No. 616 November 2019

Numerical Simulation and Benchmarking of Drops and Bubbles

S. Turek, O. Mierka

ISSN: 2190-1767

Chapter 1

Numerical Simulation and Benchmarking of Drops and Bubbles

Stefan Turek^{*,1} **and Otto Mierka***
*, *TU Dortmund, Institut für Angewandte Mathematik (LS3), Vogelpothsweg 87, 44227 Dortmund, Germany*

1.1 INTRODUCTION

The numerical simulation of immiscible multiphase flow problems, particularly including drops and bubbles, is very important in many applications, and performing accurate, robust and efficient numerical computations has been the object of numerous research and simulation projects for many years. One of the main challenges for the underlying numerical methods - besides the fact that the computational simulation of the incompressible Navier-Stokes equations is challenging by itself - is that the position of the moving interface between two fluids is unknown and must be determined as part of the boundary value problem to be solved. If we assume, in the following, a domain Ω with two immiscible fluids, then the time dependent subdomains $\Omega_1(t)$ and $\Omega_2(t)$ are bounded by an external boundary Σ and a dynamic interior boundary or interface $\Gamma(t)$ which might consist of several components (see Figure 1.1).

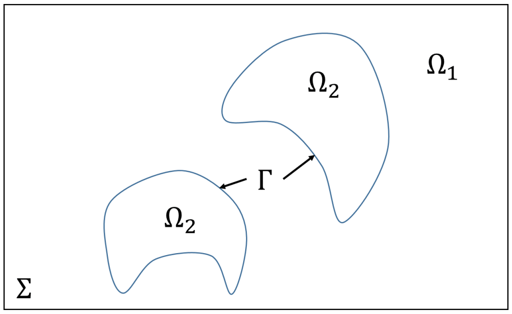

FIGURE 1.1 Illustrative sketch of the complete domain $\Omega = \Omega_1 \cup \Gamma \cup \Omega_2$.

Then, the usual model for (laminar) multiphase flow is described by the

---

[1] Corresponding author

incompressible Navier-Stokes equations

$$\rho(\mathbf{x},t)\left[\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right] - \nabla\cdot(\mu(\mathbf{x},t)[\nabla\mathbf{u} + (\nabla\mathbf{u})^T]) + \nabla p = \rho(\mathbf{x},t)\mathbf{g} + \mathbf{f}_\Gamma(\sigma) \quad (1.1)$$

$$\nabla\cdot\mathbf{u} = 0 \qquad \text{in} \qquad \Omega = \Omega_1 \cup \Gamma \cup \Omega_2 \quad (1.2)$$

which contain an additional force term $\mathbf{f}_\Gamma(\sigma)$ due to the surface tension σ at the free interface Γ. Here, the density ρ as well as the viscosity μ are variable and discontinuous, that means

$$\rho(\mathbf{x},t) = \begin{cases} \rho_1, & \forall\, \mathbf{x} \in \Omega_1(t) \\ \rho_2, & \forall\, \mathbf{x} \in \Omega_2(t) \end{cases}, \qquad \mu(\mathbf{x},t) = \begin{cases} \mu_1, & \forall\, \mathbf{x} \in \Omega_1(t) \\ \mu_2, & \forall\, \mathbf{x} \in \Omega_2(t) \end{cases} \quad (1.3)$$

which significantly influences the velocity $\mathbf{u}$ as well as the pressure p.

This contribution provides a numerical description and related computational studies of both interface capturing and interface tracking approaches, representating two prominent methodologies for handling multiphase flow problems, particularly in the framework of the Finite Element Method (FEM). Our main emphasis is laid on the description of special benchmarking tools and corresponding results for both discussed approaches which can be utilized for validation and evaluation of today's and future numerical methods as well as their realization in CFD software. Since we aim to provide such tools for (almost) 'real life' applications, we concentrate on fully 3D configurations while corresponding 2D results can be found in the literature, for instance in [29]. For this purpose, the open-source CFD package FeatFlow (`www.featflow.de`) was utilized and extended with the described multiphase flow modules such that the existing numerical ingredients of the FeatFlow approach [12, 32, 55], namely flexible, higher order FEM discretization schemes in space and time with flux correction and edge-oriented stabilization techniques, unstructured meshes with adaptive grid deformation, efficient Newton-Multigrid solvers and parallelization based on domain decomposition could be directly exploited. It should be also mentioned that the utilized techniques in this contribution are representative for many other approaches based on Finite Volume, Finite Difference or Radial Basis Function discretizations of the Navier-Stokes equations as well as of equivalent flow models based on Lattice Boltzmann methods (LBM), Smoothed Particle Hydrodynamcis (SPH) methods and many others.

The outline of the contribution is as follows: After a short description in section 1.2 of the state-of-the-art regarding interface tracking and capturing methods, we describe in section 1.3 the chosen solution techniques which are based on Discrete Projection methods [53, 54] for the Navier-Stokes equations, combined with corresponding numerical tools for the interface capturing, resp., tracking approaches. To be precise, discretization aspects regarding the incompressible Navier-Stokes equations using the Crank-Nicolson method and the

Q_2/P_1 finite element pair are discussed in section 1.3, too, whereas the details of the employed FEM approaches for the Level Set equation as well as adapting the mesh towards the dynamic interface can be found in sections 1.4,1.5 and 1.6.

Moreover, section 1.7 presents several numerical test cases which first of all shall help to evaluate the quality of the underlying single-phase flow solver, here providing quantitative results for a 3D ‘flow around a cylinder’ configuration. Subsequently, we present recently developed benchmark settings for prototypical 3D multiphase flows which can be used for ‘simple’ validation and evaluation of multiphase CFD codes without the necessity of complex postprocessing operations: Besides a 3D ‘rising bubble’ benchmark [1], we also present numerical reference values for a ‘Taylor bubble’ setting [34]. Additionally, we provide simulation results of a reactive Taylor bubble flow in a framework of which a reaction parameter estimation has been performed in order to match the corresponding experimentally measured results. Finally, the results are summarized in section 1.8.

1.2 MATHEMATICAL MODEL

The free interface Γ is permanently being deformed and moved so that its position has to be treated as an unknown and has to be determined in every time step. Depending on the technique for the representation of the interface, one can distinguish between *front tracking* and *front capturing* approaches which can be realized on fixed as well as dynamically moving meshes. For an overview of existing numerical approaches and their classification we recommend references [47] and [51]. In the following, we provide a compact summary of the most important characteristics and properties of both classes of methods.

The ‘natural’ front tracking approach [2, 21, 37, 60] is based on an explicit tracing of the dynamic interface between the two phases. Here, in the case of Lagrangian finite element methods [24], the underlying mesh has to be continuously adapted to the free interface so that the grid points move with the interface. More flexibility is promised by the *Arbitrary Lagrangian Eulerian* (ALE) formulation [6, 7, 10, 18, 19] and [44] which is based on local grid adaptation and which provides excellent results in the case of moderate deformations (for instance for small waves at the free surface). Moreover, there are many more techniques of fictitious domain and Chimera type which allow the highly accurate tracking of the dynamic interfaces via overlapping surface meshes [25].

According to the recently published works of Bäumler [8] and Basting [4] the use of front tracking methods in combination with additional numerical components establishes an extremely powerful simulation framework being suitable for benchmarking purposes of multiphase flows experiencing no topological changes. These additional numerical components are:

- isoparametric finite elements
- (semi)implicit treatment of surface tension
- PDE-based mesh deformation

which are all together inevitable to reach the resulting synergic effect guaranteeing high convergence properties of the resulting numerical scheme. The realization of above mentioned work was performed in a 2D environment (Basting [4] pure 2D; Bäumler [8] axisymmetric 3D, i.e. pseudo 3D) where its potential was clearly demonstrated on classical 2D benchmark problems such as 'static bubble' and 'rising bubble' [29] with excellent convergence properties. This is the reason why exactly these two approaches have been used as guideline to perform the corresponding extensions in a fully 3D flow solver environment in the open-source software FeatFlow. It is also worth to mention that the original 2D work showed additional potential in case of

- topological remeshing where a novel model for the relocation of the fluidic interface on the level of the otherwise fixed mesh topology was develeped [4] making it possible to use the front tracking method in the case of strong deformations of the interface, too, for the second test case of the 2D 'rising bubble' benchmark (see [29]).
- additional transport equations for the reactive transport of species as well as their backcoupling via concentration dependent surface tension (known as Marangoni effects) which has been successfully modeled in [8].

Let us summarize the properties of the up-to-date available front tracking approach first from the point of view of its advantages:

- due to its higher order geometrical approximation of the interfacial surface (via isoparametric FEM), excellent spatial convergence properties are naturally supported with excellent mass conservation which is already achieved on relatively coarse meshes. For the very same reason, any artificial mass correction mechanism is not necessary.
- thanks to the aligned mesh strategy, no smoothening of material properties is necessary to perform what makes this approach (more or less) parameter-free.
- due to the used (semi-)implicit treatment of the surface tension via the Laplace Beltrami operator, no global recovery of the normal vectors or the curvature is necessary and no capillary time-step restriction is needed to be applied.

Unfortunately, the front tracking approach is related to disadvantages, too, from which the following are important to be mentioned:

- due to the time dependent mesh motion and the employed ALE techniques, the temporal discretization may reduce to first order accuracy which can only be compensated by the use of small timesteps or special discretization schemes.
- neither topological nor drastic deformations of the interfacial surface are allowed. Moreover, special care has to be taken for the generation of the computational mesh, so to obtain meshes suitable for the overall course of unsteady simulations.

These aspects make this method suitable for benchmarking purposes of relatively simple multiphase configurations, but unfortunately not robust enough for its generic use in the field of general laminar multiphase flows.

In contrast to such (often highly accurate) Lagrangian methods, Eulerian front capturing methods seem to be much more robust and flexible. They are applicable even to free interface problems with significant topology changes (breakup of bubbles, fragmentation, coalescence, etc.). Based on the early Marker-and-Cell method of Harlow and Welch [62], the implicit reconstruction of the interface is based on an indicator function $\phi(\mathbf{x}, t)$ which contains the information about the corresponding subdomain for the point $\mathbf{x}$ at time t. The distribution in the complete domain Ω can then be calculated via the scalar transport equation

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0 \tag{1.4}$$

so that the exact position of the free interface $\Gamma(\phi)$ at any time can be reconstructed from ϕ with the help of postprocessing techniques. One of the most well known methods is the Volume-of-Fluid (VOF) method [38] in which case the indicator function ϕ can be interpreted as volume fraction which should have the discrete values 0 or 1 depending on the location of $\mathbf{x}$:

$$\phi(\mathbf{x}, t) = \begin{cases} 1, & \forall\, \mathbf{x} \in \Omega_1(t) \\ 0, & \forall\, \mathbf{x} \in \Omega_2(t) \end{cases} \tag{1.5}$$

The numerical drawback of this approach is that artificial diffusion smears out the (originally) discontinuous indicator function which arises from the solution of the discretized advection problems resulting in a boundary layer with $0 < \phi < 1$. Therefore, numerical schemes and locally adapted meshes have to be designed to address this boundary layer as thin as possible so that the corresponding error for reconstructing the free interface is reduced. Moreover, due to the steep gradients and the discontinuity of the indicator function, standard Galerkin schemes lead to unphysical oscillations which significantly deteriorate the accuracy or even lead to unphysical over- and undershoots. As conclusion, the development of corresponding higher order monotone discretization schemes in combination with unstructured, locally refined meshes still belongs to the numerical challenges one has to solve.

As a successful alternative, the Level Set approach [39, 40, 49] has been established which represents the interface as zero isoline of a continuous indicator function ϕ which should be close to the distance with respect to the free interface

$$\phi(\mathbf{x}, t) = \begin{cases} \text{dist}(\mathbf{x}, \Gamma), & \forall\, \mathbf{x} \in \Omega_1(t) \\ -\text{dist}(\mathbf{x}, \Gamma), & \forall\, \mathbf{x} \in \Omega_2(t) \end{cases} \tag{1.6}$$

so that $\Gamma(t) = \{\mathbf{x} \in \Omega \,|\, \phi(\mathbf{x}, t) = 0\}$ holds. In contrast to the VOF approach, ϕ as a distance function is smooth and allows the calculation of a globally defined normal vector $\mathbf{n}$ towards the interface Γ and of the corresponding curvature via

$$\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|}, \qquad \kappa = -\nabla \cdot \mathbf{n} = -\nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right). \tag{1.7}$$

Here, special FEM techniques for gradient recovery can be used which allow highly accurate approximations of normal vectors and curvature [51] which are necessary for the direct evaluation of the surface tension force $\mathbf{f}_\Gamma = \kappa\sigma\delta(\phi)\mathbf{n}$, with $\delta(\phi)$ denoting the corresponding Dirac Delta function. Hence, the development and implementation of a typical Level Set approach consists of performing the following sequence of tasks:

- Discretization of the Level Set transport problem in (1.4).
- Reinitialisation, resp., redistancing of the Level Set function.
- Additional correction so that mass and volume is preserved (if necessary).
- Calculation of normal vector fields (and curvature if needed) based on ϕ.
- Evaluation of the discontinuous fluid parameters $\rho(\phi)$, $\mu(\phi)$, and of $\mathbf{f}_\Gamma$, with or without reconstruction of Γ.

The above sequence of tasks involves very many different possibilities and choices which inevitably lead to numerous differing solution approaches. This is evident from the rich collection of publications on Level Set methods which also demonstrates the high potential of these methods for a wide range of applications (see for instance the books by Osher [39] and Sethian [49]). However, the resulting quality of the solutions mainly depends on the underlying numerical and computational approaches, and one has to acknowledge the fact that many existing Level Set codes are still based on uniform Cartesian meshes which is easy to implement. The drawback is that the computational cost typically is quite high since uniform mesh refinement has to be performed to resolve the necessary scales, particularly near the fluidic interfaces, but also due to complicated geometries with small-scale structures. Unstructured meshes are particularly well suited for such approaches which leads us to finite volume and finite element discretization methods which are the most prominent candidates for unstructured simulation approaches. Examples for corresponding approaches in the framework of VOF and Level Set methods can be found in [3, 10, 11, 17, 28, 35, 36, 45]. In many approaches, for example in the *Interface Proximity Adaption Method* of Barth and Sethian [3], the mesh is locally refined near the interface which is quite easy to find if ϕ is a distance function [35].

Although finite element methods together with locally refined grids seem to possess a very advantageous behaviour for the simulation of multiphase flow problems with free interfaces, many existing Level Set codes are still based on finite differences. It is only during the last years that FEM codes have been successfully applied for these special CFD problems ([42, 46, 50]; see also [16, 23, 36, 43, 51, 52, 56]). However, there is still a huge potential for improvement if '*optimal*' modern discretization and solution techniques shall be adapted to the special characteristics of FEM-Level Set methods. In constructing a modern Level Set solver it is important to focus on unstructured meshes with local grid refinement strategies for highly nonstationary multiphase flow simulations, and to make detailed studies for higher numerical stability. Additionally, stable and accurate discretization of the convective terms (for instance, VOF and Phase-

Field methods show very steep gradients near the interface, similarly as Level Set approaches without redistancing), robust treatment of large density differences, and the handling of large surface tension σ also require special attention.

Summarizing the properties of FEM-Level Set techniques for multiphase flow problems, we may conclude the following (potentially) advantageous behaviour in comparison to interface tracking methods as well as VOF and Phase-Field approaches which motivates our recent and future work for the combination of finite elements and Level Set methods:

- If the Level Set function satisfies the distance property, it is smooth so that even on highly uniform meshes qualitatively good results can be obtained. Local refinement around the interface will help to improve the accuracy, but in contrast to VOF and Phase-Field methods, which may lead to smeared interfaces due to numerical diffusion or to unphysical oscillations due to steep gradients, adaptive meshes are not necessary.
- Accurate FEM discretizations of higher order can be adapted to the special characteristics of Level Set functions, that means higher smoothness because of the distance function properties.
- Accurate representations of the interface are provided, without explicit description, but even for complex geometrical changes, which is important for handling the surface tension term.
- Auxiliary quantities like normal vectors and curvature are provided, even globally, which is particularly advantageous for the Continuous Surface Force (CSF) [9] approach.

On the other hand, there are still several problems with Level Set approaches (and some of them are also valid for VOF and Phase-Field methods) which are numerically challenging and which are in the focus of recent research activities:

- The standard Level Set formulation is not conservative which may lead to mass loss.
- Since reinitialisation is necessary to preserve the distance property, often highly expensive computational operations might be necessary, for instance via solving globally the Eikonal equation [13, 26], or redistancing is based on ‘cheaper’ methods which however change the position and shape of the interface, leading again to mass loss.
- Due to the standard explicit treatment of surface tension, the time step size is restricted by the *capillary time step restriction* [27], that means the necessary time steps depend by purely numerical reasons on the size of surface tension and on the local mesh size.

In the following sections, we first of all describe the overall solution techniques for the underlying incompressible Navier-Stokes equations which are based on the ‘Discrete Projection method’ which is followed by a discussion of the FEM discretization details, on the one hand regarding numerical techniques for treating the Level Set equation, and on the other hand discussing interface tracking methods in detail.

1.3 DISCRETE PROJECTION METHODS FOR THE INCOMPRESSIBLE NAVIER-STOKES EQUATIONS

In this section, we briefly review the 'Discrete Projection method' as a special variant of Multilevel Pressure Schur Complement (MPSC) approaches for the solution of incompressible flow problems, and we combine it with FEM discretization techniques. We will explain some characteristics of high-resolution FEM schemes as applied to incompressible flow problems and discuss the computational details regarding the efficient numerical solution of the resulting nonlinear and linear algebraic systems. Furthermore, we will discuss the coupling mechanisms between the 'basic' flow model (standard Navier-Stokes equations for velocity and pressure) and the scalar transport equations for the Level Set indicator function in our multiphase flow solver.

1.3.1 Discretization techniques

For a better illustration, we consider first of all numerical solution techniques for the (single-phase) incompressible Navier-Stokes equations,

$$\begin{aligned} &\mathbf{u}_t - \nu\Delta\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p_\rho = \mathbf{f} \\ &\nabla\cdot\mathbf{u} = 0, \quad \text{in } \Omega\times(0,T] \quad \text{with } p_\rho = \frac{p}{\rho} \quad \text{and } \nu = \frac{\mu}{\rho} \end{aligned} \tag{1.8}$$

for given force $\mathbf{f}$ which might contain the surface tension. Moreover, boundary values are prescribed on the boundary $\partial\Omega$ as well as an initial condition at $t = 0$. Solving this problem numerically is still a considerable task in the case of long time calculations and high Reynolds numbers, particularly in 3D and if the time dynamics is complex. The common solution approach is a separate discretization in space and time. We first (semi-) discretize in time by one of the usual methods known from the treatment of ordinary differential equations, such as the Forward or Backward Euler-, the Crank-Nicolson- or Fractional-Step-θ–scheme, or others, and obtain a sequence of generalized stationary Navier-Stokes problems.

Basic θ-scheme:
Given $\mathbf{u}^n$ *and* $\Delta t = t_{n+1} - t_n$*, then solve for* $\mathbf{u} = \mathbf{u}^{n+1}$ *and* $p_\rho = p_\rho^{n+1}$

$$\frac{\mathbf{u}-\mathbf{u}^n}{\Delta t} + \theta[-\nu\Delta\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u}] + \nabla p_\rho = \mathbf{g}^{n+1} \quad , \quad \nabla\cdot\mathbf{u} = 0 \quad , \quad \text{in } \Omega \tag{1.9}$$

with right hand side $\mathbf{g}^{n+1} := \theta\mathbf{f}^{n+1} + (1-\theta)\mathbf{f}^n - (1-\theta)[-\nu\Delta\mathbf{u}^n + \mathbf{u}^n\cdot\nabla\mathbf{u}^n]$

In the following simulations, the parameter θ is chosen as $\theta = 1/2$, representing the Crank-Nicolson-scheme which is of second order. Alternatively, the Fractional-Step-θ-scheme [59] which uses three different values for θ and for the time step Δt at each time level, is another excellent candidate with slightly better robustness properties.

For the spatial discretization, we choose a finite element approach which is based on a suitable variational formulation. On the finite mesh $\mathcal{T}_h$ (3D hexahedral elements in our case) covering the domain Ω with local mesh size h, one defines polynomial trial functions for velocity and pressure. These spaces H_h and L_h should lead to numerically stable approximations as $h \to 0$, i.e., they should satisfy the so-called *inf-sup* (LBB) condition [20]

$$\min_{q_h \in L_h} \max_{\mathbf{v}_h \in H_h} \frac{(q_h, \nabla\cdot \mathbf{v}_h)}{\|q_h\|_0 \; \|\nabla \mathbf{v}_h\|_0} \geq \gamma > 0 \tag{1.10}$$

with a mesh-independent constant γ. While the original FeatFlow solvers are based on *rotated multilinear* nonconforming finite element functions for the velocity and piecewise constant pressure approximations, we extended the complete solver package to higher order Stokes elements, namely conforming triquadratic ansatz functions for the velocity and discontinuous linear pressure approximations (Q_2/P_1), which belong to the 'best' finite element pairs for laminar incompressible flow w.r.t. accuracy and robustness. Since so far most of our numerical simulations have been performed for small up to moderate Reynolds numbers, the (nonlinear) convective operator was discretized using standard stabilization techniques only. Currently, we use edge-, resp., face-oriented FEM stabilization techniques [55] which can be easily realized for higher order ansatz functions, too. Here, special jump terms of the gradient of the solution as well as of the test function have to be included into the weak formulation which leads to a consistent stabilization, for stationary as well as nonstationary configurations. For an overview regarding such special FEM stabilization techniques, we refer to [41, 55] and particularly to [12] which contains corresponding results for the Q_2/P_1 approach, too.

1.3.2 Solution techniques

Using the same notation $\mathbf{u}$ and p_ρ also for the coefficient vectors in the representation of the approximate solution, the discretized Navier-Stokes equations may be written as a coupled (nonlinear) algebraic system of the form:

Given $\mathbf{u}^n$ *and* $\mathbf{f}$, *compute* $\mathbf{u} = \mathbf{u}^{n+1}$ *and* $p_\rho = p_\rho^{n+1}$ *by solving*

$$A\mathbf{u} + \Delta t B p_\rho = \mathbf{g} \quad , \quad B^T \mathbf{u} = 0, \qquad \text{where} \tag{1.11}$$

$$\mathbf{g} = [M - \theta_1 \Delta t N(\mathbf{u}^n)]\mathbf{u}^n + \theta_2 \Delta t \mathbf{f}^{n+1} + \theta_3 \Delta t \mathbf{f}^n \, . \tag{1.12}$$

Here and in the following, we use the more compact form for the diffusive and advective part

$$N(\mathbf{v})\mathbf{u} := -\nu\Delta\mathbf{u} + \mathbf{v}\cdot\nabla\mathbf{u} \tag{1.13}$$

while M is the (lumped) mass matrix [61], B is the discrete gradient operator, and $-B^T$ is the associated divergence operator. Furthermore,

$$A\mathbf{u} = [M + \theta\Delta t N(\mathbf{u})]\mathbf{u}, \qquad N(\mathbf{u}) = K(\mathbf{u}) + \nu L, \tag{1.14}$$

where L is the discrete Laplacian and $K(\mathbf{u})$ is the nonlinear transport operator incorporating a certain amount of artificial diffusion due to some appropriate FEM stabilization as described before. The solution of nonlinear algebraic systems like (1.11) is a rather difficult task and many aspects, namely the treatment of the nonlinearity and of the incompressibility as well as the outer control of the couplings, need to be taken into account. Consequently, this leads to a great variety of incompressible flow solvers which are closely related to one another but exhibit considerable differences in terms of their stability, convergence, and efficiency. The Multilevel Pressure Schur Complement (MPSC) approach outlined below makes it possible to put many existing solution techniques into a common framework and to combine their advantages so as to obtain better run-time characteristics.

The fully discretized Navier-Stokes equations (1.11) as well as the linear subproblems to be solved within the outer iteration loop for a fixed-point defect correction or, with a similar structure, for a Newton-like method admit the following representation

$$\begin{bmatrix} A & \Delta t B \\ B^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p_\rho \end{bmatrix} = \begin{bmatrix} \mathbf{g} \\ 0 \end{bmatrix}. \tag{1.15}$$

In general, we have $A = M + \beta N(\mathbf{u})$, with $\beta = \theta\Delta t$ for time-dependent problems. If the operator A is nonsingular, the velocity can be formally expressed as

$$\mathbf{u} = A^{-1}(\mathbf{g} - \Delta t B p_\rho) \tag{1.16}$$

and plugged into the discretized continuity equation

$$B^T\mathbf{u} = 0 \tag{1.17}$$

which gives a scalar *Schur complement* equation for the pressure only

$$B^T A^{-1} B p_\rho = \frac{1}{\Delta t} B^T A^{-1}\mathbf{g}. \tag{1.18}$$

Thus, the coupled system (1.15) can be handled as follows

1. Solve the Pressure Schur Complement (PSC) equation (1.18) for p_ρ.
2. Substitute p_ρ into relation (1.16) and compute the velocity $\mathbf{u}$.

It is worth mentioning that the matrix A^{-1} is full and should not be assembled explicitly. Instead, an auxiliary problem is to be solved by a direct method or by inner iterations. For instance, the velocity update (1.16) is equivalent to the solution of the discretized momentum equation $A\mathbf{u} = \mathbf{g} - \Delta t B p_\rho$. Likewise, the matrix $S := B^T A^{-1} B$ is never generated in practice. Doing so would be prohibitively expensive in terms of CPU time and memory requirements. It is instructive to consider a preconditioned Richardson method which yields the following **basic iteration** for the PSC equation

$$p_\rho^{(l+1)} = p_\rho^{(l)} - C^{-1}\left[S p_\rho^{(l)} - \frac{1}{\Delta t} B^T A^{-1} \mathbf{g}\right], \qquad l = 0, \ldots, L-1. \tag{1.19}$$

Here, C has to be chosen as suitable preconditioner to S but being easier to 'invert' in an iterative way. The number of PSC cycles L can be fixed or chosen adaptively so as to achieve a prescribed tolerance for the residual. The basic idea behind the family of global MPSC schemes is the construction of globally defined additive preconditioners for the Schur complement operator $S = B^T A^{-1} B$. Recall that the matrix A has the following structure

$$A := M + \beta K(\mathbf{u}) + \gamma L, \tag{1.20}$$

where $\beta = \theta \Delta t$ and $\gamma = \nu \beta$. Unfortunately, even today it is still a challenging task to construct a matrix $\tilde{A}$ and a preconditioner $C = B^T \tilde{A}^{-1} B$ that would be a sufficiently good approximation to all three components of A and S, respectively; particularly for the convective part with $K(\mathbf{u})$. Therefore, one may start with developing individual preconditioners for the reactive (M) and diffusive (L) part, while the convective (K) part is neglected by applying this special kind of operator splitting. In our case, the Reynolds numbers in the considered flow configurations are so far quite moderate, so that this approach can be justified, particularly if small time steps are used to resolve to complex dynamical behaviour. Therefore, the (lumped) mass matrix M proves to be a reasonable approximation to the complete operator A, so that our basic iteration (1.19) for the pressure Schur complement equation

$$p_\rho^{(l+1)} = p_\rho^{(l)} + [B^T M^{-1} B]^{-1} \frac{1}{\Delta t} B^T A^{-1} \left[\mathbf{g} - \Delta t B p_\rho^{(l)}\right] \tag{1.21}$$

can be interpreted and implemented as a *discrete projection scheme*, if $L = 1$, such as those proposed in [14, 22]. Here, the important step is that for the chosen Stokes element pair, Q_2/P_1, the matrix $P := B^T M^{-1} B$ can be explicitly built up relatively easily even in a domain decomposition framework due to the chosen discontinuous pressure. Then, the main algorithmic steps are as follows [53]:

1. Solve the ‘viscous Burgers’ equation for $\tilde{\mathbf{u}}$

$$A\tilde{\mathbf{u}} = \mathbf{g} - \Delta t B p_\rho^{(l)}.$$

2. Solve the discrete ‘Pressure-Poisson’ problem

$$P q_\rho = \frac{1}{\Delta t} B^T \tilde{\mathbf{u}}.$$

3. Correct the pressure and the velocity

$$p_\rho^{(l+1)} = p_\rho^{(l)} + q_\rho, \quad \mathbf{u} = \tilde{\mathbf{u}} - \Delta t M^{-1} B q_\rho.$$

In essence, the right-hand side of the momentum equation is assembled using the old pressure iterate and the intermediate velocity $\tilde{\mathbf{u}}$ is projected onto the subspace of solenoidal functions so as to satisfy the constraint $B^T\mathbf{u} = 0$. Moreover, the matrix P corresponds to a mixed discretization of the Laplacian operator [22] so that this method is a discrete analogue of the classical projection schemes derived by Chorin ($p_\rho^{(0)} = 0$) and Van Kan ($p_\rho^{(0)} = p_\rho(t_n)$) via operator splitting for the continuous problem.

Now, in the case of multiphase problems, the above addressed scheme serves as the core which has to be extended with additional mechanisms depending on the adopted realization (interface tracking or capturing) but still resulting in an operator splitting framework. The Navier-Stokes equations with the arising time-dependent and discontinuous material properties - density $\rho(\phi)$ and viscosity $\mu(\phi)$ - and surface tension term $\mathbf{f}_{\Gamma,\sigma}(\phi)$ can still be written for both realizations formally in a common way, as follows

$$\begin{aligned} \rho(\phi)\left[\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right] - \nabla\cdot(\mu(\phi)[\nabla\mathbf{u} + (\nabla\mathbf{u})^T]) + \nabla p = \\ = \rho(\phi)\mathbf{g} + \mathbf{f}_{\Gamma,\sigma}(\phi), \qquad \nabla\cdot\mathbf{u} = 0 \end{aligned} \tag{1.22}$$

which depending on the applied realization of the interface treatment is then extended with the corresponding additional transport equation. At first, let us extend the scheme towards the interface tracking realization according to that approach in which the additional transport equation is the mesh deformation PDE, which is governing the motion of the time-dependent computational mesh being deformed due to the tangential movement of the internal boundary points associated with the corresponding interface at every timestep. The corresponding mesh deformation equation is written as follows

$$\nabla\cdot[\nabla\mathbf{d} + (\nabla\mathbf{d})^T] = 0 \tag{1.23}$$

where $\mathbf{d}$ stands for the displacement vector. The equation above has to be extended with the corresponding boundary conditions acting on the outer boundaries of the domain, as well as on the interfacial surface, so to enable a tangential slip of the nodes on the individual (curved) surfaces. To do this, only the surface related to the interface requires special attention since that depends on the actual velocity distribution at the given time level. This coupling between the mesh deformation and the Navier-Stokes equations is realized by means of the projection method according to which the velocity field is obtained first in the framework of the ALE method for a fixed interface position Γ^n and by means of the mesh velocity ω^n: note that both originate from the previous time level t^n. Then the update of the interface position is performed from Γ^n to Γ^{n+1} by means of the interface-normal components of the velocity at the old interface, as follows

$$\Gamma^{n+1} = \Gamma^n + \mathbf{u}_n^{n+1} \cdot \Delta t \tag{1.24}$$

where $\mathbf{u}_n$ is the normal component of the velocity calculated by projecting the local velocity field into the interface normal direction evaluated from the old time level $\mathbf{n}_\Gamma^n$. After discretization in space and time, we obtain again a system of nonlinear algebraic equations which can be written in matrix form as follows:

$$A_u(\mathbf{u}^{n+1}, \Gamma^{n+1})\mathbf{u}^{n+1} + \Delta t F(\Gamma^{n+1}) + \Delta t B p^{n+1} = \mathbf{g}_u, \tag{1.25}$$

$$A_{\mathbf{d}}(\mathbf{u}^{n+1})\mathbf{d}^{n+1} = g_{\mathbf{d}}, \qquad B^T \mathbf{u}^{n+1} = 0 \tag{1.26}$$

Note that equation (1.25) in contrast to (1.11) and (1.14) is multiplied with $\rho(\Gamma)$, which gives rise to the modified operators M_ρ, $K_\rho(\mathbf{u})$ and L_μ. Here and below the superscript $n+1$ refers to the time level, while subscripts identify the origin of discrete operators ($\mathbf{u}$ for the momentum equation and ϕ for the Level Set equation); moreover, ρ and μ are evaluated w.r.t. the old time level t^n which makes this formulation semi-implicit. The system of nonlinear equations written above - corresponding to the interface tracking approach - is casted in the framework of the used Discrete Projection Method into the following solution scheme consisting of four main algorithmic steps:

1. Compute $\tilde{\mathbf{u}}$ from the momentum equation

$$A_u(\tilde{\mathbf{u}}, \Gamma^n)\tilde{\mathbf{u}} = \mathbf{g}_u - \Delta t F(\Gamma^n) - \Delta t B p^n.$$

2. Solve the discrete Pressure-Poisson problem

$$P_\rho q = \frac{1}{\Delta t} B^T \tilde{\mathbf{u}} \text{ with } P_\rho := B^T M_\rho^{-1} B.$$

3. Correct the pressure and the velocity

$$p^{n+1} = p^n + q, \quad \mathbf{u}^{n+1} = \tilde{\mathbf{u}} - \Delta t M_\rho^{-1} B q.$$

4. Solve the mesh deformation equation for $\mathbf{d}$ and obtain Γ^{n+1}

$$A_{\mathbf{d}} \mathbf{d}^{n+1} = 0$$

with boundary conditions depending on $\mathbf{u}^{n+1}$.

This solution strategy is realized for only one outer iteration step, however only under the condition that the used time step is sufficiently small.

Now, in case of addressing our multiphase flow problem in the framework of the interface capturing method, the system of equations (1.22) has to be extended with the Level Set transport equation (1.4).

After discretization in space and time, we obtain again a system of nonlinear algebraic equations which can be written in matrix form as follows

$$A_u(\mathbf{u}^{n+1}, \phi^{n+1})\mathbf{u}^{n+1} + \Delta t F(\phi^{n+1}) + \Delta t B p^{n+1} = \mathbf{g}_u, \tag{1.27}$$

$$A_\phi(\mathbf{u}^{n+1})\phi^{n+1} = g_\phi, \qquad B^T \mathbf{u}^{n+1} = 0. \tag{1.28}$$

Note that we have the freedom of using different finite element approximations and discretization schemes for the velocity $\mathbf{u}$ and indicator function ϕ, and the discrete problem (1.27),(1.28) can be solved again in the framework of the Discrete Projection Method. For relatively small time steps, this strategy works very well, and simulation software can be developed in a modular way making use of optimized multigrid solvers. Consequently, in the simplest case (just one outer iteration per time step), the sequence of algorithmic steps to be performed is as follows:

1. Compute $\tilde{\mathbf{u}}$ from the momentum equation

$$A_u(\tilde{\mathbf{u}}, \phi^n)\tilde{\mathbf{u}} = \mathbf{g}_u - \Delta t F(\phi^n) - \Delta t B p^n.$$

2. Solve the discrete Pressure-Poisson problem

$$P_\rho q = \frac{1}{\Delta t} B^T \tilde{\mathbf{u}} \text{ with } P_\rho := B^T M_\rho^{-1} B.$$

3. Correct the pressure and the velocity

$$p^{n+1} = p^n + q, \quad \mathbf{u}^{n+1} = \tilde{\mathbf{u}} - \Delta t M_\rho^{-1} B q.$$

4. Solve the Level Set equation for ϕ

$$A_\phi(\mathbf{u}^{n+1})\phi^{n+1} = g_\phi.$$

Due to the nonlinearity of the discretized convective terms, resp., of the reinitialisation step, iterative defect correction or Newton-like methods, resp., corrections via redistancing, must be invoked in steps 1 and 4. However, due to the assumed relatively small time steps, such nonlinear iteration methods are not critical for the complete flow simulation.

1.4 TREATMENT OF SURFACE TENSION EFFECTS

Surface tension effects are taken into account through the following force balance at the interface Γ:

$$[\mathbf{u}]|_\Gamma = 0, \quad [-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)]\big|_\Gamma \cdot \mathbf{n} = \sigma\kappa\mathbf{n}$$

The first condition implies continuity of the velocity across the interface, whereas the second describes the force balance on Γ. $\mathbf{n}$ is the unit normal at the interface pointing into Ω_1, $[\mathsf{A}]|_\Gamma = \mathsf{A}|_{\Omega_1\cap\Gamma} - \mathsf{A}|_{\Omega_2\cap\Gamma}$ denotes the jump of a quantity A across the interface, σ is the surface tension coefficient, and κ is the curvature of the interface Γ. The surface tension term according to the used finite element discretization is transformed into its variational equivalent by multiplying it with a suitably chosen test function and integrating it over Γ, which yields:

$$\mathbf{f}_{st} = \int_\Gamma \sigma\kappa\mathbf{n} \cdot \mathbf{v} d\Gamma$$

Since the above addressed integration has to be performed on the interfacial surface which in general might not be be aligned with the computational mesh, there are different possibilities offered in order to account for the appropriate representation of this source term. According to the more natural but at the same time also more challenging representation, the integration is indeed performed on the interfacial surface in a framework of a time dependent deformed computational mesh being aligned with the interface (front tracking method) by taking advantage of the Laplace Beltrami transformation by introducing the surface tangential derivatives $\underline{\Delta}$ and $\underline{\nabla}$

$$\mathbf{f}_{st} = \int_{\Gamma} \sigma \left(\underline{\Delta} \mathbf{x}|_{\Gamma} \right) \cdot \mathbf{v} d\Gamma$$

which then, by applying partial integration, transfers one derivative into the test function

$$\mathbf{f}_{st} = -\int_{\Gamma} \sigma \underline{\nabla} \mathbf{x}|_{\Gamma} \cdot \underline{\nabla} \mathbf{v} d\Gamma + \int_{\gamma} \sigma \partial_{\gamma} \mathbf{x}|_{\Gamma} \cdot \mathbf{v} d\gamma$$

where the second term will be vanishing for closed interfacial surfaces. In order to reduce the related time step restriction we follow the work of Bänsch [6, 7] and Hysing [27, 28] and introduce a semi-implicit treatment of the surface tension term by taking advantage of

$$(\mathbf{x}|_{\Gamma})^{n+1} = (\mathbf{x}|_{\Gamma})^{n} + \Delta t \mathbf{u}^{n+1}$$

where Δt is the time step and $\mathbf{u}^{n+1}$ is the velocity field at the new time step. Under the assumption of having a continuous ALE velocity of the computational mesh we arrive at the following representation of the surface tension term

$$\mathbf{f}_{st} = -\int_{\Gamma^n} \sigma \underline{\nabla} \left(\mathbf{x}|_{\Gamma} \right)^{n} \cdot \underline{\nabla} \mathbf{v} d\Gamma - \Delta t \int_{\Gamma^n} \sigma \underline{\nabla} \mathbf{u}^{n+1} \cdot \underline{\nabla} \mathbf{v} d\Gamma$$

where the second term represents a positive definite matrix which can be directly built into the stiffness matrix guaranteeing a relaxation of the time step restriction due to surface tension.

The second strategy, which is used in our front capturing Level Set method, is related to extending the surface integral to a volumetric one by the help of the Dirac delta funcion $\delta(\mathbf{x})$ corresponding to the location of the interface:

$$\mathbf{f}_{st} = \int_{\Omega} \sigma \kappa \mathbf{n} \cdot \mathbf{v} \delta \left(\Gamma \right) dx$$

This strategy forms the basis of the Continuous Surface Force (CSF) approach. Accordingly the Dirac Delta function δ is to be replaced by its continuous regularized counterpart which in particular has the following construction:

$$\delta(\phi) = \begin{cases} \phi < 0 & \max\left(0, \frac{1}{\epsilon} + \frac{1}{\epsilon^2}\phi\right) \\ \\ \phi \geq 0 & \max\left(0, \frac{1}{\epsilon} - \frac{1}{\epsilon^2}\phi\right) \end{cases}$$

The introduced parameter ϵ is the extent into which the volumetric integration is extended. Its choice is clearly not arbitrary, but related to the underlying mesh resolution h. Since the interface normal vector $\mathbf{n}$ and curvature κ are higher order derivatives of the Level Set function ϕ, their spatial distributions can be obtained by a combination of appropriate projection and gradient recovery techniques. Additionally, due to the employed explicit treatment in the framework of the used interface capturing method, all evaluations are related to the Level Set solutions being evaluated at the previous time step. Accordingly, the continuous (piecewise trilinear) interface normal $\mathbf{n}^n_{Q_1}$ is obtained by L_2-projection (and normalization) from the piecewise discontinuous P_1 space into the continuous Q_1 space. Finally, the continuous approximation $\kappa^n_{Q_1}$ of the curvature κ^n is reconstructed via L_2-projection, as follows

$$\int_\Omega \kappa^n_{Q_1}\, w dx = -\int_\Omega w\, \nabla \cdot \mathbf{n}^n_{Q_1} dx, \tag{1.29}$$

where w denotes the test functions from the conforming trilinear Q_1 space.

1.5 THE $Q_2/P_1/Q_1$-FEM-INTERFACE TRACKING APPROACH

In the current work in case of the front tracking method, the computational mesh is always perfectly aligned with the free interface Γ and therefore the computational mesh is deformed in every timestep in order to follow the movement of the interfacial surface. For this reason a suitable mesh deformation algorithm is needed to be integrated into the computational scheme which is in this case the simple PDE based Mesh Deformation Equation (MDE) already addressed in (1.23). The spatial discretization of the MDE is realized by FEM in the Q_1 space. Since the underlying equation is time-independent (i.e. steady state) and has to be solved at every timestep due to the time dependent boundary conditions, its solution is accelerated by the employed geometric multigrid methods. The only disadvantage of this methodology is that it is not suitable to introduce the slip condition of the displacement in the surface tangential directions as for example described in ([15]) by transformation of the corresponding degrees of freedom and related matrix entries into a local coordinate system. Therefore we have embedded the underlying slip conditions of the boundary nodes into the defect correction scheme described in [33] by transforming the (otherwise linear) MDE to a nonlinear solution algorithm which then needs to be solved only up to a relaxed tolerance. Accordingly, in every nonlinear solution step the resulting normal and tangential directions are updated which then define the direction of the applied slip condition for the individual degrees of freedom. Additional remarks to the realization of the mesh deformation are necessary due to the fact that only the motion of the Q_1 nodes (the vertices of the underlying hexahedral elements) is governed by the MDE and therefore the coordinates of the remaining Q_2 nodes need to be performed carefully on an algebraic level. This in particular means to obtain the updated coordinate values of all edge-, face- and element-centered nodes which, in case of being located on a curved surface (either outer or interfacial surface), therefore need to be projected onto the underlying curved surface. The generally used coordinate averaging algorithm might fail especially in case of already highly stretched elements where the obtained averaged face- or element-midpoints can displace the resulting point outside of the corresponding element. For this reason we propose to update the face coordinate by using a limited edge-point based averaging according to which coordinate of the face midpoint will be calculated from the shorter pair of edge-midpoints (see Figure 1.2). The same mechanism is then transferrable for the update of the element midpoints based on the face-midpoint coordinate averages.

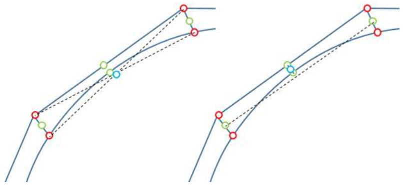

FIGURE 1.2 Midpoint calculation strategies. Left: based on the vertices of the element (red) which may result in placing the midpoint outside of the element. Right: based on the shortest edge-midpoint (green), recovering an acceptable coordinate.

After the computation of the corresponding edge-, face- and element-oriented midpoints, the surface normal and resulting surface tangential vectors are reconstructed by means of a FEM L_2-projection for the isoparametric Q_2 element.

For completeness, the mesh deformation equation after applying the FEM discretization can be written in the following simple form

$$L_{\mathbf{d}}\,\mathbf{d} = 0 \tag{1.30}$$

where the $L_{\mathbf{d}}$ operator is the discrete Q_1 diffusion operator which (as recommended in [4]) due to the motion of the computational mesh during the simulation is updated based on the last available mesh coordinates from time level t^n. This way the displacement solution vector is incremented with respect to the computational mesh from the last time step instead of the initial mesh at $t = 0$.

Concerning the ALE formulation we refer to [8] which is basically related to addressing a generic conservation equation for a quantity b (like linear momentum in our case) written as

$$\partial_t b + \operatorname{div}(b\mathbf{u}) = -\operatorname{div}\mathbf{F} + B_V, \tag{1.31}$$

where $\mathbf{F}$ corresponds to the diffusive flux, B_V refers to the volumetric or surface force terms. Here, in the framework of the used ALE formulation, we define the mesh velocity as $\omega(\mathbf{x}, t)$ so that the corresponding transport equation is, according to the used ALE formulation, transformed to

$$\widehat{\partial_t} b + (\mathbf{u} - \omega) \cdot b + (\mathrm{div}\mathbf{u})\, b = -\mathrm{div}\mathbf{F} + B_V, \tag{1.32}$$

where the $\widehat{\partial_t} b$ stands for the ALE time derivative which in the case of a general time discretization framework introduces the problem of evaluating the old velocity distribution on the new mesh. However, this difficulty can be overcome by the use of the fully implicit Euler method (as used in this work) which at the same time results in an only first order temporal discretization of the overall scheme.

The moving reference frame is introduced to fix the barycenter of a single drop or bubble to the origin and this way the coordinate system experiences acceleration which has the consequence of a transformation of the momentum balance equations in terms of the Dirichlet values and modifying the volume force $\mathbf{f} = \rho\mathbf{g}$ to $\mathbf{f} = \rho(\mathbf{g} - \mathbf{a})$ where $\mathbf{a}$ is the acceleration of the frame velocity.

1.6 THE $Q_2/P_1/Q_2$-FEM-INTERFACE CAPTURING APPROACH

In the current work in case of the front capturing method, the location of the free interface Γ is treated by means of the Level Set approach which represents the interface as a zero isosurface of a continuous indicator function ϕ and its distribution in the complete domain Ω can then be calculated via the following scalar transport equation:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0 \tag{1.33}$$

Since ϕ approximates the distance function, it is smooth and it allows the calculation of the globally defined normal vector $\mathbf{n}$ at the interface Γ and of the corresponding curvature via 1.7, which are evaluated by the help of FEM based special gradient recovery techniques (as desribed in Sec. 1.4).

The implementation of the here applied Level Set approach consists of the following sequence of tasks:

- Advection of the Level Set function based on the underlying velocity field $\mathbf{u}$ (as in (1.33))
- Geometric reinitialization of the Level Set function
- Recovery of the normal vector $\mathbf{n}$ and curvature κ
- Evaluation of the discontinous fluid parameters ρ, μ, and $\mathbf{f}_{st}$ based on the reconstructed distribution of Γ

One has to keep in mind that the reinitialization of the Level Set field is nececcary for two main reasons, and these are that by reinitialization the Level Set function becomes smooth and stabilization of the convection terms does not need to be incorporated, while the second reason is that the recovery of the curvature in (1.7) requires as input a distance field. On the other hand, it is well known from the literature, that the reinitialization of the Level Set field leads to artificial "movement" of the interface, and may exhibit even its systematic translation in cases when it is performed too often. That is the reason why the two previously mentioned requirements can be satisfied through an additionally introduced function $\tilde{\phi}$. According to the proposed approach one can reinitialize the Level Set field ϕ into $\tilde{\phi}$ and all the postprocessing steps (recovery of the normals, curvature and physical properties) are then performed only with respect to $\tilde{\phi}$ (which is the most accurate approximation of the distance distribution) without influencing the interface. This way, reinitialization of the Level Set field ϕ becomes crucial only for stability reasons. The link between the fields ϕ and $\tilde{\phi}$ is the interface Γ which is reconstructed from the Level Set distribution ϕ at every timestep. During this global surface reconstruction process all elements of the computational domain intersected by the interface are visited and according to the used higher order Q_2 element the zero level surface is triangulated in a recursive fashion so to provide subgrid resolution of the interface. The arising set of triangles is then organized into groups with respect to the elements they originate from. Due to the underlying multigrid mesh structure this data management is performed in a hierarchical manner while for the coarser level representation for efficiency reasons only the mass of points of the individual groups are to be used. This allows us to design a very efficient reinitialization algorithm, which starts on the coarse representation of the surface and ends by determination of the closest triangle to an analyzed point. The role of efficiency in such a "point to triangle distance" implementation plays a crucial role since the reinitialization described by us is related to a L_2-projection, as follows

$$\int \tilde{\phi}\; w\; dx = \int \text{dist}\; w\; dx, \tag{1.34}$$

where w is the Q_2 basis funcion and ‘dist’ is the distance which on the level of practical realization means the evaluation of the distance with respect to the surface in each cubature point during the numerical integration. This linear equation can also be solved iteratively by the help of the lumped mass matrix M_L, and according to our experience, already the first iteration of this solution process provides an accurate approximation:

$$\tilde{\phi} = M_L^{-1} \int \text{dist}\; w\; dx \tag{1.35}$$

Then, for the reinitialization of the Level Set function two procedures offer themselves. The first alternative is a full one ($\phi := \tilde{\phi}$) with a given frequency f_{reini} or according to the second alternative it can be performed in an underrelaxation approach even at every timestep:

$$\phi := M_L^{-1} \int (\alpha_{\text{reini}}\text{dist} + (1 - \alpha_{\text{reini}})\phi)\; w\; dx. \tag{1.36}$$

1.7 NUMERICAL BENCHMARKING AND SIMULATIONS

This section contains several numerical studies for examining and validating the methodology described in the previous sections. Moreover, the presented multiphase flow configurations are designed to serve as benchmarking configurations, hereby providing reference values for the quantitative evaluation of new approaches for 3D multiphase flow problems.

1.7.1 Single-phase 3D flow around a cylinder

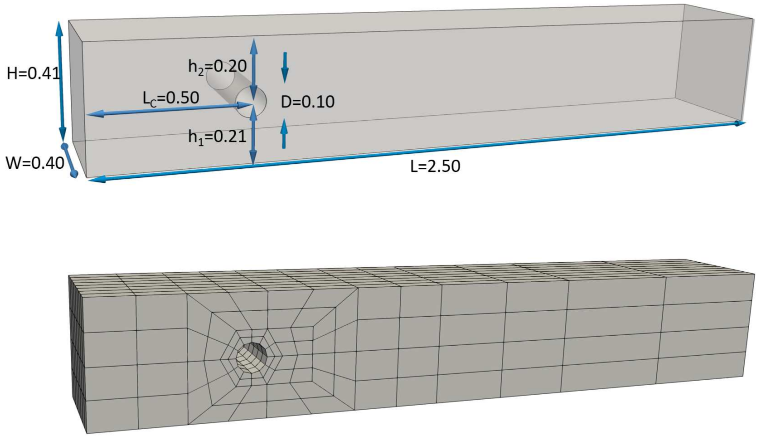

FIGURE 1.3 Geometry and coarse mesh for the ‘Flow around cylinder’ benchmark.

The first incompressible flow problem to be dealt with, particularly to demonstrate the accuracy of the higher order Q_2/P_1 approach, is the well-known benchmark *Flow around cylinder* developed in 1995 for the priority research program “Flow simulation on high-performance computers” under the auspices of DFG, the German Research Association [58]. This project was intended to facilitate the evaluation of various numerical algorithms for the incompressible Navier-Stokes equations in the laminar flow regime. A quantitative comparison of simulation results is possible on the basis of relevant flow characteristics such as pressure values as well as drag and lift coefficients, for which sufficiently accurate reference values are available (see also: `www.featflow.de/en/benchmarks/ff_benchmarks.html`). Here, we consider two test cases, namely steady and nonstationary incompressible flow around a cylinder with circular cross-section. An in-depth description of the geometrical details and boundary conditions can be found in reference [58] and particularly in [5] which contain all relevant information regarding this benchmark configuration. This single-phase flow problem is chosen here with the special aim to demonstrate the accuracy improvement related to the use of isoparametric finite elements, which allow to exploit the full order of the employed finite element approximations also in case of curved boundaries. The role of these discretization aspects becomes especially crucial in case that the quantities computed at such curved boundaries have backcoupling effects, just as it is in case of multiphase problems involving strong surface tension effects on evolving interfaces.

The first test case, the flow at $Re = 20$ is actually dominated by diffusion and could be simulated by the standard Galerkin method without any extra stabilization. The corresponding results are shown in Table 1.1 which demonstrates the numerical properties of the Q_2/P_1 approach. The advantage of the isoparametric version of the same numerical framework is clearly visible via the achieved L_2 error reduction which is especially visible at the finer resolution levels.

TABLE 1.1 Mesh convergence results - obtained by successive refinement of the coarse mesh in Fig.1.3 - in terms of drag, lift for the 'Flow around cylinder' problem at $Re = 20$. C_D and C_L are the normalized ($\frac{1}{2}\rho U^2_{\text{mean}} L_{\text{cyl}} D_{\text{cyl}}$) drag and lift coefficients. As reference values are used for drag $C_D = 6.185330$ and for lift $C_L = 0.009401$ [30].

	mesh stats		non-isoparametric		isoparametric	
Level	NEL	NDOF($\mathbf{u}, p$)	C_D $\%err$	C_L $\%err$	C_D $\%err$	C_L $\%err$
3	6,144	199,200	6.13973 0.7372	0.009569 1.7870	6.18040 0.0797	0.009881 5.1058
4	49,152	1,482,816	6.17433 0.1778	0.009381 0.2127	6.18455 0.0126	0.009464 0.6701
5	393,216	11,432,064	6.18261 0.0440	0.009387 0.1489	6.18521 0.0019	0.009407 0.0638
6	3,145,728	89,760,016	6.18465 0.0110	0.009397 0.0425	6.18531 0.0003	0.009402 0.0106

The second testcase is related to a nonstationary realization of the same geometrical and parameter setup of the first test case which is achieved by a time dependent inflow boundary condition starting with a fluid in rest reaching a maximum flowrate and Reynolds number at $t = 4.0$ and then decreasing the flowrate back to zero at $t = 8.0$. For detailed description we refer to [58] and particularly to [5] which reports the currently available reference computation results and provides comparisons with other mainstream CFD software tools. The reported reference results were obtained with the same Q_2/P_1 FEM simulation software as the one used in this work with the only difference of the isoparamtric element transformation which proves to be an essential ingredient guaranteeing additional accuracy for the price of only negligibly increased computational cost spent in matrix assembly procedures. For this test case we provide the demonstration of the improved accuracy due to the use of isoparametric transformations in terms of graphical proofs which are displayed in Figure 1.4. Due to the fact that the prediction of the target drag and lift coefficients is qualitatively already very close already on the 3rd resolution level to the reference curves, the corresponding close-up views of the interesting maximum and minimum regions are supplemented next to the overall $t = [0, 8s]$ evolution of benchmark quantities. As it may be visible from the inserted figures the isoparametric counterpart is predicting considerably more accurate drag and particularly lift values at each resolution level.

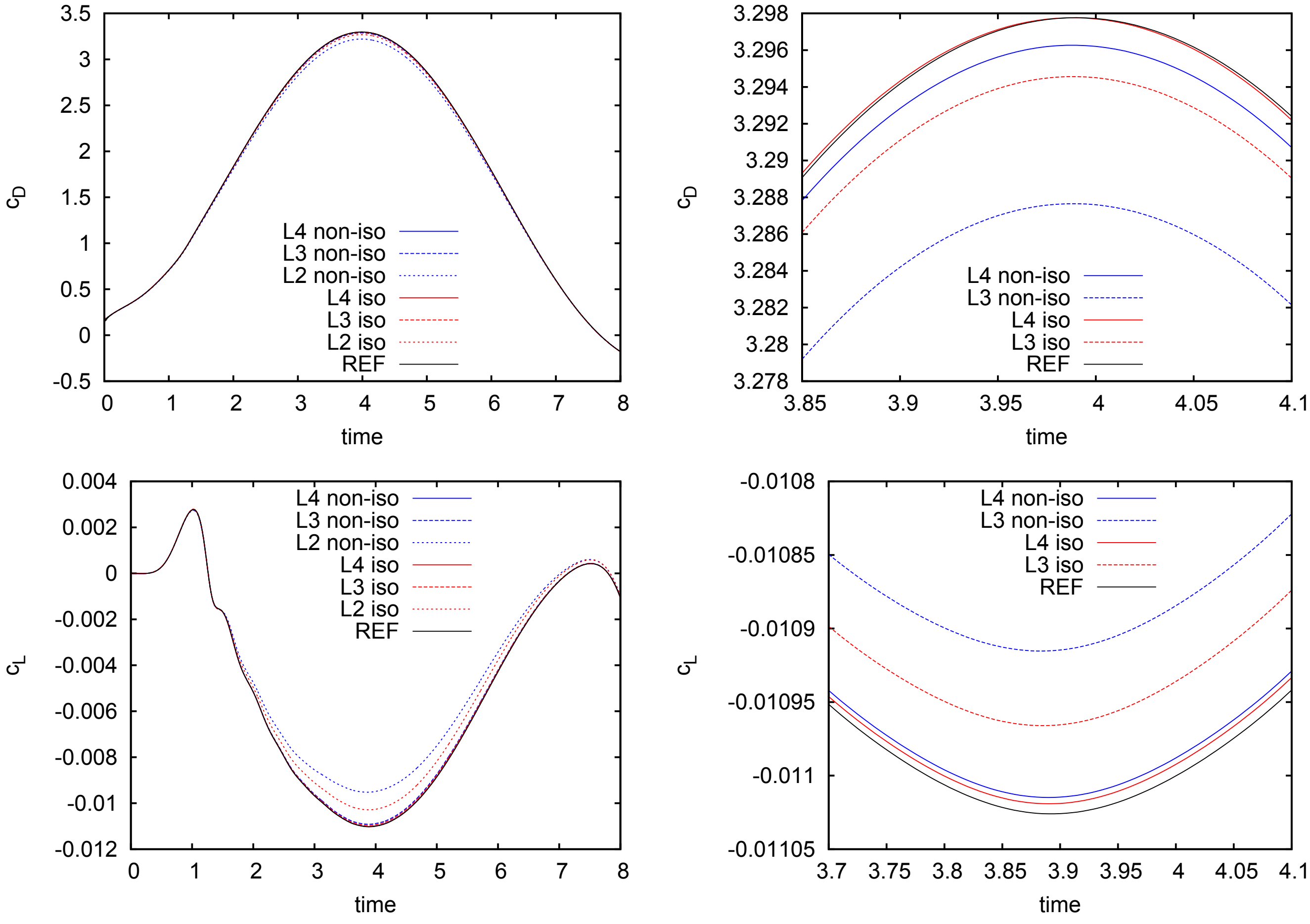

FIGURE 1.4 Results of the instationary Flow around a cylinder benchmark. Time evolution of the drag coefficient (top), time evolution of the lift coefficient (bottom). Close-ups for the drag-maximum and lift-minimum regions are visualized at the right side. REF stands for the reference results published in Bayraktar *et al.* [5].

1.7.2 3D Rising Bubble Benchmark

Having validated the 3D single-phase flow solver we provide the computational details of a 3D multiphase flow benchmark problem which follows its already well-established 2D counterpart originally introduced by Smoliansky [51] and later analyzed by Hysing *et al.* [29]. The only introduced difference with respect to the 2D benchmark case (aside of the additional spatial dimension) is the type of boundary condition applied at the side of the channel which according to the original 2D is attributed to *free slip* boundary condition, but in the 3D benchmark [29] it is related to a *no slip* boundary condition. Pioneering work on the 3D extension of this benchmark is related to Adelsberger *et al.* [1] where simulation results for different multiphase flow solvers have been presented and compared. However, due to the relatively widely spread computational results and absence of detailed convergence analysis, the determination of the benchmark results was not fully accomplished. The geometrical setup according to the 3D configuration is depicted in Figure 1.5 and the summary of all dimensionless physical parameters is displayed in Table 1.2. Both simulation frameworks (front tracking and front capturing) have been utilized for this benchmark problem and their respective spatial and temporal convergence have been taken into consideration. A partial compilation of the here presented results (missing the finest spatial and temporal resolution results) has been published by Turek *et al.* [57]. The sequence of computational meshes used for the Level Set based front capturing simulation was created on the basis of simple fully-structured and isotropic mesh generation where the corresponding mesh sizes were varied in the range of $1/24$ to $1/128$ by using two basic coarse meshes of $1/24$ and $1/32$ which have been refined to levels 2, 3 and 4 according to geometrical multigrid techniques. In case of the front tracking simulations an interface-aligned unstructured coarse mesh (displayed in the right subfigure of Figure 1.5) has been used for the simulations which was then refined according to geometrical multigrid techniques up to resolution levels 2, 3 and 4.

TABLE 1.2 The complete parameter set of dimensionless physical properties used for the 3D Benchmark problem according to Hysing *et al* [29].

ρ_1	ρ_2	μ_1	μ_2	g	σ
1000	100	10	1	0.98	24.5

The considered benchmark quantities are selected as for the original 2D benchmark, namely:

- **Bubble Size** - Size of the bubble in two different directions, namely in the rise direction (z) expressed as R_z/R_0 (normalized *w.r.t.* initial bubble size R_0) and in a perpendicular to rise direction being aligned with one of the cartesian axis (x or y) expressed as $R_{x,y}/R_0$.

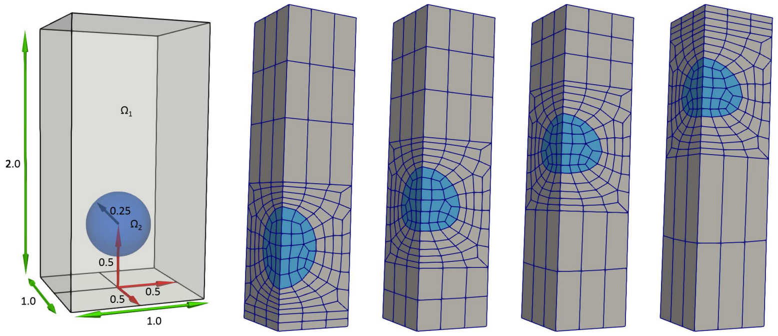

FIGURE 1.5 Left: Geometrical setup of the 3D rising bubble benchmark. Right: Sequence of coarse meshes used for the transient ALE based front-tracking simulations.

- **Bubble Sphericity** - The "degree of sphericity" in $\mathbb{R}^3$ can be defined as

$$\frac{A_0}{A} = \frac{4\pi R_0^2}{A}\,.$$

Here, A_0 denotes the interfacial area of an equivalent (in terms of volume) spherical bubble of radius R_0 (i.e. the surface area of the initial bubble) and A denotes the interfacial area of the simulated bubble at the given time level.

- **Rise Velocity** - The mean velocity with which the bubble is rising or moving and is defined as

$$\mathbf{U}_c = \frac{\int_{\Omega_2} \mathbf{u}\, dx}{\int_{\Omega_2} 1\, dx}$$

where Ω_2 denotes the region that the bubble occupies. The velocity component in the direction opposite to the gravity vector is then denoted as rise velocity V_c, for which the stationary limit is called 'terminal velocity'.

According to the definition of this benchmark problem, the evolution of all these quantities is to be predicted in the course of a transient simulation starting with a bubble at rest, which due to the buyoancy forces begins to rise in the surrounding fluid giving rise to an interplay of surface tension forces with the hydrodynamic forces in form of deformation of the bubble surface. The duration of the simulation covers a dimensionless time interval from $t = 0$ up to $t = 3$.

First, let us to present the results obtained for the front tracking benchmark computations. The results are organized into tables (see Table 1.3 and 1.4) and into graphical representations (see Figure 1.10 and 1.11) in order to obtain a visual representation of the convergence study. Since in case of the here adopted front-tracking technique - due to the used ALE scheme - the temporal discretization is only of first order, the investigation of the temporal convergence

TABLE 1.3 Absolute convergence of the monitored quantities at final time t=3.0 w.r.t. temporal and spatial refinements - Front tracking framework. Explanation of the symbols is as follows: L - spatial refinement level, Δt is the used time step, V_0, V are the initial and actual volumes of the bubble, z_c is the z location of the bubble associated with its point of mass. The remaining parameters are introduced in Section 1.7.2.

L	$\frac{\Delta t}{10^3}$	A_0/A	$\frac{V_0-V}{V_0}\%$	z_c	V_c	R_z/R_0	$R_{x,y}/R_0$
2	1/64	0.96070	0.19833	0.97311	0.34899	0.73328	1.15683
2	1/16	0.95984	0.06895	0.97282	0.34885	0.73359	1.15735
2	1/4	0.95911	0.05389	0.97238	0.34865	0.73408	1.15752
2	1/1	0.95625	0.53683	0.97060	0.34787	0.73601	1.15862
3	1/64	0.95928	0.00622	0.97267	0.34877	0.73368	1.15781
3	1/16	0.95910	0.03623	0.97256	0.34872	0.73380	1.15793
3	1/4	0.95839	0.15608	0.97211	0.34852	0.73428	1.15819
3	1/1	0.95557	0.63327	0.97034	0.34775	0.73621	1.15932
4	**1/64**	**0.95924**	**0.01178**	**0.97264**	**0.34876**	**0.73370**	**1.15786**
4	1/16	0.95906	0.04161	0.97253	0.34871	0.73382	1.15799
4	1/4	0.95836	0.16085	0.97209	0.34852	0.73430	1.15827
4	1/1	0.95556	0.63553	0.97033	0.34775	0.73623	1.15938

TABLE 1.4 Relative convergence of the monitored quantities at final time t=3.0 w.r.t. temporal and spatial refinements - Front tracking framework. Explanation of the symbols is as follows: L - spatial refinement level, Δt is the used time step, V_0, V are the initial and actual volumes of the bubble, z_c is the z location of the bubble associated with its point of mass. The remaining parameters are introduced in Section 1.7.2.

L	$\frac{\Delta t}{10^3}$	A_0/A	z_c	V_c	R_z/R_0	$R_{x,y}/R_0$
2	1/64	0.1520	0.0483	0.0659	0.0564	0.0897
2	1/16	0.0631	0.0185	0.0258	0.0145	0.0445
2	1/4	0.0132	0.0267	0.0315	0.0523	0.0298
2	1/1	0.3113	0.2097	0.2552	0.3154	0.0656
3	1/64	0.0042	0.0031	0.0029	0.0025	0.0044
3	1/16	0.0144	0.0082	0.0115	0.0142	0.0057
3	1/4	0.0884	0.0545	0.0688	0.0802	0.0283
3	1/1	0.3818	0.2365	0.2896	0.3427	0.1256
4	1/64	-	-	-	-	-
4	1/16	0.0184	0.0113	0.0143	0.0167	0.0107
4	1/4	0.0918	0.0565	0.0688	0.0829	0.0349
4	1/1	0.3831	0.2375	0.2896	0.3459	0.1312

plays a crucial role. For this reason aside of the spatial convergence, special attention has been paid to the temporal convergence, as well. Our converged results are therefore related to the Level 4 resolution level and $1/64 \cdot 10^{-3}$ time step size. In Table 1.3 are listed the benchmark quantities at the end ($t = 3.0$) of the simulation, which are then further processed in Table 1.4 which reveals the relative convergence of the underlying simulations. As it is seen from the presented results, the spatial convergence of the simulations is much faster than the temporal (as expected), and the difference between the Level 3 and Level 4 results is already achieved to a satisfactory extent. Moreover, this observation is valid for all temporal resolution levels. The influence of the temporal discretization level - due to the mentioned first order scheme - is responsible for the need to use so drastically small timestep sizes giving rise to simulations of 192,000 timesteps. The graphical representation of the evolution of the chosen benchmark quantities such as the rising speed and diameter are displayed in Figure 1.6 for bubble sphericity and relative bubble mass in Figure 1.7. Having the so far computationally estimated evolution of the benchmark quantities from the work of Adelsberger [1] makes it possible to compare the here estimated evolution of the benchmark quantities. These comparisons in terms of the bubble rise velocity and diameter are displayed in Figure 1.8 and for the bubble sphericity in Figure 1.9, respectively. As seen from the mentioned figures, the best agreement from the available data is provided by the NaSt3D simulation tool [1].

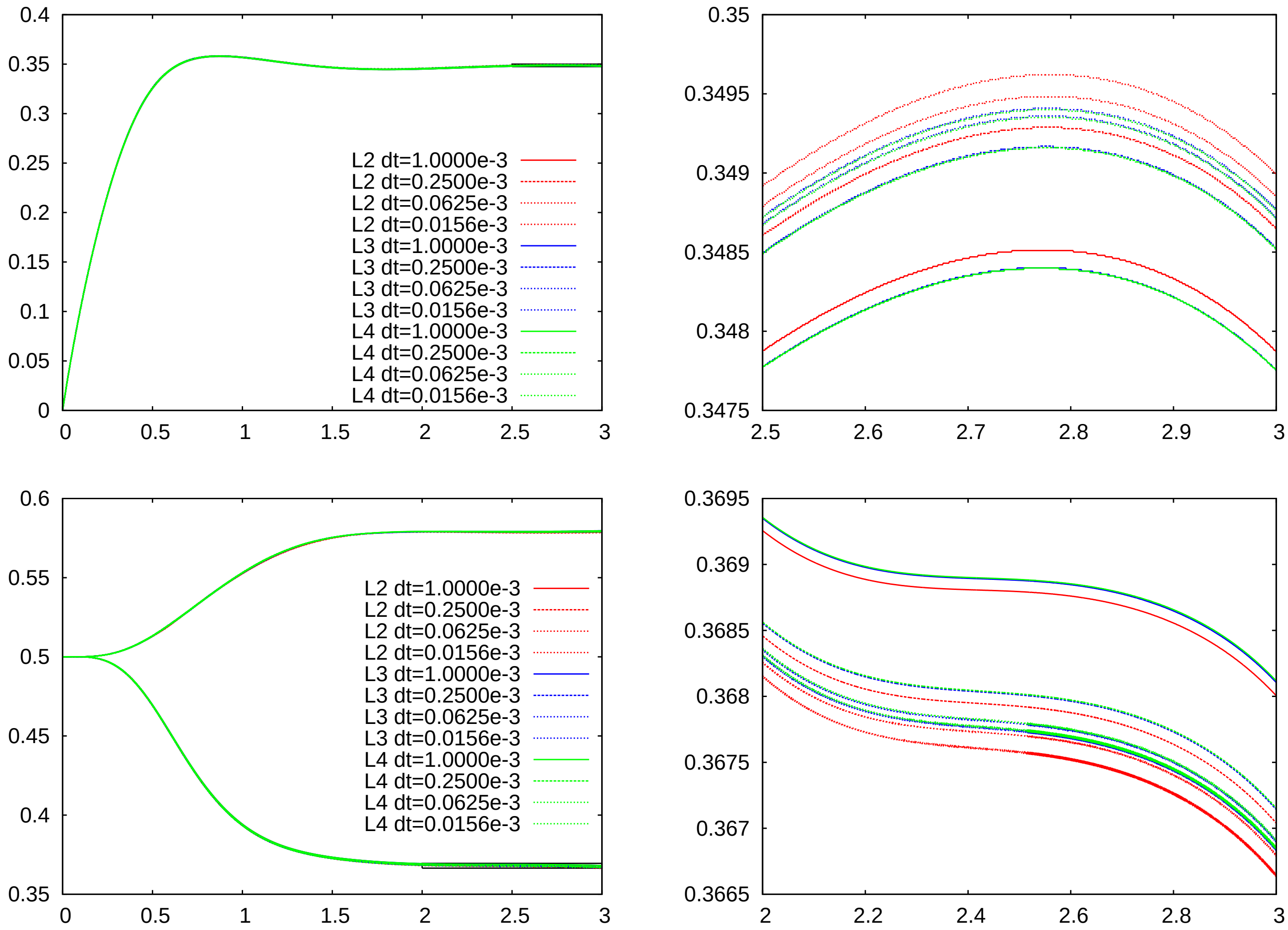

FIGURE 1.6 From top to bottom: Evolution of bubble rise velocity V_c, bubble diameter $2R_z$, $2R_{x,y}$ for the 3D benchmark problem (Front tracking method). L2, L3, L4 represent the level of spatial resolutions and dt stands for the size of the corresponding time step.

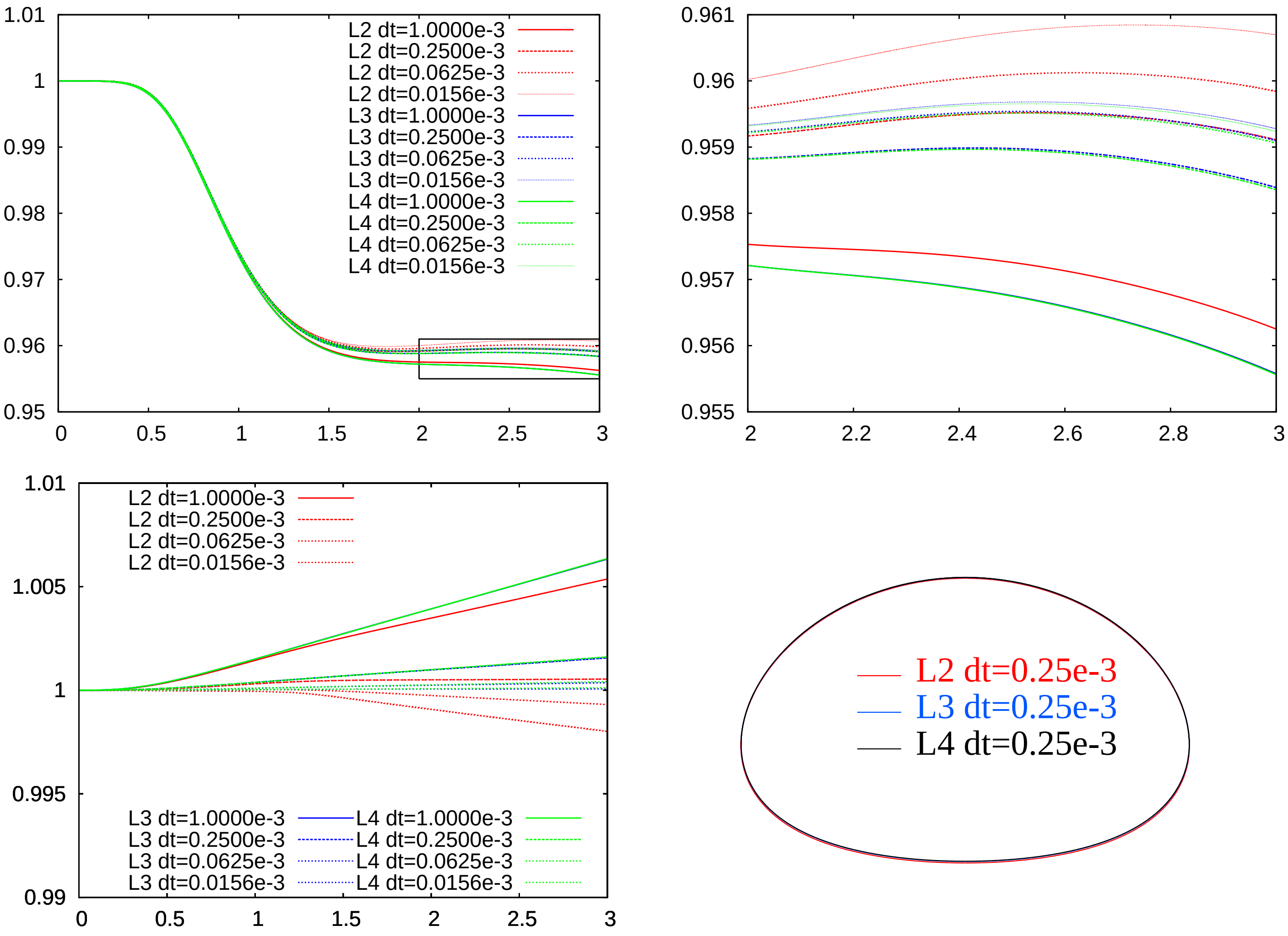

FIGURE 1.7 From top to bottom: Evolution of bubble sphericity A_0/A, relative bubble mass V/V_0 and bubble shape for the 3D benchmark problem (Front tracking method). L2, L3, L4 represent the level of spatial resolutions and dt stands for the size of the corresponding time step.

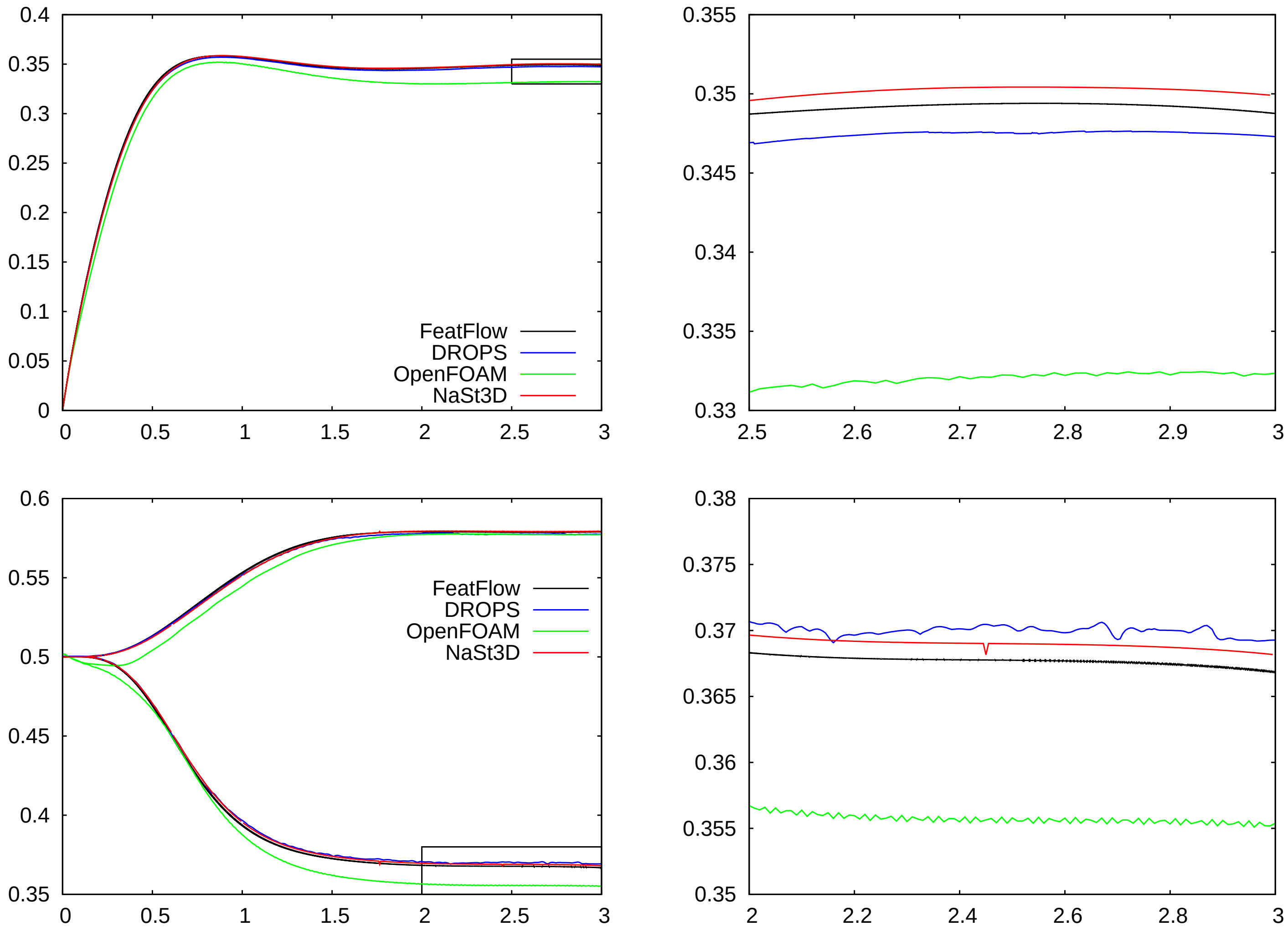

FIGURE 1.8 From top to bottom: Evolution of bubble rise velocity V_c, bubble diameter $2R_z$, $2R_{x,y}$ in comparison with the results published by Adelsberger *et al.* [1].

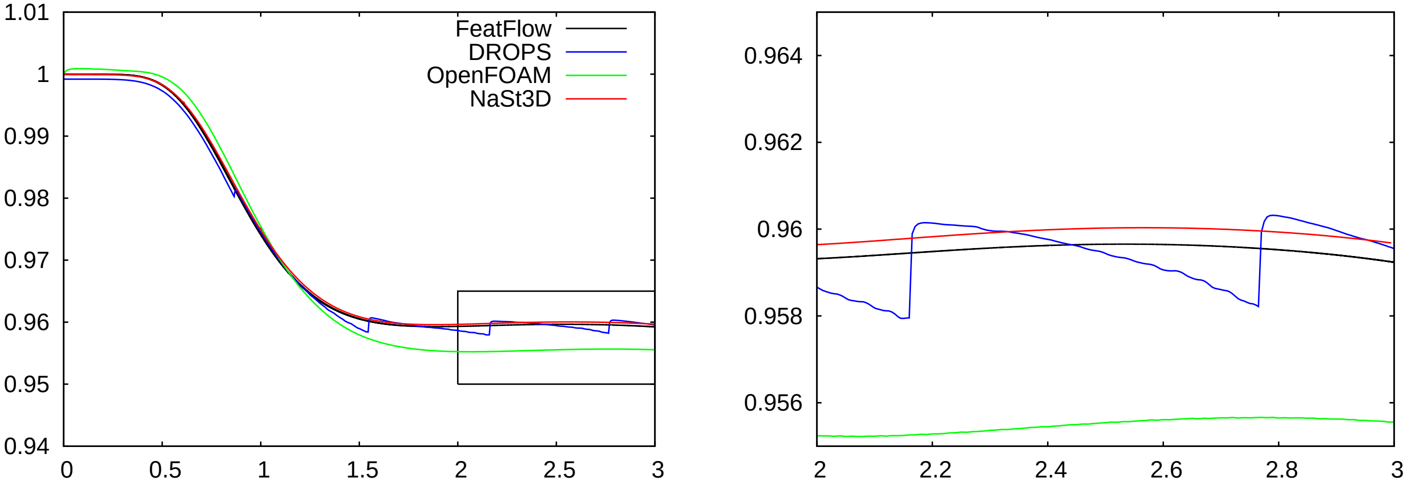

FIGURE 1.9 From top to bottom: Evolution of bubble sphericity A_0/A in comparison with the results published by Adelsberger *et al.* [1].

Due to the achieved highly accurate results, which seem to be (almost) mesh independent, we consider from now on these results to be the reference results which are subsequently used as reference for the obtained interface capturing method. In contrast to the interface tracking method, the interface capturing method is not parameter-free and, in fact, is related to two artificial parameters which have to be adjusted according to the used spatial resolution. One of them is the thickness of the smoothening distance ϵ which is directly related to the used element size and therefore is here of order h. The choice of this parameter was following the relation of $\epsilon = 2 \cdot h$ for all spatial resolution simulations. The other parameter is the underrelaxation factor α_{reini} (see (1.36)) which is a more tricky parameter than ϵ because it is not directly related to the resolution level but instead to correct the mass loss in the course of simulation. According to the simulations performed in this work the value of this underrelaxation paramter α_{reini} was always constant in the course of the individual simulations, however its value has been found iteratively so to violate mass conservation as minimally as possible. This strategy of tuning the underrelaxation parameter α_{reini} leads also to the consequence that its value would need to be newly determined for each time step value which, due to the fact that the here employed temporal discretization is second order accurate (Crank-Nicholson), was omitted by using sufficiently small timesteps guaranteeing temporal discretization independent results for each spatial resolution simulation. The results are summarized in analogous formats as for the interface tracking framework. The absolute convergence of the benchmark quantities is provided in Table 1.5 and the resulting relative convergence results are compared in Table 1.6 against the reference values obtained by the above described interface tracking framework results. As seen from the mentioned tables the reached spatial convergence here is considerably weaker than experienced for the other numerical framework, but as expected, the results are asymptotically approaching the reference results, too. The clear benefit of these simulations on the one hand is the independence of temporal discretization, at least for the here analyzed sufficiently small timesteps. On the other hand it is necessary to mention that the timestep size is linearly scaling with the spatial discretization size h due to the underlying capillary time step restriction, taking into account the underlying explicit treatment of surface tension (in case of our interface capturing framework). Graphical representations of the obtained results in terms of time evolution of the bubble rise speed and bubble diameter are displayed in Figure 1.10 and bubble sphericity and relative bubble mass in Figure 1.11. As seen from the evolution of the relative bubble mass the interface capturing framework has a much weaker conservation property as its front tracking counterpart.

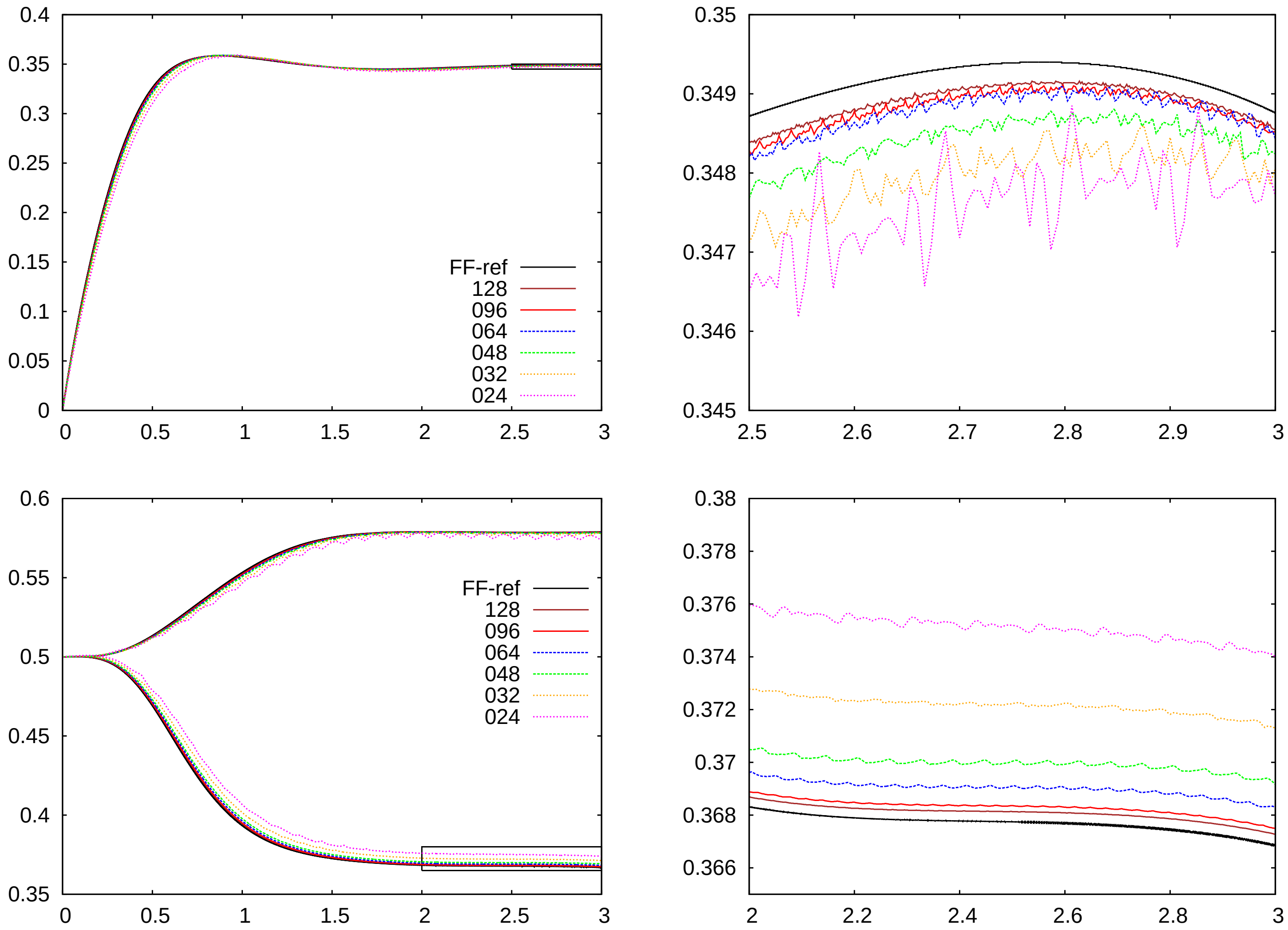

FIGURE 1.10 From top to bottom: Evolution of bubble rise velocity V_c, bubble diameter $2R_z$, $2R_{x,y}$ for the 3D benchmark problem computed by the Level Set approach. The line captions represent the number of elements corresponding to the spatial resolution of the mesh.

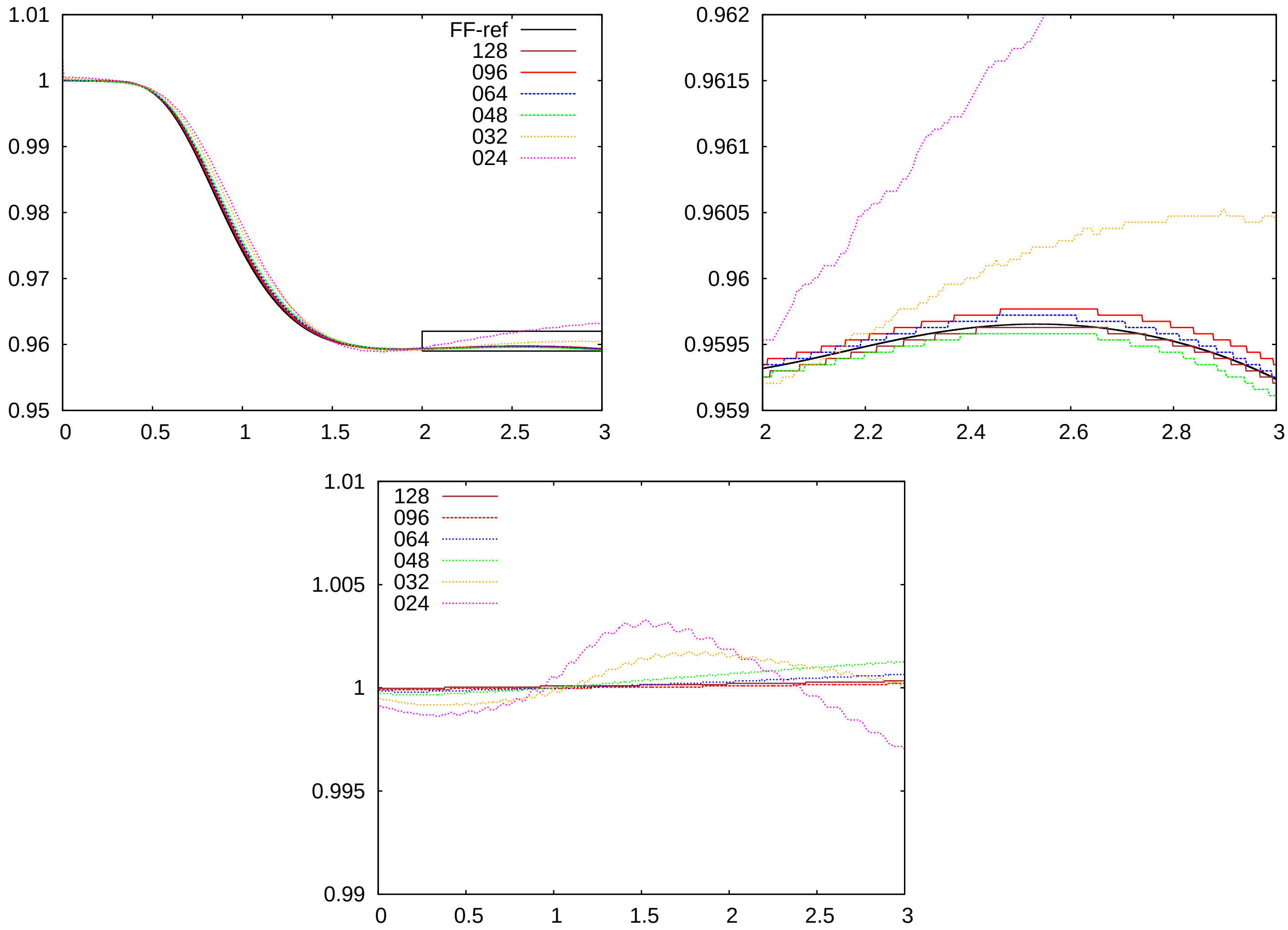

FIGURE 1.11 From top to bottom: Evolution of bubble sphericity A_0/A and relative bubble mass V/V_0 for the 3D benchmark problem computed by the Level Set approach. The line captions represent the number of elements corresponding to the spatial resolution of the mesh.

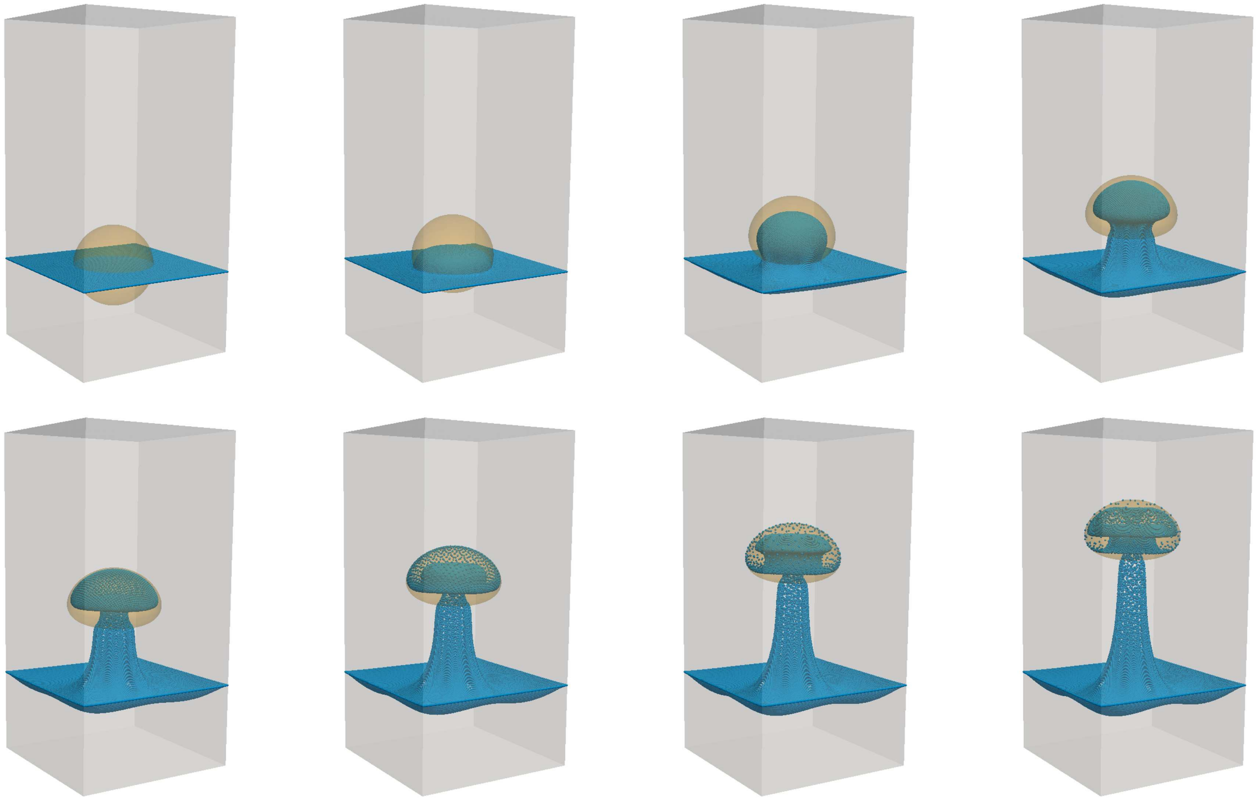

FIGURE 1.12 Particle tracing flow visualization of the 3D rising bubble benchmark.

TABLE 1.5 Absolute convergence of the monitored quantities at final time t=3.0 w.r.t. temporal and spatial refinements - Front capturing framework. Explanation of the symbols is as follows: h - element size, Δt is the used time step, V_0, V are the initial and actual volumes of the bubble, z_c is the z location of the bubble associated with its point of mass. The remaining parameters are introduced in Section 1.7.2.

h	$\frac{\Delta t}{10^3}$	A_0/A	$\frac{V_0-V}{V_0}$%	z_c	V_c	R_z/R_0	$R_{x,y}/R_0$
1/128	1.25	0.97459	0.00714	0.27504	0.35756	0.78918	1.10444
1/96	1.67	0.97480	0.00154	0.27473	0.35781	0.79009	1.10387
1/64	2.50	0.97523	0.00203	0.27387	0.35802	0.79266	1.10254
1/48	3.33	0.97572	0.00062	0.27279	0.35799	0.79555	1.10108
1/32	5.00	0.97706	0.00869	0.27005	0.35827	0.80335	1.09728
1/24	6.67	0.97803	0.05194	0.26642	0.35898	0.81327	1.09310
ref		0.97417	0.00262	0.27585	0.35709	0.78709	1.10618

TABLE 1.6 Relative convergence of the monitored quantities at final time t=3.0 w.r.t. temporal and spatial refinements - Front capturing framework. Explanation of the symbols is as follows: h - element size, Δt is the used time step, V_0, V are the initial and actual volumes of the bubble, z_c is the z location of the bubble associated with its point of mass. The remaining parameters are introduced in Section 1.7.2.

h	$\frac{\Delta t}{10^3}$	A_0/A	z_c	V_c	R_z/R_0	$R_{x,y}/R_0$
1/128	1.25	0.00253	0.13230	0.05597	0.11495	0.02076
1/96	1.67	0.01237	0.18025	0.06827	0.17787	0.05510
1/64	2.50	0.00226	0.30373	0.09548	0.39158	0.05629
1/48	3.33	0.01411	0.45560	0.14855	0.63078	0.17584
1/32	5.00	0.12567	0.83760	0.20051	1.21112	0.21355
1/24	6.67	0.41291	1.41440	0.30198	1.94954	0.82673

Finally, as a demonstration (see Figure 1.12) of the established nonstationary flow patterns during the rise of the bubble, a visualization in terms of particle tracing is displayed at several instances from the beginning ($t = 0.0$) to the end ($t = 3.0$). Due to the chosen planar initial seed the particles are located in the primary and secondary phase at the same time. The particle motion captures the internal recirculation pattern inside of the bubble leading to a mushroom shape deformation of the initially planar structure.

1.7.3 3D Taylor Bubble Benchmark

Taylor bubbles are elongated bubbles being transported in straight and smooth capillaries. The specific feature of such a multiphase flow setup is the presence of a thin liquid film which is squeezed between the capillary wall and the trans-

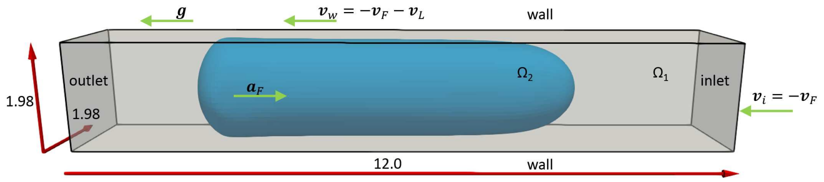

FIGURE 1.13 Geometrical definition of the Taylor bubble benchmark according to the Moving Reference Frame (MFR) technique. $\mathbf{v}_F$ stands for the frame velocity, $\mathbf{v}_L$ stands for the liquid velocity which corresponds to the required flowrate, $\mathbf{v}_w$ and $\mathbf{v}_i$ are the velocity boundary values prescribed at the channel wall and inlet surfaces. $\mathbf{a}_F$ and $\mathbf{g}$ are the accelerations of the reference frame and gravity, respectively.

ported bubble. The hydrodynamics of Taylor bubble flows is governed in steady flow regimes mainly by surface tension and viscous forces, while unsteady flow regimes arise due to the increasing contribution of inertial forces. For this reason, there are introduced two dimensionless quantities which are used to characterize the underlying flow conditions of Taylor bubble flows. These are the capillary number $\mathrm{Ca} = \eta_l U_b/\sigma$ and the Reynolds number $\mathrm{Re} = \rho_l d_h U_b/\eta_l$, where ρ_l and η_l stand for the liquid density and viscosity, U_b is the bubble velocity, d_h is the hydraulic diameter of the channel and σ is the surface tension coefficient. In this section we present numerical results for a Taylor bubble benchmark which was introduced in the work of Marschall *et al.* [34] not only numerically, but also experimentally. The experimentally measured system [34] consisting of an aqueous solution of glycerol - constituting the liquid phase - and air - representing the gas phase - was operated at such conditions that the Capillary and Reynolds numbers were as follows: $\mathrm{Ca} = 0.088$ and $\mathrm{Re} = 17.0$. The square cross section of the used capillary was 1.98 mm × 1.98 mm and the experimentally measured bubble volume was determined as $V_b = 17.5\ \mathrm{mm}^3$. The corresponding physical properties of the respective phases were as follows: $\rho_l = 1195.6\ \mathrm{kg/m}^3$, $\rho_g = 1.3\ \mathrm{kg/m}^3$ and $\eta_l = 28.54\ m\mathrm{Pa}\cdot\mathrm{s}$, $\eta_l = 20.0\cdot 10^{-3}\ m\mathrm{Pa}\cdot\mathrm{s}$. The realization of the Taylor bubble flow problem was performed in a cocurrent flow (see Figure 1.13) so that the bubble rise velocity was experimentally determined to be $205.57 \pm 0.82\ \mathrm{mm/s}$. The reference work of this Taylor bubble flow setup [34] has described several numerical frameworks which have have been used for the computational validation covering both front capturing (VOF, Level Set) and front tracking techniques. All CFD simulation results have qualitatively well captured the experimental reference results, but since they showed out small variations with respect to each other and the experimental results, unfortunately no benchmark values could have been determined. The differences between the results of the different software packages can mainly be attributed to the different realizations of the same problem by imposing periodic boundary conditions (with pressure drop), moving window techniques and the moving reference frame (MFR) technique. Our numerical realization follows the MFR technique used by the FS3D and OpenFoam software packages. Another special feature of this computational problem is related to the fact that it corresponds to a steady state solution which is to be reached by an unsteady simulational approach. Considering that some numerical schemes are either not fully conservative or are adjusted with mass correction mechanisms, makes this computational problem very challenging, especially in the framework of a mesh convergence study. According to the realization presented here, a coarse-level resolution simulation has been started by prescribing the required volume of the bubble. After reaching steady state (relative changes in the bubble surface area decreased below the prescribed tolerance) the coarse-level resolution result has been prolongated to a finer resolution level and used as a start-solution for the subsequent finer resolution simulation (see Figure 1.14). Finally, the same procedure has been performed for the finest-level resolution simulation, as well.

Due to the fact that in the course of the prolongation of the solution from coarser to finer level, the volume of the bubble has changed considerably, independent simulations have been performed for the lower resolution levels (level 2 and 3) for such an initial bubble volume which has been reached by the finest level solution (level 4) before (17.49 mm^3). This way, the comparison of the monitored quantities is related to the same bubble volume basis. The monitored quantities have been determined from each resolution-level simulation result and are organized into Table 1.7 together with the available other published results [34]. These quantities are the bubble length, transportation velocity and the minimum film thickness in the longitudinal and diagonal directions. Other important characteristics (see Table 1.8) of the underlying simulations are related to the integral quantities like the bubble volume conservation and the relative surface area deviations measured on the different resolution levels. According to the results listed in Table 1.8 it can be seen that the coarsest level resolution results are influenced by a relatively large mass loss (0.025% over the 10s simulation time) which also leads to uncertainities with regards to the steady state (influenced by permanent mass loss), however these effects of mass loss are massively cured by the higher resolution level and are completely vanishing for the finest resolution level. For this reason we believe that it is justified to speak about mesh convergence only in case of the two higher level resolution results and the overall high order convergence is to be expected to happen on a potentially next resolution level which on the one hand would be already so much CPU intensive that it did not met the aims of this publication. On the other hand the results presented here are already in the range of standard benchmark precision requirements. Additionally, in Figure 1.15 we provide the comparisons of the obtained bubble shapes in the two considered planes against the experimentally measured reference curves and a vizualization of the arising recirculation structures in terms of streamlines is demonstrated in Figure 1.16. Additional remark to the performed simulations is that all three resolution level simulations have been performed with the same 0.1 ms timestep resulting in instationary simulations requiring about $\approx 10,000$ timesteps and covering an ≈ 1 s real time simulation in order to reach steady state.

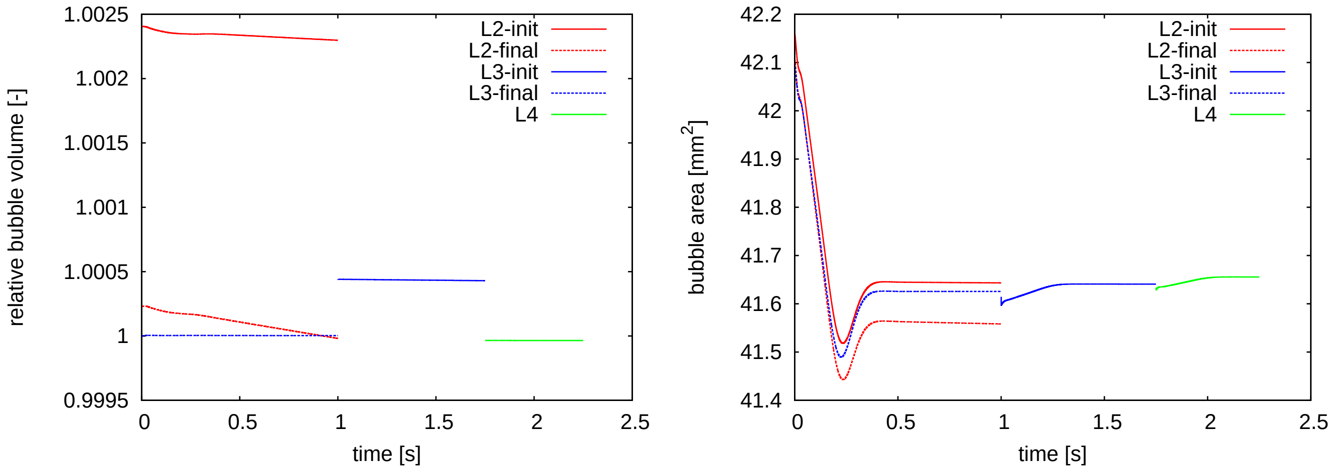

FIGURE 1.14 Mass conservation temporal evolution of the Taylor bubble simulations performed on different resolution levels (left). Time evolution of the Taylor-bubble surface area on different resolution levels (right). Solid lines (init) represent the initial sequence of simulations including solution prolongation from coarser to finer, dashed lines (final) stand for the restarted simulations with the updated initial bubble volume (17.49 mm^3).

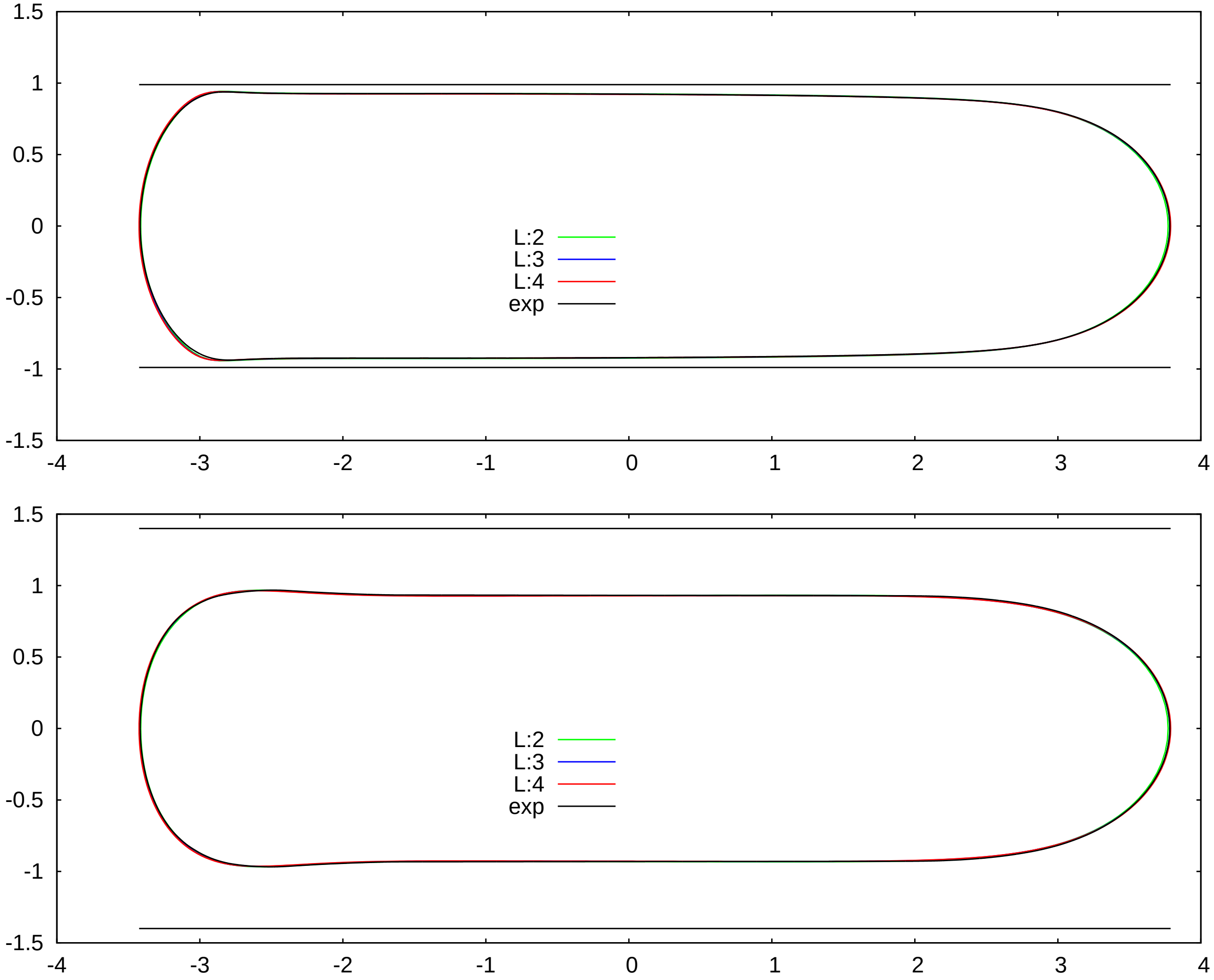

FIGURE 1.15 Spatial resolution convergence of the Taylor bubble shape in lateral (top) and diagonal (bottom) slices in comparison with the experimental results published by Marschall *et al.* [34]. Solid black line represents the wall of the capillary.

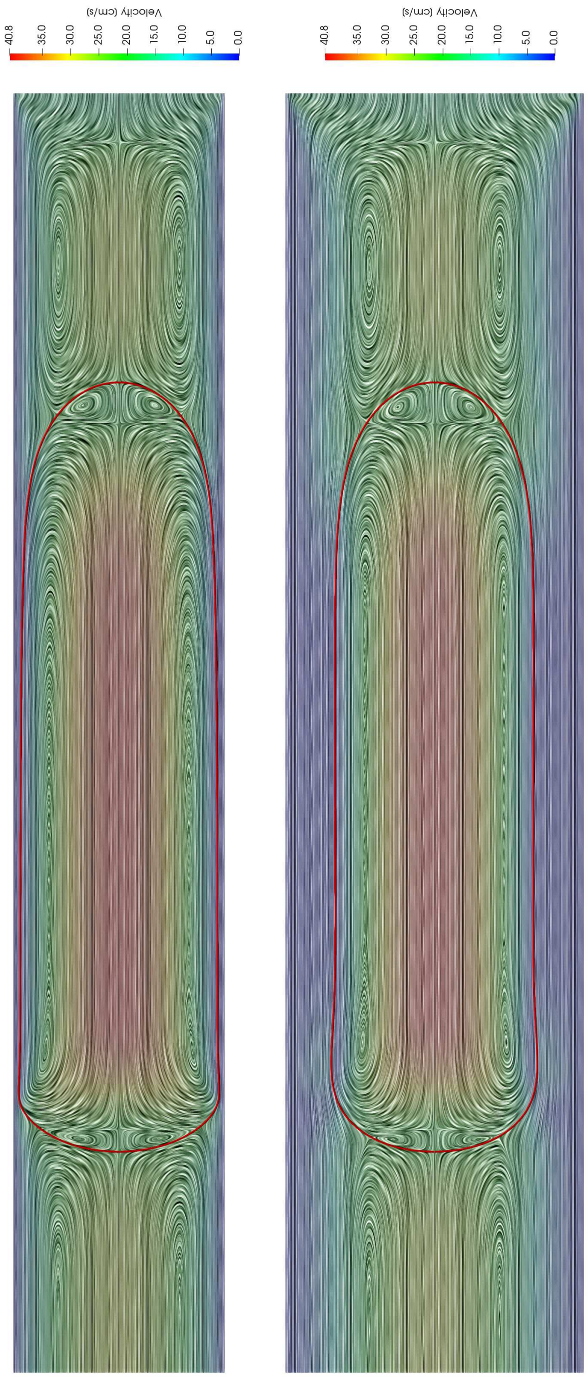

FIGURE 1.16 Visualization of the recirculation vortices for the Taylor bubble benchmark by means of a streamline rendering (Surface LIC).

TABLE 1.7 Mesh convergence of the Taylor bubble benchmark with respect to other available references. The benchmark quantities listed in the table are as follows: L - Bubble length, Φ_d - Minimum flim thickness in the diagonal direcion, Φ_l - Minimum flim thickness in the longitudinal direcion and v - bubble rise velocity.

		DROPS	FS3D	TURBIT	OpenFoam	FF-L2	FF-L3	FF-L4	EXP
L	[mm]	7.230	7.197	7.110	7.202	7.185	7.204	7.214	7.200
Φ_d	[μm]	439.2	436.2	442.0	456.6	431.6	434.4	435.4	433.1
Φ_l	[μm]	49.0	47.7	28.0	59.0	46.8	49.0	49.7	50.5
v	[cm/s]	20.69	19.75	20.78	20.58	19.17	19.78	20.07	20.56

1.7.4 Reactive multiphase flow

Multiphase flow problems are usually associated with additional transport phenomena (e.g. transport of species) which are however strongly dependent on the hydrodynamics of the involved phases. Therefore, the accurate prediction of fluid motion becomes the prerequisite for the accurate simulation of subsequent transport problems. In this section we provide simulation results of a reactive Taylor-bubble flow with the aim of reaction parameter estimation on the basis of experimentally measured results. All the here referenced experimental measurements have been carried out by the chair of Laboratory of Equipment Design, Department of Biochemical and Chemical Engineering at the TU Dortmund University. The analyzed chemical reaction system is related to a two-step indigo-carmine redox-reaction which from a reaction engineering point of view refers to a consecutive-competitive reaction scheme:

$$\mathrm{A} + \mathrm{B} \longrightarrow \mathrm{C} \tag{1.37}$$

$$\mathrm{A} + \mathrm{C} \longrightarrow \mathrm{D} \tag{1.38}$$

From the industrial point of view, such reaction schemes are extremely important to be understood, since the optimal realization of underlying chemical processes makes it possible to increase the overall yield or selectivity of the key-species which may be especially challenging in case of C being the wanted reaction product and D the unwanted one. Due to the following eminent properties of microcapillaries

- high interfacial area
- high mass transfer coefficients
- possibilities of precise control and manipulation of flows

are these realization units very promising for such sensitive reaction systems. Therefore, it is inevitable to understand the effect of the arising Taylor vortices and the interplay of diffusion and convection on the performance of the underlying chemical reaction system.

TABLE 1.8 Mesh convergence of the Taylor bubble benchmark with respect to mass conservation and final surface area. Specific mass loss stands for an asymptotical mass loss of the bubble at (or close to) steady state per 100 ms real time, surface area is the bubble surface area at (or close to) steady state, and the relative surface area relates the difference of coarser/finer level solution bubble surface area to the finest bubble surface area.

quantity	unit	FF-L2	FF-L3	FF-L4
specific mass loss	$[\%/100\ ms]$	0.0025	0.00001	$\ll$0.00001
surface area	$[\mathrm{mm}^2]$	41.5602	41.6256	41.6556
relative surface area	$[\%]$	0.229	0.072	-

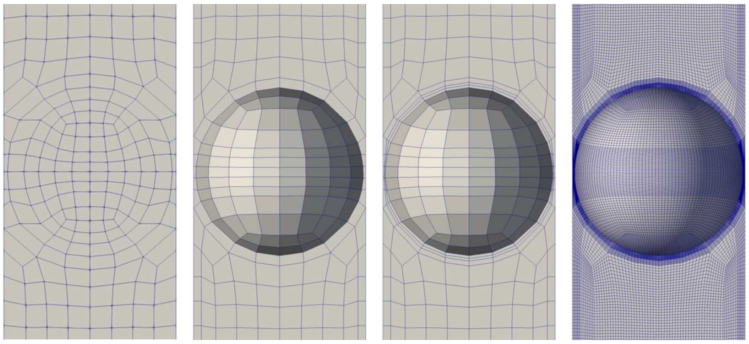

FIGURE 1.17 Graphical representation of the computational meshes. From left to right: Coarse level resolution mesh used for the CFD simulation, the coarse level resolution mesh without the bubble phase, recursively extended boundary-layer coarse mesh, the level 4 resolution fine mesh used for the transport of species problem with chemical reactions.

In the current work we have computationally analyzed the oxidation of leuco-indigo carmine (B) to an anionic radical intermediate (C) which is further reduced to leuco-indigo carmine (D). The injected aqueous solution of (B) with the initial concentration of $C_{B,t=0} = 0.30\ m\mathrm{mol} \cdot \mathrm{L}^{-1}$ is contacted with a periodically generated sequence of air bubbles having a volume of $V = 7.62\ \mathrm{mm}^3$ which are transported in a circular cross-section capillary of diameter $D = 1.6$ mm. The experimentally imposed target flowrates of the individual phases

- $\dot{V}_g = 0.8\ m\mathrm{L} \cdot \mathrm{min}^{-1}$ for the gas phase,
- $\dot{V}_l = 3.2\ m\mathrm{L} \cdot \mathrm{min}^{-1}$ for the liquid phase

were achieved in the simulation by prescribing the corresponding a) initial volumetric ratio of the phases in a circular crosssectional domain of length being equivalent to a (periodic) bubble-to-bubble distance from the experimental measurements $L = 3.32$ mm and b) an iteratively found pressure jump condition resulting in the required flowrate values. The corresponding simulation setup has

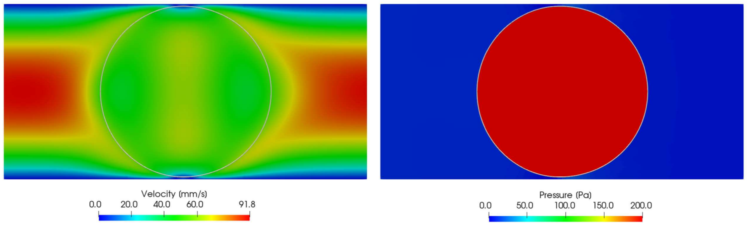

FIGURE 1.18 Visualization of computational results obtained for the reactive bubble problem. On the left side is the velocity field and on the right is the pressure distribution.

been realized in a framework of periodic boundary conditions for velocity, pressure and species concentrations, however no jump condition has been used for the species concentration quantities which is responsible for a small deviation with respect to the real experimental setup. The computational setup reflects a system of an infinite sequence of bubbles separated by mutually connected liquid slugs (due to the thin liquid film between the bubble and the capillary), which initially (at $t = 0$) have a spatially constant initial concentration $C_{A,t=0} = 0.0\ m\text{mol} \cdot \text{L}^{-1}$, $C_{B,t=0} = 0.3\ m\text{mol} \cdot \text{L}^{-1}$, $C_{C,t=0} = 0.0\ m\text{mol} \cdot \text{L}^{-1}$, $C_{D,t=0} = 0.0\ m\text{mol} \cdot \text{L}^{-1}$ of all species and are assigned to a steady (and therefore constant) velocity field. This steady state flow field is computed separately in a framework of a decoupled multiphase flow simulation before performing the corresponding simulations of the species transport problem. According to this simplification the time axis of the simulations may be interpreted as the spatial axis along which a representative bubble is transported in the real experimental setup. Another simplification introduced into the model is related to the assumption of constant material properties which however is strongly justified due to the fact that the chemical species are present at only very low concentrations and therefore have a negligible influence on the otherwise constant material properties. The material properties are listed in Table 1.9 which provides the viscosities μ, densities ρ of the present g/l phases, the interfacial tension σ and diffusion coefficients D_i [31] of the chemical species in the liquid phase. The presence of the gas phase was not considered in the transport equation of species (only in the preceding multiphase flow simulation), instead a Dirichlet boundary condition according to Henry law has been used for the Oxygen species concentration and zero flux for all other species. The value of this equilibrium concentration according to the experimental measurements was set to $C_{A,\Gamma} = 0.25\ m\text{mol} \cdot \text{L}^{-1}$.

The resulting spatially converged flow field and pressure distribution of the flow simulations is demonstrated in Figure 1.18 which has been obtained in an analogous way as in the previously described Taylor bubble benchmark. In

TABLE 1.9 The set of physical properties used for the Reactive Taylor Bubble problem.

ρ_l $[\mathrm{kg} \cdot \mathrm{m}^{-3}]$	ρ_g $[\mathrm{kg} \cdot \mathrm{m}^{-3}]$	μ_l $[m\mathrm{Pa} \cdot \mathrm{s}]$	μ_g $[m\mathrm{Pa} \cdot \mathrm{s}]$	σ $[m\mathrm{N} \cdot \mathrm{m}^{-1}]$	$D_{A/l}$ $[\mathrm{m}^2 \cdot \mathrm{s}^{-1}]$	$D_{B,C,D/l}$ $[\mathrm{m}^2 \cdot \mathrm{s}^{-1}]$
1000	1.2	1.12	0.018	75.0	$2.0 \cdot 10^{-9}$	$0.6 \cdot 10^{-9}$

particular, first a lower level solution was obtained which after prolongation was then used as an initial condition of a finer level simulation. The iteratively found pressure drop resulting in the required flowrates has been estimated as $\Delta P = 3.3$ Pa. The such obtained velocity distribution was used in the subsequent species transport simulations. Since the resolution requirements dictated by the Schmidt number of the underlying scalar transport problems were relatively high $Sc = \nu_l/D_A = 330$ (being $\nu_l = \mu_l/\rho_l$ the kinematic viscosity of the liquid phase) additional mesh manipulation procedures related to boundary layer refinement have been implemented. According to the mentioned technique the coarse mesh extracted from the fluid simulations has been extended by successively refined boundary layers of elements in the normal direction to the bubble surface. The resulting extended coarse mesh representations are displayed in Figure 1.17. The such obtained coarse mesh has been refined in the framework of standard geometrical multigrid techniques to resolution levels 4 and 5, respectively, so to ensure mesh convergence of the obtained simulation results. The velocity field obtained from the multiphase flow simulation has been interpolated via an L_2 projection onto the fine resolution meshes used by the species transport computations. The conservation properties of the arising discrete convection operators were fulfilled by the use of an Algebraic Flux Correction (AFC) schemes (see [32] for details) which is a key ingredient providing high-resolution properties to the employed FEM discretization scheme for the species transport problems. The most valuable properties of the AFC scheme are related to

- stabilized (oscillation-free) transport of scalar quantities (exhibiting steep gradients) by the use of the smallest amount of numerical diffusion, which is determined in the course of a non-linear iteration loop,
- the local extrema diminishing properties according to which maxima and minima do not grow/sink due to transportation in space.

For the transport in the space of species are responsible the reaction terms which are also treated in an operator splitting approach. This means, after the transportation step in space, systems of Ordinary Differential Equations (ODE) are to be solved for each degree of freedom independently from each other, which allows for using a substepping technique even for stiff reaction systems. Due to robustness requirements the ODE solver of John Burkardt has been integrated into the here used solver, which is a Fortran 90 adaptation of the original work of Shampine and Gordon [48]. The arising system of ODEs reads

$$\frac{d\,c_A}{d\,t} = -r_1 - r_2 \tag{1.39}$$

$$\frac{d\,c_B}{d\,t} = -r_1 \tag{1.40}$$

$$\frac{d\,c_C}{d\,t} = +r_1 - r_2 \tag{1.41}$$

$$\frac{d\,c_D}{d\,t} = +r_2 \tag{1.42}$$

where r_1 and r_2 are the reaction rates of the first and second chemical reaction, respectively. These reaction rates are modelled as second order reactions, as follows

$$r_1 = k_1\, c_A\, c_B \tag{1.43}$$

$$r_2 = k_2\, c_A\, c_C \tag{1.44}$$

where the constants k_1 and k_2 are the reaction rate constants of the first and second reaction, respectively. The values of these constants were unknown and had to be determined by means of a pararameter optimization process to provide the best agreement with the experimentally measured concentration results.

The species transport simulations have been performed with a $1/1600$ s timestep size covering a total 11.0 s time interval, from which three time level outputs ($t = [3.3, 6.6, 11.0]$ s) have been used for comparisons with the corresponding experimental results. The experimentally recorded pictures have been transformed to concentration distributions, which however were related to concentration intensities averaged through the cylindrical fluid volume and therefore the CFD results had to be transformed into the same representations via an additional postprocessing step by integrating the concentration distribution of the three monitored species (B, C, D) through the corresponding fluid volume. The such obtained 2D representations have been used for the overall parameter optimization process. In order to prevent a too detailed description of the optimisation procedure we discuss here only the results for the found parameter values providing the closest match with the experimental references. The value of the estimated reaction constants were $k_1 = k_2 = 8 \cdot 10^5$ L mol s^{-1}, which are matching reasonably with the experimentally expected values of Krieger *et al.* [31] ($k_1 = 6.3 \cdot 10^5$ L mol s^{-1} and $k_2 = 22.4 \cdot 10^5$ L mol s^{-1}). The spatial distribution of the species at time $t = 6.6$ s for the used two resolution levels with the close up views in the vicinity of the bubble interface are demonstrated in Figure 1.19. Here, it is visible that the steep concentration gradients are well resolved already at the coarser resolution what ensures the spatial convergence of the obtained results on the finest resolution level. The graphical comparison of the results with the experimental reference results is provided in Figure 1.20. This comparison demonstrates the found match between the dynamics of the

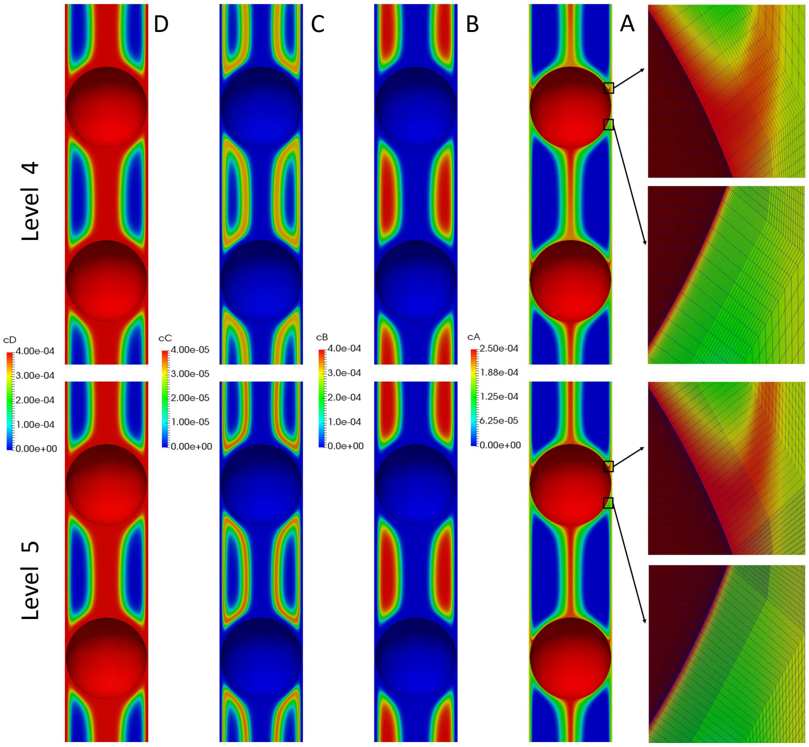

FIGURE 1.19 Graphical representation of the computational convergence between refinement Level 4 and Level 5.

simulated process and the experimentally measured one, since the concentration distributions are matching each other at all three time levels. In particular, the prediction of the dynamics of the intermediate species *C* is the most crucial since the concentration values of this species is the lowest and has the most significant spatial distribution forming a ring shape of a thickness and radial location which is possible to predict only with the right prediction of the overall reactive transport phenomena. The right interpretation of the comparison between the two sets of results requires a few comments concerning the experimental reference values, namely that the pair of the experimentally investigated bubbles were not the same, instead the camera position was fixed every time to a different position so to record the bubbles and slugs at the different time levels. Therefore, the centerline distance between the bubbles was not always the same but just close to the target value. The other necessary comment is that the extraction of concentration signals in the vicinity of the bubble surface and at the wall of the capillary was affected with a larger experimental error due to light reflection effects, this is also why the differences look to be that large in these regions between the two sets of results. The regions belonging to the slug core are affected with a much lower measurement error and therefore these are the regions based on which the parameter optimization was performed.

1.8 CONCLUSIONS

In this contribution, we have provided a compact description of state-of-the-art numerical solvers for multiphase flow problems, namely interface tracking and interface capturing methods. Corresponding discretization and solution approaches which are based on Finite Element and Discrete Projection methods for the Navier-Stokes equations, combined with corresponding numerical tools for the interface capturing, resp., tracking approaches, lead to robust, accurate, flexible and efficient simulation tools. Moreover, we have presented several numerical test cases of benchmarking type which first of all help to evaluate the quality of the underlying (single-phase) flow solvers. Subsequently, we have described the settings for a quantitative 3D Rising Bubble benchmark which can be used for 'simple' validation and evaluation of multiphase CFD codes without the necessity of complex postprocessing operations. Finally, we also provided numerical reference values for a 'Taylor bubble' setting, and we showed simulation results of a reactive Taylor bubble flow in the framework of estimating reaction parameters to match corresponding experimentally measured results. All reference benchmark quantities can be downloaded from `www.featflow.de`.

ACKNOWLEDGMENTS

The financial support of DFG (SPP 1740) is gratefully acknowledged (TU 102/53-1). Additional thank is to be expressed to the group of Prof. Norbert Kockmann (particulary to M.Sc. Waldemar Krieger) for supporting this work with experimental measurements for the reactive bubble problem and also to Dr. Evren Bayraktar for providing the interface between the computational and experimental work. We would also like to thank Dr. Kathrin Bäumler for the provided support during the endless validation procedures. The computations have been carried out on the LiDO cluster at TU Dortmund University. We would like to thank the LiDO cluster team for their help and support.

BIBLIOGRAPHY

[1] Adelsberger J., Esser P., Griebel M., Groß S., Klitz M., Rüttgers A., 3D incompressible two-phase flow benchmark computations for rising droplets. *Proceedings of the 11th World Congress on Computational Mechanics (WCCM XI)*, Barcelona, Spain, 2014.

[2] Annaland M. S., Dijkhuizen W., Deen N. G., Kuipers J. A. M., Numerical simulation of behavior of gas bubbles using a 3-D front-tracking method. *AIChE Journal* 2006, **52**(1):99–110, DOI: 10.1002/aic.10607.

[3] Barth T., Sethian J. A., Numerical Schemes for the Hamilton-Jacobi and level set equations on triangulated domains. *Journal of Computational Physics* 1999, **145**:1–40.

[4] Basting S., *An interface fitted finite element method for multiphysics simulations.* PhD Thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, Faculty of Sciences, Nürnberg, 2016.

[5] Bayraktar E., Mierka O., Turek S., Benchmark computations of 3D laminar flow around a cylinder with CFX, OpenFOAM and FeatFlow, *International Journal of Computational Science and Engineering* 2012, **7**(3):253–266, DOI: 10.1504/IJCSE.2012.048245.

[6] Bänsch E., Numerical methods for the instationary Navier-Stokes equations with a free capillary surface. Habilitation Thesis, Universität Freiburg, 1998.

[7] Bänsch E., Finite element discretization of the Navier-Stokes equations with a free capillary surface. *Numerische Mathematik* 2001, **88**(2):203–235, DOI: 10.1007/s002110000225.

[8] Bäumler K., *Simulation of single drops with variable interfacial tension.* PhD Thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, Faculty of Sciences, Erlangen-Nürnberg, 2014.

[9] Brackbill J. U., Khote D. B., Zemach C., A Continuum Method for Modeling Surface Tension. *Journal of Computational Physics* 1992, **100**:335–354.

[10] Chen T., Minev P. D., Nandakumar K., A projection scheme for incompressible multiphase flow using adaptive Eulerian grid. *International Journal for Numerical Methods in Fluids* 2001, **45**(1):1–19, DOI: 10.1002/fld.591.

[11] Croce R., Griebel M., Schweitzer M. A., Numerical simulation of bubble and droplet deformation by a level set approach with surface tension in three dimensions. *International Journal for Numerical Methods in Fluids* 2009, **62**:963–993, DOI: 10.1002/fld.2051.

[12] Damanik H., *Monolithic FEM techniques for viscoelastic fluids.* PhD Thesis, TU Dortmund, Institute of Applied Mathematics (LS III), Dortmund, 2011.

[13] Di Pietro D. A., Lo Forte S., Parolini N., Mass preserving finite element implementations of the level set method. *Applied Numerical Mathematics* 2006, **56**(9):1179–1195, DOI: 10.1016/j.apnum.2006.03.003.

[14] Donea J., Giuliani S., Laval H., Quartapelle L., Finite element solution of the unsteady Navier-Stokes equations by a fractional step method. *Comput. Methods Appl. Mech. Engrg.* 1982, **30**:53–73.

[15] Engelman, M. S., Sani L., Gresho P.M., The implementation of normal and/or tangential boundary conditions in Finite Element codes for incompressible fluid flow, *International Journal for Numerical Methods in Fluids* 1982, **2**:225–238.

[16] Farthing M. W., Kees C. E., Implementation of Discontinuous Galerkin Methods for the Level Set Equation on Unstructured Meshes. ERDC/CHL CHETN-XII-2, 2008.

[17] Frolkovič P., Logashenko D., Wittum G., Towards Finite Volume Discretization of Two-Phase Flow Using Level-Set Formulation. *SiT-Report* (SiT, University of Heidelberg), 15/12/2007.

[18] Ganesan S., *Finite Element Methods on Moving Meshes for Free Surface and Interface Flows.* PhD Thesis, Otto-von-Guericke-Universität, Fakultät für Mathematik, Magdeburg, 2006, published as book by docupoint Verlag Magdeburg, ISBN 3-939665-06-1.

[19] Ganesan S., Tobiska L., Computations on flows with interfaces using arbitrary Langrangian Eulerian method. *European Conference on Computational Fluid Synamics* 2006, ECCOMAS CFD.

[20] Girault V., Raviart P. A., *Finite Element Methods for Navier-Stokes equations.* Springer-Verlag, 1986.

[21] Glimm J., Grove J. W., Li X. L., Shyue K.-M., Zeng Y., Zhang Q., Three-Dimensional Front Tracking. *SIAM Journal on Scientific Computing* 1998, **19**:703–727, ISSN:1064-8275.

[22] Gresho P. M., Chan S. T., Lee R. L., Upson C. D., A modified finite element method for solving the time-dependent, incompressible Navier-Stokes equations. Part 1: Theory, *International Journal for Numerical Methods in Fluids* 1983, **4**(6):557–598.

[23] Groß S., Reichelt V., Reusken A., A Finite Element Based Level Set Method for Two-Phase

Incompressible Flows. *Comp. Visual. Sci.* 2006, **9**:239–257.

[24] Hansbo A., Hansbo P., An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Computer Methods in Applied Mechanics and Engineering* 2002, **191**:5537–5552.

[25] Houzeaux G., Codina R., A Chimera method based on a Dirichlet/Neumann (Robin) coupling for the Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 2003, **192**:3343–3377.

[26] Hysing S., Turek S., The Eikonal equation: numerical efficiency vs. algorithmic complexity on quadrilateral grids. *Proceedings of Algoritmy* 2005, Slovak University of Bratislava, Bratislava, ISBN 80-227-2192-1, 22–31.

[27] Hysing S., A new implicit surface tension implementation for interfacial flows. *International Journal for Numerical Methods in Fluids* 2006, **51**(6):659–672, DOI: 10.1002/fld.1147.

[28] Hysing S., *Numerical Simulation of Immiscible Fluids with FEM Level Set Techniques.* PhD Thesis, University of Dortmund, Institute of Applied Mathematics (LS III), Dortmund, 2007.

[29] Hysing, S., Turek, S., Kuzmin, D., Parolini, N., Burman, E., Ganesan, S., Tobiska, L., Quantitative benchmark computations of two-dimensional bubble dynamics. *International Journal for Numerical Methods in Fluids* 2009, **60**(11):1259–1288, DOI: 10.1002/fld.1934.

[30] John V., On the efficiency of linearization schemes and coupled multigrid methods in the simulation of a 3D flow around a cylinder. *International Journal of Numerical Methods in Fluids* 2005, **50**-(7):845–862, DOI: 10.1002/fld.1080.

[31] Krieger W., Hörbelt M., Schuster S., Hennekes J., Kockmann N., Kinetic Study of Leuco-Indigo Carmine Oxidation and Investigation of Taylor and Dean Flow Superposition in a Coiled Flow Inverter. *Chemical Engineering Technology* 2019, **42**(10):1–10, DOI: 10.1002/ceat.201800753.

[32] Kuzmin D., Turek S., High-resolution FEM-TVD schemes based on a fully multidimensional flux limiter. *Journal of Computational Physics* 2004, **198**(1):131–158, DOI: 10.1016/j.jcp.2004.01.015.

[33] Kuzmin, D., Mierka, O., Turek, S., On the implementation of the k–epsilon turbulence model in incompressible flow solvers based on a finite element discretization, *International Journal of Computing Science and Mathematics* 2008, **1**:193–206, DOI: 10.1504/IJCSM.2007.016531.

[34] Marschall H., Boden S., Lehrenfeld Ch., Falconi C. J., Hampel U., Reusken A., Wörner M., Bothe D., Validation of Interface Capturing and Tracking techniques with different surface tension treatments against a Taylor bubble benchmark problem. *Computers and Fluids* 2014, **102**:336–352.

[35] Minev P. D., Chen T., Nandakumar K., A finite element technique for multifluid incompressible flow using Eulerian grids. *Journal of Computational Physics* 2003, **187**:255–273.

[36] Nagrath S., Jansen K. E., Lahey R. T., Computation of incompressible bubble dynamics with a stabilized finite element level set method. *Comput. Methods Appl. Mech. Engrg.* 2005, **194**:4565–4587.

[37] Nguyen V.-T., Peraire J., Khoo B. C., Persson P.-O., A discontinuous Galerkin front tracking method for two-phase flows with surface tension. *Computers and Fluids* 2010, **39**:1–14, DOI: 10.1016/j.compfluid.2009.06.007.

[38] Nichols B. D., Hirt C. W., Methods for calculating multidimensional, transient free surface flows past bodies. *Proc. First Intern. Conf. Num. Ship Hydrodynamics* 1975, Gaithersburg, ML.

[39] Osher, S. J., Fedkiw, R. P., *Level Set Methods and Dynamic Implicit Surfaces.* Springer-Verlag, 2002, ISBN 0-387-95482-1.

[40] Osher S., Sethian J. A., Fronts propagating with curvature-dependent speed: Algorithms based

on Hamilton-Jacobi formulations. *Journal of Computational Physics* 1988, **79**(1):12–49, DOI: 10.1016/0021-9991(88)90002-2.

[41] Ouazzi, A., *Finite element simulation of nonlinear fluids with application to granular material and powder.*, PhD Thesis, Universität Dortmund, 2005.

[42] Owen H. C., *A Finite Element Model for Free Surface and Two Fluid Flows on Fixed Meshes.* PhD Thesis, 2009, Barcelona.

[43] Parolini N., *Computational Fluid Dynamics for Naval Engineering Problems.* PhD Thesis no. 3138, École Polytechnique Fédérale de Lausanne (EPFL), 2004.

[44] Ramaswamy B., Kawahara M., Arbitrary Lagrangian-Eulerianc finite element method for unsteady, convective, incompressible viscous free surface fluid flow. *International Journal for Numerical Methods in Fluids* 2005, **7**:1053–1075.

[45] Rumpf M., Preußer T., A Level Set Method for Anisotropic Geometric Diffusion in 3D Image Processing. *SIAM Journal of Applied Mathematics* 2002, **62**:1772–1793.

[46] Quecedo M., Pastor M., Application of the level set method to the finite element solution of two-phase flows. *Int. J. Numer. Methods Eng.* 2001, **50**(3):645–663.

[47] Scardovelli R., Zaleski S., Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics* 1999, **31**:567–603, DOI: 10.1146/annurev.fluid.31.1.567.

[48] Shampine L., Gordon M., *Computer Solution of Ordinary Differential Equations: The Initial Value Problem*, Freeman, 1975, ISBN: 0716704617.

[49] Sethian J. A., *Level Set Methods and Fast Marching Methods: Evolving Interphases in Computational Geometry, Fluid Mechanics, Computer Vision and Material Science*, Cambridge University Press, 1999.

[50] Smolianski A., Finite-element/level-set/operator-splitting (FELSOS) approach for computing two-fluid unsteady flows with free moving interfaces. *Int. J. Numer. Methods Fluids* 2005, **48**(3):231–269.

[51] Smolianski A., *Numerical Modeling of Two-Fluid Interfacial Flows.* PhD dissertation, University of Jyvaskyla, 2001, Jyvaskyla Studies in Computing 8, ISBN 951-39-0929-8.

[52] Tornberg A.-K., Interface Tracking Methods with Applications to Multiphase Flows. PhD Thesis, Stockholm, 2000, ISBN 91-7170-558-9.

[53] Turek S., On discrete projection methods for the incompressible Navier-Stokes equations: an algorithmical approach. *Computer Methods in Applied Mechanics and Engineering* 1997, **143**(3-4):271–288, DOI: 10.1016/S0045-7825(96)01155-3.

[54] Turek S., Efficient Solvers for Incompressible Flow Problems, An Algorithmic and Computational Approach. *Lecture Notes in Computational Science and Engineering* 1999, **6** Springer-Verlag.

[55] Turek, S., Ouazzi, A., Unified edge-oriented stabilization of nonconforming FEM for incompressible flow problems: Numerical investigations. *Journal of Numerical Mathetmatics* 2007, **15**(4):299–322.

[56] Turek, S., Mierka, O., Hysing, S., Kuzmin, D., Numerical study of a high order 3D FEM–Level Set approach for immiscible flow simulation, Appeared in: Repin, S., Tiihonen, T., Tuovinen, T., Numerical methods for differential equations, optimization, and technological problems, *Computational Methods in Applied Sciences* 2013, **27**:65–91, ISBN 978-94-007-5287-0.

[57] Turek, S., Mierka, O., Bäumler K., Numerical Benchmarking for 3D Multiphase Flow: New Results for a Rising Bubble, Appeared in: Radu, F., Kumar, K., Berre, I., Nordbotten, J., Pop, I., Lecture Notes in Computational Science and Engineering, *Numerical Mathematics and Advanced Applications* 2019, Springer, **27** pp. 593–601, ISBN: 978-3-319-96414-0.

[58] Turek S., Schäfer M., Benchmark computations of laminar flow around cylinder. *Flow Simulation with High–Performance Computers II (Notes on Numerical Fluid Mechanics)* 1996,

52:547–566.

[59] Turek, S., Rivkind, L., Hron, J., Glowinski, R., Numerical study of a modified time-stepping theta-scheme for incompressible flow simulations. *Journal of Scientific Computing* 2006, **28**(2–3):533–547, DOI: 10.1007/s10915-006-9083-y.

[60] Unverdi S. O., Tryggvason G., A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics* 1992, **100**(1):25–37, DOI: 10.1016/0021-9991(92)90307-K.

[61] Veneziani A., Villa U., ALADINS: an ALgebraic splitting time ADaptive solver for the Incompressible Navier-Stokes equations. Part 1: Basic settings and analysis. Technical Report: TR-2011-010, 2011.

[62] Welch J. E., Harlow F. H., Shannon J. P., Daly B. J., The MAC method: a computing technique for solving viscous, incompressible, transient fluid-flow problems involving free surfaces. *Los Alamos Scientific Laboratory Rep.* 1965, LA-3425.

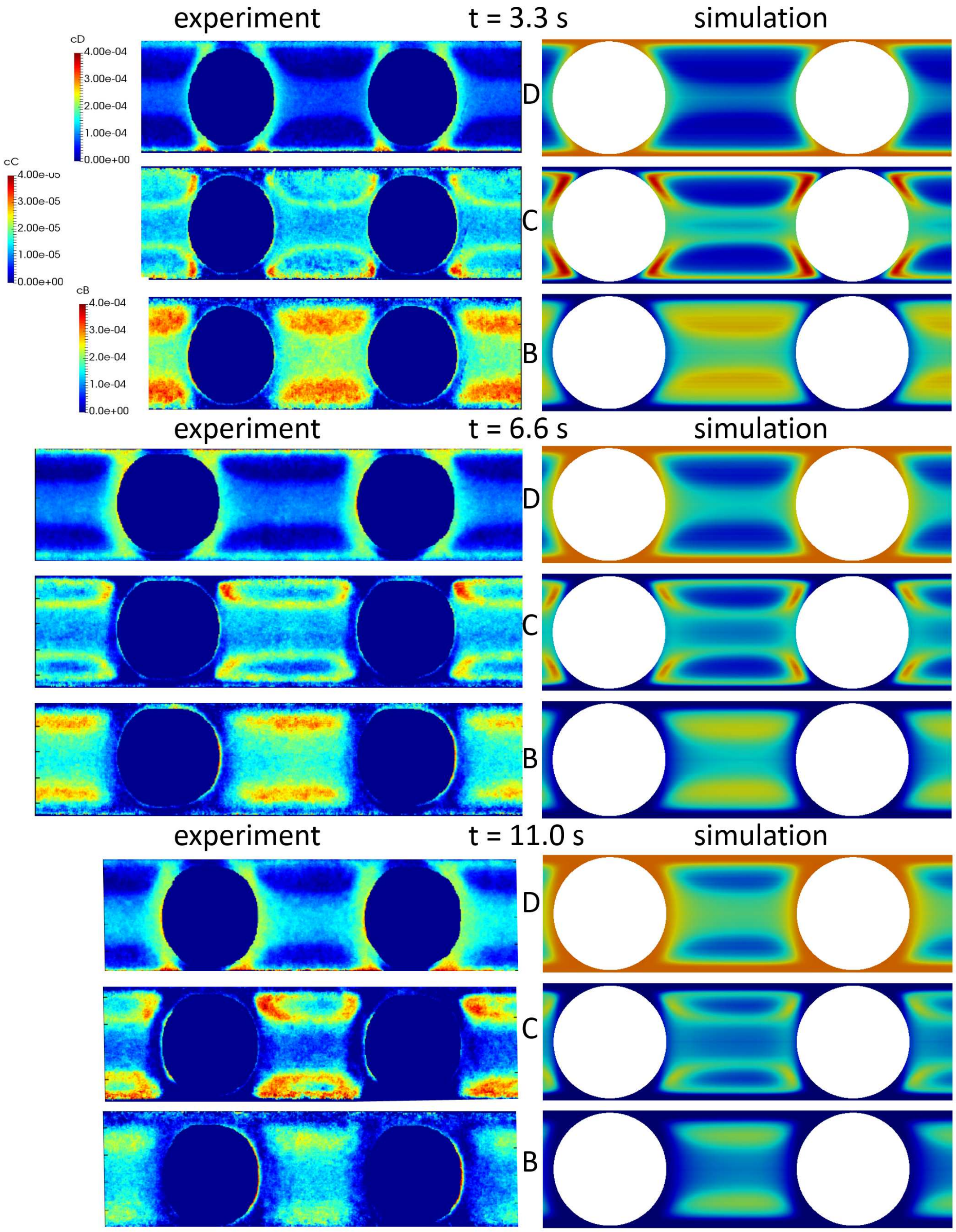

FIGURE 1.20 Comparison of the computationally predicted concentration distribution at three different time levels with the experimental reference results.