k = 0 \text{ on } \Gamma_{\text{stirrer}} \cup \Gamma_{\text{wall}}, \quad \frac{\partial k}{\partial \mathfrak{n}} = 0 \text{ on } \Gamma_{\text{surface}}, \tag{2.7}$$

and

$$\frac{\partial \varepsilon}{\partial \mathfrak{n}} = 0 \text{ on } \Gamma, \tag{2.8}$$

where $\mathfrak{n}$ denotes the outer normal vector.
Again, the initial conditions for $\bar{u}$, k, and ε are chosen such that they are consistent with the boundary conditions.

2.3 Modeling the Population Dynamical Processes

The population dynamical processes can be modeled using the following general population balance equation from [Ger99] (cf. Eq. (B.3) in Appendix B)

$$\frac{\partial f(e,x,t)}{\partial t} = -\nabla_{x,e} \cdot \phi_N(e,x,t) + s(e,x,t),$$

where $f(e,x,t)$ is the so-called number density function depending on internal and external coordinates e and x as well as on time t. The quantity f describes how many drops with internal coordinate e are located at a certain time t at a certain point x in the sense that the total number of drops N_{drops} at time t is given by

$$N_{\text{drops}}(t) = \int_{\Omega_x} \int_{\Omega_e} f(e,x,t)\, de\, dx,$$

where Ω_x and Ω_e denote the geometrical space and the space of internal coordinates, respectively. Furthermore, $\nabla_{x,e}\cdot$ denotes the divergence operator with respect to internal and external coordinates. The right-hand side consists of the transport density $\nabla_{x,e} \cdot \phi_N$ and the sum of source and sink terms s. The transport density $\nabla_{x,e} \cdot \phi_N(e,x,t)$ can be split up into the transport density concerning the internal coordinates $\nabla_e \cdot \phi_{N,e}$ and the one concerning the external coordinates $\nabla_x \cdot \phi_{N,x}$:

$$\nabla_{x,e} \cdot \phi_N(e,x,t) = \nabla_x \cdot \phi_{N,x}(e,x,t) + \nabla_e \cdot \phi_{N,e}(e,x,t),$$

where the ∇-operators are defined by

$$\nabla_x := \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_3} \end{bmatrix} \quad \text{and} \quad \nabla_e := \begin{bmatrix} \frac{\partial}{\partial e_1} \\ \vdots \\ \frac{\partial}{\partial e_a} \end{bmatrix}.$$

Furthermore, the transport flows $\phi_{N,x}$ and $\phi_{N,e}$ can be divided into a convective and a diffusive part:

$$\begin{aligned} \phi_{N,x} &= w_x \cdot f(e,x,t) + \phi^D_{N,x}, \\ \phi_{N,e} &= w_e \cdot f(e,x,t) + \phi^D_{N,e}. \end{aligned}$$

Here, the convective part in x-direction $w_x \cdot f(e,x,t)$ describes the deterministic movement of the particles. The diffusive part $\phi^D_{N,x}$ can be used to account for the Brownian motion of the particles. The convective part in direction of the internal coordinates states the rate of change of the single particle properties. The diffusive part can be used to describe the stochastic variation of the rate of change.

In the discussed application (see page 5), we consider a system without changes in the internal coordinates per time, which means that there is no growth or shrinkage of droplets due to mass transfer or reaction. Therefore, we get $\nabla_e \cdot \phi_{N,e}(e,x,t) = 0$. Furthermore, the Brownian motion is assumed to be very small compared to the deterministic particle movement and, hence, can be neglected. Additionally, we can approximate the velocity w_x by the velocity u of the surrounding continuous phase. (The drops are just moving with the surrounding fluid.)

Remark 2.4. *As already mentioned in the beginning of Section 2.2, if one omits the assumption that the drops are just moving with the continuous phase and that they do not have any influence on the flow field, a more complex modeling approach is required. If we skip the assumption that all the drops are moving with the same velocity as the continuous phase, a multi-phase modeling approach has to be used, which includes additional equations for the "movement" of the drops. If one assumes that all the drops are moving with the same velocity, a two-phase model suffices, otherwise additional equations for each "group" of drops moving with the same velocity are required.*
If we allow the drops to have an influence on the continuous phase (so-called "two-way coupling"), an additional source term, which describes the influence of the number and sizes of the drops on the flow field, is required in the (Reynolds-averaged) Navier-Stokes equations.

However, for the considered application (see page 5), both assumptions (i. e., that the drops are just moving with the continuous phase and that they have a negligible influence on the flow field) are justified, since the densities of the continuous and the dispersed phase are almost equal and the volume fraction of the dispersed phase is sufficiently small.

With the simplifications described above and using just one internal coordinate, namely the volume V, we get the following population balance equation:

$$\frac{\partial f(V,x,t)}{\partial t} = -\nabla_x \cdot (u \cdot f(V,x,t)) + s(V,x,t). \tag{2.9}$$

Note that the divergence operator $\nabla_x \cdot$ used here is equal to the divergence operator $\nabla \cdot$ in the Navier-Stokes equations (2.1), since we do not have any derivatives with respect to the internal coordinate V in both cases. Therefore, we will in the following use $\nabla \cdot$ instead of $\nabla_x \cdot$.

For f, we prescribe the following initial condition:

$$f(V,x,t)|_{t=0} = f_0(V,x), \tag{2.10}$$

where – for simplicity and for fixed x – f_0 is usually chosen to be proportional to the density of a Gaussian normal distribution. Additionally, we use as boundary conditions

$$f(V, x, t) = 0 \text{ for } x \in \Gamma, \tag{2.11}$$

$$f(V_{\min}, x, t) = f(V_{\max}, x, t) = 0, \tag{2.12}$$

where $V_{\min}$ and $V_{\max}$ denote the minimal and maximal drop volumes.

Remark 2.5. *Note that in the numerical simulation usually not the minimal and maximal possible drop volumes are used as boundary values in Eq. (2.12), but limits according to the considered application are chosen. By this, fewer drop size classes are needed and the efforts for the numerical solution are reduced. However, this may lead to problems concerning the conservation of mass if these limits are not chosen appropriate and drops of sizes outside the interval (V_{min}, V_{max}) are generated by coalescence or breakage. Note that, in some applications, limits that have been chosen in the beginning do not need to remain suitable after some time if the drop size distribution has changed drastically.*

Furthermore, s is the sum of source and sink terms describing the population dynamical processes. In the considered application, only coalescence and breakage can be observed. Therefore, the term s can be divided into

$$s = s^+_{\text{coal}} + s^-_{\text{coal}} + s^+_{\text{break}} + s^-_{\text{break}},$$

where $s^\pm_{\text{coal}}$ and $s^\pm_{\text{break}}$ are describing the rate of increase / decrease of the number density function f due to coalescence and breakage, respectively. The four different terms occurring on the right-hand side are explained graphically in Fig. 2.2. Note that we only consider binary coalescence (since the probability that three or more drops collide in a time interval $(t, t + dt)$ at a certain point is negligibly small compared to the probability that two drops meet), but allow breakage of a drop into two or more daughter drops. In the following, we will discuss the mathematical representation of these terms.

Let us start with the coalescence process. The source term s^+_{coal} due to coalescence accounts for all drops that are formed by coalescence of two smaller drops. So, the source term can be written as (cf. Eq. (B.7) in Appendix B)

$$s^+_{\text{coal}}(V, x, t) = \int_0^V R_{\text{coal}}(V', V'', y(x, t)) f(V', x, t) f(V'', x, t)\, dV',$$

where V' and V'' are the volumes of the two coalescing drops. Since the drops with volumes V' and V'' form a drop with volume V, the following relation must hold:

$$V'' = V - V'.$$

The function R_{coal} describes the probability that two individuals coalesce. The vector y denotes the so-called continuous phase vector (see, e. g., [Ram00]), i. e., the properties

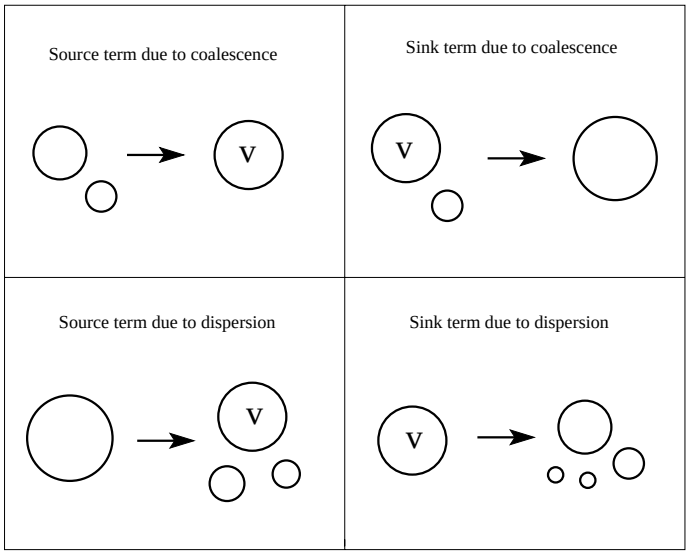


Figure 2.2: Source and sink terms for a drop of volume V due to coalescence and breakage.

of the continuous phase that influence the coalescence process. Thus, the vector y may depend on pressure, temperature, or other values that we get from the calculation of the flow field.
Note that the lower limit of the integral could also be substituted by the minimal possible volume of a drop $V_{\min}$.

Remark 2.6. *It is clear that theoretically there is no difference between the two formulations, since the number density function has to fulfill the condition $f(V, x, t) = 0$ for all $V \notin [V_{min}, V_{max}]$.*
However, in practice, it is very difficult to determine V_{min} and V_{max} for a certain application. Therefore, it can really make a difference which formulation is used. If the integration limit is chosen too small, drops of "too small" size are formed that do not occur in reality. On the other hand, if the limit is chosen too big, we may forget some drops that are really existent.

The sink term due to coalescence accounts for the droplets that are lost, because they form a larger one together with another droplet. Mathematically, the sink term due to coalescence is given by integration over all droplets that are able to coalesce with a given one (cf. Eq. (B.6)):

$$s^-_{\text{coal}}(V, x, t) = -f(V, x, t) \int_0^{V_{\max}-V} R_{\text{coal}}(V, V', y(x, t)) f(V', x, t) dV',$$

where $V_{\max}$ denotes the maximal possible volume of a drop in the liquid-liquid system.

Breakage can, like coalescence, mathematically be modeled by a source and a sink term. The source term describes the generation of daughter droplets, whereas the sink term stands for the loss of the mother drop.

The source term due to breakage is given by (cf. Eq. (B.5))

$$s^+_{\text{break}}(V,x,t) = \int_V^{V_{\max}} n(V', y(x,t))\gamma(V,V',y(x,t))R_{\text{break}}(V',y(x,t))f(V',x,t)dV',$$

where $n(V', y(x,t))$ is the number of daughter drops formed by breakage of a drop with volume V', $\gamma(V,V',y(x,t))$ is the distribution of daughter drops that describes the probability that a daughter drop with volume V is formed by the breakage of a mother drop with volume V', and $R_{\text{break}}(V',y(x,t))$ is the breakage rate that accounts for the number of dispersed drops per unit time. Since γ is a probability density function, it has to fulfill the following normalization condition:

$$\int_0^{V_{\max}} \gamma(V,V',y(x,t))dV = 1. \tag{2.13}$$

Conservation of mass leads to the additional condition:

$$n(V',y(x,t)) \int_0^{V'} m(V)\gamma(V,V',y(x,t))\, dV = m(V'), \tag{2.14}$$

where $m(V)$ denotes the mass of a drop with volume V.

The sink term due to breakage accounts for the drops that are lost, because they break up into smaller droplets. It is given by (cf. Eq. (B.4))

$$s^-_{\text{break}}(V,x,t) = -R_{\text{break}}(V,y(x,t))\ f(V,x,t).$$

The coalescence rate R_{coal}, the breakage rate R_{break}, the distribution of daughter drops γ as well as the number of daughter drops n have to be specified for certain coalescence and breakage processes depending on the considered application.

In Subsection 2.3.1, we will describe in detail some approaches how these functions can be modeled in the considered application. We will investigate three different models, which have been introduced in [CT77], [TT94], and [Rit02]. In all these models, it is assumed that the dispersed phase is homogeneously distributed in the physical space, i. e., that the tank is ideally mixed. In this case, a space-independent population balance equation can be used to describe the processes in the stirred tank. In addition, in these approaches, there does not occur any transport term in the population balance equation, but only changes of the number density function due to coalescence and breakage. But although we use a more general approach (namely a space-dependent number density function $f(V,x,t)$ and changes in f due to convective transport, see Eq. (2.9)), we can nevertheless use the coalescence and breakage rates proposed there. In the model

considered here, we account for the space-dependency in the coalescence and breakage rates by substituting the constant value ε_l used in the three mentioned papers by the energy dissipation rate ε (from the k-ε model) at a certain point (x, t).

Let us now consider how turbulence can be modeled for the population balance equation. For this, we use the same approach as for the Navier-Stokes equations, i. e., we split all occurring quantities, which here are the number density function f and the velocity u, into mean values $\bar{f}$ and $\bar{u}$, and fluctuations f' and u'. Again, we regard $\bar{f}$ and $\bar{u}$ as averaged values, received by a filter operator $\langle\cdot\rangle$:

$$\begin{aligned} f &= \bar{f} + f', \quad \bar{f} = \langle f \rangle, \\ u &= \bar{u} + u', \quad \bar{u} = \langle u \rangle. \end{aligned}$$

Inserting this into Eq. (2.9), we get

$$\frac{\partial(\bar{f} + f')}{\partial t} = -\nabla \cdot ((\bar{u} + u')(\bar{f} + f')) + s.$$

Applying the filter operator $\langle\cdot\rangle$ to this equation, we then get the averaged population balance equation

$$\frac{\partial \bar{f}}{\partial t} = -\nabla \cdot (\bar{u}\bar{f}) - \nabla \cdot \langle u' f' \rangle + \bar{s},$$

where $\bar{s}$ is the mean value of s, i. e., $\bar{s} = \langle s \rangle$. This term represents the averaged source and sink terms.

According to [Pas04b], the term $\nabla \cdot \langle u' f' \rangle$ is modeled in analogy to the approach for the Reynolds stress tensor:

$$\nabla \cdot \langle u' f' \rangle = -\nabla \cdot (c_t \nabla \bar{f}),$$

where c_t is the so-called turbulent diffusion coefficient, which is (like the turbulent viscosity ν_t) not a fluid property, but a function of k and ε. Similar to the Reynolds stress tensor, which is assumed to be proportional to the gradient of the velocity, it is assumed here that the term $\langle u' f' \rangle$ is proportional to the gradient of the averaged number density function $\nabla \bar{f}$. Furthermore, c_t is specified by the turbulent Schmidt number Sc_t

$$\mathrm{Sc}_t = \frac{\mu_t}{\rho_f c_t},$$

which is assumed to be approximately one. (In many applications the value 0.9 is used.) For further details, see [Pas04b] and [Pas04a].
The turbulent eddy viscosity μ_t is given by the turbulent kinetic energy k and the energy dissipation rate ε (cf. Eq. (2.3)):

$$\mu_t = \nu_t \rho_f = c_\mu \rho_f \frac{k^2}{\varepsilon}.$$

Together, the averaged population balance equation then reads as

$$\frac{\partial \bar{f}}{\partial t} = -\nabla \cdot (\bar{u}\bar{f}) + \nabla \cdot (c_t \nabla \bar{f}) + \bar{s}. \tag{2.15}$$

Again, we can apply the former initial and boundary conditions for f to the averaged number density function $\bar{f}$.

2.3.1 Some Models Describing Coalescence and Breakage Processes

In this subsection, we will describe some possibilities how coalescence and breakage can be modeled. In the following, we will present the idea and the approach of the physically based model by Coulaloglou and Tavlarides [CT77]. Furthermore, we will describe the differences to the later introduced models by Tsouris and Tavlarides [TT94] and by Ritter [Rit02].
It was decided to consider the models from [CT77] and [TT94], respectively, because both models are among the most popular (and also most commonly used) models concerning this topic. Furthermore, the model from [Rit02] will be considered, since this work is more recent, but, nevertheless, it directly refers to the other two models.

Remark 2.7. *In the literature, one can find, besides the three models introduced above, many more different approaches for modeling the coalescence and breakage phenomena, see, e. g., [PB90], [BS91], [Che91], [HN96], [LS96], [AKK99], [MML99a], [MML99b], [Ram00], [AKKM02], [DO02a], [DO02b], [LEMM02], [VDV02], [WWJ03], or [WWJ05]. Concerning the breakage of drops in turbulent dispersions, a review of different model approaches can be found in [LEMM02]. Furthermore, in [WWJ03], several models for the distribution of daughter drops have been reviewed. Concerning coalescence, different models for the sub-processes of collision, film drainage, and rupture have been reviewed by [Che91]. In addition, in [WWJ05], several approaches for coalescence and breakage models are compared.*

In the three models that will be presented in the following (as in most of the models in the literature), only one internal coordinate is used to describe the properties of a particle, namely its volume V (or its diameter d, respectively).
Furthermore, as already mentioned before, it is assumed that the coalescence and breakage rates are space-independent. Additionally, the influence of the continuous phase vector is not taken into account.
However, as a first approach, these models can also be used to describe the processes occurring in liquid-liquid systems that do not fulfill all the assumptions mentioned above.

Coalescence

Coalescence is the confluence of fluid particles with each other or of a fluid particle with its mother phase. This means that smaller drops are formed together to larger ones.

The process of coalescence of two drops can be divided into three phases [Che91]:

- Collision of the drops,
- drainage of the film between the drops, and
- rupture of the film (after a critical thickness is reached) and confluence of the drops.

So, a necessary condition for the coalescence of two drops is that the two drops must remain in contact for a sufficiently long time so that the processes of film drainage, film rupture, and coalescence can occur [CT77].

Coalescence rate

In [CT77] the coalescence rate R_{coal}^{CT} of two drops of volumes V and V' is modeled as the product of the collision frequency ϑ^{CT} and the coalescence efficiency $\lambda_{\text{coal}}^{CT}$:

$$R_{\text{coal}}^{CT}(V, V') = \vartheta^{CT}(V, V')\lambda_{\text{coal}}^{CT}(V, V').$$

There, an expression for the collision frequency ϑ^{CT} is derived by assuming that the mechanism of collision in a locally isotropic flow field is analogous to collisions between molecules in the kinetic theory of gases. Thus, the following formula is used:

$$\vartheta^{CT}(V, V') = \left(\frac{9\pi}{2}\right)^{1/3}(V^{2/3} + V'^{2/3})(\overline{u^2}(V) + \overline{u^2}(V'))^{1/2}, \tag{2.16}$$

where $\overline{u^2}(V)$ is the mean square fluctuation velocity of a drop of volume V, given by

$$\overline{u^2}(V) = \bar{c}_1\, \varepsilon_l^{2/3} V^{2/9}, \tag{2.17}$$

where ε_l is the local energy dissipation rate per unit mass and $\bar{c}_1$ is a dimensionless constant. In [CT77] it is stated that under the assumption that ε_l is constant throughout the tank the following relation can be used:

$$\varepsilon_l \sim N^{*3} D_{\text{stirrer}}^2, \tag{2.18}$$

where N^* is the stirrer speed (revolutions per time unit) and D_{stirrer} is the stirrer diameter. Note that in [CT77] it is assumed that the stirrer speed is constant. However, for certain applications, it makes sense to drop this assumption, especially if the stirrer speed is used to control the drop size distribution.

Inserting Eq. (2.17) into Eq. (2.16) yields

$$\vartheta^{CT}(V, V') = \bar{c}_2\, \varepsilon_l^{1/3}(V^{2/3} + V'^{2/3})(V^{2/9} + V'^{2/9})^{1/2}$$

with a constant $\bar{c}_2$.

According to [CT77], the coalescence efficiency $\lambda_{\text{coal}}^{CT}$ can be related to the physical phenomena that occur: The drops must be compressed for a sufficiently long time so that the film between them can rupture and the drops are able to coalesce. This means that the drops have to be in contact for a longer time than coalescence needs to take place. In [CT77] the following approach is used:

$$\lambda_{\text{coal}}^{CT}(V, V') = e^{-\bar{t}_{\text{coal}}^{CT}/\bar{t}_{\text{contact}}^{CT}}, \tag{2.19}$$

where $\bar{t}^{CT}_{\text{coal}}$ and $\bar{t}^{CT}_{\text{contact}}$ are the averages of coalescence and contact time, respectively.
In [CT77] the coalescence time is estimated as the time required for the drainage of the film between the drops:

$$\bar{t}^{CT}_{\text{coal}} \sim \frac{3}{16}\frac{\mu_c \mathcal{F}}{\pi \sigma^2}\left(\frac{1}{h_1^2}-\frac{1}{h_0^2}\right)\left(\frac{dd'}{d+d'}\right)^2, \tag{2.20}$$

where μ_c is the dynamic viscosity of the continuous phase and σ is the interfacial tension. Furthermore, $\mathcal{F}$ is the force compressing the drops, given by

$$\mathcal{F} \sim \rho_c \overline{u^2}\left(\frac{dd'}{d+d'}\right)^2, \tag{2.21}$$

where ρ_c denotes the density of the continuous phase and $\overline{u^2} \sim \varepsilon_l^{2/3}(d+d')^{2/3}$ due to Eq. (2.17). (The model assumes that the drops are located in a fluid eddy of size $d+d'$.) The two parameters h_0 and h_1 are the film thicknesses at initial contact of the drops and at the time, when spontaneous film rupture can occur. Inserting the formulas for $\overline{u^2}$ and $\mathcal{F}$ into Eq. (2.20) leads to

$$\bar{t}^{CT}_{\text{coal}} \sim \frac{\mu_c \rho_c \varepsilon_l^{2/3}(d+d')^{2/3}}{\sigma^2}\left(\frac{1}{h_1^2}-\frac{1}{h_0^2}\right)\left(\frac{dd'}{d+d'}\right)^4.$$

In [CT77] the contact time $\bar{t}^{CT}_{\text{contact}}$ is estimated as the time two drops of size d and d' will stay together in a turbulent flow:

$$\bar{t}^{CT}_{\text{contact}} \sim \frac{(d+d')^{2/3}}{\varepsilon_l^{1/3}}.$$

Assuming the film thicknesses h_0 and h_1 to be constant, the coalescence efficiency can be written as

$$\lambda^{CT}_{\text{coal}}(V,V') = \exp\left(-\bar{c}_3\frac{\mu_c\rho_c\varepsilon_l}{\sigma^2}\left(\frac{V^{1/3}V'^{1/3}}{V^{1/3}+V'^{1/3}}\right)^4\right). \tag{2.22}$$

In this way, the coalescence rate R^{CT}_{coal} in [CT77] is given by

$$\begin{aligned} R^{CT}_{\text{coal}}(V,V') &= c^{CT}_{\text{coal},1}\left(V^{2/3}+V'^{2/3}\right)\left(V^{2/9}+V'^{2/9}\right)^{1/2}\varepsilon_l^{1/3} \\ &\quad \exp\left(-c^{CT}_{\text{coal},2}\frac{\mu_c\rho_c\varepsilon_l}{\sigma^2}\left(\frac{V^{1/3}V'^{1/3}}{V^{1/3}+V'^{1/3}}\right)^4\right), \end{aligned}$$

where $c^{CT}_{\text{coal},1}$ and $c^{CT}_{\text{coal},2}$ are parameters that need to be identified from experimental data.

After comparing numerical and experimental results, a damping of turbulence depending on the volume fraction χ of the dispersed phase was found in [CT77]. Thus, to take this into account, in [CT77] a different formula for the mean square fluctuation velocity $\overline{u^2}$ in a liquid-liquid system is used in the derivation of the coalescence rate, namely

$$\overline{u_\chi^2} = (1+\chi)^{-2}\overline{u^2}. \tag{2.23}$$

Using this formula, in [CT77] the following coalescence rate is derived:

$$\begin{aligned} R_{\text{coal}}^{CT,\chi}(V,V') &= c_{\text{coal},1}^{CT}\frac{1}{1+\chi}\left(V^{2/3}+V'^{2/3}\right)\left(V^{2/9}+V'^{2/9}\right)^{1/2}\ \varepsilon_l^{1/3} \\ &\quad \exp\left(-c_{\text{coal},2}^{CT}\frac{\mu_c\rho_c\varepsilon_l}{\sigma^2(1+\chi)^3}\left(\frac{V^{1/3}V'^{1/3}}{V^{1/3}+V'^{1/3}}\right)^4\right). \end{aligned} \tag{2.24}$$

However, if we insert (2.23) into the Formulas (2.16) and (2.21), we derive a slightly different final form for the coalescence rate. Instead of the term $(1+\chi)^3$ in the denominator of the exponential function, we have the term $(1+\chi)^2$, but besides this, there are no further differences. Note that this difference does not have to be taken into account, but can simply be put into the adaption parameter $c_{\text{coal},2}^{CT}$, since χ is constant in the considered application.

Remark 2.8. *The collision frequency from [CT77] given in Eq. (2.16) is often referred as*

$$\vartheta^{CT,cor}(V,V') = \left(\frac{9\pi}{2}\right)^{1/3}(V^{1/3}+V'^{1/3})^2(\overline{u^2}(V)+\overline{u^2}(V'))^{1/2}, \tag{2.25}$$

even in [TT94]. Therefore, this "corrected" version will be used for the simulation, in particular with regard to the fact that this expression is easier to substantiate physically. Note that the corresponding term then also has to be substituted in the coalescence rate (2.24).

In [TT94] basically the same approach as in [CT77] is used to model the coalescence rate, i. e., the coalescence rate is also expressed by the product of collision frequency and coalescence efficiency. Furthermore, the collision frequency is also modeled by assuming that the drops in a turbulent flow behave like gas molecules. However, in [TT94], the following (corrected) formula is used (cf. Eq. (2.25)):

$$\vartheta^{TT}(d,d') = \frac{\pi}{4}\left(d+d'\right)^2\left(\overline{u^2}(d)+\overline{u^2}(d')\right)^{1/2} \quad \text{with } \overline{u^2}(d) = 1.07\varepsilon_l^{2/3}d^{2/3}, \tag{2.26}$$

where the constant ε_l is given by Eq. (2.18) as proposed in [CT77]. As approach for the coalescence efficiency, also Eq. (2.19) is used.

The main difference between the two models lies in the formulas for coalescence and contact time. In [TT94], the averaged coalescence time $\bar{t}_{\text{coal}}^{TT}$ between two drops with diameters d and d' is given by

$$\bar{t}_{\text{coal}}^{TT} = \bar{c}_4\frac{6\pi\mu_c\bar{c}_5}{\rho_c\varepsilon_l^{2/3}(d+d')^{2/3}},$$

where μ_c and ρ_c are the viscosity and density of the continuous phase, respectively. In [TT94] the value 3.44 is used for the constant $\bar{c}_4$. The parameter $\bar{c}_5$ is given by

$$\bar{c}_5 = 1.872 \ln \left[\frac{h_0^{1/2} + 1.378\bar{c}_6}{h_1^{1/2} + 1.378\bar{c}_6} \right] + 0.127 \ln \left[\frac{h_0^{1/2} + 0.312\bar{c}_6}{h_1^{1/2} + 0.312\bar{c}_6} \right],$$

where h_0 and h_1 are the film thicknesses at initial contact of the drops and at the time, when spontaneous film rupture can occur, respectively. The parameter $\bar{c}_6$ is given by

$$\bar{c}_6 = \frac{\mu_c}{\mu_d} \left(\frac{dd'}{2(d+d')} \right)^{1/2},$$

where μ_d is the viscosity of the dispersed phase. For the contact time $\bar{t}^{TT}_{\text{contact}}$ of two drops with diameters d and d', the following formula is used:

$$\bar{t}^{TT}_{\text{contact}} = \frac{(D^2_{\text{tank}} H_{\text{tank}})^{1/3}}{31.25 N^* D_{\text{stirrer}}},$$

where N^* is the stirrer speed, D_{tank} is the vessel diameter, H_{tank} is the tank height, and D_{stirrer} is the stirrer diameter (see Fig. 2.1).

Altogether, the coalescence rate in [TT94] is given by

$$\begin{aligned} R^{TT}_{\text{coal}}(d,d') &= c^{TT}_{\text{coal},1} \varepsilon_l^{1/3} \left(d+d'\right)^2 \left(d^{2/3} + d'^{2/3}\right)^{1/2} \\ &\quad \exp\left(-c^{TT}_{\text{coal},2} \frac{\mu_c \bar{c}_5}{\rho_c \varepsilon_l^{2/3} (d+d')^{2/3}} \frac{N^* D_{\text{stirrer}}}{(D^2_{\text{tank}} H_{\text{tank}})^{1/3}} \right), \end{aligned}$$

where the parameters $c^{TT}_{\text{coal},1}$ and $c^{TT}_{\text{coal},2}$ can be used as adaption parameters. In [TT94] they are given by

$$c^{TT}_{\text{coal},1} = \frac{\pi}{4}\sqrt{1.07} \approx 0.812422 \quad \text{and} \quad c^{TT}_{\text{coal},2} = c_4 \cdot 6\pi \cdot 31.25 \approx 2026.33.$$

In [Rit02] a kind of mixture of the two models introduced above is used. For the collision frequency Eq. (2.26) from [TT94] is used, whereas the coalescence efficiency is modeled by Eq. (2.22) as given in [CT77]. Furthermore, a damping factor DF, depending on the volume fraction χ, given by

$$\text{DF}(\chi) = \left[1 + 2.5\, \chi\, \frac{\mu_d + 0.4\mu_c}{\mu_d + \mu_c} \right]^2, \tag{2.27}$$

is included in both functions, the collision frequency as well as the coalescence efficiency. Here, as before, μ_d and μ_c denote the dynamic viscosity of the dispersed and the continuous phase, respectively. Note that this damping factor was already used in [TT94] in order to take into account the turbulence damping of the dispersed phase when modeling the breakage process (see below).

Altogether, the coalescence rate in [Rit02] is given by

$$R^R_{\text{coal}}(d,d') = c^R_{\text{coal},1}\frac{\varepsilon_l^{1/3}}{\text{DF}(\chi)^{1/2}}\left(d+d'\right)^2\left(d^{2/3}+d'^{2/3}\right)^{1/2} \exp\left(-c^R_{\text{coal},2}\frac{\mu_c\rho_c\varepsilon_l}{\sigma^2\text{DF}(\chi)^{3/2}}\left(\frac{dd'}{d+d'}\right)^4\right),$$

where $c^R_{\text{coal},1}$ and $c^R_{\text{coal},2}$ can be used as adaption parameters.

Breakage

Breakage means the breakup of larger drops into smaller ones. This can be originated by the collision of drops with each other or with the stirrer or by shear stresses caused by the streaming.
Deformation and breakage of a drop in a turbulent flow depends on many parameters, e. g., drop size, density, interfacial surface tension, viscosity of both phases, holdup fraction, local flow, and local energy dissipation [CT77].

In the models that will be presented in the following, binary breakage is assumed (as in most of the models in the literature). However, in [MGW$^+$06] (see also [MPK06]), it was shown that breakage into more than two daughter drops is usually much more probable. There, experimental investigations were made in order to determine the number and sizes of the daughter drops formed by the breakage of a single mother drop, and the results were used to obtain an enhanced daughter drop distribution. More theoretical modeling approaches for multiple particle breakage, however, can only rarely be found in the literature, see, e. g., [HN96] and [DO02b].

Breakage rate

Again, we first consider the approach introduced in [CT77], which is based on the nature of breakage. In this approach, it is assumed that the droplets are in a turbulent flow field that is locally isotropic. (Experimental investigations show that the flow field can be considered as locally isotropic for Re $\geq$ 10,000, where Re denotes the Reynolds number, see [CT77].) The droplet size d is within the inertial subrange, i. e., it is much larger than the Kolmogorov scale and much smaller than the macro scale, which is given by the diameter of the stirrer in the considered application. Furthermore, it is assumed that viscous effects are negligible. The drops are deformed due to local pressure fluctuations. A drop breaks up if the energy of the turbulent eddy colliding with the drop exceeds the surface energy of the drop.

In [CT77] the following approach is used for the breakage rate of a droplet of volume V:

$$R^{CT}_{\text{break}}(V) = \frac{1}{t_b}\frac{N_{\text{break}}(V)}{N_{\text{drops}}(V)},$$

where t_b is the breakage time and $\frac{N_{\text{break}}(V)}{N_{\text{drops}}(V)}$ is the fraction of drops breaking. This fraction is assumed to be proportional to the fraction of turbulent eddies colliding with the drop and having a turbulent kinetic energy greater than the surface energy of the drop. This is modeled by

$$\frac{N_{\text{break}}(d)}{N_{\text{drops}}(d)} = e^{-E_\sigma/\bar{E}_{\text{kin}}},$$

where $E_\sigma(d)$ is the surface energy of a drop with diameter d and $\bar{E}_{\text{kin}}(d)$ is the mean turbulent kinetic energy of an eddy of size d. For the surface energy E_σ the formula

$$E_\sigma(d) = \tilde{c}_1\, \sigma d^2$$

is used, whereas the mean turbulent kinetic energy $\bar{E}_{\text{kin}}$ is given by

$$\bar{E}_{\text{kin}}(d) = \tilde{c}_2\, \rho_d d^3 \overline{u^2}(d), \tag{2.28}$$

where σ is the interfacial tension and ρ_d is the density of the dispersed phase. The mean square of the relative velocity $\overline{u^2}(d)$ between two points separated by a distance d in the inertial subrange is again given by

$$\overline{u^2}(d) = \tilde{c}_3\, \varepsilon_l^{2/3} d^{2/3},$$

where for ε_l, as before, Eq. (2.18) can be used.

The breakage time t_b is estimated by assuming that the motion of the centers of mass of the daughter drops is similar to the relative motion of two lumps of fluid in a turbulent flow field:

$$t_b \sim d^{2/3} \varepsilon_l^{-1/3}.$$

Altogether the breakage rate in [CT77] is given by

$$R^{CT}_{\text{break}}(V) = c^{CT}_{\text{break},1} V^{-2/9} \varepsilon_l^{1/3} \exp\left(-\frac{c^{CT}_{\text{break},2}\sigma}{\rho_d \varepsilon_l^{2/3} V^{5/9}}\right)$$

with two dimensionless constants $c^{CT}_{\text{break},1}$ and $c^{CT}_{\text{break},2}$ that can be used as adaption parameters.

Taking again the turbulence damping due to the volume fraction χ of the dispersed phase into account, the breakage rate in [CT77] becomes

$$R^{CT,\chi}_{\text{break}}(V) = c^{CT}_{\text{break},1} \frac{\varepsilon_l^{1/3}}{1+\chi} V^{-2/9} \exp\left(-\frac{c^{CT}_{\text{break},2}\sigma(1+\chi)^2}{\rho_d \varepsilon_l^{2/3} V^{5/9}}\right). \tag{2.29}$$

Again, we get a slightly different formula if we substitute $\overline{u^2}$ by $\overline{u_\chi^2}$ in Eq. (2.28), namely the term $1+\chi$ in the denominator of $R^{CT,\chi}_{\text{break}}$ does not occur. However, this difference can again be put into the adaption parameter $c^{CT}_{\text{break},1}$.

Let us now have a closer look at the model from [TT94]. There, the approach from [PB90] is used to model the breakage rate, i. e., the breakage rate R^{TT}_{break} of a drop with diameter d is assumed to be given by the product of the eddy-drop-collision frequency ϑ^{TT} and the breakage efficiency $\lambda^{TT}_{\text{break}}$:

$$R^{TT}_{\text{break}}(d) = \vartheta^{TT}(d)\lambda^{TT}_{\text{break}}(d).$$

This model is based on the assumption that the turbulent flow in a stirred tank is isotropic and that the drop size is in the inertial subrange (cf. the model from [CT77]). Furthermore, it is assumed that a drop only breaks up if it collides with an eddy that is smaller than the drop. (Larger eddies only transport the drops.)

In this model, the eddy-drop-collision frequency is modeled (like the drop-drop collision frequency in the model from [CT77]) by assuming that the eddies and drops move like ideal gas molecules. Therefore, their collision process can be described with the help of the kinetic theory of gases.
Accordingly, the following formula for the collision frequency between drops of size d and eddies of a size range that can break these drops is used in [TT94]:

$$\vartheta^{TT}(d) = \int \frac{\pi}{4}(d_e + d)^2(u_e^2 + u_d^2)^{1/2} dn_e, \tag{2.30}$$

where d_e is the size of an eddy that can break a drop with diameter d and dn_e is the number of eddies of size between d_e and $d_e + \Delta d_e$. Eddy velocity u_e and drop velocity u_d are given by

$$u_e^2 = 8.2\left(\frac{\varepsilon_l}{\kappa_e}\right)^{2/3} \quad \text{and} \quad u_d^2 = 1.07\varepsilon_l^{2/3} d^{2/3},$$

where ε_l is chosen as before and $\kappa_e = \frac{2}{d_e}$ is the wave number of an eddy with diameter d_e.

Inserting these two formulas into Eq. (2.30) (and using $d_e = \frac{2}{\kappa_e}$) gives

$$\vartheta^{TT}(d) = \int \frac{\pi}{4}\left(\frac{2}{\kappa_e} + d\right)^2 \varepsilon_l^{1/3}\left[8.2\kappa_e^{-2/3} + 1.07d^{2/3}\right]^{1/2} dn_e.$$

In [TT94] the integration variable n_e is substituted by the eddy wave number κ_e by using the following differential equation for the number of eddies per unit mass of the fluid $n_{e,m}$:

$$\frac{dn_{e,m}(\kappa_e)}{d\kappa_e} = 0.1\frac{\kappa_e^2}{\rho_f},$$

where ρ_f is the density of the fluid. Hence, the collision frequency in [TT94] is given by

$$\vartheta^{TT}(d) = \tilde{c}_4\varepsilon_l^{1/3}\int_{2/d}^{2/d_{e,\min}}\left(\frac{2}{\kappa_e} + d\right)^2\left[8.2\kappa_e^{-2/3} + 1.07d^{2/3}\right]^{1/2}\kappa_e^2 d\kappa_e.$$

Here, the integral term represents the total number of eddies that are able to break drops with diameter d. The upper limit of the integration is the (large) wave number

that corresponds to the eddy size $d_{e,\min}$. This eddy size $d_{e,\min}$ can be taken arbitrarily as half of the critical drop size, since eddies of size less than the critical drop diameter are not able to break any drop in the system because they do not have enough kinetic energy. In [TT94] the constant $\tilde{c}_4$ includes the turbulence damping factor $\mathrm{DF}(\chi)$ due to the breakage as well as the fraction of the volume of the stirrer region V_{stirrer} to the total volume of the tank V_{tank}:

$$\tilde{c}_4 = \frac{V_{\text{stirrer}}}{V_{\text{tank}}} \frac{0.1\pi}{4} \mathrm{DF}(\chi),$$

where the damping factor $\mathrm{DF}(\chi)$ is given by Eq. (2.27).

In [TT94] the breakage efficiency is modeled by an exponential function (cf. [CT77] or [PB90]), which accounts for the different forces acting on the drop. (A drop breaks if its surface tension is exceeded by the energy of a colliding eddy.) Therefore, the breakage efficiency can be described by

$$\lambda_{\text{break}}^{TT} = \exp\left(-\frac{\bar{E}_\sigma}{\tilde{c}_5 \bar{E}_{\text{eddy}}}\right),$$

where $\bar{E}_\sigma$ is the average energy required for drop breakage, $\bar{E}_{\text{eddy}}$ is the average energy of an eddy, and $\tilde{c}_5$ is a constant. The average energy required for drop breakage is assumed to be the arithmetic mean of the minimal and maximal energy required for forming two daughter drops. The minimal energy is given by the energy needed to form the smallest and largest possible daughter drop, the maximal energy by the one needed to form two equal-size drops. By this, one gets

$$\bar{E}_\sigma = \frac{1}{2}\Bigg\{ \underbrace{2\pi\sigma\left(\frac{d}{2^{1/3}}\right)^2}_{\text{maximal energy } E_{\sigma,\max}} + \underbrace{\pi\sigma d_{\max}^2 + \pi\sigma d_{\min}^2}_{\text{minimal energy } E_{\sigma,\min}} \Bigg\} - \underbrace{\pi\sigma d^2}_{\text{energy of breaking drop}},$$

where σ is the interfacial tension and $d_{\max}$ and $d_{\min}$ are the diameters of the largest and smallest possible drops that can be formed by breakage of a drop with diameter d. (Note that, due to conservation of mass, the formula $d^3 = d_{\max}^3 + d_{\min}^3$ must hold.) The average energy of an eddy is given by

$$\bar{E}_{\text{eddy}} = 0.43\pi\rho_f \varepsilon_l^{2/3} d_e^{11/3} \overset{d_e = \frac{2}{\kappa_e}}{=} 5.47\pi\rho_f \varepsilon_l^{2/3} \kappa_e^{-11/3}.$$

Altogether the breakage rate in [TT94] is given by

$$R_{\text{break}}^{TT}(d) = c_{\text{break},1}^{TT} \mathrm{DF}(\chi) \varepsilon_l^{1/3} \int_{2/d}^{2/d_{e,\min}} \left(\frac{2}{\kappa_e} + d\right)^2 \left[8.2\kappa_e^{-2/3} + 1.07 d^{2/3}\right]^{1/2} \exp\left(-\frac{\bar{E}_\sigma}{c_{\text{break},2}^{TT} \bar{E}_{\text{eddy}}}\right) \kappa_e^2 d\kappa_e, \quad (2.31)$$

where $\mathrm{DF}(\chi)$ is the (above mentioned) damping factor dependent on the volume fraction χ of the dispersed phase. In [TT94] the following values are used for the constants $c^{TT}_{\mathrm{break},1}$ and $c^{TT}_{\mathrm{break},2}$:

$$c^{TT}_{\mathrm{break},1} = 0.0118 \quad \text{and} \quad c^{TT}_{\mathrm{break},2} = 1.3.$$

The breakage rate introduced in [Rit02] is quite similar to the one used in [TT94]. The only differences are that in [Rit02] the damping factor $\mathrm{DF}(\chi)$ does not occur in the numerator, but the square root of it is put into the denominator, and that different values for the adaption parameters are used. Thus, the breakage rate in [Rit02] is given by

$$R^R_{\mathrm{break}}(d) = \frac{c^R_{\mathrm{break},1}\varepsilon_l^{1/3}}{(\mathrm{DF}(\chi))^{1/2}} \int_{2/d}^{2/d_{e,\min}} \left(\frac{2}{\kappa_e} + d\right)^2 \left[8.2\kappa_e^{-2/3} + 1.07 d^{2/3}\right]^{1/2} \exp\left(-\frac{\bar{E}_\sigma}{c^R_{\mathrm{break},2}\bar{E}_{\mathrm{eddy}}}\right) \kappa_e^2 d\kappa_e,$$

and the constants $c^R_{\mathrm{break},1}$ and $c^R_{\mathrm{break},2}$ are set to

$$c^R_{\mathrm{break},1} = 0.1\frac{\pi}{4} \approx 0.0785 \quad \text{and} \quad c^R_{\mathrm{break},2} = 1.$$

Distribution of daughter drops

In order to be able to model the source term due to breakage, a distribution of daughter drops is needed. The distribution of daughter drops is a probability density function, which gives the probability that the breakage of a drop of diameter d' (or volume V') leads to a daughter drop of diameter d (or volume V). The distribution of daughter drops (or breakage distribution function) can be modeled in different ways. Many different approaches can be found in the literature (see again, e. g., [CT77], [TT94] and [Rit02] or the references given in Remark 2.7 on page 17).

In [CT77] it is assumed that the distribution of daughter drops takes the form of a normal distribution:

$$\gamma^{CT}(V, V') = \frac{2.4}{V'} \exp\left(-4.5\frac{(2V - V')^2}{V'^2}\right).$$

Here, the variance is chosen such that more than 99.6 percent of the droplets formed lie within the volume range between 0 and V'. Note that this function can be rewritten in the following form:

$$\gamma^{CT}(V, V') = \frac{1}{\frac{V'}{6}\sqrt{2\pi}} \exp\left(\frac{(V - \frac{1}{2}V')^2}{2\left(\frac{V'}{6}\right)^2}\right). \tag{2.32}$$

In [Rit02], a normal distribution is used, given by the following formula:

$$\gamma^R(V,V') = \frac{1}{\frac{V'}{10}\sqrt{2\pi}} \exp\left(-\frac{\left(V-\frac{V'}{2}\right)^2}{2\left(\frac{V'}{10}\right)^2}\right). \tag{2.33}$$

Comparison with Eq. (2.32) shows that the normal distribution used in [Rit02] has a smaller variance than the one used in [CT77].

It can be shown that the energy requirement for a breakage into two equal-size drops is greater than the energy requirement for a breakage into a small and a large drop [TT94]. The use of a normal distribution, which is unimodal and has a higher probability density in the middle, does not take this into account. Therefore, it is better to use a bimodal distribution with higher probability at the ends than in the middle.

In [TT94] the following function is used as distribution of daughter drops:

$$\gamma^{TT}(d,d') = \frac{E_{\text{break, min}}(d) + E_{\text{break, max}}(d) - E_{\text{break}}(d)}{\int_0^{d'} \left[E_{\text{break, min}}(\delta) + E_{\text{break, max}}(\delta) - E_{\text{break}}(\delta)\right] d\delta},$$

where $E_{\text{break, min}}(d)$ and $E_{\text{break, max}}(d)$ are the minimal and maximal energy needed for the breakage of a drop with diameter d into two daughter drops, and $E_{\text{break}}(d)$ is the energy of a dispersing drop of diameter d.

In [TT94] concrete formulas for these energies are only given for the discrete case. However, alternatively one can also use the above introduced formulas to model these energy terms:

$$\begin{aligned}
E_{\text{break, min}}(d) &:= E_{\sigma,\min}(d) = \pi\sigma d_{\max}^2 + \pi\sigma d_{\min}^2 \quad \text{with } d_{\max}^3 + d_{\min}^3 = d^3,\\
E_{\text{break, max}}(d) &:= E_{\sigma,\max}(d) = 2\pi\sigma\left(\frac{d}{2^{1/3}}\right)^2,\\
E_{\text{break}}(d) &:= \pi\sigma d^2.
\end{aligned}$$

In [Ger99] another bimodal distribution can be found:

$$\gamma_*^G(V,V') = V'(V'-V)\exp\left(-\hat{c}_1^4\left(\left(\frac{V'}{V}-\frac{1}{2}\right)^4 - \left(\frac{1-2\hat{c}_2}{\sqrt{2}}\left(\frac{V'}{V}-\frac{1}{2}\right)\right)^2\right)\right)$$

with constants $\hat{c}_1 = 0.065$ and $\hat{c}_2 = 3.5$. This function has to be scaled such that it fulfills the normalization condition (2.13). This leads to:

$$\gamma^G(V,V') = \frac{\gamma_*^G(V,V')}{\int_0^{V_{\max}} \gamma_*^G(V,V')dV'}.$$

In general, for every distribution of daughter drops, the dependency on the internal coordinate can be changed from volume V to diameter d (and vice versa) by the following transformation:

$$\gamma(d,d') = 3c_V d^2 \gamma(V,V'),$$

where c_V is the shape factor between volume and diameter.

2.4 Summary

At the end of this chapter, let us summarize all equations so that we get one coupled system describing the processes in the stirred tank. This system consists of the Reynolds-averaged Navier-Stokes equations with k-ε turbulence model given by Eq. (2.5) and the averaged population balance equation given by (2.15). Note that the source and sink terms s have to be modeled such that Eqs. (2.13) and (2.14) are satisfied. In addition, we have the following initial and boundary conditions (cf. Eqs. (2.6), (2.7), and (2.8) as well as (2.10) and (2.11), but for the averaged number density function $\bar{f}$). Altogether, the following system is derived:

$$\left.\begin{aligned}
\tfrac{\partial}{\partial t}\bar{u} + \nabla\cdot(\bar{u}\otimes\bar{u}) + \tfrac{1}{\rho_f}\nabla\bar{p} + \tfrac{2}{3}\nabla k - \nabla\cdot(\nu^*(\nabla\bar{u} + (\nabla\bar{u})^T)) &= \bar{g}\\
\nabla\cdot\bar{u} &= 0\\
\tfrac{\partial}{\partial t}k + \bar{u}\cdot\nabla k - \tfrac{\nu_t}{2}\|\nabla\bar{u} + (\nabla\bar{u})^T\|_F^2 - \nabla\cdot(\nu_t\nabla k) + \varepsilon &= 0\\
\tfrac{\partial}{\partial t}\varepsilon + \bar{u}\cdot\nabla\varepsilon - \tfrac{c_1 c_\mu}{2}k\|\nabla\bar{u} + (\nabla\bar{u})^T\|_F^2 - \nabla\cdot\left(\tfrac{1}{c_\varepsilon}\nu_t\nabla\varepsilon\right) + c_2\tfrac{\varepsilon^2}{k} &= 0
\end{aligned}\right\} \quad \text{in } \Omega\times[0,t_{\text{end}}],$$

$$\frac{\partial\bar{f}}{\partial t} + \nabla\cdot(\bar{u}\bar{f}) - \nabla\cdot(c_t\nabla\bar{f}) = \bar{s} \quad \text{in } [V_{\min},V_{\max}]\times\Omega\times[0,t_{\text{end}}],$$

$$\begin{aligned}
u(x,t=0) &= u_0(x) && \text{in } \Omega,\\
p(x,t=0) &= p_0(x) && \text{in } \Omega,\\
k(x,t=0) &= k_0(x) && \text{in } \Omega,\\
\varepsilon(x,t=0) &= \varepsilon_0(x) && \text{in } \Omega,\\
f(V,x,t=0) &= f_0(V,x) && \text{in } [V_{\min},V_{\max}]\times\Omega,\\
\bar{u} &= u_{\text{stirrer}} && \text{on } \Gamma_{\text{stirrer}}\times[0,t_{\text{end}}],\\
\bar{u} &= 0 && \text{on } \Gamma_{\text{wall}}\times[0,t_{\text{end}}],\\
\tfrac{\partial\bar{u}_1}{\partial\mathfrak{n}_1} = \tfrac{\partial\bar{u}_2}{\partial\mathfrak{n}_2} = 0,\ \bar{u}_3 &= 0 && \text{on } \Gamma_{\text{surface}}\times[0,t_{\text{end}}],\\
k &= 0 && \text{on } \Gamma_{\text{stirrer}}\cup\Gamma_{\text{wall}}\times[0,t_{\text{end}}],\\
\tfrac{\partial k}{\partial\mathfrak{n}} &= 0 && \text{on } \Gamma_{\text{surface}}\times[0,t_{\text{end}}],\\
\tfrac{\partial\varepsilon}{\partial\mathfrak{n}} &= 0 && \text{on } \Gamma\times[0,t_{\text{end}}],\\
f(V,x,t) &= 0 && \text{on } [V_{\min},V_{\max}]\times\Gamma\times[0,t_{\text{end}}],\\
f(V_{\min},x,t) = f(V_{\max},x,t) &= 0 && \text{in } \Omega\times[0,t_{\text{end}}].
\end{aligned}$$

Remark 2.9. *As already mentioned before, in the derivation of this system of equations a one-way coupling was assumed.*
However, for further applications, like for systems, where the dispersed phase has an influence on the continuous phase that cannot be neglected, a more complex modeling approach has to be considered (at least a two-way coupling, usually in combination with a two- or multi-phase fluid flow model), see Remark 2.4.

Furthermore, as soon as one likes to control the drop size distribution, also a two-way coupling has to be considered. In this case (and under the assumption that the drops are just moving with the fluid and do not have any influence on the continuous phase),

one can still use a single-phase fluid flow model. However, the influence of the input parameters (which are set according to the observed drop size distribution, for example) has to be additionally modeled, which leads to an additional source term in the Reynolds-averaged Navier-Stokes equations.

In many technical applications, like in the production of polystyrene foam, for example, mono-disperse systems with a certain drop size are desired. For such an application, the system of equations above must be extended as described in the following.
The right-hand side of the system of equations describing the fluid flow as well as the behavior of the drops in the turbulent dispersion must be extended by a control term $\mathcal{S}(U_{in})$*, where* U_{in} *summarizes the control or input parameters. The input parameters influencing the averaged population balance equation are, e. g., the pH-value, the volume fraction of the dispersed phase, the stirrer speed, or physical characteristics of the liquids, whereas the flow field is mainly influenced by one parameter, namely the stirrer speed.*
Furthermore, an output equation of the following form must be added to the system:

$$U_{out} = \mathcal{Y}(\bar{u}, \bar{p}, k, \varepsilon, \bar{f}).$$

Here, U_{out} *consists of the output parameters, which can be mean values of the distribution (like the Sauter mean diameter or the moments of the distribution) or parameters describing the width of the distribution (like the standard deviation), for example. The output parameters are usually quantities that can be measured and / or calculated. They are used to compare the calculated drop size distribution with the desired one in order to determine the input for the next optimization step.*

3 Analysis of the Dynamical Behavior of the Coupled System

In this chapter, we will apply a method of lines (MOL) approach to the system of equations describing the considered stirred liquid-liquid system in order to analyze the dynamics of the process. This spatial discretization leads to a differential-algebraic system, for which an index will be determined, since this tells us something about how strong the coupling between the equations is and, therefore, also something about numerical problems that may arise.

Generally, one can say that from the numerical point of view, it is desirable for the differential-algebraic system to have an index that is as small as possible [BCP96], since higher-index systems usually are more difficult to solve. Therefore, it is quite important to determine an index so that the problem can be treated appropriately. If an higher index is detected, index reduction may be one way to treat the problem.

Remark 3.1. *Note that if a stationary system is considered, the framework of differential-algebraic equations is not appropriate, since the typical problems of differential-algebraic equations do not occur in this case. On the other hand, this framework is particularly suitable for analyzing the dynamical behavior of a time-dependent system (like the coupled system in the considered application). Furthermore, the analysis of the underlying differential-algebraic system is important if control or optimization of a process is investigated. Even if the desired state of the system is a stationary one (as for the considered coupled system, where the goal is to derive a stationary drop size distribution with a certain mean drop size and a small standard deviation), the "way to get there" can be highly time-dependent and, thus, the dynamics of the underlying differential-algebraic system plays an important role and has to be analyzed.*
In this case, the method of lines (i. e., a semi-discretization in space), which leads to a differential-algebraic system, is a very good approach for the treatment of the system of partial differential equations. If the system was first discretized in time (using the so-called Rothe-method), it would be transformed to a stationary system, and the analysis described below could not be performed (see also Remark 4.4 in Chapter 4).

3.1 Derivation of the Differential-Algebraic System

As we have seen in Chapter 2, the system of equations describing the processes in the stirred tank is given by:

$$\begin{aligned}
\frac{\partial}{\partial t}\bar{u} + \nabla\cdot(\bar{u}\otimes\bar{u}) + \frac{1}{\rho_f}\nabla\bar{p} + \frac{2}{3}\nabla k - \nabla\cdot(\nu^*(\nabla\bar{u} + (\nabla\bar{u})^T)) &= \bar{g},\\
\frac{\partial}{\partial t}k + \bar{u}\cdot\nabla k - \frac{\nu_t}{2}\|\nabla\bar{u} + (\nabla\bar{u})^T\|_F^2 - \nabla\cdot(\nu_t\nabla k) + \varepsilon &= 0,\\
\frac{\partial}{\partial t}\varepsilon + \bar{u}\cdot\nabla\varepsilon - \frac{c_1c_\mu}{2}k\|\nabla\bar{u} + (\nabla\bar{u})^T\|_F^2 - \nabla\cdot\left(\frac{1}{c_\varepsilon}\nu_t\nabla\varepsilon\right) + c_2\frac{\varepsilon^2}{k} &= 0,\\
\frac{\partial\bar{f}}{\partial t} + \nabla\cdot(\bar{u}\bar{f}) - \nabla\cdot(c_t\nabla\bar{f}) &= \bar{s},\\
\nabla\cdot\bar{u} &= 0,
\end{aligned} \tag{3.1}$$

where the constants c_μ, c_ε, c_1, and c_2 are defined as in Section 2.2.

Note that the quantities $\nu^* = \nu + \nu_t$ with $\nu_t = c_\mu\frac{k^2}{\varepsilon}$ and $c_t = \frac{\nu_t}{\mathrm{Sc}_t} = \frac{c_\mu\frac{k^2}{\varepsilon}}{\mathrm{Sc}_t}$ are dependent on k and ε. Furthermore, since the term s is modeled in such a way that it is dependent on the energy dissipation rate ε (see Section 2.3), the averaged source term $\bar{s}$ is also dependent on ε.
Additionally, we have to take care that the condition for mass conservation, given by Eq. (2.14), is fulfilled.

In the following, the system of equations given in (3.1) will be discretized in space in order to analyze the resulting differential-algebraic system that describes the dynamics of the process.
In analogy to the construction in [BCP96], it can be shown that the spatial discretization of the Navier-Stokes equations (2.1) leads to a semi-discretized system of the following form:

$$\begin{aligned}
M\frac{du_h}{dt} + (N_{\mathrm{diff}} + N_{\mathrm{conv}}(u_h))u_h + \frac{1}{\rho_f}Cp_h &= g_h,\\
\tilde{C}^T u_h &= 0,
\end{aligned}$$

where u_h, p_h, and g_h are the semi-discretized velocity, pressure, and external forces, respectively. Furthermore, M is the mass matrix and N_{diff} and N_{conv} are the discretized diffusive and convective parts, corresponding to $-\nu\Delta u$ and $\nabla\cdot(u\otimes u)$, respectively. The matrices C and $\tilde{C}^T$ are the discretized gradient and divergence operators. (Normally the discretization is done in such a way that $\tilde{C} = C$ holds, but in general this equality does not need to be satisfied.)

For the system in (3.1), consisting of the Reynolds-averaged Navier-Stokes equations with k-ε model and the averaged population balance equation, this can be done in the

same way. Here, spatial discretization leads to

$$\begin{aligned} M_1\frac{du_h}{dt} + (N_1(u_h) + N_2(k_h,\varepsilon_h))u_h + \frac{2}{3}C_1k_h + \frac{1}{\rho_f}C_2p_h &= \mathcal{G}_1, \\ M_2\frac{dk_h}{dt} + (N_3(u_h) + N_4(k_h,\varepsilon_h))k_h + N_5(k_h,\varepsilon_h)u_h &= \mathcal{G}_2(\varepsilon_h), \\ M_3\frac{d\varepsilon_h}{dt} + (N_3(u_h) + \tilde{N}_4(k_h,\varepsilon_h))\varepsilon_h + \tilde{N}_5(k_h,\varepsilon_h)u_h &= \mathcal{G}_3(k_h,\varepsilon_h), \\ M_4\frac{df_h}{dt} + (N_6(u_h) + N_7(k_h,\varepsilon_h))f_h &= \mathcal{G}_4(\varepsilon_h,f_h), \\ C_3^Tu_h &= 0, \end{aligned} \tag{3.2}$$

where u_h and p_h are the semi-discretized averaged velocity and pressure, respectively. Furthermore, k_h, ε_h, and f_h are the corresponding quantities derived by the semi-discretization of the turbulent kinetic energy k, the energy dissipation ε, and the averaged number density function $\bar{f}$. The terms $\mathcal{G}_1$, …, $\mathcal{G}_4$ on the right-hand side are the semi-discretized source terms. The matrices M_1, …, M_4 are mass matrices, the matrices N_1, …, N_7, $\tilde{N}_4$, and $\tilde{N}_5$ are discretizations of the nonlinear terms. Finally, C_1 and C_2 are the discretized gradient operators and C_3^T is the discretized divergence operator.

Additionally, the discretization of the equation for mass conservation, given by Eq. (2.14), leads to

$$\mathcal{G}_5(m_h) = 0, \tag{3.3}$$

where m_h is the vector consisting of the discretized particle masses.

For the sake of clarity in notation, we summarize the quantities u_h, k_h, ε_h, and f_h to the variable $q_h := [u_h^T, k_h^T, \varepsilon_h^T, f_h^T]^T$. If we further define

$$M = \begin{bmatrix} M_1 & & \\ & \ddots & \\ & & M_4 \end{bmatrix}, \quad \mathcal{P} = \begin{bmatrix} -\frac{1}{\rho_f}C_2p_h \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

and

$$\mathcal{G} = \begin{bmatrix} -(N_1(u_h) + N_2(k_h,\varepsilon_h))u_h - \frac{2}{3}C_1k_h + \mathcal{G}_1 \\ -(N_3(u_h) + N_4(k_h,\varepsilon_h))k_h - N_5(k_h,\varepsilon_h)u_h + \mathcal{G}_2(\varepsilon_h) \\ -(N_3(u_h) + \tilde{N}_4(k_h,\varepsilon_h))\varepsilon_h - \tilde{N}_5(k_h,\varepsilon_h)u_h + \mathcal{G}_3(k_h,\varepsilon_h) \\ -(N_6(u_h) + N_7(k_h,\varepsilon_h))f_h + -\mathcal{G}_4(\varepsilon_h,f_h) \end{bmatrix},$$

the system in (3.2) together with Eq. (3.3) can be written as

$$\begin{bmatrix} M & & \\ & 0 & \\ & & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_h \\ \dot{m}_h \\ \dot{p}_h \end{bmatrix} = \begin{bmatrix} \mathcal{G} + \mathcal{P} \\ \mathcal{G}_5 \\ C_3^Tu_h \end{bmatrix}, \tag{3.4}$$

where $\dot{q}_h$, $\dot{m}_h$, and $\dot{p}_h$ denote the time derivatives of the corresponding variables. Note that $\mathcal{G}$ is not dependent on the semi-discretized pressure p_h.

Before we will determine an index of the differential-algebraic system derived above in the next section, we will say something about its special form. For this, we need the following definition from [Mär01].

Definition 3.2. *Consider the following nonlinear differential-algebraic equation*

$$\mathcal{V}(z(t),t)(w(z(t),t)) + b(z(t),t) = 0, \quad t \in \mathbb{I} \subset \mathbb{R},\ z(t) \in \mathbb{D}_z \subset \mathbb{R}^{\hat{n}}, \tag{3.5}$$

where the functions $\mathcal{V} : \mathbb{D}_z \times \mathbb{I} \to \mathbb{R}^{\hat{n}\times\hat{n}}$, $w : \mathbb{D}_z \times \mathbb{I} \to \mathbb{R}^{\hat{n}}$, *and* $b : \mathbb{D}_z \times \mathbb{I} \to \mathbb{R}^{\hat{n}}$ *are continuous in their arguments and have the continuous partial derivatives* $\mathcal{V}_z$, w_z, *and* b_z. *Denote* $\mathcal{W}(z,t) := w_z(z,t)$.
The leading term in (3.5) is said to be properly stated (or properly formulated) if

$$\ker\left(\mathcal{V}(z,t)\right) \oplus \operatorname{im}\left(\mathcal{W}(z,t)\right) = \mathbb{R}^{\hat{n}}, \quad z \in \mathbb{D}_z,\ t \in \mathbb{I},$$

where im *and* ker *denote the image and the kernel, respectively, and if there exists a continuously differentiable projector function* $\mathcal{U} : \mathbb{I} \to \mathbb{R}^{\hat{n}\times\hat{n}}$ *such that* $(\mathcal{U}(t))^2 = \mathcal{U}(t)$, $\ker(\mathcal{V}(z,t)) = \ker(\mathcal{U}(t))$, $\operatorname{im}(\mathcal{W}(z,t)) = \operatorname{im}(\mathcal{U}(t))$, *and* $w(z,t) = \mathcal{U}(t)w(z,t)$ *holds for* $z \in \mathbb{D}_z$, $t \in \mathbb{I}$.

Lemma 3.3. *The system (3.4) is in properly stated leading term form.*

Proof. Obvious, since Eq. (3.4) can be rewritten as

$$\begin{bmatrix} I_{n_q} & & \\ & 0 & \\ & & 0 \end{bmatrix} \left(\begin{bmatrix} I_{n_q} & & \\ & 0 & \\ & & 0 \end{bmatrix} \begin{bmatrix} q_h \\ m_h \\ p_h \end{bmatrix}^{\cdot} \right) - \begin{bmatrix} M^{-1}(\mathcal{G}+\mathcal{P}) \\ \mathcal{G}_5 \\ C_3^T u_h \end{bmatrix} = 0,$$

and it holds:

$$\ker\left(\begin{bmatrix} I_{n_q} & & \\ & 0 & \\ & & 0 \end{bmatrix} \right) \oplus \operatorname{im}\left(\begin{bmatrix} I_{n_q} & & \\ & 0 & \\ & & 0 \end{bmatrix} \right) = \mathbb{R}^{\hat{n}},$$

where n_q denotes the size of q_h and $\hat{n}$ the dimension of the whole system. Thus, the projector $\mathcal{U}$ (see Definition 3.2) can be chosen as

$$\mathcal{U} = \begin{bmatrix} I_{n_q} & & \\ & 0 & \\ & & 0 \end{bmatrix}.$$

□

Remark 3.4. *Note that if a differential-algebraic equation is in properly stated leading term form, this has some advantages concerning its analysis and numerical treatment, see, e. g., [Mär02]. First of all, from this special form, one can directly read off, which derivatives of the unknown function are involved and in what way. Thus, stronger solvability results and a better performance of numerical methods can be obtained. Furthermore, the use of the properly stated leading term form for differential-algebraic equations yields advantages in optimal control problems because of the symmetry between the original (linear) differential-algebraic equation and its adjoint equation (both having the same form). Further details on this can be found in [Mär02] and references therein.*

3.2 Index Determination

In this section, we will determine an index of the differential-algebraic system in (3.4).

In the literature, various index concepts can be found. The most widely used are the differentiation index (d-index) introduced in [Gea88], [Gea90], and [CG95] and the perturbation index (p-index) introduced in [HLR89] (see also [HW96]). But there are also other concepts, like the geometric index, see [Rhe84], the tractability index, see [GM86], or the structural index, see [Pan88].

Here, the so-called "strangeness-index" (s-index), introduced in [KM94] and [KM98] will be used for the analysis, since this concept is the most general one of all these index concepts. The strangeness-index can be seen as generalization of the differentiation index (see [KM94], [KM96b], and [KM01]) as well as of the perturbation index (see [KM96a]). Furthermore, also over- and under-determined systems can be treated with this concept (see [KM01]).

Before we will determine the index of the system in (3.4), we will give a brief introduction to the concept of the strangeness-index of general (nonlinear) differential-algebraic equations.

3.2.1 The Strangeness-Index of a General Differential-Algebraic Equation

In this subsection, we give a brief introduction how the strangeness-index of a nonlinear differential-algebraic equation is defined and how it can be determined. This introduction is a brief summary of parts of the theory presented in [KM06].

We consider a general differential-algebraic equation

$$F(t, z, \dot{z}) = 0 \tag{3.6}$$

with $F \in \mathcal{C}(\mathbb{I} \times \mathbb{D}_z \times \mathbb{D}_{\dot{z}}, \mathbb{R}^{\hat{n}})$, $\mathbb{I} \subset \mathbb{R}$ (compact) interval, $\mathbb{D}_z, \mathbb{D}_{\dot{z}} \subset \mathbb{R}^{\hat{n}}$ open.

The standard method for the solution of nonlinear problems is Newton's method. There, we first linearize around some given initial guess, then solve the so obtained linear problem, and correct the initial guess afterwards.
This suggests to consider any linearization of (3.6) and apply the theory for linear systems with variable coefficients to the resulting equations in order to determine the index of the nonlinear system given in (3.6). But counter-examples show that one must expect that away from the solution the constant rank assumptions, which are required for the analysis in the linear case (see [KM98]), do not hold anymore. Thus, it does not make sense to consider linearizations of the original equation alone. We must also include derivatives of the equation. Moreover, the common constant rank assumptions must be restricted to a subset of a neighborhood.

For these reasons, we define the so-called "inflated differential-algebraic equation", which

consists of the original equation and its derivatives up to order $l \in \mathbb{N}_0$, by

$$\begin{aligned} F_l(t,z,\dot{z},\ldots,z^{(l+1)}) &= \begin{bmatrix} F(t,z,\dot{z}) \\ \frac{d}{dt}F(t,z,\dot{z}) \\ \vdots \\ (\frac{d}{dt})^l F(t,z,\dot{z}) \end{bmatrix} \\ &= \begin{bmatrix} F(t,z,\dot{z}) \\ F_t(t,z,\dot{z}) + F_z(t,z,\dot{z})\dot{z} + F_{\dot{z}}(t,z,\dot{z})\ddot{z} \\ \vdots \end{bmatrix} = 0 \end{aligned}$$

and denote its Jacobians by

$$\begin{aligned} \mathcal{M}_l(t,z,\dot{z},\ldots,z^{(l+1)}) &= F_{l;\dot{z},\ldots,z^{(l+1)}}(t,z,\dot{z},\ldots,z^{(l+1)}), \\ \mathcal{N}_l(t,z,\dot{z},\ldots,z^{(l+1)}) &= -\left[F_{l;z}(t,z,\dot{z},\ldots,z^{(l+1)}),0,\ldots,0\right]. \end{aligned}$$

If we define

$$E := F_{\dot{z}}(t,z,\dot{z}) \quad \text{and} \quad A := -F_z(t,z,\dot{z}),$$

the Jacobians $\mathcal{M}_l$ and $\mathcal{N}_l$ can be written as in the linear case, see [KM06]:

$$\begin{aligned} \mathcal{M}_l &= \begin{bmatrix} E & & & & \\ \dot{E}-A & E & & & \\ \ddot{E}-2\dot{A} & 2\dot{E}-A & E & & \\ \vdots & & \ddots & \ddots & \\ E^{(l)}-lA^{(l-1)} & \ldots & & l\dot{E}-A & E \end{bmatrix}, \\ \mathcal{N}_l &= \begin{bmatrix} A & 0 & \ldots & 0 \\ \dot{A} & 0 & \ldots & 0 \\ \ddot{A} & 0 & \ldots & 0 \\ \vdots & \vdots & & \vdots \\ A^{(l)} & 0 & \ldots & 0 \end{bmatrix}. \end{aligned}$$

Before we can define the strangeness-index of the general nonlinear differential algebraic equation, given in Eq. (3.6), we need the following hypothesis.

Hypothesis 3.5. *There exist integers $\hat{\mu}$, $\hat{a}$, and $\hat{d}$ such that the set*

$$\mathcal{L}_{\hat{\mu}} = \{(t,z,\dot{z},\ldots,z^{(\hat{\mu}+1)}) \in \mathbb{I} \times \mathbb{R}^{\hat{n}} \times \mathbb{R}^{\hat{n}} \times \cdots \times \mathbb{R}^{\hat{n}} \mid F_{\hat{\mu}}(t,z,\dot{z},\ldots,z^{(\hat{\mu}+1)}) = 0\}$$

associated with F is nonempty and such that for all values $(t,z,\dot{z},\ldots,z^{(\hat{\mu}+1)}) \in \mathcal{L}_{\hat{\mu}}$, where $F_{\hat{\mu}}(t,z,\dot{z},\ldots,z^{(\hat{\mu}+1)}) = 0$ is considered as an algebraic equation, there exists a (sufficiently small) neighborhood, in which the following properties are satisfied.

1. *We have*
$$\operatorname{rank}(\mathcal{M}_{\hat{\mu}}(t, z, \dot{z}, \ldots, z^{(l+1)})) = (\hat{\mu}+1)\hat{n} - \hat{a},$$
such that there exists a matrix function Z_2 of size $(\hat{\mu}+1)\hat{n} \times \hat{a}$ and with pointwise maximal rank, which is smooth on $\mathcal{L}_{\hat{\mu}}$ and satisfies there $Z_2^T \mathcal{M}_{\hat{\mu}} = 0$.

2. *We have*
$$\operatorname{rank}(A_2(t, z, \dot{z}, \ldots, z^{(l+1)})) = \hat{a} \quad \textit{with} \quad A_2 = Z_2^T \mathcal{N}_{\hat{\mu}}[I_{\hat{n}}\ 0\ \ldots\ 0]^T,$$
such that there exists a matrix function T_2 of size $\hat{n} \times \hat{d}$, $\hat{d} = \hat{n} - \hat{a}$, and with pointwise maximal rank, which is smooth on $\mathcal{L}_{\hat{\mu}}$ and satisfies there $A_2 T_2 = 0$.

3. *We have*
$$\operatorname{rank}(F_{\dot{z}}(t, z, \dot{z}) T_2(t, z, \dot{z}, \ldots, z^{(l+1)})) = \hat{d},$$
such that there exists a matrix function Z_1 of size $\hat{n} \times \hat{d}$ and with pointwise maximal rank, which is smooth on $\mathcal{L}_{\hat{\mu}}$ and satisfies there $\operatorname{rank}(Z_1^T E) = \hat{d}$ *(such that $Z_1^T E T_2$ has full rank $\hat{d}$ on $\mathcal{L}_{\hat{\mu}}$).*

By means of this hypothesis we can define the strangeness-index:

Definition 3.6. *The strangeness-index of the nonlinear differential-algebraic equation given in (3.6) is the smallest integer $\hat{\mu}$ for which Hypothesis 3.5 is satisfied.*
If $\hat{\mu} = 0$, then the differential-algebraic equation is called strangeness-free.

Note that in the points, where any of the ranks changes, i. e., where any of the quantities $\hat{\mu}$, $\hat{a}$, or $\hat{d}$ jumps, the strangeness-index is not defined.

Let us now consider the implications of Hypothesis 3.5 for the solvability of (3.6). By means of the implicit function theorem, one can show that starting from a point $(t_0, z_0, \dot{z}_0, \ldots, z_0^{(\hat{\mu}+1)}) \in \mathcal{L}_{\hat{\mu}}$ the equation $F_{\hat{\mu}} = 0$ locally (i. e., in a neighborhood of $(t_0, z_0, \dot{z}_0, \ldots, z_0^{(\hat{\mu}+1)})$) implies a reduced strangeness-free differential-algebraic equation of the following form:
$$\hat{F}(t, z, \dot{z}) = \begin{bmatrix} \hat{F}_1(t, z, \dot{z}) \\ \hat{F}_2(t, z) \end{bmatrix} = 0, \tag{3.7}$$
where $\hat{F}_1$ and $\hat{F}_2$ are given by
$$\hat{F}_1(t, z, \dot{z}) = Z_1^T F(t, z, \dot{z}, \ldots, z^{(\hat{\mu}+1)}), \quad \hat{F}_2(t, z) = Z_2^T F_{\hat{\mu}}(t, z, \dot{z}, \ldots, z^{(\hat{\mu}+1)}).$$

This result is summarized by the following theorem.

Theorem 3.7. *Let F as in (3.6) be sufficiently smooth and satisfy Hypothesis 3.5 with characteristic values $\hat{\mu}$, $\hat{a}$, and $\hat{d}$. Then every sufficiently smooth solution of (3.6) also solves the reduced problem (3.7) consisting of $\hat{d}$ differential and $\hat{a}$ algebraic equations.*

Proof. See [KM06], page 163. □

However, if we do not start with a solution of the original problem, but only with a point $(t_0, z_0, \dot{z}_0, \ldots, z_0^{(\hat{\mu}+1)}) \in \mathcal{L}_{\hat{\mu}}$, it can happen that the reduced differential-algebraic equation, which is derived in this case, is not solvable. Moreover, even if the reduced problem possesses a solution, it is not clear if this solution also solves the original differential-algebraic equation (3.6). To ensure that the reduced system reflects (at least locally) the properties of the original system concerning solvability and structure of the solution set, the converse direction of Theorem 3.7 is needed. The following theorem gives sufficient conditions for this.

Theorem 3.8. *Let F as in (3.6) be sufficiently smooth and satisfy Hypothesis 3.5 with characteristic values $\hat{\mu}$, $\hat{a}$, and $\hat{d}$, as well as with characteristic values $\hat{\mu}+1$, $\hat{a}$, and $\hat{d}$. Then, for every $(t_0, z_0, \dot{z}_0, \ldots, z_0^{(\hat{\mu}+2)}) \in \mathcal{L}_{\hat{\mu}+1}$, the reduced problem (3.7) has a unique solution satisfying the initial value given by $(t_0, z_0, \dot{z}_0, \ldots, z_0^{(\hat{\mu}+2)})$. Moreover, this solution locally solves the original problem (3.6).*

Proof. See [KM06], page 165. □

Remark 3.9. *Note that the results obtained above are only local results. However, they can be globalized (as in the case of ordinary differential equations) by continuing the process (under the assumption of sufficient smoothness) until the boundary of $\mathcal{L}_{\hat{\mu}}$ or $\mathcal{L}_{\hat{\mu}+1}$, respectively, is reached.*

3.2.2 The Strangeness-Index of the System in (3.4)

To determine the index of the nonlinear differential-algebraic system in (3.4), we proceed as described in Subsection 3.2.1. Therefore, we first rewrite the system in the form

$$F(t, z, \dot{z}) = 0$$

with $t \in \mathbb{I}$, $z = \left[q_h^T, m_h^T, p_h^T\right]^T \in \mathbb{D}_z \subset \mathbb{R}^{\hat{n}}$, $\dot{z} = \left[\dot{q}_h^T, \dot{m}_h^T, \dot{p}_h^T\right]^T \in \mathbb{D}_{\dot{z}} \subset \mathbb{R}^{\hat{n}}$, $\hat{n} = n_q + n_m + n_p$, and $F \in \mathcal{C}(\mathbb{I} \times \mathbb{D}_z \times \mathbb{D}_{\dot{z}}, \mathbb{R}^{\hat{n}})$ given by

$$F = \begin{bmatrix} I_{n_q} & & \\ & 0 & \\ & & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_h \\ \dot{m}_h \\ \dot{p}_h \end{bmatrix} - \begin{bmatrix} M^{-1}(\mathcal{G} + \mathcal{P}) \\ \mathcal{G}_5 \\ C_3^T u_h \end{bmatrix}.$$

Then we determine the Jacobians $E(t) := F_{\dot{z}}(t, z, \dot{z})$ and $A(t) := -F_z(t, z, \dot{z})$ as functions of t. Here, these matrices are given by

$$E = \begin{bmatrix} I_{n_q} & & \\ & 0 & \\ & & 0 \end{bmatrix} \quad \text{and} \quad A(t) = \begin{bmatrix} G(t) & 0 & C \\ 0 & Q(t) & 0 \\ \tilde{C} & 0 & 0 \end{bmatrix},$$

where $G \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{n_q \times n_q})$ and $Q \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{n_m \times n_m})$ denote the following Jacobians:

$$G = M^{-1} \frac{\partial \mathcal{G}}{\partial q_h}, \quad Q = \frac{\partial \mathcal{G}_5}{\partial m_h}.$$

Moreover, $C \in \mathbb{R}^{n_q \times n_p}$ and $\tilde{C} \in \mathbb{R}^{n_p \times n_q}$ are given by

$$C = \begin{bmatrix} -\frac{1}{\rho_f} M_1^{-1} C_2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \tilde{C} = \begin{bmatrix} C_3^T & 0 & 0 & 0 \end{bmatrix}.$$

Remark 3.10. *Note that it does not make any difference whether the system is first discretized and then linearized or vice versa.*

Let us now apply the theory from Subsection 3.2.1 to the nonlinear system, its Jacobians, and the corresponding inflated system. As shown in Subsection 3.2.1, the strangeness-index of a differential-algebraic equation can be determined by finding the smallest value $\hat{\mu} \in \mathbb{N}_0$, for which Hypothesis 3.5 is fulfilled.
Therefore, we start with $\hat{\mu} = 0$ and check if Hypothesis 3.5 is satisfied in this case.

1. First, we have to determine $Z_2 \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{\hat{n} \times \hat{a}})$ with maximal rank such that $Z_2^T \mathcal{M}_0 = 0$ holds. Since $\mathcal{M}_0$ is given by

   $$\mathcal{M}_0 = \begin{bmatrix} I_{n_q} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

   it is clear that $\operatorname{rank}(\mathcal{M}_0) = n_q = \hat{n} - \hat{a}$ holds and we get the condition

   $$\hat{a} = n_m + n_p. \tag{3.8}$$

   Therefore, Z_2 is given by

   $$Z_2 = \begin{bmatrix} 0 & 0 \\ I_{n_m} & 0 \\ 0 & I_{n_p} \end{bmatrix}.$$

   Note that Z_2 is uniquely determined up to a transformation with a nonsingular matrix (which does not have any influence on the rank).

2. Since the matrix A_2 is given by

   $$\begin{aligned} A_2(t) \;=\; Z_2^T \mathcal{N}_0(t) \;&=\; \begin{bmatrix} 0 & I_{n_m} & 0 \\ 0 & 0 & I_{n_p} \end{bmatrix} \begin{bmatrix} G(t) & 0 & C \\ 0 & Q(t) & 0 \\ \tilde{C} & 0 & 0 \end{bmatrix} \\ &=\; \begin{bmatrix} 0 & Q(t) & 0 \\ \tilde{C} & 0 & 0 \end{bmatrix}, \end{aligned}$$

   we get the condition $\hat{a} = \operatorname{rank}(\tilde{C}) + \operatorname{rank}(Q)$ and further with (3.8)

   $$\operatorname{rank}(\tilde{C}) = n_p, \quad \operatorname{rank}(Q) = n_m,$$

which means that $\tilde{C}$ and Q must both have full rank. Furthermore, $n_q \geq n_p$ must hold.
The matrix $T_2 \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{\hat{n} \times \hat{d}})$ has to be chosen such that $Z_2^T \mathcal{N}_0 T_2 = 0$ holds. Therefore, we get

$$T_2 = \begin{bmatrix} 0 \\ 0 \\ I_{n_p} \end{bmatrix}.$$

3. It then follows that

$$\hat{d} = \operatorname{rank}(ET_2) = \operatorname{rank}(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}) = 0,$$

which leads to the contradiction $\hat{n} = \hat{d} + \hat{a} = n_m + n_p < \hat{n}$, since n_q is greater than zero. (Otherwise all the equations for the mean velocity, the turbulent kinetic energy, the energy dissipation rate, and the averaged number density function would not occur in the system.)

Therefore, the strangeness-index of the system is greater than zero.

In the next step, we try out whether Hypothesis 3.5 holds for $\hat{\mu} = 1$:

1. Again, we first determine a matrix $Z_2 \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{2\hat{n} \times \hat{a}})$ which has maximal rank $\hat{a}$ (where $\hat{a}$ is defined by the rank of the kernel of $\mathcal{M}_1$) and fulfills the condition $Z_2^T \mathcal{M}_1 = 0$. Since $\mathcal{M}_1 \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{2\hat{n} \times 2\hat{n}})$ is given by

$$\mathcal{M}_1(t) = \left[\begin{array}{ccc|ccc} I_{n_q} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline -G(t) & 0 & -C & I_{n_q} & 0 & 0 \\ 0 & -Q(t) & 0 & 0 & 0 & 0 \\ -\tilde{C} & 0 & 0 & 0 & 0 & 0 \end{array}\right],$$

we get the condition $2\hat{n} - \hat{a} = \operatorname{rank}(\mathcal{M}_1) = 2n_q + \operatorname{rank}(Q)$. Therefore, it follows that

$$\hat{a} = 2n_m + 2n_p - \operatorname{rank}(Q), \tag{3.9}$$

and Z_2 is given by

$$Z_2^T(t) = \left[\begin{array}{ccc|ccc} 0 & I_{n_m} & 0 & 0 & 0 & 0 \\ 0 & 0 & I_{n_p} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & Q^+(t) & 0 \\ \tilde{C} & 0 & 0 & 0 & 0 & I_{n_p} \end{array}\right],$$

where $Q^+ \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{n_m \times (n_m - \operatorname{rank}(Q))})$ denotes the co-range of Q.

2. Then, we can determine $A_2 \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{\hat{a}\times\hat{n}})$ by

$$A_2(t) = Z_2^T(t)\mathcal{N}_1(t)\begin{bmatrix} I_{\hat{n}} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & Q(t) & 0 \\ \tilde{C} & 0 & 0 \\ 0 & Q^+\dot{Q}(t) & 0 \\ \tilde{C}G(t) & 0 & \tilde{C}C \end{bmatrix}.$$

Since $\operatorname{rank}(A_2) = \hat{a}$, it follows that

$$\hat{a} = \operatorname{rank}\left(\begin{bmatrix} \tilde{C} & 0 \\ \tilde{C}G & \tilde{C}C \end{bmatrix}\right) + \operatorname{rank}\left(\begin{bmatrix} Q \\ Q^+\dot{Q} \end{bmatrix}\right).$$

With this, we get on one hand

$$\hat{d} = \hat{n} - \hat{a} = n_q + n_m + n_p - \operatorname{rank}\left(\begin{bmatrix} \tilde{C} & 0 \\ \tilde{C}G & \tilde{C}C \end{bmatrix}\right) - \operatorname{rank}\left(\begin{bmatrix} Q \\ Q^+\dot{Q} \end{bmatrix}\right),$$

whereas, on the other hand, Eq. (3.9) gives

$$\hat{d} = \hat{n} - \hat{a} = n_q - n_m - n_p + \operatorname{rank}(Q).$$

By equating these two expressions and using the fact that

$$\begin{aligned} \operatorname{rank}\left(\begin{bmatrix} \tilde{C} & 0 \\ \tilde{C}G & \tilde{C}C \end{bmatrix}\right) &\leq n_p + \min(n_p, n_q), \\ \operatorname{rank}(Q) &\leq n_m, \\ \operatorname{rank}\left(\begin{bmatrix} Q \\ Q^+\dot{Q} \end{bmatrix}\right) &\leq n_m, \end{aligned}$$

this leads to the following conditions:

$$n_q \geq n_p, \quad \operatorname{rank}(Q) = n_m, \quad \operatorname{rank}(\tilde{C}) = n_p, \quad \operatorname{rank}(\tilde{C}C) = n_p. \tag{3.10}$$

Therefore, it follows that the matrix Q is nonsingular, and the third row in the matrix A_2 does not occur. Furthermore, $\hat{d}$ and $\hat{a}$ are given by $\hat{d} = n_q - n_p$ and $\hat{a} = 2n_p + n_m$.
Since the condition $A_2T_2 = 0$ must hold, we get for $T_2 \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{\hat{n}\times(n_q-n_p)})$:

$$T_2(t) = \begin{bmatrix} \tilde{C}^- \\ 0 \\ -(\tilde{C}C)^{-1}\tilde{C}G(t)\tilde{C}^- \end{bmatrix},$$

where $\tilde{C}^-$ denotes the kernel of the matrix $\tilde{C}$.

3. Furthermore, since $\operatorname{rank}(\tilde{C}) = n_p$ due to (3.10), it follows that

$$\operatorname{rank}(ET_2) = \operatorname{rank}(\tilde{C}^-) = n_q - n_p = \hat{d}$$

such that we can choose $Z_1 \in \mathcal{C}(\mathbb{I}, \mathbb{R}^{\hat{n}\times\hat{d}})$ as

$$Z_1^T = \begin{bmatrix} I_{n_q} & 0 & 0 \end{bmatrix},$$

yielding that $Z_1^T E(t) T_2 = \left[\tilde{C}^-\right]$ has constant rank $\hat{d}$.

This shows that the strangeness-index of the differential-algebraic system in (3.4) corresponding to the partial differential equations, which describe the dynamical processes in the stirred tank, is equal to one if we choose the spatial discretization in such a way that the conditions in (3.10) are satisfied.

These conditions are easily fulfilled by using a sensible discretization. First of all, one can say that every discretization scheme should satisfy the condition $n_q \geq n_p$, since otherwise there would be more discretization points for the pressure than for the velocity, the turbulent kinetic energy, the energy dissipation rate, and the number density function together. Furthermore, in order to fulfill the condition that Q is nonsingular, we only have to choose a suitable quadrature formula. In addition, the conditions $\operatorname{rank}(\tilde{C}) = n_p$ and $\operatorname{rank}(\tilde{C}C) = n_p$ are satisfied if the additive constant for the pressure is fixed and the continuity equation is discretized appropriately. This follows by the fact that in the Navier-Stokes equations, the pressure is only determined up to an additive constant, since only the gradient of the pressure occurs in the equations, see, e. g., [Wes01].

Since the rank conditions for $\tilde{C}$ and $\tilde{C}C$ already occur in the Navier-Stokes equations, the considered system in (3.4) has the same strangeness-index as the Navier-Stokes equations. This means that the index of the Navier-Stokes equations was not increased by the coupling with the population balance equation. This observation can be summarized by the following corollary:

Corollary 3.11. *All implicit time discretization schemes that are adapted to the solution of the Navier-Stokes equations are also suitable for solving the discussed coupled system.*

Since the differential-algebraic system derived by the spatial discretization of the Navier-Stokes equations has strangeness-index one, it is hard to treat this system in its original form. When solving such a system as it appears originally, one can get into difficulties due to the mixing of differential and algebraic components. Therefore, it is useful to first remove this "strangeness" before solving the differential-algebraic system. Theorem 3.7 ensures that every sufficiently smooth solution of the system in (3.4) is also a solution of a corresponding strangeness-free system of the form (3.7). Theorem 3.8 states that the assumption of a smooth solution can even be relaxed if Hypothesis 3.5 is not only fulfilled for $\hat{\mu} = 1$, but also for $\hat{\mu} + 1 = 2$. However, in the considered application the existence of a sufficiently smooth solution can be assumed.

Many Navier-Stokes solution techniques carry out such an index reduction as described above, although this is often not mentioned explicitly, see [Wei96]. Among such typical solution methods are the so-called "penalty method", the pressure correction methods (also known as operator splitting methods), or the so-called "stream function-vorticity-pressure method", see, e. g., [Wes01], [Wei97].
Within our project, the solver FeatFlow [TB98] is used for the simulation of the Navier-Stokes equations. This method uses a finite element approach for the spatial discretization and a discrete projection scheme for the decoupling of pressure and velocity. In [Wei96], it has been shown that this kind of approach leads to a strangeness-free differential-algebraic system. Therefore, the approach used in FeatFlow is suitable for

the solution of the Navier-Stokes equations.
However, a order reduction in the time discretization may occur, see [Wei96].

4 Simulation of the Flow Field in a Stirred Tank Using FeatFlow

In this chapter, we will consider the simulation of the flow field in a stirred tank using the CFD code FeatFlow (version 1.2). We decided to use FeatFlow, since it is a fast and robust non-commercial CFD code.

Remark 4.1. *For the solution of the Navier-Stokes equations, several CFD codes are available. On one hand, there are commercial codes, whereof* CFX *[CFX],* StarCD *[Sta], and* Fluent *[Flu] are the most established ones. On the other hand, there are non-commercial codes, of which several have been developed by (former) members of the numerics group in Heidelberg (see also* `http://numerik.uni-hd.de/`*), namely* deal.II *[dea] (which is more appropriate for stationary problems),* FeatFlow *[TB98],* Gascoigne3D *[Gas], or* HiFlow *[HiF]. Other non-commercial CFD codes are, e. g.,* OpenFOAM *[Ope] (developed by the company OpenCFD) and* MooNMD *[Moo] (developed at the Otto-von-Guericke University Magdeburg). In addition, there are two implementations of an adaptive finite-element Navier-Stokes solver, one in FORTRAN by Bänsch and Höhn (see [Bän98] and [BH00]) and one C-implementation in the code* Alberta *[Alb] by Siebert. Furthermore, the AWFD library [AWF] can be used to implement a Navier-Stokes solver, at least for problems on simple domains with periodic boundary conditions. Another non-commercial code, which is also particularly suited for the simulation of stirred tanks, is* Fastest *[Fas]. A list of further freely available CFD codes can, e. g., be found at* `www.cfd-online.com` *(for the exact URL, see [CFD]).*

We have decided to use a non-commercial code, where the source code is freely available and can be adapted for our purposes. This is especially important for the coupling (see Chapter 5) and for control and optimization tasks, which are the long-term objectives of the considered application (although this is not the topic of this thesis).

Remark 4.2. *Many commercial codes claim to "solve everything", but the 1995 DFG benchmark [TS96] showed that, in general, this is not true. For these benchmark configurations, some of the commercial codes failed already for laminar flow calculations. Furthermore, commercial codes normally do not have all the new and improved numerical, algorithmic, and computational ideas included yet. In addition, commercial codes have often implemented some kind of (strong) damping so that a variety of flow configurations can be solved. This can lead to wrong results in some cases, since also "real" oscillations are damped by these methods and, thus, "unstable behavior" cannot be reproduced by these codes.*

In contrast, when non-commercial codes are used, the user knows exactly which schemes are used for the different subproblems and, therefore, is aware of problems that might occur. However, non-commercial codes are often badly documented so that it is necessary to work together with the developers. Without doing that, these codes usually cannot be used effectively.

The possibility to work together with the developers was another reason for the choice of FeatFlow.

Since in the considered practical application, the stirred tank is operated with a stirrer speed of about 400 to 700 rounds per minute (which corresponds to Reynolds numbers of about $\mathrm{Re} = 16{,}667$ to $\mathrm{Re} = 29{,}167$), the flow simulation has to be done under the same conditions. This means that high Reynolds numbers, i. e., a turbulent flow, must be simulated. Therefore, the simulation should include some kind of turbulence modeling.
If the simulation is done without any turbulence model, i. e., if direct numerical simulation (DNS) is used, the grid must be so fine that even the smallest eddies are resolved. It is clear then that, since the size of the smallest eddies is orders of magnitude smaller than the size of the computational domain, many grid points are needed.
In fact, the number of grid points N_{DNS} required for a direct numerical simulation in 3D can be estimated by (see [Pas04b])

$$N_{\mathrm{DNS}} = 5^3 \, \mathrm{Re}^{9/4},$$

i. e., it is proportional to $\mathrm{Re}^{9/4}$ (see also [GDN98]). Therefore, in the considered application, about one trillion grid points should be used, which is not feasible with today's computing power.

Remark 4.3. *Even the so-called "Earth Simulator" [ES], a highly parallel vector supercomputer system with a main memory of 10 TB and a theoretical performance of 40 Tflops, which is one of the world's largest computers, is "only" able to perform flow simulations for less than 70 billion grid points, see [AFIK04].*

However, since there was no k-ε turbulence model included in FeatFlow until during this PhD project, the first simulations were done by direct numerical simulation on a grid with about 360,000 to 2.8 million grid points.

This chapter is organized as follows. First, in Section 4.1, we will briefly describe the structure as well as the mathematical background of the solver FeatFlow. Then, in Section 4.2, we will discuss which preparations need to be done in order to be able to simulate a stirred tank reactor with FeatFlow. In Section 4.3, some results of the direct numerical simulation will be shown. These results as well as the corresponding time and memory requirements for the simulation show that using a turbulence model is necessary for the flow simulation. In Section 4.4, we will briefly describe the implementation of a k-ε turbulence model in FeatFlow. The chapter closes with a discussion of the pros and cons of the previously described approach.

4.1 About FeatFlow

In this section, we will describe briefly the structure as well as the mathematical background of FeatFlow. Detailed information can be found in [TB98], [Tur99] and references therein.

FeatFlow is "Finite element software for the incompressible Navier-Stokes equations". It was developed at the University of Heidelberg within the working group of R. Rannacher and S. Turek. (A list of involved persons can be found in Appendix B of the FeatFlow manual [TB98].) In the meantime, S. Turek is at the University of Dortmund, where the code is still enhanced and extended.
FeatFlow is part of the Feast project, which has the aim to develop software that realizes new mathematical and algorithmical ideas in combination with high performance computational techniques. It is based on the finite element packages Feat2D [BHM+95] and Feat3D [HST94], which are written in FORTRAN 77. Thus, also FeatFlow is mainly written in FORTRAN 77. However, some newer extensions have been programmed in FORTRAN 90 and FORTRAN 95. The source code is freely available and can be downloaded from `http://www.featflow.de`.

The structure of FeatFlow can be divided into two parts: an outer control part, which is responsible for the convergence and the accuracy of the overall problem, and an inner solution part, which can be regarded as a "solver engine" for a fixed configuration.
The outer control part splits the complete flow problem into different subproblems, which are controlled by an overall outer iteration. Therefore, it performs all discretization tasks including the definition of initial conditions, boundary values, and all other input parameters that are necessary to solve the resulting discrete problem. Furthermore, it may include error control mechanisms as well as more complex frameworks, e. g., turbulence models. Only this outer part is responsible for the quality of the difference between the discrete solution (if solved sufficiently accurate by the following inner part) and the exact solution.
The inner solution part, on the other hand, provides approximate solutions of the discretized system for a given framework (i. e., for a given mesh, a given number of time steps, etc.). This inner part can be seen as a "solution engine", which tries to find the best solution only for a fixed configuration. However, the differences between the resulting approximation of the prescribed framework and the exact solution are neglected. The control and minimization of these differences is the task of the outer part.

Let us now have a look at the discretization and solution techniques used in FeatFlow. Since FeatFlow solves the Navier-Stokes equations in their dimensionless form, let us start with these equations (cf. Eqs. (A.2) and (A.5) in Appendix A):

$$\left.\begin{aligned} \tfrac{\partial}{\partial t}u - \tfrac{1}{\mathrm{Re}}\Delta u + (u\cdot\nabla)u + \nabla p &= g \\ \nabla\cdot u &= 0 \end{aligned}\right\} \quad \text{in } \Omega\times[0,t_{\mathrm{end}}]. \tag{4.1}$$

In FeatFlow the Rothe method is applied, which means that the Navier-Stokes equations (4.1) are first discretized in time.

Remark 4.4. *The separation of the space and the time discretization is a very common approach for the solution of partial differential equations. It can be realized in two different ways. On the one hand, the method of lines can be applied, which means that the partial differential equation is first discretized in space. This leads to a system of ordinary differential equations or a system of differential-algebraic equations, respectively, which then has to be treated by a time stepping method. On the other hand, the Rothe method can be used, where the partial differential equation is first discretized in time. This leads to a system of stationary partial differential equations, which has to be solved using an appropriate space discretization technique.*
The second approach has the advantage that all well-known schemes from the solution of stationary partial-differential equations can be applied. This is also the reason why many "CFD people" use this approach, since in this field of application stationary problems have been investigated for years, leading to "optimized" solution schemes for this kind of problem. However, this approach has the disadvantage that the error bound for the time discretization is dependent on the Lipschitz constant, which is usually proportional to e^{Re}. The first approach, on the other hand, has the advantage that all the knowledge from the solution of ordinary differential equations or differential-algebraic equations, respectively, can be used. Here, also higher order time discretization schemes can be applied easily. Furthermore, this approach is more appropriate for control and feedback.

If one discretizes the Navier-Stokes equations with one of the usual methods known from the treatment of ordinary differential equations, such as the backward Euler, the Crank-Nicolson or the fractional step θ-scheme, one obtains a sequence of generalized stationary Navier-Stokes equations with prescribed boundary values for every time step. In the case of the one-step θ-schemes this reads as:

Given u^n and the time step $\delta_t = t_{n+1} - t_n$, then solve for u^{n+1} and p^{n+1}

$$\left.\begin{array}{rcl} \frac{u^{n+1}-u^n}{\delta_t} + \theta\left[-\frac{1}{\mathrm{Re}}\Delta u^{n+1} + (u^{n+1}\cdot\nabla)u^{n+1}\right] + \nabla p^{n+1} &=& \tilde{g}^{n+1} \\ \nabla \cdot u^{n+1} &=& 0 \end{array}\right\} \quad \text{in } \Omega$$

with right-hand side

$$\tilde{g}^{n+1} := \theta g^{n+1} + (1-\theta)g^n - (1-\theta)\left[-\frac{1}{\mathrm{Re}}\Delta u^n + (u^n\cdot\nabla)u^n\right].$$

The parameter θ is chosen depending on the time-stepping scheme, e. g., $\theta = 1$ for the backward Euler or $\theta = \frac{1}{2}$ for the Crank-Nicolson scheme.

Since the underlying differential-algebraic equation is highly stiff and of strangeness-index one (cf. Chapter 3), implicit time-stepping schemes must be chosen. The most frequently used schemes are the backward Euler and the Crank-Nicolson scheme, already mentioned above. However, both schemes have disadvantages. The backward Euler scheme is of first order accuracy only, whereas the Crank-Nicolson scheme is not strongly A-stable, leading to unexpected instabilities due to its weak damping property. For this

reason, in FeatFlow additionally the fractional step θ-scheme has been implemented, where every time step is subdivided into three smaller time steps and each substep is solved with a different value for the parameter θ.

In order to compare the fractional step θ-scheme with the one-step θ-schemes introduced above, we define for each (macro) time step $t_n \to t_{n+1} = t_n + \delta_t$ a sequence of three smaller time steps of (variable) size $\delta_{t_i}, i = 1, 2, 3$, which have to fulfill the following condition:

$$\delta_t = \sum_{i=1}^{3} \delta_{t_i}.$$

In the case of the backward Euler scheme or the Crank-Nicolson scheme, we perform three substeps, each with time step $\delta_{t_i} = \frac{\delta_t}{3}, i = 1, 2, 3$, and use the same θ ($\theta = 1$ or $\theta = \frac{1}{2}$, respectively) in each substep.

For the fractional step θ-scheme we proceed as follows. We choose $\theta = 1 - \frac{\sqrt{2}}{2}$, $\theta' = 1 - 2\theta$, $\alpha = \frac{1-2\theta}{1-\theta}$, $\beta = 1 - \alpha$, and split the macro time step $t_n \to t_{n+1} = t_n + \delta_t$ into three consecutive substeps δ_{t_1}, δ_{t_2}, and δ_{t_3} that fulfill the conditions $\delta_{t_1} = \delta_{t_3} = \theta\delta_t$ and $\delta_{t_2} = \theta'\delta_t$. If we further define $\tilde{\theta} := \alpha\theta\delta_t = \beta\theta'\delta_t$, this leads to the following scheme:

$$\begin{aligned}
[I + \tilde{\theta}N(u^{n+\theta})]u^{n+\theta} + \theta\delta_t\nabla p^{n+\theta} &= [I - \beta\theta\delta_t N(u^n)]u^n + \theta\delta_t g^n, \\
\nabla \cdot u^{n+\theta} &= 0, \\
[I + \tilde{\theta}N(u^{n+1-\theta})]u^{n+1-\theta} + \theta'\delta_t\nabla p^{n+1-\theta} &= [I - \alpha\theta'\delta_t N(u^{n+\theta})]u^{n+\theta} + \theta'\delta_t g^{n+1-\theta}, \\
\nabla \cdot u^{n+1-\theta} &= 0, \\
[I + \tilde{\theta}N(u^{n+1})]u^{n+1} + \theta\delta_t\nabla p^{n+1} &= [I - \beta\theta\delta_t N(u^{n+1-\theta})]u^{n+1-\theta} + \theta\delta_t g^{n+1-\theta}, \\
\nabla \cdot u^{n+1} &= 0.
\end{aligned}$$

Here, we have used the abbreviation

$$N(u)u := -\frac{1}{\mathrm{Re}}\Delta u + (u \cdot \nabla)u. \tag{4.2}$$

The fractional step θ-scheme is strongly A-stable and of second order accuracy. Therefore, it combines the advantages of both methods, the backward Euler scheme and the Crank-Nicolson scheme, but with about the same numerical effort (see [Tur96] for numerical tests).

In each case, the time discretization leads to the task to solve nonlinear problems of the following type in each time step:

Given u^n and parameters $\delta_t = \delta_t(t_{n+1})$, $\theta = \theta(t_{n+1})$, and $\theta_i = \theta_i(t_{n+1})$, $i = 1, 2, 3$, then solve for u^{n+1} and p^{n+1}

$$\left.\begin{aligned}
[I + \theta\delta_t N(u^{n+1})]u^{n+1} + \delta_t\nabla p^{n+1} &= [I - \theta_1\delta_t N(u^n)]u^n + \theta_2\delta_t g^{n+1} + \theta_3\delta_t g^n \\
\nabla \cdot u^{n+1} &= 0
\end{aligned}\right\} \text{ in } \Omega. \tag{4.3}$$

For the spatial discretization a finite element approach is used. Here, a non-conforming quadrilateral Stokes element, which uses piecewise rotated multi-linear shape functions

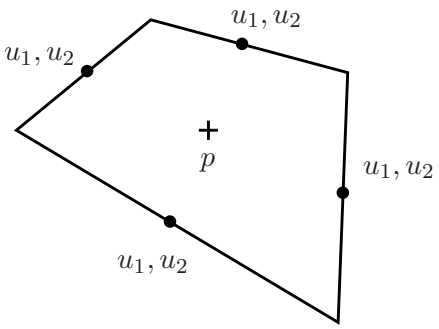


Figure 4.1: Nodal points of the non-conforming $\tilde{\mathrm{Q}}1/\mathrm{Q}0$ element pair in 2D.

for the velocities and piecewise constant pressure approximations, is chosen, see [RT92]. In 2D, the bilinear shape functions for the velocities are spanned by $\langle x_1^2 - x_2^2, x_1, x_2, 1\rangle$. Accordingly, in 3D, trilinear shape functions, spanned by $\langle x_1^2 - x_2^2, x_1^2 - x_3^2, x_1, x_2, x_3, 1\rangle$, are used. The nodal values are the mean values of the velocity vector over the element edges or faces, respectively, or in the corresponding midpoints, and the mean values of the pressure over the elements, see Fig. 4.1. Hence, it can be regarded as the natural quadrilateral analogon of the well-known triangular Stokes element of Crouzeix-Raviart (see [Bra97], for example).

Remark 4.5. *The decision whether triangles or quadrilaterals should be used in the finite element discretization depends on the considered application, see [Bra97]. The use of triangular elements has the advantage that also complex geometries can be covered easily. In the case of quadrilateral elements this is much more complicated, since so-called "hanging nodes" have to be avoided. On the other hand, quadrilaterals are more suitable for parallelization and provide a better error analysis. For further information, see, e. g., [Bra97].*

A convergence analysis for the chosen element pair, usually denoted by $\tilde{\mathrm{Q}}1/\mathrm{Q}0$, can be found in [RT92]. Here, we will only summarize some important features of the element pair, see [Tur99]. First of all, it fulfills the Ladyzhenskaya-Babuška-Brezzi condition without any additional stabilization. In addition, the stability and, hence, the approximation property is also satisfied on meshes containing cells with very large aspect ratios. Furthermore, robust upwind and streamline-diffusion mechanisms can be applied for higher Reynolds number flow, which may lead to matrices with certain M-matrix properties. Compared with other first order elements, the accuracy seems to be just as good, but the considered pair is the only one that is unconditionally stable. In addition, divergence-free nodal bases are easily constructed, which allows the elimination of the pressure variables, leading to positive definite systems for the velocity variables only. These systems can be efficiently solved by preconditioned conjugate gradient or multigrid methods. Further information can also be found in [Tur94].

If we apply this spatial discretization to the semi-discretized Navier-Stokes system and use the same symbols u and p also for the coefficient vectors in the nodal representation

for the functions u and p, the discrete version of problem (4.3) can be written as nonlinear algebraic system of the following form:

Given u^n, a right-hand side $\tilde{g}^{n+1}$ and a time step δ_t, then solve for u^{n+1} and p^{n+1}

$$Su^{n+1} + \delta_t C p^{n+1} = \tilde{g}^{n+1}, \quad C^T u^{n+1} = 0 \tag{4.4}$$

with matrix S and right-hand side $\tilde{g}^{n+1}$ such that

$$Su^{n+1} = \left[M + \theta\delta_t N(u^{n+1})\right] u^{n+1}, \quad \tilde{g}^{n+1} = \left[M - \theta_1\delta_t N(u^n)\right] u^n + \theta_2\delta_t g^{n+1} + \theta_3\delta_t g^n.$$

Here, M is the mass matrix and N is the advection term, consisting of diffusive and convective parts, corresponding to the nonlinear form defined in Eq. (4.2). Furthermore, C is the gradient matrix, whereas $-C^T$ denotes the discretization of the divergence operator.

There are at least two possible approaches how to solve this type of problem [Tur99]:

1. In the first approach, the nonlinearity is first treated by an outer nonlinear iteration of fixed-point or quasi-Newton type or by a linearization technique through extrapolation in time. Then the resulting linear subproblems (Oseen equations) are solved by a direct coupled or by a splitting approach separately for velocity and pressure.

2. In the second approach, the coupled system is first split, which results in a nonlinear Burgers equation for the velocity and a quasi-Poisson problem for the pressure. Then the nonlinearity is treated by an appropriate nonlinear iteration or a linearization technique.

In FeatFlow the first approach is realized in the solvers `cc2d` (2D) and `cc3d` (3D), respectively, in the case of solving the subproblems of Oseen type by a coupled approach. The other case (i. e., the approach, where the subproblems are split) is / will be implemented in the solvers `cp2d` (2D) and `cp3d` (3D), respectively. (The code `cp3d` has not been finished until this thesis was written.) The second approach is implemented in `pp2d` (2D) and `pp3d` (3D), respectively.

Given that the considered application is 3D, we can only choose between the codes `cc3d` and `pp3d`. Since numerical tests [Tur96] have shown that the solvers `pp2d` / `pp3d` are superior to `cc2d` / `cc3d`, especially for highly non-stationary flows, we have used the solver `pp3d` for the flow simulations, see Section 4.3. Therefore, we will focus on this part of FeatFlow and only explain the structure of this solver in more detail in the following, at least for the parts, where the solvers `cc3d` and `pp3d` differ.

Let us start with the discrete nonlinear problem given in Eq. (4.4). One way to solve this is to first eliminate the velocity u^{n+1} from the first equation (the matrix S being invertible)

$$u^{n+1} = S^{-1}(\tilde{g}^{n+1} - \delta_t C p^{n+1})$$

and then solve the so-called "Schur complement" equation for the pressure

$$C^T S^{-1} C p^{n+1} = \frac{1}{\delta_t} C^T S^{-1} \tilde{g}^{n+1}. \tag{4.5}$$

In order to describe the solution method used for this problem, let us start with a general (nonlinear) formulation of the discrete projection method. If we assume that P is a regular preconditioning operator that is chosen "close to S", then Eq. (4.5) can be solved by the following scheme:

Given $p^n =: p^{n,0}$*, then solve for* $l = 1, \ldots, L$

$$p^{n,l} = p^{n,l-1} + \frac{\hat{\alpha}}{\delta_t} \left[C^T P^{-1} C\right]^{-1} C^T S^{-1} \left(\tilde{g}^{n+1} - \delta_t C p^{n,l-1}\right),$$

where $\hat{\alpha} > 0$ *is a relaxation parameter (usually chosen as* $\hat{\alpha} = 1$ *or* $\hat{\alpha} = 2$*). Finally set* $p^{n+1} := p^{n,L}$*,* $\tilde{u}^{n+1} = S^{-1}\left(\tilde{g}^{n+1} - \delta_t C p^{n,L-1}\right)$*, and determine* u^{n+1} *via*

$$u^{n+1} = \tilde{u}^{n+1} - \frac{\delta_t}{\hat{\alpha}} P^{-1} C (p^{n,L} - p^{n,L-1}).$$

If we use the lumped mass matrix $P = M_l$ as preconditioner and set $L = 1$, then the nonlinear discrete projection algorithm can be rewritten in the following form, see [Tur97]:

Given p^n*,* $\delta_t = \delta_t(t_{n+1})$*, and* $S := M + \theta \delta_t N(\cdot)$*.*

1. *Solve for* $\tilde{u}$ *the nonlinear transport-diffusion equation*

$$S\tilde{u} = \tilde{g}^{n+1} - \delta_t C p^n. \tag{4.6}$$

2. *Solve for* q *the discrete Poisson equation*

$$C^T M_l^{-1} C q = \frac{1}{\delta_t} C^T \tilde{u}. \tag{4.7}$$

3. *Update* $p^{n+1} := p^n + \hat{\alpha} q$ *with* $\hat{\alpha} > 0$ *and calculate*

$$u^{n+1} = \tilde{u} - \delta_t M_l^{-1} C q. \tag{4.8}$$

This iteration scheme can be regarded as a special kind of nonlinear iteration scheme for the solution of the generalized Navier-Stokes problem given in (4.4). First the velocity and the pressure are decoupled in an outer iteration, whereas the nonlinear transport-diffusion equation is solved in an inner iteration. However, we only perform one iteration. Nevertheless, this scheme works well for small time steps. This well-known approach is similar to the class of the so-called "classical" projection schemes, see [Tur99].

Since the matrix $C^T M_l^{-1} C$ corresponds to a mixed discretization of the Laplacian operator, this method can also be regarded as a discrete analogon of the schemes proposed

by Chorin [Cho68] or Van Kan [Van86]. The main difference is that we discretize the equations before we perform the projection steps. This leads to a different treatment of the boundary values for the pressure, which is one of the main problems of the continuous schemes. With the approach considered here, boundary layers can be avoided, even for higher Reynolds numbers, see [Tur96].

It remains the question how to solve the nonlinear system given in (4.6). For the solution of this nonlinear transport-diffusion equation, we use the so-called "adaptive fixed point defect correction method". For Eq. (4.6) this leads to the following scheme:

Given $u^n =: \tilde{u}_0$*, then solve for* $l = 1, \ldots, L$

$$\tilde{u}_l = \tilde{u}_{l-1} - \hat{\omega}_l \left[\tilde{S}(\tilde{u}_{l-1})\right]^{-1} \left[S(\tilde{u}_{l-1})\tilde{u}_{l-1} - \tilde{g}^{n+1} + \delta_t C p^n\right], \tag{4.9}$$

where $\hat{\omega}_l$ *is chosen adaptively. The operator* $\tilde{S}(\tilde{u}_{l-1})$ *represents an approximate Frechét derivative, which can simply be chosen as* $\tilde{S}(\tilde{u}_{l-1}) = S(\tilde{u}_{l-1})$*, for example.*
Finally, set $\tilde{u} := \tilde{u}_L$ *and continue with step 2 in the nonlinear discrete projection algorithm on page 52.*

This nonlinear iteration can be terminated if the residual $S(\tilde{u}_{l-1})\tilde{u}_{l-1} - \tilde{g}^{n+1} + \delta_t C p^n$ is small enough or after a given number of steps L.
However, if the method is stopped after $L = 1$ iteration, this approach can be interpreted as a linearization of the convective term by constant extrapolation in time, e. g., replace $(u^{n+1}\cdot\nabla)u^{n+1}$ by $(u^n\cdot\nabla)u^{n+1}$. Another possibility is to use linear extrapolation in time, i. e., using $([2u^n - u^{n-1}]\cdot\nabla)u^{n+1}$ as approximation for $(u^{n+1}\cdot\nabla)u^{n+1}$, with which second order accuracy is obtained. However, the resulting linear systems are nonsymmetric and require special solution techniques. A much simpler approach is to replace $(u^{n+1}\cdot\nabla)u^{n+1}$ by $(u^n\cdot\nabla)u^n$ (fully explicit treatment of the nonlinear terms) and consider this advective term as part of the right-hand side. Then, the corresponding linear systems are symmetric, see [Tur97].

The use of the exact Frechét derivative in (4.9) instead of an approximation (corresponding to the classical Newton scheme) would lead to "convective" terms $(u^n\cdot\nabla)u^{n+1}$, which can be stabilized by upwinding, but also to "reactive" terms of the form $(u^{n+1}\cdot\nabla)u^n$, which may cause instabilities for larger time steps. By choosing an approximate Frechét derivative only (quasi-Newton), these "bad" reactive terms are deleted, see [Tur97]. Further details on the adaptive fixed point defect correction method and especially on the choice of the parameter $\hat{\omega}_l$ can be found in [Tur94] and [Tur99].

Before we summarize how one FeatFlow-step with the solver `pp3d` looks like, we will briefly describe the adaptive step size control implemented in FeatFlow. It is highly recommended to perform the simulations with step size control, especially when the solvers `pp2d` and `pp3d`, respectively, are used, see [Tur96]. In this paper, the reader can also find more details about the derivation of this control, which is based on the estimation of the truncation error.

Given the actual step size δ_t, FeatFlow performs, on one hand, three (sub)steps with this step size δ_t and, on the other hand, one step with the larger step size $3\delta_t$. Afterwards

the relative changes REL_{δ_t}, given by

$$\mathrm{REL}_{\delta_t} := \frac{\|v_{3\delta_t} - v_{\delta_t}\|}{\|v_{\delta_t}\|},$$

are calculated. Here, $v_{3\delta_t}$ and v_{δ_t} denote the solution pairs $\{u, p\}$ obtained by performing one time step with the step size $3\delta_t$ or three time steps with the step size δ_t, respectively. These relative changes are then used to compute the necessary step size $\tilde{\delta}_t$ such that the local relative error is bounded by a given tolerance parameter TOL. This can be done by the following formula:

$$\tilde{\delta}_t \;\leq\; 8\ \mathrm{TOL}\ \delta_t^2\ \frac{1}{\mathrm{REL}_{\delta_t}}.$$

If the estimated value for $\tilde{\delta}_t$ is larger than the used one or only slightly smaller, the result is accepted, and the next macro time step is performed with the step sizes $\tilde{\delta}_t$ and $3\tilde{\delta}_t$. Otherwise, the last calculation is repeated with step size $\tilde{\delta}_t$.

At the end of this section, we summarize one FeatFlow-step with the solver `pp3d`. First, all data are initialized, and all triangulations, linear matrices, pointer structures, and right-hand sides are built on all grid levels. Furthermore, all needed vectors are generated. Then, the parameters for the time stepping scheme and the adaptive time step size are selected, and the time stepping loop is entered. Within this loop, the generalized stationary Navier-Stokes equations corresponding to the actual time level, see Eq. (4.4), are solved by decoupling velocity and pressure, i. e., by applying the nonlinear discrete projection algorithm, see page 52. With given pressure as right-hand side, a nonlinear transport-diffusion equation, see Eq. (4.6), is solved by the adaptive fixed point defect correction method and a multigrid technique for the linear convection-diffusion problems. Using the divergence of this intermediate velocity as right-hand side, the corresponding update problem of quasi-Poisson type for the pressure, see Eq. (4.7), is solved by a multigrid scheme. This auxiliary solution is then added to the old pressure to obtain a new one. In addition, the corresponding velocity field is adjusted, see Eq. (4.8). After this solution part, the time step control is entered. Here, the new time step size is selected and it is decided whether the old time step has to be repeated or the simulation can be continued with the new time step size. Then, the actual solution vector is evaluated. Afterwards the output routines are entered. These routines monitor all actions and give information about the needed workspace and computer time for the performed application.

4.2 Preprocessing

In this section, we will describe which preparations have to be done in order to be able to simulate a stirred tank reactor with FeatFlow. In this context, we will also discuss some particularities of FeatFlow.
In the following subsections, we will consider the grid generation, the specification of initial and boundary conditions, as well as the choice of the solver parameters. Furthermore, we will discuss consequences of the dimensionless calculation used in FeatFlow.

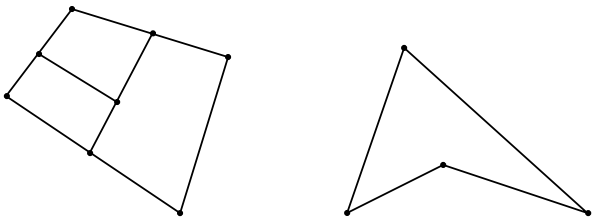


Figure 4.2: Forbidden elements. Left: hanging node, right: non-convex element.

4.2.1 Grid Generation

Before a simulation can be started, an appropriate grid has to be generated. As already mentioned before, FeatFlow uses quadrilateral elements for the finite element discretization. Therefore, in 3D, a grid consisting of hexahedrons must be constructed. As already discussed in the previous section, it is much more complicated to cover a domain with quadrilateral elements than with triangular ones. For the simulation with FeatFlow (as for most CFD codes that use hexahedrons), one has to make sure that the grid only consists of convex elements and that so-called "hanging nodes" are avoided, see Fig. 4.2. In addition, elements with large aspect ratios, i. e., very long and thin elements, should, if possible, not be used. Furthermore, neighboring elements should be of similar size.
Another important point is that the coarse grid should be as coarse as possible, because, in the multigrid algorithm, the system of equations is solved directly on the lowest (= coarsest) level.

Remark 4.6. *In addition to the CFD solver* FeatFlow*, there are several preprocessing tools available from the* FeatFlow*-homepage.*

In 2D, the program DeViSoRGrid *[BG02] can be used. With the FORTRAN-subroutine* `tr2to3`*, which is part of the* FeatFlow*-package, one can, in addition, expand 2D meshes to 3D by "popping" the 2D mesh layers in a sandwich-like technique. (An example for this is the construction of a cylinder from a circle.) More information about the creation of 2D and this special kind of 3D geometries can be found in [Str04].*

The creation of a "real" 3D geometry is much more time-consuming. Here, the preprocessing tool Grid3D *[GRV], a 3D version of* DeViSoRGrid*, can be used. At the time, when the grid for the considered application was created, there was only a beta version of the program available. This beta version was already quite helpful, but it did not avoid the need for understanding the structure of the files describing the grid in order to be able to make final modifications by hand. However, in the meantime, an "official version" has been released, see* DeViSoR*-homepage* `http://www.featflow.de/feast_hp/devisormain.html`*. More information about the program* Grid3D *can be found in the manual [GR05].*

Let us now describe how an appropriate grid for the considered application can be created. Since in FeatFlow all boundary nodes have to lie exactly on the analytic

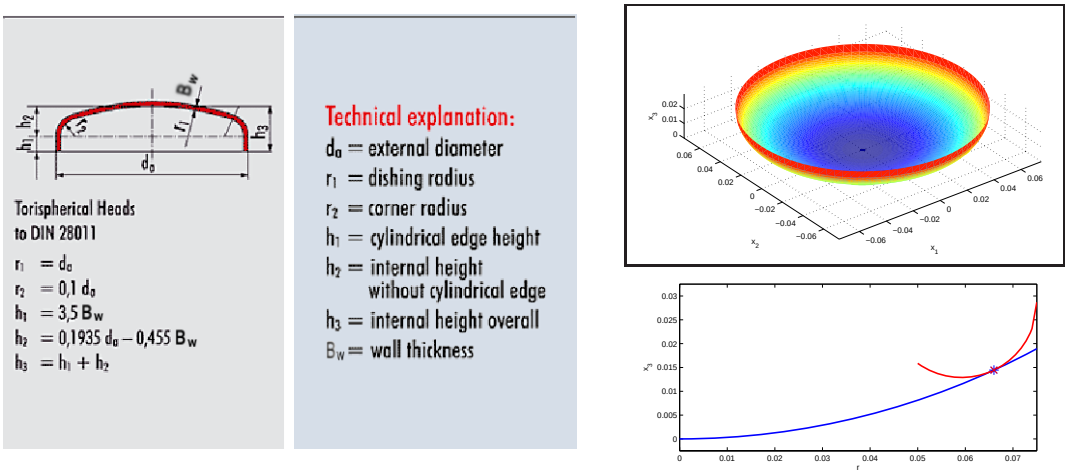


Figure 4.3: Left: Definition of a torispherical head according to DIN 28011 [Sla]. Right: Mathematical description of the torispherical head plotted with MATLAB [Mat] in 3D (top) and 2D, where the two root functions and their osculation point can be seen (bottom).

boundary, we first need a mathematical description of the domain (i. e., of the stirred tank), before we can start with the creation of the grid.

In principle, the tank can be modeled as a cylinder (see Fig. 2.1), except for its rounded bottom part, which is called torispherical head. This torispherical head can mathematically be described by a composed root function, whose measures are defined by the norm DIN 28011, see Fig. 4.3. Due to this definition, the torispherical head is given by two touching spherical lines with radii r_1 and r_2, respectively (see Fig. 4.3). These two radii as well as the height h_2 are defined by the wall thickness B_w, given by $B_w = 0.004\,\mathrm{m}$, and the external diameter $d_a = D_{\mathrm{tank}} + 2B_w$ of the tank. In the considered application, these measures are given by:

$$r_1 := d_a = 0.158\,\mathrm{m}, \quad r_2 := 0.1\ d_a = 0.0158\,\mathrm{m}, \quad h_2 := 0.1935\ d_a - 0.455\ B_w = 0.028753\,\mathrm{m}.$$

Note that the parameter h_1 (and therefore also h_3) does not need to be calculated, since it only describes the cylindric edge height of the torispherical head. Thus, the part of the tank that corresponds to h_1 can be counted to the cylindric part.

By setting up the equations for the spheres, we get two root functions describing the inner and the outer part of the torispherical head, see Fig. 4.3 at the right bottom. However, these two functions do not intersect. Therefore, in order to get a useful mathematical representation of the torispherical head, we had to slightly change one of the parameters. It was decided to "adapt" the value of h_2 from $h_2 = 0.028753\,\mathrm{m}$ to $h_2 = 0.02870696\,\mathrm{m}$, which leads to two touching functions.

Remark 4.7. *It is clear that a useful definition of the torispherical head requires that the given values for the radii and heights lead to two touching functions. Since the given values seem to be rounded, the "adaption" of h_2 (which is maybe even formed by a term of rounded values) is justified.*

By doing this, we get the following mathematical description of the torispherical head:

$$\mathcal{K}(r) = \begin{cases} r_1 - \sqrt{r_1^2 - r^2}, & |r| \leq r_s, \\ h_2 - \sqrt{r_2^2 - (r - \frac{D_{\text{tank}}}{2} + r_2)^2}, & r_s < |r| \leq \frac{D_{\text{tank}}}{2}, \end{cases} \tag{4.10}$$

where $r_s = 0.066\,\text{m}$ denotes the osculation point of the two functions. By determining the inverse function of Eq. (4.10), we get

$$\mathcal{K}^{-1}(x_3) = \begin{cases} \sqrt{r_1^2 - (r_1 - x_3)^2}, & 0 \leq x_3 \leq x_{3,s} \\ \frac{D_{\text{tank}}}{2} - r_2 + \sqrt{r_2^2 - (h_2 - x_3)^2}, & x_{3,s} < x_3 \leq H_{\text{tank}} \end{cases} \tag{4.11}$$

with $x_{3,s} = 0.0144451\,\text{m}$. This formula can be used to determine the radius r that corresponds to a point on the torispherical head with height x_3.

Once we have a mathematical description of the tank, we can calculate the coordinates of the boundary nodes exactly. Thus, we are able to create a coarse grid that satisfies the condition that all boundary nodes lie on the analytic boundary.

In the considered application, this was done in the following way. First, we used the program Blender [Ble] to compose a first version of the tank consisting of a cylinder and a torus (as approximation of the bottom part). The construction of the different parts of the tank was done with the help of the online-tutorials that can be found on the BLENDER-homepage `http://www.blender.org` and with the tutorial by Chronister [Chr04]. The resulting tank was then exported as `dxf`-file, which was used as draft for constructing a coarse mesh of the tank with the help of the program GRID3D. As already mentioned before, there was only a beta version of GRID3D available at the time the coarse grid for the considered application was created. Therefore, the experiences reported in the following refer to this version. In GRID3D, we first created the Cartesian boundary nodes and then the inner nodes. Afterwards, we combined every two neighboring nodes to one edge, four edges to one face (so-called "QuadElement"), and six faces to one hexahedron. By doing this, we achieved a coarse mesh of the tank consisting of 144 nodes and 85 hexahedrons, see Fig. 4.4. One problem of the beta version of GRID3D was that the program was only able to save nodes and elements. Therefore, it was not possible to temporarily save any edges or faces.

Note that we did not model the stirrer or the baffles yet. This will be done by simply marking the corresponding cells as boundary parts and by prescribing boundary values there, see Subsection 4.2.2.

The finer grids are constructed by FEATFLOW automatically by simply halfing every edge so that, in 3D, every element is subdivided into eight smaller ones.

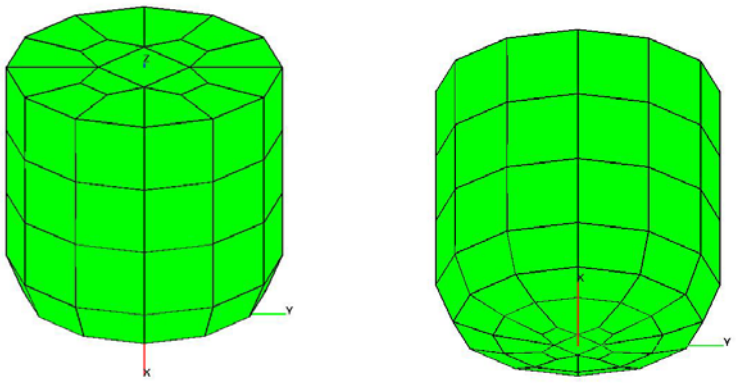


Figure 4.4: Coarse grid consisting of 144 nodes and 85 cells (hexahedrons). Left: view from above, right: view from below.

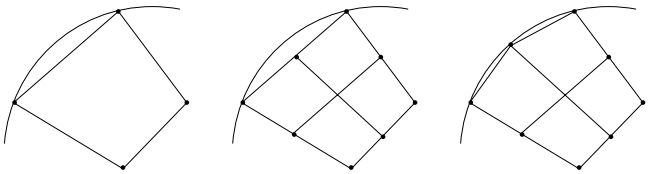


Figure 4.5: Left: coarse grid, middle: refined grid, right: projection of the new boundary node onto the analytic boundary.

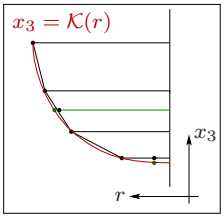


Figure 4.6: Projection in the torispherical head.

Remark 4.8. FeatFlow *does not offer the possibility to refine the grid adaptively during the simulation process. However, in 2D, the FORTRAN-subroutine* `trigen2d` *can be used to create some kind of "locally refined" coarse grid. Such an "adapted" coarse grid can be created in the preprocessing step by refining only certain elements. For doing this,* `trigen2d` *offers different refinement strategies. Further details on this can be found in the* FeatFlow*-manual [TB98]. During the simulation,* FeatFlow *then refines this "adapted" coarse mesh in a regular way.*

Since in FeatFlow all boundary nodes have to lie exactly on the analytic boundary, in every refinement step, the newly created boundary nodes must be projected onto the analytic boundary, see Fig. 4.5. This is why the user has to provide a projection algorithm by adapting the files `parq2d.f` or `parq3d.f`, respectively.

In the considered case, this projection is realized as follows. In the cylindric part of the tank, for every newly created boundary node, the x_1- and x_2-coordinates are scaled to the radius of the cylinder $r_{\text{cyl}} = {}^{D_{\text{tank}}}/_{2}$. One only has to take care that on top of the tank

only the "outer" boundary nodes are projected. However, this can be realized by using a threshold for the distance of the node from the outer part of the analytic boundary. Note that this threshold has to be chosen dependent on the refinement level, since the finer the grid is, the smaller is the difference between the radius of the newly created boundary node and the radius of the tank. In the torispherical head part, the projection can be done in the same way as in the cylindric part, except that one has to take into account that the height x_3 is dependent on the radius r. Therefore, we first have to calculate the radius r that corresponds to the height x_3 by $r = \mathcal{K}^{-1}(x_3)$, see Eq. (4.11), and can then scale x_1 and x_2 with this radius. However, in the coarse mesh, there is a small area at the middle bottom of the tank, where the rounded torispherical head is approximated by a horizontal plate, see Fig. 4.4. If the above described projection is applied to that part as well, it causes all newly created boundary nodes to be projected onto the same circle leading to a "hole" in the bottom of the tank. Therefore, for these points, we have to keep x_1 and x_2 unchanged and project the boundary nodes in vertical direction onto the analytic boundary by scaling x_3 with the height that corresponds to the radius $r = \sqrt{x_1^2 + x_2^2}$, see Fig. 4.6. This height is given by $\mathcal{K}(r)$, see Eq. (4.10).

Fig. 4.7 illustrates the implemented algorithm. The projection in the cylindric part is always in horizontal direction, whereas the projection in the torispherical head part is – depending on the considered height – in horizontal or in vertical direction.

4.2.2 Specification of Initial and Boundary Values

In FeatFlow, initial and boundary values are specified in the FORTRAN-files `indat2d.f` and `indat3d.f`, respectively. There, the user can prescribe different types of boundary conditions, like Dirichlet boundary conditions for the velocity (inflow) or natural outflow conditions, for example. In the default case, i. e., if no boundary values are prescribed by the user, all boundary parts are treated as rigid walls.

Remark 4.9. *One important point about the specification of boundary conditions in* FeatFlow *is that the prescribed values are only valid on boundary nodes. This means that, if we prescribe values for the velocity or for the pressure somewhere else, these values are simply ignored.*

Further information about what other types of boundary conditions can be prescribed in FeatFlow can be found in [TB98] and [Tur99]. An explanation of the meaning of the different variables in the files `indat2d.f` and `indat3d.f` can be found in [Str04].

For the stirred tank reactor, we use the default case of rigid walls for the whole outer boundary of the tank. In addition, for the sake of simplicity, we substitute the "free surface"-boundary conditions

$$\frac{\partial u_1}{\partial \mathbf{n}_1} = 0, \quad \frac{\partial u_2}{\partial \mathbf{n}_2} = 0, \quad u_3 = 0$$

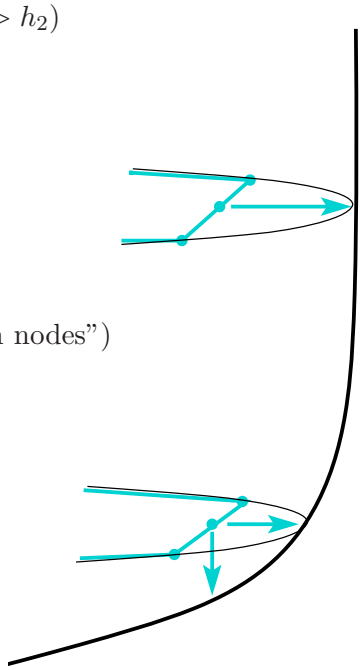
at the top of the tank (see Section 2.2) by Dirichlet-zero boundary conditions for the velocity, which corresponds to a lid on top of the tank. This can be done, since in

Algorithm:

Given: $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$: the coordinates of the newly created boundary node,
H_{tank}: the height of the tank,
h_2: the height of the torispherical head part,
$x_{3,\min}$: the height of the "bottom nodes" in the coarse grid,
$r_{\text{cyl}} = {}^{D_{\text{tank}}}/_{2}$: the radius of the cylinder,
r_{tol}: the threshold radius for "outer" boundary nodes depending on the actual refinement level.

If ($\tilde{x}_3 == H_{\text{tank}}$ and $\sqrt{\tilde{x}_1^2 + \tilde{x}_2^2} > r_{\text{tol}}$) or ($H_{\text{tank}} > \tilde{x}_3 > h_2$)
("outer" nodes in cylindric part)

- $x_3 = \tilde{x}_3$, i. e., keep x_3 unchanged
- calculate
  $x_1 = \tilde{x}_1 \cdot \frac{r_{\text{cyl}}}{\sqrt{\tilde{x}_1^2+\tilde{x}_2^2}}$
  $x_2 = \tilde{x}_2 \cdot \frac{r_{\text{cyl}}}{\sqrt{\tilde{x}_1^2+\tilde{x}_2^2}}$



else if $\tilde{x}_3 > x_{3,\min}$ (torispherical head part, no "bottom nodes")

- $x_3 = \tilde{x}_3$, i. e., keep x_3 unchanged
- calculate
  $r = r(x_3) = \mathcal{K}^{-1}(x_3)$ by Eq. (4.11)
  $x_1 = \tilde{x}_1 \cdot \frac{r}{\sqrt{\tilde{x}_1^2+\tilde{x}_2^2}}$
  $x_2 = \tilde{x}_2 \cdot \frac{r}{\sqrt{\tilde{x}_1^2+\tilde{x}_2^2}}$

else ("bottom nodes")

- $x_1 = \tilde{x}_1$, $x_2 = \tilde{x}_2$, i. e., keep x_1, x_2 unchanged
- calculate
  $x_3 = x_3(r) = \mathcal{K}(r)$ with $r = \sqrt{x_1^2 + x_2^2}$ by Eq. (4.10)

end

Figure 4.7: Algorithm describing the projection of boundary nodes in a stirred tank with torispherical head during the refinement process.

[NW90] it has been shown that the lid has only an influence on the flow field in the direct vicinity of the lid / free surface, and this part of the stirred tank reactor does not play an important role in the considered application.

It remains to implement the (moving) stirrer as well as the (fixed) baffles. Although these parts seem to be very different, in FeatFlow they are treated by the same principle, namely by the so-called "fictitious boundary method", a method that is suitable for incorporating complicated small-scale and / or time-dependent geometrical details into the flow computation [TDR02] (see also [WTR03]).

This approach belongs to the class of "fixed grid approaches", which means that the computational mesh is fixed, whereas the solid boundaries are allowed to move freely through the given mesh. One big advantage of the "fixed grid approaches" over the conventional "body-conformal approaches", where the computational mesh is kept in accordance to the geometrical details, is that the CPU costs are significantly lower, since the computational mesh remains unchanged during the simulation, i. e., the expensive regeneration or deformation of the computational grid can be avoided.

Since in FeatFlow the fictitious boundary method is used in the context of multigrid solvers, the complex geometries and their associated boundary curves or surfaces have to be approximated in a hierarchical way. Here, the design of an appropriate fine mesh, which resolves all the geometrical details, is much easier than the construction of a sequence of coarser meshes, especially if one keeps in mind that sequences of more or less nested meshes are necessary for multigrid [TDR02].
One possible approach is to start with a relatively fine coarse mesh that contains already most of the geometrical fine-scale details. However, for 3D applications, this approach is not feasible, since the resulting "coarsest" mesh usually consists of so many cells that the typical multigrid efficiency is lost due to the amount of work needed for the solution on the coarsest grid.
Another approach is to use non-nested grids, where the coarser meshes are completely different to the finest grid. However, in this case, the resulting multigrid solver spends most of its time with the interpolation from one mesh to another. Furthermore, the convergence rates may deteriorate, since they strongly depend on the choice of the subgrids and the corresponding transfer operators [TDR02].

Taking these problems into account, in FeatFlow the fictitious boundary method is applied in combination with the so-called "iterative filtering technique". In this approach, a (rough) boundary parameterization, which sufficiently describes all the large-scale structures with regard to the boundary conditions, is used on the coarse mesh, whereas all the fine-scale features are treated as interior objects. This is done in such a way that the corresponding components in all matrices and vectors are unknown degrees of freedom, which are implicitly incorporated into all iterative solution steps. By doing this, standard tools can be used for the refinement of the grid in the interior regions and very accurate approximations can be obtained. In addition, the corresponding vector components are projected onto the subspace of "correct" boundary conditions by the iterative filtering technique, before and directly after each iteration substep. This en-

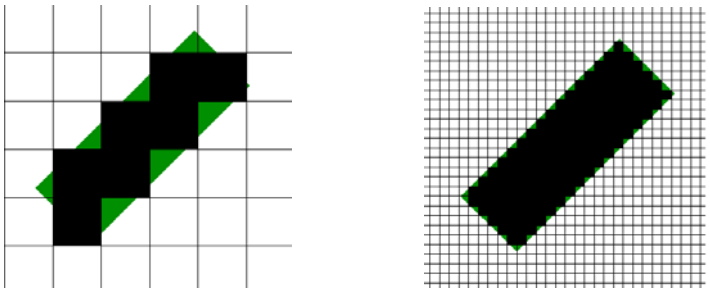


Figure 4.8: $O(h)$-approximation of a moving blade in FeatFlow. Green: analytic description of the blade, black: approximation in FeatFlow leading to a "zig-zag" approximation of the boundary. Left: very coarse mesh, right: finer mesh.

sures the typical performance of standard multigrid solvers without requiring additional modifications in the software components [TDR02].

Another advantage of this approach, at least if appropriate mesh-dependent filters are applied, is that the number and / or the complexity of the shape of "interior" objects can be decreased when the mesh is coarsened. In some cases, it might even happen that the coarsest configuration does not contain any object at all (in this case, the objects are only present on the finer levels). For further details, see [TDR02].

On the other hand, the "fictitious boundary method" only leads to an $O(h)$-approximation of the boundaries (due to the piecewise constant approximation of the interfaces), see Fig. 4.8. Therefore, this approach is particularly suitable for the simulation of the qualitative flow behavior of complex configurations, but if one likes to get quantitative results, then a good approximation of the "real" boundaries and therefore a very fine mesh is needed. If a (too) coarse grid is used, we only get a very rough approximation of the analytic boundary. However, also finer grids can lead, depending on the shape of the analytic boundary, to "zig-zag" approximations.

More information about the idea, the implementation, and the advantages of the fictitious boundary method can be found in [TDR02]. Here, we just want to give a less formal, but more demonstrative description of the approach.
In order to get a better imagination of how the fictitious boundary method works, one can think of a fixed mesh, through which the solid boundaries are moving. These "fictitious" boundaries are then incorporated into the calculation by deciding in each time step for every grid cell whether it belongs to the solid boundary or not. If a cell belongs to the fictitious boundary, it is marked so that boundary values can be prescribed there. Note that in FeatFlow boundary values can not only be prescribed for "real" boundary nodes, but also for "artificial" ones.
This "marking of cells" is realized in the file `bndry.f`. Here, the user has to implement a mathematical description of the (time-dependent) domain of the (moving) interior objects.

Remark 4.10. *Besides the fictitious boundary method, there are several other approaches how complicated boundaries and / or rotating or moving obstacles can be treated within a flow simulation.*

In process engineering, one mainly uses the following three approaches for the simulation of stirred tanks [SD03]. A review of these methods can be found in [BCGM98]. Here we will only give a short description of these approaches.

- *In the first approach, which is called "impeller boundary condition (IBC) method" in [BCGM98], the geometry of the impeller is not resolved by the computational mesh. (The impeller region is excluded from the computational domain.) The effects of the impeller are rather modeled by imposing suitable, empirically derived boundary conditions to the external flow. However, satisfactory results can only be obtained if reliable empirical data are available for the flow near the impeller.*

- *In the second method, which is called "inner-outer (IO) iterative procedure" in [BCGM98], the domain of the whole vessel is divided into two concentric, partially overlapping zones. In both regions, geometrical details – like the stirrer, the baffles and / or the geometry of the vessel – are resolved. In the inner domain, the flow field is calculated in a reference frame that is rotating with the impeller, whereas, in the outer region, the simulation is performed in the reference frame of the laboratory. In both regions, the simulations are done under steady-state assumptions with respect to their own reference frame. After azimuthally averaging and transforming for the relative motion, information is iteratively exchanged between the two regions. In the rotating domain, the transformation of the conservation laws yields two additional terms, namely the centrifugal and the Coriolis force.*
  Note that this approach is often referred to as "multiple frame of reference (MFR) method", although there are slight differences between the inner-outer iterative procedure described above and the multiple frame of reference method introduced in [LIG94]. In [SD03], one can also find the term "snapshot approach". For further details and more information concerning the differences between these approaches, see [BCGM98].
  A disadvantage of these approaches is that they only provide a stationary solution of the flow field for a given position of the stirrer with respect to the baffles. The rotating domain has to be selected in such a way that the flow at the boundary to the stationary domain is almost steady. Thus, this approach is only suitable for cases, where the impeller is small compared to the diameter of the vessel [SD03].

- *In the third approach, which is called "sliding-grid (SG) method" in [BCGM98] and "sliding or clicking mesh method" in [SD03], the tank is divided into two concentric non-overlapping blocks, where the inner one is rotating with the impeller and the outer one is stationary. Note that, unlike in the inner-outer iterative procedure, the flow equations in the rotating domain are written with respect to the fixed ("laboratory") reference frame, while it is the grid that rotates. However, the rotation of the grid results in acceleration terms that are equivalent to the forces*

Name	Short description	Pros & cons
impeller boundary condition method	exclusion of the impeller region from the computational domain, modeling of the effects of the impeller by imposing suitable, empirically derived boundary conditions	requires reliable empirical data for the flow near the impeller
inner-outer iterative procedure / multiple frame of reference method	division of the whole vessel into two (partially overlapping) zones: one inner domain, which is rotating with the impeller, and one outer domain, which is fixed, steady-state assumptions for the simulation in both regions	provides only a stationary solution for a given impeller-to-baffle position
sliding / clicking mesh method	division of the whole vessel into two non-overlapping zones: one inner domain, in which the grid is rotating with the impeller, and one outer domain, in which the grid is fixed	resolves unsteady nature of the flow, higher computational effort

Table 4.1: Comparison of the three most commonly used methods for the treatment of rotating impellers in the flow simulation of stirred tanks.

that arise in the second approach [BCGM98].
The difference between the sliding and the clicking mesh method is that in the former one, the two regions can deform, whereas in the latter one the grid lines of the inner and the outer domain must always match.
Obviously, the sliding as well as the clicking mesh technique both require a considerably higher computational effort than the second approach, since several revolutions of the impeller must be calculated in order to get a quasi-steady state solution [SD03].
An application of the sliding mesh method for the simulation of a stirred tank reactor can be found in [BLWC97] or [BLWC98], respectively.

For the simulation of stirred vessels, one mainly uses the three methods introduced above, which are also summarized in Table 4.1. However, in general, there are many more approaches for modeling complicated and / or moving boundaries within a flow simulation. Among these approaches are the following:

- *"Fictitious domain methods" (see, e. g., [BTT97] or [GPP94]).*
  The idea of these methods is to embed the original (complicated) domain into a much simpler one (the so-called "fictitious domain") so that, after a reformulation, the problem can be solved there.

- *"Level set methods" (see [OF03] and [Set99]).*
  This method tracks the motion of the fluid-structure interface by defining a so-called "level set function", which describes possible evolutions of the interface. To

derive the motion of the fluid-structure interface, an additional equation has to be solved.

- *"Chimera methods" (see [Ste91] or [SB87]).*
  In this approach, several separately generated, but overlapping unstructured grids are used for the discretization of the domain. The idea of generating all grid components independently of the each other simplifies the construction of a mesh for complicated and / or moving objects considerably. The exchange of boundary information is realized by interpolation.

- *Standard arbitrary Lagrangian-Eulerian (ALE) formulations with frequent remeshing (see [MH97]).*

All the above mentioned methods belong to the class of mesh-based methods, since they all need a mesh for the simulation.
However, also so-called mesh-free methods can be applied for the flow simulation with complicated and / or moving boundaries. The idea of these methods is to approximate the partial differential equations on a set of nodes, but without an additional mesh. Therefore, these methods are able to solve problems, where good meshes are difficult to maintain. An overview of mesh-free methods can be found in [FM04].
Another approach, which is introduced in [Fri05], is to couple mesh-free and mesh-based methods.

As already mentioned before, FeatFlow uses the "fictitious boundary method" combined with an "iterative filtering technique" for incorporating geometrically complicated objects or time-dependent boundary components.
Therefore, in the considered application, the user has to provide a mathematical description of the stirrer and of the baffles by adding (a part of) a subroutine to the file `bndry.f`. Then, in each time step, every cell that belongs to this domain is marked as a "fictitious" boundary node. Once a cell is marked, boundary values can be prescribed for it.
The following formulae give a mathematical description of the moving stirrer and of the baffles for the considered application. This description forms the basis for the implementation of the domain in the file `bndry.f`:

$$
\begin{aligned}
\Omega_{\text{stirrer}} = \quad & \left\{(x_1,x_2,x_3)^T \in \mathbb{R}^3 :\ x_1^2+x_2^2 \le \tfrac{D_{\text{shaft}}^2}{4},\ H_l \le x_3 \le H_{\text{tank}}\right\} \\
\bigcup \quad & \left\{(x_1,x_2,x_3)^T \in \mathbb{R}^3 :\ x_1^2+x_2^2 \le \tfrac{D_{\text{disk}}^2}{4},\ \left|x_3 - \left(H_l + \tfrac{H_{\text{stirrer}}}{2}\right)\right| \le \tfrac{H_{\text{disk}}}{2}\right\} \\
\bigcup \quad & \left\{(x_1,x_2,x_3)^T \in \mathbb{R}^3 :\ x_1^2+x_2^2 \le \tfrac{D_{\text{hub}}^2}{4},\ H_l \le x_3 \le H_l + H_{\text{stirrer}}\right\} \\
\bigcup_{j=1,\dots,6} \quad & \Big\{(x_1,x_2,x_3)^T \in \mathbb{R}^3 : \\
& \quad \tfrac{D_{\text{stirrer}}}{2} - D_{\text{blade}} \le \cos(j\tfrac{\pi}{3}+t\omega)x_1 + \sin(j\tfrac{\pi}{3}+t\omega)x_2 \le \tfrac{D_{\text{stirrer}}}{2}, \\
& \quad -\tfrac{B_{\text{blade}}}{2} \le -\sin(j\tfrac{\pi}{3}+t\omega)x_1 + \cos(j\tfrac{\pi}{3}+t\omega)x_2 \le \tfrac{B_{\text{blade}}}{2}, \\
& \quad H_l \le x_3 \le H_l + H_{\text{stirrer}}\Big\}
\end{aligned}
$$

$$\Omega_{\text{baffles}} = \bigcup_{j=1,\dots,4} \Big\{ (x_1, x_2, x_3)^T \in \mathbb{R}^3 : \\ \tfrac{D_{\text{tank}}}{2} - D_s - D_{\text{baffle}} \le \cos(j\tfrac{\pi}{2})x_1 + \sin(j\tfrac{\pi}{2})x_2 \le \tfrac{D_{\text{tank}}}{2} - D_s, \\ -\tfrac{B_{\text{baffle}}}{2} \le -\sin(j\tfrac{\pi}{2})x_1 + \cos(j\tfrac{\pi}{2})x_2 \le \tfrac{B_{\text{baffle}}}{2}, \\ H_l + \tfrac{H_{\text{stirrer}}}{2} \le x_3 \le H_{\text{tank}} \Big\},$$

where Ω_{stirrer} and Ω_{baffles} denote the domain of the stirrer and of the baffles, respectively. Furthermore, D_{tank} and H_{tank} are the diameter and height of the tank, H_l is the height of its bottom part, D_{stirrer} and H_{stirrer} are the diameter and the height of the stirrer, D_{blade} and B_{blade} are the diameter (width) and length of each blade, D_{disk} and H_{disk} are the diameter and the height of the disk, D_{hub} is the diameter of the hub in the middle, D_{shaft} is the diameter of the shaft, D_{baffle} and B_{baffle} are the diameter (width) and the length of each baffle, and D_s is the distance of the baffles to the wall. Some of these quantities can be seen in Fig. 2.1 on page 6. The values of all quantities are given by

$$D_{\text{tank}} = 0.15\,\text{m},\ H_{\text{tank}} = 0.15\,\text{m},\ H_l = 0.05\,\text{m},$$
$$D_{\text{stirrer}} = 0.05\,\text{m},\ H_{\text{stirrer}} = 0.01\,\text{m},\ D_{\text{blade}} = 0.012\,\text{m},\ B_{\text{blade}} = 0.001\,\text{m},$$
$$D_{\text{disk}} = 0.038\,\text{m},\ H_{\text{disk}} = 0.001\,\text{m},\ D_{\text{hub}} = 0.013\,\text{m},\ D_{\text{shaft}} = 0.009\,\text{m},$$
$$D_{\text{baffle}} = 0.013\,\text{m},\ B_{\text{baffle}} = 0.001\,\text{m},\ D_s = 0.003\,\text{m}.$$

In addition, t denotes the time, and ω is the angular velocity, which is determined by the rotational speed N^*, see Section 2.2.

4.2.3 Solver Parameters

In the files `pp2d.dat` and `pp3d.dat`, respectively, which are located in the `#data`-subdirectory, several solver parameters can be specified. Here, one can select the particular schemes as well as the corresponding parameters for the different subproblems that we have already considered in Section 4.1. Furthermore, different in- and output options can be chosen in these files.
In the following, we will exemplarily explain the meaning of some of these parameters, which have been picked from different "categories" in order to give an overview. These selected parameters are also summarized in Table 4.2. A list and short description of all parameters can be found in [TB98].

The type of time discretization used by FeatFlow is determined by the parameter `IFRSTP`, for example, whereas `THETA` specifies the value of the parameter θ if a one-step θ-scheme is used, see Section 4.1. Concerning the multigrid algorithm for the velocity, `ICYCU` specifies the type of multigrid cycle, `ILMINU` and `ILMAXU` the minimal and maximal number of linear multigrid steps, `ISLU` the type of multigrid solver, and `ISMU` the type of multigrid smoother, to give just a few examples. For the pressure, the corresponding parameters are `ICYCP`, `ILMINP`, `ILMAXP`, `ISLP`, and `ISMP`. In addition, `NLMIN` and `NLMAX` state the minimal and maximal multigrid level. Here, level i means that the coarse grid has been refined $(i-1)$ times. Obviously, if `NLMIN` equals `NLMAX`,

CPARM	name of parameterization file
CMESH	name of coarse mesh file
CFILE	name of protocol file
ISTART	input of start vector
	$= 0$: start with homogeneous vector (only boundary conditions)
	$\neq 0$: read start vector
CSTART	name of start vector file
ISOL	output of solution vector
	$= 0$: no output
	$\neq 0$: output
CSOL	name of solution vector file
NLMIN / NLMAX	minimal / maximal multigrid level
IMASS	parameter for mass matrix type (lumped or real mass matrix)
IMASSL	parameter for element type of lumped mass matrix
IUPW	parameter for convective terms (streamline diffusion or upwinding)
INLMIN / INLMAX	minimal / maximal number of nonlinear iterations used for the solution of the nonlinear transport-diffusion equation
	INLMIN = INLMAX = 1: linear extrapolation in time
	INLMIN = INLMAX = −1: constant extrapolation in time
ICYCU / ICYCP	parameter for multigrid-cycle for velocity / pressure (F-, V-, or W-cycle)
ILMINU / ILMINP	minimal number of multigrid-steps for velocity / pressure
ILMAXU / ILMAXP	maximal number of multigrid-steps for velocity / pressure
ISMU / ISMP	parameter for multigrid-smoother for velocity / pressure (Jacobi, SOR, SSOR, or ILU)
ISLU / ISLP	parameter for multigrid-solver for velocity / pressure (SOR, BiCGSTAB / CG, ILU, or BiCGSTAB+ILU / CG+ILU)
RE	parameter for viscosity (Reynolds number)
EPSUR	stopping criterion for relative changes in velocity in nonlinear iteration
IPROJ	parameter for type of projection scheme
	$= 0$: first order (Chorin)
	> 0: second order (Van Kan)
	< 0: $\|$IPROJ$\|$ steps of first order, then second order
NITNS	maximum number of macro time steps
TIMENS	absolute start time
THETA	parameter for time stepping value (Implicit Euler or Crank-Nicolson, only used if IFRSTP = 0)
TSTEP	starting time step
IFRSTP	parameter for time stepping scheme (one-step or fractional step scheme)
DTGMV	time difference for gmv-output
IGMV	level for gmv-output
IADTIM	parameter for adaptive time step control
	$= 0$: no control, fixed time step TSTEP is used
	$\neq 0$: time step control is used
TIMEMX	maximum absolute time
DTMIN / DTMAX	minimal / maximal time step during adaptive control
DTFACT	factor for largest possible time step changes
IEPSAD	parameter for type of error control

Table 4.2: Short description of some selected parameters that can be specified in the file pp2d.dat or pp3d.dat, respectively.

no multigrid algorithm is used. Other examples are the parameters `INLMIN` and `INLMAX`, which specify the minimal and maximal number of nonlinear iterations used for the solution of the nonlinear transport-diffusion equation. Thus, these parameters specify the minimal and maximal number of times the adaptive fixed point defect correction method, given by Eq. (4.9) on page 53, is repeated. There are many more parameters in the files `pp2d.dat` and `pp3d.dat`, respectively, that specify what kind of schemes or approximations are used in the different discretization steps, like the parameters `IMASS` and `IMASSL`, which specify whether any and (if so) what kind of lumped mass matrix is used, the parameter `IUPW`, which indicates if upwinding for the convective terms is applied, or the parameter `IPROJ`, which specifies the type of projection scheme.

Furthermore, in the files `pp2d.dat` and `pp3d.dat`, respectively, one has to specify the name of the file that contains the mesh. In 2D, a so-called parameterization file (`prm`-file) as well as a coarse mesh file that also contains the triangulation (so-called `tri`-file) are needed, whereas in 3D the latter one suffices. The name of these files are given by the parameters `CPARM` and `CMESH`, respectively. The output of the simulation is written to a file specified by the parameter `CFILE`. Furthermore, the parameter `ISTART` indicates if there is a solution file the simulation can start with, and `CSTART` gives the name of this file. Accordingly, the parameter `ISOL` states if the (new) solution is written to a file, and `CSOL` specifies the name of the file.

Concerning the control of the duration of a simulation, the parameters described in the following can be used. Indeed, there are several reasons for the termination of a simulation in FeatFlow, namely if the maximum number of time steps `NITNS` or the maximum absolute time `TIMEMX` is reached or if one of the stopping criteria is fulfilled, e. g., if the relative changes in the velocity in the nonlinear iteration are below a certain tolerance `EPSUR`. The parameter `IADTIM` indicates if an adaptive step size control is used, whereas `IEPSAD` specifies its type. If the simulation is done with adaptive step size control, the minimal and maximal step size as well as the maximal factor for the changes of the step size are given by `DTMIN`, `DTMAX`, and `DTFACT`, respectively. The absolute start time is given by `TIMENS`, whereas `TSTEP` specifies the start time step size.

The parameter `RE` denotes the Reynolds number of the considered application, which states how turbulent the flow is. As we will see in the following subsection as well as in Appendix A, this quantity describes the characteristics of the flow in a certain geometry sufficiently.

Another set of parameters that can be specified in the files `pp2d.dat` and `pp3d.dat`, respectively, are the options for the output. It is recommended to use the `gmv`-format for the output, since the corresponding program GMV (short for General Mesh Viewer) [Ort] is a quite powerful post-processing tool, at least in comparison to other freely available visualization tools. However, one has to keep an eye on the size of the output files. In the considered application, the `gmv`-files of level 5 (360,369 nodes) had already a size of 63 MB, and on level 6 (2,833,761 nodes) the size was 493 MB. That is why it is suggested to use a lower level (i. e., a coarser grid) for the `gmv`-output than for the simulation, at least if the output is saved for several time points. The level of the

gmv-output is specified by the parameter IGMV (same for avs- and film output). DTGMV gives the time difference for the gmv-output. If this parameter is set to zero, FeatFlow writes output after every time step. The gmv-output files are written to the subdirectory #gmv. They are called u.*int*.gmv, where *int* denotes the number of the time step, in which the output was written.

Further information about the files pp2d.dat and pp3d.dat (and also about the corresponding files cc2d.dat and cc3d.dat that are used to specify the parameters for the coupled solvers cc2d and cc3d) can be found in the FeatFlow-manual [TB98].

4.2.4 Dimensionless Calculation

As already mentioned in Subsection 4.1, FeatFlow solves the Navier-Stokes equations in their dimensionless form, which means that it does not matter which size the computational domain has or which velocity is prescribed. The only quantity that characterizes the flow is the Reynolds number, see Appendix A. Therefore, the velocity as well as the geometrical parameters can simply be normalized to one. However, in order to compare the simulation results with experimental data, one has to work with the "real" size of the domain and the "real" velocity that corresponds to a certain Reynolds number.

Nevertheless, the simulations can also be done with scaled values. In this case, one only has to determine the Reynolds number that characterizes the considered application before the simulation. In the case of a stirred tank reactor, the Reynolds number can be calculated by the following formula:

$$\mathrm{Re} = \frac{N^* {D_{\mathrm{stirrer}}}^2}{\nu},$$

where N^* is the rotational speed, D_{stirrer} is the diameter of the stirrer and ν is the kinematic viscosity of the stirred fluid. In order to get comparable simulation results, we only have to prescribe the appropriate Reynolds number and transform back the results after the simulation by scaling velocity and geometrical quantities. However, this also leads to a different time scale.

In the considered application, the simulation was done on a grid with diameter $D_{\mathrm{tank}}^{\mathrm{sim}} = 3$ for the stirred tank. As maximal velocity we prescribed $u_{\mathrm{tip}}^{\mathrm{sim}} = 0.5$ at the tip of the blades of the stirrer. However, in "reality" the tank has a diameter of $D_{\mathrm{tank}} = 0.15\,\mathrm{m}$ and the maximal velocity of the stirrer, which is reached at the outer edges, can be determined by the rotational speed N^* via

$$u_{\mathrm{tip}} = \pi D_{\mathrm{stirrer}} N^*,$$

which gives $u_{\mathrm{tip}} = 0.06283185\,{}^{m}/_{s}$ for Re = 1,000, $u_{\mathrm{tip}} = 0.6283185\,{}^{m}/_{s}$ for Re = 10,000, and $u_{\mathrm{tip}} = 1.8849556\,{}^{m}/_{s}$ for Re = 30,000. The time scaling can be calculated via

$$\frac{u_{\mathrm{tip}}^{\mathrm{sim}}}{u_{\mathrm{tip}}} = \frac{D_{\mathrm{tank}}^{\mathrm{sim}}}{D_{\mathrm{tank}}} \frac{t}{t^{\mathrm{sim}}} \quad \rightarrow \quad t = \frac{D_{\mathrm{tank}}}{D_{\mathrm{tank}}^{\mathrm{sim}}} \frac{u_{\mathrm{tip}}^{\mathrm{sim}}}{u_{\mathrm{tip}}} t^{\mathrm{sim}}.$$

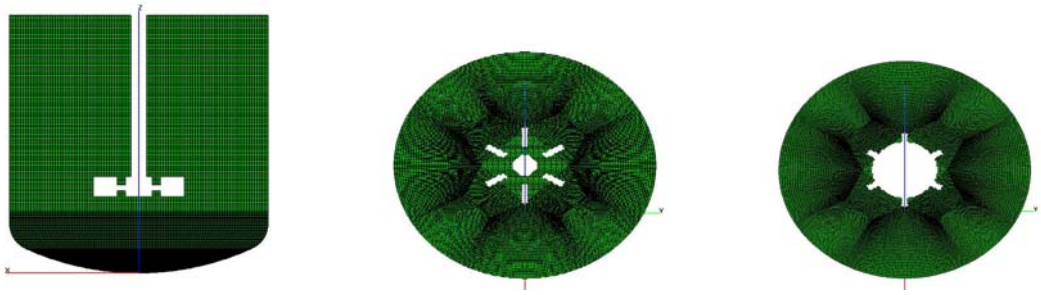


Figure 4.9: "Exact" description of the stirrer (grid of level 6). Left: longitudinal section, middle: cross section through the blades, above / below the disk, right: cross section through the disk.

This formula states that the simulation time has to be scaled by a factor of 0.3978876 for Re = 1,000, a factor of 0.03978876 for Re = 10,000, and a factor of 0.0132629 for Re = 30,000 in order to get the "real" or "experimental" time t corresponding to u_{tip} and D_{tank}.

4.3 Simulation Results

In this section, we will show the results of the flow simulation with FeatFlow. We consider a stirred tank reactor equipped with a six-bladed Rushton turbine and four baffles as already described in Section 4.2. The blades of the stirrer are 1 mm thick, 12 mm wide, and 10 mm high, the baffles are 1 mm thick, 13 mm wide, and 100 mm high (see also the exact mathematical description in Subsection 4.2.2 on page 65). Fig. 4.9 shows a longitudinal section and two cross sections through the tank, which illustrate the shape of the stirrer.

The simulations were done for three different Reynolds numbers: Re = 1,000, Re = 10,000, and Re = 30,000. As already mentioned in the beginning of this chapter, 3D flow simulations for such high Reynolds numbers require an extremely fine grid or some kind of turbulence modeling. However, the following results were produced without any turbulence model, just by applying direct numerical simulation.

In the following two subsections, we will show results of the simulation for two different refinement levels (level 5 and level 6). In Subsection 4.3.3, we will discuss why it is currently not possible to do simulations on refinement level 7 (corresponding to 22,475,457 nodes and 22,282,240 cells). At the end of this section, we will study simulation times.

4.3.1 Simulation Results for Level 5

In this subsection, we will present the results of the flow simulation using a grid of level 5. As already explained before, level i denotes the $(i-1)$th regular refinement of the coarse

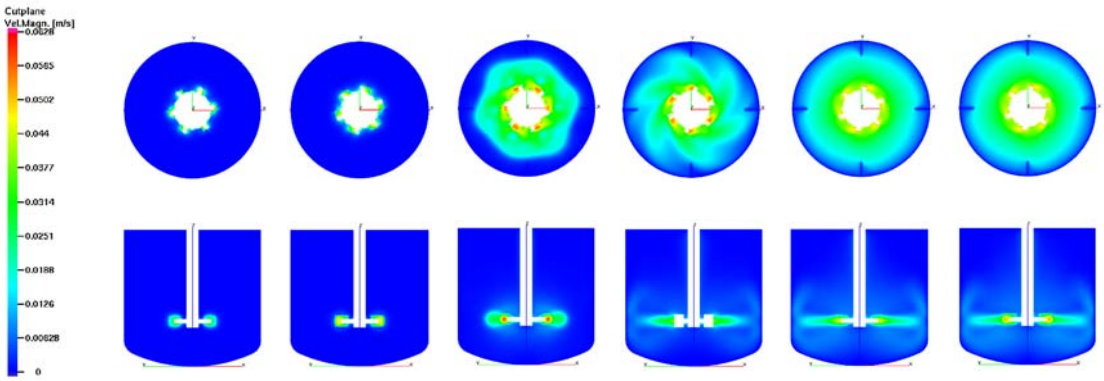


Figure 4.10: Grid of level 5, flow field for Re = 1,000. Velocity magnitude at different time points ($t_1 = 0.06\,\mathrm{s}$, $t_2 = 0.15\,\mathrm{s}$, $t_3 = 3.66\,\mathrm{s}$, $t_4 = 16.79\,\mathrm{s}$, $t_5 = 60.04\,\mathrm{s}$, $t_6 = 99.31\,\mathrm{s}$). Top: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane), bottom: longitudinal section between two baffles.

grid, which is depicted in Fig 4.4 on page 58.
We will show simulation results for three different Reynolds numbers, starting with Re = 1,000, followed by Re = 10,000, and finally Re = 30,000. For Re = 30,000, we will also show simulation results that have been obtained by using an "adapted" grid, i. e., a grid that was refined in the impeller region, where the highest velocities and velocity gradients occur.
The presentation of the simulation results for Re = 1,000 includes a more detailed description of the flow field (so that the reader gets an idea how the typical flow field induced by a Rushton turbine looks like), whereas for Re = 10,000 and Re = 30,000, we will only present a few results and mainly focus on comparing them with the previously presented results.

Simulation results for $\mathbf{Re = 1{,}000}$

In the following, we will show simulation results for Re = 1,000 corresponding to a rotational speed of about 24 rounds per minute (i. e., $N^* = 0.4\ {}^1/_\mathrm{s}$). In view of this, as boundary condition, we used a velocity of $u_\mathrm{tip} = 0.06283185\,{}^\mathrm{m}/_\mathrm{s}$ at the tip of the blades. Furthermore, Dirichlet-zero boundary conditions were prescribed on the whole boundary of the tank as well as on the four baffles.

Fig. 4.10 shows the velocity magnitude at different time points in a cross and a longitudinal section, respectively. Here, one can see how the flow field evolves over time. In the beginning, everywhere except directly at the impeller a zero-velocity is prescribed. However, after some time, the movement of the stirrer leads to a movement of the fluid in its vicinity. At the same time, the rotation of the fluid with the impeller leads to flow

in radial direction, which can be seen in the longitudinal sections in Fig. 4.10. The fluid is moved outwards in radial direction, then splits up, when the boundary of the tank is reached, and flows back building two circles, one above and one below the impeller mid-plane. The last two pictures in Fig. 4.10 show that, after some time, the shape of the flow field does not change any more.

Remark 4.11. *It is needless to say that the flow field still changes with time, since, on one hand, the impeller in the middle is turning around all the time and, on the other hand, the turbulence of the flow leads to permanently changing (random) velocity fluctuations. However, after some time, the structure of the flow field does not change any more. This means that if the fluctuations caused by the periodic passage of the impeller blades (so-called "pseudo-turbulence", see [VBS76]) as well as the "real" turbulent fluctuations of the velocity are subtracted from the velocity, the remaining quantity (the mean velocity) does not change any more. Then, the flow field is "fully developed" or "quasi-steady", see [YUNK02]. In [YUNK02] it is stated that for a Reynolds number of* $\mathrm{Re} = 4{,}500$ *twelve revolutions are sufficient to get a quasi-steady flow field throughout the whole tank.*

An example how the typical (fully developed) flow field induced by a Rushton turbine looks like is given by Fig. 4.11. This figure shows three vector-plots of the flow field in the stirred tank after $t = 99.31\,\mathrm{s}$ (corresponding to about 40 revolutions). In the left picture, a longitudinal section through the tank in the middle between two baffles is shown. Here, one can see the typical (so-called "kidney-shaped") flow field induced by a Rushton turbine. The liquid is drawn into the stirrer element axially, deflected in radial direction, and discharged in form of a radial jet stream. At the walls of the tank or at the baffles, respectively, the radial jet stream splits up and is diverted up- and downwards. Two large-scale ring vortices are formed, one above and one below the impeller mid-plane. Since the stirrer element is mounted on $1/3$ of the liquid height, the upper ring vortex only fills a part of the upper vessel area, whereas the lower ring vortex spans the entire lower vessel volume.
The bottom picture on the right-hand side shows a cross section through a quarter of the tank at height $x_3 = 5\,\mathrm{cm}$ (i. e., in the impeller mid-plane). Here, one can see how the velocity magnitude decreases from its highest values directly behind the blades of the impeller to zero-values at the wall and at the baffles. Clearly, the highest velocities throughout the tank can be found in the region around the stirrer element.
In the top picture, a cross section through the tank at height $x_3 = 10\,\mathrm{cm} = 2/3\,H_{\mathrm{tank}}$ is shown. At this height, the structure of the flow is only influenced by the four baffles. The number of blades the impeller consists of does not seem to affect the flow in this region of the tank.
Note that, in all three pictures, the velocity magnitude has been scaled by a factor of two so that the regions of smaller velocities can be seen better.

Remark 4.12. *The structure of the flow field in a baffled stirred tank operated by a Rushton turbine is well-known since decades. It has been investigated experimentally as well as numerically by a multitude of authors.*

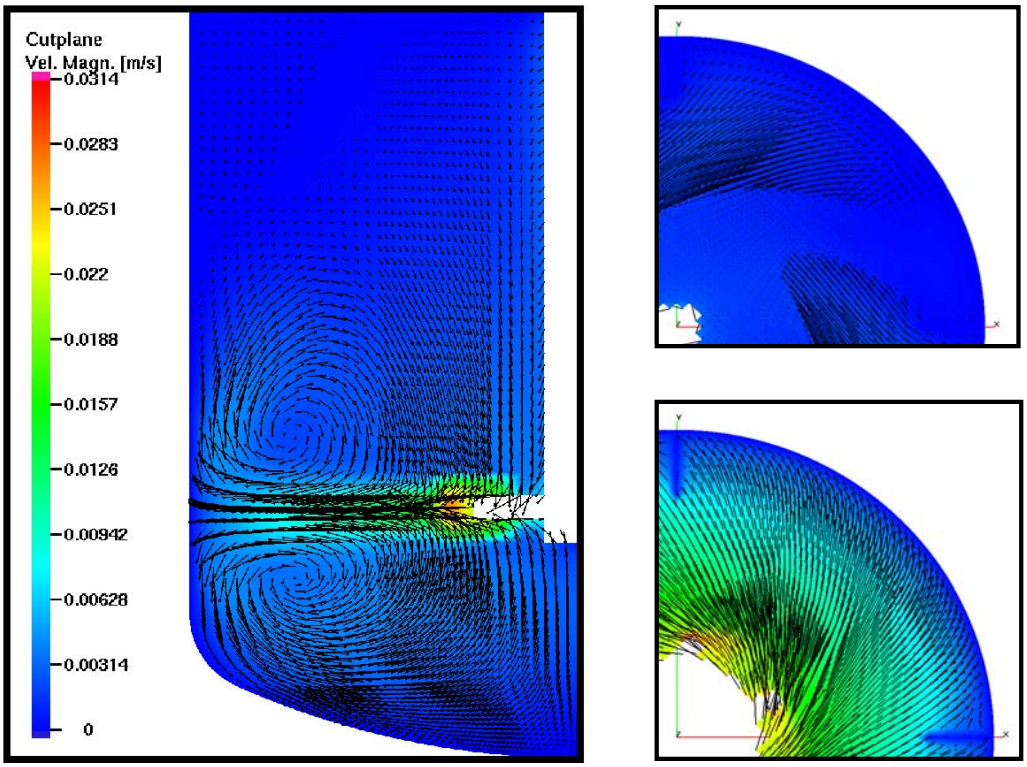


Figure 4.11: Grid of level 5, flow field for Re = 1,000. Vector-plot of the velocity magnitude after $t = 99.31\,\mathrm{s}$ (about 40 revolutions). Left: longitudinal section between two baffles, top right: cross section at height $x_3 = 10\,\mathrm{cm}$, top bottom: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane).

Experimental investigations for a Rushton turbine with diameter $D_{stirrer} = 1/3 D_{tank}$ were performed in, e. g., [VV78], [YPW87], [CC88], [WP89], [RJ90b], [DKJN93], [YW93], [SC95], and [SHD97]. A detailed review of the earlier works can be found in [RJ90b], whereas the newer literature has been summarized in [BCGM98].

Numerical simulations of the flow in a baffled stirred tank have been carried out basically by three different approaches.
In the first one, the impeller is not resolved by the computational mesh, but experimental data are used to specify boundary conditions for the impeller. This approach has been used in, e. g., [MPL86], [RJ90a], and [GIL+92]. Alternatively, in [PP87] source terms have been introduced in the momentum equation describing the drag of the impeller in the entire region swept by the impeller.
The second approach includes the so-called "multiple frames of reference" method, which was introduced in [LIG94], as well as the snapshot approach (see [RV94], [RD96]) and the so-called "inner-outer" approach by [BCGM94]. For a description of the methods

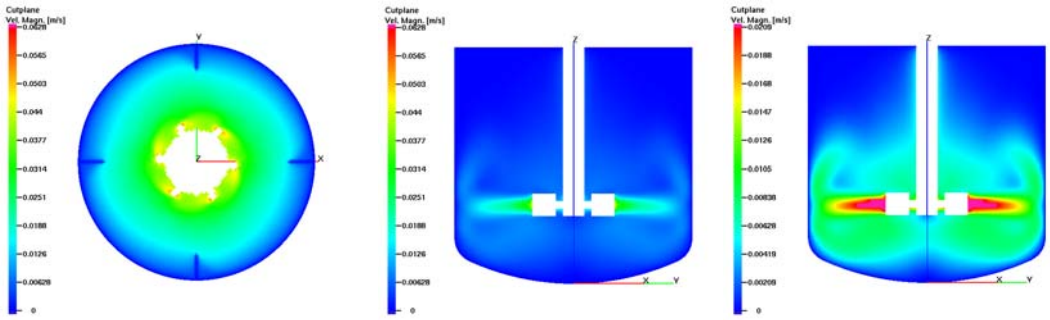


Figure 4.12: Grid of level 5, flow field for Re = 1,000. Velocity magnitude at time $t = 99.55\,\mathrm{s}$ (about 40 revolutions). Left: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane), middle: longitudinal section between two baffles, right: longitudinal section between two baffles, scaled plot.

and the differences between them, see Remark 4.10 and references therein.
Numerical simulations with the third approach, which is known as "sliding grid" or "sliding mesh" approach, were first done in [LGI+93], [PM93], and [MMC94]. More recently, the clicking and sliding mesh approaches (implemented in the parallel multigrid flow solver FeatFlow*) have been applied to different stirrer configurations, see, e. g., [BWS98], [SS01], or [SS98].*
Especially in the newer literature, one can also find other approaches for simulating the flow field in a stirred tank reactor. In [DV99], for example, large eddy simulations (LES) were used in combination with a force-field technique.
A detailed review of the different methods for modeling the impeller in a baffled stirred tank is given in [BCGM98]. Further information about the different approaches can also be found in Remark 4.10 and references cited there.

In the following, we will compare the just presented flow field for Re = 1,000 with those for Re = 10,000 and Re = 30,000. As already mentioned before, we will focus on a few simulation results, namely on the velocity magnitude in a cross section at height $x_3 = 5\,\mathrm{cm}$ (i. e., in the impeller mid-plane) and in a longitudinal section between two baffles. For the latter one, we will also show a scaled plot in order to give an idea of the flow field in the less turbulent regions of the tank. For Re = 1,000, these results are shown in Fig. 4.12.

Simulation results for $\mathbf{Re = 10{,}000}$

In the following, we will present simulation results for Re = 10,000, which corresponds to a rotational speed of about 240 rounds per minute. In view of this, a velocity of $u_{\mathrm{tip}} = 0.6283185\,\mathrm{m/s}$ was prescribed at the tip of the blades.

Fig. 4.13 shows the velocity magnitude in a cross and a longitudinal section through the tank after 10 seconds (corresponding to 40 revolutions). Again, as in the previously pre-

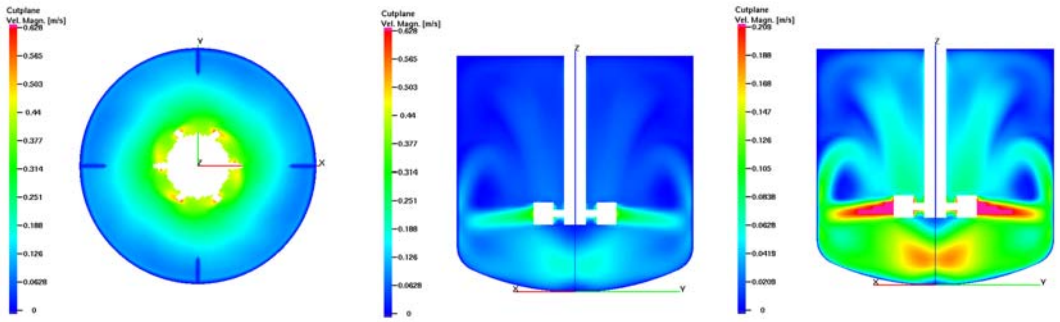


Figure 4.13: Grid of level 5, flow field for Re = 10,000. Velocity magnitude at time $t = 10.00\,\mathrm{s}$ (about 40 revolutions). Left: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane), middle: longitudinal section between two baffles, right: longitudinal section between two baffles, scaled plot.

sented results for Re = 1,000, one can see that the highest velocities occur in the region around the blades and that the velocity magnitude decreases with increasing distance from the impeller. In addition, one can again recognize the typical "kidney-shaped" flow field with a radial jet stream at the height of the impeller and two recirculation zones above and below.

However, if we compare the flow field for Re = 10,000 with the previously shown results for Re = 1,000, three major differences can be detected at once. The first one can be seen in the cross section through the impeller mid-plane. The value of the velocity magnitude, which decreases in both cases with increasing distance from the impeller in the middle, decreases slower for Re = 1,000 than for Re = 10,000. Second, for Re = 10,000, there are relatively high values of the velocity magnitude in the region below the stirrer (see middle and right picture in Fig. 4.13). In the corresponding pictures in Fig. 4.12, this phenomenon cannot be detected. A third difference between the flow fields for Re = 1,000 and Re = 10,000 can be seen in the right picture of Fig. 4.12 and Fig. 4.13, respectively. For Re = 10,000, the upper recirculation loop is more pronounced than for Re = 1,000. In addition, there seems to be more "movement" in the upper part of the tank. Furthermore, for Re = 1,000 the direction of the jet stream is horizontal, whereas it is downwards-sloping for Re = 10,000.

A comparison of the results presented above with corresponding results from the literature shows that in some regions of the tank the flow field is not calculated correctly. An example of this is the region of high velocities below the stirrer and the direction of the flow field in this area. However, a critical and detailed comparison with experimental data from the literature will be presented after some more simulation results have been shown.

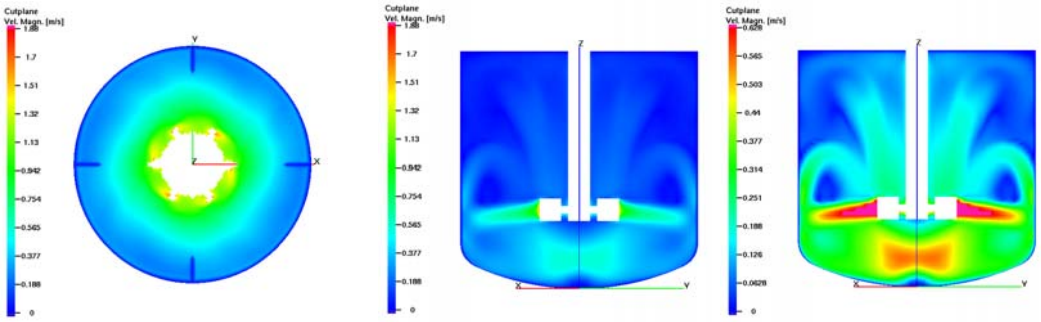


Figure 4.14: Grid of level 5, flow field for Re = 30,000. Velocity magnitude at time $t = 3.32\,\mathrm{s}$ (about 40 revolutions). Left: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane), middle: longitudinal section between two baffles, right: longitudinal section between two baffles, scaled plot.

Simulation results for Re = 30,000

In the following, we will present simulation results for Re = 30,000, which corresponds to a rotational speed of about 720 rounds per minute. In view of this, $u_{\mathrm{tip}} = 1.8849556\,{}^{\mathrm{m}}/_{\mathrm{s}}$ was used as boundary condition.

Fig. 4.14 shows the flow field in a cross and a longitudinal section through the tank after 3.32 seconds, i. e., after about 40 revolutions. First, one can say that the results look quite similar to those obtained for Re = 10,000. This finding is in accordance to the literature, where it is stated that the flow field scaled by u_{tip}, which denotes the (maximum) prescribed velocity at the tip of the blades, is the same for all Reynolds numbers in the so-called "fully turbulent" flow regime. In [PAOK04], the transition between laminar and turbulent regimes is said to occur in a range of Reynolds numbers between Re = 50 and Re = 5,000. This is in accordance to the presented results, which show the same structure of the flow field for Re = 10,000 and Re = 30,000 and a slightly different one for Re = 1,000.

However, as already mentioned before, the results are not fully satisfactory. Therefore, it was decided to refine the grid in the region around the impeller (the region where the highest velocities occur) in order to improve the results. In order to do this without increasing the computational costs, we refined the grid in this area, but, in return, coarsened the grid in the bulk region of the tank, see Fig. 4.15. Note that the refinement / coarsening of the mesh was only done in vertical direction and in such a way that the total number of nodes stayed (exactly) the same.

The resulting flow field for Re = 30,000 at time $t = 4.33\,\mathrm{s}$, which corresponds to about 52 revolutions, is shown in Fig. 4.16. As before, the left picture in Fig. 4.16 shows a cross section of the velocity magnitude in the impeller mid-plane, whereas the middle and the right picture show a longitudinal section between two baffles.
Compared to the simulation results shown before, one can immediately see that, for this

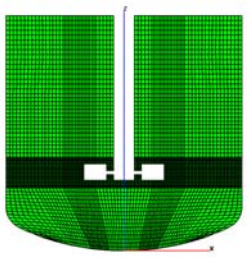


Figure 4.15: “Adapted” grid of level 5.

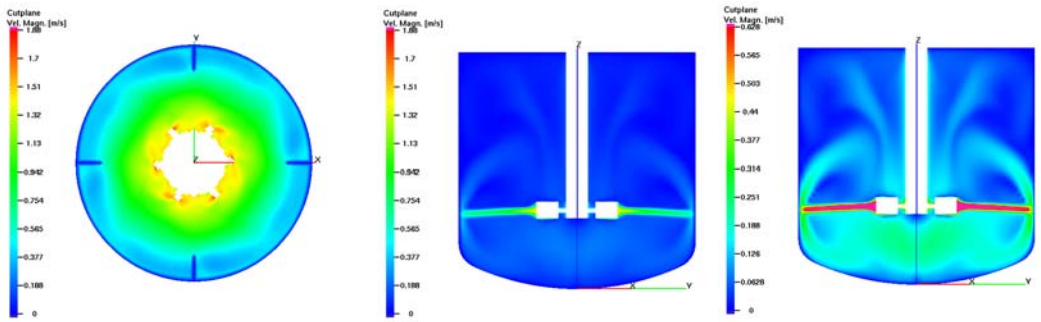


Figure 4.16: “Adapted” grid of level 5, flow field for Re = 30,000. Velocity magnitude at time $t = 4.33\,\mathrm{s}$ (about 52 revolutions). Left: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane), middle: longitudinal section between two baffles, right: longitudinal section between two baffles, scaled plot.

simulation, the jet stream is much thinner. This unphysical behavior may be caused by the local refinement of the grid and especially by the steep transition from very fine grid cells to coarser ones. The flow field seems to be geared to the grid. These problems are similar to those observed in [Str04], where the flow over a backward facing step was simulated with FeatFlow. There, it was observed that the asymmetry of the grid (caused by the local refinement of an orthogonal grid) can lead to an asymmetry in the resulting flow field.

In order to ensure that the flow field is really fully developed and that its shape does not change with time any more, a “long-term” simulation was started. The resulting flow field after 32.61 seconds (i. e., about 391 revolutions) is not presented here, since the results really show that the structure of the flow stays the same.

4.3.2 Simulation Results for Level 6

Since the simulations on level 5 did not show fully satisfactory results, it was decided to refine the grid. Thus, we will in the following consider the results of simulations with a grid of level 6. However, the graphical output was created for level 5 only, because

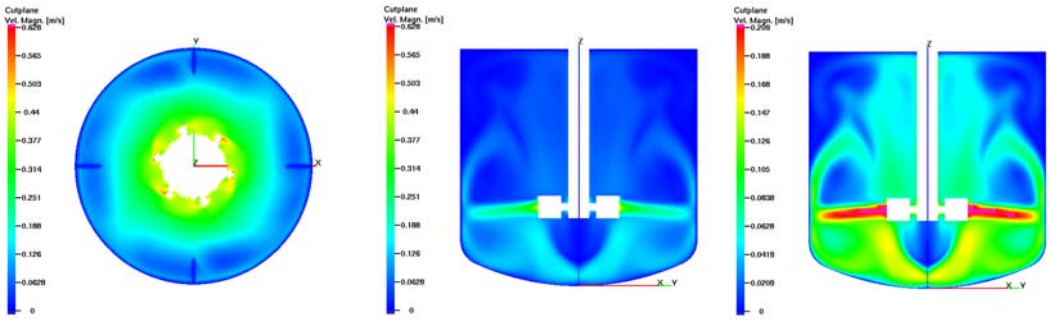


Figure 4.17: Grid of level 6, flow field for Re = 10,000. Velocity magnitude at time $t = 12.99\,\mathrm{s}$ (about 52 revolutions). Left: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane), middle: longitudinal section between two baffles, right: longitudinal section between two baffles, scaled plot.

otherwise we could not save more than a few output files due to the huge size of the `gmv`-files (see Subsection 4.2.3).

Since in the considered practical application the stirred tank is operated with a rotational speed of about 400 to 700 rounds per minute, which corresponds to Reynolds numbers of at least $\mathrm{Re} \approx 10^4$, only results for Reynolds numbers of this order of magnitude will be presented in the following. Another reason for the restriction to a few cases only is that the simulation times are really immense for the grids of level 6, see Subsection 4.3.4.

Simulation results for $\mathrm{Re} = 10{,}000$

The simulation results for Reynolds number Re = 10,000 at time $t = 12.99\,\mathrm{s}$ (corresponding to about 52 revolutions) are shown in Fig. 4.17.

If we compare these results to the results obtained by the corresponding simulation on a grid of level 5 (see Fig. 4.13), we first recognize that the region of high velocities below the stirrer is not so pronounced any more. The flow field in this area seems to be more sensible for the grid of level 6. However, there are relatively high gradients of the velocities in this area, which still are not really realistic.

Simulation results for $\mathrm{Re} = 30{,}000$

In the following, we will present the results for Reynolds number Re = 30,000. Fig. 4.18 shows a cross and a longitudinal section of the velocity magnitude at time $t = 4.33\,\mathrm{s}$, which corresponds to about 52 revolutions.

Again, as for the simulations on the grid of level 5, these results look almost identical to those for Reynolds number Re = 10,000. Only in the scaled plot in the right picture,

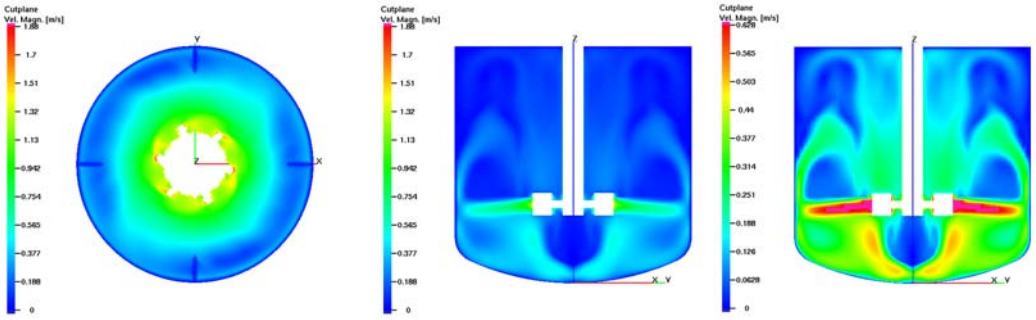


Figure 4.18: Grid of level 6, flow field for Re = 30,000. Velocity magnitude at time $t = 4.33\,\mathrm{s}$ (about 52 revolutions). Left: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane), middle: longitudinal section between two baffles, right: longitudinal section between two baffles, scaled plot.

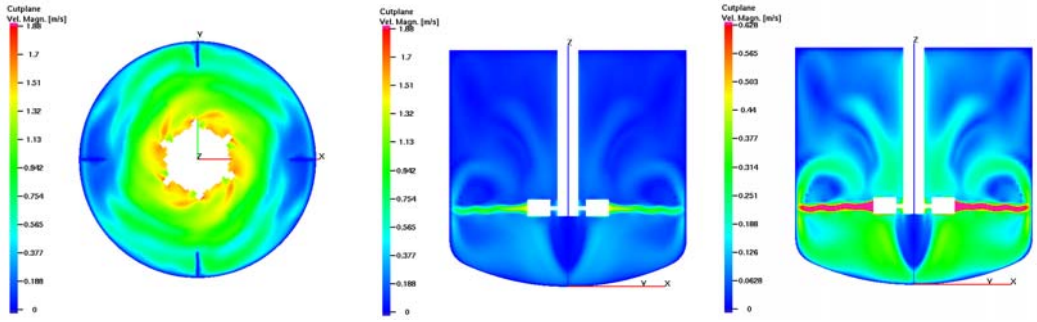


Figure 4.19: "Adapted" grid of level 6, flow field for Re = 30,000. Velocity magnitude at time $t = 3.33\,\mathrm{s}$ (about 40 revolutions). Left: cross section at height $x_3 = 5\,\mathrm{cm}$ (impeller mid-plane), middle: longitudinal section between two baffles, right: longitudinal section between two baffles, scaled plot.

some smaller differences can be detected, which can be explained by a slightly different position of the longitudinal section within the tank.

As before, for the simulations with the grid of level 5, we also tried to improve the results by using an "adapted" grid (the same grid as the one used on level 5, see Fig. 4.15 on page 77, but refined once more).

Fig. 4.19 shows the resulting flow field after $t = 3.33\,\mathrm{s}$, i. e., after about 40 revolutions. Here, one can immediately see the differences to the flow field we obtained by using the "ad-hoc" grid. With some good will, one can say that the flow field looks turbulent (at least in the cross section). Another difference to the results presented before is the "wiggling" of the jet stream, which can be seen in the middle and the right picture of Fig. 4.19. Again, as for the simulation with the "adapted" grid of level 5, the jet stream is thinner than for the "ad-hoc" grid, which seems to be caused by the only very local

refinement of the grid. Another difference to the simulation results obtained by the "ad-hoc" grid is that the region of low velocities below the stirrer is smaller, which seems to be more reasonable. On the other hand, there are still very high velocity gradients at the "boundary" of this region. However, the maximum values of the velocity magnitude in the region below the stirrer are lower, which also seems to be more reasonable.

Altogether, the results presented so far show that, in the whole, the "mainstream" of the flow field is reproduced well by the simulation. However, if we have a more detailed look at the results, we can find several regions, where the flow field is not calculated correctly (as already described above).
First of all, there is the region below the impeller, where the velocity is almost zero. This wedge and even more the high gradients of the velocity at its boundary are in contradiction to the typical shape of the recirculation loops in this area. The flow is expected to be sucked in vertically by the impeller.
Second, the velocities in the bottom part of the tank (in the lower recirculation loop) are altogether much higher than those in the upper recirculation loop, which is also not in accordance to experimental findings. However, it should be mentioned that the experimental investigations were made for a "real" cylindric tank, i. e., for a tank with a flat bottom.
The cross section in Fig. 4.19 shows an irregular distribution of the velocity in the outer regions, which seems to be unphysical at first sight. However, if we compare this cross section with a cross section at a time point a little bit later, when the blades of the impeller have another position with respect to the baffles, the relatively high velocities at two of the baffles in Fig. 4.19 have "moved" to another position. Thus, this phenomenon seems to be really induced by the interaction of the blades of the impeller with the baffles.

Altogether, the results obtained with the "adapted" grid of level 6 seem to be the best. This is one reason why we will use these results for a deeper analysis and comparison with experimental data from the literature. Another reason is that the results for $\mathrm{Re} = 30{,}000$ fit best to the considered application. (Experimental investigations of the drop size distribution in the stirred tank were done for Reynolds numbers within this range.)

Fig. 4.20 shows cross sections at four different heights. In the left picture of each row, the velocity magnitude is plotted, whereas the middle and the right picture show the square-root of the sum of the squared radial plus the squared tangential velocity and the vertical velocity, respectively. Note that some of the plots are scaled, because of the different orders of magnitude of the velocity components.
These pictures show that the "turbulent structure" of the flow field, which could be observed in the cross section in Fig. 4.19, can also be seen at other heights in the tank. Furthermore, the irregular structure appears in all the different components of the velocity.

The first row in Fig. 4.20 shows a cross section through the impeller mid-plane. Here, one can see that, at this height, the main streaming of the flow is in radial and tangential direction. The axial component of the velocity is negligible compared to the other two

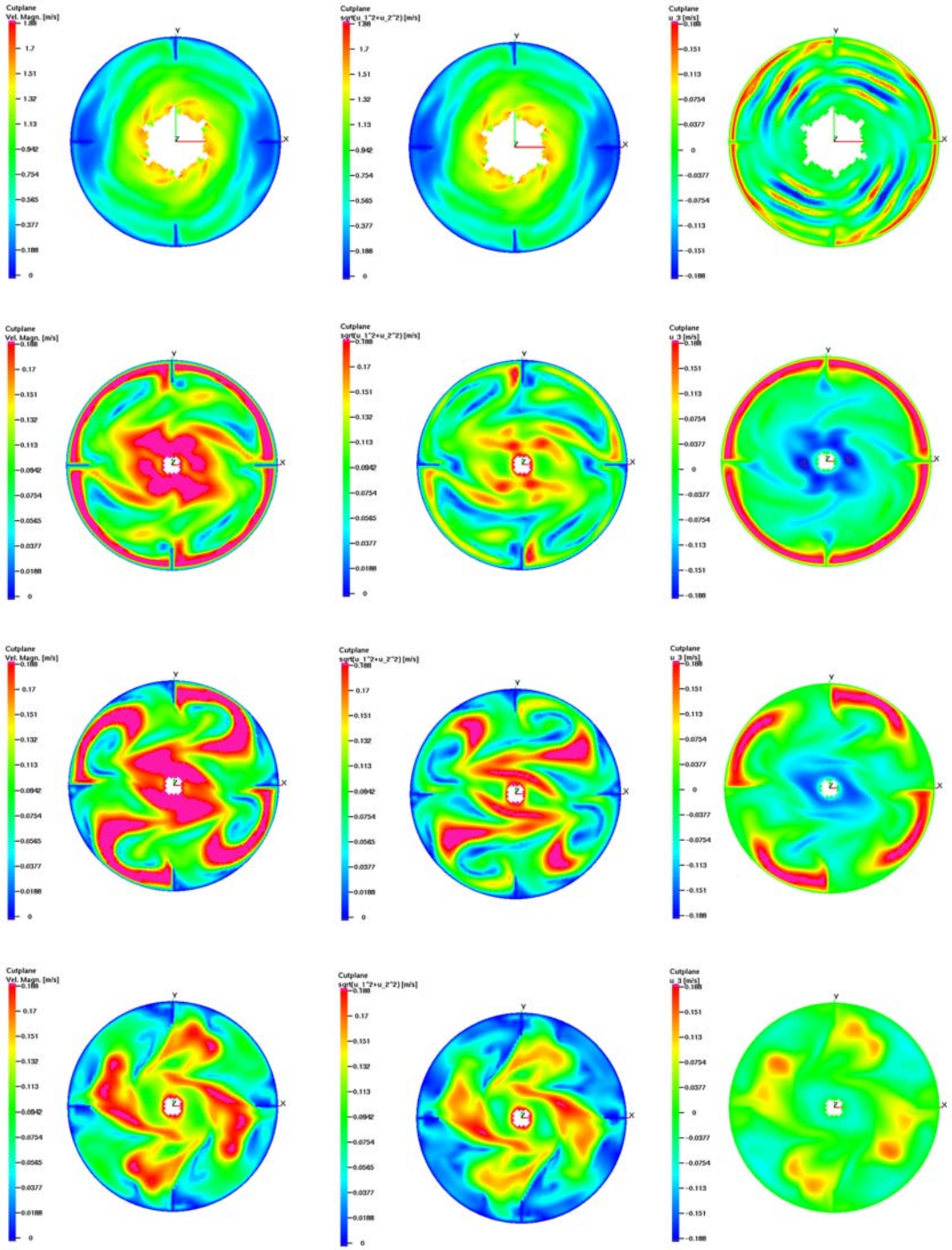


Figure 4.20: "Adapted" grid of level 6, flow field for Re = 30,000. Cross sections at four different heights (first row: $x_3 = 5\,\text{cm}$ (impeller mid-plane), second row: $x_3 = 6\,\text{cm}$, third row: $x_3 = 7.5\,\text{cm}$, fourth row: $x_3 = 10\,\text{cm}$) at time $t = 3.33\,\text{s}$ (about 40 revolutions). Left: velocity magnitude, middle: square-root of sum of squared radial plus squared tangential velocity, right: vertical velocity. (All plots except the left and the middle one in the first row are scaled.)

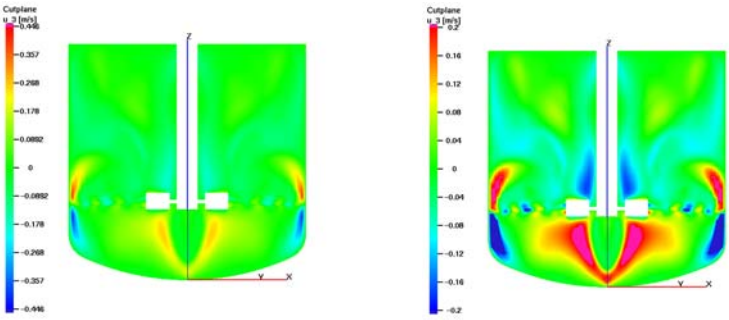


Figure 4.21: "Adapted" grid of level 6, flow field for Re = 30,000. Vertical velocity (normal and scaled value), longitudinal section between two baffles at time $t = 3.33\,\mathrm{s}$ (about 40 revolutions).

components.
In the second row, a cross section one centimeter above the impeller mid-plane is shown. Here, one can see that the radial and tangential components of the velocity are relatively random, whereas the vertical flow is directed upwards near the boundary of the tank and downwards in the middle.
The upwards and downwards flow in vertical direction can also be seen in the cross section at height $x_3 = 7.5\,\mathrm{cm}$, i. e., $2.5\,\mathrm{cm}$ above the impeller mid-plane. However, at this height, the upwards streaming is restricted to a region behind the baffles. Directly in front of the baffles, there is no significant vertical flow. This suggests the assumption that the upper recirculation loop is larger in the region behind the baffles than in the region in front of them.
In the bottom pictures, a cross section at height $x_3 = 10\,\mathrm{cm}$ (i. e., in the middle between the impeller mid-plane and the top of the tank) is shown. Here, the flow field mainly consists of radial and tangential components. However, there is still a small vertical streaming up- and downwards.

As next step, we will briefly describe the flow in vertical direction throughout the tank. Fig. 4.21 shows the vertical velocity in a longitudinal section between two baffles at time $t = 3.33\,\mathrm{s}$, which corresponds to about 40 revolutions. Here, one can see that the highest values of the vertical velocity component can be found in two regions, namely in the region near the wall (or the baffles, respectively) above and below the impeller-mid-plane and in the region in the middle below and above the stirrer. Both regions can easily be identified as parts of the two recirculation loops formed by the jet stream hitting the wall.

After giving a qualitative description of the flow field, we will now also quantitatively analyze the flow data by comparing axial and radial profiles of the three (mean) velocity components with corresponding data from the literature. For this comparison, we use experimental results obtained in [WP89], [CC88], [RJ90b], [VV78], [DKJN93], and [Hoc90]. Note that in [CC88] it is distinguished between profiles for a baffle plane and

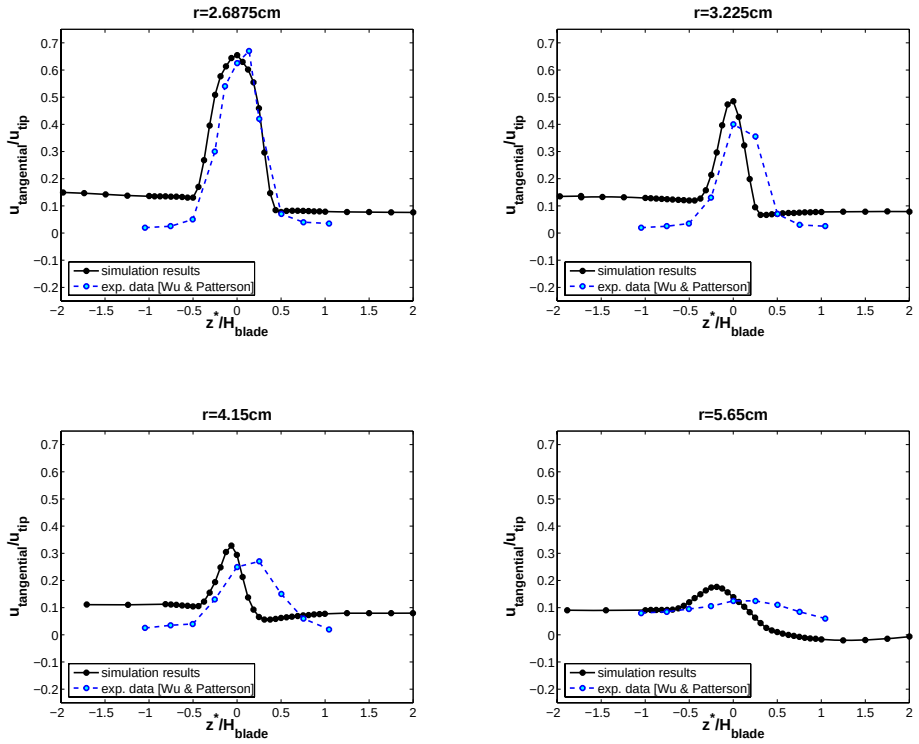


Figure 4.22: Axial profile of mean tangential velocity between two baffles at different radii (top left: $r = 2.6875\,\text{cm}$, top right: $r = 3.225\,\text{cm}$, bottom left: $r = 4.15\,\text{cm}$, bottom right: $r = 5.65\,\text{cm}$).

profiles for a plane between two baffles, and results for both cases are given. However, in the other papers it is not specified if the experimental data were obtained in a baffle plane or between two baffles, and thus, these results are used for both cases.
The radii in the presented axial profiles as well as the heights in the radial ones have been chosen so that they could be compared with as many experimental results from the literature as possible. Therefore, it was not possible to present exactly uniform profiles for all three velocity components. However, we have tried to choose the results in such a way that the presentation is as clear and as uniform as possible.

Fig. 4.22 shows axial profiles of the mean tangential velocity in a plane in the middle between two baffles for different radii. Note that these simulation results as well as all the following ones have been obtained by using an "adapted" grid of level 6. "Mean velocity" here means that the CFD results have been averaged over several time steps

$$u_{\text{mean}}(t_k) = \frac{1}{2j+1} \sum_{i=k-j}^{k+j} u(t_i), \tag{4.12}$$

where $u_{\text{mean}}(t_i)$ and $u(t_i)$ are the mean velocity and the velocity from the CFD simulation at time $t = t_i$, respectively. By this, the influence of the blades of the impeller (i. e., the exact position of the impeller) is "averaged out". These profiles are compared with experimental data from Wu & Patterson [WP89].
The top left picture in Fig. 4.22 shows the velocity profile at radius $r = 2.6875\,\text{cm}$, i. e., close to the impeller tip. Here one can see that the mean tangential velocity is reproduced well by the simulation. Only in the region below the impeller (the negative part of the axis of abscissae), the profiles do not match the experimental data from [WP89]. However, this difference may be caused by the fact that in [WP89] a cylindric, flat-bottomed tank was used, whereas the simulations were done for a tank with torispherical head.
The top right picture in Fig. 4.22 shows the velocity profile at $r = 3.225\,\text{cm}$, i. e., a little bit further away from the impeller tip. The simulation results still match the experimental data tolerably well. However, in the simulation, the region of high velocities is shifted a little bit downwards. This may also be explained by the different bottoms used for the simulation.
The two bottom pictures in Fig. 4.22 show the profiles at radii $r = 4.15\,\text{cm}$ and $r = 5.65\,\text{cm}$. Here one can see that the peak of the profile is shifted further downwards compared to the experimental data. Furthermore, the profiles in the simulation seem to be too steep.

Fig. 4.23 shows the same profiles in a baffle plane. Here one can see that the profiles are almost identical, especially in the first two pictures. This is not really surprising, since the radial positions in these pictures are close to the impeller and, thus, far away from the baffles. If we compare the two bottom pictures in Fig. 4.23 with the corresponding ones in Fig. 4.22, we recognize that the peaks are higher and steeper in the baffle plane. However, there are only small differences, since the radial positions in these pictures are still in significant distance from the radial position of the baffles.

The four pictures in Fig. 4.24 show axial profiles of the mean radial velocity at the same radial positions, i. e., for $r = 2.6875\,\text{cm}$, $r = 3.225\,\text{cm}$, $r = 4.15\,\text{cm}$, and $r = 5.65\,\text{cm}$. Again, the simulation results are compared with experimental data from Wu & Patterson [WP89]. This comparison shows that, in the simulation, the maximum value of the mean radial velocity is underestimated in the region near the impeller, whereas it is overestimated further away from it. While the peak values in the simulation are first slowly increasing and then slowly decreasing, the experimental data show a continuous decrease of the mean radial velocity with further distance from the impeller tip.
Since the corresponding results in a baffle plane are again similar, we will not show them.

Fig. 4.25 shows axial profiles of the mean axial velocity between two baffles at the same radial positions as before. Comparing the results with those in Wu & Patterson [WP89] shows that the simulation results are totally different from the experimental data. Neither the zero-crossing (i. e., the position where the flow changes its direction) nor the order of magnitude of the mean axial velocity are reproduced by the simulation. It may be possible that the vertical velocity is strongly influenced by the shape of the bottom. However, this does not explain these strong differences.

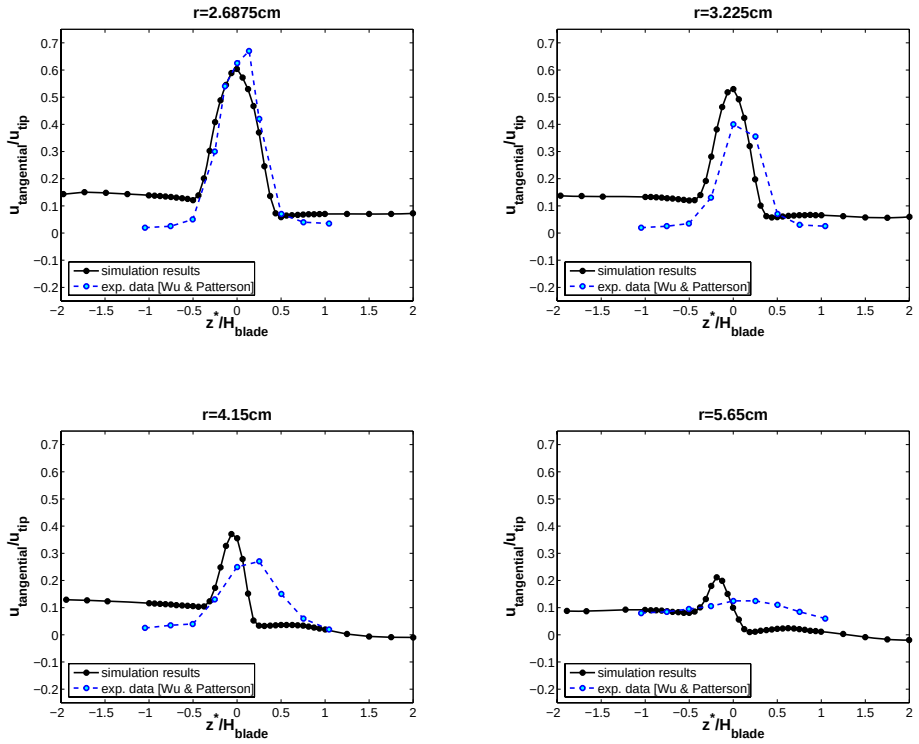


Figure 4.23: Axial profile of mean tangential velocity in baffle plane at different radii (top left: $r = 2.6875\,\text{cm}$, top right: $r = 3.225\,\text{cm}$, bottom left: $r = 4.15\,\text{cm}$, bottom right: $r = 5.65\,\text{cm}$).

Again, the corresponding profiles through a baffle differ only slightly from the presented results and are, therefore, not shown here.

Let us now have a look at the radial profiles. The top right picture in Fig. 4.26 shows a radial profile of the mean tangential velocity between two baffles at height $x_3 = 4.5\,\text{cm}$, i. e., at the bottom line of the impeller. Comparing the simulation results with experimental data obtained by Costes & Couderc [CC88] (note that the exact values have been read off from the diagrams presented in [JR99]), one can see that the simulation data match very well the experimental data.
The top right picture in Fig. 4.26 shows the same radial profile, but at height $x_3 = 5\,\text{cm}$, i. e., in the impeller mid-plane. In this case, the simulation results are not only compared with experimental data from Costes & Couderc [CC88], but also with those from Wu & Patterson [WP89]. The results match both sets of data well. (The differences between the two sets of experimental data are even larger than those between the simulation and the experimental results.)
The bottom two pictures in Fig. 4.26 show analogous profiles for $x_3 = 6.046\,\text{cm}$ and $x_3 = 6.5\,\text{cm}$, i. e., a little above the impeller. Comparing the results with experimental

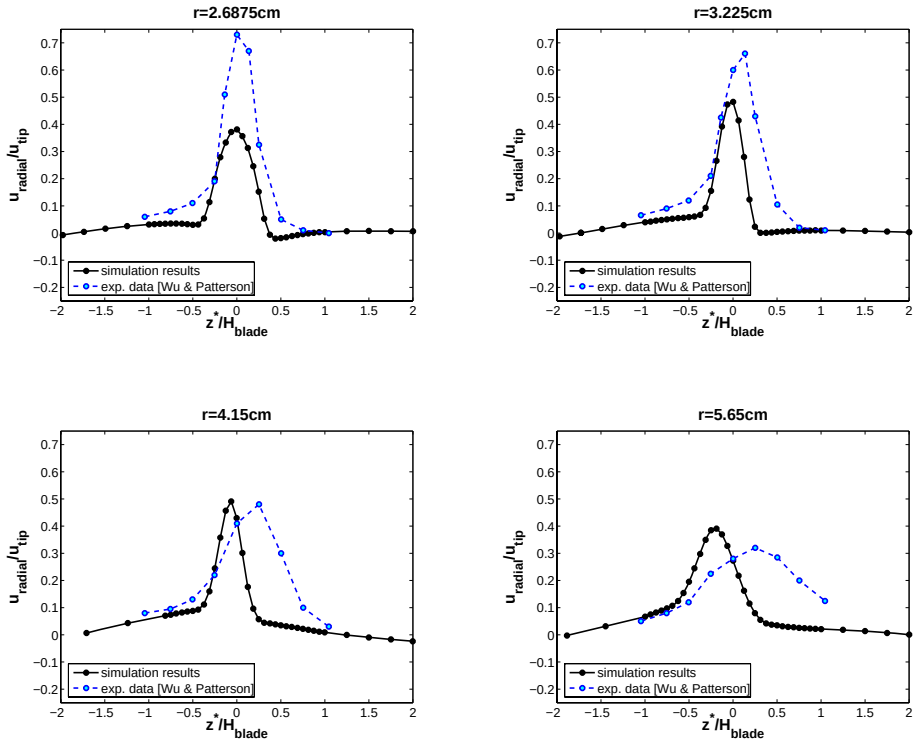


Figure 4.24: Axial profile of mean radial velocity between two baffles at different radii (top left: $r = 2.6875\,\text{cm}$, top right: $r = 3.225\,\text{cm}$, bottom left: $r = 4.15\,\text{cm}$, bottom right: $r = 5.65\,\text{cm}$).

data from Wu & Patterson [WP89] and from Ranade & Joshi [RJ90b], respectively, does not show an as good agreement as before. The lower velocities in this region are not reproduced satisfactorily by the simulation.

The corresponding profiles in a plane through a baffle are shown (for three of the four heights) in Fig. 4.27. As before, the profile at height $x_3 = 5\,\text{cm}$ matches quite well the experimental data, whereas the other two profiles are only in moderate accordance to the data from the literature.

Fig. 4.28 shows radial profiles of the mean radial velocity in a plane between two baffles. The top left picture in Fig. 4.28 shows results for $x_3 = 4.5\,\text{cm}$, which are compared with experimental data from Costes & Couderc [CC88]. (Again the values have been taken from [JR99].) There is only a moderate agreement between the simulation results and the experimental data. However, the order of magnitude of the velocity component is reproduced by the simulation.
The top right picture in Fig. 4.28 shows the corresponding profile for $x_3 = 5\,\text{cm}$. Since the flow field in the impeller mid-plane has been investigated by a multitude of authors,

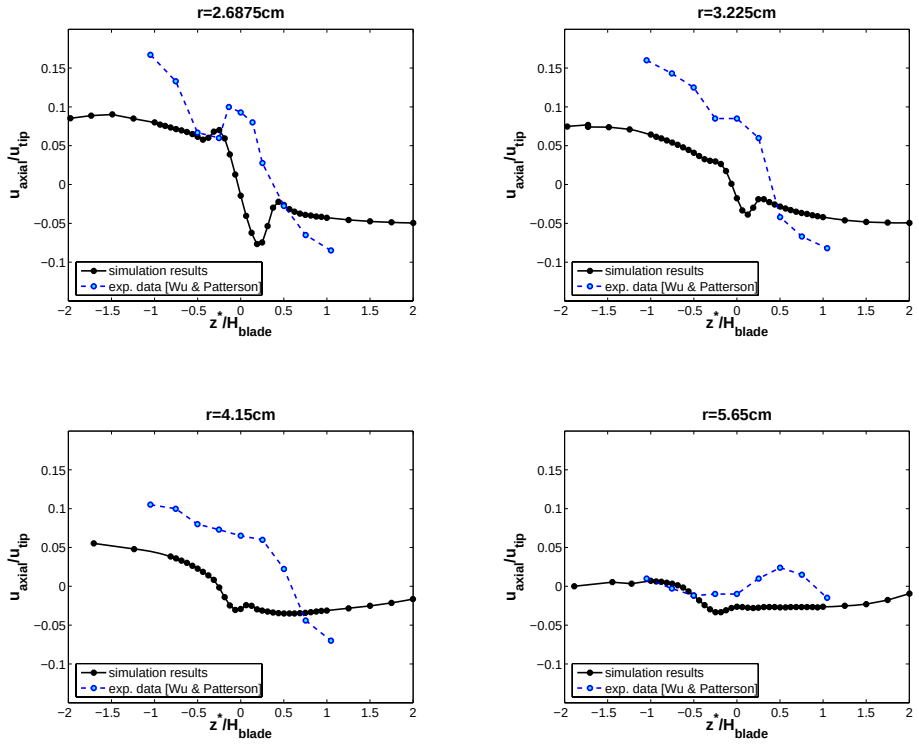


Figure 4.25: Axial profile of mean axial velocity between two baffles at different radii (top left: $r = 2.6875\,\text{cm}$, top right: $r = 3.225\,\text{cm}$, bottom left: $r = 4.15\,\text{cm}$, bottom right: $r = 5.65\,\text{cm}$).

we are able to compare the simulation results with experimental data from four different papers, namely the papers by Costes & Couderc [CC88] (read off from [JR99]), Wu & Patterson [WP89], Van der Molen & Van Maanen [VV78], and Dyster et al. [DKJN93]. The comparison shows that the simulation results agree quite well with the experimental data, except for the region near the impeller. A reason for this may be a poor specification of boundary values at the rotating impeller.

The bottom pictures in Fig. 4.28 show corresponding profiles for $x_3 = 6.046\,\text{cm}$ and $x_3 = 6.5\,\text{cm}$. While the simulation results for $x_3 = 6.046\,\text{cm}$ do not match the experimental data (neither those by Wu & Patterson [WP89] nor those by Hockey [Hoc90]), the results for $x_3 = 6.5\,\text{cm}$ are in perfect agreement with the experimental data from Ranade & Joshi [RJ90b]. Note that the experimental data in [Hoc90] are depicted here as they have been taken from [LSE00]. Furthermore, it should be mentioned that the depicted values from [Hoc90] are not really for $x_3 = 6.046\,\text{cm}$, but for $x_3 = 6.1225\,\text{cm}$. However, since the simulation results for $x_3 = 6.046\,\text{cm}$ and $x_3 = 6.1225\,\text{cm}$ are almost identical, the data from [Hoc90] can also be used for $x_3 = 6.046\,\text{cm}$.

As before in the presentation of the axial profiles, we will, also for the radial profiles,

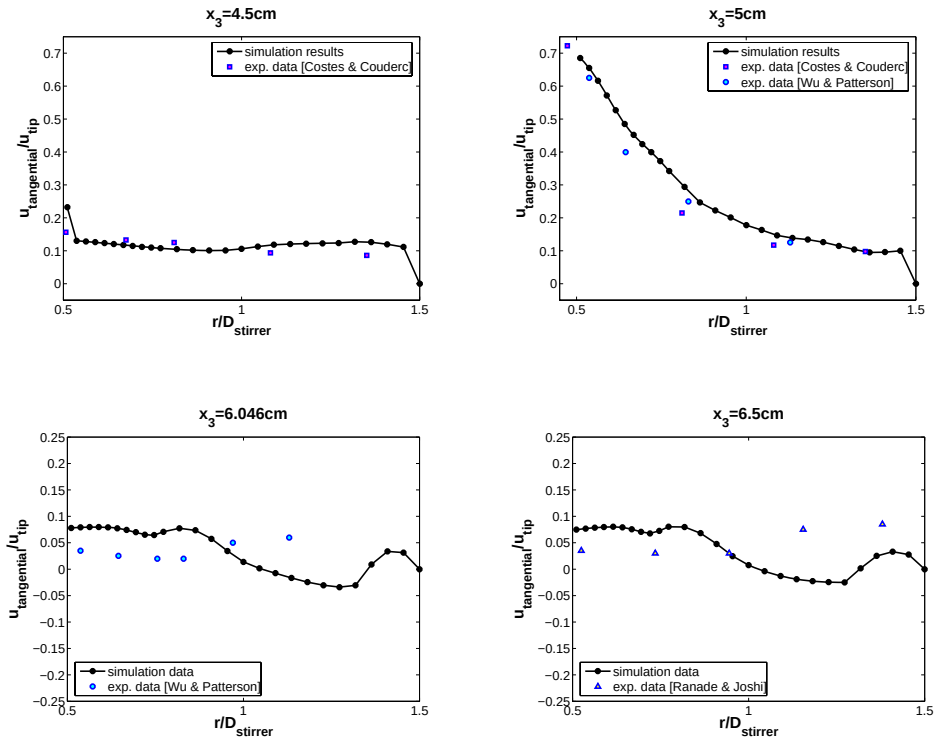


Figure 4.26: Radial profile of mean tangential velocity between two baffles at different heights (top left: $x_3 = 4.5\,\text{cm}$, top right: $x_3 = 5\,\text{cm}$, bottom left: $x_3 = 6.046\,\text{cm}$, bottom right: $x_3 = 6.5\,\text{cm}$).

skip the pictures showing the mean radial velocity in a plane though a baffle, since they do not give any new information.

Fig. 4.29 shows radial profiles of the mean axial velocity between two baffles at five different heights. The picture in the top shows simulation results for $x_3 = 5\,\text{cm}$ in comparison with corresponding experimental data from Wu & Patterson [WP89]. Here one can see that the simulation results are totally different from the data from the literature. While in [WP89] positive velocities in the region around the impeller have been measured, the simulation gives very small negative values.
The two pictures in the middle of Fig. 4.29 show corresponding profiles for $x_3 = 6.046\,\text{cm}$ and $x_3 = 6.5\,\text{cm}$. Comparing these results with experimental data from Wu & Patterson [WP89], Hockey [Hoc90] (again read off from [LSE00]), and Ranade & Joshi [RJ90b] shows that the general tendency is reproduced by the simulation. However, the magnitude of the values differs from the experimental data. Note that (as before) the experimental data in [Hoc90] is again not for $x_3 = 6.046\,\text{cm}$, but for $x_3 = 6.1225\,\text{cm}$.
The pictures in the bottom row of Fig. 4.29 show that the deviation from the experimental data gets larger with larger distance from the impeller mid-plane. Here, simulation

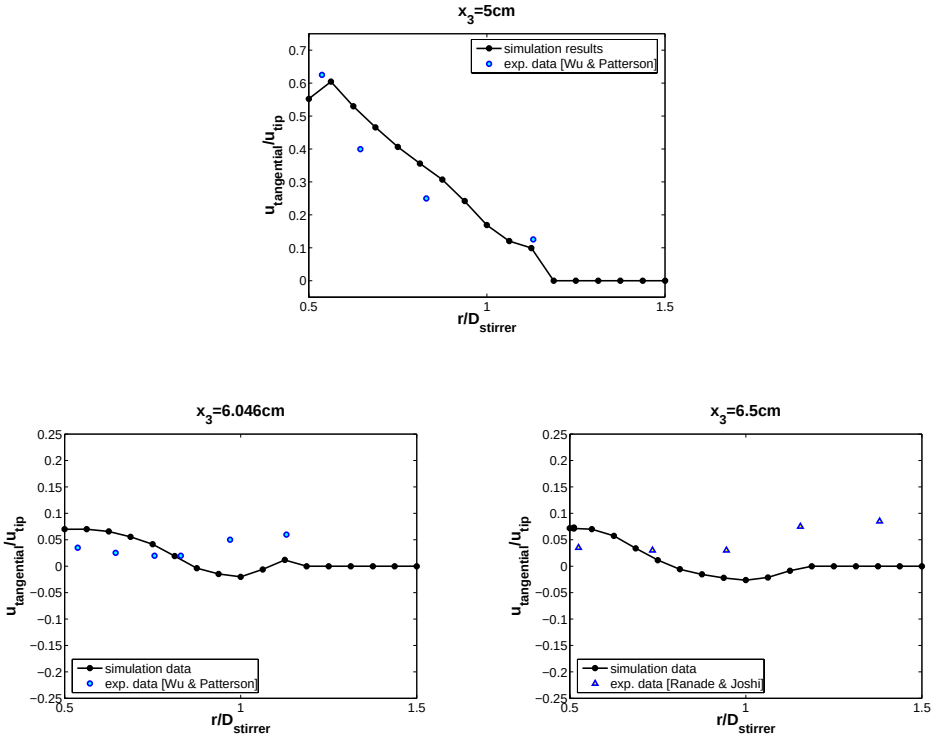


Figure 4.27: Radial profile of mean tangential velocity in baffle plane at different heights (top: $x_3 = 5\,\text{cm}$, bottom left: $x_3 = 6.046\,\text{cm}$, bottom right: $x_3 = 6.5\,\text{cm}$).

results at heights $x_3 = 7\,\text{cm}$ and $x_3 = 8.75\,\text{cm}$ have been compared with experimental data from Costes & Couderc [CC88] and Ranade & Joshi [RJ90b].

The profiles of the mean axial velocity in a baffle plane clearly differ from those between two baffles, since the highest axial velocities occur at the same radial positions, where the baffles are, i. e., in the region, where a zero-value is prescribed in the axis of the baffles. However, this is also the reason why profiles through a baffles plane are not very interesting, since the axial velocity is zero (as prescribed) in the most interesting regions. Thus, these profiles are not shown here.

At the end of this subsection, let us now summarize the results of the comparison between the simulated and the experimental velocity profiles.
First of all, one can say that for the tangential and radial velocity components a satisfactory agreement could be obtained. For these components, the general tendency of the velocity profiles from the literature was reproduced well by the numerical simulation. For some positions, even a very good quantitative agreement could be observed, like for the axial profile of the mean tangential velocity at $r = 2.6875\,\text{cm}$ (i. e., close to the tip of the impeller) or for the radial profile of the tangential velocity at heights $x_3 = 4.5\,\text{cm}$ and $x_3 = 5\,\text{cm}$ (i. e., at the bottom line of the impeller and in the impeller mid-plane).

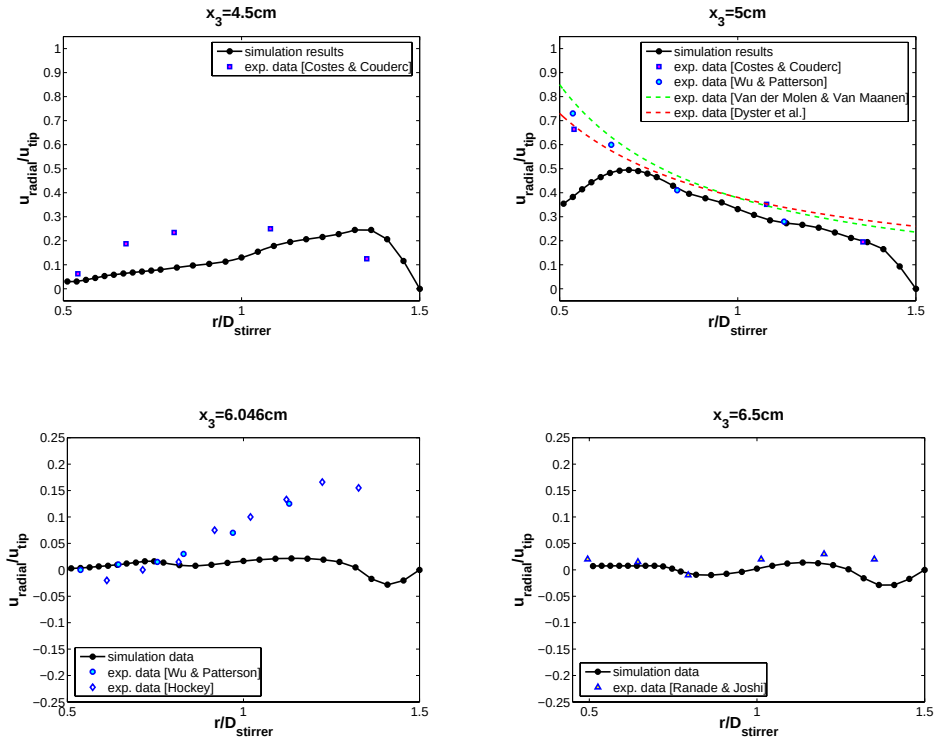


Figure 4.28: Radial profile of mean radial velocity between two baffles at different heights (top left: $x_3 = 4.5\,\text{cm}$, top right: $x_3 = 5\,\text{cm}$, bottom left: $x_3 = 6.046\,\text{cm}$, bottom right: $x_3 = 6.5\,\text{cm}$).

For the axial velocity component, however, the agreement between the experimental and simulation results was not satisfactory. For most of the profiles (namely for all presented axial profiles as well as for the radial profiles at heights $x_3 = 5\,\text{cm}$ and $x_3 = 8.75\,\text{cm}$), the simulation results were totally different from the experimental data. This shows that the flow simulation still has to be improved.

Remark 4.13. *In the literature, a deep analysis of the flow field in a stirred tank can be found. Besides mean velocity profiles, many authors also present profiles of phase-resolved mean velocities, root-mean square values of the velocity, and other turbulence data. Furthermore, vector-plots of the region around the impeller are shown, which illustrate the trailing vortices behind the impeller blades. By this, a very good description of all the time-dependent features induced by the rotating impeller is given, and the typical shape of the flow field, including the high speed discharge stream originating from the impeller towards the wall, the large recirculation zones in the bulk tank, the trailing vortices behind the rotating blades, and the vortices behind the baffles, is reproduced. However, since the quantitative comparison of the mean velocity values with experimental data shows only moderate agreement, we refrain from doing any further (and more*

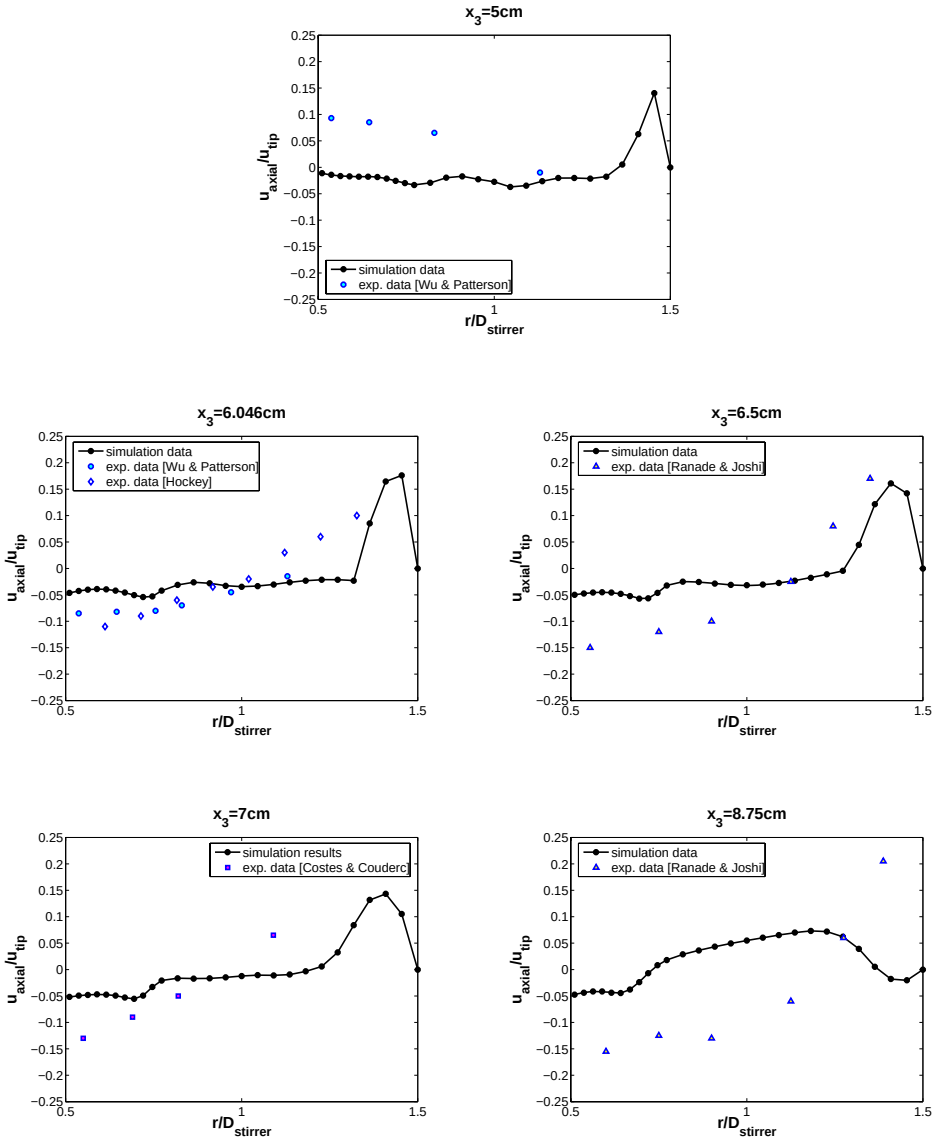


Figure 4.29: Radial profile of mean axial velocity between two baffles at different heights (first row: $x_3 = 5\,\mathrm{cm}$, second row, left: $x_3 = 6.046\,\mathrm{cm}$, second row, right: $x_3 = 6.5\,\mathrm{cm}$, third row, left: $x_3 = 7\,\mathrm{cm}$, third row, right: $x_3 = 8.75\,\mathrm{cm}$).

detailed) comparisons. In addition, a further analysis of the velocity components would go beyond the scope of this thesis, since for the coupling we will only need values of the velocity averaged over relatively large compartments.
On the other hand, the turbulence parameter ε, which is also needed for the coupling, requires a much finer resolution of the flow field or the use of some kind of turbulence model. Thus, these issues will be discussed in the following subsections.

4.3.3 Simulation on Level 7

The simulation results on the "adapted" grid of level 6 show that even in the refined region around the impeller the grid is still not fine enough to resolve all the scales of turbulence. Thus, it was decided to refine the grid again. However, it was not possible to start a simulation with FeatFlow on a grid of level 7 consisting of 22,475,457 nodes and 22,282,240 elements.

The first problem was that the parameter `NNWORK`, by which the maximal memory requirement of the program is defined, is declared as `INTEGER*4` variable, i. e., as an integer in the range $-(2^{31}-1) = -2.15 \cdot 10^9 \leq$ `NNWORK` $\leq 2.15 \cdot 10^9 = 2^{31}-1$. However, the amount of memory needed for a simulation on a grid of level 7 exceeded this range.
This problem was tried to overcome by substituting all 4-byte integers by 8-byte integers. However, the internal memory management of FeatFlow required also a doubling of the floating point variables. Thus, all `REAL*8` variables needed to be substituted by `REAL*16` variables, which led to a doubling of the whole memory requirement without changing the size of the problem.
Since for the simulation on a grid of level 6 with the "original" sizes for the variables already more than 4.5 GB were needed, it had to be expected that for the simulation of level 7 about 72 GB would be required, since, in this case, the grid is eight times finer (which corresponds to about eight times more variables) and the variables need twice as much memory. Nevertheless, we tried to start a simulation on a supercomputer with 64 GB shared memory. As already expected from the theoretical considerations, the simulation crashed due to lack of memory after less than four hours.

However, even if the simulation was successful (in the sense that it did not crash), it would not really have been possible to get results in a reasonable time frame. Since already the simulation of twelve revolutions (which are at least needed in order to get a fully-developed flow field) on a grid of level 6 took about three weeks (see Table 4.3 as well as the corresponding explanations in the following subsection), it must be expected that the same simulation on a grid of level 7 takes at least 24 weeks (factor eight due to the eight times more variables) even if the problem with `NNWORK` did not occur. In addition, since the server architecture does not support the handling of 16-byte floating point variables, the described solution would be even more time-consuming.

Grid type	Re	sim. time	rev.	comp. time
L5	1,000	10 s	4	30,384 s (approx. 8:26h)
L5	10,000	1 s	4	30,196 s (approx. 8:23h)
L5	30,000	0.3 s	4	29,923 s (approx. 8:19h)
L5, adapt.	30,000	0.3 s	4	34,213 s (approx. 9:30h)
L6	10,000	1 s	4	274,343 s (approx. 76:12h)
L6	30,000	0.3 s	4	265,356 s (approx. 73:43h)
L6, adapt.	30,000	0.3 s	4	292,255 s (approx. 81:11h)

Table 4.3: Simulation times for different grid types, refinement levels, and Reynolds numbers. Abbreviations used in the table: Re = Reynolds number, sim. time = simulated time (corresponds to the "real" time), rev. = revolutions, comp. time = computing time, L5 / L6 = grid of level 5 / level 6, adapt. = "adapted" grid.

4.3.4 Simulation Times

As already mentioned before, the simulations were done on two different refinement levels and for two different underlying coarse grids. Both types of grids had the same number of nodes: The grids of level 5 consisted of 360,369 nodes and 348,160 elements, whereas the grids of level 6 consisted of 2,833,761 nodes and 2,785,280 elements.

The simulations were done on an SGI Altix 3700, equipped with 32 Itanium II processors and a memory of 64 GB. The operating system is an SGI advanced server for Linux, Version 3.0 with SGI ProPack Version 3.0. The compiler is an Intel Fortran for ia64, Version 8.0.

All simulations were done with adaptive step size control, see page 53. However, we prescribed the same (scaled) maximum time step size for all cases. "Scaled" here means that the step sizes for different Reynolds numbers are considered as equal if the same number of time steps per revolution is used. Note that, for all simulations, this maximum time step size was used by the step size control after a few starting time steps.
Furthermore, all simulations were started from a fully developed flow field and with the same baffle-to-impeller position. In all cases, four revolutions were simulated, which corresponds to 10 s, 1 s, and about 0.3 s for Re = 1,000, Re = 10,000, and Re = 30,000, respectively.

The resulting computing times are shown in Table 4.3. Having a look at the last column in Table 4.3, one can immediately recognize the high computational costs for the simulations. The simulations on the grids of level 5 took between 29,923 and 34,213 seconds, i. e., between about eight and nine and a half hours. This means that the calculation of one revolution takes at least two hours. For level 6, the simulation times were in the range between 265,356 and 292,255 seconds, i. e., between three and 3.38 days (about three days and nine hours). Thus, the time requirement for one revolution is at least

about 20 hours.
The computing times appear even worse if we think about simulating one minute "real time" for a Reynolds number of Re = 30,000 (i. e., 720 revolutions), which would take about one year and eight month on a grid of level 6. For the simulation of the drop size distributions, time intervals of the order of magnitude of one minute are very small and thus, within the coupling approach, one would like to be able to calculate the flow field for such a time interval in a reasonable time frame.
However, as long as the stirred tank is operated with a constant stirrer speed (leading to a quasi-steady flow field after some revolutions, see Remark 4.11), it is sufficient to run simulations for a much smaller time interval and use the results also for the remaining time, which cannot be simulated due to the high computing times.

In the following, we will compare the computing times for the different simulations with each other. Let us start with the analysis of the computing times for the "ad-hoc" grid of level 5. If we compare the computing times for Re = 1,000 with those for Re = 10,000 and Re = 30,000, we can see that all simulations required about the same amount of computing time.
If we compare the computing times for the "ad-hoc" grid of level 5 with those for the corresponding "adapted" one (both for Re = 30,000), we can see that the time requirement for the latter one is a little bit higher than for the first one. The reason for this seems to be the "irregularity" of the "adapted" grid (where the differences in the sizes of neighboring elements are much larger).

The same behavior as for level 5 can also be observed for the grid of level 6. First, the simulation for Re = 30,000 ("ad-hoc" grid) took about the same time as the simulation for Re = 10,000. Second, the computing time for the "adapted" grid was again a little bit larger than the one for the "ad-hoc" grid.

Let us now compare the computing times for the grids of level 5 with those of level 6. For Re = 10,000 and the "ad-hoc" grid, the simulation on level 6 took about nine times longer than the one on level 5. Since the grid of level 6 consists of eight times more nodes than the grid of level 5, such a factor has to be expected. (Since the number of unknowns, which is given by the number of elements (corresponding to the pressure) plus three times the number of faces (corresponding to the velocity components), changes from 3,517,696 to 27,997,184, i. e., also by a factor of about eight, one would expect at least a factor of eight.) For Re = 30,000 ("ad-hoc" grid) the computing times on the different refinement levels differed by a factor of about 8.9. For the "adapted" grid, the ratio between the computing time on level 6 and the one on level 5 was 8.5.
However, it should be mentioned that for all simulations (i. e., also for those with a grid of level 6) the `gmv`-output was of level 5 only. Clearly, the ratios between the simulation times for level 6 and those for level 5 would become significantly larger if the `gmv`-output was saved for the same refinement level as the corresponding simulations.

At the end of this subsection, it should be mentioned that the simulation times needed for the bulk of the simulations presented in the previous subsections were even higher than those stated here. The reason for this is that the "reference simulations" presented

here were done after an update of the queuing system, whereas most of the previous simulations were done before. Before the update, the supercomputer, which has been used for the simulations, worked with a so-called "shared CPU set". This means that for a non-parallelized job only one processor with 2 GB memory could be "booked" in advance, even if the job needed remarkably more memory. The remaining amount of memory needed for the simulation was then taken from other processors, which did not work to full capacity, during the simulation. Clearly, in this case, the simulation times are not representative, since they are highly dependent on the load of the remaining processors. In contrast, the simulation times presented in this subsection were derived after the update of the queuing system. Here, also for a non-parallelized job that needs more than 2 GB memory, more than one processor could be reserved (so-called "exclusive CPU sets"). Therefore, enough memory was available for the job during the whole simulation, and, thus, the computing times are reproducible, and a comparison was possible.

4.4 Implementation of a k-ε Turbulence Model in FeatFlow

The simulation results presented in the previous subsections show that, in fact, a finer grid is required in order to resolve all the relevant scales of the turbulent flow field, at least, if a direct numerical simulation is used. This observation is also confirmed by theory, as already explained in the beginning of this chapter, see page 46.
However, the investigations in the previous Subsections 4.3.3 and 4.3.4 showed that the memory and time requirement for the simulation of the considered application are so high that it is not possible to run simulations on a finer mesh.
This shows that the implementation of a turbulence model is needed for the simulation of the considered practical application and, in particular, for the coupling with Parsival and for further applications like control or optimization.

In the following subsections, we will give a brief description of the newly implemented k-ε turbulence model in FeatFlow and discuss the problems that arise when this model is used in combination with moving boundaries.

Remark 4.14. *From the literature it is well-known that the k-ε turbulence model underlies some assumptions, which are usually not fulfilled for the flow field in a stirred tank reactor. One of these assumptions is the one of local isotropy, which is not satisfied in the impeller stream, see, e. g., [LSE00], [YPW87]. Another assumption, which is not fulfilled in stirred vessels, see [JR99], is the one of spectral equilibrium, which implies that, once turbulent kinetic energy is generated at the low-wave-number end of the spectrum (large eddies), it is immediately dissipated at the same location at the high-wave-number end (small eddies). Furthermore, simplifications are used in the derivation of the transport equations for k and ε, and the model constants in the k-ε model are determined from simple flow configurations, which are not comparable to stirred tanks, see [BBD00].*

Nevertheless, the k-ε model is one of the most frequently used turbulence models, at least in industry [BBD00]. Reasons for this are the successful application of the k-ε turbulence

model to many complex flow problems and the fact that the weaknesses of the model are relatively well-known and, thus, they can be accounted for.

Another reason for the use of the k-ε model in the considered application is that within the coupling approach values of the energy dissipation rate ε are needed for the calculation of the coalescence and breakage rates. Thus, the use of a turbulence model, which calculates these values directly, is obvious and sensible.

It remains the question if it makes sense to use one of the modifications / extensions of the standard k-ε model, like the Chen-Kim k-ε model or the renormalization group (RNG) methods. A description of these methods and their pros and cons compared to the standard k-ε model can be found in [JR99]. The numerical calculations presented there show that none of these k-ε models performs satisfactorily without modification. This observation was confirmed in [MLBY01], where it was concluded from test calculations that the RNG k-ε model does not perform better than the standard k-ε model. However, in [JR99] good agreement was obtained between measured and simulated data with an "optimized" version of the Chen-Kim k-ε model.

4.4.1 The Solver pp3d-ke

In order to avoid the extremely high costs of direct numerical simulation for three-dimensional flows at high Reynolds numbers, a k-ε turbulence model has recently been added to FeatFlow. This model allows to calculate such flows on meshes of moderate size. The corresponding CFD code `pp3d-ke` was developed by D. Kuzmin building on the laminar FeatFlow version, see `http://www.featflow.de`. The mathematical basis of this program is the system of Reynolds-averaged Navier-Stokes equations together with two transport equations for the turbulent kinetic energy k and its dissipation rate ε, see Eq. (2.5) on page 10. Additionally, appropriate boundary conditions for the averaged velocity $\bar{u}$ as well as for k and ε have to be prescribed. In this context, the modeling of the fluid near solid walls is of particular importance, since in these regions the k-ε model is not valid any more, see Subsection A.2.3 in Appendix A. In the solver `pp3d-ke` wall functions are used to model the region next to the wall. The corresponding equations and further details can be found in [KMS$^+$06].
The discretization in space is performed by an unstructured grid finite element method, see [Tur99]. A detailed description of the numerical algorithm for the k-ε model can be found in [KT04], [KLT05].
An application of the k-ε model in FeatFlow can be found in [HKM$^+$06]. Here, the solver `pp3d-ke` is coupled with Matlab in order to facilitate the use of standard control design techniques for fluid flow problems, like the control of the recirculation bubble behind a backward facing step.

4.4.2 Combination of pp3d-ke with Moving Boundaries

Since in the considered application some parts of the boundary are moving (rotating impeller), the moving boundary method described in Subsection 4.2.2 has to be adapted

for the k-ε turbulence model. However, the combination of the solver `pp3d-ke` with the moving boundary method meets some difficulties, which will be discussed in the following.

A first observation is that the fictitious boundary method requires a very fine mesh in the area, where the impeller is rotating. If the grid is too coarse, the thin blades of the impeller are prescribed very inaccurately. This gives rise to large undesired errors in the calculation of the velocity field.
The second difficulty arises from the fact that the available wall boundary laws for the k-ε turbulence model require the boundary to be smooth and very plane, since they were derived from physical experiments of channel flows.
From these considerations and under the assumption that the major changes in the velocity field are due to the rotating impeller, one can conclude that it is preferable to perform a coordinate transformation using a fixed impeller, around which the outside wall of the tank is rotating. This transformation, however, leads to additional terms in the Reynolds-averaged Navier-Stokes equations, namely the Coriolis and centrifugal forces (see [KMS+06]). Since, in the latter one, the averaged velocity does not appear, it can be added to the pressure term. By doing this, the equations are solved for the transformed "pressure" consisting of the "real" pressure and the part derived from the centrifugal forces term. On the other hand, the Coriolis force contains the averaged velocity $\bar{u}$ and, thus, leads to a "real" change in the equations and this additional term must be implemented in the CFD code.

Finally, we have to consider the boundary conditions in this setting. The given wall boundary conditions in the k-ε turbulence model, which are prescribed everywhere on the solid wall, are not sufficient, since we are not able to deal with the tangential part of the velocity vector field on the boundary. This problem can be overcome by realizing generalized boundary conditions according to [KFT00] and [JD92]. Further details can be found in [KMS+06] and references therein.

4.5 Discussion and Perspectives

The current simulation results show many challenges. Obviously, the results of the flow simulation can still be improved. However, if direct numerical simulation is used, a much finer grid is required, which clearly cannot be handled due to the high time and memory requirements for the simulation, see Subsections 4.3.3 and 4.3.4.

A way out would be the use of a parallel CFD solver. A parallel version of the solver `pp3d` is already available on the Internet, see `http://www.featflow.de`. However, for the considered stirred tank application, the moving boundary method is needed, i. e., the solver `pp3d-movbc` has to be used, which is not parallelized yet.
On the other hand, even if a parallel CFD code is used, the time and memory requirements for a direct numerical simulation are still enormous. Thus, it is sensible to use some kind of turbulence modeling, which is currently being completed, see Section 4.4.

Remark 4.15. *In general, there are a lot of arguments for and against the use of a k-ε turbulence model, see also Remark 4.14 on page 95. Despite the strong assumptions of the k-ε model, which are not fulfilled for most flow configurations, the k-ε turbulence model is one of the most widely used turbulence models.*
According to [JR99], it is generally accepted that Reynolds-averaging of the Navier-Stokes equations and modeling the Reynolds stresses with an appropriate turbulence model are a promising way of modeling the flow behavior. However, in [YPL03] it was stated that methods based on Reynolds-averaging predict the mean flow reasonably well, but underpredict the turbulent kinetic energy by about 50 percent. Furthermore, in [BBD00] it was claimed that for most key features the results of direct numerical simulation are closer to experimental findings than the k-ε predictions. However, their numerical simulations were only done for a Reynolds number of $\mathrm{Re} = 7{,}275$.
On the other hand, the use of direct numerical simulation requires the resolution of the motions down to the dissipative scales. Therefore, direct numerical simulation is only applicable to relatively low Reynolds number flows and not to industrial related applications [YPL03].

As already mentioned before, an important reason for the use of a k-ε model for the considered application is that within the coupling approach the value of ε is needed for the calculation of the coalescence and breakage rates. However, the use of a turbulence model, which takes into account the small scales (by modeling them), can be problematic in combination with a compartment model, which simply neglects all these scales. Note that the influence of the drops on the flow field is neglected in the considered approach and that every zone is assumed to be ideally mixed in the compartment model.

At the moment a sensible adaption of the k-ε turbulence model solver `pp3d-ke` to the application of stirred liquid-liquid systems is investigated, see Subsection 4.4.2, including the coordinate transformation described there. As already mentioned in this subsection, the combination of the k-ε turbulence model with the moving boundary method leads to additional problems, since the wall boundary laws for the k-ε model require a smooth and plane boundary. However, in general, this condition is violated by the "zig-zag" approximation used in the moving boundary method, see Fig. 4.8 on page 62.

An additional possibility to save computational costs is to use the symmetry of the problem, i. e., to simulate only the flow field in a part of the tank representing the whole. However, since the stirred tank is equipped with a six-bladed Rushton turbine, but only with four baffles, at most a halfing of the costs is possible. Note that the baffles cannot be neglected, since they prevent the liquid from performing a solid-body rotation and, thus, enhance mixing, see, e. g., [Kra03] or [DV99]. In order to use the symmetry of the problem, the CFD code must be able to handle symmetric boundary conditions, which is not realized for the three-dimensional solver `pp3d` in FeatFlow.

Another difficulty is that the grid generation is very complicated (due to the quadrilateral elements used in FeatFlow), see Subsection 4.2.1. Thus, it is really hard to get an "optimal" grid for the simulation. However, the simulation results show that a refinement of the mesh in the region of large velocity gradients is recommended. This finding is in

accordance with the analysis in [JR99], where it is stated that local refinement of the grid in several regions of the vessel is necessary in order to resolve sharp gradients of mean flow and turbulence quantities. In [YPW87] it is even stated that the numerical resolution of steep gradients present in the flow is more important to the solution of the flow field than the representation of turbulence.
On the other hand, an only local refinement of the grid can reduce the number of grid points needed for a direct numerical simulation. In the stirred tank application, for example, it should suffice to use a very fine grid (which is able to resolve all scales of turbulence) in the region around the impeller only and a coarser one in the remaining part of the tank (since the local Reynolds numbers are much smaller there). An application, where a direct numerical simulation in combination with a locally refined grid was used, can be found in [BBD00], for example.

5 Coupling the CFD Code FeatFlow with the Population Balance Solver Parsival

In this chapter, we will discuss how the coupling between the flow simulation and the solution of the population balance equation can be realized. We will use the solvers FeatFlow and Parsival in a simulator coupling approach in order to solve the system of equations describing the processes in a stirred tank reactor, see Eq. (3.1) on page 32.

Remark 5.1. *In general, the coupling that is hidden in the description / modeling of the different processes in a stirred tank reactor can be considered on different levels: on the level of physics, on the level of equations, and on the level of simulation, see Fig. 5.1. In [KS00] these levels are denoted by "physical model description", "mathematical model description", and "behavioral model description".*

On the level of physics, the coupling describes how the flow field in the stirred tank and the population dynamical processes of the drops interact with each other. On one hand, the turbulent flow influences the behavior of the drops. For example, there are more breakage and less coalescence phenomena in the turbulent region close to the impeller than in the less turbulent regions of the tank. On the other hand, the drop size distribution throughout the tank can also have an influence on the flow field. Bigger particles can enhance turbulence, for example, whereas smaller particles can lead to a damping of turbulence.

If we consider the coupling on the level of equations, we have, on one hand, the Reynolds-averaged Navier-Stokes equations with k-ε turbulence model describing the turbulent flow in the stirred tank and, on the other hand, the averaged population balance equation accounting for the behavior of the drops. In principle, the best way to treat the arising system is to solve all equations together as one coupled system (the so-called "monolithic model"). In Chapter 3, the dynamics of this coupled system have been analyzed and it has been shown that, at least from the numerical point of view, existing algorithms for the solution of the Navier-Stokes equations could be extended such that they can be used to treat the coupled system. However, the effort to build such a solver is immense, and, thus, this approach is currently not feasible, at least not in the short-term construction of a solver that can be used in practical applications.

In practice, today the only way to get a solution in a reasonable time frame is to couple existing solvers, which represents the coupling on the level of simulation. In this approach, which is known as "co-simulation", already developed and implemented methods are used, which are typically combined via a relaxation method (see, e. g., [KS00] or [MJWR99]). This means that, in each iteration step, one method uses the results of a

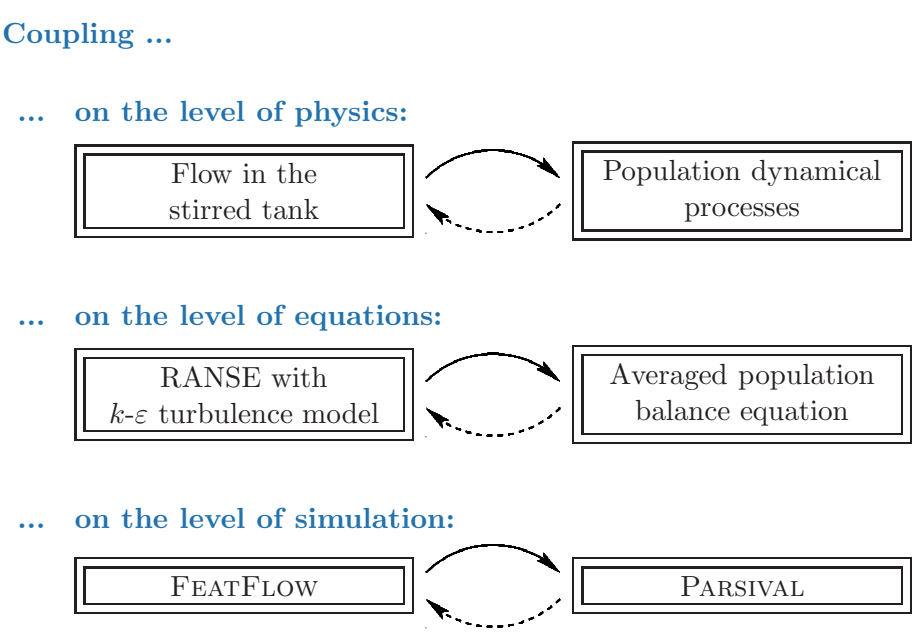


Figure 5.1: Different levels of coupling. Abbreviation used in the figure: RANSE = Reynolds-averaged Navier-Stokes equations.

previous step of the other method and vice versa.
The advantage of this approach is the enormous saving of time needed for modeling, development, and implementation. Furthermore, the subsystems can be modeled independently and in parallel, the resulting modules can be exchanged easily, and different software can be used for each module [KS00], which means that an "optimal" solver can be chosen for each subproblem.
On the other hand, the use of different tools for the description and simulation of the subproblems must be handled and the data exchange between these tools must be implemented. Another disadvantage of this approach is that the results can only rarely be verified by existing theory. Furthermore, simulator coupling can lead to convergence problems, and high computation times must be expected.
For theoretical aspects of simulator coupling approaches, including stability and convergence analyses, see, e. g., [KS00], [AG01], [Ebe04], and [RS05].

In the following, the simulator coupling approach will be used to solve the coupled system. However, we will only consider a so-called "one-way coupling" here, which means that we only consider the influence of the turbulent flow field on the drops, but do not take into account that the flow field is also affected by the drop size distribution. Note that this influence was already neglected in the modeling approach in Chapter 2.
In brief, the applied simulator coupling approach can be summarized as follows. In a first step, the flow field is calculated with FeatFlow on a very fine grid. According to the results of the CFD simulation, the whole tank is then subdivided into a much

smaller number of compartments. Since the drop size distributions are assumed to be space-independent inside these compartments, the division has to be done in such a way that this is (at least approximately) satisfied. Afterwards, the drop size distributions are calculated in PARSIVAL, where the results of the CFD simulation are used in the coalescence and breakage rates as well as for the calculation of the exchange flows between the different compartments.

Remark 5.2. *In the literature, one can already find such kind of multi-compartment models in combination with CFD simulations for different applications, like stirred liquid-liquid systems (see, e. g., [AKK99], [AKKM02], or [PB]), stirred gas-liquid systems (see, e. g., [LAA02] or [VDV02]), or imperfect mixing in reactive systems (see, e. g., [WR05]). In these papers, the compartment model approach has been combined with different solution strategies for the population balance equation, namely with the method of classes (see [AKK99], [AKKM02], [LAA02], or [VDV02]) and the method of moments (see [WR05]).*

The idea of the multi-compartment approaches is to use different grids for the CFD simulation and for the calculation of the other equations describing population dynamical processes or reaction kinetics, for example. While a very fine grid is used for the CFD simulation, the remaining equations are solved on a much coarser grid. According to the results of the CFD simulation, the reactor is subdivided into several compartments, which are then assumed to be ideally mixed. The number of compartments varies dependent on the considered application reaching from two to hundred (or even more) compartments. (In [AKK99], [AKKM02], [LAA02], and [PB] between ten and fourteen compartments are used, whereas the model in [WR05] consists of hundred compartments.)

By means of these compartment model approaches, a more detailed model (than using just one ideally mixed reactor for the whole simulation) is derived, since dependencies on locally varying quantities are included in the model. On the other hand, compartment models save enormous computing time compared to models that use an as fine grid as the one for the CFD simulation also for the calculation of the other equations.

This chapter is organized as follows. First, in Section 5.1, we will give a short introduction to the mathematical background of PARSIVAL. Furthermore, we will briefly describe how to implement a model in PARSIVAL. Then, in Section 5.2, we will summarize some results of the simulation of the drop size distributions with PARSIVAL. In this context, we will also discuss the influence of the choice of different model parameters and different coalescence and breakage rates on the resulting drop size distributions. In Section 5.3, we will explain how the coupling between FEATFLOW and PARSIVAL can be realized. After a presentation of the results of the coupling and a comparison with experimental data in Section 5.4, we will discuss the pros and cons of the presented approach in Section 5.5 and give some hints how the coupling can be improved in the future. Finally, in Section 5.6, we will present the quadrature method of moments and the direct quadrature method of moments, which both represent good alternatives for solving the population balance equation, in particular, if they are used within a coupling approach, since they can easily be integrated into a CFD code.

5.1 About Parsival

In this section, we will give a brief introduction to Parsival. Parsival, which is short for **Par**ticle **si**ze e**val**uation, is a simulation package for (space-independent) population balances, i. e., for equations of the following form:

$$\frac{\partial f(e,t)}{\partial t} = s(f(e,t),e,t). \tag{5.1}$$

Here, $f(e,t)$ denotes the number density function, which depends on internal coordinates e and on the time t. With a current version of Parsival (version 6.14.5) only one internal coordinate can be handled, namely the diameter d of the drops. However, an extension of Parsival for more than one internal coordinate is currently being developed, see [Par]. Furthermore, the function s on the right-hand side of Eq. (5.1) denotes the sum of source and sink terms, which may consist of terms due to coalescence and breakage (as in the considered application), but also on terms accounting for other population dynamical processes, like growth, shrinkage, formation, or dissolution.
Note that Eq. (5.1) is not as easy to handle as it looks like, since the right-hand side may involve partial derivatives as well as integrals depending on which processes are considered.

Further assumptions on the models that can be simulated with Parsival can be found in the Parsival manual [Wul].

Although only space-independent population balance equations can be treated with Parsival, one can nevertheless realize some kind of space-dependency by substituting the total (space-dependent) system by several smaller (ideally mixed, i. e., space-independent) subsystems, which are "connected" with each other by volume flows. Since this approach is used for the realization of the coupling between FeatFlow and Parsival, further details can be found in Section 5.3.

5.1.1 The Numerical Algorithm Implemented in Parsival

The numerical algorithm implemented in Parsival is based on a generalized finite-element scheme with self-adaptive grid and order construction and is connected to a time discretization of Rothe's type, see [WGN01] and [WGN99]. It is called Galerkin *h-p* method (*h*: variable grid, *p*: variable order). In the following, we will describe this numerical algorithm, which is shown graphically in Fig. 5.2, and give some basic features of Parsival.

As just mentioned, Parsival uses a Rothe method, which means that the population balance equation is first discretized in time. However, in contrast to the solution of space- and time-dependent partial differential equations, where the application of the Rothe method leads to a system of stationary space-dependent partial differential equations, the time-discretization here leads to a system of stationary partial differential equations that is dependent on the internal coordinate, i. e., on the diameter of the drops. The

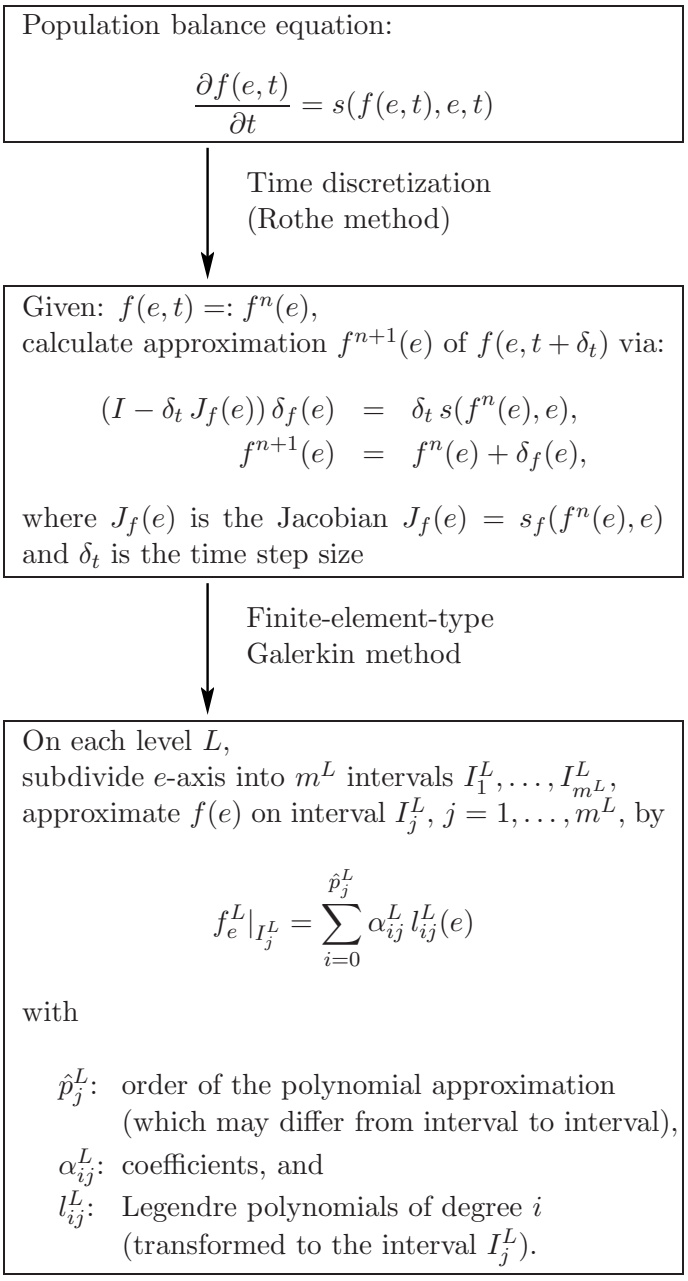


Figure 5.2: Schematic representation of the numerical algorithm implemented in PARSIVAL.

difficulty in the solution of these equations is that the discretization with respect to the diameter of the drops is much more complicated than a "normal" spatial discretization. The challenge here is that conservation laws, which ensure that the total volume of the dispersed phase (i. e., the total volume of all drops) is maintained during the coalescence and breakage processes, still have to be satisfied after the discretization with respect to the internal coordinate (see also Remark 5.3).

In Parsival, the semi-implicit Euler scheme is applied as time discretization method. Thus, the discretization of Eq. (5.1) leads to the following problem:

Given $f^n(e) := f(e,t)$ and time step size δ_t, then solve for $f^{n+1}(e)$

$$\begin{aligned} (I - \delta_t\, J_f(e))\, \delta_f(e) &= \delta_t\, s(f^n(e), e), \\ f^{n+1}(e) &= f^n(e) + \delta_f(e), \end{aligned} \tag{5.2}$$

where δ_f is a correction term and $J_f(e)$ denotes the Jacobian $J_f(e) = s_f(f^n(e), e)$.

By this, we obtain a linear system (with respect to $\delta_f(e)$), which is still a partial differential equation, but now a stationary one. Thus, it has to be solved by some kind of approximation technique.

Clearly, both, the time discretization as well as the approximation of the stationary problem, introduce errors. In Parsival, an adaptive time control tries to keep the error of the time discretization ϵ_t below a certain tolerance by controlling the time step size. The idea of the approach is that the solution of Eq. (5.2) may be slightly perturbed by the approximation of the stationary problem without affecting the time step size and accuracy significantly. The stationary approximation error ϵ_s just has to be kept below a certain accuracy. In fact, the stationary approximation error ϵ_s is matched to the time error ϵ_t such that the complete time step is performed within the overall tolerance. It is a special feature of the Rothe method that the requirement for ϵ_s is not too strict, when an appropriate time discretization scheme is used, see [WGN01].
In Parsival, the multiplicative error correction (MEC) [Bor91] is used for the time step control, since it avoids differences in the computation of the time error and uses the same left-hand side as in Eq. (5.2) so that no additional decomposition has to be done. For further details concerning the time integration and control, see [WGN01].

Within a time step, the equations that remain to be solved both have the following form:

$$(I - \delta_t\, J_f(e))\, f(e) = h(e), \tag{5.3}$$

where the right-hand side $h(e)$ and the solution $f(e)$ contain all components of the system or its numerical representation at time t or $t + \delta_t$, respectively. The task is now to find an approximation $\hat{f}(e)$ of $f(e)$ that satisfies $\|\hat{f} - f\| < \epsilon_s$. Thus, we approximate the solution of Eq. (5.3) by means of the following finite-element type Galerkin method. In this approach, the drop size axis (e-axis) is subdivided into several intervals, and on each interval $f(e)$ is approximated by using a local expansion f_e of $f(e)$.

This approximation is realized by means of a multi-level algorithm as follows. On the L-th level, the approximation f_e^L on interval I_j^L, $j = 1, \ldots, m^L$, where m^L denotes the

total number of intervals on that level, is given by

$$f_e^L|_{I_j^L} = \sum_{i=0}^{\hat{p}_j^L} \alpha_{ij}^L \, l_{ij}^L(e).$$

Here, $\hat{p}_j^L$ is the order of the polynomial approximation on interval I_j^L, α_{ij}^L are the expansion coefficients, and l_{ij}^L are the Legendre polynomials of degree i (transformed to the respective interval I_j^L). The crucial point here is that the number of expansion coefficients (i. e., the order of the approximating polynomials) may vary from interval to interval such that the form of the drop size distribution can be resolved by the varying order.
The construction is started with an initial grid (for the choice of this initial grid, see [WGN01]) and proceeds from level to level by refining the intervals or by increasing the order. (For details about the refinement strategies on each level, see also [WGN01].)
The node-order-pattern (i. e., the number and the size of the intervals and the respective polynomial orders) on the final grid, i. e., $(I_j^{\text{final}}, \hat{p}_j^{\text{final}}), j = 1, \ldots, m_{\text{final}}$, are chosen in such a way that the amount of work necessary to compute the whole approximation is as small as possible.

Further details on the Galerkin *h-p* method and especially on the construction of the subdivisions as well as on the finite representations of the drop size distribution can be found in [WGN01].

Remark 5.3. *In general, there are several approaches how the internal coordinate in a population balance equation can be treated. Among the most commonly used methods, there are the discrete formulation for the solution known as method of classes and the method of moments approaches.*

In the method of classes, the axis of the internal coordinate is subdivided into several intervals, and the population balance equation is discretized with respect to these intervals. However, for many applications, usually neither an arithmetic discretization (where an equidistant subdivision of the size axis is used) nor a geometric discretization (where the size of the next class is twice as large as the size of the previous class), see Fig. 5.3 (left), leads to satisfactory results, since, due to the high computational costs, only a few classes can be used. Moreover, if the distribution changes with time, a fixed discretization leads to a rough approximation only. Thus, an intelligent adaptive algorithm (as the one used by Parsival*) is needed, which controls the discretization of the internal coordinate depending on the shape of the distribution.*
An alternative approach, which uses $\mathcal{H}$-matrix methods for the numerical treatment of the Galerkin discretization of the integral operators in the population balance equation in order to reduce the computational costs, has been introduced in [Koc05] (see also [KHS06]).
Another difficulty of the method of classes is the conservation of mass during the population dynamical processes, see Fig. 5.3 (right). In the literature, several approaches for this problem can be found, see [BHB81], [HRM88], [LSH95], [Van99], [KR96a], and [KR96b]. A comparison of the different methods can be found in [Van00].

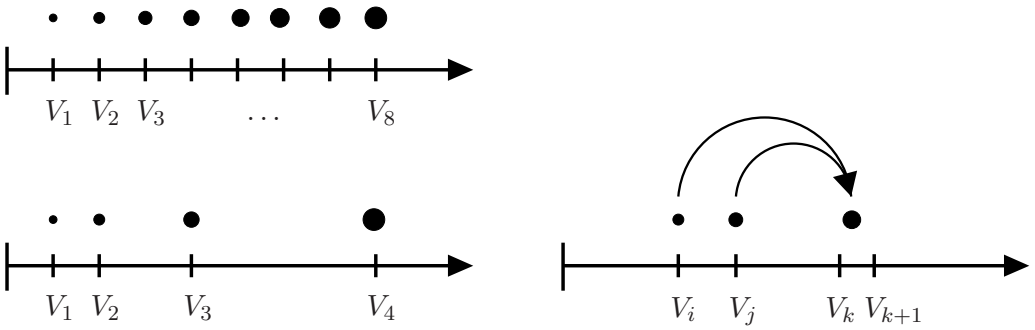


Figure 5.3: Left: Discretization of the drop volume, top: arithmetic discretization: $V_{i+1} - V_i = \text{const}$, bottom: geometric discretization: $V_{i+1}/V_i = \text{const}$. Right: Problem of mass conservation: The coalescence of two drops with the (discretized) volumes V_i and V_j leads to a drop with volume $V_i + V_j$, which usually lies between two discretization points V_k and V_{k+1}.

In the method of moments approaches, the internal coordinate is "integrated out", and the equations for a finite set of moments (that are independent of the internal coordinate) are solved instead of the original population balance equation.
For Eq. (5.1), this reads as

$$\frac{\partial m_k(t)}{\partial t} = \int_{-\infty}^{\infty} s(f(e,t),e,t)\; e^k \, de =: \bar{s}_k(t), \qquad k = 1, \ldots, N_{mom},$$

where m_k denotes the k-th moment, given by

$$m_k(t) = \int_{-\infty}^{\infty} f(e,t)\; e^k \, de.$$

However, in general (e. g., for complicated coalescence kernels), it is not possible to write the equations for the first N_{mom} moments in terms of other lower-order moments. A way out of this is given by the quadrature method of moments and the direct quadrature method of moments, which are described in Section 5.6.
The advantage of these approaches is that they are very efficient and that they can easily be implemented in a CFD code. On the other hand, the reconstruction of the whole distribution from a finite set of moments is, in general, not possible, unless assumptions on its shape are made.

Other possibilities for the solution of population balance equations are, e. g., the method of Laplace transforms or Monte Carlo simulation methods, whereof the latter ones are very common. A brief description of these approaches and further references can be found in [Ram00].

The numerical algorithm described above, see also Fig. 5.2, has been implemented in the object-oriented software package Parsival, which is a dynamic flow sheet simulator with a full graphical user interface. In Parsival, the population dynamical processes in

a multi-phase system are represented by six types of components, which can be defined by the user: *Reactors, Fluids, Distributions, Streams, Coefficients,* and *Modules*. All processes can be combined in terms of implemented basic modules (such as growth, coalescence, breakage, etc.) that consist of a pre-defined equation pattern, which can be "filled" with an appropriate kernel depending on the desired model. (A short description how a stirred tank model can be implemented in PARSIVAL can be found in the following subsection.)

Further information about the features and the use of PARSIVAL can be found in [Wul], [GB04], and [Par]. One of the additional features of PARSIVAL is the estimation of parameters from given experimental results. By this, free parameters often occurring in the models from the literature can be fitted and thus optimized with respect to a certain application (see also Section 5.2). An example of simulating a mixed suspension, mixed product removal crystallizer with PARSIVAL can be found in [WGN01] and [WGN99].

5.1.2 Implementation of a Stirred Tank Reactor in PARSIVAL

Before a simulation with PARSIVAL can be started, the complete model including the parameters and the initial values of all components and reactor variables must be provided by the user. This input data can be created and edited in the PARSIVAL-Workshop, a user-friendly tool that allows the creation of input files required for the simulation in a dialogue-guided way. Although the resulting input file is a plain ASCII-file, which may also be modified in an arbitrary text editor, this is not recommended (and also hardly possible), since the required syntax is not documented.

As already mentioned in the previous subsection, there are six types of components that can be used for modeling a stirred tank in PARSIVAL.

- In the first component, called *Reactors*, one can specify the name, the temperature, the operation mode (batch, semi-batch, or continuous) as well as the volume of the different reactors, the whole model consists of. Furthermore, one can declare a so-called control stream for each reactor. During the simulation, this control stream is used to keep the volume of the reactor constant. Note that if an ideally mixed tank is modeled, only one reactor is needed. However, in order to model a non-ideal tank, a compartment model consisting of several reactors has to be implemented (since each reactor is assumed to be ideally mixed). The exchange between these compartments can be modeled by volume flows, see Section 5.3 as well as the description of the components *Streams* and *Modules* below.
- In the component *Fluids*, one can specify the name, the initial value, the molar mass, and the density of the different fluids, the continuous phase consists of. Furthermore, one has to specify the name of the unit (i. e., the name of the reactor) the fluid component belongs to. Note that more than one fluid can be used for describing the continuous phase (e. g., if the continuous phase consists of a mixture of different fluids). If a multi-compartment model is used, at least one fluid has to be used for each compartment (i. e., for each reactor).

- Analogously, the dispersed phases are modeled by the component *Distributions*. Here, one can, besides the name, the start distribution, the density, and the unit it refers to, also specify a so-called shape factor, which describes the relationship between the drop sizes and their volumes.

- By means of the component *Streams*, the flow into, out of, and between the different reactors can be described. Here, one has to specify the name of the stream, the unit it belongs to, its kind (volume stream or mass stream), its direction (feed stream or exit stream), and its temperature. In a multi-compartment model, the exchange between different reactors can be modeled by exit streams in combination with appropriate modules describing the flow of the continuous phase as well as of the drop size distributions between the compartments.

- In the component *Coefficients*, all constants have to be specified. Only after the coefficients have been defined here, they can be used in the other components (as part of functions, for example).

- *Modules* are pre-defined patterns or templates that have to be filled with appropriate data, like with declared components (*Fluids*, *Distributions*, *Streams*, or *Coefficients*) or with kernels (describing coalescence or breakage rates, for example). Every module leads to a specific system of differential equations for the participating reactants. By collecting all these terms, the user can construct the complete system of differential equations solved by Parsival. Modules may describe population dynamical processes, like coalescence (or agglomeration), breakage, attrition, dissolution, or growth of particles. In addition, equations describing feed or exit streams as well as diffusion or convection terms can be modeled with pre-defined modules. Furthermore, boundary conditions can be specified via modules. A list of available modules can be found in the Parsival-Manual [Wul].

A short description of the components we have implemented in the four-compartment model, which we have used in the coupling approach, will be given in Section 5.3.

5.2 Results of the Simulation with Parsival

In this section, we will summarize some results of the simulation of drop size distributions with Parsival, which have been carried out in a joint project, mainly by the project partners at the Institute for Process Engineering at the TU Berlin. The aim of these simulations was to find out if existing models from the literature are capable of reproducing the measured drop size distributions and, if not, how these models can be improved and extended so that they can be used as predictive models. By means of systematic simulations it can be investigated, which parameters have an influence on the drop size distributions in a certain application. This knowledge can then also be used for the control and optimization of drop size distributions, usually with the aim to achieve a desired average drop size and a small standard deviation of the distribution.

In general, drop size distributions in stirred liquid-liquid systems are dependent on many parameters. Among them, there are:

- the power input,
- the dispersed phase fraction,
- geometrical parameters, like
  - the ratio between the diameter of the tank and the diameter of the stirrer,
  - the ratio between the tank height and the clearance, or
  - the shape of the stirrer,
- physical properties of the species, such as
  - interfacial tension,
  - viscosity,
  - ionic strength,
  - pH-value, or
  - surface tension.

Clearly, these influences should also be reflected by the mathematical models describing coalescence and breakage processes. However, in most models from the literature, not all parameters are taken into account. Instead of that, fitting parameters are introduced in the models, which can be adapted for a certain application. The problem with this approach is that the simulations with one set of parameters often cannot reproduce the experimental drop size distributions if different values for a certain input parameter (such as the rotational speed) are used. Thus, these models cannot be used as predictive models.

This difficulty is the reason why systematic simulations are required in order to improve the models from the literature by taking into account further dependencies on physical parameters, which can be detected by experimental investigations, but are not yet present in the mathematical models.

Remark 5.4. *A quantitative understanding of drop breakage and coalescence mechanisms is essential for the development of predictive models describing the drop size distributions in stirred liquid-liquid systems. Currently, it is still not possible to sufficiently describe a drop size distribution as a function of process and physical parameters in an inhomogeneous flow field typical for a stirred tank reactor. The comparison of experimental drop size distributions and simulation results with different models from the literature show significant discrepancies, see, e. g., [Weg04]. Even the most popular models, like the one by Coulaloglou and Tavlarides [CT77] or the one by Alopaeus et al. [AKK99], partially lead to only poor simulation results if system parameters, like the speed of the impeller or the volume fraction of the dispersed phase, are changed without*

adapting the model parameters. Therefore, a further analysis of the known models as well as their improvement are required for a reliable simulation and control of stirred liquid-liquid systems.
An advantage of simulations over experimental investigations is that the population dynamical processes (i. e., coalescence and breakage) can easily be analyzed separately. In experiments, however, the separation of the different population dynamical processes is much more complicated, see, e. g., [MGW+06].

The following investigations were performed for the "standard" stirred tank, equipped with a six-bladed Rushton turbine and four baffles, described in Chapter 4. (The exact dimensions can be found on page 66.)
First, we will present results for so-called "one-compartment models", where the assumption that the stirred tank is ideally mixed is used. Afterwards, we will describe the results for "more-compartment models", where the different energy dissipation rates in different regions of the stirred tank are taken into account.

5.2.1 Results for "One-Compartment Models"

- In [Weg04], a stirred liquid-liquid system, consisting of water as continuous phase and toluene as dispersed phase with a volume fraction of 10 %, was investigated. Three different rotational speeds (400, 550, and 700 rounds per minute) and seven different pH-values $(1, 3, \ldots, 13)$ were used in the experiments. However, the simulations with Parsival were only performed for a pH-value of 13. This pH-value leads to a coalescence-hindered system (see, e. g., [GWPK06a]) so that transient drop size distributions can be observed. For pH-values smaller than 11, the final steady-state distributions are reached so fast that the change of the drop size distributions with time can hardly be measured.
  For the coalescence and breakage rates, three different models were implemented in Parsival. Note that in all three models the same coalescence rate (namely the one from [CT77]) and the same distribution of daughter drops (namely a Gaussian normal distribution) were used, and that only the breakage rate was varied. For the latter one, approaches from [CT77], [AKK99], and [MML99a] were implemented. The fitting parameters in these models were optimized using Parsival's parameter estimation, which minimizes the differences between simulated and experimental data. By this, the optimal values with respect to the experimental data for a rotational speed of 400 rounds per minute were obtained. This set of parameters was then also used for the simulation of the other rotational speeds.
  A comparison of the resulting drop size distributions with the experimental data showed good agreement for all three models for the rotational speed of 400 rounds per minute. However, the results for other rotational speeds showed major discrepancies, see Fig. 5.4. The simulated drop size distributions were too broad and consisted of too many small drops. This suggests the assumption that the model parameters are not only dependent on physical parameters of the dispersion, but

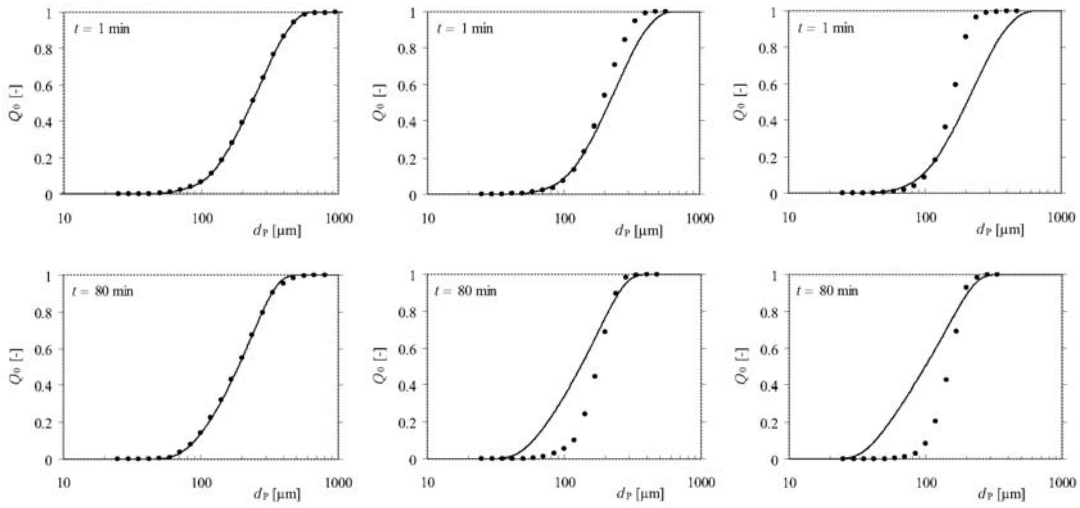


Figure 5.4: Experimental (dots) and simulated (lines) drop size distributions (cumulative number distributions) for 400 (left), 550 (middle), and 700 (right) rounds per minute (taken from [Weg04]). For the simulation, the model from [CT77] was used. Note that the model parameters were fitted with respect to experimental data for a rotational speed of 400 rounds per minute.

also on the rotational speed. Another reason for the deviation could be that the small drops could not be measured in the experiments. A surprisingly good agreement between experimental data and simulation results was found for a step in the rotational speed from 700 to 400 rounds per minute, see Fig. 5.5. The transient behavior of the drop size distribution was reproduced very well in this case.

Altogether the conclusion can be drawn that, for all models, the simulation results agreed very well with the experimental data if the rotational speed, which was used for the fitting of the parameters, was not varied for the simulation. This is not really surprising, since in this case the shape of the distribution can be "adapted" by choosing the "right" parameters. However, if the same set of parameters is used for much higher (or lower) rotational speeds, the simulated drop size distributions may vary significantly from the experimental results.

- Similar results were obtained in [GWSK05] (see also [GWSK04]). Here, the same liquid-liquid system (consisting of water as continuous and toluene as dispersed phase with a volume fraction of 10 %) was investigated. The drop size distributions were determined experimentally for the same pH-values $(1, 3, \ldots, 13)$ and for the same range of rotational speeds. However, in [GWSK05], five different rotational speeds were considered: 400, 475, 550, 625, and 700 rounds per minute. As in [Weg04], also a step in the rotational speed from 700 to 400 rounds per minute was investigated in order to observe the coalescence behavior of the drops.
  Numerical simulations were performed with PARSIVAL using the coalescence and

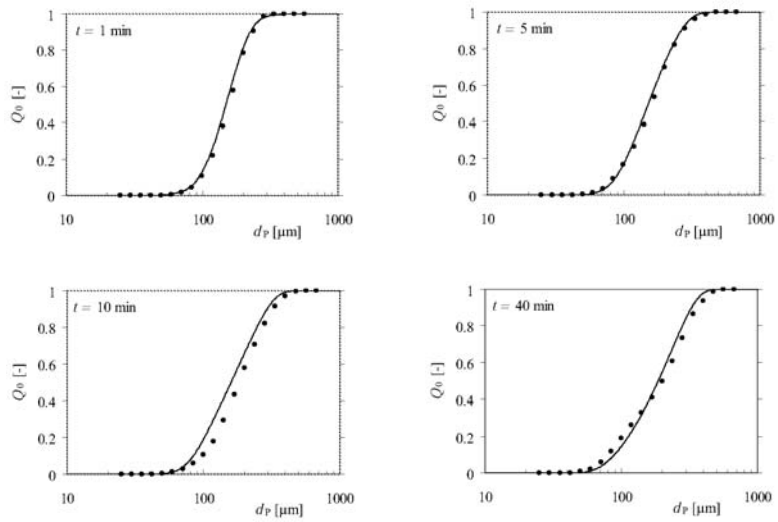


Figure 5.5: Experimental (dots) and simulated (lines) drop size distributions (cumulative number distributions) for a step in the rotational speed from 700 to 400 rounds per minute for 1, 5, 10, and 40 minutes after the step (taken from [Weg04]). For the simulation, the model from [CT77] was used.

breakage rates from [CT77]. It was decided to use this model, since it showed the best fit with experimental data. As in [Weg04], Parsival's parameter estimation was used to determine the optimal model parameters with respect to the experimental data for a rotational speed of 400 rounds per minute. The same set of parameters was then also used for the simulation with the higher rotational speeds.
A comparison with the results from the experimental investigations showed, as expected, very good agreement for the rotational speed of 400 rounds per minute, but only moderate agreement for the higher rotational speeds. This indicates that the coalescence and breakage rates are dependent on the rotational speed. Thus, one can expect better results if the influence of the rotational speed is taken into account in the modeling.

- In [GSP+05], the parameter sensitivity regarding the energy input was studied by deriving the optimal model parameters for five different rotational speeds (400, 475, 550, 625, and 700 rounds per minute) and comparing them. This analysis showed that the two parameters occurring in the breakage rate from [CT77] are dependent on the stirrer speed (the higher the rotational speed, the greater the parameters), see Fig. 5.6 (top). This confirms the indication that, in order to obtain a better description of the drop size distributions, the effect of the rotational speed has to be taken into account in the model for the breakage rate. On the other hand, the parameters in the coalescence rate did not show a dependency on the rotational speed, see Fig. 5.6 (bottom).

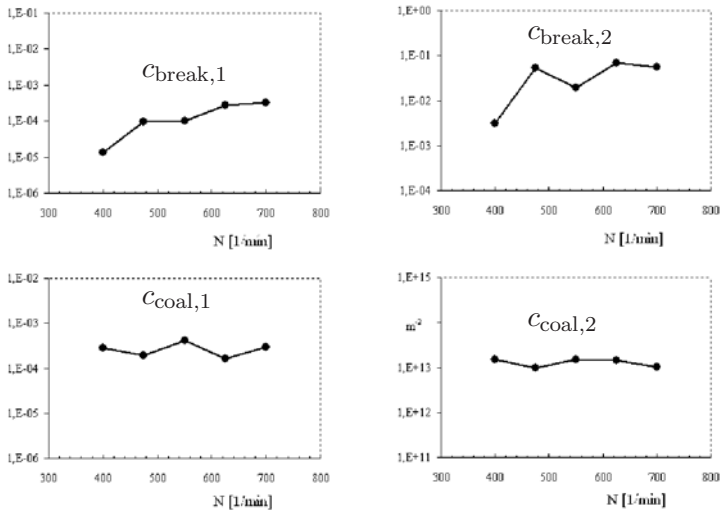


Figure 5.6: Dependence of the parameters occurring in the coalescence and breakage rate from [CT77] on the rotational speed N^* (taken from [GSP+05]). Top: parameters in breakage rate, bottom: parameters in coalescence rate.

Furthermore, the optimal parameters obtained for the five different rotational speeds were averaged, and the resulting values were used for the simulation with a rotational speed of 700 rounds per minute. For the resulting drop size distributions a much better agreement with experimental data could be achieved than for the simulation with the parameters optimized "only" for a rotational speed of 400 rounds per minute. However, a physical derivation of the model equations seems to be more reasonable.

- Also in [GWPK06a] (see also [GWPK06b] and [GWPK05]), the same dispersed system was considered. However, the volume fraction of toluene in water was varied from 5 to 50 %. As in [Weg04], the experiments were carried out for pH from 1 to 13 and rotational speeds from 400 to 700, whereas the numerical simulations with PARSIVAL were performed for the same range of rotational speeds, but only for a pH-value of 13.
  Different models for coalescence and breakage rates were considered. Results were presented for the two models, for which a reasonable parameter estimation was possible, namely the models from [CT77] and [AKK99]. In [GWPK06a], the model parameters were optimized with respect to the experimental drop size distributions for a rotational speed of 550 rounds per minute. Furthermore, a "correction" factor for the mean energy dissipation rate $\bar{\varepsilon}$ was introduced in order to take into account that the impeller zone has a stronger influence on the population dynamic phenomena (especially on breakage) than the other parts of the tank. By this, for both models, it was possible to obtain very good agreement between experimental and simulation results, not only for a rotational speed of 550, but also for 700

rounds per minute.
However, by changing the volume fraction from 10 to 30 %, but keeping the same set of parameters (which was fitted with respect to a volume fraction of 10 %), the simulation failed to give reasonable Sauter mean diameters.

The sizes of the drops were highly underestimated by the simulation. This shows that the volume fraction is another parameter that should be taken into account in the coalescence and breakage models.

- In [MGW+06] (see also [MPK06]), experimental investigations using a single drop breakage cell were carried out in order to analyze the breakage process in stirred liquid-liquid systems. The aim of the experimental investigations was to obtain information about the size and the number of daughter droplets formed by the breakage of a single mother drop. These data were then used to fit the two free parameters in the distribution of daughter drops proposed in [DO02b] (a Beta distribution, in which the two free parameters are defined as functions of a shape parameter and of the mean number of daughter drops resulting per breakage event) in order to get an enhanced distribution of daughter drops based on experimental data. This extended model was then implemented in Parsival, as well as the well-known model from [CT77].
  A comparison of the simulation results with experimental drop size distributions showed good agreement for both models, but an even better agreement for the extended model. This shows that the numerical simulations with Parsival can be used to improve models from the literature by optimizing the fitting parameters with respect to experimental investigations. Further information can be found in [MGW+06] and [MPK06].

5.2.2 Results for "More-Compartment Models"

It is well-known from experimental investigations as well as from numerical simulations that the energy dissipation rate is orders of magnitudes greater in the region around the impeller than in the bulk of the tank. Since coalescence and breakage are dependent on the (locally varying) energy dissipation rate, it is sensible to take these differences into account in the simulation of the drop size distributions.
This can be realized by dividing the tank into several zones and using different values for the energy dissipation rate in the corresponding coalescence and breakage rates. Such multi-compartment models have already been proposed in the literature, see, e. g., [AKK99], [AKKM02], [LAA02], [PB], [VDV02], or [WR05] (and also Remark 5.2). In the following, we will describe the realization of this approach as it has been considered within our project.

- In [Weg04], a two-compartment model was implemented. For this, the tank was divided into two zones, an impeller zone and a bulk zone, which were both assumed to be ideally mixed. In both zones, the same model was used for the description

of the coalescence and breakage phenomena, namely the model by [CT77], but different values for the mean energy dissipation rate. In the impeller region, the mean energy dissipation rate was set to a much higher value than in the bulk of the tank. The exchange between the two zones was modeled by two streams that had the same magnitude (which was determined by CFD calculations), but were flowing in opposite directions. Again, the parameters in the model were fitted for a rotational speed of 400 rounds per minute with the help of Parsival's parameter estimation procedure. This set of parameters was then also used for the simulation of higher rotational speeds.
The results were similar as for the one-zone model. For a rotational speed of 400 rounds per minute, the agreement between the simulation results and the measured drop size distributions was very good, whereas for rotational speeds of 550 or 700 rounds per minute only moderate agreement could be achieved. Again, a step in the rotational speed from 700 to 400 rounds per minute was simulated, and the drop size distributions obtained in experiments were reproduced very well for this case.

One can say that the results for the two-compartment model were better than those derived for the one-compartment model. The experimental data were reproduced better due to the more dynamic reaction of the system. This follows from the fact that in the two-compartment model, the differences in the turbulent flow field throughout the tank and their influence on the drop size distribution are taken into account.

The next step is a division of the tank into more than two compartments. In this case, the different energy dissipation rates for all zones as well as the volume flows between the zones (describing the exchange flows between the compartments) are needed, see Section 5.3.

- In [GSP+05], this approach of subdividing the stirred tank into several compartments was considered. Here, a more general approach of coupling the calculation of the drop size distributions with the simulation of the flow field in a stirred tank was discussed.

More details about this approach as well as about the realization of the coupling between the CFD code FeatFlow and the population balance solver Parsival can be found in the following section.

5.3 Realization of the Coupling between FeatFlow and Parsival

In this section, we will consider the coupling between the CFD code FeatFlow and the population balance solver Parsival. This coupling is realized as a so-called "one-way coupling", which means that the results of the CFD simulation are used for the

calculation of the drop size distributions, but the influence of the drops on the flow field is neglected. This simplification is justified for the considered application, since the densities of the dispersed and the continuous phase are almost equal and the drops are very small.

If we assume that the CFD code has implemented some kind of turbulence modeling, which means that it is capable of calculating not only (mean) velocity and pressure variables, but also turbulence parameters (from which the energy dissipation ε rate can be derived), the coupling with Parsival can be realized as follows.
In a first step, the flow field as well as the turbulence parameters are calculated by the CFD code using a very fine grid consisting of several millions of cells. If the same spatial discretization was also used for the calculation of the drop size distributions, a simulation with hundred classes (i. e., "discretization points" for the internal coordinate) would already lead to several hundred millions of cells, which is too costly. Thus, several (spatial) cells are combined to one compartment so that for each compartment the drop size distribution can be calculated with Parsival. Within these compartments, the drop size distribution is considered to be space-independent, and hence the compartments should be chosen in such a way that this is approximately satisfied.
As next step, a reactor consisting of N_K compartments is implemented in Parsival. In every time step, the mean energy dissipation rate $\bar{\varepsilon}_i$ for each compartment K_i is read from a file containing the CFD results such that the changes of the drop size distribution due to coalescence and breakage can be calculated. The mean energy dissipation rate $\bar{\varepsilon}_i(t)$ in compartment K_i is computed from the value $\varepsilon(x,t)$ that we get from the CFD simulation via

$$\bar{\varepsilon}_i(t) = \frac{1}{V_i} \int_{V_i} \varepsilon(x,t)\, dx, \tag{5.4}$$

where V_i denotes the volume of compartment K_i. Furthermore, the average volume flow $\dot{V}_{ij}(t)$ between two neighboring compartments K_i and K_j is calculated via

$$\dot{V}_{ij}(t) = \max\big\{\, 0,\ \int_{A_{ij}} \bar{u}(x,t) \cdot \mathfrak{n}_{ij}\, dx \,\big\}, \tag{5.5}$$

where $\bar{u}(x,t)$ is the mean velocity calculated by the CFD code, A_{ij} denotes the area of the face lying between the two compartments K_i and K_j, and $\mathfrak{n}_{ij}$ denotes the unit normal vector on A_{ij} in direction from compartment K_i to compartment K_j. With these volume flows the convective flow of the drop size distributions is modeled. In time step $t = t_{k+1}$, we get the new distribution $f_i(t_{k+1})$ in compartment K_i from

$$f_i(t_{k+1}) = f_i(t_k) \ + \sum_{j \text{ neighbor of } i} \frac{\dot{V}_{ji}(t_k)}{V_i}\, f_j(t_k) \ - \sum_{j \text{ neighbor of } i} \frac{\dot{V}_{ij}(t_k)}{V_i}\, f_i(t_k).$$

The left picture in Fig. 5.7 illustrates this way of realizing the coupling with turbulence model. The mean velocities $\bar{u}$ and the energy dissipation rates ε obtained by the CFD simulation are averaged so that mean volume flows $\dot{V}_{ij}$ and mean energy dissipation rates

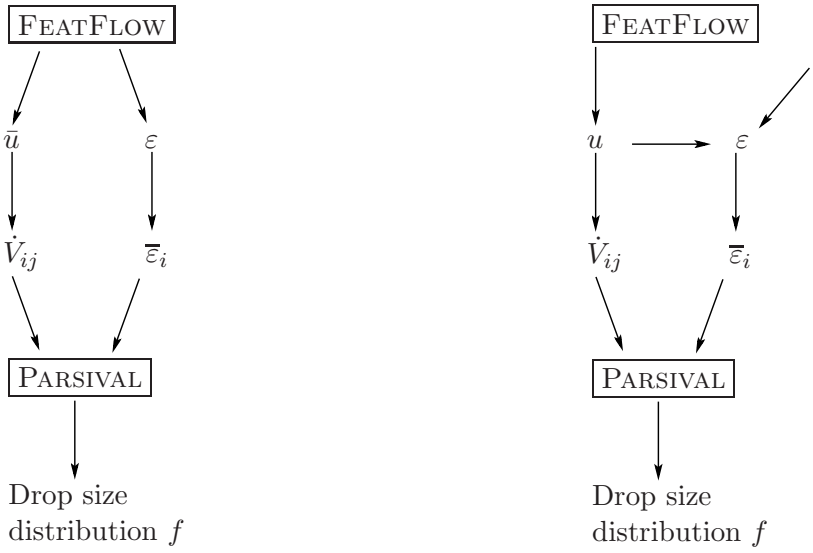


Figure 5.7: Realization of the coupling. Left: with turbulence model (i. e., the values for the energy dissipation rate ε are obtained from the CFD simulation), right: without turbulence model (i. e., the energy dissipation rates ε are calculated from the velocities u, which are obtained from the CFD simulation, or values from the literature or experimental investigations are used).

$\bar{\varepsilon}_i$ are derived, which are then used for the computation of the drop size distributions with Parsival.

However, as already discussed in Subsection 4.4.2, the adaption of the k-ε turbulence model to the moving boundary method in FeatFlow, which is needed for the simulation of the considered application, was still under investigation when this thesis was written. Thus, turbulence parameters, like the energy dissipation rate needed for the coupling, were not available from the CFD simulation, and values from the literature or experimental investigations had to be used.

Theoretically, if we do not face the difficulties discussed in the beginning of Chapter 4 (see page 46), the coupling between FeatFlow and Parsival could also be realized by using the velocities, which one obtains from the direct numerical simulation, for the calculation of the energy dissipation rates ε. In this case, the quantity ε can be calculated by its definition

$$\varepsilon := \frac{\nu}{2}\langle \| \nabla u' + \left(\nabla u'\right)^T \|_F^2 \rangle,$$

where the values of the fluctuations u' are given by

$$u' = u - \bar{u} \quad \text{with} \quad \bar{u} = \frac{1}{\mathcal{T}} \int_{t-\frac{\mathcal{T}}{2}}^{t+\frac{\mathcal{T}}{2}} u(\tau) d\tau \tag{5.6}$$

and the velocities u are the results of the direct numerical simulation.
However, in practice, this approach is usually very inaccurate, at least if post-processing

is used for the calculation of the energy dissipation rates from the simulated velocity field, which is then only given for a finite number of nodes. This typically leads to large approximation errors, in particular, in the calculation of the derivatives.

Therefore, for the simulation of the drop size distributions with Parsival, the results of the CFD simulation are only used for the calculation of the mean volume flows $\dot{V}_{ij}$ between the compartments, whereas for the mean energy dissipation rates $\bar{\varepsilon}_i$ values from the literature and experimental investigations were taken. Nevertheless, the coupling approach is realized in such a way that, as soon as the mean energy dissipation rates $\bar{\varepsilon}_i$ are available from the CFD simulation, they can easily be included in the model.

The two approaches how the coupling can be realized without turbulence model are shown schematically in the right picture in Fig. 5.7. In this case, the energy dissipation rates ε are calculated by using the results of the direct numerical simulation, or values from "outside", i. e., from the literature or experimental investigations, are taken. Once, the values for u and ε are available, we can proceed as in the case with turbulence model, i. e., $\dot{V}_{ij}$ and $\bar{\varepsilon}_i$ can be calculated and used for the simulation with Parsival. Note that the mean velocity $\bar{u}$ can simply be calculated from the velocity u, which one obtains from a direct numerical simulation, by averaging, see Eq. (5.6).

In the following subsections, we will give some more details how the coupling is realized for the considered application.

5.3.1 Division of the Tank into Compartments

As already mentioned before, we cannot use the same fine grid for the calculation of the drop size distributions as for the flow simulation. The reason for this is that the population balance equation is not only dependent on space and time, but also on at least one internal coordinate that characterizes the elements of the dispersed phase (like the drop volume V or the drop diameter d in the considered application). Since each additional coordinate increases the dimension of the problem by one, in many applications one uses the assumption of an ideally mixed tank. This means that the drop size distribution is assumed to be space-independent. However, due to the locally strong varying energy consumption throughout the tank, it is sensible to subdivide the domain at least into a few subdomains, the so-called compartments (which can be interpreted as a very coarse space discretization). In the considered application, this division is done according to the flow field that we get from the CFD simulation.

For the simulation with Parsival, the tank is divided into four compartments as shown in Fig. 5.8, where only one half of the tank is displayed. Compartment K_2, which lies directly around the stirrer, has the highest energy consumption, followed by compartment K_3, which is located in the same height, but in the outer part of the tank. Below and above these two compartments, respectively, there are the two other compartments K_1 and K_4, where the energy consumption is much smaller. Note that the division was done in such a way that the boundaries of the compartments coincide with the edges of the coarse grid, whenever this was possible without generating too irregular boundaries.

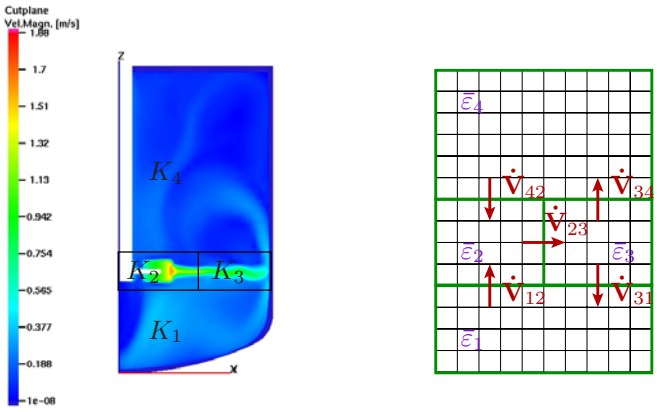


Figure 5.8: Left: division of the tank into four compartments $K_1, \ldots, K_4$ according to the velocity field. Right: Four-zone model with mean volume flows $\dot{V}_{ij}$ and mean energy dissipation rates $\bar{\varepsilon}_i$ drawn schematically.

In this way, the boundaries between all compartments coincide with edges of the coarse grid, except for the boundary between the second and the third compartment.

For the implementation of a corresponding compartment model in Parsival, the volumes of the compartments are needed. Thus, we give a mathematical description of the compartments:

$$\begin{aligned}
K_1 &: \quad 0 \le x_3 \le H_1, \\
K_2 &: \quad H_1 < x_3 \le H_2 \quad \text{and} \quad \sqrt{x_1^2 + x_2^2} \le R_1, \\
K_3 &: \quad H_1 < x_3 \le H_2 \quad \text{and} \quad \sqrt{x_1^2 + x_2^2} > R_1, \\
K_4 &: \quad H_2 < x_3 \le H_{\text{tank}}
\end{aligned}$$

with $H_1 = 0.04\,\text{m}$, $H_2 = 0.06\,\text{m}$, and $R_1 = 0.04\,\text{m}$. Since K_2, K_3, and K_4 are simply cylinders or hollow cylinders, respectively, their volumes are given by

$$\begin{aligned}
V_2 &= \pi R_1^2 (H_2 - H_1) = 1.005 \cdot 10^{-4}\,\text{m}^3, \\
V_3 &= \pi (R_2^2 - R_1^2)(H_2 - H_1) = 2.529 \cdot 10^{-4}\,\text{m}^3, \\
V_4 &= \pi R_2^2 (H_{\text{tank}} - H_2) = 1.590 \cdot 10^{-3}\,\text{m}^3,
\end{aligned} \tag{5.7}$$

where R_2 is the radius of the whole tank, i. e., $R_2 = {}^{D_{\text{tank}}}/_2 = 0.075\,\text{m}$. The volume of compartment K_1 is given by

$$V_1 = V_{\text{Klöpper}} + V_{\text{cyl}},$$

where $V_{\text{Klöpper}}$ and V_{cyl} are the volumes of the torispherical head and the cylindric part, respectively. The volume $V_{\text{Klöpper}}$ is calculated via integration over the function $\mathcal{K}^{-1}(x_3)$,

see Eq. (4.11) on page 57, describing the torispherical head:

$$\begin{aligned} V_{\text{Klöpper}} &= \int_0^{h_2} \pi \left[\mathcal{K}^{-1}(x_3)\right]^2 dx_3 \\ &= \int_0^{x_{3,s}} \pi \left[r_1^2 - (r_1 - x_3)^2\right] dx_3 + \int_{x_{3,s}}^{h_2} \pi \left[\frac{D_{\text{tank}}}{2} - r_2 + \sqrt{r_2^2 - (h_2 - x_3)^2}\right]^2 dx_3, \end{aligned}$$

where h_2 denotes the height of the torispherical head part. Calculation of the integral leads to

$$\begin{aligned} V_{\text{Klöpper}} &= \pi \left[r_1 x_3^2 - \frac{1}{3}x_3^3\right]_0^{x_{3,s}} + \pi \left[\left\{\left(\frac{D_{\text{tank}}}{2} - r_2\right)^2 + r_2^2 - h_2^2\right\} x_3 + h_2 x_3^2 - \frac{1}{3}x_3^3\right]_{x_{3,s}}^{h_2} \\ &+ \pi \left(\frac{D_{\text{tank}}}{2} - r_2\right) \left[(x_3 - h_2)\sqrt{r_2^2 - (h_2 - x_3)^2} - r_2^2 \arcsin\left(\frac{h_2 - x_3}{r_2}\right)\right]_{x_{3,s}}^{h_2}, \end{aligned}$$

where we have used the following formula from [Zei96], page 168f:

$$\begin{aligned} &\int \sqrt{\alpha_1 z^2 + \alpha_2 z + \alpha_3} dz \\ = \; &\frac{(2\alpha_1 z + \alpha_2)\sqrt{\alpha_1 z^2 + \alpha_2 z + \alpha_3}}{4\alpha_1} + \frac{4\alpha_1\alpha_3 - \alpha_2^2}{8\alpha_1} \int \frac{dz}{\sqrt{\alpha_1 z^2 + \alpha_2 z + \alpha_3}} \\ = \; &\frac{(2\alpha_1 z + \alpha_2)\sqrt{\alpha_1 z^2 + \alpha_2 z + \alpha_3}}{4\alpha_1} - \frac{4\alpha_1\alpha_3 - \alpha_2^2}{8\alpha_1} \frac{1}{\sqrt{-\alpha_1}} \arcsin\left(\frac{2\alpha_1 z + \alpha_2}{\sqrt{-4\alpha_1\alpha_3 + \alpha_2^2}}\right), \end{aligned}$$

which holds for $\alpha_1 < 0$ and $4\alpha_1\alpha_3 - \alpha_2^2 < 0$. Inserting the values for r_1, r_2, h_2, D_{tank}, and $x_{3,s}$ (see Subsection 4.2.1) gives $V_{\text{Klöpper}} = 3.359 \cdot 10^{-4}\,\text{m}^3$. Since V_{cyl} is given by $V_{\text{cyl}} = \pi \frac{D_{\text{tank}}^2}{4}(H_1 - h_2) = 1.996 \cdot 10^{-4}\,\text{m}^3$, we get

$$V_1 = 5.355 \cdot 10^{-4}\,\text{m}^3. \tag{5.8}$$

Thus, the volume of the whole tank is given by

$$V_{\text{tank}} = \sum_{i=1}^{4} V_i = 2.479 \cdot 10^{-3}\,\text{m}^3. \tag{5.9}$$

Since for the calculation of the mean volume flows $\dot{V}_{ij}$ in the following subsection we will also need the area of the faces A_{ij} lying between the compartments K_i and K_j, these values will be calculated in the following. The faces between the compartments K_1 and K_2 and between K_2 and K_4, respectively, are circles, and hence their areas are given by

$$A_{12} = A_{24} = \pi R_1^2 = 5.027 \cdot 10^{-3}\,\text{m}^2. \tag{5.10}$$

The faces between the compartments K_1 and K_3 as well as between K_3 and K_4 are annuli, and thus their areas are given by

$$A_{13} = A_{34} = \pi \left(\frac{(D_{\text{tank}})^2}{4} - R_1^2 \right) = 1.265 \cdot 10^{-2}\,\text{m}^2. \tag{5.11}$$

The face between the compartments K_2 and K_3 is given by the nappe of a cylinder. Its area is given by

$$A_{23} = 2\pi R_1 (H_2 - H_1) = 5.027 \cdot 10^{-3}\,\text{m}^2. \tag{5.12}$$

5.3.2 Calculation of the Mean Volume Flows $\dot{V}_{ij}$

The next step is the calculation of the mean volume flows between neighboring compartments from the velocities derived by the flow simulation. Theoretically, these volume flows $\dot{V}_{ij}$ can be calculated using Eq. (5.5), where $\bar{u}$ is given according to Eq. (5.6). However, in practice, we have to approximate the integral by a quadrature formula, i. e., by

$$\int_{A_{ij}} \bar{u}(x,t)\,\mathfrak{n}_{ij}\,dx \approx \sum_{\alpha=1}^{N_{ij}} \bar{u}_\alpha(t)\,\mathfrak{n}_{ij}\,A_{ij}^\alpha,$$

where N_{ij} is the total number of nodes on face A_{ij}. Here, $\bar{u}_\alpha$ denotes the mean velocity in node α and A_{ij}^α denotes the area of the part of the face A_{ij} that belongs to node α. If we assume the grid to be equidistant, then we can use the approximation

$$\int_{A_{ij}} \bar{u}(x,t)\,\mathfrak{n}_{ij}\,dx \approx \sum_{\alpha=1}^{N_{ij}} \bar{u}_\alpha(t)\,\mathfrak{n}_{ij}\,A_{ij}^\alpha = \frac{A_{ij}}{N_{ij}} \sum_{\alpha=1}^{N_{ij}} \bar{u}_\alpha(t)\,\mathfrak{n}_{ij}. \tag{5.13}$$

The resulting volume flows calculated by Eq. (5.13), where we have plugged in the values from (5.10)-(5.12), are shown in Fig. 5.9 (the diamonds). Note that the mean velocities $\bar{u}_\alpha$ were calculated from the CFD results by averaging (according to Eq. (4.12) on page 83, which can be interpreted as the discrete version of Eq. (5.6)). The size of the time interval, over which the velocities were averaged, was chosen as $\Delta t \approx {}^1\!/_{12}\,\text{s}$, which corresponds to about one revolution. The calculation of the mean volume flows from the output of the CFD calculation and the generation of an appropriate data file, which could be read by Parsival, was done with Perl [Per].
From Fig. 5.8 it can be seen that, due to the required mass conservation, the volume flows have to fulfill the following conditions

$$\begin{aligned} \dot{V}_{31} - \dot{V}_{12} &= 0, \\ \dot{V}_{12} - \dot{V}_{23} + \dot{V}_{42} &= 0, \\ \dot{V}_{23} - \dot{V}_{31} - \dot{V}_{34} &= 0, \\ \dot{V}_{34} - \dot{V}_{42} &= 0. \end{aligned}$$

However, for the computed values in Fig. 5.9, these conditions are not satisfied.
There are several reasons for this. First of all, the grid used for the CFD simulation is not

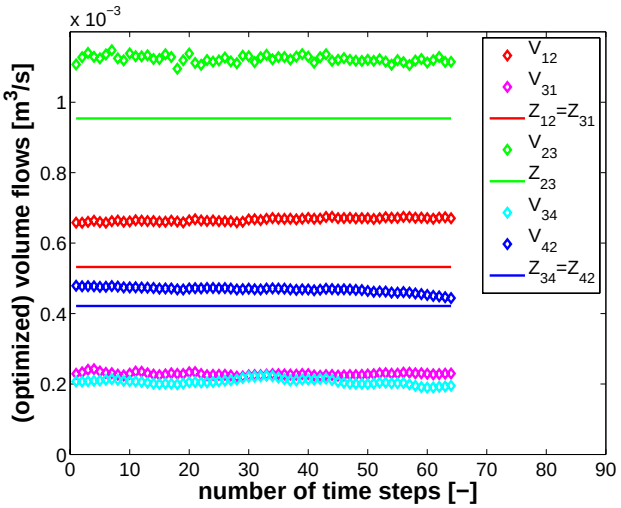


Figure 5.9: Volume flows between the four compartments. Values from CFD calculation (diamonds) and optimized values (solid lines).

equidistant and, therefore, the assumption used in Eq. (5.13) is not fulfilled. However, this does not explain this significant deviation. Second, even if the grid was equidistant, the volume flows would probably not be calculated correctly, since FeatFlow does not control the conservation of mass for every cutplane, but only over the total domain.

To improve the calculation of the volume flows, one could use a more accurate interpolation formula, which takes the distance between two neighboring grid points into account so that the assumption of an equidistant grid could be avoided. However, the calculation of these distances is computationally too expensive for an unsorted list of about 2.8 million nodes. Another approach would be to choose the compartments for the calculation of the drop size distributions in such a way that they are composed of several cells of the coarse grid used for the CFD simulation. In this case, one would have the same number of nodes in each element of the coarse grid (due to the regular refinement of the grid). Thus, one could easily calculate the distance between two neighboring grid points (or at least an average of it) within each element of the coarse grid. However, in the considered application, this approach cannot be used, since it is not possible to combine the elements of the coarse CFD grid in such a way that this leads to sensible compartments. An alternative would be to combine several cells of one of the finer grids used in the CFD simulation to obtain appropriate compartments. However, this would lead to very complicated boundaries between the different compartments, and, therefore, the calculations of the normal compounds of the velocities would be very complex.
This analysis shows that the best way would have been to create the coarse grid for the CFD simulation in such a way that an appropriate compartment model could easily be created by the combination of several coarse grid elements. However, the creation of such a "clever" grid consisting of quadrilateral elements is very sophisticated.

Since all the existent alternatives are either very complex and time-consuming or should have been applied in advance, it was decided to use the computed volume flows for the coupling and to fit them by applying a least-squares method. To do this, for the given volume flows $\dot{V}_{ij}^k$ between two compartments K_i and K_j at time t_k, we determine the optimal volume flows $\mathcal{Z}_{ij}^k$ by solving the following minimization problem:

$$\|\mathcal{Z}^k - \dot{V}^k\|_2 = \min! \quad \text{s.t.} \quad \mathcal{H}\mathcal{Z}^k = 0, \tag{5.14}$$

where $\mathcal{Z}^k = [\mathcal{Z}_{12}^k, \mathcal{Z}_{23}^k, \mathcal{Z}_{31}^k, \mathcal{Z}_{34}^k, \mathcal{Z}_{42}^k]^T$, $\dot{V}^k = [\dot{V}_{12}^k, \dot{V}_{23}^k, \dot{V}_{31}^k, \dot{V}_{34}^k, \dot{V}_{42}^k]^T$ for all time points t_k, and $\mathcal{H}$ is given by

$$\mathcal{H} = \begin{bmatrix} 1 & -1 & 0 & 0 & 1 \\ 0 & 1 & -1 & -1 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix}.$$

Since the volume flows $\dot{V}_{ij}^k$ do not change much in time (due to the constant stirrer speed), see Fig. 5.9, it is more reasonable to solve the following minimization problem instead of (5.14),

$$\|\mathcal{Z} - \dot{V}^k\|_2 = \min! \quad \text{for all time points } t_k \quad \text{s.t.} \quad \mathcal{H}\mathcal{Z} = 0, \tag{5.15}$$

where $\mathcal{Z} = [\mathcal{Z}_{12}, \mathcal{Z}_{23}, \mathcal{Z}_{31}, \mathcal{Z}_{34}, \mathcal{Z}_{42}]^T$ denotes the vector of volume flows $\mathcal{Z}_{ij}$, which are optimal with respect to the simulation results for all time points t_k. The latter formulation has the advantage that there are more data available for the calculation of the volume flows and that as much information as possible is used for the calculation of the optimal values.

Fig. 5.9 shows the optimized volume flows (the solid lines), which were determined by means of the Matlab-function `fmincon`, depending on the given volume flows $\dot{V}_{ij}^k$ (the diamonds) obtained by a FeatFlow simulation. Note that the axis of abscissae gives the number of time steps starting at some time point, when the flow field is already fully developed. The length of one time step is given by $\Delta t = 7.96 \cdot 10^{-3}$ s. These optimized volume flows are in the following used for the simulation with Parsival. Note that these volumes flows correspond to a Reynolds number of Re = 30,000, whereas the simulations with Parsival will be done for a rotational speed of 700 rounds per minute, which corresponds to a Reynolds number of about Re = 29,167. Thus, the volume flows should be scaled by a factor of $^{29,167}/_{30,000} \approx 0.972$. However, due to the low accuracy of the volume flows derived by the CFD simulation, this factor can also be neglected.

Remark 5.5. *Clearly, if possible, one would prefer to use the "original" values derived by the CFD simulation directly in* Parsival *and let the program do the optimization. Unfortunately, this is not possible, since in* Parsival *one can only prescribe a part of the volume flows (namely exactly as many as that there does not occur any contradiction concerning the conservation of mass) and let the program adjust the remaining ones during the simulation. (If more volume flows are prescribed, this will lead to a violation of the mass conservation.) A "real" optimization, however, cannot be performed with* Parsival*, but has to be done by hand and "outside" of* Parsival.

parameter	value
ρ_c	$998.3\ {}^{\mathrm{kg}}/_{\mathrm{m}^3}$
ρ_d	$866.9\ {}^{\mathrm{kg}}/_{\mathrm{m}^3}$
ν_c	$1 \cdot 10^{-6}\ {}^{\mathrm{m}^2}/_{\mathrm{s}}$
ν_d	$0.63 \cdot 10^{-6}\ {}^{\mathrm{m}^2}/_{\mathrm{s}}$
σ	$32 \cdot 10^{-3}\ {}^{\mathrm{N}}/_{\mathrm{m}}$
χ	0.1
c_V	${}^{\pi}/_{6}$

parameter	value
V_1	$5.355 \cdot 10^{-4}\,\mathrm{m}^3$
V_2	$1.005 \cdot 10^{-4}\,\mathrm{m}^3$
V_3	$2.529 \cdot 10^{-4}\,\mathrm{m}^3$
V_4	$1.590 \cdot 10^{-3}\,\mathrm{m}^3$
$\dot{V}_{12} = \dot{V}_{31}$	$5.32 \cdot 10^{-4}\ {}^{\mathrm{m}^3}/_{\mathrm{s}}$
$\dot{V}_{34} = \dot{V}_{42}$	$4.22 \cdot 10^{-4}\ {}^{\mathrm{m}^3}/_{\mathrm{s}}$
$\dot{V}_{23}$	$9.54 \cdot 10^{-4}\ {}^{\mathrm{m}^3}/_{\mathrm{s}}$

Table 5.1: Different parameters used for the simulation with Parsival. Left: physical parameters of the two phases, right: volumes of and volume flows between the compartments.

5.3.3 Implementation of the Compartment Model in Parsival

In the following, we will describe the implemented model of the stirred tank reactor that has been used for the calculation of the drop size distributions with Parsival.

The stirred tank model used in Parsival consists of the following components:

- Four continuous reactors, which model the four compartments $K_1, \ldots, K_4$, see Fig. 5.8. As volumes for the reactors the values given by Eqs. (5.7) and (5.8) are prescribed. Furthermore, we have defined three control streams, namely $\dot{V}_{23}$, $\dot{V}_{31}$, and $\dot{V}_{42}$, each belonging to the reactor, of which they flow out, see Fig. 5.8. (These and other values used for the simulation with Parsival are summarized in Table 5.1.) The temperature in the compartments was set to 20°C. However, for the specification of the temperature, not only a value, but an input file was used, which prescribes the value of 20°C at certain time points. These time points refer to exactly the time points, at which the mean volume flows (and if applicable also the mean energy dissipation rates) are read from a file, namely the so-called measured-data file (or short `mdf`-file). This circumstantial definition of the temperature is required in order to force Parsival to perform the calculation of the drop size distributions at least in these time points so that all information from the `mdf`-file is read during the simulation.

- Four fluids modeling the continuous phases in the four compartments (i. e., each fluid refers to one reactor). Since in the considered application, water is used as continuous phase, the physical characteristics of water are prescribed for the four fluids. Thus, a density of $\rho_c = 998.3\,{}^{\mathrm{kg}}/_{\mathrm{m}^3}$ is used.

- Four distributions modeling the drop size distributions, i. e., the dispersed phases in the different compartments (again, each distribution refers to one of the four reactors). In all compartments, the same start distribution is used, namely an

experimental drop size distribution for a rotational speed of 400 rounds per minute and a pH-value of 13, which has been measured at time $t = 1\,\mathrm{s}$. However, the start distributions are scaled according to the total mass of the compartment they belong to, so that the volume fraction of 10 % toluene in water is prescribed in all compartments. Since toluene is used as dispersed phase, a density of $\rho_d = 866.9\,{}^{\mathrm{kg}}/_{\mathrm{m}^3}$ is prescribed. The shape factor is set to ${}^{\pi}/_{6}$, which means that the drops are assumed to be spherical.

- Five streams, defined both as volume and as exit streams. The streams belong to the reactor, of which they flow out, see Fig 5.8. For two of the five streams, namely $\dot{V}_{12}$ and $\dot{V}_{34}$, values are prescribed, the other three, namely $\dot{V}_{23}$, $\dot{V}_{31}$, and $\dot{V}_{42}$, are simply set to zero. Since they have been defined as control streams in the reactors they refer to, their values are adapted during the simulation so that the volumes of the reactors are kept constant.

- The following groups of coefficients:

  - model parameters occurring in the coalescence and breakage rates,
  - physical parameters of the two phases (density, surface tension, etc.),
  - mean energy dissipation rates in the different compartments,
  - geometry parameters,
  - volume fraction of the dispersed phase,
  - other parameters.

  Note that, if the coupling is realized using a turbulence model for the simulation of the flow field, the mean energy dissipation rates in the different compartments have to be modeled by functions, which can be overwritten by the actual data read from the `mdf`-file (see Subsection 5.4.3).

- The following modules used to describe all the processes in the stirred tank reactor:

  - four coalescence modules (one corresponding to each distribution, i. e., to each reactor),
  - four modules modeling the source terms due to breakage (one corresponding to each distribution / reactor),
  - four modules modeling the sink terms due to breakage (one corresponding to each distribution / reactor),
  - five modules describing the exchange of the continuous phases between the different compartments (one corresponding to each stream),
  - five modules describing the exchange of the drop size distributions between the different compartments (one corresponding to each stream).

In all coalescence and breakage modules, the model from [CT77] is used for the coalescence and breakage rates, see Eq. (2.24) on page 20 and Eq. (2.29) on page 23, respectively. However, note that for the collision frequency the corrected formula (2.25) is used, which leads to a slightly different coalescence rate, see page 20. For the distribution of daughter drops, we have implemented the model from [Rit02], see Eq. (2.33) on page 27.
In principle, the same coalescence and breakage rates and the same distribution of daughter drops are used in all four compartments. However, different values for the mean energy dissipation rates $\bar{\varepsilon}_i$ are chosen in the different compartments K_i, $i = 1, \ldots, 4$, see Fig. 5.8.

As already mentioned briefly before, during the simulation, the mean volume flows between the compartments (and if applicable also the mean energy dissipation rates in the compartments) are read from a file, in which the values at different time points are saved. The calculation of the drop size distributions is performed according to this information. That all the values in the `mdf`-file are used for the calculation of the drop size distributions is made sure by the "dummy" file, in which the temperature is prescribed at exactly these (relevant) time points. Without using such a "dummy" file, it could happen that Parsival performs a time step with such a big step size that one (or even more) of the values in the `mdf`-file is jumped over and thus the corresponding values in the `mdf`-file are not taken into account.

Further information about the implemented model will be given in the presentation of the results in the next section.

Remark 5.6. *Besides the presented approach of implementing a compartment model in* Parsival*, the coupling between* FeatFlow *and* Parsival *could also be realized by simulating N_K compartments "separately" (i. e., by modeling each compartment as an ideally mixed reactor) in* Parsival *and implementing the exchange flows between the compartments by hand outside of* Parsival*. Note that, if this approach is used, the external control of the volume flows must also make sure that the sum of all volume flows is zero, i. e., that mass is conserved during the simulation. Under the assumption that N_K versions of* Parsival *could be started simultaneously, this approach could be used, having the advantage that each simulation with* Parsival *would be relatively "easy" (and thus fast) and that all the* Parsival*-simulations could be performed in parallel. However, additional time for the "outer loop" must be taken into account. Depending on the implementation of the whole model, this can lead to a higher amount of computing time than the one needed for the presented approach. In addition, M. Wulkow, the developer of* Parsival*, recommended to use the presented approach.*

A further automatization of the coupling approach presented before can be derived by analyzing the structure of the ASCII-file containing all the data needed to start a simulation with Parsival (see Section 5.1). By understanding the structure of this file, one can implement a program that automatically creates such a file, for example by means of Perl (in the same way as the `mdf`-files containing the results of the CFD simulation

have been created).
However, for a simulation with the version of PARSIVAL that was used for the investigations presented here, such an automatization is needless, since it cannot be started without pressing the "run"-button inside the program itself. Thus, we refrain from giving any further details concerning this approach.

Remark 5.7. *In [KBM06], a coupling of* PARSIVAL *with the commercial CFD code* FLUENT*, namely via the so-called "software-level integration platform"* CHEOPS *(see [Che] for further information), has been implemented and analyzed (see also [KBG+05]). In this paper, the solution of the coupled problem of crystallization and fluid dynamics by means of software integration has been considered.*
In this approach, the population balance and crystallization kinetics are modeled in PARSIVAL*, whereas the CFD simulations are done in* FLUENT*. As in the coupling approach presented in this thesis, the population balance model is not solved in each grid cell of the fluid dynamics calculation, but only for a much smaller number of compartments, which are assumed to be ideally mixed. Thus, within these compartments averaged values are used for the simulation.*
The integration platform CHEOPS *already mentioned above is then used to establish the communication between the individual tools and to provide data exchange and overall convergence of the coupled system. (More information about* CHEOPS *can be found in [KBG+05], [SYvWM04] and references therein.) However, as in the coupling approach presented here, also in [KBM06], only a one-way coupling is realized, i. e., only the influence of the fluid dynamics on the population balance is considered, whereas the effects of the particle size distribution on the fluid dynamics are neglected.*

5.4 Results of the Coupling and Comparison with Experimental Data

In this section, we will present the results of the coupling approach. As long as we do not explicitly state the contrary, the model described in the previous subsection has been used for the simulation, see Table 5.1 for the volumes of and the volume flows between the compartments as well as for the physical parameters of the two phases. Further parameters used for the simulation can be found in Table 5.2.
As already mentioned before, the model from [CT77] is implemented for the coalescence and breakage rates, whereas for the distribution of daughter drops the formula from [Rit02] is taken. For the model parameters occurring in the coalescence and breakage rates, the values from [GWPK06a] are used. These parameters are summarized in the left Table 5.2. Note that these values are optimized for the same liquid-liquid system (water-toluene), for the same volume fraction of the dispersed phase (10% toluene), and the same pH-value (pH=13, i. e., a coalescence-hindered system) as in the considered application, but for a different rotational speed (550 rounds per minute instead of 700) and for a one-compartment model instead of a multi-compartment model (which can also lead to different values for the parameters, see [Weg04]). Furthermore, in [GWPK06a],

parameter	value
$c^{CT}_{\text{coal},1}$	$1.5 \cdot 10^{-4}$
$c^{CT}_{\text{coal},2}$	$2.56 \cdot 10^{12}$
$c^{CT}_{\text{break},1}$	$6.14 \cdot 10^{-4}$
$c^{CT}_{\text{break},2}$	$5.7 \cdot 10^{-2}$

parameter	value
dispersed phase fraction χ	10 %
pH value	13
rotational speed N^*	700 $^1/_{\text{min}}$
power number Ne	3.8
mean energy dissipation rate $\bar{\varepsilon}$	0.761 $^{\text{m}^2}/_{\text{s}^3}$

Table 5.2: Parameters used for the simulation with Parsival. Left: parameters in the coalescence and breakage rates, taken from [GWPK06a], right: other parameters.

a "correction" factor for the mean energy dissipation rate $\bar{\varepsilon}$ has been used, which clearly also has an influence on the four parameters. Nevertheless, these values will be used in the considered coupling approach, since the aim of the investigations is to get an agreement between the experimental and the simulated drop size distributions without using "perfect" parameters, but by improving the model so that there is no need to adapt the model parameters as long as only minor modifications in the system are made. If this is possible, the improved model can be used for predictive simulations.

In the following, we will present simulation results for the four-compartment model. Besides different test cases, we will also consider a four-compartment model representing the stirred tank that has been used in the experimental investigations carried out by our project partners at the Institute for Process Engineering at the TU Berlin. For this case, we will in Subsection 5.4.2 compare the simulation results with the corresponding experimental data.

5.4.1 Simulation Results for the Four-Compartment Model with Constant Volume Flows between the Compartments

In this subsection, we will present simulation results for the four-compartment model with constant volume flows. First, we will show the resulting drop size distributions for the case that there are no volume flows between the four compartments. By this test simulation, the influence of different values for the energy dissipation rate on the drop size distributions can be investigated. Second, we will use the optimized volume flows from Subsection 5.3.2 for the simulation. We will see that for volume flows of this order of magnitude there is hardly any difference between the resulting drop size distributions in the different compartments. Thus, we will also present results for a simulation with smaller volume flows at the end of this subsection.

As already discussed in Chapter 4, the simulation with FeatFlow does not provide any values for the energy dissipation rate ε at the moment. Thus, for the first test cases presented here, we have chosen the mean energy dissipation rates $\bar{\varepsilon}_i$ in the different compartments K_i, $i = 1, \ldots, 4$, as follows:

$$\bar{\varepsilon}_1 = 0.329\,\bar{\varepsilon}, \quad \bar{\varepsilon}_2 = 12\,\bar{\varepsilon}, \quad \bar{\varepsilon}_3 = 2\,\bar{\varepsilon}, \quad \bar{\varepsilon}_4 = 0.3\,\bar{\varepsilon}, \tag{5.16}$$

where $\bar{\varepsilon}$ denotes the overall mean value of the energy dissipation rate ε throughout the tank. By this choice, the turbulence distribution is as expected (and, at least roughly, as in the literature, see [AKK99]). Furthermore, the condition

$$\sum_{i=1}^{4} \bar{\varepsilon}_i V_i = \bar{\varepsilon}\, V_{\text{tank}}$$

is fulfilled. To determine $\bar{\varepsilon}$, we have used the formula (see [Kra03])

$$\bar{\varepsilon} = \frac{\text{Ne}\,(N^*)^3 (D_{\text{stirrer}})^5}{V_{\text{tank}}},$$

where Ne denotes the power (or Newton) number, N^* is the rotational speed, D_{stirrer} is the diameter of the stirrer, and V_{tank} is the volume of the tank. For the considered application this formula gives $\bar{\varepsilon} = 0.761\ {}^{\text{m}^2}\!/_{\text{s}^3}$, where we have inserted $N^* = {}^{70}\!/_{6}\,{}^{1}\!/_{\text{s}}$ for the rotational speed, $D_{\text{stirrer}} = 0.05\,\text{m}$ for the impeller diameter, and Ne = 3.8 for the power number (see, e. g., [Weg04]). Furthermore, the volume of the tank is given by Eq. (5.9).

Simulation without volume flows

For this test case, we have set the volume flows between the different compartments to zero. Fig. 5.10 shows the resulting drop size distributions in the four compartments after $t = 0$, $t = 60\,\text{s}$, $t = 300\,\text{s}$, $t = 600\,\text{s}$, $t = 1{,}500\,\text{s}$, and $t = 4{,}800\,\text{s}$. Here, the cumulative number distributions $Q_{0,i}$ in the compartments K_i, $i = 1, \ldots, 4$, are plotted versus the diameter d of the drops. Note that the value $Q_{0,i}(d)$, $i = 1, \ldots, 4$, gives the total number of drops of sizes smaller than or equal to d per cubic meter.
The top left picture of Fig. 5.10 shows the start distributions (due to the normalization the same for all four compartments), for which we have chosen a measured drop size distribution for a rotational speed of 400 rounds per minute. This start distribution can be used for the $N^* = 700\ {}^{1}\!/_{\text{min}}$-simulation, since one can assume that the transient drop size distributions for a rotational speed of $N^* = 700\ {}^{1}\!/_{\text{min}}$ "go through" this value at some time point in the beginning.
The top right picture in Fig. 5.10 displays the simulated drop size distributions after $t = 60\,\text{s}$. It shows that the distributions in the first and in the fourth compartment are moved towards larger drop sizes. There are less, and, therefore (due to conservation of mass), larger drops in these compartments after $t = 60\,\text{s}$. The reason for this is that we have chosen relatively small values for the mean energy dissipation rates in these compartments. By this choice, almost only coalescence takes place. On the other hand, in the second compartment, where the mean energy dissipation rate has been chosen much higher than the average value $\bar{\varepsilon}$, only breakage occurs. (The reason for this is that the drops are not able to stay in contact as long as coalescence needs to take place.) This effect results in a significant change of the distribution to smaller drop sizes. Therefore, in the second compartment, there are many more and, hence, smaller drops. Also in

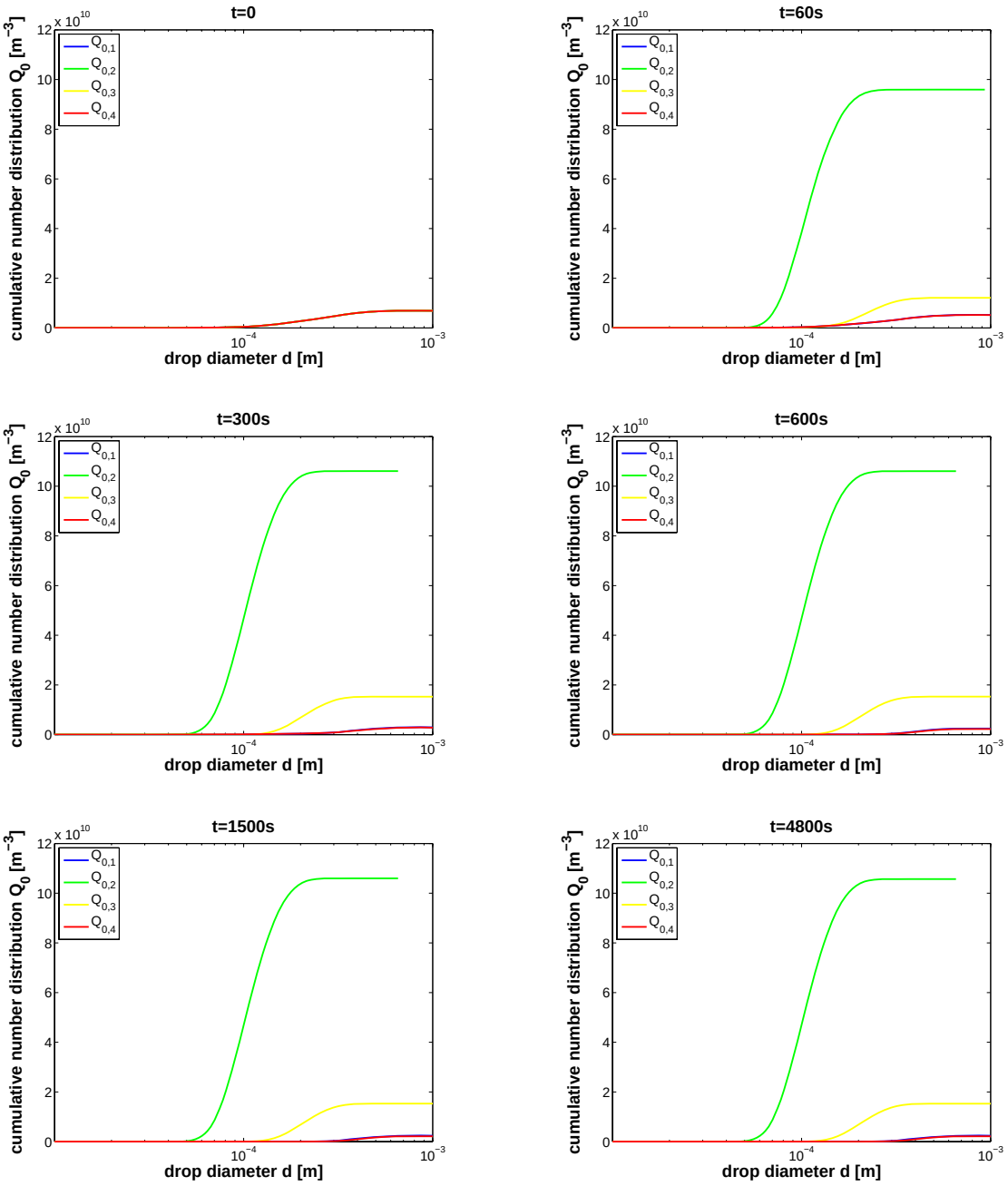


Figure 5.10: Drop size distributions in the four compartments at different time points for a simulation without volume flows. Top left: $t = 0$, top right: $t = 60\,\mathrm{s}$, middle left: $t = 300\,\mathrm{s}$, middle right: $t = 600\,\mathrm{s}$, bottom left: $t = 1{,}500\,\mathrm{s}$, bottom right: $t = 4{,}800\,\mathrm{s}$.

compartment K_3, there are more and smaller drops after $t = 60\,\mathrm{s}$ than in the beginning. Here, breakage dominates, since the mean energy dissipation rate in this compartment is twice as high as the overall mean value $\bar{\varepsilon}$.
The pictures in the middle and at the bottom of Fig. 5.10 show the simulated drop size distributions after 5, 10, 25, and 80 minutes. They demonstrate that, after some time, a stationary drop size distribution is reached in all compartments. Note that, in the second and third compartment, this stationary distribution is reached faster than in the other two compartments. However, after $t = 1{,}500\,\mathrm{s}$, one cannot recognize any further changes in any of the distributions. Note that, although the distributions do not change any more after some time, this does not mean that there is no coalescence or breakage, but only that the number of breaking drops balances the number of coalescing drops.

Fig. 5.11 shows the transient behavior of the drop size distributions in the different compartments. Here we have again plotted the cumulative number distributions $Q_{0,i}$ versus the diameter d. However, this time, we have, for each compartment K_i, $i = 1, \ldots, 4$, put different time points together in one graph so that the changes in the drop size distributions with time can be observed better. (Note the different scales in the axis of ordinates.) Fig. 5.11 confirms the findings described above. One can observe that the drop size distributions in the first and in the fourth compartment are moved towards larger drop sizes (and thus less drops can be found in these compartments) due to the relatively low values for $\bar{\varepsilon}_1$ and $\bar{\varepsilon}_4$. On the other hand, there are much more and smaller drops in the other two compartments, where the values for the mean energy dissipation rate have been chosen higher than the average value $\bar{\varepsilon}$. Furthermore, one can recognize that in these two compartments, the stationary distribution is reached much faster than in the other two compartments. (In the second and third compartment only three colors can be detected in the pictures, which means that after $t = 300\,\mathrm{s}$ the drop size distributions do not change any more, whereas between five and six colors can be seen in the other two graphs, which corresponds to stationary drop size distributions after about 900–1,500 s.)

Fig. 5.12 displays the Sauter mean diameter d_{32} versus the time t in the four compartments. The Sauter mean diameter d_{32} is defined as the ratio of the third to second moments (see Section 5.6) for the case that the diameter d is used as internal coordinate. Here, it can be observed that in all four compartments the "final" Sauter mean diameters are reached after less than 20 minutes. As already expected from the previously presented drop size distributions and according to the prescribed values for the mean energy dissipation rates $\bar{\varepsilon}_i$, the Sauter mean diameters in the second and in the third compartment decrease, whereas the Sauter mean diameters in the other two compartments increase, until a steady state is reached. Furthermore, the graph depicts the anti-proportionality of the Sauter mean diameter and the mean energy dissipation rate: the larger the value for $\bar{\varepsilon}_i$ (i. e., the higher the turbulence), the smaller the Sauter mean diameter in the corresponding compartment K_i (i. e., the smaller the average drop size).

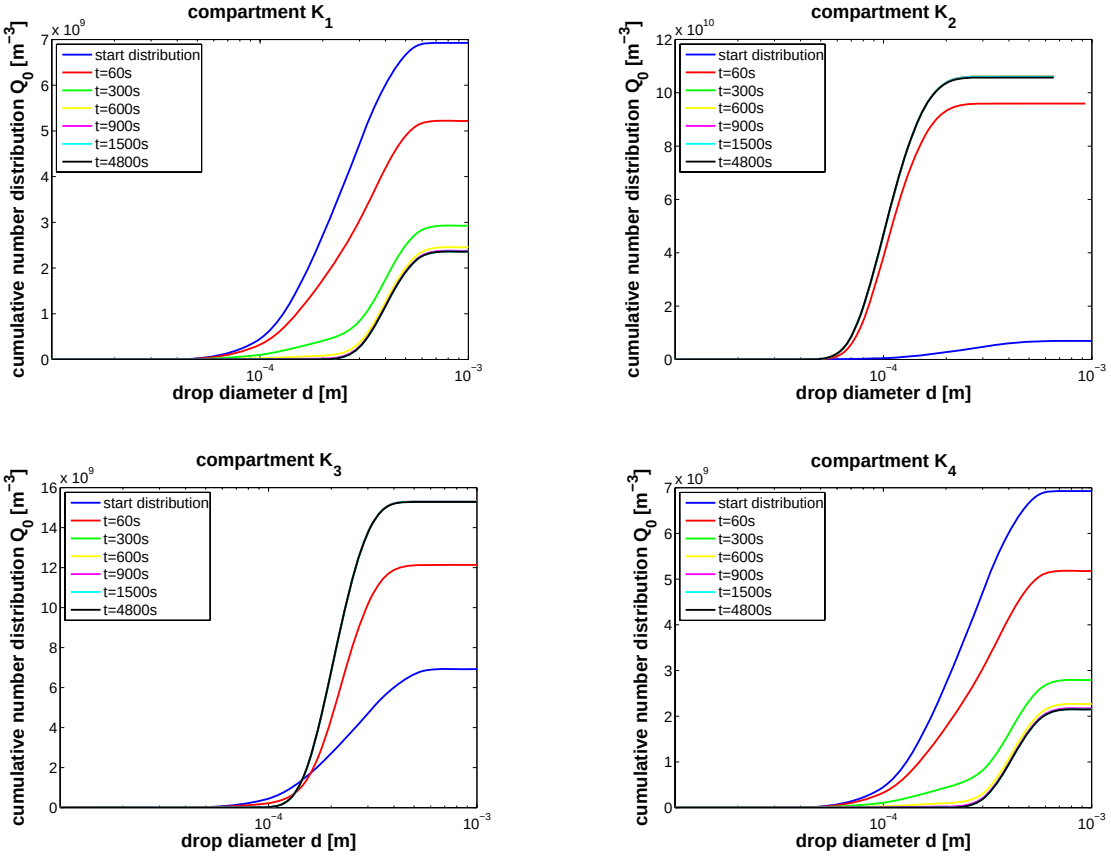


Figure 5.11: Drop size distributions in the different compartments at time points $t = 0$, $t = 60\,\mathrm{s}$, $t = 300\,\mathrm{s}$, $t = 600\,\mathrm{s}$, $t = 900\,\mathrm{s}$, $t = 1{,}500\,\mathrm{s}$, and $t = 4{,}800\,\mathrm{s}$ for a simulation without volume flows. Top left: compartment 1, top right: compartment 2, bottom left: compartment 3, bottom right: compartment 4.

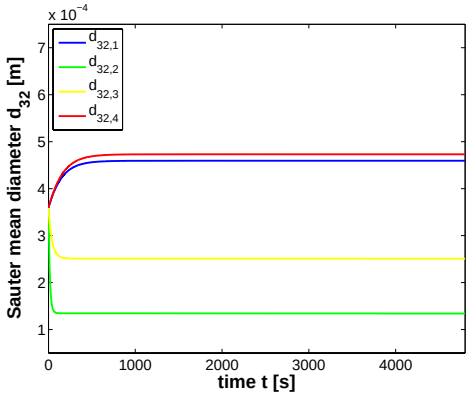


Figure 5.12: Sauter mean diameter versus time in the different compartments for a simulation without volume flows.

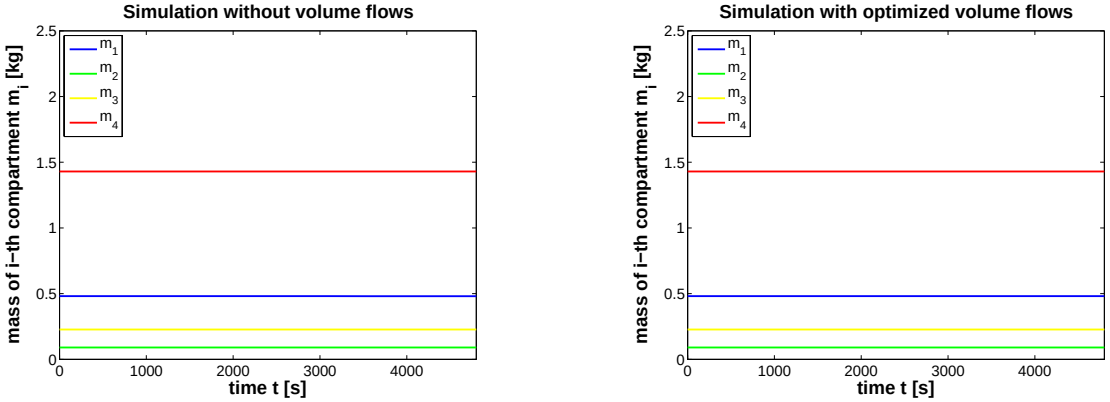


Figure 5.13: Conservation of mass during the simulation with PARSIVAL. Left: simulation without volume flows, right: simulation with optimized volume flows.

The left picture in Fig. 5.13 demonstrates that mass is conserved during the simulation. This shows that the discretization of the internal coordinate is realized in such a way that the total volume of the drops is conserved during coalescence and breakage (see Section 5.1).

Simulation with optimized volume flows $\mathcal{Z}_{ij}$

In the following, we will present the simulation results, which have been obtained using the optimized volume flows calculated in Subsection 5.3.2. In this case, the volume flows between the compartments are read from a file containing the calculated values $\mathcal{Z}_{ij}$, see Eq. (5.15). However, in PARSIVAL, only two of the five volume flows have to be prescribed, the other three are calculated by the program in such a way that

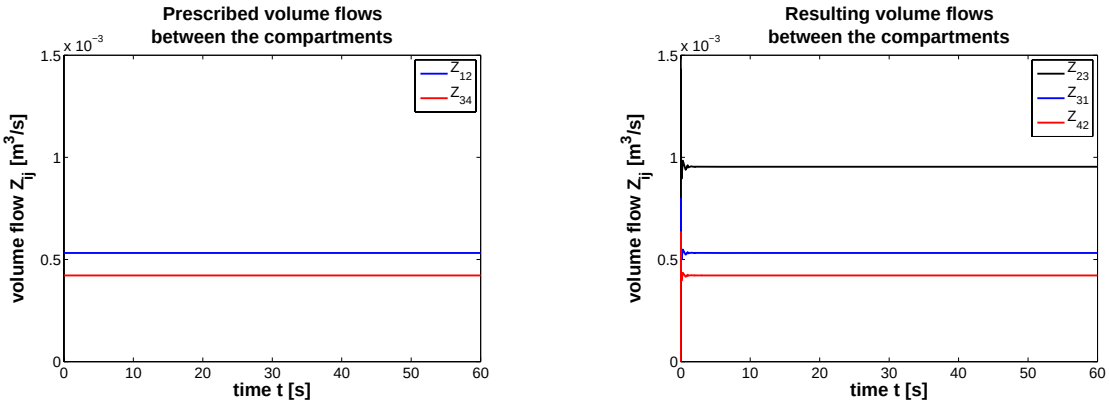


Figure 5.14: Volume flows between the different compartments. Left: prescribed volume flows $\mathcal{Z}_{12}$ and $\mathcal{Z}_{34}$, right: resulting volume flows $\mathcal{Z}_{23}$, $\mathcal{Z}_{31}$, and $\mathcal{Z}_{42}$ calculated by Parsival.

mass is conserved, see Fig. 5.14 as well as the description of the implemented stirred tank reactor in Parsival in Subsection 5.3.3. The left picture in Fig. 5.14 shows the prescribed volume flows $\mathcal{Z}_{12}$ and $\mathcal{Z}_{34}$ between the first and the second and the third and the fourth compartment, respectively. As already mentioned before, the remaining three volume flows are computed by Parsival during the simulation. The right picture in Fig. 5.14 demonstrates that these volume flows take the "right" values after a few iterations so that the constraint in Eq. (5.15) is fulfilled. (Note that only $t = 60\,\mathrm{s}$ are plotted in Fig. 5.14, because, otherwise, the oscillations in the beginning could not even be observed.)

Fig. 5.15 shows the resulting drop size distributions after $t = 0$, $t = 60\,\mathrm{s}$, $t = 300\,\mathrm{s}$, $t = 600\,\mathrm{s}$, $t = 1{,}500\,\mathrm{s}$, and $t = 4{,}800\,\mathrm{s}$ for the simulation with optimized volume flows. One observes that, in all compartments, the distributions are almost the same. The reason for this is that the volume flows are so high that they lead to a total exchange of the drop size distributions between the different compartments. Clearly, spatial variations in the drop size distributions can only occur if coalescence and breakage are at least as fast as the circulation rates in the system.
However, if one compares the drop size distributions in the four compartments after $t = 60\,\mathrm{s}$ with the prescribed start distributions, one can see that there are more (and thus smaller) drops for $t = 60\,\mathrm{s}$. This means that, in total, there are more breaking than coalescing drops. Thus, although the volumes of the compartments K_2 and K_3, where breakage dominates due to the relatively high energy dissipation rates there, are much smaller than the volumes of the other two compartments, where coalescence dominates, the breakage rates in the second and third compartment are so high that the smaller volumes are compensated for. This increase in the number of drops is still observable for larger values of t. However, after some time, a steady state is reached and, thus, for $t \geq 1{,}500\,\mathrm{s}$, no further changes in the drop size distributions can be detected.

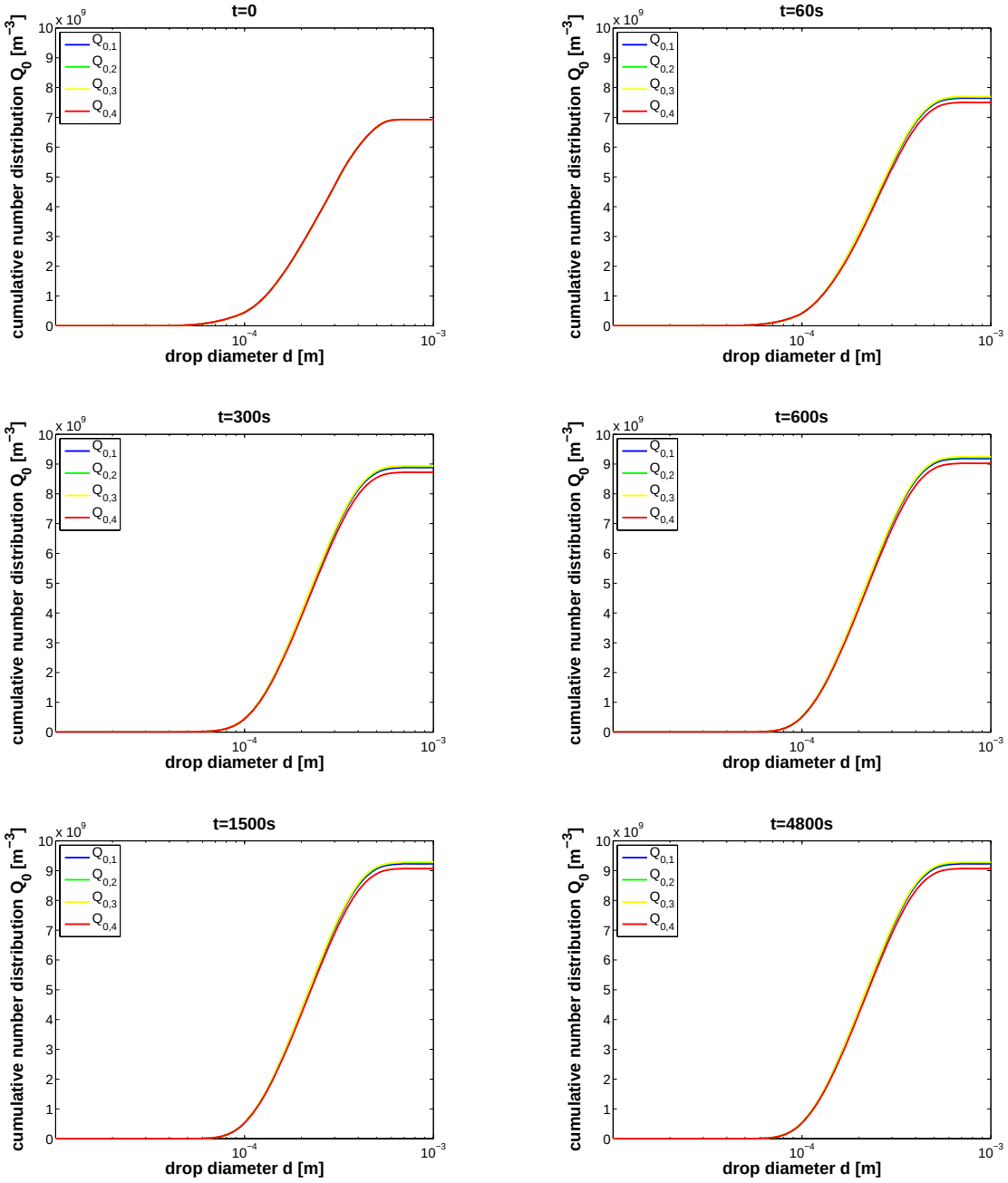


Figure 5.15: Drop size distributions in the different compartments at different time points for a simulation with optimized volume flows. Top left: $t = 0$, top right: $t = 60\,\mathrm{s}$, middle left: $t = 300\,\mathrm{s}$, middle right: $t = 600\,\mathrm{s}$, bottom left: $t = 1{,}500\,\mathrm{s}$, bottom right: $t = 4{,}800\,\mathrm{s}$.

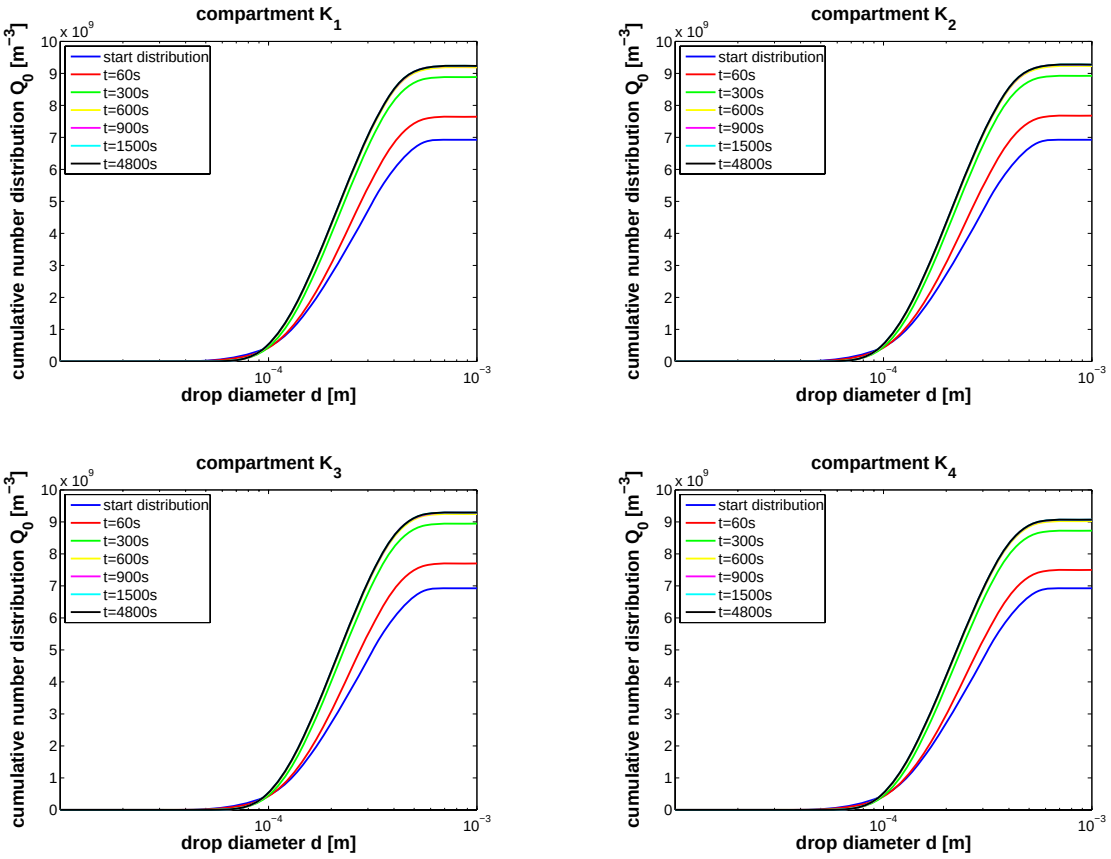


Figure 5.16: Drop size distributions in the different compartments at time points $t = 0$, $t = 60\,\mathrm{s}$, $t = 300\,\mathrm{s}$, $t = 600\,\mathrm{s}$, $t = 900\,\mathrm{s}$, $t = 1{,}500\,\mathrm{s}$, and $t = 4{,}800\,\mathrm{s}$ for a simulation with optimized volume flows. Top left: compartment 1, top right: compartment 2, bottom left: compartment 3, bottom right: compartment 4.

The findings described above are supported by Fig. 5.16, which displays the transient behavior of the drop size distributions in the different compartments. As before, we have plotted the cumulative number distributions $Q_{0,i}$, $i = 1, \ldots, 4$, versus the diameter d, but, as in Fig. 5.11, by putting different time points together in one graph so that the changes in the drop size distribution with time can be recognized better.
As already stated before, almost no differences in the drop size distributions can be detected between the different compartments. In addition, one can observe that in all compartments the distributions reach a steady state after about 900 seconds, i. e., after about 15 minutes.

Fig. 5.17 shows the Sauter mean diameter d_{32} versus the time t in the four compartments for the simulation with optimized volume flows, for which only very small differences can be observed. Since for this case already the drop size distributions have been almost

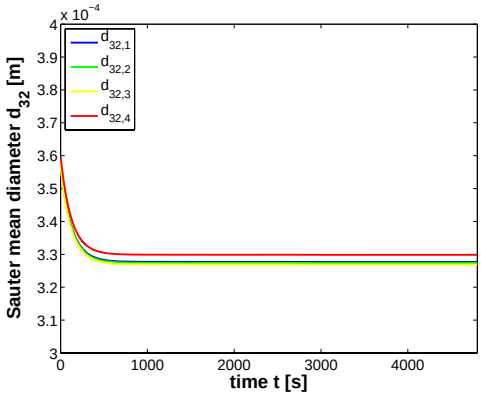


Figure 5.17: Sauter mean diameter versus time in the different compartments for a simulation with optimized volume flows.

equal, this could be expected.

In the right picture of Fig. 5.13, the masses of the different compartments are plotted versus the time. This shows that also for the simulation with volume flows mass is conserved.

Simulation with scaled volume flows

For this test case, we have scaled the optimized volume flows by a factor of $1/100$ so that the exchange between the compartments is not as high as before, and, thus, the interaction of the volume flows and the coalescence and breakage processes, both influencing the drop size distributions in the different compartments, can be seen better.
As for the simulation with optimized volume flows, only two of the five volume flows were prescribed, and the remaining three were calculated by PARSIVAL so that mass is conserved. Since the volume flows are exactly equal to those shown in Fig. 5.14, except for their scaling by the factor of $1/100$, we refrain from showing them here.

Fig. 5.18 shows the simulated drop size distributions in the four compartments after $t = 0$, $t = 60\,\text{s}$, $t = 300\,\text{s}$, $t = 600\,\text{s}$, $t = 1{,}500\,\text{s}$, and $t = 4{,}800\,\text{s}$. The top left picture in Fig. 5.18 displays the start distributions. In the top right picture, where the drop size distributions after $t = 60\,\text{s}$ are depicted, one can observe that, for the simulation with scaled volume flows, only in the fourth compartment less drops can be found, whereas in all other compartments there are more drops than in the beginning. As for the simulation without volume flows, there are the most (and also the smallest) drops in the second compartment (due to the highest value for the mean energy dissipation rate prescribed there). However, this time, an almost as large number of drops as in the second compartment can be found in the third compartment. The reason for this is that the drops of compartment K_2 are directly transported to compartment K_3. This also explains why, in total, the number of drops in the second compartment is smaller than

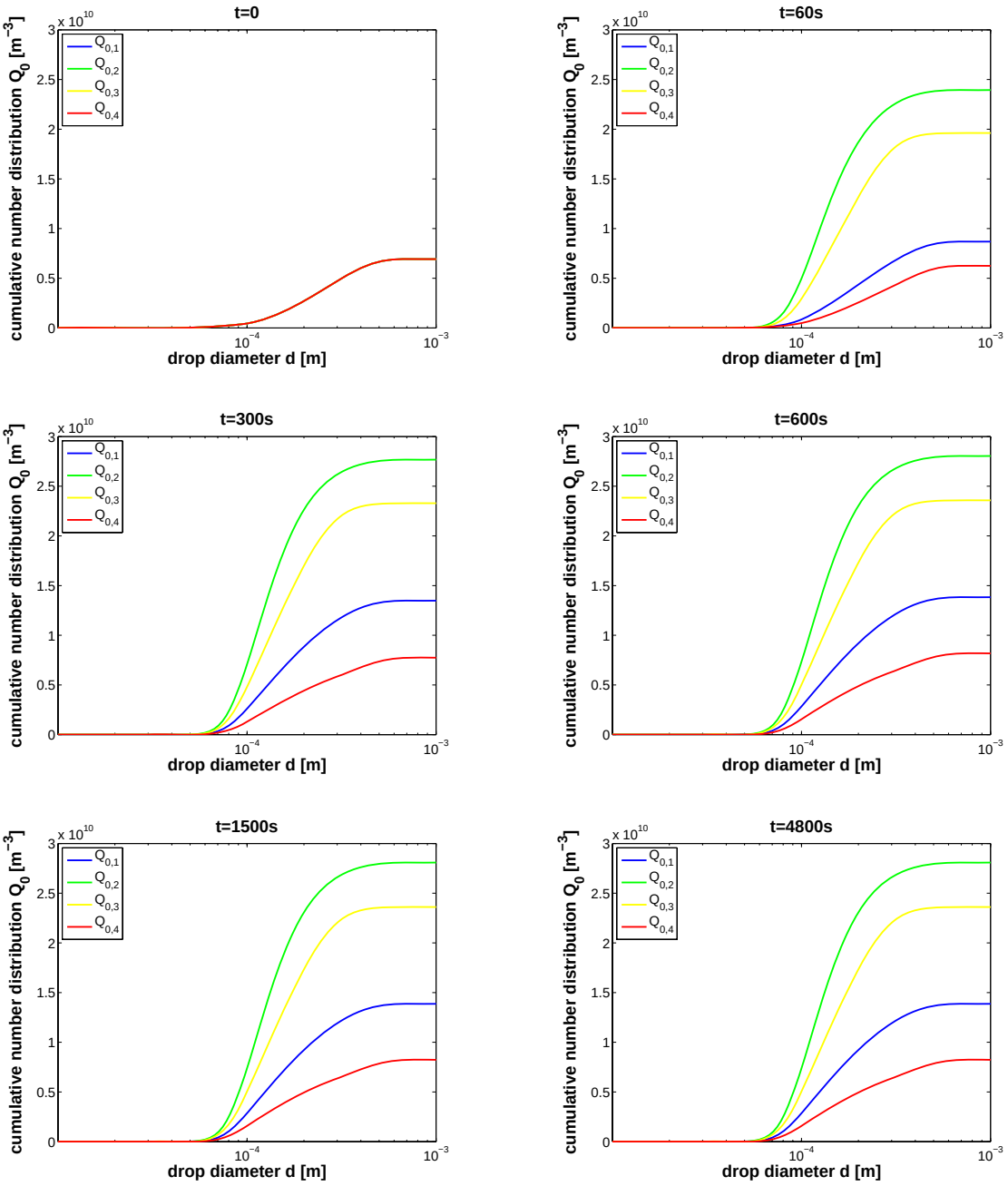


Figure 5.18: Drop size distributions in the four compartments at different time points for a simulation with scaled volume flows. Top left: $t = 0$, top right: $t = 60\,\mathrm{s}$, middle left: $t = 300\,\mathrm{s}$, middle right: $t = 600\,\mathrm{s}$, bottom left: $t = 1{,}500\,\mathrm{s}$, bottom right: $t = 4{,}800\,\mathrm{s}$.

for the simulation without volume flows. Furthermore, one observes a relatively large number of drops in the first compartment, especially compared to the number of drops in the fourth compartment, where almost the same value for the mean energy dissipation rate has been prescribed. This significant difference in the drop size distributions in the two compartments can be explained by the fact that compartment K_4 is much bigger than compartment K_1 and, thus, the incoming volume flow, which transports the relatively many and small drops from compartment K_3, does not carry so much weight there. In addition, the volume flow from compartment K_3 to the compartment K_4 is smaller than the one to compartment K_1.
The pictures in the middle and at the bottom of Fig. 5.18 show that after some time the number of drops increases in all compartments until the drop size distributions reach a steady state after about 600–900 seconds (see also Fig. 5.19).

Fig. 5.19 shows the same results, but this time, the drop size distributions are plotted separately for each compartment and together for different time points. By this, one can observe that the number of drops in the second and in the third compartment first increases very fast, then the increase becomes slower, and finally a steady-state is reached. In compartment K_1 also a movement of the number density function towards smaller drop sizes can be detected. However, the increase there is not as fast as in the second and in the third compartment, especially not in the beginning. Furthermore, one can observe that in the fourth compartment, the number of drops first decreases and then increases.

Fig. 5.20 shows the Sauter mean diameter d_{32} versus the time t in the four compartments for the simulation with scaled volume flows. The first thing that should be noticed is that the Sauter mean diameters in the second and in the third compartment are almost equal (which could, at least roughly, already be expected from the findings presented before). Furthermore, as also already expected from the considerations of the drop size distributions, the Sauter mean diameters in the second and in the third compartment decrease much faster than the one in the first compartment. However, also in the first compartment, the Sauter mean diameter decreases. Furthermore, Fig. 5.20 shows that the Sauter mean diameter in the fourth compartment first increases strongly and then decreases again, but to a value that is still larger than the Sauter mean diameter in the beginning. This non-monotone behavior of the Sauter mean diameter can be explained by the interaction of prescribing different values for the mean energy dissipation rates in the different compartments with choosing the volume flows between the compartments of such an order of magnitude that they do not lead to a total exchange of the drop size distributions. The different magnitude of the energy dissipation rates leads, on one hand, to coalescence-dominated and, on the other hand, to breakage-dominated compartments. In interaction with the choice of the volume flows, this leads to a mixture of larger and smaller drops (coming from the coalescence-dominated and breakage-dominated compartments, respectively).

Also for the simulation with scaled volume flows, mass is conserved during the simulation. However, since the masses in the four compartments are exactly the same as for the two cases considered before (see Fig. 5.13), we will not show them here.

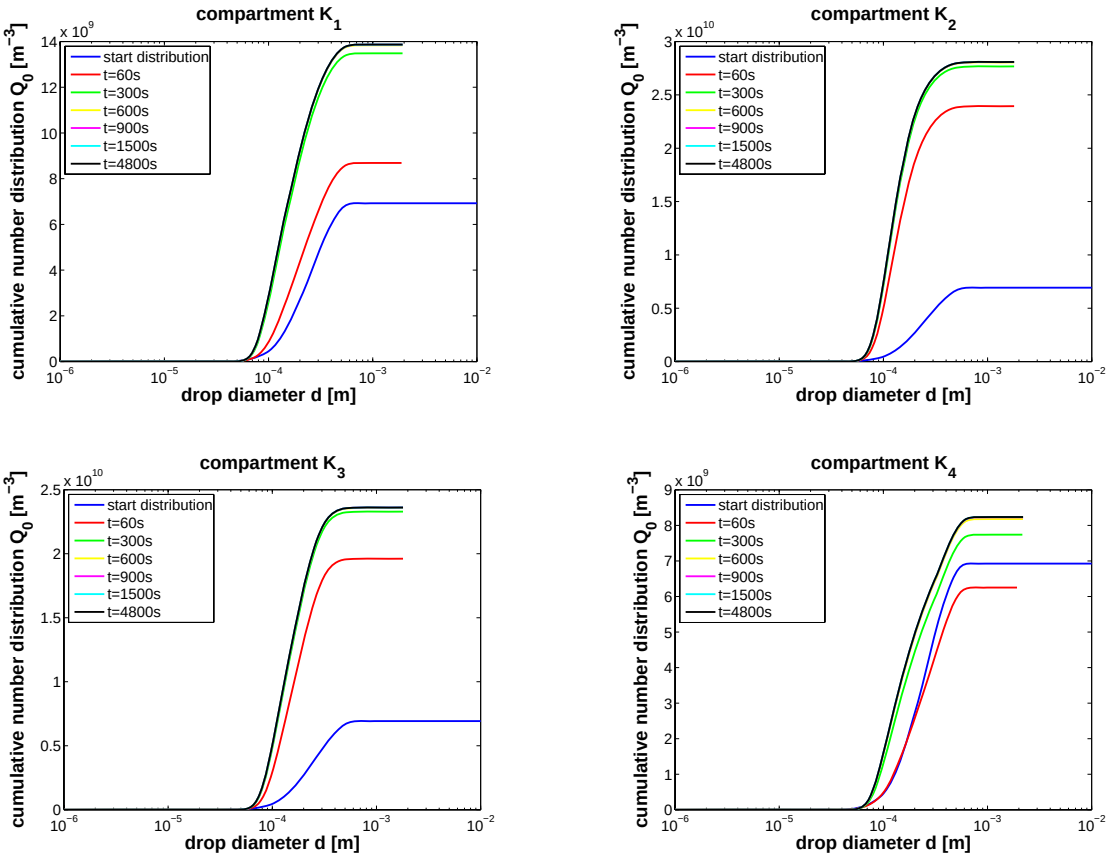


Figure 5.19: Drop size distributions in the different compartments at time points $t = 0$, $t = 60\,\mathrm{s}$, $t = 300\,\mathrm{s}$, $t = 600\,\mathrm{s}$, $t = 900\,\mathrm{s}$, $t = 1{,}500\,\mathrm{s}$, and $t = 4{,}800\,\mathrm{s}$ for a simulation with scaled volume flows. Top left: compartment K_1, top right: compartment K_2, bottom left: compartment K_3, bottom right: compartment K_4.

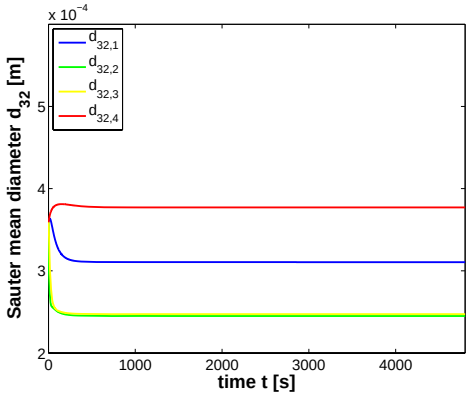


Figure 5.20: Sauter mean diameter versus time in the different compartments for a simulation with scaled volume flows.

At the end of this subsection, we will show all steady-state drop size distributions in the four compartments for the three different configurations presented above in one figure, namely Fig. 5.21. Note that we have used the same scale in all pictures so that the differences between the three cases can be seen better.
The top left picture in Fig. 5.21 shows the start distribution, which is the same for all cases and in all compartments. The top right picture displays the final distributions after $t = 4{,}800\,s$ in the different compartments for the case that the simulation is done without volume flows between the compartments. Here, and especially in comparison to the other cases, one can observe the large increase of the number of drops in the second compartment due to the much higher energy dissipation rate prescribed there. In the bottom left picture, the final distributions in the different compartments after a simulation with the optimized volume flows from Eq. (5.15) are shown. One can observe that the order of magnitude of these volume flows is so high that there is a total exchange of the drop size distributions between the four compartments. Furthermore, a slight increase in the total number of drops can be detected for that case. In the bottom right picture the resulting drop size distributions for the simulation with the scaled volume flows are depicted. These volume flows are not so large that they lead to a total exchange of the distributions. However, they lead to a more "balanced" distribution of the drops over the compartments.

5.4.2 Comparison with Experimental Data

In the following, we will compare the drop size distributions that have been obtained by the simulation using the four-compartment model with optimized volume flows (see previous subsection) with experimental drop size distributions. The corresponding measurements were carried out at the Institute for Process Engineering at the TU Berlin. The experimental data result from pictures that have been taken in the middle-plane between two baffles and at height $x_3 = 5\,\mathrm{cm}$ (i. e., in the impeller mid-plane). Since

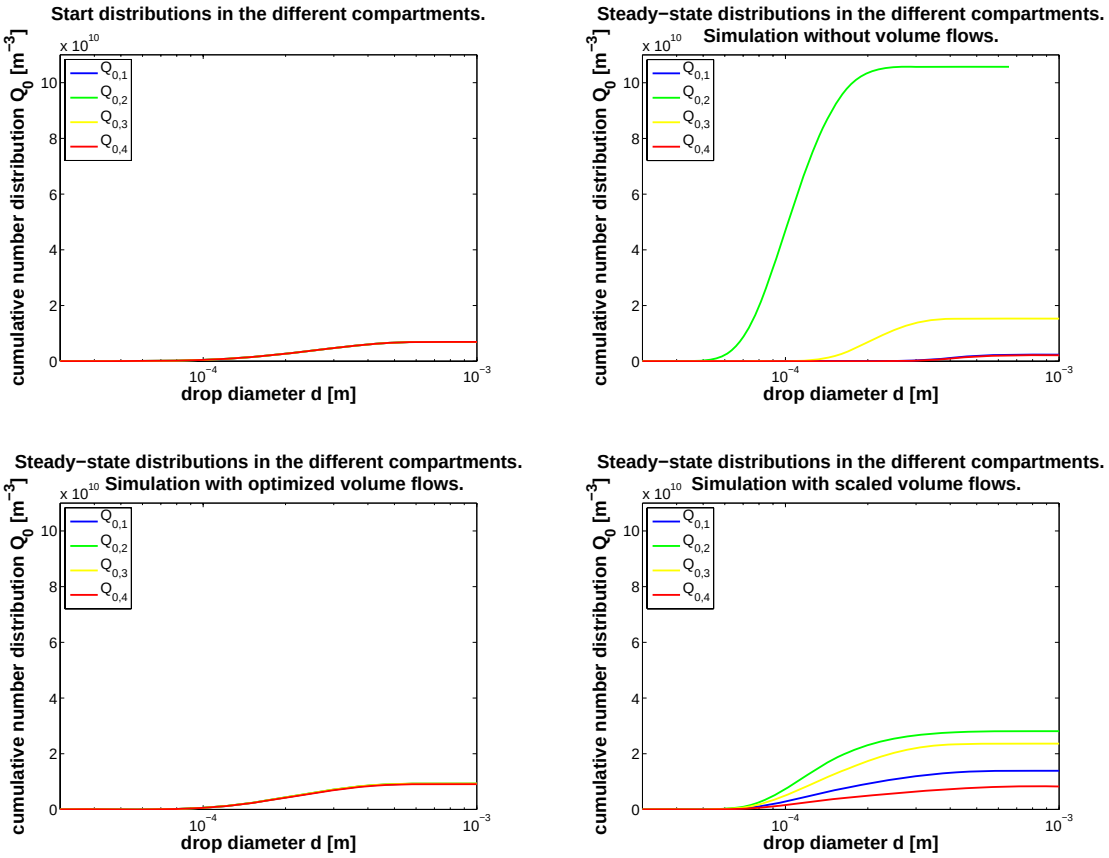


Figure 5.21: Drop size distributions in the different compartments. Top left: start distributions, top right: final distributions after $t = 4{,}800\,s$ for a simulation without volume flows between the compartments, bottom left: final distributions after $t = 4{,}800\,s$ for a simulation with optimized volume flows $\mathcal{Z}_{ij}$, bottom right: final distributions after $t = 4{,}800\,s$ for a simulation with scaled volume flows.

the experimental investigations provide only normalized drop size distributions, we have transformed the simulated drop size distributions (by dividing them by the total number of drops) so that they can be compared with the experimental data.

Fig. 5.22 shows a comparison of the experimental drop size distributions with the simulation results for the four-compartment model as well as for a corresponding one-compartment model. At first sight, one observes only moderate agreement between the experimental and the simulated drop size distributions, both for the one- and for the four-compartment model. Nevertheless, the simulation results for the four-compartment model are closer to the experimental findings, in particular, for small drop sizes.

However, the simulation of 4,800 s using the one-compartment model took about eleven hours, whereas more than 40 hours were needed for the same simulation using the four-compartment model. Note that all simulations were performed on an Intel Pentium M processor with 1.5 GHz and a 1 GB memory. PARSIVAL was run under Linux using the Windows-emulation WINE. Since the computing times for the four-compartment model are much higher than those for the one-compartment model, it is questionable whether the relatively low improvement in the resulting drop size distributions justifies these higher computational costs.

Fig. 5.23 shows experimental and simulated Sauter mean diameters, both for the one- and for the four-compartment model. Here, one can observe that both, the simulation results for the one- and for the four-compartment model, cannot reproduce the experimental Sauter mean diameters. That the Sauter mean diameters for the four-compartment model are even worse compared to the one obtained by the one-compartment model, although the drop size distributions for the four-compartment model were closer to the experimental data, can be explained by the fact that only the small drop sizes are reproduced better by the four-compartment model, which have only a very low influence on the Sauter mean diameter. On the other hand, the only slightly better agreement of the one-compartment model in the range of the largest occurring drop sizes has a much higher influence on the Sauter mean diameter.

At the end of this subsection, it should again be mentioned that the simulation results are highly dependent on the choice of the model parameters.
Theoretically, a set of parameters should be valid for different configurations as long as only minor changes in the system are made (for example, if only the rotational speed or the volume fraction of the dispersed phase are changed and as long as the modifications are not too large). However, it is clear that modifications that lead to major changes in the system (like such a small rotational speed that we do not have a fully turbulent flow field any more or such a high volume fraction of the dispersed phase that the influence of the drops on the flow field cannot be neglected any more) cannot be handled with the same set of parameters. Nevertheless, as long as the underlying assumptions made for the modeling of the system are fulfilled, the adaption parameters should not depend on quantities that are already included in the model, like the energy dissipation rate or the volume fraction of the dispersed phase. The “fitting” parameters should rather be used to account for dependencies on quantities that have not been included in the model yet,

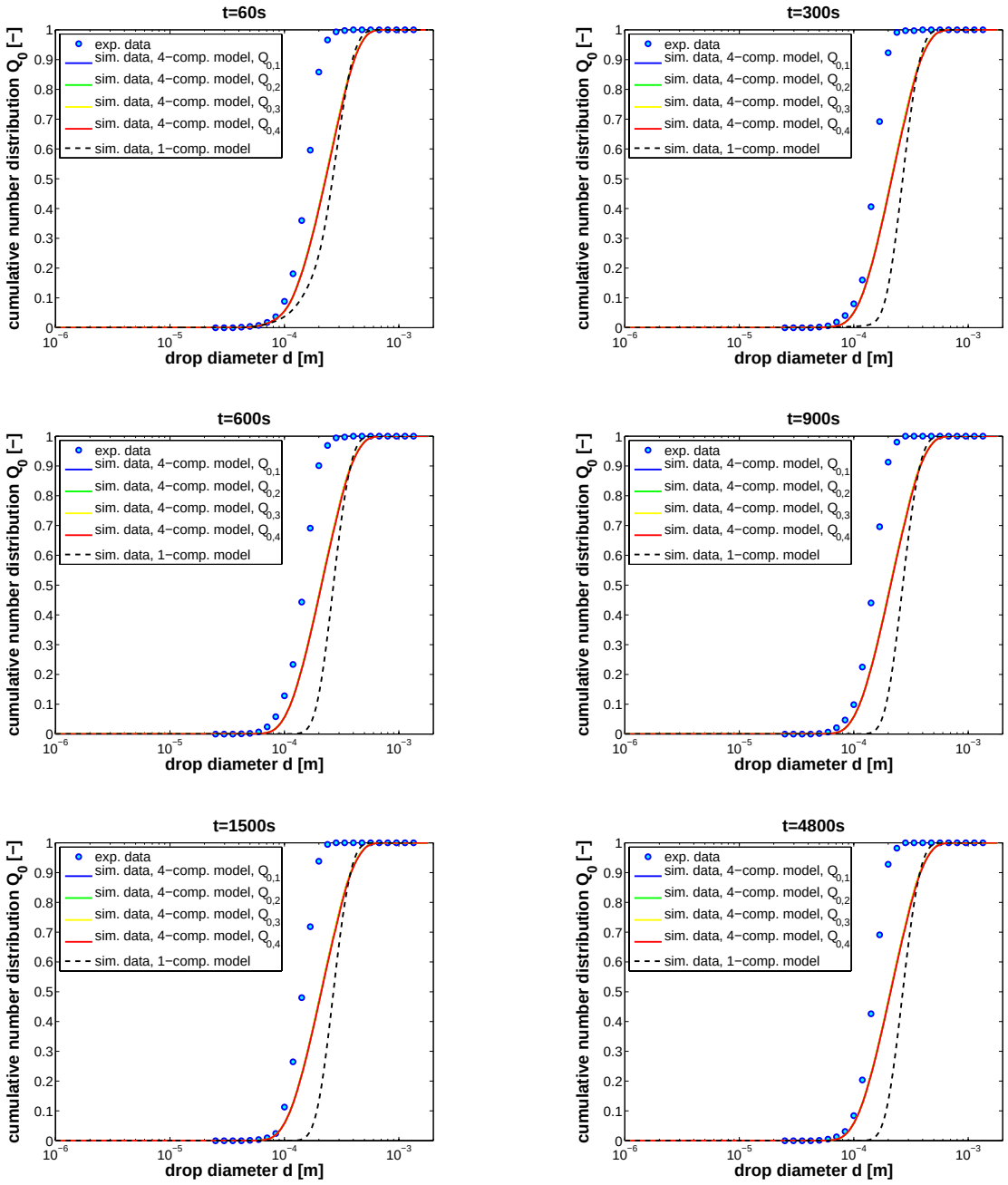


Figure 5.22: Experimental and simulated drop size distributions for the one- and four-compartment model. Top left: $t = 60\,\mathrm{s}$, top right: $t = 300\,\mathrm{s}$, middle left: $t = 600\,\mathrm{s}$, middle right: $t = 900\,\mathrm{s}$, bottom left: $t = 1{,}500\,\mathrm{s}$, bottom right: $t = 4{,}800\,\mathrm{s}$.

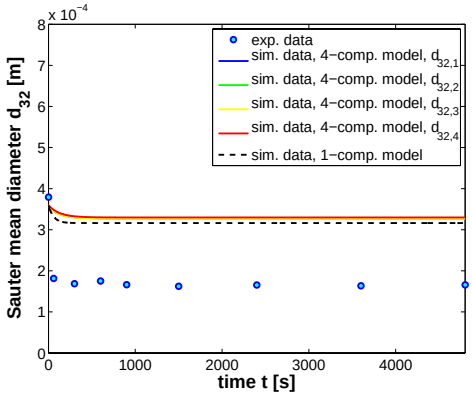


Figure 5.23: Experimental and simulated Sauter mean diameters for the one- and four-compartment model.

such as a dependency on the pH-value.
In practice, however, this is not the case. Here, the "right" choice of the adaption parameters is what mainly influences how good the agreement between experimental and simulated drop size distributions is. Without an optimization of these parameters, the mathematical models are mostly not able to reproduce the experimental drop size distributions satisfactorily. A good agreement with experimental data is often only obtained by parameter adaption, see, e. g., [Weg04], [GSP+05], [GWSK05], or [GWPK06a].
However, in view of being able to perform predictive simulations, the aim of the investigations is to be able to reproduce the experimental drop size distributions without re-adapting the parameters for every single configuration, but by taking into account further factors that have an influence on the drop size distribution. An example is the considered approach, where the inhomogeneity of the flow field in the stirred tank is taken into account by using different values for the energy dissipation rates in the different zones of the tank. By this, the strong influence of the high energy dissipation rate in the region around the impeller is included in the model.

Remark 5.8. *In [GWPK06a], a "correction" factor for the mean energy dissipation rate $\bar{\varepsilon}$ was included in order to model the strong influence of the impeller region on the drop size distribution. By this, an excellent agreement between the simulated and the experimental drop size distributions could be achieved.*
As we have seen from the results presented in this section, the division of the tank into zones and the use of different values for the energy dissipation rate in these zones can lead to different drop size distributions throughout the tank, even if the average value $\bar{\varepsilon}$ is the same as in the one-compartment model simulation. The reason for this is that the locally strong increase of the energy dissipation rate in the (relatively small) impeller zone has a stronger influence on the resulting drop size distribution than the corresponding decrease in the remaining part of the tank. Thus, the "correction" factor used in [GWPK06a] can be seen as some kind of compartment model, where the higher energy

consumption in the impeller zone is modeled, but the remaining part of the tank is simply neglected.
However, the problem with this approach is that the size of the correction factor is not known in advance and can only be determined by comparison with experimental data. Thus, it represents an additional adaption parameter, which makes a predictive simulation more difficult.

By contrast, in the presented approach, no information from the corresponding experimental investigations has been used. The model was extended only by information that could be obtained in advance (namely by the results of the CFD simulation).

Remark 5.9. *Another possibility to get rid of the dependency on the adaption parameters would be to determine one set of parameters that are optimal in a certain sense with respect to many different configurations, e. g., for different rotational speeds, different volume fractions of the dispersed phase, and different pH-values. By this, we clearly do not get the best set of parameters for every single configuration, but we may get one set of parameters, which is able to reproduce all these cases satisfactorily and can therefore still be used after some modifications in one of the quantities listed above has been made.*

However, the disadvantage of this approach is that it is very time-consuming. Using the four-compartment model approach in Parsival*, every fitting with respect to one single configuration would need already several days so that the optimization with respect to all configurations would be too extensive. Furthermore, we need, as in the approach used in [GWPK06a], experimental drop size distributions for the fitting. In order to get a set of reliable parameters, a sufficient number of experimental drop size distributions (for sufficiently many different cases) is needed. In addition, besides these efforts that have to be made, it is not clear if such a set of optimal parameters always leads to satisfactory results. It may happen, for example, that a set of parameters that is optimal with respect to the values that have been used for the fitting is not satisfactory if other values lying in the same range are used.*

In general, one can say that a physically based improvement of the coalescence and breakage models should always be preferred to such a mathematical optimization.

5.4.3 Simulation Results for the Four-Compartment Model with Time-Dependent Volume Flows

In the following, we will present results of the simulation with time-variant volume flows. This case is of special interest, since, in general, the volume flows that we get from the CFD simulation are time-variant. Especially if one likes to control the drop size distributions in the stirred tank, one cannot restrict to the case that constant values are prescribed for the stirrer speed. (One can imagine that most desired drop size distributions cannot be derived by applying only a constant stirrer speed.) Thus, the coupling with Parsival has been implemented in such a way that this can be realized.

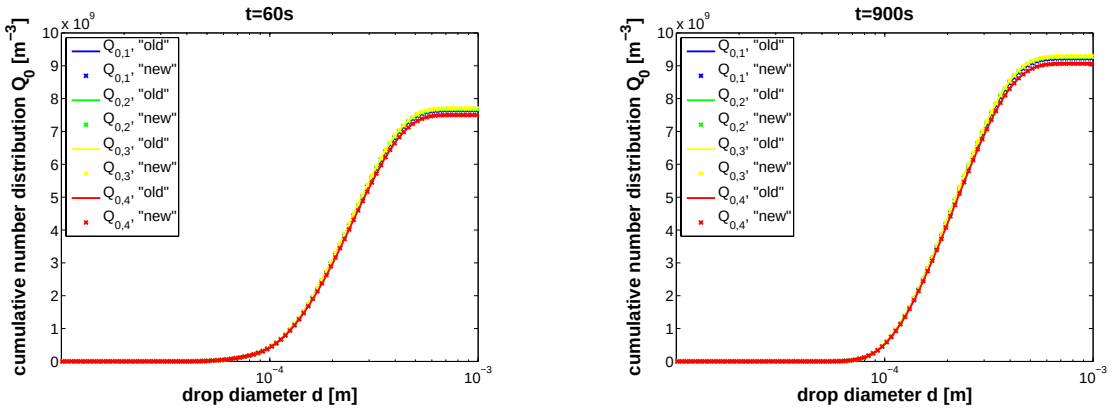


Figure 5.24: Drop size distributions in the four compartments at different time points for a simulation with optimized volume flows. Solid lines: "old" implementation as described in Subsection 5.4.1, cruxes: "new" implementation. Left: $t = 60\,\mathrm{s}$, right: $t = 900\,\mathrm{s}$.

However, due to the enormous computing times for the flow simulation, up to now only very small time intervals could be calculated with FeatFlow. Since the volume flows obtained by the simulation with FeatFlow had to be optimized before they could be used in the coupling approach (see Subsection 5.3.2), only an averaged value for each of the volume flows was derived (which is a sensible approximation for the case that a constant stirrer speed is used).
Nevertheless, in order to test the possibilities of the presented approach, we will in the following also prescribe time-variant mean volume flows and energy dissipation rates.

Test case: simulation with the constant (optimized) volume flows $\mathcal{Z}_{ij}$

In the first test case, we have implemented the stirred tank model in Parsival in such a way that time-variant values for the mean energy dissipation rates $\bar{\varepsilon}_i$ in the different compartments K_i as well as for the mean volume flows $\dot{V}_{ij}$ between the compartments K_i and K_j could be prescribed. Nevertheless, in order to test the implementation, we have used an `mdf`-file with the same (constant) values for $\bar{\varepsilon}_i$ and $\dot{V}_{ij}$ as the ones used for the simulation with optimized volume flows in Subsection 5.4.1.

Fig. 5.24 displays the resulting drop size distributions for $t = 60\,\mathrm{s}$ and $t = 900\,\mathrm{s}$ for both, the "old" implementation and the "new" one. This shows that the "new" model leads, as expected, to exactly the same results.

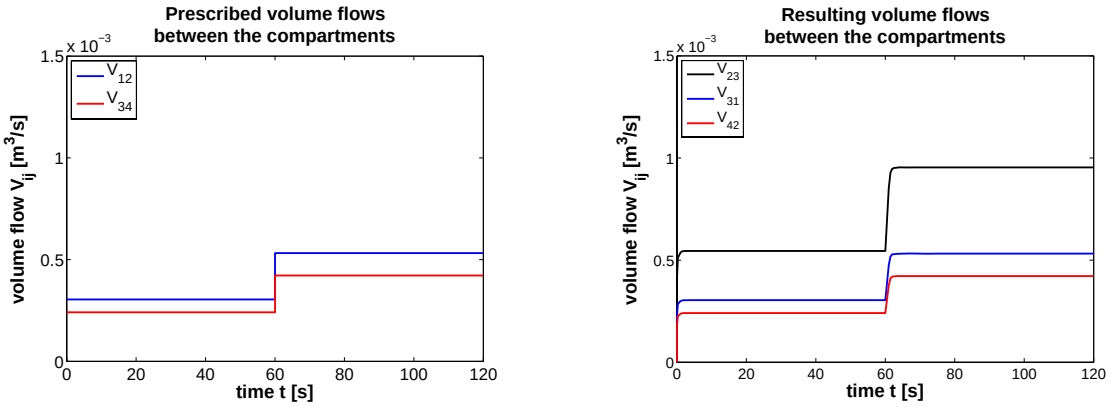


Figure 5.25: Volume flows between the different compartments. Left: prescribed volume flows $\dot{V}_{12}$ and $\dot{V}_{34}$, right: resulting volume flows $\dot{V}_{23}$, $\dot{V}_{31}$, and $\dot{V}_{42}$ calculated by Parsival.

Simulation using step functions for the volume flows

In the following, we will present the simulation results for the (test) case that we have prescribed a step function for the rotational speed of the impeller, namely

$$N^*(t) = \begin{cases} {}^{20}\!/_{3}\ {}^{1}\!/_{\mathrm{s}} & \text{for } 0 \le t \le 60\,\mathrm{s}, \\ {}^{70}\!/_{6}\ {}^{1}\!/_{\mathrm{s}} & \text{for } 60\,\mathrm{s} < t \le 120\,\mathrm{s}, \end{cases}$$

which corresponds to a step in the rotational speed from 400 to 700 rounds per minute. In Parsival this can be modeled by prescribing the following functions for the mean volume flows $\dot{V}_{12}$ and $\dot{V}_{34}$:

$$\dot{V}_{12}(t) = \begin{cases} {}^{4}\!/_{7} \cdot \mathcal{Z}_{12} = 3.04 \cdot 10^{-4}\ {}^{\mathrm{m}^3}\!/_{\mathrm{s}} & \text{for } 0 \le t \le 60\,\mathrm{s}, \\ \mathcal{Z}_{12} = 5.32 \cdot 10^{-4}\ {}^{\mathrm{m}^3}\!/_{\mathrm{s}} & \text{for } 60\,\mathrm{s} < t \le 120\,\mathrm{s} \end{cases}$$

and

$$\dot{V}_{34}(t) = \begin{cases} {}^{4}\!/_{7} \cdot \mathcal{Z}_{34} = 2.41 \cdot 10^{-4}\ {}^{\mathrm{m}^3}\!/_{\mathrm{s}} & \text{for } 0 \le t \le 60\,\mathrm{s}, \\ \mathcal{Z}_{34} = 4.22 \cdot 10^{-4}\ {}^{\mathrm{m}^3}\!/_{\mathrm{s}} & \text{for } 60\,\mathrm{s} < t \le 120\,\mathrm{s}. \end{cases}$$

Note that the values for $60\,\mathrm{s} < t \le 120\,\mathrm{s}$ are the optimal flows from Eq. (5.15) and the values for $0 \le t \le 60\,\mathrm{s}$ are derived by assuming that the volume flows are linearly dependent on the rotational speed, which is justified due to the definition of the volume flows. Fig. 5.25 shows these prescribed volume flows (left) as well as the resulting volume flows between the remaining compartments (right).
Since the mean energy dissipation rates $\bar{\varepsilon}_i$ are defined by

$$\bar{\varepsilon}_i = \check{c}_i \, \frac{\mathrm{Ne}\,(N^*)^3\,(D_{\mathrm{stirrer}})^5}{V_{\mathrm{tank}}}, \qquad i = 1, \ldots, 4,$$

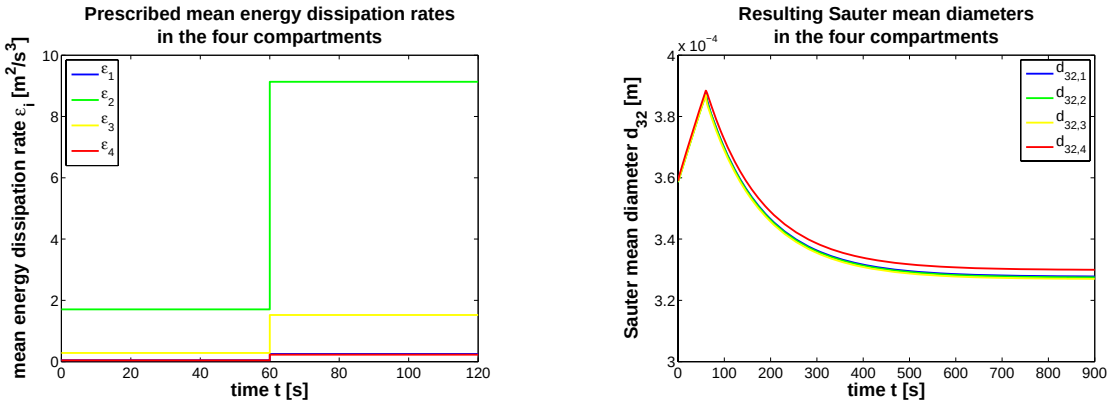


Figure 5.26: Left: prescribed values for the mean energy dissipation rates $\bar{\varepsilon}_i$ in the four compartments, right: resulting Sauter mean diameters in the four compartments.

where the constants $\check{c}_i$, $i = 1, \ldots, 4$, are given according to Eq. (5.16), we have scaled the values for $\bar{\varepsilon}_i$, $i = 1, \ldots, 4$, for $0 \leq t \leq 60\,\mathrm{s}$ by a factor of $(4/7)^3$. Note that we have used the assumption that besides D_{stirrer}, V_{tank}, and $\check{c}_i$ also the power number Ne is constant, which is justified for a fully turbulent flow field. The left picture in Fig. 5.26 shows these mean energy dissipation rates.

The right picture in Fig. 5.26 depicts the resulting Sauter mean diameters in the different compartments. Here one can observe that first, as long as the smaller values for the mean energy dissipation rates are prescribed, the Sauter mean diameters in all compartments increase and then, after the increase of the $\bar{\varepsilon}_i$, the Sauter mean diameters decrease again, until they reach a steady state. Note that the steady state values here are, as expected, the same as for the case that a rotational speed of 700 rounds per minute was used for the whole simulation (see Subsection 5.4.1).

The left picture in Fig. 5.27 demonstrates that mass is conserved during the simulation.

Simulation using sinusoids for the volume flows

In the following, we will present the results for a simulation, where we have prescribed sinusoidal functions for the mean volume flows and energy dissipation rates. This case models a stirred tank operated with a sinusoidal rotational speed, namely

$$N^*(t) = 55/6 + 5/2 \sin\left(\pi/30\, t\right).$$

The left picture in Fig. 5.28 shows the prescribed volume flows $\dot{V}_{12}$ and $\dot{V}_{34}$, defined as

$$\dot{V}_{12}(t) = \frac{N^*(2.5k)}{35/3}\, \mathcal{Z}_{12} \quad \text{for} \quad t \in [2.5k, 2.5(k-1)), \quad k = 0, \ldots, 47$$

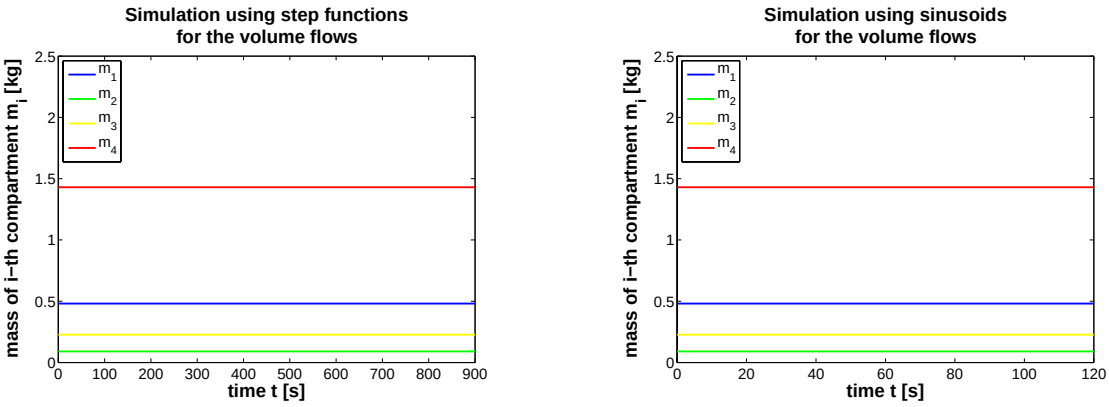


Figure 5.27: Conservation of mass during the simulation. Left: simulation with step-functions prescribed as volume flows and mean energy dissipation rates, right: simulation with sinusoidal volume flows and mean energy dissipation rates.

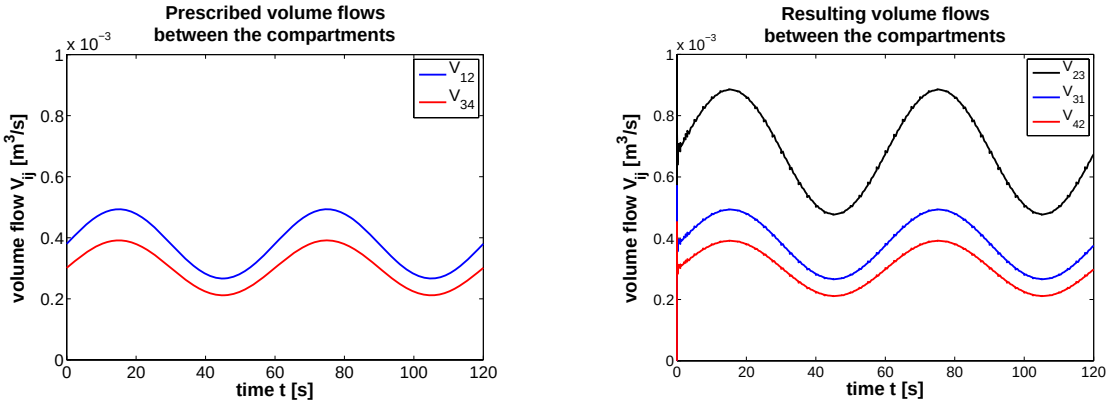


Figure 5.28: Volume flows between the different compartments. Left: prescribed volume flows $\dot{V}_{12}$ and $\dot{V}_{34}$, right: resulting volume flows $\dot{V}_{23}$, $\dot{V}_{31}$, and $\dot{V}_{42}$ calculated by Parsival.

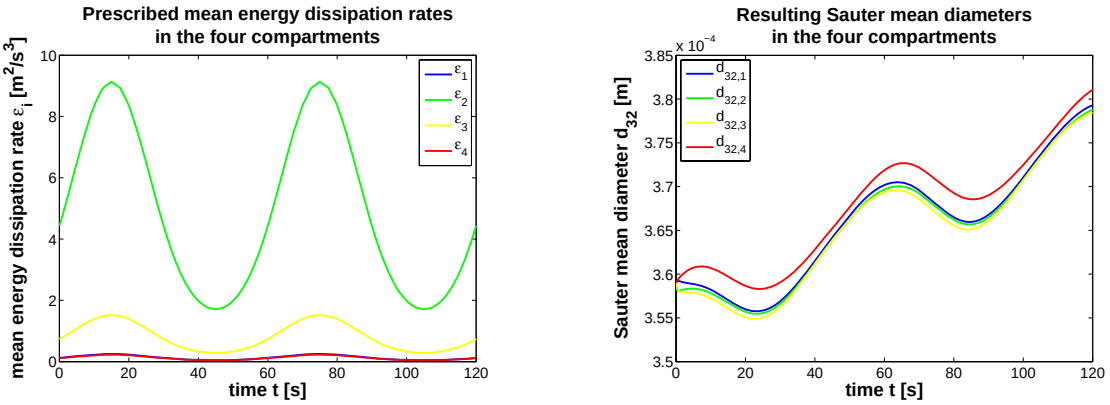


Figure 5.29: Left: prescribed values for the mean energy dissipation rates in the four compartments, right: resulting Sauter mean diameters in the four compartments.

and

$$\dot{V}_{34}(t) = \frac{N^*(2.5k)}{35/3}\,\mathcal{Z}_{34} \quad \text{for} \quad t \in [2.5k, 2.5(k-1)), \quad k = 0, \ldots, 47.$$

The other three volume flows, which are calculated by PARSIVAL during the simulation, are displayed in the right picture of Fig. 5.28.

Accordingly to the prescribed volume flows, for the mean energy dissipation rates $\bar{\varepsilon}_i$, the following values are used:

$$\bar{\varepsilon}_i(t) = \left[\frac{N^*(2.5k)}{35/3}\right]^3 \check{c}_i\,\bar{\varepsilon} \quad \text{for} \quad t \in [2.5k, 2.5(k-1)), \quad k = 0, \ldots, 47, \quad i = 1, \ldots, 4.$$

These data are depicted in the left picture of Fig. 5.29.

Note that for both, the mean volume flows and the mean energy dissipation rates, we have only prescribed discrete values instead of "real" sinusoids. The reason for this is that, in the coupling approach, the results of the CFD simulation are also discrete values instead of (continuous) functions.

The right picture in Fig. 5.29 shows the Sauter mean diameters in the different compartments. It displays how the Sauter mean diameters increase and decrease with decreasing and increasing mean energy dissipation rates, respectively. However, besides the sinusoidal behavior (according to the sinusoidal mean energy dissipation rates), one can also observe that altogether the Sauter mean diameters increase, which shows that coalescence dominates in the system. Note that we will not give a more detailed analysis of the results, since the considered configuration is only a test case, which does not have any relevance in practice. However, it demonstrates further applications and possibilities of the presented approach in the context of control.

At the end of this subsection, it should be stated that also for the case, where sinusoids are used for the volume flows, mass is conserved during the simulation (see the right picture in Fig. 5.27).

5.5 Discussion

As already mentioned in Section 5.1, Parsival is a powerful tool for the solution of population balance equations. Since in Parsival an adaptive discretization is not only used for the time, but also for the internal coordinate, a good approximation of the drop size distribution can be obtained with relatively low computational costs, at least compared to other method of classes approaches. Compared to the method of moments approaches, Parsival has the advantage that the whole distribution is calculated instead of only a few moments.
However, as soon as more complex models are considered, the relatively high computation times needed for the method of classes approaches carry more and more weight (compare the high simulation times for the four-compartment model presented in the previous section), and the method of moments approaches become superior. In general, the high costs for the method of classes makes an implementation in a CFD code hardly possible. While it suffices to extend the CFD code by about six transport equations for the case that the quadrature method of moments is used, at least 20 (for more complex problems even between 100 and 200) additional equations are needed for an implementation of the method of classes [MBBF02], see also Section 5.6.

Besides these enormous costs needed for the coupling of a method of classes approach with a CFD code, Parsival can hardly be coupled "directly" with a CFD code. The problem here is that Parsival is a commercial Windows program controlled by a graphical user interface, which cannot simply be "included" into a CFD solver. The version of Parsival that has been used for the investigations presented here can, in addition, only be started by pressing a "run"-button inside the program.
Nevertheless, calling Parsival (as a so-called "Automation Server") from "outside" (namely from Microsoft Excel or from a C++ or FORTRAN program) is realized via the so-called "COM-Interface" (where COM is short for Common Object Model), see [Wul].

Altogether, one can say that the actual way of coupling the two solvers FeatFlow and Parsival was a good first approach to investigate the problem, but for more practical problems or for further investigations, concerning the control and optimization of drop size distributions, for example, an alternative approach for the solution of the coupled system is necessary. For these applications, the system must be coupled on another level. A sensible possibility would be to "directly" include the solution of the population balance equation into the CFD simulation. If a non-commercial CFD code is used, where the source code is freely available and can be adapted, the solution of the population balance equation can directly be included by extending the system of equations by some additional transport equations. Since the quadrature method of moments and the direct

quadrature method of moments are really suitable for such a way of coupling the population balance equation with CFD, these approaches will be considered in the following section.

Another point that should be discussed is the choice of the "fitting" parameters occurring in the coalescence and breakage models. The comparison of the simulation results with experimental data in Subsection 5.4.2 showed that further difficulties arise in the context of finding the "right" values for the adaption parameters in the coalescence and breakage rates. The problem here is that predictive simulations are not possible as long as an optimization of the model parameters based on experimental data is needed for every slightly changed model. Thus, an improvement of the models is necessary in order to weaken or even avoid the undesired dependency of the simulation results on these parameters.
This can be achieved by extending the mathematical description of the coalescence and breakage processes by dependencies on physical parameters, like the pH value or the ionic strength, which can be observed in experiments (see, e. g., [KGS04] or [GWPK06a]), but are mostly not yet included in the modeling.
Another factor that has a strong influence on the coalescence and breakage processes, but is mostly not taken into account in the modeling, is the inhomogeneity of the flow field in the stirred tank. One way to include this dependency in the modeling is given by the compartment model approach presented in this chapter. Clearly, this approach can further be improved by increasing the number of compartments. However, a simulation in PARSIVAL with a significantly larger number of compartments would lead to enormous computational costs. Thus, also for this case, a method of moments approach seems to be a good alternative.

5.6 The (Direct) Quadrature Method of Moments as Alternative Approach to Solve Population Balance Equations

In this section, we will consider the quadrature method of moments (QMOM) and the direct quadrature method of moments (DQMOM) as two alternative approaches for solving population balance equations. We consider these methods, since they have the advantage that they can easily be coupled with a CFD simulation. For the coupling, the CFD code must only be extended by a few scalar transport equations, from which the moments of the distribution can be calculated. Such a coupling has already been implemented for certain applications, see, e. g., [MVF03a], [MPF$^+$03], [FMF04], [ZMBF04], [ZMBF06], [MSS$^+$06], or [MSSM06].

The quadrature method of moments is based on the method of moments, in which the internal coordinate is "integrated out" and the population balance equation is determined through its moments, see [MVF03b]. The method of moments was already described by Hulburt and Katz [HK64] in 1964 but has not found wide applicability

due to the difficulty of expressing the transport equations for the moments in terms of the moments themselves [MVF03b]. This so-called "closure problem" can, for example, be solved by using a quadrature approximation for the unclosed terms leading to the quadrature method of moments proposed in [McG97]. Here, the weights and abscissae of the quadrature approximation can be specified from the lower-order moments.
The direct quadrature method of moments, introduced in [MF] (see [FMF04]), is an extension of the quadrature method of moments. It is based on the direct solution of the transport equations for weights and abscissae of the quadrature approximation [FMF04].

In the following subsections, we will consider the application of the (direct) quadrature method of moments to the population balance equation (2.9), here given in a more general form, namely dependent on any internal coordinate e instead of the volume V,

$$\frac{\partial f(e,x,t)}{\partial t} = -\nabla \cdot (u\, f(e,x,t)) + s(e,x,t), \tag{5.17}$$

or, respectively, to the averaged population balance equation (2.15), given by

$$\frac{\partial f(e,x,t)}{\partial t} = -\nabla \cdot (u f(e,x,t)) + \nabla \cdot (c_t\, \nabla f(e,x,t)) + s(e,x,t). \tag{5.18}$$

Note that we have, for ease of notation, omitted the bars over the averaged values here.

First, we will describe the different approaches mentioned above. For simplicity, the different methods are only applied to the population balance equation (5.17). However, they can be used for the averaged population balance equation (5.18) analogously. In the last subsection, this will be shown considering the direct quadrature method of moments as example.

5.6.1 The Method of Moments

The method of moments is based on the idea of solving the governing equations for a finite set of moments instead of solving the population balance equation itself. This can be done, since there is a one-to-one correspondence between the distribution f and the so-called moment space, i. e., the set of all moments $\{m_0, m_1, m_2, \dots\}$, where the moments are defined by

$$m_k = \int_{-\infty}^{\infty} f(e) e^k de, \quad k = 0, 1, 2, \dots .$$

Note that in the considered application, where the volume is used as internal coordinate, it is sufficient to integrate from 0 to ∞, since there are no drops of negative volume.

The moments also have a physical meaning. The zeroth-order moment, for example, describes the total particle number density independent of what we use as internal coordinate. If we use length as internal coordinate, the first-order moment describes the total particle length, the second-order moment the total particle surface area and the

third-order moment the total particle volume, whereas, if we use volume as internal coordinate, the first-order moment describes the total particle volume.

In order to solve the population balance equation (5.17) by the method of moments, we apply the moment transform, which means that we "integrate out" the internal coordinate e:

$$\int_{-\infty}^{\infty} \frac{\partial f(e,x,t)}{\partial t}\, e^k\, de = -\int_{-\infty}^{\infty} \nabla\cdot(u\, f(e,x,t))\, e^k\, de + \int_{-\infty}^{\infty} s(e,x,t)\, e^k\, de, \quad k = 0,1,2,\ldots,$$

leading to the following transport equations for the moments

$$\frac{\partial m_k(x,t)}{\partial t} = -\nabla\cdot(u\, m_k(x,t)) + \int_{-\infty}^{\infty} s(e,x,t)\, e^k\, de, \quad k = 0,1,2,\ldots, \tag{5.19}$$

where the term $\int_{-\infty}^{\infty} s(e,x,t)\, e^k\, de =: \bar{s}_k(x,t)$ is given by (cf. Section B.2)

$$\begin{aligned}
\bar{s}_k(x,t) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} R_{\text{coal}}(e',e'',y(x,t))\, f(e',x,t)\, f(e'',x,t)\, de'\, e^k\, de \\
&- \int_{-\infty}^{\infty} f(e,x,t) \int_{-\infty}^{\infty} R_{\text{coal}}(e,e',y(x,t))\, f(e',x,t)\, de'\, e^k\, de \\
&+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} n(e',y(x,t))\, \gamma(e,e',y(x,t))\, R_{\text{break}}(e',y(x,t))\, f(e',x,t)\, de'\, e^k\, de \\
&- \int_{-\infty}^{\infty} R_{\text{break}}(e,y(x,t))\, f(e,x,t)\, e^k\, de, \qquad k = 0,1,2,\ldots.
\end{aligned} \tag{5.20}$$

This shows that, by applying the moment transform, it is possible to get rid of the internal coordinate e, i. e., to reduce the dimension of the problem by one. But, in exchange, we get infinitely many equations. Hence, we have to use an approximation. In the method of moments, the set of transport equations is only solved for the first N_{mom} moments.

5.6.2 The Closure Problem and the Quadrature Method of Moments

The problem of the method of moments is that for complex kernels (like for coalescence or breakage kernels that are dependent on the internal coordinate), it is usually not possible to write the equations for the first N_{mom} moments in terms of only these moments. This is because the right-hand side of the equation for the k-th-order moment may include higher-order moments $m_{k+1}, m_{k+2}, \ldots$, which is, for example, the case if we use a breakage rate that is linearly dependent on the internal coordinate e. (Set, e. g., $R_{\text{break}}(e,x,t) = e$ in Eq. (5.20)).

There are several ways to overcome this so-called "closure problem". (A description of some possible approaches can be found in [DO02a], for example.) One possibility is to

use a quadrature approximation for all terms involving the unknown distribution f, see [McG97], given by

$$\int_{-\infty}^{\infty} \pi(e) f(e) de \approx \sum_{\alpha=1}^{N_{\text{mom}}} w_\alpha \, \pi(e_\alpha), \tag{5.21}$$

where w_α are the weights and e_α are the abscissae (or nodes) of the quadrature approximation. This approximation is exact if the nodes e_α are the zeros of a polynomial of order N_{mom} that is orthogonal to the distribution f and if π is a polynomial of order less than or equal to $2N_{\text{mom}} - 1$ (see, e. g., [Sto99], page 176).

In the considered application, the terms that involve the unknown distribution f are the source and sink terms due to coalescence and breakage given by Eq. (5.20). Applying the quadrature methods of moments to Eq. (5.19) leads to an approximation $\bar{s}_k^{(N_{\text{mom}})}(x,t)$ of these source and sink terms $\bar{s}_k(x,t)$. How this approximation can be calculated, depends on the special form of the kernels and on what we choose as internal coordinate. Details concerning this can, for example, be found in [MPF$^+$03].

The essence of the quadrature-based closure lies in the fact that the weights w_α and abscissae e_α can be completely specified in terms of the lower-order moments of the unknown distribution f, see [McG97].

Using this approximation (5.21), we get for the moments:

$$m_k = \int_{-\infty}^{\infty} f(e) e^k de \approx \sum_{\alpha=1}^{N_{\text{mom}}} w_\alpha e_\alpha^k, \quad k = 0, 1, 2, \ldots. \tag{5.22}$$

This shows that it suffices to know the first $2N_{\text{mom}}$ moments in order to determine the weights and abscissae in a quadrature approximation of order N_{mom}. However, the direct solution of Eq. (5.22) for these quantities would require a nonlinear search and is not recommended [McG97]. That is why the product-difference (PD) algorithm proposed in [Gor68] is used to determine the weights and abscissae from the lower-order moments. This algorithm has many points in common with the quotient-difference (QD) algorithm described in [Rut57]. However, the product-difference algorithm has the advantage that it saves all divisions until the end.

This algorithm works as follows (see [Mar01], for example). In the first step, a matrix $\mathcal{B} = [b_{ij}] \in \mathbb{R}^{(2N_{\text{mom}}+1)\times(2N_{\text{mom}}+1)}$ is constructed. The components of the first column are given by

$$b_{i1} = \delta_{i1}, \quad i = 1, \ldots, 2N_{\text{mom}} + 1,$$

where δ_{ij} is the Kronecker delta. The second column consists of

$$\begin{aligned} b_{i2} &= (-1)^{i-1} m_{i-1}, \quad i = 1, \ldots, 2N_{\text{mom}}, \\ b_{2N_{\text{mom}}+1,2} &= 0, \end{aligned}$$

where m_0 is set to one. (This normalization can be done, since the final weights can be corrected by multiplication with the true m_0.) The remaining components are found by

means of the product-difference algorithm:

$$b_{ij} = b_{1,j-1}\, b_{i+1,j-2} - b_{1,j-2}\, b_{i+1,j-1}, \quad j = 3, \ldots, 2N_{\text{mom}} + 1, \quad i = 1, \ldots, 2N_{\text{mom}} + 2 - j.$$

The remaining entries in the lower right part of the matrix are set to zero. Next, we define $\hat{b}_1 := 0$ and

$$\hat{b}_i = \frac{b_{1,i+1}}{b_{1i}\, b_{1,i-1}}, \quad i = 2, \ldots, 2N_{\text{mom}},$$

in order to build a symmetric tridiagonal matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N_{\text{mom}} \times N_{\text{mom}}}$ from the sums and products of $\hat{b}_i$. The diagonal of the matrix $\mathcal{A}$ is given by

$$a_{ii} = \hat{b}_{2i} + \hat{b}_{2i-1}, \quad i = 1, \ldots, N_{\text{mom}},$$

while its co-diagonal elements are given by

$$a_{i,i+1} = a_{i+1,i} = -\sqrt{\hat{b}_{2i+1}\hat{b}_{2i-1}}, \quad i = 1, \ldots, N_{\text{mom}} - 1.$$

The abscissae and weights are now determined by finding the eigenvalues and -vectors of the matrix $\mathcal{A}$. The abscissae are just the eigenvalues of $\mathcal{A}$, whereas the weights can be found by

$$w_j = m_0 v_{j1}^2,$$

where v_{j1} is the first component of the jth eigenvector v_j. Once abscissae and weights have been computed, any integral involving the number density function f can be approximated using Eq. (5.21).

The quadrature approximation (5.21) is equivalent to using the following finite-mode representation for the solution f, see [Mar04]:

$$f(e, x, t) = \sum_{\alpha=1}^{N_{\text{mom}}} w_\alpha \delta(e - e_\alpha),$$

where δ denotes the Dirac delta function, which can be loosely thought of as a function on the real line which is zero everywhere except at the origin, where it is infinite, i. e.,

$$\delta(x) = \begin{cases} 0, & x \neq 0, \\ \infty, & x = 0 \end{cases}$$

and which is also constrained to satisfy the identity

$$\int_{-\infty}^{\infty} \delta(x)\, dx = 1.$$

In this approach, each node e_α of the quadrature approximation can be thought of a single particle class, see Fig. 5.30.

One great advantage of the quadrature method of moments is that only a few (usually about six) moments are needed in order to get a satisfactory approximation of the whole distribution, which makes the implementation in a CFD code feasible. As already mentioned in Section 5.5, in the method of classes, in contrast, about 20–30 classes for simple problems and about 100–200 classes for more complex problems are needed in order to get about the same accuracy, see [MBBF02].

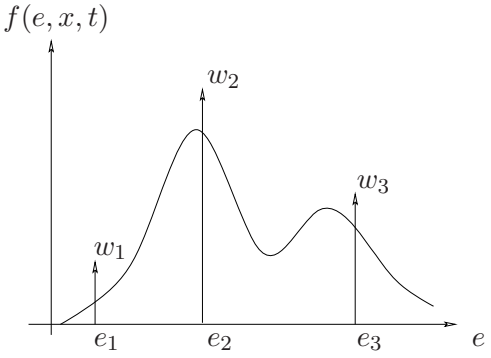


Figure 5.30: Each node e_α can be seen as a single particle class in the quadrature method of moments approach.

5.6.3 The Direct Quadrature Method of Moments

The direct quadrature method of moments directly solves transport equations for the weights w_α and abscissae e_α instead of solving the equations for the moments. Here, space- and time-dependent weights $w_\alpha(x,t)$ and nodes $e_\alpha(x,t)$ are used in the finite-mode representation of the solution

$$f(e,x,t) = \sum_{\alpha=1}^{N_{\text{mom}}} w_\alpha(x,t)\,\delta(e - e_\alpha(x,t)). \tag{5.23}$$

When the direct quadrature method of moments is implemented in a multi-fluid CFD code, each node of the quadrature approximations can be seen as a distinct dispersed phase with its own phase velocity u_α.

Inserting Eq. (5.23) into the population balance equation (5.17) leads to

$$\begin{aligned} &\sum_{\alpha=1}^{N_{\text{mom}}} \delta(e-e_\alpha)\left[\frac{\partial w_\alpha}{\partial t} + (\nabla\cdot u_\alpha)w_\alpha + u_\alpha\cdot\nabla w_\alpha\right] \\ -\ &\sum_{\alpha=1}^{N_{\text{mom}}} \delta'(e-e_\alpha)\left[\frac{\partial e_\alpha}{\partial t}w_\alpha + (u_\alpha\cdot\nabla e_\alpha)\,w_\alpha\right] \;=\; s(e), \end{aligned} \tag{5.24}$$

where u_α is the phase velocity for $e = e_\alpha$. Note that, for simplicity, we have omitted the dependency on space and time here. If we introduce weighted abscissae

$$\zeta_\alpha = w_\alpha e_\alpha$$

into Eq. (5.24) and use

$$\begin{aligned}
&\frac{\partial e_\alpha}{\partial t} w_\alpha + (u_\alpha \cdot \nabla e_\alpha) w_\alpha \\
&\qquad = \frac{\partial e_\alpha}{\partial t} w_\alpha + (u_\alpha \cdot \nabla w_\alpha) e_\alpha + (u_\alpha \cdot \nabla e_\alpha) w_\alpha - e_\alpha (u_\alpha \cdot \nabla w_\alpha) \\
&\qquad = \frac{\partial w_\alpha}{\partial t} e_\alpha + \frac{\partial e_\alpha}{\partial t} w_\alpha + (\nabla \cdot u_\alpha) w_\alpha e_\alpha + u_\alpha \cdot (\nabla \cdot (w_\alpha e_\alpha)) - e_\alpha \frac{\partial w_\alpha}{\partial t} \\
&\qquad\quad - e_\alpha (\nabla \cdot u_\alpha) w_\alpha - e_\alpha (u_\alpha \cdot \nabla w_\alpha) \\
&\qquad = \frac{\partial (w_\alpha e_\alpha)}{\partial t} + \nabla \cdot (u_\alpha w_\alpha e_\alpha) - e_\alpha \frac{\partial w_\alpha}{\partial t} - e_\alpha \nabla \cdot (u_\alpha w_\alpha) \\
&\qquad = \frac{\partial \zeta_\alpha}{\partial t} + \nabla \cdot (u_\alpha \zeta_\alpha) - e_\alpha \left\{ \frac{\partial w_\alpha}{\partial t} + \nabla \cdot (u_\alpha w_\alpha) \right\},
\end{aligned}$$

we get

$$\begin{aligned}
&\sum_{\alpha=1}^{N_{\text{mom}}} \delta(e - e_\alpha) \left[\frac{\partial w_\alpha}{\partial t} + \nabla \cdot (u_\alpha w_\alpha) \right] \\
- &\sum_{\alpha=1}^{N_{\text{mom}}} \delta'(e - e_\alpha) \left[\frac{\partial \zeta_\alpha}{\partial t} + \nabla \cdot (u_\alpha \zeta_\alpha) - e_\alpha \left\{ \frac{\partial w_\alpha}{\partial t} + \nabla \cdot (u_\alpha w_\alpha) \right\} \right] \;=\; s(e).
\end{aligned}$$

This expression is the starting point for deriving transport equations for the weights and weighted abscissae.

If we define $\xi_\alpha^{(1)}$ and $\xi_\alpha^{(2)}$ to be the source terms of the transport equations for the weights w_α and weighted abscissae ζ_α

$$\begin{aligned}
\frac{\partial w_\alpha}{\partial t} + \nabla \cdot (u_\alpha w_\alpha) &= \xi_\alpha^{(1)}, \\
\frac{\partial \zeta_\alpha}{\partial t} + \nabla \cdot (u_\alpha \zeta_\alpha) &= \xi_\alpha^{(2)},
\end{aligned}$$

then the population balance equation becomes

$$\sum_{\alpha=1}^{N_{\text{mom}}} \delta(e - e_\alpha) \xi_\alpha^{(1)} - \sum_{\alpha=1}^{N_{\text{mom}}} \delta'(e - e_\alpha)(\xi_\alpha^{(2)} - e_\alpha \xi_\alpha^{(1)}) = s(e). \tag{5.25}$$

This equation can now be used to determine $\xi_\alpha^{(1)}$ and $\xi_\alpha^{(2)}$ by applying the moment transform. Using

$$\int_{-\infty}^{\infty} e^k \delta(e - e_\alpha) de = e_\alpha^k \quad \text{and} \quad \int_{-\infty}^{\infty} e^k \delta'(e - e_\alpha) de = -k e_\alpha^{k-1}, \tag{5.26}$$

the moment transform of equation (5.25) yields

$$(1 - k) \sum_{\alpha=1}^{N_{\text{mom}}} e_\alpha^k \xi_\alpha^{(1)} + k \sum_{\alpha=1}^{N_{\text{mom}}} e_\alpha^{k-1} \xi_\alpha^{(2)} = \int_{-\infty}^{\infty} s(e) e^k de = \bar{s}_k(x, t). \tag{5.27}$$

This linear system can now be written in matrix-vector form:

$$\mathcal{E}\xi = \varsigma, \tag{5.28}$$

where the matrix $\mathcal{E} = [\mathcal{E}_1, \mathcal{E}_2] \in \mathbb{R}^{2N_{\text{mom}} \times 2N_{\text{mom}}}$ is given by

$$\mathcal{E}_1 = \begin{bmatrix} 1 & \cdots & 1 \\ 0 & \cdots & 0 \\ -e_1^2 & \cdots & -e_{N_{\text{mom}}}^2 \\ \vdots & & \vdots \\ -(2N_{\text{mom}}-2)e_1^{2N_{\text{mom}}-1} & \cdots & -(2N_{\text{mom}}-2)e_{N_{\text{mom}}}^{2N_{\text{mom}}-1} \end{bmatrix} \quad \text{and}$$

$$\mathcal{E}_2 = \begin{bmatrix} 0 & \cdots & 0 \\ 1 & \cdots & 1 \\ 2e_1 & \cdots & 2e_{N_{\text{mom}}} \\ \vdots & & \vdots \\ (2N_{\text{mom}}-1)e_1^{2N_{\text{mom}}-2} & \cdots & (2N_{\text{mom}}-1)e_{N_{\text{mom}}}^{2N_{\text{mom}}-2} \end{bmatrix}.$$

The vectors $\xi, \varsigma \in \mathbb{R}^{2N_{\text{mom}}}$ are defined by

$$\xi = \left[\xi_1^{(1)}, \ldots, \xi_{N_{\text{mom}}}^{(1)}, \xi_1^{(2)}, \ldots, \xi_{N_{\text{mom}}}^{(2)}\right]^T \qquad \text{and} \qquad \varsigma = \left[\,\bar{s}_0, \ldots, \bar{s}_{2N_{\text{mom}}-1}\right]^T.$$

If the abscissae e_α are unique, then $\mathcal{E}$ has full rank for arbitrary N_{mom}. For this case, the source terms for the transport equations for the weights and weighted abscissae can be found by simply solving the linear system (5.28). In practice, however, for $N_{\text{mom}} > 2$, the system has to be solved numerically.
If there is any point in the computational domain where two abscissae are equal, then the matrix $\mathcal{E}$ becomes singular. In this case, one possible approach to overcome this problem is to estimate the vector ξ from the average of the source vectors from neighboring cells, see [FMF04]. Further details concerning this as well as alternative approaches can be found in [FMF04].

When working with one internal coordinate, the direct quadrature method of moments is numerically equivalent to the quadrature method of moments. Both methods result in the same prediction, with the difference that the direct quadrature method of moments does not use the product-difference algorithm but tracks the quadrature approximation by solving a simple linear system, see [Mar04]. A great advantage of the direct quadrature method of moments is that it can easily be extended to bivariate (and multivariate) problems, see [FMF04] and [ZMBF04]. In [MZV+06], a detailed description of the direct quadrature method of moments applied to bivariate population balance models as well as an investigation of its numerical performance (compared to Monte Carlo simulation methods) can be found.

Applying the direct quadrature method of moments to the averaged population balance equation (5.18)

In the following, we will apply the direct quadrature method of moments also to the averaged population balance equation (5.18). This can be done in the same way as before, we only have to account for the additional term $\nabla \cdot (c_t \ \nabla f)$ due to turbulent diffusion. Inserting

$$f(e,x,t) = \sum_{\alpha=1}^{N_{\text{mom}}} w_\alpha(x,t)\delta(e - e_\alpha(x,t))$$

into the averaged population balance equation (5.18) leads to

$$\begin{aligned}
&\sum_{\alpha=1}^{N_{\text{mom}}} \delta(e-e_\alpha)\left[\frac{\partial w_\alpha}{\partial t} + (\nabla\cdot u_\alpha)w_\alpha + u_\alpha\cdot\nabla w_\alpha - \nabla c_t\cdot\nabla w_\alpha - c_t\Delta w_\alpha\right] \\
- &\sum_{\alpha=1}^{N_{\text{mom}}} \delta'(e-e_\alpha)\left[\frac{\partial e_\alpha}{\partial t}w_\alpha + (u_\alpha\cdot\nabla e_\alpha)\ w_\alpha - (\nabla c_t\cdot\nabla e_\alpha)\ w_\alpha\right. \\
&\qquad\qquad \left. - 2c_t(\nabla w_\alpha\cdot\nabla e_\alpha) - c_t\Delta e_\alpha\ w_\alpha\right] \\
- &\sum_{\alpha=1}^{N_{\text{mom}}} \delta''(e-e_\alpha)\left[c_t(\nabla e_\alpha\cdot\nabla e_\alpha)\ w_\alpha\right] \ = \ s(e),
\end{aligned}$$

where we have again omitted the dependency on space and time.

With the weighted abscissae $\zeta_\alpha = w_\alpha e_\alpha$, we get

$$\begin{aligned}
&\sum_{\alpha=1}^{N_{\text{mom}}} \delta(e-e_\alpha)\left[\frac{\partial w_\alpha}{\partial t} + \nabla\cdot(u_\alpha w_\alpha) - \nabla\cdot(c_t\ \nabla w_\alpha)\right] \\
- &\sum_{\alpha=1}^{N_{\text{mom}}} \delta'(e-e_\alpha)\left[\frac{\partial \zeta_\alpha}{\partial t} + \nabla\cdot(u_\alpha\zeta_\alpha) - \nabla\cdot(c_t\ \nabla\zeta_\alpha)\right. \\
&\qquad\qquad \left. - e_\alpha\left\{\frac{\partial w_\alpha}{\partial t} + \nabla\cdot(u_\alpha w_\alpha) - \nabla\cdot(c_t\ \nabla w_\alpha)\right\}\right] \qquad (5.29) \\
- &\sum_{\alpha=1}^{N_{\text{mom}}} \delta''(e-e_\alpha)\left[c_t\nabla e_\alpha\cdot\nabla e_\alpha\ w_\alpha\right] \ = \ s(e).
\end{aligned}$$

Again, we define $\xi_\alpha^{(1)}$ and $\xi_\alpha^{(2)}$ to be the source terms of the transport equations for the weights w_α and weighted abscissae ζ_α, this time with the additional turbulent diffusion term,

$$\begin{aligned}
\frac{\partial w_\alpha}{\partial t} + \nabla\cdot(u_\alpha w_\alpha) - \nabla\cdot(c_t\ \nabla w_\alpha) &= \xi_\alpha^{(1)}, \\
\frac{\partial \zeta_\alpha}{\partial t} + \nabla\cdot(u_\alpha \zeta_\alpha) - \nabla\cdot(c_t\ \nabla \zeta_\alpha) &= \xi_\alpha^{(2)}.
\end{aligned}$$

Further, we define

$$\xi_\alpha^{(3)} := c_t \left(\nabla e_\alpha \cdot \nabla e_\alpha\right) w_\alpha.$$

With this, Eq. (5.29) can be written as

$$\sum_{\alpha=1}^{N_{\text{mom}}} \delta(e-e_\alpha)\xi_\alpha^{(1)} - \sum_{\alpha=1}^{N_{\text{mom}}} \delta'(e-e_\alpha)(\xi_\alpha^{(2)} - e_\alpha\xi_\alpha^{(1)}) - \sum_{\alpha=1}^{N_{\text{mom}}} \delta''(e-e_\alpha)\xi_\alpha^{(3)} = s(e).$$

Again, we apply the moment transform to determine $\xi_\alpha^{(1)}$, $\xi_\alpha^{(2)}$, and $\xi_\alpha^{(3)}$.
So, using (5.26) as well as

$$\int_{-\infty}^{\infty} e^k \delta''(e-e_\alpha)de = k(k-1)e_\alpha^{k-2},$$

we end up with the following linear system

$$(1-k)\sum_{\alpha=1}^{N_{\text{mom}}} e_\alpha^k\xi_\alpha^{(1)} + k\sum_{\alpha=1}^{N_{\text{mom}}} e_\alpha^{k-1}\xi_\alpha^{(2)} = k(k-1)\sum_{\alpha=1}^{N_{\text{mom}}} e_\alpha^{k-2}\xi_\alpha^{(3)} + \int_{-\infty}^{\infty} s(e)e^k de,$$

which can be solved by the same means as Eq. (5.27).

6 Summary and Outlook

6.1 Summary

In this thesis, the modeling, analysis, and numerical simulation of a stirred liquid-liquid system was considered.

A system of equations was derived describing, on one hand, the turbulent flow field in the stirred tank and, on the other hand, the behavior of the drops. The turbulent flow field in the stirred tank was modeled by the Reynolds-averaged Navier-Stokes equations in combination with a k-ε turbulence model. For the description of the population dynamical processes an averaged population balance equation was used, in which coalescence and breakage were modeled via integral terms. For these terms, different models from the literature were presented, and it was discussed how they can be adapted so that they can be used for the considered application.

By an index analysis of the underlying differential-algebraic system, which was derived by spatial discretization, it could be shown that the index of the Navier-Stokes equations was not increased by the coupling with the population balance equation. Thus, it could be concluded that all solvers that are suitable for the solution of the Navier-Stokes equations can, at least theoretically, be extended so that they can also be applied for the solution of the coupled system.

However, due to the high effort for building such a solver, the system of equations was solved by means of a simulator coupling approach, in which the CFD code FeatFlow was used for the flow simulation and the population balance solver Parsival was applied for the calculation of the drop size distributions. The coupling was realized as a one-way coupling, which means that the results of the flow simulation were used for the calculation of the drop size distributions, but the influence of the drops on the flow field was neglected. This simplification was justified for the considered application, since the densities of the continuous and of the dispersed phase were similar and the volume fraction of the drops was sufficiently small. Due to the difficulties in the implementation of a k-ε turbulence model for the moving boundary method in FeatFlow, the coupling was realized without turbulence model. Hence, in the compartment model used for the calculation of the drop size distributions with Parsival, results from the CFD simulation were only used for the calculation of the mean volume flows, whereas for the mean energy dissipation rates values from the literature were taken.

A comparison of the simulated drop size distributions with experimental data, which have been obtained by the experimental partner project showed that by using a four-compartment model, which takes (at least roughly) the inhomogeneity of the turbulent

flow field into account, a better agreement could be obtained than by using only a one-compartment model. However, it also demonstrated that the "right" choice of the adaption parameters in the coalescence and breakage models is a crucial point concerning the quality of the agreement between experimental and simulated drop size distributions.

6.2 Outlook

The different investigations in this thesis showed that there are a lot of starting points for improvements concerning the modeling and simulation of the coupled system.

First of all, the coupling between FeatFlow and Parsival should be realized with a turbulence model so that for the mean energy dissipation rates, which are needed for the calculation of the coalescence and breakage phenomena with Parsival, values from the flow simulation can be used. Note that the simulator coupling has already been realized in such a way that, as soon as turbulence parameters are available from the CFD simulation, they can easily be included in the model (for details, see Chapter 5).
If the flow simulation is done without turbulence modeling, which means that direct numerical simulation is applied, a much finer grid is required. In fact, the number of grid points needed for a direct numerical simulation of the considered application is so high that, at the moment, this approach is not feasible, see the discussion in the beginning of Chapter 4 on page 46. However, it has to be investigated if one can avoid the enormous number of grid points needed for a direct numerical simulation by a local refinement of the grid. In [BBD00], a direct numerical simulation in combination with a locally refined grid consisting of about 1.9 million grid points was used for the simulation of the flow field in a stirred tank for a Reynolds number of Re = 7,275. A comparison of the simulation results with those obtained by using a k-ε turbulence model showed that direct numerical simulation was superior for this application, since the results were closer to experimental findings for almost all key features of the flow.
Since including a k-ε turbulence model into the moving boundary code in FeatFlow is very difficult (in particular because of the sensitivity of the k-ε model to the used "zig-zag"-approximation of the boundary, see Subsection 4.4.2), it is sensible to think about alternatives concerning both, the treatment of the moving boundaries as well as the turbulence modeling. As for the latter one, one possibility is to use a k-ω model instead, since this turbulence model is more robust with respect to the structure of the grid in the boundary regions. The k-ω model is still valid in the boundary layer near the wall (where the local Reynolds number is much smaller) and no further modeling in this area is needed. Concerning the treatment of the moving boundaries, several approaches can be found in the literature, whereof the impeller boundary condition method, the inner-outer iterative procedure (which is also known as multiple frame of reference method) and the sliding or clicking mesh method are the most popular ones, at least in chemical engineering science, see Remark 4.10 on page 63. These approaches have already been successfully applied in combination with a k-ε turbulence model, see, e. g., [JR99] for the impeller boundary condition method, [BBD00] for the inner-outer iterative procedure,

or [LSE00] and [MLBY01] for both, the inner-outer iterative procedure and the sliding mesh method.

Concerning the modeling, the description of the coalescence and breakage phenomena has to be improved in order to get rid of the undesired dependency on the adaption parameters, see Section 5.4. This can be achieved by including further dependencies on physical parameters (such as the pH value or the ionic strength), which can be detected in experiments (see, e. g., [KGS04] and [GWPK06a]), but are not yet included in most of the mathematical models. However, in order to determine the factors that have an influence on the resulting drop size distributions, systematic parameter studies in combination with appropriate experimental investigations are required. In addition, an understanding of the underlying physics of the processes is necessary in order to be able to derive an appropriate (physically based) extension of the model.

Another way to improve the simulation results is to increase the number of compartments in order to refine the modeling of the inhomogeneous turbulent flow field in the stirred tank. Furthermore, it would be desirable to do the division into compartments in such a way that it goes hand in hand with the grid used for the flow simulation, since in this case, the volume flows needed for the coupling can be calculated more precisely. An ideal approach would be to use the division of the compartment model also as coarse grid in the multigrid algorithm for the CFD simulation. However, in this approach, much more compartments are needed, which is not manageable in Parsival, since the simulation would take a too long time.

On the level of simulation, a starting point for further improvements is given by an automatization of the coupling. Here, the exchange of data via writing into and reading from files should be substituted by a direct "connection" between the two solvers. It would be desirable to include the calculation of the drop size distributions into the CFD simulation. However, since it seems to be hardly possible to "call" Parsival from FeatFlow during the simulation, other approaches for the solution of the population balance equation should be considered.

In this thesis, it was assumed that the influence of the drops on the flow field is negligible. Although this simplification is justified for the considered application, in general, the modeling of stirred liquid-liquid systems can be enhanced by taking this influence into account. In this case, the Reynolds-averaged Navier-Stokes equations must be extended by a term that models the influence of the drops on the flow field. Thus, not only the results of the CFD simulation are used for the calculation of the drop size distributions, but also the results of the population balance solver are used for the computation of the flow field.
In [VDV02], a so-called "1.5-way coupling" was used for the population balance modeling in an aerated stirred tank, which means that after the flow field had been calculated with Fluent, the velocities as well as the turbulence parameters derived by a single-phase flow simulation were corrected in order to take the influence of the gas bubbles into account. However, for a more detailed modeling of the interactions between the two phases, a "real" two-way coupling is required.

In the literature, two-way coupling approaches are usually modeled via two-phase or multi-phase flow simulations, since, in this case, also the assumption that the drops are just moving with the same velocity as the continuous phase is not satisfied any more. Thus, transport equations for both phases, the continuous and the dispersed one, have to be solved. If one assumes that all drops are moving with the same velocity, a two-phase model suffices, otherwise, additional equations for each "group" of drops moving with the same velocity are required.
An introduction to the numerical modeling of multi-phase flows can be found in [Wör03]. Further information concerning two- or multi-phase fluid flow modeling can, e. g., be found in [Ish75], [Dre83], [SD03], [Pas04b] and references therein.

For the solution of the population balance equation in the framework of multi-phase flows, several approaches can be found.
In general, the particles of the dispersed phase can be divided into several groups (usually according to their size) and each group can be treated as a separate phase in a multi-phase flow calculation. However, in this case, an additional set of coupled equations (continuity and momentum) has to be solved for each group of particles so that only a few groups can be handled (typically 2–3 dispersed phases), which leads to a poor representation of the distribution.
In the MUSIG (MUltiple SIze Group) model, introduced in [Lo96], it is therefore assumed that all particle velocities are the same so that for all particles together only one set of momentum equations has to be solved. Thus, essentially, the multi-phase model is reduced to a two-fluid approach with one velocity field for the continuous and one for the dispersed phase. However, we still have a continuity equation for each group of particles to represent the size distribution. With this approach, a larger number of groups (between 10 and 20) can be considered leading to a better representation of the particle size distribution. Applications of the MUSIG model for gas-liquid reactors can, e. g., be found in [Lo00a], [Lo00b], and [Fra05].
Another approach that can be used for the solution of the population balance equation in this context is the (direct) quadrature method of moments, see Section 5.6. In this approach, the internal coordinate is "integrated out" and the population balance equation is determined through its moments. A great advantage of this approach is that it usually suffices to include a few scalar transport equations (by which the moments are determined) into the CFD simulation in order to get a satisfactory approximation of the whole drop size distribution. Such a coupling between CFD and the solution of population balance equations has already been implemented for certain applications, see, e. g., [MVF03a], [MPF$^+$03], [FMF04], [ZMBF04], [ZMBF06], [MSS$^+$06], or [MSSM06].

The long-term objective of the project is the control of the drop size distribution in the stirred tank, which requires at least a two-way coupling (see Remark 2.9 at the end of Chapter 2) as well as the integration of a turbulence model (otherwise, the flow simulation cannot be handled due to the enormous simulation times). Thus, it is clear that the simulation of the coupled system should be realized in such a way that these extensions can be included in the model, which is hardly possible with the presented approach, as the investigations in this thesis have shown.

In general, but especially concerning the control of the drop size distribution, the ideal solution would be to solve all equations together as one coupled system, i. e., to use a so-called monolithic model instead of a simulator coupling approach, see Remark 5.1 in the beginning of Chapter 5. In this context, it would also be recommended to apply a method of lines approach, i. e., to do the space before the time discretization, so that the dynamics of the system can be treated better. Furthermore, it is clear that some kind of model reduction has to be applied due to the huge size of the coupled system.
However, for a practical realization of such a control problem, it is recommended to use an already existing CFD code due to the high effort to build a new CFD solver. Thus, there are some restrictions concerning the integration of a solution method for the population balance equation, since there are already structures in the code that can hardly be changed. In this case, a feasible approach is to include the population balance equation by means of the (direct) quadrature method of moments into the CFD code, which can simply be realized by implementing a few additional scalar transport equations.

Altogether, the conclusion can be drawn that the simulator coupling approach presented in this thesis was a good first approach to investigate the problem. However, as soon as some of the above mentioned extensions concerning modeling and simulation are required (e. g., for a more detailed description of the processes or for including control), this approach cannot be used any more.
In this thesis, we investigated the pros and cons of the presented approach and showed alternative methods, by which the modeling and simulation of the considered problem can be improved in the future.

A The Mathematical Description of Flows: Navier-Stokes and Reynolds-Averaged Navier-Stokes Equations

In this appendix, we will consider how the flow of fluids can be described mathematically.

In the first section, we will derive the Navier-Stokes equations from conservation laws for mass and momentum and give a dimensionless representation of them. In the second section, we will consider the Reynolds-averaged Navier-Stokes equations as one way to model turbulence.

This appendix is compiled from different sections of the book by Griebel et al. [GDN98].

A.1 The Navier-Stokes Equations

The Navier-Stokes equations are the basic equations of fluid dynamics modeling the flow of viscous incompressible fluids. They describe how velocity and pressure depend on space and time. The Navier-Stokes equations consist of two equations: the so-called continuity equation and the momentum equation.

A.1.1 Derivation of the Navier-Stokes Equations

In this subsection, we consider the derivation of the Navier-Stokes equations from conservation laws for mass and momentum.

For this, we consider a fluid that occupies the arbitrary (open and bounded) subdomain $\Omega_0 \subset \mathbb{R}^3$ at time $t = 0$. The function $\Psi : \Omega_0 \times [0, t_{end}] \to \Omega_t \subset \mathbb{R}$ describes the motion of the particle positions $\psi \in \Omega_0$ with time, so that, at time $t \geq 0$, the fluid occupies the domain $\Omega_t := \{\Psi(\psi, t) : \psi \in \Omega_0\}$. Therefore, Ω_t is a closed system, which means that no fluid particle flows across its boundaries. The path of a particle $\psi \in \Omega_0$ is the graph of the function $t \mapsto \Psi(\psi, t)$. The velocity of the fluid at a fixed position $x := \Psi(\psi, t) \in \Omega_t$ is given by

$$u(x, t) = \frac{\partial}{\partial t}\Psi(\psi, t).$$

The transport theorem states how the time derivative of an integral over a time-dependent domain can be computed.

Theorem A.1 (Transport theorem)**.** *For a differentiable scalar function* $F : \Omega_t \times [0, t_{end}] \to \mathbb{R}, (x,t) \mapsto F(x,t)$*, the following identity holds:*

$$\frac{d}{dt}\int_{\Omega_t} F(x,t)\,dx = \int_{\Omega_t} \left\{ \frac{\partial}{\partial t}F + \nabla\cdot(Fu) \right\}(x,t)\,dx.$$

Proof. Transformation of the integral over Ω_t to Ω_0 before the differentiation and back-transformation afterwards (see [GDN98]). □

Conservation of mass

The mass of a fluid occupying a certain domain is determined by the integral over the density ρ of the fluid. Since the same amount of fluid that occupies the domain Ω_0 at time $t = 0$ later, at time $t > 0$, occupies the domain Ω_t, the following equation must hold for all $t > 0$:

$$\int_{\Omega_0} \rho(x,0)\,dx = \int_{\Omega_t} \rho(x,t)\,dx.$$

This means the derivative of mass with respect to time must be zero. Using the transport theorem we get for all Ω_t, $t \geq 0$:

$$0 = \frac{d}{dt}\int_{\Omega_t} \rho(x,t)\,dx = \int_{\Omega_t} \left\{ \frac{\partial}{\partial t}\rho + \nabla\cdot(\rho u) \right\}(x,t)\,dx \qquad \forall\,\Omega_t,\ \ t \geq 0.$$

Since this equation holds for arbitrary regions Ω_t, it follows that

$$\frac{\partial}{\partial t}\rho + \nabla\cdot(\rho u) = 0. \tag{A.1}$$

This is the so-called continuity equation for compressible fluids. For incompressible fluids (which means that the density of the fluid is independent of space and time, i. e., $\rho(x,t) = \rho_f = \text{const.}$), Eq. (A.1) simplifies to

$$\nabla\cdot u = 0. \tag{A.2}$$

Conservation of momentum

The momentum of a solid body is given by the product of its mass and its velocity. In the case of a fluid system the velocity of the fluid may depend on space so that the momentum $i(t)$ of the fluid in the domain Ω_t must be calculated by the integral:

$$i(t) := \int_{\Omega_t} \rho(x,t)\,u(x,t)\,dx.$$

Newton's second law states that the rate of change of (linear) momentum equals the sum of forces acting on the fluid:

$$\frac{d}{dt}i(t) = \sum \text{acting forces.}$$

There are two types of forces:

- body forces (e. g., gravity, Coriolis force, magnetic force) given by

$$\int_{\Omega_t} \rho(x,t)\, g(x,t)\, dx$$

  with a given force density g per unit volume, and

- surface forces (e. g., pressure or internal friction) given by

$$\int_{\partial\Omega_t} \tau(x,t)\, \mathfrak{n}\, ds$$

  with a stress tensor $\tau = \begin{bmatrix} \tau_{11} & \tau_{12} & \tau_{13} \\ \tau_{21} & \tau_{22} & \tau_{23} \\ \tau_{31} & \tau_{32} & \tau_{33} \end{bmatrix}$ and normal vector $\mathfrak{n}$.

Newton's second law thus reads

$$\frac{d}{dt}\int_{\Omega_t} \rho(x,t)u(x,t)dx = \int_{\Omega_t} \rho(x,t)g(x,t)dx + \int_{\partial\Omega_t} \tau(x,t)\mathfrak{n}ds.$$

Applying first the transport theorem and then the product rule to the left-hand side of this equation and the Gaussian integral theorem to the second summand of the right-hand side, it follows that

$$\frac{\partial}{\partial t}(\rho u) + (u\cdot\nabla)(\rho u) + (\rho u)\nabla\cdot u - \rho g - \nabla\cdot\tau = 0. \tag{A.3}$$

The character of this equation is mainly determined by the way the stress tensor τ is modeled.
When inviscid fluids are modeled, internal friction can be neglected and the stress tensor τ is only determined by pressure. This leads to the Euler equations, which are mainly used in gas dynamics.
For viscous fluids the friction forces cannot be neglected. For a Newtonian fluid we get in this case (using Stokes' assumption) a viscous part τ_{visc} in the stress tensor τ, which is given by the following expression:

$$\tau := -pI + \tau_{\text{visc}} := (-p + \tilde{\mu}\nabla\cdot u)I + 2\mu\tau_{\text{strain}}.$$

Here, μ and $\tilde{\mu}$ are thermodynamic material constants and the strain tensor τ_{strain} is given by $\tau_{\text{strain}} := \frac{1}{2}\left(\nabla u + (\nabla u)^T\right)$. With this, (A.3) leads to

$$\frac{\partial}{\partial t}(\rho u) + (u\cdot\nabla)(\rho u) + (\rho u)\nabla\cdot u + \nabla p = (\mu+\tilde{\mu})\nabla(\nabla\cdot u) + \mu\Delta u + \rho g,$$

where we have used the identities $\nabla\cdot\nabla u = \Delta u$, $\nabla\cdot(\nabla u)^T = \nabla(\nabla\cdot u)$, and $\nabla\cdot[(\nabla\cdot u)I] = \nabla(\nabla\cdot u)$.

For incompressible fluids ($\rho(x,t) = \rho_f = \text{const.}$) and by using the continuity equation, we get the momentum equation

$$\frac{\partial}{\partial t}u + (u \cdot \nabla)u + \frac{1}{\rho_f}\nabla p = \frac{\mu}{\rho_f}\Delta u + g. \tag{A.4}$$

Here μ is the dynamic viscosity and $\nu := \frac{\mu}{\rho_f}$ is the kinematic viscosity.

The continuity equation (A.2) and momentum equation (A.4) together are called the Navier-Stokes equations for incompressible fluids.

A.1.2 Dimensionless Representation

Descriptive parameters like the dynamic viscosity μ or characteristic values for the length L_b, the velocity u_f, and the density ρ_f give information about the flow field for a certain application. To be able to say something about the correlation between some kind of flow in a large scale and the same kind of flow in a small scale, these parameters have to be put together appropriately in order to form dimensionless quantities. The dimensionless quantities are formed from the corresponding dimensional quantities by

$$\text{dimensionless quantity} = \frac{\text{dimensional quantity}}{\text{reference quantity with the same physical unit}}.$$

The used reference quantities should be constant as well as characteristic for the problem, and they have to be known in advance.

Let us now consider the incompressible Navier-Stokes equations and introduce the following dimensionless quantities:

$$x^* := \frac{x}{L_b}, \quad t^* := \frac{u_f t}{L_b}, \quad u^* := \frac{u}{u_f}, \quad p^* := \frac{p - p_f}{\rho_f u_f^2},$$

which correspond to the given scalar quantities L_b, u_f, p_f, and ρ_f. In a wind tunnel, for example, L_b can describe the length of the object in the stream and u_f, p_f, and ρ_f can be given by the velocity, the pressure, and the density of the fluid in the inflow region.

After transformation to these variables we get

$$\frac{\partial}{\partial t^*}u^* + (u^* \cdot \nabla^*)u^* + \nabla^* p^* = \frac{\mu}{\rho_f u_f L_b}\Delta^* u^* + \frac{L_b}{u_f^2}g,$$

where the operators ∇^* and Δ^* are given with respect to x^* instead of x.

Since this equation only depends on the two parameter groupings on the right-hand side, it is clear that two flows behave equivalently if both parameter groupings have the same value.

Corollary A.2. *Flows in similar geometries $\Omega_2 = L_b^{-1} \cdot \Omega_1, L_b > 0$, are dynamically similar if each of the parameters μ, u_f, ρ_f, L_b and g of the two flows are such that the dimensionless quantities*

$$\underset{(\textit{Reynolds number})}{\mathrm{Re} := \frac{\rho_f u_f L_b}{\mu}} \qquad \textit{and} \qquad \underset{(\textit{Froude number})}{\mathrm{Fr} := \frac{u_f}{\sqrt{L_b \|g\|_2}}}$$

of the flows are matching.

By introducing the dimensionless body force

$$g^* := \frac{L_b}{u_f^2} g \quad (= \frac{1}{\mathrm{Fr}^2} \frac{g}{\|g\|_2})$$

and leaving out the asterisks, we get the dimensionless momentum equation

$$\frac{\partial}{\partial t} u + (u \cdot \nabla) u + \nabla p = \frac{1}{\mathrm{Re}} \Delta u + g. \tag{A.5}$$

Continuity equation (A.2) together with the dimensionless momentum equation (A.5) are called the dimensionless Navier-Stokes equations.

The two dimensionless quantities, the Reynolds number and the Froude number, both describe properties of the flow.
The Reynolds number specifies the ratio between inertial and viscous forces. For $\mathrm{Re} \approx 0$, for example, the inertial forces are so small compared to friction that they can be neglected. In this case, we have a highly viscous fluid. On the other hand, for very large Reynolds numbers (up to 10^{10} or larger), friction can be neglected (like for air).
The Froude number describes the ratio between inertial and gravitational forces.

A.2 Modeling Turbulence: the Reynolds-Averaged Navier-Stokes Equations

In this section, we will consider the Reynolds-averaged Navier-Stokes equations as one way how turbulent flows can be modeled.

One of the characteristics of turbulent flows is that they are instationary, three-dimensional, irregular, chaotic, and unpredictable. Turbulent flows are characterized by the occurrence of eddies of different sizes. The size of the largest eddies is determined by the characteristic length of the domain, whereas the smallest eddies have sizes of about one millimeter or even less. The main part of the energy can be found in the larger eddies. From there, it is transported to smaller and smaller eddies until the smallest eddies are reached and the energy is transformed to heat. This phenomenon is called energy cascade.

Theoretically, the flow field of every incompressible, viscous fluid can be described by the Navier-Stokes equations, regardless if it is laminar or turbulent. Therefore, an obvious way to compute a turbulent flow field is to discretize and solve the Navier-Stokes equations on such a fine grid that even the smallest eddies are resolved. This approach is known as direct numerical simulation (DNS).

However, since the size of the smallest eddies is much smaller than the size of the computational domain, many grid points are needed to resolve the mixing on this level. The ratio between the smallest and the largest eddies is dependent on the Reynolds number Re, which is a measure for turbulence: the higher the Reynolds number, the more turbulent is the flow. The number of required grid points is proportional to $\mathrm{Re}^{9/4}$. Therefore, direct numerical simulation can only be used for simple geometries or problems with low (or medium) Reynolds numbers, see [RXY+02].

A.2.1 Basics of Turbulence Modeling

The following idea goes back to Reynolds [Rey95] and is the foundation of almost every turbulence model: All quantities occurring in the Navier-Stokes equations – i. e., the velocity u, the pressure p, and the right-hand side g – are decomposed into a mean part $\bar{u}$, $\bar{p}$, and $\bar{g}$, respectively, and into the small and smallest variations u', p', and g' (the so-called fluctuations) so that the following holds:

$$u = \bar{u} + u', \quad p = \bar{p} + p', \quad g = \bar{g} + g'.$$

Here, the mean part can be a statistical, temporal, or spatial average, for example. In general, the mean part is formed by applying a filter $\langle\cdot\rangle$. This means:

$$\bar{u} := \langle u\rangle, \quad \bar{p} := \langle p\rangle, \quad \bar{g} := \langle g\rangle,$$

and therefore,

$$u' = u - \langle u\rangle, \quad p' = p - \langle p\rangle, \quad g' = g - \langle g\rangle.$$

Note that for the vectors u and g, the filter is applied componentwisely.

This filter usually has the following properties:

- $\langle\cdot\rangle$ is a linear operator: $\langle\alpha_1 F_1 + \alpha_2 F_2\rangle = \alpha_1\langle F_1\rangle + \alpha_2\langle F_2\rangle$ for all quantities F_1, F_2 and all $\alpha_1, \alpha_2 \in \mathbb{R}$.
- Time and spatial derivatives commute with the filtering operator, i. e., $\langle\frac{\partial F}{\partial t}\rangle = \frac{\partial\langle F\rangle}{\partial t}$, $\langle\frac{\partial F}{\partial x_i}\rangle = \frac{\partial\langle F\rangle}{\partial x_i}$ for all quantities F and all $t, x_i \in \mathbb{R}$, $t \geq 0$.
- $\langle\langle F\rangle\rangle = \langle F\rangle$ for all quantities F.
- $\langle F_1\langle F_2\rangle\rangle = \langle F_1\rangle\langle F_2\rangle$ for all quantities F_1, F_2.

To come back to one of the examples mentioned above, the mean value $\langle F \rangle$ of a quantity $F(t)$ can be obtained by making a time average over a time interval $\mathcal{T}$ (cf., e. g., [BSL02]):

$$\langle F \rangle = \frac{1}{\mathcal{T}} \int_{t-\frac{1}{2}\mathcal{T}}^{t+\frac{1}{2}\mathcal{T}} F(\tau) d\tau.$$

Let us now apply the filter $\langle \cdot \rangle$ to the Navier-Stokes equations. Doing this, we get for the continuity equation:

$$0 = \nabla \cdot \langle u \rangle = \nabla \cdot \bar{u}.$$

Before we apply the filter $\langle \cdot \rangle$ to the momentum equation (A.4), we first transform it to vector notation:

$$\frac{\partial}{\partial t} u + \nabla \cdot (u \otimes u) + \frac{1}{\rho_f} \nabla p = \nu \Delta u + g, \tag{A.6}$$

where we have used the equality $(u \cdot \nabla) u = \nabla \cdot (u \otimes u)$. Here, $u \otimes u$ denotes a tensor with elements $[u \otimes u]_{ij} = u_i u_j,\ i, j = 1, 2, 3.$

Applying the filtering operator to Eq. (A.6) then leads to

$$\frac{\partial}{\partial t} \langle u \rangle + \nabla \cdot \langle u \otimes u \rangle + \frac{1}{\rho_f} \nabla \langle p \rangle = \nu \Delta \langle u \rangle + \langle g \rangle.$$

Using $\langle u \otimes u \rangle = \langle (\bar{u} + u') \otimes (\bar{u} + u') \rangle = \bar{u} \otimes \bar{u} + \langle u' \otimes u' \rangle$, we get the so-called Reynolds-averaged Navier-Stokes equations for the mean velocity $\bar{u}$

$$\frac{\partial}{\partial t} \bar{u} + \nabla \cdot (\bar{u} \otimes \bar{u}) + \frac{1}{\rho_f} \nabla \bar{p} - \nu \Delta \bar{u} + \nabla \cdot \langle u' \otimes u' \rangle = \bar{g}, \tag{A.7}$$

$$\nabla \cdot \bar{u} = 0. \tag{A.8}$$

The so-called Reynolds stresses are defined by the tensor

$$R(u') := -\langle u' \otimes u' \rangle.$$

Therefore, the averaging has lead to almost the same equations, just with the additional term $-\nabla \cdot R(u')$. Since R depends on the fluctuations u' of the velocities, the system consists of one additional unknown u' so that the Reynolds-averaged Navier-Stokes equations do not form a closed system any more. (There are more unknowns than equations.) This problem is known as the closure problem in turbulence modeling.

In order to solve this problem, one has to introduce new equations, which are usually based on hypotheses and approximations given by empirical information or experimental data. These equations form the so-called turbulence model. There are a lot of different turbulence models, including the k-ε model, which will be considered in the next subsection.

In general, there are basically two principles of turbulence modeling (see, e. g., [RXY+02]). The first approach is to model every component of the Reynolds stress tensor by its own

transport equation leading to the so-called Reynolds stress models, whereas in the second approach the Reynolds stress tensor is modeled by means of the following relation

$$-\langle u' \otimes u' \rangle = -\frac{2}{3} kI + \nu_t (\nabla \bar{u} + (\nabla \bar{u})^T).$$

Here, the turbulent viscosity ν_t is defined in analogy to the molecular viscosity in the laminar case. The latter models can further be distinguished into zero-, one-, and two-equation models depending on how many transport equations are additionally introduced. Theoretically, there are also models with more than two additional transport equations (see [LS72], for example), but in practice they are hardly used.

The most simple approach is to model the turbulent viscosity ν_t by an algebraic equation. These models are known as algebraic turbulence models or zero-equation models.
In the so-called one-equation models, an additional transport equation, either for the turbulent viscosity ν_t or for the turbulent kinetic energy k, is solved. This allows to model the transport of turbulence.
In the two-equations models, two transport equations for two different quantities are solved. This allows to reproduce the two most important characteristics of turbulence, intensity and structure. The most popular two-equations model is the k-ε model by Launder and Spalding [LS74], but there are also other approaches like the k-ω model by Wilcox [Wil88].

A.2.2 The k-ε Model

The k-ε model by Launder and Spalding [LS74] is a two-equations model, i. e., it introduces two additional variables k and ε (and a transport equation for each of them) to model the Reynolds stresses. The two variables are the turbulent kinetic energy k and the energy dissipation rate ε, which are given by

$$k := \frac{1}{2} \langle \|u'\|_F \rangle, \quad \varepsilon := \frac{\nu}{2} \langle \|\nabla u' + (\nabla u')^T\|_F^2 \rangle,$$

where $\|\cdot\|_F$ denotes the Frobenius norm, defined by

$$\|X\|_F^2 = \operatorname{tr}(XX^T) = \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij}^2$$

for matrices $X = [x_{ij}]_{i=1,\ldots,n, j=1,\ldots,m} \in \mathbb{R}^{n \times m}$. Note that this definition also includes the norm of a vector $x \in \mathbb{R}^n$, since it can be seen as an $n \times 1$-matrix.

The k-ε model underlies the following assumptions:

- The Reynolds hypothesis holds, i. e., it is assumed that $R(u') = -\langle u' \otimes u' \rangle$ depends only on k, ε, and $\nabla \bar{u}$:

$$R(u') \approx \mathcal{R}(\nabla \bar{u} + (\nabla \bar{u})^T, k, \varepsilon).$$

- The fluctuations u' (and hence turbulence) are isotropic, i. e., they are uniform in all directions.
- Convective transport of the fluctuations corresponds to a diffusive transport of the mean quantities.
- $\langle u' \otimes u' \rangle$ and $\nabla \bar{u} + (\nabla \bar{u})^T$ are proportional, and the proportionality factor is the turbulent eddy viscosity

$$\nu_t := c_\mu \frac{k^2}{\varepsilon}.$$

Putting the approximate Reynolds tensor

$$R(u') \approx \mathcal{R}(\nabla \bar{u}, k, \varepsilon) := -\frac{2}{3} k I + c_\mu \frac{k^2}{\varepsilon} (\nabla \bar{u} + (\nabla \bar{u})^T)$$

into the momentum equation (A.7) and setting $\nu^* := \nu + \nu_t = \nu + c_\mu \frac{k^2}{\varepsilon}$, leads to

$$\frac{\partial}{\partial t} \bar{u} + \nabla \cdot (\bar{u} \otimes \bar{u}) + \frac{1}{\rho_f} \nabla \bar{p} + \frac{2}{3} \nabla k - \nabla \cdot (\nu^* (\nabla \bar{u} + (\nabla \bar{u})^T)) = \bar{g}. \tag{A.9}$$

To determine k and ε, the following transport equations are used:

$$\frac{\partial}{\partial t} k + \bar{u} \cdot \nabla k - \frac{\nu_t}{2} \| \nabla \bar{u} + (\nabla \bar{u})^T \|_F^2 - \nabla \cdot (\nu_t \nabla k) + \varepsilon = 0, \tag{A.10}$$

$$\frac{\partial}{\partial t} \varepsilon + \bar{u} \cdot \nabla \varepsilon - \frac{c_1 c_\mu}{2} k \| \nabla \bar{u} + (\nabla \bar{u})^T \|_F^2 - \nabla \cdot (\frac{1}{c_\varepsilon} \nu_t \nabla \varepsilon) + c_2 \frac{\varepsilon^2}{k} = 0, \tag{A.11}$$

where the constants c_μ, c_ε, c_1, and c_2 are usually chosen empirically. In [LS74] the following values are proposed:

$$c_\mu = 0.09, \quad c_\varepsilon = 1.3, \quad c_1 = 1.44, \quad c_2 = 1.92,$$

whereas in [GDN98] some slightly different values are used. There, the product $c_1 c_\mu$ is set to 0.126 and c_ε is set to $\frac{0.09}{0.07} \approx 1.2857$, because for these values three well-known flow cases (grid turbulence, turbulent shear stress, and turbulent flow over a plate) are modeled correctly.

A.2.3 Boundary Conditions for the k-ε Model

To solve the system of partial differential equations, consisting of the continuity equation (A.8), the momentum equation (A.9), and the equations for the turbulent kinetic energy k (A.10) and its dissipation ε (A.11), appropriate boundary conditions have to be specified.
The most simple and natural approach would be to prescribe values for k and ε or for their normal derivatives along the boundary. But there is a problem: near the wall the

k-ε model is not valid any more. One reason for this is that, near the wall, small local Reynolds numbers occur because of the low velocities there. The k-ε model, however, is only valid for large Reynolds numbers. Another reason is that near solid walls turbulent variations and fluctuations normal to the wall are damped such that another assumption of the k-ε model, the isotropy of turbulence, is not fulfilled any more.

In turbulent flows, the thin boundary layer that is formed at fixed walls can be divided into three parts:

- Directly at the wall, the velocity of the fluid is very small because of adhesion. (The flow can be considered as laminar there.)
- Next to this region, there is a transition zone, in which the flow becomes locally unstable. (Here, the velocity profile can be considered as logarithmic.)
- Further away from the wall, there is the turbulent region.

There are different approaches to model the flow field in these regions, namely:

- with wall functions,
- by low-Reynolds-number models, or
- by two-layer models.

A survey on these approaches can be found in [GDN98] and references therein.

B The Mathematical Description of Dispersed-Phase Systems: a General Population Balance Equation

In this appendix, we will consider how dispersed-phase systems can be modeled. The following investigations are according to Gerstlauer [Ger99].

Dispersed-phase systems are two- (or more-) phase systems that consist of (at least) a continuous and a dispersed phase. The continuous phase surrounds the dispersed one, which is given by the so-called population of individuals. Every individual interacts with the continuous phase by exchanging heat, mass, and momentum.

When modeling dispersed systems, one does not only need time and space coordinates, but also so-called internal (or property) coordinates describing the characteristics of the dispersed phase. While the space coordinates, which are also called external coordinates in this context, specify the position of the individual in the space, the internal coordinates describe an arbitrary number of significant characteristics of the particle.
Both internal and external coordinates span a space, the so-called particle phase space, see [HK64]. Any individual can be represented by a point in this particle phase space. The coordinates of this point identify the phase of the particle – i. e., location and characteristics – uniquely.

For the mathematical description of the dispersed phase, we need, in addition, a so-called number density function, denoted by f in the following, that describes the expected number of individuals that are located in a certain domain of the particle phase space at a certain time t. Therefore, f is dependent on the internal coordinates e, the external coordinates x and the time t, i. e., $f = f(e, x, t)$. The total number of individuals $N_{\mathrm{ind}}(t)$ at time t is then given by integration over the whole geometrical space Ω_x and the whole space of internal coordinates Ω_e:

$$N_{\mathrm{ind}}(t) = \int_{\Omega_x} \int_{\Omega_e} f(e, x, t)\, de\, dx. \tag{B.1}$$

B.1 Derivation of a General Population Balance Equation

In this section, the derivation of a general population balance is considered. Population balances represent a modeling approach that is tailored to dispersed systems.

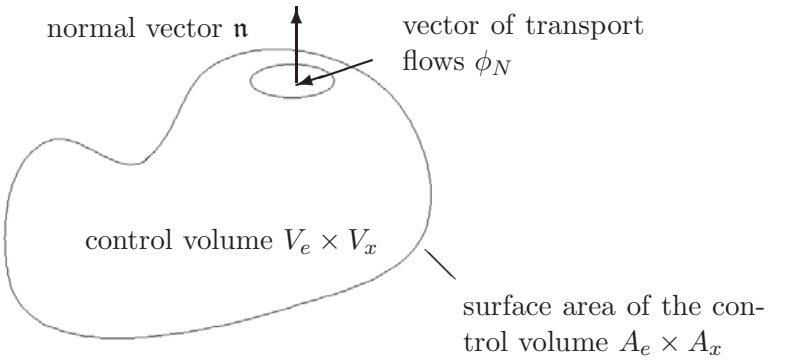


Figure B.1: Control volume

In the following, the number density function is assumed to be a continuous function. At first sight, this assumption seems to be problematic, since the number density function describes the number of individuals, which is normally given by an integer. However, it has been shown that the number density function can be considered as an average number of individuals. That is why the restriction to integer values can be lifted and the above made assumption is justified.
Therefore, a general population balance equation can be derived in an analogous way as setting up equations for mass, energy, and momentum.

The starting point for our investigations is the general integral balance equation for the total number of individuals $N_{\text{ind}}(t)$ in any fixed control volume $V_e \times V_x$ of the particle phase space

$$\frac{dN_{\text{ind}}}{dt} = \Phi_N + S_N. \tag{B.2}$$

This equation states that the rate of increase of the total number of individuals equals a transport term Φ_N plus a source term S_N. The transport term accounts for the number of individuals that are supplied from the surrounding to the control volume through the surface area $A_e \times A_x$ per time interval. The source term describes the number of individuals that are created or destroyed per time interval by processes taking place inside the considered control volume (e. g., coalescence or breakage).

In the following, a local form of (B.2) will be derived. According to (B.1), the number of individuals in the control volume can be expressed by:

$$N_{\text{ind}}(t) = \int_{V_x} \int_{V_e} f(e, x, t)\, de\, dx.$$

Differentiation leads to

$$\frac{dN_{\text{ind}}(t)}{dt} = \frac{d}{dt}\left[\int_{V_x} \int_{V_e} f(e, x, t)\, de\, dx\right].$$

The source term S_N on the right-hand side of Eq. (B.2) can be substituted by a volume integral

$$S_N = \int_{V_x} \int_{V_e} s(e,x,t)\, de\, dx,$$

where s is the source density referring to a volume element $de\, dx$ of the particle phase space. The transport term Φ_N can be replaced by

$$\Phi_N = -\int_{A_x} \int_{A_e} (\mathfrak{n} \cdot \phi_N(e,x,t))\, de\, dx.$$

Here, the term $-(\mathfrak{n} \cdot \phi_N)$ denotes normal component of the transport flow, which is supplied to the control volume $V_e \times V_x$ from the surrounding. Inserting this into Eq. (B.2) results in

$$\frac{d}{dt} \int_{V_x} \int_{V_e} f(e,x,t)\, de\, dx = -\int_{A_x} \int_{A_e} (\mathfrak{n} \cdot \phi_N(e,x,t))\, de\, dx + \int_{V_x} \int_{V_e} s(e,x,t)\, de\, dx.$$

The Gaussian integral theorem then leads to

$$\frac{d}{dt} \int_{V_x} \int_{V_e} f(e,x,t)\, de\, dx = -\int_{V_x} \int_{V_e} (\nabla_{x,e} \cdot \phi_N(e,x,t))\, de\, dx + \int_{V_x} \int_{V_e} s(e,x,t)\, de\, dx,$$

where $\nabla_{x,e}$ denotes the gradient with respect to external and internal coordinates.

Since $\nabla_{x,e} \cdot \phi_N$ has the same unit as the source density s, it is called transport density. The vector ϕ_N summarizes the different parts of the transport flow. They are counted positive if they are added to the control volume, and negative otherwise.

Since the stationary control volume $V_e \times V_x$ is a volume element of fixed size, the time derivative can be put into the integral as a partial derivative. This leads to

$$\int_{V_x} \int_{V_e} \frac{\partial f(e,x,t)}{\partial t}\, de\, dx = -\int_{V_x} \int_{V_e} \nabla_{x,e} \cdot \phi_N(e,x,t)\, de\, dx + \int_{V_x} \int_{V_e} s(e,x,t)\, de\, dx$$

or equivalently

$$\int_{V_x} \int_{V_e} \left(\frac{\partial f(e,x,t)}{\partial t} + \nabla_{x,e} \cdot \phi_N(e,x,t) - s(e,x,t) \right) de\, dx = 0.$$

Since the control volume can be chosen arbitrarily (small), we get the local form for a differential volume element

$$\frac{\partial f(e,x,t)}{\partial t} = -\nabla_{x,e} \cdot \phi_N(e,x,t) + s(e,x,t). \tag{B.3}$$

If we divide the transport density $\nabla_{x,e} \cdot \phi_N$ into a transport density with respect to the internal coordinates $\nabla_e \cdot \phi_{N,e}$ and one with respect to the external coordinates $\nabla_x \cdot \phi_{N,x}$, we get

$$\frac{\partial f(e,x,t)}{\partial t} = -\nabla_x \cdot \phi_{N,x}(e,x,t) - \nabla_e \cdot \phi_{N,e}(e,x,t) + s(e,x,t),$$

where the ∇-operators are defined as

$$\nabla_x := \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_3} \end{bmatrix} \quad \text{and} \quad \nabla_e := \begin{bmatrix} \frac{\partial}{\partial e_1} \\ \vdots \\ \frac{\partial}{\partial e_a} \end{bmatrix}.$$

Furthermore, the transport flows can be divided into a convective and a diffusive part:

$$\begin{aligned} \phi_{N,x} &= w_x \cdot f(e,x,t) + \phi^D_{N,x}, \\ \phi_{N,e} &= w_e \cdot f(e,x,t) + \phi^D_{N,e}. \end{aligned}$$

The convective part $w_x \cdot f(e,x,t)$ in x-direction describes the deterministic moving of the particles, whereas the diffusive part $\phi^D_{N,x}$ can be used to account for their Brownian motion.
The convective part $w_e \cdot f(e,x,t)$ in direction of the internal coordinates e states the rate of change of single particle properties. The diffusive part $\phi^D_{N,e}$ can be used to describe the stochastic variation of a rate of change.

The source term s summarizes different physical phenomena. On the one hand, there are formation and dissolution of individuals, which are due to mass exchange with the continuous phase. On the other hand, there are breakage (or attrition, respectively) as well as coalescence (or agglomeration or aggregation, respectively), which are affected by fluid dynamic processes and momentum exchange between the two phases. Therefore, s can be divided into the following source and sink terms

$$s = s^+_{\text{form}} + s^-_{\text{diss}} + s^+_{\text{break}} + s^-_{\text{break}} + s^+_{\text{coal}} + s^-_{\text{coal}}$$

accounting for formation and dissolution of individuals as well as for their breakage and coalescence. These terms will be specified in the following section.

B.2 Approach for Source and Sink Terms

The source term s on the right-hand side of Eq. (B.3) accounts for the different population dynamical phenomena. These phenomena can be divided into two groups:

- population dynamical phenomena due to heat and mass transfer between the continuous and the dispersed phase (e. g., formation, dissolution, growth, or shrinkage of individuals),
- population dynamical phenomena due to fluid dynamic processes and momentum transfer between the two phases (e. g., coalescence or breakage).

B.2.1 Formation

Considering dispersed systems, one can see that new individuals are formed due to mass transfer between the continuous and the dispersed phase. In the literature, these processes are often called "birth processes".

Mathematically, the formation of individuals is described by a formation or birth rate $R_{\text{form}}(e, y(x,t))$, which is dependent on the state of the continuous phase y. The vector y describes properties of the continuous phase that influence the population dynamical processes, like pressure, temperature, or chemical composition, for example. The birth rate accounts for the number of individuals that are formed per unit time and per volume element of the geometric space. The new individuals can be formed with different values in the internal coordinates. The distribution of the formed individuals can be described by a density function $f_{\text{form}}(e, x)$, which has to fulfill the following normalization condition

$$\int_{V_e} f_{\text{form}}(e, x)\, de = 1.$$

The source term due to formation is then given by

$$s^{+}_{\text{form}}(e, x, y) = R_{\text{form}}(e, y(x,t))\, f_{\text{form}}(e, x).$$

There is no sink term due to formation.

B.2.2 Dissolution

Dissolution is the inverse process to formation. In the literature, these processes are often called "death processes". The mathematical description of this process is done by introducing a dissolution or death rate $R_{\text{diss}}(e, y(x,t))$, which describes the number of individuals that dissolve per unit time and volume element of the geometric space. The dissolution rate is dependent on the internal coordinates of the individual that dissolves and on the state of the continuous phase y. The sink term due to dissolution is given by

$$s^{-}_{\text{diss}}(e, x, y) = -R_{\text{diss}}(e, y(x,t))\, f_{\text{diss}}(e, x),$$

where the number density function $f_{\text{diss}}(e, x)$ has to fulfill the following condition

$$\int_{V_e} f_{\text{diss}}(e, x)\, de = 1.$$

There is no source term due to dissolution.

B.2.3 Breakage

Breakage is a phenomenon where a larger individual breaks up into some smaller ones. The larger particle, which breaks up, is often called mother particle, whereas the newly

formed smaller particles are called daughter particles. Depending on the kind of considered system, the break-up process can be called dispersion, breakage, or attrition.

Mathematically, breakage can be modeled by a source and a sink term. The source term describes the generation of daughter particles, whereas the sink term stands for the loss of the mother particle.

If we assume that the breakage of one particle is independent of the breakage of another one, we can define a breakage rate $R_{\text{break}}(e, y(x,t))$ dependent on the internal coordinates e and the state of the neighborhood $y(x,t)$. Then, the sink term for the breakage of individuals is the product of the breakage rate with the number density function:

$$s^-_{\text{break}}(e,x,y,t) = -R_{\text{break}}(e, y(x,t))\ f(e,x,t). \tag{B.4}$$

This sink term describes the number of individuals in state (e,x) that break up per time interval and per volume of the particle phase space.

To describe the source term of the breakage, we additionally have to consider the number of produced individuals and the values of their internal coordinates. In the following, we will assume that the breakage of a mother particle at place x leads to daughter particles located at the same place. Then, the number of daughter particles $n(e', y(x,t))$ is dependent on the internal coordinates of the mother particle e' and on the state of the surrounding y. The distribution of the developing daughter particles with respect to the internal coordinates can be described by the probability density function $\gamma(e, e', y(x,t))$. The probability density function $\gamma(e, e', y(x,t))$ describes the probability that a daughter particle of state e is produced by the breakage of a particle of state e'. It has to fulfill the following normalization condition:

$$\int_{V_e} \gamma(e, e', y(x,t))\, de = 1.$$

The source term describing all the daughter particles that are produced by breakage is obtained by weighted integration over all possible sinks:

$$\begin{aligned} s^+_{\text{break}}(e,x,y,t) &= -\int_{V_{e^*}} n(e', y(x,t))\, \gamma(e, e', y(x,t))\, s^-_{\text{break}}(e', x, y, t)\, de' \\ &= \int_{V_{e^*}} n(e', y(x,t))\, \gamma(e, e', y(x,t))\, R_{\text{break}}(e', y(x,t))\, f(e', x, t)\, de'. \end{aligned} \tag{B.5}$$

Note that it is not integrated over the total domain of the internal coordinates e', but only over a partial domain, which is denoted by V_{e^*}.
When a specific case is considered, these integration limits are determined due to physical restrictions on the internal coordinates (like the maximal or minimal diameter of a drop, for example).

Conservation of mass leads to the following condition:

$$n(e', y(x,t)) \int_{V_e} m(e)\, \gamma(e, e', y(x,t))\, de = m(e'),$$

where $m(e)$ is the mass of an individual with internal coordinates e.

The breaking rate $R_{\text{break}}(e', y(x,t))$, the number of daughter particles $n(e', y(x,t))$ and the density function $\gamma(e, e', y(x,t))$ have to be determined for a specific application. The question whether there exist theoretically based approaches for these rates depends highly on what kind of dispersed system is considered. However, even in these cases, these functions must still additionally be fitted to experimental data for many applications.

B.2.4 Coalescence

Coalescence is a process, where some individuals collide and together form a new individual. Depending on the type of dispersed system this process can also be called agglomeration or aggregation.

Mathematically, coalescence (like breakage) is described by a source and a sink term. Since the probability that three or more individuals collide in a time interval $(t, t+dt)$ is negligibly small compared to the probability that two particles collide, we will assume binary coalescence in the following. Furthermore, we will assume that all individuals have an infinitesimal small size such that we do not have to distinguish between the positions of two coalescing individuals.

Under these assumptions the probability that an individual of state (e, x) coalesces with an individual of state (e', x) in a time interval $(t, t+dt)$ can be described by a function $R_{\text{coal}}(e, e', y(x,t))$. It is clear that the coalescence rate R_{coal} must be symmetric with respect to the internal coordinates e and e':

$$R_{\text{coal}}(e, e', y(x,t)) = R_{\text{coal}}(e', e, y(x,t)).$$

To be able to characterize the source and sink term for the coalescence of two individuals we need a term that describes the number of pairs of individuals with state (e, x) and (e', x) per volume of the particle phase space per time t. This number density function can be defined as follows

$$f_2(e, e', x, t) := f(e, x, t)\, f(e', x, t).$$

This product approach is correct if the individuals fulfill the statistical independency condition. (In this context, the statistical independency condition means qualitatively that the fact that a certain drop with internal coordinates e can be found at time t at place x does not have any influence on the probability that any other drop with internal coordinates e' is also there at the same time.) Otherwise it is a good approximation.

To get the mathematical model for the sink term due to coalescence, the loss of an individual of state (e, x) because of coalescence is considered. The integration over all individuals that are able to coalesce with a given one leads to

$$\begin{aligned} s^-_{\text{coal}}(e, x, y, t) &= -\int_{V_{e^*}} R_{\text{coal}}(e, e', y(x,t))\, f_2(e, e', x, t)\, de' \\ &= -f(e, x, t) \int_{V_{e^*}} R_{\text{coal}}(e, e', y(x,t))\, f(e', x, t)\, de'. \end{aligned} \tag{B.6}$$

The source term due to coalescence is given by:

$$\begin{aligned} s^{+}_{\text{coal}}(e,x,y,t) &= \int_{V_{e^*}} R_{\text{coal}}(e',e'',y(x,t))\, f_2(e',e'',x,t)\, de' \\ &= \int_{V_{e^*}} R_{\text{coal}}(e',e'',y(x,t))\, f(e',x,t)\, f(e'',x,t)\, de', \end{aligned} \tag{B.7}$$

where e' and e'' denote the internal coordinates of the two coalescing individuals. It is clear that the characteristics of the produced individual can be determined by the characteristics of the two coalescing individuals. Therefore, the state (e'',x) of one of the coalescing individuals can be defined by the state (e,x) of the produced individual and the state (e',x) of the other coalescing individual. This can be expressed by the following condition

$$e'' = e''(e,e',x),$$

which states that the state (e'',x) must be explicitly computable by the state of the two other concerned individuals.

Note that in Eq. (B.6) and Eq. (B.7) it is again only integrated over a partial volume V_{e^*} of the space of internal coordinates.

Furthermore, the coalescence rate $R_{\text{coal}}(e',e'',y(x,t))$ must be determined more precisely for every specific application.

Nomenclature

A	negative of Jacobian of F with respect to z, page 36
A_2	matrix function, page 37
A_e / A_x	surface area of control volume with respect to the internal / external coordinates, page 182
A_{ij}	area of the face lying between the compartments K_i and K_j, page 118
A_{ij}^{α}	area of the part of the face A_{ij} that belongs to node α, page 123
$\mathcal{A} = [a_{ij}]$	matrix in the product-difference algorithm, page 159
a	number of internal coordinates, page 6
$\hat{a}$	number of algebraic equations, page 36
B_{baffle}	length of the baffle, page 66
B_{blade}	length of the blade, page 66
B_w	wall thickness of the tank, page 56
$\mathcal{B} = [b_{ij}]$	matrix in product-difference algorithm, page 158
b	coefficient function in the differential-algebraic equation (3.5), page 34
$\hat{b}_i$	parameters in the product-difference algorithm, page 159
C	spatially discretized gradient operator, page 32
$\tilde{C}$	transposed of spatially discretized divergence operator, page 32
$\tilde{C}^-$	kernel of $\tilde{C}$, page 41
C_1, C_2	spatially discretized gradient operators, page 33
C_3	transposed of spatially discretized divergence operator, page 33
CFD	computational fluid dynamics, page 2
$\mathcal{C}$	space of continuous functions, page 35
c_1, c_2, c_ε, c_μ	constants in the k-ε model, page 10
$\bar{c}_1, \ldots, \bar{c}_6$	constants in the coalescence model, page 18
$\tilde{c}_1, \ldots, \tilde{c}_5$	constants in the breakage model, page 23
$\hat{c}_1$, $\hat{c}_2$	constants in the model for the distribution of daughter drops, page 27

$\check{c}_i$	constant in definition of $\bar{\varepsilon}_i$, page 151
$c_{\text{break},1}$, $c_{\text{break},2}$	adaption parameters in the breakage model, page 23
$c_{\text{coal},1}, c_{\text{coal},2}$	adaption parameters in the coalescence model, page 19
c_t	turbulent diffusion coefficient, page 16
c_V	shape factor, page 7
D_{baffle}	diameter of the baffle, page 5
D_{blade}	diameter of the blade, page 66
D_{disk}	diameter of the disk, page 66
D_{hub}	diameter of the hub, page 66
D_s	distance of the baffles to the wall, page 66
D_{shaft}	diameter of the shaft, page 66
D_{stirrer}	diameter of the stirrer, page 5
D_{tank}	diameter of the tank, page 5
DF	damping factor, page 21
$\mathbb{D}_z$	open set $\subset \mathbb{R}^{\hat{n}}$, page 34
$\mathbb{D}_{\dot{z}}$	open set $\subset \mathbb{R}^{\hat{n}}$, page 35
d	diameter of a drop, page 7
$\hat{d}$	number of differential equations, page 36
d_{32}	Sauter mean diameter, page 133
d_a	external diameter of the tank, page 56
d_e	size of an eddy, page 24
$d_{e,\min}$	minimal size of an eddy, page 25
$d_{\max}$ / $d_{\min}$	maximal / minimal drop diameter, page 25
E	Jacobian of F with respect to $\dot{z}$, page 36
E_{break}	energy needed for drop breakage, page 27
$E_{\text{break, max}}$ / $E_{\text{break, min}}$	maximal / minimal energy needed for drop breakage, page 27
$\bar{E}_{\text{eddy}}$	average energy of an eddy, page 25
$\bar{E}_{\text{kin}}$	mean turbulent kinetic energy of an eddy, page 23
E_σ	surface energy of a drop, page 23
$E_{\sigma,\max}$ / $E_{\sigma,\min}$	maximal / minimal energy needed for forming two daughter drops, page 25
$\bar{E}_\sigma$	average energy required for drop breakage, page 25
$\mathcal{E}$, $\mathcal{E}_1$, $\mathcal{E}_2$	matrices in the direct quadrature method of moments, page 162
$e = [e_1, \ldots, e_a]^T$	internal coordinates, page 6
e_α	abscissae of the quadrature approximation, page 158
F	function in the differential-algebraic equation (3.6), page 35

$\hat{F}$	function in the reduced strangeness-free differential-algebraic equation (3.7), page 37
$\hat{F}_1$ / $\hat{F}_2$	differential / algebraic part in Eq. (3.7), page 37
F_l	derivative array of F, page 36
Fr	Froude number, page 175
$\mathcal{F}$	force compressing the drops, page 19
f	number density function, page 11
$\bar{f}$	averaged number density function, page 16
f'	fluctuations of the number density function, page 16
$\hat{f}$	approximation of f in the finite-element type Galerkin method, page 106
f_0	initial value for the number density function, page 13
f_2	number density function for a coalescing pair, page 187
f_i	number density of drop size distribution in compartment K_i, page 118
f_{diss} / f_{form}	distribution of dissolved / formed individuals, page 185
f_e	approximation of f with respect to the internal coordinates e in the finite-element type Galerkin method, page 106
f_e^L	approximation of f with respect to the internal coordinates e on the L-th level in the finite-element type Galerkin method, page 106
f_h	spatially discretized averaged number density function, page 33
f^n	temporally discretized number density function, page 106
G	Jacobian, page 38
$\mathcal{G}$, $\mathcal{G}_1, \ldots, \mathcal{G}_5$	spatially discretized terms in Eq. (3.4), page 33
g	external forces, page 8
$\bar{g} = [\bar{g}_1, \bar{g}_2, \bar{g}_3]^T$	averaged external forces, page 9
$g' = [g'_1, g'_2, g'_3]^T$	fluctuations of the external forces, page 9
g^*	dimensionless external forces, page 175
$g_h = [g_{1,h}, g_{2,h}, g_{3,h}]^T$	spatially discretized external forces, page 32
g^n	temporally discretized external forces, page 48
$\tilde{g}^n$	right-hand side in the one-step θ-scheme, page 48
H_1, H_2	heights in the description of the compartments, page 121
H_{disk}	height of the disk, page 66
H_l	height of the bottom part of the tank, page 66
H_{stirrer}	height of the stirrer, page 66
H_{tank}	height of the tank, page 5

$\mathcal{H}$ . . . . . matrix in minimization problem, page 125

h . . . . . right-hand side in Eq. (5.3), page 106

h_0, h_1 . . . . . film thicknesses, page 19

$h_1, \ldots, h_3$ . . . . . heights in the description of the torispherical head, page 56

I . . . . . identity matrix, page 10

I_j^L . . . . . j-th interval in the subdivision of the e-axis on the L-th level in the finite-element type Galerkin method, page 106

$\mathbb{I}$ . . . . . time interval, page 34

i . . . . . momentum, page 172

im . . . . . image, page 34

$J_f(e)$ . . . . . Jacobian of the source term in the population balance equation with respect to the internal coordinates e, page 106

K_i . . . . . i-th compartment, page 118

$\mathcal{K}$ . . . . . function describing the torispherical head, page 57

k . . . . . turbulent kinetic energy, page 10

k_0 . . . . . initial value for the turbulent kinetic energy, page 28

k_h . . . . . spatially discretized turbulent kinetic energy, page 33

ker . . . . . kernel, page 34

L_b . . . . . characteristic length, page 174

$\mathcal{L}_{\hat{\mu}}$ . . . . . solution set, page 36

l_{ij}^L . . . . . Legendre polynomials of degree i (transformed to the interval I_j^L) used in the finite-element-type Galerkin method, page 107

M . . . . . mass matrix, page 32

$M_1, \ldots, M_4$ . . . . . mass matrices, page 33

M_l . . . . . lumped mass matrix, page 52

$\mathcal{M}_l$ . . . . . Jacobian of F_l with respect to $\dot{z}, \ldots, z^{(l+1)}$, page 36

m . . . . . mass (of a drop), page 15

m_k . . . . . k-th moment, page 156

m_h . . . . . spatially discretized mass, page 33

m^L . . . . . number of intervals on the L-th level in the finite-element type Galerkin method, page 107

N . . . . . convective and diffusive parts in the Navier-Stokes equations, page 49

N^* . . . . . rotational speed, page 8

$N_1, \ldots, N_7$, $\tilde{N}_4$, $\tilde{N}_5$ . . . . . spatially discretized nonlinear terms in the Reynolds-averaged Navier-Stokes equations, page 33

N_{ij} . . . . . number of nodes on face A_{ij}, page 123

N_{break} . . . . . number of breaking drops, page 23

N_{conv} / N_{diff} . . . . . discretized convective / diffusive part in the Navier-Stokes equations, page 32

N_{DNS} . . . . . number of grid points required for a direct numerical simulation (DNS), page 46

N_{drops} . . . . . total number of drops, page 11

N_{ind} . . . . . total number of individuals, page 181

N_K . . . . . number of compartments, page 118

N_{mom} . . . . . number of moments, page 157

Ne . . . . . power (or Newton) number, page 131

$\mathbb{N}_0$ . . . . . set of natural numbers including zero, page 36

$\mathcal{N}_l$ . . . . . negative of Jacobian of F_l with respect to z, page 36

n . . . . . number of daughter drops, page 15

$\hat{n}$ . . . . . number of differential-algebraic equations, page 35

n_e . . . . . number of eddies, page 24

$n_{e,m}$ . . . . . number of eddies per unit mass of the fluid, page 24

n_m / n_p / n_q . . . . . number of equations corresponding to m_h / p_h / q_h in Eq. (3.4), page 38

$\mathfrak{n} = [\mathfrak{n}_1, \mathfrak{n}_2, \mathfrak{n}_3]^T$ . . . . . outer normal vector, page 8

$\mathfrak{n}_{ij}$ . . . . . unit normal vector on A_{ij} in direction from compartment K_i to compartment K_j, page 118

P . . . . . regular preconditioning operator, page 52

$\mathcal{P}$ . . . . . spatially discretized pressure term, page 33

p . . . . . pressure, page 8

$\bar{p}$ . . . . . averaged pressure, page 9

p' . . . . . pressure fluctuations, page 9

p^* . . . . . dimensionless pressure, page 174

p_0 . . . . . initial value for the pressure, page 28

p_h . . . . . spatially discretized pressure, page 32

p^n . . . . . temporally discretized pressure, page 48

$p^{n,l}$ . . . . . iterate for the discrete pressure in the discrete projection method, page 52

$\hat{p}_j^L$ . . . . . order of the polynomial approximation of the number density function on the interval I_j^L in the finite-element-type Galerkin method, page 107

Q . . . . . Jacobian, page 38

Q^+ . . . . . co-range of Q, page 40

$Q_{0,i}$ . . . . . cumulative number distribution in compartment K_i, page 131

q . . . . . pressure correction (determined by means of the discrete Poisson equation), page 52

$q_h = [u_h^T, k_h^T, \varepsilon_h^T, f_h^T]^T$ . . . . . vector consisting of the spatially discretized quantities u_h, k_h, ε_h, and f_h, page 33

R . . . . . Reynolds stress tensor, page 9

R_1, R_2 . . . . . radii in the description of the compartments, page 121

R_{break} / R_{coal} . . . . . breakage / coalescence rate, page 13

R_{diss} / R_{form} . . . . . dissolution / formation rate, page 185

Re . . . . . Reynolds number, page 8

REL_{δ_t} . . . . . relative changes (used for the calculation of the step size in the adaptive step size control), page 54

$\mathbb{R}$ . . . . . set of real numbers, page 6

$\mathbb{R}^a$ . . . . . space of real a-dimensional vectors, page 6

$\mathbb{R}^{a_1 \times a_2}$ . . . . . space of real $a_1 \times a_2$-matrices, page 34

$\mathcal{R}$ . . . . . approximate Reynolds stress tensor, page 10

r . . . . . radius, page 8

r_1, r_2 . . . . . radii in the description of the torispherical head, page 56

r_{cyl} . . . . . radius of the cylindric part of the tank, page 58

r_s . . . . . osculation point of the two functions describing the torispherical head, page 57

r_{tol} . . . . . threshold radius for the "outer" boundary nodes, page 60

S . . . . . matrix in the discrete formulation of the Navier-Stokes equations, page 51

$\tilde{S}$ . . . . . approximate Frechét derivative of S, page 53

S_N . . . . . source term, page 182

Sc_t . . . . . turbulent Schmidt number, page 16

$\mathcal{S}$ . . . . . control term, page 29

s . . . . . sum of source and sink terms in population balance equation, page 11

$\bar{s}$ . . . . . sum of averaged source and sink terms in population balance equation, page 16

$\bar{s}_k$ . . . . . moment transform of s, page 157

$\bar{s}_k^{(N_{\text{mom}})}$ . . . . . approximation of $\bar{s}_k$, page 158

s_{break}^+ / s_{break}^- . . . . . source / sink term due to breakage, page 13

s_{coal}^+ / s_{coal}^- . . . . . source / sink term due to coalescence, page 13

s_{diss}^- . . . . . sink term due to dissolution, page 184

s^+_{form} . . . . . source term due to formation, page 184
T_2 . . . . . matrix function, page 37
TOL . . . . . tolerance parameter, page 54
$\mathcal{T}$ . . . . . time interval (for Reynolds-averaging), page 119
t . . . . . time coordinate, page 6
t^* . . . . . dimensionless time coordinate, page 174
t_n . . . . . discrete time points, page 48
t_b . . . . . breakage time, page 23
$\bar{t}_{\text{coal}}$ / $\bar{t}_{\text{contact}}$ . . . . . average coalescence / contact time, page 19
t_{end} . . . . . end time, page 6
U_{in} / U_{out} . . . . . control (or input) / output parameters, page 29
$\mathcal{U}$ . . . . . projector, page 34
$u = [u_1, u_2, u_3]^T$ . . . . . velocity, page 8
$\bar{u} = [\bar{u}_1, \bar{u}_2, \bar{u}_3]^T$ . . . . . averaged velocity, page 9
$u' = [u'_1, u'_2, u'_3]^T$ . . . . . velocity fluctuations, page 9
u^* . . . . . dimensionless velocity, page 174
$\tilde{u}$ . . . . . discrete velocity in the nonlinear transport-diffusion equation, page 52
$u_0 = [u_{1,0}, u_{2,0}, u_{3,0}]^T$ . . . . . initial value for the velocity, page 28
$\tilde{u}_l$ . . . . . iterate for the discrete velocity in the adaptive fixed point defect correction method, page 53
$\bar{u}_\alpha$ . . . . . mean velocity in node α, page 123
u_d . . . . . drop velocity, page 24
u_e . . . . . eddy velocity, page 24
u_f . . . . . characteristic velocity of the fluid, page 174
$u_h = [u_{1,h}, u_{2,h}, u_{3,h}]^T$ . . . . . spatially discretized velocity, page 32
u_{mean} . . . . . mean velocity, page 84
u_{stirrer} . . . . . velocity prescribed on the stirrer, page 8
$\overline{u^2}$ . . . . . mean square fluctuation velocity, page 18
u^n . . . . . temporally discretized velocity, page 48
$\tilde{u}^n$ . . . . . iterate for the discrete velocity in the discrete projection method, page 52
V . . . . . volume of a drop, page 7
V_i . . . . . volume of compartment K_i, page 118
V_{cyl} . . . . . volume of the cylindric part of compartment K_1, page 121
V_e / V_x . . . . . control volume with respect to the internal / external coordinates, page 182
V_{e^*} . . . . . partial domain $\subset V_e$, page 186

$V_{\text{Klöpper}}$ . . . . . volume of the torispherical head part of compartment K_1, page 121

$V_{\max}$ / $V_{\min}$ . . . . . maximal / minimal drop volume, page 13

V_{stirrer} . . . . . volume of the stirrer region, page 25

V_{tank} . . . . . volume of the tank, page 25

$\dot{V}_{ij}$ . . . . . mean volume flow between the compartments K_i and K_j, page 118

$\dot{V}_{ij}^k$ . . . . . mean volume flow between the compartments K_i and K_j at time point t_k, page 125

$\dot{V}^k = [\dot{V}_{12}^k, \dot{V}_{23}^k, \dot{V}_{31}^k, \dot{V}_{34}^k, \dot{V}_{42}^k]^T$. vector of mean volume flows at time point t_k, page 125

$\mathcal{V}$ . . . . . coefficient function in the differential-algebraic equation (3.5), page 34

v_j . . . . . jth eigenvector of $\mathcal{A}$, page 159

v_{δ_t} . . . . . solution pair $\{u, p\}$ obtained by using the step size δ_t, page 54

$\mathcal{W}$ . . . . . partial derivative of w with respect to z, page 34

w . . . . . coefficient function in the differential-algebraic equation (3.5), page 34

w_e / w_x . . . . . "velocity" of the dispersed phase in direction of the internal / external coordinates, page 12

w_α . . . . . weights of the quadrature approximation, page 158

$X = [x_{ij}]$ . . . . . matrix in the definition of the Frobenius norm, page 178

$x = [x_1, x_2, x_3]^T$ . . . . . space coordinates, page 6

x^* . . . . . dimensionless space coordinate, page 174

$\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$ . . . . . coordinates of the newly created boundary node (after the refinement of the grid), page 60

$x_{3,\min}$ . . . . . height of the "bottom nodes" in the coarse grid, page 60

$x_{3,s}$ . . . . . osculation point of the two functions describing the torispherical head, page 57

$\mathcal{Y}$ . . . . . term in the output equation, page 29

y . . . . . continuous phase vector, page 13

Z_1, Z_2 . . . . . matrix functions, page 37

$\mathcal{Z} = [\mathcal{Z}_{12}, \mathcal{Z}_{23}, \mathcal{Z}_{31}, \mathcal{Z}_{34}, \mathcal{Z}_{42}]^T$. vector of optimized volume flows, page 125

$\mathcal{Z}_{ij}$ . . . . . optimized volume flow between the compartments K_i and K_j, page 125

$\mathcal{Z}^k = [\mathcal{Z}_{12}^k, \mathcal{Z}_{23}^k, \mathcal{Z}_{31}^k, \mathcal{Z}_{34}^k, \mathcal{Z}_{42}^k]^T$ vector of optimized volume flows at time point t_k, page 125

$\mathcal{Z}_{ij}^k$ . . . . . optimized volume flow between the compartments K_i and K_j at time point t_k, page 125

α . . . . . parameter in the fractional step θ-scheme, page 49

$\hat{\alpha}$ . . . . . relaxation parameter in the discrete projection method, page 52

α_{ij}^{L} . . . . . expansion coefficients on the interval I_j^L in the finite-element-type Galerkin method, page 107

β . . . . . parameter in the fractional step θ-scheme, page 49

Γ . . . . . boundary of the domain Ω, page 8

Γ_{stirrer} . . . . . stirrer part of the boundary, page 8

Γ_{surface} . . . . . surface part of the boundary, page 8

Γ_{wall} . . . . . wall part of the boundary, page 8

γ . . . . . distribution of daughter drops, page 15

δ . . . . . Dirac delta function, page 159

δ_{ij} . . . . . Kronecker delta, page 158

δ_f . . . . . correction term for the number density function, page 106

δ_t . . . . . time step size, page 48

$\delta_{t_i}, i = 1, 2, 3$ . . . . . time step size of substeps in the time stepping scheme, page 49

$\tilde{\delta}_t$ . . . . . necessary time step size calculated by the adaptive step size control, page 54

ε . . . . . energy dissipation rate, page 10

$\bar{\varepsilon}$ . . . . . mean energy dissipation rate throughout the tank, page 131

ε_0 . . . . . initial value for the energy dissipation rate, page 28

$\bar{\varepsilon}_i$ . . . . . mean energy dissipation rate in compartment K_i, page 118

ε_h . . . . . spatially discretized energy dissipation rate, page 33

ε_l . . . . . local energy dissipation rate, page 16

ϵ_s / ϵ_t . . . . . error of the space / time discretization scheme used in PARSIVAL, page 106

ζ_α . . . . . weighted abscissae, page 161

ϑ . . . . . collision frequency, page 18

θ . . . . . parameter in the one-step θ-scheme, page 48

θ', $\tilde{\theta}$ . . . . . parameters in the fractional step θ-scheme, page 49

θ_i . . . . . parameters in the time discretization scheme, page 49

κ_e . . . . . wave number of an eddy, page 24

λ_{coal} . . . . . coalescence efficiency, page 18

μ . . . . . dynamic viscosity, page 8

$\tilde{\mu}$ . . . . . thermodynamic material constant, page 173

$\hat{\mu}$ . . . . . strangeness-index, page 36

μ_c . . . . . dynamic viscosity of the continuous phase, page 19

μ_d . . . . . dynamic viscosity of the dispersed phase, page 21

μ_t . . . . . turbulent eddy viscosity, page 16

ν . . . . . kinematic viscosity, page 8

$\nu^* = \nu + \nu_t$ . . . . . sum of kinematic and turbulent viscosity, page 10

ν_t . . . . . turbulent viscosity, page 10

$\xi = [\xi_1^{(1)}, \ldots, \xi_{N_{\text{mom}}}^{(1)}, \xi_1^{(2)}, \ldots, \xi_{N_{\text{mom}}}^{(2)}]^T$ vector consisting of the source terms $\xi_\alpha^{(1)}$ and $\xi_\alpha^{(2)}$, page 162

$\xi_\alpha^{(1)}$, $\xi_\alpha^{(2)}$ . . . . . source terms of the transport equations in the direct quadrature method of moments, page 161

$\xi_\alpha^{(3)}$ . . . . . source term of the transport equation in the direct quadrature method of moments, page 164

π . . . . . polynomial, page 158

ρ . . . . . density, page 172

ρ_c . . . . . density of the continuous phase, page 19

ρ_d . . . . . density of the dispersed phase, page 23

ρ_f . . . . . density of the fluid, page 8

σ . . . . . interfacial tension, page 19

$\varsigma = [\bar{s}_0, \ldots, \bar{s}_{2N_{\text{mom}}-1}]^T$ . . . . . vector consisting of the source terms $\bar{s}_k$, page 162

$\tau = [\tau_{ij}]$ . . . . . stress tensor, page 173

τ_{strain} . . . . . strain tensor, page 173

τ_{visc} . . . . . viscous part of the stress tensor, page 173

Φ_N . . . . . transport term, page 182

ϕ_N . . . . . transport flow, page 11

$\phi_{N,e}$ / $\phi_{N,x}$ . . . . . transport flow in direction of the internal / external coordinates, page 11

$\phi_{N,e}^D$ / $\phi_{N,x}^D$ . . . . . diffusive part of the transport flow in direction of the internal / external coordinates, page 12

φ . . . . . angle, page 8

χ . . . . . volume fraction of the dispersed phase, page 5

Ψ . . . . . function in the derivation of the Navier-Stokes equations, page 171

ψ . . . . . path of a particle, page 171

Ω . . . . . spatial domain, page 6

Ω_0, Ω_t . . . . . subdomains of $\mathbb{R}^3$, page 171

Ω_1, Ω_2 . . . . . similar geometries, page 175

Ω_{baffles} . . . . . domain of the baffles, page 66

Ω_e / Ω_x . . . . . space of internal coordinates/ geometrical space, page 11

Ω_{stirrer} . . . . . domain of the stirrer, page 66

ω . . . . . angular velocity, page 8

$\hat{\omega}_l$	parameter in adaptive fixed point defect correction method, page 53
∇	gradient operator, page 8
$\nabla\cdot$	divergence operator, page 8
∇_e / ∇_x	gradient operator with respect to the internal / external coordinates, page 11
$\Delta = \nabla^2$	Laplace operator, page 8
$\otimes$	tensor product operator, page 177
$\langle\cdot\rangle$	filter operator, page 176

Bibliography

[AFIK04] C. Arakawa, O. Fleig, T. Ishihara, and Y. Kaneda. Direct numerical simulations of fundamental turbulent flows with the largest grid numbers in the world and its application of modeling for engineering turbulent flows. In *Annual Report of the Earth Simulator Center. April 2003 – March 2004*, chapter 4 (Epoch-Making Simulation), pages 195–198. The Earth Simulator Center, Japan Agency for Marine-Earth Science and Technology, Yokohama, July 2004. ISSN 1348-5830.

[AG01] M. Arnold and M. Günther. Preconditioned dynamic iteration for coupled differential-algebraic systems. *BIT. Numerical Mathematics*, 41:1–25, 2001.

[AKK99] V. Alopaeus, J. Koskinen, and K. I. Keskinen. Simulation of the population balances for liquid-liquid systems in a nonideal stirred tank. Part 1: Description and qualitative validation of the model. *Chemical Engineering Science*, 54:5887–5899, 1999.

[AKKM02] V. Alopaeus, J. Koskinen, K. I. Keskinen, and J. Majander. Simulation of the population balances for liquid-liquid systems in a nonideal stirred tank. Part 2: Parameter fitting and the use of the multiblock model for dense dispersions. *Chemical Engineering Science*, 57:1815–1825, 2002.

[Alb] ALBERTA – An adaptive hierarchical finite element toolbox. See `http://www.alberta-fem.de/index.html` (15.10.2006).

[AWF] AWFD – Adaptivity, Wavelets & Finite Differences. See `http://wissrech.iam.uni-bonn.de/research/projects/AWFD/` (18.08.2006).

[Bän98] E. Bänsch. Simulation of stationary, incompressible flows. *Acta Mathematica Universitatis Comenianae*, LXVII:101–114, 1998.

[BBD00] C. Bartels, M. Breuer, and F. Durst. Comparison between Direct Numerical Simulation and k-ε prediction of the flow in a vessel stirred by a Rushton turbine. In H. E. A. Van den Akker and J. J. Derksen, editors, *10th European Conference on Mixing*, pages 239–246, Kramers Laboratorium voor Fysische Technologie, TU Delft, July 2–5, 2000. Elsevier.

[BCGM94] A. Brucato, M. Ciofalo, F. Grisafi, and G. Micale. Complete numerical simulation of flow fields in baffled stirred vessels: the inner-outer approach. *Institution of Chemical Engineers (IChemE) Symposium Series*, 136:155–162, 1994.

[BCGM98] A. Brucato, M. Ciofalo, F. Grisafi, and G. Micale. Numerical prediction of flow fields in baffled stirred vessels: A comparison of alternative modelling approaches. *Chemical Engineering Science*, 53:3653–3684, 1998.

[BCP96] K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, volume 14 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1996.

[BG02] C. Becker and D. Göddeke. DeViSoRGrid 2. User's manual, 2002. See `http://www.featflow.de/feast_hp/devisormain.html` (10.02.2006).

[BH00] E. Bänsch and B. Höhn. Numerical treatment of the Navier-Stokes equations with slip boundary condition. *The Society for Industrial and Applied Mathematics (SIAM) Journal on Scientific Computing*, 21:2144–2162, 2000.

[BHB81] R. J. Batterham, J. S. Hall, and G. Barton. Pelletizing kinetics and simulation of full scale balling circuits. In *Proceedings of the 3rd International Symposium on Agglomeration*, pages A 136–A 151, Nürnberg, 1981.

[BHM+95] H. Blum, J. Harig, S. Müller, P. Schreiber, and S. Turek. FEAT2D. Finite Element Analysis Tools. User manual, release 1.3, Universität Heidelberg, 1995.

[Ble] Blender. Open source 3D graphics creation. See `http://www.blender.org` (10.02.2006).

[BLWC97] A. Bakker, R. D. LaRoche, M. H. Wang, and R. V. Calabrese. Sliding mesh simulation of laminar flow in stirred reactors. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 75:42–44, 1997.

[BLWC98] A. Bakker, R. D. LaRoche, M. H. Wang, and R. V. Calabrese. Sliding mesh simulation of laminar flow in stirred reactors. Published in "The Online CFM Book" at `http://www.bakker.org/cfmbook/cfmbook.html`, 1998. Updated: February 15, 2000.

[Bor91] F. A. Bornemann. An adaptive multilevel approach to parabolic equations. II. variable-order time discretization based on a multiplicative error correction. *Impact of Computing in Science and Engineering*, 3:93–122, 1991.

[Bra97] D. Braess. *Finite Elemente. Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer, Berlin, 1997.

[BS91] H. Bahmanyar and M. J. Slater. Studies of drop break-up in liquid-liquid systems in a rotating disc contactor. Part I: Conditions of no mass transfer. *Chemical Engineering & Technology*, 14:79–89, 1991.

[BSL02] R. B. Bird, W. E. Stuart, and E. N. Lightfoot. *Transport Phenomena*. John Wiley & Sons, Inc., New York, 2002.

[BTT97] F. Bertrand, P. A. Tanguy, and F. Thibault. Three-dimensional fictitious domain method for incompressible fluid flow problems. *International Journal for Numerical Methods in Fluids*, 25:136–719, 1997.

[BWS98] M. Böhm, K. Wechsler, and M. Schäfer. A parallel moving grid multigrid method for flow simulation in rotor-stator configurations. *International Journal for Numerical Methods in Engineering*, 42:175–189, 1998.

[CC88] J. Costes and J. P. Couderc. Study by laser Doppler anemometry of the turbulent flow induced by a Rushton turbine in a stirred tank: Influence of the size of the units – I. Mean flow and turbulence. *Chemical Engineering Science*, 43:2751–2764, 1988.

[CFD] CFD Online. Links – Software. See `http://www.cfd-online.com/Links/soft.html#cfd` (13.06.2006).

[CFX] ANSYS, Inc. ANSYS CFX. See `http://www.ansys.com/products/cfx.asp` (03.03.2006).

[CG95] S. L. Campbell and C. W. Gear. The index of general nonlinear DAEs. *Numerische Mathematik*, 72:173–196, 1995.

[Che91] A. K. Chesters. The modelling of coalescence processes in fluid-liquid dispersions: A review of current understanding. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 69:259–270, 1991.

[Che] Heterogeneous Process Simulation and Optimization (Cheops). See `http://www.lpt.rwth-aachen.de/Research/cheops.php` (03.08.2006).

[Cho68] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Mathematics of Computation*, 22:745–762, 1968.

[Chr04] J. Chronister. Blender basics. Classroom tutorial book, 2004. See `http://www.statikonline.com/Blender/Blender_Basics-Part1.pdf` (10.02.2006).

[CT77] C. A. Coulaloglou and L. L. Tavlarides. Description of interaction processes in agitated liquid-liquid dispersions. *Chemical Engineering Science*, 32:1298–1297, 1977.

[dea] deal.II. See `http://www.dealii.org/` (03.03.2006).

[DKJN93] K. N. Dyster, E. Koutsakos, Z. Jaworski, and A. W. Nienow. An LDA study of the radial discharge velocities generated by a Rushton turbine:

Newtonian fluids, $Re \geq 5$. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 71:11–23, 1993.

[DO02a] R. B. Diemer and J. H. Olson. A moment methodology for coagulation and breakage problems: Part 2 – moment models and distribution reconstruction. *Chemical Engineering Science*, 57:2211–2228, 2002.

[DO02b] R. B. Diemer and J. H. Olson. A moment methodology for coagulation and breakage problems: Part 3 – generalized daughter distribution functions. *Chemical Engineering Science*, 57:4187–4198, 2002.

[Dre83] D. A. Drew. Mathematical modeling of two-phase flow. *Annual Review of Fluid Mechanics*, 1983.

[DV99] J. Derksen and H. E. A. Van den Akker. Large Eddy Simulations on the flow driven by a Rushton turbine. *The American Institute of Chemical Engineers (AIChE) Journal*, 2:209–221, 1999.

[Ebe04] F. Ebert. A control-theoretic approach to simulator coupling. Diplomarbeit, TU Berlin, 2004.

[ES] The earth simulator. See `https://www.es.jamstec.go.jp/` (22.2.2006).

[Fas] FASTEST. See `http://www.fnb.maschinenbau.tu-darmstadt.de/de/software/fastest/` (13.06.2006).

[Flu] Fluent Inc. FLUENT Flow Modeling Software.
See `http://www.fluent.com/software/fluent/` (03.03.2006).

[FM04] T.-P. Fries and H.-G. Matthies. Classification and overview of meshfree methods. Informatikbericht 2003-03, Institut für Wissenschaftliches Rechnen, TU Braunschweig, 2004. See `http://opus.tu-bs.de/opus/volltexte/2003/418/` (16.02.2006).

[FMF04] R. Fan, D. L. Marchisio, and R. O. Fox. Application of the direct quadrature method of moments to polydisperse gas-solid fluidized beds. *Powder Technology*, 139:7–20, 2004.

[Fra05] T. Frank. Advances in Computational Fluid Dynamics (CFD) of 3-dimensional gas-liquid multiphase flow. In *NAFEMS Seminar "Simulation of Complex Flows (CFD)"*, Niedernhausen / Wiesbaden, April 25–26, 2005.

[Fri05] T.-P. Fries. *A Stabilized and Coupled Meshfree/Meshbased Method for Fluid-Structure Interaction Problems*. PhD thesis, TU Braunschweig, 2005.

[Gas] Gascoigne3D. High Performance Adaptive Finite Element Toolkit. See `http://www.gascoigne.uni-hd.de/index.html` (18.08.2006).

[GB04] R. Grosch and H. Briesen. *Getting started with Parsival. A tutorial introduction*. RWTH Aachen, 2004.

[GDN98] M. Griebel, T. Dornseifer, and T. Neunhoeffer. *Numerical Simulation in Fluid Dynamics: A Practical Introduction*. SIAM Monographs on Mathematical Modeling and Computation. The Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1998.

[Gea88] C. W. Gear. Differential-algebraic equation index transformations. *The Society for Industrial and Applied Mathematics (SIAM) Journal on Scientific and Statistic Computing*, 9:39–47, 1988.

[Gea90] C. W. Gear. Differential algebraic equations, indices, and integral algebraic equations. *The Society for Industrial and Applied Mathematics (SIAM) Journal on Numerical Analysis*, 27:1527–1534, 1990.

[Ger99] A. Gerstlauer. *Herleitung und Reduktion populationsdynamischer Modelle am Beispiel der Flüssig-Flüssig-Extraktion*, volume 3 (Verfahrenstechnik) of *Fortschritt-Berichte*. VDI Verlag, Düsseldorf, 1999.

[GIL+92] A. D. Gosman, R. I. Issa, C. Lekakou, M. K. Looney, and S. Politis. Multidimensional modelling of turbulent two-phase flow in stirred vessels. *The American Institute of Chemical Engineers (AIChE) Journal*, 38:1946–1956, 1992.

[GM86] E. Griepentrog and R. März. *Differential-Algebraic Equations and Their Numerical Treatment*, volume 88 of *Teubner-Texte zur Mathematik*. BSB B. G. Teubner Verlagsgesellschaft, Leipzig, 1986.

[Gor68] R. G. Gordon. Error bounds in equilibrium statistical mechanics. *Journal of Mathematical Physics*, 9:655–663, 1968.

[GPP94] R. Glowinski, T.-W. Pan, and J. Périaux. A fictitious domain method for Dirichlet problem and applications. *Computer Methods in Applied Mechanics and Engineering*, 111:283–303, 1994.

[GR05] D. Göddeke and T. Rohkämper. DeViSoR Grid3D. Manual, DeViSoR Group, Institute of Applied Mathematics, University of Dortmund, 25.05.2005. See `http://www.featflow.de/feast_hp/devisormain.html` (10.02.2006).

[GRV] D. Göddeke, T. Rohkämper, and T. Vollerthun. DeViSoR Grid3D. Design & visualisation software resource. See `http://www.featflow.de/feast_hp/devisormain.html` (10.02.2006).

[GSP+05] A. Gäbler, S. Schlauch, A. R. Paschedag, M. Kraume, and V. Mehrmann. Transient drop size distributions in stirred liquid liquid dispersions. In M. Sommerfeld, editor, *Proceedings of the 11th Workshop on Two-Phase Flow Predictions*, Merseburg, April 5–8, 2005.

[GWPK05] A. Gäbler, M. Wegener, A. R. Paschedag, and M. Kraume. Experiments, modelling and simulation of drop size distributions in stirred liquid/liquid systems. In *3rd International Berlin Workshop (IBW3) on Transport Phenomena with Moving Boundaries*, pages 20–35, Berlin, October, 6–7, 2005.

[GWPK06a] A. Gäbler, M. Wegener, A. R. Paschedag, and M. Kraume. The effect of pH on experimental and simulation results of transient drop size distributions in stirred liquid-liquid dispersions. *Chemical Engineering Science*, 61:3018–3024, 2006.

[GWPK06b] A. Gäbler, M. Wegener, A. R. Paschedag, and M. Kraume. The effect of pH on experimental and simulation results of transient drop size distributions in stirred liquid/liquid dispersions. In *Fluid Mixing VIII*, London, April 10–12, 2006.

[GWSK04] A. Gäbler, M. Wegener, S. Schlauch, and M. Kraume. Transient drop size distributions in stirred liquid liquid dispersions. In *15th Symposium of Chemical Engineering*, Cracow, 2004.

[GWSK05] A. Gäbler, M. Wegener, S. Schlauch, and M. Kraume. Transiente Tropfengrößenverteilungen in gerührten Flüssig/Flüssig-Dispersionen. *Chemie Ingenieur Technik*, 77:80–84, 2005.

[HiF] HiFlow. See `http://www.hiflow.de/` (03.03.2006).

[HK64] H. M. Hulburt and S. Katz. Some problems in particle technology. A statistical mechanical formulation. *Chemical Engineering Science*, 19:555–574, 1964.

[HKM+06] L. Henning, D. Kuzmin, V. Mehrmann, M. Schmidt, A. Sokolov, and S. Turek. Flow control on the basis of a Featflow-Matlab coupling. In *Notes on Numerical Fluid Mechanics and Multidisciplinary Design (NNFM)*. Springer, Berlin, Active Flow Control 2006, September 27–29, 2006. Accepted for publication.

[HLR89] E. Hairer, C. Lubich, and M. Roche. *The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods*. Springer Verlag, 1989.

[HN96] P. J. Hill and K. M. Ng. Statistics of multiple particle breakage. *The American Institute of Chemical Engineers (AIChE) Journal*, 42(6):1600–1611, 1996.

[Hoc90] R. M. Hockey. *Turbulent Newtonian and non-Newtonian flows in a stirred reactor*. PhD thesis, Department of Mechanical Engineering, Imperial College of Science, Technology and Medicine, London, 1990.

[HRM88] M. J. Hounslow, R. L. Ryall, and V. R. Marshall. Discretized population balance for nucleation, growth, and agglomeration. *The American Institute of Chemical Engineers (AIChE) Journal*, 34:1821–1832, 1988.

[HST94] J. Harig, P. Schreiber, and S. Turek. FEAT3D. Finite Element Analysis Tools in 3 Dimensions. Technical report, Universität Heidelberg, 1994.

[HW96] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer, Berlin, 1996.

[Ish75] M. Ishii. *Thermo-Fluid Dynamic Theory of Two-Phase Flow*, volume 22 of *Collection de la direction des études et recherches d'électricité de France*. Eyrolles, Paris, 1975.

[JD92] M. Jaeger and G. Dhatt. An extended k-ε finite element model. *International Journal for Numerical Methods in Fluids*, 14:1325–1345, 1992.

[JR99] M. Jenne and M. Reuss. A critical assessment on the use of k-ε turbulence models for simulation of the turbulent liquid flow induced by a Rushton-turbine in baffled stirred-tank reactors. *Chemical Engineering Science*, 54:3921–3941, 1999.

[KBG+05] V. Kulikov, H. Briesen, R. Grosch, A. Yang, L. von Wedel, and W. Marquardt. Modular dynamic simulation for integrated particulate processes by means of tool integration. *Chemical Engineering Science*, 60:2069–2083, 2005.

[KBM06] V. Kulikov, H. Briesen, and W. Marquardt. A framework for the simulation of mass crystallization considering the effect of fluid dynamics. *Chemical Engineering and Processing*, 45:886–899, 2006.

[KFT00] B. Koobus, C. Farhat, and H. Tran. Computation of unsteady viscous flows around moving bodies using the k-ε turbulence model on unstructured dynamic grids. *Computer Methods in Applied Mechanics and Engineering*, 190:1441–1466, 2000.

[KGS04] M. Kraume, A. Gäbler, and K. Schulze. Influence of physical properties on drop size distributions of stirred liquid-liquid dispersions. *Chemical Engineering & Technology*, 27:330–334, 2004.

[KHS06] J. Koch, W. Hackbusch, and K. Sundmacher. Simulation of the population balance for droplet breakage in a liquid-liquid stirred tank reactor using $\mathcal{H}$-matrix methods. In W. Marquardt and C. Pantelides, editors, *16th European Symposium on Computer Aided Process Engineering and 9th International Symposium on Process Systems Engineering*, pages 261–266. Elsevier B.V., 2006.

[KLT05] D. Kuzmin, R. Löhner, and S. Turek. *Flux-Corrected Transport*. Springer, 2005.

[KM94] P. Kunkel and V. Mehrmann. Canonical forms for linear differential-algebraic equations with variable coefficients. *Journal of Computational and Applied Mathematics*, 56:225–259, 1994.

[KM96a] P. Kunkel and V. Mehrmann. Generalized inverses of differential-algebraic operators. *The Society for Industrial and Applied Mathematics (SIAM) Journal on Matrix Analysis and Applications*, 17:426–442, 1996.

[KM96b] P. Kunkel and V. Mehrmann. Local and global invariants of linear differential-algebraic equations and their relations. *Electronic Transactions on Numerical Analysis*, 4:138–157, 1996.

[KM98] P. Kunkel and V. Mehrmann. Regular solutions of nonlinear differential-algebraic equations and their numerical determination. *Numerische Mathematik*, 79:581–600, 1998.

[KM01] P. Kunkel and V. Mehrmann. Analysis of over- and underdetermined nonlinear differential-algebraic systems with application to nonlinear control problems. *Mathematics of Control, Signals, and Systems*, 14:233–256, 2001.

[KM06] P. Kunkel and V. Mehrmann. *Differential-Algebraic Equations. Analysis and Numerical Solution*. EMS Publishing House, Zürich, 2006.

[KMS+06] D. Kuzmin, V. Mehrmann, S. Schlauch, A. Sokolov, and S. Turek. Population balances coupled with the CFD-code FEATFLOW. Preprint 12-2006, Institut für Mathematik, TU Berlin, 2006.

[Koc05] J. Koch. *Effiziente Behandlung von Integraloperatoren bei populationsdynamischen Modellen*. PhD thesis, Otto-von-Guericke-Universität Magdeburg, 2005.

[KR96a] S. Kumar and D. Ramkrishna. On the solution of population balance equations by discretization – I. A fixed pivot technique. *Chemical Engineering Science*, 51:1311–1332, 1996.

[KR96b] S. Kumar and D. Ramkrishna. On the solution of population balance equations by discretization – II. A moving pivot technique. *Chemical Engineering Science*, 51:1333–1342, 1996.

[Kra03] M. Kraume, editor. *Mischen und Rühren. Grundlagen und moderne Verfahren*. Wiley-VCH, Weinheim, 2003.

[KS00] R. Kübler and W. Schiehlen. Two methods of simulator coupling. *Mathematical and Computer Modelling of Dynamical Systems*, 6:93–113, 2000.

[KT04] D. Kuzmin and S. Turek. Numerical simulation of turbulent bubbly flows. In *3rd International Symposium on Two-Phase Flow Modelling and Experimentation*, Pisa, September 22–24, 2004.

[LAA02] M. Laakkonen, V. Alopaeus, and J. Aittamaa. The determination of parameters for bubble breakage and coalescence functions for gas-liquid systems in a mixed tank. In *The American Institute of Chemical Engineers (AIChE) Annual meeting 2002*, Indianapolis, 2002. See `http://www.tkk.fi/Units/ChemEng/news/2002/aiche2002/aiche-2002_177h.pdf#search=%22%22the%20determination%20of%20parameters%20for%20bubble%22%22` (24.08.2006).

[LEMM02] J. C. Lasheras, C. Eastwood, C. Martínez-Bazán, and J. L. Montañés. A review of statistical models for the break-up of an immiscible fluid immersed into a fully developed turbulent flow. *International Journal of Multiphase Flow*, 28:247–278, 2002.

[LGI+93] J. Y. Luo, A. D. Gosman, R. I. Issa, J. C. Middleton, and M. K. Fitzgerald. Full flow field computation of mixing in baffled stirred vessels. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 71:342–344, 1993.

[LIG94] J. Y. Luo, R. I. Issa, and A. D. Gosman. Prediction of impeller-induced flow in mixing vessels using multiple frames of references. *Institution of Chemical Engineers (IChemE) Symposium Series*, 136:549–556, 1994.

[Lo96] S. Lo. Application of the MUSIG model to bubbly flows. Technical Report AEAT-1096, AEA Technology, June 1996.

[Lo00a] S. Lo. Application of population balance to CFD modelling of gas-liquid reactors. In *Trends in Numerical and Physical Modelling for Industrial Multiphase Flows*, Corse, September 27–29, 2000. See `http://www.cmla.ens-cachan.fr/Utilisateurs/perfortmans/Cargese00/Cargese00_pro/lo.pdf` (28.10.2006).

[Lo00b] S. Lo. Some recent developments and applications of CFD to multiphase flows in stirred reactors. In *AMIF-ESF Workshop: Computing Methods for Two-Phase Flow*, Aussois, January 12–14, 2000.

[LS72] B. E. Launder and D. B. Spalding. *Mathematical Models of Turbulence*. Academic Press, London, 1972.

[LS74] B. E. Launder and D. B. Spalding. The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 3:269–289, 1974.

[LS96] H. Luo and H. F. Svendsen. Theoretical model for drop and bubble breakup in turbulent dispersions. *The American Institute of Chemical Engineers (AIChE) Journal*, 42:1225–1233, 1996.

[LSE00] G. L. Lane, M. P. Schwarz, and G. M. Evans. Comparison of CFD methods for modelling of stirred tanks. In H. E. A. Van den Akker and J. J. Derksen, editors, *10th European Conference on Mixing*, pages 273–280, Kramers Laboratorium voor Fysische Technologie, TU Delft, July 2–5, 2000. Elsevier.

[LSH95] J. D. Litster, D. J. Smit, and M. J. Hounslow. Adjustable discretized population balance for growth and aggregation. *The American Institute of Chemical Engineers (AIChE) Journal*, 41:591–603, 1995.

[Mar01] D. L. Marchisio. *Precipitation in Turbulent Fluids*. PhD thesis, Politecnico di Torino, Dipartimento di Scienza dei Materiali ed Ingegneria Chimica, 2001.

[Mär01] R. März. Nonlinear differential-algebraic equations with properly formulated leading term. Preprint 2001-03, Institut für Mathematik, Humboldt-Universität zu Berlin, 2001. See `http://www.mathematik.hu-berlin.de/publ/pre/2001/M-01-3.html` (08.09.2006).

[Mär02] R. März. The index of linear differential algebraic equations with properly stated leading terms. *Results in Mathematics. Resultate der Mathematik*, 42:308–338, 2002.

[Mar04] D. L. Marchisio. Workshop on population balance methods in computational fluid dynamics. Course material, Esbjerg, June 28 – July 2, 2004.

[Mat] The MathWorks Inc. MATLAB – The Language of Technical Computing. See `http://www.mathworks.de/products/matlab/` (13.08.2006).

[MBBF02] D. L. Marchisio, A. A. Barresi, G. Baldi, and R. O. Fox. Comparison between the classes method and the quadrature method of moments for describing aggregation, coalescence, and breakage problems in multiphase systems. In *Proceedings of the 8th International Conference on Multiphase Flow in Industrial Plants*, Alba, September 18–20, 2002. See also `http://www.emse.fr/pratsolis/list/list.html` or `http://www.emse.fr/pratsolis/list/animp2002barresi.pdf`, respectively (24.08.2006).

[McG97] R. McGraw. Description of aerosol dynamics by the quadrature method of moments. *Aerosol Science and Technology*, 27:255–265, 1997.

[MF] D. L. Marchisio and R. O. Fox. Direct quadrature method of moments: Derivation, analysis and applications. *Aerosol Science and Technology*. Submitted (2004).

[MGW+06] S. Maaß, A. Gäbler, M. Wegener, A. Zaccone, A. Paschedag, and M. Kraume. Drop breakage and daughter drop distributions in stirred liquid/liquid systems and their modelling within the population balance equation. In *12th European Conference on Mixing*, Bologna, June 27–30, 2006.

[MH97] A. Masud and T. J. R. Hughes. A space-time Galerkin/least-squares finite element formulation of the Navier-Stokes equations for moving domain problems. *Computer Methods in Applied Mechanics and Engineering*, 146:91–126, 1997.

[MJWR99] D. P. Mok, M. Jung, W. A. Wall, and E. Ramm. Iterative Substrukturalgorithmen für die Finite Elemente Analyse dynamischer Ein- und Mehrfeldsysteme. Chemnitzer FEM-Symposium, 1999.

[MLBY01] G. Montante, K. C. Lee, A. Brucato, and M. Yianneskis. Numerical simulations of the dependency of flow pattern on impeller clearance in stirred vessels. *Chemical Engineering Science*, 56:3751–3770, 2001.

[MMC94] J. Y. Murthy, S. R. Mathur, and D. Choudhury. CFD simulation of flows in stirred tank reactors using a sliding mesh technique. *Institution of Chemical Engineers (IChemE) Symposium Series*, 136:341–348, 1994.

[MML99a] C. Martínez-Bazán, J. L. Montañés, and J. C. Lasheras. On the breakup of an air bubble injected into a fully developed turbulent flow. Part 1. Breakup frequency. *Journal of Fluid Mechanics*, 401:157–182, 1999.

[MML99b] C. Martínez-Bazán, J. L. Montañés, and J. C. Lasheras. On the breakup of an air bubble injected into a fully developed turbulent flow. Part 2. Size PDF of the resulting daughter bubbles. *Journal of Fluid Mechanics*, 401:183–207, 1999.

[Moo] MooNMD – Mathematics and object oriented Numerics in Magdeburg. See `http://www-ian.math.uni-magdeburg.de/home/mitkova/MooNMD_html/` (18.08.2006).

[MPF+03] D. L. Marchisio, J. T. Pikturna, R. O. Fox, R. D. Vigil, and A. A. Barresi. Quadrature method of moments for population-balance equations. *The American Institute of Chemical Engineers (AIChE) Journal*, 49:1266–1276, 2003.

[MPK06] S. Maaß, A. R. Paschedag, and M. Kraume. Experimental and numerical investigations of coalescence and breakage phenomena in stirred liquid/liquid systems. In *16th International Conference "Chemical Engineering and Plant Design"*, Berlin, October 10–11, 2006.

[MPL86] J. C. Middleton, F. Pierce, and P. M. Lynch. Computation of flow fields and complex reaction yield in turbulent stirred reactors and comparison with experimental data. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 64:18–22, 1986.

[MSS+06] D. L. Marchisio, M. Soos, J. Sefcik, M. Morbidelli, A. A. Barresi, and G. Baldi. Effect of fluid dynamics on particle size distribution in particulate processes. *Chemical Engineering & Technology*, 29:191–199, 2006.

[MSSM06] D. L. Marchisio, M. Soos, J. Sefcik, and M. Morbidelli. Role of turbulent shear rate distribution in aggregation and breakage processes. *The American Institute of Chemical Engineers (AIChE) Journal*, 52:158–173, 2006.

[MVF03a] D. L. Marchisio, R. D. Vigil, and R. O. Fox. Implementation of the quadrature method of moments in CFD codes for aggregation-breakage problems. *Chemical Engineering Science*, 58:3337–3351, 2003.

[MVF03b] D. L. Marchisio, R. D. Vigil, and R. O. Fox. Quadrature method of moments for aggregation-breakage processes. *Journal of Colloid and Interface Science*, 258:322–334, 2003.

[MZV+06] D. L. Marchisio, A. Zucca, M. Vanni, A. A. Barresi, and G. Baldi. Formulation and validation of bivariate population balance models and their implementation in computational fluid dymanics codes. In *12th European Conference on Mixing*, Bologna, June 27–30, 2006.

[NW90] J. M. Nouri and J. H. Whitelaw. Effect of size and confinement on the flow characteristics in stirred reactors. In *Proceedings of the Fifth International Symposium on Application of Laser Techniques to Fluid Mechanics*, pages 23.2.1–23.2.8, Lisbon, July 9–12, 1990.

[OF03] S. Osher and R. Fedkiw. *Level set methods and dynamic implicit surfaces*, volume 153 of *Applied Mathematical Sciences*. Springer, New York, 2003.

[Ope] OpenFOAM: The Open Source CFD Toolbox. See `http://www.opencfd.co.uk/openfoam/` (18.08.2006).

[Ort] F. A. Ortega. GMV. General Mesh Viewer, version 3.4. User's manual, Los Alamos National Labratory.

[Pan88] C. C. Pantelides. The consistent initialization of differential-algebraic systems. *The Society for Industrial and Applied Mathematics (SIAM) Journal on Scientific and Statistic Computing*, 9:213–231, 1988.

[PAOK04] E. L. Paul, V. A. Antiemo-Obeng, and S. M. Kresta, editors. *Handbook of Industrial Mixing: Science and Practice*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2004.

[Par] PARSIVAL – Simulation of particle processes. See `http://www.cit-wulkow.de/pdf/Broschueres/Parsival/Brochure_Parsival_LQ.pdf` (25.08.2006).

[Pas04a] A. R. Paschedag. Modeling of mixing and precipitation using CFD and population balances. *Chemical Engineering & Technology*, 27:232–236, 2004.

[Pas04b] A. Paschedag. *Computational Fluid Dynamics in der Verfahrenstechnik*. Wiley-VCH-Verlag GmbH & Co, Weinheim, Bergstr., 2004.

[PB90] M. J. Prince and H. W. Blanch. Bubble coalescence and break-up in air-sparged bubble columns. *The American Institute of Chemical Engineers (AIChE) Journal*, 36:1485–1499, 1990.

[PB] W. Podgórska and J. Baldyga. Drop break-up and coalescence in the stirred tank. See `http://www.if.pw.edu.pl/~dynamika/spraw/pub/Baldyga/Drop_break-up.doc` (4.10.2006).

[Per] The Perl Directory at Perl.org. See `http://www.perl.org/` (13.08.2006).

[PM93] C. Y. Perng and J. Y. Murthy. A moving-deforming-mesh technique for simulation of flow in mixing tanks. *The American Institute of Chemical Engineers (AIChE) Symposium Series*, 89:37–41, 1993.

[PP87] K. A. Pericleous and M. K. Patel. The modelling of tangential and axial agitators in chemical reactors. *PhysicoChemical Hydrodynamics*, 8:105–123, 1987.

[Ram00] D. Ramkrishna. *Population Balances. Theory and Applications to Particulate Systems in Engineering*. Academic Press, San Diego, 2000.

[RD96] V. V. Ranade and S. M. S. Dommeti. Computational snapshot of flow generated by axial impellers in baffled stirred vessels. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 74:476–484, 1996.

[Rey95] O. Reynolds. On the dynamical theory of incompressible viscous fluids and the determination of the criterion. *Philosophical Transactions of the Royal Society of London. A (1887–1895)*, 186:123–164, 1895.

[Rhe84] W. C. Rheinboldt. Differential-algebraic systems as differential equations on manifolds. *Mathematics of Computation*, 43:473–482, 1984.

[Rit02] J. Ritter. *Dispergierung und Phasentrennung in gerührten Flüssig/flüssig-Systemen*. PhD thesis, TU Berlin, 2002.

[RJ90a] V. V. Ranade and J. B. Joshi. Flow generated by a disc turbine: Part II Mathematical modelling and comparison with experimental data. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 68:34–50, 1990.

[RJ90b] V. V. Ranade and J. B. Joshi. Flow generated by a disc turbine: Part I Experimental. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 68:19–33, 1990.

[RS05] T. Reis and T. Stykel. Stability analysis and model order reduction for coupled systems. Preprint 241, DFG Research Center Matheon, TU Berlin, 2005. Submitted for publication.

[RT92] R. Rannacher and S. Turek. Simple nonconforming quadrilateral Stokes element. *Numerical Methods for Partial Differential Equations*, 8:97–111, 1992.

[Rut57] H. Rutishauser. *Der Quotienten-Differenzen-Algorithmus*, volume 7 of *Mitteilungen aus dem Institut für angewandte Mathematik an der Eidgenössischen Technischen Hochschule in Zürich*. Birkhäuser Verlag, Basel, 1957.

[RV94] V. V. Ranade and H. E. A. Van den Akker. A computational snapshot of gas-liquid flow in baffled stirred reactors. *Chemical Engineering Science*, 49:5175–5192, 1994.

[RXY+02] T. Rung, L. Xue, J. Yan, M. Schatz, and F. Thiele. Numerische Methoden der Thermo- und Fluiddynamik. Lecture notes, 2002.

[SB87] J. L. Steger and J. A. Benek. On the use of composite grid schemes in computational aerodynamics. In *Proceedings of the first world congress on computational mechanics (Austin, Texas, 1986)*, volume 64, pages 301–320, 1987.

[SC95] C. M. Stoots and R. V. Calabrese. Mean velocity field relative to a Rushton turbine blade. *The American Institute of Chemical Engineers (AIChE) Journal*, 41:1–11, 1995.

[SD03] M. Sommerfeld and S. Decker. State of the art and future trends in CFD simulation of stirred vessels. In VDI-GVC, editor, *11th European Conference on Mixing*, pages 1–20, Bamberg, October 14–17, 2003. VDI-Gesellschaft Verfahrenstechnik und Chemieingenieurwesen GVC.

[Set99] J. A. Sethian. *Level set methods and fast marching methods*, volume 3 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, Cambridge, 1999.

[SHD97] M. Schäfer, M. Höfken, and F. Durst. Detailed LDV measurements for visualization of the flow field within a stirred-tank reactor equipped with a Rushton turbine. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 75:729–736, 1997.

[Sla] Slawinski. Product information. Torispherical heads acc. to DIN 28011. See `http://www.slawinski.de/english/index.htm` (07.02.2006).

[SS98] R. Sieber and M. Schäfer. Performance of a block-structured clicking-mesh multigrid scheme for flow computations in time-varying geometries. In *Computational Fluid Dynamics '98. Proceedings of the Fourth European Computational Fluid Dynamics Conference, September 7–11, 1998, Athens, Greece*, pages 1291–1295, Chichester, 1998. Wiley.

[SS01] R. Sieber and M. Schäfer. Dynamic meshes for fluid-structure interaction. In *Large-Scale Scientific Computing*, volume 2179 of *Lecture Notes in Computer Science*, pages 387–397, Berlin, 2001. Springer.

[Sta] CD-adapco. STAR-CD. See `http://www.cd-adapco.com/products/STAR-CD/` (03.03.2006).

[Ste91] J. L. Steger. The Chimera method of flow simulation. In *Workshop on Applied CFD*, University of Tennessee Space Institute, August 1991.

[Sto99] J. Stoer. *Numerische Mathematik 1.* Springer, Berlin, 1999.

[Str04] A. Streller. Finite element discretization of the Navier-Stokes equation for the backward facing step. Diplomarbeit, TU Berlin, 2004.

[SYvWM04] G. Schopfer, A. Yang, L. von Wedel, and W. Marquardt. CHEOPS: A tool integration platform for chemical process modelling and simulation. *International Journal on Software Tools for Technology Transfer*, 6:186–202, 2004.

[TB98] S. Turek and C. Becker. FEATFLOW. Finite element software for the incompressible Navier-Stokes equations. User manual, release 1.1, Universität Heidelberg, 1998.

[TDR02] S. Turek, W. Decheng, and L. Rivkind. The fictitious boundary method for the implicit treatment of Dirichlet boundary conditions with applications to incompressible flow simulation. In E. Bänsch, editor, *Challenges in Scientific Computing CISC 2002*, LNCSE, pages 37–68, Berlin, 2002. Springer.

[TS96] S. Turek and M. Schäfer. Benchmark computations of laminar flow around cylinder. In E. H. Hirschel, editor, *Flow Simulation with High-Performance Computers II*, volume 52 of *Notes on Numerical Fluid Mechanics*, pages 547–566. Vieweg, 1996.

[TT94] C. Tsouris and L. L. Tavlarides. Breakage and coalescence models for drops in turbulent dispersions. *The American Institute of Chemical Engineers (AIChE) Journal*, 40:395–406, 1994.

[Tur94] S. Turek. Tools for simulating non-stationary incompressible flow via discretely divergence-free finite element models. *International Journal for Numerical Methods in Fluids*, 18:71–105, 1994.

[Tur96] S. Turek. A comparative study of time stepping techniques for the incompressible Navier-Stokes equations: From fully implicit nonlinear schemes to semi-implicit projection methods. *International Journal for Numerical Methods in Fluids*, 22:987–1011, 1996.

[Tur97] S. Turek. On discrete projection methods for the incompressible Navier-Stokes equations: An algorithmical approach. *Computer Methods in Applied Mechanics and Engineering*, 143:271–288, 1997.

[Tur99] S. Turek. *Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach.* Lecture Notes in Computational Science and Engineering. Springer, Berlin, 1999.

[Van86] J. Van Kan. A second-order accurate pressure-correction scheme for viscous incompressible flow. *The Society for Industrial and Applied Mathematics (SIAM) Journal on Scientific and Statistic Computing*, 7:870–891, 1986.

[Van99] M. Vanni. Discretized procedure for the breakage equation. *The American Institute of Chemical Engineers (AIChE) Journal*, 45:916–919, 1999.

[Van00] M. Vanni. Approximate population balance equations for aggregation-breakage processes. *Journal of Colloid and Interface Science*, 221:143–160, 2000.

[VBS76] K. Van't Riet, W. Bruijn, and J. M. Smith. Real and pseudo-turbulence in the discharge stream from a Rushton turbine. *Chemical Engineering Science*, 31:407–412, 1976.

[VDV02] B. C. H. Venneker, J. J. Derksen, and H. E. A. Van den Akker. Population balance modeling of aerated stirred vessels based on CFD. *The American Institute of Chemical Engineers (AIChE) Journal*, 48:673–685, 2002.

[VV78] K. Van der Molen and H. R. E. Van Maanen. Laser-Doppler measurements of the turbulent flow in stirred vessels to establish scaling rules. *Chemical Engineering Science*, 33:1161–1168, 1978.

[Weg04] M. Wegener. Experimentelle Untersuchungen und Modellierung von transienten Tropfengrößenverteilungen in gerührten Flüssig-flüssig-Systemen. Diplomarbeit, TU Berlin, 2004.

[Wei96] J. Weickert. Navier-Stokes equations as a differential-algebraic system. Preprint, TU Chemnitz-Zwickau, 1996.

[Wei97] J. Weickert. *Applications of the Theory of Differential-Algebraic Equations to Partial Differential Equations of Fluid Dynamics*. PhD thesis, TU Chemnitz, 1997.

[Wes01] P. Wesseling. *Principles of Computational Fluid Dynamics*, volume 29 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2001.

[WGN99] M. Wulkow, A. Gerstlauer, and U. Nieken. PARSIVAL – A tool for the simulation of crystallization processes. *Chemische Technik*, 51:249–252, 1999.

[WGN01] M. Wulkow, A. Gerstlauer, and U. Nieken. Modeling and simulation of crystallization processes using parsival. *Chemical Engineering Science*, 56:2575–2588, 2001.

[Wil88] D. C. Wilcox. Reassessment of the scale-determining equation for advanced turbulence models. *The American Institute of Aeronautics and Astronautics (AIAA) Journal*, 26:1299–1310, 1988.

[Wör03] M. Wörner. A compact introduction to the numerical modeling of multiphase flows. Wissenschaftlicher Bericht FZKA 6932, Institut für Reaktorsicherheit, Programm Nukleare Sicherheitsforschung, Forschungszentrum Karlsruhe, November 2003.

[WP89] H. Wu and G. K. Patterson. Laser-Doppler measurements of the turbulent-flow parameters in a stirred mixer. *Chemical Engineering Science*, 44:2207–2221, 1989.

[WR05] G. J. Wells and W. H. Ray. Methodology for modeling detailed imperfect mixing effects in complex reactors. *The American Institute of Chemical Engineers (AIChE) Journal*, 51:1508–1520, 2005.

[WTR03] D. Wan, S. Turek, and L. Rivkind. An efficient multigrid FEM solution technique for incompressible flow with moving rigid bodies. In M. Feistauer, V. Dolejsi, P. Knobloch, and K. Najzar, editors, *Numerical Mathematics and Advanced Applications*, pages 844–853, Berlin, 2003. Springer.

[Wul] M. Wulkow. *PARSIVAL. Simulation Package for Particle Balances*. CiT GmbH.

[WWJ03] T. Wang, J. Wang, and Y. Jin. A novel theoretical breakup kernel function for bubbles/droplets in a turbulent flow. *Chemical Engineering Science*, 58:4629–4637, 2003.

[WWJ05] T. Wang, J. Wang, and Y. Jin. Population balance model for gas-liquid flows: Influence of bubble coalescence and breakup models. *Industrial & Engineering Chemistry Research*, 44:7540–7549, 2005.

[YPL03] S. L. Yeoh, G. Papadakis, and K. C. Lee. Large eddy simulation of turbulent flow in Rushton impeller stirred reactor with a sliding-deforming mesh methodology. In VDI-GVC, editor, *11th European Conference on Mixing*, pages 39–46, Bamberg, October 14–17, 2003. VDI-Gesellschaft Verfahrenstechnik und Chemieingenieurwesen GVC.

[YPW87] M. Yianneskis, Z. Popiolek, and J. H. Whitelaw. An experimental study of the steady and unsteady flow characteristics of stirred reactors. *Journal of Fluid Mechanics*, 175:537–555, 1987.

[YUNK02] S. Yuu, T. Umekage, Y. Nakano, and S. Kawakami. Numerical simulation of a medium Reynolds number baffled stirred tank (Re = 4500) and experimental verification. *The Japan Society of Mechanical Engineers (JSME) International Journal Series B (Fluids and Thermal Engineering)*, 45:752–758, 2002.

[YW93] M. Yianneskis and J. H. Whitelaw. On the structure of the trailing vortices around Rushton turbine blades. *Transactions of the Institution of Chemical Engineers (IChemE), Part A, Chemical Engineering Research and Design*, 71:543–550, 1993.

[Zei96] E. Zeidler, editor. *Teubner-Taschenbuch der Mathemtik*. B. G. Teubner, Stuttgart, 1996. Begründet von I. N. Bronstein und K. A. Semendjajew. Weitergeführt von G. Grosche, V. Ziegler und D. Ziegler.

[ZMBF04] A. Zucca, D. L. Marchisio, A. A. Barresi, and R. O. Fox. Implementation of the population balance equation in CFD codes for modelling soot formation in turbulent flames. In *Proceedings 2nd International Conference on Population Balance Modelling*, Valencia, May 5–7, 2004.

[ZMBF06] A. Zucca, D. L. Marchisio, A. A. Barresi, and R. O. Fox. Implementation of the population balance equation in CFD codes for modelling soot formation in turbulent flames. *Chemical Engineering Science*, 61:87–95, 2006.