# Numerical Methods for the Simulation of the Acoustics of Multiphase Flows

Vom Fachbereich Maschinenbau
an der Technischen Universität Darmstadt
zur Erlangung des Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte

## Dissertation

vorgelegt von

**Jonas Friedrich, M.Sc.**
aus Wiesbaden

| | |
|---|---|
| 1. Gutachten: | Prof. Dr. rer. nat. Michael Schäfer |
| 2. Gutachten: | Prof. Dr. rer. nat. Oliver Weeger |
| Tag der Einreichung: | 29. April 2024 |
| Tag der mündlichen Prüfung: | 09. Juli 2024 |

Darmstadt 2024
D17

Numerical Methods for the Simulation of the Acoustics of Multiphase Flows

Submitted doctoral thesis by Jonas Friedrich

1. Review: Prof. Dr. rer. nat. Michael Schäfer
2. Review: Prof. Dr. rer. nat. Oliver Weeger

Date of submission: 29. April 2024
Date of thesis defense: 09. July 2024

Darmstadt, Technische Universität Darmstadt

Year thesis published in TUprints: 2024
URN of the doctoral thesis: urn:nbn:de:tuda-tuprints-280364
URL of the doctoral thesis: https://tuprints.ulb.tu-darmstadt.de/id/eprint/28036

**Acknowledgments**

Die vorliegende Dissertation wurde im Rahmen meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Fachgebiet Numerische Berechnungsverfahren im Maschinenbau an der Technischen Universität Darmstadt erarbeitet. An dieser Stelle möchte ich allen Personen meine Dankbarkeit aussprechen, die mich auf meinem bisherigen Weg begleitet und unterstützt haben.

Mein erster Dank gilt Herrn Prof. Dr. rer. nat. Michael Schäfer, der mich am Fachgebiet aufgenommen und damit meine Promotion ermöglicht hat. Darüber hinaus möchte ich mich bei ihm für den gegebenen Freiraum bedanken, in welchem er dennoch immer ein offenes Ohr hatte. Bei Prof. Dr. rer. nat. Oliver Weeger bedanke ich mich für die freundliche Übernahme des Korreferats.
Einen besonderen Dank möchte ich Monika Müller und Michael Fladerer aussprechen. Monika stand mir vom ersten Tag an in allen administrativen und organisatorischen Themen zur Seite und das immer mit einem Lächeln. Abgesehen zum unbeliebten Thema Windows, konnte ich mich jederzeit mit meinen IT-Fragen an Michael wenden und mir wurde stets mit seiner ausführlichen Antwort geholfen.

Sowohl für die durchweg positive Arbeitsatmosphäre als auch für den fachliche Austausch möchte ich mich bei allen Kollegen bedanken, angefangen bei meinen Bürokollegen Awais Ali und Jessica Mariño Salguero. Einen herzlichen Dank gilt auch Elena Kolb, Artem Karev, Andreas Schmitt und Maximilian Kannapinn für die unvergessliche Zeit am Fachgebiet. Ausgesprochen dankbar bin ich für diejenigen Kollegen aus denen echte Freunde geworden sind: Felix Köhler, Dominic Jekel und Jakob Munz.

Zu guter Letzt gebührt mein größter Dank meiner Familie. Ohne meine Eltern, die mich zu jedem Zeitpunkt in meinem Leben unterstützt haben und damit bis zum heutigen Tage nicht aufgehört haben, wäre mein bisheriger Werdegang nicht möglich gewesen. Bei meinen tollen Kindern, Levi und Henri, möchte ich mich für das Verständnis meiner Abwesenheit während der Fertigstellung dieser Arbeit bedanken. Für die Liebe, den Rückhalt und die Unterstützung in allen Lebenslagen bin ich meiner Frau Elisabeth unendlich dankbar.

# Abstract

The noise reduction of technical systems has become one of the main challenges in the twenty-first century. In order to tackle this challenge, numerical simulations are a crucial part in the process of understanding the physical mechanisms of sound generation. Nevertheless, experiments have been traditionally used to study the complex phenomenon of sound generated by multiphase flows since there were no suitable numerical methods available. For this reason, a numerical framework for simulating acoustics produced by low Mach number multiphase flows is presented within this work.
The motion of the multiphase flow is described by a single set of the Navier-Stokes equations. Various phases in the computational domain are treated as one fluid with variable material properties. The distribution of the phases is advected with the Volume of Fluid method based on a high resolution scheme methodology. To account surface tension, a singular term formulated by the Continuous Surface Tension method is added to the governing equations. For surface tension dominated flows, the accuracy of the forces around the interface is greatly dependent on the interface curvature computation. Therefore, different improvement strategies applied to curvature computation methods along with a machine learning approach are introduced and analyzed.
Different orders in energy, length and time scales of the acoustics and the flow lead to a segregated handling. An acoustic/viscous splitting approach is employed for the computation of the acoustics. After the multiphase flow field is obtained, the generation and propagation of the acoustic waves is determined on the basis of the linearized Euler equations. The fluid and acoustic sets of equations are both discretized with the Finite Volume method in the in-house solver FASTEST. A one-way coupling between the two physical disciplines, justified by the difference in the characteristic scales, is accomplished by an acoustic source term derived from the unsteady flow field. By adapting the acoustic source term to the multiphase environment, the difficulties of a moving interface are overcome. The coupled validation test cases show the expected results.
For the final part of this work, the developed numerical methods are applied on a three-dimensional test case. A complex and not yet fully understood example of multiphase acoustics is the sound of a water drop impacting into a water pool, as can be heard from a tripping tap. First, the necessary of a very high resolution in space and time as well as an accurate curvature model is demonstrated. Afterwards, the results of the simulations are compared to experimental data with satisfactory agreement and further insights on the sound producing physics are given.

# Zusammenfassung

Die Lärmminderung bei technischen Systemen ist zu einer der größten Herausforderungen des einundzwanzigsten Jahrhunderts geworden. Um diese Herausforderung zu bewältigen, sind numerische Simulationen ein entscheidender Bestandteil bei der Untersuchung von physikalischen Mechanismen der Schallerzeugung. Aufgrund des Fehlens geeigneten numerische Methoden, wurden in der Vergangenheit meistens Experimente verwendet, um das komplexe Phänomen des von Mehrphasenströmungen erzeugten Schalls zu untersuchen. Aus diesem Grund wird in dieser Arbeit eine Softwareumgebung für die Simulation der von Mehrphasenströmungen mit niedriger Machzahl erzeugten Akustik vorgestellt.
Die Bewegung der mehrphasigen Strömung wird durch einen einzigen Satz der Navier-Stokes-Gleichungen beschrieben. Die verschiedenen Phasen im Berechnungsgebiet werden als ein Fluid mit variablen Materialeigenschaften behandelt. Die Verteilung der Phasen wird mit der Volume-of-Fluid-Methode auf der Grundlage hochauflösender Grenzflächenerhaltungs-Schemata berechnet. Um die Oberflächenspannung zu berücksichtigen, wird ein einzelner Term, der mit der Methode der kontinuierlichen Oberflächenspannung formuliert wird, zu den Erhaltungsgleichungen hinzugefügt. Bei Strömungen, die von Oberflächenspannung dominiert werden, hängt die Genauigkeit der Kräfte an der Grenzfläche stark von der Berechnung der Grenzflächenkrümmung ab. Daher werden verschiedene Verbesserungsstrategien für die Krümmungsberechnungsmethoden zusammen mit einem maschinellen Lernansatz eingeführt und analysiert.
Unterschiedliche Größenordnungen in Energie-, Längen- und Zeitskalen zwischen der Akustik und der Strömung führen zu einer getrennten Behandlung. Für die Berechnung der Akustik wird ein akustisch-viskoser Splitting-Ansatz verwendet. Nachdem das mehrphasige Strömungsfeld berechnet wurde, wird die Erzeugung und Ausbreitung der akustischen Wellen auf Basis der linearisierten Euler-Gleichungen bestimmt. Die Strömungs- als auch die Akustikgleichungen werden mit der Finite-Volumen-Methode im institutseigenen Gleichungslöser FASTEST diskretisiert. Eine einseitige Kopplung zwischen den beiden physikalischen Disziplinen, die durch den Unterschied in den charakteristischen Skalen begründet ist, wird durch einen aus dem instationären Strömungsfeld abgeleiteten akustischen Quellterm erreicht. Durch die Anpassung des akustischen Quellterms an die mehrphasige Umgebung werden die Schwierigkeiten einer sich bewegenden Grenzfläche überwunden. Die gekoppelten Validierungstestfälle zeigen die erwarteten Ergebnisse.
Im letzten Teil dieser Arbeit werden die entwickelten numerischen Methoden auf einen dreidimensionalen Testfall angewandt. Ein komplexes und noch nicht vollständig verstandenes Beispiel für die Mehrphasenakustik ist das Geräusch eines in ein Wasserbecken fallenden Wassertropfens, wie beispielweise bei einem undichten Wasserhahn. Zunächst wird die Notwendigkeit einer sehr hohen räumlichen und zeitlichen Auflösung sowie eines genauen Krümmungsmodells demonstriert. Anschließend werden die Ergebnisse der Simulationen mit experimentellen Daten verglichen. Eine zufriedenstellende Übereinstimmung wird festgestellt und weitere Einblicke in die schallerzeugende Physik gegeben.

# Table of Contents

# 1. Introduction

## 1.1 Motivation

Product development in engineering disciplines has been driven by physical prototyping. The process of finding an ideal design can be very costly and time-consuming. With the rise of computer technology a field called Computational Fluid Dynamics (CFD) emerged, which simulates a fluid flow of any kind in or around a geometry. By integrating CFD in the development process, the number of physical prototypes and hence, the time-to-market can be reduced. But not only engineers are profiting from CFD, also scientists are able to gain deeper knowledge of complex physical behavior.
A focus in recent researches is the reduction of noise, which is defined as unwanted and/or harmful sound [32]. In general, sound is produced mainly by two mechanisms: flow induced sound (aeroacoustics) and sound generated by mechanical vibrations (vibroacoustics). However, there is sound which is not directly explainable with one of the main mechanisms. Consider rain falling on a lake, waves breaking on a beach or a leaking tap releasing water drops into a sink full of water. All those examples can be categorized as multiphase acoustics, since the sound is produced by the presence of an interface between air and water. Although the first experimental results on this topic have been presented in the beginning of the nineteenth century [73], the airborne sound producing mechanisms have not yet been fully discovered.
As most of the naturally occurring flows, the velocity is subsonic, so that the computation of the acoustics imposes a multi scale problem. Different orders of magnitudes in energy, length and time scales lead to a partitioned approach. Once the flow field has been precisely determined, the acoustics are then computed, fed by an acoustic source term. In most of these hybrid models, turbulence effects are the main contributor to the acoustic source term. The extension of an hybrid aeroacoustic simulation approach to multiphase flows is part of the ongoing research.
Despite the physically correct coupling of the acoustics to the multiphase flow, the accurate representation of the multiphase interface is required. For interface capturing methods, as the employed Volume of Fluid (VOF) method, the transition region between phases is implicitly present. This implies that the exact interface position is not directly available

imposing difficulties for the curvature determination. Especially surface tension dominated multiphase flows rely on proper methods estimating the interface curvature from the transition region.
Finally, building a numerical framework which includes the computation of surface tension dominated multiphase flows, the propagation of sound within a multiphase domain and their coupling remains for the mentioned reasons a challenge and motivates this work.

## 1.2 State of the Art

This chapter covers the developments concerning the numerical methods for computing multiphase flow with surface tension, acoustics and the few approaches covering a coupled framework.

#### Multiphase Flow

For most problems analyzed with CFD, the physical basis is given by the partial differential equations derived by Navier and Stokes. By adding a transport equation for a variable that separates the physical properties in the Navier-Stokes equations (NSE) for different phases, the most commonly used method for computing multiphase flows, namely the Volume of Fluid (VOF) method, is obtained.
The VOF method to track free boundaries was first introduced in 1981 by Hirt et al. [49]. A color function whose value is unity when a cell is fully occupied by fluid and zero if a cell contains no fluid is transported through the domain. A value between unity and zero represents the volume fraction of the fluid occupied in the cell, thus the interface between phases. For the prediction of incompressible, immiscible multiphase flows, two fundamental types of VOF methods emerged: Geometric and compressive methods.
Geometric VOF methods reconstruct the interface explicitly from the volume fraction field. The accuracy of geometric VOF methods and the sharpness of the interface are their strengths, however the computational effort is their weakness [20]. Due to the geometrical reconstruction algorithms, the computational resources and the implementation complexity are increasing significantly for three dimensions or on arbitrary grids [144]. After presenting the Simple Line Interface Calculation (SLIC) algorithm [87], many improvements of geometric VOF methods have been introduced in the literature [148, 108, 106, 7].
In contrast compressive, or algebraic, VOF methods solve an additional transport equation for the color function, hence no explicit interface reconstruction is applied. However, in order to keep the interface as sharp as possible, special discretization techniques in space and time are necessary [21]. Algebraic VOF methods have been modified and improved to compensate the drawbacks of the original algorithm. To reduce numerical diffusion and to increase curvature approximations, the steep color function is replaced by a smooth

level-set approach [90, 129]. Another attempt to reduce the diffusion is to apply a high resolution differencing scheme on the discretization of the transport equation for the color function. As the original VOF method, the High Resolution Interface Capturing scheme (HRIC) or Compressive Interface Capturing Scheme for Arbitrary Meshes (CICSAM) are based on a donor-acceptor formulation [84, 135]. Other schemes like M-CICSAM [140] or CUIBS [96] followed with enhanced performance concerning interface sharpness, interface stability at higher Courant numbers or accuracy on different grid types.
The VOF method relies on a finite volume discretization, which solves partial differential equations by dividing the domain into discrete control volumes (CV). These volumes are then used to approximate the integral form of the governing equations, allowing for the calculation of flow variables at each volume. One advantage of this Finite Volume Method (FVM) is its ability to conserve mass, momentum, and energy within each volume. However, over the past years other numerical approaches have gained interest: the Discontinous Galerkin (DG) method, the Lattice Boltzmann Method (LBM) and the Smoothed Particel Hydrodynamics (SPH) method. Although the DG method is not available in commercial CFD codes, several scientific open-source [59] or industrial in-house codes [138] are in development. By combining features of the FVM and the Finite Element Method (FEM), the DG method has advantages in the context of high-order, parallel and unstructured CFD codes. On the other side, the DG struggles with the typical drawbacks of high-order methods such as instabilities [74]. A different class of numerical methods is the LBM, which is motivated by the idea of capturing the physics of macroscopic flow through simple local microscale operations [6]. On a discrete regular grid of lattices, a distribution function and its temporal evolution are described by the Boltzmann equation. All relevant macroscopic flow variables (velocity, pressure, etc.) are calculated from this molecular distribution function. Its popularity is based on its ability to simulate flows beyond the limits of the NSE and its efficient parallelization. Meshless methods, such as the SPH, are ideal for time dependent complex geometries due to their flexibility and therefore, popular for free surface flows. Generation, organization and adapting the numerical grid is often difficult for mesh based methods in terms of computational time and complexity [112]. For solving partial differential equations with the SPH method, flow describing mass particles are distributed equally in the fluid. Within a smoothing length, neighboring particles interact through a kernel function and the flow variables are obtained by a summation approximation. Since the SPH method is not as mature as the FVM, some general challenges remain, e.g. adaptive particle resolution or turbulent flows with high-Reynolds numbers [103].

### Surface Tension

In some multiphase flow problems surface tension is not negligible, e.g. when the relation between the dynamic viscosity times a characteristic velocity and the surface tension (capillary number) is high. For the computation of the naturally as surface force act-

ing surface tension, Brackbill et al. [14] introduced the Continuum Surface Force (CSF) model. Surface tension is modeled as a body force, which is imposed as a continuous, three-dimensional effect across an interface. Although the CSF model has been employed in all kinds of interfacial flow environments [2, 147, 4], its disadvantageous property of producing unphysical flow ("spurious currents") around the interface has been an issue since. Francois et al. [33] stated three key conclusions for surface-tension-driven flows: First, an exact balance between the surface tension and the pressure gradient forces can be achieved, if the interfacial curvature is known exactly, by identical discretization of both. Second, sharp and continuous surface tension representations within a balanced-force flow algorithm and the same curvature approximation only differ in the spatial distribution of the pressure jump across the interface. Third, the origin of spurious currents within a balanced-force algorithm lies in the errors of the curvature estimation. Apart from these three conclusions, it should be mentioned that if other forces, like buoyancy, are dominant, there are no evident differences between surface tension models or curvature estimation methods. Francois et al. worked with a geometric VOF method, nevertheless, others successfully adapted the ideas to different interfacial flow algorithms [48, 22, 147, 145].
However, in real flow simulations where no a priori knowledge of the interface curvature is present, the accuracy of surface force estimates mostly depends on the accuracy of the interface curvature estimation [70]. Therefore, different methods have been developed, which rely on either the direct derivation from the volume fraction field or on discrete differential operators applied to an explicit description of the location of the interface [102].
Within the VOF, the curvature estimation methods are based on the implicit representation of the interface. Brackbill et al. [14] stated that the discontinuous volume fraction field needs to be smoothed by a convolution kernel, before it is differentiated twice for the curvature. The studies of Williams et al. [143] compared different smoothing kernels and concluded that a kernel for convolution methods needs to satisfy certain requirements. Although such methods are used for comparisons [16, 33] or in hybrid approaches [94], they are not convergent with mesh refinement and labeled as outdated [102]. Apart from an abruptly varying volume fraction field, a level-set function can be used either for interface advection, as mentioned, or only for curvature computation. In the latter, the level-set function is temporarily constructed from the volume fraction field for computing the interface normals and curvatures. It is not advected nor it has an explicit effect on the advection of the VOF field [92]. Other versions of this CLSVOF method, in which the level-set field is advected as well, can be found exemplary in [128, 119, 15].
A different class of curvature computation methods are based on computing height functions. Sussman et al. [125] first presented the idea in the context of curvature computations in multiphase flows. In case of VOF methods, the discrete volume fractions are summed up along the columns ("heights") in the direction of the dominant normal vector component. Then, the curvature is derived by differentiation of the heights around an interface cell. Simple implementation [16] and yet the only method with convergence at mesh refinement [27] are the main reasons for the intensive research over the past years [5, 34, 28, 27, 13].

In general, the task of the mentioned methods is to identify the interface from the volume fractions by a mathematical operation and use the outcome to estimate the curvature. A new approach with promising results was presented by Qi et al. [105], which uses machine learning techniques to find a relation between volume fractions and their curvature. Patel et al. [95] further investigated the machine learning methodology and evaluated its performance with a flow solver.

**Computational Aeroacoustics**

In contrast ot the multiphase computation, the numerical prediction of sound is a comparable new field of research, although it origins from the work of Lighthill [69] in 1952. The reformulation of the compressible Navier-Stokes equations into an inhomogeneous wave equation, known as Lighthill's acoustic analogy, set the path for the hybrid modeling approaches in computational aeroacoustics (CAA). In this context, hybrid expresses the separate handling of flow and acoustics overcoming the multi scale problem. Sources from the solution of the transient flow field are derived, which are used to compute the acoustics. Lighthill's findings are based on the investigations of unbounded free stream flows and therefore, other configurations are not well represented. Extensions are given by Curle's formulation [17], which includes fixed rigid bodies or the most common Ffowcs Williams-Hawkings (FW-H) formulation [142], which is able to model the sound generation by moving surfaces. In comparison to the resources consumed by the flow field computation, all these integral methods are cost-efficient independent of the observer location.
In addition to the use of analogies, there are methods based on disturbance variables. The acoustic/viscous splitting technique decomposes the compressible quantities into a superposition of an incompressible part with an acoustic perturbation, referred as expansion about incompressible flow (EIF). With this method the acoustic sources are derived from the incompressible Navier-Stokes equations (NSE). Shen and Sørenson [116] modified the original method from Hardin and Pope [45] by changing slightly the basic decomposition of the variables in order to account for non-isentropic flows. Bailly et al. [8] and Bogey et al. [12] showed that the linearized Euler equations (LEE) with their source term definition are able to predict aerodynamic noise. Seo and Moon [113] proposed a different version of the perturbation equations for handling near field compressibility effects, which are named perturbed compressible equations (PCE). Later, Seo and Moon [114] linearized their compressible perturbation equations (LPCE) to ensure that grid-independent acoustic solutions are obtained by suppressing the generation of perturbed vorticity in the formulation. The acoustic perturbation equations (APE) by Ewert and Schröder [29] were presented in several versions, which are exited by source terms determined from a compressible or incompressible flow problem. Hüppe et al. [50] proposed a computationally efficient reformulation of an APE system leading to the perturbed convective wave equation (PCWE). A comprehensive review of the existing hybrid aeroacoustic approaches is given

by Schoder et al. [111]. Advantageous in all hybrid methods is the possibility of using different computational grids and methods to compute each physical discipline achieving the highest accuracy. A common feature of all hybrid models based on incompressible conservation equations is the one-sided coupling, in which the feedback of the acoustic field on the flow is neglected.
Instead of modeling the acoustics, the Direct Noise Computation (DNC) solves the compressible NSE resulting in the united field of flow and acoustics. Since all relevant (turbulent) structures in the flow have to be resolved, the computational resources are immense. Even with the current advancements in high performance clusters (HPC), the application of DNC is restricted to certain flow configurations in the mid and high Mach number regime [53]. With a certain degree of turbulence modeling as in a Large Eddy Simulation (LES), a DNC is applicable to technical problems [77, 35]. However, if the underlying flow is characterized by a low Mach number, the scale discrepancy of the characteristic lengths between flow and acoustics increases, causing the DNC to become inefficient. Since many industrial flow applications operate at low speeds with Mach numbers smaller 0.3, hybrid methodologies are considered as the most practical methods for aeroacoustics [53].
The combination of incompressible multiphase flows and acoustics are not very frequent in the literature of numerical methods. Mostly compressible phenomena and the effect of the acoustics on the multiphase flow are investigated. Munz et al. [82] extended the EIF approach for low Mach number flow with variable density, variable temperature and heat transfer by replacing the incompressible solution of the NSE with the solution of the compressible NSE at zero Mach number. Denner et el. [24] proposed a pressure-based algorithm, which is able to retain the acoustic properties of shock waves at an interface. The sound generation and propagation in two phases were simulated by Tajiri et al. [131] with the help of a finite difference lattice Boltzmann method.

## 1.3 Objectives and Outline

In this work a simulation approach for predicting acoustics emerging from low Mach multiphase flows are investigated. In order to achieve the overarching objective of a reliable coupled simulation framework, all numerical components must function optimally both individually and with each other. In the particular case of this thesis, this means that for the multiphase part a fast, accurate and in parallel usable curvature model has to be developed. The surface tension computation is responsible for the pressure field, which imposes a jump across an interface. The coupling mechanism to the acoustics is addressed due to the abruptly varying pressure, especially if the interface is moving. The acoustic computation is attached after the determination of the flow variables by treating the multi scale problem of acoustics at low Mach number flows with a splitting technique. Both, the multiphase and the acoustic part, are realized within the computational fluid dynamics

solver FASTEST [25] of the Institute of Numerical Methods in Mechanical Engineering at the Technical University of Darmstadt.
After the introductory Chapter 1, this thesis is structured as follows. Chapter 2 explains the mathematical and physical principles with their equations. Valid assumptions, e.g. one-fluid formulation for the multiphase flow or the linearization of the acoustic terms, are given as well. In Chapter 3, the employed numerical methods for solving coupled multiphase acoustic problems are presented. Due to the special treatment, the numerical approaches around the interface are addressed in Chapter 4. Different curvature computation methods and the strategy for handling the acoustic source term are introduced. Chapter 5 analyses the different curvature computation methods and demonstrates the correct coupling between multiphase flow and acoustics. Additionally, a complex three dimensional drop impact test case is shown, analyzed and compared to experimental results. The final Chapter 6 summarizes the findings of this thesis and gives an outlook on future research opportunities.

# 2. Physical Models

Every numerical simulation is based on different physical laws and models. Dimensionless numbers characterize the problem and support the engineer in the decision of which assumptions and simplifications are valid. The Knudsen number is defined as the relation between the mean free path of a particle and a characteristic length of the flow field. If the Knudsen number is below or equal 0.01, the flow can be described as a continuum and it is not necessary to consider every molecule. Hence, the fluid flow in the problem domain can be described by a set of differential equations and boundary conditions. The Navier-Stokes equations are the most common mathematical model to describe viscous fluids in a continuum. However, assumptions are made to model the flows presented in this work, such as mass transfer between phases is omitted as well as mass sources or sinks and chemical reactions. Further, a critical Reynolds number is not reached in the problems dealt with in this work. This means that the ratio between the inertia and viscosity forces is comparatively low, so that the flow can be described as laminar and turbulence is not considered.
Sound is understood as small fluctuations of pressure which are superimposed to the pressure field. A sound source oscillates and brings the surrounding fluid or gas phase into motion. Due to its compressibility and mass these oscillations are transmitted to the receiver. In case of low Mach number fluid flows, a simultaneously computation of the acoustics with the Navier-Stokes equations is inefficient. The different length scales of the fluid motion and the acoustics would require very small time steps accounting the speed of sound in combination with a fine numerical grid able to resolve the flow. Therefore, the model employed to describe acoustics is based on the Euler equations linearized around a stationary field.
Let the flow variables be the density $\rho(t, x_i)$, the velocity vector $u_i(t, x_i)$ and the pressure $p(t, x_i)$. Geometrical quantities are given by $n_i$ as the outward unit normal vector and $t_i$ as the tangential vector of the corresponding surface. For clarity the dependency of space and time of the flow variables will be left out.
In this chapter the physical models to describe the Navier-Stokes equations are presented in Section 2.1, followed by the additional models for immiscible two-phase flows in Section 2.2. Finally, the physics to describe acoustics in Section 2.3 conclude this chapter.

## 2.1 Fluid Mechanics

A conservation law states that for an isolated system a particular physical measurable property is conserved as the system changes over time. The conservation of mass, linear momentum and energy are the basis for solving a fluid dynamics problem. Independent of the fluid properties, these conservation laws completely describe the physical behavior. In computational fluid dynamics an Eulerian formulation is the most common approach for their mathematical description. Flow variables are measured at a fixed control point or volume as the time passes. In contrast, the Lagrangian formulation focuses on a flow particle and how its properties change over space and time by following its path. For a given situation, one or the other of these techniques is easier to use, but regardless of their ease of application, both must give the same results. The Reynolds transport theorem (2.1) converts the conservation laws from a closed system formulation (Lagrangian) to that of a control volume (Eulerian):

$$\frac{\mathrm{D}}{\mathrm{D}t}\int_{V(t)}\varphi\,\mathrm{d}V=\int_{V}\frac{\partial\varphi}{\partial t}\,\mathrm{d}V+\int_{S}(\varphi u_i)n_i\,\mathrm{d}S. \tag{2.1}$$

A fixed mass or closed system time derivative $\frac{\mathrm{D}}{\mathrm{D}t}$ of an integral with varying region $V(t)$ for a quantity $\varphi$ is equal to this quantity integrated over a fixed region $V$ and to the flux $u_i n_i$ of this quantity through the bounding surfaces $S$, if both regions coincide at time $t$ [120]. The differential form of the Reynolds transport theorem applied on the conservation equations for mass, momentum and energy are commonly know as Navier-Stokes equations.

### 2.1.1 Navier-Stokes Equations

Initially the name Navier-Stokes referred to the conservation equation of linear momentum, however, nowadays it is understood as the collective of mass, momentum and energy conservation equations [81]. Due to the assumption of constant temperature throughout any problem considered in this work, the energy equation is omitted.

#### Conservation of Mass

Using the Eulerian part of the relation (2.1) for density in place of a general variable $\varphi$ leads to

$$\int_{V}\frac{\partial\rho}{\partial t}\,\mathrm{d}V+\int_{S}(\rho u_i)n_i\,\mathrm{d}S=0. \tag{2.2}$$

Mass sources or sinks are not present thus Equation (2.2) states that the temporal change of mass is unequal to zero if mass enters or leaves the volume. Written in differential form

the mass conservation is called continuity equation and is given by

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0. \tag{2.3}$$

**Conservation of Momentum**

Newton's second law of motion asserts that the momentum of a body only changes over time if a net force is acting on it. In case of fluid mechanics momentum is defined as the volume integral of the product between density and velocity. Similar to the mass, the Reynolds transport theorem is applied to the linear momentum which becomes

$$\int_V \frac{\partial(\rho u_i)}{\partial t}\,\mathrm{d}V + \int_S (\rho u_i) u_i n_i\,\mathrm{d}S = F_i. \tag{2.4}$$

The sum of forces $F_i$ generates the motion of fluid and can be divided into body forces, which act on every fluid particle and into surface forces, which are exerted by the surrounding environment on the surface of the considered part of the fluid:

$$F_i = \underbrace{\int_V \rho f_i\,\mathrm{d}V}_{\text{Body forces}} + \underbrace{\int_S t_i^*\,\mathrm{d}S}_{\text{Surface forces}}\ . \tag{2.5}$$

Whereas body or also volume force is trivial, a closer look is taken at the surface force and its stress vector $t_i$. Consider a material volume for which the stress principle of Cauchy states that on any plane there is a force that depends on the orientation of the plane. The stress vector becomes a linear function of the outward pointing unit normal vector of the plane $t_i^* = t_i^*(n_j)$. For example in the i-th direction follows

$$t_i^* = \sigma_{1i} n_1 + \sigma_{2i} n_2 + \sigma_{3i} n_3 \quad \text{for} \quad i = 1, 2, 3. \tag{2.6}$$

This results in nine components $\sigma_{ji}$ to describe the state of stress in any point inside a material. Cauchy introduced the components in a tensor of second order, also known as Cauchy stress tensor which is multiplied with the unit normal vector of the plane:

$$t_i^* = \begin{bmatrix} n_1 & n_2 & n_3 \end{bmatrix} \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}. \tag{2.7}$$

Each component of $\sigma_{ji}$ represents the stress acting on face $i$ in the $j$ direction. Each row represents the face and each column stands for a force in the corresponding direction.

Elements on the main diagonal ($j = i$) are normal stresses and on the secondary diagonal ($j \neq i$) are shear stresses. With this formulation the differential form of the linear momentum conservation is

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial (\rho u_i) u_j}{\partial x_j} = \rho f_i + \frac{\partial \sigma_{ji}}{\partial x_j}. \tag{2.8}$$

The same principles are valid for the conservation of angular momentum, which is defined as the product of the moment of inertia and the angular velocity. It states that the temporal change of momentum is equal to the moment acting on the body. Without further description the conservation of angular momentum concludes that the Cauchy stress tensor is symmetric

$$\sigma_{ji} = \sigma_{ij}. \tag{2.9}$$

For further details on the derivation of the conservation laws the reader is referred to [81, 120].

**Assumptions**

The conservation laws in Equations (2.3) and (2.8) require certain assumptions and relations for solving fluid mechanical problems. In the first instance the stress tensor is split into normal and shear stresses

$$\sigma_{ij} = \tau_{ij} - p\delta_{ij}, \tag{2.10}$$

where in fluid mechanics normal stresses are expressed with pressure $p$ multiplied with the Kronecker delta $\delta_{ij}$ ($\delta_{ij} = 1, \quad \text{if} \quad i = j \quad \text{and} \quad \delta_{ij} = 0 \quad \text{otherwise}$). A linear viscous isotropic fluid, named Newtonian fluid, is characterized by linear dependency of the viscous stress and the strain rate with the dynamic viscosity $\mu$ as a constant of proportionality ($\tau = \mu\dot{\gamma}$). Thus, the viscous stress tensor in Equation (2.10) for a general Newtonian fluid is formulated in index notation as

$$\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3}\mu \left( \frac{\partial u_j}{\partial x_j} \right) \delta_{ij}, \tag{2.11}$$

where the second term accounts the bulk viscosity. Using Equation (2.11) on a material volume, a relation between the stress tensor $\sigma_{ij}$ and the flow variables can be established:

$$\sigma_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \left( \frac{2}{3}\mu \frac{\partial u_j}{\partial x_j} + p \right) \delta_{ij}. \tag{2.12}$$

Adding the material law of Equation (2.12) to the differential form of the linear momentum in Equation (2.8) yields the compressible Navier-Stokes equations without the energy

equation.
Compressibility can be neglected if the Mach number, which is the relation between the characteristic flow velocity and the prevailing speed of sound, is smaller than 0.3. In such case the density of gases may be assumed constant, whereas in general at practical velocity liquids are assumed incompressible. Due to the negligible density change the divergence of the velocity vector in Equation (2.3) becomes zero ($\frac{\partial u_i}{\partial x_i} = 0$). Consequently, the material law in Equation (2.12) simplifies and the incompressible Navier-Stokes Equations written in non-conservative form read as

$$\frac{\partial u_i}{\partial x_i} = 0, \tag{2.13}$$

$$\rho \left( \frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) = \rho f_i + \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] - \frac{\partial p}{\partial x_i} . \tag{2.14}$$

In addition to the conservation formulations for incompressible flows, the physical models of two-phase flows with different material properties will be explained in the following section.

## 2.2 Multiphase Flow

This section deals with the physical models for the computation of multiphase flows. The term multiphase implies that in the domain there are different phases. However, different fluids with the same phase are included as well. Here, the focus lies on immiscible gas-liquid flows, without identifying any wall or body as a third phase. Sharp interfaces separate different fluids or phases, whereby the transition of their properties (density, viscosity) occurs on very small scales. By means of the continuum hypothesis the interface is assumed to have vanishing thickness [134]. Nevertheless, constitutive models have to be added to the Navier-Stokes equations in order to compute two-phase flows.

### 2.2.1 One-Fluid Formulation

One approach for multiphase flows is to employ the Navier-Stokes equations for each phase coupled with appropriate conditions at the interface. The one-fluid formulation on the contrary uses one set of governing equations for the whole flow domain occupied by various immiscible phases [134]. The differentiation between phases is employed by variable material properties that change abruptly at the interface. For accounting two phases, denoted with superscripts $^A$ and $^B$, the dynamic viscosity and the density in the incompressible

Navier-Stokes equations depend on an indicator function $F$ as

$$\begin{aligned} \mu &= \mu^A F^A + \mu^B F^B, \\ \rho &= \rho^A F^A + \rho^B F^B. \end{aligned} \tag{2.15}$$

Important to mention is that for incompressible two-phase flows, in which the density of each phase is constant, the mass conservation equation (2.13) is equivalent to single-phase flows [19]. Since the velocity field is continuous among phases ($u_i^A$=$u_i^B$), there is no difference in normal or tangential velocity at the interface $\Sigma$:

$$u_i^A n_i^\Sigma = -u_i^B n_i^\Sigma, \tag{2.16}$$

$$u_i^A t_i^\Sigma = u_i^B t_i^\Sigma. \tag{2.17}$$

Equation (2.16) represents an impermeable interface without mass transfer and the statement in Equation (2.17) expresses a no-slip condition [141].

### 2.2.2 Volume Fraction Transport

Considering a volume $V$, the function $F$ describes the proportion of a phase in it. For a fluid $A$ the function reads as

$$F^A = \frac{V^A}{V}. \tag{2.18}$$

Due to the relation in Equation (2.18) the function $F$ is called volume fraction. In case of only fluid $A$ occupies volume $V$, the volume fraction of fluid $B$ is zero. Therefore, the volume fraction of two phases (or in general of endless phases) in sum must equal one:

$$F^A + F^B = 1. \tag{2.19}$$

However, in further explanations the volume fraction is referred to $F$ without superscript. Every value between zero and one represents the interface of finite thickness, thus the representation of the scalar volume fraction in a fluid domain is described by

$$F(x_i, t) = \begin{cases} 1, & \text{if } x_i \in \text{fluid } A, \\ 0, & \text{if } x_i \in \text{fluid } B, \\ 0 < F < 1, & \text{if } x_i \in \text{between fluid } A \text{ and fluid } B. \end{cases} \tag{2.20}$$

Since the volume fraction moves with the continuous velocity field, it is employed in the transport equation (2.1) and given in differential form for its advection:

$$\frac{\partial F}{\partial t} + u_i \frac{\partial F}{\partial x_i} = 0. \tag{2.21}$$

The volume fraction transport equation (2.21) or the conservation equations of mass and momentum as in (2.13)-(2.14) do not capture interface effects such as surface tension; hence are described afterwards.

### 2.2.3 Surface Tension

When dealing with multiphase flows, additional physics at the phase boundaries occur. Some have been neglected in the introduction of this chapter, however, surface tension as an interfacial phenomena of liquids will be included. The microscopic origin of surface tension lies in the intermolecular interactions [75]. Molecules of liquids share cohesive forces equally with all their neighbors. On the surface, for example in a water-air system, there are fewer neighbor molecules to cling to, see Figure 2.1. This results in establishing a stronger bond

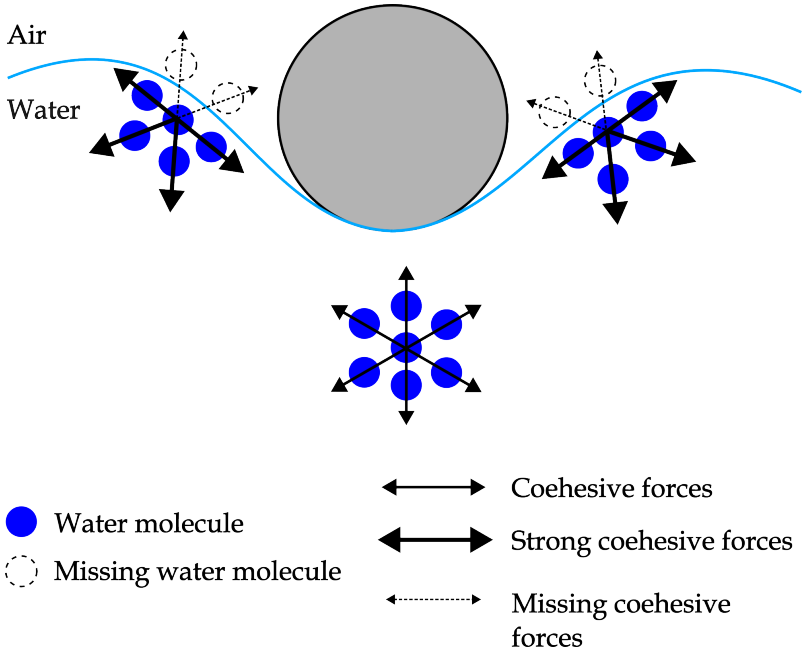



Figure 2.1: Microscopic view on surface tension (in reference to [136])

with the remaining neighbors and in a net force directed into the liquid. The latter creates an internal pressure which is only observable at curved surfaces, where the unbalanced surface tension forces are equilibrated by excess pressure on the concave side [11]. For a curved interface without external forces the Young-Laplace equation approximates the jump in fluid pressure $\Delta p$ as function of the surface tension coefficient $\sigma$ and the principal radii of curvature $R_1$ and $R_2$ as

$$\Delta p = \sigma \left( \frac{1}{R_1} + \frac{1}{R_2} \right). \tag{2.22}$$

The surface tension coefficient $\sigma$ depends on the fluids and phases sharing the interface, the temperature and the presence of surfactants [9], but in case of isothermal immiscible flows without any contamination it may be regarded as constant. Evidently interfacial

phenomena are concentrated only at phase boundaries, therefore the force due to surface tension is modeled as a force per unit volume by

$$f_{i,\sigma} = \int_{\sigma} \sigma \kappa n_{i,S} \delta_S \, \mathrm{d}S, \tag{2.23}$$

with the mean curvature $\kappa$, the unit normal to the interface $n_{i,S}$, the Dirac delta function with respect to the interface $\delta_S = \delta(x_i - x_{i,S})$ and the points $x_{i,S}$ on the interface $\sigma$. Equation (2.23) is known as CSF model proposed by Brackbill et al. [14]. For a general indicator function $C$ over a transition region of finite thickness Equation (2.23) approximates the volume force as

$$f_{i,S} = \sigma \kappa \frac{1}{\Delta C} \frac{\partial C}{\partial x_i}, \tag{2.24}$$

where $\Delta C = C^B - C^A$ is the jump from fluid $A$ to $B$. Considering the volume fraction (2.19) as the interface defining function, the jump from fluid $A$ with $C^A = 0$ to fluid $B$ with $C^B = 1$ results in $\Delta C = 1$ and turns the integral in Equation (2.24) into

$$f_{i,S} = \sigma \kappa \frac{\partial F}{\partial x_i}. \tag{2.25}$$

Hence, to take the surface tension into account, Equation (2.25) is added as a source term to the one-fluid formulation of the Navier-Stokes equations. The final form for immiscible two-phase flows in presence of surface tension becomes

$$\frac{\partial u_i}{\partial x_i} = 0, \tag{2.26}$$

$$\rho \left( \frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) = \rho g_i + \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] - \frac{\partial p}{\partial x_i} + \sigma \kappa \frac{\partial F}{\partial x_i}, \tag{2.27}$$

in which the general body forces $f_i$ are limited to the gravitational acceleration $g_i$ . In order to solve a flow field coupled with acoustics the necessary physical fundamentals are outlined in the next section.

## 2.3 Acoustics

Acoustics is the science of sound, its properties, its origin and propagation, its generation and perception, its measurement and its application. Sound is described as mechanical waves of an elastic medium, which propagate as sound waves through gases, liquids and solids [66]. For example, a sound wave in air is an oscillatory fluctuation of ambient pressure and density traveling through the medium. Hence, the frequency of sound is the number

Table 2.1: Frequency ranges in acoustics [66]

| Frequency range | Label | Examples |
|---|---|---|
| 0 Hz to 20 Hz | Infrasound | Monitoring earthquakes, whale communication |
| 20 Hz to 20 kHz | Audible sound | Human speech, jet noise |
| 20 kHz to 10 GHz | Ultrasound | Sonogram, industrial cleaning |
| 10 GHz to 1 THz | Hypersound | Phononic crystals |

of pressure fluctuations per second, measured in Hertz (Hz). Acoustic sources can emerge from anything that vibrates in a receiver dependent audible frequency range. In Table 2.1 an overview of frequency ranges are given. Not only is sound described by its frequency but also by its pressure fluctuation amplitude and the speed of sound. Möser [85] states that sound fields are almost always described by means of their pressure distribution, which is justified by the simple detection by microphones. In contrast the sound density can only be determined indirectly by the pressure. So, in order to quantify the sound signal the acoustic pressure is measured. The human ear is able to perceive sound pressures from $20 \times 10^{-6}$ Pa to 200 Pa, thereby a logarithmic scale is more handy to represent this wide range [85]. The sound pressure level $L_p$ is given in decibel (dB) and defined by

$$L_p = 20 \log_{10} \left( \frac{\hat{p}}{p_0} \right), \tag{2.28}$$

where the reference pressure $p_0 = 2 \times 10^{-5}$ Pa represents roughly the hearing threshold at a frequency of 1 kHz and $\hat{p}$ expresses the root mean square value of the time domain signal. A small overview of situations and their absolute pressure with equivalent sound pressure level is shown in Table 2.2. The acoustic values in Table 2.2 fluctuate around the ambient pressure, which is comparable high at $10^5$ Pa. This disparity in pressure scales will be

Table 2.2: Sound pressure and corresponding sound pressure level with examples [85]

| Sound pressure $\hat{p}$ (Pa) | Sound pressure level $L_p$ (dB) | Situation |
|---|---|---|
| $2 \times 10^{-5}$ | 0 | hearing threshold |
| $2 \times 10^{-4}$ | 20 | forest, slow winds |
| $2 \times 10^{-3}$ | 40 | library |
| $2 \times 10^{-2}$ | 60 | office |
| $2 \times 10^{-1}$ | 80 | busy street |
| $2 \times 10^{0}$ | 100 | pneumatic hammer, siren |
| $2 \times 10^{1}$ | 120 | jet plane during take-off |
| $2 \times 10^{2}$ | 140 | threshold of pain, hearing loss |

revisited in the following section. A second characteristic of acoustics is the frequency of the pressure fluctuation. Only their combination determines the actual hearing level. For example, a tone at 50 Hz with 80 dB is perceived with the same loudness as a 1 kHz tone with 60 dB [85]. While on the subject, a tone is created by harmonic oscillations of the acoustic pressure whereas noise is produced by highly erratic pressure variations composed by many frequencies combined together. The frequency $f$ can be expressed in terms of the angular frequency $\omega$ or the time period $T$ as

$$f = \frac{\omega}{2\pi} = \frac{1}{T}. \tag{2.29}$$

In non-dispersive media[1] the fraction of the speed of sound and the frequency is the spatial period known as wavelength $\lambda$:

$$\lambda = \frac{c}{f}. \tag{2.30}$$

With the basic quantities (2.28) to (2.30) an acoustic signal can be described. However, typical sounds are complex with many oscillations of varying amplitude and duration and with distinct pattern [98]. Depending on the signal, different characteristics or measurements are advantageous, for example an A-weighting applied to sound pressure levels to account for relative loudness perceived by the human ear. In the following section, the reason for not using the NSE for the acoustics is given. Afterwards, the equations employed for generation and propagation of acoustic quantities are derived.

### 2.3.1 Coupling of Acoustics and Fluid Flow

When dealing with acoustics and fluid flow at low Mach numbers, different time and space scales are present for each discipline. An average audible sound with a frequency of $f = 2000\,\mathrm{Hz}$ traveling through air with a speed of sound of $c = 343\,\mathrm{m\,s^{-1}}$ results in a characteristic length scale given by the wavelength $\lambda = 1.715 \times 10^{-1}\,\mathrm{m}$. In contrast, the problems discussed in this work contain geometries about $1 \times 10^{-4}\,\mathrm{m}$. Besides this length scale disparity [53], the time scales differ by orders of magnitude as well. From the previous mentioned speed of sound of air and the impact velocity of falling water drop given by $U_\mathrm{I} \approx 1\,\mathrm{m\,s^{-1}}$ follows the temporal disparity, which emerges if both velocities are resolved properly. Without any modeling or assumptions, all physical disciplines can be computed with the compressible NSE. A so-called Direct Noise Computation (DNC) requires high numerical resolutions in time and space, ergo consumes much computational resources. Due to the large disparities in the scales, the application of DNC for low Mach number flows is highly inefficient and this topic is understood as multi scale problem. Hence, hybrid methodologies, which separate the treatment of the fluid and the acoustics,

[1] In a dispersive medium different frequencies propagate with different phase velocities. An example for non-dispersive medium is air for acoustic waves.

have been established as the most practical methods for aeroacoustic computations [53]. In those hybrid approaches, a one-way coupling from the fluid flow to the acoustics is generally assumed. That there are no significant physical effects from the acoustics to the fluid field is justified by the disparity of the pressure scales. As mentioned in the previous section, the flow field pressure is about $1 \times 10^4$ times higher than the acoustic pressure. In this work, the hybrid methodology originally developed for the underlying framework by Kornhaas [57] is employed. The derivation of the equations to compute the acoustics is presented in the upcoming section.

### 2.3.2 Equations of Linear Acoustics

Acoustics can be regarded as perturbations with small amplitudes indicated by the superscript ($'$) to an ambient state, known as acoustic/viscous splitting technique [45]. Hence, overall pressure, density and velocity are described as

$$\begin{aligned} p &= p_0 + p', \\ \rho &= \rho_0 + \rho', \\ u_i &= u_{i,0} + u_i'. \end{aligned} \tag{2.31}$$

The definition (2.31) inserted into the compressible Navier-Stokes equations (2.3) and (2.14) without viscosity taking into account, also known as compressible Euler equations:

$$\frac{\partial(\rho_0 + \rho')}{\partial t} + (u_{i,0} + u_i') \frac{\partial(\rho_0 + \rho')}{\partial x_i} + (\rho_0 + \rho') \frac{\partial(u_{i,0} + u_i')}{\partial x_i} = 0, \tag{2.32}$$

$$(\rho_0 + \rho') \frac{\partial(u_{i,0} + u_i')}{\partial t} + \rho(u_{j,0} + u_j') \frac{\partial(u_{i,0} + u_i')}{\partial x_j} = \rho f_i - \frac{\partial(p_0 + p')}{\partial x_i}. \tag{2.33}$$

The terms in Equation (2.32) and (2.33) can be grouped into zero-order, first-order (one primed variable) and second-order (two primed variables). By applying a linearization around a stationary flow field (ambient flow variables are constant) the higher-order terms are neglected, so that Equation (2.32) and (2.33) become

$$\frac{\partial \rho'}{\partial t} + u_{i,0} \frac{\partial \rho'}{\partial x_i} + \rho_0 \frac{\partial u_i'}{\partial x_i} = 0 \tag{2.34}$$

$$\rho_0 \frac{\partial u_i'}{\partial t} + \rho_0 u_{j,0} \frac{\partial u_i'}{\partial x_j} + \frac{\partial p'}{\partial x_i} = 0 \tag{2.35}$$

where Equation (2.34) and (2.35) represent the evolution of acoustic density and velocity, respectively. The acoustic pressure is derived with the equation of state which is determined by thermodynamic properties and relates the changes in pressure. With the definition in Equation (2.31) follows

$$p_0 + p' = p(\rho_0 + \rho', s). \tag{2.36}$$

The propagation of a sound wave in an ideal gas[2] is so fast that an isentropic process[3] entropy $s = constant$ can be assumed. However, by employing a Taylor-series expansion for $p$ as a function of $\rho$ and $s$ in $\rho'$, grouping of terms by their order results in

$$p_0 + p' = p_0 + \left(\frac{\partial p}{\partial \rho}\right)_s (\rho_0 + \rho' - \rho_0) + \left(\frac{\partial^2 p}{\partial \rho^2}\right)_s (\rho_0 + \rho' - \rho_0)^2 + ... \quad , \tag{2.37}$$

with the derivatives evaluated at constant entropy. Further, simplifying and linearizing the expression in (2.37) gives

$$p' = \left(\frac{\partial p}{\partial \rho}\right)_s \rho'. \tag{2.38}$$

With $K$ being the bulk modulus of fluids and gases the speed of sound is given by

$$c = \sqrt{\frac{K}{\rho}}. \tag{2.39}$$

The bulk modulus can be interpreted as the stiffness of a medium,

$$K = -V \frac{\partial p}{\partial V} = \rho \frac{\partial p}{\partial \rho}, \tag{2.40}$$

in which the volume is inversely proportional to the density [124]. Using expression (2.40) in (2.39), the speed of sound becomes [66]

$$c = \sqrt{\left(\frac{\partial p}{\partial \rho}\right)_s} = \sqrt{\frac{\gamma^* p}{\rho}} = \sqrt{\gamma^* RT}, \tag{2.41}$$

and the correlation (2.38) can be written as

$$p' = c^2 \rho'. \tag{2.42}$$

To make use of the correlation in Equation (2.42), the decomposition is differentiated with respect to time, so that

$$\frac{\partial p_0}{\partial t} + \frac{\partial p'}{\partial t} = c^2 \frac{\partial \rho_0}{\partial t} + c^2 \frac{\partial \rho'}{\partial t}. \tag{2.43}$$

In a homogeneous medium, the ambient density does not change for flows with low velocity and according to Shen et al. [117] Equation (2.43) yields

$$\frac{\partial p'}{\partial t} - c^2 \frac{\partial \rho'}{\partial t} = - \frac{\partial p_0}{\partial t}. \tag{2.44}$$

[2]The equation of state of an ideal gas is defined as $p = \rho RT$ with $R$ being the universal gas constant and $T$ being the absolute temperature [124].

[3]A process is described isentropic if the entropy $S$ does not change so that $pV^{\gamma^*} = constant$. For an ideal gas the bulk modulus is only dependent on the pressure and the heat capacity ratio $\gamma^*$ as $K = \gamma^* p$.

The equation for acoustic density (2.34) is placed in formulation (2.44) to obtain the final form for the acoustic pressure as

$$\frac{\partial p'}{\partial t} + u_{i,0} \frac{\partial p'}{\partial x_i} + c^2 \rho_0 \frac{\partial u_i'}{\partial x_i} = - \frac{\partial p_0}{\partial t} \,. \tag{2.45}$$

The equations for acoustic density (2.46), velocity (2.47) and pressure (2.48) are known as Linearized Euler equations (LEE) and in summary shown without the ambient notation for better representation:

$$\frac{\partial \rho'}{\partial t} + u_i \frac{\partial \rho'}{\partial x_i} + \rho \frac{\partial u_i'}{\partial x_i} = 0 \tag{2.46}$$

$$\rho \frac{\partial u_i'}{\partial t} + \rho u_j \frac{\partial u_i'}{\partial x_j} + \frac{\partial p'}{\partial x_i} = 0 \tag{2.47}$$

$$\frac{\partial p'}{\partial t} + u_i \frac{\partial p'}{\partial x_i} + c^2 \rho \frac{\partial u_i'}{\partial x_i} = - \frac{\partial p}{\partial t} \,. \tag{2.48}$$

These were originally derived by Hardin et al. [45] and adapted by Shen et al. [116, 117] for determining aerodynamically generated acoustic fields for low Mach number flows. The temporal derivative of the pressure on the right hand side of Equation (2.48) is the source term for the acoustics, which originates directly from the previously calculated incompressible flow field. The approach of superposition the acoustic perturbations are generally known as Expansion about Incompressible Flow (EIF). Different derivations can be found in the literature, such as Shen et al. [117], Slimon et al. [118], Seo et al. [113] or Ewert et al. [29].

### 2.3.3 Sound Generation

In general, acoustic signals are generated by forcing the particles in the underlying medium to move around their position of rest causing a time and space dependent oscillation of the density [30]. The force emerges from pressure differences acting on the medium, which in turn have different sources.

#### Structural Acoustics

If mechanical waves in structures radiate sound into the surrounding acoustic fluid or the vibrating structure itself emits sound, both are often referred to as vibroacoustics. Typcial examples are the cones on a speaker which are vibrating structures that radiate sound into the air [43] or the swinging strings of a musical instrument. A modeling approach is that this kind of sound generation is based on the assumption that the normal velocity

at the structure surface is equal to the acoustic velocity of the corresponding neighboring fluid particle [44]. However, the moving structure compresses and expands the surrounding medium leading to an acoustic signal.

### Aeroacoustics

By means of aeroacoustics the sound is generated by the flow and its pressure disturbances. In 1952 Lighthill [69] developed his theory for aerodynamically generated sound, initiating today's Computational Aeroacoustics (CAA) with his definition of the acoustic source terms. The physical acoustic sources are reduced to simple types of emitters, which is why the theory by Lighthill is often referred as acoustic analogies. According to Lighthill the three basic sound generation mechanisms are: monopole, associated with fluctuating mass injection or volume flow, e.g. a loudspeaker in a box; dipole, associated with unsteady external forces or fluid pressure on a solid boundary, e.g. a loudspeaker without a box; and quadrupole, associated with an unsteady Reynolds stress, e.g. free turbulent flows [3].

### Multiphase Acoustics

In case of multiple phases the pressure fluctuation is caused by phase transition, e.g. implosion of cavity bubbles, by coalescence and breakup or by the movement of an interface between phases. The latter can be observed by oscillating gas bubbles in a liquid medium. A combination of the mechanisms is found in the sound emitted by crashing waves, which is characterized by the creation of bubbles and their sizes, hence their oscillation frequencies [18].

# 3. Numerical Framework

The accuracy of a Computational Fluid Dynamics (CFD) computation is determined by the discretization methods of the domain and the equations. In this chapter the numerical methods used for this work are introduced. Every implementation is done in the in-house flow solver FASTEST [25]. The previous introduced Navier-Stokes Equations form the base of FASTEST, which has been developed over several decades and therefore includes a wide range of functionalities such as different turbulence models [51], optimization [83], Fluid-Structure Interaction (FSI) [109] or the coupling of disciplines (Thermal-Fluid-Structure Interaction (TFSI) [99], Fluid-Structure-Acoustic Interaction (FSA) [55]) for dealing with complex three dimensional configurations.

## 3.1 Domain Discretization

For a successful CFD computation the preparation of the Computer Aided Design (CAD) geometry and its discretization is crucial for producing reliable results. There are multiple ways of dividing the continuous geometry into a grid of elements. FASTEST uses the fully conservative Finite Volume Method (FVM) on block structured collocated grids. In collocated grids all variables are stored in the center, whereas in staggered grids scalar quantities are stored in the center and velocity components are stored at faces of a Control Volumes (CV). By using hexahedron elements in three dimensional space[1], a regular connectivity of neighboring elements is secured. This attribute of structured grids allows simple and efficient implementations as well as fast memory access. The elements or in this case Control Volumes (CV) are body-fitted to the (irregular) physical boundary and its coordinates $x_i$ are mapped into a (regular) Cartesian system with logical coordinates $\xi_i$ as

$$x_i = x_i(\xi_j) \quad \text{or} \quad \xi_j = \xi_j(x_i), \quad \text{with} \quad i,j = 1,2,3. \tag{3.1}$$

[1] If a computation provides only a two-dimensional domain, hexahedron elements are used as well. The third dimension is set to unity, making the simulation quasi two dimensional.

A unique mapping is mandatory and can be achieved by an algebraic or an elliptic approach, for example by a transfinite interpolation or by solving an elliptic differential equation, respectively [110]. Cell nodes are in the center of each CV and denoted by capital letters, while cell faces are stated by small letters. For example, starting from cell center P, face e is shared with neighbor E in the east.
If the domain is decomposed into subdomains along the block boundaries for parallel computations with distributed memory, one layer of ghost cells is employed in which variables are exchanged. As in Figure 3.1, which describes the topology of FASTEST for two dimensions, next neighbor EE on processor 0 is a ghost cell receiving its values from the equivalent CV on processor 1 across the subdomain interface. In an identical notation,

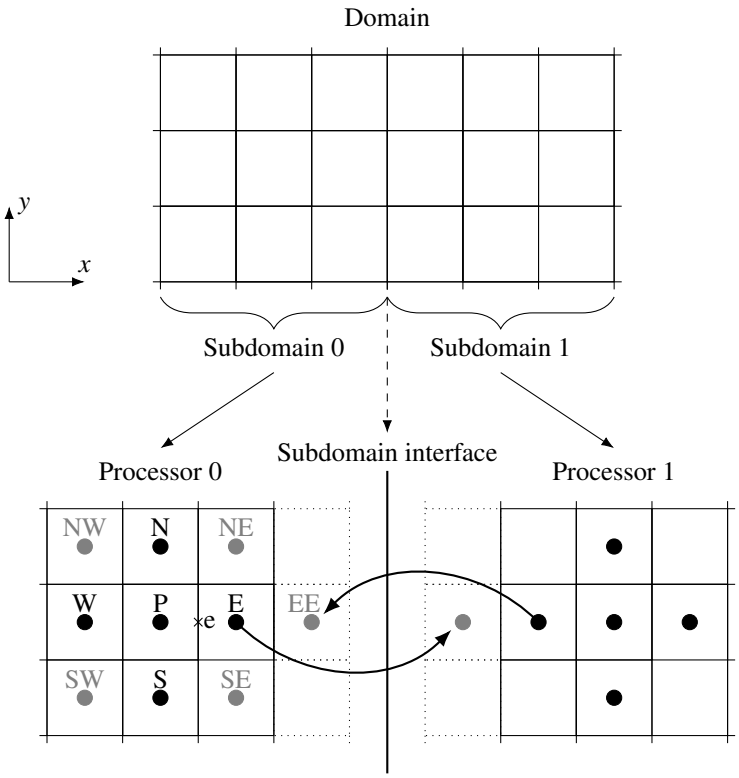



Figure 3.1: Two dimensional schema of FASEST topology at block boundaries

appending the third dimension results in CVs in positive direction with top T and in negative direction with bottom B. However, from an application engineer's point of view, the chosen grid generation method should accurately represent the geometry as well as fit the solution methods and modeling approaches.

## 3.2 Equation Discretization

After the determination of the physical models in Chapter 2, the appropriate numerical methods for the underlying differential equations are presented. By employing the FVM for the spatial and the Finite Difference Method (FDM) for the temporal discretization a non-linear algebraic system is obtained and needs to be solved.

### 3.2.1 Finite Volume Method

As stated in Section 3.1, the domain is divided into a finite set of control volumes. A generic transport equation for variable $\phi$ serves as an example for the procedure consisting of the temporal change, the convective transport, the diffusive transport with constant diffusion coefficient $\Gamma_\phi$ and a source term $f_\phi$. First, for each CV the integral balances are formulated:

$$\int_V \frac{\partial(\rho\phi)}{\partial t}\,\mathrm{d}V + \int_V \frac{\partial(\rho\phi)u_i}{\partial x_i}\,\mathrm{d}V = \int_V \frac{\partial}{\partial x_i}\left(\Gamma_\phi \frac{\partial(\rho\phi)}{\partial x_i}\right)\mathrm{d}V + \int_V f_\phi\,\mathrm{d}V. \tag{3.2}$$

Applying the Gauss theorem, which reformulates volume integrals of spatial gradients into integrals at volume surfaces with corresponding unit normal vectors, leads to the following equation

$$\int_V \frac{\partial(\rho\phi)}{\partial t}\,\mathrm{d}V + \int_S (\rho\phi)u_i n_i\,\mathrm{d}S = \int_S \left(\Gamma_\phi \frac{\partial(\rho\phi)}{\partial x_i}\right) n_i\,\mathrm{d}S + \int_V f_\phi\,\mathrm{d}V. \tag{3.3}$$

Since the domain discretization is based on hexahedron elements, the surface integrals are summed over all surfaces $S_c$ denoted by $c = e, w, n, s, t, b$ so that Equation (3.3) can be written equivalently

$$\int_V \frac{\partial(\rho\phi)}{\partial t}\,\mathrm{d}V + \sum_c \int_{S_c} (\rho\phi)u_i n_i\,\mathrm{d}S_c = \sum_c \int_{S_c} \left(\Gamma_\phi \frac{\partial(\rho\phi)}{\partial x_i}\right) n_i\,\mathrm{d}S_c + \int_V f_\phi\,\mathrm{d}V. \tag{3.4}$$

The integral formulation applies on every CV as in equation (3.4) and on the whole domain. So if summation is done over the whole domain, the integrals of the inner CV faces cancel each other out and the global conservation is preserved. This is the main advantage of the FVM and the reason for its broad usage in CFD [31].
In the next step, the surface and volume integrals are approximated by numerical integration. The simplest approximation is the midpoint rule, for which the surface integrals are expressed as values on the CV face midpoint, whereas volume integrals are approximated by the variables in the CV node. Although the midpoint rule achieves second order for a surface integral, other approximations like the trapezoidal rule or the simpson rule, which

include more values of the integrated entity, can be employed as well but are not considered here. Therefore, the convective transport at any face $c$ approximated with the midpoint rule gives

$$\int_{S_c} (\rho\phi) u_i n_i \, \mathrm{d}S_c \approx (\rho\phi u_i n_i)_c \Delta S_c, \tag{3.5}$$

which expresses the component of the convective transport normal to the cell face $c$ with surface area $\Delta S_c$. The same procedure is carried out for the diffusive transport as

$$\int_{S_e} \left( \Gamma_\phi \frac{\partial(\rho\phi)}{\partial x_i} \right) n_i \, \mathrm{d}S_e \approx \left( \Gamma_\phi \frac{\partial(\rho\phi)}{\partial x_i} \right)_e \Delta S_e, \tag{3.6}$$

and the for the source term as

$$\int_V f_\phi \, \mathrm{d}V \approx (f_\phi)_P \Delta V. \tag{3.7}$$

Assembling each part of initial balance equation (3.2) after integral approximation yields

$$\underbrace{\left( \frac{\partial(\rho\phi)}{\partial t} \right)_P \Delta V}_{\text{transient}} + \underbrace{\sum_c (\rho\phi u_i n_i)_c \Delta S_c}_{\text{conv. transport}} - \underbrace{\sum_c \left( \Gamma_\phi \frac{\partial(\rho\phi)}{\partial x_i} \right)_c \Delta S_c}_{\text{diff. transport}} = \underbrace{(f_\phi)_P \Delta V}_{\text{source term}}, \tag{3.8}$$

in which the integral of temporal derivative is approximated in the same manner as for the source term. For the final part of the FVM discretization procedure the temporal derivative, the scalars and the spatial derivatives in the CV center and faces need to be discretized.

### 3.2.2 Temporal Discretization

Although the temporal derivative can be understand as a FDM with discrete points in time [31], it is presented here. The temporal derivative can be rewritten into a source term $f(t, \phi)$ containing convective, diffusive and source term following

$$\int_V \frac{\partial(\rho\phi)}{\partial t} \, \mathrm{d}V = -\int_V \frac{\partial(\rho\phi) u_i}{\partial x_i} \, \mathrm{d}V + \int_V \frac{\partial}{\partial x_i} \left( \Gamma_\phi \frac{\partial(\rho\phi)}{\partial x_i} \right) \mathrm{d}V + \int_V f_\phi \, \mathrm{d}V = \int_V f(t, \phi) \, \mathrm{d}V. \tag{3.9}$$

Employing the midpoint rule for the volume integral yields

$$\frac{\partial(\rho\phi)}{\partial t} \Delta V = f(t, \phi) \Delta V. \tag{3.10}$$

There are two main classes of temporal discretization techniques. If the approximation is carried out at the initial time step $t_n$ with $n$ being the time step index, the explicit Euler method is obtained by

$$\frac{\partial(\rho\phi)}{\partial t}(t_n) \approx \frac{(\rho\phi)^{n+1} - (\rho\phi)^n}{\Delta t} = f(t_n, \phi^n). \tag{3.11}$$

The time step size $\Delta t_n = \Delta t_{n+1} = \Delta t$ is held constant. In contrary using the end value $t_{n+1}$ for the approximation results in the implicit Euler method as

$$\frac{\partial(\rho\phi)}{\partial t}(t_{n+1}) \approx \frac{(\rho\phi)^{n+1} - (\rho\phi)^n}{\Delta t} = f(t_{n+1}, \phi^{n+1}). \tag{3.12}$$

If a linear change is assumed over the time interval both methods can be combined resulting in the principle of the Crank-Nicolson method

$$\frac{\partial(\rho\phi)}{\partial t}(t_{n+1/2}) \approx \frac{(\rho\phi)^{n+1} - (\rho\phi)^n}{\Delta t} = \frac{1}{2}(f(t_n, \phi^n) + f(t_{n+1}, \phi^{n+1})). \tag{3.13}$$

For example, the unknown $(\rho\phi)$ at the next time step $t_{n+1}$ with implicit Euler method is determined by

$$(\rho\phi)^{n+1} = (\rho\phi)^n + f(t_{n+1}, \phi^{n+1})\Delta t. \tag{3.14}$$

Explicit methods tend to get unstable, if the time step is not small enough for the underlying grid. A dimensionless parameter called Courant number $Co$ (Courant-Friedrichs-Lewy (CFL) number) characterizes this behavior and is defined as

$$Co_{i,n} = \frac{u_i \Delta t}{\Delta x_i}. \tag{3.15}$$

For $\max(Co_{i,n}) < 1$ an explicit method is stable and does not diverge. To obtain a solution with implicit methods a system of equations has to be solved, which results from the new values on the right hand side of Equation (3.12). Therefore, stability is not an issue, yet memory consumption and implementation complexity increases. Both, implicit and explicit Euler method are first order accurate while the Crank-Nicolson method can be seen as Central Differencing Scheme (CDS) in time and thus has second order temporal accuracy [110].
By including additional points between $t_n$ and $t_{n+1}$ higher order methods can be derived. The Runge-Kutta method of order four consists of the following steps to obtain the un-

known at the new time step:

$$(\rho\phi)^{*}_{n+1/2} = (\rho\phi)^{n} + \frac{\Delta t}{2}\, f(t_n, \phi^n), \tag{3.16}$$

$$(\rho\phi)^{**}_{n+1/2} = (\rho\phi)^{n} + \frac{\Delta t}{2}\, f(t_{n+1/2}, \phi^{*}_{n+1/2}), \tag{3.17}$$

$$(\rho\phi)^{*}_{n+1} = (\rho\phi)^{n} + \Delta t\ \ f(t_{n+1/2}, \phi^{**}_{n+1/2}), \tag{3.18}$$

$$(\rho\phi)^{n+1} = (\rho\phi)^{n} + \frac{\Delta t}{6}[f(t_n, \phi^n) + 2f(t_{n+1/2}, \phi^{*}_{n+1/2}) \tag{3.19}$$

$$+ 2f(t_{n+1/2}, \phi^{**}_{n+1/2}) + f(t_{n+1}, \phi^{*}_{n+1})]. \tag{3.20}$$

Due to the multiple evaluation and storage of the system $f(t, \phi)$, the Runge-Kutta method of higher orders are intensive with respect to computation time [31].

### 3.2.3 Interpolation Techniques

Even though some of the techniques in the upcoming subsections are well known basic principles, they are presented briefly due to several mentions throughout this work. First the methods for approximating scalar values at face centers are presented. Despite the standard methods for spatial derivatives, a coordinate transformation is briefly explained. For a more detailed description the reader is referred to [141, 110, 31].

#### Upwind Differencing Scheme (UDS)

The first approximation method depends on the direction of the convective transport in Equation (3.8). The scalar value on the surface $\phi_e$ is taken from either of the adjacent nodes as

$$\phi_e = \begin{cases} \phi_P, \text{ if } (u_i n_i)_e > 0, \\ \phi_E, \text{ if } (u_i n_i)_e < 0, \end{cases} \tag{3.21}$$

which corresponds to a first-order backward or forward differencing scheme. Evaluating the Taylor series expansion of $\phi_P$ at location $x_{i,e}$ shows that the Upwind Differencing Scheme (UDS) is unconditionally stable yet numerically diffusive. The numerical diffusion increases if the flow direction deviates from the face normal and very fine grids would be necessary for a solution of proper accuracy.

#### Central Differencing Scheme (CDS)

A second order approximation of the face value $\phi_e$ can be achieved by the CDS of gradients with

$$\left(\frac{\partial \phi}{\partial x}\right)_i \approx \frac{\phi_{i+1} - \phi_{i-1}}{x_{i+1} - x_{i-1}}. \tag{3.22}$$

The values of the two adjacent center nodes $P$ and $E$ are linear interpolated with a geometric interpolation factor $\gamma_e$, so that Equation (3.22) can be formally written for the face value as

$$\phi_e = \phi_E \gamma_e + \phi_P (1 - \gamma_e) \text{ , with } \gamma_e = \frac{x_e - x_P}{x_E - x_P}. \tag{3.23}$$

With the increased order, in contrast to the UDS, follows a less stable behavior causing numerical oscillations. However, second order accuracy holds for equidistant and non-equidistant grids [110]. If the grid is distorted and non-orthogonal, more neighboring CV center are incorporated for keeping the second order. The scheme for this case is the Multidimensionale Lineare Interpolation (MuLI), which is given by

$$\phi_e = \phi_E \gamma_e + \phi_P (1 - \gamma_e) + \gamma_{ns}(\phi_N - \phi_S) + \gamma_{tb}(\phi_T - \phi_B). \tag{3.24}$$

Further details and the interpolation factors can be found in [62].

### High Resolution Schemes (HR Schemes)

As well as for the velocities, the multiphase and the acoustic part, more advanced numerical methods are employed. In general, numerical methods of a higher order tend to oscillate and lead to numerical instabilities while methods of lower order introduce numerical dissipation [68]. High resolution schemes are characterized by the combination of a lower order and a higher order method to get an accurate yet bounded solution. There are two approaches on which a cell face value can be determined: schemes based on the Normalized Variable Diagram (NVD) and Total Variation Diminishing (TVD) schemes [52]. Both use indicators to identify, if the boundedness of a scheme is preserved in the cell of interest. The required high resolution schemes are explained in here and will be referenced in the following sections. For creating the Normalized Variable Diagram (NVD), cell $C$ (central) and cell face $f$ of a convected scalar $\phi$ in a one-dimensional case are normalized according to

$$\tilde{\phi} = \frac{\phi - \phi_U}{\phi_D - \phi_U}, \tag{3.25}$$

where $U$ stands for upstream and $D$ for downstream [64, 52]. Which neighboring cell is down- or upstream depends on the flow direction, and an exemplary representation of the naming convention can be seen in Figure 3.2. If either condition

$$\phi_U \leq \phi_C \leq \phi_D \quad \text{or} \tag{3.26}$$

$$\phi_U \geq \phi_C \geq \phi_D \tag{3.27}$$

is fulfilled in every CV, the solution will be without any unphysical oscillations. In terms of the normalized variables, the criterion is known as Convective-Boundedness-Criterion (CBC) [40] with

$$0 \leq \tilde{\phi}_C \leq 1, \tag{3.28}$$

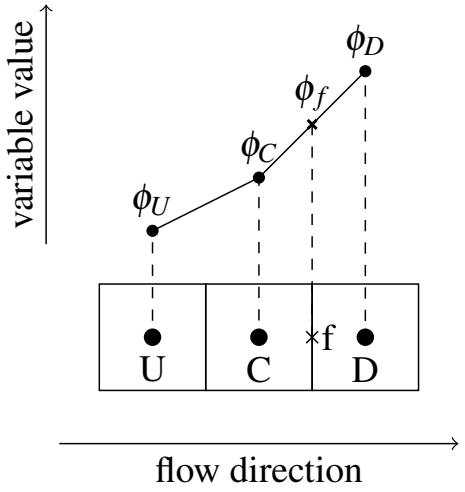



Figure 3.2: Naming convention in the context of NVD and Total Variation Diminishing (TVD)

bounded by $\phi_C = \phi_U$ and $\phi_C = \phi_D$. The normalized cell face value obtained via Equation (3.25) can be rewritten into

$$\phi_f = (1 - \tilde{\beta}_f)\phi_C + \tilde{\beta}_f \phi_D \quad \text{with} \tag{3.29}$$

$$\tilde{\beta}_f = \frac{\tilde{\phi}_f - \tilde{\phi}_C}{1 - \tilde{\phi}_C}. \tag{3.30}$$

In the actual NVD the normalized cell face $\tilde{\phi}_f$ is plotted against the normalized center cell $\tilde{\phi}_C$. The hatched area in Figure 3.3 including the line $\tilde{\phi}_f = \tilde{\phi}_C$ represents the CBC, which means that for any differencing scheme located inside the boundedness is fulfilled. For the convection of the flow velocities the Xue-MUSCL scheme with QUICK coefficients ($\kappa = 0.5$) is employed [86, 56, 146], which gives the face values as

$$\tilde{\phi}_{f,\text{XUE-MUSCL}} = \begin{cases} 2\tilde{\phi}_U & \text{for } 0 \leq \tilde{\phi}_U < \frac{1+\kappa}{4+2\kappa}, \\ (1 - \frac{\kappa}{2})\tilde{\phi}_U + \frac{1+\kappa}{4} & \text{for } \frac{1+\kappa}{4+2\kappa} \leq \tilde{\phi}_U \leq \frac{3-\kappa}{4-2\kappa}, \\ 1 & \text{for } \frac{3-\kappa}{4-2\kappa} < \tilde{\phi}_U \leq 1, \\ \tilde{\phi}_U & \text{elsewhere.} \end{cases} \tag{3.31}$$

For the volume fraction transport the Modified Compressive Interface Capturing Scheme for Arbitrary Meshes (M-CICSAM) [141] is employed, which blends a steady version of the Steady Hyper-C (SHC) scheme [65] and the Fromm scheme [39] upon the interface orientation. The blending factor $\gamma_f$ depends on the angle $\alpha_f$ between the unit normal to

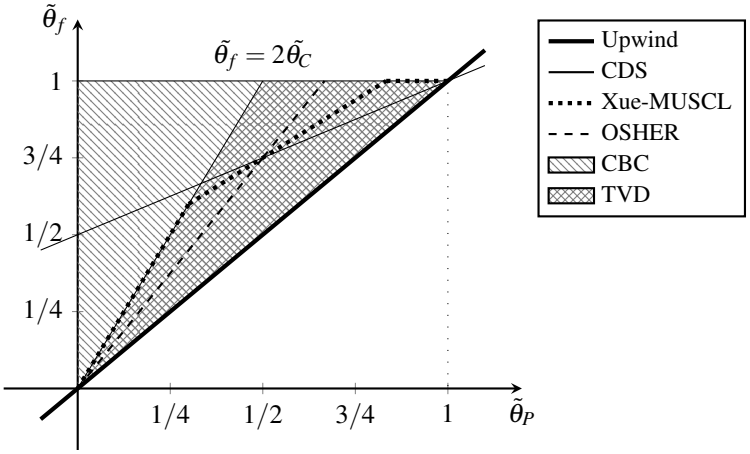



Figure 3.3: NVD with Upwind, CDS, Xue-MUSCL and OSHER

the interface and the unit vector between the cell centers in flow direction.

$$\alpha_f = \arccos\left|\frac{\nabla F_{P,i}}{\|\nabla FP,i\|} \cdot \frac{(x_{U,i} - x_{P,i})}{\|(x_{U,i} - x_{P,i})\|}\right|, \tag{3.32}$$

$$\gamma_f = \| \cos \alpha_f \|^{\frac{1}{4}}, \tag{3.33}$$

$$\tilde{\phi}_f = = \gamma_f \tilde{\phi}_{\text{SHC}} + (1 - \gamma_f)\tilde{\phi}_{\text{Fromm}}. \tag{3.34}$$

In case the interface unit normal of the flow is parallel to the unit vector of the CV centers between the face of interest, Equation (3.35) determines the face value. The SHC, which does not rely on the Courant number in contrast to the Hyper-C scheme, has a compressive character which keeps the interface compact.

$$\tilde{\phi}_{f,\text{SHC}} = \begin{cases} \min(1, 2\tilde{\phi}_U) & \text{for } 0 \leq \tilde{\phi}_U \leq 1, \\ \tilde{\phi}_U & \text{elsewhere.} \end{cases} \tag{3.35}$$

For the situation that the unit normal of the fluid is perpendicular to the unit vector of the CV centers, the second order accurate bounded linear Fromm Scheme in Equation (3.36) sets the face value:

$$\tilde{\phi}_{f,\text{Fromm}} = \begin{cases} \min(\frac{1}{4} + \tilde{\phi}_U, \tilde{\phi}_{SHC}) & \text{for } 0 \leq \tilde{\phi}_U \leq 1, \\ \tilde{\phi}_U & \text{elsewhere.} \end{cases} \tag{3.36}$$

A more comprehensive study and comparison of the NVD-schemes in general or for the underlying solver FASTEST can be found in [86, 139, 81, 121].
If a scheme should satisfy the more constrained TVD condition as well, it needs to stay in the crosshatched area in Fig 3.3. The NVD and TVD formulations can be transformed

into each other. Depending on the origin of the scheme it is mostly described in its original framework. The basis for developing a High Resolution (HR)-scheme in the TVD framework is the restriction that the Total Variation (TV) can not increase from one time step to the following:

$$TV(\phi^{t+\Delta t}) \leq TV(\phi^t). \tag{3.37}$$

The TV is defined as

$$TV = \sum_i |\phi_{i+1} - \phi_i|, \tag{3.38}$$

in which index $i$ counts over the spatial CV in the domain. Instead of using normalized variables, the flux through a face is limited by a scheme dependent flux-limiter $\psi(r_f)$ with a gradient coefficient

$$r_f = \frac{\phi_C - \phi_U}{\phi_D - \phi_C}, \tag{3.39}$$

so that the face value is calculated as

$$\phi_f = \phi_C + \frac{1}{2}\psi(r_f)(\phi_D - \phi_C). \tag{3.40}$$

A relation between the NVD and TVD formulations can be established through [81]

$$\tilde{\phi}_C = \frac{r_f}{1 + r_f}. \tag{3.41}$$

If a limiter lies within the boundaries of $0 \leq \psi(r_f) \leq 2r_f$ it satisfies the more restrictive TVD-CBC, which leads to $\tilde{\phi}_f = 2\tilde{\phi}_C$ in the NVD formulation [130]. Exemplary the Osher limiter [89] ($1 \leq \beta \leq 2$) is given by

$$\psi(r_f) = \max(0, \min(\beta, r_f)). \tag{3.42}$$

Since the TVD concept has been introduced in the 1980's [46, 47, 89, 130], a wide range of limiter are present in the literature and have been compared for the acoustic development of the in-house flow solver FASTEST [57].

### Derivatives through transformation in local coordinates

In FASTEST, the derivatives are obtained either with the Coordinate Transformation Scheme (CTS) or the Derivative Approximation Based on Multi Dimensional Taylor Series Expansion (DABT) in order to improve convergence and robustness properties of the pressure-correction scheme (Section 3.4.1) on non-orthogonal grids [61]. Both methods have the same structure but differ in terms of approximation points and interpolation coefficients. Since the CTS is employed for the pressure and of importance for the surface tension it is briefly explained.

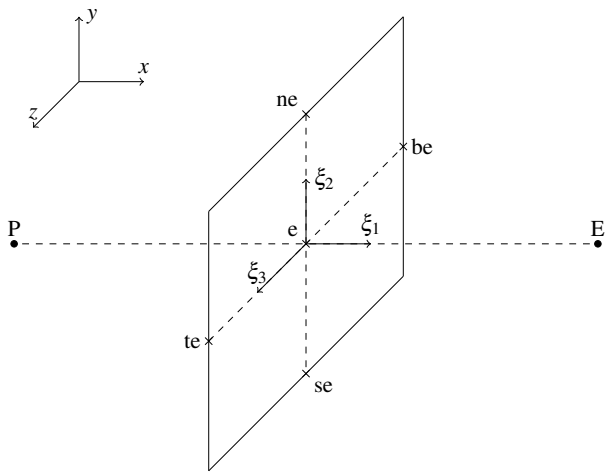



Figure 3.4: Local coordinates on the east face of a CV

As previously stated in Section 3.1, the global coordinates $x_i = (x_1, x_2, x_3)$ are transformed into local coordinates $\xi_j = (\xi_1, \xi_2, \xi_3)$ at each CV face to handle the irregular physical domain (often referred as complex geometry) in combination with structured grids. The correlation between the local and the global coordinates is shown in Figure 3.4. The transformation for a generic scalar $\phi$ reads as

$$\frac{\partial \phi}{\partial x_i} = \frac{\partial \phi}{\partial \xi_j}\frac{\partial \xi_j}{\partial x_i} = \frac{\partial \phi}{\partial \xi_j}\frac{\beta_{ij}}{J}, \tag{3.43}$$

with Jacobian $J = \det(\partial x_i \,/\, \partial \xi_j)$ and $\beta_{ij}$ representing the cofactor matrix to $(\partial x_i \,/\, \partial \xi_j)$ in the Jacobian matrix [31]. With the transformation, the derivative at a cell face can be approximated as [61, 141]

$$\left(\frac{\partial \phi}{\partial x_i}\right)_f \approx \frac{b^f_{ji}}{J\delta V}\Delta_j \phi, \quad i, j = 1, 2, 3. \tag{3.44}$$

An auxiliary CV is built in the local coordinate system $\delta V = \Delta\xi_1 \Delta\xi_2 \Delta\xi_3$ and $b_{ji}$ is derived by CDS at the cell face (denoted by superscript $f$) given by

$$b^f_{ji} = \begin{pmatrix} \Delta_2 y \Delta_3 z - \Delta_2 z \Delta_3 y & \Delta_2 z \Delta_3 x - \Delta_2 x \Delta_3 z & \Delta_2 x \Delta_3 y - \Delta_2 y \Delta_3 x \\ \Delta_3 y \Delta_1 z - \Delta_1 y \Delta_3 z & \Delta_1 x \Delta_3 z - \Delta_3 x \Delta_1 z & \Delta_1 y \Delta_3 x - \Delta_3 y \Delta_1 x \\ \Delta_1 y \Delta_2 z - \Delta_1 z \Delta_2 y & \Delta_1 z \Delta_2 x - \Delta_1 x \Delta_2 z & \Delta_1 x \Delta_2 y - \Delta_1 y \Delta_2 x \end{pmatrix}. \tag{3.45}$$

In order to increase the readability, the global coordinates $x_i$ have been replaced by $(x, y, z)$. The delta variable $\Delta$ stands for the geometric difference in the local coordinate system [141]

Table 3.1: Replacement of the variables in the generic transport variables to obtain the NSE

| Equation | $\phi$ | $\Gamma_\phi$ | $f_\phi$ |
|---|---|---|---|
| continuity | $u_i$ | 0 | 0 |
| momentum | $u_i$ | $\mu$ | $\frac{\partial}{\partial x_j}\mu\left(\frac{\partial u_j}{\partial x_i}\right) - \frac{\partial p}{\partial x_i} + \rho g_i + \sigma\kappa\frac{\partial F}{\partial x_i}$ |
| volume fraction | $F$ | 0 | 0 |

or exemplary for an east face

$$\Delta_1 x_i = x_{E,i} - x_{P,i} \tag{3.46}$$
$$\Delta_2 x_i = x_{ne,i} - x_{se,i} \tag{3.47}$$
$$\Delta_3 x_i = x_{te,i} - x_{be,i}. \tag{3.48}$$

The first local coordinate $\xi_1$ points perpendicular to the face while the others are along the face area, see Figure 3.4. The corner points $ne, se, te, be$ are approximated by interpolation of the surrounding CV centers. Matrix $b_{ij}$ also contains the face area resulting from the cross product of $\Delta_2 x_i$ with $\Delta_3 x_i$, which can be found in the first row. By transforming the generic transport equation (3.8) every term is expressed as the sum of three components.

### 3.2.4 Navier-Stokes Equations Discretization

Since the NSE contain the same elements as the generic transport equation (3.2), the numerical methods presented can be employed. By replacing the generic variable $\phi$ with the variables given in Table 3.1, the NSE represented by the continuity equation (2.26) and the momentum equation (2.27) are obtained. Although the transformation in local coordinates is employed, the discretization of the transient term as well as the convective transport term have the same form as in the generic transport equation. However, the diffusive flux $F^D$ of the momentum in the NSE becomes more complex. The first term of the diffusive flux contains an implicit treated part and an explicit treated part which is added to the source term. The second term is exclusively treated explicitly and therefore, added to the source term. Its contribution vanishes if the viscosity $\mu$ is constant or if the underlying grid is orthogonal [31, 141].
Aside from the explicit parts of the diffusive flux and the part from time discretization, the source term $f_\phi$ contains the gravitational acceleration, a negative pressure gradient and the surface tension. An accurate computation in the presence of fluids with different material properties complicates the discretization at this point. At the interface between the different fluids, an abrupt jump in body forces occur. If this discontinuity is not

modeled appropriate, unphysical spikes in the velocity field appear and can lead to a diverging solution [78]. Consider a zero velocity field $u_i = 0$, then the momentum equation of the NSE in the one-fluid formulation reduces to

$$0 = -\frac{\partial p}{\partial x_i} + \underbrace{\rho g_i + \sigma\kappa \frac{\partial F}{\partial x_i}}_{f_i},$$

$$\frac{\partial p}{\partial x_i} = f_i. \tag{3.49}$$

In order to maintain the zero velocity flow field, the discretization of Equation (3.49) needs to be in balance on both the CV center and faces [78]. Employing the local coordinate system at cell faces and its derivative approximation yields

$$\left(\frac{\partial p}{\partial x_i}\right)_f \approx \frac{b^f_{ji}}{J\delta V}\Delta_j p_f. \tag{3.50}$$

With the three unknowns of $\Delta_j p_f$, a system of linear equations emerges from Equations (3.49) and (3.50) in

$$b^S_{ji}\Delta_j p_f = J\delta V f_{i,f}. \tag{3.51}$$

By employing Cramer's rule, the solution can be found through

$$\Delta_j p_f = \left(\frac{W_j}{W}\right)_f \quad \text{, with} \tag{3.52}$$

$$W = \det(b^S_{ji}) \tag{3.53}$$

$$W_j = \det(\tilde{b}^S_{ji}). \tag{3.54}$$

In the deduced matrix $\tilde{b}^S_{ji}$, the j-th column of $b^S_{ji}$ is replaced by the vector $J\delta V f_{i,f}$. The discretization of Equation (3.50) in cell centers reads

$$f_{i,P} = \left(\frac{\partial p}{\partial x_i}\right)_P \approx \frac{b^V_{ji}}{J}\Delta_j p_P = \frac{1}{J}[b^V_{1i}(p_e - p_w) + b^V_{2i}(p_n - p_s) + b^V_{3i}(p_t - p_b)], \tag{3.55}$$

and with linear interpolation in combination with the discretization derivation of the pressure gradient on cell faces, the final expression is

$$\begin{aligned} f_{i,P} =& \frac{b^V_{1i}}{J}\left[\gamma_e\left(\frac{W_i}{W}\right) + \gamma_w\left(\frac{W_i}{W}\right)\right] \\ &+ \frac{b^V_{2i}}{J}\left[\gamma_n\left(\frac{W_i}{W}\right) + \gamma_s\left(\frac{W_i}{W}\right)\right] \\ &+ \frac{b^V_{3i}}{J}\left[\gamma_t\left(\frac{W_i}{W}\right) + \gamma_b\left(\frac{W_i}{W}\right)\right]. \end{aligned} \tag{3.56}$$

The difference to the standard midpoint rule is that every component of the volume force in (3.56) depends on the other components as well. A more comprehensive description can be found in [141, 121, 76].
After describing the momentum equation of the NSE, the continuity equation follows. Since incompressibility is assumed, the continuity equation (2.3) does not contain any information about the pressure and is discretized as

$$\int_S \frac{\partial u_i}{\partial x_i}\,\mathrm{d}S \approx \sum_c (u_i n_i)_c \Delta S_c = \sum_c \frac{\dot{m}_c}{\rho_c}, \tag{3.57}$$

which can be reformulated into mass fluxes. The coupling between the momentum and continuity equation used in this work, will be presented in Section 3.4.

## 3.3 Volume of Fluid Discretization

The volume fraction transport equation in (2.21) requires temporal and spatial discretization. As part of the FVM the integral formulation after applying the Gauss theorem yields

$$\int_V \frac{\partial F}{\partial t}\,\mathrm{d}V + \int_S F u_i n_i\,\mathrm{d}S = 0. \tag{3.58}$$

Following the work of Ubbink and Issa [135], the Crank-Nicolson method described in Section 3.2.2 is employed and the discretized volume fraction transport reads as

$$\frac{F_\mathrm{P}^{n+1} - F_\mathrm{P}^n}{\Delta t}\Delta V_\mathrm{P} = -\sum_{f=1}^{n_\mathrm{b}} \frac{1}{2}(F_f^n u_{i,f}^n \Delta S_f + F_f^{n+1} u_{i,f}^{n+1} \Delta S_f), \tag{3.59}$$

where index $n$ denotes the different time levels and $S_f$ is the face area. The assumption that the variation of the velocity field does not influence the volume fraction transport during one time step is reasonable, when the time step size is small enough [135]. Subsequently $u_{i,f}^{n+1} = u_{i,f}^n$ and Equation (3.59) can be reduced and rearranged for the unknown new value of the volume fraction as

$$F_\mathrm{P}^{n+1} = F_\mathrm{P}^t - \frac{\Delta t}{\Delta V_\mathrm{P}} \sum_{f=1}^{n_\mathrm{b}} \frac{1}{2}(F_f^n + F_f^{n+1}) u_{i,f}^n \Delta S_f. \tag{3.60}$$

To guarantee a bounded solution while maintaining the sharpness of the interface, HR schemes from Section 3.2.3 are necessary to calculate the face values of the volume fractions $F_f$. Here, the M-CICSAM [141] consisting of the compressive Hyper-C-Scheme [65] and the high-order FROMM-Scheme [39] is employed. The blending of these two schemes depends on the angle between the normal vector of the interface and the connecting line of the donor and the acceptor CV center. Comparisons between different HR-schemes can be found in [135, 139].

## 3.4 Numerical Solution

In order to find the unknown variables $u_i, p, F$ in the equations (2.26), (2.27) and (2.21) the particular discretization methods from the previous sections are employed. At certain stages throughout a time step a discrete system of equations needs to be solved. The discretization of every CV in the domain for an unknown $\phi$ is assembled into

$$a_p^{\phi,i}\phi_P^i + \sum_c a_c^{\phi,i}\phi_c^i = b_p^i \quad \text{for all} \quad i = 1, ..., N, \tag{3.61}$$

with $N$ being the number of CV in the domain, $a$ being the corresponding coefficient and $b$ being the source. FASTEST uses the incomplete LU decomposition with Stone's approximation [123], also known as Strongly Implicit Procedure (SIP), for computing each solution of any linear equation system (3.61).

### 3.4.1 Pressure-Velocity Coupling

By taking a look at the incompressible NSE, it is noticeable that the velocity components appear in both equations while the pressure only takes place in the momentum equation. If the flow would be compressible, the continuity equation could be used for the density transport and the pressure could be obtained through the equation of state $p = p(\rho, T)$. However, incompressibility is assumed here, which defines the density as constant and removes the connection between the velocities and the pressure. A missing equation for pressure is overcome by a numerical coupling between velocity and pressure. The pressure-correction algorithm Semi-Implicit Method for Pressure Linked Equations (SIMPLE) [93] is employed to find a solution which fulfills simultaneously both the momentum and the continuity equation.
The iterative procedure starts with an initial guess (e.g. from the previous iteration or time step) for the pressure and velocity field. The discrete momentum equation is solved with the initial values, which results in preliminary velocities. At this point, the preliminary velocities satisfy the momentum equation but not the continuity equation and hence, they need to be corrected. A pressure-correction equation is derived from the continuity and the momentum equation, in which the unknown neighbor velocity corrections are neglected (exclusive in the SIMPLE algorithm). After solving the pressure correction equation, the pressure correction itself is used to update the initial flow field. With the updated flow field the scalar equations are solved, e.g. temperature or the volume fraction transport. The procedure is repeated until convergence is reached and the computation continues with the next time step.
Within the pressure-correction algorithm some assumptions are made, e.g. the guessed initial flow field or the neglecting of the neighboring velocity corrections. In order to keep

the algorithm stable, an under-relaxation of the velocity and the pressure is employed. The new velocity is a combination of the previous velocity and the current velocity from the iteration procedure, in which the relaxation factor $\lambda_{u_i}$ determines the proportion. The under-relaxation of the velocity components is incorporated in the pressure-correction algorithm so that Equation 3.61 for a single CV reads

$$\frac{a_p^{u_i}}{\lambda_{u_i}} u_{i,P}^{n+1} + \sum_c a_c^{u_i} u_{i,c}^{n+1} = b_P^{u_i} + (1 - \lambda_{u_i}) u_{i,P}^n, \tag{3.62}$$

while the pressure is simply the summation of the initial guess $p^*$ and the under-relaxed correction term

$$p^{n+1} = p^* + \lambda_p p', \tag{3.63}$$

with relaxation parameter $0 \leq \lambda_p \leq 1$ [110]. For the SIMPLE algorithm, an optimal relation $\lambda_p = 1 - \lambda_{u_i}$ should be kept [31].
The solver FASTEST used for this thesis is based on a collocated grid arrangement, which means that the flow variables are computed for cell centers. In combination with the incompressible NSE this arrangement can lead to a decoupling between the pressure and the velocity fields [110]. Rather than discretizing the pressure gradient in the momentum equation with linear interpolation methods, it is selective interpolated as proposed by Rhie and Chow [107]. Thus, the velocity corrections in the SIMPLE algorithm depend on the pressure corrections of the directly neighboring CVs and not on those of the neighbors at twice the distance.
Additional corrections to the algorithm are necessary if rapidly changing volume forces are present as in multifluid systems. The Rhie-Chow interpolation is modified with Gu's body force correction [78], which corrects the interpolation of the face velocities by the face approximations of the body forces.

### 3.4.2 Boundary Conditions

Initial and boundary conditions are necessary to generate a unique solution of each system of equations. In CVs located on boundaries of the domain, there are different types of conditions to which the CV can be assigned to. The most basic boundary conditions are

- Dirichlet with $\phi = \phi_b$
- Neumann with $\dfrac{\partial \phi}{\partial x_i} n_i = b_{i,b}$

where $\phi_b$ and $b_{i,b}$ are prescribed values or functions. Other boundary conditions can be derived from the two basic ones, e.g. the symmetry boundary condition is of type Neumann

and is set to zero ($\partial\phi/\partial x_i n_i = 0$). Application examples are no-slip walls, for which the Dirichlet boundary condition for the velocity is applied with value zero. However, each boundary condition alters the coefficient and the source for a CV in the linear system of equations in (3.61).

## 3.5 Acoustic Discretization

To overcome the multi scale problem computing fluid flow at low Mach numbers and acoustics, a sub-cycling technique employed for the Linearized Euler Equations (LEE), presented in Section 2.3, solves a number of acoustic time steps $N_{ac}$ with the acoustic time step size $\Delta t_{ac}$. Sub-cycling or also known as frozen fluid approach [57] in this context implies that for a single CFD time step multiple acoustic time steps are evaluated in order to meet the overall computational time. In order to set the acoustic time quantities the initial CFL number is calculated with the speed of sound $c$ acting as velocity:

$$CFL_{ac,init} = \frac{\Delta t \cdot c}{\Delta h}. \tag{3.64}$$

Afterwards, a predefined acoustic CFL number $CFL_{ac,\mathrm{ref}}$ is used to set the acoustic time step size as well as the number of the acoustic time steps as

$$N_{ac} = \frac{CFL_{ac,\mathrm{init}}}{CFL_{ac,\mathrm{ref}}}, \tag{3.65}$$

$$\Delta t_{ac} = \frac{\Delta t}{N_{ac}}. \tag{3.66}$$

The reference CFL number is set to $CFL_{ac,\mathrm{ref}} \leq 0.5$. The numerical methods for solving acoustics are implemented in FASTEST and therefore, the FVM with a dimensional splitting-approach is employed for the LEE. For multiphase computations, the speed of sound depends on the medium present in the CV of interest. Hence, the acoustic time step size is determined by the maximum speed of sound in the whole computational domain. In the following more details on the acoustic discretization are given and further insights can be found in [67, 132, 57, 54].

### 3.5.1 Finite Volume Method for Linearized Euler Equations

For the discretization of the LEE given in Equations (2.32)-(2.34), the FVM is employed. Since the LEE form a hyperbolic system of partial differential equations, the procedure of the numerical solution differs from the NSE. For explanation purpose, the system is rewritten into

$$\frac{\partial U}{\partial t} + A_x \frac{\partial U}{\partial x} + A_y \frac{\partial U}{\partial y} + A_z \frac{\partial U}{\partial z} = S, \tag{3.67}$$

in which $U(x,t) = U$ is defined as the variable vector for the acoustic quantities ($'$)

$$U = [\rho', u', v', w', p']^T. \tag{3.68}$$

For each Cartesian coordinate $x$, $y$ and $z$ a coefficient matrix $A$ is introduced, e.g.

$$A_x = \begin{bmatrix} u & \rho & 0 & 0 & 0 \\ 0 & u & 0 & 0 & 1/\rho \\ 0 & 0 & u & 0 & 0 \\ 0 & 0 & 0 & u & 0 \\ 0 & c^2\rho & 0 & 0 & u \end{bmatrix}. \tag{3.69}$$

Finally the source term $S$ only contains the temporal pressure derivative as

$$S = \left[0, 0, 0, 0, \frac{\partial p}{\partial t}\right]^T. \tag{3.70}$$

The flux formulation separated by coordinates (3.67) is reasoned by the employed dimensional splitting scheme, which is a fractional step method for spatial directions. The idea is to split multi-dimensional problems into a sequence of one-dimensional subproblems solved by one-dimensional methods in a relative inexpensive way [67]. The solution of the initial value problem after the first sweep (e.g. in the x-direction) is the Initial Condition (IC) of the following sweep and so on. The solution of the last sweep is the final solution for the new acoustic time step at $(n+1)$. Formally written in the flux form of Equation (3.67) and the notation of Toro [132] yields

$$\left.\begin{array}{ll} \text{PDEs:} & \dfrac{\partial U}{\partial t_{ac}} + A_x \dfrac{\partial U}{\partial x} = 0, \\ \text{IC:} & U^n, \end{array}\right\} \xRightarrow{\Delta t_{ac}} U^{n+1/3} \tag{3.71}$$

$$\left.\begin{array}{ll} \text{PDEs:} & \dfrac{\partial U}{\partial t_{ac}} + A_y \dfrac{\partial U}{\partial y} = 0, \\ \text{IC:} & U^{n+1/3}, \end{array}\right\} \xRightarrow{\Delta t_{ac}} U^{n+2/3} \tag{3.72}$$

$$\left.\begin{array}{ll} \text{PDEs:} & \dfrac{\partial U}{\partial t_{ac}} + A_z \dfrac{\partial U}{\partial z} = 0, \\ \text{IC:} & U^{n+2/3}. \end{array}\right\} \xRightarrow{\Delta t_{ac}} U^{n+1} \tag{3.73}$$

The three systems of Partial Differntial Equations (PDEs) are completed by adding the source term to the IC of the first sweep, which can be expressed as an system of Ordinary Differential Equations (ODEs)

$$\left.\begin{array}{ll} \text{ODEs:} & \dfrac{\partial U}{\partial t_{ac}} = S \\ \text{IC:} & U^{n*} \end{array}\right\} \xRightarrow{\Delta t_{ac}} U^n \tag{3.74}$$

The state before source term addition is marked as $n*$. The sweep direction and the variable vector are transformed at the cell face in the direction of the outgoing normal vector (similar to Section 3.2.3). In case the numerical grid is aligned with the Cartesian coordinate system, the sweep direction from east to west corresponds to the pure x-direction. However, the set of hyperbolic equations (3.71)-(3.73) are discretized with the FVM, for which cell centers contain the solution of the unknown variables and the fluxes at the cell faces have to be computed. Therefore, the numerical solution of each augmented one-dimensional hyperbolic system is based on solving the Riemann problem at the corresponding cell face.

In order to understand the Riemann problem and its solution, a brief explanation is given in advance. Consider a system of hyperbolic equations such as a single sweep in the dimensional splitting approach or such as

$$\begin{aligned} &\text{PDEs:} \quad \frac{\partial U}{\partial t} + A\,\frac{\partial U}{\partial x} = 0, \\ &\text{IC:} \quad U(x,t_0) = \begin{cases} U_L & \text{if } x < 0, \\ U_R & \text{if } x > 0. \end{cases} \end{aligned} \tag{3.75}$$

where $U_L$ (left) and $U_R$ (right) are two constant values between a single jump discontinuity at $x = 0$. The PDEs in combination with the special IC are defined as the Riemann problem [67]. Due to the assumption of hyperbolicity the coefficient matrix $A$ has $m$ real eigenvalues $\lambda_i$ and $m$ linearly independent eigenvectors $k^i$ with $i = 1, \ldots, m$, so that

$$Ak^i = \lambda_i k^i \qquad \text{and} \tag{3.76}$$

$$A^{\pm} = K\Lambda^{\pm}K^{-1}, \tag{3.77}$$

with matrix $K = [k^1, \ldots, k^m]$ containing the eigenvectors and $K^{-1}$ being the inverse of $K$. The eigenvalues can be split into positive and negative values as

$$\begin{aligned} \lambda_i^+ &= \max(\lambda_i, 0), \\ \lambda_i^+ &= \min(\lambda_i, 0), \end{aligned} \tag{3.78}$$

and the components of the diagonal matrix $\Lambda$ are given by

$$\Lambda^{\pm} = \begin{pmatrix} \lambda_1^{\pm} & & 0 \\ & \ddots & \\ 0 & & \lambda_m^{\pm} \end{pmatrix}. \tag{3.79}$$

The splitting matrices satisfy the following conditions:

$$\lambda_i = \lambda_i^+ + \lambda_i^- \;, |\lambda_i| = \lambda_i^+ - \lambda_i^-, \tag{3.80}$$

$$\Lambda \;= \Lambda^+ + \Lambda^- \;, |\Lambda| \;= \Lambda^+ - \Lambda^-, \tag{3.81}$$

$$A \;= A^+ + A^- \;, |A| \;= A_i^+ - A_i^-. \tag{3.82}$$

However, the one-dimensional system (3.75) is rewritten into its characteristic form as

$$\frac{\partial w}{\partial t} + \Lambda \frac{\partial w}{\partial x} = 0 \quad \text{or} \tag{3.83}$$

$$\frac{\partial w^i}{\partial t} + \lambda_i \frac{\partial w^i}{\partial x} = 0 \quad , \tag{3.84}$$

with the definition $w = K^{-1}U$ or its components $w^i$ since $\Lambda$ is diagonal. The characteristic form expresses that a set of linear combination of $m$ waves travel at characteristic speeds $\lambda_1, \ldots, \lambda_m$ on characteristic curves $x(t) = x(t_0) + \lambda_i t$. The eigenvalues are called characteristic values and the function $w^i(x,t)$ characteristic variables [67]. The decoupled system in (3.84) of $m$ independent advection equations is used to find the solution of the Riemann problem with the piecewise constant initial values. The discontinuity at $x = 0$ of the Riemann problem is expected to propagate along the characteristic curve, so that the solution of each advection equation is found at

$$w^i(x,t) = \begin{cases} w_l^i \text{ if } x - \lambda_i t < 0, \\ w_r^i \text{ if } x - \lambda_i t > 0. \end{cases} \tag{3.85}$$

The overall solution of the Riemann problem is achieved by composing left and right solutions of each advection equation

$$U(x,t) = \sum_{i:\lambda_i < x/t} w_r^i k^i + \sum_{i:\lambda_i > x/t} w_l^i k^i \tag{3.86}$$

at a given point $(x,t)$. The jump of the variables in $U$ over the $i$-th wave $W^i$ in the case of a linear system can be decomposed such as

$$(w_r^i - w_l^i)k^i = \delta^i k^i = W^i, \tag{3.87}$$

with the components $\delta^i$ of vector $K^{-1}(U_R - U_L)$ , also labeled as wave strength [132]. Equation (3.86) rewritten with the wave definition of (3.87) gives

$$U(x,t) = U_L + \sum_{i:\lambda_i < x/t} W^i \tag{3.88}$$

$$= U_R - \sum_{i:\lambda_i > x/t} W^i. \tag{3.89}$$

For the LEE, $m = 5$ and the eigenvalues of coefficient matrix $A$ normal to the cell face are given by

$$\lambda_1 = u - c\,,\ \lambda_2 = \lambda_3 = \lambda_4 = u\,,\ \lambda_5 = u + c, \tag{3.90}$$

with corresponding right eigenvectors [132]

$$K = [k^1, k^2, k^3, k^4, k^5] = \begin{bmatrix} \rho & 1 & \rho & \rho & \rho \\ -c & 0 & 0 & 0 & c \\ 0 & v & 1 & v & 0 \\ 0 & w & w & 1 & 0 \\ \rho c^2 & 0 & 0 & 0 & \rho c^2 \end{bmatrix}. \tag{3.91}$$

Recalling on finding a solution to the equations of the LEE in the dimensional splitting approach. By employing FVM in space and the explicit Euler method in time, Equation (3.75) becomes

$$U_P^{n+1} = U_P^n - \frac{\Delta t}{\Delta x}(F_e^n - F_w^n). \tag{3.92}$$

As for the multiphase computation a simple first- or second-order advection scheme is not sufficient for the acoustics computation, for which reason high resolution methods from Section 3.2.3 are employed here as well. The numerical flux $F$ is constructed as a combination of the low-order Godunov and high-order Lax-Wendroff method.
The Godunov first-order upwind method[2] utilizes Equation (3.92) and requires the solution of the local Riemann problem to compute the interface flux [132]. The numerical flux $F$ approximated with the positive/negative splitting (3.82) at the east face yields

$$F_e^n = A^+ U_P + A^- U_E, \tag{3.93}$$

With those identical flux definitions, the first-order Godunov method yields

$$U_P^{n+1} = U_P^n - \frac{\Delta t}{\Delta x}(A^+ \Delta U_w + A^- \Delta U_e), \tag{3.94}$$

for which $A^{\pm}\Delta U_{w/e}$ are computed in the wave form. The second-order Lax-Wendroff is derived by a Taylor expansion series and central difference approximations, which leads to the numerical flux

$$F_e^n = (A^+ U_P^n + A^- U_E^n) + \frac{1}{2}|A| \left( I - \frac{\Delta t}{\Delta x}|A| \right) (U_E^n - U_P^n), \tag{3.95}$$

being the first-order upwind (3.93) with an correction term. The update to the variable vector for the Lax-Wendroff method results in

$$U_P^{n+1} = U_P^n - \frac{\Delta t}{\Delta x}(A^+ \Delta U_w^n + A^- \Delta U_e^n) - \frac{\Delta t}{\Delta x}(F_e^n - F_w^n). \tag{3.96}$$

[2]Godunov proposed the following scheme to update a cell value $U_P^n$ to a new value $U_P^{n+1}$: solve the two Riemann problems at $(U_W^n, U_P^n)$ and at $(U_P^n, U_E^n)$ for the conservation law (3.75), take an integral average in cell $P$ of the combined solutions of these two local problems and assign the value to $u^{n+1}$ [132].

In this context a HR method sets the proportions of the upwind method and the Lax-Wendroff to the flux dependent on how the acoustic field is varying, expressed by a flux limiter $\psi(r_f)$ with gradient coefficient or smoothness indicator $r_f$ (see Section 3.2.3). The smoothness indicator is evaluated for the east face $f = e$ depending on

$$r_e^i = \frac{\delta_f^i}{\delta_e^i} \quad \text{with } f = \begin{cases} w & \text{if } \lambda_i > 0, \\ ee & \text{if } \lambda_i < 0. \end{cases} \tag{3.97}$$

The situation of each flux and its eigencomponents might be different, therefore, the correction term in (3.95) is decomposed into the eigencomponents with the corresponding flux-limiter given as

$$\tilde{F}_e = \frac{1}{2} \sum_{i=1}^{m} |\lambda_i| \left(1 - \frac{\Delta t}{\Delta x} |\lambda_i|\right) \psi(r_e^i) \delta_e^i k^i. \tag{3.98}$$

Finally Equation (3.75) discretized with flux-limiter yields

$$U_P^{n+1} = U_P^n - \frac{\Delta t}{\Delta x}(A^+ \Delta U_w^n + A^- \Delta U_e^n) - \frac{\Delta t}{\Delta x}(\tilde{F}_e^n - \tilde{F}_w^n). \tag{3.99}$$

The Osher limiter in (3.42) among other limiters have been tested for the underlying framework in [57]. Note that if the limiter tends to zero, the pure upwind method determines the flux and on the other hand if it becomes one only the Lax-Wendroff is in charge. However, the smoothness indicator in Equation (3.97) might need a neighbor next to the direct neighboring CV. Therefore, at block boundaries only the upwind method will be employed due to the lack of more than one ghost CV.
To conclude the FVM for the LEE, acoustic boundary conditions have to be set. Except walls, all flow boundaries are non-reflective, zero-order extrapolation acoustic outlets. At walls the acoustics are completely reflected. Multiphase interfaces are special case and explained in the following section.

### 3.5.2 Multiphase Acoustics

In the underlying framework different phases are taken into account by varying material properties, so that $\rho = \rho(x_i)$ and $c = c(x_i)$. The impedance $Z(x_i) = \rho(x_i)c(x_i)$ is constant within each CV, however, it may vary through the domain referred as layered medium [67]. The solution algorithm of the acoustics, described in the previous section, splits the multidimensional Riemann problem for a CV into three one-dimensional systems at the corresponding cell faces in their normal direction. Additional to the piecewise constant initial data (3.75), the one-dimensional Riemann problem in normal direction to the cell face $\xi$ reads as

$$Z(\xi) = \begin{cases} Z_l & \text{if } \xi < 0, \\ Z_r & \text{if } \xi > 0, \end{cases} \tag{3.100}$$

with the discontinuity at $\xi = 0$. For example, the waves on each side of the discontinuity for the acoustic pressure are computed with the impedance as

$$W^{p',1} = Z_l \frac{\Delta p' - Z_r \Delta u'_\xi}{Z_l + Z_r}, \tag{3.101}$$

$$W^{p',5} = Z_r \frac{\Delta p' + Z_l \Delta u'_\xi}{Z_l + Z_r}. \tag{3.102}$$

The acoustic velocity in normal direction $u_\xi$ is obtained by projection on the vector normal to the face. By employing the high resolution method, which solves the Riemann problem for every CV, the varying material properties are taken into account naturally. Reflection and transmission at an fixed interface between fluids of different impedance has been verified in [122] with second order accuracy.

### 3.5.3 Acoustic Source

The one-way coupling of the flow field to the acoustics is achieved by the source term on the right hand side of the acoustic pressure transport equation (2.34). The temporal pressure derivative is approximated with the explicit Euler method (see Subsection 3.2.2), so that

$$S = -\frac{\partial p}{\partial t} \approx -\frac{p^{n+1} - p^n}{\Delta t}. \tag{3.103}$$

Within the sub-cycles, the source term is split into $N_{ac}$ parts for the computation of the sound propagation, so that

$$U^{n_{ac}+1} = U^{n_{ac}} + \frac{\Delta t}{N_{ac}} S. \tag{3.104}$$

# 4. Interface Treatment

Before running a simulation, each phase has to be initialized either as an inflow boundary condition or as a geometry present in domain. In this work, only problems with an initial volume fraction distribution are discussed. The methods implemented for interface initialization are presented at first. Within a VOF framework an accurate estimation of the interface surface tension force between two fluids depends on the discretization of the surface tension and on the curvature model. The discretization strategy is explained in previous Section 3.2.4 and the curvature models employed are presented secondly.

## 4.1 Interface Initialization

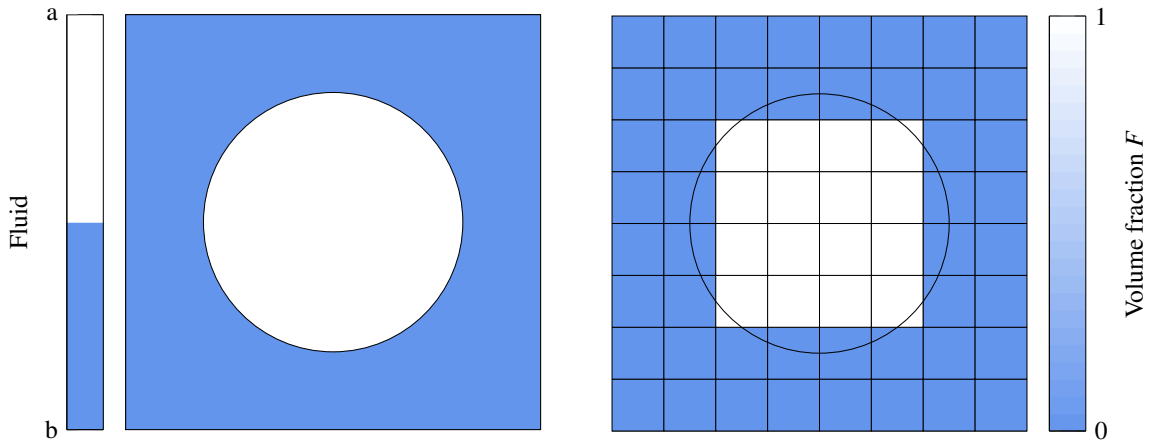
For some problem configurations certain geometrical shapes with curves are desired, e.g. ellipses or spheres. For the initial distribution of the volume fraction field different approaches are existent. Here, two approaches are presented which set the volume fraction for each CV center. Afterwards the values are smoothed for further computation. Depending on the distribution method used, the resulting volume fraction field shows different width around the exact interface location.

### 4.1.1 Binary Initialization

First a simple weighted binary initialization is presented, which follows a discontinuous indicator function as

$$\alpha(x_i) = \begin{cases} 1 & \text{if } x_i \in \text{fluid a} \\ 0 & \text{if } x_i \in \text{fluid b.} \end{cases} \tag{4.1}$$

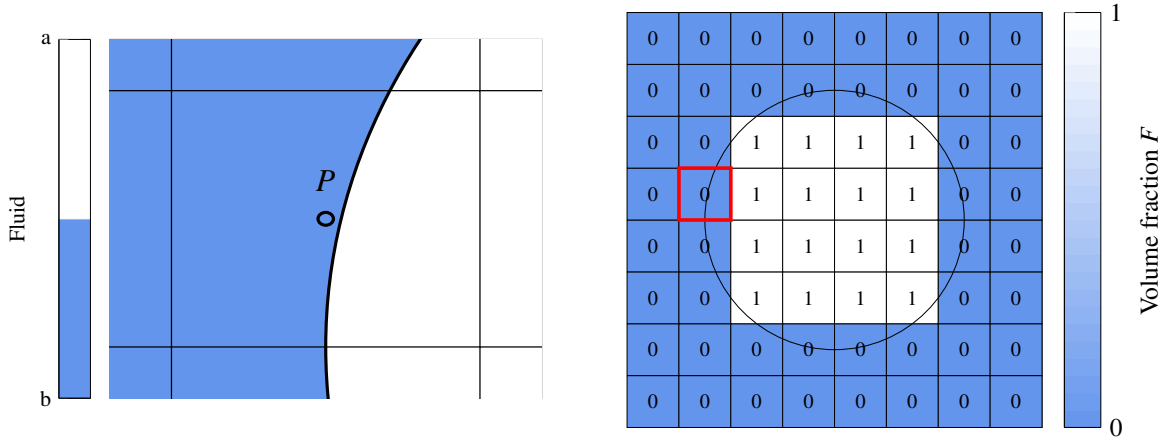
On the basis of the center node $P$ coordinates of each CV the binary decision (4.1) is made, so that the volume fraction equals the indicator function $F_{\text{init}}^{P} = \alpha(x_i^P)$.




(a) Circle of fluid a surrounded by fluid b

(b) Volume fraction field

Figure 4.1: Discontinuous and discrete distribution of fluid a and fluid b




(a) Center node $P$ of the red CV in Figure 4.2b is locate outside the circle

(b) Resulting volume fraction field with binary initialization

Figure 4.2: Binary intialization

### 4.1.2 Numerical Initialization

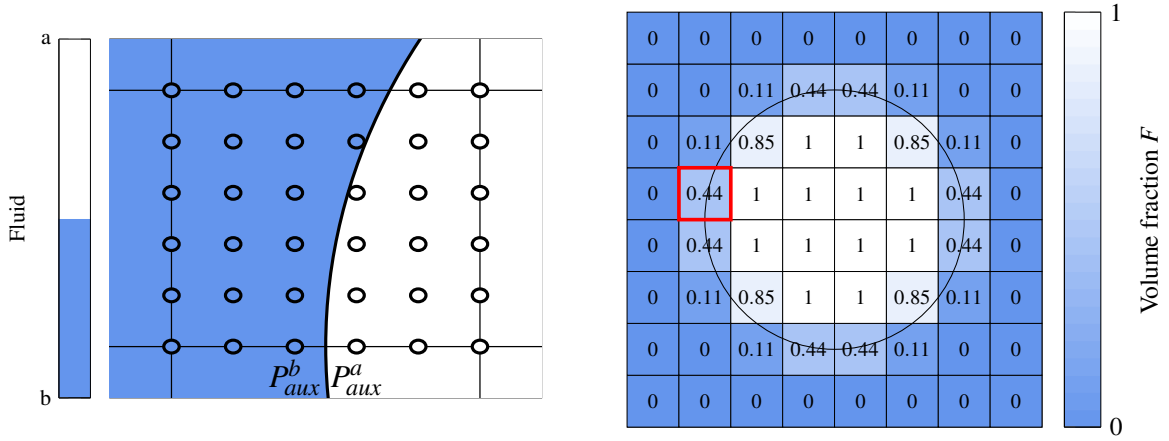
For the second variant for each CV a set of $N_{\text{aux}}$ auxiliary vertices $P_{\text{aux}}$ are evenly placed in each CV, see Figure 4.3a. Instead of the center node each auxiliary vertex is assigned to a fluid as

$$P_{\text{aux}}(x_i) = \begin{cases} P^a_{\text{aux}} & \text{if } x_i \in \text{fluid a} \\ P^b_{\text{aux}} & \text{if } x_i \in \text{fluid b.} \end{cases} \tag{4.2}$$

By summing the vertices which are located within the geometry of fluid a divided by the overall auxiliary vertices of the CV, the volume fraction is obtained by

$$F_{init}^{P} = \frac{\sum P_{aux}^{a}}{N_{aux}}. \tag{4.3}$$

Considering the example shown in Figure 4.3, $\sum P_{\text{aux}}^{a}$ equals 16 and the number of auxiliary vertices $N_{\text{aux}}$ is 36 which results in a initial volume fraction of $F_{\text{init}} = 0.44$.




(a) Auxiliary vertices within the red CV in Figure 4.3b

(b) Resulting volume fraction field with numerical initialization

Figure 4.3: Numerical initialization

### 4.1.3 Smoothing of the volume fraction field

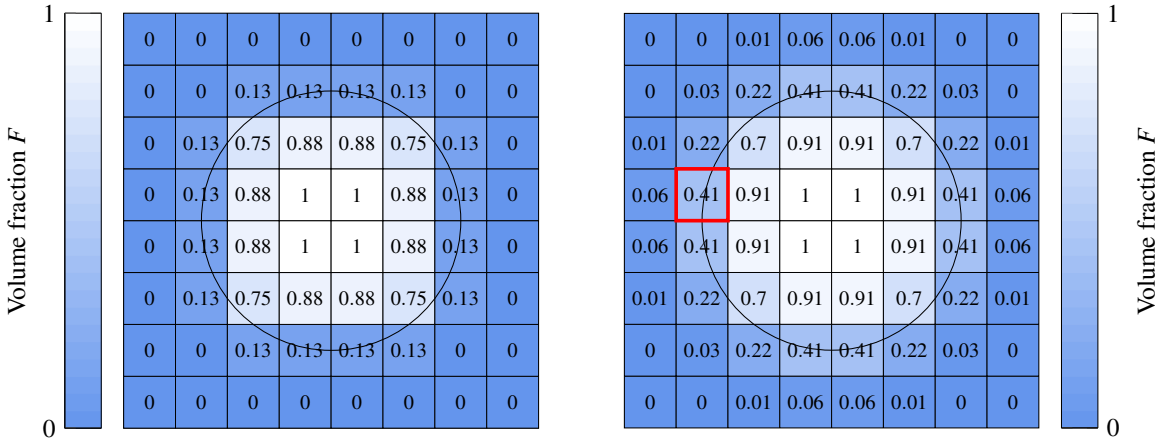
After the smoothing, the initial values of the volume fraction are further distributed around the exact interface locations. With the help of the cell geometry the weighted averages of the volume fraction are computed. First the cell face values $F_f$ are determined by numerical interpolation for which different methods can be employed. For example, employing the CDS for the east face between the central node $P$ and its neighbor $E$ yields

$$F_{e}^{P} \approx F^{P}(1 - \lambda_e) + F^{E}(\lambda_e) \tag{4.4}$$

with the interpolation factor $\lambda_e$ taken from Section 3.2.3. Afterwards the cell face values are summed for each CV and multiplied by their geometric weights. Justified by the interface capturing methods and the regular grids used in this work, the weights $w_c$ are set to unity. Every connecting CV face $F_c$ is used for the final smoothed volume fraction value in the node as

$$\tilde{F} = \frac{\sum_c F_c w_c}{\sum_c w_c}. \tag{4.5}$$

For the exemplary red CV in Figure 4.4b, which holds the initial value $F_{\text{init}} = 0.44$, the smoothed value is determined by the four cell face values $c = e, w, n, s$ and $\sum_c w_c = 4$ resulting in $\tilde{F} = (0.22 + 0.72 + 0.275 + 0.44)/4 = 0.41$. As both Figures indicate, the




(a) Smoothing after binary initialization (b) Smoothing after numerical initialization

Figure 4.4: Smoothing of volume fraction field

interface is smeared around the exact interface location. For the binary initialization the interface stays within two cells along the coordinate axis whereas with the numerical initialization the interface width spans over three CVs. However, throughout this work a sharp interface will never occur.

## 4.2 Interface Curvature Estimation

For collocated variable arrangement in conjunction with interface capturing methods, the determination of an accurate interface curvature has been challenging ever since [143, 26]. The present Eulerian interface description leads to a volumetric formulation of the surface tension $f_{i,S}$ as in Equation (2.25)

$$f_{i,S} = \sigma\kappa \frac{\partial F}{\partial x_i} . \tag{4.6}$$

While the interface curvature $\kappa$ and the volume fraction gradient have to be determined, the material dependent surface tension coefficient $\sigma$ is set constant. The volume fraction gradient becomes zero outside the interface, since surface tension only acts on the interface. Note that the volume fraction around the interface will always be diffuse to overcome the difficulty of differentiating a discontinuous function. However, additional smoothing of the

volume fraction field only for curvature determination could be employed. For the non-constant part of the surface tension four methods and their improvements are presented hereafter and evaluated in Section 5.1.

### 4.2.1 Curvature by Finite Difference Method

The curvature $\kappa$ in Equation (4.6) can be written for three dimensions in Cartesian coordinates with spatial derivatives denoted by indices $x$, $y$ and $z$ as

$$\kappa = -\nabla \cdot n_i = -\nabla \cdot \frac{\nabla F}{|\nabla F|} \tag{4.7}$$

$$= -\frac{\begin{array}{c} F_{xx}F_y^2 + F_{xx}F_z^2 + F_{yy}F_x^2 + F_{yy}F_z^2 + F_{zz}F_x^2 + F_{zz}F_y^2 \\ -2F_xF_yF_{xy} - 2F_xF_zF_{xz} - 2F_yF_zF_{yz} \end{array}}{(F_x^2 + F_y^2 + F_z^2)^{\frac{3}{2}}} \tag{4.8}$$

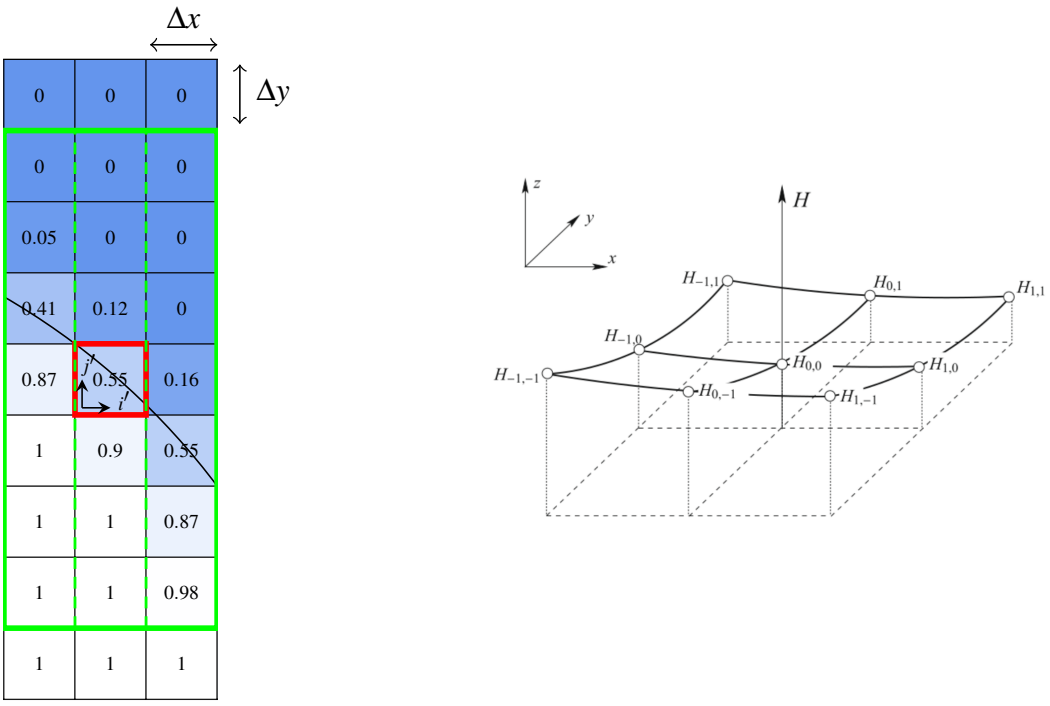
In a two-dimensional case the derivatives including the $z$-coordinate vanish. To obtain the curvature in each CV the first and second spatial derivative of the volume fraction are needed. The simplest method to discretize expression (4.8) is to apply the CDS from Section 3.2.3, which gives the first derivative in $x$-direction on uniform grid spacing ($\lambda_x = \frac{1}{2}$) as

$$F_x \approx \frac{F_e - F_w}{x_e - x_w} = \frac{\lambda_x F_E + (1-\lambda_x)F_P - (\lambda_x F_P + (1-\lambda_x)F_W)}{x_e - x_w} = \frac{\frac{1}{2}F_E - \frac{1}{2}F_W}{\Delta x}. \tag{4.9}$$

In the context of curvature computation, the finite difference method will be referred as Curvature by Finite-Difference-Method (CFDM). Since only direct neighbors are included for the curvature calculation, the advantage of this method is the low computation time. At the same time, it is also the disadvantage, since large radii cannot be represented well. Although the method has second order accuracy, it tends to oscillate and to overshoot the solution especially if the fraction field gradient is steep. To improve the accuracy of the curvature values in each cell and in the mean, some additional features are applied which are shown in Section 4.2.3.

### 4.2.2 Height Function Method

A different approach of the curvature estimation is the Height-Function (HF) method. Although the principle was initially mentioned in the 1980's ([133],[100]), Popinet [102] states that its popularity began to grow with the work by Sussmann [125] and Cummins et al. [16]. The HF method sums the volume fractions in a stencil along the largest component of the unit vector $n_i = (n_x, n_y, n_z)$ and differentiates a set of according heights.




(a) $3 \times 7$ HF stencil in 2D (b) $3 \times 3$ HF stencil in 3D, taken from [72]

Figure 4.5: HF stencils in 2D and 3D

Consider a two dimensional volume fraction distribution as in Figure 4.5a with $|n_y| > |n_x|$ at the center of a $3 \times 7$ stencil. The CV indexing with $i$ and $j$ correspond to the $x$- and $y$-coordinate directions, respectively. In the center of a stencil, where the curvature will be approximated, a local indexing is introduced with $i'$ and $j'$. Following, the heights along the $x$-coordinate, or the $i$-indexing, can be derived by

$$H_{i'} = \sum_{j'=j-N_H}^{j'=j+N_H} F_{i',j'} \Delta y \,, \quad \text{with } i' = \frac{N_N - 1}{2}, i, \frac{N_N + 1}{2} \quad . \tag{4.10}$$

The number of heights to be included is set by $N_N$ and the number of CV to be summed in positive and negative $j$-direction per height is set with $N_H$. Francois et al. [33] found that the stencil with three heights $N_N = 3$ and three CV above and below the reference $N_H = 3$ produces the most accurate results. With three heights computed with Equation (4.10), first and second derivative are estimated with second order finite differences as

$$H_x = \frac{H_{i+1} - H_{i-1}}{2\Delta x}, \tag{4.11}$$

$$H_{xx} = \frac{H_{i+1} - 2H_i + H_{i-1}}{(\Delta x)^2}. \tag{4.12}$$

For three dimensional computations the derivatives in (4.11) and (4.12) are extended about a cross derivative and the index $k$ in the $z$-coordinate direction. In case of a $3 \times 3 \times 7$ stencil with the largest normal in the $z$-direction ($|n_z| > |n_y|$ and $|n_z| > |n_x|$), the central differences formulas constructed for an interface cell are

$$H_x = \frac{H_{i+1,j} - H_{i-1,j}}{2\Delta x}, \tag{4.13}$$

$$H_y = \frac{H_{i,j+1} - H_{i,j-1}}{2\Delta y}, \tag{4.14}$$

$$H_{xx} = \frac{H_{i+1,j} - 2H_{i,j} + H_{i-1,j}}{(\Delta x)^2}, \tag{4.15}$$

$$H_{yy} = \frac{H_{i,j+1} - 2H_{i,j} + H_{i,j-1}}{(\Delta y)^2}, \tag{4.16}$$

$$H_{xy} = \frac{H_{i+1,j+1} + H_{i-1,j-1} - H_{i+1,j-1} - H_{i-1,j+1}}{4(\Delta y \Delta x)}. \tag{4.17}$$

$$\tag{4.18}$$

The curvature in the central CV at index $(i, j)$ and $(i, j, k)$ can be determined by

$$\kappa = \begin{cases} -\dfrac{H_{xx}}{(1 + H_x^2)^{\frac{3}{2}}} & \text{, in 2D} \\ \\ -\dfrac{H_{xx} + H_{yy} + H_{xx}H_y^2 + H_{yy}H_x^2 - 2H_xH_yH_{xy}}{(1 + H_x^2 + H_y^2)^{\frac{3}{2}}} & \text{, in 3D.} \end{cases} \tag{4.19}$$

The HF curvature is second-order accurate [13]. Nevertheless, Lopez et al. [72] found that the accuracy in three dimensions can be improved by smoothing the first and second derivative with a factor $\gamma$ as

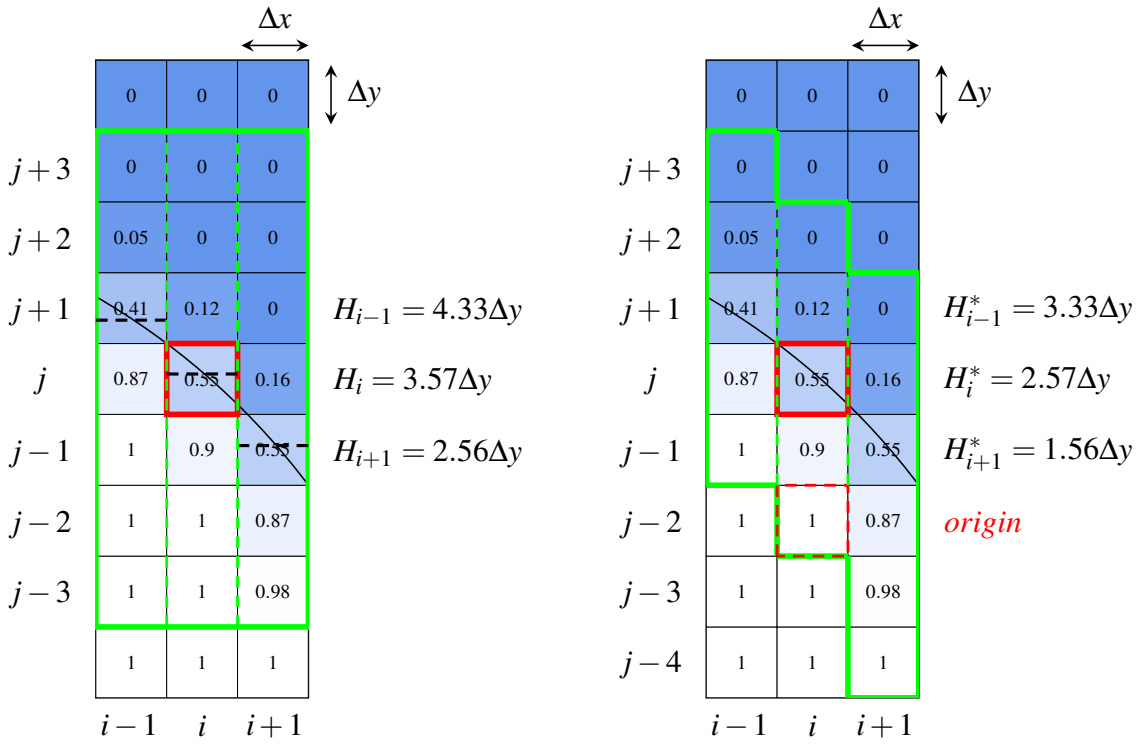
$$\gamma = \begin{cases} 0.0 & \text{, if } \theta < \theta_{crit} \\ 0.2 & \text{, otherwise.} \end{cases} \tag{4.20}$$

The smoothing parameter $\gamma$ depends on the angle $\theta$, which is measured between the height function direction and the interface normal vector according to $\theta = \arccos(\max(|n_x|, |n_y|, |n_z|))$. In case the angle reaches a critical value of $\theta_{crit} = 0.8$ [72], the smoothing takes effect on the derivatives following

$$H_x = \frac{\gamma(H_{i+1,j+1} - H_{i-1,j+1}) + H_{i+1,j} - H_{i-1,j} + \gamma(H_{i+1,j-1} - H_{i-1,j-1})}{2\Delta x(1 + 2\gamma)}, \tag{4.21}$$

$$H_{xx} = \frac{\begin{array}{c}\gamma(H_{i+1,j+1} - 2H_{i,j+1} + H_{i-1,j+1}) \\ +H_{i+1,j} - H_{i,j} + H_{i-1,j} \\ +\gamma(H_{i+1,j-1} - 2H_{i,j-1} + H_{i-1,j-1})\end{array}}{\Delta x^2(1 + 2\gamma)} \tag{4.22}$$

One important drawback of the fixed stencil method is the fact that curvature calculation only converges with mesh refinement, if the calculated heights in the column center is located in the reference cell (red CV in Figure 4.6a). Hence, the implemented algorithm has to disregard those cells, which do not satisfy this so called HF condition [94]. However, Popinet et al. [101] proposes an improvement to the standard HF methods with the use of adaptive stencils. By employing an adaptive stencil, which considers the topology around the reference cell, the Adaptive Height-Functions (AHF) method is supposed to handle difficult volume fractions configurations more robust [102]. In each column the heights are




(a) Fixed $3 \times 7$ HF stencil with heights (b) Adaptive HF stencil with normelized heights

Figure 4.6: HF stencils in 2D inside green bounds, center CV in red bounds and *origin* CV in dashed red bounds

constructed from a *base* ($F = 1$) to a *top* ($F = 0$) with a strictly monotonic variation of the volume fraction, as in Figure 4.6b. If that is the case the column is flagged as consistent and if all columns fulfill this requirement, the heights are summed by

$$H_{i'} = \sum_{j'=base}^{j'=top} F_{i',j'} \Delta y \,, \quad \text{with } i' = \frac{N_N - 1}{2}, i, \frac{N_N + 1}{2} \quad . \tag{4.23}$$

Afterwards the heights are normalized relative to the base of the reference cell, called *origin*, as

$$H^*_{i'} = H_{i'} - \mathrm{sgn}(n_y)(base_{i'} - origin)\Delta y. \tag{4.24}$$

Otherwise, the stencil direction is defined as inconsistent and the method is applied in the other normal direction. In case the other stencil direction turns out to be inconsistent as well, accurate curvature computation within the adaptive stencil is not possible and a fallback strategy has to take over (e.g. previous curvature value or finite difference curvature computation 4.2.1). However, if a direction is found to be consistent the curvature is computed whether the HF condition is fulfilled or not.

### 4.2.3 Improvements to CFDM and HF

The fastest methods for curvature computation are the CFDM approach and the basic HF. Both can lack of accuracy at certain conditions, yet after employing some additional features the results can be improved significantly. A better estimation of the first derivatives is achieved by filtering the volume fraction with a smooth least-squares polynomial (LSP) filter of degree two [91]:

$$\tilde{F}_{\mathrm{P}} = \frac{1}{3}\sum_{x_i=1}^{3}\left(\frac{\sum_{k=-2}^{2} b_k F^{x_i}_{\mathrm{P}-k}}{\sum_{k=-2}^{2} b_k}\right). \tag{4.25}$$

The mean value of all spatial directions $x_i = (x, y, z)$, derived by the sum of the coefficients $b_k$ times the corresponding concentrations divided by the sum of the coefficients, leads to a smoother distribution of the concentration. Using the central differencing scheme on the first derivatives to obtain the second derivatives, once again applying a filter on the first derivatives does not improve the outcome.
Missing values for the second neighbor at block boundaries are linear extrapolated with the help of the spatial and concentration gradients. Considering a CV next to the boundary in positive $x$-direction, so that the coordinate extrapolation yields

$$\overline{x_{EE}} = x_E + \big(2(x_E - x_P) - (x_P - x_W)\big). \tag{4.26}$$

The extrapolated coordinate is used for the concentration extrapolation in

$$\overline{F_{EE}} = F_E + \frac{\overline{x_{EE}} - x_E}{x_P - x_E}(F_P - F_W). \tag{4.27}$$

After checking for boundedness[1], the extrapolated value at the second neighbor is then placed into the filter.
For further improvement of the curvature a 5-point Chebyshev filter [91] in the manner

[1] $\overline{F_{EE}} = \min\Big(\max(\overline{F_{EE}}, 1), 0\Big)$

of Equation (4.25) and a volume fraction dependent weighting [22] is applied on the curvature itself. The coefficients for the different filters are listed in Table 4.1. As stated

Table 4.1: Filter coefficients $b_k$

| | $b_{-2}$ | $b_{-1}$ | $b_0$ | $b_1$ | $b_2$ |
|---|---|---|---|---|---|
| LSP degree 2 | $-3$ | 12 | 17 | 12 | $-3$ |
| 5-point Chebyshev | $-1$ | 4 | 10 | 4 | $-1$ |

by Renardy et al. [106] the curvature is expected to be more accurate when the volume fraction is not close to zero or unity. Therefore, the weighting for the curvature is defined as

$$\kappa_{\mathrm{P}}^{*} = \frac{\kappa_{\mathrm{P}}\omega_{F,\mathrm{P}} + \sum_{Q} \kappa_{Q}\omega_{F,Q}}{\omega_{F,\mathrm{P}} + \sum_{Q} \omega_{F,Q}}. \tag{4.28}$$

Index $Q$ denotes all the direct neighboring CVs and the weighting factor with the exponent $\lambda$ is given by

$$\omega_{F,i} = \left[1 - 2(|0.5 - F_i|)\right]^{\lambda}. \tag{4.29}$$

In Equation (4.29) it can be seen that the factor reaches zero if the volume fraction tends zero or unity and becomes maximum when the volume fraction equals 0.5.

### 4.2.4 Coupled VOF Level-Set

In order to track the position of the interface, there are VOF-based methods as well as the Level-Set (LS) method. With a combination of both methods in the context of curvature computation, called Coupled-VOF-Level-Set (CVOFLS) method, the advantages of both are combined, which are conserving the mass for the VOF method and an accurate representation of interface geometry for the LS method [92]. For each curvature computation, the LS function $\phi_F$ is initially constructed upon the VOF field with $\Delta h = \max(\Delta x_i)$ as

$$\phi_F^0 = 2\Delta h(F - 0.5), \tag{4.30}$$

which is larger than zero on one side, greater than zero on the opposite site and exact zero on the interface ($F = 0.5$ in the VOF field). The so-called distance function indicates the shortest distance to the interface in each cell center and its equation reads as

$$\frac{\partial \phi_F}{\partial \tau} = \mathrm{sgn}(\phi_F^0)\left(1 - \left|\frac{\partial \phi_F}{\partial x_i}\right|\right), \tag{4.31}$$

with $\tau$ being an artificial time. A smoothed Heaviside function is employed for the sign of the LS function $\text{sgn}(\phi_F^0)$ for better and more stable approximation [92], which are given by

$$\text{sgn}(\phi_F) = \begin{cases} -1 & \text{when } \phi_F < \epsilon, \\ 2\left(H(\phi_F) - \frac{1}{2}\right) & \text{when } |\phi_F| \leq \epsilon, \\ 1 & \text{when } \phi_F > \epsilon \end{cases} \tag{4.32}$$

and

$$H(\phi_F) = \begin{cases} 0 & \text{when } \phi_F < \epsilon, \\ \frac{1}{2}(1 + \frac{\phi_F}{\epsilon} + \frac{1}{\pi}\sin\left(\frac{\pi\phi_F}{\epsilon}\right) & \text{when } |\phi_F| \leq \epsilon, \\ 1 & \text{when } \phi_F > \epsilon. \end{cases} \tag{4.33}$$

In both functions the grid depended interface thickness on one side serves as criterion and is set to $\epsilon = 2\Delta h$. The LS distance function (4.31) is discretized in (artificial) time with a fourth order Runge-Kutta method (Section 3.2.2) and in space with the Weighted Essentially Non-Oscillatory (WENO) scheme [71]. The solution of the signed distance function (4.31) should keep the interface in the initial position, which requires a reinitialization procedure after each artificial time step [127]. For reducing the computational effort, the signed distance function (4.31) is restricted to an area around the interface so that it is solved only up to $\tau = \alpha\Delta h$, with $\alpha$ being a factor.
At the end of the solving procedure, the derivatives of $\phi_F$ are determined by also using the WENO scheme and placed into the curvature computation of Equation (4.8). Only for the CVOFLS method, the surface tension computation changes to

$$f_{i,S} = \sigma\kappa \frac{\partial H(\phi_F)}{\partial \phi_F} \frac{\partial \phi_F}{\partial x_i}. \tag{4.34}$$

For more details of the coupled method or the implementation the reader is referred to [127, 126, 41, 92, 121].

### 4.2.5 Machine Learning

In the previous methods for computing curvature some sort of stencil around the interface cell was taken into account. Then, an algorithm is applied to the stencil to calculate the curvature. For the method described in this subsection, the algorithm is replaced by a machine learning approach. The resulting curvature might be corrected with the methods shown in Section 4.2.3. The overall procedure and findings are the results of a joint work presented in [58]. Figure 4.7 outlines this procedure performed for each CV of the domain along the interface. To feed the artificial neural network (ANN), a stencil of volume fraction

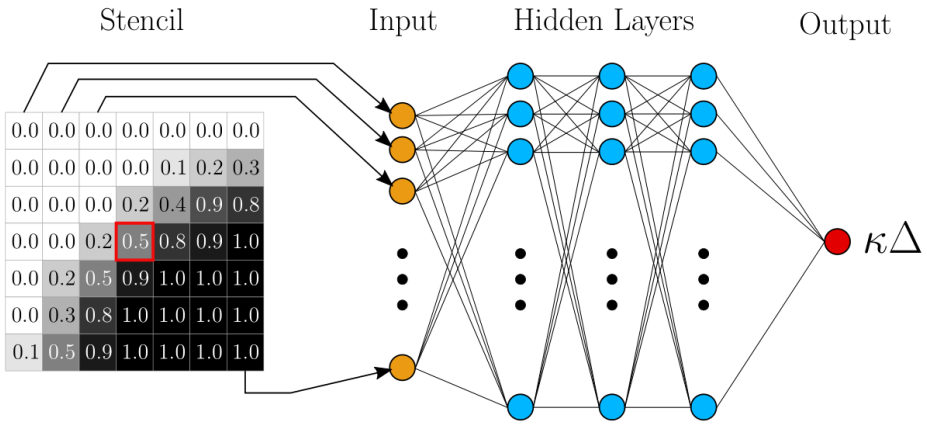



Figure 4.7: Neural network-based curvature prediction from volume fraction stencil [58]

values is used around each computational point as input features. Before being passed to the neural network, the individual volume fractions are always arranged in a vector of the same order. The corresponding curvature value is obtained by the neural network's output, which is normalized by the cell size $\Delta = \max(\Delta x_i)$ of the computational grid. For the neural network to be used in a flow simulation, it has to be trained and tested on a diverse and well-distributed data set. During training, the weights and biases of the neural network are determined and optimized to predict the curvature accurately. The neural network is considered deep if it has at least two hidden layers between the input and output layer [42]. The trained neural network, along with its corresponding weights and biases, is then implemented into the flow solver.

### Data Generation

To train and test neural networks, a comprehensive data set of pairs of volume fraction stencils and corresponding normalized curvatures of arbitrary fluid-fluid interface configurations is required. The data set should cover a wide range of different interface configurations to represent as many real situations as possible. The algorithm used in this study to generate such a data set is loosely based on the work of [95] and is explained in the following. Since the curvature is normalized, the domain with length $L = 1\,\mathrm{m}$ and the equidistant grid with CV size $\Delta = 10^{-3}\,\mathrm{m}$ do not change throughout the data generation process. The generation starts with choosing a curvature in the bounds of $\Delta\kappa_{\min} = 10^{-5}$ and $\Delta\kappa_{\max} = 0.4$, which are found by the precision of the neural network and by the minimal possible geometry in a stencil, respectively. Therefore, the obtained data is uniformly distributed over the curvature when a large data set is created and the neural network is trained equally well on different curvature values. Three geometries are used to generate the data set: a circle, an ellipse and a sine wave. Every geometry is parameterized in order

to automatically generate different configurations as a function of few parameters. The data set generation procedure can be summarized in:

1. Randomly choose a normalized curvature in the range of $\Delta\kappa_{\min} \leq \Delta\kappa \leq \Delta\kappa_{\max}$.
2. Set the geometry parameters randomly within boundaries that guarantee the existence of the chosen curvature.
3. Find the specific location of the chosen curvature.
4. Randomly rotate (around an angle $\theta$) and translate (up to one CV) the geometry and the location of the curvature.
5. Set the stencil of size $7 \times 7$.
6. Shift the stencil along the interface normal by a randomly generated length in the bounds of $\pm 2L\Delta$ to account for CVs located next to the exact interface.
7. Compute initial volume fractions in the stencil.
8. Create two separate smoothed volume fractions stencils and blend those with the initial stencil by a random number between 0.5 and 1.5.
9. Keep or invert volume fractions and curvature by a 50 % chance.
10. Create data set from curvature and volume fraction stencil.
11. Repeat steps 1-10 until desired amount of data sets are created.

**Interface Resharpening**

Although the data generation covers a broad range of interface configurations, the input stencil is optimized with an interface resharpening algorithm. It is introduced due to the diffuse input stencil in the flow simulation with the algebraic VOF method. The resharpening algorithm is employed for the input stencil in the data training as well as within the flow solver, thereby the consistency of the neural network-based curvature prediction on generic data and the usage within an actual simulation increases. The resharpening is performed in each coordinate direction separately and uniform borders are eliminated, which changes the $7 \times 7$ stencil into two $5 \times 5$ stencils (input elements only increase from 49 to 50). The algorithm sequence reads as:

1. Initialize the new $5 \times 5$ stencils.

2. Find CVs in the original stencil for which a neighbor is on the other side of the interface.

3. Determine the sharp interface location by interpolation between the CV found in step 2.

4. Assign a new volume fraction containing the sharp interface according to it (or the mean, if each direction proposes a new value).

5. Set the volume fraction in CVs outside the sharp interface to one or zero, depending on the previous affiliation.

Additionally the final volume fraction stencil is rotated and mirrored so that the normal on the interface always shows in the same octant. Hence, the effort in training the neural network is reduced and robustness is increased.

**ANN**

The trained neural network is directly available in FASTEST, but the data generation is carried out in a Python environment and the training and testing is done with Google's Tensorflow. First the Artificial Neural Network (ANN) is being initialized with the number of layers, the kind of layer, the nodes per layer, the activation function of the nodes as well as the initial distribution of the weights between the layers. Afterwards the ANN is compiled with setting the optimization method, the learning rate and the loss-function. The methods and functions chosen for the ANN can be found in Table 4.2. After the data

Table 4.2: Settings for ANN

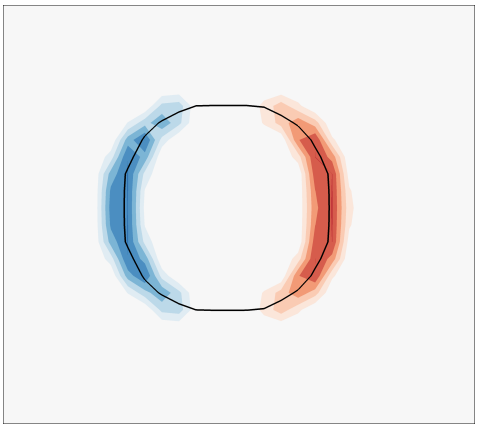
| Setting | Method or value |
|---|---|
| Activation function | Rectified Linear Unit (ReLU) |
| Loss function | Mean Square Error (MSE) of $\Delta\kappa$ |
| Initial weight distribution | He-Uniform |
| Optimization method | Adam |
| Learning rate | $10^{-4}$ |

generation procedure there are $1 \times 10^6$ data points available, which are divided into 70 %, 15 % and 15 % for training, testing and validation, respectively. Before passing the data sets from the generation and the interface resharpening to the training, they are divided into batches of size 128. An early stopping criterion is employed, so that the training is stopped if the validation error does not reduce further for several epochs (one epoch equals all batches) of training. Different neural networks varying up to five hidden layers and

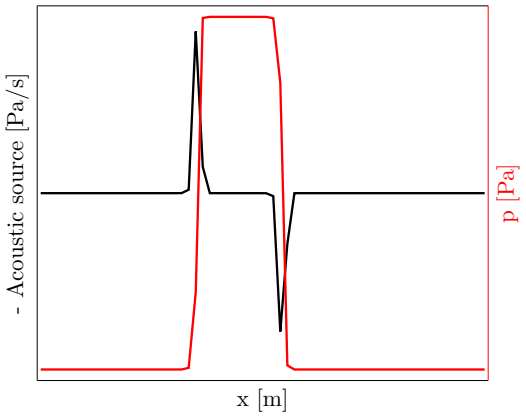
up to 200 nodes each layer are evaluated, with and without interface sharpening and the findings can be found in [58]. The most promising network configuration is trained with 200 nodes on each of three layers and with the interface resharpening algorithm for the input stencil.

## 4.3 Interface Acoustics

For computing acoustics with the LEE as presented in Section 2.3, the source term on the right-hand side of the acoustic pressure transport Equation (2.48) is the driver of every acoustic appearance. The change of the pressure from one time step to the next produces the acoustic source. In case of multifluid systems the pressure across an interface between the fluids changes, which is expressed by the equilibrium in Equation (3.49). Consider a moving water drop inside an air stream traveling with identical velocity, hence no relative movement between both phases exist. Figure 4.8a shows the acoustic source term at the interface in such configuration, additionally in Figure 4.8b the pressure jump and the acoustic source along the horizontal center line of the drop is displayed.



(a) Schematic of acoustic source term (blue=low, red=high) around moving interface)




(b) Pressure and acoustic source along horizontal middle line of water drop

Figure 4.8: Unphysical acoustic sources around fluid-fluid interface

However, when computing acoustics in the underlying framework consisting of the VOF method and the CSF formulation for the surface tension including their special discretization techniques (Section 3.2.3 and Section 3.2.4), the interface spans over several CV. Not only but especially if the interface moves, the pressure in these interface CV changes locally and introduces unphysical acoustic sources (see Figure 4.8b). In order to suppress these

erroneous sources in the interface region the temporal pressure gradient is corrected as

$$-\frac{\partial p}{\partial t} = \begin{cases} 0 & \text{if } \kappa \neq 0, \\ -\dfrac{\partial p}{\partial t} & \text{if } \kappa = 0. \end{cases} \tag{4.35}$$

Additionally, in case the interface movement should be the main contributor to the acoustic source, the pressure is corrected in each phase with the values from a point far away of the area of interest. For a two-phase system of fluid A and fluid B the correction yields

$$-\frac{\partial p}{\partial t} = \begin{cases} -\dfrac{\partial (p - p_{\text{ref, fluid A}})}{\partial t} & \text{if CV located in fluid A,} \\ -\dfrac{\partial (p - p_{\text{ref, fluid B}})}{\partial t} & \text{if CV located in fluid B.} \end{cases} \tag{4.36}$$

# 5. Numerical Examples

In this chapter numerical validations and examples are presented upon the framework presented in Chapter 3 and Chapter 4. First the curvature computation methods are validated on different configurations. Second the coupling between the acoustics and the multiphase part is examined. At the end, a full three-dimensional test case covering every aspect of this work is presented.

## 5.1 Curvature Validation

In order to get a fast, robust and accurate curvature value in each interface cell, different methods have been presented in Chapter 4. To validate these methods simple yet meaningful test cases are required. Since the CVOFLS curvature computation method has been analyzed in previous work [121], it is not explicitly examined. The accuracy of this model is outweighed by its high computational time, making it unusable for complex, three-dimensional and multiblock distributed simulations. However, for some test cases it will be considered as reference.
The process of the curvature validation begins with a static test case, for which the curvature computation models CFDM, HF, ANN and the effect of each improvement of Section 4.2.3 are investigated. The curvature validation is performed in two dimensions, to gain a general overview of the most promising combinations of each method and improvement. Except the machine learning approach, all curvature computation models are available in three dimensions. In a second validation test case, capillary oscillations around an equilibrium solution further characterize the performance of each curvature method. Due to the surface tension, the initially ellipse or square geometry oscillates around an equivalent circle until steady state is reached. The capillary oscillations are particularly suitable for testing curvature calculation methods because this represents the extreme case in which the fluid motion is caused purely by surface tension. In other applications, the motion induced by the surface tension is sometimes superimposed by a much stronger convective flow, which reduces the influence of the surface force and thus reduces the tolerance for incorrect curvature calculations.

Each method and its best combination with the possible improvements is carried out on different grid sizes to gain a comprehensive overview. After each test case a comparison across the methods and a conclusion are given.

### 5.1.1 Static Drop

The static drop test case in any form is based on Laplace's equilibrium (2.22) and the standard for most of the surface tension dominated multiphase work given in the literature. By placing a spherical drop in a static equilibrium the net surface force at each point should be zero due to the balancing of the surface tension and the pressure gradient. The correct solution to this problem is a zero velocity field and a pressure field that rises from a constant value $p_0$ outside the drop to $p_0 + \sigma/R$ inside the drop. The exact curvature for a circle is known as $\kappa_{\text{exact}} = 1/r$. Measuring the deviation from the correct solution with the curvature value itself, the magnitude of the velocity and the pressure jump across the interface evaluates the curvature computation for each model. Different norms and errors are employed to the physical quantities, which read as

$$L_2(\kappa) = \frac{1}{\kappa_{\text{exact}}} \sqrt{\frac{\sum_{j=1}^{N_I} (\kappa_j - \kappa_{\text{exact}})^2}{N_I}}, \tag{5.1}$$

$$L_\infty(\kappa) = \frac{1}{\kappa_{\text{exact}}} \max(|\kappa_j - \kappa_{\text{exact}}|), \tag{5.2}$$

$$L_\infty(u_i) = \max(|u_i|), \tag{5.3}$$

$$E(\Delta p_{\max}) = \frac{|\Delta p_{\max} - \Delta p_{\text{exact}}|}{\Delta p_{\text{exact}}}, \tag{5.4}$$

$$E(\Delta p_{\text{inout}}) = \frac{|\Delta p_{\text{inout}} - \Delta p_{\text{exact}}|}{\Delta p_{\text{exact}}}. \tag{5.5}$$

In the curvature norms (5.1) and (5.2) index $j$ counts to $N_I$, which is the number of interface CVs found by $\kappa_j \neq 0$. The difference between maximum and minimum pressure present in the domain $\Delta p_{\max} = |\max(p) - \min(p)|$ defines the error $E(\Delta p_{\max})$. The pressure error $E(\Delta p_{\text{inout}})$ depends on the average pressure inside and outside the drop as $\Delta p_{\text{inout}} = |\frac{1}{N_{\text{in}}} \sum_{l=1}^{N_{\text{in}}} p_l - \frac{1}{N_{\text{out}}} \sum_{m=1}^{N_{\text{out}}} p_m|$ with $N_{\text{in}}$ and $N_{\text{out}}$ being the number of CVs inside and outside the drop, respectively.
The computational domain of quadratic size with length $L = 8\,\text{m}$ is partitioned by 16, 32, 64, 128 and 256 CVs. Within the background fluid resides the drop of radius $R = 2\,\text{m}$ centered in the domain. The surface tension coefficient is increased to $\sigma = 73\,\text{N}\,\text{m}^{-1}$ leading to a pressure jump of $\Delta p_{\text{exact}} = 36.5\,\text{Pa}$ (2D) or $\Delta p_{\text{exact}} = 73\,\text{Pa}$ (3D). Initially every flow variable is set to zero demanding the solver to establish the pressure inside the drop. Convergence is reached if the change of every flow variable is beneath the criterion

of $\epsilon = 1 \times 10^{-4}$.
Each curvature computation method of Section 4.2 and every improvement (see Section 4.2.3) employed on it will be analyzed separately followed by a comparison of each best combination. In order to gain a first insight the analysis is carried out after one time step on $64^2$ CVs. The material properties can be found in Table 5.1.

Table 5.1: Material properties of static drop test case

| Property | Symbol | Unit | Surrounding | Drop |
|---|---|---|---|---|
| Density | $\rho$ | $[\mathrm{kg\,m^{-3}}]$ | 1 | 1000 |
| Viscosity | $\mu$ | $[\mathrm{kg\,m^{-1}\,s^{-1}}]$ | $1 \times 10^{-3}$ | $2 \times 10^{-5}$ |
| Surface tension coefficient | $\sigma$ | $[\mathrm{N\,m^{-1}}]$ | 73 | |

**CFDM**

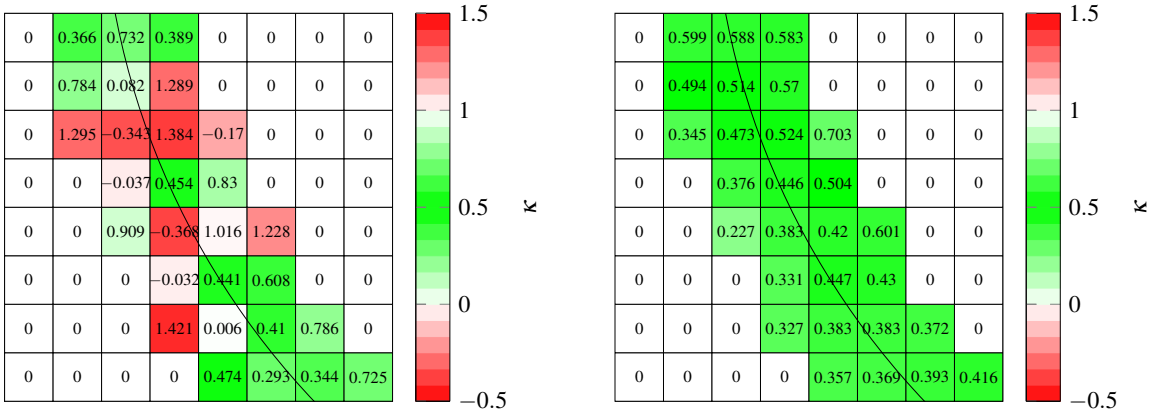
The CFDM differentiates the volume fraction distribution in every spatial direction twice, so that the curvature is estimated by a finite difference approximation as presented in Section 4.2.1. In Figure 5.1a discrete curvature values in the interface region around a small sector of the static drop are shown. Taking a look at the curvature of the standard method in Figure 5.1a, it is obvious that the values are far away from the exact curvature of $\kappa = 0.5\,\mathrm{m^{-1}}$. By employing a smoothing (denoted by $\tilde{}$) on the concentration and on the curvature and in addition weighting (denoted by $^*$) the (smoothed) curvature, the results improve significantly. Indicated by the green color, the improved discrete curvature values are shown in Figure 5.1b.

Every modification combination was tested and its outcome is shown in Figure 5.2. Although the least curvature error in Figure 5.2a is achieved by applying all of the named steps, the spurious velocities in Figure 5.2b are at the lowest by only weighting the smoothed curvature.

However, the pressure errors in Figure 5.2c are consistent with the best curvature result, so that the optimal combination taking all measurements into account is obtained by applying all improvements $(\tilde{F}, \tilde{\kappa}^*)$ to the standard CFDM (general denoted by Std.). The effect of the improvements on the curvature values is contrasted to the standard in Figure 5.1b.

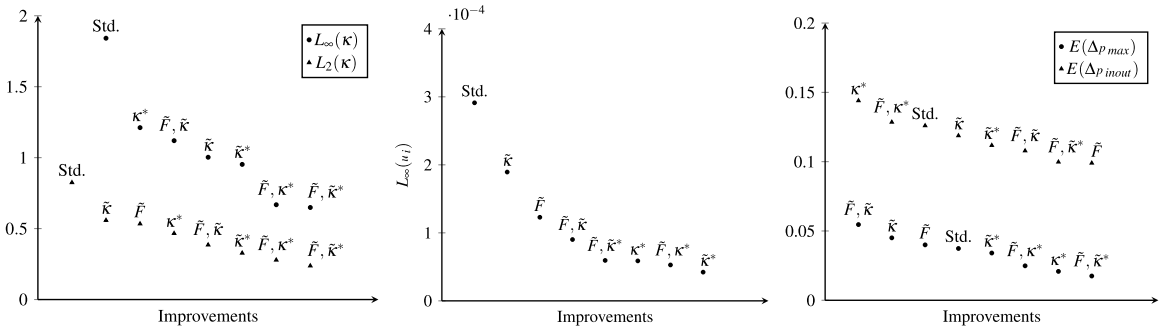
**HF**

The HF method consists of creating heights along the interface which are differentiated to obtain the curvature as described in Section 4.2.2. Since the improvements can be employed for the HF curvature computation as well, the impact is investigated and the results are




(a) CFDM (b) CFDM with improvements

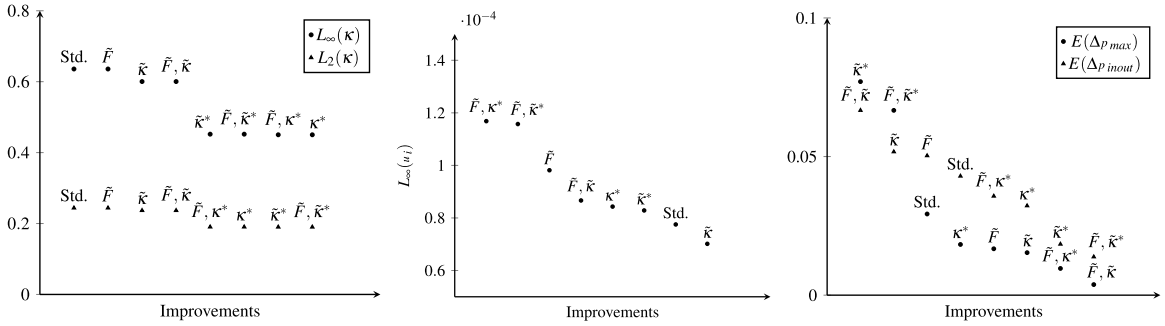
Figure 5.1: Discrete curvature values of CFDM without and with improvements after one time step on $64^2$ CVs




(a) Curvature error norms (b) Spurious velocities (c) Pressure jump errors

Figure 5.2: Curvature errors, spurious velocities and pressure jump error of CFDM and improvements

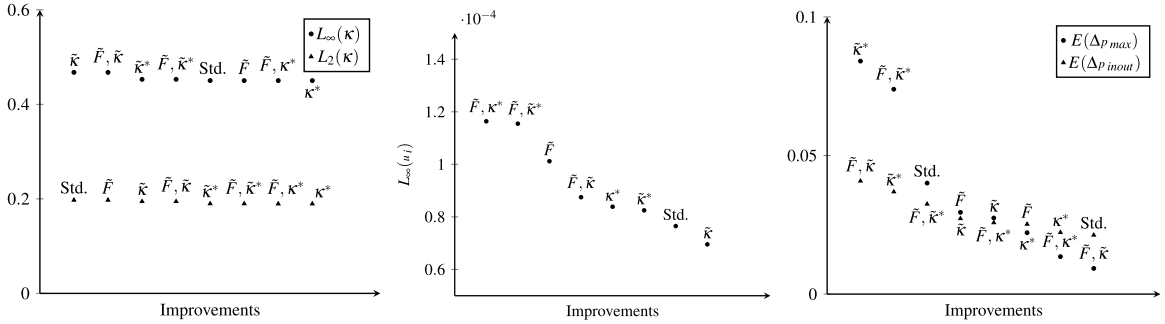
shown in Figure 5.3. The smoothing of the volume fraction ($\tilde{F}$) only changes the spatial derivatives for the height direction determination and the surface tension term (4.6), yet the smoothed values are not used for constructing the heights. For the standard HF method finding the best overall improvement is not as distinct as for the CFDM method. For the curvature error norms, most impact is achieved by smoothing the volume fraction and the curvature and weighting the curvature ($\tilde{F}, \tilde{\kappa}^*$) or only weighting the curvature ($\kappa^*$) as seen in Figure 5.3a. Least spurious velocities are produced after smoothing the curvature ($\tilde{\kappa}$ in Figure 5.3b). Best results for the pressure in Figure 5.3c are achieved by employing all improvements ($\tilde{F}, \tilde{\kappa}^*$) and by smoothing the volume fraction and the curvature ($\tilde{F}, \tilde{\kappa}$), followed by only curvature smoohting ($\tilde{\kappa}$) and curvature smoothing and weighting ($\tilde{\kappa}^*$). By ranking all results from one to seven and adding the ranks, the best combination is




(a) Curvature error norms (b) Spurious velocities (c) Pressure jump errors

Figure 5.3: Curvature errors, spurious velocities and pressure jump error of HF and improvements

smoothing the curvature.
The stencil on which the height is constructed can be set adaptively instead of a fixed size. On this HF approach the different improvement methods are employed and the results are shown in Figures 5.4.



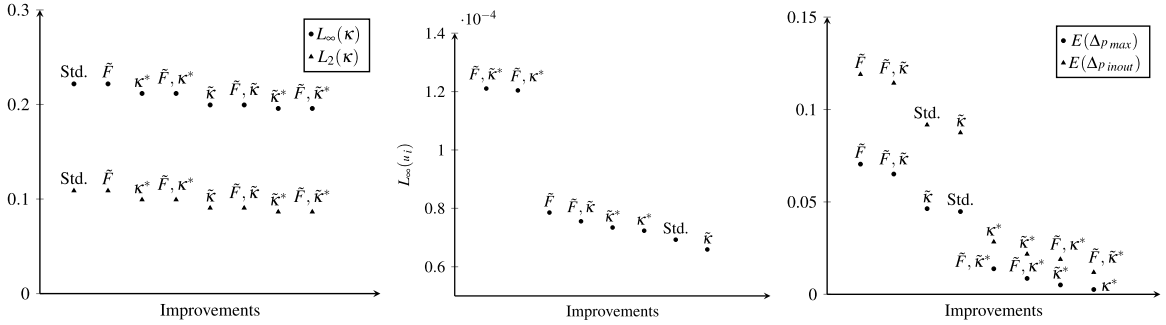

(a) Curvature error norms (b) Spurious velocities (c) Pressure jump error

Figure 5.4: Curvature errors, spurious velocities and pressure jump error of AHF and improvements

Identical to the fixed height, the weighting of the curvature shows the best overall result as well as for the curvature norms in Figure 5.4a. Spurious velocities in Figure 5.4b are low for curvature smoothing ($\tilde{\kappa}$). The in-out pressure jump error in Figure 5.4c gives the best results for the standard AHF, whereas the smoothing of the volume fraction and the curvature ($\tilde{F}, \tilde{\kappa}$). The ranking across all disciplines gives the smoothing of the curvature to be the best combination.

### ANN

Instead of computing the curvature by differentiating the volume fraction field in some manner, a trained ANN is able to return the curvature. The input data is a stencil of size $7 \times 7$ and the result is the curvature value of the interface cell, described in Section 4.2.5. As a curvature computation method rarely discussed in the literature, the ANN is only available for two-dimensional problems. Here, the network is trained with 200 nodes on each of three layers and the pre- and post-processing steps for the input stencil, explained in Section 4.2.5, are employed. The results are shown in Figure 5.5 additionally with the improvements to the curvature and the volume fraction.




(a) Curvature error norms (b) Spurious velocities (c) Pressure jump error

Figure 5.5: Curvature errors, spurious velocities and pressure jump error of ANN and improvements

Note that volume fraction is manipulated exclusively for the first derivatives, as in the HF method. The input of the ANN should be as identical as possible to the training data in order to produce accurate curvature results. The curvature error norms show decreasing behavior for all improvements in comparison to the standard method, with smoothing and weighting the curvature and smoothing the volume fraction ($\tilde{F}, \tilde{\kappa}^*$) having the least errors, shown in Figure 5.5a. In contrast the spurious velocities are at the highest with the least curvature error and the lowest for smoothing the curvature ($\tilde{\kappa}$), to be seen in Figure 5.5b. The least in-out pressure difference error is found to be with all improvements employed ($\tilde{F}, \tilde{\kappa}^*$), while the lowest maximum pressure difference results by only weighting the curvature ($\kappa^*$). Both noticeable in Figure 5.5c. Equal to the HF method, a ranking determines the best combination which is the curvature smoothing and weighting.

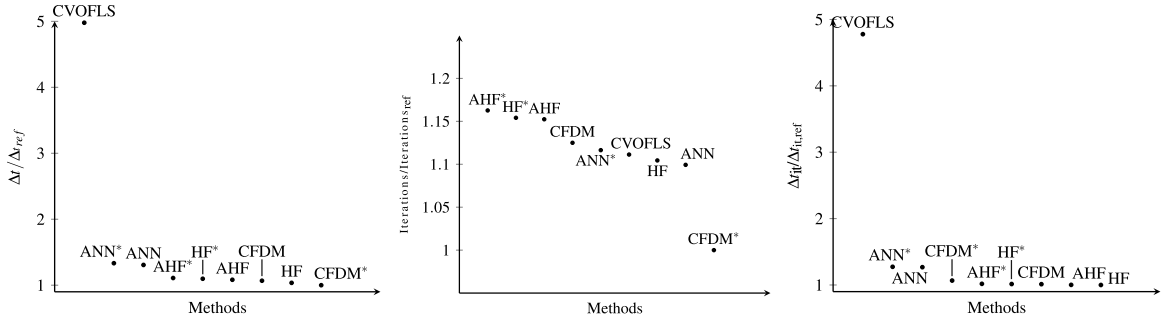
### Comparisons

After identifying the most promising combinations of each method with the improvements, an overview of the outcome is given in Table 5.2. With the most effective improvements and without them, a direct comparison of the curvature computation methods will be given in the following.

Table 5.2: Standard methods and their most effective improvement

| Method | Volume Fraction Smoothing | Curvature Smoothing | Curvature Weighting |
|---|---|---|---|
| CFDM | ✓ | ✓ | ✓ |
| HF | | | ✓ |
| AHF | | | ✓ |
| ANN | | ✓ | ✓ |

Starting with the computational time for one time step on the finest grid of $256^2$ CVs, the initial statement to the CVOFLS as being the slowest is proven by Figure 5.6a, in which the time is plotted relative to the time consumed by the fastest method. The computational time includes all inner iterations of the SIMPLE algorithm. The improved CFDM (CFDM*) is faster than the standard CFDM, reasoned by the fewer iterations taken for reaching convergence. The number of inner iterations are shown in Figure 5.6b and the time spent per iteration in Figure 5.6c, both scaled ot the smallest value in each plot.

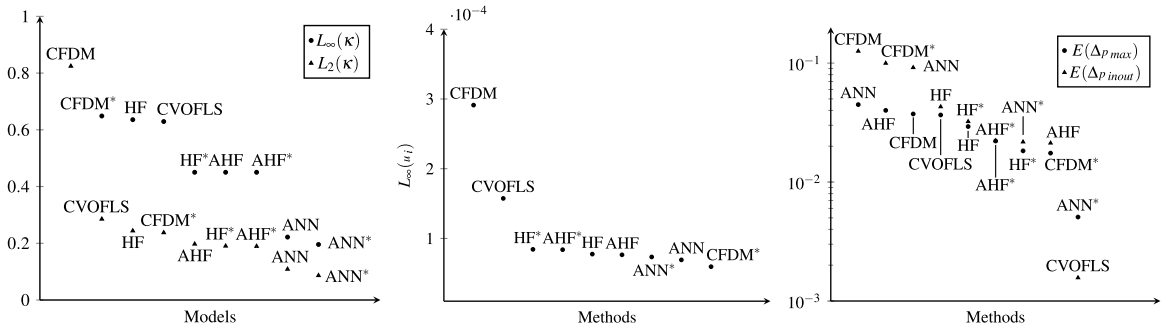



(a) Computational time (b) Iterations (c) Time per iteration

Figure 5.6: Computational time, inner iterations and time spent per iteration for one time step on $256^2$ CVs scaled to smallest value

However, considering the computational time in Figure 5.6a all HF and CFDM methods are in close range, while the ANN method is slightly slower. The time necessary to run through the improvement routines are less than 5 %.
In Figure 5.7 all methods without and with each best improvement employed, as per Table

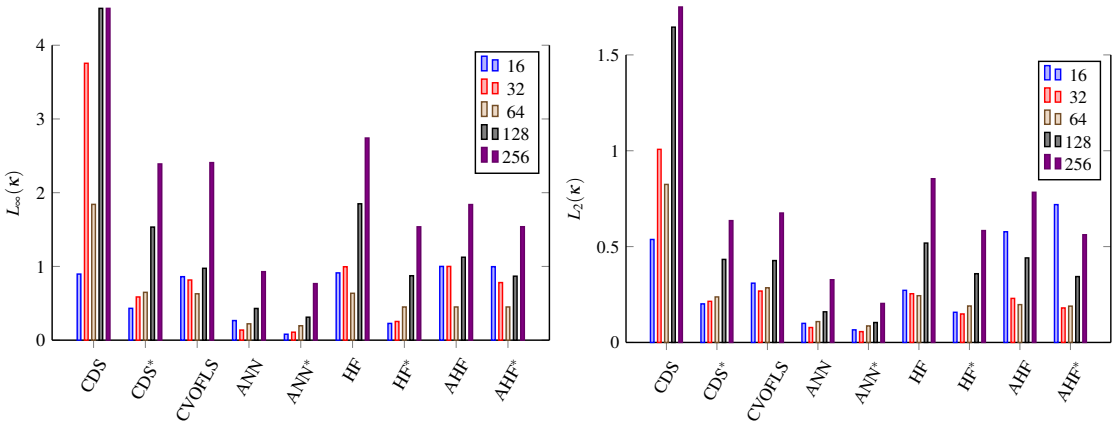
5.2, are compared on the reference grid of $64^2$ CVs. The ANN offers the best curvature values and the least maximum pressure error. However, the improved CFDM produces the least spurious velocities.




(a) Curvature error norms (b) Spurious velocities (c) Pressue jump error

Figure 5.7: Curvature errors of CFDM, HF and ANN without and with their best improvements (*) and CVOFLS after one time step on $64^2$ CVs

The accuracy of each method on different grid spacing is addressed next, starting at $16^2$ CVs up to $256^2$ CVs. Note that this not solely a geometrical review rather than the results after one time step. The errors shown in Figure 5.8 indicate that none of the presented methods truly converges with grid refinement in every discipline analyzed.

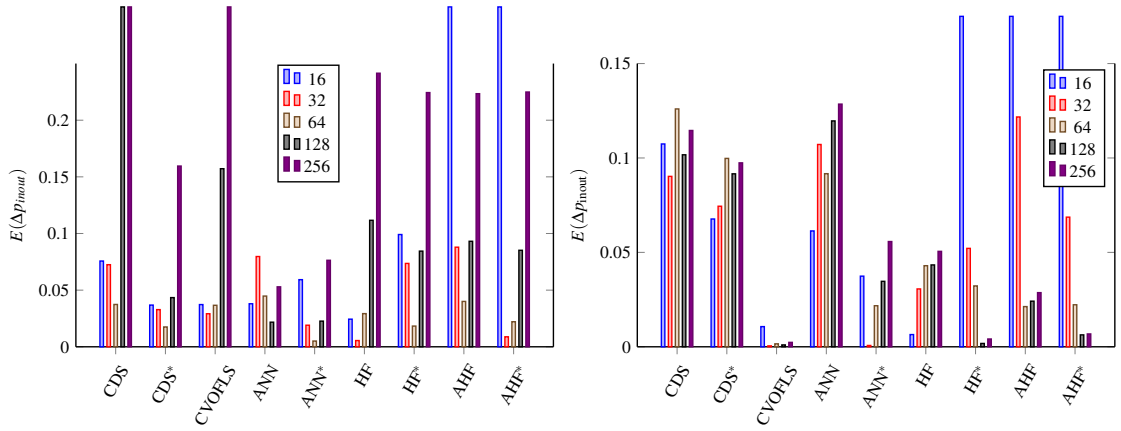



(a) Curvature error norm (b) Curvature error norm

Figure 5.8: Curvature grid convergence results for curvature methods

Especially with the HF methods, an optimal grid resolution for the stencil is found at 64 CVs. Note that the literature [16, 94] includes only cells meeting the HF condition for convergence studies, while here every interface cell is included into the error determination. However, except the pressure error on $16^2$ and $32^2$ CVs of the standard HF method (see

Figure 5.9), the best improvements of each method enhances the results on every grid level and in every aspect. The most impact of the improvements is found to be with the CFDM curvature computation method.




(a) Maximum pressure error

(b) In-out pressure error

Figure 5.9: Pressure grid convergence results for curvature methods

### 5.1.2 Capillary Oscillations

In this dynamic test case an ellipse or square is initially placed in the middle of the domain. Due to the acting surface tension the geometry tries to transform into a circle and overshoots the counterpart of its origin geometry until the equilibrium solution is reached. The ellipse oscillates in the second mode ($m = 2$) and the square in the fourth mode ($m = 4$), so that its oscillating frequency $\omega_m$ and period $\tau_m$ can be described analytically [60, 1] with

$$\omega_m = \sqrt{\frac{\sigma m(m-1)(m+1)}{(\rho_1+\rho_2)\overline{R}^3}}, \tag{5.6}$$

$$\tau_m^* = \frac{\rho_1 \overline{R}^2}{\mu_1(2m(m-1))}, \tag{5.7}$$

$$\omega_m^* = \sqrt{\omega_m(1-(\omega_m\tau_m)^{-2})}, \tag{5.8}$$

$$\tau_m = \frac{2\pi}{\omega_m^*}. \tag{5.9}$$

The viscosity of the fluid reduces the natural oscillation frequency [10], which is considered through the intermediate variables $\omega^*$ and $\tau^*$. For the geometries an average radius $\overline{R}$ is determined from the semi-axes $a$ and $b$ of each geometry. The geometrical data is listed in

Table 5.3 and the material properties in Table 5.4, both are chosen in reference to [149] and [137]. The two dimensional domain is of size $L = 0.0075\,\mathrm{m}$, discretized with 64, 128 and

Table 5.3: Geometrical properties of capillary oscillation test case

| Property | Symbol | Unit | Ellipse | Square |
|---|---|---|---|---|
| Semi-axis | $a$ | [m] | 0.016893 | 0.04 |
| Semi-axis | $b$ | [m] | 0.028226 | 0.04 |
| Average radius | $\overline{R}$ | [m] | 0.02256 | |
| Domain | $L$ | [m] | 0.075 | |
| Analytical period | $\tau_0$ | [s] | 1.5918 | 0.5025 |

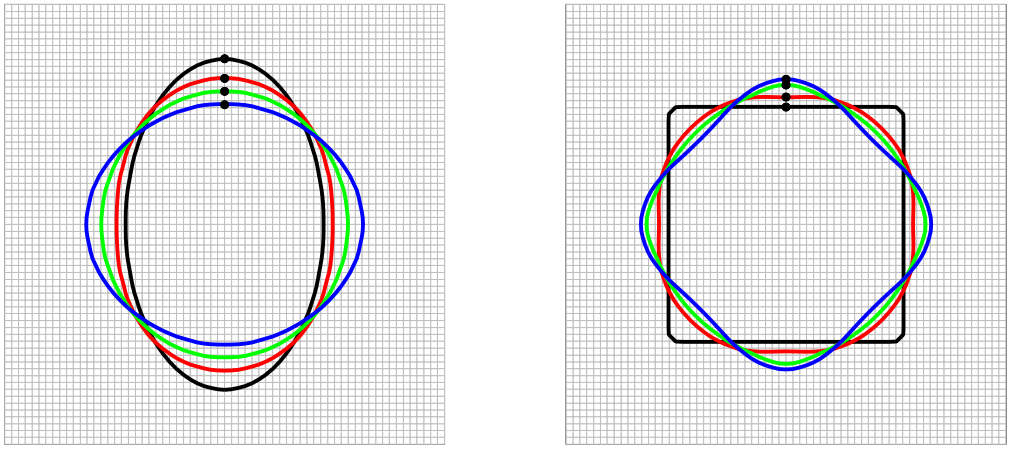
Table 5.4: Material properties of capillary oscillation test case

| Property | Symbol | Unit | Surrounding | Geometry |
|---|---|---|---|---|
| Density | $\rho$ | $[\mathrm{kg\,m^{-3}}]$ | 1.1768 | 787.88 |
| Viscosity | $\mu$ | $[\mathrm{kg\,m^{-1}\,s^{-1}}]$ | 0.002 | 0.024 |
| Surface tension coefficient | $\sigma$ | $[\mathrm{N\,m^{-1}}]$ | 0.02361 | |

256 CVs in each spatial direction. The time step size in the elliptical drop test case is set to $\Delta t = 2.5 \times 10^{-3}\,\mathrm{s}$ for the lowest grid size and $\Delta t = 1 \times 10^{-3}\,\mathrm{s}$ for the middle and finest grid. The motion of the square droplet is more dynamic, so that $\Delta t = 1 \times 10^{-3}\,\mathrm{s}$ is set for 64 CVs, $\Delta t = 6.25 \times 10^{-4}\,\mathrm{s}$ in case of 128 CVs and $\Delta t = 3.125 \times 10^{-4}\,\mathrm{s}$ for 256 CVs. For the investigation of the oscillation period, the north point of each geometry is tracked as shown in Figures 5.10a and 5.10b. The position of the north point is approximated with linear interpolation along the vertical middle line, starting at the center of the geometry. If the derivative of the y-motion changes its sign, the time is stored and the error in percentage is computed as

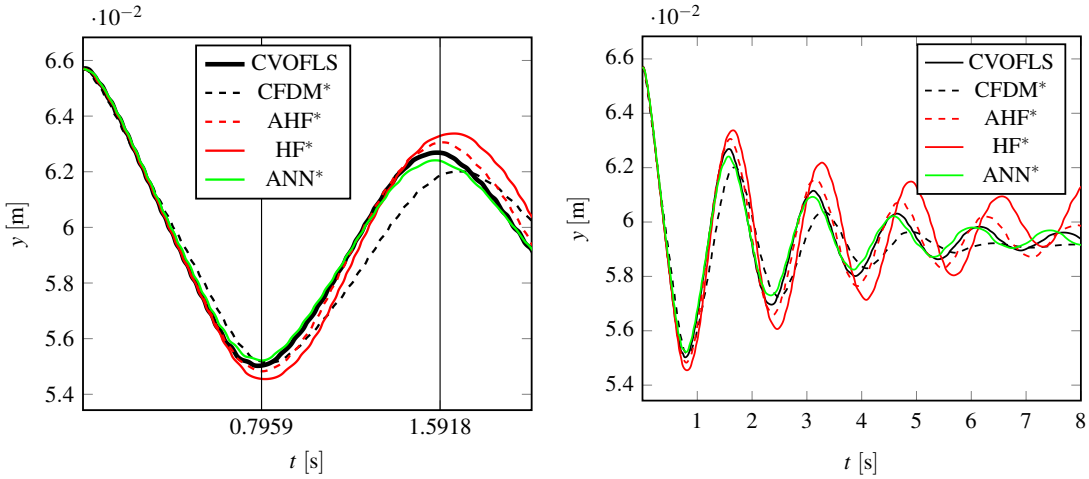
$$E(\tau_m) = \frac{|\tau_m - \tau_{m,\mathrm{analytic}}|}{\tau_{m,\mathrm{analytic}}} \cdot 100. \tag{5.10}$$

After initially distorted by 25 % the elliptical droplet starts to oscillate in a certain frequency with decreasing amplitude. The graphical representation of the simulation results on $128^2$ CVs are shown in Figure 5.11, accompanied by the analytical period and its multiples marked by solid vertical lines in the enlarged view of Figure 5.11a. The first period of each curvature method and the corresponding error $E(\tau_2)$ are listed in Table 5.5. The CVOFLS method serves as a reference for the amplitude as it agrees with the analytical period across all grids within 1.5 % and lower. It is difficult to make a general and universally valid statement about the best curvature method. As seen in the presented results, the improved ANN shows the best results on $64^2$ CVs and $256^2$ CVs, while the improved



(a) Ellipse (b) Square

Figure 5.10: North point movement in capillary test cases




(a) Ellipse until first period (b) Ellipse until $t = 8\,\mathrm{s}$

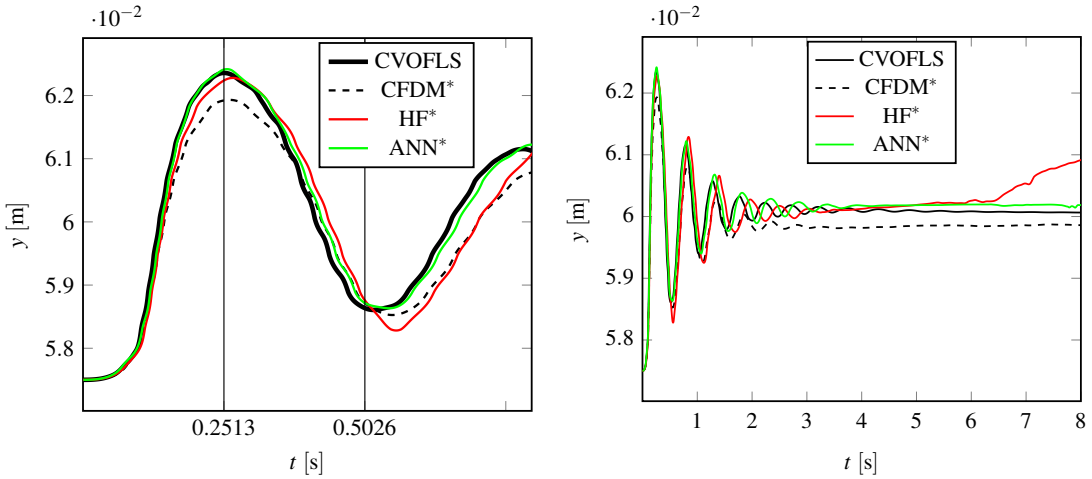
Figure 5.11: North point movement of oscillating ellipse

AHF is the best on $128^2$ CVs. However, by looking at the amplitude of the oscillation in Figure 5.11b it is obvious that the ANN* is in good agreement with the reference CVOFLS. Although the improved CFDM produces the worst results, the error is in the range of 6 %. Considering a longer simulation time, the curvature values in the improved HF method lead to lesser damping with tendencies to instabilities towards the end at $t = 8\,\mathrm{s}$. In contrary, the improved CFDM shows an excessive damping behavior leading to smaller amplitudes with a stable ending.
Replacing the elliptic geometry in the middle of the domain with a square, the frequency

Table 5.5: Ellipse oscillation time period results

| Method | $64^2$ | | $128^2$ | | $256^2$ | |
|---|---|---|---|---|---|---|
| | $\tau_2$ [s] | $E(\tau_2)$ [%] | $\tau_2$ [s] | $E(\tau_2)$ [%] | $\tau_2$ [s] | $E(\tau_2)$ [%] |
| Analytical | 1.592 | 0 | 1.592 | 0 | 1.592 | 0 |
| CVOFLS | 1.5675 | 1.530 | 1.5800 | 0.744 | 1.585 | 0.430 |
| CFDM* | 1.6850 | 5.852 | 1.6850 | 5.852 | 1.69 | 6.166 |
| HF* | 1.6450 | 3.339 | 1.6575 | 4.124 | 1.699 | 6.731 |
| AHF* | 1.6025 | 0.669 | 1.6050 | 0.826 | 1.63 | 2.397 |
| ANN* | 1.5925 | 0.041 | 1.5750 | 1.058 | 1.579 | 0.807 |

mode increases to $m = 4$. Figure 5.12 shows the north point movement of the simulation on $128^2$ CVs at the beginning and until $t = 8\,\mathrm{s}$.




(a) Square until first period

(b) Square until $t = 8\,\mathrm{s}$

Figure 5.12: North point movement of oscillating square

Table 5.6 supplements the corresponding numerical values determined in the same way as for the ellipse. Initially, the distribution of the volume fraction values of the square geometry is quite challenging for the curvature methods as it has four sharp edges connected by straight lines. Although the sharp edges are smoothed the curvature is high, while on the straight lines the curvature tends to be infinite small. The AHF fails to compute the curvature for the square at the beginning, hence no desired oscillation occurs. For that reason, it is left out of the results.
Nevertheless, the simulation of the higher frequency mode shows similar results to the lower case. The ANN with its improvements is the most accurate across all grid resolutions in

Table 5.6: Square oscillation time period results

| Method | $64^2$ | | $128^2$ | | $256^2$ | |
|---|---|---|---|---|---|---|
| | $\tau_4$ [s] | $E(\tau_4)$ [%] | $\tau_4$ [s] | $E(\tau_4)$ [%] | $\tau_4$ [s] | $E(\tau_4)$ [%] |
| Analytical | 0.5026 | | 0.5026 | | 0.5026 | |
| CVOFLS | 0.522 | 3.869 | 0.518 | 3.099 | 0.519 | 3.348 |
| CFDM* | 0.561 | 11.629 | 0.552 | 9.815 | 0.548 | 9.005 |
| HF* | 0.569 | 13.221 | 0.559 | 11.307 | 0.566 | 12.551 |
| ANN* | 0.560 | 11.430 | 0.546 | 8.694 | 0.543 | 7.948 |

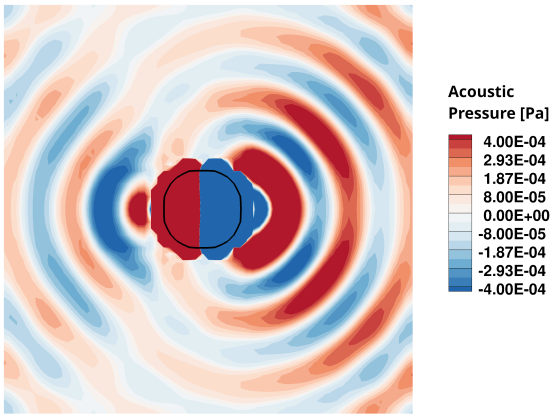
terms of period and amplitude, shown in Figure 5.12a. As before, the HF fails to keep the simulation stable while the CFDM* and its over proportional damping shows no sign of movement after $t = 4\,\mathrm{s}$, to be seen in Figure 5.12b.
In summary, the CVOFLS justifies its role as reference since it produces small errors while being stable for any grid resolution. However, the computational time is five times greater than with other methods, which makes its application impractical. Depending on the grid resolution and the test case, the AHF shows promising results, which may be enhanced by an even more sophisticated stencil construction [28] or by combining other methods [101]. Nevertheless, for the square geometry the AHF was not able to compute the initial curvature values at the edges leading to an overall failure.
Although having similar results to the CFDM in the static drop test case, the HF with fixed stencil size is the least accurate curvature method for the capillary test case. Especially, the unstable behavior for both geometries is not tolerable. As fastest curvature method of those tested and simultaneously being unconditionally stable, the CFDM with its improvements turns out as versatile employable. At least, the ANN curvature computation shows excellent results in every aspect and more effort in its development has to be made.

## 5.2 Coupling between Acoustics and Multiphase

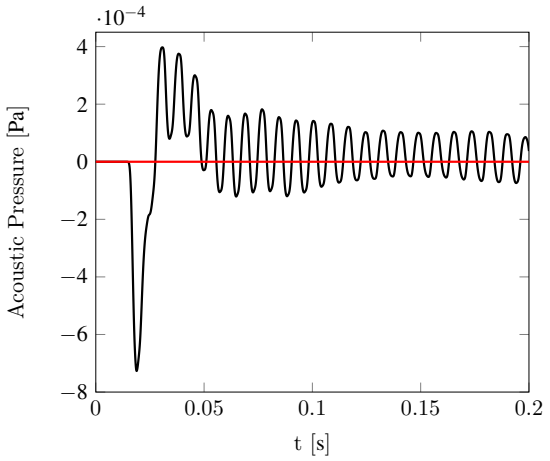
In this section, the generation of the acoustic sources by surface-tension-driven multiphase flow is presented. At first, the interface treatment of Section 4.3 is verified by two test cases. In both test cases different phases and acoustics are computed, but in the first configuration an acoustic signal should not occur while in the second it should be produced by the interface movement [37]. Secondly, a three-dimensional test case considering airborne acoustics triggered indirectly through nearby oscillations is presented. Note that each single part of the solver has been verified in [141, 57].

### 5.2.1 Verification of the Interface Acoustics

For the first verification of the coupling between the acoustics and the multiphase part, a water drop (fluid 1) in form of an ellipsoid is placed inside a duct filled with air (fluid 2) having a density ratio of $\rho_1/\rho_2 = 1 \times 10^3$ and a viscosity ratio of $\mu_1/\mu_2 = 1 \times 10^2$. A uniform flow field with a constant velocity in $x$-direction $u = 0.1\,\mathrm{m\,s^{-1}}$ is set. The time step size is $\Delta t = 2 \times 10^{-5}\,\mathrm{s}$ and the simulation time is $T = 0.2\,\mathrm{s}$. The duct has a length in both directions of 0.05 m, divided by $64^2$ CVs resulting in the uniform cell size $\Delta h = 7.8125 \times 10^{-4}\,\mathrm{m}$. The drop radius is $r = 5 \times 10^{-3}\,\mathrm{m}$ and the drop starts at $(x, y) = (0.0125\,\mathrm{m}, 0.025\,\mathrm{m})$. Referring to Equation (2.22) with constant surface tension $\sigma = 0.08\,\mathrm{N\,m^{-1}}$, the exact pressure jump is $\Delta p = 16\,\mathrm{Pa}$. To focus on the moving interface the curvature is kept constant and moreover the speed of sound for both fluids is set to $c = 1\,\mathrm{m\,s^{-1}}$. Beside the inlet on the left side and the outlet on the right side, for the top and the bottom of the domain symmetric boundary conditions are applied. All four boundary conditions act like acoustic outlets.



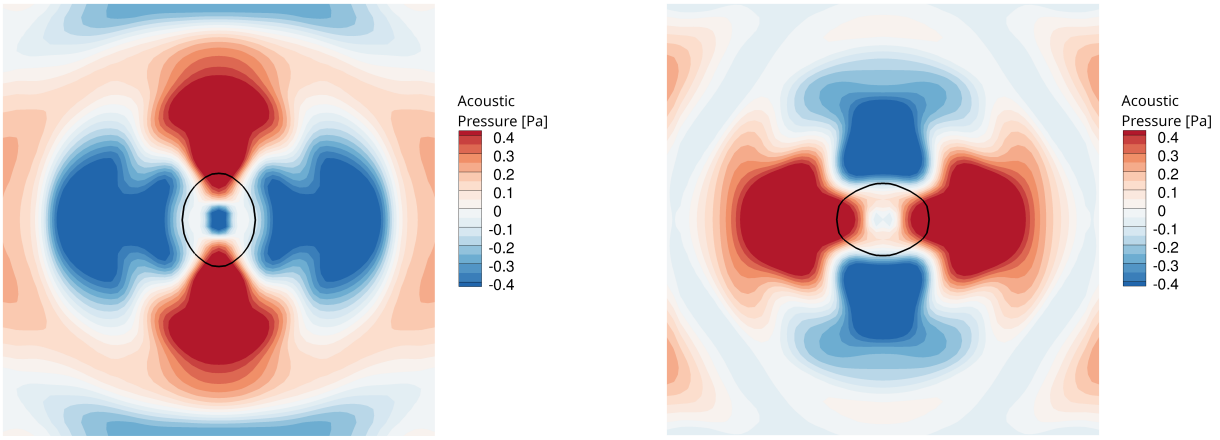

(a) Acoustic pressure at $t = 0.1177\,\mathrm{s}$ with volume fraction isoline for $F = 0.5$




(b) Comparison of acoustic pressure at point $P$

Since the constant drop and the background fluid move simultaneously no acoustics should be produced. In Figure 5.13b, the acoustic pressure contour plot is shown at time $t = 0.011\,77\,\mathrm{s}$. The sources at the front and the back of the drop produce acoustic waves which are emitted into the domain and inside the drop. Waves inside the drop travel until the opposite side of the interface where they are nearly completely reflected due to the material parameters of the fluids. Exemplary taking the acoustic pressure at point $P = (0.025\,\mathrm{m}, 0.045\,\mathrm{m})$ over the simulation time, one can see in Figure 5.13b that the magnitude is in a range of $10^{-4}\,\mathrm{Pa}$. If the simulation is repeated without the sources of the interface, as presented in Section 4.3, the outcome is as expected. The magnitude of the acoustic pressure in point $P$ is about $10^{-17}\,\mathrm{Pa}$. Comparing the two simulations, the

regime with the suppressed sources appears as a red line in Figure 5.13b. For this test case suppressing the sources in the interface region leads to negligible acoustic pressure in the domain.
Whereas in the previous test case a moving interface should not produce sound, the moving interface in the second test case should emit acoustic waves. Similar to Section 5.1.2, an elliptic drop with two semi-axes of $a = 0.005\,\mathrm{m}$ and $b = 0.003\,\mathrm{m}$ is placed in the center of the domain. The lengths and discretization are the same as in the previous test case. Due to the acting surface tension the ellipse oscillates around its equivalent circle shape, while the background fluid is at rest. It is expected that the acoustic pressure emitted from the ellipse has the same frequency as the oscillation itself. The simulation parameters are: $\Delta t = 3 \times 10^{-7}\,\mathrm{s}$, $\sigma = 0.08\,\mathrm{N\,m^{-1}}$, $\rho_1/\rho_2 = 1$, $\mu_1/\mu_2 = 1$ and $T = 0.15\,\mathrm{s}$.




(a) Maximum deflection in $y$-direction

(b) Maximum deflection in $x$-direction

Figure 5.14: Acoustic pressure contour of oscillating drop with concentration isoline for $F = 0.5$

Figures 5.14a and 5.14b show the acoustic pressure contour plots at approximately maximum drop deflections. In Figure 5.14a it can be seen that the drop emits acoustic signals in the north and south direction with positive acoustic pressure, corresponding to the ellipse deflection. As expected, the results in Figure 5.14b are inverted as the orientation of ellipse deflection. The frequency of the drop oscillation is 60.1 Hz. In Figure 5.15 the Fourier analysis of the acoustic pressure is taken at point $Q = (0.25\,\mathrm{m}, 0.35\,\mathrm{m})$. The maximum amplitude occurs at a frequency of 58.6 Hz. The difference between the frequencies of the drop oscillation and the acoustic pressure is within 2.5 % and therefore in a good agreement.

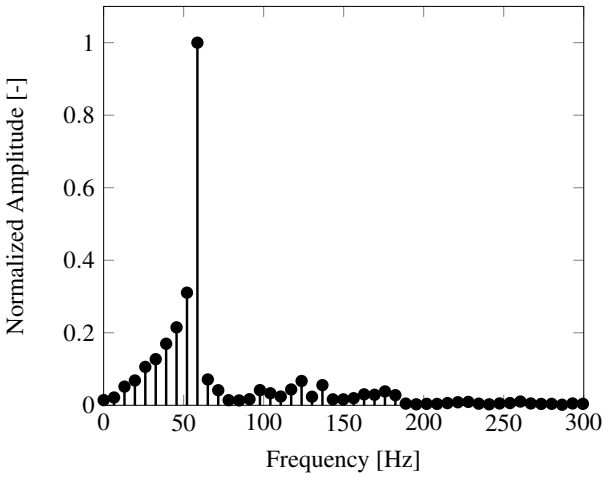



Figure 5.15: Fourier analysis of acoustic pressure at point $Q$

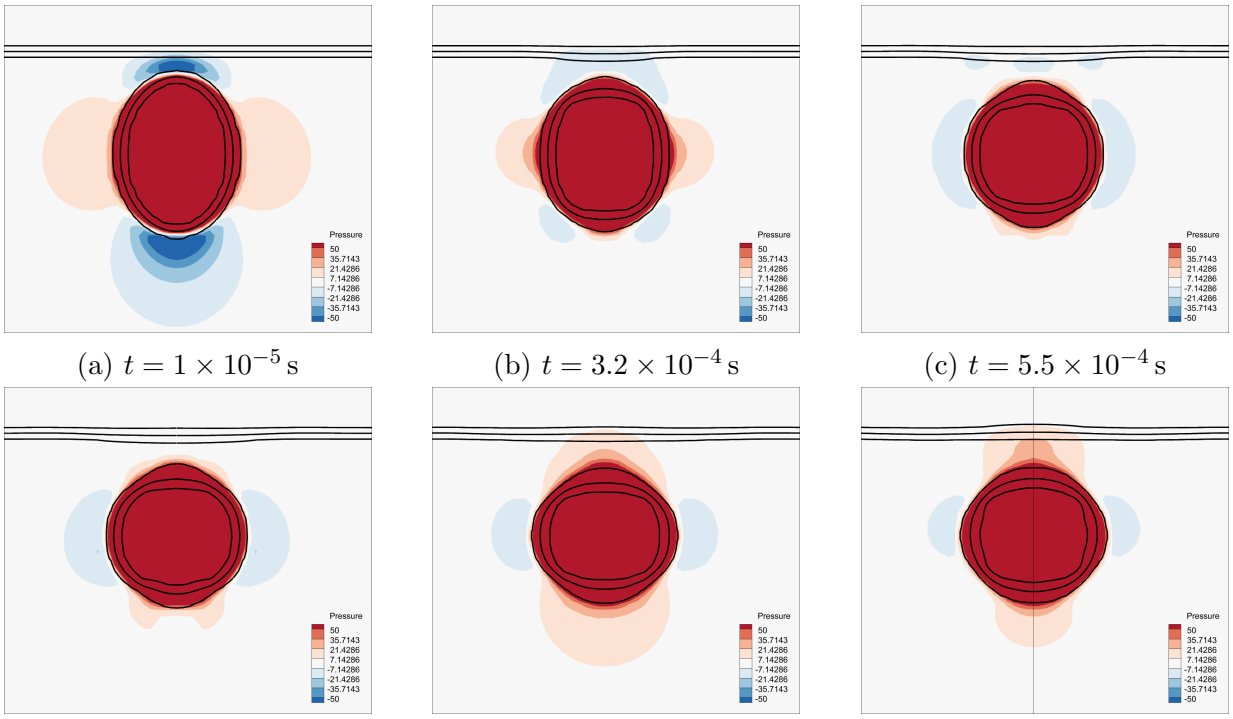
### 5.2.2 Acoustics at Nearby Surface

In order to check if the numerical methods presented in this work are suitable for the following drop impact test case, a preliminary three-dimensional test case is set up [38]. According to Phillips et al. [97] the main driver of the drop impact acoustics is the bubble oscillation after the detachment. Therefore, an ellipsoid-shaped air bubble, defined by the major axis of $0.923 \times 10^{-3}\,\mathrm{m}$ and the minor axis of $0.675 \times 10^{-3}\,\mathrm{m}$, is initialized at $(0.0025\,\mathrm{m}, 0.0015\,\mathrm{m}, 0.0025\,\mathrm{m})$ directly under a water surface at $0.139 \times 10^{-3}\,\mathrm{m}$. The grid consists of $128^3$ CVs and the material parameters for air and water are listed in Table 5.7.

Table 5.7: Material properties of air and water

| Property | Symbol | Unit | Air | Water |
|---|---|---|---|---|
| Density | $\rho$ | $[\mathrm{kg\,m^{-3}}]$ | 1 | 1000 |
| Viscosity | $\mu$ | $[\mathrm{kg\,m^{-1}\,s^{-1}}]$ | $1.7 \times 10^{-5}$ | $1 \times 10^{-3}$ |
| Speed of Sound | $c$ | $[\mathrm{m\,s^{-1}}]$ | 1 | 4.32 |
| Surface tension | $\sigma$ | $[\mathrm{kg\,s^{-2}}]$ | 0.074 | |

The speed of sound is scaled due to the comparable small cubic domain of $L = 5 \times 10^{-3}\,\mathrm{m}$. The surface tension model is chosen to be the improved CFDM, as it is a full three-dimensional test case running on eight processors in parallel. Similar to the capillary oscillation test cases, surface tension will force the ellipsoid to return into a spherical shape which leads to an oscillation of the bubble. In Figure 5.16 a series of six time steps showing a slice through the $xy$-plane at constant $z = 2.5 \times 10^{-3}\,\mathrm{m}$ with the pressure as contour, ranging from positive in red to negative in blue. The black isolines represent




(a) $t = 1 \times 10^{-5}\,\mathrm{s}$ (b) $t = 3.2 \times 10^{-4}\,\mathrm{s}$ (c) $t = 5.5 \times 10^{-4}\,\mathrm{s}$

(d) $t = 6.1 \times 10^{-4}\,\mathrm{s}$ (e) $t = 9.3 \times 10^{-4}\,\mathrm{s}$ (f) $t = 1.15 \times 10^{-3}\,\mathrm{s}$

Figure 5.16: Pressure of oscillating ellipsoid driving the motion of above surface with volume fraction isolines at $F = 0.1, 0.5, 0.9$

the volume fraction at values $F = 0.1, 0.5, 0.9$. The under-pressure between the bubble and the surface in Figure 5.16a leads to a downward movement, see Figures 5.16b and 5.16c. After the point of inflection in Figure 5.16d, the pressure rises and pushes the above surface in the upward direction (Figures 5.16e and 5.16f). The relative movement of the water surface to the surrounding air drives the acoustics in the air. The acoustic pressure above the surface and the surface motion are compared with respect to their frequency and phase shift. The airborne signal taken at a monitoring point $(0.003\,\mathrm{m}, 0.004\,\mathrm{m}, 0.003\,\mathrm{m})$ and the surface movement right above the bubble are shown in Figure 5.17. Not taking into account the first peak of each signal, the frequency of the following four peaks for the airborne regime can be found at 664 Hz, which is less than 10 % diverging from the surface oscillation frequency with 724 Hz. In addition to the frequency, the shift in time of the two signals is used for validation. Between the first upward directed peak of the surface at $1.38 \times 10^{-3}\,\mathrm{s}$ and the first positive peak of the acoustic pressure at $2.52 \times 10^{-3}\,\mathrm{s}$, there is a difference of $\Delta t^1_{\mathrm{signals}} = 1.14 \times 10^{-3}\,\mathrm{s}$. Between the first downward directed peak of the surface at $2.13 \times 10^{-3}\,\mathrm{s}$ and the first negative peak of the acoustic pressure at $3.46 \times 10^{-3}\,\mathrm{s}$, the difference in time is $\Delta t^2_{\mathrm{signals}} = 1.33 \times 10^{-3}\,\mathrm{s}$. The distance of the first CVs above the surface, in which the acoustic source is not suppressed, to the monitoring point is

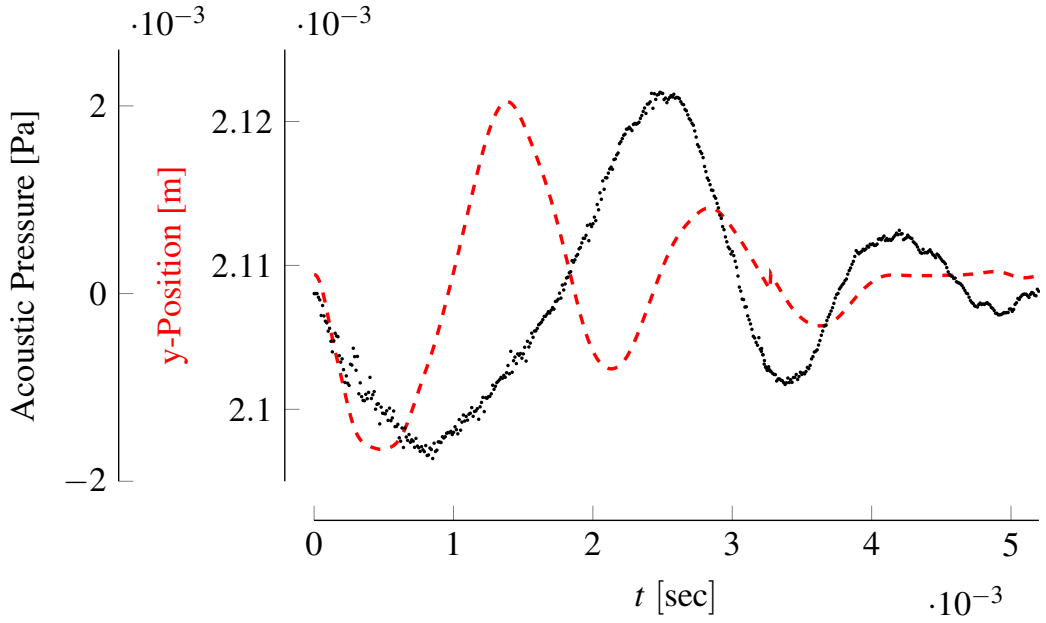
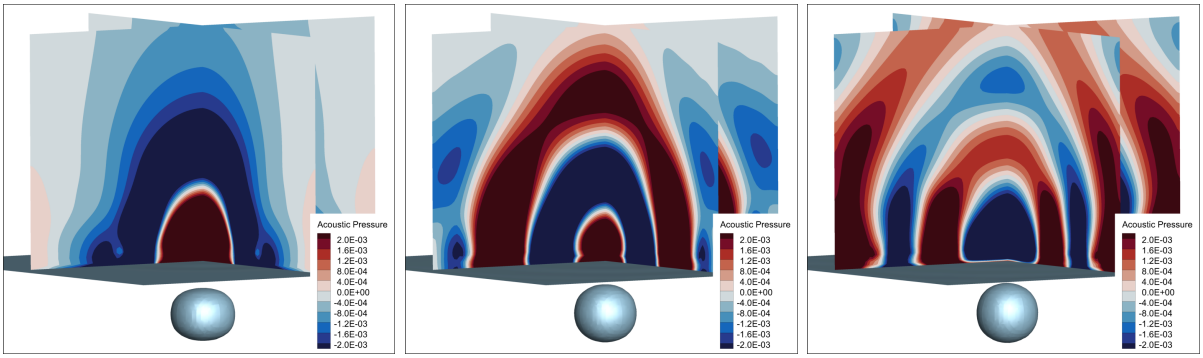



Figure 5.17: Airborne acoustics of preliminary test case (black) and surface position (red)

$\Delta d = 1.52 \times 10^{-3}\,\mathrm{m}$. With a speed of sound of $1\,\mathrm{m\,s^{-1}}$, the phase shift between the surface movement and the acoustic signal is in plausible range so that $\Delta d \approx c \cdot \Delta t^{1,2}_{\text{signals}}$. Due to the scaled speed of sound, the acoustic pressure above the surface at three points in time is made visible in Figure 5.18. The preliminary test case provides a promising result and therefore the next step is to apply the methods to the more complex drop impact test case.




(a) $t = 1 \times 10^{-3}\,\mathrm{s}$ (b) $t = 2.5 \times 10^{-3}\,\mathrm{s}$ (c) $t = 3.5 \times 10^{-3}\,\mathrm{s}$

Figure 5.18: Airborne acoustic pressure of oscillating surface driven by the motion of an oscillating bubble and volume fraction isosurface at $F = 0.5$

## 5.3 Drop Impact Acoustics

The final numerical example combines all models presented in this work. The physical background of this test case was first investigated in the 1920's [73]. A liquid drop impacts under certain conditions into a larger body of liquid, which then produces a sound. A common phenomena as it can be observed by a tripping tap releasing water drops into a sink full of water or raindrops falling onto a lake. Various researches tried to find the source of the sound. In 1959 Franz [36] concluded through the comparison of photographs and hydrophones that the formation of a bubble in the liquid was found to coincide with the start of a sound pulse. Thirty years later, Pumphrey et al. [104] investigated the circumstances in which the air bubble is formed and stated that the sound pulse is emitted by the pulsating air bubble under the surface of the water during the drop impact. They concluded by their studies that the "regular bubble entrainment" only occurs under certain conditions, which were later characterized by the Weber number

$$\mathrm{We} = \frac{\rho U_\mathrm{I}^2 D}{\sigma} \tag{5.11}$$

and the Froude number

$$\mathrm{Fr} = \frac{U_\mathrm{I}^2}{gD}, \tag{5.12}$$

where $g$ is the acceleration due to gravity, $U_\mathrm{I}$ is the impact velocity, $D$ is the drop diameter and $\sigma$ is the surface tension. Oguz et al. [88] found that the boundaries of the regular

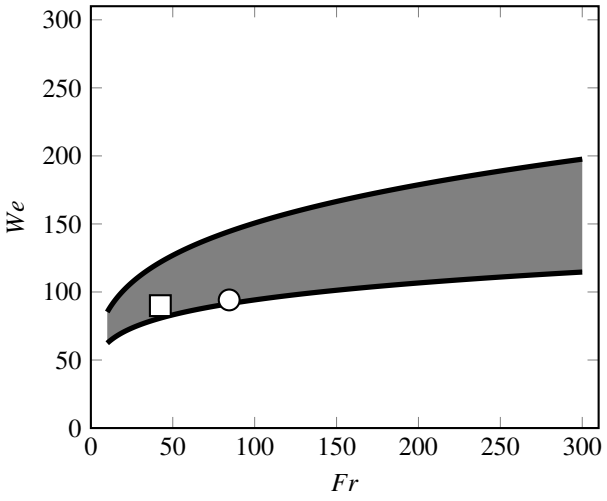



Figure 5.19: Froude-Weber diagram with shaded regular entrainment region based on Oguz et al. [88]; square represents the test case by Phillips et al. [97]; circle represents the test case by Morton et al. [80]

entrainment region, shown as shaded area in Figure 5.19, could be approximated by the following power law relation:

$$We = A \cdot Fr^{\beta}, \tag{5.13}$$

in which the upper boundary is given by $A = 48.3$, $\beta = 0.247$ and the lower boundary by $A = 41.3$, $\beta = 0.179$. In this region a bubble detaches from the bottom of the cavity. It was assumed by Leighton [63] in 2012 that the airborne sound results from the underwater sound field propagating through the water-air interface. However, recently progress was made in determining the source of the characteristic so-called "pling" sound by Phillips et al. [97]. By increasing the performance of the experimental equipment used for capturing the fluid and the acoustics, the group of Phillips contradicts Leighton's assumption by proposing the theory that the airborne sound is produced by an oscillation of the cavity bottom. Like a piston in a baffle, the motion of the cavity bottom is driven by the oscillation of the detached air bubble shown in Figure 5.20.

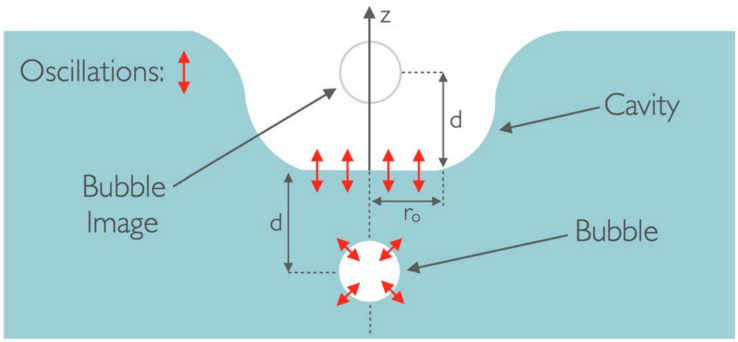



Figure 5.20: Schema of drop impact airborne sound source by Phillips et al. [97]

Various methods solving the acoustics and multiphase part are existent, but few works handle a coupled simulation. For example, the sound generation and propagation in two phases were simulated by Tajiri et al. [131] with the help of a finite difference lattice Boltzmann method. For simulating a such physically rich test case, the task is to find an optimal balance between computational time and physical correctness. The considerations concerning spatial and temporal resolutions are presented in the following.

### 5.3.1 Spatial Resolution

Morton et al. [80] stated in their work that a certain resolution is necessary to capture all the effects from the experiments. To achieve a sufficient resolution within an acceptable amount of computational time, the grid gets adapted in r-refinement manner [110] as explained in the following. First the simulation is performed on a uniform grid to identify the area of interest. Afterwards the CVs are concentrated in this area while keeping the

overall number of CVs constant. Inside the refined area the grid has a Cartesian structure. To investigate the influence of the grid, the cavity depth over time is computed on four different grid levels using a $2.9 \times 10^{-3}$ m diameter drop impacting with $U_\mathrm{I} = 1.55\,\mathrm{m\,s^{-1}}$ (We = 84, Fr = 94, circle in Figure 5.19). An example of the coarsest mesh can be seen in Figure 5.21 and the grid level details can be found in Table 5.8. The results of the mesh study in Figure 5.22 are scaled in length by the initial drop diameter $D$ and in time by $D/U_\mathrm{I}$.

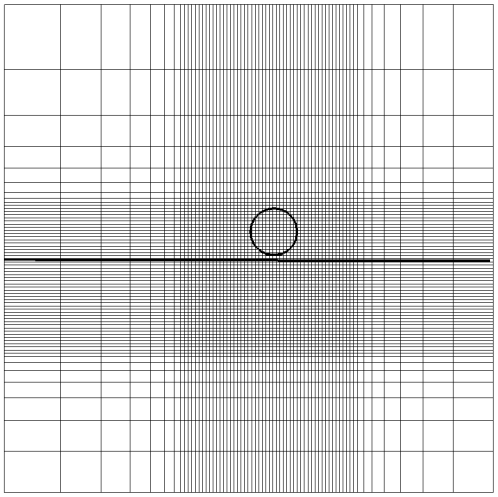


Figure 5.21: Example mesh in xy-plane with coarsest spacing and initial concentration $\alpha = 0.5$

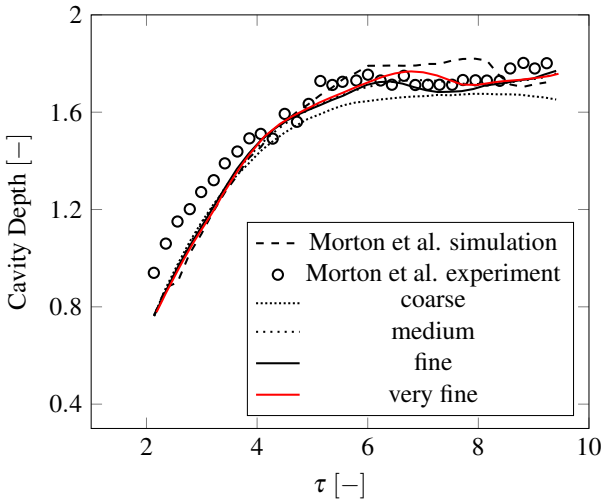



Figure 5.22: Comparison between results by Morton et al. [80] and simulations on four grid levels

Table 5.8: Grid level details

| Grid level | CV | Minimum cell size | CPU cores |
|---|---|---|---|
| Coarse | $64^3$ | $2.266 \times 10^{-4}$ m | 2 |
| Medium | $128^3$ | $1.133 \times 10^{-4}$ m | 8 |
| Fine | $256^3$ | $5.664 \times 10^{-5}$ m | 64 |
| Very fine | $416^3$ | $3.090 \times 10^{-5}$ m | 216 |

No bubble entrapment occurs on the coarse grid, in comparison to the higher grid levels. Since the detachment of a bubble is crucial for the acoustic part, the coarse grid will not be included for the coupled simulation. However, the path of the four grid levels are very similar until $\tau = 4$ where the coarse grid regime stays at a lower cavity depth. The other three finer resolutions differ slightly in the interval between $\tau = 6$ and $\tau = 8$. The regime of the fine and the very fine grid resolution deflect slightly to the surface. With the very

fine grid, the crater moves further to the depth before recoiling, visible as discrepancy between the fine grid around $\tau = 6.5$.
Additionally Figure 5.22 compares the results of the four simulations with the numerical and the corresponding experimental results of Morton et al. [80]. The simulation and experimental data by Morton et al. show different courses over time. The experimental data starts with an offset between $\tau = 2$ to $\tau = 4$, while from $\tau = 4$ to $\tau = 6$ the mean of the data points lies next to their simulation results. After $\tau = 6$, the simulation data of Morton stays constant until $\tau = 8$, from where on the cavity bottom rises again. The experimental data in contrast shows some sort of wave from $\tau = 6$ on with a rise of depth at the end. It can be seen that the results from the simulation on the fine and the very fine grid level also behave wave like from $\tau = 6$ on with a rise at the end. At the beginning, simulations on all grid levels run for the mesh study show the same offset as the simulation data of Morton et al. However, from $\tau = 5$ to $\tau = 8$ the fine and very fine grid level regimes agree within a 4 % range to the experimental data of Morton.

### 5.3.2 Temporal Resolution

The maximum temporal resolution is either given by the expected acoustic signal or the capillary time-step constraint of the multiphase flow. From theory the acoustics are produced by an oscillating bubble which introduces oscillations in the bottom of the cavity. The oscillation frequency was found to be in range of the natural oscillation frequency $f$ for a submerged gas bubble derived by Minnaert [79]:

$$f = \frac{1}{2\pi r}\sqrt{\frac{3\gamma_H P_0}{\rho}}. \tag{5.14}$$

With bubble radius $r$, specific heat ratio for the gas inside the bubble $\gamma_H$, static pressure on the exterior surface of the bubble $P_0$ and the density of the surrounding liquid $\rho$, the frequency is a function of the bubble radius. In order to capture variations of the bubble radius, a diameter range of $0.7 \times 10^{-3}$ m to $0.9 \times 10^{-3}$ m is considered and leads to a frequency range of approximately 9500 Hz to 7250 Hz, respectively. To ensure that the multiphase simulation is able to produce this oscillation, a maximum time step size is determined following the Nyquist-Shannon sampling theorem [115], which states that the sampling frequency should be twice as high as the signal frequency. In order to not only ensure a distinguish association of a frequency but rather a clear resolution, the sampling rate is being increased tenfold. These considerations concerning the acoustics lead to a maximum time step size of $\Delta t_{\max} = 1.053 \times 10^{-5}$ s.
Within multiphase flows with surface tension, the time step size needs to resolve the propagation of capillary waves for a stable simulation [14]. In Denner et al. [23] it is demonstrated that the constraint needs to be applied irrespective of an explicit or implicit

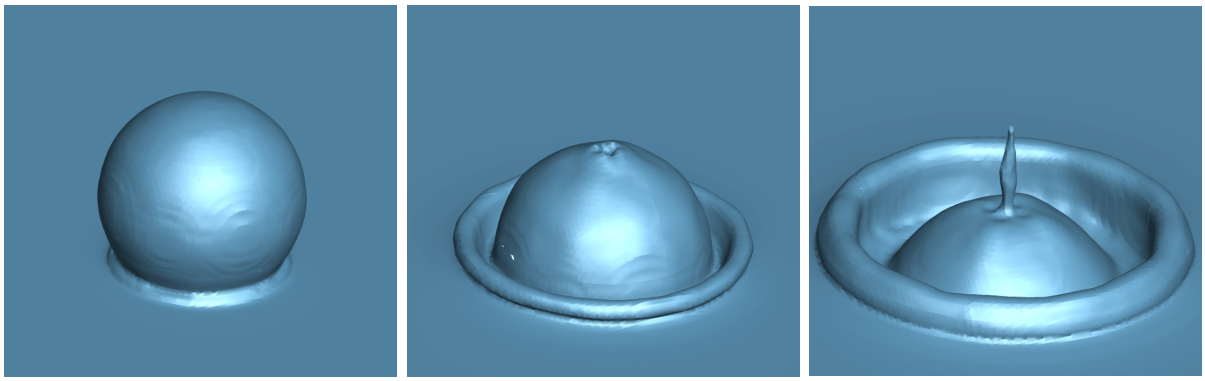
surface tension implementation. According to Brackbill et al. [14] the capillary time step constraint leads to a maximum time step size of

$$\Delta t_{\max} < \left[ \frac{(\rho_1 + \rho_2)(\Delta h)^3}{4\pi\sigma} \right]^{1/2} . \tag{5.15}$$

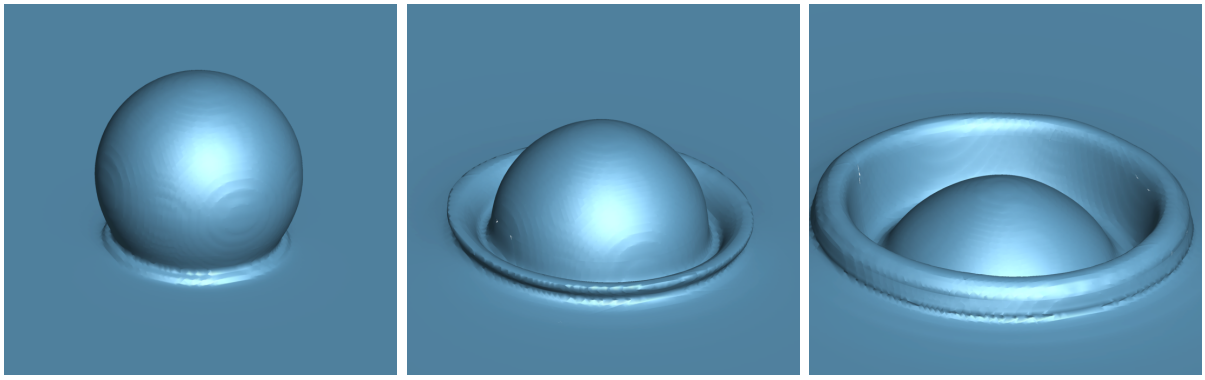
With the properties of air and water as listed in Table 5.7 and the minimum cell size of the very fine grid level in Table 5.8, a maximum time step size of $\Delta t_{\max} = 5.634 \times 10^{-5}\,\mathrm{s}$ results.
Considering both criteria and in order to be in distance to a CFL number smaller than 0.3, the maximum time step size for the coarse, medium and fine grid simulations are set to a maximum of $\Delta t_{\max} = 1 \times 10^{-6}\,\mathrm{s}$ and for the very fine grid $\Delta t_{\max} = 1 \times 10^{-7}\,\mathrm{s}$.

### 5.3.3 Importance of Surface Tension

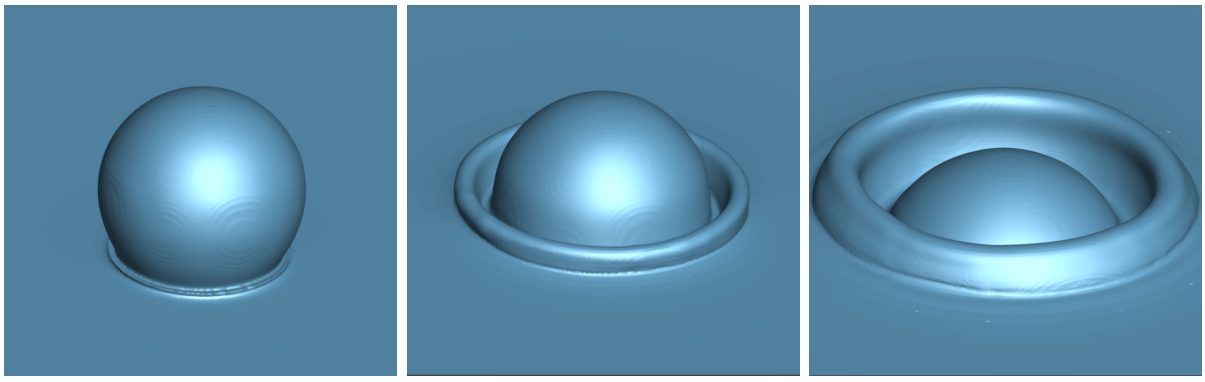
Before comparing the results, the importance of the curvature computation on the fine grid is exposed in Figure 5.23a. Due to an inadequate curvature computation, the drop loses its shape during the impact into the surface and some fluid emerges at the top of the drop. In general, the surface appears as irregular and wavy leading to an unstable surface representation. With the described improvements to the CFDM model from Section 4.2.3, the surface tension stays stable as well as the shape of the drop in Figure 5.23b. Even in the very fine grid configuration, the improved CFDM curvature computation works well and shows no signs of faulty behavior. The isosurface of the volume fraction in Figure 5.23c appeals smoother than in the fine grid simulation.



(a) CFDM on $256^3$ CVs



(b) Improved CFDM on $256^3$ CVs



(c) Improved CFDM on $416^3$ CVs

Figure 5.23: Difference between the standard CFDM and the improved CFDM on $256^3$ and $416^3$ of the initial stages of the drop impact test case, shown by volume fraction isosurface for $F = 0.5$

### 5.3.4 Results and Discussion

Phillips et al. [97] deliver snapshots of a high-speed video at key stages of a 4.0 mm drop impacting with $U_\mathrm{I} = 1.29\,\mathrm{m\,s^{-1}}$ (We = 90, Fr = 42, square in Figure 5.19). With the settings of Morton et al. [80] the coupled multiphase acoustics simulation is carried out on the fine [38] and the very grid resolution from Section 5.3.1. The material properties are taken from Table 5.7, except the speeds of sound. The unscaled speeds of sound are employed for this test case, which are $343\,\mathrm{m\,s^{-1}}$ and $1484\,\mathrm{m\,s^{-1}}$ for air and water, respectively.
Figure 5.24 and Figure 5.26 show snapshots of the experiment, which are compared with the numerical results on the very fine grid produced in this work. The Figures 5.25 and 5.27 represent slices in the $x, y$-plane at $z = 0.017\,35\,\mathrm{m}$ taken from the coupled three dimensional simulation.

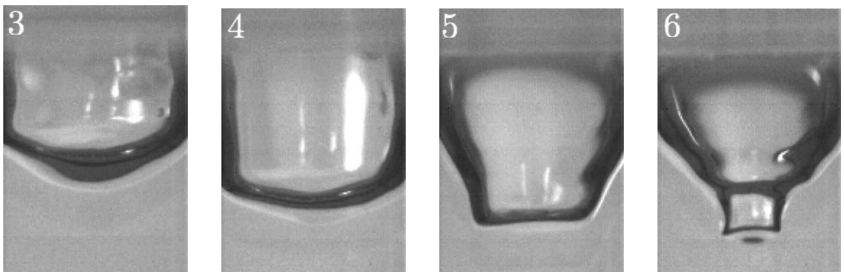



Figure 5.24: Experiment [97] at $10.42\times10^{-3}\,\mathrm{s}$, $15.82\times10^{-3}\,\mathrm{s}$, $20.89\times10^{-3}\,\mathrm{s}$ and $22.92\times10^{-3}\,\mathrm{s}$

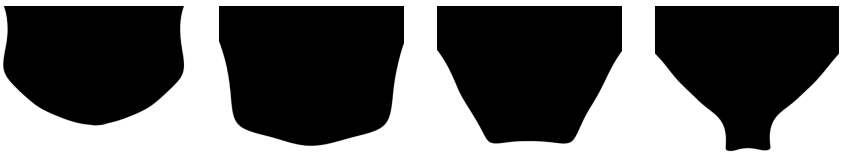


Figure 5.25: Simulation at $7.724 \times 10^{-3}\,\mathrm{s}$, $13.10 \times 10^{-3}\,\mathrm{s}$, $17.19 \times 10^{-3}\,\mathrm{s}$ and $19.08 \times 10^{-3}\,\mathrm{s}$

Due to the different drop diameter and impact velocity compared to the experiment, the snapshots of the simulation results are shifted in time. However, the difference between each frame for the simulation are in close range to the experiment. Every basic feature stated by Franz [36] during a drop impact can be seen in the simulation: cavity creation (Figure 5.25 at $7.724 \times 10^{-3}\,\mathrm{s}$ and $13.10 \times 10^{-3}\,\mathrm{s}$), begin of recoiling due to surface tension (Figure 5.25 at $1.719 \times 10^{-3}\,\mathrm{s}$ and $19.08 \times 10^{-3}\,\mathrm{s}$), entrapment of a small air bubble (Figure 5.27 at $19.87 \times 10^{-3}\,\mathrm{s}$, $20.15 \times 10^{-3}\,\mathrm{s}$ and $20.36 \times 10^{-3}\,\mathrm{s}$).
For a better understanding of the separation process in the simulation, more frames between $19.84\times10^{-3}\,\mathrm{s}$ and $20.31\times10^{-3}\,\mathrm{s}$ are shown in Figure 5.28. In the first row, the bubble

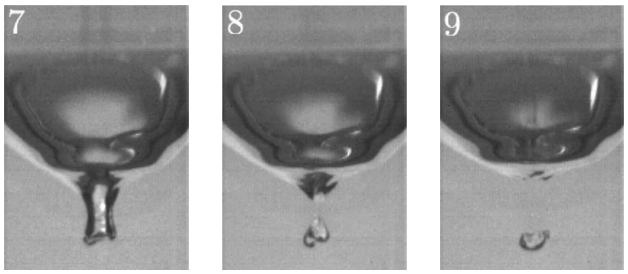



Figure 5.26: Experiment [97] at $24.34 \times 10^{-3}$ s, $24.55 \times 10^{-3}$ s and $24.94 \times 10^{-3}$ s

Figure 5.27: Simulation at $19.87 \times 10^{-3}$ s, $20.15 \times 10^{-3}$ s and $20.36 \times 10^{-3}$ s

detaches from the cavity bottom followed by a contraction of the bubble. The pinch-off creates a retracting of the air, which impacts into the bubble and initiates the oscillation as seen in the second and third row. The cavity bottom seems to move monotonous in upward direction after the bubble detachment. A direct influence of the bubble oscillation onto the cavity bottom is not observable in the frames of Figure 5.28.
Nevertheless, by analyzing the pressure and the acoustic source around the bubble detachment an acoustic initiating behavior is present. Before the crater collapses as in the first frame of Figure 5.29, in both phases the pressure rises after the tightest point in the neck. Going upwards from the tightest point to the top of the crater, the pressure sinks and is less than in the surrounding air. After the pinch-off and above the newly created surface, the pressure increases again indicated by the lighter nuance of the blue color. Below the surface, the high pressure mostly keeps its position and pushes the bubble and cavity bottom away from each other. By examining the acoustic source term in the same interval in Figure 5.30, deflections in negative and positive directions are observable. During the retraction phase negative acoustic sources emerge above the neck of the crater, as seen in the first frame. Positive acoustic sources starting to occur right before the pinch-off. These positive acoustic sources are the highest right after the pinch-off and seem to penetrate the previous present negative sources. The last frame of Figure 5.30 shows again some negative sources coming from the surface. Some CVs above the surface contain small volume fraction values indicating water in which acoustic source artifacts emerge. Due to the material properties, the signal produced inside these artifacts does not interrupt the airborne acoustics. From the perspective of the flow variables, there should be an acoustic signal coming from the drop impact. Therefore, the acoustic pressure in air and water

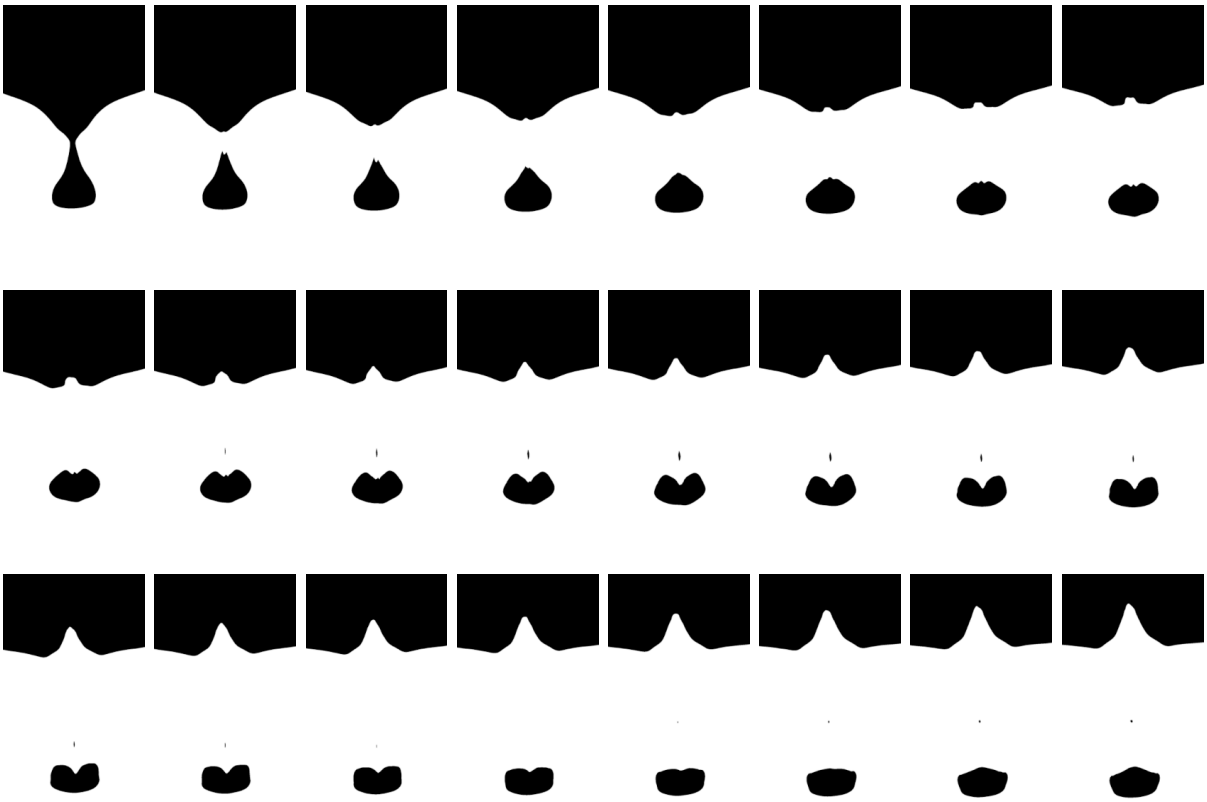


Figure 5.28: Bubble detachment from $20.11 \times 10^{-3}$ s to $21.32 \times 10^{-3}$ s in $0.04 \times 10^{-3}$ s steps

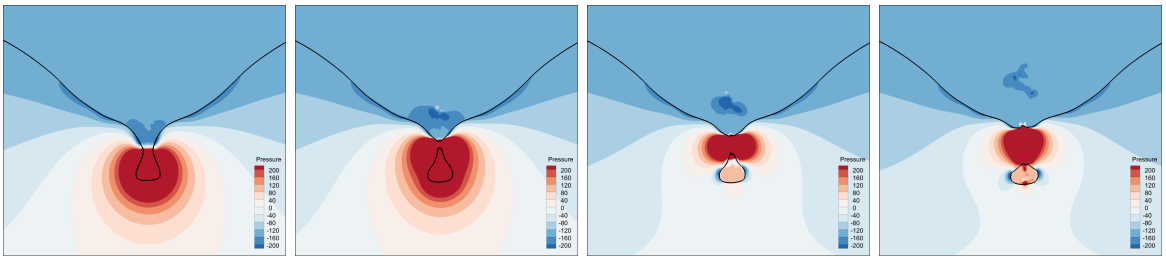


Figure 5.29: Pressure of bubble detachment at $t = 20.03 \times 10^{-3}$ s, $20.13 \times 10^{-3}$ s, $20.18 \times 10^{-3}$ s, $20.30 \times 10^{-3}$ s

at monitoring points (0.0173 m, 0.0208 m, 0.0173 m) and (0.0173 m, 0.0065 m, 0.0173 m) are evaluated, respectively. The results of the airborne acoustics for the fine and the very fine grid are presented in Figure 5.31.

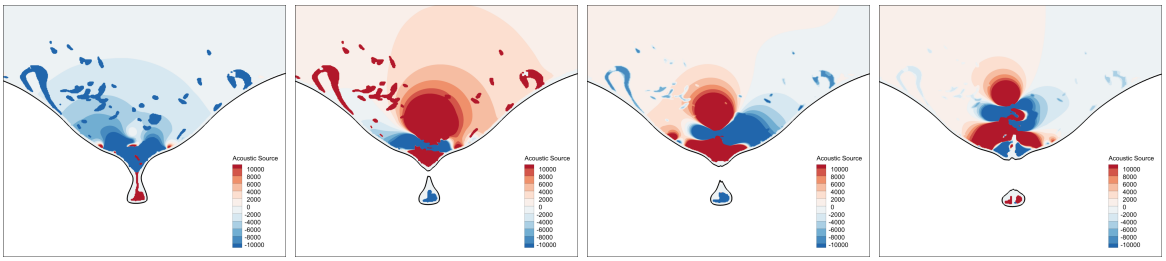
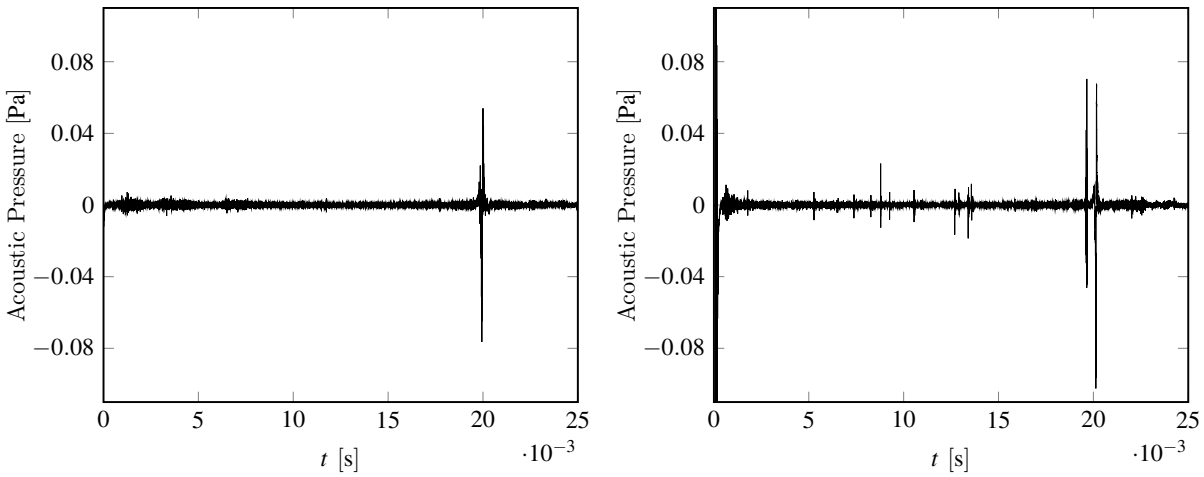


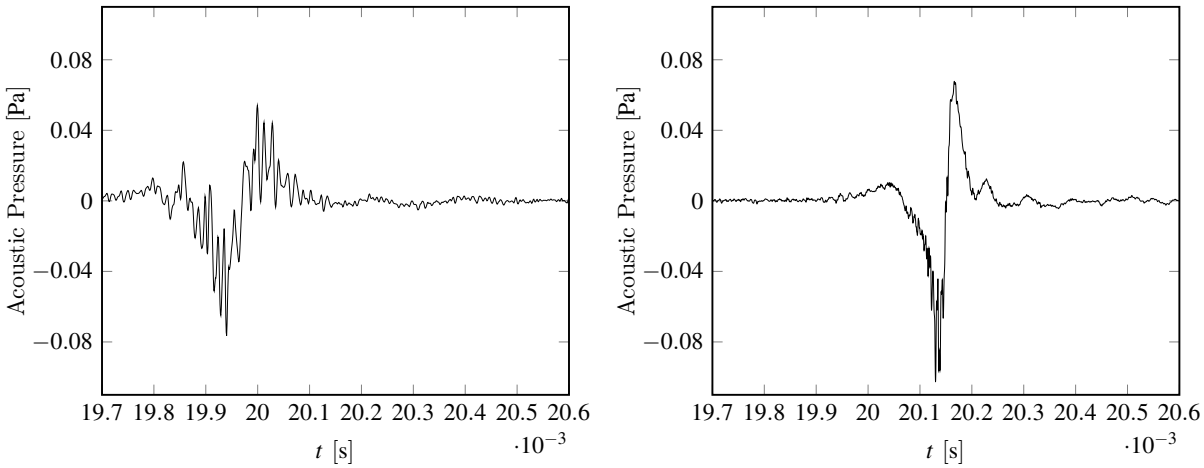
Figure 5.30: Acoustic source of bubble detachment at $t = 20.03 \times 10^{-3}$ s, $20.13 \times 10^{-3}$ s, $20.18 \times 10^{-3}$ s, $20.30 \times 10^{-3}$ s




(a) $256^3$ CVs (b) $416^3$ CVs

Figure 5.31: Airborne signal of drop impact simulation

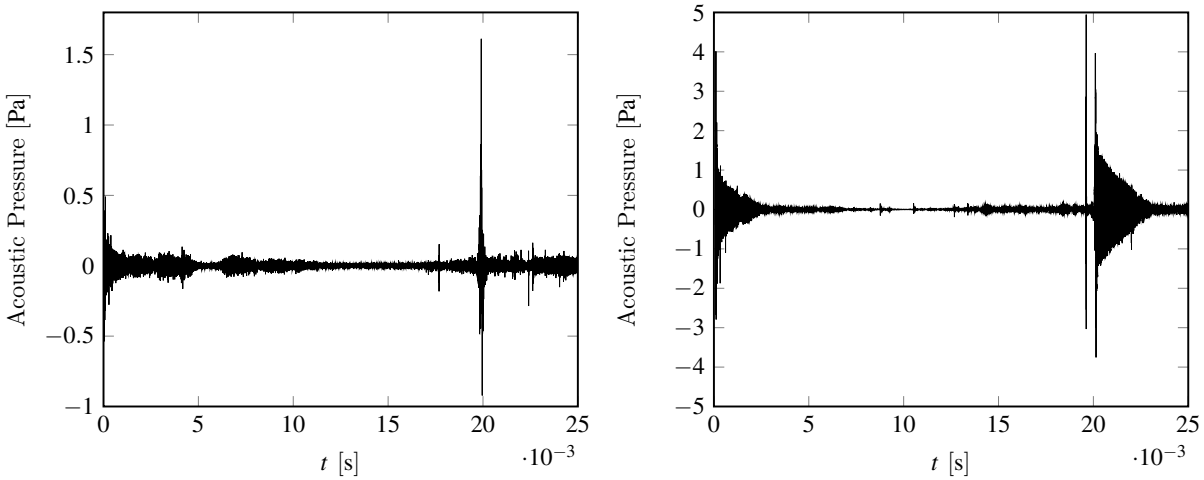
On both grid levels, the regime of the acoustic pressure is identical, as there are some small spikes before the bubble detachment and a single great amplitude around the bubble detachment. For a better correlation, the acoustic pressure around the pinch-off is shown in Figure 5.32. On the fine grid with $256^3$ CVs the pinch-off occurs $0.3 \times 10^{-3}$ s earlier than on the very fine grid and the signal appears more disturbed than on the very fine grid. However, comparing the images of the very fine grid in Figures 5.29 and 5.30, the sound pulse coincides with the acoustic source as expected. As seen in Figure 5.32b, the acoustic pressure starts with a negative deflection and after reaching its peak around $t = 20.13 \times 10^{-3}$ s, it is followed by an increase with a high at $t = 20.17 \times 10^{-3}$ s. Afterwards very small oscillation with positive peaks at $t = 20.228 \times 10^{-3}$ s and $t = 20.303 \times 10^{-3}$ s are present, yet they are not distinguishable from the signal noise before and after the pinch-off. However, the previous experimental findings that the acoustic signal emerges from the bubble entrainment and its pinch-off could be reproduced in the simulations carried out in this work.




(a) $256^3$ CVs (b) $416^3$ CVs

Figure 5.32: Close-up of airborne signal at drop detachment

Although the focus lies on the airborne signal, the acoustic pressure underwater for the fine and the very fine grid are shown in Figures 5.33. From the scales of the acoustic pressure

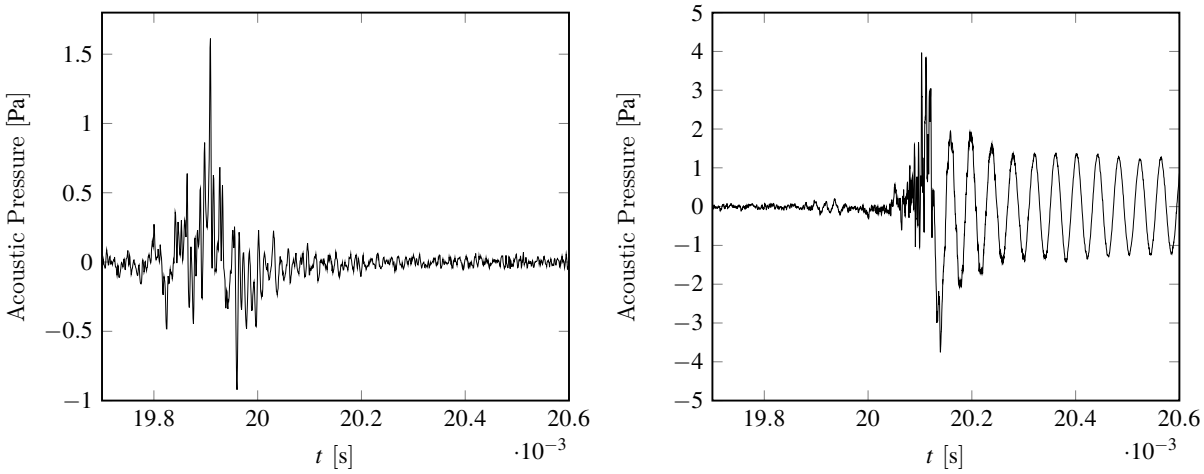



(a) $256^3$ CVs (b) $416^3$ CVs

Figure 5.33: Underwater signal of drop impact simulation

above and below the surface, it can be seen that the amplitude difference between air and water is in the expected range. At the beginning a deflection in the underwater signal occurs, which is caused by the initialization of the simulation and not by the impact of the

drop. As in the airborne signal, the underwater signal shows a big deflection at the bubble entrainment. A close-up of the acoustic pressure in water during the detachment process is shown in Figure 5.34.




(a) $256^3$ CVs (b) $416^3$ CVs

Figure 5.34: Close-up of underwater signal at drop detachment

Contrary to the acoustic pressure above the surface, the underwater signal is starting with a positive deflection in the acoustic pressure signal. The regime on the fine grid level is not as clear as the airborne results. Note that the underwater acoustics are more likely sensible to general disturbances due the high speed of sound and the comparable small domain. The signal of the very fine grid resolution in Figure 5.34b starts to oscillate after the pinch-off. Comparing the high frequency of the oscillation with the time series pictures in Figure 5.28 and the theoretical considerations in Section 5.3.2, it is more likely to originate from a different source than the bubble oscillation.

Concluding the results of the drop impact test case, the developed framework is able to reproduce the corresponding experiment. Furthermore, the simulation gives a more detailed insight in the physical mechanisms leading to the airborne acoustics and the characteristic drop impact sound. The theoretical hypothesis of the induced oscillation on the cavity bottom was not directly visible, although every other physical mechanism, including the acoustic signal, occurs in the simulation. It should be noted that in the experiments the effect of reverberation had influence on the damping of the signal oscillation [97]. In the simulations, the employed non-reflecting acoustic outlets do not have this effect which could be the reason for the missing oscillation after the pinch-off. However, by conducting a parameter study with varying drop diameters, impact velocities or surface tension coefficients different correlations might be revealed.

# 6. Summary and Outlook

The physical phenomena of acoustics emerging from multiphase flows is in the scope of researches since the beginning of the last century. Experiments have been the main source of theories concerning the sound producing mechanisms. With the development of reliable computational simulations, a deeper understanding of the physics is achieved. In return, the knowledge is used for reducing noise in technical systems which is one of the main problems being faced in current century.
Within this thesis, a numerical framework for simulating acoustics emerging from surface tension dominated two-phase flows at low Mach numbers was presented. Both physical disciplines were computed on the basis of the finite volume method (FVM) and implemented in the computational fluid dynamics solver FASTEST. In order to account for the multiphase flow, the modified compressive interface capturing scheme for arbitrary meshes (M-CICSAM) as part of the volume of fluid method (VOF) were employed. Surface tension was modeled with the continuum surface force (CSF), for which the determination of the interface curvature is responsible for its accuracy. A main part of this work consisted in improving the curvature computation. A smoothing kernel applied to the volume fraction field as well as to the curvature values and a volume fraction dependent weighting were implemented. Their impact individually and in all possible combinations were investigated. It was found that the curvature computation methods have been improved in all measured quantities with the appropriate combination. In addition new approaches, such as the adaptive height function (AHF) or the machine learning approach, have been implemented. All methods are validated on two surface-tension-driven test cases. The best improvement was shown by the curvature by finite difference method (CFDM), while the most promising results have been achieved by the machine learning approach using an artificial neural network (ANN).
The second main part of this thesis was to develop a proper coupling between the multiphase flow and the acoustics. Due to the disparity of the scales between the low Mach number flow and the acoustics, the latter is obtained by solving linearized Euler equations (LEE) after the flow field has been computed. A sub-cycle technique for the necessary small acoustic time steps and high resolution schemes with flux limiter for the comparable large acoustic waves were employed. Without dealing with surface tension dominated

multiphase flows, the coupling between the fluid flow and the acoustics is achieved by a source term consisting of the pressure derivative with respect to the time. In the presence of curved multiphase interfaces, a material dependent pressure jump across the interface occurs naturally. A numerical difficulty of dealing with moving multiphase interfaces, ergo a moving pressure jump, arises. Unphysical acoustic sources are produced, which have been suppressed with a curvature based source term correction introduced in this work. A verification and a validation test case showed the expected results.
Finally, the developed numerical framework was used to investigate the acoustics of a water drop impacting into a water pool. With the improved CFDM method, full three dimensional coupled simulations up to seventy million control volumes (CVs) were compared to experimental results with satisfactory agreement. Furthermore, the results provided physical insights into the sound producing mechanisms, being the first numerical work of this kind. The simulation carried out in this thesis confirmed that the characteristic sound is only produced right before and right after a small air bubble detaches from the cavity bottom.

With the developed numerical methods, the ability of simulating a physical complex test case is given. However, some challenges and questions of the multiphase and the acoustic part are still remaining. Although the newly implemented improvements to the curvature methods showed good results, an all-purpose curvature computation algorithm is not available. Especially with distributed, multiblock domains, mainly the improved CFDM method is able to deliver stable values. Since the machine-learning approach for the curvature showed impressive results, the development should be carried on and extended to three dimension.
Recalling the final drop impact test case at low grid resolutions, physical aspects did not occur. For the bubble detachment as the initiator of the characteristic drop impact sound, a very fine grid resolution was necessary. Even though the simulation ran on several cores within a high performance cluster, the computational cost as well as the file sizes of the results were challenging. For reducing the number of CVs, an adaptive grid should be implemented. The mesh resolution around the multiphase interface could be increased while in flow areas without any steep gradients it could be decreased.
The drop impact test case itself should be investigated further. Different configurations concerning the drop impact velocity, drop radius or surface tension could give more insights on the sound producing mechanism and the acoustic signal. A cooperation with the authors of the experimental data would be advantageous.

# References


[1] A. Aalilija, Ch.-A. Gandin, and E. Hachem. On the analytical and numerical simulation of an oscillating drop in zero-gravity. *Computers & Fluids*, 197:104362, 2020.

[2] T. Abadie, J. Aubin, and D. Legendre. On the combined effects of surface tension force calculation and interface advection on spurious currents within volume of fluid and level set frameworks. *Journal of Computational Physics*, 297:611 – 636, 2015.

[3] M. Abom. *An Introduction to Flow Acoustics*. KTH, Stockholm, Sweden, 2010.

[4] S. Adami, X. Y. Hu, and N. A. Adams. A new surface-tension formulation for multi-phase sph using a reproducing divergence approximation. *Journal of Computational Physics*, 229:5011 – 5021, 2010.

[5] S. Afkhami and M. Bussmann. Height functions for applying contact angles to 3d vof simulations. *International Journal for Numerical Methods in Fluids*, 61:827 – 847, 2008.

[6] C. K. Aidun and J. R. Clausen. Lattice-boltzmann method for complex flows. *Annual Review of Fluid Mechanics*, 42:439–472, 2010.

[7] E. Aulisa, S. Manservisi, R. Scardovelli, and S. Zaleski. Interface reconstruction with least-squares fit and split advection in three-dimensional cartesian geometry. *Journal of Computational Physics*, 225:2301 – 2319, 2007.

[8] C. Bailly and D. Juvé. Numerical solution of acoustic propagation problems using linearized euler equations. *AIAA Journal*, 38(1):22–29, 2000.

[9] G. K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge Mathematical Library. Cambridge University Press, 2000.

[10] E. Becker, W. J. Hiller, and T. A. Kowalewski. Experimental and theoretical investigation of large-amplitude oscillations of liquid droplets. *Journal of Fluid Mechanics*, 231:189–210, 1991.

[11] M. V. Berry. The molecular mechanism of surface tension. *Physics Education*, 6(2):79–84, 1971.

[12] C. Bogey, C. Bailly, and D. Juvé. Computation of flow noise using source terms in linearized euler's equations. *AIAA Journal*, 40(2):235–243, 2002.

[13] G. Bornia, A. Cervone, S. Manservisi, R. Scardovelli, and S. Zaleski. On the properties and limitations of the height function method in two-dimensional cartesian geometry. *Journal of Computational Physics*, 230:851 – 862, 2011.

[14] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *Journal of Computational Physics*, 100:335 – 354, 1992.

[15] I. Chakraborty, G. Biswas, and P. S. Ghoshdastidar. A coupled level-set and volume-of-fluid method for the buoyant rise of gas bubbles in liquids. *International Journal of Heat and Mass Transfer*, 58:240 – 259, 2013.

[16] S. J. Cummins, M. M. Francois, and D. B. Kothe. Estimation curvature from volume fractions. *Computers and Structures*, 83:425 – 434, 2005.

[17] N. Curle. The influence of solid boundaries upon aerodynamic sound. *Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences*, 231:505–514, 09 1955.

[18] G. B. Deane and M. D. Stokes. Scale dependence of bubble creation mechanisms in breaking waves. *Nature*, 418(6900):839–844, 2002.

[19] F. Denner. *Balanced-force two-phase flow modelling on unstructured and adaptive meshes*. PhD thesis, Imperial College London, 2013.

[20] F. Denner, D. R. van der Heul, G. T. Oud, M. M. Villar, A. d. S. Neto, and B. G. M. van Wachem. Comparative study of mass-conserving interface capturing frameworks for two-phase flows with surface tension. *International Journal of Multiphase Flow*, 141:37 – 47, 2014.

[21] F. Denner and B. G. M. van Wachem. Compressive vof method with skewness correction to capture sharp interfaces on arbitrary meshes. *Journal of Computational Physics*, 279:127 – 144, 2014.

[22] F. Denner and B. G. M. van Wachem. Fully-coupled balanced-force vof framework for arbitrary meshes with least-squares curvature evaluation from volume fractions. *Numerical Heat Transfer Fundamentals, Part B*, 65:218 – 255, 2014.

[23] F. Denner and B. G. M. van Wachem. Numerical time-step restrictions as a result of capillary waves. *Journal of Computational Physics*, 285:24–40, 2015.

[24] F. Denner, C.-N. Xiao, and B. G. M. van Wachem. Pressure-based algorithm for compressible interfacial flows with acoustically-conservative interface discretisation. *Journal of Computational Physics*, 367:192–234, 2018.

[25] F. Durst and M. Schäfer. A parallel block-structured multigrid method for the prediction of incompressible flows. *International Journal for Numerical Methods in Fluids*, 22:549–565, 1996.

[26] F. Evrard, F. Denner, and B. van Wachem. Estimation of curvature from volume fractions using parabolic reconstruction on two-dimensional unstructured meshes. *J. Comput. Phys.*, 351(C):271–294, 2017.

[27] F. Evrard, F. Denner, and B. G. M. van Wachem. Height-function curvature estimation with arbitrary order on non-uniform cartesian grids. *Journal of Computational Physics: X*, 7:100060, 2020.

[28] F. Evrard, F. Denner, and B. G. M. van Wachem. Height-function curvature estimation with arbitrary order on non-uniform cartesian grids. *Journal of Computational Physics: X*, 7:100060, 2020.

[29] R. Ewert and W. Schröder. Acoustic perturbation equations based on flow decomposition via source filtering. *Journal of Computational Physics*, 188(2):365–398, 2003.

[30] W. Fasold and E. Veres. *Schallschutz und Raumakustik in der Praxis*. Huss-Medien, Berlin, Germany, 1998.

[31] J. H. Ferziger, M. Peric, and R. L. Street. *Numerische Strömungsmechanik*. Springer, Wiesbaden, Germany, 2 edition, 2019.

[32] D. Fink. A new definition of noise: noise is unwanted and/or harmful sound. Noise is the new 'secondhand smoke'. *Proceedings of Meetings on Acoustics*, 39(1):050002, 2020.

[33] M. M. Francois, S. J. Cummins, E. D. Dendy, D. Edward, D. B. Kothe, J. M. Sicilian, and M. W. Williams. A balanced-force algorithm for continuois and sharp interfacial surface tension models within a volume tracking framework. *Journal of Computational Physics*, 213:141 – 173, 2006.

[34] M. M. Francois and B. K. Schwartz. Interface curvature via volume fractions, heights, and mean values on nonuniform rectangular grids. *Journal of Computational Physics*, 229:527 – 540, 2010.

[35] H. M. Frank and C.-D. Munz. Direct aeroacoustic simulation of acoustic feedback phenomena on a side-view mirror. *Journal of Sound and Vibration*, 371:132–149, 2016.

[36] G. J. Franz. Splashes as Sources of Sound in Liquids. *The Journal of the Acoustical Society of America*, 31(8):1080–1096, 1959.

[37] J. Friedrich and M. Schäfer. Acoustics simulation in the presence of moving interfaces in multiphase flows. In *Proceedings of ECCM 6-ECFD 7*, pp. 2176–2186, 2018.

[38] J. Friedrich and M. Schäfer. Towards an acoustic simulation of a water drop impacting in a water pool. *Flow, Turbulence and Combustion*, 105(4):1231–1247, 2020.

[39] J. Fromm. A method for reducing dispersion in convective difference schemes. *Journal of Computational Physics*, 3:1765 – 189, 1968.

[40] P. H. Gaskell and A. K. C. Lau. Curvature-compensated convective transport: Smart, a new boundedness- preserving transport algorithm. *International Journal for Numerical Methods in Fluids*, 8(6):617–641, 1988.

[41] D. Gerlach, G. Tomar, G. Biswas, and F. Durst. Comparison of volume-of-fluid methods for surface tension-dominant two-phase flows. *International Journal of Heat and Mass Transfer*, 49:740 – 754, 2006.

[42] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[43] S. Hambric. Structural acoustics tutorial—part 1: Vibrations in structures. *Acoustics Today*, 2:21–33, 2006.

[44] S. Hambric and J. Fahnline. Structural acoustics tutorial—part 2: Sound—structure interaction. *Acoustics Today*, 3:9–27, 2007.

[45] J. C. Hardin and D. S. Pope. An acoustic/viscous splitting technique for computational aeroacoustics. *Theoretical and Computational Fluid Dynamics*, 6(5-6):323–340, 1994.

[46] A. Harten. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 49(3):357–393, 1983.

[47] A. Harten. On a class of high resolution total-variation-stable finite-difference schemes. *SIAM Journal on Numerical Analysis*, 21(1):1–23, 1984.

[48] M. Herrmann. A balanced force refined level set grid method for two-phase flows on unstructured flow solver grids. *Journal of Computational Physics*, 227:2674 – 2706, 2008.

[49] C. W. Hirt and B. D. Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39:201 – 225, 1981.

[50] A. Hüppe, J. Grabinger, M. Kaltenbacher, A. Reppenhagen, G. Dutzler, and W. Kühnel. A non-conforming finite element method for computational aeroacoustics in rotating systems. In *20th AIAA/CEAS Aeroacoustics Conference*. American Institute of Aeronautics and Astronautics, 2014.

[51] S. Jakirlić, G. Kadavelil, M. Kornhaas, M. Schäfer, D. C. Sternel, and C. Tropea. Numerical and physical aspects in les and hybrid les/rans of turbulent flow separation in a 3-d diffuser. *International Journal of Heat and Fluid Flow*, 31(5):820–832, 2010. Sixth International Symposium on Turbulence, Heat and Mass Transfer, Rome, Italy, 14-18 September 2009.

[52] H. Jasak, H. G. Weller, and A. D. Gosman. High resolution nvd differencing scheme for arbitrarily unstructured meshes. *International Journal for Numerical Methods in Fluids*, 31(2):431–449, 1999.

[53] M. Kaltenbacher, editor. *Computational Acoustics*. Springer International Publishing, 2018.

[54] E. Kolb. *Aeroacoustic Simulation of Turbulent Fluid-Structure Interactions at Low Mach Numbers*. PhD thesis, Technische Universität Darmstadt, 2023.

[55] E. Kolb and M. Schäfer. Aeroacoustic simulation of flexible structures in low mach number turbulent flows. *Computers & Fluids*, 227:105020, 2021.

[56] A. Kondratyuk. *Investigation of the Very Large Eddy Simulation model in the context of fluid-structure interaction*. PhD thesis, Technische Universität Darmstadt, 2017.

[57] M. Kornhaas. *Effiziente numerische Methoden für die Simulation aeroakustischer Probleme mit kleinen Machzahlen*. PhD thesis, Technische Universität Darmstadt, 2011.

[58] Z. Kraus, F. Köhler, J. Friedrich, and M. Schäfer. A deep learning approach for curvature prediction in algebraic volume of fluid methods. In *9th edition of the International Conference on Computational Methods for Coupled Problems in Science and Engineering (COUPLED PROBLEMS 2021)*, 2021.

[59] F. Kummer, J. Weber, and M. Smuda. Bosss: A package for multigrid extended discontinuous galerkin methods. *Computers & Mathematics with Applications*, 81:237–257, 2021. Development and Application of Open-source Software for Problems with Numerical PDEs.

[60] H. Lamb. *Hydrodynamics*. Dover Books on Physics. Dover publications, 1945.

[61] T. Lehnhäuser and M. Schäfer. Efficient discretization of pressure-correction equations on non-orthogonal grids. *International Journal for Numerical Methods in Fluids*, 42(2):211–231, 2003.

[62] T. Lehnhäuser and M. Schäfer. Improved linear interpolation practice for finite-volume schemes on complex grids. *International Journal for Numerical Methods in Fluids*, 38(7):625–645, 2002.

[63] T. G. Leighton. How can humans, in air, hear sound generated underwater (and can goldfish hear their owners talking)? *The Journal of the Acoustical Society of America*, 131(3):2539–2542, 2012.

[64] B. P. Leonard. Simple high-accuracy resolution program for convective modelling of discontinuities. *International Journal for Numerical Methods in Fluids*, 8(10):1291–1318, 1988.

[65] B. P. Leonard. The ultimate conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering*, 88(1):17–74, 1991.

[66] R. Lerch, G. Sessler, and D. Wolf. *Technische Akustik*. Springer Berlin Heidelberg, 2009.

[67] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002.

[68] F. S. Lien and M. A. Leschziner. Upstream monotonic interpolation for scalar transport with application to complex turbulent flows. *International Journal for Numerical Methods in Fluids*, 19(6):527–548, 1994.

[69] M. J. Lighthill. On sound generated aerodynamically i. general theory. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 211(1107):564–587, 1952.

[70] P. Liovic, M. M. Francois, M. Rudman, and R. Manasseh. Efficient simulation of surface tension-dominated flows through enhanced interface geometry interrogation. *Journal of Computational Physics*, 229:7520 – 7544, 2010.

[71] Xu-D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.

[72] J. López, C. Zanzi, P. Gómez, R. Zamora, F. Faura, and J. Hernández. An improved height function technique for computing interface curvature from volume fractions. *Computer Methods in Applied Mechanics and Engineering*, 198(33):2555–2564, 2009.

[73] A. Mallock. Sounds produced by drops falling on water. *Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences*, 95:138–143, 1918.

[74] V. Maltsev, D. Yuan, K. W. Jenkins, M.Skote, and P. Tsoutsanis. Hybrid discontinuous galerkin-finite volume techniques for compressible flows on unstructured meshes. *Journal of Computational Physics*, 473:111755, 2023.

[75] A. Marchand, J. H. Weijs, J. Weijs, J. H. Snoeijer, and B. Andreotti. Why is surface tension a force parallel to the interface? *American journal of physics*, 79(10):999–1008, 2011.

[76] J. M. Mariño Salguero. *Numerical simulation of free surface flows interacting with flexible structures*. PhD thesis, Technische Universität Darmstadt, 2021.

[77] O. Marsden, C. Bogey, and C. Bailly. Direct noise computation of the turbulent flow around a zero-incidence airfoil. *AIAA Journal*, 46(4):874–883, 2008.

[78] J. Mencinger and I. Žun. On the finite volume discretization of discontinuous body force field on collocated grid: Application to vof method. *Journal of Computational Physics*, 221(2):524–538, 2007.

[79] M. Minnaert. Xvi. on musical air-bubbles and the sounds of running water. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 16(104):235–248, 1933.

[80] D. Morton, M. Rudman, and L. Jong-Leng. An investigation of the flow regimes resulting from splashing drops. *Physics of Fluids*, 12(4):747–763, 2000.

[81] F. Moukalled, L. Mangani, and M. Darwish. *High Resolution Schemes*, pp. 429–488. Springer International Publishing, Cham, 2016.

[82] C.-D. Munz, M. Dumbser, and S. Roller. Linearized acoustic perturbation equations for low mach number flow with variable density and temperature. *Journal of Computational Physics*, 224(1):352–364, 2007. Special Issue Dedicated to Professor Piet Wesseling on the occasion of his retirement from Delft University of Technology.

[83] J. Munz. *Numerische Topologieoptimierung multiphysikalischer Probleme*. PhD thesis, Technische Universität Darmstadt, 2023.

[84] S. Muzaferija, M. Peric, P. Sames, and Schelin T. A two-fluid navier- stokes solver to simulate water entry. *Proc. Twenty-Second Symposium on Naval Hydrodynamics*, 1998.

[85] M. Möser. *Engineering Acoustics*. Springer Berlin Heidelberg, 2009.

[86] K. C. Ng, M. Z. Yusoff, and E. Y. K. Ng. Higher-order bounded differencing schemes for compressible and incompressible flows. *International Journal for Numerical Methods in Fluids*, 53(1):57–80, 2007.

[87] W. F. Noh and P. Woodward. Slic (simple line interface calculation). *Lecture Notes in Physics*, 59:330 – 340, 1976.

[88] H. N. Oguz and A. Prosperetti. Bubble entrainment by the impact of drops on liquid surfaces. *Journal of Fluid Mechanics*, 219:143–179, 1990.

[89] S. Osher and S. Chakravarthy. High resolution schemes and the entropy condition. *SIAM Journal on Numerical Analysis*, 21(5):955–984, 1984.

[90] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79:12 – 49, 1988.

[91] C. Palacio, C. Pascual, F. Suarez, and I. Lloret. Smoothing of digital spectroscopic data by using a chebyshev filter. *Vacuum*, 64:481 – 485, 2002.

[92] I. R. Park, K. S. Kim, J. Kim, and S. H. Van. A volume-of-fluid method for incompressible free surface flows. *International Journal for Numerical Methods in Fluids*, 61:1331 – 1362, 2009.

[93] S.V Patankar and D.B Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer*, 15(10):1787–1806, 1972.

[94] H. V. Patel, J. A. M. Kuipers, and E. A. J. F. Peters. Computing interface curvature from volume fractions: a hybrid approach. *Computers & Fluids*, 161:74 – 88, 2018.

[95] H. V. Patel, A. Panda, J. A. M. Kuipers, and E. A. J. F. Peters. Computing interface curvature from volume fractions: A machine learning approach. *Computers & Fluids*, 193:104263, 2019.

[96] J. K. Patel and G. Natarajan. A generic framework for design of interface capturing schemes for multi-fluid flows. *Computers & Fluids*, 106:108 – 118, 2015.

[97] S. Phillips, A. Agarwal, and P. Jordan. The sound produced by a dripping tap is driven by resonant oscillations of an entrapped air bubble. *Scientific Reports*, 8, 2018.

[98] A. D. Pierce. *Acoustics*. Springer International Publishing, 2019.

[99] P. Pironkov. *Numerical Simulation of Thermal Fluid-Structure Interaction*. PhD thesis, Technische Universität Darmstadt, 2010.

[100] J.Y Poo and N Ashgriz. A computational method for determining curvatures. *Journal of Computational Physics*, 84(2):483–491, 1989.

[101] S. Popinet. An accurate adaptive solver for surface-tension-driven interfacial flows. *Journal of Computational Physics*, 228(16):5838–5866, 2009.

[102] S. Popinet. Numerical models of surface tension. *Journal of Computational Physics*, 229:7520 – 7544, 2010.

[103] J. Pozorski and M. Olejnik. Smoothed particle hydrodynamics modelling of multiphase flows: an overview. *Acta Mechanica*, 235(4):1685–1714, 2023.

[104] H. C. Pumphrey, L. A. Crum, and L. Bjo/rno/. Underwater sound produced by individual drop impacts and rainfall. *The Journal of the Acoustical Society of America*, 85(4):1518–1526, 1989.

[105] Y. Qi, J. Lu, R. Scardovelli, S. Zaleski, and G. Tryggvason. Computing curvature for volume of fluid methods using machine learning. *Journal of Computational Physics*, 377:155 – 161, 2018.

[106] Y. Renardy and M. Renardy. Prost: A parabolic reconstruction of surface tension for the volume-of-fluid method. *Journal of Computational Physics*, 183:400 – 421, 2002.

[107] C. M. Rhie and W. L. Chow. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal*, 21(11):1525–1532, 1983.

[108] W. J. Rider and D. B. Kothe. Reconstructing volume tracking. *Journal of Computational Physics*, 141:112 – 152, 1998.

[109] F. Schäfer, S. Müller, T. Uffinger, S. Becker, J. Grabinger, and M. Kaltenbacher. Fluid-structure-acoustic interaction of the flow past a thin flexible structure. *AIAA Journal*, 48(4):738–748, 2010.

[110] M. Schäfer. *Computational Engineering - Introduction to Numerical Methods*. Springer International Publishing, 2022.

[111] S. Schoder and M. Kaltenbacher. Hybrid aeroacoustic computations: State of art and new achievements. *Journal of Theoretical and Computational Acoustics*, 27(04):1950020, 2019.

[112] T. Seifarth. *Numerische Algorithmen für gitterfreie Methoden zur Lösung von Transportproblemen*. PhD thesis, Universität Kassel, 2018.

[113] J. H. Seo and Y. J. Moon. Perturbed compressible equations for aeroacoustic noise prediction at low mach numbers. *AIAA Journal*, 43(8):1716–1724, 2005.

[114] J. H. Seo and Y. J. Moon. Linearized perturbed compressible equations for low mach number aeroacoustics. *Journal of Computational Physics*, 218(2):702–719, 2006.

[115] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[116] W. Z. Shen and J. N. Sørensen. Aeroacoustic modelling of low-speed flows. *Theoretical and Computational Fluid Dynamics*, 13(4):271–289, 1999.

[117] W. Z. Shen and J. N. Sørensen. Comment on the aeroacoustic formulation of hardin and pope. *AIAA Journal*, 37(1):141–143, 1999.

[118] S. A. Slimon, M. C. Soteriou, and D. W. Davis. Computational aeroacoustics simulations using the expansion about incompressible flow approach. *AIAA Journal*, 37(4):409–416, 1999.

[119] G. Son and N. Hur. A coupled level-set and volume-of-fluid method for the buoyancy-driven motion of fluid particles. *Numerical Heat Transfer, Part B*, 42:523 – 542, 2002.

[120] J. H. Spurk and N. Aksel. *Fluid Mechanics*. Springer International Publishing, 2020.

[121] D. Staab. *Numerical treatment of multiphase flows coupled with acoustics for surface tension dominated flows*. PhD thesis, Technische Universität Darmstadt, 2016.

[122] D. Staab, S. Nowak, D. C. Sternel, and M. Schäfer. Numerical simulation of acoustics in heterogeneous media. *COUPLED PROBLEMS 2015 - Proceedings of the 6th International Conference on Coupled Problems in Science and Engineering*, pp. 791–799, 2015.

[123] H. L. Stone. Iterative solution of implicit approximations of multidimensional partial differential equations. *SIAM Journal on Numerical Analysis*, 5(3):530–558, 1968.

[124] D. Surek and S. Stempin. *Angewandte Strömungsmechanik*. Vieweg+Teubner, 2007.

[125] M. Sussman. A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles. *Journal of Computational Physics*, 187:110 – 136, 2003.

[126] M. Sussman and E. Fatemi. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM Journal on Scientific Computing*, 20(4):1165–1191, 1999.

[127] M. Sussman, E. Fatemi, P. Smereka, and S. Osher. An improved level set method for incompressible two-phase flows. *Computers & Fluids*, 27(5):663–680, 1998.

[128] M. Sussman and E. G. Puckett. A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows. *Journal of Computational Physics*, 162:301 – 337, 2000.

[129] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114:146 – 159, 1994.

[130] P. K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 21(5):995–1011, 1984.

[131] S. Tajiri, M. Tsutahara, and H. Tanaka. Direct simulation of sound and underwater sound generated by a water drop hitting a water surface using the finite difference lattice boltzmann method. *Computers & Mathematics with Applications*, 59(7):2411–2420, 2010. Mesoscopic Methods in Engineering and Science.

[132] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer Berlin Heidelberg, 2009.

[133] M. D. Torrey, L. D. Cloutman, R. C. Mjolsness, and C. W. Hirt. Nasa-vof2d: a computer program for incompressible flows with free surfaces, 1985.

[134] G. Tryggvason, R. Scardovelli, and S. Zaleski. *Direct Numerical Simulations of Gas–Liquid Multiphase Flows*. Cambridge University Press, 2011.

[135] O. Ubbink and R. I. Issa. A method for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics*, 153:26 – 50, 1999.

[136] E. van Dornshuld. *Chemistry, Student Handbook*. Mississippi State University, 2024. `https://dornshuld.chemistry.msstate.edu/books/chemistry/surface-tension.html`.

[137] S. Vincent and J.-P. Caltagirone. Test-case no 10: Parasitic currents induced by surface tension (pc). *Multiphase Science and Technology*, 16(1-3), 2004.

[138] P. S. Volpiani, J.-B. Chapelier, A. Schwöppe, J. Jägersküpper, and S. Champagneux. Aircraft simulations using the new cfd software from onera, dlr, and airbus. *Journal of Aircraft*, pp. 1–13, 2024.

[139] T. Waclawczyk and T. Koronowicz. Comparison of cicsam and hric high.resolution schemes for interface capturing. *Journal of Theoretical and Applied Mechanics*, 64:325 – 345, 2008.

[140] T. Waclawczyk and T. Koronowicz. Remarks on prediction of wave drag using vof method with interface capturing approach. *Arch Civ Mech Eng*, 8, 2008.

[141] Tomasz Waclawczyk. *Numerical modelling of free surface flows in ship hydrodynamics*. PhD thesis, Polish Academy of Sciences, Gdansk, 2007.

[142] J. E. Ffowcs Williams and D. L. Hawkings. Sound generation by turbulence and surfaces in arbitrary motion. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 264(1151):321–342, 1969.

[143] M. W. Williams, D. B. Kothe, and E. G. Puckett. Accuracy and convergence of continuum surface tension models. *Fluid Dynamics at Interfaces*, pp. 294 – 305, 1998.

[144] C. S. Wu, D. L. Young, and H. C. Wu. Simulations of multidimensional interfacial flows by an improved volume-of-fluid method. *International Journal of Heat and Mass Transfer*, 60:739 – 755, 2013.

[145] Z. Xie, D. Pavlidis, P. Salinas, J. R. Percival, C. C. Pain, and O. K. Matar. A balanced-force control volume finite element method for interfacial flows with surface tension using adaptive anisotropic unstructured meshes. *Computers & Fluids*, 138:38 – 50, 2016.

[146] L. Xue. *Entwicklung eines effizienten parallelen Lösungsalgorithmus zur dreidimensionalen Simulation komplexer turbulenter Strömungen*. PhD thesis, Technische Universität Berlin, 1998.

[147] K. Yokoi. A density-scaled continuum surface force model within a balanced force formulation. *Journal of Computational Physics*, 278:221 – 228, 2014.

[148] D. L. Youngs. Time-dependent multi-material flow with large fluid distortion. *Numerical Methods for Fluid Dynamics*, pp. 273 – 285, 1982.

[149] L. Štrubelj, I. Tiselj, and B. Mavko. Simulations of free surface flows with implementation of surface tension and interface sharpening in the two-fluid model. *International Journal of Heat and Fluid Flow*, 30:741–750, 2009.

# List of Figures

# List of Tables

# Abbreviations

**AHF** Adaptive Height-Functions 52, 65, 67, 71–73, 105

**ANN** Artificial Neural Network 58, 61, 66–68, 70–73, 106, 108

**CAA** Computational Aeroacoustics 21

**CAD** Computer Aided Design 22

**CBC** Convective-Boundedness-Criterion 28, 29, 31

**CDS** Central Differencing Scheme 26, 27, 32, 47, 49

**CFD** Computational Fluid Dynamics 1–3, 22, 24, 38

**CFDM** Curvature by Finite-Difference-Method 49, 53, 61, 63, 64, 67–69, 71, 73, 76, 83, 84, 92, 105, 106

**CFL** Courant-Friedrichs-Lewy 26, 38, 83

**CICSAM** Compressive Interface Capturing Scheme for Arbitrary Meshes 3

**CSF** Continuum Surface Force 4, 15, 59

**CTS** Coordinate Transformation Scheme 31

**CV** Control Volumes 22–25, 28, 30–34, 36–38, 43, 44, 47–53, 55–60, 62–64, 67, 68, 70–72, 74, 76, 77, 84, 86, 88–90, 92, 105, 106

**CVOFLS** Coupled-VOF-Level-Set 54, 55, 61, 67, 68, 70, 71, 73, 106

**DABT** Derivative Approximation Based on Multi Dimensional Taylor Series Expansion 31

**DG** Discontinous Galerkin 3

**DNC** Direct Noise Computation 6, 17

**EIF** Expansion about Incompressible Flow 20

**FDM** Finite Difference Method 24, 25

**FEM** Finite Element Method 3

**FSA** Fluid-Structure-Acoustic Interaction 22

**FSI** Fluid-Structure Interaction 22

**FVM** Finite Volume Method 3, 22, 24, 25, 35, 38, 40, 42, 43

**HF** Height-Function 49–53, 61, 63–68, 71, 73, 105, 106

**HR** High Resolution 31, 35, 43

**HRIC** High Resolution Interface Capturing scheme 3

**IC** Initial Condition 39, 40

**LBM** Lattice Boltzmann Method 3

**LEE** Linearized Euler Equations 38, 41–43, 59

**LES** Large Eddy Simulation 6

**LS** Level-Set 54, 55

**M-CICSAM** Modified Compressive Interface Capturing Scheme for Arbitrary Meshes 29

**MSE** Mean Square Error 58

**MuLI** Multidimensionale Lineare Interpolation 28

**NSE** Navier-Stokes equations 2, 3, 6, 17, 33–38, 108

**NVD** Normalized Variable Diagram 28–31, 105

**ODEs** Ordinary Differential Equations 39

**PDEs** Partial Differntial Equations 39, 40

**ReLU** Rectified Linear Unit 58

**SHC** Steady Hyper-C 29, 30

**SIMPLE** Semi-Implicit Method for Pressure Linked Equations 36, 37, 67

**SIP** Strongly Implicit Procedure 36

**SLIC** Simple Line Interface Calculation 2

**SPH** Smoothed Particel Hydrodynamics 3

**TFSI** Thermal-Fluid-Structure Interaction 22

**TV** Total Variation 31

**TVD** Total Variation Diminishing 28–31, 105

**UDS** Upwind Differencing Scheme 27, 28

**VOF** Volume of Fluid 1–4, 45, 54, 57, 59

**WENO** Weighted Essentially Non-Oscillatory 55