

**Numerische Simulation der Interaktion
von inkompressiblen Zweiphasenströmungen
mit Starrkörpern in drei Raumdimensionen**

DISSERTATION

zur

Erlangung des Doktorgrades (Dr. rer. nat.)

der

Mathematisch-Naturwissenschaftlichen Fakultät

der

Rheinischen Friedrich-Wilhelms-Universität Bonn

VORGELEGT VON

ROBERTO CROCE

AUS

ESSEN

BONN 2010

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der
Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Referent: Prof. Dr. Michael Griebel

2. Referent: Prof. Dr. Marc Alexander Schweitzer

Tag der Promotion: 25. Juni 2010

Erscheinungsjahr: 2010

Zusammenfassung

In dieser Arbeit stellen wir erstmalig einen parallelen Löser für inkompressible Zweiphasenströmungen mit Oberflächenspannung in drei Raumdimensionen vor, der Masseerhaltung und hohe Konvergenzordnung vereint. Als zweite wesentliche Neuerung kombinieren wir diesen Strömungslöser über die Level-Set-Technik mit der Dynamik komplexer Starrkörpergeometrien.

Im ersten Teil der Arbeit wird der dreidimensionale Navier-Stokes-Löser für inkompressible Zweiphasenströmungen mit Oberflächenspannung vorgestellt und auf die Simulation von Blasen- und Tröpfchendeformationen angewendet. Wir untersuchen hierbei den Einfluss der Oberflächenspannung und ihrer Diskretisierung, sowie der Fixpunktiteration zur Masseerhaltung auf das globale Konvergenzverhalten des Verfahrens. Unser Schema verwendet eine Finite-Differenzen-Diskretisierung zusammen mit einer Projektionsmethode und einem WENO-Verfahren fünfter Ordnung für die konvektiven Terme. Die freie Oberfläche zwischen den beiden Fluidphasen wird mit einer Level-Set-Technik approximiert. Dabei sind die Oberflächenspannungskräfte implizit über die *CSF*-Methode in den Navier-Stokes-Impulsgleichungen enthalten und werden über ein geglättetes Delta-Funktional mittels einer Interpolation dritter Ordnung ausgewertet. Unsere Approximation liefert eine Konvergenz zweiter Ordnung außerhalb einer Umgebung der freien Oberfläche und eine Konvergenz erster Ordnung in Nähe der freien Oberfläche, wo das geglättete Delta-Funktional zum Einsatz kommt. Um die bekannten Masseerhaltungsschwierigkeiten der Level-Set-Approximation zu bewältigen, setzen wir eine verbesserte Reinitialisierungsmethode ein und kombinieren diese mit einer globalen Fixpunktiteration, sodass die Masse auch für Langzeitrechnungen erhalten bleibt. Wir diskutieren detailliert unser numerisches Verfahren und präsentieren einige Ergebnisse hinsichtlich Masseerhaltung, Konvergenz der Krümmung und die Anwendung unseres Löser auf die Simulation von zwei Problemen mit aufsteigenden Bläschen, eines mit kleinem und eines mit großem Sprung in den Materialparametern, und auf die Simulation von Tropfendeformationen in einer Scherströmung in drei Raumdimensionen. Weiterhin vergleichen wir unsere dreidimensionalen Resultate mit quasi-zweidimensionalen und zweidimensionalen Simulationsergebnissen. Der Vergleich zeigt klar die Notwendigkeit einer dreidimensionalen Simulation zur korrekten Wiedergabe des physikalischen Verhaltens.

Im zweiten Teil dieser Arbeit koppeln wir bewegliche, komplexe, dreidimensionale Starrkörpergeometrien mit dem Zweiphasen-Strömungslöser. Wir beschreiben die Transformation einer vorgegebenen komplexen triangulierten Starrkörpergeometrie in eine diskrete Level-Set-Funktion. Die Bewegungsgesetze des Starrkörpers können hierbei über eine semi-Lagrangesche Diskretisierung der Transportgleichung so umgesetzt werden, dass die charakteristische Deformationsfreiheit des Starrkörpers erhalten bleibt. Die vollständige Einbindung der Starrkörperdynamik in den inkompressiblen Zweiphasen-Strömungslöser wird über einen zusätzlichen Lagrange-Multiplikator-Ansatz umgesetzt. Ähnlich dem Druckfeld aus der Projektionsmethode, welches die Inkompressibilität der Fluide forciert, wird ein zusätzlicher Druck innerhalb des Starrkörpers ermittelt, der die Deformationsfreiheit und die physikalische Starrkörperbewegung erzwingt und diese mit dem Geschwindigkeits- und Druckfeld der Zweiphasenströmung koppelt. In dieser Arbeit wurde der Lagrange-Multiplikator-

Ansatz mit der Level-Set-Technik gekoppelt und erstmalig auf reale dreidimensionale Zweiphasen-Starrkörper-Interaktionen mittels einer semi-impliziten Lösungsmethode erfolgreich umgesetzt. Das gesamte Verfahren ist darüber hinaus parallelisiert und um ein Large-Eddy-Turbulenzmodell erweitert worden, sodass dreidimensionale makrodynamische Strömungsprozesse realitätsnah simuliert werden können. Die Ergebnisse einiger numerischer Experimente, bei denen einerseits der hydrodynamisch stationäre Zustand einer Kugel innerhalb einer Phase und auf der freien Oberfläche untersucht und andererseits die numerische Konvergenzrate am Beispiel eines auftauchenden Zylinders ermittelt wird, belegen deutlich das von uns vorhergesagte globale Konvergenzverhalten erster Ordnung.

Abschließend simulieren wir das Strömungsverhalten einer zwangsgeführten Schiffskörpergeometrie und das Auftauchen eines U-Bootes. Diese herausfordernden numerischen Experimente belegen die vielseitige Einsetzbarkeit des für diese Arbeit entwickelten Simulationsprogramms.

Danksagung

An dieser Stelle möchte ich allen danken, die mich bei dieser Arbeit unterstützt haben. Meinem Betreuer Prof. Dr. Michael Griebel möchte ich für die optimalen Rahmenbedingungen und wertvollen Diskussionen danken. Ebenso möchte ich Priv. Doz. Dr. Marc Alexander Schweitzer danken für die Übernahme des Zweitgutachtens und für die Zeit, die er stets für Diskussionen mit mir übrig hatte. Insbesondere danke ich Dr. Martin Engel, ohne den der Arbeitsalltag nur halb so viel Freude gemacht hätte und mit dem ich jede Idee ausdiskutieren konnte. Dr. Martin Engel, Dr. Sven Gross und Dipl. Math. Patrick Utler danke ich für das akribische Korrekturlesen dieser Arbeit und Dipl.-Inf. Peter Zaspel für die Unterstützung bei der Erstellung der gerenderten Bilder. Desweiteren danke ich Priv. Doz. Dr. Marc Alexander Schweitzer, Dr. Martin Engel und Dipl.-Math. Bram Metsch für die erstklassige Rechnerinfrastruktur, die ich exzessiv nutzen konnte.

Teile dieser Arbeit sind während Forschungsaufenthalten am Institut für Computerwissenschaften der Florida-State-University in den USA, am Institut für Computerwissenschaften der Katholieke Universiteit Leuven in Belgien und am Laboratoire d'Hydraulique Environnementale der Ecole Polytechnique Fédérale de Lausanne in der Schweiz entstanden. Für die Gastfreundlichkeit und Großzügigkeit, die ich seitens Prof. Dr. Max Gunzburger, Prof. Dr. Dirk Roose und Prof. Dr. Christophe Ancey erfahren durfte, bin ich sehr dankbar.

Diese Arbeit wäre nicht möglich gewesen ohne die Unterstützung der DFG, die über das Projekt C3 des SFB-611 "Singuläre Phänomene und Skalierung in mathematischen Modellen" meine Tätigkeit als wissenschaftlicher Mitarbeiter finanzierte.

Auf persönlicher Ebene danke ich meiner Frau und meinen Eltern, die das Entstehen dieser Arbeit stets moralisch unterstützten. Danke Dir Emine für Deine Geduld und langjährige Bereitschaft, gemeinsame Zeit für diese Arbeit zu opfern. Es hat alles länger gedauert als es sollte. Özur dilerim.

Inhaltsverzeichnis

1	Einleitung	1
I	Simulation von inkompressiblen Zweiphasenströmungen	11
2	Das Modell für Zweiphasenströmungen mit Oberflächenspannung	12
2.1	Die Navier-Stokes-Gleichungen	12
2.1.1	Die Erhaltung der Masse	13
2.1.2	Die Erhaltung des Impulses	14
2.2	Das Modell für inkompressible Zweiphasenströmungen	15
2.2.1	Physikalisches Modell	15
2.2.2	Einbindung der Oberflächenspannungskraft in die Impulsgleichungen	18
2.2.3	Oberflächenspannung als Volumenintegral via der CSF-Methode . .	19
2.2.4	Dimensionslose Formulierung	21
2.2.5	Randbedingungen an Gebietsrändern	21
3	Diskretisierungstechniken für freie Oberflächen	23
3.1	Tracking-Methoden	24
3.1.1	Moving-Mesh-Methoden	24
3.1.2	Front-Tracking-Verfahren	25
3.1.3	Randintegral-Verfahren	27
3.1.4	Partikel-Methoden	27
3.2	Capturing-Methoden	29
3.2.1	Continuum-Advection-Verfahren	30
3.2.2	Volume-Capturing-Methoden	31
3.2.3	Phasenfeld-Methoden	33
3.2.4	Level-Set-Methoden	34
4	Diskretisierung des Zweiphasenmodells mit Oberflächenspannung	40
4.1	Diskretisierungsgitter und Interpolationsregeln	40
4.2	Diskretisierung der Level-Set Darstellung für freie Oberflächen	42
4.2.1	Glättung, Oberflächenspannung, Normalenvektor und Krümmung .	43
4.2.2	Diskretisierung der Level-Set-Transportgleichung	45
4.2.3	Diskretisierung der Level-Set-Reinitialisierung	48

IV INHALTSVERZEICHNIS

4.2.4	Verbesserte Masseerhaltung über eine CFL-erhaltende Glättung der signum Funktion	50
4.2.5	Picard-Fixpunktiteration zur Masseerhaltung bis auf Maschinengenauigkeit	52
4.3	Die Projektionsmethode für inkompressible Zweiphasenströmungen	53
4.3.1	Diskretisierung der konvektiven Terme mit dem WENO-Verfahren fünfter Ordnung	55
4.3.2	Diskretisierung der viskosen Terme	57
4.3.3	Löser für die Poissongleichung mit nichtkonstanter Dichte	58
4.4	Die adaptive Zeitschrittweitensteuerung	59
4.5	Parallelisierung des Zweiphasen-Strömungslösers	61
4.5.1	Gebietszerlegung als Parallelisierungsstrategie	61
4.5.2	Parallelisierung des Strömungsprogramms	62
4.6	Vollständiger paralleler Algorithmus für inkompressible Zweiphasenströmungen	63
5	Numerische Ergebnisse des parallelen Zweiphasenlösers	66
5.1	Reine Advektion einer Sphäre in einem Rotationsfeld	67
5.2	Die Reinitialisierung für eine komplexe Oberfläche	69
5.3	Masseerhaltung und Konvergenz der Krümmung	71
5.4	Konvergenzanalyse einer aufsteigenden Blase mit Oberflächenspannung	74
5.5	Deformationsverhalten von Tropfen in Scherströmungen	79
II	Kopplung mit beweglichen Starrkörpern	86
6	Generierung und Transport von komplexen Starrkörpergeometrien	87
6.1	Traditionelle Darstellungstechniken komplexer Geometrien	87
6.2	Generierung komplexer Geometrien mittels der Level-Set-Technik	89
6.2.1	Grundidee des Level-Set-Verfahrens zur Beschreibung komplexer Geometrien	90
6.2.2	Berechnung des Vorzeichens für Innen- und Aussendifferenzierung	90
6.2.3	Maschinengenaue Berechnung einer Abstandsfunktion	91
6.2.4	Effizienzsteigerung aufgrund der Lokalitätseigenschaft	94
6.3	Semi-Lagrangischer Transport einer Starrkörpergeometrie	96
6.3.1	Bewegungsparameter eines dreidimensionalen Starrkörpers	96
6.3.2	Reine Bewegungsgleichung für Starrkörper	101
6.3.3	Semi-Lagrangesche Diskretisierung der Starrkörperbewegung	102
7	Erweiterung des Modells auf Fluid-Starrkörper Wechselwirkung	106
7.1	Die Fictitious-Domain-Methode	106
7.2	Behandlung der Fluid-Starrkörper Wechselwirkung	107
7.2.1	Physikalisches Modell	107
7.2.2	Anwendung auf geglättetes Heaviside-Starrkörper-Profil	110

7.3	Erweiterte Chorin-Temam-Projektionsmethode	112
7.4	Diskretisierung der erweiterten Projektionsmethode	115
7.5	Parallelisierung der Starrkörperdynamik	117
7.5.1	Gebietsausschnitt als Parallelisierungsstrategie für dynamische Starrkörper	117
7.6	Vollständiger Algorithmus für inkompressible Zweiphasenströmungen mit dynamischen Starrkörpern	119
8	Numerische Ergebnisse für bewegte Starrkörper in Zweiphasenströmungen	122
8.1	Vergleich der Simulationsergebnisse mit Experimentaldaten	123
8.1.1	Unidirektionale Kopplung: Deformation der freien Oberfläche aufgrund eines mit konstanter Geschwindigkeit aufsteigenden Zylinders	123
8.1.2	Bidirektionale Kopplung: Deformation der freien Oberfläche aufgrund eines frei fallenden Zylinders	126
8.2	Konvergenzuntersuchungen eines bidirektional gekoppelten Starrkörpers in Zweiphasenströmungen	128
8.2.1	Stationärer Fall: Kugel im Kräftegleichgewicht innerhalb einer Fluidphase	128
8.2.2	Dynamischer Fall in quasi-2D: aufsteigender Zylinder	131
8.2.3	Stationärer Fall: Kugel im Kräftegleichgewicht an der freien Oberfläche	133
8.3	Simulationsergebnisse mit komplexen Geometrien und Large-Eddy-Turbulenzmodell	136
8.3.1	Unidirektionale Kopplung: Rotation und Translation des Schiffes RV-Tangaroa	138
8.3.2	Bidirektionale Koppelung: Auftauchendes U-Boot	151
9	Zusammenfassung und Ausblick	158
9.1	Zusammenfassung	158
9.2	Ausblick	160

Symbolverzeichnis

Es werden jeweils die Bezeichnungen für den dreidimensionalen Fall angegeben:

$\vec{a} \cdot \vec{b}$	$= \sum_{i=1}^3 a_i b_i$ ist das Skalarprodukt der beiden Vektoren $\vec{a} = (a_1, a_2, a_3)$ und $\vec{b} = (b_1, b_2, b_3)$
$\vec{a} \otimes \vec{b}$	Tensorprodukt zweier Vektoren ist definiert als 3×3 Matrix mit den Komponenten $(\vec{a} \otimes \vec{b})_{i,j} := a_i b_j$ für $i, j = 1, 2, 3$
\mathbf{n}	$= (n_1, n_2, n_3)^T$ ist der nach außen gerichtete Einheitsnormalenvektor
\mathbf{t}, \mathbf{s}	sind zwei zueinander orthogonal stehende Einheitstangentialvektoren
$\partial_x f; f_x$	erste partielle Ableitung der Funktion f nach x
$\partial_{xx} f; f_{xx}$	zweite partielle Ableitung der Funktion f nach x
∇f	$= \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right)$ ist der Gradient von f
$\partial_{\mathbf{n}} f$	$= (\nabla f) \cdot \mathbf{n}$ ist die Ableitung von f in Richtung der äußeren Normalen der freien Oberfläche oder eines Hindernisgebietes
$\partial_{\mathbf{t}} f$	$= (\nabla f) \cdot \mathbf{t}$ ist die Ableitung von f in Richtung des Tangentialvektors \mathbf{t} der freien Oberfläche oder eines Hindernisgebietes
Δf	$= \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2}$ ist der Laplace-Operator von f
\mathbf{x}	$= (x, y, z)^T$ ist der Ortsvektor
\mathbf{u}	$= (u, v, w)^T$ ist der Geschwindigkeitsvektor
u, v, w	Geschwindigkeitskomponenten in x -, y - und z -Richtung

$\nabla \cdot \mathbf{u}$	$= \operatorname{div} \mathbf{u} := \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}$ ist die Divergenz von \mathbf{u}
$\Omega \subset \mathbb{R}^3$	bezeichnet das Fluidgebiet
$\delta x, \delta y, \delta z$	Schrittweiten in x -, y - und z -Richtung
h	äquidistante Raugittermaschenweite $h = \delta x = \delta y = \delta z$
Ω_h	diskretes Gitter über Ω mit Maschenweite h
$\Omega_h^1, \dots, \Omega_h^P$	Aufteilung des diskreten Gesamtgitters in P Teilblöcke
p	Druck
ρ_i	Dichte der Phase $i = 1, 2$
μ_i	Viskosität der Phase $i = 1, 2$
\mathbf{g}	$= (g_1, g_2, g_3)^T$ sind Volumenkräfte
\mathbf{D}	$= \nabla \mathbf{u} + \{\nabla \mathbf{u}\}^T$ ist der Deformationstensor
\mathbf{I}	Einheitsmatrix
\mathbf{T}	$= -p\mathbf{I} + \mu\mathbf{D}$ ist der Spannungstensor
Re	$= \frac{\rho_i LU}{\mu_i}$ ist die Reynoldszahl (siehe (2.31))
Fr	$= \frac{U^2}{\ \mathbf{g}\ L}$ ist die Froudezahl (siehe (2.32))
We	$= \frac{\rho_i LU^2}{\sigma}$ ist die Weberzahl (siehe (2.33))
σ	Oberflächenspannungskoeffizient
$p_{i,j,k}$	Druck im Zellmittelpunkt $((i - \frac{1}{2})\delta x, (j - \frac{1}{2})\delta y, (k - \frac{1}{2})\delta z)$
$u_{i,j,k}$	Geschwindigkeit in x -Richtung im Zellmittelpunkt (analog für die Geschwindigkeiten in y - und z -Richtung)
$\phi_{i,j,k}$	Level-Set-Funktion im Zellmittelpunkt
δt	Zeitschrittweite
t_n	$= t_{n-1} + \delta t$ ist die Zeit nach dem n -ten Zeitschritt
p^n	Druck zum Zeitpunkt t_n , analog für \mathbf{u} und ϕ

VIII

$\delta\tau$	künstliche Zeitschrittweite in der Reinitialisierungsfunktion
$\delta(\phi)$	Deltafunktional in Abhängigkeit von der Level-Set-Funktion
$H(\phi)$	Heavyside'sche Gewichtsfunktion in Abhängigkeit von der Level-Set-Funktion
$\delta^\epsilon(\phi), H^\epsilon(\phi)$	regularisierte Deltafunktion und regularisierte Heavyside-Funktion
$S(\phi), \text{sign}(\phi)$	signum-Funktion in Abhängigkeit von der Level-Set-Funktion
$S_h(\phi)$	diskrete signum-Funktion
N	Anzahl der Diskretisierungspunkte
$\alpha(\mathbf{x})$	Parameter für das Maß der Regularisierung am freien Rand
$\epsilon(\mathbf{x})$	$= \alpha(\mathbf{x})h$, Maß für die Regularisierung der freien Oberfläche
dof	degrees of freedom (Anzahl der Freiheitsgrade)

Kapitel 1

Einleitung

In fast allen Lebensbereichen lässt sich das Verhalten von Fluiden, das heißt von Flüssigkeiten und Gasen, insbesondere in der Interaktion mit Starrkörpern beobachten. So hat jeder schon mal die Bugwellen eines Schiffes, aufsteigende Kohlensäure–Gasbläschen im Mineralwasser oder auch einfach in Wasserpfützen fallende Regentropfen gesehen.

Fluidströmungen werden durch den Einfluß externer Kräfte, wie zum Beispiel der Gravitation, der Scher- oder Oberflächenspannung oder auch der Bewegungsenergie eines Starrkörpers erzeugt. Für kleine Mach-Zahlen¹, $Ma < 0.3$, können Fluide als *inkompressibel* betrachtet werden, das heißt, die Dichte des Fluids bleibt über alle Zeiten hinweg konstant.

Diese Arbeit behandelt im ersten Teil inkompressible Zweiphasenströmungen mit Oberflächenspannungskräften in drei Raumdimensionen und im zweiten Teil zusätzlich die Interaktion mit komplexen Starrkörpergeometrien. Dabei wird angenommen, daß beide Phasen sich nicht miteinander vermischen können. Zwei nicht mischbare Fluide enthalten immer eine gemeinsame Grenzschicht, an der sie sich berühren. Diese Kontaktfläche wird als *freier Rand* oder auch als *freie Oberfläche* bezeichnet. Charakteristisch für freie Ränder ist, daß sie sich zeitabhängig verändern.

Die Fotos in Abbildung 1.1 zeigen herausfordernde Beispiele für das Deformationsverhalten eines Fluidtropfens in einer Scherströmung. In Abhängigkeit von der ausgeübten Scherrate und den Materialparametern des Tropfens, wie beispielsweise dessen Viskosität und Oberflächenspannungskraft, kann die induzierte Oberflächendynamik zu vollständig unterschiedlichem Deformationsverhalten des Tropfens führen. Hierbei zeigen die ersten beiden Kolumnen in Abbildung 1.1 stationäre Endzustände einer Tropfendeformation und in der letzten Kolumne ist der Fall einer instationären Tropfendeformation abgebildet, bei der die Scherrate des Fluids dazu führt, dass der Tropfen zunächst sehr lang gedehnt wird und schließlich eine Topologieveränderung erfährt, da die Oberflächenspannungskraft nicht mehr die wirkenden Scherkräfte bilanzieren kann. Im Rahmen der Topologieveränderung werden kleinere Satellitentropfchen erzeugt, bei denen das Oberfläche-zu-Volumen-Verhältnis entsprechend gewachsen ist und somit nun die Oberflächenspannungskräfte eine stärkere Wirkung ausüben als zuvor. Dieser Prozess der Topologieveränderung wird soweit fortschreiten, bis die Scherkräfte vollständig von den Oberflächenspannungskräften der Tropfen bilanziert werden können.

¹Die Mach-Zahl ist der Quotient aus Strömungs- und Schallgeschwindigkeit.

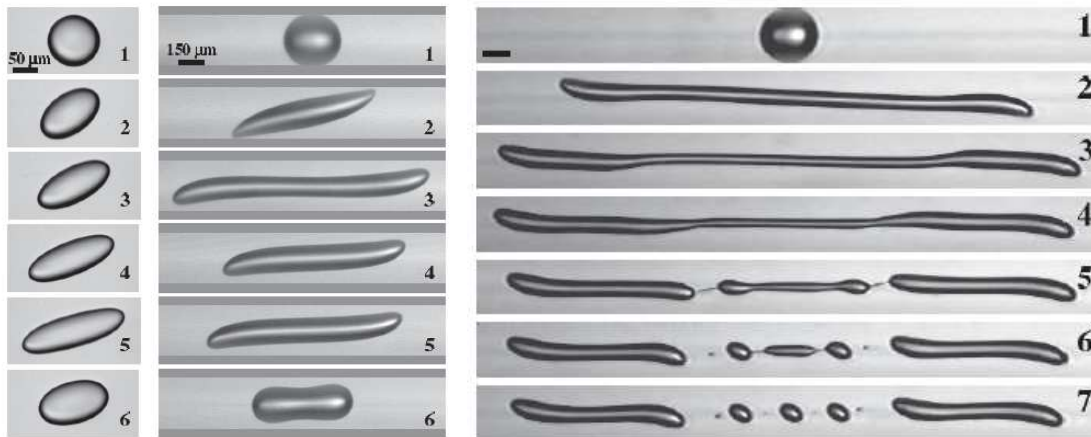


Abbildung 1.1: Tropfen in einer Scherströmung. Erste zwei Kolumnen: Stationäre Endzustände einer Tropfendehformation; Rechts: Instationäre Deformation eines Tropfens bei dem sich Satellitentropfen ablösen. Fotos stammen aus dem Artikel [177].

Zur Analyse der Dynamik freier Oberflächen sind grundsätzlich hochgenaue physikalische Experimente wünschenswert. Im Allgemeinen jedoch können derartige Experimente aus diversen Gründen zu aufwendig oder sogar unmöglich sein. Schwierigkeiten treten in der Regel auf bei sehr kleinen oder sehr großen Längen- und Zeitskalen oder auch aus Sicherheits- und Kostengründen.

Daher wurden mathematische Modelle für die Fluidodynamik entwickelt, die die physikalische Welt einigermaßen exakt beschreiben. Die Modellierung der physikalischen Abläufe stellt hierbei große Herausforderungen an Ingenieure, Physiker und angewandte Mathematiker. Denn fluiddynamische Instabilitäten, wie zum Beispiel Kelvin-Helmholtz- und Rayleigh-Taylor-Instabilitäten [20], führen zu sehr komplizierten topologischen Strukturen des freien Randes. Zusätzlich ist der Einfluss von regularisierenden Effekten, wie der Oberflächenspannung, zu beachten.

Für das in dieser Arbeit verwendete Fluid-Modell der Navier-Stokes-Gleichungen lässt sich in der Regel keine analytische Lösung angeben, weil diese Gleichungen meist stark nichtlinear und nichtlokal sind. Aus diesem Grund wird versucht, die Phänomene der Zweiphasenströmungen mit Hilfe von numerischen Simulationen zu analysieren. Seitdem Hochleistungsrechner, insbesondere Parallelrechner, die hierfür erforderlichen und umfangreichen Berechnungen ermöglichen, ist die Simulation von Zweiphasenströmungen zu einem festen Bestandteil der numerischen Strömungsdynamik (engl. *Computational Fluid Dynamics (CFD)*) geworden.

Eine klassische Strömung, in der diese Schwierigkeiten auftreten, ist beispielsweise der Aufprall eines Fluidtropfens auf ein Substrat. Abbildung 1.2 zeigt eine von uns erzeugte Simulation eines solchen Problems. Diese Forschungsaufgaben stammen aus dem Bereich der Tintendruckertechnologie und aus dem Bereich der Biotechnologie. Diese verfolgen das Ziel, die Tintendrucker hinsichtlich Schriftbild und Patronenverbrauch zu optimieren oder auch Teststreifen für analytische Schnelltests wie beispielsweise Schwangerschafts- oder

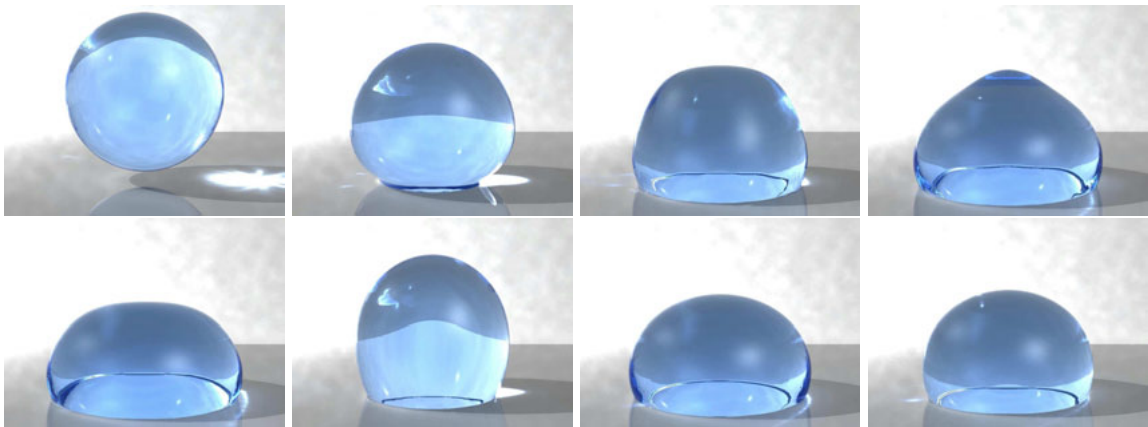


Abbildung 1.2: Simulation eines auf ein Substrat fallenden Fluidtropfens (durchgeführt im Rahmen des Sonderforschungsbereiches 611, Projekt C3). Der zeitliche Verlauf des Tropfenaufpralls ist von links nach rechts dargestellt.

Blutzuckertests zu entwickeln.

Um den gesamten Prozess des Aufpralls zu verstehen, müßte man eine systematische Studie dieses Phänomens mit einer Serie von Laborexperimenten durchführen, welche den vollständigen Parameterraum abdecken. Die Kosten hierfür wären immens. Die numerische Simulation dahingegen ermöglicht eine umfangreiche Parameterstudie zu einem Bruchteil an Zeitaufwand und Kosten. Somit haben numerische Simulationen in vielen Anwendungen aus dem Mikrobereich der Fluidodynamik (Druckerdüsen, Aerosolgemische, Mikropumpen, etc.) einen substantiellen Einfluss auf die Entwicklung von Prototypen.

Das Ziel der numerischen Simulation besteht in der Berechnung einer approximativen Lösung, welche die realen Abläufe möglichst genau wiedergibt. Hierbei besteht ein beträchtliches Problem darin, Diskretisierungsmethoden für die beschreibenden nichtlinearen partiellen Differentialgleichungen zu entwickeln, die eine feste Konvergenzordnung besitzen und keine unphysikalischen Oszillationen bzw. Ergebnisse produzieren. Die Schwierigkeiten liegen im physikalischen System. Dieses ist wegen der Nichtlinearität sehr empfindlich gegenüber kleinen Störungen und kann zusätzlich Singularitäten enthalten, wie zum Beispiel im Kontaktpunkt von zwei sich berührenden Oberflächen.

Im ersten Teil der Arbeit stellen wir zunächst unser Zweiphasenmodell, dessen Diskretisierung und die Parallelisierung vor. Anschließend führen wir einige numerische Validierungen durch, die sowohl die Approximationsgüte untersuchen als auch physikalisch akkurate Simulationen liefern, bei denen sämtliche zu erwartende Oberflächendeformationen und Topologieveränderungen vollständig wiedergegeben werden konnten.

Nach der Validierung des Zweiphasen-Simulationsprogramms präsentieren wir im zweiten Teil der Arbeit eine maßgebliche Erweiterung des ersten Teils. Hier werden die Navier-Stokes-Gleichungen um ein Modell zur Beschreibung der Wechselwirkung mit beweglichen Starrkörpern erweitert.

Die Interaktion von inkompressiblen Zweiphasenströmungen mit Starrkörpern stellt eine substantielle Erweiterung dieses Simulationsprogramms dar, mit der viele zusätzliche



Abbildung 1.3: Links ist das fahrende Forschungsschiff RV-Tangaroa [1] abgebildet und rechts ein fahrendes U-Bootes [2].

natur- und ingenieurwissenschaftliche Fragestellungen simulationstechnisch untersucht werden können. Ob es um die Effizienzsteigerung eines Mixers, die Optimierung des hydrodynamischen Widerstands eines Wasserfahrzeugs oder die Maximierung der Energieausbeute eines Wellenkraftwerks geht, in allen Anwendungen spielt die Wechselwirkung von Zweiphasenströmungen mit beweglichen Starrkörpern eine wichtige Rolle.

So ist beispielsweise eine genaue Vorhersage der hydrodynamischen Kräfte auf ein sich bewegendes Schiff oder U-Boot (siehe Abbildung 1.3) essentiell für das Rumpfdesign. Der Wasserwiderstand zu bestimmten Geschwindigkeiten bestimmt die erforderliche Motorleistung und den Kraftstoffverbrauch. Die Minimierung des Wasserwiderstands ist daher ein wichtiges Ziel beim Schiffsrumpfdesign. Weiterhin induziert die Anregung von Wellenmustern durch das fahrende Schiff nicht nur Wellenwiderstände sondern limitiert dadurch auch die Fahrtgeschwindigkeit in Küstennähe aufgrund von wasserbaulichen Schutzbestimmungen.

Die Simulation von sich bewegendem Starrkörpern birgt weitere beträchtliche Herausforderungen für die Numerik. So ist neben der Darstellung der Starrkörpergeometrien, welche im Rahmen von Gittergenerierungstechniken einen Forschungsbereich für sich darstellt, auch der Transport der Starrkörpergeometrie und schließlich die Einbindung in den Zweiphasenströmungslöser mit zahlreichen Fragestellungen verknüpft. Beim Transport der Starrkörpergeometrie möchte man den Einfluss von numerischen Diffusionsartefakten ausschließen, da ansonsten die für den Starrkörper charakteristische Eigenschaft der Deformationsfreiheit verletzt wird. Bei der Einbindung des beweglichen Starrkörpers in den Zweiphasenlöser ist darauf zu achten, dass während der Bewegung des Starrkörpers die Inkompressibilitätseigenschaft der Fluide erhalten bleibt und der Starrkörper dabei seine Impermeabilitätsbedingung zu jedem Zeitpunkt erfüllt. Weiterhin benötigt man für die Simulation makrodynamischer Fluidströmungen, wie in Abbildung 1.3 dargestellt, Diskretisierungsgitter mit einer enorm hohen Anzahl an Freiheitsgraden. Diese wiederum führen zu einem extrem hohen Rechenaufwand, der nur mittels paralleler Hochleistungsrechner in vertretbarer Zeit zu bewältigen ist und selbst dann ist man in den meisten Fällen noch weit davon entfernt, die Turbulenzen auf sämtlichen Größenskalen aufzulösen. Mit anderen Worten, neben der Parallelisierung des gesamten Algorithmus ist für die Simulation makroskopischer Strömungen mit beweg-

ten Starrkörpern der Einsatz eines Turbulenzmodells unumgänglich.

Numerische Simulation freier Oberflächen

Erste Versuche, aussagekräftige Simulationen von Strömungsproblemen mit freien Rändern zu berechnen, sind in den **60er Jahren** unternommen worden. Die sehr populäre *Marker-and-Cell-Methode* (MAC) wurde von Harlow und Welch [87] im Jahr 1965 entwickelt. Dabei wird die Lage eines Fluides mit freiem Rand durch masselose Partikel dargestellt. Daly [48] erweiterte diese Methode im Jahr 1967 für echte Zweiphasenströmungen. Seitdem bauen viele andere Entwicklungen darauf auf, wie zum Beispiel Daly [49, 50], Amsdon und Harlow [10] und Griebel et. al [77]. Anstelle von Partikeln haben Hirt und Nichols [94] in den **70er Jahren** mit der *Volume-of-Fluid-Methode* (VOF) eine Volumenanteilsfunktion zur Darstellung der freien Oberfläche verwendet. Aus dieser skalaren Funktion, die in jeder Gitterzelle den prozentualen Volumenanteil des Fluids zum Gesamtzellvolumen angibt, wird der freie Rand im zweidimensionalen Fall durch Liniensegmente rekonstruiert (siehe Abbildung 3.4). Die erste von Noh und Woodward [140] entwickelte *SLIC* Rekonstruktion approximiert die Oberfläche mittels horizontaler und vertikaler Liniensegmente. Diese wurde in [94] durch das *PLIC* Verfahren verbessert, welches Liniensegmente mit einer beliebigen Steigung verwendet. In den **80er Jahren** ist die *Level-Set-Methode* von Osher und Sethian [142] entwickelt worden. Hier wird der freie Rand mit der Niveaumenge zum Niveauewert Null einer höherdimensionalen skalaren Funktion identifiziert. Erste Anwendungen der Level-Set-Methode im Bereich der Fluidodynamik sind von Mulder, Osher und Sethian [136] unternommen worden. Sie simulierten das Verhalten von aufsteigenden Gasblasen. In den **90er Jahren** wurde erstmals die Modellierung von Zweiphasenströmungen um die Oberflächenspannung erweitert. Dabei haben Brackbill et al. [23] die *Continuous-Surface-Force-Methode* (CSF) entwickelt und mit dem VOF-Verfahren gekoppelt. Unabhängig davon entwickelten Unverdi et al. [191] eine CSF-ähnliche Methode und setzten sie im Zusammenhang mit der MAC-Technik ein. Diese CSF-Verfahren beschreiben die Oberflächenspannungskraft als einen Quellterm, der über ein in der Nähe der freien Oberfläche definiertes Deltafunktional aktiviert wird. Lafaurie, Nardonne, Scardovelli, Zaleski und Zanetti [118] lieferten mit der *Continuous-Surface-Stress-Methode* (CSS) ein ähnliches Modell, das sie mit der VOF-Technik koppelten.

Erste Simulationen von Fluid-Starrkörper-Interaktionen wurden ebenfalls in den **90er Jahren** im Rahmen von Fluid-Partikel-Strömungen entwickelt. Hierbei haben sich Glowinski et al. [70–72], Hu et al. [58, 96, 97] und Patankar et al. [145] auf zweidimensionale Strömungen von Partikeln innerhalb einer Fluidphase beschränkt, die sie mit einer Finite-Elemente-Diskretisierung behandelt haben. In den **letzten zehn Jahren** wurden Entwicklungen auf dem Gebiet der Fluid-Starrkörper-Interaktionen von Peric et al. [52], Nakayama et al. [110, 138] und Karniadakis et al. [129] vorangetrieben. Dabei haben Peric et al. ihre Simulationen auf Basis einer zweidimensionalen Arbitrary Lagrangian-Eulerian (ALE) Formulierung umgesetzt. Nakayama et al. hingegen haben in [110, 138] die zweidimensionale Strömung von Partikeln innerhalb einer Fluidphase mittels eines geglätteten Starrkörperprofils betrachtet, und Karniadakis et al. haben in [129] diese Methode einer numerischen

Fehleranalyse unterzogen.

Viele der vorgestellten Verfahren und Artikel sind Pionierarbeiten, die sich ausschließlich auf zweidimensionale Simulationen beschränken und einige Probleme, wie beispielsweise die Masseerhaltung, Topologieveränderungen oder auch die Konvergenzordnung der Krümmung und die Zweiphasen-Starrkörper-Interaktion, unbeachtet lassen. Auch auf eine effiziente Nutzung der Rechnerressourcen durch parallel ablaufende Algorithmen wird nicht eingegangen. Daraus ergeben sich eine Reihe interessanter Fragestellungen und Verbesserungsmöglichkeiten, die den Ausgangspunkt dieser Arbeit bilden.

Zu dieser Arbeit

Diese Arbeit ist in zwei Teile gegliedert. Im ersten Teil wird ein physikalisches Modell für inkompressible Zweiphasenströmungen mit freien Oberflächen inklusive Oberflächenspannung vorgestellt. Dabei werden Zweiphasenströmungen betrachtet, die an der Phasengrenze einen Dichte- und Viskositätssprung enthalten. Die Diskretisierung des dreidimensionalen Modells enthält mehrere Schwierigkeiten. So ist zum Beispiel eine möglichst gute Darstellung der freien Oberfläche wichtig für eine genaue Berechnung ihrer Krümmung, die wiederum zur Beschreibung der Oberflächenspannungskräfte benötigt wird. Hierbei ist die Level-Set-Methode zur Approximation der freien Oberfläche eingesetzt worden. Sie betrachtet den freien Rand im Allgemeinen als die Niveaumenge zum Niveaufwert Null einer höherdimensionalen skalaren Funktion, welche als Level-Set-Funktion bezeichnet wird. Im Rahmen dieser Arbeit wurde die Level-Set-Funktion derart gewählt, dass sie in jedem diskreten Punkt des gesamten Rechengebietes den Abstand zum freien Rand angibt. Damit können der Normalenvektor und die Krümmung auf effiziente Weise implizit aus der Level-Set-Funktion bestimmt werden. Ein grundsätzlicher Vorteil dieser Methode ist die einfache und robuste Handhabung von topologischen Änderungen wie zum Beispiel das Zusammentreffen oder Abspalten von Tröpfchen.

Die zeitliche Entwicklung der freien Oberfläche wird über den Eulerschen Transport der Level-Set-Funktion mit den Strömungsgeschwindigkeiten ermittelt. Deshalb ist es notwendig, dieses Geschwindigkeitsfeld möglichst gut zu approximieren. Bei der Berechnung der Geschwindigkeiten müssen unphysikalische Oszillationen verhindert werden, die insbesondere im Fall konvektionsdominanter Strömungen auftreten können. Solche Oszillationen übertragen sich auf den freien Rand und führen zu unbrauchbaren Ergebnissen. Diskretisierungsmethoden erster Ordnung verhindern zwar unerwünschte Oszillationen, aber sie induzieren eine starke numerische Diffusion, wodurch ein substantieller Masseverlust entsteht, der ebenfalls zu unbrauchbaren Ergebnissen führt. Naive Verfahren höherer Ordnung, wie beispielsweise zentrale Differenzen, zeigen ein deutlich geringeres diffusives Verhalten, sind aber in der Regel nicht robust.

Aus diesem Grund verwenden wir ein WENO (weighted essentially non-oscillatory) Verfahren fünfter Ordnung zur Diskretisierung sämtlicher anfallender konvektiver Terme. Hierbei zeigen wir anhand zahlreicher numerischer Untersuchungen, dass dieses Verfahren eine sehr gute Kombination aus hoher Approximationsgüte und Robustheit darstellt.

Die Oberflächenspannung wird mittels des *CSF*-Modells durch einen zusätzlichen Quell-

term in den Navier-Stokes-Impulsgleichungen modelliert. Die Wirkung des Quelltermes muß dabei auf die freie Oberfläche eingeschränkt werden. Dies geschieht über ein Delta-funktional, welches in einer kleinen Umgebung des freien Randes die Kräfte des Quelltermes aktiviert. Weiterhin erfordert die Oberflächenspannung die Berechnung des Normalenvektors und der Krümmung der freien Oberfläche. Diese geometrischen Größen lassen sich hierbei, im Gegensatz zu VOF-Methoden, sehr effizient und akkurat aus der Level-Set-Funktion ermitteln.

Das wohl bekannteste Problem der Level-Set-Methode ist die Erhaltung der Masse. Eigene Untersuchungen ergaben, dass diese umso mehr verletzt wird, je größer die Oberflächenspannungskräfte sind, denn durch den Transportschritt werden in einer kleinen Umgebung der freien Oberfläche die Niveaumengen der Level-Set-Funktion aufgrund der Oberflächenspannungskräfte komprimiert oder auch gestreckt. Dadurch entstehen sehr große oder sehr kleine Gradienten in der Level-Set-Funktion, die ihrerseits dann keine Abstandsfunktion mehr ist. Über einen Reinitialisierungsprozeß wird die Level-Set-Funktion wieder zu einer Abstandsfunktion transformiert. Dieser Prozeß induziert aber auch eine künstliche Bewegung des freien Randes, wodurch die Erhaltung der Masse nicht mehr garantiert werden kann. Der Grund hierfür liegt unter anderem darin, dass eine herkömmliche Diskretisierung gegen die CFL-Bedingung verstößt und sich die freie Oberfläche auf unphysikalische Weise weiter als eine Gittermaschenweite bewegen kann. Durch eine modifizierte Approximation der im Reinitialisierungsprozess auftretenden Vorzeichenfunktion kann die Einhaltung der CFL-Bedingung forciert werden. In Kombination mit dem WENO-Verfahren fünfter Ordnung und einer anschließenden Picard-Iteration für die Masseerhaltung kann diese bis auf Maschinengenauigkeit insbesondere für Langzeitrechnungen erhalten werden.

Die Navier-Stokes-Gleichungen werden nach der Chorin-Temam-Projektionsmethode [37, 186] gelöst. Ein Teilschritt dieser Methode erfordert das Lösen eines großen linearen Gleichungssystems. Da der Dichtesprung der beiden Phasen in das lineare Gleichungssystem eingeht, entsteht folglich bei hohen Dichteunterschieden ein schlecht konditioniertes Problem. Hierzu ist neben einem parallelen Jacobi vorkonditionierten BiCGSTAB [193] auch ein paralleles Algebraisches-Mehrgitter-Verfahren (AMG) [130] implementiert worden, welches das Gleichungssystem auch im Fall schlechter Konditionierung effizient löst.

Am Beispiel einer Sphäre in einem Rotationsfeld untersuchen wir zunächst das numerische Verhalten der reinen Advektion und zeigen, dass der Einfluss numerischer Diffusion aufgrund des WENO-Verfahrens fünfter Ordnung gering ist. Danach untersuchen wir die Reinitialisierung einer Level-Set-Funktion, die zu Beginn keine Abstandsfunktion ist und deren Niveaumenge zum Niveauwert Null eine komplexe Oberfläche darstellt. Nach der Reinitialisierung erhalten wir eine globale vorzeichenbehaftete Abstandsfunktion, die dieselbe Niveaumenge zum Wert Null aufweist wie zu Beginn. Daraufhin untersuchen wir die Masseerhaltung und die Konvergenz der Krümmung einer mit einem konstanten Geschwindigkeitsfeld diagonal durch das Rechengebiet transportierten Sphäre. Anschließend führen wir eine Konvergenzanalyse des gesamten Zweiphasenlösers am Beispiel einer aufsteigenden Blase mit Oberflächenspannung durch, wobei erstmalig der Einfluss der Picard-Iteration für die globale Masseerhaltung auf die gesamte Konvergenzordnung des Verfahrens untersucht wird. Weiterhin vergleichen wir unsere numerischen Ergebnisse einer aufsteigenden Luftblase mit denen aus [109, 192] und vergleichen diese darüber hinaus mit unseren zusätz-

lichen Ergebnissen einer realen, dreidimensionalen Version des Testproblems. Abschließend führen wir eine Parameterstudie für Zweiphasenströmungen durch und zeigen zahlreiche Simulationen von Tropfen in einer Scherströmung, bei denen die Kapillar- und Reynoldszahl variieren, siehe Abbildung 1.1. Entsprechend dem (Ca, Re) -Phasendiagramm konnten alle zu erwartenden Topologieveränderungen der Tröpfchendeformation in drei Dimensionen simuliert werden.

Der zweite Teil dieser Arbeit befasst sich mit der Generierung einer Level-Set-Funktion aus einer entsprechend vorgegebenen komplexen triangulierten Starrkörpergeometrie, dann mit der Starrkörperdynamik und ihrer semi-Lagrangeschen Diskretisierung und letztendlich mit der vollständigen Einbindung in den inkompressiblen Zweiphasenströmungslöser aus dem ersten Teils dieser Arbeit. Die Kopplung mit der Starrkörperdynamik verläuft über einen zusätzlichen Lagrange-Multiplikator-Ansatz. Ähnlich dem Druckfeld aus der Chorin-Temam-Projektionsmethode, welcher die Inkompressibilität der Fluide forciert, wird hier ein zusätzlicher Druck innerhalb des Starrkörpers ermittelt, der die Deformationsfreiheit und die physikalisch akkurate Starrkörperbewegung forciert und diese mit dem Geschwindigkeits- und Druckfeld der Zweiphasenströmung koppelt.

Die numerische Umsetzung verläuft hierbei über eine Eulersche Beschreibung auf einem festen Gitter bei dem die Interaktion zwischen den Zweiphasen und dem Starrkörper mittels über dem Starrkörper definierten Volumenkräften vollzogen wird. Die Darstellung der Starrkörpergeometrie als Level-Set-Funktion ermöglicht es uns, ein über 1-2 Gitterzellen geglättetes Dichteprofil der Starrkörpergeometrie vollständig in die Navier-Stokes-Gleichungen einzubinden, wobei eine über den Starrkörper wirkende glatte Volumenkraft direkt in die Interaktion mit den Navier-Stokes-Gleichungen eingeht. Diese Technik hat im Vergleich zu den Methoden, in denen die Starrkörpergeometrie nur über Randbedingungen in die Strömungsgeschwindigkeiten eingeht, den großen Vorteil, dass keine regelmäßig anfallenden Gittergenerierungen aufgrund von Stabilitätsbedingungen durchgeführt werden müssen. Andererseits muss man hier ein Transportschema für den Starrkörper einsetzen, welches nur die reine Starrkörperbewegung, die aus Translation und Rotation besteht, in Betracht zieht und keine zusätzlichen numerischen Diffusionsartefakte induziert, wie sie bei Eulerschen Transportschemata auftreten. Aus diesem Grund verwenden wir ein semi-Lagrangesches Verfahren für den Transport des Starrkörpers, das die Formerhaltung des Starrkörpers auch im Rahmen von Langzeitrechnungen hervorragend erfüllt.

Das in dieser Arbeit entwickelte Verfahren ist artverwandt zu den Verfahren von Glowinski et al. [70–72] und Nakayama et al. [110, 138], die im Rahmen von zweidimensionalen partikelhaltigen Strömungen ebenfalls einen Lagrange-Multiplikator-Ansatz wählten. Sie simulierten hierbei einfache zweidimensionale Kreisgeometrien als Partikel innerhalb einer Fluidphase.

In dieser Arbeit wurde, soweit dem Autor bekannt, erstmals der Lagrange-Multiplikator-Ansatz mit der Level-Set-Technik gekoppelt und auf reale dreidimensionale Zweiphasen-Starrkörper-Interaktionen mittels einer semi-impliziten Lösungsmethode erfolgreich umgesetzt. Das gesamte Verfahren ist darüber hinaus erstmalig parallelisiert und mit einem Large-Eddy-Turbulenzmodell versehen worden, sodass auch massiv parallel berechnete Simulationen realer dreidimensionaler makrodynamischer Strömungsprozesse ermöglicht werden.

Der gesamte Algorithmus ist in C++ programmiert und mit der Kommunikationsbibliothek *Message-Passing-Interface* (MPI) [211] parallelisiert worden, um dreidimensionale Probleme mit hoher Auflösung effizient berechnen zu können. Als Strategie für die Parallelisierung wurde hierbei für die Strömungsgleichungen und die Kraftberechnungsroutinen die Gebietszerlegungsmethode [98] eingesetzt, durch die das gesamte räumliche Rechengebiet in Blöcke eingeteilt und dann auf einzelne Prozessoren verteilt wird. Der Transport des Starrkörpers, welcher im Allgemeinen nur einen Bruchteil des gesamten Strömungsgebietes einnimmt, wurde mittels der *Replicated-Data*-Technik parallelisiert.

Am Beispiel eines Zylinders, welcher ins Wasser fällt, wird das gesamte gekoppelte numerische Verfahren anhand von Labormessdaten validiert. Darüber hinaus untersuchen wir die globale Approximationsgüte des numerischen Verfahrens am Beispiel eines aufsteigenden Zylinders und untersuchen das hydrodynamische Gleichgewicht einer Kugel innerhalb einer Fluidphase und auf der freien Oberfläche.

Abschließend zeigen parallel berechnete Simulationen, inklusive Large-Eddy-Turbulenzmodell, am Beispiel des fahrenden Forschungsschiffes Tangaroa und am Beispiel eines auftauchenden U-Bootes – vergleiche Abbildung 1.3 – die breiten ingenieurstechnischen Anwendungsbereiche dieses hierfür entwickelten Simulationsprogramms.

Kurzer Überblick

Diese Arbeit ist in zwei Hauptteile gegliedert. Der erste Teil, welcher die Kapitel 2,3,4 und 5 umfasst, beschreibt das inkompressible Zweiphasenmodell mit Oberflächenspannung, dessen Diskretisierung und Validierung. Der zweite Teil, bestehend aus den Kapiteln 6,7 und 8, befasst sich mit der Kopplung von Zweiphasenströmungen und beweglichen Starrkörpern. Es wird die Generierung komplexer Starrkörpergeometrien, deren Bewegungsgesetze und schließlich die Kopplung mit Zweiphasenströmungen beschrieben. Das diskretisierte Modell wird anhand von Labormessdaten validiert, und es wird die numerische Approximationsgüte des Modells bestimmt. Abschließend zeigen zwei herausfordernde Simulationen inklusive Large-Eddy-Turbulenzmodell die vielseitige Anwendbarkeit des hier entwickelten Simulationsprogramms.

Kapitel 2 erklärt die physikalisch-mathematischen Gleichungen, die zur Modellierung von inkompressiblen Zweiphasenströmungen mit Oberflächenspannung verwendet werden.

Kapitel 3 gibt einen Überblick über aktuelle numerische Techniken zur Diskretisierung von freien Oberflächen. Die Methoden können hierbei in zwei Klassen unterteilt werden, die Tracking-Verfahren und die Capturing-Techniken.

Kapitel 4 beschreibt die Diskretisierung des Zweiphasenmodells mit Oberflächenspannung. Neben der eingesetzten Projektionsmethode zur Lösung der Navier-Stokes-Gleichungen, beschreiben wir die Glättungsprozedur der Fluid-Materialparameter, die Behandlung der freien Oberfläche, die Berechnung der Oberflächenspannung, wie auch die Parallelisierung des gesamten Zweiphasen-Algorithmus.

Kapitel 5 zeigt detaillierte numerische Ergebnisse des parallelen Zweiphasenlösers. Hier untersuchen wir zunächst die reine Advektion einer Sphäre in einem Rotationsfeld, dann die Generierung einer Abstandsfunktion bezüglich einer komplexen Oberfläche. Anschließend

wird die Masseerhaltung und die Konvergenz der Krümmung untersucht und die Approximationsgüte des gesamten Algorithmus am Beispiel einer aufsteigenden Blase mit Oberflächenspannung. Abschließend wird eine Parameterstudie bezüglich eines Tropfens in einer Scherströmung durchgeführt, bei der sämtliche Topologieveränderungen simuliert werden konnten.

Kapitel 6 befasst sich mit der Generierung von komplexen Starrkörpergeometrien und deren Transport, der sich auf Translation und Rotation beschränkt. Die Diskretisierung der Transportgleichung für Starrkörper wird mittels eines semi-Lagrangischen Verfahrens umgesetzt, welches die Deformationsfreiheit eines Starrkörpers sehr gut erhält.

Kapitel 7 erklärt das physikalische Modell zur Beschreibung von Fluid-Starrkörper-Wechselwirkungen, dessen Diskretisierung in Form einer erweiterten Chorin-Temam-Projektionsmethode und dessen Parallelisierung.

Kapitel 8 zeigt detaillierte numerische Ergebnisse des parallelen Zweiphasen-Starrkörper-Lösers. Hierbei validieren wir das entwickelte Simulationsprogramm anhand von physikalischen Messdaten, bestimmen dessen numerische Approximationsgüte und untersuchen das numerische Verhalten bezüglich hydrostatischer Gleichgewichte einer Kugel innerhalb einer Phase und auf der freien Oberfläche. Abschließend simulieren wir das Strömungsverhalten einer zwangsgeführten Schiffskörpergeometrie und das Auftauchen eines U-Bootes.

Kapitel 9 beendet diese Arbeit mit einer kurzen Zusammenfassung der Ergebnisse und einem Ausblick auf mögliche Weiterentwicklungen des Simulationsprogramms.

Teil I

Simulation von inkompressiblen Zweiphasenströmungen

Kapitel 2

Das Modell für Zweiphasenströmungen mit Oberflächenspannung

In diesem Kapitel werden die dem Modell zugrundeliegenden physikalischen Gleichungen behandelt. Nach einer kurzen Herleitung der Navier-Stokes-Gleichungen [119] wird ausführlich auf die Modellierung von inkompressiblen Zweiphasenströmungen mit Oberflächenspannung eingegangen, wobei insbesondere die Randbedingungen an der freien Oberfläche und an den festen Rändern beschrieben werden [43].

2.1 Die Navier-Stokes-Gleichungen

Mit den Navier-Stokes-Gleichungen läßt sich das Verhalten instationärer Strömungen von viskosen, inkompressiblen Fluiden beschreiben. Sie leiten sich aus den Erhaltungssätzen für Masse und Impuls her.

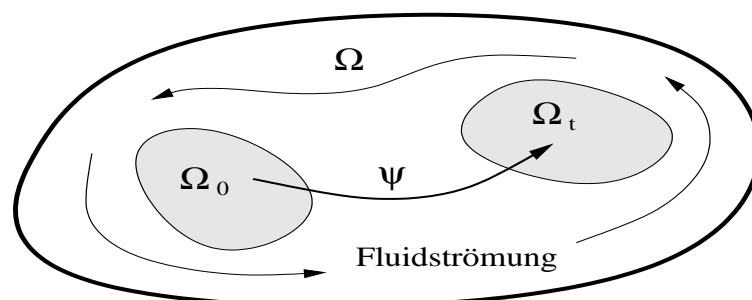


Abbildung 2.1: Zeitliche Entwicklung von Ω_0 nach Ω_t .

Im Folgenden sei $\Omega \subset \mathbb{R}^3$ ein festes Gebiet, welches mit einem Fluid gefüllt ist. Für $\mathbf{x} \in \Omega$ beschreibt die Abbildung $\vec{\psi}_{\mathbf{x}} : t \mapsto \vec{\psi}(\mathbf{x}, t)$ mit $\vec{\psi} : \Omega \times [0, t_{end}] \mapsto \Omega$ die Trajektorie des Partikels \mathbf{x} . Mit $\vec{\psi}_t : \mathbf{x} \mapsto \vec{\psi}(\mathbf{x}, t)$ wird die Position \mathbf{x} eines Partikels auf seine Position zur Zeit t abgebildet. Die Abbildung $\vec{\psi}$ wird *Fluidströmungsabbildung* genannt. Hier und im

Folgendes sei vorausgesetzt, dass $\vec{\psi}$ die jeweils benötigten Regularitätsbedingungen erfüllt, und dass $\vec{\psi}_t$ für alle $t \geq 0$ invertierbar ist. Ein beliebiges Teilgebiet $\Omega_0 \subset \Omega$ wird so durch die Fluidströmung in das Teilgebiet $\vec{\psi}_t(\Omega_0) =: \Omega_t \subset \Omega$ transportiert (siehe Abbildung 2.1). Die zugehörige Geschwindigkeit \mathbf{u} des Fluids an einem festen Ort $\mathbf{x} \in \Omega_t$ zum Zeitpunkt t folgt aus der Fluidströmungsabbildung über zeitliches Differenzieren

$$\mathbf{u}(\mathbf{x}, t) := \frac{\partial \vec{\psi}}{\partial t}(\vec{a}, t) \quad \text{mit} \quad \vec{a} := \vec{\psi}_t^{-1}(\mathbf{x}) \in \Omega_0. \quad (2.1)$$

Die Herleitung der Navier-Stokes-Gleichungen benötigt das Transporttheorem. Aus diesem geht hervor, wie die Zeitableitung eines Integrals über ein zeitabhängiges Gebiet berechnet werden kann.

Theorem 2.1. (Transporttheorem)

Für eine differenzierbare, skalare Funktion $f : \Omega \times [0, t_{end}] \rightarrow \mathbb{R}$, $(\mathbf{x}, t) \mapsto f(\mathbf{x}, t)$ gilt:

$$\frac{d}{dt} \int_{\Omega_t} f(\mathbf{x}, t) d\mathbf{x} = \int_{\Omega_t} \{ \partial_t f + \nabla \cdot (f \mathbf{u}) \} (\mathbf{x}, t) d\mathbf{x}. \quad (2.2)$$

Beweis:

siehe Chorin & Marsden [38]. \square

Mit Hilfe dieses Theorems kann die Kontinuitätsgleichung aus der Masseerhaltung gefolgt werden.

2.1.1 Die Erhaltung der Masse

Die Masse eines Fluids, welches zum Zeitpunkt t das Volumen Ω_t einnimmt, ist gleich dem Integral über seine Dichte $\rho(\mathbf{x}, t)$. Die Masseerhaltung besagt dann, dass die Ableitung der Masse nach der Zeit gleich Null sein muß. Zusammen mit dem Transporttheorem folgt

$$0 = \frac{d}{dt} \int_{\Omega_t} \rho(\mathbf{x}, t) d\mathbf{x} = \int_{\Omega_t} (\partial_t \rho + \nabla \cdot (\rho \mathbf{u})) (\mathbf{x}, t) d\mathbf{x} \quad \forall \Omega_t, t \geq 0.$$

Weil Ω_0 , und deshalb auch Ω_t , beliebig gewählt werden darf und die Integranden stetig sind, lautet die Kontinuitätsgleichung für kompressible Fluide in differentieller Form

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0. \quad (2.3)$$

Für inkompressible Fluide, deren Dichte unabhängig von Ort und Zeit ist, reduziert sich die Kontinuitätsgleichung zu

$$\nabla \cdot \mathbf{u} = 0. \quad (2.4)$$

Vorweggenommen sei, dass die Beschreibung von inkompressiblen Zweiphasenströmungen darauf beruhen wird, dass für jede Phase Gleichung (2.4) erfüllt sein muß, aber die freie Oberfläche analog zu (2.3) transportiert wird.

2.1.2 Die Erhaltung des Impulses

Der Impuls eines Fluides mit dem Anfangsvolumen Ω_0 zur Zeit t läßt sich darstellen als

$$\mathbf{m}(\Omega_0, t) := \int_{\Omega_t} \rho(\mathbf{x}, t) \mathbf{u}(\mathbf{x}, t) \, d\mathbf{x} .$$

Newtons zweites Gesetz besagt, dass die zeitliche Änderung des Impulses gleich der Summe der Kräfte ist, die auf das Fluid wirken, also

$$\frac{d}{dt} \mathbf{m}(\Omega_0, t) = \sum \text{Kräfte} . \quad (2.5)$$

Hierbei setzen sich die Kräfte aus verschiedenen Kraftarten zusammen:

- *Volumenkräfte* (zum Beispiel Gravitation, Coriolis-Kraft, magnetische Kraft, ...), die sich in der Form $\int_{\Omega_t} \rho(\mathbf{x}, t) \mathbf{g}(\mathbf{x}, t) \, d\mathbf{x}$ ausdrücken lassen, mit einer gegebenen Kraftdichte $\rho \mathbf{g}$ (Kraft pro Einheitsvolumen). Im Rahmen dieser Arbeit betrachten wir als Volumenkraft ausschließlich die Gravitation.
- *Oberflächenkräfte* (zum Beispiel Druck, innere Reibung, ...), die sich in der Form $\int_{\partial\Omega_t} \mathbf{T}(\mathbf{x}, t) \mathbf{n} \, dF$ ausdrücken lassen, mit $\mathbf{T} \in \mathbb{R}^{3 \times 3}$ als dreidimensionaler Spannungstensor und \mathbf{n} als Normalenvektor auf $\partial\Omega_t$.

Das zweite Newtonsche Gesetz (2.5) lautet somit

$$\frac{d}{dt} \int_{\Omega_t} \rho(\mathbf{x}, t) \mathbf{u}(\mathbf{x}, t) \, d\mathbf{x} = \int_{\Omega_t} \rho(\mathbf{x}, t) \mathbf{g}(\mathbf{x}, t) \, d\mathbf{x} + \int_{\partial\Omega_t} \mathbf{T}(\mathbf{x}, t) \mathbf{n} \, dF .$$

Nach Anwendung des Transporttheorems (2.2) gilt

$$\int_{\Omega_t} (\partial_t(\rho \mathbf{u}) + \nabla \cdot (\mathbf{u} \otimes (\rho \mathbf{u}))) \, d\mathbf{x} = \int_{\Omega_t} \rho \mathbf{g} \, d\mathbf{x} + \int_{\partial\Omega_t} \nabla \cdot \mathbf{T} \, d\mathbf{x} , \quad (2.6)$$

wobei zusätzlich das Randintegral auf der rechten Seite nach dem Satz von Gauß in ein Volumenintegral umgewandelt wurde. Da dies für beliebige Ω_t gilt, lautet die Impulsgleichung in differentieller Form

$$\partial_t(\rho \mathbf{u}) + \nabla \cdot (\mathbf{u} \otimes (\rho \mathbf{u})) - \rho \mathbf{g} - \nabla \cdot \mathbf{T} = 0 . \quad (2.7)$$

Der Spannungstensor \mathbf{T} modelliert die inneren Reibungskräfte. Für *viskose Newtonsche Fluide* hat er die Form

$$\mathbf{T} := (-p + \underbrace{\lambda(\nabla \cdot \mathbf{u})}_{=0, \text{ nach (2.4)}}) \mathbf{I} + \mu \mathbf{D} = -p \mathbf{I} + \mu \mathbf{D} , \quad (2.8)$$

mit p als Druck, \mathbf{I} als Einheitsmatrix, μ und λ als Viskositätskonstanten und mit dem *Deformationstensor*

$$\mathbf{D} = \nabla \mathbf{u} + \{\nabla \mathbf{u}\}^T .$$

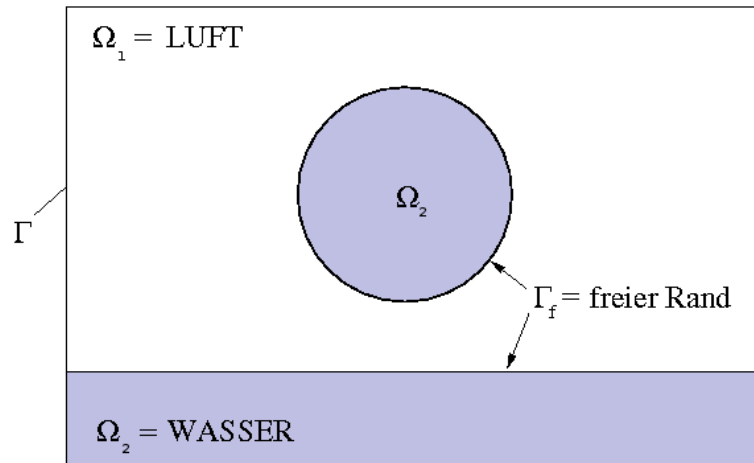


Abbildung 2.2: Zur Zweiphasenmodellierung.

Die Annahme der Inkompressibilität führt dazu, dass der Term $\lambda(\nabla \cdot \mathbf{u})$ in (2.8) wegfällt. Wird schließlich der Spannungstensor (2.8) in die Gleichung (2.7) eingesetzt, so ergibt sich ein System partieller Differentialgleichungen zweiter Ordnung

$$\rho(\partial_t(\mathbf{u}) + \nabla \cdot (\mathbf{u} \otimes (\mathbf{u}))) + \nabla p = \nabla \cdot \mu \mathbf{D} + \rho \mathbf{g} \quad \text{mit } \rho = \text{konstant} . \quad (2.9)$$

Für die nun folgende Beschreibung des Zweiphasensystems wird das gekoppelte Gleichungssystem aus (2.4) und (2.9) verwendet. Aufgrund der Voraussetzung von Inkompressibilität und konstanter Temperatur des Fluides ist die Energiegleichung entkoppelt und wird daher zur Berechnung der Strömung nicht benötigt.

2.2 Das Modell für inkompressible Zweiphasenströmungen

Das nun folgende Modell für inkompressible Zweiphasenströmungen basiert im wesentlichen auf den Navier-Stokes-Gleichungen. Diese werden jedoch zusätzlich um Randbedingungen entlang der Zweiphasengrenzschicht erweitert, so dass Oberflächenspannungskräfte zwischen beiden Phasen wirken können.

2.2.1 Physikalisches Modell

Das Verhalten von zwei unmischbaren inkompressiblen Fluiden wird durch die inkompressiblen Navier-Stokes-Gleichungen beschrieben, die im Folgenden auf einer offenen Menge $\Omega = \Omega_1 \cup \Omega_2 \cup \Gamma_f \subset \mathbb{R}^3$ mit Lipschitz-Rand $\Gamma := \partial\Omega$ definiert sind. Hierbei bezeichnen Ω_1 und Ω_2 die beiden Teilgebiete der zwei unterschiedlichen Fluide. Der freie Rand zwischen den zwei Fluidphasen wird als $\Gamma_f := \partial\Omega_1 \cap \partial\Omega_2$ definiert. Die zwei Fluidgebiete Ω_1 und Ω_2 sind zusammen mit dem freien Rand von der Zeit abhängig. Die zeitliche Entwicklung beider Fluide wird über die klassischen Navier-Stokes-Gleichungen in den betreffenden

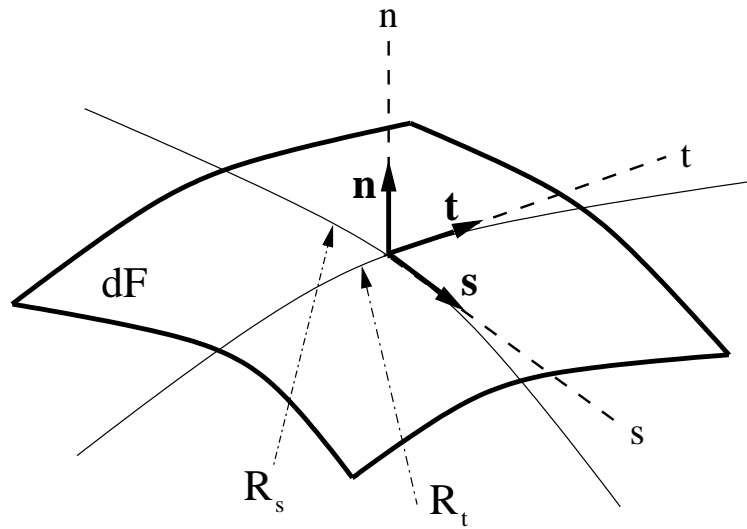


Abbildung 2.3: Zur Beschreibung der Oberflächenspannungskraft.

Teilgebieten $\Omega_i(t)$ für $t \in [0, T]$ beschrieben. Somit lautet das System wie folgt

$$\begin{aligned}
 \rho_i \frac{D\mathbf{u}_i}{Dt} &= -\nabla p_i + \nabla \cdot (\mu_i \mathbf{D}_i) + \rho_i \mathbf{g} && \text{in } \Omega_i, \\
 \nabla \cdot \mathbf{u}_i &= 0 && \text{in } \Omega_i, \\
 \mathbf{u}_i|_{\Gamma} &= 0 && \text{in } [0, T], \\
 \mathbf{u}_i|_{t=0} &= u_{0_i} && \text{in } \Omega_i,
 \end{aligned} \tag{2.10}$$

wobei $i \in \{1, 2\}$ die jeweilige Fluidphase bezeichnet und $\frac{D\mathbf{u}}{Dt} = \frac{d\mathbf{u}}{dt} + \mathbf{u} \cdot \nabla \mathbf{u}$ die Materialableitung ist. Die Viskositäten μ_i und Dichten ρ_i beider Phase werden als konstant vorausgesetzt. Allerdings ist das System (2.10) nicht vollständig ohne zusätzliche Bedingungen auf der freien Oberfläche Γ_f .

Die grundlegende Annahme in der Kontinuumsmechanik für freie Oberflächen Γ_f ist, dass diese als scharfe Grenzschichten betrachtet werden kann; das heißt, die Grenzschicht ist eine zweidimensionale Oberfläche, welche die zwei Fluidgebiete voneinander trennt. Für unmischbare Fluide kann dieses über Dimensionsanalyse gezeigt werden, weiterführende Betrachtungen findet man in [119]. Daher enthält unser Zweiphasenmodell einen Sprung in der Dichte und einen Sprung in der Viskosität entlang der freien Oberfläche. Eine Folgerung dieser Voraussetzung ist, dass die Grenzschicht keine Masse besitzt. Deshalb verschwinden die Netto-Spannungen entlang der Grenzschicht. Darüber hinaus müssen die Geschwindigkeiten quer durch die Grenzschicht stetig sein, das heißt $\mathbf{u}_1 = \mathbf{u}_2$ auf Γ_f . Für eine detailliertere Betrachtung sei auf [119] verwiesen. Falls Wechsel des Aggregatzustandes, wie z.B. Gefrier- oder Kondensationsprozesse, ignoriert werden können, so gelten die folgenden Randbedingungen auf der Grenzschicht Γ_f zwischen den beiden Fluidphasen:

- *Kinematische Bedingung:* Die freie Oberfläche bildet eine scharfe Trennfläche, durch die keine Masse fließt. Also sind insbesondere die Geschwindigkeiten entlang der

Grenzschicht stetig

$$\mathbf{u}_1 = \mathbf{u}_2 \quad \text{auf } \Gamma_f. \quad (2.11)$$

- *Dynamische Bedingung:* An der Grenzfläche zwischen zwei sich bewegenden Fluiden ohne Oberflächenspannung ist die Sprung-Randbedingung durch

$$(\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{n} = 0, \quad (\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{t} = 0, \quad \text{und} \quad (\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{s} = 0,$$

gegeben [119], wobei $\mathbf{T}_i := -p_i \mathbf{I} + \mu_i \mathbf{D}_i$ den Spannungstensor des jeweiligen Fluides bezeichnet. Diese beschreibt die Bilanzierung der viskosen Reibungskräfte entlang des freien Randes. Wenn die Oberflächenspannung mit einbezogen wird, so tritt auf der rechten Seite der Gleichungen ein zusätzlicher Quellterm auf,

$$(\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{n} = \sigma \kappa \mathbf{n}, \quad (2.12)$$

$$(\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{t} = \frac{\partial \sigma}{\partial \mathbf{t}}, \quad (2.13)$$

$$(\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{s} = \frac{\partial \sigma}{\partial \mathbf{s}}, \quad (2.14)$$

mit σ als Koeffizient der Oberflächenspannungskraft. Mit \mathbf{n} wird die äußere Oberflächennormale auf Γ_f bezeichnet, das heißt die äußere Normale auf $\partial\Omega_1$. Weiterhin ist κ die lokale Krümmung

$$\kappa = \frac{1}{R_t} + \frac{1}{R_s} \quad (2.15)$$

mit R_t und R_s als Hauptkrümmungsradien der Grenzfläche entlang der lokalen orthogonalen Koordinaten t und s . Abbildung 2.3 veranschaulicht die geometrischen Größen R_t und R_s , dabei sind \mathbf{n} , \mathbf{t} und \mathbf{s} die Einheitsvektoren des lokalen orthogonalen Koordinatensystem (n, t, s) . Gleichung (2.12) besagt, dass auf dem freien Rand die viskosen Spannungskräfte in Normalenrichtung proportional zur lokalen Krümmung sind.

Da wir in dieser Arbeit isotherme Reinstoffe simulieren, wird im Folgenden σ als eine Materialkonstante betrachtet. Deshalb reduzieren sich die Randbedingungen auf der Grenzfläche Γ_f zu

$$\mathbf{u}_1 = \mathbf{u}_2, \quad (2.16)$$

$$(\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{n} = \sigma \kappa \mathbf{n}, \quad (2.17)$$

$$(\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{t} = 0, \quad (2.18)$$

$$(\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{s} = 0, \quad (2.19)$$

wobei nun die tangentialen Oberflächenspannungskräfte (2.18) und (2.19) aufgrund der Homogenität der Oberflächenspannungskonstante und der Stetigkeit des Geschwindigkeitsfeldes (2.16) im Vergleich zu der normalen Oberflächenspannungskraft vernachlässigbar klein sind und deshalb in unserem Fall nicht weiter betrachtet werden.

Folglich vervollständigen wir die Navier–Stokes Gleichungen (2.10) in jedem der Teilgebiete Ω_i mit den entsprechenden Randbedingungen für die Oberflächenspannung (2.16) und (2.17). Somit ergibt sich das vollständige Modell für inkompressible Zweiphasenströmungen inklusive Oberflächenspannungskraft wie folgt

$$\begin{aligned}
 \rho_i \frac{D\mathbf{u}_i}{Dt} &= -\nabla p_i + \nabla \cdot (\mu_i \mathbf{D}_i) + \rho_i \mathbf{g} && \text{in } \Omega_i, \\
 \nabla \cdot \mathbf{u}_i &= 0 && \text{in } \Omega_i, \\
 \mathbf{u}_i|_{\Gamma} &= 0 && \text{in } [0, T], \\
 \mathbf{u}_i|_{t=0} &= u_{0_i} && \text{in } \Omega_i \\
 \mathbf{u}_1 &= \mathbf{u}_2 && \text{auf } \Gamma_f, \\
 (\mathbf{T}_1 - \mathbf{T}_2) \cdot \mathbf{n} &= \sigma \kappa \mathbf{n} && \text{auf } \Gamma_f,
 \end{aligned} \tag{2.20}$$

für $i \in \{1, 2\}$.

2.2.2 Einbindung der Oberflächenspannungskraft in die Impulsgleichungen

Das Ziel ist, die beiden ersten Gleichungen in (2.20) als ein System über ganz Ω zu formulieren. Dies geschieht mit Hilfe der freien Randbedingung aus (2.12), die sich implizit als singulärer Quellterm in den Impulsgleichungen darstellen läßt. Als Ausgangspunkt für die Herleitung wird hierbei die Integralform der ersten Gleichung von (2.10) verwendet:

$$\int_{\Omega_i} \rho_i \frac{D(\mathbf{u}_i)}{Dt} d\mathbf{x} = \int_{\partial\Omega_i} \mathbf{T}_i \cdot \mathbf{n} dF + \int_{\Omega_i} \rho_i \mathbf{g} d\mathbf{x}, \tag{2.21}$$

für $i \in \{1, 2\}$. Summation beider Impulsgleichungen aus (2.21) liefert

$$\begin{aligned}
 \rho_1 \int_{\Omega_1} \frac{D\mathbf{u}}{Dt} d\mathbf{x} + \rho_2 \int_{\Omega_2} \frac{D\mathbf{u}}{Dt} d\mathbf{x} = \\
 \int_{\partial\Omega} \mathbf{T} \cdot \mathbf{n} dF - \int_{\Gamma_f} [\mathbf{T}] \cdot \mathbf{n} dF + \rho_1 \int_{\Omega_1} \mathbf{g} d\mathbf{x} + \rho_2 \int_{\Omega_2} \mathbf{g} d\mathbf{x},
 \end{aligned} \tag{2.22}$$

wobei $\mathbf{T} := \mathbf{T}_1 \chi_{\Omega_1} + \mathbf{T}_2 \chi_{\Omega_2}$ definiert wird unter Einsatz der charakteristischen Funktionen χ_{Ω_1} und χ_{Ω_2} , die den Fluidgebieten Ω_1 und Ω_2 zugehörig sind. Desweiteren bezeichnet $[\mathbf{T}]$ den Sprung im Spannungstensor \mathbf{T} auf der Grenzfläche Γ_f , das heißt $[\mathbf{T}] = \mathbf{T}_1 - \mathbf{T}_2$. Mit (2.17) und dem Satz von Gauß erhalten wir

$$\int_{\partial\Omega} \mathbf{T} \cdot \mathbf{n} dF - \int_{\Gamma_f} [\mathbf{T}] \cdot \mathbf{n} dF = \int_{\Omega} \nabla \cdot \mathbf{T} d\mathbf{x} - \int_{\Gamma_f} \sigma \kappa \mathbf{n} dF. \tag{2.23}$$

Diese Umwandlung ist im distributionellen Sinne aufzufassen, weil der Spannungstensor, bedingt durch den Sprung in der Viskosität μ , über Γ_f unstetig ist.

Aufgrund von (2.16) ist das Geschwindigkeitsfeld $\mathbf{u} = \mathbf{u}_1 \chi_{\Omega_1} + \mathbf{u}_2 \chi_{\Omega_2}$ für viskose Fluide stetig auf Ω und es gilt somit die Beziehung

$$\rho_1 \int_{\Omega_1} \frac{D\mathbf{u}}{Dt} d\mathbf{x} + \rho_2 \int_{\Omega_2} \frac{D\mathbf{u}}{Dt} d\mathbf{x} = \int_{\Omega} \rho \frac{D\mathbf{u}}{Dt} d\mathbf{x} \quad \text{mit} \quad \rho := \rho = \rho_1 \chi_{\Omega_1} + \rho_2 \chi_{\Omega_2}. \tag{2.24}$$

Durch Substitution von (2.23) und (2.24) in (2.22) erhalten wir die Integralformulierung der Impulsgleichung für inkompressible Zweiphasen-Strömungen mit Oberflächenspannung

$$\int_{\Omega} \rho \frac{D\mathbf{u}}{Dt} d\mathbf{x} = \int_{\Omega} \nabla \cdot \mathbf{T} d\mathbf{x} - \int_{\Gamma_f} \sigma \kappa \mathbf{n} dF + \int_{\Omega} \rho \mathbf{g} d\mathbf{x}. \quad (2.25)$$

Der Vorteil von Gleichung (2.25) liegt darin, dass sie die Randbedingung der freien Oberfläche implizit enthält. Allerdings ist diese immer noch als freies Randintegral enthalten. Für die Diskretisierung ist es von Vorteil, wenn dieses Randintegral mittels der *Continuum Surface Force Methode (CSF)* [23, 191] in ein Volumenintegral umgewandelt wird. Das lässt sich später dahingehend ausnutzen, dass ohne explizite Rekonstruktion der freien Oberfläche die freien Randbedingungen implizit bei der Diskretisierung der Impulsgleichungen erfüllt werden, vergleiche hierzu [61, 158]. Man beachte, dass dies im Gegensatz zu den Methoden, die eine explizite Rekonstruktion der Oberfläche erfordern, eine substantiell effizientere Lösungstechnik ist, die darüber hinaus numerisch stabiler ist und sich gut parallelisieren lässt.

2.2.3 Oberflächenspannung als Volumenintegral via der CSF-Methode

Das *Continuum-Surface-Force (CSF)*-Modell wurde erstmalig von Brackbill et al. [23] für VOF-Methoden und von Unverdi et al. [191] für MAC-Methoden präsentiert. Im Folgenden geben wir eine kurze Beschreibung der CSF-Approximation im Rahmen der Level-Set-Technik, nähere Details findet man in [33, 185]. Die Koppelung mit der Level-Set-Methode hat im Gegensatz zu der Volume-of-Fluid-Methode den Vorteil, dass sich aufgrund der Stetigkeit der Level-Set Funktion sowohl der Normalenvektor als auch die Krümmung genauer und kostengünstiger berechnen lassen.

Das Hauptanliegen des CSF-Modells besteht darin, das freie Randintegral aus Gleichung (2.25)

$$\int_{\Gamma_f} \sigma \kappa \mathbf{n} dF$$

in ein äquivalentes Volumenintegral über Ω umzuwandeln, um damit eine vereinheitlichte Kopplung mit der Impulsgleichung zu erhalten. Desweiteren sind die Viskosität und die Dichte nicht mehr global konstant, sondern sie weisen nun jeweils einen Sprung entlang der Phasengrenze auf. Deshalb wird die Dichte- und die Viskositätsverteilung in Abhängigkeit einer skalaren Markerfunktion ϕ als

$$\begin{aligned} \rho(\phi) &= \rho_1 + (\rho_2 - \rho_1)H(\phi) \\ \mu(\phi) &= \mu_1 + (\mu_2 - \mu_1)H(\phi) \end{aligned}$$

beschrieben, wobei $H(\phi)$ die Heaviside'sche Sprungfunktion ist. Sie ist definiert als

$$H(\phi) := \begin{cases} 0 & \text{falls } \phi < 0 \\ \frac{1}{2} & \text{falls } \phi = 0 \\ 1 & \text{falls } \phi > 0 \end{cases} .$$

Typisch für Level-Set-Techniken ist, dass als skalare Markerfunktion ϕ eine vorzeichenbehaftete Abstandsfunktion gewählt wird, die in einer Umgebung der physikalisch relevanten freien Oberfläche $\Gamma_f(t) := \{\mathbf{x} \in \Omega \mid \phi(\mathbf{x}, t) = 0\}$ den euklidischen Abstand zur freien Oberfläche und über das Vorzeichen die entsprechende Phasenzugehörigkeit angibt. Mit den ϕ -abhängigen Formulierungen kann die Impulsgleichung (2.25) umgeschrieben werden als

$$\int_{\Omega} \rho(\phi) \frac{D\mathbf{u}}{Dt} d\mathbf{x} = \int_{\Omega} \nabla \cdot \mathbf{T} d\mathbf{x} - \int_{\Gamma_f} \sigma \kappa \mathbf{n} dF + \int_{\Omega} \rho(\phi) \mathbf{g} d\mathbf{x}. \quad (2.26)$$

Das Randintegral läßt sich nun mit Hilfe der Dirac δ -Funktion [60, 113] in ein Volumenintegral umwandeln. Dabei ist die δ -Funktion auf der freien Oberfläche Γ_f konzentriert.

Satz 2.1. *Es gilt*

$$\int_{\Gamma_f} \sigma \kappa \mathbf{n} dF = \int_{\Omega} \sigma \kappa \delta(\phi(\mathbf{x}, t)) \mathbf{n} d\mathbf{x} \quad (2.27)$$

wobei $\delta : \mathbb{R} \rightarrow \overline{\mathbb{R}}$, $\phi \mapsto \delta(\phi)$ die eindimensionale Dirac δ -Funktion ist mit den Eigenschaften $\delta(0) = \infty$ und $\int_{\mathbb{R}} \delta dx = 1$.

Beweis:

siehe Brackbill, Kothe, Zemach [23]. \square

Wenn nun (2.27) in (2.26) eingesetzt wird, ergibt sich

$$\int_{\Omega} \left(\rho(\phi) \frac{D\mathbf{u}}{Dt} - \nabla \cdot \mathbf{T} + \sigma \kappa \delta(\phi) \mathbf{n} - \rho(\phi) \mathbf{g} \right) d\mathbf{x} = 0.$$

Da Ω beliebig gewählt werden kann, lautet die zugehörige differentielle Form

$$\rho(\phi) \frac{D\mathbf{u}}{Dt} - \nabla \cdot \mathbf{T} + \sigma \kappa \delta(\phi) \mathbf{n} - \rho(\phi) \mathbf{g} = 0,$$

oder etwas umformuliert

$$\rho(\phi) \frac{D\mathbf{u}}{Dt} + \nabla p = \nabla \cdot (\mu(\phi) \mathbf{D}) - \sigma \kappa \delta(\phi) \mathbf{n} + \rho(\phi) \mathbf{g}. \quad (2.28)$$

Man beachte, dass sich aufgrund der Stetigkeit der Level-Set-Funktion ϕ der Einheitsnormalenvektor und die Krümmung der freien Oberfläche einfach mittels Gradienten- beziehungsweise Divergenzbildung der Level-Set-Funktion berechnen lassen, es gilt also insbesondere

$$\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|} \quad \text{und} \quad \kappa = \nabla \cdot \mathbf{n}.$$

2.2.4 Dimensionslose Formulierung

Oft ist es nützlich, Gleichung (2.28) in dimensionslosen Größen zu formulieren, um aus ähnlichen Modellgrößen (Modellgrößen, die sich nur um einen Faktor unterscheiden) dynamisch ähnliche Strömungen zu berechnen. Dazu werden aus den dimensionsbehafteten Größen die zugehörigen dimensionslosen Größen durch

$$\mathbf{x}' = \frac{\mathbf{x}}{L}, \quad \mathbf{u}' = \frac{\mathbf{u}}{U}, \quad t' = \frac{U}{L}t, \quad p' = \frac{p}{\rho_1 U^2}, \quad \rho' = \frac{\rho}{\rho_1}, \quad \mu' = \frac{\mu}{\mu_1}, \quad (2.29)$$

gebildet, wobei L und U konstante skalare Vergleichsgrößen sind. Durch Einsetzen dieser dimensionslosen Größen in Gleichung (2.28) ergibt sich die dimensionslose Formulierung der Navier-Stokes-Gleichungen

$$\mathbf{u}'_{t'} + \mathbf{u}' \cdot \nabla' \mathbf{u}' + \frac{\nabla' p'}{\rho'(\phi)} = \frac{1}{\rho'(\phi)} \left(\frac{1}{Re} \nabla' \cdot (\mu'(\phi) \mathbf{D}') - \frac{1}{We} \kappa(\phi) \delta(\phi) \nabla' \phi \right) + \frac{\mathbf{g}'}{Fr} \quad (2.30)$$

mit $\mathbf{D}' = \{\nabla' \mathbf{u}'\} + \{\nabla' \mathbf{u}'\}^T$ und $\mathbf{g}' = \frac{\mathbf{g}}{\|\mathbf{g}\|}$. Der Operator ∇' bezieht sich auf die Variable \mathbf{x}' . Die Dichte und die Viskosität werden berechnet als Konvexkombination der Verhältnisse zwischen beiden Phasen, also

$$\rho'(\phi) = \frac{\rho_2}{\rho_1} + \left(1 - \frac{\rho_2}{\rho_1}\right) H(\phi) \quad \text{und} \quad \mu'(\phi) = \frac{\mu_2}{\mu_1} + \left(1 - \frac{\mu_2}{\mu_1}\right) H(\phi).$$

Desweiteren enthält die Gleichung (2.30) folgende dimensionslose Parametergruppen, die die Eigenschaften der Strömung beschreiben

$$Re = \frac{\rho_1 L U}{\mu_1} \quad (\text{Reynoldszahl}) \quad (2.31)$$

$$Fr = \frac{U^2}{\|\mathbf{g}\|L} \quad (\text{Froudezahl}) \quad (2.32)$$

$$We = \frac{\rho_1 L U^2}{\sigma} \quad (\text{Weberzahl}). \quad (2.33)$$

Hierbei stehen die Reynoldszahl für das Verhältnis zwischen Trägheits- und Reibungskräften, die Froudezahl für das Verhältnis zwischen Trägheits- und Gravitationskräften und die Weberzahl für das Verhältnis zwischen Trägheits- und Grenzflächenkräften. Die Oberflächenspannung σ geht als Materialkonstante in der Weberzahl ein. Da (2.30) nur von diesen dimensionslosen Parametergruppen abhängt, verhalten sich Strömungen in ähnlichen Geometrien genau dann dynamisch ähnlich, wenn jeweils ihre Reynolds-, Froude- und Weberzahlen übereinstimmen.

Die Diskretisierung des Zweiphasenmodells in Kapitel 4 betrachtet nur die dimensionsbehafteten Navier-Stokes-Gleichungen (2.28). Um mühsame Umrechnungen zu sparen, sind aus praktischen Gründen jedoch beide Formen der Navier-Stokes-Gleichungen implementiert worden.

2.2.5 Randbedingungen an Gebietsrändern

Um das System von partiellen Differentialgleichungen zu schließen, sind Randbedingungen für die Geschwindigkeiten an den Gebietsrändern notwendig. Im Folgenden bezeichne \mathbf{n} den

Einheitsnormalenvektor, der vom festen Rand ins Fluidgebiet zeigt, und \mathbf{t} und \mathbf{s} seien zwei linear unabhängige Tangentialvektoren. Üblicherweise wird eine Variable am Rand entweder auf einen vorgegebenen Wert gesetzt (Dirichlet-Randbedingung), oder ihr Gradient wird in eine bestimmte Richtung festgesetzt (Neumann Randbedingung). Insgesamt lassen sich damit folgende physikalische Randeigenschaften modellieren:

1. *Haftbedingung (no-slip)*: Das Fluid haftet auf dem Rand $\Gamma = \partial\Omega$, also muß dort der Geschwindigkeitsvektor verschwinden: $\mathbf{u}|_{\Gamma} = 0$.
2. *Fixe Ein-/Ausströmbedingung (in-/outflow)*: Das Fluid strömt mit einer fest vorgegebenen Geschwindigkeit \mathbf{u}_o ein/aus, also gilt auf diesem Rand $\Gamma_{EA} \subset \Gamma$ für den Geschwindigkeitsvektor: $\mathbf{u}|_{\Gamma_{EA}} = \mathbf{u}_o$.
3. *Rutschbedingung (free-slip)*: Das Fluid gleitet ohne Reibungsverluste auf diesem Rand $\Gamma_R \subset \Gamma$, das heißt nur die tangentialen Geschwindigkeiten, die keine Änderung in Richtung des Normalenvektors haben, bleiben erhalten:

$$(\mathbf{u} \cdot \mathbf{n})|_{\Gamma_R} = 0 \quad , \quad (\partial_{\mathbf{n}}(\mathbf{u} \cdot \mathbf{t}))|_{\Gamma_R} = 0 \quad , \quad (\partial_{\mathbf{n}}(\mathbf{u} \cdot \mathbf{s}))|_{\Gamma_R} = 0 \quad .$$

4. *Ausströmbedingung (outflow)*: Die Fluidgeschwindigkeit ändert sich nicht in Normalenrichtung zum Ausflußrand, also gilt auf dem Ausflußrand $\Gamma_A \subset \Gamma$: $(\partial_{\mathbf{n}}\mathbf{u})|_{\Gamma_A} = 0$, wobei hier \mathbf{n} den äußeren Normalenvektor des Ausflußrandes bezeichnet.

Beim Setzen der Randbedingungen muß darauf geachtet werden, dass das Randintegral über die Geschwindigkeiten senkrecht zum festen Rand Null ist, denn aus der Kontinuitätsgleichung zusammen mit dem Satz von Gauß folgt:

$$0 = \int_{\Omega} \nabla \cdot \mathbf{u} \, d\mathbf{x} = \int_{\partial\Omega} \mathbf{u} \cdot \mathbf{n} \, dF \quad . \quad (2.34)$$

Eine Verletzung dieser Bedingung führt zu Fehlern in der Inkompressibilität und dadurch zu Problemen in der Masserhaltung. Insbesondere wenn Ausflußrandbedingungen gegeben sind, wird in der Regel die Kompatibilitätsbedingung (2.34) verletzt. Hierzu müssen die extrapolierten Geschwindigkeiten so korrigiert werden, dass der Massefluß am Ausströmrand gleich dem Massefluß am Einströmrand ist.

Die Randbedingungen für den Druck werden zusammen mit der in dieser Arbeit eingesetzten Projektionsmethode zur Diskretisierung der Navier-Stokes-Gleichungen in Kapitel 4 erklärt.

Kapitel 3

Diskretisierungstechniken für freie Oberflächen

Mehrphasenströmungen treten von jeher in vielen Fragestellungen der Ingenieurwissenschaften und der Mathematik auf. Aus diesem Grund sind in den letzten dreißig Jahren viele Modelle und zahlreiche numerische Methoden entwickelt worden, die Simulationen von Strömungen mit freien Oberflächen in unterschiedlichsten Einsatzbereichen ermöglichen.

Komplexe Strömungen mit einer Liquid-Gas-Grenzfläche wurden bereits in zahlreichen Veröffentlichungen berücksichtigt. Hierbei hat sich die Mehrheit der eingesetzten Diskretisierungstechniken für freie Oberflächen [13, 33, 35, 45, 86, 101, 128, 132, 182, 183, 185, 189, 192, 204] auf das Lösen von inkompressiblen Zweiphasenströmungen konzentriert. Kompressibilitätseffekte in Zweiphasenströmungen wurden in [8, 9, 114, 136, 137, 174, 197] betrachtet und Methoden, in denen ein Fluid als inkompressibel und das andere als kompressibel betrachtet wird, wurden in [27, 30, 57] untersucht.

Darüber hinaus treten freie Oberflächen auch in den unterschiedlichsten Bereichen der Physik und Ingenieurwissenschaften auf, wie beispielsweise in der Fluid-Struktur-Wechselwirkung [51, 71, 75, 170, 207], Blutströmungen in elastischen Arterien [31, 153], in Mehrphasenproblemen mit unmischbaren Fluiden [109, 127, 152], in der Gletscherdynamik [150], in viskoelastischen Fluidströmungen [19, 159, 206], in Formfüllungsverfahren [39, 47, 126], in der Blasen- und Tröpfchendynamik [45, 62, 128, 157, 162, 210], im Korn- und Kristallwachstum [78, 209] und im Schiff- und Wasserbau [17, 25, 144, 180]. Alle diese Anwendungen erfordern problemangepasste numerische Methoden für freie Oberflächen. Daher wurden zahlreiche numerische Techniken in den letzten drei Jahrzehnten entwickelt, die im Folgenden kurz beschrieben werden. Detailliertere Übersichten finden sich in [95, 115, 165].

Zusammengefasst können die numerischen Techniken für freie Oberflächen in zwei Klassen eingeteilt werden:

1. die Tracking-Methoden und **2. die Capturing-Methoden** ,

diesen kann dann wiederum eine feinere Unterteilung in Form von *Surface-Tracking-Methoden*, *Volume-Tracking-Techniken*, *Front-Capturing-Methoden* etc. zugeordnet werden, auf die in den folgenden Abschnitten näher eingegangen wird.

3.1 Tracking-Methoden

Tracking-Methoden sind von Natur aus Lagrangesche Techniken, die beispielsweise in [100] beschrieben werden. Die Position eines diskreten Punktes \mathbf{x}_i auf der freien Oberfläche wird für jeden Zeitpunkt durch Integration der Evolutionsgleichung

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{u}_i \quad (3.1)$$

ermittelt, wobei \mathbf{u}_i die Geschwindigkeit ist, mit der sich der diskrete Punkt \mathbf{x}_i fortbewegt. Zur Klasse der Tracking-Methoden gehören die *Moving-Mesh-* [55, 161], die *Front-Tracking-* [67, 189], die *Randintegral-* [151] und die *Partikel-* [15, 89, 132] Verfahren. Für *Moving-Mesh-*, *Front-Tracking-* und *Randintegral-*Verfahren entsprechen die Punkte \mathbf{x}_i den diskreten Punkten auf den Gitterlinien, welche die Oberfläche repräsentieren. Für partikelbasierte Methoden entsprechen die Punkte \mathbf{x}_i den individuellen Partikeln mit vorgegebenen physikalischen Eigenschaften. Diese sind dann entlang der freien Oberfläche verteilt.

Im Folgenden wird auf jedes dieser Tracking-Verfahren detaillierter eingegangen.

3.1.1 Moving-Mesh-Methoden

Moving-Mesh-Verfahren beschreiben die freie Oberfläche entlang wohldefinierter Zellränder. Die Idee dabei ist, das Gitter zusammen mit der Fluidgeschwindigkeit zu transportieren, so dass es im zeitlichen Verlauf an die freie Oberfläche angepasst bleibt. Lagrange- und Lagrange-Euler-Methoden zählen zu den bekanntesten Vertretern der Moving-Mesh-Klasse.

Es gibt verschiedene Lagrangesche Ansätze. Eine Möglichkeit ist, auf einem rechteckigen Gitter zu rechnen, welches sich während der zeitlichen Entwicklung verzerrt. Eine andere Möglichkeit ist, auf verallgemeinerten orthogonalen Gittern zu rechnen, die der Form der freien Oberfläche angepasst sind. Darüber hinaus gibt es auch Lagrangesche Methoden auf triangulierten Gittern.

Sie alle haben den gemeinsamen Vorteil, dass sich die freie Oberfläche zu jedem Zeitpunkt gut darstellen lässt, solange die Deformation des Gitters moderat bleibt. Wenn aber die Fluidströmung das Gitter extrem verformt, dann kann es zu Einbußen in der Konvergenzordnung bis hin zu numerischen Instabilitäten kommen.

Ein Weg, starke Gitterverzerrungen zu beheben, besteht darin, neue Gitter zu erzeugen. Dabei wird, wenn notwendig, ein weniger verformtes Gitter konstruiert, auf dem die physikalischen Größen durch Interpolation aus den alten Gitterwerten berechnet werden. Diese Interpolationen induzieren numerische Diffusion, wodurch ein künstliches Verschwimmen der freien Oberfläche erfolgt. Die Verwendung von Dreiecksgittern reduziert diese Diffusion auf Kosten einer komplizierteren Verwaltung der Gitterpunkte und eines erhöhten Speicherplatzbedarfs.

Ausschlaggebend für den Nachteil von rechteckigen Gittern ist die unflexible Verbindung unter den Gitterpunkten. Jeder Gitterpunkt innerhalb des Rechengebietes gehört zu genau vier Rechtecken (in zwei Raumdimensionen). Diese Einschränkung hat bei der Neukonstruktion des Gitters zur Folge, dass im Allgemeinen mehr Interpolationen notwendig werden als auf Dreiecksgittern, denn triangulierte Gebiete können jedem Gitterpunkt beliebig viele Dreiecke zuordnen. Diese höhere Flexibilität ermöglicht die Konstruktion eines

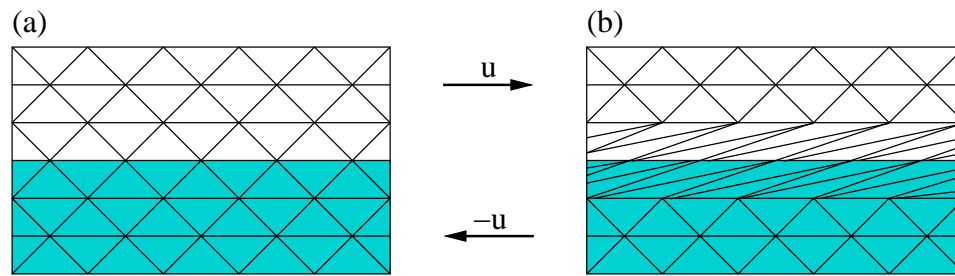


Abbildung 3.1: Scherströmung mit Geschwindigkeit u in der oberen Fluidhälfte (weiss) und $-u$ in der unteren Fluidhälfte (blau): (a): Anfangsbedingung der Zweiphasen-Scherströmung. (b): Gitterverzerrung durch die Fluidgeschwindigkeit u und $-u$. Neue Verknüpfung der Gitterpunkte ergibt wieder (a).

uniformeren Gitters mit weniger Interpolationen. Zum Beispiel zeigt Abbildung 3.1(b) in die Länge gestreckte dreieckige Gitterzellen an der Grenzschicht einer Scherströmung. Das Ersetzen durch symmetrischere Dreiecke benötigt im Idealfall keine zusätzlichen Gitterpunktinterpolationen, wie in Abbildung 3.1(a) zu sehen ist.

Weiterhin gibt es auch Mischformen zwischen Lagrangeschen und Eulerschen Methoden, sogenannte Lagrange-Euler-Verfahren. Dabei wird die Fluidströmung auf dem Eulerschen Gitter ausgewertet und die freie Oberfläche mit einem darübergelegten Lagrangeschen Gitter dargestellt. Diese Idee kombiniert die Vorteile eines rechteckigen Eulerschen Gitters, auf dem sich die Strömung leicht berechnen läßt, mit den Vorteilen eines Lagrangeschen Gitters, durch das die freie Oberfläche gut approximiert wird.

Die Berechnung eines Zeitschrittes verläuft dabei über mehrere Stationen. Zuerst wird das Lagrangesche Gitter, über das die freie Oberfläche dargestellt wird, mit der Fluidgeschwindigkeit des Eulerschen Gitters transportiert. Über dieses neue Lagrangesche Gitter kann nun jeder Punkt auf dem Eulerschen Gitter genau einer Phase zugeordnet werden. Dadurch lassen sich die Phaseneigenschaften, wie z.B. Viskosität, Dichte, etc., in jedem Punkt des Eulerschen Gitters bestimmen. Schließlich wird die Berechnung der neuen Strömungsdaten bezüglich der verschiedenen Phasen auf dem Eulerschen Gitter durchgeführt. Mit den neuen Geschwindigkeits- und Druckwerten werden dann wieder die aktuellen Lagrange-schen Positionen ermittelt. Diese dienen als Ausgangspunkt für den nächsten Zeitschritt.

Lagrange-Euler-Methoden haben den Nachteil, dass Informationen beider Gitter zu speichern sind und zusätzlich zwischen beiden Gittern interpoliert werden muß, wodurch wiederum numerische Diffusion induziert wird, die das Endresultat empfindlich stören kann.

3.1.2 Front-Tracking-Verfahren

Explizites *Front-Tracking* hat seine Ursprünge in den *Marker and Cell (MAC)* Verfahren [87] und dessen Erweiterungen durch Daly [49, 50]. Dabei wird die freie Oberfläche durch Lagrange-Marker-Partikel repräsentiert, die innerhalb eines stationären Eulerschen Gitters transportiert werden. Während sich die Oberfläche bewegt und deformiert, können Ober-

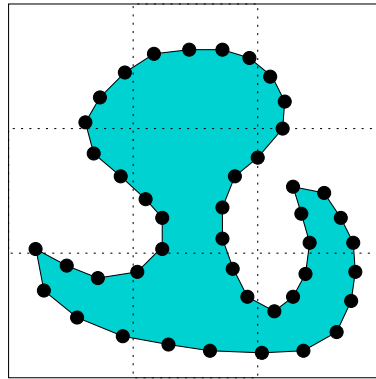


Abbildung 3.2: Einsatz von Lagrange-Marker-Partikel zur Oberflächendarstellung für die Front-Tracking-Methode.

flächenpunkte nach Bedarf in die Partikelliste hinzugefügt, gelöscht und wieder eingebunden werden. Die Oberfläche kann hierbei prinzipiell beliebig komplexe Topologieänderungen durchlaufen, vorausgesetzt der Algorithmus ist in der Lage topologische Entscheidungen zu treffen. Topologische Änderungen resultieren hierbei nicht aus einer Menge von lokal eingesetzten physikalischen Oberflächenmodellen, sondern algorithmischen Interventionen, die darüber entscheiden ob eine Oberfläche in zwei Teilstücke zerfällt oder umgekehrt zwei Oberflächen sich zu einer vereinen. Diese Algorithmen sind in der Regel nicht masseerhaltend, obwohl die Erhaltungseigenschaft für topologisch reguläre Oberflächen und mit hohen Partikeldichten zur Oberflächendarstellung sehr gut erfüllt wird. Eine Weiterentwicklung stellt zum Beispiel die *Surface-Marker-Technik* dar, welche im Jahr 1991 von S. Chen, D.B. Johnson und P.E. Raad [34, 35] entwickelt wurde. Sie haben dabei Einphasenprobleme mit freien Rändern in zwei Raumdimensionen betrachtet, also ein Fluid im Vakuum. Eine Erweiterung auf dreidimensionale Probleme wurde kürzlich in [12, 155] entwickelt. Weiterführende Details über Front-Tracking-Methoden kann in den Referenzen [175, 176, 188, 189] gefunden werden.

Der notwendige Informationsaustausch zwischen der Oberfläche und dem stationären Eulerischen Gitter wird mittels Ideen aus der *Immersed Interface Method (IMM)* [124, 148] umgesetzt. Mit dieser Technik wird die scharfe Oberfläche über eine glatte Distributionsfunktion approximiert, die wiederum dazu genutzt wird, die Quellterme, welche auf der Oberfläche definiert sind, auf die benachbarten Gitterpunkte zu verteilen. Die Oberfläche erhält damit eine endliche Breite der Größenordnung $O(h)$, um damit Stabilität und Glätte zu gewährleisten. Numerische Diffusionsartefakte treten hierbei nicht auf, da die Oberflächenbreite über alle Zeiten konstant bleibt. Das Volumen wird in Front-Tracking-Methoden nicht exakt erhalten, da es im Allgemeinen keine Oberfläche gibt, die aus den Lagrange-Marker-Partikeln resultiert und mit dem Volumen der entsprechenden Konturlinie der Distributionsfunktion C exakt übereinstimmt.

Front-Tracking-Verfahren wurden erfolgreich in einer Vielzahl von unterschiedlichsten Oberflächenströmungen eingesetzt, wie beispielsweise in der Gasdynamik [36, 68], in inkompressiblen Strömungen mit und ohne Phasenübergänge [107, 108, 190, 191] und in der

Mikrostrukturevolution von Erstarrungsprozessen [105, 106]. Desweiteren wurde in [191] gezeigt, dass Front-Tracking-Methoden auch komplexe topologische Änderungen in 3-D standhalten können.

3.1.3 Randintegral-Verfahren

Randintegralmethoden können freie Oberflächenströmungen mit einer hohen Approximationsordnung berechnen und somit sehr genaue Ergebnisse liefern, insbesondere im Rahmen von zweidimensionalen Strömungen mit ausreichend regulären Oberflächentopologien [65, 95, 151, 156]. In diesen Methoden wird die Oberfläche explizit transportiert, ähnlich wie in den Moving-Mesh-Methoden oder Front-Tracking-Techniken, aber die Lösung des Strömungsproblems auf dem gegebenen Strömungsgebiet wird vollständig aus Informationen seitens diskreter Punkte auf der freien Oberfläche gewonnen. Für inkompressible Strömungen wird die Oberfläche über ein Geschwindigkeitspotential charakterisiert, welches über eine Verteilung von Punktdipolen repräsentiert wird. Randintegral-Verfahren wurden erstmalig von Rosenhead [163] in den dreissiger Jahren verwendet um sich aufrollende Wirbelschichten zu studieren, aber es brauchte weitere dreissig Jahre bevor Weiterentwicklungen es ermöglichten allgemeingültigere freie Oberflächenprobleme zu betrachten [18]. Zahlreiche Verbesserungen und Erweiterungen der Randintegral Verfahren sind hieraus in den letzten zwei Dekaden entstanden, der Leser sei hier auf Yeung [205] und Hou [95] für eine Übersicht auf aktuellere Arbeiten verwiesen. Desweiteren zählen auch die *Vortex-Methoden* zur Klasse der Randintegral-Verfahren, die umfassend von Leonard [120, 121] beschrieben werden.

Die Hauptvorteile von Randintegral-Verfahren sind zum einen die Reduktion des Strömungsproblems um eine Dimension, da nur noch Größen auf der Oberfläche geführt werden, und zum anderen das Potential sehr genaue Lösungen zu liefern, sofern die freie Oberfläche topologisch regulär gestaltet ist. Nachteile der Randintegral-Methoden bestehen zum einen in den Schwierigkeiten diese Technik auf drei Dimensionen zu erweitern (obwohl es in [121] umgesetzt wurde), und zum anderen ist das Verfahren sehr sensitiv bezüglich numerischer Instabilitäten, da die zugrundeliegenden Probleme singular sind. Desweiteren besteht die Notwendigkeit von lokalen Eingriffen in der Oberfläche im Fall von topologischen Änderungen, ähnlich wie bei den Front-Tracking-Verfahren. Dennoch präsentieren randintegralbasierte Simulationen mit freien Oberflächen weiterhin beeindruckende Lösungen für eine Vielfalt von Strömungsanwendungen [184].

3.1.4 Partikel-Methoden

Partikel-basierte Methoden werden über den Gebrauch von diskreten Partikeln charakterisiert, die makroskopische Fluidparzellen repräsentieren [131]. Die Lagrangeschen Navier-Stokes-Gleichungen werden hierbei über kleine Umgebungen der jeweiligen Partikel integriert, in denen sie die physikalische Eigenschaften wie Masse, Impuls und Energie mit sich führen. Der Einsatz von Partikel-basierten Methoden zur Modellierung von freien Oberflächenströmungen ist attraktiv, da die schwer zu handhabenden nichtlinearen Advektions-terme in den Navier-Stokes-Gleichungen schlicht über die Partikelbewegung dargestellt wer-

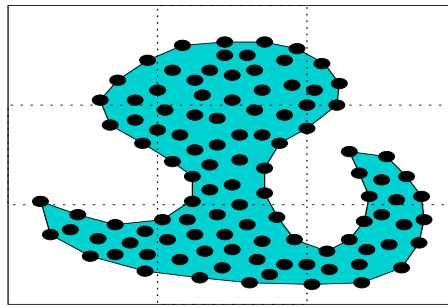


Abbildung 3.3: Volumenbeschreibung mit der *particle-in-cell* (PIC) Technik. Im Falle der *Smoothed-Particle-Hydrodynamics* (SPH) Methode würde das Eulersche Gitter im Hintergrund wegfallen, da SPH ein gitterloses Verfahren ist.

den. Darüber hinaus führt das Wissen über die Lage und Position eines jeden Partikels dazu, dass Materialoberflächen automatisch transportiert werden. Desweiteren führt der Gebrauch von Partikelbewegungen zur Approximation von Advektions-Termen dazu, dass die numerische Diffusion entlang der freien Oberflächen verschwindend gering ist. Partikel-basierte Methoden können hierbei in zwei Hauptklassen eingeteilt werden:

1. Partikel-Verfahren, die in Verbindung mit einem Gitter eingesetzt werden, sogenannte *Particle-In-Cell* (PIC) Techniken [89].
2. Partikel-Verfahren, die gitterlos arbeiten [16], wie beispielsweise die *Smoothed-Particle-Hydrodynamics* (SPH) Methode [66, 132]. Vertiefende Literatur für gitterlose Verfahren findet man unter anderem auch in [80–83].

F. Harlow und seine Mitarbeiter haben vor fünfzig Jahren die erste Partikel-basierte Methode erfunden und diese als *Particle-In-Cell* (PIC) Technik bezeichnet [88, 90]. Über die folgenden zwanzig Jahre ist das PIC-Verfahren intensiv eingesetzt worden und hat zahlreiche Weiterentwicklungen durchlaufen [89], sodass es *die* Methode zur Modellierung von stark verzerrten Oberflächenströmungen wurde. Mitte der achtziger Jahre hat dieses Verfahren aufgrund innovativer Entwicklungen und Verbesserungen [22, 24, 116] eine Wiederbelebung erfahren. Eine Stärke von PIC-Methoden besteht darin, dass Oberflächendetails, die feiner sind als die Schrittweite des Eulerschen Gitters, mit Hilfe der Partikel aufgelöst werden können. Schwierigkeiten treten auf, wenn Fluidfronten aufeinandertreffen oder sich abspalten. Solche topologischen Änderungen erfordern spezielle Erkennungsverfahren. Diese müssen angeben, wann zwei Partikellisten zu einer einzigen zusammengefügt werden oder eine Partikelliste in zwei getrennte aufgeteilt wird.

Im Vergleich zu Harlow's klassischer PIC-Methode veranlassen moderne *vollständige* PIC-Verfahren die Partikel dazu alle relevanten Fluid Informationen zu tragen anstatt nur dessen Position und Masse. Diese Formulierung wurde erstmalig in der SPH-Methode von J. Monaghan [131–133] und seinen Mitarbeitern vor dreissig Jahren eingesetzt und unterscheidet sich von den PIC-Verfahren darin, dass nicht ein begleitendes Gitter verwendet wird.

Ähnlich wie PIC-Methoden sind SPH-Verfahren besonders gut für kompressible Strömungen im Überschallbereich geeignet, wie sie beispielsweise in astrophysikalischen Anwendungen und in der Schock-Dynamik auftreten. Eine Schwäche von SPH-basierten Verfahren besteht im höheren Speicheraufwand und in der höheren Rechenzeit. Für die Simulation von inkompressiblen Strömungen können diese Verfahren erst seit Mitte der neunziger Jahre [133–135] eingesetzt werden, denn sie scheinen sehr anfällig hinsichtlich numerischer Instabilitäten [21] zu sein. Eine Stärke von Partikel-Methoden ist ihre einfache Umsetzung, denn in der Regel liefert die Umsetzung auf ein 3D-Problem keine weiteren Komplikationen im Vergleich zu einem 2D-Problem.

3.2 Capturing-Methoden

Capturing-Methoden sind von Natur aus Eulersche Techniken, die beispielsweise in [100] beschrieben werden. Bei Capturing-Techniken wird die freie Oberfläche nicht explizit transportiert, sondern über eine charakteristische Markerfunktion C eingefangen ("captured"), die beispielsweise die unstetige Heaviside-Funktion ergibt, wenn die Gittermaschenweite gegen Null läuft. Für den Zweiphasen-Fall kann C zum Beispiel durch

$$C = \begin{cases} C_1 & \text{für Fluid 1} \\ C_2 & \text{für Fluid 2} \\ > C_1, < C_2 & \text{auf der freien Oberfläche} \end{cases}$$

definiert werden, wobei $C_2 > C_1$ vorausgesetzt wird. Für eine endlich kleine Gittermaschenweite kann C nicht perfekt unstetig sein, deshalb hat die Region $C_1 < C < C_2$ eine endliche Breite von der Ordnung $O(h)$. Da ein Punkt auf der freien Oberfläche diese nicht verlassen darf, folgt die Evolutionsgleichung für C aus der Lagrangeschen Invarianz

$$\frac{DC}{Dt} = \frac{\partial C}{\partial t} + (\mathbf{u} \cdot \nabla)C = 0, \quad (3.2)$$

wobei \mathbf{u} die Transportgeschwindigkeit ist. Exakte Lagrangesche Informationen können in *Capturing*-Methoden nicht garantiert werden, denn diese sind nur in C eingebettet und nicht explizit vorhanden. Aus diesem Grund ist die exakte Lage der freien Oberfläche im Allgemeinen nicht bekannt. Dessen Lage ist definiert als die Übergangsregion in der $C_1 < C < C_2$ gilt. Sofern C bekannt ist, gestatten *Capturing*-Techniken den Einsatz ein und derselben Fluid-Modell-Gleichung auf dem gesamten Rechenggebiet Ω . Entfernt von der freien Oberfläche reduzieren sich die Fluid-Gleichungen auf die entsprechenden einphasigen Fluid-Gleichungen und innerhalb der Grenzschicht enthalten diese Gleichungen angepasste diskrete Deltafunktionen (in Abhängigkeit von C) für entsprechende Oberflächenterme [23, 124, 148, 191]. In *Capturing*-Methoden ist das Diskretisierungsgitter in der Regel fest vorgegeben und zeitlich invariant. Daher ist diesen Methoden eine topologische Robustheit inne, die eine Verschmelzung oder eine Trennung von Fluidteilgebieten ohne zusätzlichen Aufwand numerisch berechenbar machen. Andererseits ist eine kompakte Grenzflächendichte schwierig zu erhalten, da hier numerische Diffusionsartefakte bei der Diskretisierung des konvektiven Terms $(\mathbf{u} \cdot \nabla)C$ auftreten. Diese treten insbesondere in der Nähe der freien

Oberfläche auf, wo C stark variiert, und führen somit zu künstlich verschmierten Ergebnissen, welche dann beispielsweise die Volumen- oder Masseerhaltung verletzen. Das Ziel aktueller Forschung liegt im Beheben dieses Nachteils.

Zur Klasse der *Capturing*-Methoden gehören insbesondere die *Continuum-Advection*-, die *Volume-of-Fluid*-, die *Phasenfeld*- und die *Level-Set*-Verfahren, auf die nun im Folgenden näher eingegangen wird.

3.2.1 Continuum-Advection-Verfahren

Die *Continuum-Advection*-Methoden beruhen auf einer Diskretisierung hoher Ordnung der Kontinuums-Transportgleichung (3.2). Diese Idee ist sehr verführerisch, da ein solcher Algorithmus integraler Bestandteil eines jeden Strömungslösers ist und somit wenig Aufwand für die Umsetzung verspricht. Die *Continuum-Advection*-Methoden lassen sich auf die traditionellen Diskretisierungstechniken für hyperbolische Gleichungen zurückführen. Hierunter zählt Gleichung (3.2) als klassischer Standardfall, weshalb es zahllose Veröffentlichungen über Numerik hyperbolischer Erhaltungsgleichungen gibt, die sich auf (3.2) beziehen [59, 117, 123, 201, 208]. Ein einfaches aber mit starker numerischer Diffusion behaftetes Beispiel ist das Upwind-„Donor-Cell“-Schema [94] von erster Ordnung aus den achtziger Jahren. In den darauffolgenden Jahren wurden eine Reihe Schemata höherer Ordnung für hyperbolische Erhaltungsgleichungen entwickelt, wie beispielsweise das QUICK-Verfahren [122], die PPM-Technik [40], Universal-Limiter-Methoden [64, 195] und Adaptive-Stencil-Techniken [91, 173]. Solche Techniken werden zur Diskretisierung von Gleichung (3.2) verwendet, falls C die Energie, den Impuls, die Dichte oder einen passiven Skalar beschreibt.

Darüber hinaus wurden Continuum-Advection-Schemata in zahlreichen Freie-Oberflächen-Codes eingesetzt. Dabei ist es beim Oberflächentransport in der Regel die Aufgabe, zu einer unstetigen skalaren charakteristischen Funktion C eine Lösung der Transportgleichung (3.2) zu finden. Das ist mit herkömmlichen Verfahren meistens ein Problem, da diese eine gewisse Regularität von C voraussetzen. Die Quelle des Problems liegt in der algebraischen Behandlung des konvektiven Terms $(\mathbf{u} \cdot \nabla)C$. Selbst der Einsatz von höherer-Ordnung-Approximationen führt zu einem unphysikalischen Verschwinden der Grenzflächendichte, wie beispielsweise auch in [160] der Einsatz eines vierter-Ordnung PPM-Schemas zu einer immer noch unakzeptablen Aufweitung der freien Oberfläche führte. Der Grund liegt darin, dass die Unstetigkeiten in C nur als solche bestehen bleiben können, wenn die Lösung zu (3.2) aus einer *geometrischen* Approximation von $(\mathbf{u} \cdot \nabla)C$ herrührt. Continuum-Advection-Schemata approximieren diesen Term algebraisch, das heißt, sie bilden räumliche Ableitungen auch an den Unstetigkeitsstellen von C ohne diese im geometrischen Sinne gesondert zu behandeln. Weitere Studien in [160] haben ergeben, dass die Grenzflächenbreite im Laufe des Transportes bis auf 4-8 Zellen anwachsen kann, selbst mit Einsatz von Methoden höherer Ordnung [63]. Diese Beobachtungen sind nicht zufriedenstellend für eine Transport Methode mit freien Oberflächen und entsprechenden Sprüngen in der Dichte, der Viskosität, der Energie oder der passiven skalaren Funktion.

Eine interessante Idee [202, 203], mit der diese Oberflächendiffusionsproblematik in den Griff zu bekommen ist, besteht in der Transformation der unstetigen Funktion C in eine *glatte* Funktion \tilde{C} . Nun kann Gleichung (3.2) problemlos mit \tilde{C} gelöst werden und daraufhin

\tilde{C} wieder in die entsprechende unstetige Funktion C rücktransformiert werden. Diese Idee ist ebenfalls ein Hauptbestandteil in Level-Set-Methoden [141, 168].

3.2.2 Volume-Capturing-Methoden

Volume-Capturing-Techniken sind Eulersche Verfahren, die über eine Volumenanteilsfunktion C den Fluid-Füllstand in jeder Gitterzelle angeben. Es wird nur der Transport des Fluid-Volumens berechnet und dann aus ihm, wenn nötig, die freie Oberfläche rekonstruiert. Eine der populärsten Volume-Capturing-Techniken ist die Volume-of-Fluid (VOF) Methode.

Die Volume-of-Fluid-Technik wurde 1981 von Hirt und Nichols [94] entwickelt. Sie ermöglicht die Berechnung von sich ablösenden und aufeinandertreffen Fluidpartien in zwei und drei Raumdimensionen.

Im zweidimensionalen Fall wird die freie Oberfläche mit einer geschlossenen Kurve $\Gamma_f \subset \Omega \subset \mathbb{R}^2$ über einem Euler'schen Gitter identifiziert. Die Idee ist nun, eine sogenannte Volumenanteilsfunktion $C : \Omega \times [0, t_{end}] \mapsto [0, 1]$ einzuführen, die in jeder Gitterzelle den Anteil des Fluidvolumens innerhalb der freien Oberfläche angibt. Für eine Gitterzelle, die vollständig außerhalb von Γ_f liegt, gilt $C = 0$. Für eine Gitterzelle, die vollständig innerhalb Γ_f liegt, gilt $C = 1$. Für Gitterzellen, die von Γ_f durchlaufen werden, ist $C \in (0, 1)$, wobei der genaue Wert den Volumenanteil innerhalb von Γ_f am gesamten Zellvolumen repräsentiert. Alle nötigen geometrischen Daten, wie zum Beispiel der Normalenvektor und

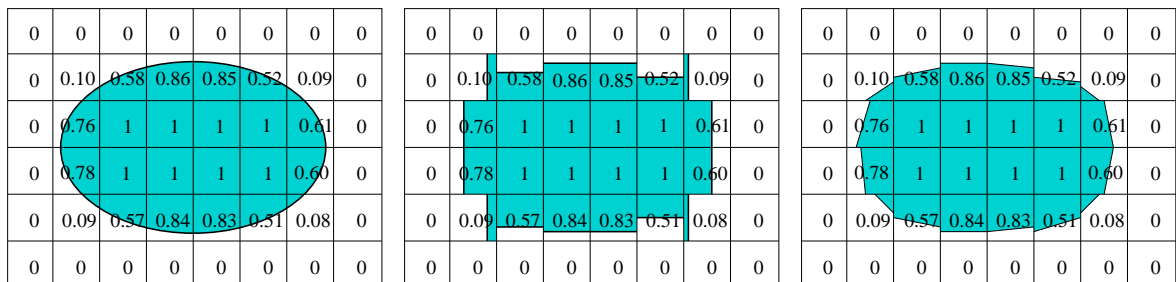


Abbildung 3.4: Exakte Oberfläche, SLIC und PLIC Rekonstruktion.

die Krümmung, werden ausschließlich aus der Volumenanteilsfunktion mit Hilfe verschiedenster Approximationstechniken ermittelt, die Gegenstand aktueller Forschung sind.

Der im Jahr 1976 von Noh und Woodward [140] entwickelte *Simple-Line-Interface-Calculation* (SLIC) Algorithmus gehört zu den ersten Oberflächenrekonstruktionsansätzen. Der freie Rand wird dabei durch horizontale oder vertikale Linien approximiert, die die Gitterzellen dem Volumenanteil nach schneiden, wie das mittlere Bild von Abbildung 3.4 darstellt. Die Orientierung der Linien wird so gewählt, dass die Kontaktfläche gleicher Fluidsorten aus benachbarten Zellen möglichst beibehalten wird. Dadurch wird die numerische Diffusion reduziert, weil eine Zelle sich zuerst vollständig mit einem Fluidtyp füllen muß, bevor sie diesen an andere Nachbarzellen weitergeben kann. Die Advektionsberechnung beruht auf einer Splitting-Methode. Dabei wird die Oberfläche zuerst rekonstruiert und dann in

x-Richtung transportiert. Mit diesen neuen Daten wird die Lage der Oberfläche erneut rekonstruiert und schließlich in y-Richtung transportiert. Dabei ist zu beachten, dass während der x- bzw. y-Advektionsberechnung die Liniensegmente jeweils unterschiedliche Oberflächen darstellen. Damit keine zu große (fehlerhafte) Oberflächenverformung stattfindet, ist es vorteilhaft, die Reihenfolge der Advektionsberechnung in x- und y-Richtung in jedem Zeitschritt zu wechseln. Eine Erweiterung auf drei Dimensionen ist leicht möglich.

Die grobe Annäherung der Oberfläche mit der SLIC-Methode wurde im Jahr 1981 von Hirt und Nichols [94] verbessert. Sie verwenden Linien mit einer beliebigen Steigung als Oberflächenapproximation, wie das letzte Bild von Abbildung 3.4 darstellt. Diese Idee wird heute als *Piecewise Linear Interface Construction (PLIC)* Technik bezeichnet. Sie ermöglicht neben einer besseren Oberflächenannäherung auch eine bessere Berechnung des Normalenvektors und der Krümmung. Dabei kann die Steigung einer Linie innerhalb einer Oberflächenzelle, also wenn $C \in (0, 1)$ gilt, im zweidimensionalen Fall durch einwertige Funktionen, nämlich $X(y)$ oder $Y(x)$ dargestellt werden. Als Approximationen zu $X(y)$ und $Y(x)$ werden

$$\begin{aligned} Y_i = Y(x_i) &= C_{i,j-1}\delta y_{j-1} + C_{i,j}\delta y_j + C_{i,j+1}\delta y_{j+1} \quad , \\ X_j = X(y_j) &= C_{i-1,j}\delta x_{i-1} + C_{i,j}\delta x_i + C_{i+1,j}\delta x_{i+1} \end{aligned}$$

verwendet. Anhand dieser Werte lassen sich Näherungen für die Ableitung der Volumenfunktionsfunktion in x-Richtung durch

$$\left[\frac{dY}{dx} \right]_i = \frac{\frac{Y_{i+1}+Y_i}{2} - \frac{Y_i+Y_{i-1}}{2}}{\frac{1}{2} \left(\frac{\delta x_{i+1}+\delta x_i}{2} + \frac{\delta x_i+\delta x_{i-1}}{2} \right)} = 2 \frac{Y_{i+1} - Y_{i-1}}{\delta x_{i+1} + 2\delta x_i + \delta x_{i-1}} \quad (3.3)$$

und analog die Ableitung in y-Richtung

$$\left[\frac{dX}{dy} \right]_j = 2 \frac{X_{j+1} - X_{j-1}}{\delta y_{j+1} + 2\delta y_j + \delta y_{j-1}} \quad (3.4)$$

berechnen. Falls nun $|\left[\frac{dY}{dx} \right]_i|$ kleiner als $|\left[\frac{dX}{dy} \right]_j|$ ist, dann liegt die Oberfläche mehr horizontal als vertikal, und die Ableitung in x-Richtung wird als Steigung der Linie verwendet. Umgekehrt wird für eine mehr horizontal verlaufende Oberfläche die Ableitung in y-Richtung als Steigung der Linie eingesetzt. Auf welcher Seite der approximierten Oberfläche sich das Fluid befindet, hängt davon ab, welche Richtungsableitung als Steigung ausgewählt wird. Falls $|\left[\frac{dY}{dx} \right]_i|$ verwendet wird, dann bestimmt das Vorzeichen von $\left[\frac{dX}{dy} \right]_j$, ob das Fluid oberhalb (positiv) oder unterhalb (negativ) des Randes liegt. Wenn $|\left[\frac{dX}{dy} \right]_j|$ eingesetzt wird, dann bestimmt das Vorzeichen $\left[\frac{dY}{dx} \right]_i$ ob das Fluid rechtsseitig (positiv) oder linksseitig (negativ) der Oberfläche liegt. Damit ist der Rand und die Lage des Fluides eindeutig bestimmt. Diese Rechnung wird für alle Gitterzellen mit $C_{i,j} \in (0, 1)$ durchgeführt.

Zusätzlich haben Hirt und Nichols ein verbessertes Advektionsschema entwickelt, welches als *Donor-Akzeptor-Verfahren* bekannt ist. Dort wird mit Kontrollfunktionen gesichert, dass nur soviel Volumen in eine Akzeptor-Gitterzelle wandern kann, wie sich in der Donor-Gitterzelle befindet. Allerdings ergeben sich dabei Rundungsfehler, bei denen die Volumen-

anteilsfunktion Werte außerhalb des Intervalls $[0, 1]$ annehmen kann. Eine Rücksetzung dieser Werte auf den nächstliegenden legalen Wert ist dabei in jedem Rechenzeitschritt erforderlich. Dies führt im Laufe der Rechnung zu einer empfindlichen Verletzung der Volumenerhaltung.

3.2.3 Phasenfeld-Methoden

In der letzten Dekade wurden Phasenfeld-Techniken intensiv für Kristallwachstums- und Hele-Shaw-Strömungssimulationen eingesetzt [28, 112, 198, 199], und desweiteren ist diese Methode auch erfolgreich für Navier-Stokes-Strömungen eingesetzt worden [11, 101]. Phasenfeld-Verfahren modellieren hierbei ähnlich wie andere Eulersche Techniken die Oberflächenspannungskräfte als Kontinuumskräfte mittels einer künstlichen Glättung der Oberflächenunstetigkeiten (Dichte, Viskosität) und Kräfte entlang einer dünnen, aber numerisch auflösbaren, Schicht. Diese Glättung erlaubt den Einsatz von klassischen numerischen Approximationsverfahren für die Oberflächenkinematik auf festen Gittern. Die Phasenfeld-Methode liefert hierbei ein kontinuierliches Oberflächenmodell [23, 191], welches energetisch und thermodynamisch konsistent ist [101]. Desweiteren wurde ein Phasenfeldmodell für Erstarrungsprozesse in Gegenwart von Schmelzkonvektionsphänomenen entwickelt [187]. All diese erfolgreichen Einsätze der Phasenfeld-Technik zeigen, dass es in der Tat ein mächtiges Werkzeug für die direkte numerische Simulation von Grenzschichtphänomenen darstellt.

In Fall von Strömungen mit freien Oberflächen ist der Ausgangspunkt die van-der-Waals-Hypothese, in der die Oberflächenenergiedichte abhängt von dem Phasenfeld ϕ und den Gradienten von ϕ . Cahn und Hilliard [29] erweiterten diese Hypothese auf dynamische Situationen indem sie die Flüsse bezüglich der Oberflächendiffusion als proportional zu den Gradienten der chemischen Potentiale approximierten. Jacqmin [101] erweiterte die Cahn-Hilliard-Gleichung um die strömungsdynamische Komponente. Gleichung (3.2) liefert Jacqmin's Evolutionsgleichung für ϕ (wobei hier $C = \phi$), bis auf den Unterschied, dass die rechte Seite nicht Null sondern der Laplace eines chemischen Potentials ist. Dieser neue Laplace-Term auf der rechten Seite kann wiederum diffusiven oder auch anti-diffusiven Charakter

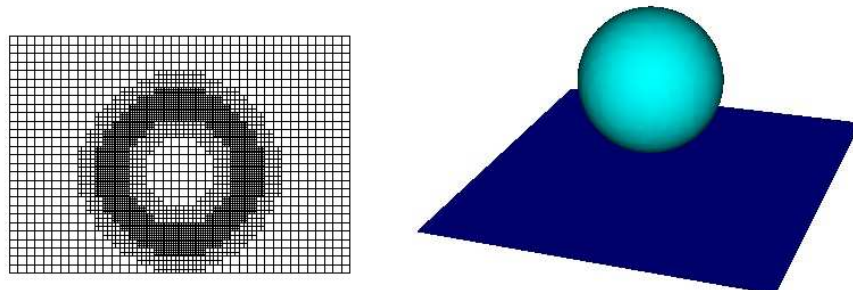


Abbildung 3.5: Adaptive Verfeinerung der Grenzfläche in Phasenfeldmethoden.

haben und damit die Grenzflächenbreite auf dynamische Weise gleichmäßig regularisieren. Anstatt speziell angepasste numerische Methoden zu entwickeln, die die Grenzflächenbreite gleichmässig halten, haben wir hier physikalische Mechanismen, die innerhalb der Phasenfeldmethode diese Funktion übernehmen. Deshalb können einfache Diskretisierungen mit zentralen Differenzen für den konvektiven Term $(\mathbf{u} \cdot \nabla)C$ eingesetzt werden [101], ohne die Notwendigkeit für den Einsatz von Upwind-Techniken.

Der (anti)-diffusive Term ist dennoch problematisch, denn es werden mindestens drei Gitterzellen für die Grenzschichtbreite benötigt, so dass der Laplace-Operator dort auch angemessen diskretisiert werden kann. Ansonsten neigt die freie Oberfläche dazu am Gitter zu haften [101]. Aktuelle Entwicklungen versuchen dieses Problem zu umgehen, indem sie in einer Umgebung der freien Oberfläche eine adaptive Verfeinerung der Gitter einsetzen, wie in Abbildung 3.5 dargestellt ist. Solche Methoden werden insbesondere für dreidimensionale Phasenfeld-Simulationen benötigt, um ausreichend genaue Ergebnisse zu erhalten.

3.2.4 Level-Set-Methoden

Level-Set-Verfahren gehören ebenfalls zur Klasse der Front-Capturing-Verfahren und sind artverwandt mit den Phasenfeldmethoden. Level-Set-Verfahren definieren die freie Oberfläche als eingebettete Niveaumenge einer höherdimensionalen Lipschitz-stetigen skalaren Funktion. Das Fortbewegen der Oberfläche reduziert sich auf den Transport dieser skalaren Funktion. Der freie Rand wird schließlich durch einen Konturplot der entsprechenden Niveaumenge visualisiert.

Im Rahmen dieser Arbeit ist die Level-Set-Technik verwendet worden, um inkompressible Zweiphasenströmungen zu simulieren. Aus diesem Grund wird diese Technik im Folgenden detaillierter beschrieben.

Level-Set-Darstellung als Anfangswertproblem

Die Level-Set-Approximation wurde im Jahr 1988 von Osher und Sethian [142, 168] eingeführt und wird seitdem erfolgreich für zahlreiche Probleme aus der Bildverarbeitung, der Materialkunde, den Verbrennungs- und Ätzprozessen, der Seismologie, der Robotik und der Fluidodynamik angewandt.

Zur Veranschaulichung wird eine geschlossene Kurve Γ_f in zwei Raumdimensionen betrachtet, die zwei Gebiete voneinander trennt. Das Ziel ist es, die Bewegung der Kurve zu verfolgen, die unter einer gegebenen Geschwindigkeitsfunktion F in Normalenrichtung fortschreitet. Einflüsse in tangentialer Richtung werden hierbei vernachlässigt, wie in Abbildung 3.6 dargestellt.

Es sei $\Omega \subset \mathbb{R}^2$ ein festes Gebiet mit einer zeitlich evolvierenden Kurve $\Gamma_f(t) \subset \Omega$ für alle $t \in [t_0, t_{end}]$. Die Idee besteht nun darin, eine möglichst glatte Funktion $\phi(\mathbf{x}, t)$ zu konstruieren, die auf Ω definiert ist und deren Niveaumenge zum Wert Null $\{\phi = 0\}$ gleich der Kurve $\Gamma_f(t)$ ist, also

$$\Gamma_f(t) = \{\mathbf{x} : \phi(\mathbf{x}, t) = 0\} \text{ für alle } t \in [t_0; t_{end}].$$

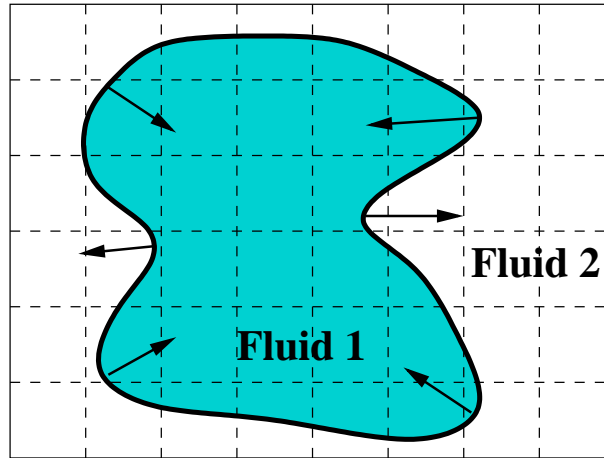


Abbildung 3.6: Oberflächenbewegung in Normalenrichtung.

Mit anderen Worten, der Level-Set-Wert eines Partikels $\mathbf{x}(t) \in \Gamma_f(t)$ ist immer gleich Null:

$$\phi(\mathbf{x}(t), t) = 0 \text{ für alle } t \in [t_0, t_{end}]. \quad (3.6)$$

Durch Differenzieren von (3.6) mittels der Kettenregel folgt

$$\phi_t + \nabla\phi(\mathbf{x}(t), t) \cdot \mathbf{x}'(t) = 0. \quad (3.7)$$

Im Fall der Navier-Stokes-Gleichungen ist $\mathbf{x}'(t)$ das Navier-Stokes-Geschwindigkeitsfeld \mathbf{u} eingeschränkt auf die freie Oberfläche, also $\mathbf{x}'(t) = \mathbf{u}|_{\Gamma_f}$. Übertragen auf das gesamte Gebiet Ω , erhält man schliesslich die globale Transportgleichung der Level-Set Funktion mit dem entsprechenden Geschwindigkeitsfeld \mathbf{u} der Navier-Stokes-Gleichungen

$$\begin{aligned} \phi_t + \mathbf{u} \cdot \nabla\phi &= 0 \\ \text{Anfangswert: } \phi(\mathbf{x}, t=0) &= \phi_0. \end{aligned} \quad (3.8)$$

Weil ausschließlich die Nullniveaumenge die physikalisch relevante freie Oberfläche darstellt, hat man viel Freiraum, die Level-Set-Funktion außerhalb ihrer Nullmenge zu bewegen. Dies kann dahingehend ausgenutzt werden, um möglichst glatte Level-Set-Funktionen während der numerischen Berechnung zu verwenden, wodurch die Stabilität und die Genauigkeit des Algorithmus verbessert werden. In dieser Arbeit soll die Level-Set-Funktion zusätzlich folgende Bedingungen erfüllen

$$\phi(\mathbf{x}, t) \begin{cases} > 0 & \text{falls } \mathbf{x} \in \{\text{Phase 1}\} \\ = 0 & \text{falls } \mathbf{x} \in \Gamma_f \\ < 0 & \text{falls } \mathbf{x} \in \{\text{Phase 2}\} \end{cases} \quad (3.9)$$

und $|\nabla\phi| = 1$ (Eikonalgleichung).

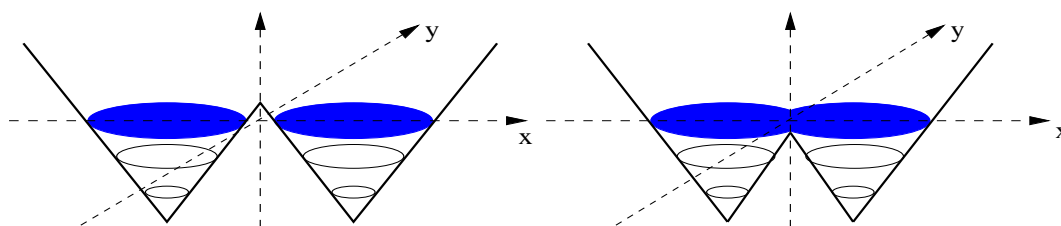


Abbildung 3.7: Level-Set-Darstellung: Getrennte (links) und vereinigte Oberflächen.

Damit wird sie zu einer vorzeichenbehafteten Lipschitz-stetigen Abstandsfunktion. Überall dort, wo zwei Ränder sich berühren oder schneiden, ist diese Level-Set-Funktion Lipschitz-stetig. Ansonsten ist sie in einer Umgebung des freien Randes immer glatt.

Die Transportgleichung (3.8) wird hierbei im Ort mit dem WENO-Schema fünfter Ordnung diskretisiert, das in Kapitel 3.4 näher beschrieben wird. Dadurch führen Ecken, Kanten oder Sprungstellen, die innerhalb der Level-Set-Funktion entstehen könnten, nicht zu numerischen Instabilitäten, was die Handhabung von topologischen Singularitäten wie zum Beispiel das Zusammentreffen und Ablösen von zwei Fluidpartien ermöglicht.

In Abbildung 3.7 ist der Graph der Level-Set-Funktion für zwei Raumdimensionen in Form einer breiten Linie inklusive den Niveaulinien als Kreise dargestellt. Die Level-Set-Funktion ist zu zwei unterschiedlichen Zeitpunkten dargestellt. Das linke Bild zeigt zwei getrennte Oberflächen, im rechten Bild sind die Oberflächen vereinigt. Der Rand der farbigen Flächen ist die Niveaumenge zum Wert Null der Level-Set-Funktion ist. Beachtenswert ist, dass Topologieveränderungen der freien Oberfläche nicht zu Singularitäten in der Level-Set-Funktion führen, diese bleibt Lipschitz-stetig.

Die Level-Set-Reinitialisierung

Im Allgemeinen ist eine zeitlich evolvierte Level-Set-Funktion keine Abstandsfunktion mehr. Wichtig für die Berechnung der Oberflächenspannungskräfte ist aber, dass die Level-Set-Funktion während der zeitlichen Entwicklung immer eine vorzeichenbehaftete Abstandsfunktion innerhalb einer ϵ -Umgebung des freien Randes Γ_f bleibt. Dies ist notwendig für eine stabile und konsistente Berechnung der Oberflächenspannung. Falls die Abstandseigenschaft der Level-Set-Funktion nicht aufrecht erhalten wird, entstehen mit der Zeit sehr große oder auch sehr kleine Level-Set-Gradienten an der freien Oberfläche (vgl. Abbildung 3.8), die zu verzerrten Oberflächenspannungskräften führen und unphysikalische Oszillationen erzeugen [19].

Es gibt verschiedene Methoden mit dem Ziel, die Abstandseigenschaft der Level-Set-Funktion wiederherzustellen, ohne dabei ihre Nullniveaumenge zu verändern. Die *Fast-Marching*- und *Narrow-Band-Techniken* von J.A. Sethian [168] beruhen auf einer expliziten Rekonstruktion der Niveaumenge zum Wert Null, von dem aus über eine Binärbaum-Datenstruktur genau einmal durch die Gitterzellen gewandert wird, in denen die zugehörigen Abstandswerte neu errechnet werden.

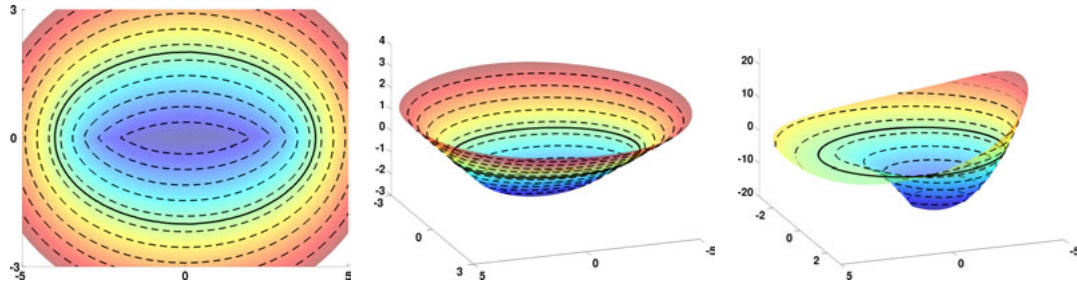


Abbildung 3.8: Level-Set-Funktion, einmal als 2D-Plot mit entsprechenden Niveaumengen, einmal als Graph der vorzeichenbehafteten Abstandsfunktion und einmal als Graph der verzerrten Funktion.

Basierend auf einer Beobachtung von J.-M. Morel wurde eine elegantere Weise zur Erhaltung der Abstandseigenschaft von Sussman, Smereka und Osher [185] eingeführt. Die Idee hierbei ist, iterativ in *künstlicher Zeit* τ eine stationäre Lösung der folgenden partiellen Differentialgleichung vom Hamilton-Jacobi-Typ

$$d_\tau + \text{sign}(\phi)(|\nabla d| - 1) = 0 \quad (3.10)$$

mit der Anfangsbedingung

$$d(\mathbf{x}, 0) = \phi(\mathbf{x})$$

zu berechnen, wobei die signum-Funktion definiert ist durch

$$\text{sign}(\phi) := \begin{cases} -1 & \text{falls } \phi < 0 \\ 0 & \text{falls } \phi = 0 \\ 1 & \text{falls } \phi > 0. \end{cases} \quad (3.11)$$

Dieser Ansatz benötigt keine explizite Oberflächenrekonstruktion. Die stationäre Lösung von Gleichung (3.10) ist eine vorzeichenbehaftete Abstandsfunktion, welche die Niveaumenge zum Wert Null von ϕ unverändert läßt, weil $\text{sign}(0)=0$. Wenn also die stationäre Lösung erreicht ist, wird einfach $\phi(\mathbf{x})$ durch $d(\mathbf{x}, \tau_{\text{stationär}})$ ersetzt. Desweiteren genügt es, dass die Level-Set-Funktion nur in einer ϵ -Umgebung des freien Randes eine Abstandsfunktion ist, denn bei der Diskretisierung des Zweiphasenmodells wird ein über der ϵ -Umgebung regularisiertes Deltafunktional und eine regularisierte Heaviside'sche Gewichtsfunktion eingesetzt. Um den Dichte- und Viskositätssprung und die Oberflächenspannungskräfte auf beiden Seiten der freien Oberfläche numerisch gleichmäßig aufzulösen, ist es hinreichend, wenn in dieser ϵ -Umgebung eine Abstandsfunktion gegeben ist. Somit ist der stationäre Zustand erreicht, wenn

$$|\nabla d| = 1 \quad \text{für} \quad |d| \leq \epsilon$$

gilt. Eine vorteilhafte Eigenschaft von Gleichung (3.10) besteht darin, dass die Level-Set-Funktion zuerst in der Nähe des freien Randes reinitialisiert wird. Dies ist leicht ersichtlich,

wenn (3.10) umgeschrieben wird zu

$$d_\tau + \vec{c} \cdot \nabla d = \text{sign}(\phi) \quad \text{mit} \quad \vec{c} = \text{sign}(\phi) \frac{\nabla d}{|\nabla d|}.$$

Diese Darstellung zeigt, dass es sich um eine nichtlineare Differentialgleichung handelt, dessen Charakteristiken von der Nullniveaumenge aus in Normalenrichtung mit der Geschwindigkeit ± 1 verlaufen. Dies bedeutet, dass d zuerst in der Nähe der freien Oberfläche reinitialisiert wird, so dass dann dort $|\nabla d| = 1$ gilt. Also ist es nicht nötig, Gleichung (3.10) auf dem ganzen Gebiet bis zum stationären Zustand zu lösen, womit eine erhebliche Anzahl an Iterationen eingespart werden kann. Beispielsweise sei $\delta\tau$ die Zeitschrittweite für den Iterationsprozeß und 2ϵ der Glättungsbereich des freien Randes, dann reicht es den Iterationsprozeß (3.10) für $\tau \leq \epsilon$ zu lösen. In der Praxis hat sich herausgestellt, dass meistens nicht mehr als drei bis fünf Iterationen notwendig sind.

Wir fassen die wesentlichen Vorteile der Level-Set-Technik zusammen:

1. Während andere Verfahren, die auf einer Parametrisierung der Oberfläche beruhen, bei topologischen Änderungen mehrwertig oder singular werden, behält diese Methode immer die Wohldefiniertheit der Level-Set-Funktion.
2. Die Level-Set-Methode ist ein vollständig Euler'sches Verfahren, deshalb können Finite-Differenzen-Verfahren auf einem festen Gitter angewendet werden, um die anfallenden Gleichungen in Raum und Zeit zu diskretisieren.
3. Die Berechnung des Normalenvektors und der Krümmung gehen auf natürliche Weise aus der Level-Set-Funktion hervor. Dabei ist für zwei Raumdimensionen die mittlere Krümmung κ der Level-Set-Kurve Γ_f in einem Punkt $(\vec{x}, t) \in \Gamma_f$ definiert als Divergenz des Normalenvektors in (\vec{x}, t) , also

$$\kappa = \nabla \cdot \mathbf{n}(\vec{x}, t) = \nabla \cdot \left(\frac{\nabla \phi(\vec{x}, t)}{|\nabla \phi(\vec{x}, t)|} \right) = - \frac{\phi_y^2 \phi_{xx} - 2\phi_x \phi_y \phi_{xy} + \phi_x^2 \phi_{yy}}{(\phi_x^2 + \phi_y^2)^{\frac{3}{2}}}.$$

Sie wird im Allgemeinen mit zentralen Differenzen diskretisiert.

4. Weiterhin ermöglicht die einfache Datenstruktur des Eulerschen Gitters eine Parallelisierung des Algorithmus, sodass insbesondere Probleme in drei Raumdimensionen effizient berechnet werden können.

Ein Nachteil von Level-Set-Methoden liegt in der Qualität der Volumenerhaltung. Da Level-Set-Verfahren Volumengrößen nicht explizit mitführen und aufgrund des Transportes und der Reinitialisierung auf einem kartesischen Gitter können numerische Diffusionsartefakte bei der Diskretisierung der konvektiven Terme auftreten. Dies kann insbesondere bei Langzeitsimulationen die Masseerhaltung empfindlich stören. Aus diesem Grund sind in den letzten Jahren zahlreiche Hybrid-Techniken entwickelt worden, mit denen die Vorteile der unterschiedlichen Diskretisierungsmethoden vereint werden sollen. Hinsichtlich der Masseerhaltung sind hierbei Verbesserungen erreicht worden, aber leider wird durch die Level-Set-VOF- und Partikel-Level-Set-Verfahren die Approximationsgüte der Krümmung stark verschlechtert, da die Hybrid-Methoden gleichzeitig auch die Regularität der freien Oberfläche zerstören

und damit die Auswertung der Krümmung mittels Berechnung zweiter Ableitungen schlecht konvergiert. Gerade aber die Krümmung spielt in unserem Fall eine wichtige Rolle. Deshalb haben wir eine andere Methode entwickelt mit der die lokale Regularität der freien Oberfläche erhalten bleibt und gleichzeitig die Masse erhalten wird. Auf die detaillierte Beschreibung und Diskretisierung unseres numerischen Verfahrens gehen wir im Folgenden Kapitel ein.

Kapitel 4

Diskretisierung des Zweiphasenmodells mit Oberflächenspannung

In diesem Kapitel wird die Diskretisierung des Zweiphasenmodells mit Oberflächenspannung beschrieben. Diese beruht im wesentlichen auf einer Erweiterung des Chorin-Temam-Projektionschemas [37, 186]. Im Folgenden werden insbesondere die Diskretisierung des Level-Set-Modells inklusive Transport und Reinitialisierung des freien Randes sowie die Diskretisierung der Navier-Stokes-Gleichungen beschrieben. Hierbei wird nach dem Reinitialisierungsprozess zusätzlich eine Fixpunktiteration durchgeführt mit der eine Masseerhaltung bis auf Maschinengenauigkeit ermöglicht wird. Diese ist insbesondere notwendig, um ein durch numerische Diffusion induziertes Verschieben der freien Oberfläche so zu korrigieren, dass die Massebilanz über alle Zeiten erhalten bleibt. Zusammen mit dem Einsatz eines WENO-Verfahrens fünfter Ordnung für die Berechnung sämtlicher Gradienten und einer verbesserten Approximation der Reinitialisierungsgleichung mit nachfolgender Picard-Fixpunktiteration wird in dieser Arbeit für die Masseerhaltung eine substantielle Verbesserung erzielt [45, 46].

Ein weiteres Problem ist der Dichtesprung zwischen den beiden Phasen, der in die Kondition der Systemmatrix bezüglich der Poissongleichung eingeht. Dies führt bei einem großen Dichtesprung und feinen Gittermaschenweiten zu einem schlechten Konvergenzverhalten. Daher wurde der Löser zusätzlich zu einem Jacobi-vorkonditionierten parallelen BiCGStab-Verfahren um einen AMG-Vorkonditionierer [130] erweitert. Dieser Löser konvergiert robust bezüglich des Dichtesprungs und mit optimaler Komplexität, das heißt die Konvergenzrate ist unabhängig von der Gittermaschenweite und der Rechenaufwand steigt mit linearem Aufwand bezüglich der Anzahl der Gitterpunkte.

4.1 Diskretisierungsgitter und Interpolationsregeln

Zur Diskretisierung des Gebietes Ω wird ein versetztes Gitter (staggered grid) verwendet, bei dem die Unbekannten für die Geschwindigkeiten, den Druck und die Level-Set-Funktion nicht in denselben Gitterpunkten, sondern versetzt liegen. Dieses diskrete Gitter Ω_h wird im Allgemeinen aus Stabilitätsgründen bevorzugt. Die Anordnung der verschiedenen Varia-

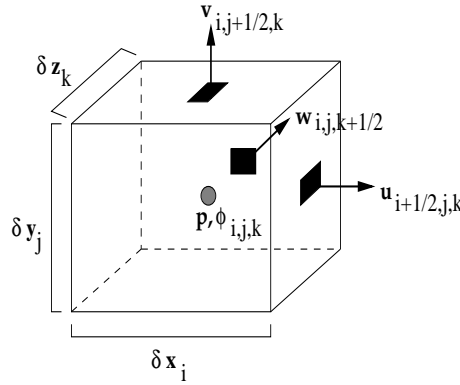


Abbildung 4.1: Positionierung der Geschwindigkeiten, des Druckes und der Level-Set Werte innerhalb einer Gitterzelle mit Index (i, j, k) .

blen an unterschiedlichen Positionen des kartesischen Tensorproduktgitters orientiert sich am Marker and Cell (MAC) Schema von Harlow und Welch [87]. Hierbei werden die vektoriellen Größen (Geschwindigkeiten (u, v, w)) in den Mittelpunkten der Zellseitenflächen und sämtliche skalare Größen (Druck p und die Level-Set-Werte ϕ) im Zentrum einer diskreten Gitterzelle positioniert, wie in Abbildung 4.1 dargestellt.

Für den Lösungsprozess benötigt man zusätzliche Interpolationen, da aufgrund des versetzten Gitters die Variablen jeweils um eine halbe Gittermaschenweite versetzt zueinander liegen. Hierzu ist in dieser Arbeit eine Lagrange-Polynominterpolation dritten Grades eingesetzt worden. Dabei wird für äquidistante Gitter zum Beispiel die Geschwindigkeitskomponente u im Zellmittelpunkt (i, j, k) berechnet über

$$\text{Mitte : } u_{i,j,k} = \frac{1}{16} \left(-u_{i-\frac{3}{2},j,k} + 9u_{i-\frac{1}{2},j,k} + 9u_{i+\frac{1}{2},j,k} - u_{i+\frac{3}{2},j,k} \right),$$

entsprechendes gilt für die Berechnung von $v_{i,j,k}$ und $w_{i,j,k}$, siehe Abbildung 4.2 oben. In Fluidzellen, die an Gebietsränder oder Hinderniszellen grenzen, wird nur auf Werte aus dem Fluidgebiet zugegriffen, sodass dort links- beziehungsweise rechtsseitig gewichtete Interpolationen entstehen:

$$\text{Links : } u_{i,j,k} = \frac{1}{16} \left(5u_{i-\frac{1}{2},j,k} + 15u_{i+\frac{1}{2},j,k} - 5u_{i+\frac{3}{2},j,k} + u_{i+\frac{5}{2},j,k} \right),$$

$$\text{Rechts : } u_{i,j,k} = \frac{1}{16} \left(u_{i-\frac{5}{2},j,k} - 5u_{i-\frac{3}{2},j,k} + 15u_{i-\frac{1}{2},j,k} + 5u_{i+\frac{1}{2},j,k} \right).$$

Analoges gilt für $v_{i,j,k}$ und $w_{i,j,k}$, siehe Abbildung 4.2 unten. Gegebenenfalls werden Level-Set-Werte an Mittelpunkten der Zellseitenflächen auf dieselbe Art ermittelt.

Die Bestimmung der Geschwindigkeitswerte an den Mittelpunkten der Zellseitenflächen, wo die jeweilige Komponente nicht lebt, verläuft über zweidimensionale Lagrange-Interpolation. Zum Beispiel wird v an der Stelle $(i + \frac{1}{2}, j, k)$ ermittelt über

$$v_{i+\frac{1}{2},j,k} = \frac{1}{32} \left(-v_{i-1,j+\frac{3}{2},k} - v_{i-1,j-\frac{3}{2},k} - v_{i+2,j+\frac{3}{2},k} - v_{i+2,j-\frac{3}{2},k} + 9v_{i,j+\frac{1}{2},k} + 9v_{i,j-\frac{1}{2},k} + 9v_{i+1,j+\frac{1}{2},k} + 9v_{i+1,j-\frac{1}{2},k} \right).$$

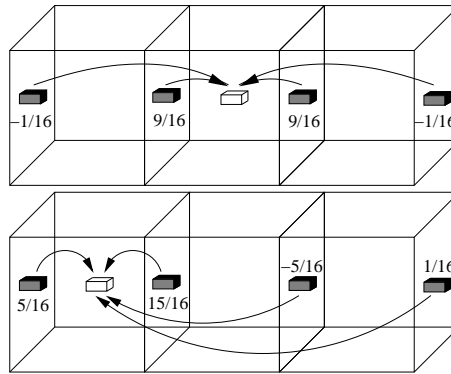


Abbildung 4.2: Zentrale (oben) und linksseitige (unten) eindimensionale Lagrange-Interpolation.

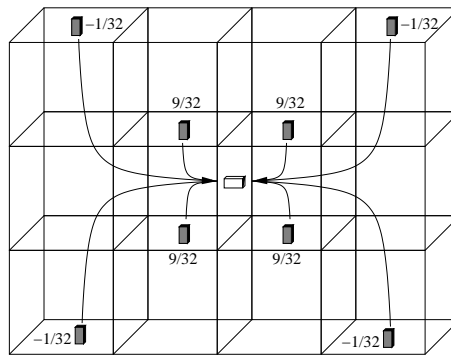


Abbildung 4.3: Zweidimensionale Lagrange-Interpolation.

In Abbildung 4.3 sind die Positionen und die Gewichte der v -Geschwindigkeitswerte eingezeichnet, die zur Interpolation notwendig sind. Auf analoge Art werden die u - und w -Geschwindigkeitskomponenten an anderen Zellseitenflächen bestimmt. Desweiteren wird bei der zweidimensionalen Lagrange-Polynominterpolation in Fluidzellen, die an Gebietsränder oder Hinderniszellen grenzen, auf bilineare Interpolation reduziert.

Man beachte, dass die Diskretisierungsgitter aller Variablen an den Gebietsrändern um entsprechende Randzellenschichten zu erweitern sind, sodass das genaue Setzen von Randbedingungen auf dem versetzten Gitter ermöglicht wird.

4.2 Diskretisierung der Level-Set Darstellung für freie Oberflächen

In diesem Abschnitt betrachten wir die Diskretisierung der freien Oberfläche mittels der Level-Set Technik inklusive des Transport- und des Reinitialisierungsprozesses, sowie des anschließenden Picard-Fixpunktiteration, welche eine Masseerhaltung bis auf Maschinen-

genauigkeit ermöglicht.

Ein Hauptproblem bei Zweiphasenströmungen liegt darin, dass sowohl die Dichte als auch die Viskosität im Allgemeinen einen Sprung an der Phasengrenze aufweisen. Hierbei führt eine scharfe Auswertung der Dichte- und Viskositätsprünge im Allgemeinen zu numerischen Instabilitäten. Um Zweiphasensysteme mit großen Viskositäts- und Dichtesprüngen inklusive Oberflächenspannung stabil berechnen zu können, müssen diese innerhalb einer geschickt gewählten ϵ -Umgebung der freien Oberfläche regularisiert werden. Sowohl die Wahl der ϵ -Umgebung als auch die Auswertung der Oberflächenspannung, der Materialparameter, des Normalenvektors und der Krümmung wird im Folgenden beschrieben.

4.2.1 Glättung, Oberflächenspannung, Normalenvektor und Krümmung

Gegeben sei im Folgenden eine diskrete vorzeichenbehaftete Abstandsfunktion ϕ_h mit den Eigenschaften aus (3.9). Die Materialparameter Dichte und Viskosität weisen einen Sprung entlang der freien Oberfläche auf. Diese Unstetigkeiten können einen ungünstigen Einfluß auf die Stabilität und die Approximationsgüte des globalen numerischen Schemas haben. Daher wird ein angemessenes Glättungsschema für die Materialparameter benötigt, sodass die Dichte und die Viskosität zumindest stetig sind entlang der freien Oberfläche. Dieses Glättungsschema sollte ausserdem die globale Konvergenzordnung erhalten. Weiterhin ist auch die Diskretisierung des Dirac-Delta-Funktional in der „Continuum Surface Force (CSF)“-Methode (siehe Abschnitt 2.2.3) entsprechend auf die Glättungsregion anzupassen.

Wie in Abbildung 4.4 dargestellt, wird zu diesem Zweck die freie Oberfläche mit einer Grenzflächenbreite $\epsilon(\mathbf{x})$ versehen, welche proportional zu der räumlichen Maschenweite h ist. Dann wird die diskrete Dichte $\rho(\phi_h)$ und die diskrete Viskosität $\mu(\phi_h)$ ersetzt durch

$$\rho^\epsilon(\phi_h) = \rho_2 + (\rho_1 - \rho_2)H^\epsilon(\phi_h) \quad \text{und} \quad \mu^\epsilon(\phi_h) = \mu_2 + (\mu_1 - \mu_2)H^\epsilon(\phi_h) \quad (4.1)$$

wobei

$$H^\epsilon(\phi_h(\mathbf{x})) := \begin{cases} 0 & \text{falls } \phi_h(\mathbf{x}) < -\epsilon(\mathbf{x}), \\ \frac{1}{2}(1 + \frac{\phi_h(\mathbf{x})}{\epsilon} + \frac{1}{\pi} \sin(\frac{\pi\phi_h(\mathbf{x})}{\epsilon})) & \text{falls } |\phi_h(\mathbf{x})| \leq \epsilon(\mathbf{x}), \\ 1 & \text{falls } \phi_h(\mathbf{x}) > \epsilon(\mathbf{x}) \end{cases} \quad (4.2)$$

die geglättete Heaviside-Funktion beschreibt. Das dazugehörige Deltafunktional ist über die Ableitung von (4.2) gegeben durch:

$$\delta^\epsilon(\phi_h(\mathbf{x})) := \partial_{\phi_h} H^\epsilon(\phi_h(\mathbf{x})) = \begin{cases} \frac{1}{2\epsilon(\mathbf{x})}(1 + \cos(\frac{\pi\phi_h(\mathbf{x})}{\epsilon})) & \text{falls } |\phi_h(\mathbf{x})| \leq \epsilon(\mathbf{x}) \\ 0 & \text{sonst.} \end{cases} \quad (4.3)$$

Entsprechend zu [56] wird eine Raum- und ϕ_h -abhängige Glättungsregion verwendet, um entsprechende Gittereffekte ausschließen zu können und somit die Konvergenzordnung des numerischen Verfahrens zu erhalten. Aus diesem Grund wird als Glättungsregion $|\phi_h(\mathbf{x})| \leq \epsilon(\mathbf{x})$ verwendet mit

$$\epsilon(\mathbf{x}) = \alpha(\mathbf{x})h \quad \text{und} \quad \alpha(\mathbf{x}) := c \frac{\|\nabla\phi_h(\mathbf{x})\|_1}{\|\nabla\phi_h(\mathbf{x})\|_2} < C, \quad (4.4)$$

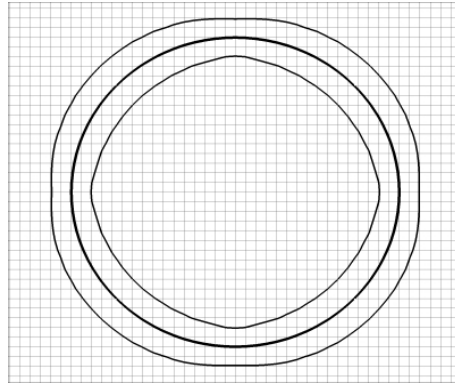


Abbildung 4.4: Beispiel einer numerischen Grenzflächenbreite $\epsilon(\mathbf{x})$ für $c = 2$. Die fettgedruckte mittlere Linie beschreibt den physikalisch relevanten freien Rand Γ_f .

wobei in unserern numerischen Experimenten $c = 1.5$ gesetzt wird. Abbildung 4.4 veranschaulicht die Raum- und ϕ_h -abhängige Glättungsregion für den Parameter $c = 2$.

Die fundamentale Voraussetzung für das obige Glättungsschema ist, dass der Gradient der Level-Set Funktion konstant in der $\epsilon(\mathbf{x})$ -Region ist, das heißt, dass die Konturen von ϕ_h in der $\epsilon(\mathbf{x})$ -Region parallel zu der Niveaumenge zum Niveaufwert Null verlaufen. Aufgrund der Wahl einer vorzeichenbehafteten Abstandsfunktion ist in unserer Implementation diese Voraussetzung automatisch erfüllt.

Man beachte, dass die geglättete Heaviside-Funktion später auch zur Approximation der Volumina $V_1(\phi^0)$ und $V_1(\phi^{n+1})$ in der Picard-Iteration (4.26) verwendet wird. Die entsprechend für die Masseerhaltung notwendigen Integrale der Form $\int_{\Omega} H^\epsilon(\phi_h) dx$ werden hierbei über eine Mittelpunkregel approximiert.

Der Einsatz des geglätteten Delta-Funktional aus (4.3) reduziert die Qualität der Approximation lokal innerhalb der Glättungsregion um eine Ordnung. Zu diesem Zweck betrachten wir eine Funktion g und deren Approximation g_h mit einem punktwweisen Approximationsfehler $g - g_h$ der Ordnung $O(h^p)$. Dann ist der geglättete Fehler $(g - g_h)\delta^\epsilon$ von der Ordnung $O(h^p)O(\epsilon(\mathbf{x})^{-1}) \approx O(h^{p-1})$ da $\epsilon(\mathbf{x}) \approx h$.

Schliesslich betrachten wir die Approximation der Oberflächenspannung für die Diskretisierung der Impulsgleichung (2.28). Da die Oberflächenspannung als singulärer Quellterm in die Impulsgleichung eingeht, wird sie mittels des Delta-Funktional und der CSF-Approximation in ein Volumenintegral umgeformt. Daher ist der Hauptbestandteil der Approximation der Oberflächenspannung der Einsatz eines geglätteten Delta-Funktional $\delta^\epsilon(\phi_i)$, welches in (4.3) mit dem aus (4.4) definierten ϵ gegeben ist. Dies führt zu einer volumetrischen Approximation der Oberflächenspannungskraft mit einem Träger in der ϵ -Umgebung der freien Oberfläche, das heißt

$$\sigma \kappa \delta(\phi) \mathbf{n} \approx \sigma \kappa_h \delta^\epsilon(\phi_h) \mathbf{n}_h.$$

Man erinnere sich, dass die diskrete Einheitsnormale \mathbf{n}_h auf der Oberfläche gegeben ist durch

$$\mathbf{n}_h := \frac{\nabla \phi_h}{|\nabla \phi_h|} = \frac{(\phi_x, \phi_y, \phi_z)_h^T}{\sqrt{(\phi_x^2 + \phi_y^2 + \phi_z^2)_h}},$$

und die diskrete Krümmung in drei Dimensionen wie folgt definiert ist durch

$$\kappa_h := \nabla \cdot \mathbf{n}_h. \quad (4.5)$$

Zu beachten ist, dass trotz des Einsatzes einer vorzeichenbehafteten Abstandsfunktion, also $|\nabla \phi_h| = 1$ in der ϵ -Umgebung der freien Oberfläche, der Normalenvektor \mathbf{n}_h statt durch $\nabla \phi_h$ durch $\frac{\nabla \phi_h}{|\nabla \phi_h|}$ approximiert wird. Auf ähnliche Weise wird die diskrete Krümmung κ_h anstatt durch $\Delta \phi_h$ durch $\nabla \cdot \mathbf{n}_h$ berechnet. Diese Prozedur führt zu einer stabileren Auswertung der Oberflächennormalen und der Krümmung im Fall von Topologieveränderungen der freien Oberfläche und die entsprechenden Differentialoperatoren werden über zentrale Differenzen approximiert.

4.2.2 Diskretisierung der Level-Set-Transportgleichung

Ein wesentlicher Punkt bei der Simulation von Zweiphasenströmungen ist die Berechnung der diskreten Dichte ρ_h und der diskreten Viskosität μ_h in Abhängigkeit von den zwei Fluidphasen. Anstatt diese Größen getrennt voneinander über die Transportgleichungen

$$\partial_t \mu_h + \mathbf{u}_h \cdot \nabla \mu_h = 0 \quad \text{und} \quad \partial_t \rho_h + \mathbf{u}_h \cdot \nabla \rho_h = 0, \quad (4.6)$$

zu entwickeln, werden in dieser Arbeit die beiden Gleichungen (4.6) auf den Transport der Level-Set-Funktion ϕ_h

$$\partial_t \phi_h + \mathbf{u}_h \cdot \nabla \phi_h = 0 \quad (4.7)$$

zurückgeführt, wobei sich anschliessend die Dichte ρ_h und die Viskosität μ_h zu jedem Zeitpunkt aus den Level-Set-Werten ϕ_h über Gleichung (4.1) berechnen lassen. Damit muss die Transportgleichung nur noch einmal pro Zeitschritt gelöst werden. Zusätzlich ist dieser Ansatz numerisch stabiler und enthält weniger numerische Diffusion, weil die Level-Set-Funktion glatter ist (Lipschitz-stetig) als das Dichte- und das Viskositätsfeld aus (4.6), die einen Sprung an der Phasengrenze haben. Entscheidend ist, dass jede Phase immer exakt (bis auf den fest vorgegebenen Glättungsbereich ϵ) ihren zugehörigen Viskositäts- und Dichtewert erhält. Dies ist bei einem getrennten Transport von Viskosität, Dichte und freier Oberfläche nicht garantiert.

Die zeitliche Entwicklung der freien Oberfläche und der Materialparameter erfolgt somit allein über den Transport der diskreten Level-Set-Funktion ϕ_h . Hierzu ist die Diskretisierung der Transportgleichung (4.7) notwendig, die im Folgenden beschrieben wird.

Raumdiskretisierung mit dem WENO-Verfahren fünfter Ordnung

Die Konvektionsgeschwindigkeiten $\mathbf{u} = (u, v, w)$ des konvektiven Termes $\mathbf{u} \cdot \nabla \phi$ der Transportgleichung (4.7) werden über Lagrange-Polynominterpolation dritten Grades (siehe Kapitel 4.1) in den Mittelpunkten der Gitterzellen bestimmt. Die einzelnen Komponenten von $\nabla \phi = (\phi_x, \phi_y, \phi_z)$ werden mit dem WENO-Verfahren fünfter Ordnung aus Kapitel 4.3.1 diskretisiert.

Zeitdiskretisierung mit dem Adams-Bashforth-Verfahren zweiter Ordnung

Die Zeitdiskretisierung erfordert ein Zeitgitter, das entweder im Sinne von finiten Differenzen als diskrete Punkte betrachtet oder als finite Volumen, also als Zeitvolumen angesehen werden kann. Nach der Diskretisierung des Raumoperators L wird die Zeit im Sinne eines Anfangswertproblems für gewöhnliche Differentialgleichungen diskretisiert. Für die Zeitdiskretisierung betrachten wir im Folgenden die gewöhnliche Differentialgleichung erster Ordnung mit gegebenem Anfangswert:

$$\frac{d\phi(t)}{dt} = L(\phi(t)), \quad \phi(t_0) = \phi^0. \quad (4.8)$$

Der Raumoperator L enthält die Raumableitungen der skalaren Funktion ϕ zum Zeitpunkt t . Gleichung (4.8) wird in drei Raumdimensionen jeweils für die x -, y - und z -Komponente angewendet.

Das Problem besteht darin, eine Lösung ϕ^1 zu finden, die $\phi(t_0 + \delta t)$ approximiert, also um einen Zeitschritt δt vom Anfangspunkt entfernt ist. Die Lösung ϕ^1 zum Zeitpunkt $t_1 = t_0 + \delta t$ kann als neue Anfangsbedingung angesehen werden und damit dann rekursiv um δt transportiert werden zu den Zeitpunkten $t_2 = t_1 + \delta t$, $t_3 = t_2 + \delta t$, etc..

Für den Transportschritt von t_n nach $t_{n+1} = t_n + \delta t$ betrachtet man häufig das Integral von (4.8) über das Intervall $[t_n, t_{n+1}]$

$$\int_{t_n}^{t_{n+1}} \frac{d\phi}{dt} dt = \phi^{n+1} - \phi^n = \int_{t_n}^{t_{n+1}} L(\phi(t)) dt, \quad (4.9)$$

mit $\phi^{n+1} = \phi(t_{n+1})$. Das Integral auf der rechten Seite von Gleichung (4.9) muss nun geeignet approximiert werden.

Eine Möglichkeit hierzu bietet die *Adams-Bashforth-Methode* [32]. Sie zählt zu den besten Mehrpunkt-Verfahren und ihre Idee ist, das Zeitintegral mittels Lagrange-Polynomextrapolation zu berechnen. Für ein explizites Adams-Bashforth-Verfahren der Ordnung $m + 1$ wird ein Lagrange-Polynom durch die Punkte $L(\phi^{n-m}), L(\phi^{n-m+1}), \dots, L(\phi^n)$ konstruiert welches dann mit der Simpson-Regel [166] exakt integriert wird. Das Adams-Bashforth-Verfahren zweiter Ordnung für *äquidistante* Zeitschritte lautet:

$$\phi^{n+1} = \phi^n + \frac{\delta t}{2} [3L(\phi^n) - L(\phi^{n-1})]. \quad (4.10)$$

Bis auf $L(\phi^n)$ ist also keine zusätzliche Berechnung des Raumoperators L notwendig. Allerdings müssen Werte $L(\phi^{n-k})$ aus alten Zeitpunkten gespeichert werden, um höhere Ordnungen zu erhalten. Der prinzipielle Nachteil der Adams-Bashforth-Methode besteht darin,

dass zu Beginn der Rechnung nur der Anfangswert gegeben ist, obwohl zur Erhaltung der Ordnung mehrere Werte benötigt werden. Ein Ausweg ist, am Anfang der Berechnungen ein Verfahren niedriger Ordnung (z.B. explizites Euler-Verfahren) mit kleinen Zeitschritten zu verwenden, sodass die gewünschte Genauigkeit erhalten bleibt. Sobald genügend Werte erhältlich sind, kann man die Ordnung entsprechend erhöhen und die Zeitschritte langsam vergrößern. Dies impliziert aber nicht-äquidistante Zeitschrittweiten, für die Gleichung (4.10) nicht gilt. In diesen Fällen sollte ein Adams-Bashforth-Verfahren zweiter Ordnung für *nicht äquidistante* Zeitschritte verwendet werden.

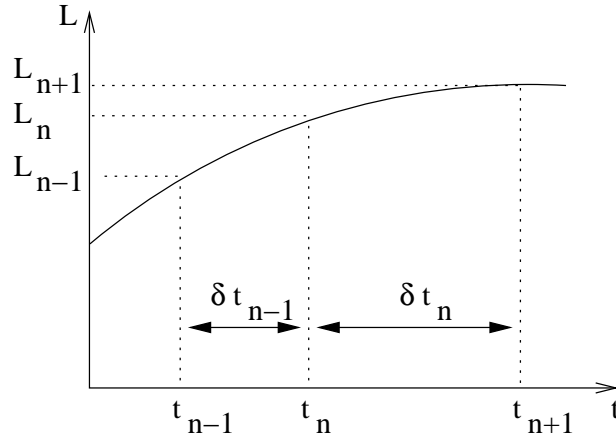


Abbildung 4.5: Zur nicht äquidistanten Zeitdiskretisierung.

Hierzu werden zwei gegebene Punkte L_{n-1} und L_n zu den Stützstellen t_{n-1} und t_n betrachtet, mit $L_i = L(\phi(t_i))$, $i = n-1, n$, vergleiche Abbildung 4.5. Zusätzlich sei $\delta t_{n-1} := t_n - t_{n-1}$ und $\delta t_n := t_{n+1} - t_n$. Wie zuvor ist das Ziel die Approximation des Integrals aus (4.9). Das Lagrange-Polynom ersten Grades durch die Punkte (t_{n-1}, L_{n-1}) und (t_n, L_n) hat die Form

$$p(t) = L_{n-1} \frac{t - t_n}{t_{n-1} - t_n} + L_n \frac{t - t_{n-1}}{t_n - t_{n-1}}. \quad (4.11)$$

Dieses wird nun mittels der Trapezregel exakt integriert und man erhält

$$\begin{aligned} \int_{t_n}^{t_{n+1}} p(t) dt &= \frac{\delta t_n}{2} (p(t_{n+1}) + p(t_n)) \\ &= \frac{\delta t_n}{2} \left(L_{n-1} \frac{t_{n+1} - t_n}{t_{n-1} - t_n} + L_n \frac{t_{n+1} - t_{n-1}}{t_n - t_{n-1}} + L_n \right) \\ &= \frac{\delta t_n}{2} \left(L_n \left(\frac{t_{n+1} - t_{n-1}}{t_n - t_{n-1}} + 1 \right) + L_{n-1} \left(\frac{t_{n+1} - t_n}{t_{n-1} - t_n} \right) \right) \\ &= \frac{\delta t_n}{2} \left(L_n \left(\frac{\delta t_n + 2\delta t_{n-1}}{\delta t_{n-1}} \right) - L_{n-1} \frac{\delta t_n}{\delta t_{n-1}} \right). \end{aligned}$$

Damit hat das verallgemeinerte Adams-Bashforth-Verfahren zweiter Ordnung die Form

$$\phi^{n+1} = \phi^n + \frac{\delta t_n}{2} \left(\frac{\delta t_n + 2\delta t_{n-1}}{\delta t_{n-1}} L_n - \frac{\delta t_n}{\delta t_{n-1}} L_{n-1} \right) \quad (4.12)$$

Im Fall äquidistanter Zeitschrittweiten (also $\delta t_n = \delta t_{n-1}$) ergibt sich aus (4.12) die Gleichung (4.10).

Das Adams-Bashfort-Verfahren zweiter Ordnung kann im Rahmen der Chorin-Temam-Projektionsmethode aus Kapitel 4.3 eingesetzt werden, wodurch sich die Strömungsgeschwindigkeiten der Navier-Stokes-Gleichungen in der Zeit von zweiter Ordnung genau berechnen lassen. Diese Idee ist erstmals von van Kahn in [194] beschrieben worden.

4.2.3 Diskretisierung der Level-Set-Reinitialisierung

Für die nach dem Transportschritt entstandene vorläufige Level-Set-Funktion ϕ^* (siehe Kapitel 4.3) wird die folgende Reinitialisierungsgleichung in Form eines Anfangswertproblems

$$d_\tau + \text{sign}(\phi^*) (|\nabla d| - 1) = 0 \quad (4.13)$$

Anfangswert: $d(0) = \phi^*(\mathbf{x}, t)$

in künstlicher Zeit τ iterativ gelöst, um wieder eine vorzeichenbehaftete Abstandsfunktion zu erhalten.

In [43, 45] durchgeführte numerische Testrechnungen haben gezeigt, dass die Diskretisierungsordnung in Zeit und Raum substantiell zur Erhaltung der Masse beiträgt und eine genauere Berechnung der Krümmung ermöglicht. Desweiteren können nach einem Transportschritt in Nähe der freien Oberfläche sehr große beziehungsweise sehr kleine Level-Set-Gradienten entstehen. Die Reinitialisierung wird dann in Bereichen mit flachem Gradienten sehr langsam. Nochschwieriger ist, dass in Bereichen mit sehr steilen Gradienten eine Verletzung der CFL-Bedingung eintreten kann, wodurch die Masseerhaltung empfindlich gestört wird.

Zu beiden Problemen werden im Folgenden Verbesserungen implementiert und mit numerischen Ergebnissen bezüglich der Masseerhaltung und schließlich auch bezüglich der Krümmung in Kapitel 5 validiert.

Zeitdiskretisierung mit dem Runge-Kutta-Verfahren dritter Ordnung

Für die im Rahmen der Reinitialisierung auftretende Zeitdiskretisierung verwenden wir *Runge-Kutta-Methoden* [32]. Das explizite Runge-Kutta-Verfahren zweiter Ordnung wird auch als klassische Heuns-Methode oder modifiziertes Euler-Verfahren bezeichnet. Es besteht aus zwei Teilschritten. Zunächst wird ein halber Zeitschritt als Prädiktor-Wert über einen expliziten Euler-Schritt berechnet, dann folgt mit der Mittelpunkregel ein Korrektor-Schritt, wodurch die Methode ihre zweite Ordnung erhält:

$$\begin{aligned} \text{Prädiktor: } \quad \phi^{n+\frac{1}{2}} &= \phi^n + \frac{\delta t}{2} L(\phi^n), \\ \text{Korrektor: } \quad \phi^{n+1} &= \phi^n + \delta t L(\phi^{n+\frac{1}{2}}). \end{aligned}$$

Der Vorteil dieses Verfahrens ist seine bessere Stabilität (TVD-Stabilität [74] im Vergleich zu den Adams-Bashforth-Methoden). Zusätzlich benötigen Runge-Kutta-Methoden bis auf die Anfangsbedingung keine weiteren Daten, um vom Beginn einer Rechnung an die Ordnung der Zeitdiskretisierung einzuhalten.

Das *Runge-Kutta-Verfahren dritter Ordnung* besteht aus einer Konvexkombination von drei Euler-Schritten.

$$\begin{aligned}\phi^{n+\frac{1}{3}} &= \phi^n + \delta t L(\phi^n) \\ \phi^{n+\frac{2}{3}} &= \phi^n + \frac{\delta t}{4} \left[L(\phi^n) + L(\phi^{n+\frac{1}{3}}) \right] \\ \phi^{n+1} &= \phi^n + \frac{\delta t}{6} \left[L(\phi^n) + 4L(\phi^{n+\frac{2}{3}}) + L(\phi^{n+\frac{1}{3}}) \right].\end{aligned}\quad (4.14)$$

In sämtlichen numerischen Experimenten aus Kapitel 5 wird dieses Runge-Kutta-Verfahren für die Level-Set Reinitialisierung eingesetzt. Dabei sind vor jeder Berechnung des Raumoperators L die Randbedingungen zu setzen.

Raumdiskretisierung mit dem WENO-Verfahren fünfter Ordnung

Der Raumoperator L bezüglich Gleichung (4.13) ist

$$\begin{aligned}L(d) &= \text{sign}(\phi^*) (|\nabla d| - 1) \\ &= \left(\frac{S(\phi^*) d_x}{\sqrt{d_x^2 + d_y^2 + d_z^2}} \right) d_x + \left(\frac{S(\phi^*) d_y}{\sqrt{d_x^2 + d_y^2 + d_z^2}} \right) d_y \\ &\quad + \left(\frac{S(\phi^*) d_z}{\sqrt{d_x^2 + d_y^2 + d_z^2}} \right) d_z - S(\phi^*),\end{aligned}\quad (4.15)$$

mit $S = \text{sign}$. Aus Gründen der numerischen Stabilität ist es nützlich die signum-Funktion $S(\phi^*)$ zu glätten.

Die Diskretisierung des Raumoperators L wirft zwei Fragen auf: Erstens, auf welche Weise soll ∇d berechnet werden? Zweitens, wie sollte die signum-Funktion regularisiert werden, ohne dass dabei der freie Rand künstlich bewegt wird?

Zur ersten Frage wurde in [43] der Einfluß der Diskretisierung von ∇d auf die Erhaltung der Masse untersucht. Hier hat sich gezeigt, dass das WENO-Verfahren fünfter Ordnung zu substantiell besserer Masseerhaltung führt als Verfahren von geringerer Approximationsordnung. Dabei wird für die Berechnung von $[d_x]_{i,j,k}$ ein Newton-Interpolationspolynom $P_{i-1,i}$ konstruiert, sodass d im Intervall $I = [x_{i-1}, x_i] \times \{y_j\} \times \{z_k\}$ von r -ter Ordnung approximiert, und ein weiteres Polynom $P_{i,i+1}$, das d im Intervall $I = [x_i, x_{i+1}] \times \{y_j\} \times \{z_k\}$ von r -ter Ordnung approximiert. Für $[d_x]_{i,j,k}$ stehen nun die zwei Werte

$$d_x^- = \left[\frac{dP_{i-1,i}}{dx} \right]_{i,j,k} \quad \text{und} \quad d_x^+ = \left[\frac{dP_{i,i+1}}{dx} \right]_{i,j,k}\quad (4.16)$$

zur Verfügung (analog für d_y und d_z). Welcher dieser beiden Werte ausgewählt wird, ermittelt

sich über folgende Upwindabfrage [181]:

$$[d_x]_{i,j,k} = \begin{cases} d_x^+ & \text{falls } d_x^- S_h(\phi_{i,j,k}) < -d_x^+ S_h(\phi_{i,j,k}) \text{ und } d_x^+ S_h(\phi_{i,j,k}) < 0 \\ d_x^- & \text{falls } d_x^- S_h(\phi_{i,j,k}) > 0 \text{ und } d_x^+ S_h(\phi_{i,j,k}) > -d_x^- S_h(\phi_{i,j,k}) \\ 0 & \text{sonst} \end{cases} .$$

Zur zweiten Frage wurde in [185] von den Autoren die folgende geglättete signum-Funktion

$$S_h(\phi) = \frac{\phi}{\sqrt{\phi^2 + h^2}} \quad (4.17)$$

verwendet. Diese Funktion führt zu einer guten Masseerhaltung, solange $\nabla\phi$ in der ϵ -Umgebung nicht zu groß oder zu klein ist. Im Rahmen von dynamischen Zweiphasenströmungen ist es aber der Regelfall, dass nach dem Transportschritt der Gradient der Level-Set Funktion $\nabla\phi$ insbesondere in der ϵ -Umgebung stark variieren kann. Dies führt dazu, dass bei einer expliziten Diskretisierung der Reinitialisierungsgleichung (4.13) eine Glättung der signum-Funktion mittels (4.17) die CFL-Bedingung verletzt. Diese Verletzung der CFL-Bedingung hat wiederum zu Folge, dass sich die freie Oberfläche mehr als eine Gittermaschenweite künstlich bewegt und somit den Masseerhalt substantiell stört. Aus diesem Grund ist es, neben einem WENO-Verfahren fünfter Ordnung für die Diskretisierung von $\nabla\phi$, essentiell, die signum-Funktion derart zu glätten, dass die CFL-Bedingung unabhängig von $\nabla\phi$ erfüllt ist. Darauf gehen wir im nun folgenden Abschnitt näher ein.

4.2.4 Verbesserte Masseerhaltung über eine CFL-erhaltende Glättung der signum Funktion

Wir betrachten zunächst die Auswirkung der herkömmlich geglätteten signum-Funktion aus (4.17) auf die CFL-Bedingung. Dazu wird in [146] zur Vereinfachung der *eindimensionale Fall* mit dem Upwind-Verfahren erster Ordnung im Raum und dem expliziten Euler-Verfahren in der Zeit betrachtet. Angenommen, die freie Oberfläche befindet sich im Intervall (x_i, x_{i+1}) , dann gilt $d_i < 0 < d_{i+1}$ für die Level-Set-Werte d_i und d_{i+1} in den Gitterpunkten x_i und x_{i+1} . Ein Reinitialisierungszeitschritt für die Gleichung (4.13) mit Upwind- und Euler-Verfahren führt zu

$$d_i^1 = d_i + s_i \delta\tau \left(1 - \frac{d_{i+1} - d_i}{\delta x} \right) \quad (4.18)$$

$$d_{i+1}^1 = d_{i+1} + s_{i+1} \delta\tau \left(1 - \frac{d_{i+1} - d_i}{\delta x} \right) , \quad (4.19)$$

wobei $-1 \leq s_i \leq 0$ und $0 \leq s_{i+1} \leq 1$ Approximationen zu $\text{sign}(d_i)$ und $\text{sign}(d_{i+1})$ aus (4.13) sind.

Das obige Schema ist monoton, das heißt d_i^1 und d_{i+1}^1 sind nicht-fallend in d_i und d_{i+1} , solange die CFL-Bedingung

$$\delta\tau \leq \delta x \quad (4.20)$$

erfüllt ist. Wenn $(d_{i+1} - d_i)/\delta x \leq 1$ ist, dann gilt $d_i^1 < 0 < d_{i+1}^1$, unabhängig von $\delta\tau$, s_i und s_{i+1} . Aber falls $(d_{i+1} - d_i)/\delta x > 1$ gilt und $\text{sign}(d)$ durch $S_h(d)$ aus (4.17) approximiert wird, dann kann nicht garantiert werden, dass d_i und d_{i+1} ihr Vorzeichen nicht wechseln. Falls zum Beispiel $\delta\tau = c(\delta x)$, $d_i = -m(\delta x)$ und $d_{i+1} = n(\delta x)$ mit $c, m, n \in \mathbb{R}^+$ gesetzt wird, dann ist

$$d_i^1 = m(\delta x) \left[-1 + \frac{c(m+n-1)}{\sqrt{m^2+1}} \right] \quad (4.21)$$

$$d_{i+1}^1 = n(\delta x) \left[1 - \frac{c(m+n-1)}{\sqrt{n^2+1}} \right]. \quad (4.22)$$

Zu beliebig kleinem, fest vorgegebenen $c = \delta\tau/\delta x$ können m und n leicht so gewählt werden, dass d_i^1 oder d_{i+1}^1 gegenüber d_i beziehungsweise d_{i+1} das Vorzeichen wechselt, wodurch sich die freie Oberfläche eventuell weiter als eine Gitterzelle bewegt. Aus (4.18) und (4.19) folgt, dass diese Vorzeichenwechsel von d durch die modifizierte CFL-Bedingung

$$\frac{\delta\tau}{\delta x} \leq \frac{|d_k|}{\delta x |s_k| \left| 1 - \frac{d_{i+1}-d_i}{\delta x} \right|} \quad \text{für } k = i, i+1 \quad (4.23)$$

verhindert werden können. Damit keine größere Zeitschrittweitereinschränkung entsteht als durch die ursprüngliche CFL-Bedingung (4.20), wird folgende Approximation für $\text{sign}(d)$ gewählt:

$$s_k = \frac{d_k}{\sqrt{d_k^2 + \left(1 - \left| \frac{d_{i+1}-d_i}{\delta x} \right| \right)^2 \delta x^2}}. \quad (4.24)$$

Diese neue Form der regularisierten signum-Funktion ist von der lokalen Steigung der Level-Set-Funktion abhängig. Dort, wo ∇d lokal sehr steil ist, nähert sich die signum-Funktion dem Wert Null und damit werden die Geschwindigkeiten aus (4.15) im Reinitialisierungsprozeß (4.13) herabgesetzt. Dort, wo ∇d lokal sehr flach ist, nähert sich der Betrag der signum-Funktion dem Wert Eins, wodurch die Geschwindigkeiten aus (4.15) im Reinitialisierungsprozeß (4.13) erhöht werden. Wenn $\nabla d = 1$, dann ist s_k die exakte Darstellung der signum-Funktion.

Die obige Analyse deutet an, dass auch im mehrdimensionalen Fall, wenn als Zeitdiskretisierung das Euler-Verfahren eingesetzt und $\nabla d(x)$ durch einen diskreten Operator Dd approximiert wird, $S(d)$ durch

$$s_k = \frac{d_k}{\sqrt{d_k^2 + (1 - |Dd|)^2 \delta x^2}} \quad (4.25)$$

approximiert werden kann. Dadurch erhalten alle Level-Set-Werte während der Reinitialisierung ihr Vorzeichen, wodurch ein unphysikalisches Bewegen der freien Oberfläche reduziert wird. Dies führt zu einer verbesserten Masseerhaltung. Für Zeitdiskretisierungen höherer Ordnung bieten sich Runge-Kutta-Methoden an, weil diese aus Konvexkombinationen von Euler-Schritten bestehen. Deshalb verwenden wir für die Reinitialisierung, wie im Abschnitt über die Zeitdiskretisierung der Reinitialisierungsfunktion bereits erwähnt, ein Runge-Kutta-Verfahren dritter Ordnung.

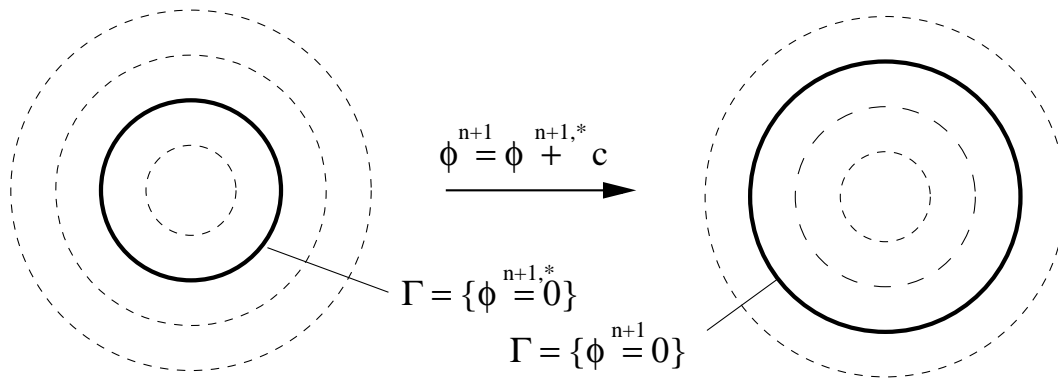


Abbildung 4.6: Volumenkorrektur der Level-Set Funktion.

4.2.5 Picard-Fixpunktiteration zur Masseerhaltung bis auf Maschinengenauigkeit

Die Reinitialisierung der Level-Set Funktion ist sehr wichtig für eine homogene Berechnung sowohl der Viskositäts- und Dichtesprünge als auch der Oberflächenspannung entlang der freien Oberfläche. Die im vorigen Kapitel dargestellte CFL-erhaltende Glättung der signum-Funktion sorgt zusammen mit dem WENO-Verfahren fünfter Ordnung für eine substantiell verbesserte Volumenerhaltung, da hiermit der künstliche Bewegungsfreiraum der freien Oberfläche auf höchstens eine Gittermaschenweite eingeschränkt wird.

In der Regel ist aber bei instationären Problemen in jedem Zeitschritt ein Reinitialisierungsschritt notwendig. Daher besteht die Gefahr, dass sehr kleine Massefehler sich im Laufe der Zeit aufaddieren. Dies führt dazu, dass gerade bei Langzeitrechnungen, also insbesondere für Simulationsprozesse, in denen man in einen stationären Zustand hineinkonvergieren möchte, die Volumenerhaltung wiederum empfindlich gestört werden kann. Um im Folgenden Langzeitrechnungen zu ermöglichen, führen wir nach der Reinitialisierung der Level-Set-Funktion eine zusätzliche Picard-Fixpunktiteration durch, welche die Masseerhaltung bis auf Maschinengenauigkeit forciert [46].

Das Verfahren verläuft wie folgt: Nach der Level-Set-Reinitialisierung ist die Level-Set-Funktion in einer Umgebung der Nullkontur wieder eine vorzeichenbehaftete Abstandsfunktion, die aber in der Regel eine sehr leichte künstliche Verschiebung im Rahmen einer Gittermaschenweite erfahren hat. Der daraus resultierende Volumenverlust kann korrigiert werden, indem eine neue Isolinie von ϕ zu einem leicht von Null verschobenen Niveauewert betrachtet wird. Diese Operation kann umgesetzt werden, indem zum ursprünglichen Niveauewert Null eine Konstante c bestimmt wird, bei der $|c|$ den Abstand zwischen der alten und der neuen Null-Konturlinie beschreibt, siehe Abbildung 4.6. Die Tatsache, dass ϕ in einer ausreichend grossen Umgebung der freien Oberfläche die Eigenschaft einer vorzeichenbehafteten Abstandsfunktion hat, spielt hierbei eine wichtige Rolle.

Die eingesetzte Picard-Fixpunktiteration hat hierbei die folgende Form,

$$\phi^{n+1} \leftarrow \phi^{n+1} + \underbrace{\omega(V_1(\phi^0) - V_1(\phi^{n+1}))}_{= c^{n+1}}, \quad (4.26)$$

wobei $V_1(\phi^0)$ für das Anfangsvolumen einer Phase Ω_1^0 steht und $V_1(\phi^{n+1}) := \int_{\Omega} H(\phi^{n+1}) dx$ das Volumen von Ω_1^{n+1} nach der Reinitialisierung ist. Mit jedem Iterationsschritt wird dabei die freie Oberfläche global in Normalenrichtung um einen bestimmten Abstandswert verschoben. Dieser Abstandswert wird von Iterationsschritt zu Iterationsschritt kleiner, bis das Volumen auf eine beliebig vorgegebene Fehlerschranke dem Anfangsvolumen gleicht. Der Rechenaufwand für dieses Verfahren ist sehr gering. Hierbei kann ω als Transformationsfaktor einer volumetrischen Größe in eine Abstandsgröße angesehen werden. Wenn man beispielsweise die gegebenen Volumina $V_1(\phi^0)$ und $V_1(\phi^{n+1})$ als Kugelvolumina annimmt und ihre zugehörigen Radien r über die Formel

$$r = \sqrt[3]{\frac{3}{4\pi}V} \quad (4.27)$$

berechnet, dann stellt die Differenz beider Radien eine gute Approximation für den Korrekturterm der Level-Set-Funktion dar, der in unseren Anwendungen in wenigen Iterationsschritten die gewünschte Genauigkeit lieferte. Die Anzahl der Iterationen ist dabei von der Geometrie der Level-Set-Funktion abhängig.

Die Verschiebung der Level-Set-Funktion zur Erhaltung der Masse führt zu einer zusätzlichen Fehlerquelle, welche weiterhin in Kapitel 5 numerisch untersucht wird.

4.3 Die Projektionsmethode für inkompressible Zweiphasenströmungen

Die Projektionsmethode nach Chorin-Temam [37, 186] ist ein klassisches Lösungsverfahren für die Navier-Stokes-Gleichungen, welches ursprünglich zur Simulation einphasiger Strömungen verwendet wurde. Aufgrund des Zweiphasensystems erweitern sich die Impulsgleichungen um einen Quellterm für die Oberflächenspannung, ferner kommen der Dichte- und Viskositätssprung hinzu. Diese drei zusätzlichen Größen werden hierbei aus der Level-Set-Funktion ϕ ermittelt. Die genaue Berechnung aller Größen, welche in Abhängigkeit der Level-Set-Funktion ϕ stehen, wurde zuvor in Abschnitt 4.2 näher beschrieben.

Insgesamt hat die für Zweiphasenströmungen erweiterte Chorin-Temam-Projektionsmethode den folgenden Ablauf:

Berechne zuerst ein vorläufiges Geschwindigkeitsfeld \mathbf{u}^* , welches von den Geschwindigkeiten des vorherigen Zeitschritts abhängt, jedoch nicht vom Druckgradienten

$$\mathbf{u}^* = \mathbf{u}^n + \delta t \left(\mathbf{g} - \nabla \cdot (\mathbf{u}^n \otimes \mathbf{u}^n) + \frac{1}{\rho^\epsilon(\phi^n)} (\nabla \cdot (\mu^\epsilon(\phi^n) \mathbf{D}^n) - \sigma \kappa(\phi^n) \delta^\epsilon(\phi^n) \nabla \phi^n) \right). \quad (4.28)$$

Ermittle anschließend eine vorläufige, neue Level-Set-Funktion ϕ^* durch explizites Lösen der Transportgleichung nach Abschnitt 4.2.2,

$$\phi^* = Z(R(\phi^n, \mathbf{u}^n)),$$

wobei Z der diskrete Zeitoperator und R der diskrete Raumoperator ist. Bestimme nun ϕ^{**} durch Reinitialisierung von ϕ^* nach Abschnitt 4.2.3 und berechne aus ϕ^{**} mittels Picard-Iteration die neue Level-Set-Funktion ϕ^{n+1} nach Abschnitt 4.2.5. Aus dem System

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\delta t} + \frac{\nabla p^{n+1}}{\rho(\phi^{n+1})} = 0, \quad (4.29)$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0. \quad (4.30)$$

wird durch Anwendung des Divergenzoperators auf (4.29) und unter Verwendung von (4.30) die Poissongleichung

$$\nabla \cdot \left(\frac{1}{\rho(\phi^{n+1})} \nabla p^{n+1} \right) = \nabla \cdot \frac{\mathbf{u}^*}{\delta t} \quad (4.31)$$

hergeleitet. Hierdurch wird der neue Druck p^{n+1} implizit über das Lösen der Poissongleichung berechnet und liefert schließlich die neuen Geschwindigkeiten \mathbf{u}^{n+1} durch den Korrekturschritt

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\delta t}{\rho(\phi^{n+1})} \nabla p^{n+1}. \quad (4.32)$$

Damit (4.31) wohldefiniert ist, sind noch Randbedingungen für den Druck notwendig. Dazu wird (4.29) auf die äußere Einheitsnormale des Gebietsrandes projiziert, also

$$\left. \frac{\partial p^{n+1}}{\partial \mathbf{n}} \right|_{\Gamma} = \frac{\rho(\phi^{n+1})}{\delta t} (\mathbf{u}_{\Gamma}^* - u_{\Gamma}^{n+1}) \cdot \mathbf{n}. \quad (4.33)$$

Falls nun am Rand die Schätzwgeschwindigkeiten $u_{\Gamma}^* = u_{\Gamma}^{n+1}$ gesetzt wird, ergeben sich homogene Neumann-Randbedingungen für den Druck, also

$$\left. \frac{\partial p^{n+1}}{\partial \mathbf{n}} \right|_{\Gamma} = 0. \quad (4.34)$$

Im Fall konstanter Dichte läßt sich ein skalierter Druck $\hat{p}^{n+1} := \frac{p \Delta t}{\rho}$ definieren. Damit vereinfacht sich die Poissongleichung (4.31) zu

$$\Delta \hat{p}^{n+1} = \nabla \cdot \mathbf{u}^*. \quad (4.35)$$

Der Druck ist durch (4.34) und (4.35) nur bis auf eine Konstante genau bestimmt. Diese kann fixiert werden, indem zusätzlich die Nebenbedingung

$$\int_{\Omega} p^{n+1} = 0$$

gefordert wird. Damit ist es möglich, unterschiedliche Druckergebnisse miteinander zu vergleichen.

Im Folgenden werden die Diskretisierungen der auftretenden Differentialoperatoren und der Löser für die Poissongleichung näher beschrieben.

4.3.1 Diskretisierung der konvektiven Terme mit dem WENO-Verfahren fünfter Ordnung

Wenn Diskretisierungs-, Abbruch- und Rundungsfehler während einer Rechnung nicht unbeschränkt wachsen, dann wird die numerische Lösung als *stabil* bezeichnet. Im Fall konvektionsdominanter Probleme treten bei der Diskretisierung mit zentralen Differenzen Stabilitätsprobleme auf. Es werden unphysikalische Oszillationen in der Lösung erzeugt, die das Ergebnis unbrauchbar machen. Dies liegt daran, dass die diskrete Gleichung bestimmte Eigenschaften der kontinuierlichen Gleichung nicht mehr widerspiegelt (siehe [84]).

Aus diesem Grund müssen alternative Diskretisierungsmethoden für die konvektiven Terme konstruiert werden, die für beliebige Gitterschrittweiten zu beschränkten Lösungen führen. Insbesondere im Hinblick auf die Behandlung der freien Oberfläche mit der Level-Set-Technik ist sowohl die Stabilität als auch eine möglichst hohe Approximationsordnung der Raumdiskretisierung notwendig. Denn einerseits würden sich Instabilitäten sofort auf die freie Oberfläche übertragen, andererseits würde eine zu geringe Ordnung der Raumdiskretisierung wegen des größeren Abbruchfehlers einen durch numerische Diffusion verfälschten freien Rand ergeben.

Die stabile Diskretisierung der konvektiven Terme ist ein Schwerpunkt aus dem Bereich der Numerik von hyperbolischen Erhaltungsgleichungen, wo mit sogenannten *Upwind-Verfahren* effiziente Techniken entwickelt werden. Als eine der effizientesten Methoden gelten die *Weighted-Essentially-Non-Oscillatory (WENO)*-Verfahren.

Ein Vorteil der WENO-Techniken ist, dass sich, bedingt durch eine glatte Gewichtung der Differenzensterne [102], der Normalenvektor und die Krümmung der freien Oberfläche stabiler berechnen lassen.

Ein effizientes WENO-Schema fünfter Ordnung ist von Jiang und Shu [103] entwickelt worden. Die Breite des Diskretisierungsters beträgt sieben Gitterpunkte. In Bereichen, wo die Funktion ϕ glatt ist, schaltet es ein zentrales Schema fünfter Ordnung ein, in Bereichen, wo die Funktion ϕ nicht glatt ist, wird auf ein ENO-Verfahren dritter Ordnung reduziert. Ein zusätzlicher Vorteil bezüglich der Effizienz ist, dass alle logischen Abfragen wegfallen, die beim ENO-Verfahren zu behandeln sind. Die genaue Herleitung der Verfahrens und die Transformation in die Finite-Differenzen-Darstellung ist sehr technisch, hierzu sei auf die Artikel [103, 173] verwiesen.

In der Finite-Differenzen-Darstellung wird die Berechnung der x -Ableitung einer skalaren Grösse ϕ im Zellmittelpunkt, also $[\phi_x]_{i,j,k}$, auf folgende Weise durchgeführt:

Für die Bestimmung von ϕ_x^- setze

$$q_1 = \frac{\phi_{i-2,j,k} - \phi_{i-3,j,k}}{\delta x}, \quad q_2 = \frac{\phi_{i-1,j,k} - \phi_{i-2,j,k}}{\delta x}, \quad q_3 = \frac{\phi_{i,j,k} - \phi_{i-1,j,k}}{\delta x},$$

$$q_4 = \frac{\phi_{i+1,j,k} - \phi_{i,j,k}}{\delta x}, \quad q_5 = \frac{\phi_{i+2,j,k} - \phi_{i+1,j,k}}{\delta x}$$

und um ϕ_x^+ zu finden, setze

$$q_1 = \frac{\phi_{i+3,j,k} - \phi_{i+2,j,k}}{\delta x}, \quad q_2 = \frac{\phi_{i+2,j,k} - \phi_{i+1,j,k}}{\delta x}, \quad q_3 = \frac{\phi_{i+1,j,k} - \phi_{i,j,k}}{\delta x},$$

$$q_4 = \frac{\phi_{i,j,k} - \phi_{i-1,j,k}}{\delta x}, \quad q_5 = \frac{\phi_{i-1,j,k} - \phi_{i-2,j,k}}{\delta x}.$$

Als nächstes werden die Glattheitsmaße über

$$\begin{aligned} IS_1 &= \frac{13}{12}(q_1 - 2q_2 + q_3)^2 + \frac{1}{4}(q_1 - 4q_2 + 3q_3)^2 \\ IS_2 &= \frac{13}{12}(q_2 - 2q_3 + q_4)^2 + \frac{1}{4}(q_2 - q_4)^2 \\ IS_3 &= \frac{13}{12}(q_3 - 2q_4 + q_5)^2 + \frac{1}{4}(3q_3 - 4q_4 + q_5)^2 \end{aligned}$$

und die Gewichte über

$$\omega_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2 + \alpha_3}, \quad \omega_2 = \frac{\alpha_2}{\alpha_1 + \alpha_2 + \alpha_3}, \quad \omega_3 = \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3}$$

mit

$$\alpha_1 = \frac{1}{10} \frac{1}{(\tilde{\epsilon} + IS_1)^2}, \quad \alpha_2 = \frac{6}{10} \frac{1}{(\tilde{\epsilon} + IS_2)^2}, \quad \alpha_3 = \frac{3}{10} \frac{1}{(\tilde{\epsilon} + IS_3)^2}$$

berechnet. Dann wird $[\phi_x^\pm]_{i,j,k}$ ermittelt durch

$$[\phi_x^\pm]_{i,j,k} = \omega_1 \left(\frac{q_1}{3} - \frac{7q_2}{6} + \frac{11q_3}{6} \right) + \omega_2 \left(\frac{-q_2}{6} + \frac{5q_3}{6} + \frac{q_4}{3} \right) + \omega_3 \left(\frac{q_3}{3} + \frac{5q_4}{6} - \frac{q_5}{6} \right).$$

und über die Upwindschaltung wird schließlich $[\phi_x]_{i,j,k}$ bestimmt als

$$[\phi_x]_{i,j,k} = \begin{cases} [\phi_x^-]_{i,j,k} & \text{falls } u_{i,j,k} > 0 \\ [\phi_x^+]_{i,j,k} & \text{falls } u_{i,j,k} < 0 \\ 0 & \text{sonst} \end{cases} \quad (4.36)$$

Zur Berechnung von α_1, α_2 und α_3 wurde in dieser Arbeit für alle Testrechnungen mit dem WENO-Schema der Wert $\tilde{\epsilon} = 10^{-6}$ verwendet.

Um eine gute Konvergenzordnung für die Krümmungsberechnung zu erreichen, ist das WENO-Schema fünfter Ordnung sowohl für die konvektiven Terme der Navier-Stokes-Gleichungen als auch für die konvektiven Terme der Transportgleichung und der Reinitialisierung implementiert worden.

Randbehandlung der konvektiven Terme

Die Breite der Diskretisierungssterne des WENO-Verfahrens bilden ein Problem in der Nähe von Gebietsrändern. Hier gibt es zwei äquivalente Ansätze: entweder man modifiziert den Stern in Randnähe oder man verwendet den ursprünglichen Stern, der dann auf Gitterzellen zugreift, die außerhalb des Rechengebietes liegen. Die Werte in diesen Zellen sind dann geeignet zu setzen.

Die natürlichste Art die Ränder zu behandeln, besteht darin, nur Werte zu verwenden die innerhalb des Rechengebietes liegen. Anders formuliert, alle Teilsterne eines WENO-Schemas sollten so gewählt werden, dass sie nur auf vorhandene Gitterpunkte des Rechengebietes zugreifen, inklusive einer Randbordüre, in der entsprechend vorgegebene Randwerte mittels einer Newton-Polynominterpolation dritten Grades gesetzt werden. Um teure logische Operationen zu umgehen, wie beispielsweise Randabfragen, werden zu Beginn einer

Rechnung alle Randzellen außerhalb der Randbordüre und des Fluidgebietes mit sehr großen und stark variierenden Werten besetzt. Im Rahmen dieser Arbeit wurden diese Randzellen mit den folgenden Werten belegt: $R_{-i,j,k} = (-1)^i 10^{10}$ für $x_{-i,j,k}$ als Randzellenpunkte mit $j = 1, \dots, J$, $k = 1, \dots, K$, und $i = 1, 2, \dots$ bestimmt die Anzahl der Randzellenreihen linksseitig des Rechengebietes; analoges gilt für alle anderen Gebietsränder. Dadurch greift das WENO-Schema ausschließlich auf finite Differenzen zu, die aus dem Fluidgebiet inklusive der Randbordüre stammen, weil diese glatter sind. Mit dieser Technik ermöglicht das WENO-Schema fünfter Ordnung eine Randbehandlung dritter Ordnung.

4.3.2 Diskretisierung der viskosen Terme

Der viskose Anteil $\nabla \cdot \mu(\phi)\mathbf{D}$ des Spannungstensors \mathbf{T} bezieht die partiellen Ableitungen der Viskosität $\mu(\phi)$ mit ein, die im Falle von Zweiphasenströmungen keine globalen Konstanten sind. Zur Berechnung des Hilfgeschwindigkeitsfeldes \mathbf{u}^* in (4.28) müssen die viskosen Terme der x -Geschwindigkeitskomponente

$$2(\mu(\phi)u_x)_x + (\mu(\phi)(u_y + v_x))_y + (\mu(\phi)(u_z + w_x))_z \quad (4.37)$$

an den Stellen $(i \pm \frac{1}{2}, j, k)$ diskretisiert werden. Genauso wird für die viskosen Terme der y -Geschwindigkeitskomponente die Diskretisierung von

$$(\mu(\phi)(u_y + v_x))_x + 2(\mu(\phi)u_y)_y + (\mu(\phi)(v_z + w_y))_z \quad (4.38)$$

an den Stellen $(i, j \pm \frac{1}{2}, k)$ und für die viskosen Terme der z -Geschwindigkeitskomponente die Diskretisierung von

$$(\mu(\phi)(u_z + w_x))_x + (\mu(\phi)(v_z + w_y))_y + 2(\mu(\phi)w_z)_z \quad (4.39)$$

an den Stellen $(i, j, k \pm \frac{1}{2})$ benötigt. Da für viskose Fluide die Geschwindigkeiten glatt sind, können die ersten Ableitungen mit zentralen Differenzen ermittelt werden. So berechnen sich zum Beispiel die Ableitungen von u über

$$\begin{aligned} [u_x]_{i,j,k} &= \frac{u_{i+\frac{1}{2},j,k} - u_{i-\frac{1}{2},j,k}}{\delta x}, \\ [u_y]_{i+\frac{1}{2},j+\frac{1}{2},k} &= \frac{u_{i+\frac{1}{2},j+1,k} - u_{i+\frac{1}{2},j,k}}{\delta y}, \\ [u_z]_{i+\frac{1}{2},j,k+\frac{1}{2}} &= \frac{u_{i+\frac{1}{2},j,k+1} - u_{i+\frac{1}{2},j,k}}{\delta z}. \end{aligned}$$

Die ersten Ableitungen der anderen Geschwindigkeitskomponenten werden analog dazu berechnet. Desweiteren können die zweiten partiellen Ableitungen aus (4.37), (4.38) und (4.39) ebenfalls mit zentralen Differenzen bestimmt werden, weil der Viskositätssprung an der Phasengrenze durch die Heaviside'sche Gewichtsfunktion H^ϵ regularisiert wurde. Damit lassen

sich sämtliche Terme aus (4.37) an der Stelle $(i + \frac{1}{2}, j, k)$ approximieren über

$$\begin{aligned}
 & [(2\mu(\phi)u_x)_x]_{i+\frac{1}{2},j,k} \\
 &= 2 \frac{\mu(\phi_{i,j,k})[u_x]_{i,j,k} - \mu(\phi_{i+1,j,k})[u_x]_{i+1,j,k}}{\delta x}, \\
 & [(\mu(\phi)(u_y + v_x))_y]_{i+\frac{1}{2},j,k} \\
 &= \frac{\mu(\phi_{i+\frac{1}{2},j+\frac{1}{2},k})[u_y+v_x]_{i+\frac{1}{2},j+\frac{1}{2},k} - \mu(\phi_{i+\frac{1}{2},j-\frac{1}{2},k})[u_y+v_x]_{i+\frac{1}{2},j-\frac{1}{2},k}}{\delta y}, \\
 & [(\mu(\phi)(u_z + w_x))_z]_{i+\frac{1}{2},j,k} \\
 &= \frac{\mu(\phi_{i+\frac{1}{2},j,k+\frac{1}{2}})[u_z+w_x]_{i+\frac{1}{2},j,k+\frac{1}{2}} - \mu(\phi_{i+\frac{1}{2},j,k-\frac{1}{2}})[u_z+w_x]_{i+\frac{1}{2},j,k-\frac{1}{2}}}{\delta z}.
 \end{aligned}$$

Vor der Berechnung der Viskositätswerte $\mu(\phi)$ wird die Level-Set-Funktion ϕ durch passende Lagrange-Polynominterpolation dritten Grades an den richtigen Stellen ausgewertet (siehe Abbildungen 4.2 und 4.3). Ebenso werden alle Terme aus (4.38) an den Stellen $(i, j + \frac{1}{2}, k)$ und alle Terme aus (4.39) an den Stellen $(i, j, k + \frac{1}{2})$ berechnet.

4.3.3 Löser für die Poissongleichung mit nichtkonstanter Dichte

In diesem Abschnitt behandeln wir die Diskretisierung und das Lösen der Druck-Poisson-Gleichung mit nichtkonstanten Koeffizienten

$$-\nabla \cdot \frac{1}{\rho^\epsilon(\phi_h^{n+1})} \nabla p_h^{n+1} = -\nabla \cdot \mathbf{u}_h^*, \quad (4.40)$$

wobei $\rho^\epsilon(\phi_h^{n+1})$ das geglättete Dichtefeld (4.1) nach dem Advektionsschritt (4.2.2), der Reinitialisierung (4.13) und der darauf folgenden Picard-Fixpunktiteration (4.26) bezeichnet. Da die Geschwindigkeiten eines viskosen Fluids ausreichend glatt sind, berechnen wir die rechte Seite von (4.40) mit zentralen Differenzen

$$[\nabla \cdot \mathbf{u}^*]_{i,j,k} = \frac{u_{i+\frac{1}{2},j,k}^* - u_{i-\frac{1}{2},j,k}^*}{\delta x} + \frac{v_{i,j+\frac{1}{2},k}^* - v_{i,j-\frac{1}{2},k}^*}{\delta y} + \frac{w_{i,j,k+\frac{1}{2}}^* - w_{i,j,k-\frac{1}{2}}^*}{\delta z}.$$

Die linke Seite von (4.40) wird mit dem herkömmlichen Sieben-Punkte-Stern in den Zellmittelpunkten diskretisiert. Man erhält

$$\begin{aligned}
 \left[\nabla \cdot \frac{1}{\rho^\epsilon(\phi)} \nabla p^{n+1} \right]_{i,j,k} &= \frac{1}{(\delta x)^2} \left(\frac{p_{i+1,j,k}^{n+1} - p_{i,j,k}^{n+1}}{\rho^\epsilon(\phi_{i+\frac{1}{2},j,k})} - \frac{p_{i,j,k}^{n+1} - p_{i-1,j,k}^{n+1}}{\rho^\epsilon(\phi_{i-\frac{1}{2},j,k})} \right) \\
 &+ \frac{1}{(\delta y)^2} \left(\frac{p_{i,j+1,k}^{n+1} - p_{i,j,k}^{n+1}}{\rho^\epsilon(\phi_{i,j+\frac{1}{2},k})} - \frac{p_{i,j,k}^{n+1} - p_{i,j-1,k}^{n+1}}{\rho^\epsilon(\phi_{i,j-\frac{1}{2},k})} \right) \\
 &+ \frac{1}{(\delta z)^2} \left(\frac{p_{i,j,k+1}^{n+1} - p_{i,j,k}^{n+1}}{\rho^\epsilon(\phi_{i,j,k+\frac{1}{2}})} - \frac{p_{i,j,k}^{n+1} - p_{i,j,k-1}^{n+1}}{\rho^\epsilon(\phi_{i,j,k-\frac{1}{2}})} \right),
 \end{aligned}$$

wobei das geglättete Dichtefeld in den Mittelpunkten der Zellseitenflächen über eine Lagrange-Interpolation dritter Ordnung aus den zellzentrierten Level-Set-Werten ermittelt wird.

Entsprechend dem Abschnitt 4.3 ergänzen wir die Druckgleichung mit homogenen Neumann-Randbedingungen und normieren den globalen Mittelwert des Drucks auf Null. Zur iterativen Lösung der resultierenden semi-definiten Diffusionsgleichung werden in dieser Arbeit die folgenden zwei Verfahren eingesetzt:

- ein paralleler BiCGStab-Löser mit Jacobi-Vorkonditionierer [43, 44],
- ein paralleler AMG-vorkonditionierter BiCGStab-Löser [79, 130].

In Problemen mit großen Variationen in der Dichte, komplexen Geometrien der freien Oberfläche und sehr feinen Gittermaschenweiten ist der AMG-vorkonditionierte BiCGStab-Löser deutlich schneller als der BiCGStab-Löser mit Jacobi-Vorkonditionierer. Man beachte aber, dass AMG-basierte Löser aufgrund des gebietsabhängigen Operators regelmässig eine neue Matrixassemblierung benötigen, welche im Fall des BiCGStab-Lösers mit Jacobi-Vorkonditionierer nicht anfällt.

4.4 Die adaptive Zeitschrittweitensteuerung

Die Berechnung der Simulationen wird mit einer adaptiven Zeitschrittweitensteuerung kontrolliert. Bei expliziten Integrationsverfahren wird die Zeitschrittweite aus Stabilitätsgründen beschränkt durch die konvektiven und viskosen Terme und durch die Volumen- und Oberflächenspannungskräfte der Navier-Stokes-Gleichungen. Zur Bestimmung der Stabilitätsbedingungen werden die von Neumannsche oder Fourier-Stabilitätsanalyse [149] für lineare Gleichungen und die Methode der modifizierten Gleichung von C. W. Hirt [93, 143], welche auch für nichtlineare Gleichungen anwendbar ist, hinzugezogen. Eine Stabilitätsanalyse der diskreten Erhaltungsgleichungen für Zweiphasenströmungen ist allerdings ein äußerst aufwendiges Unterfangen. In der Praxis hat sich der vergleichsweise einfache Ansatz bewährt, die Ergebnisse der Analyse einfacher linearer Modellgleichungen durch Analogiebildung auf die Navier-Stokes-Gleichungen zu übertragen.

Im Folgenden wird schrittweise auf die verwendete Zeitschrittweitensteuerung hingeführt, mit der sichergestellt werden soll, dass die Informationen pro Zeitschritt maximal eine Gitterzelle weit fließen können, da die diskreten Differenzgleichungen nur Flüsse zwischen benachbarten Gitterzellen betrachten.

Die Zeitschrittweitenbeschränkung bezüglich der konvektiven Terme sorgt dafür, dass keine Konvektion weiter als eine Gitterzelle pro Zeitschritt stattfindet. Diese Bedingung ist als Courant-Friedrichs-Lewy-Bedingung (CFL-Bedingung) [154] bekannt und fordert

$$\delta t_k \leq \min_{\Omega} \left(\frac{\delta x}{|u|_{max}}, \frac{\delta y}{|v|_{max}}, \frac{\delta z}{|w|_{max}} \right), \quad (4.41)$$

wobei $|\cdot|_{max}$ für die maximale Absolutgeschwindigkeit steht.

Wenn die dynamischen Viskositäten $\mu_l > 0$ und $\mu_g > 0$ sind, dann darf die Diffusion ebenfalls nicht weiter als eine Gitterzellenweite pro Zeitschritt wirken. Dies führt zur Zeitschrittweitenbeschränkung bezüglich der viskosen Terme,

$$\delta t_v \leq \left(\max \left\{ \frac{\mu_1}{\rho_1}, \frac{\mu_2}{\rho_2} \right\} \left(\frac{2}{(\delta x)^2} + \frac{2}{(\delta y)^2} + \frac{2}{(\delta z)^2} \right) \right)^{-1}. \quad (4.42)$$

Die Ungleichung (4.42) resultiert aus der Stabilitätsanalyse [149] für eine explizite Zeitdiskretisierung der Diffusionsgleichung.

Die Volumenkräfte, wie zum Beispiel die Gravitation, können in der Konvektionsabschätzung (4.41) mitberücksichtigt werden, denn $|u|_{max} + |g_1|\delta t$ ist eine lineare Approximation an die obere Schranke der Geschwindigkeitskomponente in x -Richtung inklusive der Volumenkraft. Damit lautet die um die Volumenkraft g_1 erweiterte CFL-Bedingung für die Zeitschrittweite δt_{gk}^u bezüglich der u -Komponente des Geschwindigkeitsfeldes

$$\delta t_{gk}^u \leq \left(\frac{|u|_{max} + |g_1|\delta t_{gk}^u}{\delta x} \right)^{-1}.$$

Nach δt_{gk}^u aufgelöst ergibt sich die Darstellung

$$\delta t_{gk}^u \leq 2 \left(\frac{|u|_{max}}{\delta x} + \sqrt{\left(\frac{|u|_{max}}{\delta x} \right)^2 + \frac{4|g_1|}{\delta x}} \right)^{-1}. \quad (4.43)$$

Analog dazu errechnen sich die Abschätzungen bezüglich der v - und w -Geschwindigkeitskomponente inklusive der Volumenkräfte. Ferner können die konvektiven und diffusiven Kräfte gemeinsam wirken. Aus diesem Grund ist die Zeitschrittweitenbeschränkung (4.42) mit der Abschätzung (4.43) zu kombinieren.

Zusammen mit $V := \max \left\{ \frac{\mu_1}{\rho_1}, \frac{\mu_2}{\rho_2} \right\} \left(\frac{2}{(\delta x)^2} + \frac{2}{(\delta y)^2} + \frac{2}{(\delta z)^2} \right)$ erhalten wir

$$\delta t_{gkv}^u \leq 2 \left(\left(\frac{|u|_{max}}{\delta x} + V \right) + \sqrt{\left(\frac{|u|_{max}}{\delta x} + V \right)^2 + \frac{4|g_1|}{\delta x}} \right)^{-1} \quad (4.44)$$

als die Beschränkung der Zeitschrittweite für die u -Geschwindigkeitskomponente bezüglich der Konvektion, der Diffusion und der Volumenkräfte.

Desweiteren wird die Oberflächenspannung als Volumenkraftterm der Form $\frac{1}{\rho}(\sigma\delta^\epsilon\kappa)\mathbf{n}$ in den Impulsgleichungen (4.28) aktiviert. Dabei hat die δ^ϵ -Funktion (4.3) als maximalen Wert $\frac{1}{\epsilon(\mathbf{x})}$, wobei $\epsilon(\mathbf{x})$ in (4.4) definiert ist und die Stärke der Regularisierung angibt, welche von der Größe $O(\delta x)$ ist. Die regularisierte Dichte hat an dieser Stelle den Wert $\frac{1}{2}(\rho_l + \rho_g)$. Somit ist

$$\mathbf{a} = \frac{2|\kappa|\sigma}{\epsilon(\mathbf{x})(\rho_l + \rho_g)} \mathbf{n} \quad (4.45)$$

eine Approximation an die von der Oberflächenspannung induzierten Geschwindigkeit. Da die Oberflächenspannungskraftkomponenten $\mathbf{a} = (a_1, a_2, a_3)$ jeweils analog zur Volumenkraft g_1 in Gleichung (4.44) eingebaut werden können, ergibt sich damit als Zeitschrittweiteinschränkung bezüglich der u -Geschwindigkeitskomponente die erweiterte Darstellung

$$\delta t_{gkvo}^u \leq 2 \underbrace{\left(\left(\frac{|u|_{max}}{\delta x} + V \right) + \sqrt{\left(\frac{|u|_{max}}{\delta x} + V \right)^2 + \frac{4|g_1|}{\delta x} + \frac{4a_1}{\delta x}} \right)^{-1}}_{=:C_u}. \quad (4.46)$$

C_v und C_w lassen sich analog definieren, so dass man $\delta t_{gvko}^\alpha \leq 2C_\alpha$, $\alpha = u, v, w$ erhält. Aus (4.46) erhält man die Ordnung $O((\delta x)^{\frac{3}{2}})$ für die Zeitschrittweitereinschränkung bezüglich der Oberflächenspannungskraft, wenn man in (4.45) die kleinstmögliche Wellenzahl $\kappa = \pi/\delta x$ einsetzt, welche zur kleinsten auflösbaren Wellenlänge $2\delta x$ führt. Die gleiche Ordnung für δt wurde auch in [23] angegeben.

Die vollständige Zeitschrittweitensteuerung ergibt sich aus dem Minimum der zulässigen Zeitschritte bezüglich aller kartesischen Geschwindigkeitskomponenten und lautet

$$\delta t \leq 2\tau \min_{\Omega} (C_u, C_v, C_w), \quad (4.47)$$

dabei ist $\tau \in (0, 1]$ eine Sicherheitskonstante für die Zeitschrittweite. In unseren Simulationsexperimenten wurde $\tau = 0.2$ gesetzt.

4.5 Parallelisierung des Zweiphasen-Strömungslösers

Die rasante Entwicklung von Groß- beziehungsweise Hochleistungsrechnern ermöglicht es, immer komplexere Problemstellungen in wesentlich kürzerer Zeit zu berechnen. Bis vor einem Jahrzehnt galten Vektorrechner als die schnellsten und leistungsstärksten Großrechner für numerische Probleme. Mittlerweile assoziiert man mit dem Begriff des Hochleistungsrechners hauptsächlich den des Parallelrechners. Sowohl dessen Skalierbarkeit als auch die Standardisierung paralleler Programmiersprachen und Bibliotheken haben den Parallelrechner weltweit etabliert.

4.5.1 Gebietszerlegung als Parallelisierungsstrategie

Ein naheliegender Ansatz zur Parallelisierung numerischer Algorithmen ist, das gesamte Rechengebiet Ω beziehungsweise dessen diskretes Gegenstück Ω_h in einzelne Teilgebiete $\Omega_h^1, \dots, \Omega_h^P$ zu zerlegen. Dabei gibt P die Anzahl der Teilgebiete an. Derartige *Gebietszerlegungsmethoden* sind über hundert Jahre alt und werden oftmals, nach ihrem Entwickler H. A. Schwarz [167], auch als Schwarz'sche Methoden bezeichnet. Die Strategie zur Zerteilung des Lösungsgebietes Ω_h in einzelne Teilgebiete $\Omega_h^1, \dots, \Omega_h^P$ hängt stark von der Problemstellung und dem verwendeten Gitter ab. Für das in dieser Arbeit zur Diskretisierung eingesetzte kartesische Gitter wird eine Gebietszerlegung in angrenzende Blöcke verwendet. Alle diese Blöcke werden um künstliche Überlappungszellen, sogenannte „Randbordüren“ [77], an den Rändern erweitert, damit die Diskretisierungssterne während der Berechnungen nicht auf undefinierte Werte zugreifen. Die Werte in den Randbordüren sind gleich den Werten der entsprechenden inneren Gitterzellen des angrenzenden Gebietes zu setzen.

Um immer die aktuellen Werte in den Randbordüren zu verwenden, erfolgt ein regelmäßiger Datenaustausch (Kommunikation) zwischen den einzelnen Prozessoren, vergleiche Abbildung 4.7. Dabei ist darauf zu achten, dass der Kommunikationsaufwand geringer ist als der Berechnungsaufwand. Demzufolge sollten die Überlappungsgebiete möglichst klein gehalten werden. Entscheidend für die Effizienz der Parallelisierung ist auch eine bestmögliche Lastbalancierung, bei der alle Blöcke annähernd die gleiche Anzahl an Unbekannten enthalten.

Jedes Teilgebiet läßt sich dann durch einen eigenen Prozess realisieren und kann somit je einem Prozessor zur Berechnung der Unbekannten zugeordnet werden. Damit reduziert sich der erforderliche Speicher- und Rechenaufwand pro Prozessor in Abhängigkeit der Anzahl an Teilgebieten.

Im Idealfall sollte sich bei gleichbleibender Anzahl an Gitterpunkten (also Unbekannten) die gesamte Rechenzeit beim Einsatz von P Prozessoren für P Teilgebiete entsprechend um das P -fache verkürzen. Aufgrund des zwingend notwendigen Kommunikationsaufwandes läßt sich dieser optimale Wert in der Praxis nicht erreichen.

4.5.2 Parallelisierung des Strömungsprogramms

Die Parallelisierung des im Rahmen dieser Arbeit erstellten Simulationsprogramms für Zweiphasenströmungen erfolgte mit der Kommunikationsbibliothek *MPI (message passing interface)* [6].

Weil MPI von Anfang an standardisiert wurde, hat es seit seiner Einführung eine weite Verbreitung und Akzeptanz gefunden. Mittlerweile existieren für fast alle Parallelrechner und Rechnercluster MPI-Portierungen. Ebenso ist mit MPICH des Argonne National Laboratory [211] eine Public-Domain-Version erhältlich. Charakteristisch für MPI ist, dass es heterogenes paralleles Rechnen unterstützt. Dies bedeutet, dass verschiedenste Rechner zu einem großen virtuellen Parallelrechner verbunden werden können.

Mit dem Ziel eines möglichst geringen Kommunikationsaufwandes wird für die Umsetzung der Gebietszerlegungsmethode zunächst das gesamte diskrete Rechengebiet Ω_h über die Minimierung der Kostenfunktion K

$$K(P^x, P^y, P^z) = \frac{I}{P^x} \cdot \frac{J}{P^y} + \frac{J}{P^y} \cdot \frac{K}{P^z} + \frac{I}{P^x} \cdot \frac{K}{P^z}$$

unter der Nebenbedingung: $P = P^x \cdot P^y \cdot P^z$ mit $P^x, P^y, P^z \in \mathbb{N}$

in P Teilgebiete unterteilt, wobei I , J und K die vorgegebenen Anzahlen der Gitterzellen in x -, y - und z -Richtung sind und P^x, P^y und P^z die Anzahl der Zerlegungen in x -, y - beziehungsweise z -Richtung beschreiben. Falls P^x, P^y und P^z , wie bei allen Testrechnungen dieser Arbeit, Teiler von I, J beziehungsweise K sind, entsteht hierdurch eine uniforme Gebietszerlegung in Quader.

Desweiteren werden unterschiedliche Anzahlen an Randbordüren benötigt, weil die Diskretisierungssterne der einzelnen Differentialoperatoren verschieden groß sind. Schematisch zeigt Abbildung 4.7 den Kommunikationsschritt bezüglich einer Randbordüre für die Geschwindigkeiten an den Zellseitenflächen und die skalaren Werte (Druck, Level-Set-Funktion) in den Zellmittelpunkten. Analog dazu wird der Kommunikationsprozeß auf zwei oder drei Randbordüren erweitert. In dieser Arbeit sind für die unterschiedlichen Variablen folgende Anzahlen an künstlichen Randpunktschichten eingesetzt worden:

- die Geschwindigkeiten und die Level-Set-Funktion benötigen jeweils drei Randzellenschichten für das WENO-Verfahren fünfter Ordnung,
- der Druck benötigt für den Poisson-Löser nur eine Randzellenschicht,

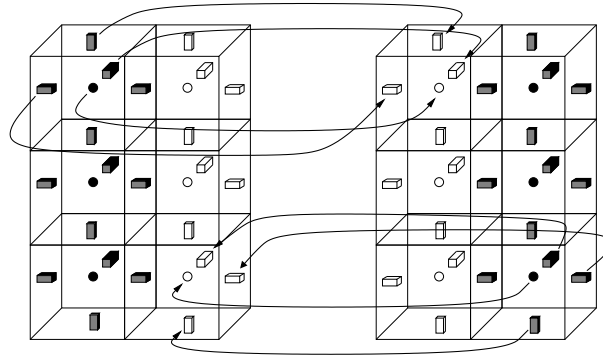


Abbildung 4.7: Datenaustausch am Beispiel einer „künstlichen Randpunktreihe“.

Während des Programmablaufs werden ausschließlich die aktuell geforderten Randzellenwerte versendet. So findet unter anderem vor der Berechnung der nicht divergenzfreen Geschwindigkeiten \mathbf{u}^* eine Kommunikation für $\mathbf{u} = (u, v, w)$ statt. Das Lösen der Poissongleichung benötigt vor jedem Iterationsschritt eine Kommunikation für den Druck. Schließlich werden für die Berechnung der Transportgleichung und der Level-Set-Reinitialisierung zusätzliche Kommunikationen für die Level-Set-Funktion notwendig. Während der Kommunikationsschritte ist insbesondere darauf zu achten, dass die Geschwindigkeits-, und Level-Set-Werte in den Ecken- und Kantenbordüren eines jeden Teilblocks auch korrekt ausgetauscht werden. Ist dies nicht der Fall, so ergeben sich falsche Werte bei der Berechnung der zweidimensionalen Lagrange-Interpolation aus Abschnitt 4.1.

Aus diesem Grund ist eine zusätzliche Kommunikationsroutine programmiert worden, die ausschließlich Werte aus den Ecken und Kanten der jeweiligen Teilblöcke in die Ecken- und Kantenbordüren aller angrenzenden Blöcke überträgt, also auch in die Teilblöcke, welche in Diagonal- beziehungsweise in Raumdiagonalrichtung angrenzen. Der vollständige parallele Algorithmus inklusive Kommunikationsroutinen wird im folgenden Abschnitt formuliert.

4.6 Vollständiger paralleler Algorithmus für inkompressible Zweiphasenströmungen

Der vollständige parallele Algorithmus für inkompressible Zweiphasenströmungen ist wie folgt aufgebaut:

ALGORITHMUS 1: Paralleler Löser für Zweiphasenströmungen.

1. Initialisiere $\delta x, \delta y, \delta z, T, \phi_0$ und Ω .
2. Setze die lokalen Teilgebiete $\Omega^q, q \in \{1, \dots, P\}$ nach (4.48).

3. Setze $t := 0$, $n := 0$, $h := \max(\delta x, \delta y, \delta z)$.
4. Setze Anfangswerte für $\mathbf{u}^n := \mathbf{u}_0$, $p^n := p_0$, und $\phi^n := \phi_0$ auf den Teilgebieten Ω^q , $q \in \{1, \dots, P\}$.
5. Berechne das Volumen von Ω_1^0 .
6. Setze $\epsilon(\mathbf{x})$ nach (4.4).
7. Setze Randwerte für \mathbf{u}^n auf den lokalen Rändern $\partial\Omega^q \cap \partial\Omega$.
8. ZEIT-SCHLEIFE: Solange $t \leq T$:
 - (a) BERECHNUNG DER ZEITSCHRITTWEITE:
 - Berechne den lokalen Zeitschritt δt^q nach Abschnitt 4.4.
 - Berechne den globalen Zeitschritt $\delta t := \min_{\hat{q}} \delta t^{\hat{q}}$.
 - (b) IMPULSGLEICHUNG I:
 - Kommunikation der Randbordürenwerte für u^n und ϕ^n .
 - Berechne das Hilfgeschwindigkeitsfeld $\mathbf{u}^*(\mathbf{u}^n, \phi^n)$ auf den Teilgebieten Ω^q , $q \in \{1, \dots, P\}$ nach Abschnitt 4.3 mit Gleichung (4.10) als Zeitdiskretisierung.
 - Setze Randwerte für \mathbf{u}^* auf den lokalen Rändern $\partial\Omega^q \cap \partial\Omega$.
 - (c) DRUCK-POISSON-GLEICHUNG I:
 - Kommuniziere die Randbordürenwerte für u^* .
 - Setze die rechte Hand Seite $b := \nabla \cdot \mathbf{u}^*$ auf lokalem Gebiet Ω^q für die Druck-Poisson-Gleichung.
 - (d) TRANSPORTGLEICHUNG:
 - Berechne den Transport der Level-Set-Funktion $\phi^*(\mathbf{u}^n, \phi^n)$ nach Abschnitt 4.2.2.
 - (e) Setze $\psi^n := \phi^*$.
 - (f) REINITALISIERUNGS-SCHLEIFE: Solange $\tau < 7 \epsilon$:
 - i. Setze Randbedingungen für ψ^n und tausche die Randbordürenwerte der Teilgebiete Ω^q , $q \in \{1, \dots, P\}$ aus.
 - ii. Berechne $\psi^{n+\frac{1}{3}}$ auf den Teilgebieten Ω^q , $q \in \{1, \dots, P\}$ nach Gleichung (4.14).
 - iii. Setze Randbedingungen für $\psi^{(n+\frac{1}{3})}$ und tausche die Randbordürenwerte der Teilgebiete Ω^q , $q \in \{1, \dots, P\}$ aus.
 - iv. Berechne $\psi^{n+\frac{2}{3}}$ auf den Teilgebieten Ω^q , $q \in \{1, \dots, P\}$ nach Gleichung (4.14).
 - v. Setze Randbedingungen für $\psi^{(n+\frac{2}{3})}$ und tausche die Randbordürenwerte der Teilgebiete Ω^q , $q \in \{1, \dots, P\}$ aus.

4.6. VOLLSTÄNDIGER PARALLELER ALGORITHMUS FÜR INKOMPRESSIBLE
ZWEIPHASENSTRÖMUNGEN 65

- vi. Berechne ψ^{n+1} auf den Teilgebieten Ω^q , $q \in \{1, \dots, P\}$ nach Gleichung (4.14).
- vii. Setze $\tau = \tau + \delta\tau$.
- (g) PICARD-FIXPUNKTITERATION: Solange $|V_1(\phi^0) - V_1(\phi^{n+1})| \leq \epsilon_{vol}$:
Berechne das Volumen von Ω_1^{n+1} .
Korrigiere die Level-Set-Funktion nach Gleichung (4.26).
- (h) Setze $\phi^{n+1} = \psi^{n+1}$ und $\hat{p}_{it} := p^n \delta t$ und $\epsilon(\mathbf{x})$ nach Gleichung (4.4).
- (i) DRUCK-POISSON-GLEICHUNG II:
Löse $\hat{p}^{n+1}(\mathbf{u}^*, \phi^{n+1})$ mit Jacobi/AMG-vorkonditioniertem Krylov-Verfahren.
Kommuniziere die Randbordüerwerte für \hat{p}_{it} pro Iteration it .
- (j) Setze $p^{n+1} := (\delta t)^{-1} \hat{p}_{it}$.
- (k) IMPULSGLEICHUNG II:
Setze \mathbf{u}^{n+1} nach Abschnitt 4.3 auf den Teilgebieten Ω^q .
Setze Randwerte für \mathbf{u}^{n+1} auf den Teilgebieten $\partial\Omega^q \cap \partial\Omega$.
- (l) Setze $t = t + \delta t$ und $n = n + 1$.

Kapitel 5

Numerische Ergebnisse des parallelen Zweiphasenlösers

In diesem Kapitel präsentieren wir verschiedene Testrechnungen, die die Funktionalität beziehungsweise das Verhalten des Zweiphasenlösers und seiner einzelnen Komponenten auf eine detaillierte Weise darstellen sollen. Zunächst betrachten wir den reinen Transport einer Sphäre in einem Rotationsfeld, um den Einfluß der diskretisierten Transportgleichung auf Form und Volumen der Sphere zu beleuchten. Danach untersuchen wir das Verhalten der Reinitialisierung auf die Form einer Niveaumenge zum Niveauwert Null. Hierbei geben wir als Anfangswert eine komplexe Level-Set-Funktion vor, die stark von einer Abstandsfunktion abweicht. Diese soll dann über den Reinitialisierungsprozess in eine entsprechende vorzeichenbehaftete Abstandsfunktion umgewandelt werden ohne dabei das Nullmengenniveau zu verändern.

Nachdem die Transportgleichung und die Reinitialisierung getrennt voneinander betrachtet worden sind, werden die Masseerhaltung und die Konvergenz der Krümmung am Beispiel einer diagonal transportierten Sphäre untersucht. Hier legen wir besonderes Augenmerk auf den Einfluß der Fixpunktiteration hinsichtlich der Masseerhaltung und des Konvergenzverhaltens.

Als erstes anwendungsorientiertes Testproblem untersuchen wir die Konvergenzrate des Zweiphasenlösers mit Oberflächenspannung am Beispiel einer aufsteigenden Blase. Hierzu betrachten wir zu Beginn eine sphärische Blase bestehend aus einem leichteren Fluid als das sie umgebene Fluid. Aufgrund der wirkenden Gravitationskraft werden Auftriebskräfte induziert, die zusammen mit der Oberflächenspannungskraft eine Deformation der Blase im zeitlichen Verlauf hervorrufen.

Abschliessend präsentieren wir Simulationsergebnisse, die das Deformationsverhalten eines Tropfens in einer Scherströmung zeigen. Hierzu führen wir systematische Studien bezüglich der Kapillarzah Ca (siehe (5.10)) und der Reynoldszahl Re (siehe (2.31)) durch, die eine Vielzahl von unterschiedlichen Tropfenformen und Stömungsstrukturen beinhalten. Die hinsichtlich des (Ca, Re) -Phasendiagramms zu erwartenden Topologieveränderungen können dabei vollständig wiedergegeben werden.

5.1 Reine Advektion einer Sphäre in einem Rotationsfeld

Im Folgenden betrachten wir den Einheitswürfel $\Omega = (0, 1)^3$, in dem sich eine Kugel der Größe

$$B = \{\mathbf{x} \in \mathbb{R}^3 : \|\mathbf{x} - \mathbf{x}_M\| < 0.2\}$$

mit dem Mittelpunkt $\mathbf{x}_M = (0.5, 0.25, 0.5)$ befindet. Analog zu dem Test in [85] verwenden wir das Geschwindigkeitsfeld

$$\mathbf{u}^*(\mathbf{x}) = c(\mathbf{x}) \cdot \begin{pmatrix} y_2 \\ -y_1 \\ 0 \end{pmatrix}$$

mit $\mathbf{y} = \mathbf{x} - (0.5, 0.5, 0.5)$ und $c(\mathbf{y}) = \max(4\|\mathbf{y}\|(0.5 - \|\mathbf{y}\|), 0)$. Somit beschreibt \mathbf{u}^* ein Rotationsfeld, welches zum Gebietsrand $\partial\Omega$ hin verschwindet. Abbildung 5.1 oben veranschaulicht die Anfangsbedingung für die Level-Set Funktion und das zeitlich konstante Geschwindigkeitsfeld. Desweiteren betrachten wir das Zeitintervall $[t_0, t_e] = [0, 20]$ und definieren das Geschwindigkeitsfeld \mathbf{u} so, dass es zum Mittelpunkt $t = 10$ des Zeitintervalls sein Vorzeichen ändert und damit den deformierten Ball wieder an seinem Ausgangspunkt zurückführt, das heißt

$$\mathbf{u}(\mathbf{x}, t) = \begin{cases} \mathbf{u}^*(\mathbf{x}) & t \leq 10 \\ -\mathbf{u}^*(\mathbf{x}) & t > 10. \end{cases}$$

Für eine freie Oberfläche $\Gamma \subset \Omega$, welche mit dem Geschwindigkeitsfeld $\mathbf{u}(\mathbf{x}, t)$ transportiert wird, gilt

$$\Gamma(t_0 + t) = \Gamma(t_e - t) \quad \text{für } t \in \left[t_0, \frac{t_0 + t_e}{2} \right] = [0, 10]. \quad (5.1)$$

Diese Eigenschaft ermöglicht es uns, auf eine graphische Art und Weise den transportbedingten Masseverlust des Balles zu untersuchen, indem wir beispielsweise die Visualisierungen der Zeitpunkte $t = 0$ und $t = 20$ und die Visualisierungen der Zeitpunkte $t = 5$ und $t = 15$ miteinander vergleichen.

Als Anfangswert für die Level-Set Funktion verwenden wir die vorzeichenbehaftete Abstandsfunktion für die Sphäre $\Gamma = \partial B$, die sich durch Auswertung der Funktion

$$\phi(\mathbf{x}) = 0.2 - \sqrt{(x_1 - x_{M_1})^2 + (x_2 - x_{M_2})^2 + (x_3 - x_{M_3})^2}$$

in den jeweiligen Gitterpunkten direkt berechnen läßt. Das Rechengebiet Ω wird mit 64^3 äquidistant verteilten Gitterpunkten diskretisiert und der Transport der Sphäre über den Zeitraum $[0, 20]$ wird mit 800 Zeitschritten der Zeitschrittweite $\delta t = 0.025$ diskretisiert.

Die Anfangskonfiguration und die Ergebnisse dieser Rechnung zu den Zeitenpunkten $t = 5, 10, 15$ und 20 sind in Abbildung 5.1 dargestellt. Hieraus läßt sich gut erkennen, dass die freie Oberfläche zu den Zeitpunkten $t \in \{0, 20\}$ und $t \in \{5, 15\}$ jeweils nahezu identisch ist, wie aufgrund von (5.1) idealerweise zu erwarten wäre. Weiterhin veranschaulicht diese Testrechnung, dass der Einfluss von numerischen Diffusionsartefakten hinsichtlich der diskretisierten Transportgleichung trotz starker Deformierung der Sphäre und moderater Gitterauflösung nur sehr gering ist.

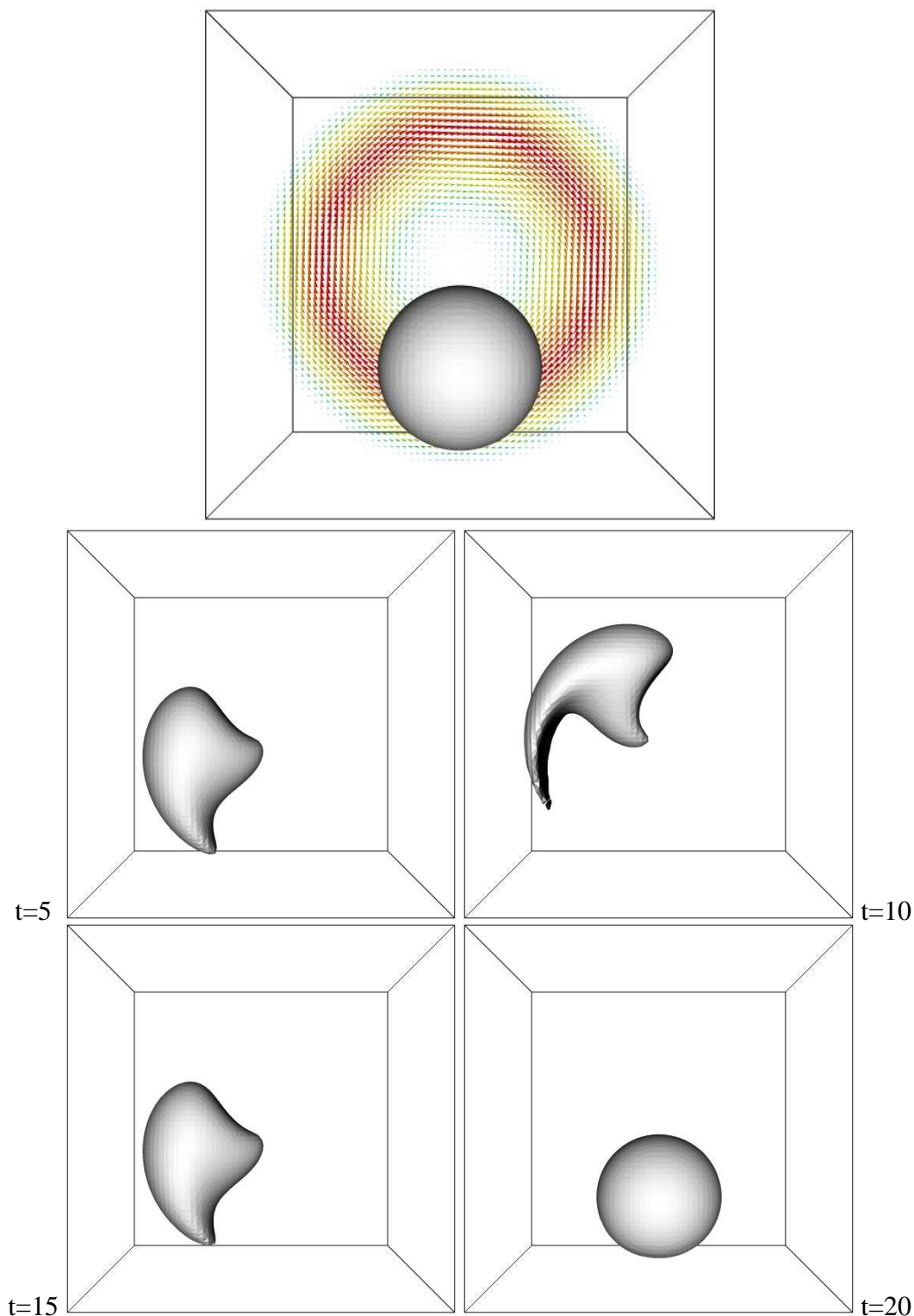


Abbildung 5.1: Advektion einer Sphäre durch ein vorgegebenes divergenzfreies Rotationsfeld. Das oberste Bild zeigt die Anfangskonfiguration, die unteren vier Bilder zeigen die chronologische Abfolge des Transportes der Sphäre zu den Zeiten $t=5$, $t=10$, $t=15$, $t=20$.

5.2 Die Reinitialisierung für eine komplexe Oberfläche

Als zweites numerisches Experiment untersuchen wir den Effekt der Reinitialisierung auf die freie Oberfläche. Hierzu betrachten wir analog zu [85] das würfelförmige Gebiet $\Omega = (-1, 1)^3$ und die skalare Funktion

$$\phi(\mathbf{x}) = \|(1 + 0.2g(\mathbf{x})) \mathbf{x}\| - 0.5, \quad \mathbf{x} \in \Omega$$

wobei $g(\mathbf{x}) = \cos(10x) \cos(10y) \cos(10z)$ für $\mathbf{x} = (x, y, z) \in \Omega$. Das Nullmengeniveau von ϕ und die Konturlinien von ϕ entlang der Mittelschnittebene $\{z = 0\}$ sind auf dem linken Bild in Abbildung 5.2 zu sehen. Hierbei ist ϕ offensichtlich keine Abstandsfunktion, da die Konturlinien auf der Mittelschnittebene nicht äquidistant zueinander verlaufen.

Das Rechengebiet Ω wird wie im vorigen Experiment mit einer Auflösung von 64^3 äquidistant verteilten Gitterpunkten diskretisiert. Die iterative Anwendung der Reinitialisierungsgleichung (3.10) mit insgesamt 128 Iterationen führt, wie zu erwarten, zu einer approximierten globalen vorzeichenbehafteten Abstandsfunktion $\tilde{\phi}$, die im rechten Bild von Abbildung 5.2 dargestellt ist. Der optische Vergleich zwischen dem Nullmengeniveau von ϕ und $\tilde{\phi}$ zeigt hier keine substantielle Änderung der Lage des Nullmengeniveaus. Um den feinskaligen Unterschied graphisch hervorzuheben, ist in Abbildung 5.3 im linken Bild eine Überlagerung beider Isofächen dargestellt. Hierbei gehören die grauen Flächenbereiche zur ursprünglichen Funktion ϕ und die schwarzen Flächenbereiche zu der aus ϕ generierten Abstandsfunktion $\tilde{\phi}$. Man beobachtet, dass die Reinitialisierung in stark gekrümmten Bereichen der freien Oberfläche verstärkt dazu neigt numerische Diffusion zu induzieren. Offensichtlich wird die freie Oberfläche dort, im Rahmen einer Gittermaschenweite, künstlich geglättet. Dieses Glättungsverhalten wird nochmals im rechten Bild verdeutlicht. Hierbei entspricht die durchgezogene schwarze Linie der ursprünglichen Funktion ϕ und die gestrichelte Linie der daraus generierten Abstandsfunktion $\tilde{\phi}$. Hier ist klar ersichtlich, dass die Krümmung im Rahmen einer Gittermaschenweite eine Glättung erfährt. Deshalb gibt es insbesondere bei Langzeitrechnungen, in denen in jedem Zeitschritt reinitialisiert wird, empfindliche Probleme mit der Masseerhaltung. Aus diesem Grund werden wir für Langzeitrechnungen zusätzlich eine Picard-Fixpunktiteration durchführen, die eine globale Volumenerhaltung garantiert. Der Einfluss dieser zusätzlichen Fixpunktiteration auf die Konvergenzeigenschaften der Krümmung und der Volumenerhaltung wird im nächsten Abschnitt untersucht.

An dieser Stelle sei angemerkt, dass für herkömmliche Zweiphasensimulationen eine Abstandsfunktion nur in einer Umgebung der freien Oberfläche notwendig ist, um die Auswertung der Heaviside-Funktion und des Delta-Funktional beidseitig der freien Oberfläche symmetrisch zu gestalten. Dies gestattet eine substantielle Reduktion der Iterationsschritte für die Reinitialisierung auf fünf bis sieben Iterationen, was wiederum eine geringere Störung der freien Oberfläche zur Folge hat, als in Abbildung 5.3 dargestellt ist.

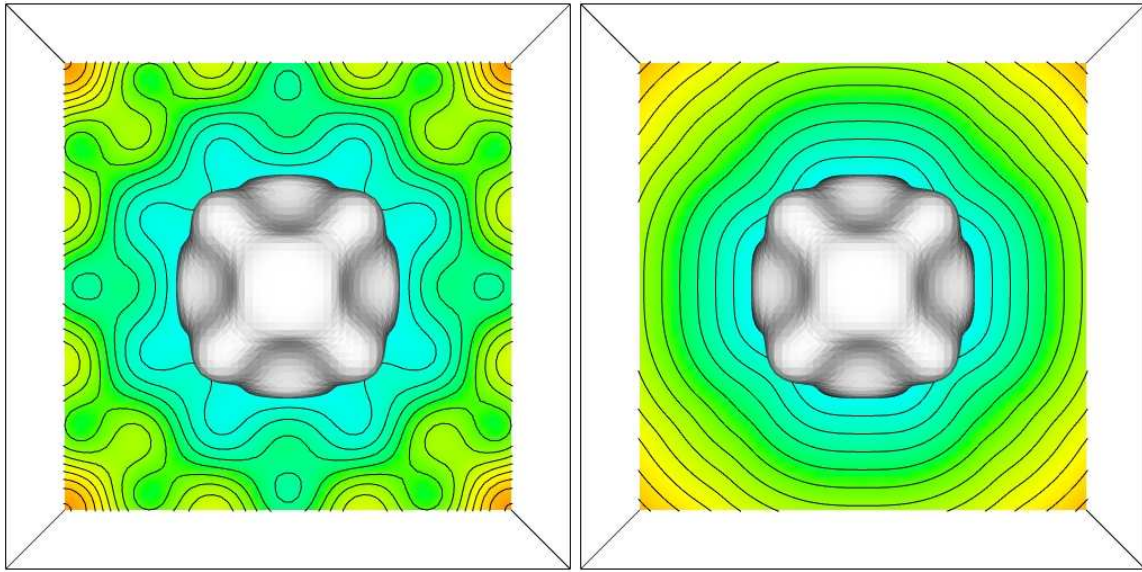


Abbildung 5.2: Nullmengenisofläche und Mittelschnittebene mit dazugehörigen Konturlinien der Level-Set Funktion vor der Reinitialisierung (links) und danach (rechts).

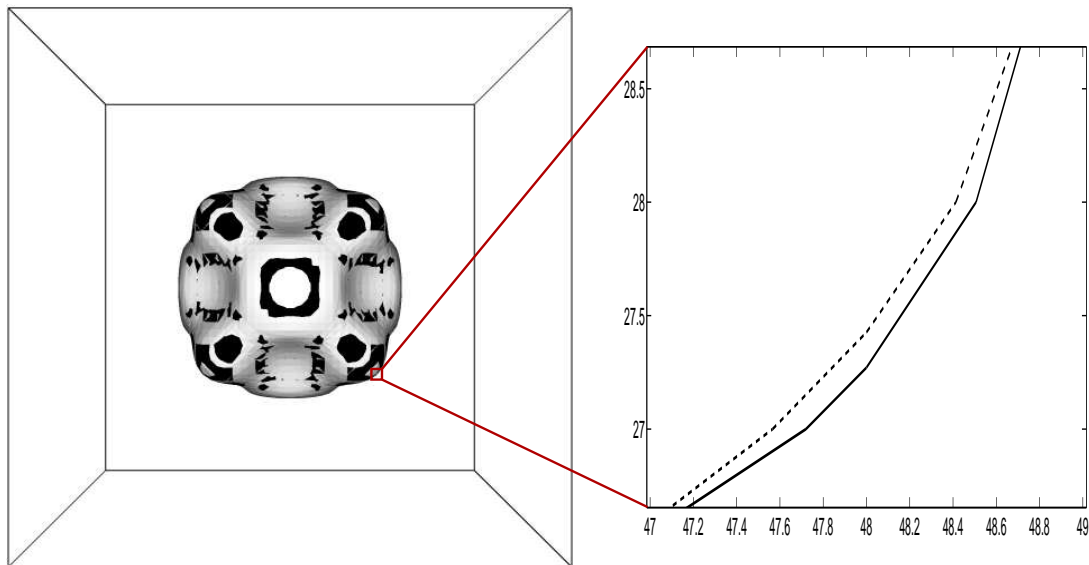


Abbildung 5.3: Das linke Bild zeigt die übereinandergelegten Isoflächen beider Funktion zum Niveauwert Null. Die graue Isofläche gehört zur Funktion ϕ und die schwarzen Flächenbereiche zu der generierten Abstandsfunktion $\tilde{\phi}$, die sich über die ursprüngliche legt. Das rechte Bild ist die Vergrößerung des Schnittes der Isoflächen mit dem kleinen roten Quadrat. Hierbei ist die durchgezogene Linie zur ursprünglichen Funktion ϕ und die gestrichelte Linie zur generierten Abstandsfunktion $\tilde{\phi}$.

5.3 Masseerhaltung und Konvergenz der Krümmung

Im Folgenden untersuchen wir die Konvergenzeigenschaften des numerischen Verfahrens bezüglich Masse und Krümmung am Testfall einer advektierten Einheitssphäre. Da die analytische Krümmung der Einheitssphäre bekannt ist, können wir die approximierte Krümmung direkt mit der analytischen Lösung vergleichen.

Hierzu betrachten wir das Gebiet $\Omega = (0, 4)^3$ welches wir mit einer Sequenz von uniform verfeinerten äquidistanten Gittern diskretisieren. Desweiteren verwenden wir periodische Randbedingungen auf $\partial\Omega$ und ein stationäres, global konstantes Geschwindigkeitsfeld

$$u(x, y, z) = 1, \quad v(x, y, z) = 1, \quad w(x, y, z) = 0.$$

Als Anfangsbedingung für die Level-Set-Funktion verwenden wir die analytische Abstandsfunktion

$$\phi(x, y, z) = 1 - \sqrt{(x - 2)^2 + (y - 2)^2 + (z - 2)^2} \tag{5.3}$$

deren Niveaumenge zum Wert Null eine Sphäre mit Radius $r = 1$ und Mittelpunkt in $(2, 2, 2)$ beschreibt. Diese Sphäre wird über die Zeit entlang der Diagonalen auf der xy -Ebene transportiert.

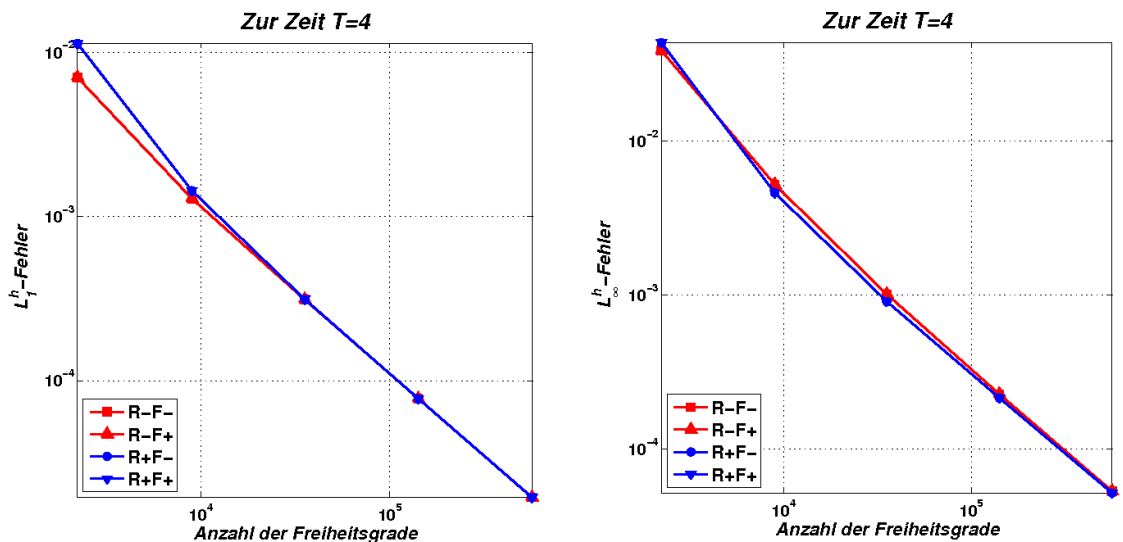


Abbildung 5.4: Fehlerkonvergenz der Krümmungsauswertung einer transportierten Einheitssphäre in der L^1 - und L^∞ -Norm nach einer Periode. $R-$ und $R+$ bezeichnen die ein- bzw. ausgeschaltete Reinitialisierung der Level-Set-Funktion, analoges gilt für $F-$ und $F+$ bezüglich der Picard-Fixpunktiteration.

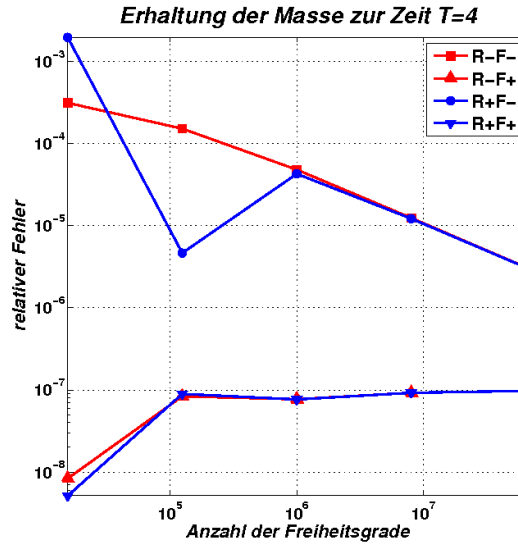


Abbildung 5.5: Konvergenz des relativen Fehlers der Masse. $R-$ und $R+$ bezeichnen die ein- bzw. ausgeschaltete Reinitialisierung der Level-Set-Funktion, analoges gilt für $F-$ und $F+$ bezüglich der Picard-Fixpunktiteration.

In diesem Experiment betrachten wir vier unterschiedliche Konfigurationen unseres numerischen Schemas, um die unterschiedlichen Effekte der Reinitialisierung und der Fixpunktiteration auf das Konvergenzverhalten zu untersuchen. Hierzu führen wir Simulationen mit und ohne Reinitialisierung (4.13) sowie mit und ohne Fixpunktiteration (4.26) durch.

Die Krümmung läßt sich darstellen als

$$\kappa_h(\phi_h)(\vec{x}) = \nabla \cdot \frac{\nabla \phi_h(\vec{x})}{\|\nabla \phi_h(\vec{x})\|}.$$

Desweiteren definieren wir die diskreten Fehlernormen

$$e_{\epsilon}^{\kappa,h}(\mathbf{x}) := \frac{|\kappa_h(\phi_h)(\vec{x}) - \kappa(\phi)(\vec{x})|}{|\kappa(\phi)(\vec{x})|}, \quad \text{mit } \kappa(\phi)(\vec{x}) = \frac{2}{1 - \phi(\vec{x})} \quad (5.4)$$

$$e_{\epsilon,1}^{\kappa,h} := \sum_{\mathbf{x}_i, |\phi_h(\mathbf{x}_i)| < \alpha(\mathbf{x}_i)h} e_{\epsilon}^{\kappa,h}(\mathbf{x}_i), \quad (5.5)$$

$$e_{\epsilon,\infty}^{\kappa,h} := \max_{\mathbf{x}_i, |\phi_h(\mathbf{x}_i)| < \alpha(\mathbf{x}_i)h} e_{\epsilon}^{\kappa,h}(\mathbf{x}_i), \quad (5.6)$$

in der $\epsilon(\mathbf{x})$ -Region gegeben durch $|\phi_h(\mathbf{x}_i)| < \alpha(\mathbf{x}_i)h = \epsilon(\mathbf{x}_i)$, man vergleiche mit (4.4). Das heißt, wir messen den Fehler nicht nur auf der freien Oberfläche, sondern in einer Umgebung der freien Oberfläche. Man beachte, dass wir nicht das geglättete Deltafunktional δ_{ϵ} in die Auswertung der Normen miteinbeziehen, denn δ_{ϵ} wird nur für die Auswertung der Oberflächenspannungskraft benötigt, aber nicht für die Approximation der Krümmung.

Desweiteren messen wir den Fehler bezüglich der Masseerhaltung, indem wir die Differenz zwischen der Masse zur Zeit $t = 4$ und der Anfangsmasse mittels der diskreten Normen

Tabelle 5.1: Relativer Fehler e und Fehlerkonvergenz η der Krümmungsberechnung κ in der L^1 - und L^∞ -Norm und der Masse m nach einer transportierten Periode der Einheitskugel.

$1/h$	dof^h	dof_ϵ^h	$e_{\epsilon,1}^{\kappa,h}$	$\eta_{\epsilon,1}^{\kappa,h}$	$e_{\epsilon,\infty}^{\kappa,h}$	$\eta_{\epsilon,\infty}^{\kappa,h}$	$e_\epsilon^{m,h}$	$\eta_\epsilon^{m,h}$
Reinitialisierung (–) Fixpunkt (–)								
25	15625	2214	7.034 ₋₃	–	3.890 ₋₂	–	3.130 ₋₄	–
50	125000	8928	1.288 ₋₃	1.218	5.236 ₋₃	1.438	1.521 ₋₄	0.346
100	1000000	35376	3.119 ₋₄	1.030	1.015 ₋₃	1.192	4.823 ₋₅	0.552
200	8000000	141440	7.773 ₋₅	1.003	2.271 ₋₄	1.080	1.244 ₋₅	0.651
400	64000000	565768	1.943 ₋₅	1.000	5.320 ₋₅	1.047	3.123 ₋₆	0.664
Reinitialisierung (+) Fixpunkt (–)								
25	15625	2214	1.133 ₋₂	–	4.370 ₋₂	–	1.969 ₋₃	–
50	125000	8932	1.437 ₋₃	1.481	4.622 ₋₃	1.611	4.652 ₋₆	2.909
100	1000000	35380	3.128 ₋₄	1.108	9.112 ₋₄	1.180	4.289 ₋₅	–1.068
200	8000000	141440	7.769 ₋₅	1.005	2.147 ₋₄	1.043	1.220 ₋₅	0.604
400	64000000	565768	1.943 ₋₅	0.999	5.180 ₋₅	1.026	3.105 ₋₆	0.658
Reinitialisierung (–) Fixpunkt (+)								
25	15625	2214	7.034 ₋₃	–	3.890 ₋₂	–	8.338 ₋₉	–
50	125000	8928	1.288 ₋₃	1.218	5.236 ₋₃	1.438	8.365 ₋₈	–1.109
100	1000000	35376	3.120 ₋₄	1.030	1.015 ₋₃	1.192	7.702 ₋₈	0.039
200	8000000	141436	7.773 ₋₅	1.003	2.271 ₋₄	1.080	9.244 ₋₈	–0.087
400	64000000	565768	1.943 ₋₅	1.000	5.320 ₋₅	1.047	9.727 ₋₈	–0.024
Reinitialisierung (+) Fixpunkt (+)								
25	15625	2214	1.131 ₋₂	–	4.373 ₋₂	–	5.157 ₋₉	–
50	125000	8932	1.437 ₋₃	1.479	4.623 ₋₃	1.611	8.952 ₋₈	–1.373
100	1000000	35380	3.129 ₋₄	1.107	9.113 ₋₄	1.180	7.659 ₋₈	0.075
200	8000000	141436	7.769 ₋₅	1.005	2.147 ₋₄	1.043	9.238 ₋₈	–0.090
400	64000000	565768	1.943 ₋₅	0.999	5.180 ₋₅	1.026	9.725 ₋₈	–0.024

$$m_\epsilon^{h,t} := \sum_{\mathbf{x}_i, |\phi_h(\mathbf{x}_i)| < \alpha(\mathbf{x}_i)h} H^\epsilon(\phi_h^t(\mathbf{x}_i)) h^3, \quad e_\epsilon^{m,h} := \frac{|m_\epsilon^{h,t} - m_\epsilon^{h,0}|}{|m_\epsilon^{h,0}|}, \quad (5.7)$$

bestimmen. Die entsprechenden algebraischen Konvergenzraten η werden mittels

$$\eta := -\frac{\log \frac{e_\epsilon^{2h}}{e_\epsilon^h}}{\log \frac{\text{dof}_{2h}}{\text{dof}_h}} \quad (5.8)$$

bestimmt.

Da wir ein absolutes Abbruchkriterium von $e_\epsilon^{m,h} < 10^{-7}$ in der Fixpunktiteration für die Masse verwenden, sagen wir einen konstanten Fehler der Größe e_ϵ^m für die Masseerhaltung voraus, wenn die Fixpunktiteration eingesetzt wird. Aufgrund des eingesetzten konstanten Geschwindigkeitsfeldes wird in diesem Experiment die vorzeichenbehaftete Abstandseigenschaft der Level-Set-Funktion nur leicht über die Zeit gestört, so dass wir sehr ähnliche Ergebnisse für die Fälle mit und ohne Reinitialisierung voraussagen.

Die Ergebnisse dieser numerischen Studie sind in Tabelle 5.1 und den Abbildungen 5.4 und 5.5 aufgeführt. Die Konvergenzraten η_ϵ^m sind bezüglich der gesamten Anzahl an Gitterzellen dof und die Konvergenzraten η_ϵ^κ sind bezüglich der Anzahl an Gitterzellen dof_ϵ in

der $\epsilon(\mathbf{x})$ -Region ermittelt worden. Da $\text{dof}_\epsilon \approx h^{-2}$ gilt, ist eine klassische Konvergenz zweiter Ordnung vorhanden, wenn $\eta = 1$ ist. Analog erhalten wir eine klassische Konvergenz zweiter Ordnung für die Masse falls $\eta = \frac{2}{3}$, da hier $\text{dof} \approx h^{-3}$ gilt.

An den aufgeführten Zahlen in Tabelle 5.1 und den Graphen in Abbildung 5.4 und Abbildung 5.5 können wir deutlich das optimale Konvergenzverhalten von zweiter Ordnung unseres numerischen Schemas beobachten. Wie zu erwarten, ist der Fehler bezüglich der Masse konstant $< 10^{-7}$ falls die Fixpunktiteration mit dem absoluten Abbruchkriterium verwendet wird, das heißt, wir erhalten eine Rate $\eta_\epsilon^m \approx 0$ aufgrund des absoluten Abbruchkriteriums. Die Reinitialisierung (R+) hat nahezu keinen Effekt auf das Konvergenzverhalten, da in diesem Testfall die Abstandseigenschaft der Level-Set-Funktion aufgrund der konstanten Advektionsgeschwindigkeit über die Zeit gut erhalten bleibt.

5.4 Konvergenzanalyse einer aufsteigenden Blase mit Oberflächenspannung

Das Konvergenzverhalten des Zweiphasenlösers mit Oberflächenspannung untersuchen wir nun am Beispiel einer aufsteigenden Blase. Hierzu betrachten wir eine sphärische Blase aus einem Fluid, welches leichter ist als das umgebende Fluid. Aufgrund der Gravitationskraft werden Auftriebskräfte induziert, die zusammen mit der Oberflächenspannungskraft die Deformation der Blase im zeitlichen Verlauf bestimmen. Daher wird hier die Einbindung der Oberflächenspannungskraft über das geglättete Deltafunktional δ_ϵ in unserem numerischen Verfahren mitbetrachtet. Desweiteren nimmt die Deformation der Blase über die Zeit immer mehr zu, so dass hier eine Reinitialisierung der Level-Set-Funktion in jedem Zeitschritt durchgeführt werden muss, um die notwendige vorzeichenbehaftete Abstandsfunktion über die Zeit zu erhalten. Um das Problem der Masseerhaltung, bedingt durch den Transport der Level-Set Funktion und deren Reinitialisierung, in den Griff zu bekommen, verwenden wir die globale Fixpunktiteration für die Masse. Zum Vergleich geben wir auch die Ergebnisse an, welche ohne Fixpunktiteration für die Masse berechnet wurden.

Wir antizipieren eine Konvergenz erster Ordnung nahe der freien Oberfläche aufgrund der Glättung des Delta Funktionals, welche von erster Ordnung ist. Außerhalb der ϵ -Umgebung erwarten wir eine Approximation zweiter Ordnung, da dort das geglättete Delta-Funktional nicht aktiv ist. Insgesamt erwarten wir eine globale Konvergenzordnung nahe bei zwei, da die freie Oberfläche nur eine Untermannigfaltigkeit des vollständigen Rechengebietes darstellt.

Die physikalischen Eigenschaften der betrachteten Fluide sind in Tabelle 5.2 aufgeführt. Die Anfangsbedingung für die Level-Set Funktion ϕ entspricht der einer Kugel mit Mittelpunkt $(0.075m, 0.05m, 0.075m)$ und Radius $r = 0.025m$. Das Rechengebiet hat die Ausmaße $\Omega = (0, 0.15m)^3$ welches wir mit drei uniform verfeinerten äquidistanten Gittern der Maschenweite $h = 1/37, 1/73, 1/145$ diskretisieren und dabei jeweils die dazugehörige äquidistante Zeitschrittweite $\delta t_h = 1.00_{-3}, 2.50_{-4}, 6.25_{-5}$ über den gesamten Zeitraum verwenden. Da eine analytische Lösung zu diesem Problem nicht bekannt ist, verwenden wir als Referenzlösung die Lösung des Gitters mit der Maschenweite $1/289$ und der Zeitschrittweite $\delta t = 1.5625_{-5}$, um die Approximationsgüte unseres numerischen Verfahren zu ermitteln.

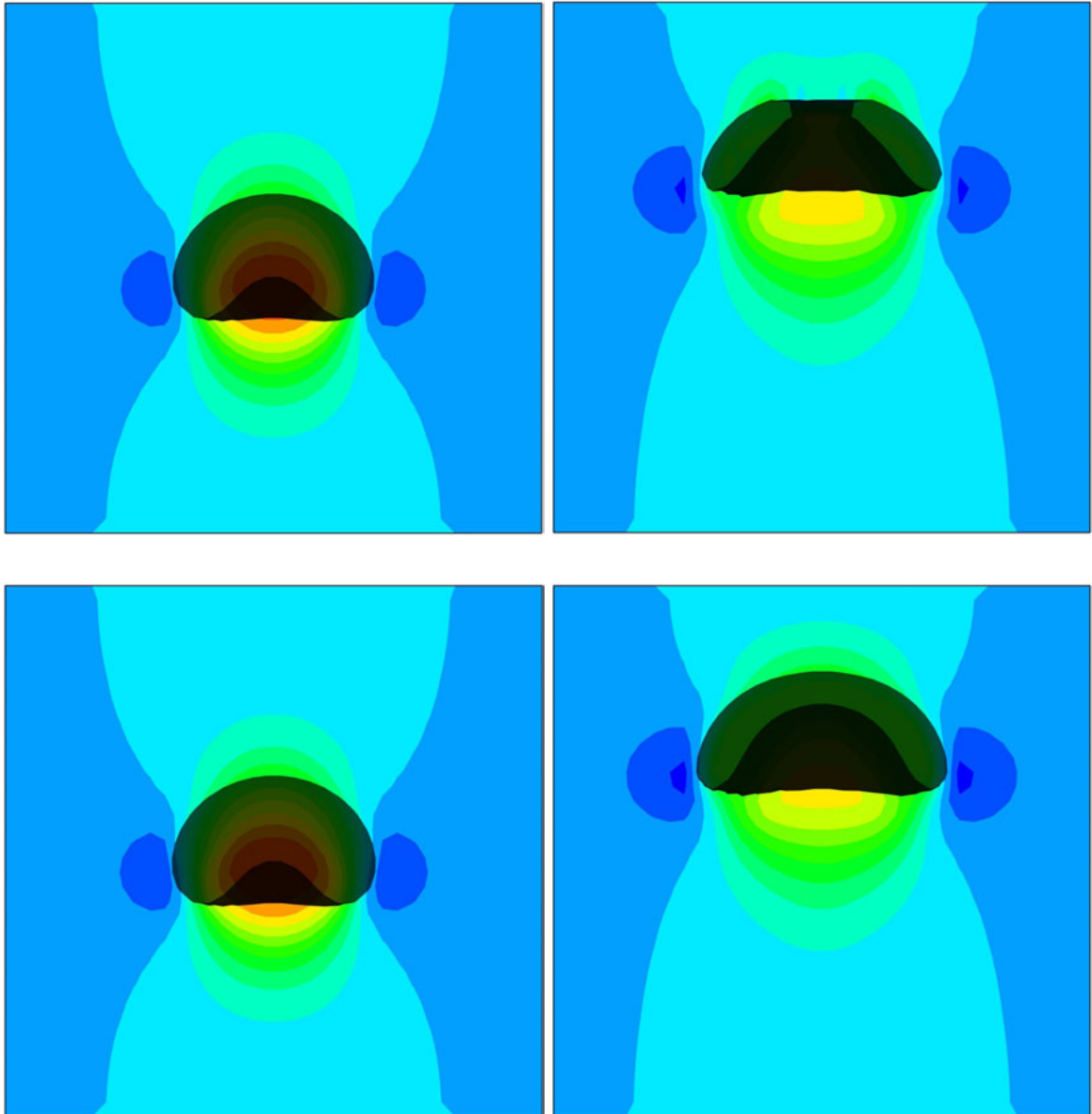


Abbildung 5.6: Aufsteigende Blase ohne (obere Zeile) und mit (untere Zeile) Fixpunktiteration für verbesserte Masseerhaltung zur Zeit $t = 0.1s$ (linke Spalte) und $t = 0.15s$ (rechte Spalte) für $h = 1/37$. Dargestellt ist die semi-transparente Isofläche der Level-Set-Funktion und ein Konturplot der v -Geschwindigkeitskomponente entlang der zentralen Schnittebene.

Tabelle 5.2: Materialparameter für die Simulation der im Fluid aufsteigenden Blase.

Blase:	$\mu_1 = 2.5_{-4} \text{kg/ms}, \rho_1 = 1.0_0 \text{kg/m}^3$	Oberflächenspannung:	$\sigma = 5.0_{-3} \text{N/m}$
umgebendes Fluid:	$\mu_2 = 5.0_{-4} \text{kg/ms}, \rho_2 = 1.0_1 \text{kg/m}^3$	Randbedingungen:	slip

Hierfür benötigen wir die Definition

$$E_h^{\text{ref}}(\psi_h) := I_h^{\text{ref}}\psi_h - \psi_{\text{ref}},$$

wobei I_h^{ref} die trilineare Interpolation der Daten des Gitters mit der Maschenweite h auf das Referenzgitter "ref" der feinsten Maschenweite darstellt, sowie die Normen

$$\|\psi_{\text{ref}}\|_{L^2} := \frac{1}{N_{\text{ref}}} \left(\sum_{i=1}^{N_{\text{ref}}} |\psi_{\text{ref}}(\vec{x}_i)|^2 \right)^{1/2}, \quad \|\psi_{\text{ref}}\|_{L^\infty} := \max_{i=1, \dots, N_{\text{ref}}} |\psi_{\text{ref}}(\vec{x}_i)|,$$

und den Fehler

$$e_p^{\psi, h} := \frac{\|E_h^{\text{ref}}(\psi_h)\|_{L^p}}{\|\psi_{\text{ref}}\|_{L^p}}. \quad (5.9)$$

Wir betrachten in diesem Experiment die Konvergenz der Geschwindigkeiten u_h, v_h, w_h , der Level-Set-Funktion ϕ_h und des Druckes p_h . Hierzu bestimmen wir die algebraischen Konvergenzraten nach Gleichung (5.8), so dass die klassische Konvergenz zweiter Ordnung angenommen wird, wenn $\eta = \frac{2}{3}$ gilt. Da wir das geglättete Delta-Funktional in unserem numerischen Verfahren nun mitbetrachten, um die Oberflächenspannung in unseren Simulationen mitzuberechnen, können wir nicht eine globale Konvergenzordnung von $\eta = \frac{2}{3}$ erwarten. Aber wir antizipieren eine Ordnung zwischen $\frac{1}{3}$ und $\frac{2}{3}$. Je feiner das Gitter wird, desto näher wird η an $\frac{2}{3}$ liegen, aufgrund des kleineren Verhältnisses von Oberfläche zu Volumen, und damit des kleineren Einflusses des geglätteten Delta-Funktional auf die gesamte Approximationsgüte. Die gemessenen relativen Fehler (5.9) und entsprechenden Konvergenzraten (5.8) sind in Tabelle 5.3 und 5.4 aufgeführt. Von den dargestellten Zahlen können wir unser vorhergesagtes Konvergenzverhalten ablesen. Wir beobachten Konvergenzraten η zwischen $\frac{1}{3}$ und $\frac{2}{3}$, die sich mit feiner werdender Gittermaschenweite für die Geschwindigkeiten in der L^2 -Norm und der L^∞ -Norm den Wert $\frac{2}{3}$ annähern. Desweiteren sind die Ergebnisse für die u und w Geschwindigkeitskomponenten identisch aufgrund der Symmetrie des betrachteten

Tabelle 5.3: Konvergenzstudie für die aufsteigende Blase zur Zeit $t = 0.075s$.

$1/h$	$e_\infty^{u,h}$	$\eta_\infty^{u,h}$	$e_2^{u,h}$	$\eta_2^{u,h}$	$e_\infty^{v,h}$	$\eta_\infty^{v,h}$	$e_2^{v,h}$	$\eta_2^{v,h}$	$e_\infty^{w,h}$	$\eta_\infty^{w,h}$	$e_2^{w,h}$	$\eta_2^{w,h}$
Fixpunkt (-)												
37	3.92 ₋₁	—	2.08 ₋₁	—	1.13 ₋₁	—	1.09 ₋₁	—	3.92 ₋₁	—	2.08 ₋₁	—
73	2.09 ₋₁	0.31	8.47 ₋₂	0.44	7.67 ₋₂	0.19	4.92 ₋₂	0.39	2.09 ₋₁	0.31	8.47 ₋₂	0.44
145	6.94 ₋₂	0.54	2.60 ₋₂	0.57	3.52 ₋₂	0.38	1.59 ₋₂	0.55	6.94 ₋₂	0.54	2.60 ₋₂	0.57
Fixpunkt (+)												
37	3.87 ₋₁	—	2.04 ₋₁	—	1.20 ₋₁	—	1.11 ₋₁	—	3.87 ₋₁	—	2.04 ₋₁	—
73	2.06 ₋₁	0.31	8.35 ₋₂	0.44	6.96 ₋₂	0.27	5.01 ₋₂	0.39	2.06 ₋₁	0.31	8.35 ₋₂	0.44
145	6.76 ₋₂	0.54	2.56 ₋₂	0.57	2.97 ₋₂	0.41	1.60 ₋₂	0.56	6.76 ₋₂	0.54	2.56 ₋₂	0.57

Tabelle 5.4: Konvergenzstudie für die aufsteigende Blase zur Zeit $t = 0.075s$.

$1/h$	$e_\epsilon^{m,h}$	$\eta_\epsilon^{m,h}$	$e_\infty^{\phi,h}$	$\eta_\infty^{\phi,h}$	$e_2^{\phi,h}$	$\eta_2^{\phi,h}$	$e_\infty^{p,h}$	$\eta_\infty^{p,h}$	$e_2^{p,h}$	$\eta_2^{p,h}$
Fixpunkt (-)										
37	4.621 ₋₂	—	3.14 ₋₂	—	1.89 ₋₂	—	6.51 ₋₁	—	1.14 ₋₂	—
73	1.158 ₋₂	0.679	1.27 ₋₂	0.44	6.90 ₋₃	0.50	5.89 ₋₁	0.05	4.90 ₋₃	0.41
145	2.402 ₋₃	0.764	4.66 ₋₃	0.49	2.19 ₋₃	0.56	4.26 ₋₁	0.16	1.92 ₋₃	0.45
Fixpunkt (+)										
37	4.035 ₋₅	—	3.49 ₋₂	—	1.48 ₋₂	—	6.52 ₋₁	—	1.04 ₋₂	—
73	2.343 ₋₅	0.267	1.32 ₋₂	0.48	6.32 ₋₃	0.42	5.90 ₋₁	0.05	4.55 ₋₃	0.41
145	6.541 ₋₆	0.620	4.46 ₋₃	0.53	2.15 ₋₃	0.52	4.27 ₋₁	0.16	1.82 ₋₃	0.45

Problems. Die relativen Fehler und Konvergenzraten für die Level-Set-Funktion ϕ korrespondieren sehr gut mit denen der Geschwindigkeiten. Wieder beobachten wir hier, dass η zwischen $\frac{1}{3}$ und $\frac{2}{3}$ liegt und sich dem Wert $\frac{2}{3}$ auf den feiner werdenden Gittern nähert, was einer Konvergenz zweiter Ordnung für die Level-Set Funktion sehr nahe kommt. Der positive Effekt der Fixpunktiteration auf die Masseerhaltung ist deutlich sichtbar an den gemessenen relativen Fehlern $e_\epsilon^{m,h}$ in Tabelle 5.4. Wir erhalten mit der Fixpunktiteration in etwa eine Genauigkeit von 10^{-5} in der Masseerhaltung auf allen Gittern, wobei ohne Fixpunktiteration die Masse nur bis auf 10^{-3} erhalten bleibt. Das Konvergenzverhalten des Druckes p ist im Vergleich zu den Geschwindigkeiten, wie zu erwarten, leicht reduziert. Der Grund hierfür liegt in der Tatsache, dass der Druck entlang der freien Oberfläche unstetig ist.

Als nächstes Testproblem simulieren wir eine im Wasser aufsteigende Luftblase. Zu diesem Zweck betrachten wir die dreidimensionale Version des Modellproblems aus [109, 192]. Das Gebiet und die Anfangskonfiguration der Fluidphasen sind wie folgt gegeben: $\Omega = (0, 0.2)m \times (0, 0.3)m \times (0, 0.2)m$, der Blasenradius ist $r_0 = 1/30m$ und das Zentrum der Blase liegt in $(0.1m, 0.1m, 0.1m)$. Die Gravitationskraft und die Materialparameter sind $\vec{g} = (0, -9.8, 0)m/s^2$, $\sigma = 0.0728N/m$, $\rho_{wasser} = 10^3kg/m^3$, $\rho_{luft} = 1.226kg/m^3$, $\mu_{wasser} = 1.137 \times 10^{-3}kg/ms$ und $\mu_{luft} = 1.78 \times 10^{-5}kg/ms$. An allen Gebietsrändern, das heißt an $\partial\Omega$, haben wir Haft-Randbedingungen gesetzt.

In der oberen Reihe von Abbildung 5.7 ist die zeitliche Entwicklung der dreidimensionalen freien Oberfläche zu den Zeitenpunkten $t = 0.01s, 0.02s, 0.035s, 0.05s$ zu sehen, die auf dem feinsten Gitter mit der Auflösung $160 \times 240 \times 160$ berechnet wurde. In der unteren Reihe von Abbildung 5.7 ist dieser Testfall zusätzlich auf den zwei gröberen Gittermaschenweiten $40 \times 60 \times 40$ und $80 \times 120 \times 80$ berechnet worden, um einen Vergleich zu der feinsten Auflösung durchführen zu können. Zur besseren Vergleichbarkeit der entsprechenden Lösungen haben wir in der unteren Reihe von Abbildung 5.7 die Visualisierung auf die Mittelschnittebene des 3D Gebietes beschränkt. Die Ergebnisse lassen sich bis zum Zeitpunkt $t = 0.035s$ sehr gut mit denen aus [109, 192] vergleichen und die Lösungen auf den drei betrachteten Gittern sind nahezu identisch. Für spätere Zeiten $t > 0.035s$ wird die Blase im Vergleich zu ihrer Ausgangsform stark deformiert und es ist eine deutliche Verbesserung der Lösung aufgrund der Gitterverfeinerung zu erkennen. Allerdings unterscheiden sich die Simulationsergebnisse für $t > 0.035s$ von denen aus [109, 192] aufgrund der Tatsache, dass wir einen dreidimensionalen Aufbau des Problems betrachten, bei dem das Verhältnis zwischen Oberfläche und Volumen kleiner ist als in zwei Dimensionen. Um diese Aussage zu unter-

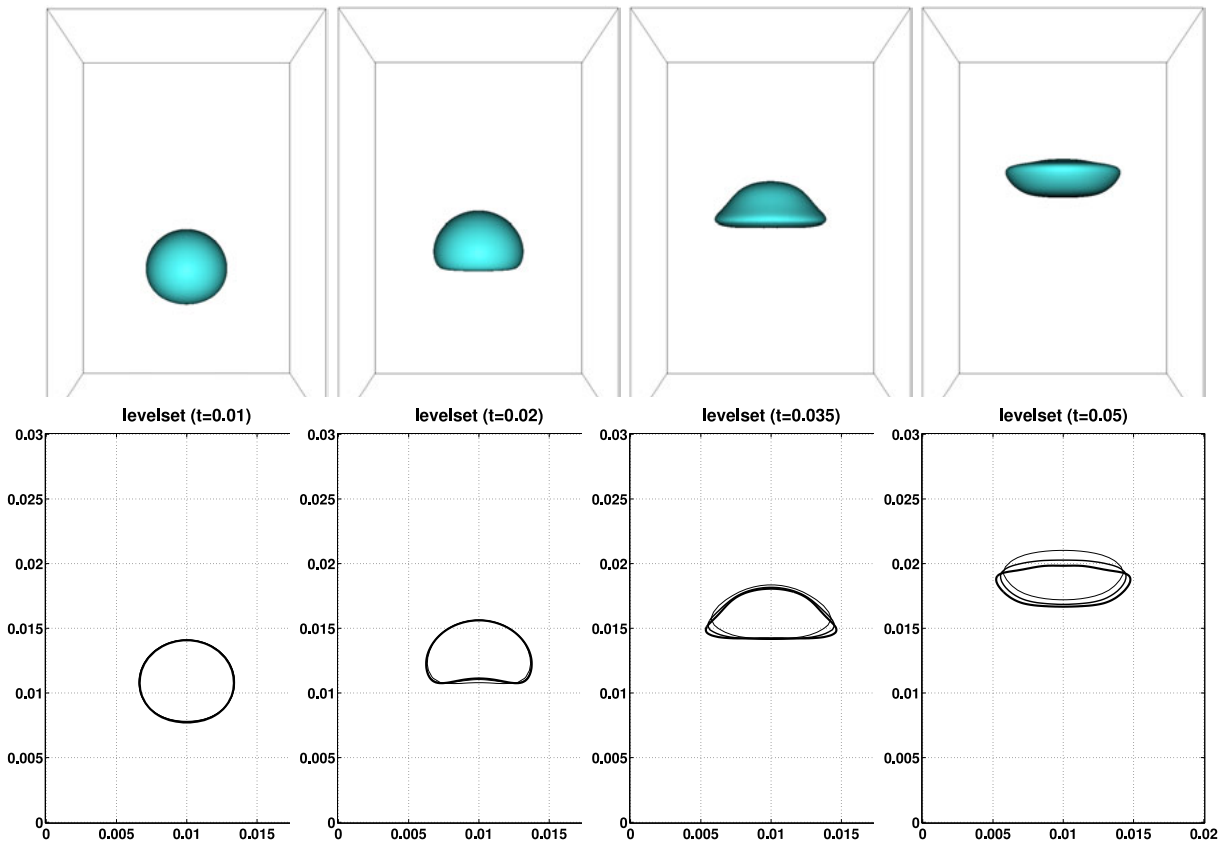


Abbildung 5.7: Im Wasser aufsteigende Luftblase zu den Zeitpunkten $t = 0.01s$, $0.02s$, $0.035s$, und $0.05s$ (von links nach rechts). In der oberen Reihe sind die Oberflächenpositionen für die Gitterweiten $= (1/160, 1.5/240, 1/160)$ abgebildet. In der unteren Reihe sind die Schnitte der freien Oberfläche mit der Mittelebene des Rechengebietes für die Gitterweiten $= (1/40, 1.5/60, 1/40)$, $(1/80, 1.5/120, 1/80)$, und $(1/160, 1.5/240, 1/160)$ abgebildet (die fettgedruckte Linie entspricht der feinsten Auflösung).

mauern, wurde ein quasi-zwei-dimensionales Analogon mit periodischen Randbedingungen in z -Richtung berechnet, das heißt es wurde eine zylindrische Blase betrachtet, bei der die Geschwindigkeitskomponente in z -Richtung gleich Null ist. Die berechneten Ergebnisse sind in Abbildung 5.8 zu sehen.

Ein Vergleich der Ergebnisse aus Abbildung 5.8 mit den zweidimensionalen Simulationsergebnissen aus [109] für die *ghost-fluid*-Methode zeigt eine sehr gute Übereinstimmung bis zum Zeitpunkt $t \leq 0.035s$. Für spätere Zeitpunkte $t > 0.035s$ wird der Masseverlust in der *ghost-fluid*-Methode [109] substantiell und führt somit zu einer Verschlechterung der Qualität der Simulation. Unser numerisches Schema sichert dahingegen die Masseerhaltung und ermöglicht somit auch qualitativ verlässliche Langzeitsimulationen. Das numerische Schema aus [192] sichert ebenfalls die Masseerhaltung, aber die Level-Set-Funktion wird dort nur mit erster Ordnung transportiert, wodurch insbesondere Langzeitsimulationen durch numerische Diffusionsartefakte beeinträchtigt werden. In unserer Approximation verwenden wir ein Transportschema höherer Ordnung. Dieses liefert für $t > 0.035s$ bessere Ergebnisse als

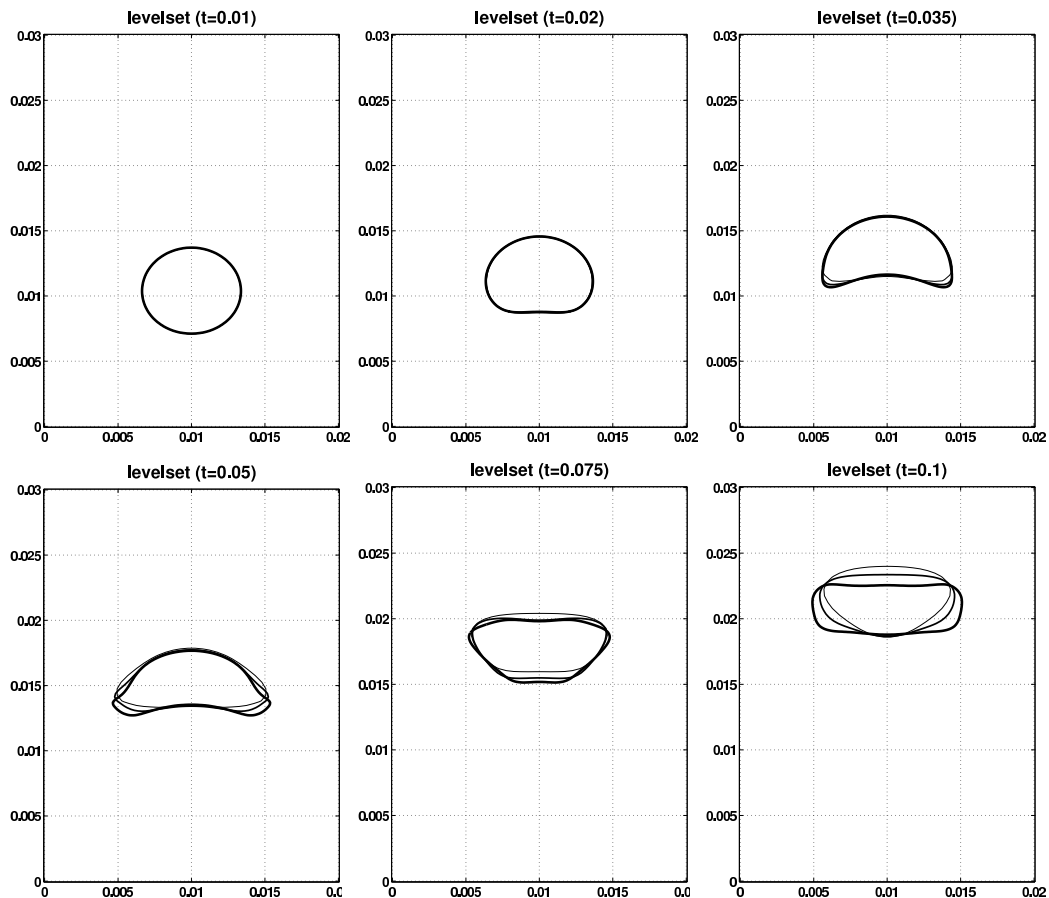


Abbildung 5.8: Aufsteigende quasi 2D Luftblase im Wasser zu den Zeitpunkten $t = 0.01s$, $0.02s$, $0.035s$, $0.05s$, $0.075s$ und $0.1s$. Zu sehen sind die Schnitte zwischen der Oberfläche und der Mittelebene des Rechengebietes für die Gittermaschenweiten $= (1/40, 1.5/60, 1/4)$, $(1/80, 1.5/120, 1/8)$, und $(1/160, 1.5/240, 1/16)$ (fettgedruckte Linien entsprechen der feinsten Auflösung).

in [192].

Zusammenfassend zeigen die Ergebnisse in Abbildung 5.7 und 5.8 die Notwendigkeit von *dreidimensionalen* Simulationen für Probleme aus der Blasen- und Tropfendynamik. Die Blase aus Abbildung 5.7 ist substantiell schneller und erfährt ein anderes Deformationsmuster als ihr zweidimensionales Gegenstück aus Abbildung 5.8. Der Grund hierfür liegt im unterschiedlichen Verhältnis zwischen Volumen und Oberfläche und der unterschiedlich starken Wechselwirkung mit den Gebietsrändern, denn der quasi 2D-Fall entspricht in 3D einem aufsteigenden Luftzylinder.

5.5 Deformationsverhalten von Tropfen in Scherströmungen

Als letztes numerisches Experiment dieses ersten Teils der Arbeit betrachten wir die Deformation eines kleinen Tropfens innerhalb einer Scherströmung. Hierzu wird ein sphäri-

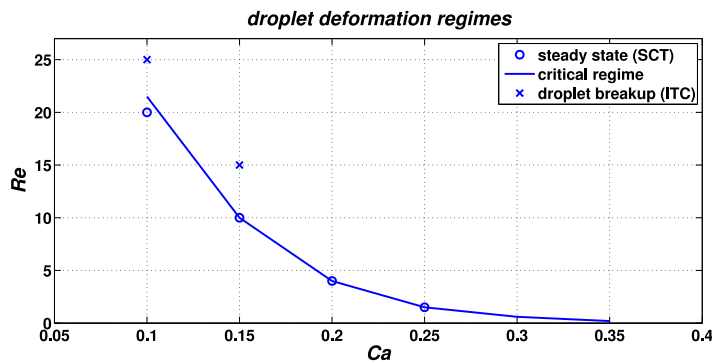


Abbildung 5.9: Phasendiagramm der Tropfendeformation bezüglich der Kapillanzahl Ca und der Reynoldszahl Re . Das stationäre Regime mit konstanter Topologie wird mit ST bezeichnet, das instationäre Regime, bei dem topologische Änderungen auftreten wird mit IT bezeichnet.

scher Tropfen mit dem Radius r_D durch die ihn umgebende Scherströmung in die Länge gezogen [14, 127, 128]. Das Deformationsverhalten hängt von den Materialeigenschaften der Fluide für den Tropfen, der umgebenden Flüssigkeit und der Scherrate ab.

Eine stationäre Lösung mit einer nahezu ellipsenförmig deformierten Tropfengeometrie erhalten wir beispielsweise, wenn die Oberflächenspannungskraft σ groß genug gewählt wird. Wenn hingegen die gewählte Oberflächenspannungskraft zu klein ist, um die entgegengesetzten Kräfte der Scherspannung zu bilanzieren, dann erfährt der Tropfen eine Topologieänderung. Hierbei zerfällt der Tropfen solange in immer kleinere Satellitentropfen, bis die Oberflächenspannungskraft wieder in der Lage ist, für die kleineren Tropfenvolumen die Scherspannungen zu bilanzieren [128].

Im Folgenden bezeichnen wir die Viskosität des umgebenden Fluides mit ν_M und die Viskosität des Tropfens mit ν_D . Der charakteristische dimensionslose Parameter eines solchen Problems ist die Kapillanzahl

$$Ca := \frac{\dot{\gamma} \nu_M r_D}{\sigma}. \quad (5.10)$$

Sie gibt das Verhältnis von Viskositätskräften zu Oberflächenspannungen an den Phasengrenzen von Flüssigkeiten und Gas oder unmischbaren Flüssigkeiten wieder. Das Deformationsverhalten des Tröpfchens lässt sich qualitativ über die Kapillanzahl Ca und die Reynoldszahl Re bestimmen. Gemäß [128] existieren zwei Strömungsbereiche: Ein stationäres Regime (ST), bei dem der Tropfen seine Ausgangstopologie erhält, und ein instationäres Regime (IT), bei dem der Tropfen eine topologische Änderungen erfährt und über die Zeit in entsprechende Satellitentropfen zerfällt, siehe Abbildung 5.9.

Zunächst betrachten wir vier repräsentative Parameterkonfigurationen für das ST-Regime mit Reynoldszahlen, die wenig unterhalb der kritischen Werte liegen, nämlich $(Ca, Re) = (0.25, 1.5)$, $(0.2, 4)$, $(0.15, 10)$, und $(0.1, 20)$. Diese Parameterwahlen wurden auch in [127, 128] im Rahmen einer VOF-Simulation betrachtet. Die finalen stationären Ergebnisse für diese Konfigurationen sind in den Abbildungen 5.10 und 5.11 dargestellt. Diese Ergebnisse sind den Ergebnissen aus [127, 128] sehr ähnlich. Mit steigender Reynoldszahl beobachten

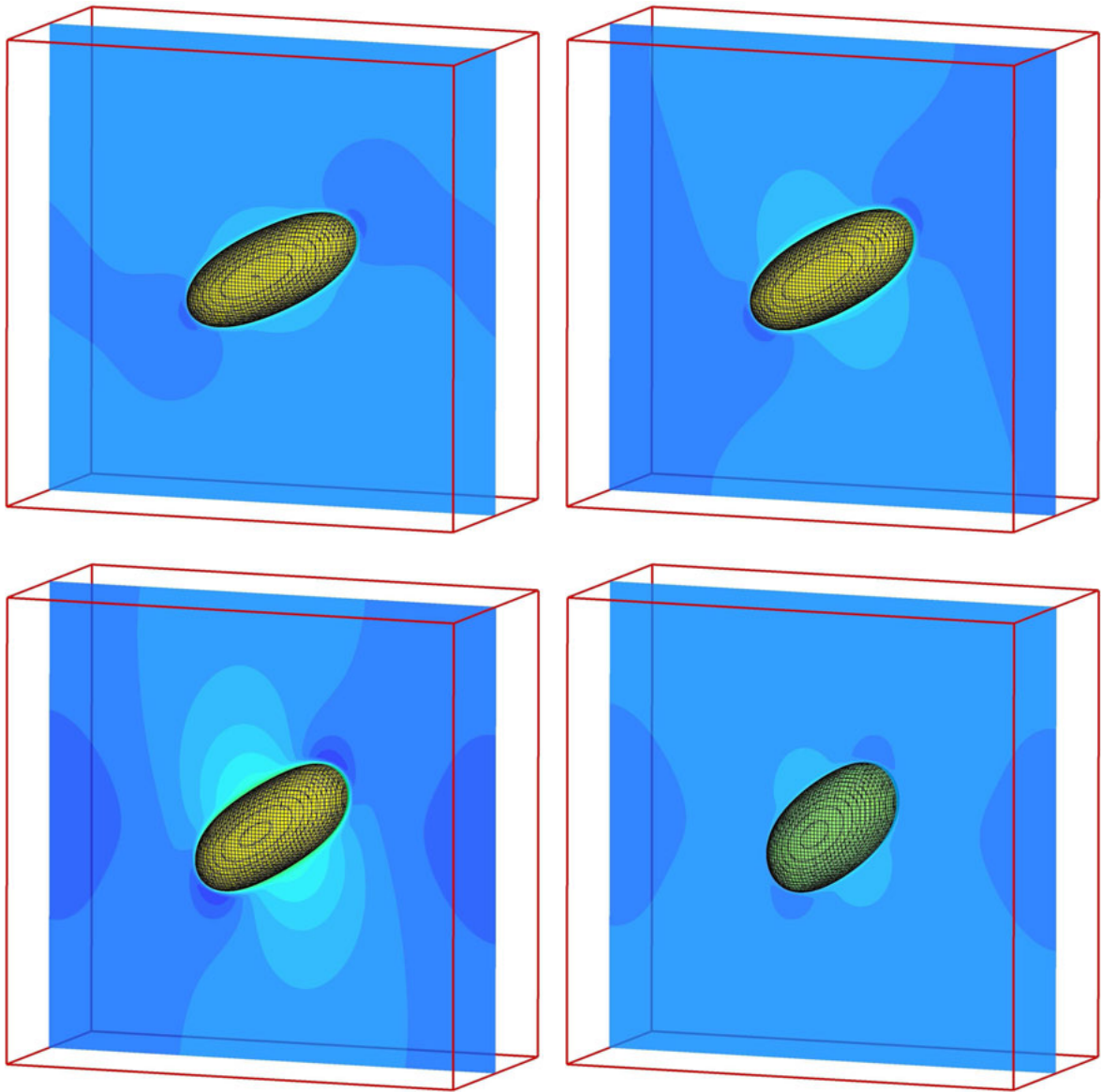


Abbildung 5.10: Stationäre Ergebnisse der Tropfendeformation in einer Scherströmung für die vier Parameterkonfigurationen $(Ca, Re) = (0.25, 1.5)$, $(0.2, 4)$, $(0.15, 10)$, und $(0.1, 20)$ (oben links bis unten rechts). Zu sehen ist die freie Oberfläche im stationären Limit. Das entsprechende Druckfeld ist mit der gleichen Farbskalierung auf allen Bildern mittels einer Mittelschnittebene dargestellt. Die stationären Zustände werden zu den Zeiten $t = 203.34$, $t = 538.41$, $t = 829.68$, und $t = 1649.36$ erreicht (von oben links bis unten rechts). Das Simulationsgebiet ist $2 \times 1 \times 2$ groß und der Tropfen hat einen Radius von $r = 0.25$. Die Gittermaschenweite ist $(2/128, 1/64, 2/128)$.

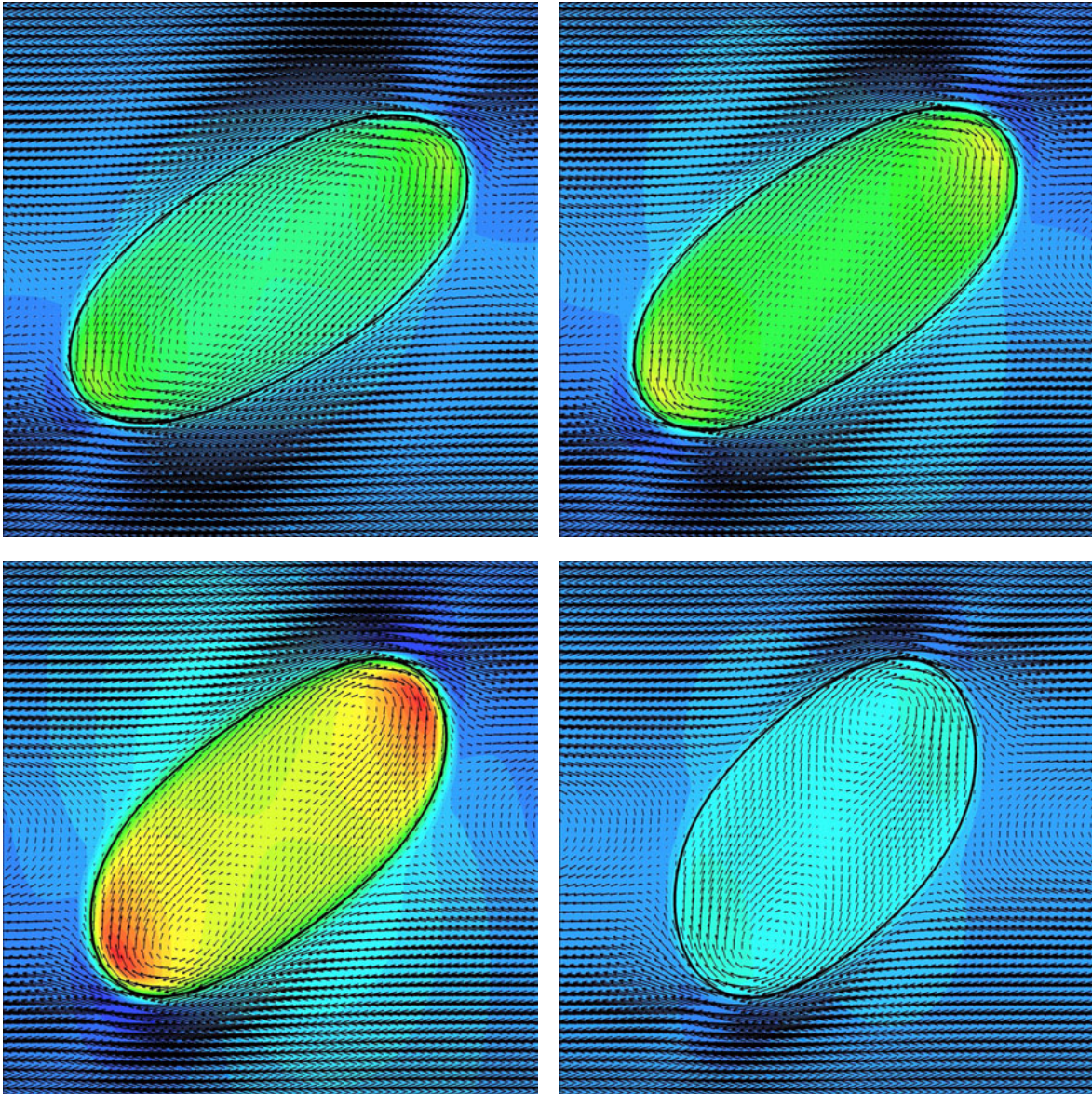


Abbildung 5.11: Stationäre Ergebnisse der Tropfendeformation in einer Scherströmung für vier unterschiedliche Parameterkonfigurationen. Zu sehen sind Vergrößerungen der freien Oberfläche in dem umgebenden Strömungsfeld entlang der Mittelschnittebene. Die entsprechenden Druckfelder sind mit der gleichen Farbskalierung auf allen Bildern mittels einer Mittelschnittebene dargestellt. Die stationären Zustände werden zu den Zeiten $t = 203.34$, $t = 538.41$ (obere Reihe) und $t = 829.68$, $t = 1649.36$ (untere Reihe) erreicht. Das Simulationsgebiet ist $2 \times 1 \times 2$ groß und der Tropfen hat einen Radius von $r = 0.25$. Die Gittermaschenweite ist $(2/128, 1/64, 2/128)$. Für die Parameter $(Ca, Re) = (0.25, 1.5)$, $(0.2, 4)$ (obere Reihe) bildet das Strömungsfeld innerhalb des Tropfens einen einzigen Wirbel aus, wobei für die Parameter $(Ca, Re) = (0.15, 10)$, $(0.1, 20)$ (untere Reihe) das Strömungsfeld innerhalb des Tropfens zwei Wirbel ausbildet.



Abbildung 5.12: Instationäre Ergebnisse der Tropfendeformation in einer Scherströmung für die Parameterkonfiguration $(Ca, Re) = (0.15, 15)$. Zu sehen ist die zeitliche Entwicklung der freien Oberfläche zu $t = 0$, $t = 334.8$, $t = 379.2$, $t = 396.0$, $t = 397.2$, $t = 416.40$ (oben links bis unten rechts). Das Simulationsgebiet ist $(0, 3) \times (0, 1) \times (0, 2)$ groß und der Tropfen hat den Radius $r = 0.25$. Die Gittermaschenweite ist $(3/192, 1/64, 2/128)$.

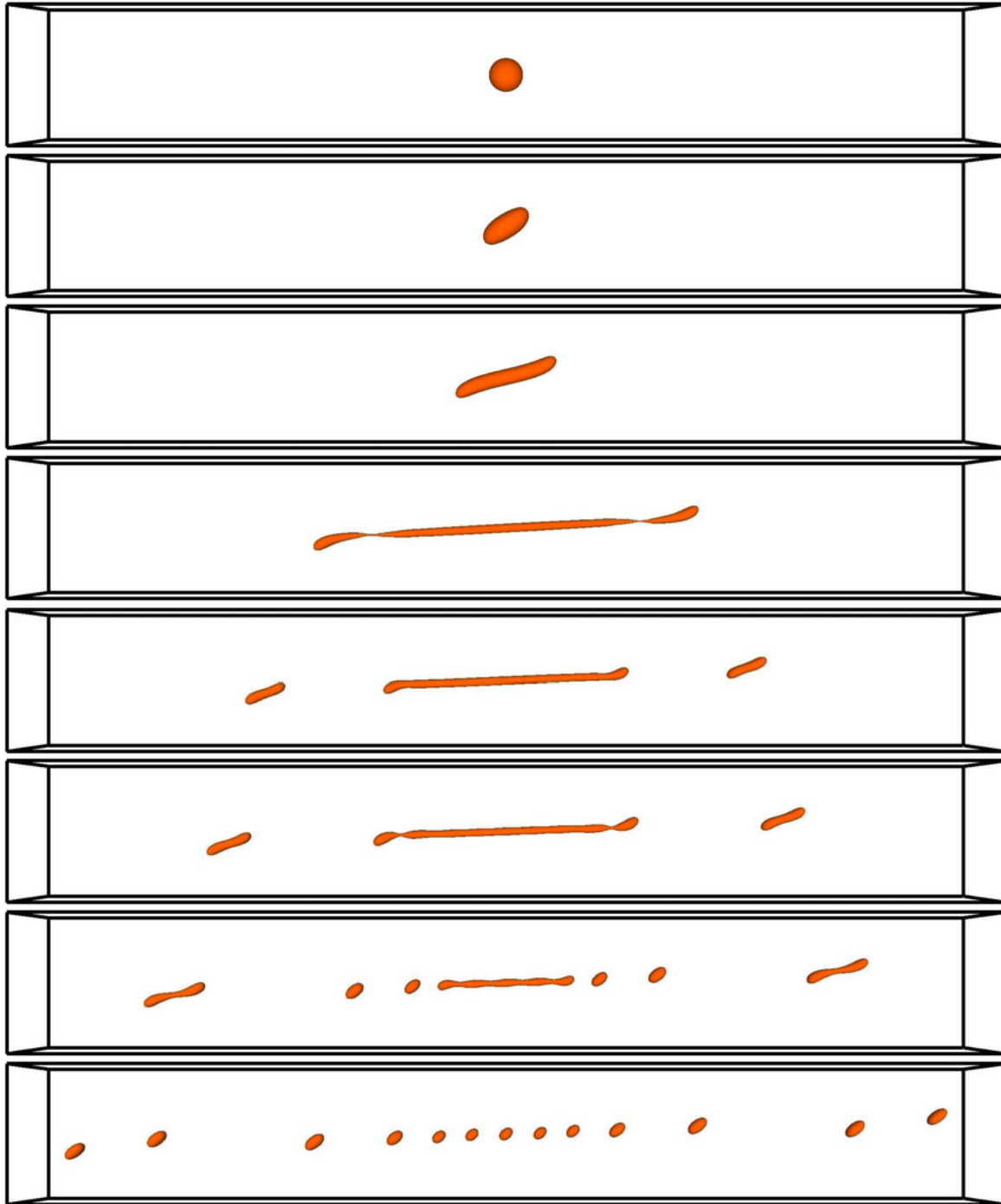


Abbildung 5.13: Instationäre Ergebnisse der Tropfendeformation in einer Scherströmung für die Parameterkonfiguration $(Ca, Re) = (0.1, 25)$. Zu sehen ist die zeitliche Entwicklung der freien Oberfläche zu $t = 0, t = 178.8, t = 238.8, t = 330.0, t = 358.8, t = 373.2, t = 394.8, t = 418.8$ (von oben nach unten). Das Simulationsgebiet ist $(0, 14) \times (0, 1) \times (0, 2)$ groß und der Tropfen hat einen Radius von $r = 0.25$. Die Gittermaschenweite ist $(14/896, 1/64, 2/128)$.

auch wir eine Brechung der Symmetrie des Tröpfchens entlang der Mittelschnittebene und eine stärkere Abweichung der Tröpfchenform von der eines Ellipsoids. Desweiteren entwickelt das Strömungsfeld innerhalb des Tröpfchens multiple Wirbel mit steigender Reynoldszahl. Wir beobachten die Ausbildung je eines Wirbels für die Fälle $(Ca, Re) = (0.25, 1.5)$ und $(Ca, Re) = (0.2, 4)$ und die Ausbildung von je zwei Wirbeln für die Fälle $(Ca, Re) = (0.15, 10)$ und $(Ca, Re) = (0.1, 20)$.

Als letztes numerisches Experiment betrachten wir zwei Konfigurationen des IT-Regimes (siehe Abbildung 5.9). Die Ergebnisse für die Parameter $(Ca, Re) = (0.15, 15)$ sind in Abbildung 5.12 aufgeführt. Hier beobachten wir zunächst eine Dehnung des Tropfens von einer sphärischen hin zu einer Ellipsenform, dann hin zu einer hantelähnlichen Form, bevor er schließlich seine Topologie ändert und in zwei Satellitentropfen zerfällt. Diese zwei Satellitentropfen konvergieren in einen stationären Zustand hinsichtlich ihrer Topologie, weil die Oberflächenspannung jetzt stark genug ist, die Scherkräfte, die auf diese Tropfenmassen wirken, auszugleichen.

Für die Parameterkonfiguration $(Ca, Re) = (0.1, 25)$ sagen wir voraus, dass das Tröpfchen im zeitlichen Verlauf in viele kleine und kleiner werdende Tröpfchen zerfällt, da das Kräftegleichgewicht zwischen der Oberflächenspannungskraft und der Scherspannungskraft sich erst sehr spät nach einer starken Deformationsphase einstellt. Der Zerfallsprozess tritt erst nach einer sehr langen Dehnungsphase des Tröpfchens in eine hantelartige Form ein. Hier lösen sich dann zunächst die Enden ab und bilden drei eher große Satellitentropfen, deren Deformationsverhalten sich ähnlich zu dem Ursprungströpfchen verhält. Die umgebende Scherströmung dehnt im weiteren Verlauf diese drei Tröpfchen wieder zu einer hantelähnlichen Form aus, bis diese sich wiederum in weitere kleinere Satellitentropfen aufteilen. Dieser charakteristische Prozess wiederholt sich solange, bis die Oberflächenspannungskräfte das Deformationsverhalten der Tröpfchen bilanzieren können, das heißt, bis die Oberflächenspannungskräfte stärker sind als die wirkenden Scherkräfte. Dieses vorhergesagte Verhalten ist deutlich in den Konturplots der Abbildung 5.13 zu beobachten.

Teil II

Kopplung mit beweglichen Starrkörpern

Kapitel 6

Generierung und Transport von komplexen Starrkörpergeometrien

In diesem Kapitel beschreiben wir die Umwandlung einer vorgegebenen triangulierten Starrkörpergeometrie in eine auf einem Eulerschen Gitter definierte Level-Set-Funktion. Diese Umwandlung besteht im wesentlichen aus zwei Schritten: erstens aus der maschinengenauen Berechnung des Abstandes eines Punktes zu einem gegebenen Dreieck, zweitens aus der Innen/Außen-Differenzierung des Starrkörpers für die Vorzeichenbestimmung der entsprechenden Level-Set-Werte.

Um schließlich Fluid-Starrkörper-Wechselwirkungen simulieren zu können, muss darüber hinaus ein bis auf Gittergenauigkeit verlustfreier Transport der Level-Set-Geometrie garantiert werden, sodass die Starrkörpereigenschaft zu jedem Zeitpunkt einer Simulation erhalten bleibt. Für solche Problemstellungen bietet sich eine Semi-Lagrangesche Diskretisierungstechnik an. Zu jedem Zeitpunkt wird eine entsprechende Trajektorie mitgeführt, sodass die aktuelle Lage des Starrkörpers über Interpolation aus den maschinengenauen Werten der Anfangsbedingung des Starrkörpers ermittelt werden kann.

6.1 Traditionelle Darstellungstechniken komplexer Geometrien

Eine klassische Darstellungstechnik komplexer Geometrien verläuft über Verfahren zur Erlangung von randangepassten Gittern. Diese Methode ist bis heute ein eigenes Fachgebiet des Wissenschaftlichen Rechnens, da es nach wie vor sehr komplexe und bisher unbeantwortete Fragestellungen beinhaltet. Ein wichtiger Interessenspunkt in der Gittergenerierung ist der zeitliche Aufwand, welcher benötigt wird, um eine Geometrie so zu vernetzen, dass eine stabile Diskretisierung einer PDE ermöglicht wird.

Im Zuge dessen haben sich in der Vergangenheit eine Anzahl unterschiedlicher Gittergenerierungstechniken entwickelt, die im Zusammenhang mit Finite-Elemente-, Finite-Volumen- und Finite-Differenzen-Lösern eingesetzt werden. Zu den populärsten Methoden zählen sicher die Methoden zur Generierung von randangepassten, unstrukturierten sowie block-strukturierten Gittern. Für eine gegebene komplexe Geometrie versucht man ein Git-

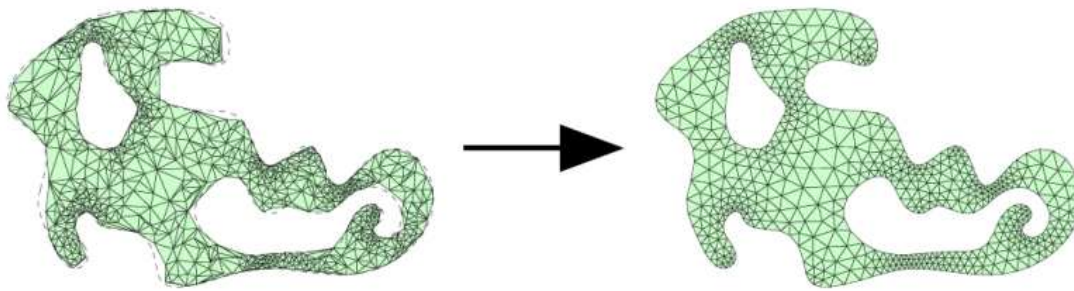


Abbildung 6.1: Gitterglättung einer komplexen Geometrie.

ter zu erzeugen, welches weitestgehend sämtliche Skalen- und Krümmungsvariationen der Geometrie einfängt.

Das Problem hierbei ist, dass eine erste Gittergenerierung in der Regel sehr zeitaufwendig ist. Weiterhin ist diese erste Gitterversion meistens so verzerrt, dass in weiteren Präprozessing-Schritten elaborierte Gitterglättungsalgorithmen eine für das Simulationsprogramm gutmütige (d.h. glatte) Verteilung der Gitterknotenpunkte produzieren sollen, wie in Abbildung 6.1¹ dargestellt.

Eine weitere im Bereich Finiter-Volumen-Techniken oft eingesetzte Gittervernetzungs-technik ist die *Chimera*- oder auch *Composite-Grids*-Methode. Hier wird die Geometrie zunächst in einfachere Teilobjekte zerlegt, welche dann einzeln mit einem Gitter versehen werden. Beim Zusammenfügen aller Teilgitter entstehen Gitter-Überlappungszonen zwischen den Teilobjekten, wie in Abbildung 6.2² dargestellt ist. In diesen Überlappungsbereichen müssen dann die zu berechnenden Strömungsgrößen mit geeigneten Interpolationsoperatoren auf das jeweilige Gitter transferiert werden.

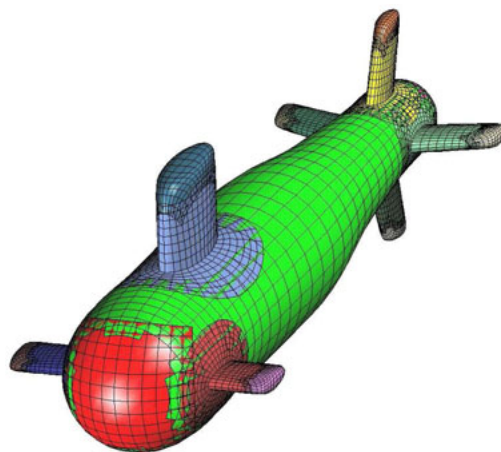


Abbildung 6.2: Gittergenerierung für ein U-Boot mittels der Composite-Grids-Methode.

¹Das Bild stammt von Per-Olof Persson [3].

²Das Bild stammt von Bill Henshaw und Kyle Chand [4].

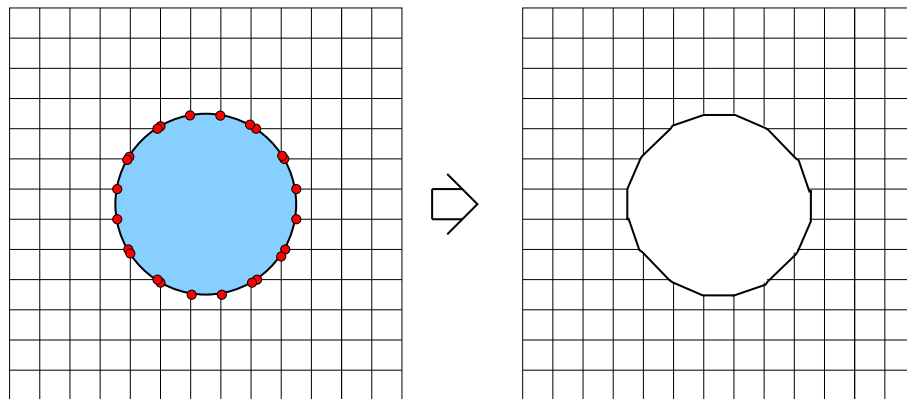


Abbildung 6.3: Randangepasstes Gitter für eine Kreisgeometrie mit der *Cut-Cell*-Technik.

Die Probleme dieser Methode bestehen darin, dass der Schritt der Objektaufteilung je nach gegebener Geometrie sehr zeitaufwendig und teuer sein kann. Weiterhin ist eine effiziente Interpolation zwischen den Gitterkomponenten, welche Masse- und Energieerhaltung gewährleistet, im Allgemeinen aufwändig.

Eine weitere Technik zur Darstellung komplexer Geometrien ist die sogenannte kartesische *Cut-Cell*-Technik. Die kartesische *Cut-Cell*-Methode ist vom Konzept her recht einfach und verwendet keine Gittergenerierung im konventionellen Sinne. Anstatt einer Gittergenerierung werden hier die Schnittpunkte für jedes Randsegment der Geometrie mit dem stationären Eulerschen Gitter berechnet. Die ermittelten Schnittpunkte zwischen dem Geometrierand und dem Eulerschen Gitter definieren die entsprechenden Schnittzellen ("cut-cells"), sodass das resultierende Gitter schliesslich ein an die Geometrie randangepasstes Gitter ergibt, wie Abbildung 6.3 veranschaulicht.

Probleme bei dieser Technik treten auf, wenn marginal kleine Teilstücke aus den *Cut-Cells* herausgeschnitten werden müssen. Insbesondere werden dann die notwendigen CFL-Stabilitätsbedingungen aus Abschnitt 4.4 für eine Strömungssimulation verletzt. Diese sichert im Rahmen einer expliziten Diskretisierung der konvektiven Terme, dass der Informationsfluss innerhalb eines Zeitschritts nicht weiter als eine Gittermaschenweite evolvieren darf. Sehr kleine Gitterteilstücke würden somit eine sehr kleine Zeitschrittweite zur Folge haben, die den gesamten Rechenaufwand der Simulation extrem in die Höhe treiben würde.

6.2 Generierung komplexer Geometrien mittels der Level-Set-Technik

Um die Problematiken von randangepassten und zusammengesetzten Gittern zu umgehen, verwenden wir eine Level-Set-Darstellung zur Geometriebeschreibung. Das heißt, im Unterschied zu den problemangepassten unstrukturierten oder blockstrukturierten Gittern sowie den zusammengesetzten Gittern, wie auch den *Cut-Cell* Gittern, verwenden wir die Einbet-

tungstechnik der Level-Set-Methode zur Darstellung komplexer Starrkörpergeometrien. Dazu beschreiben wir im Folgenden, wie wir eine vorgegebene triangulierte Geometrie in eine vorzeichenbehaftete Abstandsfunktion, also in eine Level-Set-Funktion, auf einem kartesischen Gitter transformieren. Mit anderen Worten, es muss in jedem kartesischen Gitterpunkt der euklidische Abstand zur vorgegebenen Geometrie ermittelt werden.

6.2.1 Grundidee des Level-Set-Verfahrens zur Beschreibung komplexer Geometrien

Level-Set-Verfahren gehören zur Klasse der Front-Capturing-Methoden, vgl. Abschnitt 3.2.4. Diese definieren den freien Rand als eingebettete Niveaumenge einer höherdimensionalen Lipschitz-stetigen skalaren Funktion. In unserem Fall werden wir aus einer vorgegebenen triangulierten Geometrie eine entsprechende vorzeichenbehaftete Abstandsfunktion auf einem kartesischen Gitter generieren. Diese ermöglicht es uns einerseits geometrische Größen, wie beispielsweise den Normalenvektor und die Krümmung, berechnen zu können und kann andererseits als Indikatorfunktion im Rahmen der zu lösenden Navier-Stokes-Gleichungen eingesetzt werden. Das Fortbewegen der Starrkörperoberfläche wird hierbei auf den Transport der entsprechenden Level-Set-Funktion zurückgeführt. Weiterhin kann der Starrkörpertrand durch einen Konturplot der entsprechenden Niveaumenge zum Niveaufwert Null visualisiert werden.

Um eine vorzeichenbehaftete Abstandsfunktionen für eine beliebige vorgegebene triangulierte Starrkörpergeometrie zu konstruieren, benötigen wir einerseits ein effizientes Verfahren, welches das Vorzeichen (d.h. Innen/Aussen-Differenzierung) bestimmt und andererseits ein effizientes Verfahren, welches den euklidischen Abstand eines kartesischen Gitterpunktes zu der vorgegebenen triangulierten Geometrie bis auf Maschinengenauigkeit berechnet.

6.2.2 Berechnung des Vorzeichens für Innen- und Aussendifferenzierung

Zur Bestimmung des Vorzeichens der Level-Set-Werte auf dem kartesischen Gitter, also der Differenzierung von Innen und Aussen einer vorgegebenen triangulierten Geometrie, verwenden wir ein effizientes Polygonscan-Verfahren. Die Idee von Polygonscan-Techniken beruht im Wesentlichen auf dem Jordankurven-Theorem. Dieses besagt, dass jede einfach zusammenhängende geschlossene Kurve die Ebene, in der sie liegt, in zwei disjunkte Gebiete aufteilt, ein Äußeres und ein Inneres. Wie Abbildung 6.4 links zeigt, kann man aufgrund dieses Theorems von allen Gitterpunkten einer Randebene des Rechengebietes aus eine Halbgerade durch das Rechengebiet verlaufen lassen. Die Anzahl und genaue Position ihrer Schnittpunkte mit der Geometrie werden in einem Vektor einmalig gespeichert. Aus diesem läßt sich für jeden Gitterpunkt des kartesischen Gitters über einfache Vergleichsabfragen sein entsprechendes Vorzeichen ermitteln. Die Behandlung geometrisch degenerierter Fälle, wie beispielsweise eine auf der Halbgeraden liegende Geometriekante oder ein einzelner Berührungspunkt mit einer Geometrieecke werden über die "Simulation of Simplicity"-

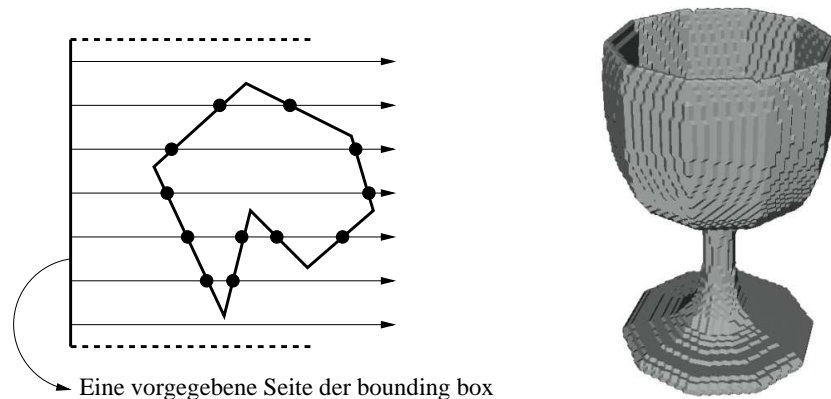


Abbildung 6.4: Polygonscan-Verfahren für eine vorgegebene Polygonzug-Geometrie. Der Becher auf der rechten Seite ist mit der bitweisen Innen- und Aussendifferenzierung auf einem kartesischen Gitter berechnet worden.

Technik [54] abgefangen. Diese täuscht in solchen Fällen eine infinitesimal kleine Verschiebung der Halbgeraden in eine Richtung vor, sodass sie den Geometrierand entweder eindeutig schneidet oder gar nicht erst berührt. Mit dieser Technik erhält man für jede geschlossene Starrkörpergeometrie immer entweder eine gerade Anzahl an Schnittpunkten oder keinen Schnittpunkt, woraus sich das Vorzeichen für die Level-Set-Werte eindeutig ermitteln läßt.

Abbildung 6.4 rechts zeigt die Null-Isofläche einer mit dem Polygonscan-Verfahren auf ein kartesisches Gitter transformierten komplexen 3D-Geometrie in Form eines Bechers. Im Vergleich zu der originalen triangulierten Geometrie ist diese Geometrie natürlich rauher, da auf dem kartesischen Gitter der Wertebereich sich nur auf $\{-1=\text{Außen}, 1=\text{Innen}\}$ beschränkt und deshalb in der Visualisierung auch die Gitterauflösung erkennbar ist.

6.2.3 Maschinengenaue Berechnung einer Abstandsfunktion

Mit dem Polygonscan-Verfahren haben wir zunächst unsere vorgegebene triangulierte Geometrie insoweit auf ein 3D kartesisches Gitter übertragen, als dass jetzt jeder kartesische Gitterpunkt mit dem Wert 1 oder -1 versehen ist und damit wiedergibt, ob er sich innerhalb oder außerhalb der vorgegebenen Geometrie befindet. Jedoch fehlen noch jegliche Informationen über den genauen Verlauf beziehungsweise die genaue Lage des Geometrierandes. Diese zusätzliche Information wird nun in einem zweiten Schritt über die Einführung einer Abstandsfunktion ermittelt. Hierbei berechnen wir in jedem kartesischen Gitterpunkt den euklidischen Abstand zu der Geometrie und multiplizieren diesen mit dem Gitterwert aus dem Polygonscanverfahren (also mit 1 oder -1). Damit erhalten wir schließlich die zu einer vorgegebenen triangulierten Geometrie auf ein 3D-Gitter übertragene adäquate vorzeichen-behaftete Abstandsfunktion.

Da wir uns auf triangulierte Geometrien beschränken, liegt das Ziel in einer schnellen Berechnung des euklidischen Abstandes zwischen einem Punkt und einem Dreieck. Dies ist ein klassisches Problem aus der Echtzeit-Computergraphik [53, 104].

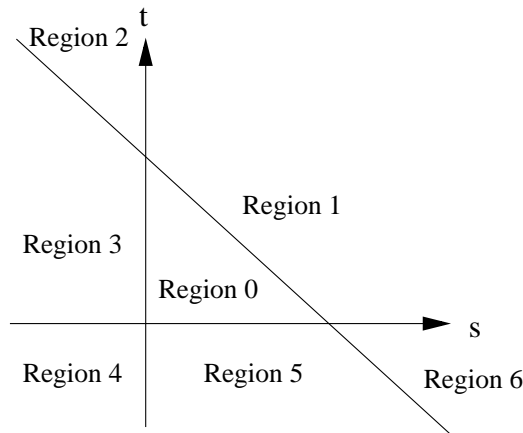


Abbildung 6.5: Aufteilung der Dreiecksebene in sieben disjunkte Regionen.

Mathematisch ausgedrückt wird in [53] die minimale euklidische Distanz gesucht zwischen einem gegebenen Punkt $\mathbf{p}(x, y, z)$ und einem gegebenen Dreieck $\mathbf{t}(s, t) = \mathbf{b} + s\mathbf{e}_0 + t\mathbf{e}_1$ für $(s, t) \in D = \{(s, t) : s \in [0, 1], t \in [0, 1], s + t \leq 1\}$. Diese berechnen wir über die Suche des Tupels $(\hat{s}, \hat{t}) \in D$, welches dem Punkt $\mathbf{t}(\hat{s}, \hat{t})$ auf dem Dreieck entspricht, der den geringsten Abstand zu dem Punkt $\mathbf{p}(x, y, z)$ hat. Der quadrierte Abstand für einen Punkt des Dreiecks zu dem Punkt \mathbf{p} ist $q(s, t) = |\mathbf{t}(s, t) - \mathbf{p}|^2$ für $(s, t) \in D$. Die Funktion q ist ein Polynom zweiten Grades in s und t ,

$$q(s, t) = as^2 + 2bst + ct^2 + ds + 2et + f, \quad (6.1)$$

wobei $a = \mathbf{e}_0^T \mathbf{e}_0$, $b = \mathbf{e}_0^T \mathbf{e}_1$, $c = \mathbf{e}_1^T \mathbf{e}_1$, $d = \mathbf{e}_0^T (\mathbf{b} - \mathbf{p})$, $e = \mathbf{e}_1^T (\mathbf{b} - \mathbf{p})$, und $f = (\mathbf{b} - \mathbf{p})^T (\mathbf{b} - \mathbf{p})$. Polynome zweiten Grades werden über das Vorzeichen von $ac - b^2$ klassifiziert. In unserem Fall gilt für $q(s, t)$

$$ac - b^2 = (\mathbf{e}_0 \cdot \mathbf{e}_0) \cdot (\mathbf{e}_1 \cdot \mathbf{e}_1) - (\mathbf{e}_0 \cdot \mathbf{e}_1)^2 = |\mathbf{e}_0 \times \mathbf{e}_1|^2 > 0. \quad (6.2)$$

Diese Positivität basiert auf der Annahme, dass die Kanten \mathbf{e}_0 und \mathbf{e}_1 des Dreiecks linear unabhängig sind, so dass ihr Kreuzprodukt keinen Nullvektor ergeben kann.

Damit wissen wir, dass $q(s, t)$ über D minimiert werden muss, um den Abstand des Punktes zum Dreieck zu ermitteln. Da q eine stetig differenzierbare Funktion ist, muss das Minimum entweder im Inneren von D , wobei der Gradient

$$\nabla q = 2(as + bt + d, bs + ct + e) = (0, 0)$$

ist, oder auf einem Randpunkt von D liegen. Der zugehörige Algorithmus verläuft wie folgt: Der Gradient von q ist genau dann Null, wenn $\hat{s} = (be - cd)/(ac - b^2)$ und $\hat{t} = (bd - ae)/(ac - b^2)$. Falls $(\hat{s}, \hat{t}) \in D$ gilt haben wir das Minimum von q gefunden. Anderenfalls muss das Minimum auf einem der Randkanten des Dreiecks liegen. Um den richtigen Rand zu finden, betrachten wir Abbildung 6.5. Hier wird die Ebene, in der das Dreieck liegt, in

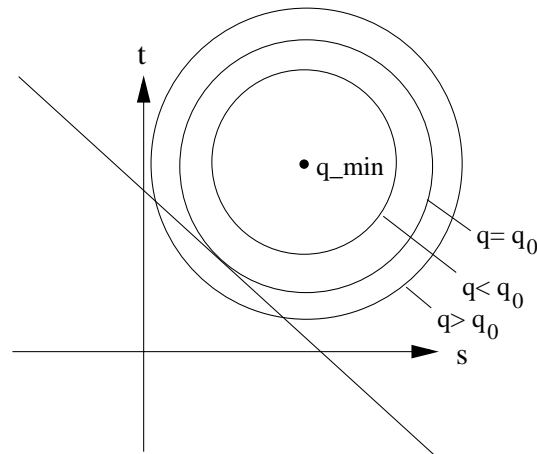


Abbildung 6.6: Verschiedene Niveaulinien zu unterschiedlichen Niveauebenen für $q(s, t)$.

sieben disjunkte Teilstücke beziehungsweise Regionen zerteilt. Das als Region 0 bezeichnete zentrale Dreieck entspricht gerade D . Falls (\hat{s}, \hat{t}) in Region 0 liegt, dann liegt der Punkt mit dem geringsten Abstand zu \mathbf{p} im Inneren des Dreiecks.

Angenommen, (\hat{s}, \hat{t}) liegt in Region 1, dann wissen wir, dass die Niveaulinien von $q(s, t)$ Ellipsen sind, da $q(s, t)$ ein Paraboloid ist. Dort wo $\nabla q = (0, 0)$ gilt, degenerieren die Ellipsen zu einem Punkt, dem Scheitelpunkt $q(\hat{s}, \hat{t}) = q_{min}$. Von q_{min} aus wachsen die Niveauebenen der Niveaulinien streng monoton. Daher gibt es einen kleinsten Niveauewert q_0 , für den die entsprechende Ellipse $q = q_0$ die Dreiecksseite $s + t = 1$ an einem Punkt $s = s_0 \in [0, 1], t = t_0 = 1 - s_0$ berührt. Dieser Punkt ist der am nächsten zum Punkt \mathbf{p} gelegene. Ermittelt wird er über den Schnitt des Graphs von $q(s, t)$ mit der Dreiecksseite $s + t = 1$. Dieser Schnitt wiederum bildet eine Parabel und ist der Graph von $f(s) = q(s, 1 - s)$ für $s \in [0, 1]$. Damit haben wir unser Minimierungsproblem auf $f(s)$ für $s \in [0, 1]$ eingeschränkt und somit um eine Dimension reduziert. Das Minimum von $f(s)$ tritt entweder im Inneren von $[0, 1]$, wo $f'(s) = 0$ ist, oder an einem Endpunkt auf, das heißt $s = 0$ oder $s = 1$. Um zwischen inneren Punkten und Endpunkten zu unterscheiden, verwenden wir dieselbe Partitionierungs-idee eingeschränkt auf den 1D-Fall. Das Intervall $[0, 1]$ partitioniert die reelle Achse in drei Teilintervalle, $s < 0, s \in [0, 1]$ und $s > 1$. Sei $f'(\hat{s}) = 0$. Falls $\hat{s} < 0$, dann ist $f(s)$ eine streng monoton wachsende Funktion für $s \in [0, 1]$. Das Minimum, eingeschränkt auf das Intervall $[0, 1]$, muss bei $s = 0$ angenommen werden, das heißt q nimmt sein Minimum bei $(s, t) = (0, 1)$ an. Falls $\hat{s} > 1$, dann ist $f(s)$ eine streng monoton fallende Funktion für $s \in [0, 1]$. Das Minimum, eingeschränkt auf das Intervall $[0, 1]$, muss bei $s = 1$ angenommen werden, das heißt q nimmt sein Minimum bei $(s, t) = (1, 0)$ an. Andererseits, falls $\hat{s} \in [0, 1]$, dann nimmt f sein Minimum in \hat{s} an und q erhält sein Minimum im Punkt $(s, t) = (\hat{s}, 1 - \hat{s})$.

Falls (\hat{s}, \hat{t}) in Region 3 liegt, dann wird das Minimum in $(0, t_0)$ für ein $t_0 \in [0, 1]$ angenommen. Falls (\hat{s}, \hat{t}) in Region 5 liegt, dann wird das Minimum in $(s_0, 0)$ für ein $s_0 \in [0, 1]$ angenommen. Die Ermittlung, ob der Berührungspunkt ein innerer Punkt oder ein Endpunkt des Intervalls ist, wird analog zum oben erklärten Verfahren durchgeführt.



Abbildung 6.7: Vergleich der Isoflächen zum Niveauwert Null des Polygonscan-Verfahrens (linkes Bild) und der vorzeichenbehafteten Abstandsfunktion (mittleres Bild); das rechte Bild zeigt die direkte Überlagerung beider Ergebnisse.

Falls (\hat{s}, \hat{t}) sich in Region 2 befindet, ist es möglich, dass die Niveaulinie von q entweder als erstes die Kante $s + t = 1$ oder als erstes die Kante $s = 0$ berührt. Da das globale Minimum in Region 2 liegt, kann der Gradient an dem Punkt $(0, 1)$ nicht ausserhalb von D zeigen. Das heißt, $(0, -1)(\nabla q(0, 1))^T$ und $(1, -1)(\nabla q(0, 1))^T$ können nicht beide negativ sein. Die zwei Vektoren $(0, -1)$ und $(1, -1)$ sind die Richtungsvektoren für die Kanten $s = 0$ und $s + t = 1$. Die Wahl der Kante $s + t = 1$ oder $s = 0$ wird in Abhängigkeit von den Vorzeichen von $(0, -1)(\nabla q(0, 1))^T$ und $(1, -1)(\nabla q(0, 1))^T$ getroffen. Dasselbe Verfahren wird auch für die Region 6 durchgeführt. In Region 4 geben die Vorzeichen der beiden Größen $(1, 0)(\nabla q(0, 0))^T$ und $(0, 1)(\nabla q(0, 0))^T$ an, welche Dreiecksseite das entsprechende Minimum enthält.

Als Beispiel ist in Abbildung 6.7 ein Direktvergleich der verschiedenen Darstellungen des Bechers aufgeführt. Das Polygonscan-Verfahren im linken Bild erzeugt ein skalares Feld auf dem kartesischen Gitter, welches nur aus den Werten $\{-1, 1\}$ besteht und damit angibt, ob sich der Gitterpunkt im Inneren oder im Äußeren befindet. Das mittlere Bild enthält nun zusätzlich die Information des euklidischen Abstandes zum geometrischen Rand, der hier durch die Niveaumenge zum Niveauwert Null dargestellt ist. Das rechte Bild zeigt eine Überlagerung beider Methoden, wodurch die Differenz beider Darstellungen des Bechers klar herausgestellt wird und der substantielle Informationsgewinn in der Geometriedarstellung mittels einer vorzeichenbehafteten Abstandsfunktion deutlich wird.

6.2.4 Effizienzsteigerung aufgrund der Lokalitätseigenschaft

Die genaue Transformation einer triangulierten Geometrie auf ein kartesisches Gitter in Form einer vorzeichenbehafteten Abstandsfunktion ist wichtig für den Navier-Stokes-Strömungslöser. Die aus der Abstandsfunktion resultierende charakteristische Funktion und die zusätzlich daraus gewonnenen geometrischen Größen, wie Normalenvektor und Krümmung,

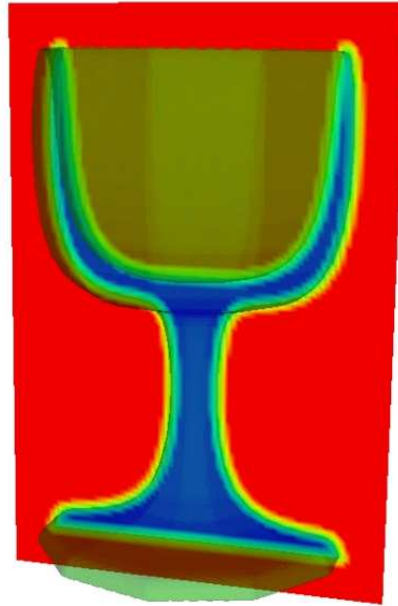


Abbildung 6.8: Darstellung der transparenten Isofläche zum Niveauwert Null inklusive einer zentralen Schnittebene, welche die Abstandswerte in farbcodierter Form wiedergibt.

können nun entsprechend in die Operatoren der Impuls- und Kontinuumsgleichung als Gewichte mit eingehen, um die Starrkörpergeometrie in das numerische Lösungsverfahren einzubeziehen. Von Vorteil ist hierbei, dass die Information einer vorzeichenbehafteten Abstandsfunktion nur in einer Umgebung der physikalisch relevanten Geometrieoberfläche gegeben sein muss. Das heißt, dass wir die Generierung der Abstandsfunktion, aufgrund der Sternbreite des WENO-Verfahrens fünfter Ordnung, bis zu sieben Gitterzellen vom Geometrierand entfernt durchführen müssen. Die restlichen Gitterzellen werden ohne Rechenaufwand direkt mit einer Abstandskonstanten von $const = 8h$ mit entsprechendem Vorzeichen versehen. Abbildung 6.8 zeigt die Isofläche zum Niveauwert Null und eine Mittelschnittebene einer entsprechend generierten Level-Set-Funktion des Bechers. Die Farbverteilung der Mittelschnittebene zeigt, dass innerhalb der ϵ -Umgebung die Abstandswerte zur Starrkörpergeometrie berechnet wurden (grüner Bereich) und ausserhalb davon ein konstanter Wert gesetzt wurde mit entsprechendem Vorzeichen in Abhängigkeit des Polygonscans (blauer/negativer und roter/positiver Flächenbereich). Hierbei wird der Entscheidungsprozess, ob innerhalb einer Gitterzelle der euklidische Abstand zum Geometrierand berechnet wird oder nicht, über die zugehörige "Bounding-Box"³ eines jeden Dreiecks gesteuert. Diese Bounding-Box wird zuvor entsprechend um sieben Gitterzellen in alle Raumrichtungen aufgeweitet. Wenn ein Gitterpunkt innerhalb der Bounding-Box eines Dreiecks der Starrkörpergeometrie liegt, dann wird dort der euklidische Abstand zu diesem Dreieck berechnet.

Dieses Verfahren reduziert den Rechenaufwand für die vorzeichenbehaftete Abstands-

³Die Bounding-Box ist hier definiert als der L^∞ -Umkreis eines Dreiecks.

funktion substantiell. Es wird ausserdem nur einmal zu Beginn der Simulationsrechnung in Form eines Vorverarbeitungsschrittes durchgeführt und hat somit keinen Einfluß auf die Rechenzeit des Navier-Stokes-Strömungslösers.

6.3 Semi-Lagrangischer Transport einer Starrkörpergeometrie

Die Starrkörpergeometrien, welche wir im Folgenden betrachten, werden – wie im letzten Abschnitt beschrieben – aus einer ursprünglich triangulierten Oberfläche in eine auf einem 3D-kartesischen Gitter definierte vorzeichenbehaftete Abstandsfunktion (Level-Set-Funktion) umgewandelt. Der Transport dieser Starrkörper-Level-Set-Funktion muss so umgesetzt werden, dass er die charakteristische Eigenschaft der Deformationsfreiheit aufrecht erhält. Hierbei ist insbesondere der Einfluß numerischer Diffusionsartefakte auf die Starrkörpergeometrie, welche im transienten Verlauf zu einer unphysikalischen Deformation führen würden, weitestgehend auszuschalten. Semi-Lagrangische Diskretisierungstechniken für die Transportgleichung der Starrkörpergeometrie sind für diese Zielsetzung besonders geeignet, da sie aufgrund ihres Lagrangeschen Ansatzes keine Diffusionsartefakte enthalten, wie sie typischerweise bei den Eulerschen Methoden auftreten.

In den folgenden Abschnitten werden wir zunächst die Bewegungsparameter eines Starrkörpers, bestehend aus Translation und Rotation, näher erläutert. Danach wird die Diskretisierung der Starrkörper-Transportgleichung mittels der Semi-Lagrangischen Technik beschrieben.

6.3.1 Bewegungsparameter eines dreidimensionalen Starrkörpers

Überall dort, wo Strukturen oder Teile von Strukturen ohne oder mit sehr kleinen Verformungen vorkommen, empfiehlt sich bei der numerischen Simulation die Verwendung von Starrkörpermodellen. So ist beispielsweise in [196] die Simulation von Steinschlagprozessen simuliert worden. Durch den zusätzlichen Einsatz entsprechender Zwangsbedingungen, wollen wir die Interaktion mit Fluiden simulieren, sodass die Bewegung von Starrkörpern in Zweiphasenströmungen effizient und akkurat simuliert werden kann. Hierzu leiten wir zunächst, analog zu [196], die reine Bewegung eines dreidimensionalen Starrkörpers her, welche neben der Translation zusätzlich aus der Rotation besteht.

Die Lage eines Starrkörpers im dreidimensionalen Raum läßt sich eindeutig durch unabhängige Parameter darstellen, wie zum Beispiel durch die Position von drei voneinander verschiedenen Körperpunkten ($3 \cdot 3 = 9$ Koordinaten mit 3 Bindungsgleichungen) oder durch die Koordinaten von zwei Punkten und der Verdrehung um die Verbindungsachse ($2 \cdot 3 + 1 = 7$ Werte bei einer Bindungsgleichung) oder auch durch die Koordinaten eines Bezugspunkts und die Verdrehung des Körpers um diesen Punkt ($3+3 = 6$ voneinander unabhängige Größen). Wenn ein Starrkörper in der Simulationsberechnung über seinen Schwerpunkt definiert wird, ist die dritte Option für die Beschreibung der Lage des Starrkörpers im Raum kanonisch. Im Schwerpunkt befindet sich zudem die Körpermasse im Gleichgewicht, das heißt

translatorische Beschleunigungen erzeugen keine Momente und Rotationen keine Kräfte im Schwerpunkt, was wiederum für die Berechnung der Starrkörperdynamik von Vorteil ist.

Das linke Bild in Abbildung 6.9 zeigt einen hexagonalen Starrkörper im globalen, zeitunabhängigen kartesischen Koordinatensystem. Der Ursprung des Koordinatensystems liegt im Schwerpunkt $\mathbf{x}_s(0)$. Die wesentliche Eigenschaft eines Starrkörpers ist die unveränderliche Lage eines jeden Körperpunktes bezüglich des lokalen Koordinatensystems, welches fest mit dem Starrkörperbezugspunkt $\mathbf{x}_s(t)$ verbunden ist, vgl. Abbildung 6.9 rechts. Damit läßt sich die Position jedes Starrkörperpunktes P bezüglich des globalen Koordinatensystems durch $x_P(t)$ und bezüglich des lokalen Koordinatensystems durch den konstanten Vektor x_P^l beschreiben. Das heißt, nach dem Satz von Chasles ist für die numerische Beschreibung des Starrkörpers die Formulierung der translatorischen Bewegung des Schwerpunkts sowie die Orientierung des lokalen Koordinatensystems ausreichend [73, 200].

Translation eines Starrkörpers im Raum

Sei $\mathbf{x}_S(t)$ die Position des Schwerpunktes einer Starrkörpergeometrie und $\mathbf{R}(t)$ die Orientierung des Starrkörpers zur Zeit t . Das Ziel ist im Folgenden die zeitliche Änderung der Position und der Orientierung zu definieren. Das heißt insbesondere, dass wir entsprechende Darstellungen für $\dot{\mathbf{x}}_S(t)$ und $\dot{\mathbf{R}}(t)$ benötigen. Da $\mathbf{x}_S(t)$ die Position des Schwerpunktes im globalen System (Realwelt) ist, ist $\dot{\mathbf{x}}_S(t)$ die Geschwindigkeit des Schwerpunktes im globalem System. Wir definieren die lineare Geschwindigkeit $\mathbf{v}(t)$ als die entsprechende Translationsgeschwindigkeit

$$\mathbf{v}(t) = \dot{\mathbf{x}}_S(t) \quad (6.4)$$

eines Starrkörpers. Wenn man die Orientierung eines Starrkörpers fixiert, dann ist die einzige Bewegung, die ein Starrkörper erfahren kann, die reine Translation und $\mathbf{v}(t)$ gibt die

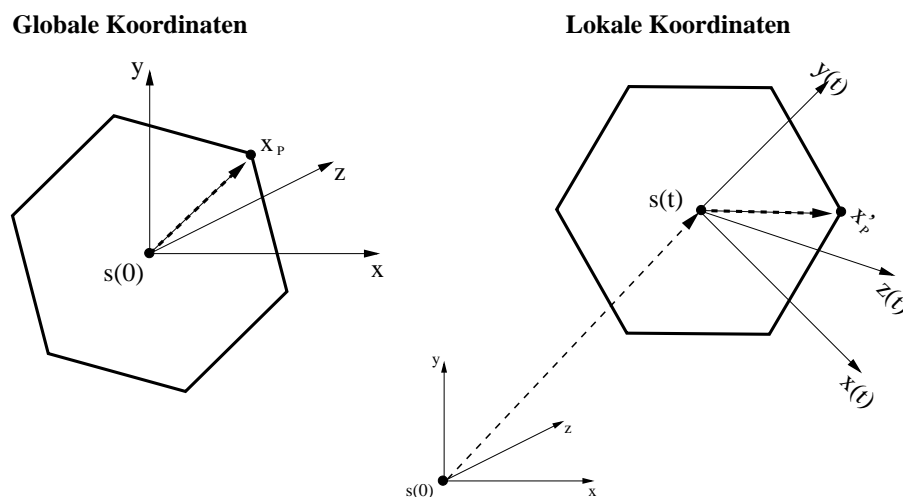
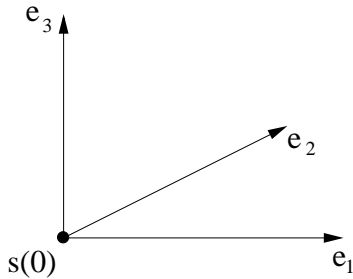


Abbildung 6.9: Hexagonaler Starrkörper: Globale, zeitunabhängige Koordinaten der Starrkörpergeometrie (links) und entsprechende lokale, zeitabhängige Koordinaten des Starrkörpers nach einer Translation und Rotation.

Globale Koordinatenachsen



Lokale Koordinatenachsen

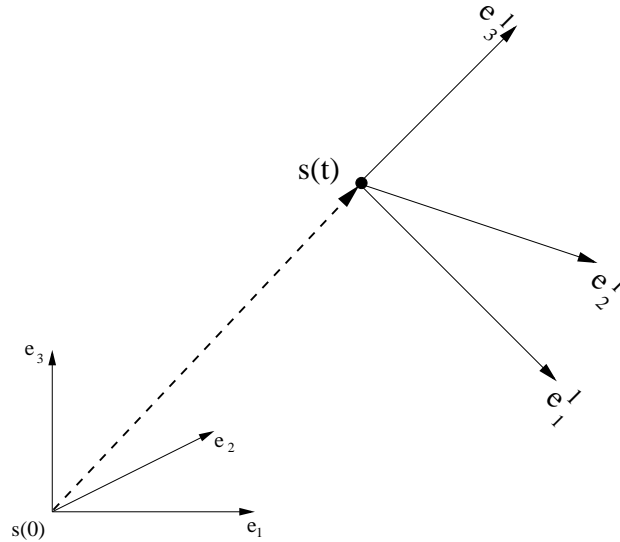


Abbildung 6.10: Das linke Bild zeigt die globalen, zeitunabhängigen Achsen einer nicht mitabgebildeten Starrkörpergeometrie und das rechte Bild zeigt die entsprechenden lokalen, zeitabhängigen Achsen nach einer Translation und Rotation.

Geschwindigkeit dieser Translation an.

Wenn eine rein translatorische Bewegung Δs_t des Schwerpunktes eines Starrkörpers stattfindet, so gilt die gleiche Verschiebung für den gesamten Starrkörper. Erst die Rotation des lokalen Bezugssystems erfordert eine zusätzliche Berechnung [139]. Ziel ist es nun, die Lage x_P des Punkts P im globalen Bezugssystem in Abhängigkeit von der Lage des Körperbezugs punkts, der Orientierung des lokalen Koordinatensystems, sowie der lokalen Koordinaten x_P^l zu beschreiben.

Rotation eines Starrkörpers im Raum

Die drei paarweise orthogonalen Basiseinheitsvektoren (e_1, e_2, e_3) spannen einen dreidimensionalen *euklidischen* Vektorraum auf. Durch ihre lineare Unabhängigkeit gilt

$$e_i \cdot e_j = \delta_{ij} \text{ mit } i, j = 1, 2, 3 \text{ und Kronecker-Delta } \delta_{ij} = \begin{cases} 1 & \text{falls } i = j, \\ 0 & \text{sonst.} \end{cases} \quad (6.5)$$

Die gleichen Eigenschaften gelten für das lokale Koordinatensystem (e_1^l, e_2^l, e_3^l) . Desweiteren lässt sich jeder der globalen Basisvektoren durch die lokalen Basisvektoren ausdrücken:

$$e_i = r_{i1}e_1^l + r_{i2}e_2^l + r_{i3}e_3^l = \sum_{j=1}^3 r_{ij}e_j^l \text{ mit } i = 1, 2, 3. \quad (6.6)$$

Bildet man nun jeweils aus den drei Basisvektoren die (3×3) -Matrizen $\underline{e}^l = [\mathbf{e}_1^l \mathbf{e}_2^l \mathbf{e}_3^l]$ und $\underline{e} = [\mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_3]$ lassen sich die obigen Gleichungen wie folgt formulieren:

$$\underline{e} = \mathbf{R} \underline{e}^l. \quad (6.7)$$

Die Matrix \mathbf{R} ist eine (3×3) -Matrix mit den Koeffizienten r_{ij} . Durch sie lässt sich die lokale Vektorbasis in globalen Koordinaten beschreiben und sie stellt somit eine Transformationsmatrix zwischen lokalem und globalem Koordinatensystem dar, wie in Abbildung 6.10 dargestellt.

Die Transformationsmatrix \mathbf{R} hat zudem folgende Eigenschaften:

- Die einzelnen Koeffizienten r_{ij} enthalten den Kosinus des Winkels ϕ_{ij} zwischen den beiden Vektoren \mathbf{e}_i^l und \mathbf{e}_j , da die Definition des Skalarprodukts $\mathbf{e}_j \cdot \mathbf{e}_i = |\mathbf{e}_j| |\mathbf{e}_i| \cos \phi$ für $|\mathbf{e}_i| = |\mathbf{e}_j| = 1$ bei der Berechnung der Matrixkoeffizienten wiederzufinden ist.
- Die Determinante von \mathbf{R} ist für Rechtssysteme gleich eins: $\det \mathbf{R} = 1$.
- Die Matrix ist orthonormal, das heißt, ihre Transponierte ist gleich ihrer Inversen:

$$\mathbf{R}^T = \mathbf{R}^{-1} \implies \mathbf{R}^T \mathbf{R} = \mathbf{I} \quad (\mathbf{I} = \text{Einheitsmatrix}) \quad (6.8)$$

Eulers Theorem [200] besagt nun, dass zwei beliebig orientierte Inertialsysteme mit einem gemeinsamen Ursprung durch eine einzige Drehung um eine bestimmte Achse ineinander überführt werden können. Die Drehachse verläuft dabei durch den Ursprung der Systeme und hat als Richtung den Eigenvektor \mathbf{r} der Transformationsmatrix für den Eigenwert $\lambda = 1$: $\mathbf{R} \mathbf{r} = \mathbf{r}$. Interpretiert man nun obige Transformation mit Hilfe dieses Theorems, so ergeben sich folgende Anwendungen für die Rotationsmatrix:

- Ein Vektor \mathbf{x}^l in lokalen Koordinaten kann mit Hilfe von \mathbf{R} in globale Koordinaten \mathbf{x} umgewandelt werden,

$$\mathbf{x} = \mathbf{x}_S(t) + \mathbf{R} \mathbf{x}^l \quad \text{bzw.} \quad \mathbf{x}^l = \mathbf{R}^T (\mathbf{x} - \mathbf{x}_S(t)). \quad (6.9)$$

- Eine lokale Matrix \mathbf{A}^l kann mit Hilfe von \mathbf{R} zu \mathbf{A} globalisiert werden:

$$\mathbf{A} = \mathbf{R} \mathbf{A}^l \mathbf{R}^T \quad \text{bzw.} \quad \mathbf{A}^l = \mathbf{R}^T \mathbf{A} \mathbf{R}. \quad (6.10)$$

- Es seien die drei Bezugssysteme \underline{e}^a , \underline{e}^b und \underline{e}^c gegeben. Dann lässt sich ein Vektor \mathbf{x}^a durch \mathbf{R}^{ba} in Koordinaten von \underline{e}^b , sowie \mathbf{x}^b durch \mathbf{R}^{cb} in Koordinaten von \underline{e}^c ausdrücken. Damit gilt für die Umwandlung von \mathbf{x}^a in \mathbf{x}^c : $\mathbf{x}^c = \mathbf{R}^{cb} \mathbf{x}^b = \mathbf{R}^{cb} (\mathbf{R}^{ba} \mathbf{x}^a)$. Das heißt, die Hintereinanderschaltung von zwei Transformationen lässt sich durch eine neue Transformationsmatrix ausdrücken

$$\mathbf{R}^{ca} = \mathbf{R}^{cb} \mathbf{R}^{ba}. \quad (6.11)$$

Dabei ist jedoch zu beachten, dass die Kombination der Transformationen nicht kommutativ ist, also im Allgemeinen $\mathbf{R}^{cb} \mathbf{R}^{ba} \neq \mathbf{R}^{ba} \mathbf{R}^{cb}$ gilt.

- Ein einzelner Vektor kann mit Hilfe der Transformationsmatrix innerhalb desselben Inertialsystems gedreht werden

$$\mathbf{x}_1 = \mathbf{R}\mathbf{x}_0 . \quad (6.12)$$

- Wird ein Körper ausgelenkt und dabei an einem Punkt festgehalten, so erfolgt eine Drehung um eine Achse durch diesen Punkt (Eulers Theorem).

Für die Rotationsmatrix gibt es verschiedene Darstellungen. Eine Auswahl wird im folgenden Abschnitt diskutiert.

Mathematische Beschreibungen von räumlichen Rotationsmatrizen

Bei der Verwendung sogenannter Eulerwinkel nimmt man an, dass die gesuchte Drehung das Ergebnis von drei aufeinander folgenden Drehungen um die Achsen z, x, z des Koordinatensystems ist. Die sogenannten Bryantwinkelverfahren sind ähnlich mit der Achsendrehungsreihenfolge x, y, z und entsprechend angepassten Teilrotationsmatrizen. Die Rotationsmatrix für beispielsweise die Bryantwinkel ergibt sich aus dem Produkt der drei aufeinander folgenden Drehungen mit den Winkeln χ, ψ und ϕ um die drei Koordinatenachsen x, y, z ($c := \cos, s := \sin$):

$$\mathbf{R}_\chi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\chi & s\chi \\ 0 & -s\chi & c\chi \end{bmatrix}, \quad \mathbf{R}_\psi = \begin{bmatrix} c\psi & 0 & s\psi \\ 0 & 1 & 0 \\ -s\psi & 0 & c\psi \end{bmatrix}, \quad \mathbf{R}_\phi = \begin{bmatrix} c\phi & s\phi & 0 \\ -s\phi & c\phi & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{R} = \mathbf{R}_\phi \mathbf{R}_\psi \mathbf{R}_\chi = \begin{bmatrix} c\psi c\chi & c\phi s\chi + s\phi s\psi c\chi & s\phi s\chi - c\phi s\psi c\chi \\ -c\psi s\chi & c\phi c\chi - s\phi s\psi s\chi & s\phi c\chi + c\phi s\psi s\chi \\ s\psi & -s\phi c\psi & c\phi c\psi \end{bmatrix}. \quad (6.13)$$

Hier wurden alle drei Koordinatenachsen als Drehachsen verwendet. Leider existieren aber verschiedene Definitionen für die Bryant- und Eulerwinkel, die sich in der Wahl der Drehachsen unterscheiden und insgesamt zwölf gleichwertige Beschreibungen liefern. Problematisch dabei ist, dass die Abbildung, die den Winkeln die zugehörige Drehmatrix zuordnet, kritische Punkte besitzt. In diesen Punkten ist die Zuordnung nicht lokal umkehrbar, man spricht vom sogenannten Gimbal Lock. Im Fall der (z, x, z) - oder (z, y, z) -Drehkonvention tritt dies auf, wenn der zweite Winkel gleich $k\pi$ ($k = 0, 1, 2, \dots$) ist, der Drehvektor der ersten Drehung ist dann gleich dem Drehvektor der zweiten Drehung. Das bedeutet, dass für eine Rotation um die z -Achse beliebig viele Eulerwinkel existieren, da die Achsen des ersten und des dritten Winkels zusammenfallen und beide Drehungen nicht mehr unterschieden werden können. Desweiteren ist die korrekte Kombination von Drehungen im Euler-System nicht intuitiv anzugeben, da sich die Drehachsen verändern.

Eine andere Möglichkeit, die Orientierung zu beschreiben und diese Nachteile zu umgehen, liegt im Einsatz von Quaternionen beziehungsweise Rotationsmatrizen, die den Starrkörper um eine eindeutig vorgegebene Achse mit einem eindeutig vorgegebenen Winkel rotieren. Die Rotationsmatrix mit beliebigem Einheitsvektor $\mathbf{v} = (v_1, v_2, v_3)^T$ als Drehachse

und α als Drehwinkel hat folgende Darstellung ($c := \cos$, $s := \sin$):

$$\mathbf{R} = \begin{bmatrix} c\alpha + v_1^2(1 - c\alpha) & v_1v_2(1 - c\alpha) - v_3s\alpha & v_1v_3(1 - c\alpha) + v_2s\alpha \\ v_2v_1(1 - c\alpha) + v_3s\alpha & c\alpha + v_2^2(1 - c\alpha) & v_2v_3(1 - c\alpha) - v_1s\alpha \\ v_3v_1(1 - c\alpha) - v_2s\alpha & v_3v_2(1 - c\alpha) + v_1s\alpha & c\alpha + v_3^2(1 - c\alpha) \end{bmatrix}.$$

Diese Drehung lässt sich auch über drei aufeinanderfolgende Drehungen mit den Eulerschen Winkeln um bestimmte Koordinatenachsen erzielen. Aber aufgrund der Reduktion auf nur eine Drehachse und nur eines Drehwinkels existieren hier keine Singularitäten mehr (Gimbal Lock). Die vier Parameter bestehend aus Drehachse und Drehwinkel werden auch als Euler-Parameter beziehungsweise als Quaternionen bezeichnet.

Der Einsatz einer solchen Rotationsmatrix für die Beschreibung der Rotation von schwimmenden Starrkörpern erfordert in jedem Zeitschritt der Navier-Stokes-Gleichungen eine Matrix-Vektor-Multiplikation für jeden Gitterpunkt. Dieser Rechenaufwand für die Rotation lässt sich aber bei genauer Betrachtung auf ein Kreuzprodukt reduzieren, wie im folgenden Abschnitt gezeigt wird.

6.3.2 Reine Bewegungsgleichung für Starrkörper

Verschiedene Rotationen können dann addiert werden, wenn sie eine Drehung um eine identische Achse beschreiben. Im Allgemeinen jedoch verändern sukzessive Drehungen auch die Richtung des Drehvektors und erfordern somit die Beschreibung der Drehgeschwindigkeit über eine Differentialgleichung. Die Drehgeschwindigkeit ist deshalb nicht einfach die Ableitung des Drehwinkels.

Ein beliebiger Starrkörperpunkt \mathbf{x}_P ist beschreibbar durch die Summe seines Starrkörperschwerpunktes \mathbf{x}_S mit der Starrkörperrotation \mathbf{R} des konstanten lokalen Ortsvektors $\mathbf{x}_P^l = \mathbf{R}^T(\mathbf{x}_P - \mathbf{x}_S)$ um einen entsprechenden Winkel α , vgl. Abbildung 6.9 :

$$\mathbf{x}_P = \mathbf{x}_S + \mathbf{x}_{PS} = \mathbf{x}_S + \mathbf{R}\mathbf{x}_P^l. \quad (6.14)$$

Die absolute Geschwindigkeitsverteilung für die Punkte P des Starrkörpers ergibt sich für $\mathbf{x}_P^l = \text{const.} (\rightarrow \dot{\mathbf{x}}_P^l = 0)$ aus der Differentiation nach der Zeit zu

$$\dot{\mathbf{x}}_P = \dot{\mathbf{x}}_S + \dot{\mathbf{R}}\mathbf{x}_P^l \quad (6.15)$$

mit $\dot{\mathbf{R}}$ als noch zu ermittelnder Grösse. Wegen der Orthonormalität der Matrix \mathbf{R} gilt

$$\mathbf{R}\mathbf{R}^{-1} = \mathbf{R}\mathbf{R}^T = \mathbf{I} \quad (6.16)$$

und von (6.16) abgeleitet nach der Zeit folgt

$$\dot{\mathbf{R}}\mathbf{R}^T + \mathbf{R}\dot{\mathbf{R}}^T = 0 \Leftrightarrow \dot{\mathbf{R}}\mathbf{R}^T = -\mathbf{R}\dot{\mathbf{R}}^T = -(\dot{\mathbf{R}}\mathbf{R}^T)^T. \quad (6.17)$$

Das heißt, der Ausdruck $(\dot{\mathbf{R}}\mathbf{R}^T)^T$ ist eine schiefsymmetrische Matrix. Die drei charakteristischen Koeffizienten dieser Matrix werden nun mit $\omega_1, \omega_2, \omega_3$ bezeichnet und der entsprechende Vektor $\boldsymbol{\omega}$ sei die Winkelgeschwindigkeit

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \quad \text{mit} \quad \dot{\mathbf{R}}\mathbf{R}^T = \boldsymbol{\Omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \quad (6.18)$$

Damit gelten für die Ableitungen der Rotationsmatrix die sogenannten Poissonschen Gleichungen [92]:

$$\dot{\mathbf{R}} = \boldsymbol{\Omega} \mathbf{R} \quad (6.19)$$

und für die Geschwindigkeit $\dot{\mathbf{x}}_P$ des Starrkörperpunkts P in globalen Koordinaten ergibt sich

$$\dot{\mathbf{x}}_P = \dot{\mathbf{x}}_S + \boldsymbol{\Omega} \mathbf{R} \mathbf{x}_P^l = \dot{\mathbf{x}}_S + \boldsymbol{\Omega} (\mathbf{x}_P - \mathbf{x}_S) = \dot{\mathbf{x}}_S + \boldsymbol{\omega} \times \mathbf{r} \quad (6.20)$$

mit $\mathbf{r} := (\mathbf{x}_P - \mathbf{x}_S)$. Die Darstellung mittels Kreuzprodukt entspricht der Definition der Geschwindigkeitsverteilung für die Punkte eines Starrkörpers nach [169]. Nach Gleichung (6.4) gilt $\dot{\mathbf{x}}_S = \mathbf{v}$, somit erhalten wir

$$\dot{\mathbf{x}}_P = \mathbf{v} + \boldsymbol{\omega} \times \mathbf{r} . \quad (6.21)$$

Im folgenden Abschnitt wird die semi-Lagrangesche Diskretisierung der Transportgleichung (6.21) für Starrkörper näher erläutert.

6.3.3 Semi-Lagrangesche Diskretisierung der Starrkörperbewegung

Der Vorteil von semi-Lagrangeschen Schemata für Level-Set-Funktionen [179] ist, dass sie bedingungslos stabil sind und Approximationen von höherer Ordnung liefern können. Desweiteren kann im Rahmen von sich bewegenden Starrkörpern der Vorteil genutzt werden, dass diese über lange Simulationszeiten hinweg ihre Form erhalten und nicht – wie im herkömmlichen Transportprozess – aufgrund von numerischen Diffusionsartefakten deformieren. Semi-Lagrangesche Schemata bestehen aus drei Hauptschritten: Raum-Zeit-Integration, Interpolation (oder Advektion) und Diskretisierung. Auf diese wird im Folgenden näher eingegangen.

Raum-Zeit-Integration

Wir betrachten die lineare hyperbolische partielle Differentialgleichung

$$\phi_t - F(\mathbf{x}, t) \cdot \nabla \phi = 0 . \quad (6.22)$$

Angenommen, ϕ sei auf einem regulären Gitter zur Zeit t bekannt und gesucht seien die Werte $\phi(\mathbf{x}, t + \Delta t)$ zur Zeit $t + \Delta t$. Dann folgt aus dem Fundamentalsatz der Differential- und Integralrechnung und Gleichung (6.22)

$$\phi(\mathbf{x}, t + \Delta t) = \phi(\mathbf{y}, t) + \int_C \nabla \phi \cdot (d\mathbf{x} + F dt) , \quad (6.23)$$

wobei C eine beliebige Trajektorie im Raum-Zeit-Gebiet ist, die (\mathbf{y}, t) mit $(\mathbf{x}, t + \Delta t)$ verbindet.

Die verschiedenen Methoden, mit denen Gleichung (6.22) numerisch behandelt werden kann, unterscheiden sich hauptsächlich in der Wahl von C . Eulersche Schemata setzen $\mathbf{x} = \mathbf{y}$

und C als ein gerades Liniensegment, welches parallel zur t -Achse verläuft, wie in Abbildung 6.11 oben dargestellt. Reine Lagrangesche Verfahren setzen C als lagrangesche Trajektorie T definiert durch

$$\dot{\mathbf{x}}(\sigma) = -F(\mathbf{x}(\sigma), \sigma), \quad (6.24)$$

die am Punkt \mathbf{y} beginnt, wie das mittlere Bild in Abbildung 6.11 veranschaulicht. Da $d\mathbf{x} + Fdt = 0$ auf T ist, transportieren Lagrangesche Techniken ϕ -Werte unverändert entlang T und sichern somit die bedingungslose Stabilität

$$\phi(\mathbf{x}, t + \Delta t) = \phi(\mathbf{y}, t). \quad (6.25)$$

Der größte Nachteil von Lagrangeschen Methoden liegt in der schnellen und starken Verzerrung von regulären Gittern, welche zu einem substantiellen Verlust der Diskretisierungsgenauigkeit führt. Semi-Lagrangesche Methoden kombinieren das reguläre Gitter eines Eulerschen Verfahren mit der bedingungslosen Stabilität eines Lagrangeschen Schemas. Sie berechnen Werte von ϕ an regulären Gitterpunkten \mathbf{x} zur Zeit $t + \Delta t$, indem sie eine Lagrangesche Trajektorie T rückwärts in der Zeit von $(\mathbf{x}, t + \Delta t)$ zu einem Punkt (\mathbf{y}, t) laufen und dann über die vorhandenen Gitterwerte zur Zeit t über Interpolation den entsprechenden Wert für ϕ im Punkt \mathbf{y} ermitteln, vgl. Abbildung 6.11 unten. Da für die Interpolation auf der t -Zeitscheibe $dt = 0$ ist und $d\mathbf{x} + Fdt = 0$ auf T gilt, erhalten wir

$$\phi(\mathbf{x}, t + \Delta t) = \phi(\mathbf{z}, t) + \int_L \nabla \phi \cdot d\mathbf{x} = \phi(\mathbf{y}, t). \quad (6.26)$$

Deshalb brauchen semi-Lagrangesche Verfahren nur die ϕ -Auswertung von \mathbf{x} auf \mathbf{y} mittels Interpolation auszuwerten.

Diskretisierung

Bei zwangsgeführten Starrkörpern ist die Translations- und Rotationsgeschwindigkeit in jedem Zeitschritt analytisch vorgegeben. Bei bidirektional gekoppelten dynamischen Festkörpern (Schwimmkörper etc.) wird die Translations- und Rotationsgeschwindigkeit vor Berechnung des Starrkörpertransportes in jedem Zeitschritt aus den wirkenden hydraulischen Kräften ermittelt, dieser Vorgang wird in Kapitel 7 beschrieben. Mit der bekannten Starrkörpergeschwindigkeit $F(\mathbf{x}, t) = \mathbf{v} + \boldsymbol{\omega} \times \mathbf{x}_p$ betrachten wir nochmals die partielle Differentialgleichung

$$\phi_t - F(\mathbf{x}, t) \cdot \nabla \phi = 0. \quad (6.27)$$

Gleichung (6.27) transportiert ϕ -Werte entlang der charakteristischen Kurven $\mathbf{s}(t)$ definiert durch

$$\dot{\mathbf{x}}(t) = -F(\mathbf{s}(t), t), \quad \mathbf{s}(0) = \mathbf{x}_0, \quad (6.28)$$

weil

$$\frac{d}{dt} \phi(\mathbf{s}(t), t) = \phi_t + \dot{\mathbf{x}} \cdot \phi_x = \phi_t - F \cdot \nabla \phi = 0 \quad (6.29)$$

gilt, wenn ϕ Gleichung (6.27) löst. Also können wir ϕ -Werte zu jeder Zeit t konstruieren, indem wir die charakteristische Kurve $\mathbf{s}(t)$ finden, die durch den Punkt (\mathbf{x}, t) verläuft, und

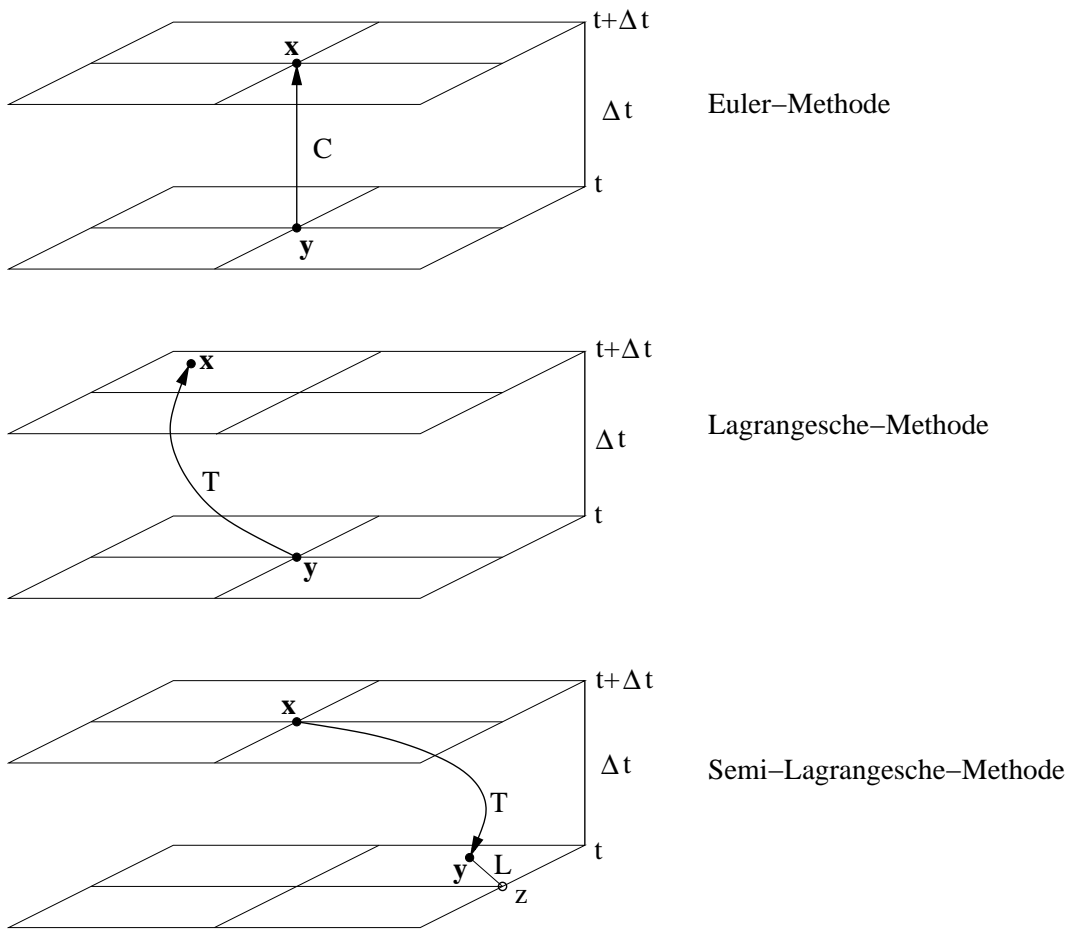


Abbildung 6.11: Raum-Zeit Integrationstechnik für den Pfad C einer Euler-Methode und den Pfad T einer Lagrangeschen- und Semi-Lagrangeschen-Methode. Um bei der Semi-Lagrangeschen-Methode den Level-Set Wert im Punkt y zu ermitteln, muss von einem bekannten Wert z aus über den Pfad L aufintegriert werden. Dies steht analog zu einer Interpolation aus bekannten umgebenen Gitterwerten.

diese dann rückwärts in der Zeit bis zu dem entsprechenden Anfangspunkt (\mathbf{x}_0, t_0) zurückverfolgen. Da zur Anfangszeit t_0 der Anfangswert von ϕ bekannt ist, können wir $\phi(\mathbf{x}_0, t_0)$ über Interpolation ermitteln und entsprechend $\phi(\mathbf{x}, t) = \phi(\mathbf{x}_0, t_0)$ setzen. Um die Charakteristik immer bis zur Anfangszeit t_0 zurückverfolgen zu können, müssen wir in jedem Zeitschritt die zurückgelegte Strecke mit der jeweils vorangegangenen aufsummieren. Denn für die Starrkörpergeschwindigkeit

$$F(\mathbf{x}, t) := \dot{\mathbf{x}} = \mathbf{v} + \boldsymbol{\omega} \times \mathbf{r} \quad (6.30)$$

mit $\mathbf{r} = \mathbf{x} - \mathbf{x}_S$ und \mathbf{x}_S als Starrkörperschwerpunkt gilt in diskreter Form mit einem Euler-Verfahren für die Zeitbehandlung

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t^n (\mathbf{v}^n + \boldsymbol{\omega}^n \times \mathbf{r}^n) . \quad (6.31)$$

Wenn wir dies induktiv bis zum Anfangszeitpunkt $t = 0$ fortsetzen, erhalten wir die Darstellung

$$\mathbf{x}^{n+1} = \mathbf{x}^0 + \sum_{j=0}^n \Delta t^j (\mathbf{v}^j + \boldsymbol{\omega}^j \times \mathbf{r}^j) \quad (6.32)$$

und schliesslich können wir nach \mathbf{x}^0 auflösen

$$\mathbf{x}^0 = \mathbf{x}^{n+1} - \sum_{j=0}^n \Delta t^j (\mathbf{v}^j + \boldsymbol{\omega}^j \times \mathbf{r}^j) . \quad (6.33)$$

Das heißt, wir können jeden Mittelpunkt \mathbf{x}^{n+1} einer Gitterzelle des ϕ -Wertes aus der Zeitebene t^{n+1} zurückführen auf seine Lage in dem Gitter der Anfangsbedingung ϕ_0 zur Zeit t^0 . Über trilineare Interpolation wird dann der Wert $\phi(\mathbf{x}^0, t^0)$ ermittelt und $\phi(\mathbf{x}^{n+1}, t^{n+1})$ auf den interpolierten Wert $\phi(\mathbf{x}^0, t^0)$ gesetzt. In unserem Fall haben wir für die Zeitdiskretisierung der Translations- und Rotationsgeschwindigkeit anstatt eines wie hier zur Veranschaulichung verwendeten Euler-Verfahrens ein Adams-Bashforth-Verfahren zweiter Ordnung umgesetzt.

Diese Diskretisierung der Transportgleichung wird in ihrer ursprünglichen Version, bei der die Charakteristiken nur zwischen zwei Zeitebenen rekonstruiert werden, nach ihren Entwicklern Courant, Isaacson und Rees [42] als *CIR*-Methode bezeichnet. Diese Methode enthält aber eine starke numerische Diffusion, da in jedem Zeitschritt eine lineare Rekonstruktion aus den Vorgängerwerten stattfindet. Die von uns umgesetzte Methode verwendet in jedem Zeitschritt nur einmalig eine lineare Rekonstruktion aus dem gegebenen Anfangswert ϕ^0 , so dass hier die numerische Diffusion nicht in jedem Zeitschritt kumuliert wird und somit eine sehr gute Formerhaltung der Starrkörpergeometrie garantiert wird.

Kapitel 7

Erweiterung des Modells auf Fluid-Starrkörper Wechselwirkung

In diesem Kapitel gehen wir näher auf das Modell ein, welches das Zusammenspiel beschreibt zwischen einem Starrkörper und dem ihn umgebenden viskosen, inkompressiblen Fluid mit freien Oberflächen. Wir verwenden hierbei die *Fictitious-Domain-Methode*, oft auch als Gebietseinbettungsmethode bezeichnet, mit einer verteilten Lagrange-Multiplikator-Technik. Dies stellt eine bidirektionale Koppelung zwischen dem Starrkörper und der Flüssigkeit sicher und ermöglicht somit die approximative Berechnung realer Interaktionen zwischen Starrkörpern und Fluiden. Die Starrheit des soliden Körpers wird dabei erhalten, indem das Geschwindigkeitsfeld innerhalb des Starrkörpers über dessen entsprechende Level-Set-Darstellung identifiziert wird. Zusätzlich wird das Geschwindigkeitsfeld auf eine Starrkörperbewegung eingeschränkt.

7.1 Die Fictitious-Domain-Methode

Fictitious-Domain-Verfahren umfassen eine grosse Klasse von Lösungsmethoden für partielle Differentialgleichungen, zu denen die Level-Set-Methode sehr artverwandt ist. Die Grundidee ist, das Gebiet eines Problemes mit komplexem geometrischen Rand auf ein größeres und einfacher berandetes Gebiet zu erweitern, das sogenannte "fiktive Gebiet". Dieser konzeptionelle Rahmen liefert zwei wesentliche Vorteile für die Konstruktion entsprechender numerischer Schemata:

- Das erweiterte Gebiet besitzt einen geometrisch einfacheren Rand, sodass regulärrere Gitter eingesetzt werden können, welche zu genaueren und stabileren Lösungen führen.
- Das erweiterte Gebiet kann zeitunabhängig sein, auch wenn das ursprüngliche Gebiet zeitabhängig ist. Daher kann ein stationäres festes Gitter für die gesamte Simulationsrechnung verwendet werden. Die bei unstrukturierten Gittertechniken aufgrund von starken Gitterverzerrungen regelmäßig anfallenden neuen Gittergenerierungen und Projektionen fallen hierbei vollständig weg. Dies führt schließlich zu einer

deutlich besseren Recheneffizienz und zu einer geringeren numerischen Diffusion, da keine Interpolationsoperatoren zwischen unterschiedlichen Gittern für die physikalischen Größen notwendig sind.

Natürlich müssen die Randbedingungen auf dem ursprünglichen komplexen Rand gesetzt werden, damit die auf dem ursprünglichen Gebiet eingeschränkte Lösung des erweiterten Gebietes auch die physikalisch richtige Lösung approximiert.

Fiktive Gebietsmethoden wurden erstmalig von Hyman [99] im Jahr 1952 eingeführt. Daraufhin wurden sie von Saul'ev [164], der den Begriff "Fictitious Domain" formte, und von Buzbee et al. [26] analysiert. Bis heute wird die fiktive Gebietsmethode in zahlreichen Publikationen eingesetzt und weiterentwickelt, wie zum Beispiel von Glowinski et al. [69–71] im Rahmen von partikelhaltigen, inkompressiblen, viskosen, einphasigen 2D Strömungen und von C. Peskin [41, 111, 148] im Rahmen von inkompressiblen, viskosen Strömungen mit elastischen, sich bewegenden, Rändern und LeVeque und Li [124, 125] im Rahmen von dynamischen elastischen Rändern oder Oberflächenspannungseffekten.

Falls die Bewegung von Starrkörpern, anstatt im voraus analytisch gegeben zu sein, aus entsprechend wirkenden hydrodynamischen Kräften und Momenten resultiert, so müssen die Bewegungsgleichungen für die Fluide mit den Bewegungsgleichungen eines Starrkörpers gekoppelt werden. Hierbei wird in der Methode von Hu [96] eine kombinierte (schwache) Formulierung des Problems aufgestellt, in der sich die hydrodynamischen Kräfte und Momente gegenseitig aufheben. Um eine solche kombinierte Gleichung zu ermöglichen, müssen die Einschränkungen der Starrkörperbewegung, insbesondere die Haft-Randbedingungen am Starrkörpertrand, ins Innere des Starrkörpers erweitert werden. Diese erweiterte Einschränkung wird in unserem Fall über einen zusätzlichen Druck innerhalb des Starrkörpers erzwungen, welcher – ähnlich dem Druck der Navier-Stokes Gleichungen – auch als Lagrange-Multiplikator angesehen werden kann. Dies hat zur Folge, dass die Wechselwirkung des Fluides mit einem Starrkörper auf eine Weise ins Navier-Stokes-Strömungsfeld eingebettet wird, welche zu jedem Zeitpunkt ein global divergenzfreies Geschwindigkeitsfeld garantiert.

7.2 Behandlung der Fluid-Starrkörper Wechselwirkung

Im Folgenden wird detailliert auf das Modell eingegangen, welches die Fluid-Starrkörper-Wechselwirkung beschreibt. Daraus wird deutlich, dass die Starrkörperdynamik ähnliche Eigenschaften aufweist, wie sie in inkompressiblen Fluiden vorzufinden sind.

Danach wird numerische Umsetzung des Modells im Zusammenhang mit unserer Level-Set-Darstellung von komplexen Starrkörpergeometrien beschrieben, sodass schließlich die Hafttrandbedingung und die Impermeabilitätbedingung auf dem Starrkörpertrand aktiviert werden.

7.2.1 Physikalisches Modell

Bevor wir die Gleichungen diskutieren, welche die Wechselwirkung von Starrkörpern mit Fluiden beschreiben, benötigen wir zunächst ein paar zusätzliche Notationen. Im Folgenden unterteilen wir das Rechengebiet in zwei disjunkte Teilgebiete. Das Teilgebiet, welches nur

die Fluide enthält, bezeichnen wir mit \mathcal{F} , und das Teilgebiet, welches den Starrkörper einnimmt, wird mit \mathcal{R} bezeichnet. Der gemeinsame Rand beider Teilgebiete ist $\partial\mathcal{R}$ und die Vereinigung beider Teilgebiete führt zu dem vollständigen Gebiet $\mathcal{V} = \mathcal{F} \cup \mathcal{R}$.

Wenn ein Starrkörper im Fluid simuliert wird, ist es nützlich, seine Dynamik in einen Translationsanteil und einen Rotationsanteil um seinen Schwerpunkt aufzuteilen. Dabei werden wir im Folgenden unseren Strömungslöser um zusätzliche Routinen erweitern, sodass die Dynamik des Starrkörpers über die Darstellung der Starrkörpergeschwindigkeiten in Form eines Translationsanteils und Rotationsanteils in die Impulsgleichungen eingeht.

Als Ausgangsbasis verwenden wir hierfür die physikalischen Modellgleichungen eines Fluid-Starrkörper-Systems [70], welches auf unseren Kontext der Zweiphasenströmungen ($i \in \{1, 2\}$) erweitert wurde:

Fluid-Bewegungsgleichungen in 3D:

$$\rho_i \frac{d\mathbf{u}_i}{dt} = \rho_i \mathbf{g} + \nabla \cdot \mathbf{T}_i + \sigma \kappa \delta(\phi) \mathbf{n} \quad \text{in } \Omega_i \setminus \overline{\mathcal{R}(t)}, \quad (7.1)$$

$$\nabla \cdot \mathbf{u}_i = 0 \quad \text{in } \Omega_i \setminus \overline{\mathcal{R}(t)}, \quad (7.2)$$

$$\mathbf{u}_i = \mathbf{u}_\Gamma(t) \quad \text{auf } \Gamma, \quad (7.3)$$

$$\mathbf{u}_i = \mathbf{v} + \boldsymbol{\omega} \times \mathbf{r} \quad \text{auf } \partial\mathcal{R}(t), \quad (7.4)$$

$$\mathbf{u}_i|_{t=0} = \mathbf{u}_{i,0} \quad \text{in } \Omega \setminus \overline{\mathcal{R}(0)}. \quad (7.5)$$

Starrkörper-Bewegungsgleichungen in 3D:

$$M \frac{d\mathbf{v}}{dt} = M \mathbf{g} + \mathbf{F}(\mathbf{u}_i, p), \quad (7.6)$$

$$\frac{d}{dt} \mathbf{J} \boldsymbol{\omega} = \mathbf{J} \frac{d\boldsymbol{\omega}}{dt} + \boldsymbol{\omega} \times \mathbf{J} \boldsymbol{\omega} = \mathbf{W}(\mathbf{u}_i, p), \quad (7.7)$$

$$\mathbf{v}|_{t=0} = \mathbf{v}_0, \quad (7.8)$$

$$\boldsymbol{\omega}|_{t=0} = \boldsymbol{\omega}_0. \quad (7.9)$$

Kinematische Gleichungen in 3D:

$$\frac{d\mathbf{x}_s}{dt} = \mathbf{v}, \quad (7.10)$$

$$\frac{d\boldsymbol{\Theta}}{dt} = \boldsymbol{\omega}, \quad (7.11)$$

$$\mathbf{x}_s|_{t=0} = \mathbf{x}_{s,0}, \quad (7.12)$$

$$\boldsymbol{\Theta}|_{t=0} = \boldsymbol{\Theta}_0. \quad (7.13)$$

Hierbei ist

$$\frac{d\mathbf{u}_i}{dt} = \frac{\partial \mathbf{u}_i}{\partial t} + (\mathbf{u}_i \cdot \nabla) \mathbf{u}_i \quad (7.14)$$

die Materialableitung und $\mathbf{r} := \mathbf{x} - \mathbf{x}_S$ aus Gleichung (7.4) ist der Richtungsvektor vom Schwerpunkt \mathbf{x}_S des Starrkörpers zu einem Starrkörperpunkt \mathbf{x} . Weiterhin sind ρ_l , \mathbf{u}_i und \mathbf{T}_i die Fluidichte, die Fluidgeschwindigkeit und der Newtonsche Spannungstensor; M , \mathbf{J} , \mathbf{v} , $\boldsymbol{\omega}$ und $\boldsymbol{\Theta}$ sind die Masse, der Trägheitstensor, die Translationsgeschwindigkeit, die Winkelgeschwindigkeit und die Winkelorientierung des Starrkörpers. Die beiden Größen

$$\mathbf{F} = - \int_{\partial\mathcal{R}(t)} \mathbf{T}_i \mathbf{n} \, ds \quad \text{und} \quad \mathbf{W} = - \int_{\partial\mathcal{R}(t)} \mathbf{r} \times \mathbf{T}_i \mathbf{n} \, ds \quad (7.15)$$

beschreiben die hydrodynamische Kraft und das Drehmoment um den Schwerpunkt über die Oberfläche des Starrkörpers zum Zeitpunkt t , wobei \mathbf{n} der nach außen gerichtete Einheitsnormalenvektor bezüglich $\partial\mathcal{R}$ ist. In typischen numerischen Methoden werden die beiden Randintegrale in Gleichung (7.15) approximativ gelöst und dann als explizite Randbedingungen entlang des Fluid-Starrkörper-Randes gesetzt. In unserem Fall aber wird die Fluid-Starrkörperdynamik implizit aktiviert über eine kombinierte Bewegungsgleichung, in der die Fluid-Starrkörper-Randbedingung ersetzt wird durch die Inkompressibilitätsbedingung und eine Volumenkraft, welche ausschließlich innerhalb des Starrkörpers wirkt. Dies hat den Vorteil einer deutlich höheren numerischen Stabilität und eines deutlich geringeren Rechenaufwandes, welcher sich darüber hinaus gut parallelisieren läßt.

Desweiteren betrachten wir in unserem Fall Starrkörpermaterialien mit homogener Dichte ρ_s , dass heisst

$$M = \rho_s \int_{\mathcal{R}(t)} d\mathbf{x} \quad \text{und} \quad \mathbf{J} = \begin{pmatrix} J_{11} & -J_{12} & -J_{13} \\ -J_{12} & J_{22} & -J_{23} \\ -J_{13} & -J_{23} & J_{33} \end{pmatrix}, \quad (7.16)$$

wobei hier $d\mathbf{x} = dx_1 dx_2 dx_3$ gilt und die Komponenten des Trägheitstensors die folgende Form haben:

$$J_{11} = \rho_s \int_{\mathcal{R}(t)} (x_2^2 + x_3^2) d\mathbf{x}, \quad (7.17)$$

$$J_{22} = \rho_s \int_{\mathcal{R}(t)} (x_3^2 + x_1^2) d\mathbf{x}, \quad (7.18)$$

$$J_{33} = \rho_s \int_{\mathcal{R}(t)} (x_1^2 + x_2^2) d\mathbf{x}, \quad (7.19)$$

$$J_{12} = \rho_s \int_{\mathcal{R}(t)} x_1 x_2 d\mathbf{x}, \quad (7.20)$$

$$J_{23} = \rho_s \int_{\mathcal{R}(t)} x_2 x_3 d\mathbf{x}, \quad (7.21)$$

$$J_{13} = \rho_s \int_{\mathcal{R}(t)} x_1 x_3 d\mathbf{x}. \quad (7.22)$$

Man beachte, dass die Anfangsgeschwindigkeit $(\mathbf{u}_i)_0$ und die Randwerte $\mathbf{u}_\Gamma(t)$ die Kompatibilitätsbedingungen

$$(\nabla \cdot \mathbf{u}_i)_0 = 0 \quad \text{in} \quad \Omega \setminus \overline{\mathcal{R}(0)} \quad \text{und} \quad \int_{\Gamma} \mathbf{u}_\Gamma(t) \cdot \hat{\mathbf{n}} \, ds = 0 \quad (7.23)$$

erfüllen müssen. Da wir ausschließlich Newtonsche Fluide betrachten, gilt für den Spannungstensor $\mathbf{T}_i = -p_i \mathbf{I} + \mu_i \mathbf{D}[\mathbf{u}_i]$, wobei p_i der Druck, μ_i die Viskosität und $\mathbf{D}[\mathbf{u}_i]$ der Deformationstensor bezüglich der beiden Fluidphasen ist. Desweiteren verwenden wir in den Gleichungen (7.11) und (7.13) die Eulerwinkel zur Beschreibung der Orientierung (Winkel-lage) von Objekten im dreidimensionalen Raum. Hierbei handelt sich um drei Winkel, welche jeweils eine Drehung (Rotation) um bestimmte Achsen beschreiben und so eine Transformation zwischen zwei (kartesischen) Koordinatensystemen, dem Laborsystem und dem körperfesten System, definieren.

Im Folgenden werden wir die Bewegungsgleichungen für den Starrkörper über eine entsprechend modifizierte Impuls- und Kontinuumsgleichung in die Navier-Stokes-Gleichungen einflechten. Dies ist möglich, da die Starrheitsbedingung des Starrkörpers sehr ähnlich zu der Inkompressibilitätsbedingung der Navier-Stokes-Gleichungen ist. Die Starrheitsbedingung ist darüber hinaus noch strikter, denn neben der geforderten Divergenzfreiheit des Geschwindigkeitsfeldes muss zusätzlich die Deformationsfreiheit für den Starrkörper erfüllt werden.

Die Starrheitsbedingung besagt, dass für jeden Punkt \mathbf{x} innerhalb eines Starrkörpers die Bewegungsgleichung

$$\dot{\mathbf{x}} = \mathbf{v} + \boldsymbol{\omega} \times \mathbf{r} \quad (7.24)$$

gilt mit einer Translationsgeschwindigkeit \mathbf{v} und einer Rotationsgeschwindigkeit $\boldsymbol{\omega}$ entlang der Achse $\boldsymbol{\omega}/|\boldsymbol{\omega}|$ mit dem Geschwindigkeitswert $|\boldsymbol{\omega}|$. Die Starrheitsbedingung kann mit Hilfe des Deformationstensors $\mathbf{D}[\mathbf{u}]$ ausgedrückt werden, welcher für ein gegebenes Geschwindigkeitsfeld $\mathbf{u} = (u, v, w)$ definiert ist durch

$$\mathbf{D}[\mathbf{u}] = [\nabla \mathbf{u} + \nabla \mathbf{u}^T] = \begin{bmatrix} 2u_x & (v_x + u_y) & (w_x + u_z) \\ (v_x + u_y) & 2v_y & (w_y + v_z) \\ (w_x + u_z) & (w_y + v_z) & 2w_z \end{bmatrix}. \quad (7.25)$$

Dieser symmetrische 3×3 Tensor $\mathbf{D}[\mathbf{u}]$ mißt die räumliche Änderung von \mathbf{u} . Somit sichert die Einschränkung

$$\mathbf{D}[\mathbf{u}] = \mathbf{0}, \quad (7.26)$$

dass die Bewegung in \mathcal{R} einer Starrkörperbewegung entspricht. Man beachte, dass diese Einschränkung uns nicht sagt wie diese Bewegung genau aussieht, aber wir wissen, dass das Geschwindigkeitsfeld Gleichung (7.24) erfüllen muss.

Desweiteren kann im Fall statischer Starrkörpergeometrien die Translationsgeschwindigkeit $\mathbf{v} = \mathbf{0}$ und die Winkelgeschwindigkeit auf $\boldsymbol{\omega} = \mathbf{0}$ gesetzt werden. Im Fall unidirektional gekoppelter Geometrien werden entsprechende Geschwindigkeitsvorgaben für \mathbf{v} und $\boldsymbol{\omega}$ gemacht.

7.2.2 Anwendung auf geglättetes Heaviside-Starrkörper-Profil

Basierend auf unsere Level-Set Darstellung von Starrkörpergeometrien werden wir im folgenden – analog zu den Zweiphasenströmungen, vgl. Abschnitt 4.2.1 – eine geglättete Darstellung von Starrkörpern einführen, welche vollständig in den Strömungslöser eingebunden

werden. Hierzu reformulieren wir die *smoothed profile method (SPM)* [138], die lediglich für einphasige, zweidimensionale, partikelhaltige Strömungen beschrieben wurde, für unseren Fall inkompressibler, dreidimensionaler Zweiphasen-Strömungen, inklusive der zuvor beschriebenen Level-Set-Darstellung komplexer dreidimensionale Starrkörper. Ein Hauptvorteil der SPM-Methode besteht in der stabilen und effizienten Darstellung der Bedingungen an der Fluid-Starrkörper-Grenzfläche.

Sei ϕ_s eine Level-Set-Darstellung einer Starrkörpergeometrie und $H^\epsilon(\phi_s)$ deren zugehörige geglättete Heaviside'sche Gewichtsfunktion. Basierend auf diesem Konzentrationsfeld wird das totale Geschwindigkeitsfeld \mathbf{u} als glatte Konvexkombination der Starrkörpergeschwindigkeit \mathbf{u}_s und der Fluidgeschwindigkeit \mathbf{u}_f gemäß

$$\mathbf{u}(x, t) \equiv (1 - H^\epsilon(\phi_s))\mathbf{u}_f + H^\epsilon(\phi_s)\mathbf{u}_s = \mathbf{u}_f + H^\epsilon(\phi_s)(\mathbf{u}_s - \mathbf{u}_f) \quad (7.27)$$

definiert, wobei zuvor die Starrkörpergeschwindigkeit $H^\epsilon(\phi_s)\mathbf{u}_s$ über

$$H^\epsilon(\phi_s)\mathbf{u}_s(\mathbf{x}, t) = H^\epsilon(\phi_s)(\mathbf{v}(t) + \boldsymbol{\omega}(t) \times [\mathbf{x} - \mathbf{x}_S(t)]) \quad \text{in } \mathcal{R} + \epsilon \quad (7.28)$$

ermittelt wird. Da das Starrkörpergeschwindigkeitsfeld \mathbf{u}_s aus Translation und Rotation zusammengesetzt ist, erfüllt es insbesondere auch die Bedingung der Divergenzfreiheit, es gilt

$$\nabla \cdot \mathbf{u}_s = \{\mathbf{v}(t) + \boldsymbol{\omega}(t) \times [\mathbf{x} - \mathbf{x}_S(t)]\} \cdot \nabla \left(\frac{H^\epsilon(\phi_s)}{H^\epsilon(\phi_s)} \right) = 0. \quad (7.29)$$

Implizite Randbedingungen

Ein weiterer Vorteil der SPM-Methode ist, dass sie auf implizite Weise die Hafttrandbedingungen entlang der Starrkörpergeometrie erfüllt. Dies geht aus der Anwendung des Rotationsoperators auf das Geschwindigkeitsfeld aus Gleichung (7.27) hervor [129]:

$$\begin{aligned} \nabla \times \mathbf{u} &= \nabla \times [\mathbf{u}_f + H^\epsilon(\phi_s)(\mathbf{u}_s - \mathbf{u}_f)] \\ &= (\nabla H^\epsilon(\phi_s)) \times (\mathbf{u}_s - \mathbf{u}_f) + H^\epsilon(\phi_s)(\nabla \times \mathbf{u}_s) + (1 - H^\epsilon(\phi_s))(\nabla \times \mathbf{u}_f). \end{aligned}$$

Da $\nabla H^\epsilon(\phi_s)$ nur entlang des Randes der Starrkörpergeometrie ungleich Null ist und $\nabla H^\epsilon \rightarrow \delta$ für $\epsilon \rightarrow 0$ gilt, kann die Hafttrandbedingung ausgedrückt werden als

$$(\nabla H^\epsilon(\phi_s)) \times (\mathbf{u}_s - \mathbf{u}_f) = \mathbf{0} \iff \mathbf{u}_s = \mathbf{u}_f \quad \text{auf } \Gamma_s \text{ für } \epsilon \rightarrow 0. \quad (7.30)$$

Das heißt insbesondere, dass die Hafttrandbedingung von der Annahme herrührt, dass das Wirbelfeld des totalen Geschwindigkeitsfeldes aus Wirbeln der Fluid- und der Starrkörpergeschwindigkeit besteht, also

$$\nabla \times \mathbf{u} = H^\epsilon(\phi_s)(\nabla \times \mathbf{u}_s) + (1 - H^\epsilon(\phi_s))(\nabla \times \mathbf{u}_f) \quad (7.31)$$

gilt. Desweiteren setzt die SPM-Methode implizit die Impermeabilitätsbedingung an der Starrkörperoberfläche. Wenn die Divergenz des gesamten Geschwindigkeitsfeldes unter der Annahme analysiert wird, dass das Fluid inkompressibel und die Starrkörpergeschwindigkeit divergenzfrei ist, ergibt sich

$$0 = \nabla \cdot \mathbf{u} = \nabla \cdot [\mathbf{u}_f + H^\epsilon(\phi_s)(\mathbf{u}_s - \mathbf{u}_f)] = (\nabla H^\epsilon(\phi_s)) \cdot (\mathbf{u}_s - \mathbf{u}_f). \quad (7.32)$$

Hierbei repräsentiert $(\nabla H^\epsilon(\phi_s)) \cdot (\mathbf{u}_s - \mathbf{u}_f) = 0$ die Impermeabilitätsbedingung an der Starrkörperoberfläche. Das gesamte Geschwindigkeitsfeld ist somit genau dann divergenzfrei, wenn die Impermeabilitätsbedingung am Starrkörpertrand erfüllt ist.

Da – wie zuvor gezeigt – die Starrkörperdynamik ähnliche Eigenschaften aufweist wie sie in inkompressiblen Fluiden vorzufinden sind, motiviert dies schließlich die Einbettung der Starrkörpergeschwindigkeiten in das Navier-Stokes-Strömungsfeld mittels eines zusätzlichen Lagrange-Multiplikator-Ansatzes, auf den wir nun im Rahmen der erweiterten Chorin-Temam-Projektionsmethode näher eingehen werden.

7.3 Erweiterte Chorin-Temam-Projektionsmethode

Im Kontext unseres Zweiphasen-Lösers wird über die *SPM*-Technik das gesamte Geschwindigkeitsfeld \mathbf{u} auf dem gesamten Gebiet Ω inklusive des Starrkörpergebietes berechnet, indem die Impulsgleichungen der Zweiphasen-Navier-Stokes-Gleichungen mit einem zusätzlichen Kraftdichteterm \mathbf{f}_s versehen werden, der die Interaktion zwischen Starrkörper und Fluid darstellt, so dass man für die Strömung die Gleichungen

$$\begin{aligned} \rho_i \frac{d\mathbf{u}_i}{dt} &= -\nabla p_i + \nabla \cdot (\mu_i \mathbf{D}_i) + \rho_i \mathbf{g} + \sigma \kappa \delta \mathbf{n} + \mathbf{f}_s && \text{in } \Omega_i, \\ \nabla \cdot \mathbf{u}_i &= 0 && \text{in } \Omega_i, \\ \mathbf{u}_i|_\Gamma &= 0 && \text{in } [0, T], \\ \mathbf{u}_i|_{t=0} &= \mathbf{u}_{0_i} && \text{in } \Omega_i \end{aligned} \quad (7.33)$$

für $i \in \{1, 2\}$ erhält. Für den Starrkörper gelten die Gleichungen

$$\rho_s \frac{\partial \mathbf{u}_s}{\partial t} = -\nabla p_s + \rho_s \mathbf{g} \quad \text{in } \Omega_s, \quad (7.34)$$

$$\nabla \cdot \mathbf{u}_s = 0 \quad \text{in } \Omega_s. \quad (7.35)$$

Hier gehen die Materialparameter und der Druck innerhalb des soliden Starrkörpers mit ein.

Der zugrunde liegende numerische Lösungsalgorithmus ergibt sich aus einer Erweiterung der Chorin-Temam-Projektionsmethode. Zuerst werden der Schwerpunkt \mathbf{x}_S und die Masse M des Starrkörpers über

$$\mathbf{x}_S^0 = \int_{\Omega} H^\epsilon(\phi_s^0(\mathbf{x})) \mathbf{x} \, d\mathbf{x} \quad \text{und} \quad M = \int_{\Omega} \rho^\epsilon(\phi_s^0(\mathbf{x})) \, d\mathbf{x} \quad (7.36)$$

berechnet, wobei eine Mittelpunkt-Methode als Integrationsschema verwendet wird.

Danach werden die aktuelle Translationsgeschwindigkeit \mathbf{v}^0 des Schwerpunktes und die Rotationsgeschwindigkeit $\boldsymbol{\omega}^0$ um den Schwerpunkt initialisiert:

$$\mathbf{v}^0 = \mathbf{v}_{init} \quad \text{und} \quad \boldsymbol{\omega}^0 = \boldsymbol{\omega}_{init}. \quad (7.37)$$

Analog zum ursprünglichen Verfahren aus Gleichung (4.28) lösen wir zunächst nach einem intermediärem Feld \mathbf{u}^* auf, welches die Geschwindigkeiten des vorherigen Zeitschritts ohne

den Druckgradienten enthält,

$$\mathbf{u}^* = \mathbf{u}^n + \delta t(-\nabla \cdot (\mathbf{u}^n \otimes \mathbf{u}^n) + \mathbf{g} + \frac{1}{\rho^\epsilon(\phi^n, \phi_s^n)}(\nabla \cdot (\mu^\epsilon(\phi^n, \phi_s^n)\mathbf{D}^n) - (1 - H^\epsilon(\phi_s^n))\sigma\kappa(\phi^n)\delta(\phi^n)\nabla\phi^n), \quad (7.38)$$

wobei ϕ^n die Level-Set-Funktion bezüglich der Zweiphasenverteilung ist und ϕ_s^n die Level-Set-Funktion bezüglich des Starrkörpers. Der Hauptunterschied zu (4.28) besteht darin, dass die Materialparameter nun zusätzlich von der Level-Set-Funktion des Starrkörpers abhängen. Dies hat zur Folge, dass innerhalb des Starrkörpers das reine Newton'sche Bewegungsgesetz gilt und ausserhalb des Starrkörpers die Zweiphasen-Navier-Stokes-Gleichungen gelten.

Anschließend wird eine vorläufige, neue Level-Set-Funktion

$$\phi^* = Z(R(\phi^n, \mathbf{u}^n)),$$

durch explizites Lösen der Transportgleichung nach Abschnitt 4.2.2 ermittelt, wobei Z der diskrete Zeitoperator und R der diskrete Raumoperator sind. Nun wird ϕ^{**} durch Reinitialisierung (*REINIT*) von ϕ^* nach Abschnitt 4.2.3 bestimmt und aus ϕ^{**} mittels Picard-Iteration (*PIT*) die neue Level-Set-Funktion ϕ^{n+1} nach Abschnitt 4.2.5 berechnet, also

$$\phi^{n+1} = PIT(\phi^{**}) \quad \text{mit} \quad \phi^{**} = REINIT(\phi^*). \quad (7.39)$$

Danach wird die zur Starrkörperdarstellung gehörige Level-Set-Funktion ϕ_s^n mittels dem in Abschnitt 6.3.3 dargestellten Semi-Lagrangeschen-Verfahren (*SL*) unter Verwendung der Translationsgeschwindigkeit \mathbf{v}^n und Winkelgeschwindigkeit transportiert ω^n ,

$$\phi_s^{n+1} = SL(\mathbf{v}^n, \omega^n). \quad (7.40)$$

Anschließend wird aufgrund der neuen Lage des Starrkörpers gemäß

$$\mathbf{x}_S^{n+1} = \int_{\Omega} H^\epsilon(\phi_s^{n+1}(\mathbf{x}))\mathbf{x} \, d\mathbf{x} \quad (7.41)$$

sein aktueller Schwerpunkt berechnet. Schließlich wird aus dem System

$$\frac{\mathbf{u}^{*,n+1} - \mathbf{u}^*}{\delta t} + \frac{\nabla p^{*,n+1}}{\rho^\epsilon(\phi^{n+1}, \phi_s^{n+1})} = 0, \quad (7.42)$$

$$\nabla \cdot \mathbf{u}^{*,n+1} = 0 \quad (7.43)$$

durch Anwendung des Divergenzoperators auf Gleichung (7.42) und unter Verwendung von Gleichung (7.43) die Poissongleichung

$$\nabla \cdot \left(\frac{\nabla p^{*,n+1}}{\rho^\epsilon(\phi^{n+1}, \phi_s^{n+1})} \right) = \nabla \cdot \frac{\mathbf{u}^*}{\delta t} \quad (7.44)$$

hergeleitet und gelöst. Hierdurch wird der neue Druck $p^{*,n+1}$ implizit durch Lösen der Poissongleichung berechnet und liefert somit ein neues temporäres Geschwindigkeitsfeld $\mathbf{u}^{*,n+1}$ durch den Korrekturschritt

$$\mathbf{u}^{*,n+1} = \mathbf{u}^* - \frac{\delta t}{\rho^\epsilon(\phi^{n+1})} \nabla p^{*,n+1}. \quad (7.45)$$

Als nächstes werden die hydrodynamischen Kräfte \mathbf{F}^n und das Drehmoment \mathbf{W}^n , welche durch die Fluide auf den Starrkörper wirken, aus der Impulserhaltung zwischen dem Starrkörper und den Fluiden hergeleitet. Die hydrodynamische Kraft ist definiert als Impulsfluß zwischen dem Fluid und dem Starrkörper. Nach Newtons drittem Gesetz bedeutet dies insbesondere, dass die hydrodynamische Kraft die Gegenwirkung des Fluides ist,

$$\mathbf{F}^n = -\frac{1}{\Delta t} \int_{\Omega} H^\epsilon(\phi_s^{n+1}) \rho(\phi_s^{n+1}) (\mathbf{u}_s^n - \mathbf{u}^{*,n+1}) d\mathbf{x}, \quad (7.46)$$

$$\mathbf{W}^n = -\frac{1}{\Delta t} \int_{\Omega} H^\epsilon(\phi_s^{n+1}) \rho(\phi_s^{n+1}) (\mathbf{u}_s^n - \mathbf{u}^{*,n+1}) \times \mathbf{r}^{n+1} d\mathbf{x}, \quad (7.47)$$

wobei $\mathbf{r}^{n+1} := \mathbf{x} - \mathbf{x}_S^{n+1}$, wie zuvor im physikalischen Modell definiert, der neue Abstandsvektor vom Schwerpunkt \mathbf{x}_S^{n+1} des Starrkörpers zu einem beliebigen Gitterpunkt \mathbf{x} ist. Mittels der hydrodynamischen Kraft wird im Folgenden die Translations- und Winkelgeschwindigkeit des Starrkörpers mittels der Newtonschen Bewegungsgleichung aktualisiert,

$$\mathbf{v}^{n+1} = \mathbf{v}^n + M^{-1} \Delta t \mathbf{F}^n, \quad (7.48)$$

$$\boldsymbol{\omega}^{n+1} = \boldsymbol{\omega}^n + \mathbf{J}^{-1} \Delta t (\mathbf{W}^n - \boldsymbol{\omega}^n \times \mathbf{J} \boldsymbol{\omega}^n), \quad (7.49)$$

wobei M und \mathbf{J}^{-1} die Masse des Starrkörpers und der Trägheitstensor sind. Als nächstes wird mittels der Translations- und Winkelgeschwindigkeit das neue Geschwindigkeitsfeld des Starrkörpers berechnet,

$$H^\epsilon(\phi_s^{n+1}) \mathbf{u}_s^{n+1} = H^\epsilon(\phi_s^{n+1}) (\mathbf{v}^{n+1} + \boldsymbol{\omega}^{n+1} \times [\mathbf{x} - \mathbf{x}_S^{n+1}]). \quad (7.50)$$

Die Einbindung der neuen Starrkörpergeschwindigkeit \mathbf{u}_s^{n+1} in das temporäre Geschwindigkeitsfeld $\mathbf{u}^{*,n+1}$ verläuft über die Berechnung eines zusätzlichen Druckfeldes p_s innerhalb des Starrkörpers,

$$\nabla \cdot \left(\frac{\nabla p_s^{*,n+1}}{\rho^\epsilon(\phi^{n+1}, \phi_s^{n+1})} \right) = \nabla \cdot \left(\frac{H^\epsilon(\phi_s^{n+1}) (\mathbf{u}_s^{n+1} - \mathbf{u}^{*,n+1})}{\delta t} \right). \quad (7.51)$$

Dieses Druckfeld führt schließlich zur Starrheit des Starrkörpers, indem durch die Korrektur

$$\mathbf{u}^{n+1} = \mathbf{u}^{*,n+1} + H^\epsilon(\phi_s^{n+1}) (\mathbf{u}_s^{n+1} - \mathbf{u}^{*,n+1}) - \frac{\delta t}{\rho^\epsilon(\phi^{n+1}, \phi_s^{n+1})} \nabla p_s^{*,n+1} \quad (7.52)$$

die globale Divergenzfreiheit des Geschwindigkeitsfeldes \mathbf{u}^{n+1} garantiert wird. Das gesamte Druckfeld berechnet sich aus den beiden Teildruckfeldern $p^{*,n+1}$ und $p_s^{*,n+1}$ mittels einfacher Summation

$$p^{n+1} = p^{*,n+1} + p_s^{*,n+1}. \quad (7.53)$$

Damit (7.44) und (7.51) wohlgestellt sind, sind noch Randbedingungen für den Druck notwendig. Dazu wird (7.42) auf die äußere Einheitsnormale des Gebietsrandes projiziert, also

$$\left. \frac{\partial p^{*,n+1}}{\partial \mathbf{n}} \right|_{\Gamma} = \frac{\rho^\epsilon(\phi^{n+1}, \phi_s^{n+1})}{\delta t} (\mathbf{u}_{\Gamma}^* - \mathbf{u}_{\Gamma}^{*,n+1}) \cdot \mathbf{n} \quad (7.54)$$

und analog auch für das zusätzliche Druckfeld p_s

$$\left. \frac{\partial p_s^{*,n+1}}{\partial \mathbf{n}} \right|_{\Gamma} = \frac{\rho^\epsilon(\phi^{n+1}, \phi_s^{n+1})}{\delta t} (\mathbf{u}_{s,\Gamma}^{n+1} - \mathbf{u}_{\Gamma}^{*,n+1}) \cdot \mathbf{n}. \quad (7.55)$$

Falls nun am Rand für die Schätzgeschwindigkeiten $\mathbf{u}_{\Gamma}^* = \mathbf{u}_{\Gamma}^{*,n+1}$ beziehungsweise $\mathbf{u}_{s,\Gamma}^* = \mathbf{u}_{\Gamma}^{*,n+1}$ gesetzt wird, ergeben sich homogene Neumann-Randbedingungen für den Druck, also

$$\left. \frac{\partial p_{(s)}^{*,n+1}}{\partial \mathbf{n}} \right|_{\Gamma} = 0. \quad (7.56)$$

Desweiteren wird der Druck mit diesem Verfahren nur bis auf eine Konstante genau bestimmt. Analog zu den Zweiphasenproblemen kann diese fixiert werden, indem zusätzlich die Nebenbedingung

$$\int_{\Omega} p^{n+1} = 0$$

gefordert wird. Damit ist es möglich, Druckergebnisse unterschiedlicher Zeitpunkte für Zweiphasen-Starrkörper-Simulationen miteinander zu vergleichen.

7.4 Diskretisierung der erweiterten Projektionsmethode

Die Diskretisierung der erweiterten Projektionsmethode verläuft in weiten Teilen analog zu der Diskretisierung des Zweiphasenmodells mit Oberflächenspannung aus Kapitel 4. Hinzu kommen die Diskretisierung des semi-Lagrangeschen Transports der Level-Set-Funktion des Starrkörpers in Gleichung (7.40), welche in Abschnitt 6.3.3 detailliert beschrieben wird. Desweiteren sind sowohl die Integrale aus Gleichung (7.36) zur Berechnung des Starrkörperschwerpunktes und dessen Masse als auch die Integrale aus den Gleichungen (7.46) und (7.47) zur Berechnung der hydrodynamischen Kräfte mittels eines einfachen Integrations-

schemas der Form

$$\mathbf{x}_S^0 = \int_{\Omega} H^\epsilon(\phi_s^0(\mathbf{x})) \mathbf{x} \, d\mathbf{x} \quad (7.57)$$

$$\approx h^3 \sum_{i,j,k} H^\epsilon(\phi_s^0(\mathbf{x}_{i,j,k})) \mathbf{x}_{i,j,k} ,$$

$$M = \int_{\Omega} \rho^\epsilon(\phi_s^0(\mathbf{x})) \, d\mathbf{x} \quad (7.58)$$

$$\approx h^3 \sum_{i,j,k} \rho^\epsilon(\phi_s^0(\mathbf{x}_{i,j,k})) \mathbf{x}_{i,j,k} ,$$

$$\mathbf{F}^n = -\frac{1}{\Delta t} \int_{\Omega} \rho_s H^\epsilon(\phi_s^{n+1}) (\mathbf{u}_s^n - \mathbf{u}^{*,n+1}) \, d\mathbf{x} \quad (7.59)$$

$$\approx -\frac{h^3}{\Delta t} \sum_{i,j,k} \rho_s H^\epsilon(\phi_s^{n+1}(\mathbf{x}_{i,j,k})) (\mathbf{u}_s^n(\mathbf{x}_{i,j,k}) - \mathbf{u}^{*,n+1}(\mathbf{x}_{i,j,k})) ,$$

$$\mathbf{W}^n = -\frac{1}{\Delta t} \int_{\Omega} \rho_s H^\epsilon(\phi_s^{n+1}) (\mathbf{u}_s^n - \mathbf{u}^{*,n+1}) \times \mathbf{r}^{n+1} \, d\mathbf{x} \quad (7.60)$$

$$\approx -\frac{h^3}{\Delta t} \sum_{i,j,k} \rho_s H^\epsilon(\phi_s^{n+1}(\mathbf{x}_{i,j,k})) (\mathbf{u}_s^n(\mathbf{x}_{i,j,k}) - \mathbf{u}^{*,n+1}(\mathbf{x}_{i,j,k})) \times (\mathbf{x}_{i,j,k} - \mathbf{x}_S^{n+1}) ,$$

ausgewertet worden. Hierbei wird der Geschwindigkeitsvektor \mathbf{u}_s^n im Zellmittelpunkt $\mathbf{x}_{i,j,k}$ mittels linearer Interpolation

$$\mathbf{u}_s^n(\mathbf{x}_{i,j,k}) = \frac{1}{2} \begin{pmatrix} (u_s^n)_{i+\frac{1}{2},j,k} + (u_s^n)_{i-\frac{1}{2},j,k} \\ (v_s^n)_{i,j+\frac{1}{2},k} + (v_s^n)_{i,j-\frac{1}{2},k} \\ (w_s^n)_{i,j,k+\frac{1}{2}} + (w_s^n)_{i,j,k-\frac{1}{2}} \end{pmatrix}$$

aus den Gitterwerten berechnet, analoges gilt für $\mathbf{u}^{*,n+1}$. Hinsichtlich der Zeitdiskretisierung sämtlicher Transportschritte für die Level-Set-Funktionen, wie auch für die zeitliche Evolution der Translations- und Winkelgeschwindigkeit des Starrkörpers aus den Gleichungen (7.46) und (7.47) ist statt des expliziten Euler-Verfahrens ein explizites Adams-Bashforth-Verfahren zweiter Ordnung eingesetzt worden. Die Berechnung der Zeitschrittweite erfolgt analog zu Abschnitt 4.4, wobei innerhalb der Starrkörperphase die Zeitschrittweitereinschränkung nur bezüglich der konvektiven Terme mit der entsprechenden Starrkörperdichte durchzuführen ist.

Im folgenden Abschnitt wird auf die um dynamische Starrkörper erweiterte Parallelisierung aus Abschnitt 4.5 eingegangen, damit aufwendige dreidimensionale Simulationen effizienter berechnet werden können.

7.5 Parallelisierung der Starrkörperdynamik

In diesem Abschnitt wird die Parallelisierungsstrategie der zusätzlich im Zweiphasen-Löser eingebundenen dynamischen Starrkörper beschrieben. Im Unterschied zur Gebietszerlegungsmethode für die Eulersche Diskretisierung des Zweiphasen-Lösers aus Abschnitt 4.5.2 ist hier das Hauptproblem, dass die Starrkörper im semi-Lagrangeschen Sinne diskretisiert werden und dabei im Allgemeinen einzelne Teilgebiete Ω_h^p , $p \in \{1, \dots, P\}$ bezüglich der von der Gebietszerlegung vorgegebenen Aufteilung $\Omega_h = \Omega_h^1 \cup \dots \cup \Omega_h^P$ durchwandern. Dabei muss pro Navier-Stokes-Zeitschritt die Trajektorie jeder Starrkörper-Level-Set-Gitterzelle bis zu ihrem Anfangswert ϕ^0 zurückverfolgt werden.

7.5.1 Gebietsausschnitt als Parallelisierungsstrategie für dynamische Starrkörper

Bei der semi-Lagrangeschen Diskretisierung des Starrkörpertransportes muss die Trajektorie jeder Gitterzelle eines Level-Set-Wertes bis zu ihrem Anfangswert ϕ^0 zurückverfolgt werden. Deshalb durchwandern die Trajektorien im Allgemeinen einzelne Teilgebiete Ω_h^p , $p \in \{1, \dots, P\}$ bezüglich der von der Gebietszerlegung vorgegebenen Aufteilung $\Omega_h = \Omega_h^1 \cup \dots \cup \Omega_h^P$. Eine vollständige Rückverfolgung einer jeden Trajektorie würde somit zu einem hohen Kommunikationsaufwand und einem komplexen Algorithmus führen. Um diese Komplikationen zu umgehen, nutzen wir im Folgenden zwei Tatsachen aus. Erstens nehmen dynamische Starrkörper im Allgemeinen nur einen kleinen Anteil des gesamten Rechengebietes Ω_h ein, und zweitens benötigen wir die Starrkörper-Level-Set-Darstellung nur in einer $O(h)$ -Umgebung, um dessen vollständige Dynamik in den Navier-Stokes-Löser einzubinden. Aufgrund dieser Eigenschaften bietet es sich an, die Level-Set-Darstellung der Starrkörpergeometrie ausschließlich auf die Geometrie inklusive ihrer $O(h)$ -Umgebung einzuschränken. Hierbei muss die $O(h)$ -Umgebung der Starrkörpergeometrie so gewählt werden, dass innerhalb der entsprechenden Gebiets- und Level-Set-abhängigen Glättungsregion $\epsilon_s(\mathbf{x})$ aus Gleichung (4.4) ein WENO-Diskretisierungstern vollständig ausgewertet werden kann. Für unsere numerischen Experimente verwenden wir $c = 1.5$ für $\epsilon_s(\mathbf{x})$ in Gleichung (4.4), und der verwendete WENO-Diskretisierungstern fünfter Ordnung hat einen Radius von drei Gitterzellen. Das heißt, wir benötigen aufgrund der zusätzlichen Berechnung des Gradienten $\nabla\phi_s$ in Gleichung (4.4) insgesamt eine Starrkörper-Umgebung der Größe von 6-7 Gitterzellen; wir verwenden in unseren numerischen Experimenten $O(h) = 7h$.

Die um die Starrkörper-Umgebung $O(h) = 7h$ erweiterte "Bounding-Box" liefert schließlich das lokale Level-Set-Feld der Starrkörpergeometrie in Form eines Gebietsausschnitts aus Ω_h . Daraufhin wird das lokale Level-Set-Feld ϕ_{lok}^0 zusammen mit den entsprechenden Anfangskoordinaten der Bounding-Box des Starrkörpers auf alle für die parallele Simulation verwendeten Prozessoren $1, \dots, P$ verteilt. Abbildung 7.1 veranschaulicht das Vorgehen am Beispiel einer lokalen Level-Set-Funktion für die Starrkörpergeometrie eines Fahrzeugs, welche auf die von der Gebietszerlegung vorgegebenen Aufteilung von zwölf Teilprozessen $\Omega_h = \Omega_h^1 \cup \dots \cup \Omega_h^{12}$ verteilt wird.

Im Laufe der Simulationsberechnung werden in jedem Zeitschritt n die entsprechenden Trajektorien aller Gitterzellenmittelpunkte bezüglich der Starrkörperdynamik (Translation

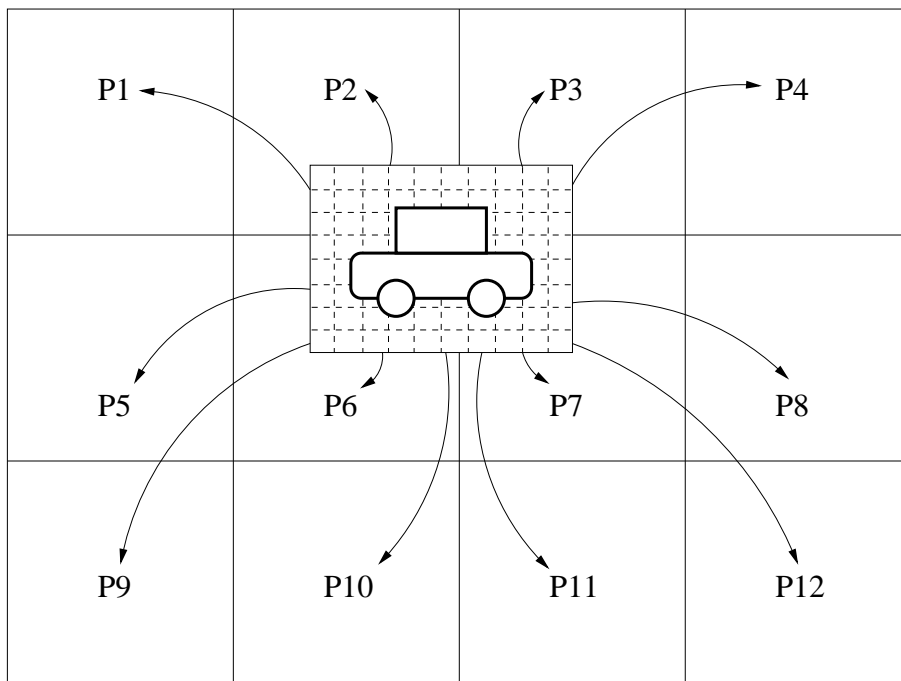


Abbildung 7.1: Verteilung der lokalen Level-Set-Darstellung einer Starrkörpergeometrie auf das mit zwölf Prozessoren parallelisierte globale Rechengitter. Ausserhalb dieses lokalen Gitters ist die Level-Set-Funktion für den Starrkörper konstant und somit nicht explizit erforderlich, womit substantieller Arbeitspeicherplatz gespart werden kann.

und Rotation) nach Gleichung (6.33) aktualisiert und gespeichert. Wird dann der Starrkörper mittels der semi-Lagrangeschen Methode transportiert, so kann das neue Level-Set-Feld bezüglich des Starrkörpers über die Trajektorienwerte schnell mittels folgender Abfrage konstruiert werden:

- Liegt der Trajektorienwert innerhalb der Bounding-Box des Starrkörpers, so ermittle über trilineare Interpolation den entsprechenden Level-Set-Wert aus den gegebenen Werten.
- Liegt der Trajektorienwert nicht innerhalb der Bounding-Box des Starrkörpers, so setze den Level-Set-Wert der Gitterzelle auf den maximalen Level-Set-Wert einer Eckzelle der Bounding-Box. Das heißt, dass ausserhalb der bounding-box alle Level-Set-Werte mit einem konstanten Wert versehen werden. Die Eigenschaft einer Abstandsfunktion ist hier nicht mehr nötig. Es muss aber das richtige Vorzeichen, welches das Innere und das Äußere des Starrkörpers markiert, gesetzt sein.

Ausgehend von der neuen Lage des Starrkörpers werden die bereits beschriebenen parallelen Auswertungsroutinen nach der Gebietszerlegung aus Abschnitt 4.5.2 für die Berechnung der Lage des neuen Starrkörperschwerpunktes aus Gleichung (7.41), für die Berechnung der am Starrkörper wirkenden Kräfte aus Gleichung (7.46) und (7.47) und für die Berechnung des zusätzlichen Druckfeldes p_s innerhalb des Starrkörpers aus Gleichung (7.51) verwendet.

7.6 Vollständiger Algorithmus für inkompressible Zweiphasenströmungen mit dynamischen Starrkörpern

Abschließend sei zur Übersicht der vollständige Algorithmus für Zweiphasenströmungen mit Starrkörpergeometrien aufgeführt:

ALGORITHMUS 2: Paralleler Zweiphasenlöser mit dynamischen Starrkörpern.

1. Initialisiere $\delta x, \delta y, \delta z$ und T .
2. Setze die lokalen Teilgebiete $\Omega^q, q \in \{1, \dots, P\}$ nach (4.48).
3. Setze $t := 0, n := 0, h := \max(\delta x, \delta y, \delta z)$.
4. Setze Anfangswerte für $\mathbf{u}^n := \mathbf{u}^0, p^n := p^0, \phi^n := \phi^0, \phi_s^n := \phi_s^0, \mathbf{v} := \mathbf{v}^0$ und $\boldsymbol{\omega}^n := \boldsymbol{\omega}^0$ auf den Teilgebieten $\Omega^q, q \in \{1, \dots, P\}$.
5. Berechne den Schwerpunkt \mathbf{x}_S^0 und die Masse M des Starrkörpers ϕ_s^0 nach Gleichung (7.57) und Gleichung (7.58).
6. Berechne das Volumen von $V_1(\phi^0)$.
7. Setze $\epsilon(\mathbf{x})$ und $\epsilon_s(\mathbf{x})$ nach (4.4).
8. Setze Randwerte für \mathbf{u}^n auf den lokalen Rändern $\partial\Omega^q \cap \partial\Omega$.
9. ZEIT-SCHLEIFE: Solange $t \leq T$:
 - (a) BERECHNUNG DER ZEITSCHRITTWEITE:
Berechne den lokalen Zeitschritt δt^q nach Abschnitt 4.4.
Berechne den globalen Zeitschritt $\delta t := \min_q \delta t^q$.
 - (b) IMPULSGLEICHUNG I:
Kommuniziere die Randbordürenwerte für \mathbf{u}^n, ϕ^n und ϕ_s^n .
Berechne das Hilfgeschwindigkeitsfeld $\mathbf{u}^*(\mathbf{u}^n, \phi^n, \phi_s^n)$ auf den Teilgebieten $\Omega^q, q \in \{1, \dots, P\}$ nach Gleichung (7.38) mit Gleichung (4.10) als Zeitdiskretisierung.
Kommuniziere die Randbordürenwerte für \mathbf{u}^* .
 - (c) TRANSPORTGLEICHUNG FÜR DEN STARRKÖRPER ϕ_s :
Berechne den Transport der Level-Set Funktion $\phi_s^{n+1}(\mathbf{v}^n, \mathbf{v}^{n-1}, \boldsymbol{\omega}^n, \boldsymbol{\omega}^{n-1})$ nach Abschnitt 6.3.3.
Berechne den Schwerpunkt \mathbf{x}_S^{n+1} des Starrkörpers ϕ_s^{n+1} nach Gleichung (7.57).
 - (d) TRANSPORTGLEICHUNG FÜR DIE ZWEIPHASENOBERFLÄCHE ϕ :
Berechne den Transport der Level-Set-Funktion $\phi^*(\mathbf{u}^n, \phi^n)$ nach Abschnitt 4.2.2.

- (e) Setze $\psi^n := \phi^*$.
- (f) REINITALISIERUNGS-SCHLEIFE: Solange $\tau < 7 \epsilon$:
- i. Setze Randbedingungen für ψ^n und tausche die Randbordürenwerte der Teilgebiete Ω^q , $q \in \{1, \dots, P\}$ aus.
 - ii. Berechne $\psi^{n+\frac{1}{3}}$ auf den Teilgebieten Ω^q , $q \in \{1, \dots, P\}$ nach Gleichung (4.14).
 - iii. Setze Randbedingungen für $\psi^{(n+\frac{1}{3})}$ und tausche die Randbordürenwerte der Teilgebiete Ω^q , $q \in \{1, \dots, P\}$ aus.
 - iv. Berechne $\psi^{n+\frac{2}{3}}$ auf den Teilgebieten Ω^q , $q \in \{1, \dots, P\}$ nach Gleichung (4.14).
 - v. Setze Randbedingungen für $\psi^{(n+\frac{2}{3})}$ und tausche die Randbordürenwerte der Teilgebiete Ω^q , $q \in \{1, \dots, P\}$ aus.
 - vi. Berechne ψ^{n+1} auf den Teilgebieten Ω^q , $q \in \{1, \dots, P\}$ nach Gleichung (4.14).
 - vii. Setze $\tau = \tau + \delta\tau$.
- (g) PICARD-FIXPUNKTITERATION: Solange $|V_1(\phi^0) - V_1(\psi^{n+1})| \leq \epsilon_{vol}$:
 Berechne das Volumen von $V_1(\psi^{n+1})$.
 Korrigiere die Level-Set-Funktion nach Gleichung (4.26).
- (h) Setze $\phi^{n+1} = \psi^{n+1}$, $\hat{p}_{it} := p^n \delta t$, $\epsilon(\mathbf{x})$ und $\epsilon_s(\mathbf{x})$ nach Gleichung (4.4).
- (i) DRUCK-POISSON-GLEICHUNG I:
 Setze die rechte Seite $\mathbf{b} := \nabla \cdot \mathbf{u}^*$ auf den lokalen Gebieten Ω^q , $q=1, \dots, P$, für die Druck-Poisson-Gleichung.
- (j) DRUCK-POISSON-GLEICHUNG II: Solange $\|\mathbf{b} - \mathbf{A}\hat{p}^{it}\|_2 > \epsilon_f$
 Setze Randwerte für den Druck nach Gleichung (7.54).
 Löse $\hat{p}^{n+1}(\mathbf{u}^*, \mathbf{b}, \phi^{n+1}, \phi_s^{n+1})$ nach Gleichung (7.44) mit Jacobi/AMG vorkonditioniertem Krylov-Verfahren.
 Kommuniziere die Randbordürenwerte für \hat{p}_{it} pro Iteration it .
- (k) Setze $p^{*,n+1} := (\delta t)^{-1} \hat{p}_{it}$.
- (l) IMPULSGLEICHUNG II:
 Berechne $\mathbf{u}^{*,n+1}$ nach Gleichung (7.45) auf den Teilgebieten Ω^q .
- (m) STARRKÖRPERGESCHWINDIGKEIT I:
 Berechne \mathbf{F}^n und \mathbf{W}^n nach Gleichung (7.59) und (7.60).
 Berechne \mathbf{v}^{n+1} und $\boldsymbol{\omega}^{n+1}$ nach Gleichung (7.48) und (7.49).
 Berechne die Starrkörpergeschwindigkeit \mathbf{u}_s^{n+1} nach Gleichung (7.51).
- (n) DRUCK-POISSON-GLEICHUNG III:
 Setze die rechte Seite \mathbf{b} von Gleichung (7.51) auf den lokalen Gebieten Ω^q , $q=1, \dots, P$, für die Druck-Poisson-Gleichung.

- (o) DRUCK-POISSON-GLEICHUNG IV: Solange $\|\mathbf{b} - \mathbf{A}\hat{p}_s^{it}\|_2 > \epsilon_s$
Setze Randwerte für den Druck nach Gleichung (7.55).
Löse Poisson-Gleichung (7.51) mit Jacobi/AMG-vorkonditioniertem
Krylov-Verfahren.
Kommuniziere die Randbordürenwerte für \hat{p}_s^{it} pro Iteration it .
- (p) Setze $p_s^{*,n+1} := (\delta t)^{-1} \hat{p}_s$.
- (q) IMPULSGLEICHUNG III:
Berechne \mathbf{u}^{n+1} nach Gleichung (7.52) auf den Teilgebieten Ω^q .
Berechne p^{n+1} nach Gleichung (7.53) auf den Teilgebieten Ω^q .
Setze Randwerte für \mathbf{u}^n auf den lokalen Rändern $\partial\Omega^q \cap \partial\Omega$.
- (r) Setze $t = t + \delta t$ und $n = n + 1$.

10. Datenausgabe des berechneten Ergebnisses.

Kapitel 8

Numerische Ergebnisse für bewegte Starrkörper in Zweiphasenströmungen

In diesem Kapitel untersuchen wir mit unserem Simulationsprogramm uni- und bidirektional gekoppelte Starrkörper mit Zweiphasenströmungen. In den unidirektional gekoppelten Problemen wird der Starrkörper ausschließlich zwangsgeführt, wohingegen in den bidirektional gekoppelten Problemen die auf den Starrkörper wirkenden hydrodynamischen Kräfte berücksichtigt werden.

In Abschnitt 8.1 untersuchen wir zuerst das Herausziehen eines Zylinders aus dem Wasser und anschließend den freien Fall eines Zylinders ins Wasser. Hierbei führen wir Direktvergleiche durch zwischen Labor-Messdaten und Fotos aus dem Artikel von Greenhow und Lin [76] und unseren numerischen Simulationsergebnissen.

In Abschnitt 8.2 untersuchen wir die numerische Konvergenzrate des gesamten Verfahrens. Es werden dabei drei charakteristische Konfigurationen betrachtet. Zunächst zeigen wir, dass die auf eine Kugel wirkenden Kräfte, welche vollständig im Wasser ist und sich im hydrostatischen Gleichgewicht befindet, verschwinden. Danach bestimmen wir die numerische Konvergenzordnung des gesamten bidirektional gekoppelten Verfahrens, indem die numerischen Lösungen auf einer Sequenz von regelmäßig verfeinerten Gittern bestimmt werden und hieraus die Fehlerreduktionsrate ermittelt wird. Als drittes Experiment untersuchen wir die auf eine Kugel wirkenden Kräfte, welche auf der Wasseroberfläche schwimmt und so initialisiert wird, dass sie sich von Beginn an im hydrostatischen Gleichgewicht befindet. Im Gegensatz zu der vollständig im Wasser befindlichen Kugel beobachten wir hier eine aus der Numerik resultierende gitterabhängige Kräftebilanz, welche aufgrund der numerischen Glättungsregion entlang der freien Oberfläche und entlang des Starrkörpers entsteht.

In Abschnitt 8.3 stellen wir Simulationen mit komplexen Starrkörpergeometrien vor. Hierbei erweitern wir unser Simulationsprogramm um ein Large-Eddy-Turbulenzmodell, da sich bewegende komplexe Starrkörper im Allgemeinen starke Turbulenzquellen sind. Die gezeigten Simulationen mit einer Schiffskörpergeometrie und mit einer U-Boot-Geometrie belegen die vielseitige Anwendbarkeit des Simulationsprogramms für ingenieurtechnische und physikalische Problemstellungen.

8.1 Vergleich der Simulationsergebnisse mit Experimentaldaten

Im Folgenden wird das physikalische Verhalten von uni- und bidirektional gekoppelten Starrkörpern in Zweiphasenströmungen am Beispiel eines mit konstanter Geschwindigkeit aufsteigenden Zylinders und eines frei fallenden Zylinders verglichen. Hierbei ist das Setup für die numerischen Simulationen aus einer Publikation von Greenhow und Lin [76] entnommen, das für die Durchführung von physikalischen Laborexperimenten verwendet wurde. Chronologische Fotografien und Messungen aus [76] nutzen wir im Folgenden zum Direktvergleich mit unseren numerischen Ergebnissen, um damit die Verlässlichkeit des numerischen Verfahrens zur Simulation von realen physikalischen Prozessen zu validieren.

8.1.1 Unidirektionale Kopplung: Deformation der freien Oberfläche aufgrund eines mit konstanter Geschwindigkeit aufsteigenden Zylinders

Als erstes numerisches Experiment betrachten wir einen vollständig in ruhendem Wasser befindlichen Zylinder, welcher dieselbe Dichte hat wie das ihn umgebende Wasser und mit konstanter Geschwindigkeit aus diesem gezogen wird. Dies ist ein sehr kompliziertes Problem, zumal die Wasseroberfläche aufgrund der Zylinderdynamik eine starke Deformation erfährt, vergleiche Abbildung 8.1. Im Laborversuch von Greenhow und Lin [76] ist hierbei ein Behälter der Größe $116.8\text{cm} \times 90.2\text{cm} \times 10.2\text{cm}$ (Breite \times Höhe \times Tiefe) eingesetzt worden, wobei dieser bis auf eine Höhe von 30cm mit Wasser gefüllt wurde. Der Zylinder hat einen Durchmesser von 11cm und wird zu Beginn so positioniert, dass der obere Rand des Zylinders 11cm unterhalb der Wasseroberfläche liegt. Dann wird der Zylinder mit einer konstanten Geschwindigkeit von $V_{cyl} = 0.87\text{m/s}$ aus dem Wasser gezogen.

In unserem Simulationsexperiment haben wir ein $116\text{cm} \times 90\text{cm} \times 10\text{cm}$ großes Rechengebiet verwendet, dieses entsprechend mit Wasser und Luft gefüllt und den Zylinder mit dem Durchmesser von 11cm und einer Tiefe von 10cm versehen. Der obere Rand des Zylinders wurde so positioniert, dass er sich 11cm unterhalb der Wasseroberfläche befindet. Es sei hier angemerkt, dass im Artikel von Greenhow und Lin [76] keine exakten Angaben bezüglich der Zylindertiefe gegeben wurde. Der Einfluss der Zylindertiefe ist aber vernachlässigbar, da das Experiment einen zweidimensionalen Charakter annimmt, wie auch von Greenhow und Lin [76] angemerkt. Das Gebiet ist mit einem äquidistanten Gitter, welches aus $232 \times 180 \times 20$ Gitterzellen besteht, aufgelöst worden. Für die Wasserphase wurde die Dichte auf $\rho_w = 998.2\text{kg/m}^3$ und die Viskosität auf $\nu_w = 1.002_{-3}$ und für die Luftphase wurde die Dichte auf $\rho_l = 1.205\text{kg/m}^3$ und die Viskosität auf $\nu_l = 1.81_{-5}$ gesetzt. Für die Glättungsregion $\epsilon(\mathbf{x})$ aus Gleichung (4.4) ist $c = 1.5$ gesetzt worden.

Abbildung 8.1 zeigt den Direktvergleich von gerenderten Bildern unserer numerischen Simulation mit den entsprechenden Fotografien des Laborexperimentes von Greenhow und Lin [76] zu den Zeitpunkten $t = 0.198\text{s}$ und $t = 0.208\text{s}$. Hierbei ist zu beobachten, dass die Lage des Zylinders und der freien Oberfläche sehr gut wiedergegeben wird. Zur Zeit $t = 0.198\text{s}$ tritt der Zylinder bis zur Hälfte seines Durchmessers aus der ursprünglichen

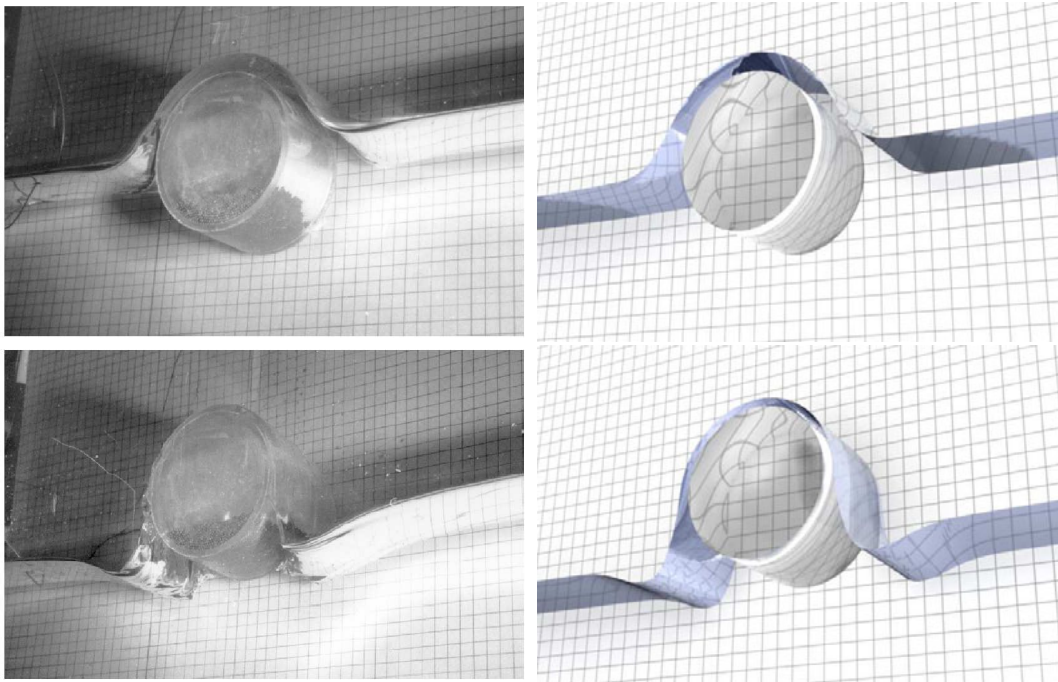


Abbildung 8.1: Direktvergleich des Laborexperiments von M. Greenhow und W.-M. Lin [76] (links) mit unserem numerischen Simulationsergebnis (rechts). Die Bilder zeigen einen mit konstanter Geschwindigkeit aufsteigenden Zylinder von 11cm Durchmesser zu den Zeitpunkten 0.198s (oben) und 0.208s (unten).

Wasseroberfläche hervor. Dies hat zur Folge, dass die freie Oberfläche sich hauptsächlich in einer Umgebung des Zylinders deformiert und ausserhalb dieser Umgebung die Wasseroberfläche nahezu unverändert bleibt. Die Wasserschicht, welche sich zu diesem Zeitpunkt über die obere Hälfte des Zylinders legt, besitzt eine homogene Schichtdicke. Aufgrund der wirkenden Gravitationskraft, der konstanten Zylindergeschwindigkeit und der geringen Viskosität des Wassers wird diese Wasserschicht im Laufe der Zeit zu beiden Seiten des Zylinders abfliessen und kontinuierlich ihre Schichtdicke reduzieren. Dieses zu erwartende Verhalten ist deutlich in dem Bild zu dem späteren Zeitpunkt $t = 0.208\text{s}$ erkennbar. Hier ist der Zylinder nahezu vollständig aus der ursprünglichen Wasseroberfläche herausgetreten und die ihn bedeckende Wasserschicht hat substantiell an Schichtdicke verloren. Dieses abgebaute Wasservolumen ist über die Zeit seitlich an dem Zylinder ins Wasserbecken abgeflossen. Das hat wiederum zur Folge, dass sich zwei Vertiefungen neben dem Zylinder herausbilden. Diese sorgen dafür, dass die ursprünglich horizontale Wasseroberfläche nun eine starke lokale Krümmung aufweist, welche zum Gebietsrand hin kontinuierlich abnimmt. Entscheidend neben der sehr guten Übereinstimmung der Bilder ist auch die Beobachtung, dass die Wasseroberfläche über die gesamte Zeit nicht den Starrkörper durchdringt, das heißt, der Algorithmus gibt die gewünschte Impermeabilität des Starrkörpers richtig wieder.

Abbildung 8.2 zeigt, dass bei fortgesetzter Simulationsrechnung, in der die Wasserschicht sich vollständig um den Zylinder legt und die Schichtdicke gegen Null läuft, die physikalische Eigenschaft der Impermeabilität des Starrkörpers aufrecht erhalten bleibt. Darüber

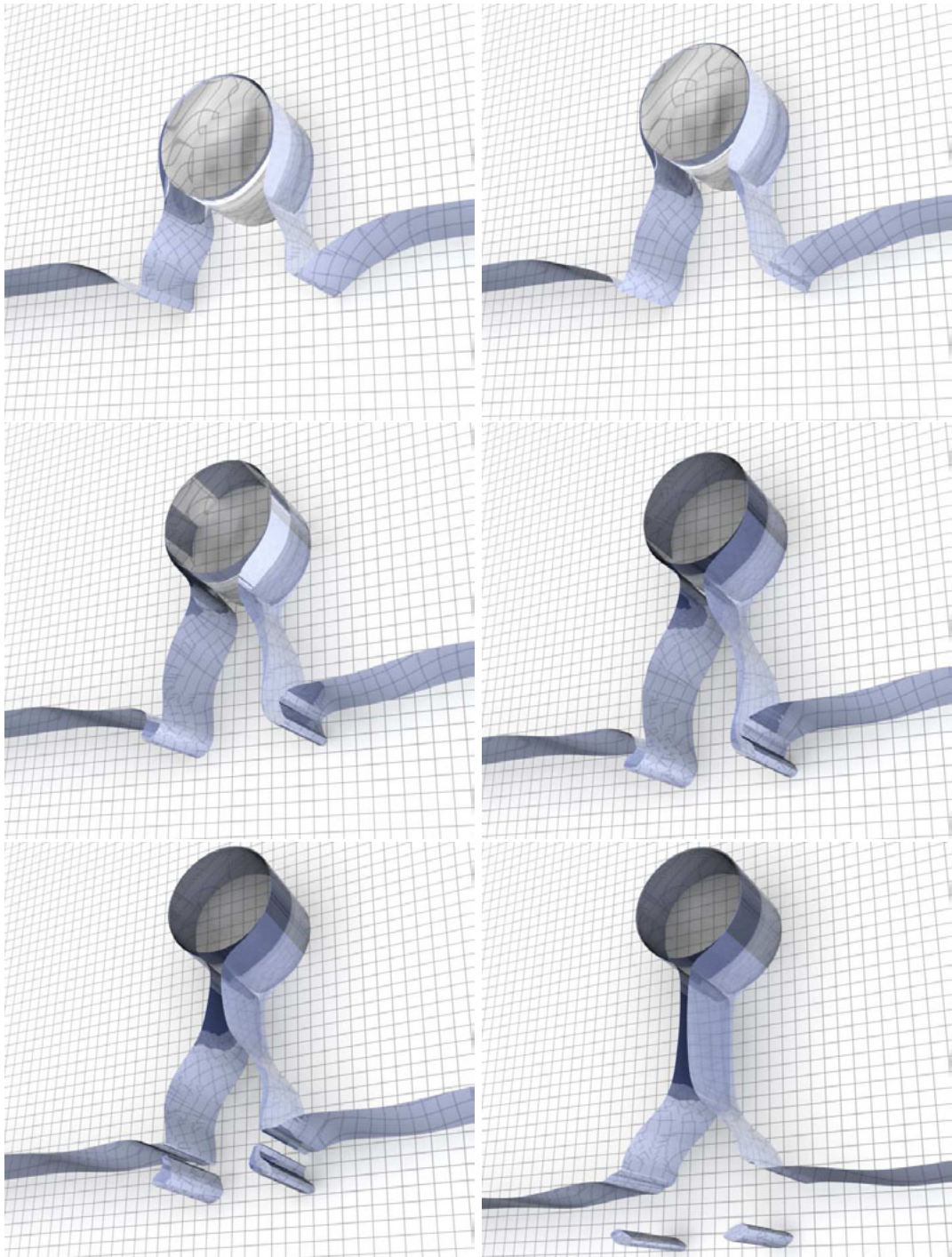


Abbildung 8.2: Der weitere chronologische Verlauf der numerischen Simulation zeigt deutlich das von Peregrine [147] als "abbrechender Wasserfall" (breaking waterfall) bezeichnete physikalische Verhalten.

hinaus wird in Abbildung 8.2 auch das von Peregrine [147] als "abbrechender Wasserfall" (breaking waterfall) bezeichnete physikalische Verhalten, soweit dem Autor bekannt, erstmalig inklusive Topologieveränderung der freien Oberfläche wiedergegeben. Hierbei zieht der Zylinder im zeitlichen Verlauf einen Wasserschweif nach sich, der solange von der restlichen Wasserschicht auf der Zylinderoberfläche genährt wird, bis diese vollkommen abgeflossen ist. Das daraus entstehende Strömungsverhalten liefert den Eindruck eines abbrechenden Wasserfalls, passend zu Peregrines Bezeichnung. Messdaten zu diesem Experiment sind in [76] nicht aufgeführt. Deshalb konnten wir hier einen Vergleich mit dem aufgeführten Experiment nur anhand der Fotografien durchführen. Die zu erwartenden physikalischen Strömungsstrukturen haben wir mit der Simulation wiedergegeben.

Im nächsten Abschnitt wird nun in einem ähnlichen Experimentaufbau eine bidirektionale Kopplung zwischen Zylinder und Wasser untersucht. Dabei taucht der Zylinder aus freiem Fall in das Wasserbecken ein. Dieses Experiment wurde ebenfalls von Greenhow und Lin [76] durchgeführt, wobei sie hier zusätzlich Messungen vorgenommen haben. Diese Messdaten ermöglichen, das Simulationsprogramm über einen bildlichen Vergleich hinaus auch quantitativ mit einem realen Experimentverlauf zu validieren.

8.1.2 Bidirektionale Kopplung: Deformation der freien Oberfläche aufgrund eines frei fallenden Zylinders

Im Folgenden betrachten wir ein weiteres Laborexperiment aus [76]. Hierbei wird ein in ruhendes Wasser frei fallender Zylinder untersucht, dessen Penetrationstiefe bezüglich der Zeit gemessen wird. Dies ermöglicht eine quantitative Validierung der bidirektionalen Kopplung unseres Simulationsprogramms mit realen Daten. In diesem Testfall besitzt der Zylinder ei-

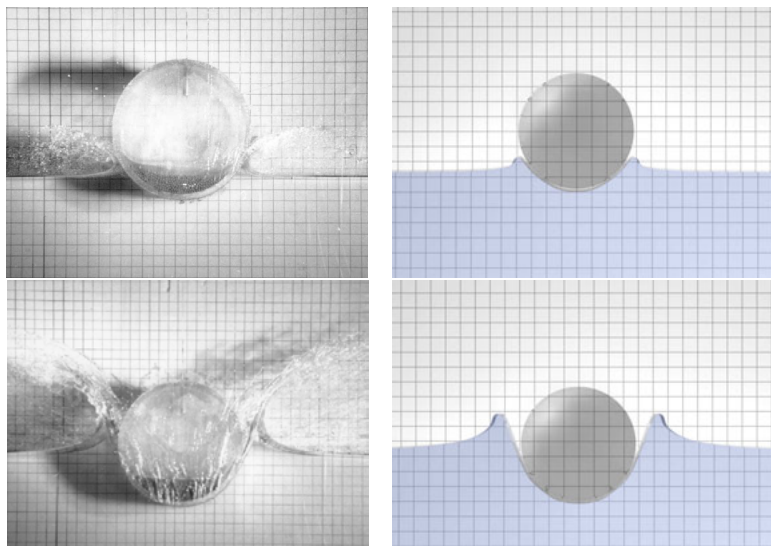


Abbildung 8.3: Direktvergleich der Penetrationstiefe des Zylinders aus dem Laborexperiment von M. Greenhow und W.-M. Lin [76] (links) mit unserem Simulationsergebnis (rechts). Deformation der freien Oberfläche zu den Zeitpunkten $t = 0.305s$ (oben) und $t = 0.335s$ (unten).

ne halb so große Dichte wie das Wasser und induziert somit während des Penetrationsprozesses zusätzliche Auftriebskräfte, die in die Kraftberechnung miteingehen. Zu Beginn der Rechnung wird der Zylinder 44cm über der Wasseroberfläche mittig positioniert. Es sei hier angemerkt, dass in [76] keine exakte Anfangshöhe angegeben wurde, daher wurde hier die Höhe aus dem in einer Tabelle angegebenen Kontaktzeitpunkt zwischen dem Zylinder und dem Wasser ermittelt. Das Wasserbecken selber hat eine Tiefe von 30cm. Die physikalischen Materialparameter für das Wasser und die Luft sind ebenfalls dieselben wie im vorigen Abschnitt. Das Rechengebiet Ω ist mit einer Auflösung von $464 \times 400 \times 7$ Gitterpunkten versehen, was zu einer Auflösung von $44 \times 44 \times 7$ Gitterpunkten für den Zylinder führt. Die Simulationsberechnung ist auf 36 Prozessoren durchgeführt worden.

Abbildung 8.3 zeigt einen graphischen Direktvergleich zwischen den Experimentalfotos und den entsprechenden Simulationsergebnissen zu den Zeitpunkten $t = 0.305s$ und $t = 0.315s$. Zu beobachten ist, dass die Lage des Zylinders und der freien Oberfläche gut übereinstimmen. Allerdings ist die gewählte Gitterauflösung für die Darstellung der Ablösung von kleinen Satellitentropfen zu niedrig.

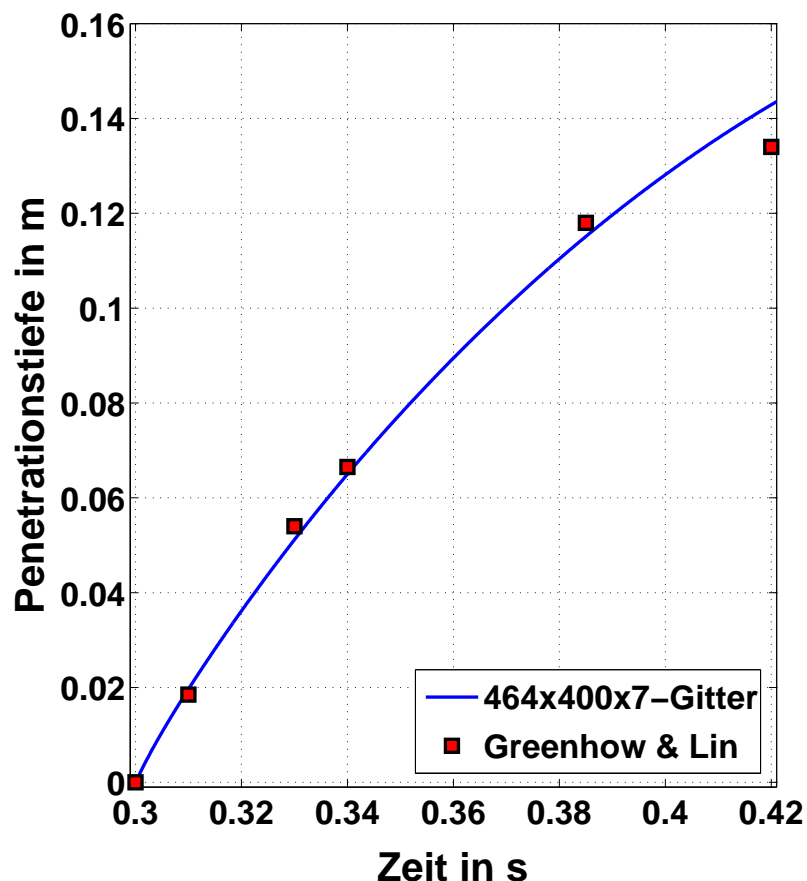


Abbildung 8.4: Direktvergleich der Messdaten des Laborexperiments von M. Greenhow und W.-M. Lin [76] mit unserem Simulationsergebnis.

Weiterhin ist in Abbildung 8.4 ein quantitativer Vergleich bezüglich der Penetrationstiefe des Zylinders über die Zeit aufgeführt. Unsere Simulation zeigt eine sehr gute Übereinstimmung mit den Labormessdaten aus [76]. Es sei hier angemerkt, dass der Zylinder aufgrund seiner geringeren Dichte im Vergleich zu Wasser im weiteren Simulationsverlauf wieder aufsteigt. Dieser Prozess ist aber nicht in den Messtabellen in [76] dargestellt, da dort nicht über den Zeitpunkt der maximalen Penetration hinaus gemessen wurde.

8.2 Konvergenzuntersuchungen eines bidirektional gekoppelten Starrkörpers in Zweiphasenströmungen

In diesem Abschnitt betrachten wir drei vollständig bidirektional gekoppelte Probleme, in denen wir jeweils das Archimedische Prinzip numerisch überprüfen und Konvergenzuntersuchungen durchführen. Das Archimedische Prinzip besagt, dass der Verlust an Gewichtskraft eines Starrkörpers beim Eintauchen in eine Flüssigkeit der Gewichtskraft der Flüssigkeit entspricht, die von dem Starrkörper verdrängt wird. Das bedeutet, dass die Auftriebskraft eines Starrkörpers der Gewichtskraft der verdrängten Flüssigkeit entspricht. Ist daher ein Starrkörper schwerer als die von ihm verdrängte Flüssigkeit, sinkt er, ist er leichter, steigt er, und ist er gleich schwer, befindet er sich im hydrostatischen Gleichgewicht und schwebt.

Diese wichtigen physikalischen Eigenschaften werden wir im Folgenden numerisch untersuchen. Zunächst betrachten wir eine Kugel mit der Dichte von Wasser, die sich vollständig in der Wasserphase und somit im hydrostatischen Gleichgewicht befindet. Wir zeigen numerisch, dass keine künstlichen Geschwindigkeiten aufgrund der Diskretisierung oder des Algorithmus generiert werden. Daraufhin werden wir detaillierte Konvergenzanalysen anhand eines aufgrund geringerer Dichte aufsteigenden quasi-2D-Zylinders durchführen. Wir betrachten einen quasi-2D Zylinder, um eine möglichst hohe Gitterauflösung einsetzen zu können, damit sich eine Konvergenzrate abzeichnen kann. Schließlich betrachten wir in diesem Abschnitt das numerische Verhalten einer Kugel mit der halben Dichte von Wasser im hydrostatischen Gleichgewicht, das heißt, genau die untere Halbkugel befindet sich im Wasser. Entscheidend hierbei ist, dass nun die wirkenden Auftriebskräfte das Gewicht der Kugel bilanzieren müssen. Es wird sich zeigen, dass die Qualität der Bilanzierung gitterabhängig ist.

8.2.1 Stationärer Fall: Kugel im Kräftegleichgewicht innerhalb einer Fluidphase

Als ersten Testfall betrachten wir eine Kugel, welche dieselbe Dichte besitzt wie das sie umgebende Wasser. Abbildung 8.5 veranschaulicht eine solche Anfangskonfiguration. Hierbei besitzt das physikalische Gebiet eine Grösse von $30\text{cm} \times 20\text{cm} \times 20\text{cm}$ (Höhe \times Breite \times Tiefe). Dieses ist, die Kugel mit eingeschlossen, zu $2/3$ mit Wasser gefüllt und zu $1/3$ mit Luft, die entsprechenden Materialparameter der Fluide sind aus Abschnitt. 8.1.1 entnommen. Die Kugel selber hat einen Radius von 3.5cm , und ihr Mittelpunkt wird auf den Punkt $(10\text{cm}, 10\text{cm}, 10\text{cm})$ initialisiert. Das Gebiet wurde mit einer Auflösung von $45 \times 30 \times 30$,

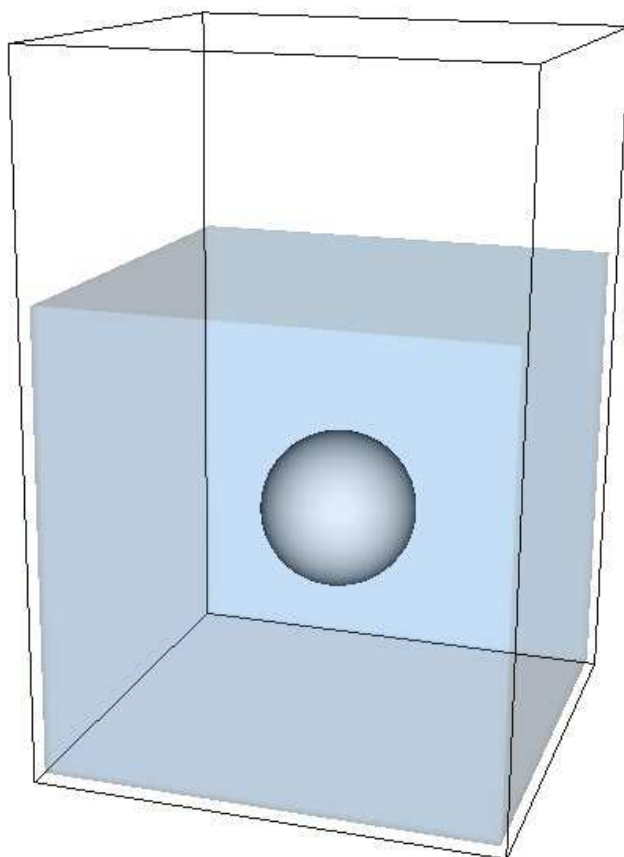


Abbildung 8.5: Initialisierung der Kugel im Kräftegleichgewicht innerhalb der Wasserphase.

$90 \times 60 \times 60$ beziehungsweise $180 \times 120 \times 120$ jeweils äquidistant verteilter Gitterpunkte versehen. Wir erwarten, dass die Kugel über alle Zeiten hinweg ihren stationären Zustand beibehält.

Abbildung 8.6 zeigt die kartesischen Geschwindigkeitskomponenten (u, v, w) , welche am Schwerpunkt der Kugel wirken, über einen Zeitraum von zwei Sekunden. Klar zu erkennen ist, dass sich alle Geschwindigkeitskomponenten über den gesamten Zeitraum hinweg in einer Größenordnung von $\pm 10^{-12} m/s$ bewegen. Es sei angemerkt, dass es essentiell ist, die Druckgleichung sowohl für die Fluide als auch für den Starrkörper bis auf Maschinengenauigkeit zu lösen. Ein frühzeitiger Abbruch des Drucklösers, etwa im Rahmen der Diskretisierungsgenauigkeit, führt zu entsprechend größeren unphysikalischen Geschwindigkeiten des Starrkörpers, die sich insbesondere bei Langzeitrechnungen empfindlich auf die Qualität der Simulationsergebnisse auswirken können.

Nachdem wir die Abwesenheit von Auftriebskräften durch das vorgestellte Experiment numerisch gezeigt haben, untersuchen wir im nächsten Abschnitt die numerische Konvergenzrate am Beispiel eines aufsteigenden Zylinders.

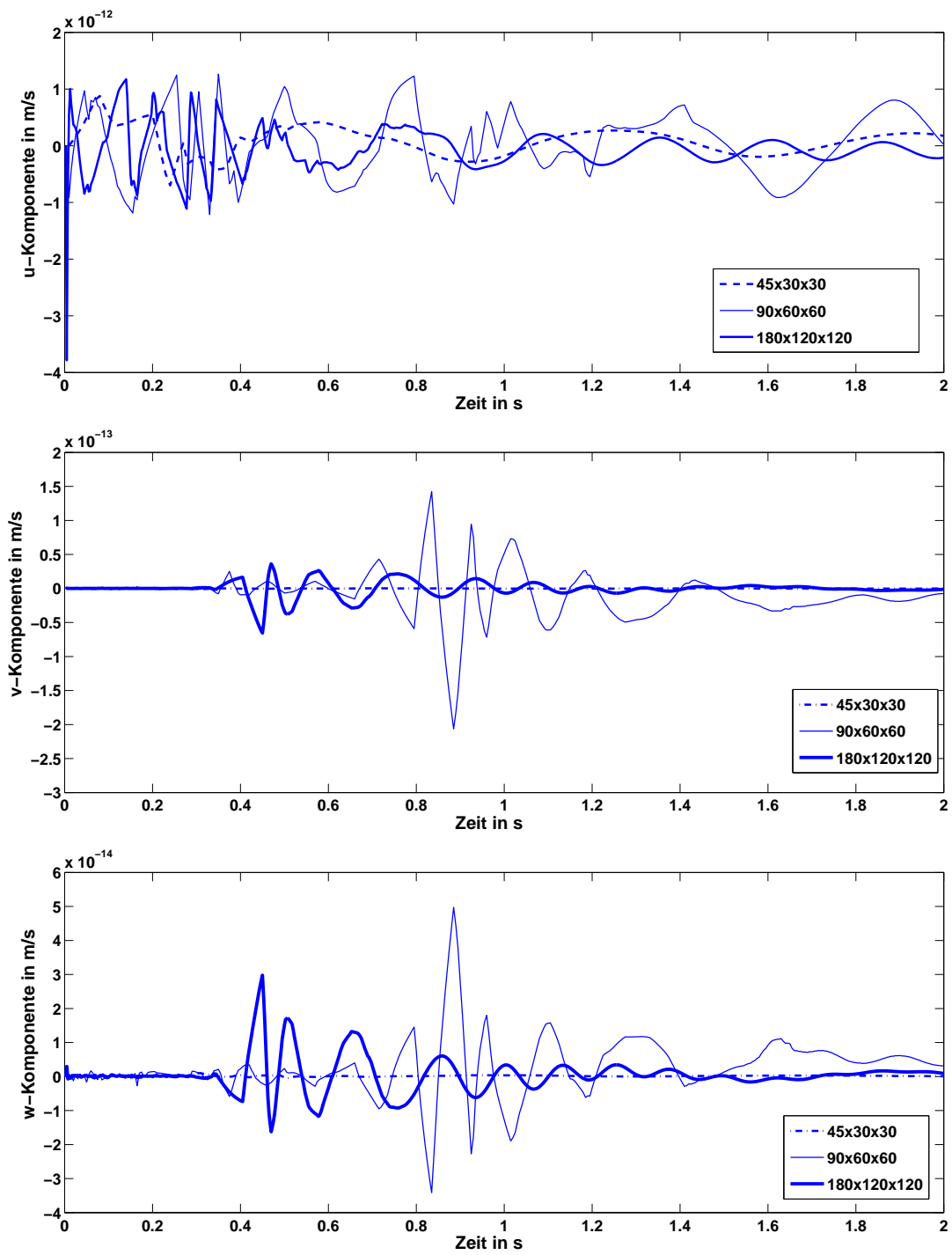


Abbildung 8.6: Auf den Schwerpunkt der Kugel wirkende Geschwindigkeitskomponenten u (oben), v und w (unten), berechnet auf drei unterschiedlich feinen Gitterauflösungen über einen Zeitraum von $2s$.

8.2.2 Dynamischer Fall in quasi-2D: aufsteigender Zylinder

Im Folgenden untersuchen wir die numerische Konvergenzrate des Algorithmus am Beispiel eines quasi-2D-Simulationsproblems. Aufgrund des zweidimensionalen Charakters ermöglicht dieses Testproblem den Einsatz einer sehr feinen Gittermaschenweite, sodass wir schließlich eine gute Aussage über die Konvergenzrate treffen können. Wir betrachten einen aufsteigenden Zylinder, welcher einen Radius von $r = 2.5\text{cm}$ besitzt und sich im Gebiet $\Omega = [0, 30]\text{cm} \times [0, 20]\text{cm} \times [0, 7]\text{cm}$ (Höhe \times Breite \times Tiefe) befindet. Das Gebiet ist einschliesslich des Zylinders bis zu einer Höhe von 20cm mit Wasser gefüllt und darüber befindet sich Luft. Die physikalischen Eigenschaften der betrachteten Fluide sind aus dem Abschnitt 8.1.1 entnommen. Die Anfangslage des Zylinders ist so gewählt, dass sein Mittelpunkt sich in $10\text{cm} \times 10\text{cm} \times 3.5\text{cm}$ befindet. Desweiteren hat der Zylinder eine Dichte von $\rho = 800\text{kg}/\text{m}^3$ und erfährt aufgrund dessen eine Auftriebskraft, welche ihn aufsteigen lässt. Um den 2D-Charakter des Strömungsproblems aufrecht zu erhalten, haben wir periodische Randbedingungen in z-Richtung gesetzt.

Da die Fluidströmung ausschließlich durch die Bewegung des aufsteigenden Starrkörpers angeregt wird, erwarten wir eine Konvergenzordnung, die substantiell durch die Konvergenzordnung für den Starrkörper dominiert wird. Wir antizipieren daher eine Konvergenz erster Ordnung aufgrund der Glättung erster Ordnung der Heaviside-Funktion bezüglich des Starrkörpers und die damit einhergehende Auswertung erster Ordnung der Integrale für die auf den Starrkörper wirkenden Kräfte.

Das Rechengebiet Ω wird mit vier uniform verfeinerten äquidistanten Gittern $30 \times 20 \times 7$, $60 \times 40 \times 7$, $120 \times 80 \times 7$ und $240 \times 160 \times 7$ diskretisiert und dabei wird jeweils die dazugehörige äquidistante Zeitschrittweite $\delta t_h = 8.0_{-3}, 2.0_{-3}, 5.0_{-4}, 1.25_{-4}$ über den gesamten Zeitraum verwendet. Man beachte, dass wir aufgrund des 2D-Charakters der Strömung die

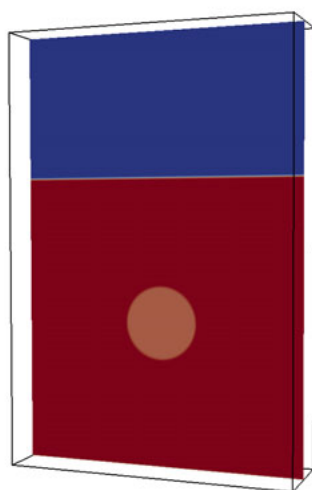


Abbildung 8.7: Mittelschnittebene der Anfangsbedingung bezüglich der Dichteverteilung des aufsteigenden Zylinders im Wasser auf dem $240 \times 160 \times 7$ -Gitter. Der Zylinder hat eine Dichte von $800\text{kg}/\text{m}^3$.

Tabelle 8.1: Konvergenzstudie für den aufsteigenden Zylinder zur Zeit $t = 0.26s$.

dof	$e^{u,h}$	$\eta^{u,h}$	$e^{v,h}$	$\eta^{v,h}$	$e^{p,h}$	$\eta^{p,h}$	$e^{ls,h}$	$\eta^{ls,h}$	$e^{gls,h}$	$\eta^{gls,h}$
Relativer L^2 -Fehler zur Zeit $t = 0.26s$										
4200	6.45_{-1}	—	7.31_{-1}	—	1.96_{-2}	—	5.29_{-3}	—	9.95_{-1}	—
16800	5.51_{-1}	0.11	6.30_{-1}	0.11	1.41_{-2}	0.24	2.96_{-3}	0.42	7.24_{-1}	0.23
67200	4.00_{-1}	0.23	4.95_{-1}	0.17	8.79_{-3}	0.34	1.44_{-3}	0.51	4.12_{-1}	0.41
268800	2.18_{-1}	0.44	2.81_{-1}	0.41	3.83_{-3}	0.60	5.33_{-4}	0.71	1.67_{-1}	0.65

Tabelle 8.2: Relativer L^∞ -Fehler der Aufstiegsgeschwindigkeit des Zylinders zur Zeit $t = 0.26s$.

dof	$e_{zyl}^{u,h}$	$\eta_{zyl}^{u,h}$
4200	4.42_{-1}	—
16800	3.07_{-1}	0.26
67200	1.74_{-1}	0.41
268800	7.02_{-2}	0.65

Gittermaschenweite in z-Richtung nicht verfeinert haben.

Analog zu Abschnitt 5.4 existiert auch hier keine analytische Lösung. Deshalb verwenden wir als Referenzlösung die Lösung, welche auf dem $480 \times 320 \times 7$ -Gitter berechnet wurde, um die Approximationsgüte unseres numerischen Verfahren zu ermitteln. Hierbei berechnen wir den relativen L^2 -Fehler $e_2^{\psi,h}$, wie in (5.9) definiert, für die Geschwindigkeitskomponenten u_h und v_h (w_h ist Null aufgrund des 2D-Charakters des Strömungsproblems) und der Level-Set-Funktionen ϕ_h und ϕ_h^s , und des globalen Druckfeldes p_h . Zusätzlich betrachten wir den relativen L^∞ -Fehler der Aufstiegsgeschwindigkeit des Zylinders.

Wir bestimmen die algebraischen Konvergenzraten nach Gleichung (5.8), so dass bei diesem 2D-Problem die klassische Konvergenz zweiter Ordnung angenommen wird, wenn $\eta = 1$ ist. Wir antizipieren eine Konvergenzordnung, die sich dem Wert $\eta = \frac{1}{2}$ mit feiner werdender Maschenweite annähert, da die Dynamik der Strömung vom Zylinder aus induziert wird. Je feiner das Gitter wird, desto näher wird η an $\frac{1}{2}$ liegen, da der Einfluß der geglätteten Heaviside-Funktion sich auf die gesamte Approximationsgüte auswirkt.

Die gemessenen relativen L^2 -Fehler (5.9) und entsprechenden Konvergenzraten (5.8) sämtlicher berechneter Größen sind in Tabelle 8.1 aufgeführt. Die aufgeführten Werte belegen unser erwartetes Konvergenzverhalten. Für die Geschwindigkeitskomponenten beobachten wir eine Konvergenzrate, welche mit feiner werdender Maschenweite gegen den η -Wert $\frac{1}{2}$ läuft. Die Konvergenzraten für den Druck und die Level-Set-Funktion der Zweiphasenströmung und des Starrkörpers weisen dagegen einen leicht höheren η -Wert als $\frac{1}{2}$ auf.

Da in unserem numerischen Modell die Masse des Zylinders approximiert wird, erwarten wir für die auf den Zylinder wirkenden Auftriebskräfte eine gitterabhängige Konvergenz. Mit anderen Worten, die Aufstiegsgeschwindigkeit des Zylinders sollte gegen einen Grenzwert laufen. Dieses Verhalten geht deutlich aus Tabelle 8.2 hervor, in der der relative L^∞ -Fehler der Aufstiegsgeschwindigkeit des Zylinders aufgeführt ist. Dieser konvergiert mit einem leicht höheren η -Wert als $\frac{1}{2}$.

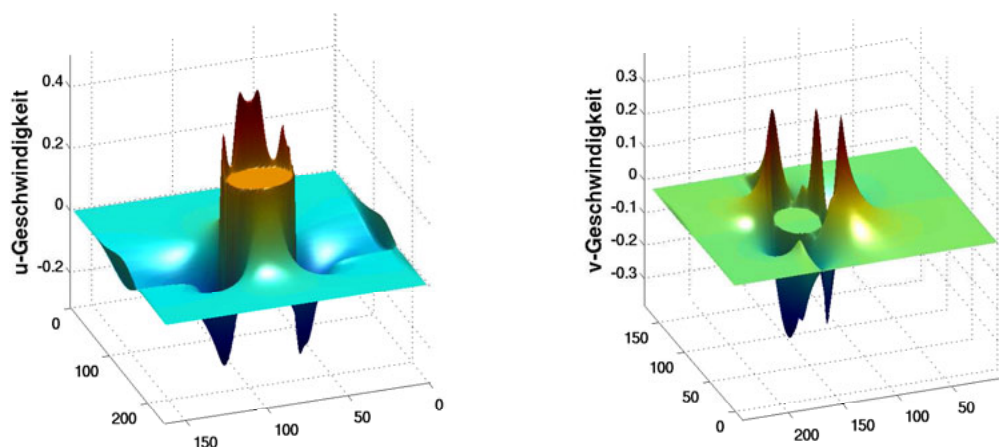


Abbildung 8.8: Graph der Mittelschnittebene von der u - und v -Geschwindigkeitskomponente des aufsteigenden Zylinders, gerechnet auf dem $240 \times 160 \times 7$ -Gitter zur Zeit $t = 0.46s$.

Eine weitere Beobachtung ist, dass die Geschwindigkeit innerhalb des Zylinders in jedem Zeitschritt konstant ist. Abbildung 8.8 zeigt die u - und v -Geschwindigkeitskomponente der Mittelschnittebene des Rechengebietes, welche auf dem $240 \times 160 \times 7$ -Gitter zur Zeit $t = 0.46s$ entnommen wurde. Der Graph der vertikalen Geschwindigkeitskomponente u im linken Bild von Abbildung 8.8 veranschaulicht, wie der Zylinder zur Zeit $t = 0.46s$ mit einer Geschwindigkeit von $u = 0.2185m/s$ aufsteigt. Je nach Zylindergeschwindigkeit und Wahl der Materialparameter der Fluide bilden sich entlang der Zylinderoberfläche Grenzschichten aus, in denen die Fluidgeschwindigkeiten lokal stark variieren können. Weiterhin zeigt der Graph der horizontalen Geschwindigkeitskomponente v im rechten Bild von Abbildung 8.8, dass der Geschwindigkeitsbeitrag innerhalb des Zylinders exakt Null ist. Am Zylinderrand aber nimmt v seine Extremalwerte an, die mit wachsendem Abstand vom Zylinderrand stark abklingen.

8.2.3 Stationärer Fall: Kugel im Kräftegleichgewicht an der freien Oberfläche

Im Folgenden betrachten wir eine Kugel mit der halben Dichte von Wasser im hydrostatischen Gleichgewicht. Gemäß des Archimedischen Prinzips müßte die Kugel für alle Zeiten zur Hälfte ins Wasser eingetaucht ruhen. Da aber im Rahmen der Numerik die Masse der Kugel und der Dichteübergang zwischen den zwei Fluidphasen in Abhängigkeit von der Gittermaschenweite mit einer Genauigkeit erster Ordnung approximiert wird, erwarten wir ein entsprechendes Ungleichgewicht zwischen dem Gewicht des verdrängten Wassers und dem Gewicht der Kugel. Dieses Ungleichgewicht führt dazu, dass die Kugel auf der Wasseroberfläche in Abhängigkeit von der Gittermaschenweite leicht vertikal zu oszillieren beginnt,

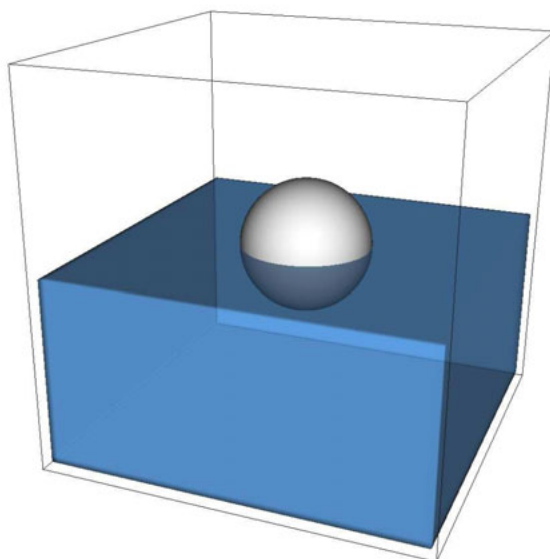


Abbildung 8.9: Simulationstest des Kräftegleichgewichts einer Kugel auf der freien Oberfläche. Zu Beginn der Simulation liegt die Kugel im hydrostatischen Gleichgewicht zur Hälfte im Wasser.

also der kinematische Zustand den stationären Zustand approximiert. Aufgrund der Massenapproximation bezüglich der Kugel und der Zweiphasenapproximation erwarten wir darüber hinaus, dass die Amplitude der Oszillationen mit feiner werdender Gittermaschenweite entsprechend von erster Ordnung gegen den kinematisch stationären Zustand, das heißt gegen Null, konvergiert.

Für dieses numerische Experiment verwenden wir das Gebiet $\Omega = [0, 30\text{cm}]^3$, welches Wasser, Luft und die Kugel enthält. Die bis zur Hälfte im Wasser eingetauchte Kugel hat ihren Mittelpunkt in $(15\text{cm}, 15\text{cm}, 15\text{cm})$, besitzt einen Radius von 5cm und hat die halbe Dichte von Wasser. Das Rechengebiet Ω wird mit drei uniform verfeinerten äquidistanten Gittern bestehend aus 40^3 , 80^3 beziehungsweise 160^3 Zellen diskretisiert. Die entsprechenden Zeitschrittweiten werden auf jedem Gitter über die Zeitschrittweitensteuerung aus Abschnitt 4.4 berechnet. Abbildung 8.10 zeigt die Mittelschnittebene der Materialdichte für diesen Testfall in den drei betrachteten Gitterauflösungen 40^3 , 80^3 und 160^3 . Diese Bilder veranschaulichen, wie die Masse der Kugel über ihre geglättete Dichteverteilung mit feiner werdendem Gitter approximiert wird. Diese Approximation konvergiert gegen das Gewicht des von der Kugel verdrängten Wasservolumens. Desweiteren erkennt man auch die Approximation des Dichteübergangs zwischen Wasser und Luft. Diese trägt gleichfalls zu dem Kräftegleichgewicht bei, da nur im Fall einer scharfen Trennfläche für die Dichte beider Flüssigkeiten die Kugel exakt zentriert an der Wasseroberfläche schwimmen würde. Mit anderen Worten, die Approximation der globalen Dichteverteilung sorgt dafür, dass auch das Kräftegleichgewicht von Schwimmkörpern approximativ eingefangen werden kann. Diese Vorhersage des physikalischen Verhaltens wird in Abbildung 8.11 bestätigt, die den zeitlichen Verlauf des Kugelmittelpunktes bezüglich der drei verschiedenen Gitterauflösungen zeigt. Hierbei entspricht die gestrichelte Linie dem Verlauf des Kugelmittelpunktes auf dem

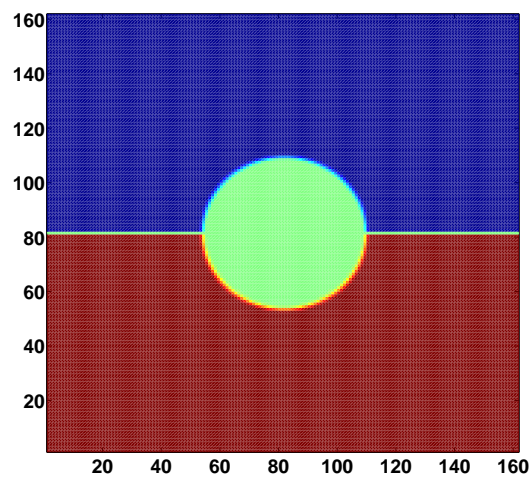
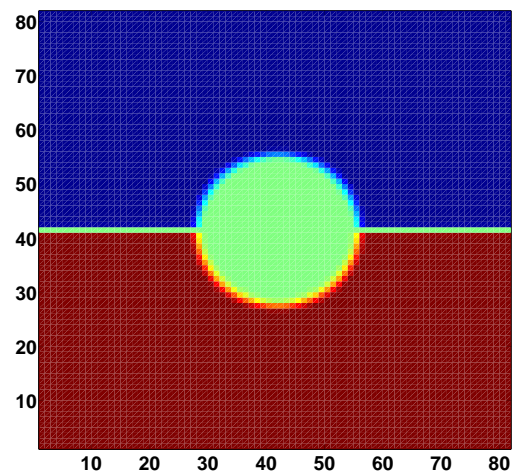
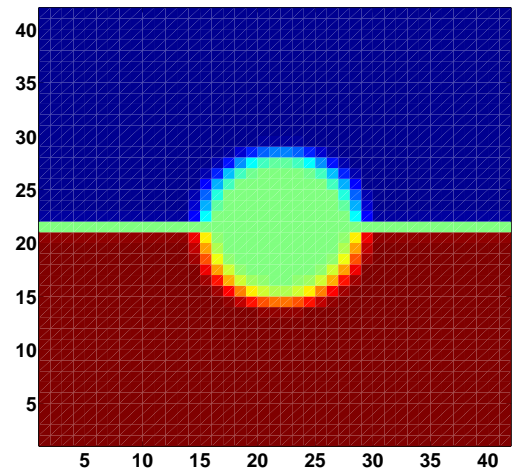


Abbildung 8.10: Mittelschnittebene der Materialdichte für den Testfall der Kugel an der freien Oberfläche in den drei betrachteten Gitterauflösungen 40^3 , 80^3 und 160^3 . Die blaue Farbe ist die Dichte der Luft, rot die des Wassers und grün die der Starrkörperkugel.

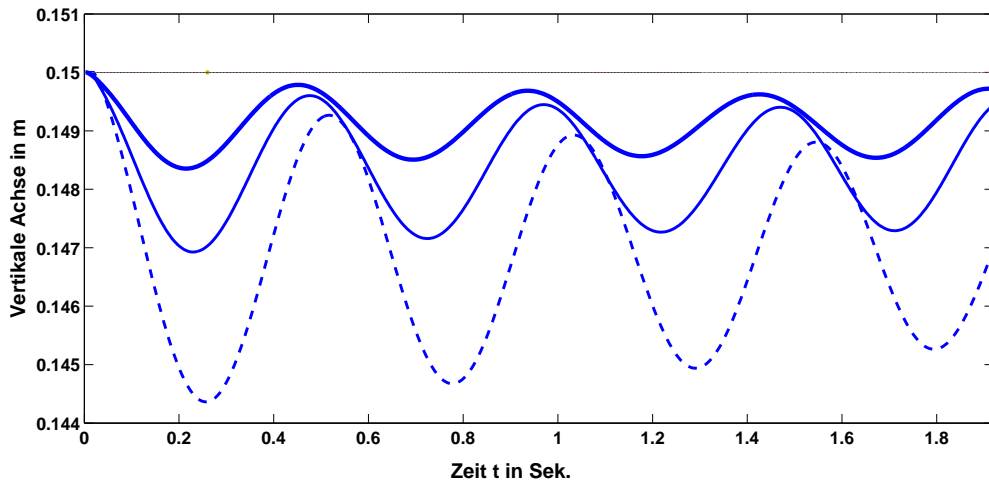


Abbildung 8.11: Untersuchung des Kräftegleichgewichts einer an der freien Oberfläche schwimmenden Kugel. Abgebildet ist die vertikale Bewegung des Kugelmittelpunktes bezüglich drei unterschiedlich feinen Gittern (gestrichelte Linie= 40^3 , durchgezogene Linie= 80^3 und fette Linie= 160^3 Gitterpunkte) über einen Zeitraum von 1.9 Sekunden. Die Wasseroberfläche befindet sich hierbei auf 0.15m Höhe (gerade Linie).

größten Gitter mit 40^3 -Gitterpunkten und die fette durchgezogene Linie dem Verlauf auf dem feinsten Gitter mit 160^3 -Gitterpunkten. Die exakte Lösung für den Kugelmittelpunkt ist eine gerade Linie mit dem konstanten Wert 15cm . Aufgrund der Approximation des globalen Dichtefeldes äußert sich das Kräftegleichgewicht in einem vertikal oszillierenden Verhalten der Kugel. Die Amplitude der Oszillationen konvergiert bei Gitterverfeinerung gegen Null von der Ordnung $O(h)$, denn wie aus Abbildung 8.11 ablesbar ist, führt eine Halbierung der Gittermaschenweite etwa zu einer Halbierung der Amplitude. Der Grund dafür, dass die Oszillationen ausschließlich unterhalb der Wasseroberfläche stattfinden, beruht auf der Dichteglättung entlang der Kugeloberfläche und entlang der Fluidoberfläche und der entsprechend wirkenden Gravitation.

8.3 Simulationsergebnisse mit komplexen Geometrien und Large-Eddy-Turbulenzmodell

Nachdem wir in den letzten beiden Abschnitten einerseits unsere Simulationsergebnisse mit Experimentaldaten verglichen haben und andererseits die drei physikalischen Hauptzustände einer Starrkörperkugel in Zweiphasenströmungen hinsichtlich ihrer Konvergenzordnung untersucht haben, zeigen wir im Folgenden praxisnähere Anwendungen. Mit anderen Worten, im Gegensatz zu der bisher betrachteten Zylinder- und Kugelgeometrie werden wir nun deutlich komplexere Geometrien einsetzen.

Als erste Geometrie betrachten wir die eines Schiffes, welches unidirektional, das heißt zwangsgeführt, mit dem Wasser gekoppelt wird. Das Schiff wird im ersten Experiment mit

einer vorgegebenen Rotationsgeschwindigkeit um seinen Mittelpunkt gedreht und fährt in einem anschließenden Experiment mit einer vorgegebenen Translationsgeschwindigkeit geradeaus.

Als zweite Geometrie betrachten wir die eines U-Bootes, welches bidirektional mit dem Wasser gekoppelt wird. In diesem Fall wird kein Geschwindigkeitsfeld vorgegeben. Statt dessen werden die Geschwindigkeiten aus den wirkenden hydrodynamischen Kräften in jedem Zeitschritt neu ermittelt. Hierbei wird das U-Boot mit einer Dichte versehen, die kleiner ist als die des Wassers, sodass es schließlich aufgrund der Auftriebskräfte aufsteigt.

Im Zusammenhang mit dem Einsatz komplexer Starrkörpergeometrien wird auch die Verwendung eines Turbulenzmodells unumgänglich, denn je komplexer eine Starrkörpergeometrie ist, desto stärker wirkt ihre Oberfläche im Verlauf von Fluid-Starrkörper-Wechselwirkungen als Turbulenzquelle. In [180] haben wir unser Simulationsprogramm um ein Large-Eddy-Turbulenzmodell erweitert. Dieses Large-Eddy-Modell wird zur Simulation der folgenden beiden Zweiphasen-Starrkörper-Wechselwirkungsprobleme miteinbezogen, da es sich für die hier untersuchten ingenieurstechnischen Anwendungen im Hinblick auf Rechenaufwand, Genauigkeit und Parallelisierbarkeit besonders gut eignet.

Im Rahmen der Large-Eddy-Turbulenz-Darstellung verwenden wir das Smagorinsky-Modell [178] als Feinstruktur-Approximation für den Spannungstensor. Dieses ergibt sich mittels Anwendung eines Raumfilters auf die Navier-Stokes-Gleichungen

$$\rho(\partial_t(\bar{\mathbf{u}}) + \nabla \cdot (\bar{\mathbf{u}} \otimes (\bar{\mathbf{u}}))) + \nabla \bar{p} = \nabla \cdot \mu \bar{\mathbf{D}} + \rho \mathbf{g} - \nabla \cdot \boldsymbol{\tau} \quad (8.1)$$

wobei $\bar{\mathbf{u}}$ und \bar{p} die gefilterten Größen sind. Gleichung (8.1) enthält nun den zusätzlichen Feinstruktur-Spannungstensor $\boldsymbol{\tau} = [\tau_{ij}]$. In Analogie zum molekülbedingten Impulstransport, der sich bei laminaren Strömungen Newtonscher Fluide als Produkt von Viskosität und Deformationstensor ausdrücken lässt, wird für den durch die Feinstruktur-Turbulenz bedingten Transport ebenfalls eine Proportionalität zwischen dem Feinstruktur-Tensor τ_{ij} und dem Grobstruktur-Deformationstensor

$$\bar{D}_{ij} = \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \quad (8.2)$$

angenommen. Laut Smagorinsky [178] ist die Proportionalität über die Wirbelviskosität ν_t gegeben via

$$\tau_{ij} = -\nu_t \bar{D}_{ij} \quad \text{mit} \quad \nu_t = l^2 |\bar{D}| \quad \text{und} \quad \bar{D} = \sqrt{\frac{1}{2} \bar{D}_{ij} \bar{D}_{ij}}, \quad (8.3)$$

wobei l die charakteristische Feinstruktur-Länge ist, welche sich aus der lokalen Filtergröße Δ und der Smagorinsky-Konstanten C_s zu

$$l = C_s \Delta \quad (8.4)$$

ergibt. Hierbei bestimmt die lokale Filtergröße den Einflussgrad des Turbulenzmodells auf die Navier-Stokes-Gleichungen und ist gitterabhängig zu wählen. Die Diskretisierung des Feinstruktur-Spannungstensors verläuft über herkömmliche zentrale Differenzen, welche als zusätzlicher Rechenschritt bei der Berechnung der Navier-Stokes-Impulsgleichungen anfallen. Die Wahl der Smagorinsky-Konstanten C_s hängt im Allgemeinen stark von dem



Abbildung 8.12: Das Forschungsschiff RV-Tangaroa in Fahrt. Man beachte, dass aufgrund des Wustbugs kaum Bugwellen generiert werden. Das Bild stammt aus Quelle [1].

Strömungsproblem (homogen, inhomogen, isotrop, anisotrop), den Fluiden (Viskosität, Dichte) und den betrachteten Starrkörpergeometrien ab. Typische Werte für die Smagorinsky-Konstante, die sich bei praktischen Anwendungen des Modells bewährt haben, liegen im Bereich $0.065 \leq C_s \leq 0.1$, wir verwenden in unseren Simulationen den Wert 0.1 für beide Fluidphasen. Man beachte, dass wir für die beiden Phasen auch verschiedene Smagorinsky-Konstanten verwenden können, die entlang der freien Oberfläche glatt ineinander überführt werden. Die lokale Filterweite Δ wählen wir als

$$\Delta = (\Delta x \cdot \Delta y \cdot \Delta z)^{\frac{1}{3}} = \Delta x = \Delta y = \Delta z, \quad (8.5)$$

da wir in den betrachteten Fällen ein äquidistantes Diskretisierungsgitter verwenden.

8.3.1 Unidirektionale Kopplung: Rotation und Translation des Schiffes RV-Tangaroa

In diesem Abschnitt betrachten wir zwei Zweiphasen-Simulationen mit der Geometrie des Forschungsschiffes RV-Tangaroa [7]. Das erste Experiment, die Schiffsrotation, wurde aufgrund fehlender Realdaten in Modellskalen, das zweite Experiment, die Schiffstranslation hingegen, in Realskalen gerechnet. In beiden Experimenten wird für die Glättungsregion $\epsilon(\mathbf{x})$ aus Gleichung (4.4) erneut $c = 1.5$ gesetzt. Die Dynamik des Schiffes wird unidirektional mit dem Wasser gekoppelt, das heißt, die Rotations- und Translationsgeschwindigkeit des Schiffes werden als zeitabhängige Funktionen explizit vorgeschrieben. Das Schiff wird im ersten Simulationsexperiment mit einer vorgegebenen Rotationsgeschwindigkeit von 0.5

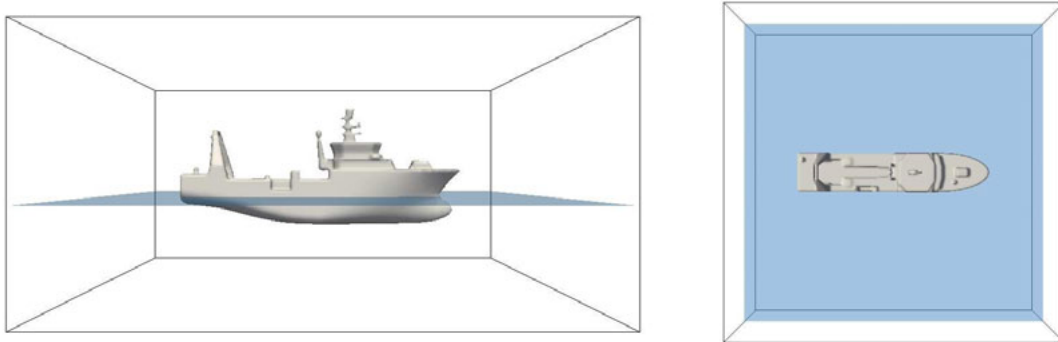


Abbildung 8.13: Rechengebiet und Ausgangsposition der RV-Tangaroa für die Simulation der Schiffs-Rotation.

Umdrehungen pro Sekunde um den Mittelpunkt des Gebietes Ω gedreht und in einem zweiten Simulationsexperiment fährt es geradeaus mit einer vorgegebenen Translationsgeschwindigkeit von 12 Knoten.

Rotation der RV-Tangaroa

Das Schiff hat in diesem Experiment eine Länge von 25.0cm , eine Breite von 5.0cm und eine Höhe von 10.4cm . Unser betrachtetes Rechengebiet für die Schiffsrotation hat die Größe $40\text{cm} \times 40\text{cm} \times 20\text{cm}$ und wird durch ein äquidistantes Gitter mit $200 \times 200 \times 100$ Gitterpunkten diskretisiert. Zu Beginn ist das Schiff in der Mitte des Gebietes positioniert und der Wasserpegel ist auf eine Höhe von 8cm gesetzt, sodass der Wulstbug des Schiffes sich vollständig im Wasser befindet. Wir verwenden Wasser und Luft mit den Materialparametern aus Abschnitt 8.1.1. Für die Glättungsregion $\epsilon(\mathbf{x})$ aus Gleichung (4.4) ist $c = 1.5$ gesetzt worden. Die Simulation wird bis zur physikalischen Zeit $t = 2\text{s}$ durchgeführt. Wir betrachten also eine vollständige Rotation des Schiffskörpers um den Mittelpunkt des Gebietes Ω . Diese Simulation ist parallel auf 24 Prozessoren berechnet worden.

Abbildung 8.13 zeigt die Startkonfiguration des Simulationsproblems. Der Wulstbug befindet sich vollständig unter Wasser und die RV-Tangaroa ist mittig im Rechengebiet positioniert. Aufgrund dieser Lage des Schiffes erwarten wir, dass sich zu Beginn der Rotation auf der Steuerbordseite am Wulstbug ein starkes Unterdruckgebiet bildet, welches einen tiefen Wasserwirbel erzeugt, und auf der Backbordseite am Wulstbug das Wasser bis zu einem gewissen Grad angestaut wird und dort eine Bugwelle bildet. Der flache Anstieg der Heckgeometrie sorgt dafür, dass sich bei der Rotation des Schiffes das Wasser auf der Steuerbordseite am Hecks staut und anhebt bis es schließlich in Wellenform um das Schiff herum ausläuft. Auf der Backbordseite am Heck entsteht im Gegensatz zu der Strömung am Wulstbug kein ausgezeichneter Wasserwirbel, da dort die Geometrie deutlich flacher ist und somit keinen starken Unterdruck produzieren kann, welcher zu einem Wasserwirbel führen würde. Die Hauptströmungsbereiche während der Rotation der RV-Tangaroa entstehen am Bug- und am Heckende des Schiffes, da hier aufgrund des Drehmoments die stärksten Kräfte

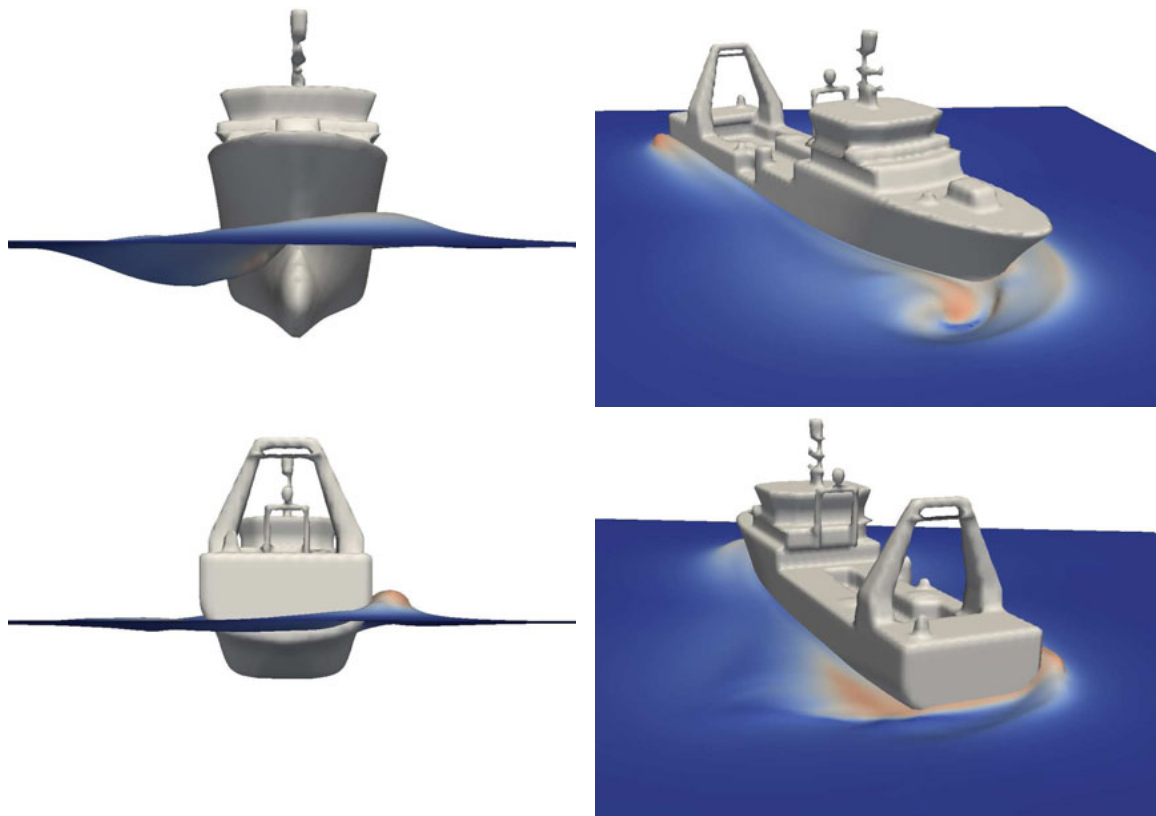


Abbildung 8.14: Bug- und Heckansicht der rotierenden Schiffskörpergeometrie aus Frontal- und Vogelperspektive zur Zeit $0.15s$. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, die mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist.

auf das Wasser und die Luft übertragen werden. Insbesondere erwarten wir daher, dass das Large-Eddy-Turbulenzmodell genau in diesen Bereichen einen entsprechend höheren Anteil an turbulenter Viskosität generiert.

Abbildung 8.14 zeigt die Bug- und Heckansicht der rotierenden Schiffskörpergeometrie aus Frontal- und Vogelperspektive zur Zeit $0.15s$. Wie erwartet, bildet sich auf den Seitenabschnitten der Schiffsgeometrie, welche der Rotationsrichtung zugewandt sind, aufgrund des Staudrucks eine Welle heraus. Auf den Seitenabschnitten der Schiffsgeometrie, welche der Rotationsrichtung abgewandt sind, beobachten wir das erwartete Absinken der Wasseroberfläche bedingt durch den aufgrund der Rotation entstehenden Unterdruck. Hierbei führt erwartungsgemäß die Geometrie des Wulstbuchs dazu, dass sich auf der Bugseite des Schiffes ein deutlich stärkerer Wasserwirbel abzeichnet als dies im Heckbereich der Fall ist. In den Abbildungen 8.15, 8.16 und 8.17 ist der zeitliche Verlauf der vollständigen Rotation aus der Vogel- und Fischperspektive aufgeführt. Diese zeigen sowohl die Schiffskörpergeometrie als auch die Wasseroberfläche, welche mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist. Dabei ist zu erkennen, dass auf der freien Oberfläche der zwei Phasen von der Bug- und Heckgeometrie Geschwindigkeiten von bis zu $0.6 \frac{m}{s}$ generiert werden, wobei die am Bug generierten Geschwindigkeiten erwartungsgemäß stärker sind als die am Heck.

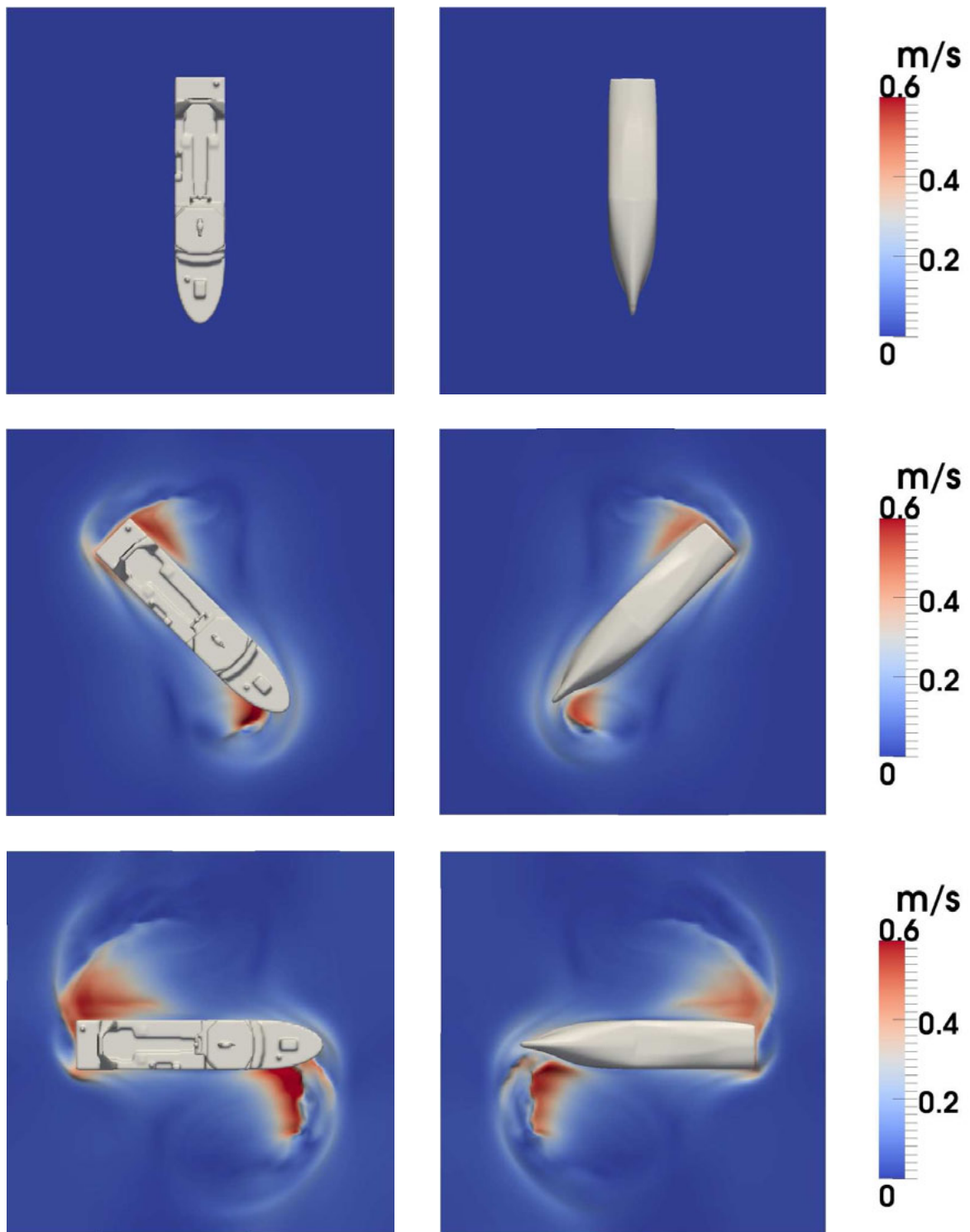


Abbildung 8.15: Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu den Zeiten $t = 0s, 0.25s, 0.5s$. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist.

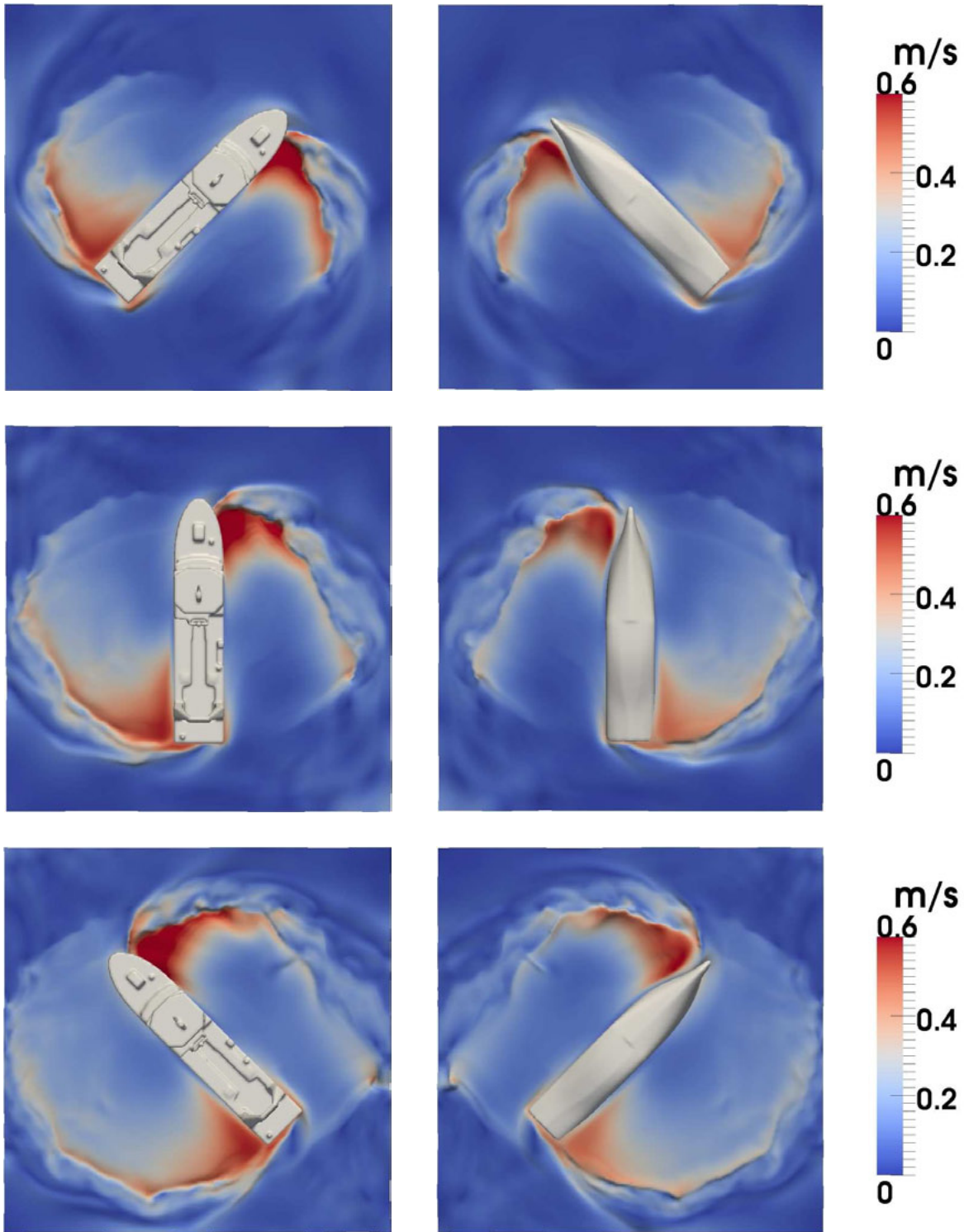


Abbildung 8.16: Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu den Zeiten $t = 0.75s, 1.0s, 1.25s$. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist.

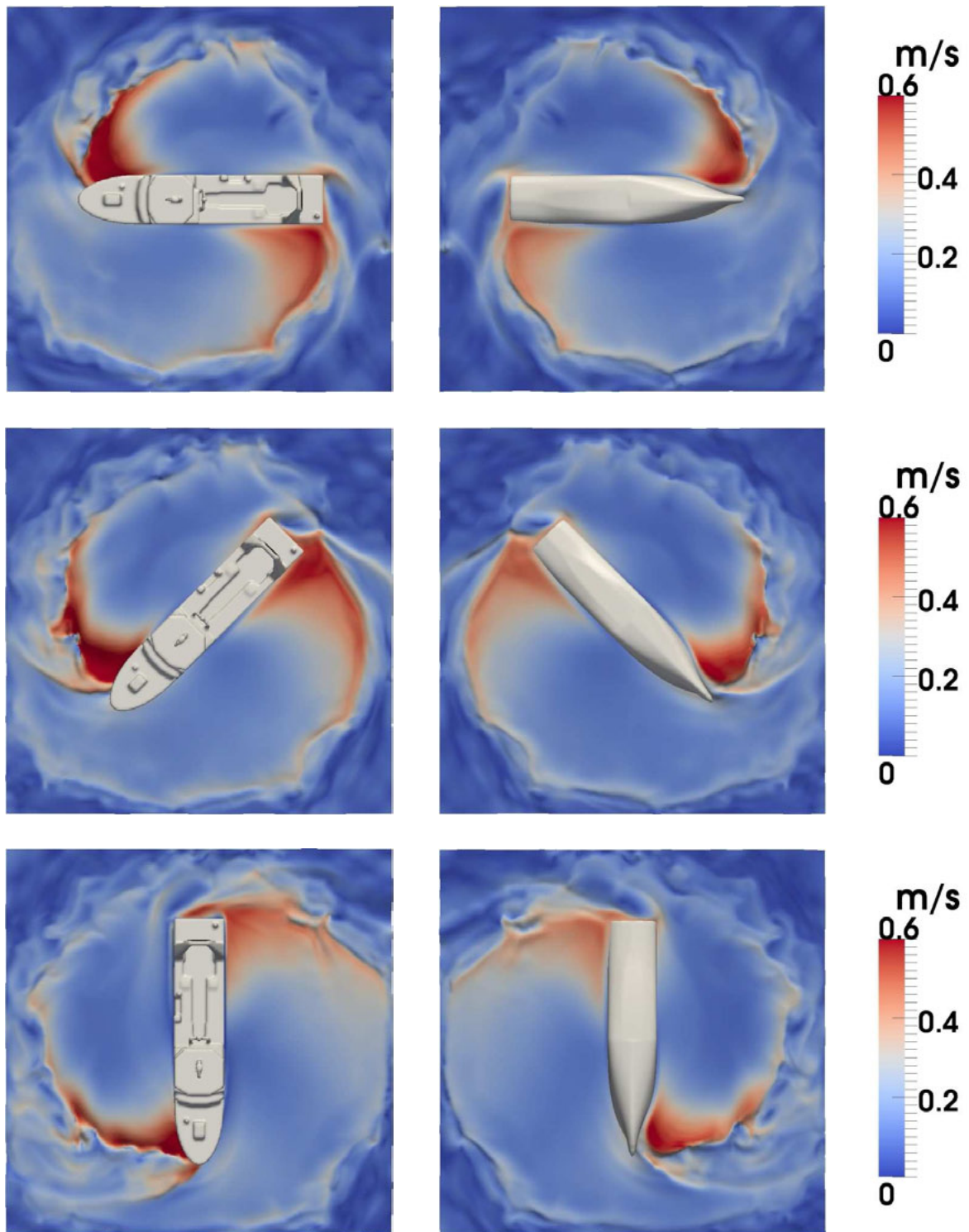


Abbildung 8.17: Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu den Zeiten $t = 1.5s, 1.75s, 2.0s$. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist.

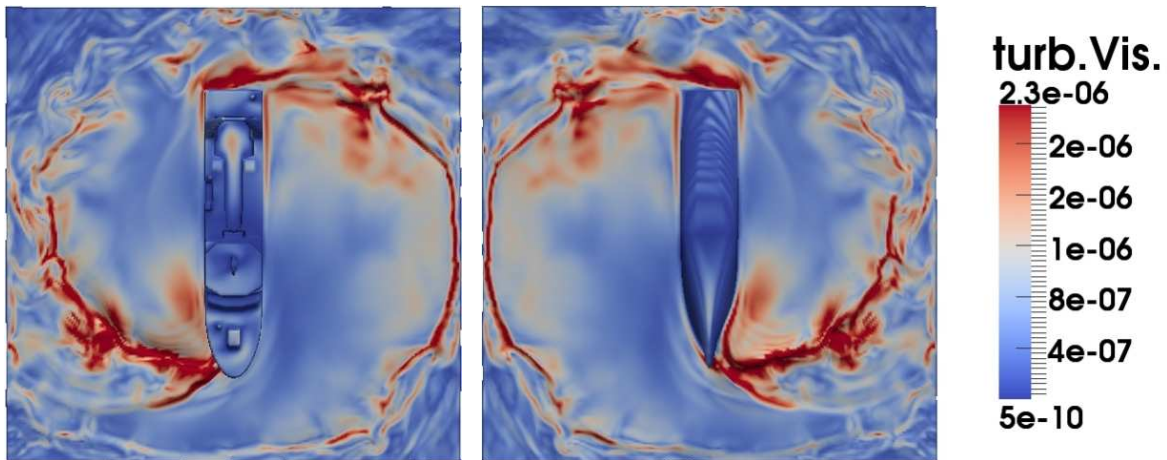


Abbildung 8.18: Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu der Zeit 2.0s. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit den Werten der turbulenten Viskosität (Eddy-Viskosität) koloriert sind.

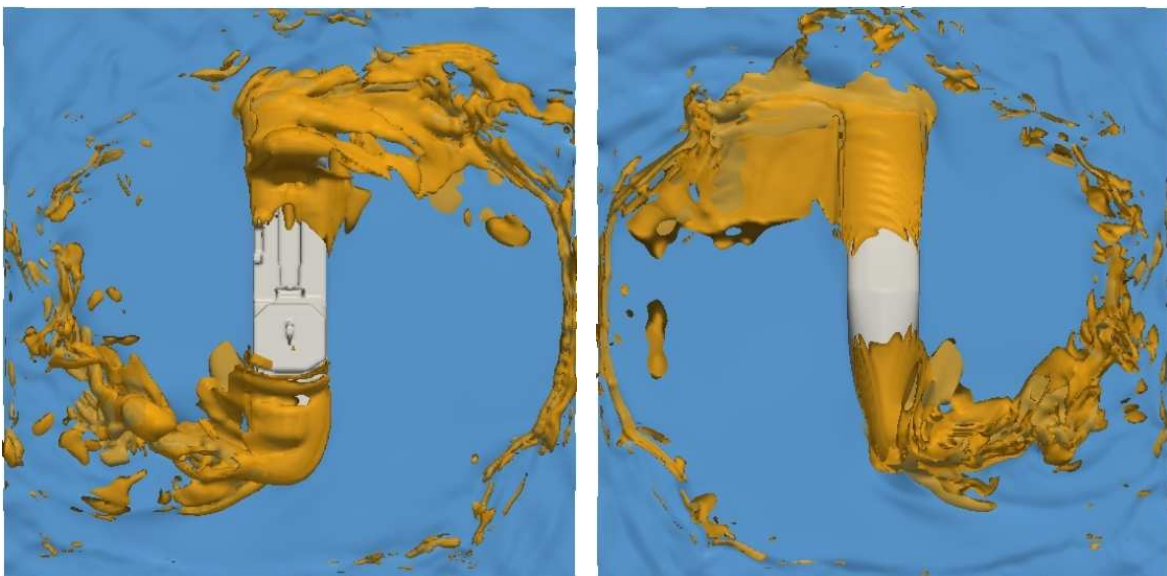


Abbildung 8.19: Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu der Zeit 2.0s. Dargestellt ist die Schiffskörpergeometrie, die Wasseroberfläche und die Isofläche der turbulenten Viskosität (Eddy-Viskosität) zum Niveauwert 1.5_{-6} . Im linken Bild ist derjenige Teil der Isofläche visualisiert, der sich in der Luft befindet. Das rechte Bild dagegen zeigt den im Wasser liegenden Teil der Isofläche.

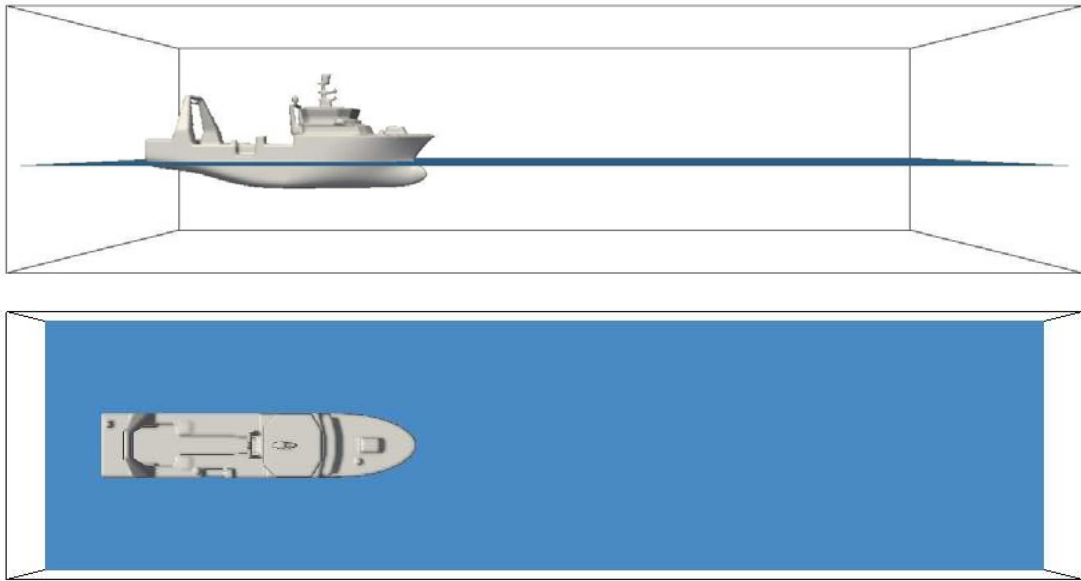


Abbildung 8.20: Rechengebiet und Ausgangsposition der RV-Tangaroa für die Simulation der Schiffs-Translation.

Weiterhin ist zu beobachten, wie die Wasserverwirbelungen sich im zeitlichen Verlauf vom Bug und Heck spiralarig ausbreiten. Bedingt durch das Drehmoment nimmt zur Schiffsmitte hin die Turbulenz deutlich ab. Diese Beobachtung wird in Abbildung 8.18, in der die Wasseroberfläche und die Schiffskörpergeometrie mit der turbulenten Viskosität zur Zeit $t = 2s$ koloriert sind, nochmals bestätigt.

In Abbildung 8.19 ist die Niveaumenge der turbulenten Viskosität zum Niveauewert 1.5_{-6} zur Zeit $t = 2s$ volumetrisch visualisiert. Dabei zeigt das linke Bild den in der Luft befindlichen Teil der Isofläche und das rechte Bild den im Wasser befindlichen Teil. Wie zu erwarten, setzt sich die turbulente Viskosität sowohl in der gesamten Luft- als auch in der gesamten Wasserphase spiralarig fort.

Translation der RV-Tangaroa

In der Realität hat die RV-Tangaroa eine Länge von $70m$ und fährt mit einer Reisegeschwindigkeit von 12 Knoten. Sie verfügt über einen Wulstbug, dessen Funktion darin besteht, die Bildung von Bugwellen zu erschweren. Auf dem Foto in Abbildung 8.12 ist dieser Effekt des Wulstbugs gut zu erkennen.

Um die Realität naturgemäß abzubilden, haben wir in unserer Simulation die RV-Tangaroa auf $70m$ Länge skaliert und diese mit einer Geschwindigkeit von 12 Knoten geradeaus fahren lassen. Für die zwei Phasen verwenden wir Wasser und Luft. Das betrachtete Rechengebiet für die Schiffs-Translation hat die Größe $224m \times 56m \times 56m$ und wird mit einem äquidistantem Gitter von $400 \times 100 \times 100$ Gitterpunkten diskretisiert. Der Schwerpunkt des Schiffes ist zu Beginn der Fahrt in $(174.4m, 28.0m, 24.46m)$ positioniert, und der Wasserpe-

gel ist auf eine Höhe von $22.4m$ gesetzt, sodass der Wulstbug des Schiffes sich vollständig im Wasser befindet, siehe Abbildung 8.20. Für die Glättungsregion $\epsilon(\mathbf{x})$ aus Gleichung (4.4) ist $c = 1.5$ gesetzt worden. Diese Simulation ist bis zur physikalischen Zeit $t = 22s$ parallel auf 64 Prozessoren berechnet worden.

Aufgrund der Möglichkeit, die Resultate der Simulation qualitativ mit dem Foto aus Abbildung 8.12 zu vergleichen, erwarten wir, dass keine substantiellen Bugwellen erzeugt werden. Weiterhin ist der im Wasser befindliche Teil der Schiffsgeometrie deutlich stromlinienförmiger als der in der Luft befindliche Teil. Deshalb sind in der Luft mehr Turbulenzen zu erwarten als im Wasser.

Abbildung 8.21 zeigt die Geradeausfahrt des RV-Tangaroa-Schiffskörpers mit einer Geschwindigkeit von 12 Knoten. Visualisiert ist die Schiffskörpergeometrie, die Wasseroberfläche und die kolorierte turbulente Viskosität auf der Mittelschnittebene. Die Bilder zeigen von oben nach unten die Positionen des Schiffes zu den Zeiten $0s, 1.9s, 3.9s, 9.9s, 15.9s$ und $21.9s$. Zur Zeit $t = 1.9s$ beobachten wir, dass sich die Wasseroberfläche am Bug leicht angehoben hat. Die Turbulenzen in der Luftphase entwickeln über die Zeit einen deutlich ausgeprägteren Nachlauf als die Turbulenzen in der Wasserphase. Das heißt, der Turbulenzgrad in der Luftphase ist erwartungsgemäß höher als das im Geometrieabschnitt unter Wasser der Fall ist. Weiterhin erkennt man, dass sich die Bugwelle zu Beginn der Simulation sich erwartungsgemäß bis zu einem moderaten Grad anhebt und dann kontinuierlich auf dieser Höhe in einer Art Gleichgewichtszustand erhalten bleibt.

Abbildung 8.22 zeigt die Geradeausfahrt der RV-Tangaroa aus der Fischperspektive. Visualisiert ist hierbei die Schiffskörpergeometrie und die freie Wasseroberfläche mit kolorierten Werten der euklidischen Norm des Geschwindigkeitsfeldes. Die Bilder sind von oben nach unten entsprechend den Zeiten $0s, 1.9s, 3.9s, 9.9s, 15.9s$ und $21.9s$ aufgeführt. Zu beobachten ist, wie zu Beginn der Geradeausfahrt im Bild zur Zeit $t = 1.9s$ die Bugwellen sich konvex um den Bug formen. Im weiteren Verlauf wechselt die Ausbreitungsstruktur der Bugwellen von einer anfänglich konvexen hin zu einer konkaven Struktur, die schließlich über den gesamten Simulationszeitraum erhalten bleibt. Zudem ist die Entwicklung des Nachlaufs der RV-Tangaroa zu sehen, der zu Beginn der Fahrt eine symmetrische Strömungsstruktur aufweist und über die Zeit eine zunehmende Asymmetrie ausbildet. Die turbulente Viskosität zur Zeit $t = 21.9s$ ist in Abbildung 8.23 visualisiert. Im obersten Bild die Fischperspektive der Geradeausfahrt des Schiffskörpers zur Zeit $t = 21.9s$ zu sehen. Dort ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit den Werten der turbulenten Viskosität koloriert wurde, dargestellt. In den drei darauffolgenden Bildern ist die Isofläche der turbulenten Viskosität zum Niveaufwert 0.003 aus der Seiten-, Vogel- und Fischperspektive zusehen. Erwartungsgemäß weist die turbulente Viskosität eine asymmetrische Verteilung auf und desweiteren ist klar zu erkennen, dass sich die turbulente Viskosität über die gesamte gefahrene Strecke des Schiffes erstreckt. Dabei nimmt sie, wie die kolorierte Wasseroberfläche deutlich zeigt, in ihrer Intensität kontinuierlich ab je weiter sie vom fahrenden Schiff entfernt ist.

Abbildung 8.24 zeigt eine Studie zur Strömungsstruktur um den Schiffsbug herum zur Zeit $t = 21.9s$. Zu sehen ist im oberen Bild die Schiffskörpergeometrie und die Wasseroberfläche, welche die Wellenstruktur um den Bug herum veranschaulicht. Dabei erkennt man, dass keine substantielle Bugwelle vom Schiff generiert wird, sondern kurz vor dem Bug der

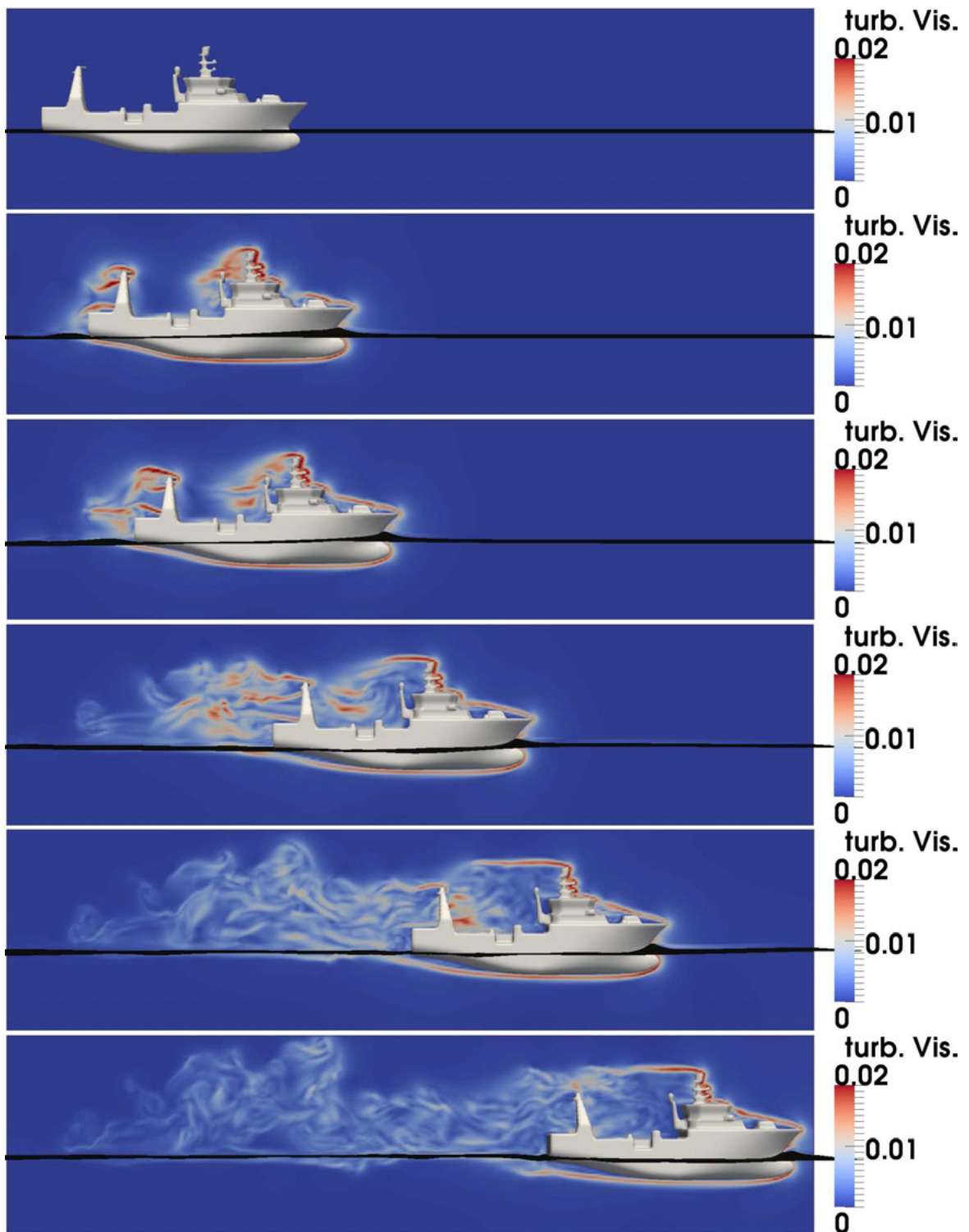


Abbildung 8.21: Geradeausfahrt der RV-Tangaroa-Schiffskörpergeometrie mit 12 Knoten. Zu sehen ist die Schiffskörpergeometrie, die freie Wasseroberfläche und die kolorierte turbulente Viskosität auf der Mittelschnittebene. Die Bilder sind chronologisch von oben nach unten entsprechend den Zeiten 0s, 1.9s, 3.9s, 9.9s, 15.9s und 21.9s aufgeführt.

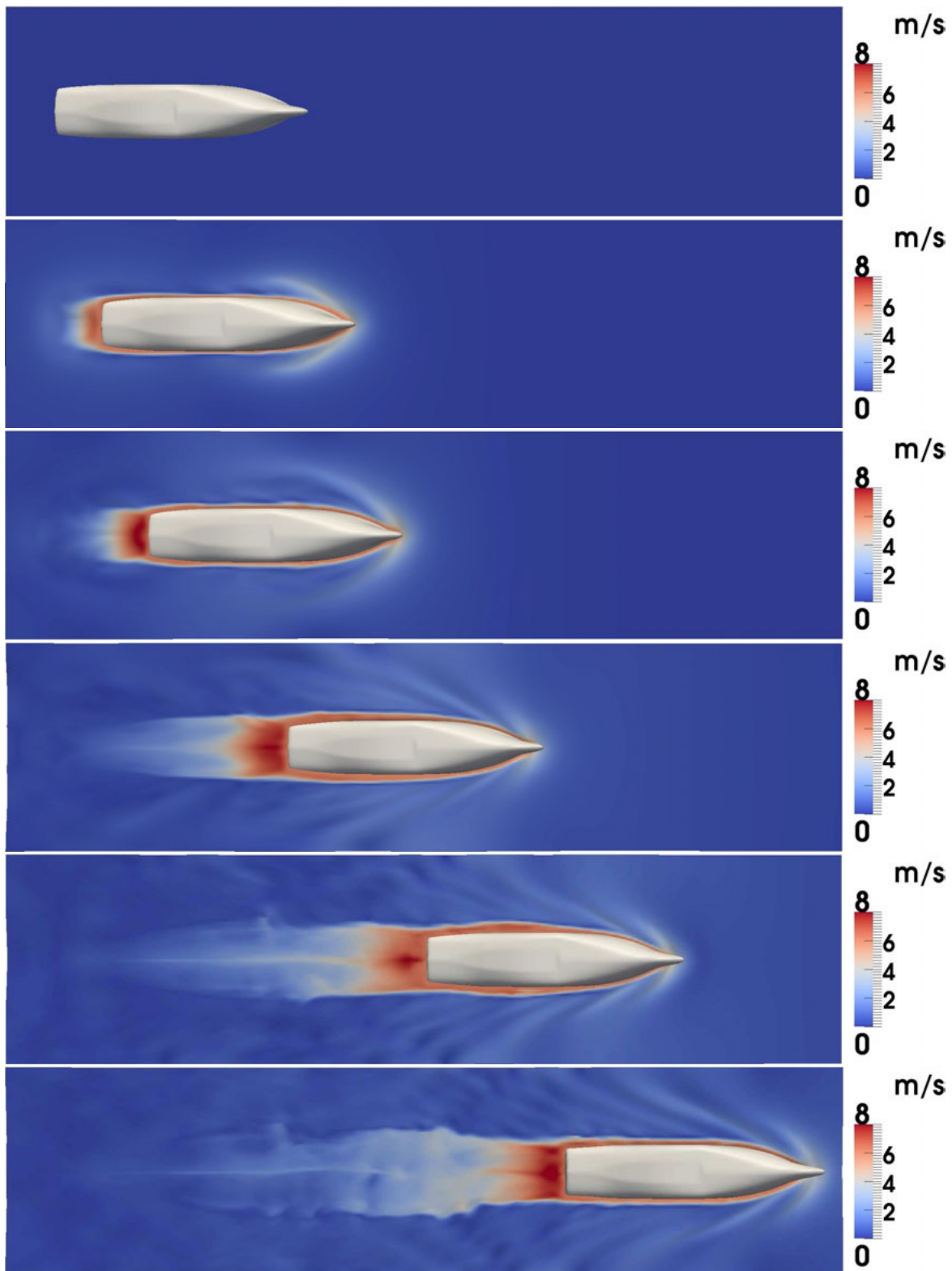


Abbildung 8.22: Fischperspektive der Geradeausfahrt der RV-Tangaroa-Schiffskörpergeometrie mit 12 Knoten. Zu sehen ist die Schiffskörpergeometrie, die freie Wasseroberfläche koloriert mit der euklidischen Norm des Geschwindigkeitsfeldes. Die Bilder sind chronologisch von oben nach unten entsprechend den Zeiten $0s$, $1.9s$, $3.9s$, $9.9s$, $15.9s$ und $21.9s$ aufgeführt.

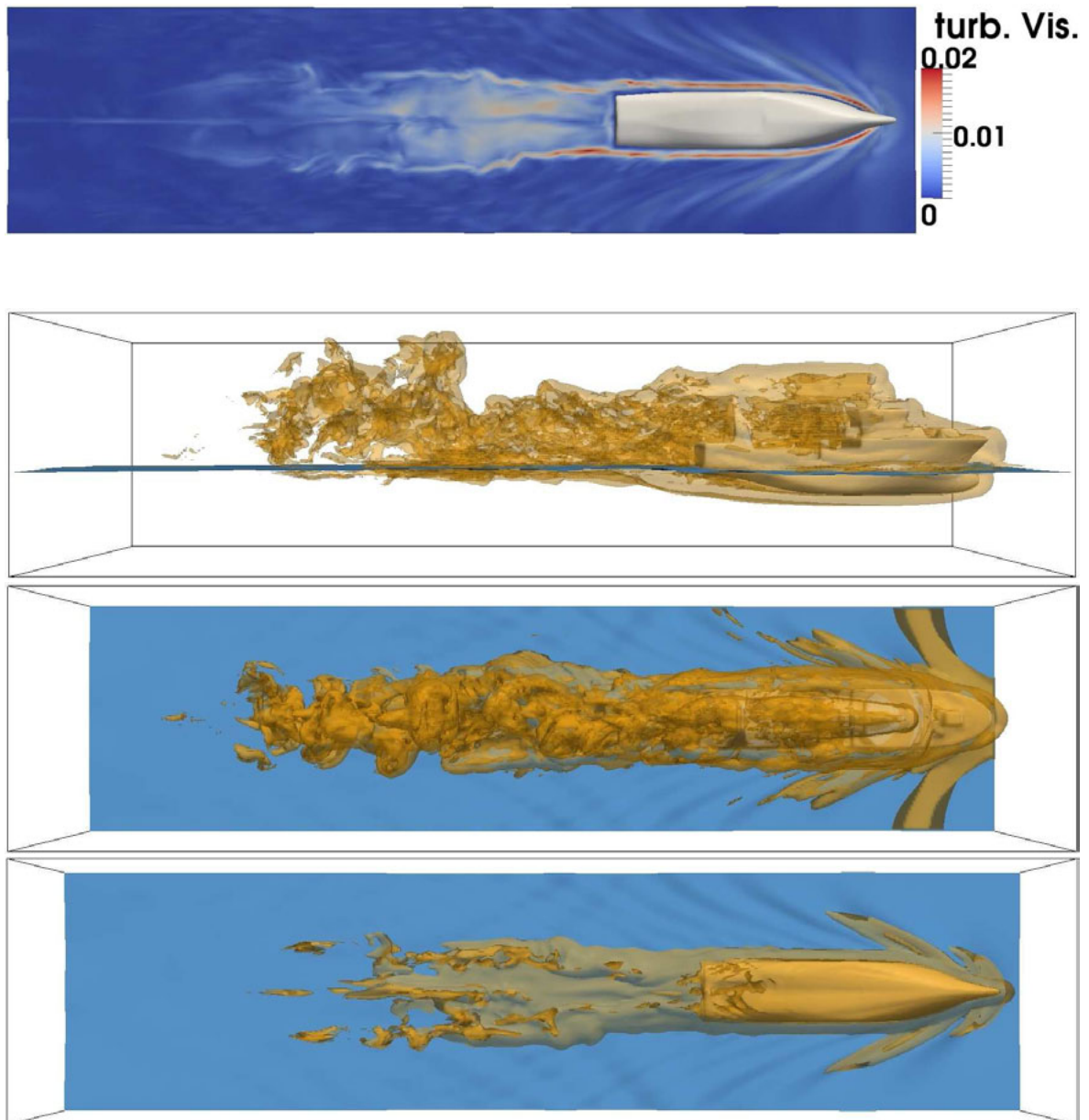


Abbildung 8.23: Geradeausfahrt der RV-Tangaroa-Schiffskörpergeometrie mit 12 Knoten. Zu sehen ist in dem obersten Bild die Fischperspektive auf die Schiffskörpergeometrie und die Wasseroberfläche zur Zeit $t = 21.9s$. Die Wasseroberfläche ist mit den Werten der turbulenten Viskosität koloriert. In den unteren drei Bildern ist zum selben Zeitpunkt die Schiffskörpergeometrie, die freie Wasseroberfläche und die Isofläche der turbulenten Viskosität zum Niveaufwert 3.0_{-3} aus der Seiten-, der Vogel- und der Fischperspektive dargestellt.

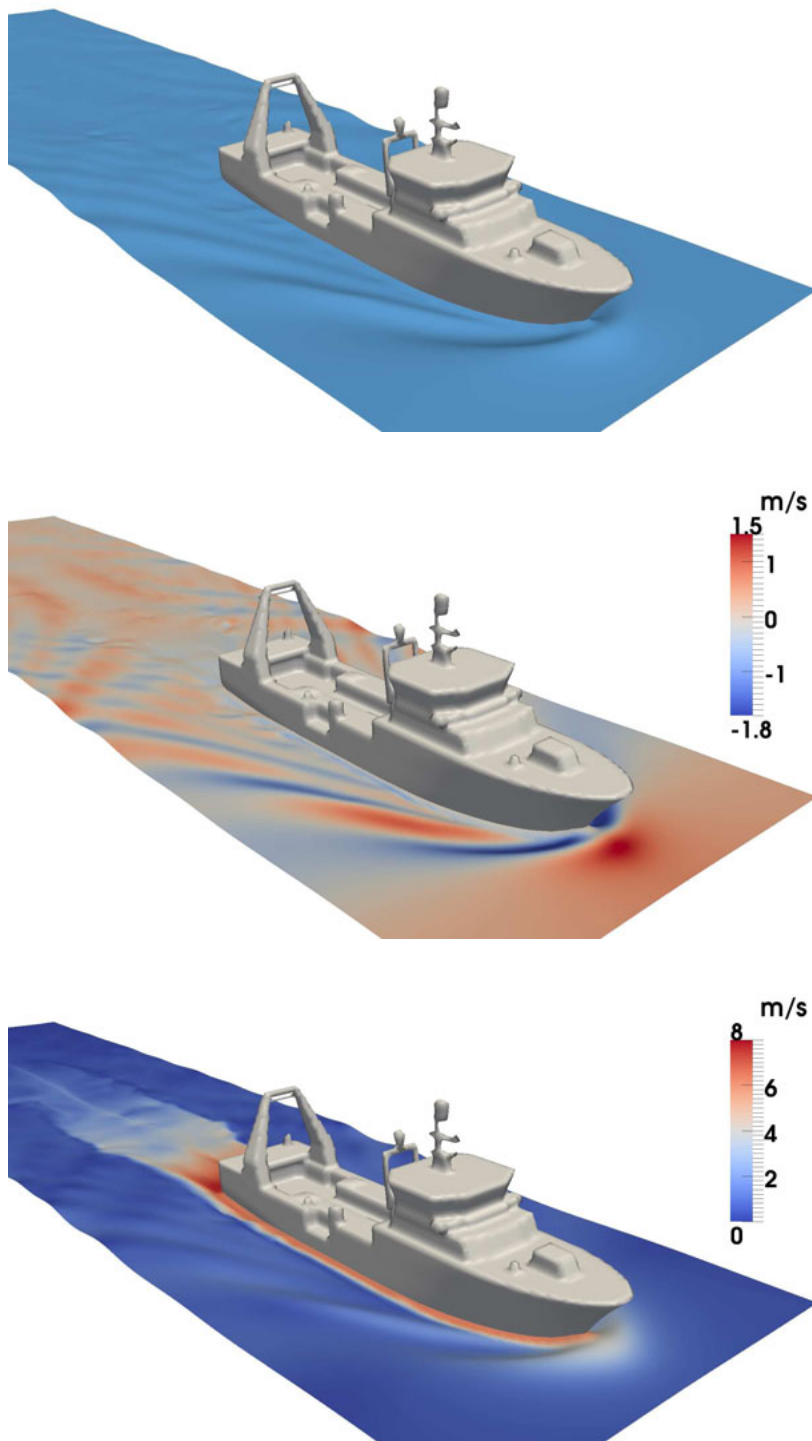


Abbildung 8.24: Geradeausfahrt der RV-Tangaroa-Schiffskörpergeometrie mit 12 Knoten zur Zeit $t = 21.4s$. Zu sehen ist im oberen Bild die Schiffskörpergeometrie und die Wasseroberfläche. Im mittleren Bild ist die Wasseroberfläche mit der Vertikalgeschwindigkeitskomponente koloriert. Im unteren Bild ist sie mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert.

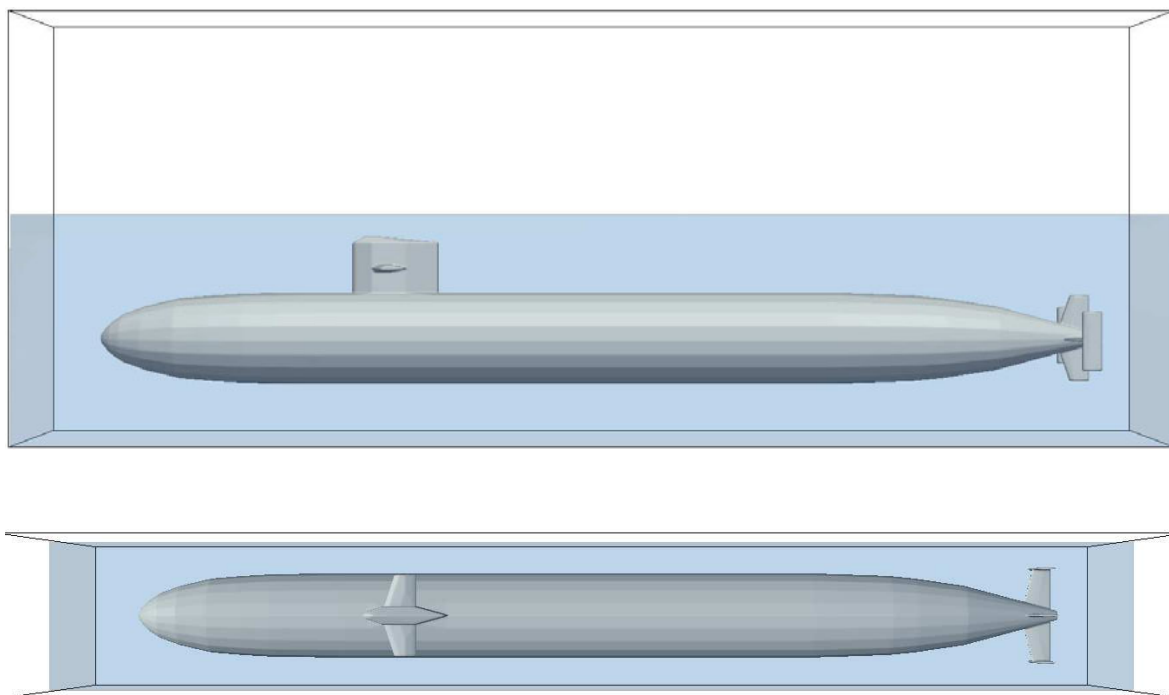


Abbildung 8.25: Initialisierung des aufsteigenden U-Bootes. Zu sehen ist die Geometrie des U-Bootes und das Wasser, in dem es sich zu Beginn befindet. Das obere Bild zeigt die Seitenansicht, das untere Bild die Ansicht von oben.

Wulstbug des Schiffes das Wasser leicht anhebt. Im mittleren Bild ist die Wasseroberfläche koloriert mit der Vertikalgeschwindigkeitskomponente. Hier wird der Effekt des Wulstbuchs deutlich erkennbar, denn dieser sorgt dafür, dass das Wasser vor dem Bug leicht angehoben wird und schließlich aufgrund der Gravitationskraft seitlich am Bug vorbeifließt, anstatt sich wie im Fall eines herkömmlichen Bugs aufzustauen. Im unteren Bild ist die Wasseroberfläche mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert. Dabei stellt man fest, dass die Vertikalkomponente der Geschwindigkeit im Verhältnis zur gesamten Geschwindigkeit nur einen kleinen Beitrag liefert und das Schiff in seiner Geradeausfahrt somit nur gering beeinflusst.

8.3.2 Bidirektionale Koppelung: Auftauchendes U-Boot

In diesem letzten Abschnitt betrachten wir die Geometrie eines U-Bootes [5], welches aus dem Wasser aufsteigt. Das U-Boot besitzt eine Länge von $355.0m$, eine Breite von $36.4m$ und eine Höhe von $52.0m$. Das betrachtete Rechengebiet hat eine Größe von $400m \times 56m \times 150m$ und ist mit einem äquidistanten Gitter der Auflösung $800 \times 112 \times 300$ diskretisiert worden. Weiterhin ist das Gebiet bis zu einer Höhe von $80m$ mit Wasser gefüllt und darüber befindet sich Luft. Die Materialparameter für Wasser und Luft können aus dem Abschnitt 8.1.1 entnommen werden. Die Dichte des U-Bootes setzen wir auf $750kg/m^3$. Zu Beginn

der Simulation befindet sich das U-Boot in waagerechter Lage, vollständig unter Wasser, wobei sein Schwerpunkt in $(28.0, 207.5, 36.30)$ positioniert ist. In Abbildung 8.25 ist die Anfangskonfiguration des aufsteigenden U-Bootes aus der Seiten- und Vogelperspektive dargestellt. Im weiteren zeitlichen Verlauf erwarten wir, dass das U-Boot aufsteigt, bis es die Wasseroberfläche durchstößt und sich dort in einen Gleichgewichtszustand einpendelt.

Dieser Vorgang ist in Abbildung 8.26 dargestellt. Dort ist der zeitlicher Verlauf des aufsteigenden U-Bootes zu den Zeiten $t = 0s, 7s, 9s$ und $11s$ aufgeführt. Dabei ist die U-Boot-Geometrie, die Wasseroberfläche und eine Mittelschnittebene, welche mit der euklidischen Norm der Geschwindigkeitswerte koloriert ist, zu sehen. Wie zu erwarten, hebt das U-Boot zu Beginn der Aufstiegsphase das über ihm befindliche Wasser leicht an. Der U-Boot-Turm durchstößt hierbei als erstes die Wasseroberfläche, wobei die seitlich vom Turm rausragenden Tiefenruder im weiteren Verlauf das Wasser in diesem Bereich stark verwirbeln. Weiterhin ist zu beobachten, wie sich mit zunehmender Aufstiegshöhe die Wasseroberfläche vollständig um die U-Boot-Geometrie legt und dieses mit einer dünnen Wasserschicht bedeckt. In Bereichen wo die Geometrie des U-Bootes eine starke Krümmung aufweist, wie beispielsweise am Turm, dem Tiefenruder und dem Steuerkreuz, löst sich die dünne Wasserschicht erwartungsgemäß als erstes auf. Deweiteren treten im Bereich des Turms und des Steuerkreuzes aufgrund der starken Turbulenzen, die von diesen Geometriebereichen ausgehen, lokale Geschwindigkeiten von bis zu $12m/s$ auf. Die anfänglich horizontale Lage des U-Bootes geht während des Aufstiegs in eine leichte Neigung über, die darauf hindeutet, dass der Bug etwas schneller als das Heck aufsteigt. Der Grund hierfür ist, dass die vordere Hälfte des U-Bootes mit dem Turm insgesamt mehr Volumen einschließt als die hintere Hälfte. Dies wiederum induziert eine ungleich verteilte Auftriebskraft, die den vorderen Teil etwas schneller aufsteigen lässt als den hinteren Teil.

Abbildung 8.27 zeigt einen Vergleich der Wasseroberflächenstruktur in der das U-Boot einerseits seinen höchsten Aufstiegsunkt erreicht (linkes Bild) und andererseits zu sinken beginnt (rechtes Bild), um in den Gleichgewichtszustand auf der Wasseroberfläche zu konvergieren. Wie erwartet, bildet die Wasseroberfläche zum Zeitpunkt des höchsten Aufstiegs eine Rinnenstruktur um das U-Boot herum, die durch das von der U-Boot-Oberfläche ablaufende Wasser und das unterhalb des U-Bootes vorhandene Unterdruckgebiet erzeugt wird. Wenn auch deutlich makroskopischer und bidirektional gekoppelt, so ist dieser Prozess doch sehr artverwandt mit dem Zylinderexperiment aus Abschnitt 8.1.1 dieses Kapitels. Im rechten Bild beginnt das U-Boot zu sinken und induziert somit einen Überdruck auf das Wasser, sodass erwartungsgemäß eine Wasserwelle von der U-Boot-Geometrie aus erzeugt wird und sich radial von ihr ausbreitet. Man beachte, dass dieser Prozess auch auf dem sehr kleinskaligen Steuerruder im Heckbereich des U-Bootes deutlich zu erkennen ist.

In Abbildung 8.28 ist sowohl der zeitliche Verlauf der Höhe des U-Boot-Schwerpunktes als auch die Aufstiegsgeschwindigkeit graphisch bis zur Zeit $t = 11.2s$ dargestellt. Die Höhe des U-Boot-Schwerpunktes nimmt zu Beginn der Simulation langsam zu und wächst bis zur Zeit $t = 7.8s$ immer schneller an. Dann beginnen die wirkenden Auftriebskräfte im Verhältnis zu der wirkenden Gravitationskraft nachzulassen, sodass die Aufstiegsgeschwindigkeit und damit der Höhenzuwachs der U-Boot-Geometrie substantiell abnimmt. Zur Zeit $t = 11s$ erreicht der U-Boot-Schwerpunkt seine maximale Höhe, wobei dessen Aufstiegsgeschwin-

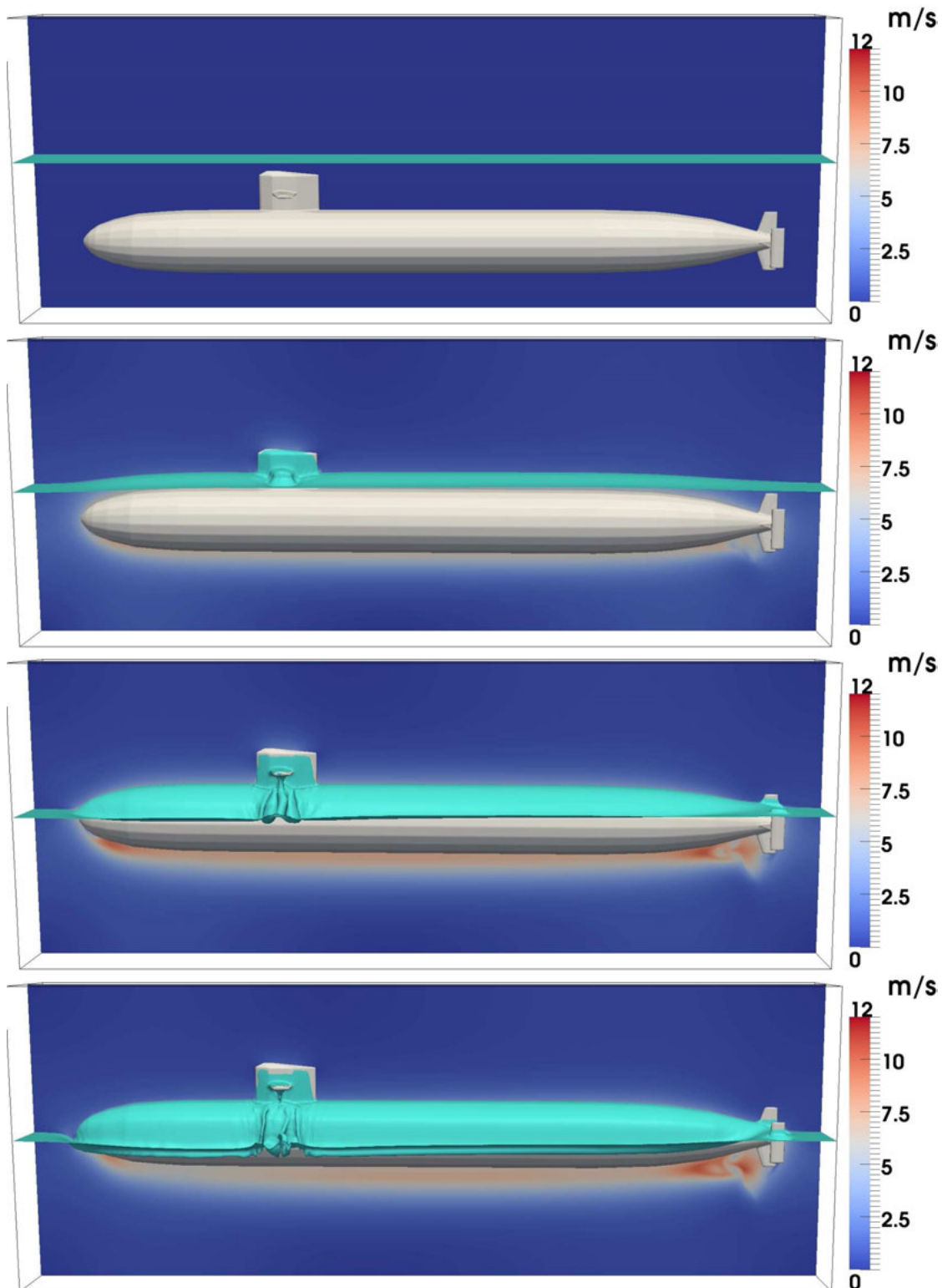


Abbildung 8.26: Aufsteigendes U-Boot zu den Zeiten $t=0s$, $7s$, $9s$, $11s$ (von oben nach unten). Zu sehen ist die U-Boot-Geometrie und die Wasseroberfläche und eine Mittelschnittebene, welche mit der euklidischen Norm der Geschwindigkeitswerte koloriert ist.

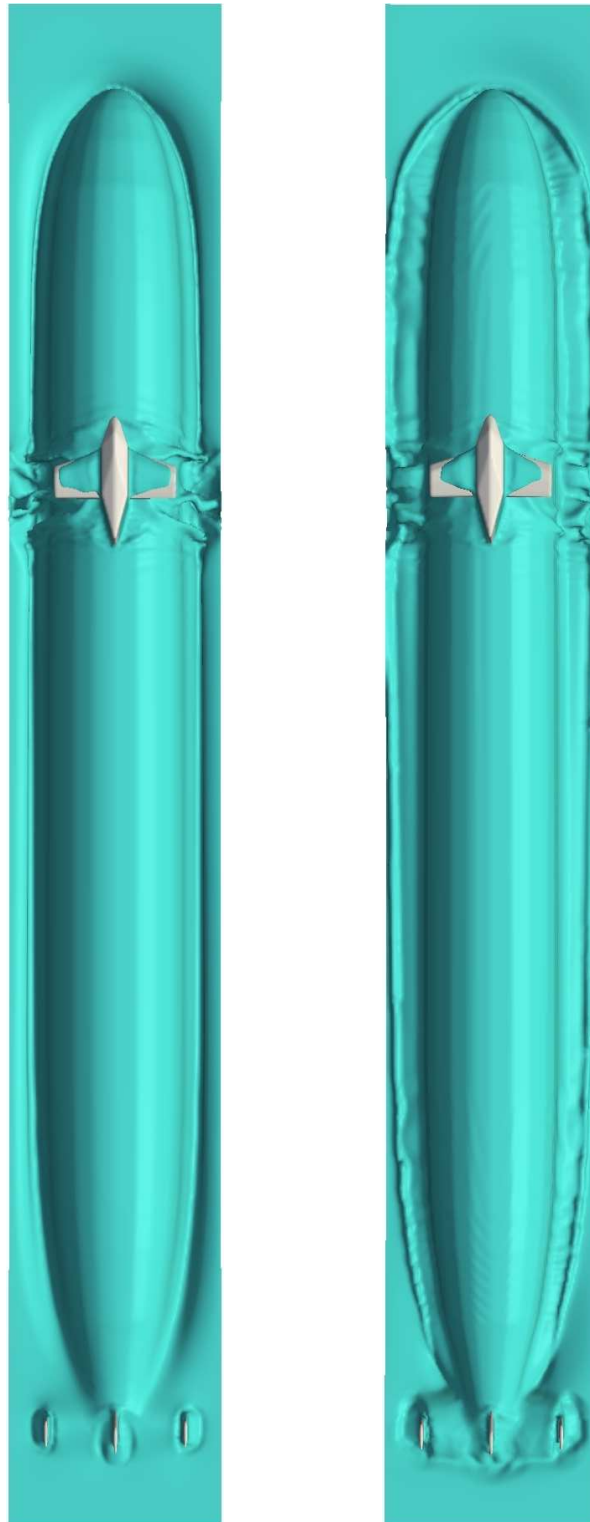


Abbildung 8.27: Vogelperspektive des aufsteigendes U-Bootes. Das linke Bild zeigt wie das U-Boot gerade seinen höchsten Aufstiegsunkt erreicht. Das rechte Bild zeigt wie das U-Boot wieder absinkt, um in den Gleichgewichtszustand auf der Wasseroberfläche zu konvergieren.

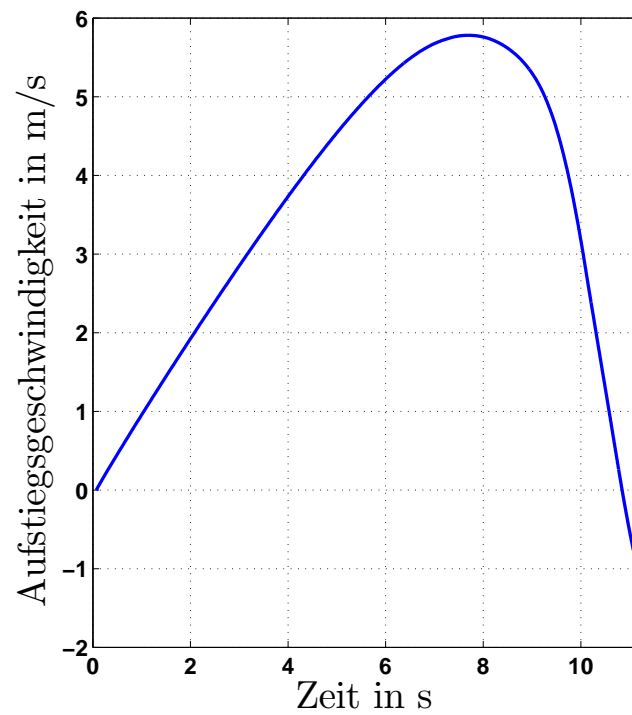
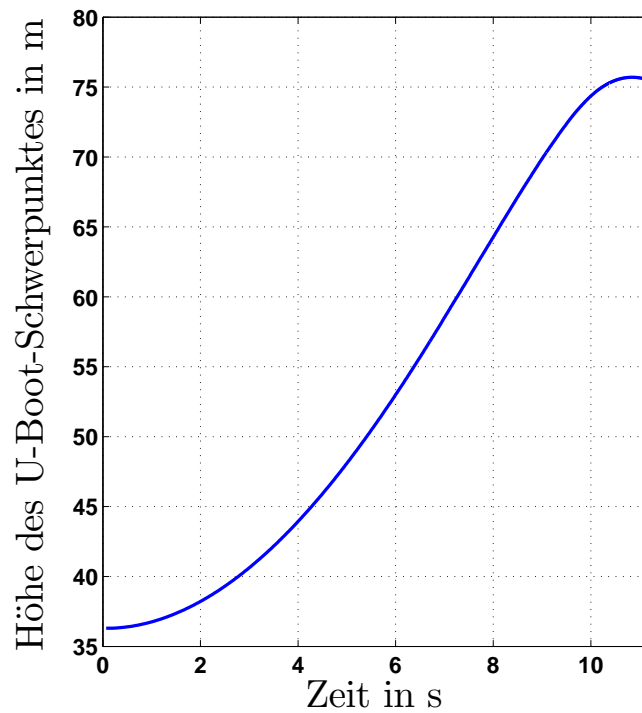


Abbildung 8.28: Aufsteigendes U-Boot: zeitlicher Verlauf der Höhe des U-Boot-Schwerpunktes (oberer Graph) und die Aufstiegsgeschwindigkeit (unterer Graph).

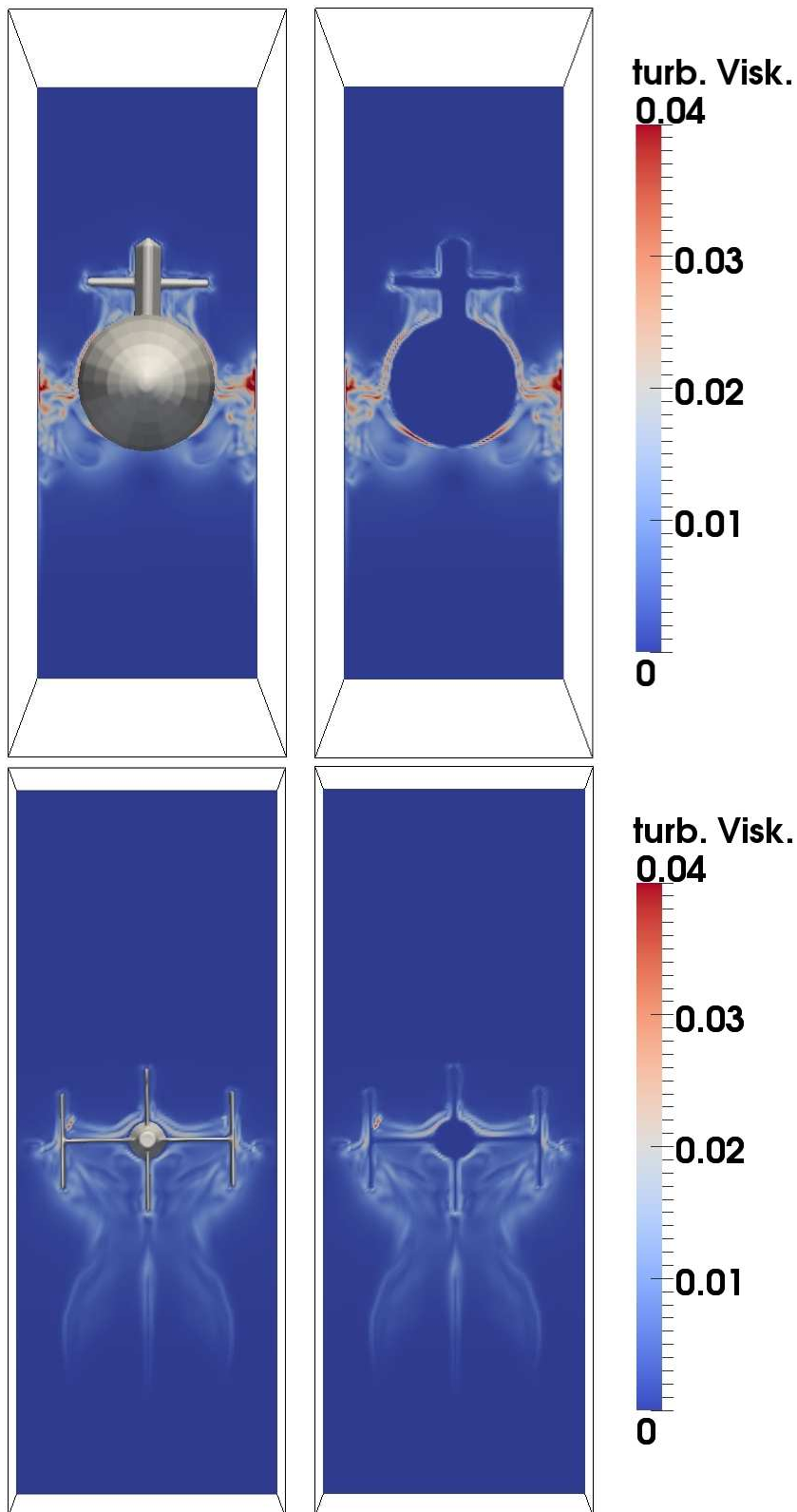


Abbildung 8.29: Aufsteigendes U-Boot zur Zeit 11.2s. Linke Bildspalte zeigt die Bug- und Heckansicht der U-Boot-Geometrie inklusive einer Schnittebene, die mit der turbulenten Viskosität (Eddy-Viskosität) koloriert ist. In der rechten Bildspalte ist dieselbe Schnittebene dargestellt, nur ohne die U-Boot-Geometrie.

digkeit erwartungsgemäß den Wert $0 \frac{m}{s}$ durchläuft und damit der Sinkprozess des U-Bootes beginnt. Darüber hinaus ist zu erkennen, dass die Graphen der Aufstiegshöhe und der Aufstiegs geschwindigkeit konsistent verlaufen, da die Aufstiegs geschwindigkeit der Ableitung der Aufstiegshöhe entspricht.

Abschließend ist in Abbildung 8.29 die Verteilung der turbulenten Viskosität (Eddy-Viskosität) auf zwei Schnittebenen zur Zeit $t = 11.2s$ dargestellt, einmal durch den Turm und Rumpf des U-Bootes in der oberen Bildreihe und einmal durch das Steuerkreuz im Heckbereich des U-Bootes in der unteren Bildreihe. Wie auf der Schnittebene durch den Turm und Rumpf des U-Bootes zu sehen, wird eine starke turbulente Viskosität generiert, da das Wasser zu der Zeit sowohl vom Turm als auch vom Rumpf des U-Bootes abfließt und zusätzlich das U-Boot mit seinem Sinkprozess starke Wellen erzeugt. Dahingegen fällt die turbulente Viskosität auf der Schnittebene durch das Steuerkreuz des U-Bootes in weiten Teilen deutlich geringer aus. Aber aufgrund der Kreuzform des Hecks wirkt dieser Geometrieabschnitt über einen deutlich längere Wegstrecke auf die Turbulenzstruktur der Strömung als das in der oberen Bildreihe der Fall ist. Die turbulente Viskosität hat keine so starke Intensität, ist aber über einen großflächigeren Bereich präsent. Schließlich beobachten wir in der rechten Bildspalte, dass innerhalb der U-Boot-Geometrie erwartungsgemäß keine turbulente Viskosität generiert wird, selbst für das feinskalige Steuerkreuz nicht.

Kapitel 9

Zusammenfassung und Ausblick

9.1 Zusammenfassung

Für diese Arbeit ist ein paralleles, numerisches Simulationsprogramm erstellt worden, das dreidimensionale, inkompressible Zweiphasenströmungen mit Oberflächenspannung und beweglichen Starrkörpern in drei Raumdimensionen stabil berechnen kann. Dabei wurden reale Zweiphasen-Starrkörpersysteme betrachtet, die an der Phasengrenze im Allgemeinen einen Dichte- und Viskositätssprung aufweisen.

Die Strömungsgleichungen werden auf einem Eulerschen Gitter mittels der Chorin-Temam-Projektionsmethode gelöst. Die konvektiven Terme der Navier-Stokes-Gleichungen, der Transportgleichung und der Level-Set-Reinitialisierung werden über das WENO-Verfahren fünfter Ordnung diskretisiert. Für die Einbindung von uni- und bidirektional gekoppelten komplexen Starrkörpern in Zweiphasenströmungen ist die Chorin-Temam-Projektionsmethode in drei Raumdimensionen um eine Kräfte- und eine Starrkörperdruck-Berechnungsroutine erweitert und parallelisiert worden. Hierbei wird neben dem klassischen Druck-Poisson-Löser, welcher als Lagrange-Multiplikator die Inkompressibilität der Zweiphasenströmung sichert, ein zusätzlicher Druck innerhalb des Starrkörpers ermittelt, der die Deformationsfreiheit des Starrkörpers wie auch dessen physikalische Bewegung in Form eines zusätzlichen Lagrange-Multiplikators erzwingt.

Die freie Oberfläche der Zweiphasenströmung und des beweglichen Starrkörpers wird mit der Nivaumenge zum Niveuwert Null einer skalaren vorzeichenbehafteten Abstandsfunktion, der Level-Set-Funktion, identifiziert. Die diskrete Level-Set-Funktion zu einer gegebenen triangulierten Starrkörpergeometrie berechnet sich hierbei in zwei Schritten. Zuerst wird in jedem Gitterpunkt das Vorzeichen über ein Polygonscan-Verfahren ermittelt und anschließend eine effiziente Abstandsberechnung zu allen Dreiecken der triangulierten Starrkörpergeometrie durchgeführt, die in ihrer entsprechenden "bounding-box" den jeweiligen Gitterpunkt enthalten. Das Minimum der Abstände liefert schließlich in jedem Gitterpunkt den gewünschten vorzeichenbehafteten Abstand.

Level-Set-Funktionen haben den Vorteil, dass sich aus ihnen geometrische Größen, wie zum Beispiel die Krümmung des freien Randes, effizient berechnen lassen. Weiterhin benötigen Level-Set-Methoden im Verlauf der Rechnung keine explizite Rekonstruktion des freien Randes. Sowohl der Dichte- und Viskositätssprung als auch die Oberflächenspannung

können in Abhängigkeit von der Level-Set-Funktion auf natürliche Weise in die Navier-Stokes-Impulsgleichungen eingebaut werden. Die Materialsprünge gehen über eine geglättete Heaviside-Funktion und die Oberflächenspannung geht als lokaler Quellterm in Form eines geglätteten Deltafunktionals in die Impulsgleichungen der Navier-Stokes-Gleichungen ein. Die Größe der Glättungsumgebung ist von der Ordnung $O(h) \approx \frac{3}{2}h$ und sie bleibt während der gesamten Simulationsberechnung, mittels einer Reinitialisierungstechnik für die Level-Set-Funktion, unverändert. Die Reinitialisierung erfolgt durch iteratives Lösen einer zusätzlichen Differentialgleichung, die dafür sorgt, dass die Level-Set-Funktion über alle Zeiten hinweg eine vorzeichenbehaftete Abstandsfunktion bleibt. Untersuchungen ergaben, dass durch die Reinitialisierung Probleme in der Masseerhaltung auftreten. Substantielle Verbesserungen wurden sowohl durch den Einsatz eines WENO-Verfahrens fünfter Ordnung als auch durch eine geschicktere Approximation der in der Reinitialisierungsfunktion enthaltenen signum-Funktion erreicht. Um darüber hinaus die Masseerhaltung für Langzeitsimulationen, welche zur Berechnung stationärer Zustände notwendig sind, zu sichern, ist nach der Reinitialisierung eine Picard-Fixpunktiteration für die Level-Set-Funktion hinzugefügt worden, die die Masseerhaltung über alle Zeiten hinweg garantiert.

Die zeitliche Entwicklung der freien Oberfläche der Zweiphasenströmung erfolgt über den Eulerschen Transport dieser Funktion mit den Strömungsgeschwindigkeiten. Die freie Oberfläche des Starrkörpers wird aufgrund der Einschränkung der Starrkörperbewegung auf Translation und Rotation über eine semi-Lagrangesche Transportmethode durchgeführt. Diese hat in unserer Implementation den substantiellen Vorteil, dass eine zeitliche Akkumulation von numerischen Diffusionsartefakten, welche die Verletzung der Deformationsfreiheit des Starrkörpers zur Folge hätte, nicht auftreten.

Die Parallelisierung des gesamten Verfahrens erfolgte mit der Programmbibliothek *Message Passing Interface* (MPI) basierend auf der Gebietszerlegungsmethode für sämtliche Strömungsgrößen und der "Replicated-Data"-Technik für den Starrkörper. Um den Kommunikationsaufwand minimal zu halten, wurden unterschiedlich viele Randbordüren für die Teilgebiete benötigt, die den verschiedenen großen Diskretisierungsskizzen angepasst sind.

Numerische Untersuchungen von Teilen des gesamten Zweiphasenlösers, wie der Transportgleichung und der Reinitialisierung der Level-Set-Funktion, zeigen deutlich, dass der Einsatz eines WENO-Verfahrens fünfter Ordnung für die Diskretisierung der konvektiven Terme, trotz stark verformter freier Oberflächen, zu einer sehr guten Masseerhaltung führt. Eine Analyse der numerischen Konvergenzordnung des vollständig gekoppelten dreidimensionalen Zweiphasen-Lösers am Beispiel einer aufsteigenden Fluidblase inklusive Oberflächenspannung hat ergeben, dass dieser mit einer Ordnung nahe zwei konvergiert. Ein direkter Vergleich mit zwei aktuellen Publikationen hat ergeben, dass unser Verfahren beiden Methoden überlegen ist, da unser Verfahren beiden Methoden in Bezug auf Masseerhaltung und Konvergenzordnung überlegen ist. Abschließend haben wir den Zweiphasenlöser im Rahmen einer Parameterstudie am Beispiel eines Tropfens in einer Scherströmung untersucht. Dabei konnten wir alle bezüglich eines Phasendiagramm zu erwartenden Topologieveränderungen in den Simulationen wiedergeben.

Numerische Untersuchungen des vollständig gekoppelten dreidimensionalen Zweiphasen-Starrkörper-Lösers zeigen, dass die Ergebnisse des hier entwickelten Simulationsprogramms die Messdaten und Momentaufnahmen eines realen Laborexperimentes von Green-

how & Lin sehr gut wiedergeben. Physikalische Phänomene, wie der von Pelegrin bezeichnete "breaking-waterfall"-Effekt konnten erfolgreich simuliert werden. Eine Analyse der numerischen Konvergenzordnung am Beispiel eines aufgrund wirkender Auftriebskräfte aufsteigenden Zylinders hat ergeben, dass dieses von der Ordnung eins konvergiert. Diese Approximationsgüte wird auch im Fall einer auf der freien Oberfläche im hydrodynamischen Gleichgewicht schwimmenden Kugel angenommen. Der Grund für die im Vergleich zum Zweiphasenlöser etwas geringere Approximationsgüte besteht darin, dass sowohl die Integralauswertung der auf den Starrkörper wirkenden Kräfte als auch die Dichteglättung mit einer Genauigkeit erster Ordnung durchgeführt wurde. Abschließend zeigen zwei um ein paralleles Large-Eddy-Turbulenzmodell erweiterte Simulationsrechnungen, die eine mit der Geometrie eines Schiffkörpers und die andere mit der Geometrie eines U-Bootes, dass das in dieser Arbeit entwickelte Simulationsprogramm ingenieurstechnische Probleme realitätsnah simulieren kann.

All diese ermutigenden Ergebnisse geben Anlass zu einer Reihe von möglichen Weiterentwicklungen, von denen nur einige im Folgenden aufgeführt werden.

9.2 Ausblick

Eine vollständig adaptive Behandlung des parallelen Zweiphasen-Starrkörper-Lösers ist für Simulationen, in denen die freie Oberfläche der Zweiphasen oder des Starrkörpers große Skalenunterschiede aufweist, von großem Vorteil. Hierbei ist neben den zu entwickelnden Prolongations- und Restriktionsoperatoren für die jeweiligen Gitterlevel insbesondere auch eine dynamische adaptive Lastbalancierung essentiell, beispielsweise über raumfüllende Kurven.

Von großem Interesse ist auch die Erweiterung dieses Simulationsprogramms um einen Formoptimierungsalgorithmus, sodass zum Beispiel der Schiffsrumpf von Wasserfahrzeugen hinsichtlich Wasserwiderstand und Kraftstoffverbrauch optimiert werden kann.

Eine weitere interessante Aufgabe bestünde darin, das Shikhmurzaev-Modell [171, 172] für die Simulation von Vorhangbeschichtungen mit diesem Simulationsprogramm zu koppeln und damit insbesondere Fluid-Starrkörper-Untersuchungen vorzunehmen. Denn der auf das Substrat wirkende Fluidimpuls ist im Modell von Shikhmurzaev nicht berücksichtigt und die in Form von Vibrationen resultierende Wechselwirkung mit dem Fluidvorhang kann den gesamten Beschichtungsprozess hinsichtlich homogener Beschichtungsdicke, Stabilität des Fluidvorhangs etc. substantiell beeinflussen.

Abbildungsverzeichnis

1.1	Tropfen in einer Scherströmung. Erste zwei Spalten: Stationäre Endzustände einer Tropfendeformation; Rechts: Instationäre Deformation eines Tropfens bei dem sich Satellitentropfen ablösen. Fotos stammen aus dem Artikel [177].	2
1.2	Simulation eines auf ein Substrat fallenden Fluidtropfens (durchgeführt im Rahmen des Sonderforschungsbereiches 611, Projekt C3). Der zeitliche Verlauf des Tropfenaufpralls ist von links nach rechts dargestellt.	3
1.3	Links ist das fahrende Forschungsschiff RV-Tangaroa [1] abgebildet und rechts ein fahrendes U-Boot [2].	4
2.1	Zeitliche Entwicklung von Ω_0 nach Ω_t .	12
2.2	Zur Zweiphasenmodellierung.	15
2.3	Zur Beschreibung der Oberflächenspannungskraft.	16
3.1	Scherströmung mit Geschwindigkeit u in der oberen Fluidhälfte (weiss) und $-u$ in der unteren Fluidhälfte (blau): (a): Anfangsbedingung der Zweiphasen-Scherströmung. (b): Gitterverzerrung durch die Fluidgeschwindigkeit u und $-u$. Neue Verknüpfung der Gitterpunkte ergibt wieder (a).	25
3.2	Einsatz von Lagrange-Marker-Partikel zur Oberflächendarstellung für die Front-Tracking-Methode.	26
3.3	Volumenbeschreibung mit der <i>particle-in-cell (PIC)</i> Technik. Im Falle der <i>Smoothed-Particle-Hydrodynamics (SPH)</i> Methode würde das Eulersche Gitter im Hintergrund wegfallen, da SPH ein gitterloses Verfahren ist.	28
3.4	Exakte Oberfläche, SLIC und PLIC Rekonstruktion.	31
3.5	Adaptive Verfeinerung der Grenzfläche in Phasenfeldmethoden.	33
3.6	Oberflächenbewegung in Normalenrichtung.	35
3.7	Level-Set-Darstellung: Getrennte (links) und vereinigte Oberflächen.	36
3.8	Level-Set-Funktion, einmal als 2D-Plot mit entsprechenden Niveaumengen, einmal als Graph der vorzeichenbehafteten Abstandsfunktion und einmal als Graph der verzerrten Funktion.	37
4.1	Positionierung der Geschwindigkeiten, des Druckes und der Level-Set Werte innerhalb einer Gitterzelle mit Index (i, j, k) .	41
4.2	Zentrale (oben) und linksseitige (unten) eindimensionale Lagrange-Interpolation.	42
4.3	Zweidimensionale Lagrange-Interpolation.	42

4.4	Beispiel einer numerischen Grenzflächenbreite $\epsilon(\mathbf{x})$ für $c = 2$. Die fettgedruckte mittlere Linie beschreibt den physikalisch relevanten freien Rand Γ_f .	44
4.5	Zur nicht äquidistanten Zeitdiskretisierung.	47
4.6	Volumenkorrektur der Level-Set Funktion.	52
4.7	Datenaustausch am Beispiel einer „künstlichen Randpunktreihe“.	63
5.1	Advektion einer Sphäre durch ein vorgegebenes divergenzfreies Rotationsfeld. Das oberste Bild zeigt die Anfangskonfiguration, die unteren vier Bilder zeigen die chronologische Abfolge des Transportes der Sphäre zu den Zeiten $t=5, t=10, t=15, t=20$	68
5.2	Nullmengenisofläche und Mittelschnittebene mit dazugehörigen Konturlinien der Level-Set Funktion vor der Reinitialisierung (links) und danach (rechts).	70
5.3	Das linke Bild zeigt die übereinandergelegten Isoflächen beider Funktion zum Niveauwert Null. Die graue Isofläche gehört zur Funktion ϕ und die schwarzen Flächenbereiche zu der generierten Abstandsfunktion $\tilde{\phi}$, die sich über die ursprüngliche legt. Das rechte Bild ist die Vergrößerung des Schnittes der Isoflächen mit dem kleinen roten Quadrat. Hierbei ist die durchgezogene Linie zur ursprünglichen Funktion ϕ und die gestrichelte Linie zur generierten Abstandsfunktion $\tilde{\phi}$	70
5.4	Fehlerkonvergenz der Krümmungsauswertung einer transportierten Einheits-sphäre in der L^1 - und L^∞ -Norm nach einer Periode. $R-$ und $R+$ bezeichnen die ein- bzw. ausgeschaltete Reinitialisierung der Level-Set-Funktion, analoges gilt für $F-$ und $F+$ bezüglich der Picard-Fixpunktiteration.	71
5.5	Konvergenz des relativen Fehlers der Masse. $R-$ und $R+$ bezeichnen die ein- bzw. ausgeschaltete Reinitialisierung der Level-Set-Funktion, analoges gilt für $F-$ und $F+$ bezüglich der Picard-Fixpunktiteration.	72
5.6	Aufsteigende Blase ohne (obere Zeile) und mit (untere Zeile) Fixpunktiteration für verbesserte Masseerhaltung zur Zeit $t = 0.1s$ (linke Spalte) und $t = 0.15s$ (rechte Spalte) für $h = 1/37$. Dargestellt ist die semi-transparente Isofläche der Level-Set-Funktion und ein Konturplot der v-Geschwindigkeitskomponente entlang der zentralen Schnittebene.	75
5.7	Im Wasser aufsteigende Luftblase zu den Zeitpunkten $t = 0.01s, 0.02s, 0.035s$, und $0.05s$ (von links nach rechts). In der oberen Reihe sind die Oberflächenpositionen für die Gitterweiten $= (1/160, 1.5/240, 1/160)$ abgebildet. In der unteren Reihe sind die Schnitte der freien Oberfläche mit der Mittelebene des Rechengebietes für die Gitterweiten $= (1/40, 1.5/60, 1/40)$, $(1/80, 1.5/120, 1/80)$, und $(1/160, 1.5/240, 1/160)$ abgebildet (die fettgedruckte Linie entspricht der feinsten Auflösung).	78
5.8	Aufsteigende quasi 2D Luftblase im Wasser zu den Zeitpunkten $t = 0.01s, 0.02s, 0.035s, 0.05s, 0.075s$ und $0.1s$. Zu sehen sind die Schnitte zwischen der Oberfläche und der Mittelebene des Rechengebietes für die Gittermaschenweiten $= (1/40, 1.5/60, 1/4)$, $(1/80, 1.5/120, 1/8)$, und $(1/160, 1.5/240, 1/16)$ (fettgedruckte Linien entsprechen der feinsten Auflösung).	79

5.9 Phasendiagramm der Tropfendeformation bezüglich der Kapillarzahl Ca und der Reynoldszahl Re . Das stationäre Regime mit konstanter Topologie wird mit ST bezeichnet, das instationäre Regime, bei dem topologische Änderungen auftreten wird mit IT bezeichnet. 80

5.10 Stationäre Ergebnisse der Tropfendeformation in einer Scherströmung für die vier Parameterkonfigurationen $(Ca, Re) = (0.25, 1.5), (0.2, 4), (0.15, 10)$, und $(0.1, 20)$ (oben links bis unten rechts). Zu sehen ist die freie Oberfläche im stationären Limit. Das entsprechende Druckfeld ist mit der gleichen Farbskalierung auf allen Bildern mittels einer Mittelschnittebene dargestellt. Die stationären Zustände werden zu den Zeiten $t = 203.34, t = 538.41, t = 829.68$, und $t = 1649.36$ erreicht (von oben links bis unten rechts). Das Simulationsgebiet ist $2 \times 1 \times 2$ groß und der Tropfen hat einen Radius von $r = 0.25$. Die Gittermaschenweite ist $(2/128, 1/64, 2/128)$. . . 81

5.11 Stationäre Ergebnisse der Tropfendeformation in einer Scherströmung für vier unterschiedliche Parameterkonfigurationen. Zu sehen sind Vergrößerungen der freien Oberfläche in dem umgebenden Strömungsfeld entlang der Mittelschnittebene. Die entsprechenden Druckfelder sind mit der gleichen Farbskalierung auf allen Bildern mittels einer Mittelschnittebene dargestellt. Die stationären Zustände werden zu den Zeiten $t = 203.34, t = 538.41$ (obere Reihe) und $t = 829.68, t = 1649.36$ (untere Reihe) erreicht. Das Simulationsgebiet ist $2 \times 1 \times 2$ groß und der Tropfen hat einen Radius von $r = 0.25$. Die Gittermaschenweite ist $(2/128, 1/64, 2/128)$. Für die Parameter $(Ca, Re) = (0.25, 1.5), (0.2, 4)$ (obere Reihe) bildet das Strömungsfeld innerhalb des Tropfens einen einzigen Wirbel aus, wobei für die Parameter $(Ca, Re) = (0.15, 10), (0.1, 20)$ (untere Reihe) das Strömungsfeld innerhalb des Tropfens zwei Wirbel ausbildet. 82

5.12 Instationäre Ergebnisse der Tropfendeformation in einer Scherströmung für die Parameterkonfiguration $(Ca, Re) = (0.15, 15)$. Zu sehen ist die zeitliche Entwicklung der freien Oberfläche zu $t = 0, t = 334.8, t = 379.2, t = 396.0, t = 397.2, t = 416.40$ (oben links bis unten rechts). Das Simulationsgebiet ist $(0, 3) \times (0, 1) \times (0, 2)$ groß und der Tropfen hat den Radius $r = 0.25$. Die Gittermaschenweite ist $(3/192, 1/64, 2/128)$ 83

5.13 Instationäre Ergebnisse der Tropfendeformation in einer Scherströmung für die Parameterkonfiguration $(Ca, Re) = (0.1, 25)$. Zu sehen ist die zeitliche Entwicklung der freien Oberfläche zu $t = 0, t = 178.8, t = 238.8, t = 330.0, t = 358.8, t = 373.2, t = 394.8, t = 418.8$ (von oben nach unten). Das Simulationsgebiet ist $(0, 14) \times (0, 1) \times (0, 2)$ groß und der Tropfen hat einen Radius von $r = 0.25$. Die Gittermaschenweite ist $(14/896, 1/64, 2/128)$. 84

6.1 Gitterglättung einer komplexen Geometrie. 88

6.2 Gittergenerierung für ein U-Boot mittels der Composite-Grids-Methode. . . 88

6.3 Randangepasstes Gitter für eine Kreisgeometrie mit der *Cut-Cell*-Technik. . 89

6.4	Polygonscan-Verfahren für eine vorgegebene Polygonzug-Geometrie. Der Becher auf der rechten Seite ist mit der bitweisen Innen- und Aussendifferenzierung auf einem kartesischen Gitter berechnet worden.	91
6.5	Aufteilung der Dreiecksebene in sieben disjunkte Regionen.	92
6.6	Verschiedene Niveaulinien zu unterschiedlichen Niveauwerten für $q(s, t)$	93
6.7	Vergleich der Isoflächen zum Niveauwert Null des Polygonscan-Verfahrens (linkes Bild) und der vorzeichenbehafteten Abstandsfunktion (mittleres Bild); das rechte Bild zeigt die direkte Überlagerung beider Ergebnisse.	94
6.8	Darstellung der transparenten Isofläche zum Niveauwert Null inklusive einer zentralen Schnittebene, welche die Abstandswerte in farbcodierter Form wiedergibt.	95
6.9	Hexagonaler Starrkörper: Globale, zeitunabhängige Koordinaten der Starrkörpergeometrie (links) und entsprechende lokale, zeitabhängige Koordinaten des Starrkörpers nach einer Translation und Rotation.	97
6.10	Das linke Bild zeigt die globalen, zeitunabhängigen Achsen einer nicht mit abgebildeten Starrkörpergeometrie und das rechte Bild zeigt die entsprechenden lokalen, zeitabhängigen Achsen nach einer Translation und Rotation.	98
6.11	Raum-Zeit Integrationstechnik für den Pfad C einer Euler-Methode und den Pfad T einer Lagrangeschen- und Semi-Lagrangeschen-Methode. Um bei der Semi-Lagrangeschen-Methode den Level-Set Wert im Punkt y zu ermitteln, muss von einem bekannten Wert z aus über den Pfad L aufintegriert werden. Dies steht analog zu einer Interpolation aus bekannten umgebenen Gitterwerten.	104
7.1	Verteilung der lokalen Level-Set-Darstellung einer Starrkörpergeometrie auf das mit zwölf Prozessoren parallelisierte globale Rechengitter. Ausserhalb dieses lokalen Gitters ist die Level-Set-Funktion für den Starrkörper konstant und somit nicht explizit erforderlich, womit substantieller Arbeitspeicherplatz gespart werden kann.	118
8.1	Direktvergleich des Laborexperiments von M. Greenhow und W.-M. Lin [76] (links) mit unserem numerischen Simulationsergebnis (rechts). Die Bilder zeigen einen mit konstanter Geschwindigkeit aufsteigenden Zylinder von 11cm Durchmesser zu den Zeitpunkten 0.198s (oben) und 0.208s (unten).	124
8.2	Der weitere chronologische Verlauf der numerischen Simulation zeigt deutlich das von Peregrine [147] als "abbrechender Wasserfall" (breaking waterfall) bezeichnete physikalische Verhalten.	125
8.3	Direktvergleich der Penetrationstiefe des Zylinders aus dem Laborexperiment von M. Greenhow und W.-M. Lin [76] (links) mit unserem Simulationsergebnis (rechts). Deformation der freien Oberfläche zu den Zeitpunkten $t = 0.305s$ (oben) und $t = 0.335s$ (unten).	126
8.4	Direktvergleich der Messdaten des Laborexperiments von M. Greenhow und W.-M. Lin [76] mit unserem Simulationsergebnis.	127
8.5	Initialisierung der Kugel im Kräftegleichgewicht innerhalb der Wasserphase.	129

8.6	Auf den Schwerpunkt der Kugel wirkende Geschwindigkeitskomponenten u (oben), v und w (unten), berechnet auf drei unterschiedlich feinen Gitterauflösungen über einen Zeitraum von $2s$	130
8.7	Mittelschnittebene der Anfangsbedingung bezüglich der Dichteverteilung des aufsteigenden Zylinders im Wasser auf dem $240 \times 160 \times 7$ -Gitter. Der Zylinder hat eine Dichte von $800kg/m^3$	131
8.8	Graph der Mittelschnittebene von der u - und v -Geschwindigkeitskomponente des aufsteigenden Zylinders, gerechnet auf dem $240 \times 160 \times 7$ -Gitter zur Zeit $t = 0.46s$	133
8.9	Simulationstest des Kräftegleichgewichts einer Kugel auf der freien Oberfläche. Zu Beginn der Simulation liegt die Kugel im hydrostatischen Gleichgewicht zur Hälfte im Wasser.	134
8.10	Mittelschnittebene der Materialdichte für den Testfall der Kugel an der freien Oberfläche in den drei betrachteten Gitterauflösungen 40^3 , 80^3 und 160^3 . Die blaue Farbe ist die Dichte der Luft, rot die des Wassers und grün die der Starrkörperkugel.	135
8.11	Untersuchung des Kräftegleichgewichts einer an der freien Oberfläche schwimmenden Kugel. Abgebildet ist die vertikale Bewegung des Kugelmittelpunktes bezüglich drei unterschiedlich feinen Gittern (gestrichelte Linie= 40^3 , durchgezogene Linie= 80^3 und fette Linie= 160^3 Gitterpunkte) über einen Zeitraum von 1.9 Sekunden. Die Wasseroberfläche befindet sich hierbei auf $0.15m$ Höhe (gerade Linie).	136
8.12	Das Forschungsschiff RV-Tangaroa in Fahrt. Man beachte, dass aufgrund des Wustbugs kaum Bugwellen generiert werden. Das Bild stammt aus Quelle [1].	138
8.13	Rechengebiet und Ausgangsposition der RV-Tangaroa für die Simulation der Schiffs-Rotation.	139
8.14	Bug- und Heckansicht der rotierenden Schiffskörpergeometrie aus Frontal- und Vogelperspektive zur Zeit $0.15s$. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, die mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist.	140
8.15	Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu den Zeiten $t = 0s, 0.25s, 0.5s$. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist.	141
8.16	Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu den Zeiten $t = 0.75s, 1.0s, 1.25s$. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist.	142
8.17	Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu den Zeiten $t = 1.5s, 1.75s, 2.0s$. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert ist.	143

8.18	Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu der Zeit 2.0s. Dargestellt ist die Schiffskörpergeometrie und die Wasseroberfläche, welche mit den Werten der turbulenten Viskosität (Eddy-Viskosität) koloriert sind.	144
8.19	Vogel- und Fischperspektive der rotierenden Schiffskörpergeometrie zu der Zeit 2.0s. Dargestellt ist die Schiffskörpergeometrie, die Wasseroberfläche und die Isofläche der turbulenten Viskosität (Eddy-Viskosität) zum Niveauwert 1.5 ₋₆ . Im linken Bild ist derjenige Teil der Isofläche visualisiert, der sich in der Luft befindet. Das rechte Bild dagegen zeigt den im Wasser liegenden Teil der Isofläche.	144
8.20	Rechengebiet und Ausgangsposition der RV-Tangaroa für die Simulation der Schiffs-Translation.	145
8.21	Geradeausfahrt der RV-Tangaroa-Schiffskörpergeometrie mit 12 Knoten. Zu sehen ist die Schiffskörpergeometrie, die freie Wasseroberfläche und die kolorierte turbulente Viskosität auf der Mittelschnittebene. Die Bilder sind chronologisch von oben nach unten entsprechend den Zeiten 0s, 1.9s, 3.9s, 9.9s, 15.9s und 21.9s aufgeführt.	147
8.22	Fischperspektive der Geradeausfahrt der RV-Tangaroa-Schiffskörpergeometrie mit 12 Knoten. Zu sehen ist die Schiffskörpergeometrie, die freie Wasseroberfläche koloriert mit der euklidischen Norm des Geschwindigkeitsfeldes. Die Bilder sind chronologisch von oben nach unten entsprechend den Zeiten 0s, 1.9s, 3.9s, 9.9s, 15.9s und 21.9s aufgeführt.	148
8.23	Geradeausfahrt der RV-Tangaroa-Schiffskörpergeometrie mit 12 Knoten. Zu sehen ist in dem obersten Bild die Fischperspektive auf die Schiffskörpergeometrie und die Wasseroberfläche zur Zeit $t = 21.9s$. Die Wasseroberfläche ist mit den Werten der turbulenten Viskosität koloriert. In den unteren drei Bildern ist zum selben Zeitpunkt die Schiffskörpergeometrie, die freie Wasseroberfläche und die Isofläche der turbulenten Viskosität zum Niveauwert 3.0 ₋₃ aus der Seiten-, der Vogel- und der Fischperspektive dargestellt. . . .	149
8.24	Geradeausfahrt der RV-Tangaroa-Schiffskörpergeometrie mit 12 Knoten zur Zeit $t = 21.4s$. Zu sehen ist im oberen Bild die Schiffskörpergeometrie und die Wasseroberfläche. Im mittleren Bild ist die Wasseroberfläche mit der Vertikalgeschwindigkeitskomponente koloriert. Im unteren Bild ist sie mit der euklidischen Norm des Geschwindigkeitsfeldes koloriert.	150
8.25	Initialisierung des aufsteigenden U-Bootes. Zu sehen ist die Geometrie des U-Bootes und das Wasser, in dem es sich zu Beginn befindet. Das obere Bild zeigt die Seitenansicht, das untere Bild die Ansicht von oben.	151
8.26	Aufsteigendes U-Boot zu den Zeiten $t=0s, 7s, 9s, 11s$ (von oben nach unten). Zu sehen ist die U-Boot-Geometrie und die Wasseroberfläche und eine Mittelschnittebene, welche mit der euklidischen Norm der Geschwindigkeitswerte koloriert ist.	153

8.27	Vogelperspektive des aufsteigendes U-Bootes. Das linke Bild zeigt wie das U-Boot gerade seinen höchsten Aufstiegspunkt erreicht. Das rechte Bild zeigt wie das U-Boot wieder absinkt, um in den Gleichgewichtszustand auf der Wasseroberfläche zu konvergieren.	154
8.28	Aufsteigendes U-Boot: zeitlicher Verlauf der Höhe des U-Boot-Schwerpunktes (oberer Graph) und die Aufstiegsgeschwindigkeit (unterer Graph).	155
8.29	Aufsteigendes U-Boot zur Zeit 11.2s. Linke Bildspalte zeigt die Bug- und Heckansicht der U-Boot-Geometrie inklusive einer Schnittebene, die mit der turbulenten Viskosität (Eddy-Viskosität) koloriert ist. In der rechten Bildspalte ist dieselbe Schnittebene dargestellt, nur ohne die U-Boot-Geometrie.	156

Tabellenverzeichnis

5.1	Relativer Fehler e und Fehlerkonvergenz η der Krümmungsberechnung κ in der L^1 - und L^∞ -Norm und der Masse m nach einer transportierten Periode der Einheitskugel.	73
5.2	Materialparameter für die Simulation der im Fluid aufsteigenden Blase. . .	76
5.3	Konvergenzstudie für die aufsteigende Blase zur Zeit $t = 0.075s$	76
5.4	Konvergenzstudie für die aufsteigende Blase zur Zeit $t = 0.075s$	77
8.1	Konvergenzstudie für den aufsteigenden Zylinder zur Zeit $t = 0.26s$	132
8.2	Relativer L^∞ -Fehler der Aufstiegsgeschwindigkeit des Zylinders zur Zeit $t = 0.26s$	132

Literaturverzeichnis

- [1] www.niwavessels.co.nz/tangaroa/.
- [2] <http://de.wikipedia.org/wiki/U-Boot>.
- [3] <http://math.mit.edu/~persson/thesis/>.
- [4] <https://computation.llnl.gov/casc/Overture/henshaw/overtureFigures/node4.html>.
- [5] www.econ.univpm.it/recchioni/w12/.
- [6] Message Passing Interface (MPI). <http://www.mcs.anl.gov/research/projects/mpi/>.
- [7] RV-Tangaroa. <http://gfs.sourceforge.net/examples/examples/tangaroa/tangaroa.gts>.
- [8] R. Abgrall, B. Nkonga, and R. Saurel. Efficient numerical approximation of compressible multi-material flow for unstructured meshes. *Computers and Fluids*, 32(4):571–605, 2003.
- [9] R. Abgrall and R. Saurel. Discrete equations for for physical and numerical compressible multiphase mixtures. *J. Comput. Phys.*, 186(2):361–396, April 2003.
- [10] A. Amsden and F. Harlow. A simplified MAC technique for incompressible fluid flow calculations. *J. Comput. Phys.*, 6:322, 1970.
- [11] L. K. Antanovskii. A phase field model of capillarity. *Phys. Fluids*, 7:747–753, 1995.
- [12] E. Aulisa, S. Manservigi, and R. Scardovelli. A surface marker algorithm coupled to an area-preserving marker redistribution method for three-dimensional interface tracking. *J. Comput. Phys.*, 197(2):555–584, July 2004.
- [13] E. Bänsch. Finite element discretization of the Navier-Stokes equations with a free capillary surface. *Numer. Math.*, 88(2):203–235, 2001.
- [14] A. Béliveau, A. Fortin, and Y. Demay. A two-dimensional numerical method for the deformation of drops with surface tension. *Int. J. Comput. Fluid Dyn.*, 10:225–240, 1998.
- [15] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl. Meshless methods: An overview and recent developments. *Comp. Meth. App. Mech. Engrg.*, 139:3–47, 1996.

- [16] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl. Meshless methods: An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, 139:3–47, 1996.
- [17] F. Bet, D. Hänel, and S. D. Sharma. Numerical simulation of ship flow by a method of artificial compressibility. In *Proc. of the Twenty-Second Symposium on Naval Hydrodynamics*, Washington D. C., 1998. National Academy Press.
- [18] G. Birkhoff. Helmholtz and Taylor instabilities. In *Symposium on Applied Mathematics*, volume 13, pages 55–76. American Mathematical Society, Providence, RI, 1962.
- [19] A. Bonito, M. Picasso, and M. Laso. Numerical simulation of 3d viscoelastic flows with free surfaces. *J. Comput. Phys.*, 215(2):691–716, 2006.
- [20] J. P. Boris and E. S. Oran. *Numerical Simulation of Reactive Flow*. Cambridge University Press, 2 edition, 2001.
- [21] J. U. Brackbill. The ringing instability in particle-in-cell calculations of low-speed flow. *J. Comput. Phys.*, 75:469–484, 1988.
- [22] J. U. Brackbill, D. B. Kothe, and H. M. Ruppel. FLIP: A low-dissipation, particle-in-cell method for fluid flow. *Computer Physics Communications*, 48:25–38, 1988.
- [23] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *J. Comput. Phys.*, 100:335, 1992.
- [24] J. U. Brackbill and H. M. Ruppel. FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. *J. Comput. Phys.*, 65:314–343, 1985.
- [25] U. P. Bulgarelli, C. Lugni, and M. Landrini. Numerical modelling of free-surface flows in ship hydrodynamics. *Int. J. Numer. Meth. Fluids*, 43:465–481, 2003.
- [26] B. L. Buzbee, F. W. Dorr, J. A. George, and G. H. Golub. The direct solution of the discrete Poisson equation on irregular regions. *SIAM J. Num. Anal.*, 8:722–736, 1971.
- [27] A. Caboussat, M. Picasso, and J. Rappaz. Numerical simulation of free surface incompressible liquid flows surrounded by compressible gas. *J. Comput. Phys.*, 203(2):626–649, 2005.
- [28] G. Caginalp. Stefan and Hele-Shaw models as asymptotic limits of the phase-field equations. *Physical Review A*, 39:5887–5896, 1989.
- [29] J. W. Cahn and J. E. Hilliard. Free energy of a nonuniform system. iii. nucleation in a two-component incompressible fluid. *J. of Chem. Phys.*, 31:688–699, 1959.
- [30] R. Caiden, R. P. Fedkiw, and C. Anderson. A numerical method for two-phase flow consisting of separate compressible and incompressible regions. *J. Comput. Phys.*, 166:1–27, 2001.

- [31] S. Canić, D. Lamponi, A. Mikelić, and J. Tampaca. Self-consistent effective equations modeling blood flow in medium-to-large compliant arteries. *SIAM J. Multiscale Analysis and Simulation*, 3(3):559–606, 2005.
- [32] C. Canuto, M. Y. Hussaini, A. Quaterioni, and T. A. Zang. *Spectral Methods in Fluid Dynamics*. Springer, Berlin/Heidelberg/New York, 1988.
- [33] Y. C. Chang, T. Y. Hou, B. Merriman, and S. Osher. A level set formulation of Eulerian interface capturing methods for incompressible fluid flows. *J. Comput. Phys.*, 124:449–464, 1996.
- [34] S. Chen, D. B. Johnson, and P. E. Raad. The surface marker method. In *Computational Modelling of Free and Moving Boundary Problems: Fluid Flow*, volume 1, pages 223–234, Southampton, Great Britain, 1991. Computational Mechanics Publications.
- [35] S. Chen, D. B. Johnson, P. E. Raad, and D. Fadda. The surface marker and micro cell method. *International Journal for Numerical Methods in Fluids*, 25:749–778, 1997.
- [36] I.-L. Chern, J. Glimm, O. McBrien, B. Plohr, and S. Yaniv. Front tracking for gas dynamics. *J. Comput. Phys.*, 62:83–110, 1986.
- [37] A. J. Chorin. Numerical solution of the Navier–Stokes equations. *Math. Comput.*, 22(104):745–762, 1968.
- [38] A. J. Chorin and J. E. Marsden. *A Mathematical Introduction to Fluid Mechanics*. Springer, New York, 3 edition, 1993. Corr. 1998.
- [39] R. Codina and O. Soto. A numerical model to track two-fluid interfaces based on a stabilized finite element method and a level set technique. *Int. J. Numer. Meth. Fluids*, 40:293–301, 2002.
- [40] P. Colella and P. Woodward. The piecewise parabolic method (PPM) for gasdynamical simulations. *J. Comput. Phys.*, 54:174–201, 1984.
- [41] R. Cortez, C. S. Peskin, J. Stockie, and D. Varela. Parametric resonance in immersed elastic boundaries. *SIAM J. Appl. Math.*, 65:494–520, 2004.
- [42] R. Courant, E. Isaacson, and M. Rees. On the solution of nonlinear hyperbolic differential equations by finite differences. *Comm. Pure Appl. Math.*, 5:243–249, 1952.
- [43] R. Croce. Ein paralleler, dreidimensionaler Navier-Stokes-Löser für inkompressible Zweiphasenströmungen mit Oberflächenspannung, Hindernissen und dynamischen Kontaktflächen. Diplomarbeit, University of Bonn, Institute for Applied Mathematics, April 2002.
- [44] R. Croce, M. Griebel, and M. A. Schweitzer. A parallel level-set approach for two-phase flow problems with surface tension in three space dimensions. Preprint 157, Sonderforschungsbereich 611, Universität Bonn, 2004.

- [45] R. Croce, M. Griebel, and M. A. Schweitzer. Numerical simulation of droplet-deformation by a level set approach with surface tension. SFB preprint 395, Institut für Numerische Simulation, Universität Bonn, 2008.
- [46] R. Croce, M. Griebel, and M. A. Schweitzer. Numerical simulation of bubble and droplet-deformation by a level set approach with surface tension in three dimensions. *Int. J. Numer. Meth. Fluids*, Published Online:n/a, Apr 2009.
- [47] K. L. D. Kothe, D. Juric and B. Lally. Numerical recipes for mold filling simulation. In *Proceedings of the Eighth International Conference on Modeling of Casting, Welding, and Advanced Solidification Processes*, San Diego, CA, 1998.
- [48] B. J. Daly. Numerical study of two fluid Rayleigh–Taylor instability. *The Physics of Fluids*, 10(2):297–307, 1967.
- [49] B. J. Daly. Numerical study of the effect of surface tension on interface instability. *The Physics of Fluids*, 12(7):1340–1354, 1969.
- [50] B. J. Daly. A technique for including surface tension effects in hydrodynamic calculations. *J. Comput. Phys.*, 4:97, 1969.
- [51] B. Desjardins, M. J. Esteban, C. Grandmont, and P. L. Tallec. Weak solutions for a fluid-elastic structure interaction model. *Rev. Mat. Comput.*, 14(2):523–538, 2001.
- [52] W. Dettmer and D. Peric. A computational framework for fluid-rigid body interaction: Finite element formulation and applications. *Comput. Methods Appl. Mech. Engrg.*, 195:1633–1666, 2006.
- [53] D. Eberly. Distance between point and triangle in 3d. Geometric Tools, LLC, 1999.
- [54] H. Edelsbrunner and E. P. Mücke. Simulation of Simplicity: A technique to cope with degenerate cases in geometric algorithms. *ACM Transactions on Graphics*, 9(1):66–104, 1990.
- [55] M. Engel and M. Griebel. Flow simulation on moving boundary-fitted grids and application to fluid-structure interaction problems. *Int. J. Numer. Meth. Fluids*, 50(4):437–468, 2006.
- [56] B. Engquist, A. K. Tornberg, and R. Tsai. Discretization of Dirac delta functions in level set methods. *J. Comput. Phys.*, 207:28–51, 2005.
- [57] R. P. Fedkiw, B. Merriman, and S. Osher. Numerical methods for a one-dimensional interface separating compressible and incompressible flows. In V. Venkatakrishnan, M. Salas, and S. Chakravarthy, editors, *Barriers and Challenges in Computational Fluid Dynamics*, pages 155–194. Kluwer Academic Publishers, 1998.
- [58] J. Feng, H. H. Hu, and D. D. Joseph. Direct simulation of initial value problems for the motion of solid bodies in a newtonian fluid. 1. sedimentation. *J. Fluid Mech.*, 261:95–134, 1994.

- [59] J. H. Ferziger and M. Peric. *Computational Methods for Fluid Dynamics*. Springer-Verlag, 1996.
- [60] O. Forster. *Analysis 3*. Vieweg, Braunschweig/Wiesbaden, 1992.
- [61] M. M. Francois, S. J. Cummins, E. D. Dendy, D. B. Kothe, J. M. Sicilian, and M. W. Williams. A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework. *J. Comput. Phys.*, 213:141–173, 2006.
- [62] A. Frohn and N. Roth. *Dynamics of Droplets*. Springer, Berlin/Heidelberg/New York, 2000.
- [63] B. Fryxell, E. Müller, and D. Arnett. Instabilities and clumping in sn 1987a. i. early evolution in two dimensions. *Astrophysical Journal*, 367:619–634, 1991.
- [64] P. H. Gaskell and A. K. C. Lau. Curvature-compensated convective transport: Smart a new boundedness-preserving transport algorithm. *Int. J. Numer. Methods Fluids*, 8:617, 1988.
- [65] A. S. Geller, S. H. Lee, and L. G. Leal. Motion of a particle normal to a deformable surface. *J. of Fluid Mech.*, 169:27–69, 1986.
- [66] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics: Theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181:375–389, 1977.
- [67] J. Glimm, J. W. Grove, X. L. Li, K.-M. Shyue, Y. Zeng, and Q. Zhang. 3-dimensional front tracking. *SIAM J. Sci. Computing*, 19:703–727, 1998.
- [68] J. Glimm and O. A. McBrien. A computational model for interfaces. *Advances in Applied Mathematics*, 6:422–435, 1985.
- [69] R. Glowinski and L. H. Juarez. Finite element method and operator-splitting for a time-dependent viscous incompressible free-surface flow. *Comp. Fluid Dynam. J.*, 12(3):459–468, 2003.
- [70] R. Glowinski, T.-W. Pan, T. I. Hesla, and D. D. Joseph. A distributed lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 25:755–794, 1999.
- [71] R. Glowinski, T. W. Pan, T. I. Hesla, D. D. Joseph, and J. Périaux. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: Application to particulate flow. *J. Comput. Phys.*, 169(2):363–426, 2001.
- [72] R. Glowinski, T. W. Pan, and J. Périaux. A fictitious domain method for dirichlet problem and applications. *Comput. Methods Appl. Mech. Eng.*, 111(3-4):283–303, 1994.

- [73] H. Goldstein. *Klassische Mechanik*. Aula-Verlag, Wiesbaden, 1989.
- [74] S. Gottlieb, C. W. Shu, and E. Tadmor. Strong stability preserving high-order time discretization methods. Report 2000-15, NASA/CR-2000-210093 ICASE, April 2000.
- [75] C. Grandmont and Y. Maday. Existence for an unsteady fluid-structure interaction problem. *M²AN*, 34(3):609-636, 2000.
- [76] M. Greenhow and W.-M. Lin. Nonlinear free surface effects: experiments and theory. Technical Report 83-19, Massachusetts Inst Of Tech Cambridge Dept Of Ocean Engineering, sep 1983.
- [77] M. Griebel, T. Dornseifer, and T. Neunhoeffler. *Numerical Simulation in Fluid Dynamics, A Practical Introduction*. SIAM, Philadelphia, 1998.
- [78] M. Griebel, W. Merz, and T. Neunhoeffler. Mathematical modelling and numerical simulation of freezing processes of a supercooled melt under consideration of density changes. *Computing and Visualization in Science*, 3:1-19, 1999.
- [79] M. Griebel, B. Metsch, D. Oeltz, and M. A. Schweitzer. Coarse grid classification: A parallel coarsening scheme for algebraic multigrid methods. *Numerical Linear Algebra with Applications*, 13(2-3):193-214, 2006.
- [80] M. Griebel and M. A. Schweitzer, editors. *Meshfree Methods for Partial Differential Equations*, volume 26 of *Lecture Notes in Computational Science and Engineering*. Springer, 2002.
- [81] M. Griebel and M. A. Schweitzer, editors. *Meshfree Methods for Partial Differential Equations II*, volume 43 of *Lecture Notes in Computational Science and Engineering*. Springer, 2005.
- [82] M. Griebel and M. A. Schweitzer, editors. *Meshfree Methods for Partial Differential Equations III*, volume 57 of *Lecture Notes in Computational Science and Engineering*. Springer, 2006.
- [83] M. Griebel and M. A. Schweitzer, editors. *Meshfree Methods for Partial Differential Equations IV*, volume 65 of *Lecture Notes in Computational Science and Engineering*. Springer, 2008.
- [84] C. Großmann and H. G. Roos. *Numerik partieller Differentialgleichungen*. Teubner Studienbücher, Mathematik, 1994.
- [85] S. Gross. *Numerical methods for three-dimensional incompressible two-phase flow*. PhD thesis, RWTH Aachen University, 2008.
- [86] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, and S. Zaleski. Volume-of-Fluid interface tracking with smoothed surface stress methods for three-dimensional flows. *J. Comput. Phys.*, 152:423-456, 1999.

- [87] F. Harlow and J. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Phys. Fluids*, 8:2182–2189, 1965.
- [88] F. H. Harlow. Hydrodynamic problems involving large fluid distortions. *Journal of the Association for Computing Machinery*, 4:137–142, 1957.
- [89] F. H. Harlow. PIC and its progeny. *Computer Physics Communications*, 48:1–11, 1988.
- [90] F. H. Harlow and M. W. Evans. A machine calculation method for hydrodynamics problems. Technical Report LAMS-1956, Los Alamos National Laboratory, 1955.
- [91] A. Harten, B. Enquist, S. Osher, and S. Chakravarthy. Uniformly high order essentially non-oscillatory schemes, III. *J. Comput. Phys.*, 71:231, 1987.
- [92] W. Hauger, W. Schnell, and D. Gross. *Technische Mechanik*, volume 1-3. Springer-Verlag, Berlin, 1993.
- [93] C. W. Hirt. Heuristic stability theory for finite-difference-equations. *J. Comput. Phys.*, 2:339–355, 1968.
- [94] C. W. Hirt and B. D. Nichols. Volume of fluid VOF method for the dynamics of free boundaries. *J. Comput. Phys.*, 39:201–225, 1981.
- [95] T. Y. Hou. Numerical solutions to free boundary problems. *Acta Numerica*, 4:335–415, 1995.
- [96] H. H. Hu. Direct simulation of flows of solid-liquid mixtures. *Int. J. Multiphase Flow*, 22:335–352, 1996.
- [97] H. H. Hu, D. D. Joseph, and A. F. Fortes. Experiments and direct simulation of fluid particle motions. *Int. Vid. J. Eng. Res.*, 2:17, 1997.
- [98] W. Huber. *Paralleles Rechnen*. Oldenbourg, München/Wien, 1997.
- [99] M. A. Hyman. Non-iterative numerical solution of boundary-value problems. *Appl. Sci. Res.*, B 2:325–351, 1952.
- [100] S. R. Idelsohn, M. A. Storti, and E. O’ate. Lagrangian formulations to solve free surface incompressible inviscid fluid flows. *Comput. Meth. Appl. Mech. Engrg.*, 191:583–593, 2001.
- [101] D. Jacqmin. Calculation of two-phase Navier-Stokes flows using phase-field modeling. *J. Comput. Phys.*, 155:96–127, 1999.
- [102] G. S. Jiang and D. Peng. Weighted ENO schemes for Hamilton–Jacobi equations. *SIAM J. Sci. Comput.*, 21(6):2126–2143, 2000.
- [103] G. S. Jiang and C. W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126:202–228, 1996.

- [104] M. W. Jones. 3d distance from a point to a triangle. Technical report, Department of Computer Science, University of Wales Swansea, February 1995.
- [105] D. Juric. Direct numerical simulation of solidification microstructures affected by fluid flow. In *Modeling of Casting, Welding, and Advanced Solidification Processes VIII*. TMS Publishers, New York, 1998.
- [106] D. Juric and G. Tryggvason. A front-tracking method for dendritic solidification. *J. Comput. Phys.*, 123:127–148, 1996.
- [107] D. Juric and G. Tryggvason. Numerical simulations of phase change in microgravity. *Heat Transfer in Microgravity Systems*, 332:33–44, 1996.
- [108] D. Juric and G. Tryggvason. Computations of boiling flows. *International journal of multiphase flow*, 24:387–410, 1998.
- [109] M. Kang, R. P. Fedkiw, and X.-D. Liu. A boundary condition capturing method for multiphase incompressible flow. *J. Sci. Comput.*, 15:323–360, 2000.
- [110] K. Kim, Y. Nakayama, and R. Yamamoto. A smoothed profile method for simulating charged colloidal dispersions. *Comput. Phys. Commun.*, 169:104–106, 2005.
- [111] Y. Kim and C. S. Peskin. Penalty immersed boundary method for an elastic boundary with mass. *Phys. Fluids*, 19(5):053103 (18 Seiten), 2007.
- [112] R. Kobayashi. Modeling and numerical simulations of dendritic crystal growth. *Physica D*, 63:410–423, 1993.
- [113] K. Königsberger. *Analysis 2*. Springer, Berlin/Heidelberg/New York, 1997.
- [114] B. Koren, M. R. Lewis, E. H. van Brummelen, and B. van Leer. Riemann-problem and level-set approaches for two-fluid flow computations i. linearized godunov scheme. Technical report, MAS-R0112, ISSN 1386-3703, CWI, Amsterdam, 2001.
- [115] D. B. Kothe. Perspectives on Eulerian finite volume methods for incompressible interfacial flows. In H. C. Kuhlmann and H.-J. Rath, editors, *Free Surface Flows*, pages 267–331. Springer, 1999.
- [116] D. B. Kothe, J. U. Brackbill, and C. K. Choi. Implosion symmetry of heavy-ion-driven inertia confinement fusion targets. *Physics of Fluids B*, 2:1898–1906, 1990.
- [117] D. Kröner. *Numerical Schemes for Conservation Laws*. Wiley and Teubner, 1996.
- [118] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti. Modelling merging and fragmentation in multiphase flows with SURFER. *J. Comput. Phys.*, 113(1):134–147, 1994.
- [119] L. D. Landau and E. M. Lifshitz. *Fluid Mechanics*. Pergamon Press, Oxford, 1959.

- [120] A. Leonard. Vortex methods for flow simulation. *J. of Comp. Phys.*, 37:289–335, 1980.
- [121] A. Leonard. Computing three-dimensional incompressible flows with vortex elements. *Annual Reviews in Fluid Mechanics*, 17:523–559, 1985.
- [122] B. P. Leonard. A stable and accurate convective modelling procedure based on quadric upstream interpolation. *Comput. Methods Appl. Mech. Eng.*, 19:59, 1979.
- [123] R. J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser, Basel, 1990.
- [124] R. J. LeVeque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 31:1019–1044, 1994.
- [125] R. J. LeVeque and Z. Li. Immersed interface methods for stokes flow with elastic boundaries or surface tension. *SIAM J. Sci. Stat. Comp.*, 18:709–735, 1997.
- [126] R. W. Lewis, S. E. Navti, and C. Taylor. A mixed lagrangian-eulerian approach to modelling fluid flow during mould filling. *Int. J. Numer. Meth. Fluids*, 25:931–952, 1997.
- [127] J. Li and Y. Y. Renardy. Numerical study of flows of two immiscible liquids at low reynolds number. *SIAM Review*, 42(3):417–439, 2000.
- [128] J. Li, Y. Y. Renardy, and M. Renardy. Numerical simulation of breakup of a viscous drop in simple shear flow through a volume-of-fluid method. *Phys. Fluids*, 12:269–282, 2000.
- [129] X. Luo, M. R. Maxey, and G. E. Karniadakis. Smoothed profile method for particulate flows: Error analysis and simulations. *J. Comput. Phys.*, 228:1750–1769, 2009.
- [130] B. Metsch. Ein paralleles graphenbasiertes algebraisches Mehrgitterverfahren. Diplomarbeit, Institut für Numerische Simulation, Universität Bonn, 2004.
- [131] J. J. Monaghan. Particle methods for hydrodynamics. *Computer Physics Reports*, 3:71–124, 1985.
- [132] J. J. Monaghan. Smoothed particle hydrodynamics. *Ann. Rev. Astron. Astrophys.*, 30:543–574, 1992.
- [133] J. J. Monaghan. Simulating free surface flows with SPH. *J. Comput. Phys.*, 110:399–406, 1994.
- [134] J. P. Morris. Simulating surface tension with smoothed particle hydrodynamics. *International Journal for Numerical Methods in Fluids*, 33(3):333–353, 1999.
- [135] J. P. Morris, P. J. Fox, and Y. Zhu. Modeling low Reynolds number incompressible flows using SPH. *J. Comput. Phys.*, 136:214–226, 1997.

- [136] W. Mulder, S. Osher, and J. A. Sethian. Computing interface motion in compressible gas dynamics. *J. Comput. Phys.*, 100:209–228, 1992.
- [137] S. Müller, P. Helluy, and J. Ballmann. Numerical simulation of cavitation bubbles by compressible two-phase fluids. Technical Report 273, IGPM, RWTH Aachen, February 2007.
- [138] Y. Nakayama and R. Yamamoto. Simulation method to resolve hydrodynamic interactions in colloidal dispersions. *Physical Review E*, 71, 036707:1–7, 2005.
- [139] P. E. Nikravesch and I. S. Chung. Application of Euler parameters to the dynamic analysis of three-dimensional constrained mechanical systems. *Journal of Mechanical Design*, 104:785–791, 1982.
- [140] W. F. Noh and P. Woodward. SLIC—Simple line interface calculation. In A. V. Voooren and P. Zandbergen, editors, *Fifth International Conference on Fluid Dynamics, Lecture Notes in Physics*, volume 59, pages 330–340. Springer, 1976.
- [141] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Number 153 in Applied Mathematical Sciences. Springer, New York/Berlin/Heidelberg, 2003.
- [142] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations. *J. Comput. Phys.*, 79(1):12–49, November 1988.
- [143] P.-J.-Roache. *Fundamentals of computational fluid dynamics*. NM: Hermosa Publishers Publ., 1998.
- [144] N. Parolini and A. Quarteroni. Mathematical models and numerical simulations for the america’s cup. *Comp. Meth. Appl. Mech. Eng.*, 194:1001–1026, 2004.
- [145] N. A. Patankar, P. Singh, D. D. Joseph, R. Glowinski, and T.-W. Pan. A new formulation of the distributed lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 26:1509–1524, 2000.
- [146] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang. A PDE-based fast local level set method. *J. Comput. Phys.*, 155(2):410–438, November 1999.
- [147] D. H. Peregrine. The fascination of fluid dynamics. *J. Fluid Mech.*, 106:59–80, 1981.
- [148] C. S. Peskin. Numerical analysis of blood flow in the heart. *J. Comput. Phys.*, 25:220–252, 1977.
- [149] R. Peyret and T. D. Taylor. *Computational Methods for Fluid Flow*. Springer Series in Computational Physics. Springer, 1983.
- [150] M. Picasso, J. Rappaz, A. Reist, M. Funk, and H. Blatter. Numerical simulation of the motion of a two dimensional glacier. *Int. J. Num. Meth. Engrg.*, 60:995–1009, 2004.

- [151] C. Pozrikidis. *Boundary integral and singularity methods for linearized viscous flow*. Cambridge University Press, University of California, San Diego, 1992.
- [152] S. Quan and D. P. Schmidt. A moving mesh interface tracking method for 3d incompressible two-phase flows. *J. Comput. Phys.*, Volume 221(2):761–780, February 2007.
- [153] A. Quarteroni and L. Formaggia. *Mathematical Modelling and Numerical Simulation of the Cardiovascular System*, chapter Modelling of Living Systems. Elsevier, 2003.
- [154] K. F. R. Courant and H. Lewy. über die partiellen differenzgleichungen der mathematischen physik. *Mathematische Annalen*, 100:32–74, 1928.
- [155] P. E. Raad and R. Bidoae. The three-dimensional Eulerian-Lagrangian marker and micro cell method for the simulation of free surface flows. *J. Comput. Phys.*, 203:668–699, 2005.
- [156] J. M. Rallison and A. Acrivos. A numerical study of the deformation and burst of a viscous drop in an extensional flow. *J. of Fluid Mech.*, 89:191–200, 1978.
- [157] Y. Renardy, S. Popinet, L. Duchemin, M. Renardy, S. Zaleski, C. Josserand, M. A. Drumright-Clarke, D. Richard, C. Clanet, and D. Quèrè. Pyramidal and toroidal water drops after impact on a solid surface. *J. Fluid. Mech.*, 484:69–83, 2003.
- [158] Y. Renardy and M. Renardy. Prost: A parabolic reconstruction of surface tension for the volume-of-fluid method. *J. Comput. Phys.*, 183:400–421, 2002.
- [159] Y. Renardy, M. Renardy, T. Chinyoka, D. B. Khismatullin, and J. Li. A viscoelastic vof-prost code for the study of drop deformation. In *Proceedings of the ASME Heat Transfer/Fluids Engineering Summer Conference*, Charlotte, North Carolina, 2004.
- [160] W. J. Rider and D. B. Kothe. Stretching and tearing interface tracking methods. Technical Report AIAA 95-1717, Presented at the 12th AIAA CFD conference, AIAA, 1995.
- [161] R. N. Rieben, D. A. White, B. K. Wallin, and J. M. Solberg. An arbitrary lagrangian-eulerian discretization of mhd on 3d unstructured grids. *J. Comput. Phys.*, 226(1):534–570, 2007.
- [162] M. Rieber and A. Frohn. Numerical study on the mechanism of splashing. *Int. Journal of Heat and Fluid Flow*, 20:455–461, 1999.
- [163] L. Rosenhead. The point vortex approximation of a vortex sheet. In *Proceedings of the Royal Society of London, Series A*, number 134, pages 170–192, 1932.
- [164] V. K. Saul’ev. On solving boundary-value problems on high-performance computers by fictitious-domain methods. *Sib. Math. J.*, 4:912–925, 1963.

- [165] R. Scardovelli and S. Zaleski. Direct numerical simulation of free surface and interfacial flow. *Annu. Rev. Fluid Mech.*, 31:567–603, 1999.
- [166] R. Schaback and H. Werner. *Numerische Mathematik*. Springer, Berlin/Heidelberg/New York, 1993.
- [167] H. A. Schwarz. *Gesammelte mathematische Abhandlungen*. Band 2. Springer, Berlin/Heidelberg/New York, 1890.
- [168] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.
- [169] A. A. Shabana. *Computational Dynamics*. John Wiley & Sons, New York, 1994.
- [170] X. Shi and S. P. Lim. A lbm-dlm/fd method for 3d fluid-structure interactions. *J. Comput. Phys.*, 226(2):2028–2043, 2007.
- [171] Y. D. Shikhmurzaev. The moving contact line on smooth solid surface. *Int. J. Multiphase Flow*, 19:589, 1993.
- [172] Y. D. Shikhmurzaev. Moving contact lines in liquid/liquid/solid systems. *J. Fluid Mechanics*, 334:211, 1997.
- [173] C. W. Shu. High order ENO and WENO schemes for computational fluid dynamics. In T. Barth and H. Deconinck, editors, *High-Order Methods for computational physics*. Springer, Berlin/Heidelberg/New York, 1999.
- [174] K.-M. Shyue. A fluid-mixture type algorithm for compressible multicomponent flow with van der Waals equation of state. *J. Comput. Phys.*, 156:43–88, 1999.
- [175] W. Shyy. *Computational Modeling for Fluid Flow and Interfacial Transport*. Elsevier Science, 1994.
- [176] W. Shyy, H. Udaykumar, M. Rao, and R. Smith. *Computational Fluid Dynamics with Moving Boundaries*. Taylor and Francis, 1996.
- [177] V. Sibillo, G. Pasquariello, M. Simeone, V. Cristini, and S. Guido. Drop Deformation in Microconfined Shear Flow. *Phys. Rev. Lett.*, 97:054502, 2006.
- [178] J. Smagorinsky. General circulation experiments with the primitive equations, i, the basic experiment. *Mon. Weather Rev.*, 91:99–165, 1963.
- [179] J. Strain. Semi-lagrangian methods for level set equations. *J. Comput. Phys.*, 151(2):498–533, 1999.
- [180] J. Strybny, C. Thorenz, R. Croce, and M. Engel. A parallel 3d free surface navier-stokes solver for high performance computing at the german waterways administration. In *The 7th Int. Conf. on Hydrosience and Engineering (ICHE-2006)*, Philadelphia, USA, September 2006.

- [181] M. Sussman and E. Fatemi. An efficient, interface preserving level set re-distancing algorithm and its application to incompressible fluid flow. *SIAM J. Sci. Comput.*, 20:1165–1191, 1999.
- [182] M. Sussman and M. Y. Hussaini. A discontinuous spectral element method for the level set equation. *J. Sci. Comput.*, 19(1–3):479–500, 2003.
- [183] M. Sussman and G. Puckett. A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows. *J. Comput. Phys.*, 162:301–337, 2000.
- [184] M. Sussman and P. Smereka. Axisymmetric free boundary problems. *J. of Fluid Mech.*, 341:269–294, 1997.
- [185] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.*, 114(1):146–159, 1994.
- [186] R. Temam. Sur l’approximation de la solution des equations de Navier-Stokes par la méthode des pas fractionnaires II. *Arch. Rational Mech. Anal.*, 33:377–385, 1969.
- [187] X. Tong, C. Beckermann, and A. Karma. Phase field simulation of dendritic growth with convection. In *Modeling of Casting, Welding, and Advanced Solidification Processes VIII*. TMS Publishers, 1998.
- [188] G. Tryggvason, B. Bunner, O. Ebrat, and W. Tauber. Computations of multiphase flow by a finite difference/front tracking method. i. multi-fluid flows. In *Lecture Notes for the 29th Computational Fluid Dynamics Lecture Series*. Karman Institute for Fluid Mechanics, Belgium, February 23-27 1998.
- [189] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, and Y. J. Jan. A front tracking method for the computations of multiphase flow. *J. Comput. Phys.*, 169:708–759, 2001.
- [190] S. O. Unverdi and G. Tryggvason. Computations of multi-fluid flows. *Physica D*, 60:70–83, 1992.
- [191] S. O. Unverdi and G. Tryggvason. A front-tracking method for viscous, incompressible multi-fluids flow. *J. Comput. Phys.*, 100:25–37, 1992.
- [192] S. P. van der Pijl, A. Segal, C. Vuik, and P. Wesseling. A mass conserving level-set method for modelling of multiphase flows. *Int. J. Numer. Meth. Fluids*, 47(4):339–361, 2005.
- [193] H. van der Vorst. Bi-cgstab: A fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 13:631, 1992.
- [194] J. van Kan. A second-order accurate pressure-correction scheme for viscous incompressible flow. *SIAM J. Sci. Stat. Comput.*, 7:870, 1986.

- [195] A. Varonos and G. Bergeles. Development and assessment of a variable-order non-oscillatory scheme for convection term discretization. *Int. J. Numer. Methods Fluids*, 26:1, 1998.
- [196] A. Volkwein. *Numerische Simulation von flexiblen Steinschlagschutzsystemen*. PhD thesis, Eidgenössische Technische Hochschule Zürich, 2004.
- [197] C. W. Wang, T. G. Liu, and B. C. Khoo. A real ghost fluid method for the simulation of multimediuim compressible flow. *SIAM J. Sci. Comput.*, 28(1):278–302, 2006.
- [198] S. L. Wang, R. F. Sekerka, A. A. Wheeler, B. T. Murray, S. R. Coriell, and G. B. McFadden. Thermodynamically consistent phase-field models for solidification. *Physica D*, 69:189–200, 1993.
- [199] A. A. Wheeler, B. T. Murray, and R. J. Schaefer. Computation of dendrites using a phase-field model. *Physica D*, 66:243–262, 1993.
- [200] J. Wittenburg. *Dynamics of Systems of Rigid Bodies*. B. G. Teubner, Stuttgart, 1977.
- [201] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.*, 54:115–173, 1984.
- [202] T. Yabe, Y. Ogata, K. Takizawa, T. Kawai, A. Segawa, and K. Sakurai. The next generation CIP as a conservative semi-lagrangian solver for solid, liquid and gas. *J. Comput. and Appl. Math.*, 149:267–277, 2002.
- [203] T. Yabe, F. Xiao, and T. Utsumi. The constrained interpolation profile method for multiphase analysis. *J. Comput. Phys.*, 169:556–593, 2001.
- [204] B. Yang and A. Prosperetti. A second-order boundary-fitted projection method for free-surface flow computations. *J. Comput. Phys.*, 213:574–590, 2006.
- [205] R. W. Yeung. Numerical methods in free-surface flows. *Annual Reviews in Fluid Mechanics*, 14:395–442, 1982.
- [206] J.-D. Yu, S. Sakai, and J. A. Sethian. Two-phase viscoelastic jetting. *J. Comput. Phys.*, 220(2):568–585, January 2007.
- [207] Z. Yu. A DLM/FD method for fluid/flexible-body interactions. *J. Comput. Phys.*, 207(1):1–27, 2005.
- [208] S. T. Zalesak. A preliminary comparison of modern shock-capturing schemes: Linear advection. In R. Vichnevetsky and R. S. Stepleman, editors, *Advances in Computer Methods for Partial Differential Equations*, volume 6, pages 15–22. IMACS, 1987.
- [209] X. Zhang, J.-S. Chen, and S. Osher. A multiple level set method for modeling grain boundary evolution of polycrystalline materials. *Interaction and Multiscale Mechanics*, 1(2):191–209, 2008.

- [210] Y. L. Zhang, K. S. Yeo, B. C. Khoo, and W. K. Chong. Three-dimensional computation of bubbles near a free surface. *J. Comput. Phys.*, 146:105–123, 1998.
- [211] MPICH-Software-Package. <http://www.mcs.anl.gov/research/projects/mpi/mpich1/>.