

TU Berlin
Institut für Mathematik
Lehrstuhl für Numerische Mathematik

Bachelorarbeit

Modellierung und Simulation von Dispersionen in turbulenten Strömungen

Manuel Baumann
Matrikelnr.: 314394
mbaumann@math.tu-berlin.de

Erstgutachter: Prof. Dr. Volker Mehrmann
Zweitgutachter: Prof. Dr. Jörg Liesen
Betreuung: Jan Heiland

12. August 2011

Inhaltsverzeichnis

1	Einleitung	7
1.1	Problemstellung und Motivation	7
1.2	Einordnung der Arbeit in den wissenschaftlichen Kontext	8
1.3	Gliederung	9
2	Mathematische Modellierung	10
2.1	Die Reynolds-Averaged-Navier-Stokes Gleichung (RANS)	10
2.1.1	Herleitung der Navier-Stokes Gleichung für inkompressible NEWTON- Fluide	10
2.1.2	Turbulenz und Filterungsoperatoren	13
2.1.3	Das Schließungsproblem der RANS-Gleichung	14
2.2	Turbulenzmodellierung mit dem k - ϵ Modell	15
2.2.1	Turbulente kinetische Energie und Dissipation	15
2.2.2	Das k - ϵ Turbulenzmodell	15
2.3	Populations-Bilanz Gleichungen (PBE)	16
2.3.1	Die Quadratur-Momenten-Methode (QMOM)	19
2.3.2	Modellierung der Quellterme nach Coulaloglou und Tavlarides	20
2.3.3	Sauter-Durchmesser und Varianz	24
2.4	Zusammenfassung	25
3	Anwendung auf einen zweidimensionalen Rührer	26
3.1	Gittergenerierung	26
3.1.1	Koordinatentransformation	27
3.1.2	Versetzte Gitter	28
3.2	Ortsdiskretisierung in Polarkoordinaten	29
3.2.1	Diskretisierung der RANS-Gleichung	29
3.2.2	Diskretisierung der skalaren Transportgleichungen	32
3.3	Einbau von Randbedingungen	34

3.3.1	Randbedingungen für die Geschwindigkeit	34
3.3.2	Wandfunktion des k - ϵ Modells	35
3.3.3	Randbedingungen für die Momente	36
3.4	Zeitintegration	37
3.4.1	Lösen der Druck-Poissongleichung (PPE)	37
3.4.2	Zeitintegration der skalaren Transportgleichungen	38
3.5	Realisierung in rotierenden Polarkoordinaten	39
3.6	Algorithmus	41
4	Numerische Tests	42
4.1	Simulationsergebnisse für $Re = 3.200$	42
4.2	Simulationsergebnisse für $Re = 12.800$	44
5	Zusammenfassung und Ausblick: Steuerung der Drehgeschwindigkeit ω	46
A	Mathematischer Anhang	48
A.1	Der Gauß'sche Integralsatz	48
A.2	Das Reynolds'sche Transporttheorem	48
B	Der PD-Algorithmus	49
C	Kleines Solver-Handbuch	50
C.1	inputfile.m	50
C.2	main.m	51

Die selbstständige und eigenhändige Anfertigung versichere ich an Eides statt.

Berlin, den

Notations- und Abkürzungstabelle

c_T	Turbulenter Diffusionskoeffizient
d_p	Tropfendurchmesser (maximaler Tropfendurchmesser: d_{max})
d_{32}	Sauter-Durchmesser
D	Durchmesser des Rührstabs
E_n	Einheitsmatrix der Dimension n
\vec{g}	Vektor der Erdbeschleunigung
k	Turbulente kinetische Energie
$m^{(l)}$	l -tes Moment der Anzahl-Dichtefunktion q , $l \in \mathbb{N}_0$
n	Anzahl der Tochtertropfen bei Zerfall ($n \equiv 2$ für binären Zerfall)
q	Anzahl-Dichtefunktion
Re	Reynolds-Zahl
T	Reynolds'scher Spannungstensor
\vec{u}	Geschwindigkeit $\vec{u} \in \mathbb{R}^d$, $d \in \{2, 3\}$
u_{tip}	Geschwindigkeit an der Spitze der Rührblätter
β	Wahrscheinlichkeit, dass bei Zerfall ein Tropfen festen Durchmessers entsteht
Γ	Rand des Diskretisierungsgebiets, also $\Gamma := \partial\Omega$
ϵ	Dissipationsrate
κ	Zerfallswahrscheinlichkeit eines Tropfens
μ	Dynamische Viskosität des Fluids
ν	Kinematische Viskosität des Fluids
ν_T	Turbulente Viskosität
ξ	Eigenschaftskordinate, $\xi \in \Omega_e$
ξ_α	Abszissen der Approximation des l -ten Moments
π	Druck des Fluids (p bezeichnet den um $\frac{1}{\rho_c}$ skalierten Druck)
ρ	Dichte des Fluids (Schreibe ρ_c im Falle konstanter Dichte)
σ	Varianz der Anzahl-Dichtefunktion q
ψ	Wahrscheinlichkeit für Koaleszenz zweier Tropfen
ω	Drehgeschwindigkeit des Rührers
ω_α	Gewichte der Approximation des l -ten Moments
Ω	Diskretisierungsgebiet $\Omega \subset \mathbb{R}^d$, $d \in \{2, 3\}$
Ω_e	Gebiet der Eigenschaftskordinaten, hier: $\Omega_e = [0, d_{max}]$
∇	Gradient-Operator
Δ	Laplace-Operator
div	Divergenz-Operator
\cdot	Skalarprodukt
CFD	Numerische Fluidodynamik
FDM	Finite-Differenzen Methode
NSE	Navier-Stokes Gleichung
PBE	Populations-Bilanz Gleichung
PDE	Partielle Differentialgleichung
PPE	Druck-Poisson Gleichung
QMOM	Quadratur-Momenten Methode
RANS	Reynolds-Averaged-Navier-Stokes Gleichung
RB	Randbedingung

Abstract

In the field of Computational Fluid Dynamics (CFD) the Navier-Stokes equation is used to describe the flow behavior of fluids and gases. For the modeling of turbulent flows the Navier-Stokes equation is extended to the Reynolds-Averaged-Navier-Stokes equation (RANS) by applying a filter operator. Since this leads to an unclosed system of partial differential equations, the k - ϵ turbulence model is used to approximate the so-called Reynolds stress tensor and to close the system of equations.

The main interest of this work is the modeling of a second, immiscible dispersed phase which is surrounded by the fluid. Therefore, a population balance equation (PBE) is derived to describe the drop distribution of the second fluid during the mixing process in a stirrer. The PBE is solved using the Quadrature Method of Moments (QMOM) whereas the coalescence and breakage of drops during the mixing process are modeled via integral terms which appear as source terms of the PBE.

In the second part of this work, the derived equations are used to simulate the flow behavior during the mixing process in a two-dimensional stirrer. For this, the partial differential equations are formulated in cylindrical coordinates and discretized in space using the Method of Finite Differences. Furthermore, the instationary part of the equations is discretized with the explicit EULER Method. To complete the modeling of the stirrer, boundary conditions for the velocity \vec{u} and the k - ϵ model are presented and the rotating of the stirring staff is realized by rotating coordinates.

The modeling and the implementation are evaluated by numerical tests in the last chapter.

1 Einleitung

1.1 Problemstellung und Motivation

In vielen technischen Anwendungen wie beispielsweise der chemischen und pharmazeutischen Industrie aber auch in der Ölindustrie spielt das Mischen zweier nicht-mischbarer Flüssigkeiten in einem Rührer eine zentrale Rolle. Da experimentelle Untersuchungen sehr teuer sein können und auch nur begrenzt Einblick in die Strömung gewähren, ist man daran interessiert, das Strömungsverhalten der Fluide numerisch zu simulieren. Des Weiteren erhofft man sich durch die numerischen Ergebnisse den Prozess des Vermischens zu optimieren, in dem Sinne, dass durch Vorgabe der Drehgeschwindigkeit des Rührstabs ein gewünschter Zustand im Rührer möglichst schnell erreicht wird (Zustandseinstellung).

Nimmt man an, dass eines der Fluide in einem deutlich niedrigeren Anteil vorliegt, so kann das Gemisch in dem Rührer in eine *disperse* Phase und eine *kontinuierliche* Phase unterteilt werden. Die disperse Phase (Dispersion) ist hierbei das niedriger konzentrierte Fluid. Sie ist deutlich von der kontinuierlichen Phase abgetrennt und liegt in Tropfen unterschiedlicher Größe innerhalb der umgebenen kontinuierlichen Phase vor (siehe Abbildung 1). Es wird angenommen, dass sie das Strömungsverhalten des Gemischs im Wesentlichen nicht beeinflusst.

Diese Bachelorarbeit befasst sich mit der Modellierung und Simulation von Dispersionen in turbulenten Strömungen. Die zur Beschreibung des Strömungsverhaltens innerhalb des Rührers hergeleiteten Gleichungen werden im zweiten Teil dieser Arbeit für die numerische Simulation eines vereinfachten zweidimensionalen Rührermodells verwendet.

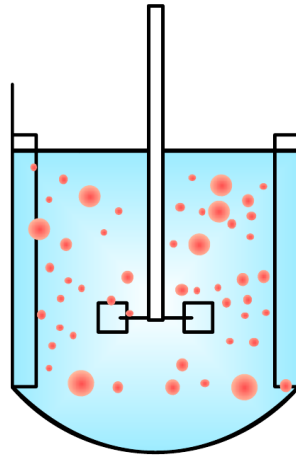


Abbildung 1: Modellierung und Simulation von Mischvorgängen in einem Rührer

Zur Beschreibung des Strömungsverhaltens eines Fluids wird zunächst die Navier-Stokes Gleichung hergeleitet und für turbulente Strömungen zur so genannten Reynolds-Averaged-Navier-Stokes Gleichung erweitert. Um dieses System abzuschließen, werden zwei weitere Transportgleichungen für die turbulente kinetische Energie k sowie die Dissipation ϵ eingeführt (k - ϵ Turbulenzmodell).

Die Beschreibung der zweiten, unlöslichen Flüssigkeit erfolgt über so genannte Populations-Bilanz Gleichungen (PBE), die die Verteilung der Tropfengröße innerhalb des umgebenen Fluids beschreiben. Hierzu wird eine allgemeine PBE zunächst hergeleitet und anschließend die Modellierung des Tropfenzerfalls und -entstehens während des Rührvorgangs auf der rechten Seite der PBE als Anregung berücksichtigt. Dies geschieht basierend auf experimentellen Untersuchungen von Coulaloglou und Tavlarides [3].

Für die numerische Simulation werden die partiellen Differentialgleichungen im Ort mit der Finiten-Differenzen Methode und in der Zeit mit der expliziten EULER-Methode diskretisiert. Hierfür werden die Gleichungen zunächst in Zylinderkoordinaten formuliert. Außerdem werden numerische Tests für unterschiedliche Reynolds-Zahlen zur Validierung durchgeführt.

1.2 Einordnung der Arbeit in den wissenschaftlichen Kontext

Die Modellierung turbulenter Strömungen durch die RANS-Gleichung gekoppelt mit dem k - ϵ Modell gehört zu den Standardverfahren in der Numerischen Fluidodynamik (CFD). Das sogenannte Standard k - ϵ Modell wird dabei aus [43] übernommen und ist für große Reynolds-Zahlen gültig. Das k - ϵ Modell gehört zu den gängigen Turbulenzmodellen. Für die numerische Simulation ist die Verwendung einer Finiten-Differenzen Diskretisierung, sofern es die geometrischen Anforderungen zulassen, ein üblicher Ansatz, insbesondere weil Aussagen über die Konvergenzordnung des Verfahrens bekannt sind (siehe beispielsweise [20, Kap. 3]). Die Anwendung von CFD-Techniken zur Simulation von Rührvorgängen wurde bereits in den Arbeiten [9] und [35] beschrieben. Als wesentliche Schwierigkeit gestaltet sich hierbei die Realisierung der anspruchsvollen Geometrie des Rührers, wie beispielsweise die Arbeiten [9] und [32] zeigen. Als Standardwerk im Bereich der *CFD in der Verfahrenstechnik* gilt unter anderem das Buch von A. Paschedag [28].

Die Modellierung einer zweiten, dispersen Phase innerhalb der Strömung kann beispielsweise durch die in [41] vorgestellten Mehrphasenmodelle realisiert werden. Hierbei werden die Gleichungen, die die Strömung beschreiben, für jede Phase einzeln aufgestellt. Ein zweiter Ansatz zur Beschreibung der dispersen Phase ist das Aufstellen einer Populations-Bilanz Gleichung (PBE). Diese werden ausführlich in [7] und [29] behandelt und verfolgen den Ansatz, die disperse Phase durch die Verteilung ihrer Tropfen zu beschreiben.

Für die numerische Lösung der PBE gibt es verschiedene Methoden, die beispielsweise in [4] vorgestellt werden. Die unterschiedlichen Methoden können in die *Klassenmethoden (MOC)* [16] und die *Momentenmethode (MOM)* [11] eingeteilt werden. Die hier verwendete Quadratur-Momenten Methode (QMOM) ist eine Variante der MOM und wird in [25] ausführlich beschrieben. Eine weitere Variante der MOM wurde beispielsweise vor kurzem in den Open Source Code OPENFOAM [36] implementiert. Allerdings steht in OPENFOAM kein Netzgenerierer zur einfachen Diskretisierung von Rührergeometrien zur Verfügung.

Gegenstand aktueller Forschung ist die mathematische Modellierung sowie das genaue Verständnis von Tropfenbruch und -koaleszenz [23]. Diese beiden gegenläufigen Prozesse werden in den Quelltermen der PBE berücksichtigt und beeinflussen maßgeblich das Simulationsverhalten. Zur Beschreibung von Zerfall und Koaleszenz von Tropfen wurde eine Vielzahl von

Modellen entwickelt, die in [18] und [19] zusammengefasst sind.

Zur Validierung dieser Modelle werden in der Regel verfahrenstechnische Untersuchungen in einem Zerfallskanal herangezogen. Inwiefern diese Ergebnisse auf einen Rührbehälter übertragbar sind, haben beispielsweise experimentelle Untersuchungen von [22] gezeigt.

1.3 Gliederung

In dieser Bachelorarbeit wird in **Kapitel 2** zunächst die Navier-Stokes Gleichung (NSE) aus der Massen- und Impulsbilanz für inkompressible NEWTON-Fluide hergeleitet. Um turbulente Strömungen zu modellieren, werden zusätzlich Filterungsoperatoren auf die NSE angewandt, was auf die Reynolds-Averaged-Navier-Stokes Gleichung (RANS) führt. In der RANS Gleichung tritt als zusätzlicher Term der so genannte Reynolds'sche Spannungstensor auf, der mit Hilfe des k - ϵ Turbulenzmodells approximiert wird.

Das Hauptinteresse dieser Arbeit liegt in der Modellierung und Simulation von Populations-Bilanz Gleichungen (PBE), die das Verhalten einer zweiten, dispersen Phase innerhalb des Fluids beschreiben. Die hergeleitete PBE wird mit Hilfe der Quadratur-Momenten Methode (QMOM) in ihren Momenten gelöst und zur Modellierung der Quell- bzw. Senkterme wird ein Tropfenmodell nach Coualoglou und Tavlarides verwendet.

Die Modellierung führt auf ein System gekoppelter partieller Differentialgleichungen.

In **Kapitel 3** der Arbeit werden die hergeleiteten Gleichungen zur Simulation von Mischvorgängen in einem zweidimensionalen Rührer angewendet. Hierfür werden die Gleichungen zunächst in Polarkoordinaten transformiert und im Ort mit der Finiten-Differenzen Methode diskretisiert sowie anschließend in der Zeit mit der EULER-Methode integriert. Darüber hinaus werden spezielle Randbedingungen für den Rührer diskutiert und eine algorithmische Umsetzung der Drehbewegung des Rührstabs präsentiert.

Zur Validierung der Modellierung und der numerischen Methoden werden in **Kapitel 4** numerische Tests für unterschiedliche Reynolds-Zahlen durchgeführt.

Abschließend wird ein kurzer Ausblick auf die Nutzung des entstandenen Strömungslösers zur Kontrolle einer gewünschten Tropfenverteilung durch Steuerung der Drehgeschwindigkeit ω gegeben.

2 Mathematische Modellierung

Die folgende Modellierung von Dispersionen in turbulenter Strömung beschränkt sich auf Gebiete $\Omega \subset \mathbb{R}^d$ mit $d = 2$ oder $d = 3$, an die hinreichende Glattheit gefordert wird, so dass der Integralsatz von Gauß anwendbar ist (siehe Anhang A.1). Darüber hinaus soll eine instationäre Simulation der Strömung modelliert werden, so dass außerdem ein Zeitintervall $T = [0, T_{end}]$ eingeführt wird. Die betrachteten physikalischen Größen sind dann auf dem Produktraum $\Omega \times [0, T_{end}]$ definiert.

Um die Strömung innerhalb des Gebiets Ω zu beschreiben, sei die Geschwindigkeit des Fluids $\vec{u} : \Omega \times T \rightarrow \mathbb{R}^d$ sowie der Druck $\pi : \Omega \times T \rightarrow \mathbb{R}$ und die Dichte $\rho : \Omega \times T \rightarrow \mathbb{R}$ des Fluids in Analogie zu [2, Kap. 1] eingeführt. Da in dieser Anwendung von inkompressiblen Fluiden ausgegangen wird, wird im Folgenden die Annahme $\rho(\vec{x}, t) = \text{const} = \rho_c$ getroffen. Es wird darüber hinaus die Annahme getroffen, dass es sich bei dem simulierten Fluid um ein NEWTON-Fluid handelt, also um eine Flüssigkeit, deren Scherspannung proportional zur Schergeschwindigkeit ist (Definition nach [31, S. 20]).

Zur Beschreibung inkompressibler Strömungen wird in diesem Kapitel zunächst die Navier-Stokes Gleichung nach [2] hergeleitet und diese für turbulente Strömungen in die sogenannte Reynolds-Averaged-Navier-Stokes Gleichung (RANS) überführt. Das System der RANS-Gleichungen ist nicht abgeschlossen, weshalb das in [43] vorgestellte k - ϵ Turbulenzmodell verwendet wird. Das Hauptinteresse dieser Arbeit ist die Modellierung der Tropfenverteilungen, wenn man eine zweite, nicht-mischbare Flüssigkeit im Gebiet Ω annimmt. Hierfür wird eine Populations-Bilanz Gleichung (PBE) für die Anzahl-Dichtefunktion q aufgestellt und in ihren Momenten gelöst. Dieses Vorgehen basiert auf der Arbeit von D. Marchisio [24] und wurde unter anderem bereits in [32] verwendet.

2.1 Die Reynolds-Averaged-Navier-Stokes Gleichung (RANS)

Die Navier-Stokes Gleichung (NSE) ist ein System gekoppelter, partieller Differentialgleichungen (PDE) und beschreibt das Strömungsverhalten von Flüssigkeiten und Gasen. Zur Herleitung der NSE wird im Folgenden die Annahme der Massenerhaltung verwendet sowie die Impulsbilanz nach dem zweiten NEWTON'schen Gesetz aufgestellt. Zur Modellierung von Turbulenz werden im Weiteren Filterungsoperatoren eingeführt und auf die NSE angewendet. Dies führt auf die RANS-Gleichung. Die Herleitung der NSE über Massen- und Impulsbilanz orientiert sich an verschiedenen Kapiteln aus [2, 33, 34], eine Einführung in Turbulenzmodellierung ist beispielsweise in [8, Kap. 10] gegeben.

2.1.1 Herleitung der Navier-Stokes Gleichung für inkompressible NEWTON-Fluide

Die Masse M eines Fluids in einem Gebiet Ω ist definiert über das Integral der Dichte des Fluids, also $M := \int_{\Omega} \rho(\vec{x}, t) d\vec{x}$. Wie in Abbildung 2 angedeutet, bedeutet **Massenerhaltung** im Gebiet Ω , dass die zeitliche Änderung von M gleich der Bilanz aus herein- und hinausfließendem Fluid ist, sofern angenommen wird, dass innerhalb des Gebiets keine Quellen und

Senken existieren:

$$\frac{d}{dt} \int_{\Omega} \rho(\vec{x}, t) d\vec{x} = - \int_{\partial\Omega} \rho(\vec{x}, t) \vec{u}(\vec{x}, t) \cdot \vec{n} ds,$$

wobei \vec{n} der nach außen zeigende Normalenvektor auf $\partial\Omega$ ist, so dass die rechte Seite der obigen Gleichung bilanziert, wieviel Masse über den Rand des betrachteten Gebiets fließt.

Nach dem Integralsatz von Gauß [15] ist dies äquivalent zu:

$$\int_{\Omega} \frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \vec{u}) d\vec{x} = 0.$$

Da diese Gleichheit für beliebige Gebiete Ω gilt, kann folgende Gleichung gefolgert werden:

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \vec{u}) = 0,$$

die als differentielle Form der Massenerhaltung bzw. **Kontinuitätsgleichung** bekannt ist, siehe [2, S. 3].

Unter der Annahme, dass das Fluid inkompressibel ist, $\rho(\vec{x}, t) = \rho_c$, folgt die bekannte Gleichung der **Divergenzfreiheit** der Strömung:

$$\operatorname{div} \vec{u} = 0. \quad (1)$$

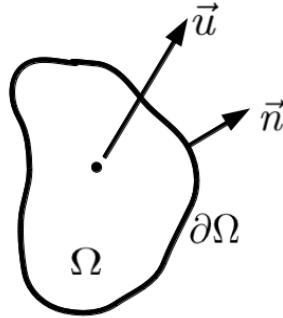


Abbildung 2: Bilanzierung des Massenstroms über den Rand des Gebiets Ω nach [2]

Die Herleitung der NSE erfolgt über die **Impulsbilanz**, wobei die aus der Kontinuitätsgleichung folgende Divergenzfreiheit der Geschwindigkeit verwendet wird.

Der Impuls I eines starren Körpers ist definiert als Produkt aus Masse und Geschwindigkeit. Im Falle eines Fluids sind diese beiden Größen zeit- und ortsabhängig und man definiert den Impuls daher als:

$$I(t) := \int_{\Omega} \rho(\vec{x}, t) \vec{u}(\vec{x}, t) d\vec{x}.$$

Nach dem zweiten NEWTON'schen Gesetz ist die Änderung des Impulses gleich der Summe aller angreifenden Kräfte, vgl. [2, S. 16]:

$$\frac{d}{dt} I(t) = \sum \text{angreifende Kräfte} =: F_{bf} + F_{sf}$$

Die auf das Fluid wirkenden Kräfte können dabei in zwei Arten aufgeteilt werden:

- **Volumenkräfte:**

In dieser Anwendung wird ausschließlich das Eigengewicht des Fluids als Volumenkraft angenommen, so dass $F_{bf} = \int_{\Omega} \rho(\vec{x}, t) \vec{g}(\vec{x}, t) d\vec{x}$ gilt, wobei \vec{g} der Vektor der Erdbeschleunigung im gewählten Koordinatensystem ist. Dieses wird in Kapitel 3 genauer spezifiziert.

- **Oberflächenkräfte:**

Als Oberflächenkräfte seien hier Druck und Reibung angenommen, die durch das Integral $F_{sf} = \int_{\partial\Omega} \sigma(\vec{x}, t) \vec{n} ds$ repräsentiert werden, wobei σ als Spannungstensor bezeichnet wird.

Der Spannungstensor σ bei einem NEWTON'schen Fluid ist gegeben durch:

$$\sigma := -\pi E_3 + \tau, \quad \tau = \lambda \operatorname{div} \vec{u} E_3 + 2\mu \delta, \quad (2)$$

wobei λ und μ Materialeigenschaften sind, die vom Ort abhängen können und π der zuvor eingeführte Druck des Fluids ist. In der obigen Formel wird außerdem der Verzerrungstensor $\delta := \frac{1}{2}(\nabla \vec{u} + (\nabla \vec{u})^T)$ verwendet, so dass tatsächlich Druck und Reibung modelliert werden.

Für die Impulsbilanz folgt somit:

$$\frac{d}{dt} \int_{\Omega} \rho(\vec{x}, t) \vec{u}(\vec{x}, t) d\vec{x} = \int_{\Omega} \rho(\vec{x}, t) \vec{g}(\vec{x}, t) d\vec{x} + \int_{\partial\Omega} \sigma(\vec{x}, t) \vec{n} ds.$$

Wendet man auf der linken Seite dieser Gleichung das Reynolds'sche Transporttheorem (siehe Anhang A.2) an und auf der rechten Seite erneut den Satz von Gauß (siehe Anhang A.1), so erhält man in differentieller Form:

$$\frac{\partial}{\partial t}(\rho \vec{u}) + (\vec{u} \cdot \nabla)(\rho \vec{u}) + \underbrace{(\rho \vec{u}) \operatorname{div} \vec{u}}_{=0 \text{ nach (1)}} - \rho \vec{g} - \operatorname{div} \sigma = 0.$$

Setzt man jetzt (2) ein, so ergibt sich:

$$\frac{\partial}{\partial t}(\rho \vec{u}) + (\vec{u} \cdot \nabla)(\rho \vec{u}) - \rho \vec{g} + \nabla \pi - \underbrace{\nabla \lambda \operatorname{div} \vec{u}}_{=0 \text{ nach (1)}} - \operatorname{div} \mu(\nabla \vec{u} + (\nabla \vec{u})^T) = 0.$$

Verwendet man wieder die Annahme, dass es sich um inkompressible Fluide handelt, so kann die obige Differentialgleichung durch ρ_c geteilt werden. Mit der formalen Skalierung $p := \frac{\pi}{\rho_c}$ ergibt sich insgesamt die NSE mit der Nebenbedingung, dass die Strömung divergenzfrei ist:

$$\frac{\partial}{\partial t} \vec{u} + (\vec{u} \cdot \nabla) \vec{u} + \nabla p - \operatorname{div} \nu(\nabla \vec{u} + (\nabla \vec{u})^T) - \vec{g} = 0, \quad (3)$$

$$\operatorname{div} \vec{u} = 0. \quad (4)$$

Hierbei wird außerdem die kinematische Viskosität des Fluids $\nu := \frac{\mu}{\rho_c}$ gemäß [31, S. 20] definiert, wobei zu beachten ist, dass diese hier als ortsabhängig angenommen wird.

2.1.2 Turbulenz und Filterungsoperatoren

Im Folgenden werden die zu modellierenden Strömungen als *turbulent* angenommen. Damit ist ein Strömungsverhalten gemeint, dass unvorhersehbar und chaotisch ist und zur Ausbildung von Wirbeln unterschiedlicher Größe führt. Um insbesondere die mikroskopische Struktur solcher Strömungen hinreichend genau aufzulösen, wäre lokal ein sehr feines Netz erforderlich, was in technischen Anwendungen die heutigen Rechenleistungen überschreitet. Nach Kolmogorov [14] kann für große Reynolds-Zahlen eine kleinste Längenskala für die Betrachtung von turbulenter Strömung angegeben werden.

Daher verwendet man in der Turbulenzmodellierung häufig den Ansatz, dass alle physikalischen Größen, die in der NSE auftreten, aus einem Mittelwert und einer kleinen Fluktuation zusammengesetzt sind:

$$\begin{aligned}\vec{u} &= \vec{U} + \vec{u}', \\ p &= P + p', \\ \vec{g} &= \vec{G} + \vec{g}'.\end{aligned}$$

Hierbei sind die groß geschriebenen Größen die Mittelwerte, die in den meisten Anwendungen von größerem Interesse sind als die Fluktuationen (gestrichen notiert), welche man oft vernachlässigt. Die Mittelwerte sind durch Anwenden eines zeitlichen Filters $\langle \cdot \rangle$ entstanden:

$$\vec{U} := \langle \vec{u} \rangle, \quad P := \langle p \rangle, \quad \vec{G} := \langle \vec{g} \rangle.$$

Der Filter $\langle \cdot \rangle$ bildet den zeitlichen Mittelwert einer Funktion $f : \mathbb{R}^d \times [0, T_{end}] \rightarrow \mathbb{R}$, in dem über ein Zeithorizont $\tau \in \mathbb{R}_{>0}$ integriert und gemittelt wird:

$$\langle f(\vec{x}, t) \rangle := \frac{1}{\tau} \int_{t-\frac{1}{2}\tau}^{t+\frac{1}{2}\tau} f(\vec{x}, s) ds.$$

Bei vektorwertigen Größen wird dieser Filter komponentenweise angewendet.

Der zuvor definierte Filterungsoperator hat für zwei physikalische Größen der Form $f_1, f_2 : \mathbb{R}^d \times [0, T_{end}] \rightarrow \mathbb{R}$ und Koeffizienten $\alpha \in \mathbb{R}$ die folgenden Eigenschaften:

1. $\langle \cdot \rangle$ ist linear, d.h. $\langle f_1 + f_2 \rangle = \langle f_1 \rangle + \langle f_2 \rangle$ und $\langle \alpha f_1 \rangle = \alpha \cdot \langle f_1 \rangle$;
2. Partielle Ableitungen kommutieren mit der Filterung: $\langle \partial f_1 / \partial t \rangle = \partial \langle f_1 \rangle / \partial t$ sowie $\langle \partial f_1 / \partial x^i \rangle = \partial \langle f_1 \rangle / \partial x^i$, $i \in \{1, 2, 3\}$;
3. Es gilt: $\langle \langle f_1 \rangle f_2 \rangle = \langle f_1 \rangle \langle f_2 \rangle$, aber im Allgemeinen $\langle f_1 f_2 \rangle \neq \langle f_1 \rangle \langle f_2 \rangle$;
4. Invarianz unter doppelter Filterung: $\langle \langle f_1 \rangle \rangle = \langle F_1 \rangle = F_1 = \langle f_1 \rangle$;
5. Die Fluktuationen werden herausgemittelt: $\langle f_1' \rangle = 0$.

Im Folgenden werden das System aus NSE (3) und Kontinuitätsgleichung (4) durch anwenden des Filterungsoperators in den Mittelwerten ausgedrückt und dabei wie in [8] die obigen Rechenregeln verwendet.

2.1.3 Das Schließungsproblem der RANS-Gleichung

Als erster Schritt soll der Filterungsoperator auf die Kontinuitätsgleichung (4) angewendet werden:

$$0 = \langle \operatorname{div} \vec{u} \rangle \stackrel{2.}{=} \operatorname{div} \langle \vec{u} \rangle \stackrel{\text{Def.}}{=} \operatorname{div} \vec{U}$$

Diese Rechnung zeigt, dass die Divergenzfreiheit auch für den Mittelwert der Geschwindigkeit gilt. Dabei werden die Rechenregeln aus Abschnitt 2.1.2 verwendet, wobei insbesondere die Vertauschung von partieller Ableitung und Filterung benutzt wird.

Um die NSE (3) zu filtern, wird diese zunächst in der folgenden Form geschrieben:

$$\frac{\partial}{\partial t} \vec{u} + \operatorname{div}(\vec{u} \otimes \vec{u}) + \nabla p - \operatorname{div} \nu(\nabla \vec{u} + (\nabla \vec{u})^T) - \vec{g} = 0, \quad (3a)$$

wobei für den Konvektionsterm auf Grund der Divergenzfreiheit von \vec{u} die Beziehung $(\vec{u} \cdot \nabla) \vec{u} = \operatorname{div}(\vec{u} \otimes \vec{u})$ gilt. Hierbei ist $(\vec{u} \otimes \vec{u})$ als Tensor aufzufassen, der komponentenweise via $[\vec{u} \otimes \vec{u}]_{i,j=1}^3 = u^i u^j$ definiert ist. Die hochgestellten Indizes bezeichnen dabei die Komponenten des Vektors $\vec{u} = [u^1, u^2, u^3]^T$.

Wendet man die Filterung auf Gleichung (3a) an, so kann in einem ersten Schritt sofort die Vertauschung von partieller Differentiation und Mittelung verwendet werden und man erhält:

$$\frac{\partial}{\partial t} \langle \vec{u} \rangle + \operatorname{div} \langle \vec{u} \otimes \vec{u} \rangle + \nabla \langle p \rangle - \operatorname{div} \nu(\nabla \langle \vec{u} \rangle + (\nabla \langle \vec{u} \rangle)^T) - \langle \vec{g} \rangle = 0.$$

Auf Grund von Eigenschaft 3 und 5 des Filterungsoperators gilt zudem:

$$\begin{aligned} \langle \vec{u} \otimes \vec{u} \rangle &= \langle (\vec{U} + \vec{u}') \otimes (\vec{U} + \vec{u}') \rangle \\ &= \langle \vec{U} \otimes \vec{U} \rangle + \langle \vec{U} \otimes \vec{u}' \rangle + \langle \vec{u}' \otimes \vec{U} \rangle + \langle \vec{u}' \otimes \vec{u}' \rangle \\ &= \vec{U} \otimes \vec{U} + \langle \vec{u}' \otimes \vec{u}' \rangle. \end{aligned}$$

Somit erhält man insgesamt die so genannte Reynolds-Averaged-Navier-Stokes Gleichung (RANS)

$$\frac{\partial}{\partial t} \vec{U} + \operatorname{div}(\vec{U} \otimes \vec{U}) + \operatorname{div} \overbrace{\langle \vec{u}' \otimes \vec{u}' \rangle}^{=: -T(\vec{u}')} + \nabla P - \operatorname{div} \nu(\nabla \vec{U} + (\nabla \vec{U})^T) - \vec{G} = 0 \quad (5)$$

$$\operatorname{div} \vec{U} = 0 \quad (6)$$

mit dem Reynolds-Spannungstensor $T(\vec{u}') := -\langle \vec{u}' \otimes \vec{u}' \rangle$. Da das Produkt zweier Fluktuationen eine weitere Unbekannte ist, besitzt das System aus (5) und (6) mehr Unbekannte als Gleichungen. Dieses Problem bezeichnet man als Schließungsproblem der RANS-Gleichung. Die Approximation des Terms $T(\vec{u}')$ ist die wesentliche Aufgabe der Turbulenzmodellierung.

2.2 Turbulenzmodellierung mit dem k - ϵ Modell

Neben einer Approximation des Reynolds-Spannungstensors T werden im k - ϵ Modell außerdem die Größen *turbulente kinetische Energie* k und *Dissipationsrate* ϵ eingeführt, sowie zwei weitere Transportgleichungen für diese Größen entwickelt. Die Approximation von T ist dann eine Funktion der mittleren Geschwindigkeit und der neu eingeführten Größen k und ϵ :

$$-\langle \vec{u}' \otimes \vec{u}' \rangle \approx T(\vec{U}, k, \epsilon)$$

2.2.1 Turbulente kinetische Energie und Dissipation

An dieser Stelle sollen die beiden skalaren Größen turbulente kinetische Energie k und Dissipation ϵ eingeführt werden. Diese werden beispielsweise in [27] und [43] eingeführt durch:

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

wobei $\|\cdot\|_F$ die Frobenius Norm eines Vektors bzw. einer Matrix ist. Beide Größen sind, wie die Geschwindigkeit \vec{u} , auf dem Produktraum $\Omega \times [0, T_{end}]$ definiert und werden im Folgenden als zwei weitere Unbekannte behandelt.

2.2.2 Das k - ϵ Turbulenzmodell

Im k - ϵ Modell wird der Reynolds-Spannungstensor approximiert durch:

$$T(\vec{u}') \approx -\frac{2}{3} k E_3 + c_\mu \frac{k^2}{\epsilon} (\nabla \vec{U} + \nabla \vec{U}^T),$$

wobei mit $c_\mu \frac{k^2}{\epsilon} =: \nu_T$ die turbulente Viskosität als Proportionalitätsfaktor zwischen dem Reynolds'schen Spannungstensor $\langle \vec{u}' \otimes \vec{u}' \rangle$ und dem symmetrischen Tensor $\tilde{T} := \nabla \vec{U} + \nabla \vec{U}^T$ bestimmt wird. Hierbei geht auch die empirische Erfahrung ein, dass Turbulenz häufig in Regionen großer Geschwindigkeitsgradienten auftritt (vgl. [8, 43]).

Definiert man, der in der Literatur üblichen Notation folgend, $\nu^* := \nu + \nu_T$, so erweitert sich die RANS-Gleichung (5) zu

$$\frac{\partial}{\partial t} \vec{U} + \text{div}(\vec{U} \otimes \vec{U}) + \nabla P + \frac{2}{3} \nabla k - \text{div}(\nu^* (\nabla \vec{U} + \nabla \vec{U}^T)) - \vec{G} = 0 \quad (7)$$

und beinhaltet somit eine Unbekannte weniger.

Darüber hinaus werden für das k - ϵ Modell in [43] zwei Transportgleichungen für k und ϵ entwickelt:

$$\frac{\partial}{\partial t} k + \vec{U} \cdot \nabla k - \frac{\nu_T}{2} \|\nabla \vec{U} + \nabla \vec{U}^T\|_F^2 - \text{div}(\nu_T \nabla k) + \epsilon = 0, \quad (8)$$

$$\frac{\partial}{\partial t} \epsilon + \vec{U} \cdot \nabla \epsilon - \frac{c_1}{2} k \|\nabla \vec{U} + \nabla \vec{U}^T\|_F^2 - \text{div}\left(\frac{c_\epsilon}{c_\mu} \nu_T \nabla \epsilon\right) + c_2 \frac{\epsilon^2}{k} = 0. \quad (9)$$

Hierbei tritt in beiden Gleichungen die quadratische Frobenius Norm des Tensors \tilde{T} auf, die im Weiteren mit \tilde{T}_F bezeichnet werden soll, also $\tilde{T}_F := \|\tilde{T}\|_F^2$.

Die Wahl der Parameter in den beiden Gleichungen beruht auf experimentellen Untersuchungen und kann in [17] nachgeschlagen werden:

$$c_\mu = 0.09, \quad c_\epsilon = 0.07, \quad c_1 = 1.126, \quad c_2 = 1.92.$$

2.3 Populations-Bilanz Gleichungen (PBE)

In diesem Abschnitt soll die eigentliche Modellierung der dispersen Phase geschehen. Dazu wird angenommen, dass in dem Simulationsgebiet Ω ein Gemisch aus zwei Phasen vorliegt, wobei eine Phase die andere vollständig umgibt (wie beispielsweise bei Wasser und Öl). Nimmt man darüber hinaus an, dass die umgebende Flüssigkeit in deutlich höherer Konzentration auftritt und beide Fluide ähnliche physikalische Eigenschaften besitzen (Dichte, Viskosität), so ist die Annahme gerechtfertigt, dass die Strömung weiterhin von einer Flüssigkeit bestimmt wird und die disperse Phase durch die Verteilung ihrer Tropfen beschrieben werden kann.

Zur mathematischen Beschreibung der dispersen Phase wird die so genannte *Anzahl-Dichtefunktion* $q : \Omega_e \times \Omega \times [0, T_{end}] \rightarrow \mathbb{R}$ eingeführt. Sie beschreibt die Anzahl der Individuen (Tropfen), die sich zu einem bestimmten Zeitpunkt $t \in [0, T_{end}]$ in einem Teilraum des Partikelzustandsraums $\Omega \times \Omega_e$ befinden. Dabei ist Ω_e der Raum der so genannten *Eigenschaftskordinaten*, der für spezielle Charakteristiken der dispersen Phase steht. Für die in Kapitel 3 diskutierte Anwendung ist es ausreichend, den Tropfendurchmesser als einzige weitere Eigenschaft der Partikelphase anzunehmen, so dass $\Omega_e = [0, d_{max}]$ gilt.

Die Gesamttropfenanzahl N_{drops} ist dann gegeben durch Integration über das Simulationsgebiet Ω sowie den Eigenschaftsraum Ω_e :

$$N_{drops}(t) = \int_{\Omega} \int_{\Omega_e} q(\xi, \vec{x}, t) \, d\xi \, d\vec{x},$$

wobei $\xi \in \Omega_e$ und $\vec{x} \in \Omega$.

Gesucht ist im Folgenden also eine Gleichung für die zeitliche Änderung der Tropfenanzahl

$$\frac{dN_{drops}(t)}{dt} = \frac{d}{dt} \left(\int_{\Omega} \int_{\Omega_e} q(\xi, \vec{x}, t) \, d\xi \, d\vec{x} \right).$$

Die nun folgende Herleitung einer allgemeinen Populations-Bilanz Gleichung (PBE) orientiert sich an den Ausführungen in [7]. Dazu wird zunächst angenommen, dass die Anzahl-Dichtefunktion q die *mittlere* Anzahl von Partikeln angibt.

In Analogie zur Modellierung der Massen- und Impulsbilanzgleichung, wird für die Änderung der Tropfenanzahl in einem betrachteten Kontrollvolumen $V \times V_e \subseteq \Omega \times \Omega_e$ von der Beziehung

$$\frac{dN_{drops}}{dt} = \Phi_N + S_N$$

ausgegangen, in der die Terme auf der rechten Seite für die zeitliche Änderung der Tropfenanzahl stehen. Dabei ist Φ_N ein Transportterm, der die Anzahl der Individuen beschreibt, die

pro Zeiteinheit über den Rand $A \times A_e$ des Kontrollvolumens zu- oder abgeführt werden. Der Quellterm S_N beschreibt die pro Zeiteinheit im Inneren des Kontrollvolumens erzeugte bzw. vernichtete Anzahl an Individuen.

Der Quellterm S_N kann durch ein Volumenintegral substituiert werden

$$S_N = \int_V \int_{V_e} s(\xi, \vec{x}, t) d\xi d\vec{x},$$

wobei s die Quelldichte bezogen auf ein Volumenelement des Partikelzustandsraums ist.

Der Transportterm Φ_N wird analog durch ein Oberflächenintegral ersetzt

$$\Phi_N = - \int_A \int_{A_e} \vec{n} \cdot \phi_N(\xi, \vec{x}, t) d\xi d\vec{x}.$$

Hierbei beschreibt der Term $-(\vec{n} \cdot \phi_N)$ die Komponente des Transportstroms bezogen auf ein Oberflächenelement des Partikelzustandsraums in Normalenrichtung. Das Minuszeichen ergibt sich auf Grund der Orientierung des Normalenvektors.

Für die Bilanzgleichung folgt mit dem Satz von Gauß [15]:

$$\frac{d}{dt} \int_V \int_{V_e} q(\xi, \vec{x}, t) d\xi d\vec{x} = - \int_V \int_{V_e} \nabla_{e,\vec{x}} \cdot \phi_N(\xi, \vec{x}, t) d\xi d\vec{x} + \int_V \int_{V_e} s(\xi, \vec{x}, t) d\xi d\vec{x},$$

wobei $\nabla_{e,\vec{x}} := \nabla_e + \nabla_{\vec{x}}$ die Summe der Gradienten bezogen auf die Orts- und Eigenschaftskordinaten ist.

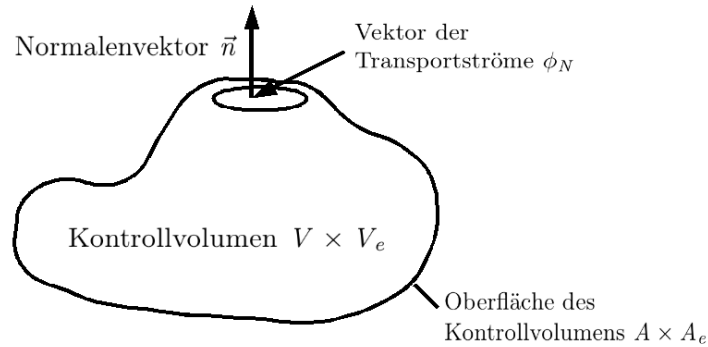


Abbildung 3: Kontrollvolumen $V \times V_e$ mit Transportstrom ϕ_N nach [7]

Da diese Überlegungen für beliebige Kontrollvolumen gelten, folgt für die **differentielle Form einer Populations-Bilanz Gleichung**:

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

Im Folgenden trifft man die Annahme, dass die Komponenten des Transport-Flusses ϕ_N in einen Konvektions- und Diffusionsterm aufgeteilt werden können:

$$\begin{aligned}\phi_{N,\vec{x}} &= w_x \cdot q(\xi, \vec{x}, t) + \phi_{N,\vec{x}}^D, \\ \phi_{N,e} &= w_e \cdot q(\xi, \vec{x}, t) + \phi_{N,e}^D,\end{aligned}$$

wobei w_x und w_e die Geschwindigkeiten der Partikel sind, so dass der Term $w_x \cdot q(\xi, \vec{x}, t)$ die deterministische Bewegung der Partikel beschreibt. Wie einleitend erwähnt, wird bei einer niedrigen Konzentration der dispersen Phase davon ausgegangen, dass sich die Partikel mit dem umgebenden Fluid *mitbewegen* und somit $w_x = \vec{u}$ angenommen werden kann. Die übrigen Terme werden in dieser Anwendung vernachlässigt. Sie werden in allgemeineren Ansätzen verwendet, um stochastische Bewegungen der Tropfen sowie die Produktion von Partikeln mit der Zeit durch beispielsweise chemische Reaktionen zu modellieren.

Die PBE vereinfacht sich somit zu folgender Gleichung:

$$\frac{\partial q}{\partial t} + \text{div}(\vec{u}q) = s. \quad (10)$$

Im Weiteren soll diskutiert werden, wie man Turbulenz in der Populations-Bilanz Gleichung (10) berücksichtigen kann. Dazu wird analog zur Herleitung der RANS-Gleichung der Ansatz gemacht, dass sich die Anzahl-Dichtefunktion q aus einem Mittelwert Q und kleinen Fluktuationen q' zusammensetzt. Mit dem in Abschnitt 2.1.2 definierten Filterungsoperator $\langle \cdot \rangle$ ergibt sich analog:

$$q = Q + q', \quad \text{mit } Q := \langle q \rangle.$$

Setzt man diesen Ansatz in die Gleichung (10) ein, so erhält man:

$$\frac{\partial(Q + q')}{\partial t} = -\text{div}((\vec{U} + \vec{u}') \cdot (Q + q')) + s.$$

Mit den Rechenregeln aus Kapitel 2.1.2 ist dies äquivalent zu:

$$\frac{\partial Q}{\partial t} = -\text{div}(\vec{U} \cdot Q) - \text{div}\langle \vec{u}' \cdot q' \rangle + S,$$

wobei $S := \langle s \rangle$ der gemittelte Quellterm ist.

Ähnlich zu den Ansätzen zur Approximation des Reynolds'schen Spannungstensors, wird in [28, Kap. 3.2.1.2] eine Approximation für den Term $\text{div}\langle u' \cdot q' \rangle$ entwickelt:

$$\text{div}\langle u' \cdot q' \rangle \approx -\text{div}(c_T \nabla Q).$$

Dabei ist c_T der sogenannte turbulente Diffusionskoeffizient, den man auch durch die Schmidt-Zahl $Sc_T = \frac{\mu_T}{\rho c_T}$ charakterisieren kann. Unter der Annahme isotroper Turbulenz sowie der Annahme, dass der turbulente Impulsaustausch gleich dem turbulenten Massenaustausch ist, kann gefolgert werden, dass die Schmidt-Zahl ungefähr 1 ist (siehe [28, Kap. 2.4.4.1]). Die turbulenten Transportkoeffizienten μ_T und c_T haben somit die gleiche Größenordnung und wegen der Beziehung $\nu_T = \frac{\mu_T}{\rho}$, folgt insgesamt $c_T \approx \nu_T$. Für die Populations-Bilanz Gleichung ergibt sich somit:

$$\frac{\partial Q}{\partial t} + \text{div}(\vec{U}Q) - \text{div}(\nu_T \nabla Q) = S. \quad (11)$$

2.3.1 Die Quadratur-Momenten-Methode (QMOM)

Da die Anzahl-Dichtefunktion q auf dem Produktraum $\Omega_e \times \Omega \times [0, T_{end}]$ definiert ist, würde die Diskretisierung dieses Raums zu hohem numerischen Aufwand führen. Daher soll q durch seine Momente charakterisiert werden:

$$m^{(l)}(\vec{x}, t) := \int_{-\infty}^{\infty} \xi^l \cdot q(\xi, \vec{x}, t) d\xi, \quad l \in \mathbb{N}_0,$$

wobei an dieser Stelle angenommen wird, dass Ω_e eindimensional ist. Wie bereits erwähnt, handelt es sich in der hier besprochenen Anwendung um den Tropfendurchmesser d_p , weshalb es ausreicht von 0 bis d_{max} zu integrieren.

Um Gleichung (11) in den Momenten auszudrücken, wird diese mit ξ^l multipliziert und über $\Omega_e = [0, d_{max}]$ integriert. Dies führt auf:

$$\frac{\partial}{\partial t} m^{(l)} + \operatorname{div}(\vec{U} m^{(l)}) - \operatorname{div}(\nu_T \nabla m^{(l)}) = S^{(l)}, \quad l \in \mathbb{N}_0, \quad (12)$$

wobei für die rechte Seite $S^{(l)}(d_p) := \int_0^{d_{max}} d_p^l S(d_p) dd_p$ eingeführt wird.

Die mathematische Struktur von Gleichung (12) ist zu (11) identisch. Der Vorteil ist jedoch, dass die Momente $m^{(l)} : \Omega \times [0, T_{end}] \rightarrow \mathbb{R}$ einen niedriger dimensional Definitionsbereich haben, da im Vergleich zum Definitionsbereich der Anzahl-Dichtefunktion q das kartesische Produkt ohne den Raum der Eigenschaftskordinaten Ω_e gebildet wird.

An dieser Stelle existiert noch das Modellierungsproblem, dass die Funktion q , die in den Momenten und in den rechten Seiten $S^{(l)}$ vorkommt (vgl. Kapitel 2.3.2), im Allgemeinen unbekannt ist. Dazu soll im Folgenden die **Quadratur-Momenten Methode (QMOM)** vorgestellt werden, die dieses Problem durch numerische Quadratur löst. Die QMOM sowie der allgemeine Ansatz, die Anzahl-Dichtefunktion über ihre Momente zu charakterisieren, beruht auf der Arbeit von D. Marchisio [24].

Zunächst ist die Beobachtung wichtig, dass q nur in Integralen vorkommt und im Integranden mit Funktionen g multipliziert wird. Die unbekannte Größe hat also die Form:

$$\int_{\Omega_e} g(\xi) q(\xi, \vec{x}, t) d\xi.$$

Da q außerdem eine Dichte ist, ist die Funktion insbesondere nicht-negativ und es kann via $\nu_{\vec{x},t}(E) := \int_E q(\xi, \vec{x}, t) d\xi$ für messbare Mengen E ein neues Maß definiert werden. Dieses Maß hängt, wie q , von den Parametern \vec{x} und t ab und ist ein Maß auf dem Raum Ω_e . Das übrig bleibende Integral kann dann mit Hilfe numerischer Quadratur bezüglich $\nu_{\vec{x},t}$ approximiert werden :

$$\int_{\Omega_e} g(\xi) q(\xi, \vec{x}, t) d\xi = \int_{\Omega_e} g(\xi) d\nu_{\vec{x},t}(\xi) \approx \sum_{\alpha=1}^N \omega_{\alpha}(\vec{x}, t) g(\xi_{\alpha}(\vec{x}, t)), \quad (13)$$

wobei die so entstehenden Gewichte ω_{α} und Abszissen (bzw. Stützstellen) ξ_{α} die $2N$ neuen Unbekannten sind, die vom Ort \vec{x} und der Zeit t abhängen. Diese Formel wird später zur Approximation der rechten Seiten in (12) benutzt.

Die Formel (13) kann insbesondere auf die Momente angewendet werden, denn mit $g(\xi) = \xi^l$ folgt:

$$m^{(l)} \approx \sum_{\alpha=1}^N \omega_{\alpha} \xi_{\alpha}^l. \quad (14)$$

Das so entstandene System aus (12) und (14) ist für die Wahl $l = \{0, \dots, 2N - 1\}$ **abgeschlossen**. Mit Hilfe des in Anhang B vorgestellten Produkt-Differenzen (PD) Algorithmus' können in jedem Zeitschritt Gewichte ω_{α} und Abszissen ξ_{α} nach Vorgabe der Momente $m^{(l)}$ so berechnet werden, dass die ersten $2N$ Momente exakt berechnet werden. Diese Gewichte und Abszissen werden auch verwendet, um höhere Momente, die in den rechten Seiten auftreten, gemäß (14) zu approximieren.

2.3.2 Modellierung der Quellterme nach Coulaloglou und Tavlarides

An dieser Stelle sollen die Quellterme von Gleichung (12) modelliert und diskretisiert werden. Die Modellierung ist dabei durch das Tropfenmodell von Coulaloglou und Tavlarides [3] gegeben. Eine Übersicht über weitere Tropfenmodelle befindet sich zum Beispiel in [32].

Betrachtet man Tropfen eines festen Durchmessers d_p , so bestimmen vor allem Zerfall und Verschmelzen (Koaleszenz) die Verteilung der Tropfengröße. In der Bilanz erhält man einen Term der Form:

$$S^{(l)}(d_p) = S_{break}^{(l)}(d_p) + S_{coal}^{(l)}(d_p) - D_{break}^{(l)}(d_p) - D_{coal}^{(l)}(d_p), \quad (15)$$

wobei zum einen Tropfen des Durchmessers d_p entstehen können, durch Zerbrechen von größeren Tropfen (S_{break}) und durch Verschmelzen von kleineren Tropfen (S_{coal}), und zum anderen Tropfen des Durchmessers d_p verschwinden, in dem ein solcher Tropfen mit anderen Tropfen koalesziert (D_{coal}) oder ein solcher Tropfen gerade zerbricht (D_{break}). Diese vier Möglichkeiten sind in Abbildung 4 schematisch dargestellt.

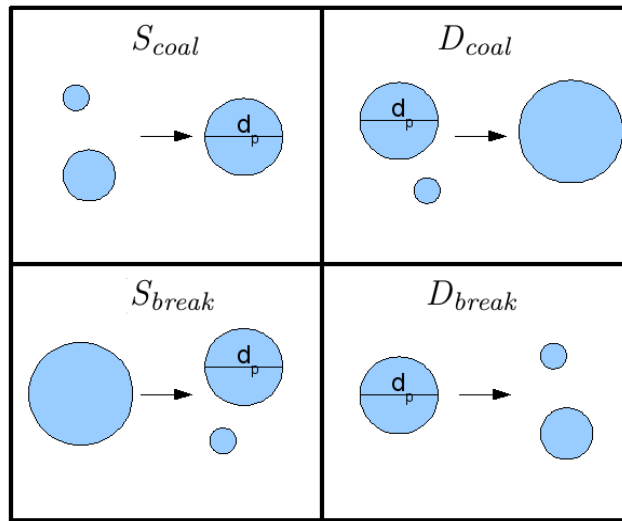


Abbildung 4: Bilanzierung von Tropfen mit festem Durchmesser d_p

Da man die Quellen an festem Ort und zu fester Zeit betrachtet, wird im Folgenden aus Gründen der Übersicht die Anzahl-Dichtefunktion q als Funktion einer Veränderlichen betrachtet $q = q(\cdot, \vec{x}, t)$.

Um die Entstehung von Tropfen des Durchmessers d_p durch Zerbrechen zu erhalten, müssen ausschließlich Tropfen mit größerem Durchmesser betrachtet werden. In [3] wird dieser Term beschrieben durch:

$$S_{break}(d_p) = \int_{d_p}^{d_{max}} n(d'_p) \beta(d_p, d'_p) \kappa(d'_p) q(d'_p) dd'_p,$$

wobei κ die Zerfallswahrscheinlichkeit eines Tropfens beschreibt, $\beta(d_p, d'_p)$ die Wahrscheinlichkeit, dass der Zerfall eines Tropfens des Durchmessers d'_p zu einem Tropfen mit Durchmesser d_p führt und n die Anzahl der entstehenden Tochtertropfen beschreibt. In dieser Anwendung wird mit $n \equiv 2$ stets binärer Zerfall angenommen. Die Integrationsgrenzen sind dabei so gewählt, dass nur Tropfen deren Durchmesser größer als d_p sind betrachtet werden und außerdem kleiner als ein maximal zugelassener Durchmesser d_{max} sind.

Analog erhält man für den Zerfall eines Tropfens des Durchmessers d_p :

$$D_{break}(d_p) = \kappa(d_p) q(d_p).$$

Um das Entstehen von Tropfen durch Verschmelzen zu modellieren, betrachtet man nur kleinere Tropfen:

$$S_{coal}(d_p) = \frac{1}{2} \int_0^{d_p} \psi(d'_p, d''_p) q(d'_p) q(d''_p) dd'_p,$$

wobei $d''_p := \sqrt[3]{d_p^3 - d'^3_p}$ und ψ die Wahrscheinlichkeit angibt, dass zwei Tropfen koaleszieren. Die Beziehung zwischen den Durchmessern ist bei kugelförmigen Tropfen äquivalent zu $Vol(d''_p) = Vol(d_p) - Vol(d'_p)$, so dass in S_{coal} genau zwei Tropfen koaleszieren, die vereinigt einen Tropfen des Durchmessers d_p ergeben. Der Faktor $1/2$ vor dem Integral kompensiert die doppelte Berücksichtigung beim Durchlaufen des Integrals.

Durch ψ kann man auch modellieren, dass ein Tropfen des Durchmessers d_p mit einem Tropfen des Durchmessers d'_p koalesziert und somit in der Bilanz (15) verschwindet:

$$D_{coal}(d_p) = q(d_p) \int_0^{\sqrt[3]{d_{max}^3 - d_p^3}} \psi(d_p, d'_p) q(d'_p) dd'_p$$

Hierbei wurde erneut die obere Integrationsgrenze so gewählt, dass keine Tropfen entstehen können, die größer als ein maximaler Durchmesser d_{max} sind.

Zur Implementierung sollen die einzelnen Quellterme noch approximiert werden.

Führe dazu die charakteristische Funktion χ ein

$$\chi(a > b) := \begin{cases} 1, & \text{wenn } a > b \\ 0, & \text{sonst} \end{cases}$$

und verwende im Folgenden die Rechenregel (13).

Somit folgt für die Entstehung von Tropfen durch Zerfall:

$$\begin{aligned}
S_{break}^{(l)} &= \int_0^{d_{max}} d_p^l \int_{d_p}^{d_{max}} n(d_p') \beta(d_p, d_p') \kappa(d_p') q(d_p') dd_p' dd_p \\
&= \int_0^{d_{max}} d_p^l \int_0^{d_{max}} \chi(d_p' > d_p) n(d_p') \beta(d_p, d_p') \kappa(d_p') q(d_p') dd_p' dd_p \\
&\approx \int_0^{d_{max}} d_p^l \left[\sum_{\alpha=1}^N \chi(\xi_\alpha > d_p) w_\alpha n(\xi_\alpha) \beta(d_p, \xi_\alpha) \kappa(\xi_\alpha) \right] dd_p \\
&\approx \sum_{\alpha=1}^N w_\alpha n(\xi_\alpha) \kappa(\xi_\alpha) \int_0^{\xi_\alpha} d_p^l \beta(d_p, \xi_\alpha) dd_p,
\end{aligned}$$

wobei das Integral in der letzten Zeile nach Spezifizierung von β numerisch approximiert werden kann (siehe hierzu beispielsweise [1, Kap. 6]).

Für das Verschwinden von Tropfen des Durchmesser d_p durch Zerfall in kleine Tropfen ergibt sich sofort:

$$D_{break}^{(l)} = \int_0^{d_{max}} d_p^l \kappa(d_p) q(d_p) dd_p \approx \sum_{\alpha=1}^N w_\alpha \xi_\alpha^l \kappa(\xi_\alpha).$$

Die Entstehung von Tropfen durch Koaleszenz kann approximiert werden durch:

$$\begin{aligned}
S_{coal}^{(l)} &= \int_0^{d_{max}} d_p^l \frac{1}{2} \int_0^{d_{max}} \chi(d_p > d_p') \psi(d_p', \sqrt[3]{d_p^3 - d_p'^3}) q(d_p') q(\sqrt[3]{d_p^3 - d_p'^3}) dd_p' dd_p \\
&\approx \frac{1}{2} \int_0^{d_{max}} d_p^l \left[\sum_{\alpha=1}^N \chi(d_p > \xi_\alpha) w_\alpha \psi(\xi_\alpha, \sqrt[3]{d_p^3 - \xi_\alpha^3}) q(\sqrt[3]{d_p^3 - \xi_\alpha^3}) \right] dd_p \\
&\approx \frac{1}{2} \sum_{\alpha=1}^N w_\alpha \int_0^{d_{max}} \chi(d_p > \xi_\alpha) d_p^l \psi(\xi_\alpha, \sqrt[3]{d_p^3 - \xi_\alpha^3}) \sum_{\gamma=1}^N w_\gamma \delta(\sqrt[3]{d_p^3 - \xi_\alpha^3} - \xi_\gamma^3) dd_p \\
&\stackrel{(*)}{=} \frac{1}{2} \sum_{\alpha=1}^N \sum_{\gamma=1}^N w_\alpha w_\gamma \frac{1}{|\xi_\gamma^{-2} (\xi_\gamma^3 + \xi_\alpha^3)^{2/3}|} \int_0^{d_{max}} \chi(d_p > \xi_\alpha) d_p^l \psi(\xi_\alpha, \sqrt[3]{d_p^3 - \xi_\alpha^3}) \delta(d_p - \sqrt[3]{\xi_\gamma^3 + \xi_\alpha^3}) dd_p \\
&= \frac{1}{2} \sum_{\alpha=1}^N \sum_{\gamma=1}^N \frac{w_\alpha w_\gamma}{|\xi_\gamma^{-2} (\xi_\gamma^3 + \xi_\alpha^3)^{2/3}|} \chi(d_{max} > \sqrt[3]{\xi_\alpha^3 + \xi_\gamma^3}) \underbrace{\chi(\sqrt[3]{\xi_\alpha^3 + \xi_\gamma^3} > \xi_\alpha)}_{\equiv 1} (\sqrt[3]{\xi_\alpha^3 + \xi_\gamma^3})^l \psi(\xi_\alpha, \xi_\gamma) \\
&= \frac{1}{2} \sum_{\alpha=1}^N \sum_{\gamma=1}^N \frac{w_\alpha w_\gamma}{|\xi_\gamma^{-2} (\xi_\gamma^3 + \xi_\alpha^3)^{2/3}|} \chi(d_{max} > \sqrt[3]{\xi_\alpha^3 + \xi_\gamma^3}) (\sqrt[3]{\xi_\alpha^3 + \xi_\gamma^3})^l \psi(\xi_\alpha, \xi_\gamma).
\end{aligned}$$

Hierfür wurde im Schritt (*) verwendet, dass für Verkettungen der Form $\delta \circ h$ der Delta-Distribution δ mit einer reellen Funktion h gilt:

$$\delta(h(x)) = \sum_{i=1}^n \frac{\delta(x - x_i)}{|h'(x_i)|}, \quad x_i \text{ ist einfache Nullstelle von } h.$$

In der zuvor verwendeten Umformung wurde diese Formel für $h(x) := \sqrt[3]{x^3 - \xi_\alpha^3} - \xi_\gamma$ mit Nullstelle $x_0 = \sqrt[3]{\xi_\gamma^3 + \xi_\alpha^3}$ und $h'(x_0) = \xi_\gamma^{-2}(\xi_\gamma^3 + \xi_\alpha^3)^{2/3}$ benutzt.

Letztlich folgt in ähnlicher Weise für das Verschwinden von Tropfen durch Verschmelzen zu größeren Tropfen:

$$\begin{aligned} D_{coal}^{(l)}(d_p) &= \int_0^{d_{max}} d_p^l q(d_p) \int_0^{d_{max}} \chi(\sqrt[3]{d_{max}^3 - d_p^3} > d_p') \psi(d_p, d_p') q(d_p') dd_p' dd_p \\ &\approx \sum_{\gamma=1}^N \sum_{\alpha=1}^N \chi(\sqrt[3]{d_{max}^3 - \xi_\gamma^3} > \xi_\alpha) w_\alpha w_\gamma \xi_\gamma^l \psi(\xi_\gamma, \xi_\alpha). \end{aligned}$$

Die experimentellen Untersuchungen von Coulaloglou und Tavlarides [3] spezifizieren die Funktionen κ , ψ und β , wobei zu beachten ist, dass die Formeln in [3] in Abhängigkeit des Tropfenvolumens geschrieben sind. Eine Formulierung in Abhängigkeit des Tropfendurchmessers ist zum Teil in [6] zu finden.

2.3.3 Sauter-Durchmesser und Varianz

Einige Momente der Anzahl-Dichtefunktion haben eine physikalische Bedeutung, weshalb ingenieurwissenschaftliche Untersuchungen [30] den Sauter-Durchmesser d_{32} als charakteristische Größe der Tropfenverteilung bestimmt haben. In praktischen Anwendungen wird gefordert, dass möglichst viele Tropfen einen gewünschten Sauter-Durchmesser \hat{d}_{32} haben und die Varianz σ der Tropfenverteilung klein sein soll.

Beide Größen können nach [38] aus den ersten vier Momenten berechnet werden:

$$d_{32} := \frac{m^{(3)}}{m^{(2)}}, \quad \sigma = \sqrt{\frac{m^{(2)}}{m^{(0)}} - \left(\frac{m^{(1)}}{m^{(0)}}\right)^2}. \quad (16)$$

Daher ist es naheliegend, in der Implementierung des PD-Algorithmus' $N = 2$ zu wählen, da man so die ersten vier Momente exakt berechnet.

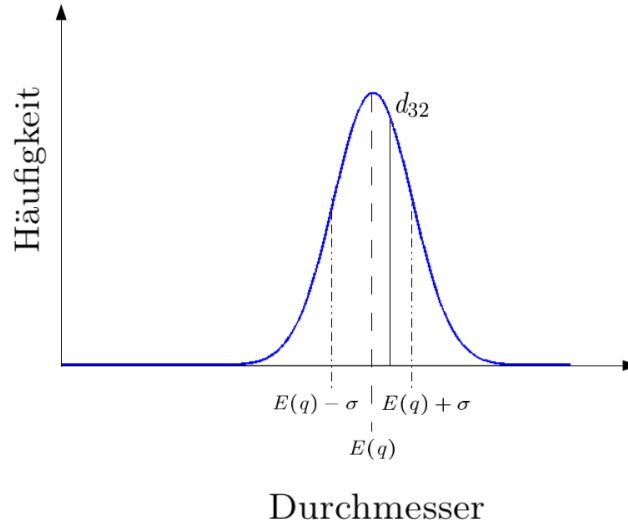


Abbildung 5: Die stochastische Verteilung der Tropfendurchmesser ist nicht bekannt und wird über Momente charakterisiert.

Wie in Abbildung 5 angedeutet, ist der Sauter-Durchmesser d_{32} größer als der Erwartungswert $E(q)$ der Dichtefunktion q , vgl. Ausführungen in [30].

Eine Übersicht über numerische Methoden, die eine unbekannte Dichtefunktion aus einer gegebenen endlichen Anzahl ihrer Momente approximiert, ist in [12] gegeben, wobei darauf hingewiesen wird, dass es sich hierbei um ein schlecht gestelltes inverses Problem handelt.

2.4 Zusammenfassung

Insgesamt führt die Modellierung von Dispersionen in turbulenter Strömung auf ein System aus gekoppelten partiellen Differentialgleichungen in den Unbekannten \vec{U} , P , k , ϵ und $m^{(l)}$ für $l \in \mathbb{N}_0$, wobei zu beachten ist, dass die Koeffizienten ν^* und ν_T ebenfalls von k und ϵ abhängen.

Dieses System (17) ist **abgeschlossen**. Bei einem dreidimensionalen Simulationsgebiet Ω liefert die approximierte RANS-Gleichung komponentenweise drei unabhängige Gleichungen. Durch die Nebenbedingung der divergenzfreien Geschwindigkeit erhält man eine weitere Gleichung. Das k - ϵ Turbulenzmodell ist ein Zwei-Gleichungs-Modell, in dem zwei weitere Transportgleichungen für k und ϵ entwickelt werden.

Darüber hinaus werden die Momente $m^{(l)}$ der Anzahl-Dichtefunktion q wie in Kapitel 2.3.1 beschrieben durch Gewichte ω_α und Abszissen ξ_α derart approximiert, dass zum einen $m^{(l)} \approx \sum_{\alpha=1}^N \omega_\alpha \xi_\alpha^l$ gilt und zum anderen die rechte Seite in der Momentengleichung ausschließlich von diesen $2N$ neu eingeführten Hilfsgrößen abhängt, $S^{(l)} \sim f(\omega_1, \xi_1, \dots, \omega_N, \xi_N)$, und hier keine höheren Momente auftreten.

Für $l \in \{0, \dots, 2N - 1\}$ ist das System (17) somit abgeschlossen.

$$\left. \begin{aligned} \frac{\partial}{\partial t} \vec{U} + \operatorname{div}(\vec{U} \otimes \vec{U}) + \nabla P + \frac{2}{3} \nabla k - \operatorname{div}(\nu^*(\nabla \vec{U} + \nabla \vec{U}^T)) - \vec{G} &= 0 \\ \operatorname{div} \vec{U} &= 0 \\ \frac{\partial}{\partial t} k + \vec{U} \cdot \nabla k - \frac{\nu_T}{2} \|\nabla \vec{U} + \nabla \vec{U}^T\|_F^2 - \operatorname{div}(\nu_T \nabla k) + \epsilon &= 0 \\ \frac{\partial}{\partial t} \epsilon + \vec{U} \cdot \nabla \epsilon - \frac{c_1}{2} k \|\nabla \vec{U} + \nabla \vec{U}^T\|_F^2 - \operatorname{div}(\frac{c_\epsilon}{c_\mu} \nu_T \nabla \epsilon) + c_2 \frac{\epsilon^2}{k} &= 0 \\ \frac{\partial}{\partial t} m^{(l)} + \operatorname{div}(\vec{U} m^{(l)}) - \operatorname{div}(\nu_T \nabla m^{(l)}) &= S^{(l)} \end{aligned} \right\} \quad (17)$$

Es ist zu beachten, dass die Gültigkeit des obigen Systems nur im Inneren des Raums $\Omega \times [0, T_{end}]$ gilt. Um das Problem mathematisch vollständig zu beschreiben, ist die Vorgabe von Randbedingungen notwendig. Diese werden im folgenden Kapitel anhand einer konkreter Anwendung diskutiert.

3 Anwendung auf einen zweidimensionalen Rührer

Das in Kapitel 2 hergeleitete System (17) zur Beschreibung von Dispersionen in turbulenten Strömungen soll nun zur Simulation eines Mischvorgangs in einem Rührer angewendet werden. In der hier präsentierten Anwendung wird dies auf einen **zweidimensionalen Rührer** beschränkt, dessen Geometrie durch die Vorgabe des Innenradius' r_1 , der Länge der Rührblätter r_2 sowie des Behälterradius' R eindeutig beschrieben ist (siehe Abbildung 6).

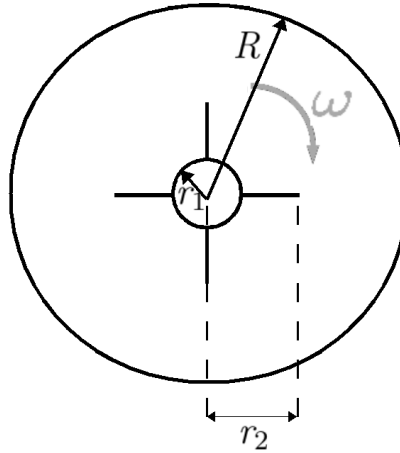


Abbildung 6: Die Geometrie des Rührers ist durch Vorgabe der Abmaße r_1, r_2, R vollständig beschrieben.

Auf Grund der speziellen Form des Rührers bietet sich für die Diskretisierung die Wahl von Polarkoordinaten an. Im Folgenden wird daher zunächst auf die Generierung eines geeigneten Gitters eingegangen. Die Diskretisierung der in (17) enthaltenen Differentialoperatoren wird mit Hilfe der Methode der Finiten Differenzen durchgeführt. Angepasst an das konkrete Problem werden in Abschnitt 3.3 Randbedingungen am Rührbehälter sowie den Rührblättern vorgegeben. Darüber hinaus wird das Drehen des Rührers durch ein rotierendes Koordinatensystem in Abschnitt 3.5 beschrieben.

Die Implementierung wurde in MATLAB [39] durchgeführt.

3.1 Gittergenerierung

Um die Ortsdiskretisierung mit der Finiten-Differenzen Methode durchzuführen, ist man an einer äquidistanten Diskretisierung in beide Raumdimensionen interessiert. Da, wie bereits erwähnt, das Diskretisierungsgebiet $\Omega \subset \mathbb{R}^2$ kreisförmig ist, bietet sich die Benutzung von Polarkoordinaten an.

3.1.1 Koordinatentransformation

In Polarkoordinaten ist das kreisförmige Rührergebiet ein Rechteck $[r_1, R] \times [0, 2\pi)$, das leicht mit Schrittweite h_r in Richtung \vec{e}_r und h_θ in Richtung \vec{e}_θ äquidistant diskretisiert werden kann (vgl. Abbildung 7, links). Die Vektoren

$$\vec{e}_r = \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} \quad \text{und} \quad \vec{e}_\theta = \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix}$$

bilden eine Orthonormalbasis, wobei zu beachten ist, dass die partiellen Ableitungen $\frac{\partial \vec{e}_r}{\partial \theta} = \vec{e}_\theta$ und $\frac{\partial \vec{e}_\theta}{\partial \theta} = -\vec{e}_r$ nicht wie im kartesischen Fall verschwinden. Ein Punkt in Polarkoordinaten besitzt folglich die Darstellung $\vec{x} = r\vec{e}_r + \theta\vec{e}_\theta$, mit Koordinaten r und θ .

Um die kartesischen Koordinaten dieses Punkts bezüglich der Einheitsvektoren e_1 und e_2 zu berechnen, wird die Koordiantentransformation Φ eingeführt:

$$\Phi : [r_1, R] \times [0, 2\pi) \rightarrow \mathbb{R}^2, \Phi(r, \theta) = \begin{bmatrix} r \cdot \cos \theta \\ r \cdot \sin \theta \end{bmatrix}.$$

Für den Punkt \vec{x} gilt dann $\vec{x} = \Phi_1(r, \theta)\vec{e}_1 + \Phi_2(r, \theta)\vec{e}_2 = \Phi(r, \theta)$.

In Abbildung 7 ist dargestellt, dass das Bild von $[r_1, R] \times [0, \frac{\pi}{3}]$ unter Φ im kartesischen Koordinatensystem das erste Sechstel des Rührers ergibt:

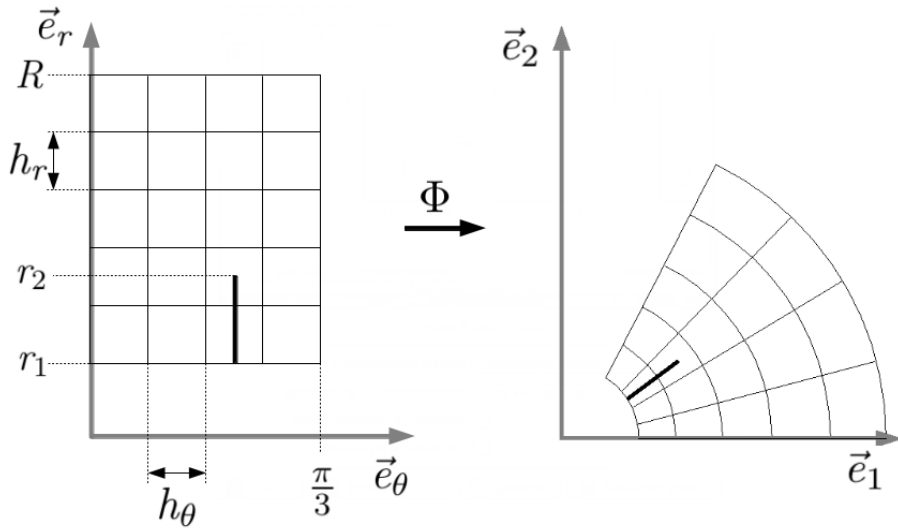


Abbildung 7: In Polarkoordinaten (links) ist der Rührer ein rechteckiges Gebiet.

Analog erhält man $\vec{U} = U^r \vec{e}_r + U^\theta \vec{e}_\theta$ für die Darstellung der Geschwindigkeit in Polarkoordinaten. Eine Rücktransformation in kartesische Koordianten ist dann durch

$$\begin{aligned} U^1 &= U^r \cos \theta - U^\theta \sin \theta, \\ U^2 &= U^r \sin \theta + U^\theta \cos \theta, \end{aligned} \tag{18}$$

gegeben, wobei $\vec{U} = U^1 \vec{e}_1 + U^2 \vec{e}_2$.

Es ist im Weiteren zu beachten, dass die Differentialoperatoren in Polarkoordinaten eine andere Form haben (siehe beispielsweise [37]).

3.1.2 Versetzte Gitter

Oft ist für die Implementierung das Arbeiten mit so genannten versetzten Gittern (engl. *staggered grids*) notwendig um die Eindeutigkeit der Lösung im Diskreten zu garantieren (vgl. [8, S. 27]). Das Arbeiten mit versetzten Gittern hat außerdem den Vorteil, dass beispielsweise ein Differenzenquotient aus Geschwindigkeitswert im Norden und im Süden direkt im Zellmittelpunkt des Hauptnetzes vorliegt.

In Abbildung 8 ist in schwarz das so genannte Hauptnetz dargestellt, auf dem unter anderem der Druck $P_{i,j}$ vorliegt. In den Mittelpunkten der beiden versetzten Gitter liegen die r -Komponente der Geschwindigkeit (blau) bzw. die θ -Komponente der Geschwindigkeit (rot). Die Indizierung ist relativ zum Mittelpunkt des Hauptnetzes gegeben, wobei $\frac{1}{2}$ jeweils für halbe Schrittweiten steht.

Randbedingungen sind immer auf dem Hauptnetz gegeben, so dass diese auf den versetzten Gittern aus dem Randwert und einem randnahen Punkt extrapoliert werden müssen.

Werte mit der in Abbildung 8 eingeführten Indizierung sollen im Folgenden immer auf dem entsprechenden Gitter liegen, wobei zwischen den Gittern linear interpoliert werden kann.

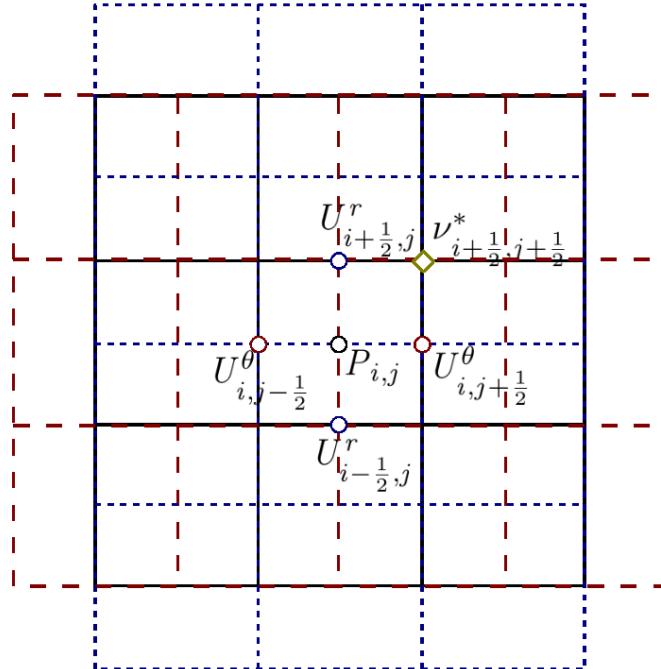


Abbildung 8: Versetzte Gitter mit Druckwert $P_{i,j}$ im Zellmittelpunkt des Hauptnetzes (schwarz) sowie r -Komponente der Geschwindigkeit im Norden und Süden (blau) und θ -Komponente der Geschwindigkeit im Osten und Westen (rot).

3.2 Ortsdiskretisierung in Polarkoordinaten

Zur Diskretisierung der im System (17) enthaltenen Differentialoperatoren, sollen diese in einem ersten Schritt zunächst in Polarkoordinaten dargestellt werden. Anschließend werden die auftretenden partiellen Ableitungen mit Hilfe der Finite-Differenzen Methode (FDM) diskretisiert. Die FDM ist ein Standardverfahren zur numerischen Lösung von partiellen Differentialgleichungen und besagt im Wesentlichen, dass die Ableitung einer skalaren Funktion $f_{FD} : \mathbb{R} \rightarrow \mathbb{R}$ durch den Differenzenquotienten approximiert werden kann:

$$\frac{\partial f_{FD}}{\partial x}(x_0) := \lim_{\epsilon \rightarrow 0} \frac{f_{FD}(x_0 + \epsilon) - f_{FD}(x_0)}{\epsilon} \approx \frac{f_{FD}(x_0 + h) - f_{FD}(x_0)}{h}, \quad \text{mit } h \ll 1.$$

Eine Einführung in die Finite-Differenzen Methode sowie Aussagen über die Konvergenzordnung der FDM befinden sich in [20, Kap. 3].

3.2.1 Diskretisierung der RANS-Gleichung

Die RANS-Gleichung (7) ist mit $\vec{G} = 0$ gegeben durch:

$$\frac{\partial}{\partial t} \vec{U} = - \underbrace{(\vec{U} \cdot \nabla) \vec{U}}_{1.)} - \underbrace{\frac{2}{3} \nabla k}_{2.)} + \underbrace{\text{div}(\nu^* (\nabla \vec{U} + \nabla \vec{U}^T))}_{3.)} - \nabla P. \quad (19)$$

Für die weitere Diskretisierung werden die Terme in (19) zunächst einzeln betrachtet:

1.) Für den Konvektionsterm gilt:

$$(\vec{U} \cdot \nabla) \vec{U} = \left(\begin{bmatrix} U^r \\ U^\theta \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial}{\partial r} \\ \frac{1}{r} \frac{\partial}{\partial \theta} \end{bmatrix} \right) \begin{bmatrix} U^r \\ U^\theta \end{bmatrix} = \begin{bmatrix} U^r \frac{\partial}{\partial r} U^r + \frac{U^\theta}{r} \frac{\partial}{\partial \theta} U^r \\ U^r \frac{\partial}{\partial r} U^\theta + \frac{U^\theta}{r} \frac{\partial}{\partial \theta} U^\theta \end{bmatrix} = \begin{bmatrix} U^r \frac{\partial U^r}{\partial r} + \frac{U^\theta}{r} \frac{\partial U^r}{\partial \theta} - \frac{U^\theta U^\theta}{r} \\ U^r \frac{\partial U^\theta}{\partial r} + \frac{U^\theta}{r} \frac{\partial U^\theta}{\partial \theta} + \frac{U^\theta}{r} \frac{\partial U^\theta}{\partial \theta} \end{bmatrix}.$$

2.) Für den Gradienten der turbulenten kinetischen Energie k gilt:

$$\frac{2}{3} \nabla k = \frac{2}{3} \begin{bmatrix} \frac{\partial k}{\partial r} \\ \frac{1}{r} \frac{\partial k}{\partial \theta} \end{bmatrix}.$$

3.) Der Spannungstensor $\tilde{T} = \nabla \vec{U} + \nabla \vec{U}^T$ wird in Kapitel 3.2.2 hergeleitet. Die Divergenz ist somit gegeben durch:

$$\begin{aligned} \text{div}(\nu^* (\nabla \vec{U} + \nabla \vec{U}^T)) &= \text{div} \left(\nu^* \begin{bmatrix} 2 \frac{\partial U^r}{\partial r} & \frac{1}{r} \frac{\partial U^r}{\partial \theta} + \frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r} \\ \frac{1}{r} \frac{\partial U^r}{\partial \theta} + \frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r} & 2 \left(\frac{\partial U^\theta}{\partial \theta} + U^r \right) \end{bmatrix} \right) \\ &= \frac{1}{r} \begin{bmatrix} \frac{\partial}{\partial r} \left(r \cdot 2\nu^* \frac{\partial U^r}{\partial r} \right) + \frac{\partial}{\partial \theta} \left(\nu^* \left(\frac{1}{r} \frac{\partial U^r}{\partial \theta} + \frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r} \right) \right) \\ \frac{\partial}{\partial r} \left(r \cdot \nu^* \left(\frac{1}{r} \frac{\partial U^r}{\partial \theta} + \frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r} \right) \right) + \frac{\partial}{\partial \theta} \left(\nu^* \left(\frac{2}{r} \frac{\partial U^\theta}{\partial \theta} + \frac{2U^r}{r} \right) \right) \end{bmatrix} \\ &= \begin{bmatrix} \frac{\partial}{\partial r} \left(2\nu^* \frac{\partial U^r}{\partial r} \right) + \frac{2\nu^*}{r} \frac{\partial U^r}{\partial r} + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\nu^* \left(\frac{1}{r} \frac{\partial U^r}{\partial \theta} \right) \right) + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\nu^* \left(\frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r} \right) \right) \\ \frac{\nu^*}{r} \left(\frac{1}{r} \left(\frac{\partial U^r}{\partial \theta} - U^\theta \right) + \frac{\partial U^\theta}{\partial r} \right) + \frac{\partial}{\partial r} \left(\frac{\nu^*}{r} \left(\frac{\partial U^r}{\partial \theta} - U^\theta \right) \right) + \frac{\partial}{\partial r} \left(\nu^* \frac{\partial U^\theta}{\partial r} \right) + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\frac{2\nu^*}{r} \frac{\partial U^\theta}{\partial \theta} \right) + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\frac{2\nu^*}{r} U^r \right) \end{bmatrix}. \end{aligned}$$

Nachdem die Differentialoperatoren in Polarkoordinaten ausgeschrieben wurden, ergibt sich (19) komponentenweise zu:

$$\begin{aligned} \frac{\partial U^r}{\partial t} = & - \left[\overbrace{U^r \frac{\partial U^r}{\partial r} + \frac{U^\theta}{r} \left(\frac{\partial U^r}{\partial \theta} - U^\theta \right)}^{=:F^1} \right] - \overbrace{\frac{2}{3} \frac{\partial k}{\partial r}}^{=:F^2} \\ & + \underbrace{\frac{\partial}{\partial r} \left(2\nu^* \frac{\partial U^r}{\partial r} \right) + \frac{2\nu^*}{r} \frac{\partial U^r}{\partial r} + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\nu^* \left(\frac{1}{r} \frac{\partial U^r}{\partial \theta} \right) \right) + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\nu^* \left(\frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r} \right) \right)}_{=:F^3} - \frac{\partial P}{\partial r} \end{aligned}$$

und:

$$\begin{aligned} \frac{\partial U^\theta}{\partial t} = & - \left[\overbrace{U^r \frac{\partial U^\theta}{\partial r} + \frac{U^\theta}{r} \left(U^r + \frac{\partial U^\theta}{\partial \theta} \right)}^{=:G^1} \right] - \overbrace{\frac{2}{3} \frac{1}{r} \frac{\partial k}{\partial \theta}}^{=:G^2} - \frac{1}{r} \frac{\partial P}{\partial \theta} \\ & + \underbrace{\frac{\nu^*}{r} \left(\frac{1}{r} \left(\frac{\partial U^r}{\partial \theta} - U^\theta \right) + \frac{\partial U^\theta}{\partial r} \right) + \frac{\partial}{\partial r} \left(\frac{\nu^*}{r} \left(\frac{\partial U^r}{\partial \theta} - U^\theta \right) \right) + \frac{\partial}{\partial r} \left(\nu^* \frac{\partial U^\theta}{\partial r} \right) + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\frac{2\nu^*}{r} \frac{\partial U^\theta}{\partial \theta} \right) + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\frac{2\nu^*}{r} U^r \right)}_{=:G^3} \end{aligned}$$

Wie in Abschnitt 3.4.1 beschrieben, kann Gleichung (19) nach Diskretisierung von

$$F_{i+\frac{1}{2},j} := -F_{i+\frac{1}{2},j}^1 - F_{i+\frac{1}{2},j}^2 + F_{i+\frac{1}{2},j}^3 \quad (20)$$

$$G_{i,j+\frac{1}{2}} := -G_{i,j+\frac{1}{2}}^1 - G_{i,j+\frac{1}{2}}^2 + G_{i,j+\frac{1}{2}}^3 \quad (21)$$

in die so genannte **Druck-Poissongleichung (PPE)** überführt werden. Hierfür wird der Druckgradient zunächst nicht diskretisiert und die Terme F und G auf den versetzten Gittern berechnet, da diese später in entsprechenden partiellen Ableitungen auf der rechten Seite einer Poissongleichung für den Druck auftauchen (vgl. Ausführungen in [8, Kap. 3.2.3]).

Darüber hinaus wird an manchen Stellen eine **Upwind-Diskretisierung** durchgeführt. Hierbei wird die erste Ableitung der Geschwindigkeit abhängig vom Vorzeichen der Geschwindigkeit an dieser Stelle durch links- bzw. rechtsseitige Differenz approximiert. Das Upwind-Verfahren wird zum Beispiel in [5, Kap. 3] näher beschrieben.

Eine vollständige Finite-Differenzen Diskretisierung mit Upwind-Verfahren von (20) und (21) ist gegeben durch:

$$F_{i+\frac{1}{2},j}^1 \approx \max(U_{i+\frac{1}{2},j}^r, 0) \cdot \frac{U_{i+\frac{1}{2},j}^r - U_{i-\frac{1}{2},j}^r}{h_r} + \min(U_{i+\frac{1}{2},j}^r, 0) \cdot \frac{U_{i+\frac{3}{2},j}^r - U_{i+\frac{1}{2},j}^r}{h_r} \\ + \frac{\max(U_{i+\frac{1}{2},j}^\theta, 0)}{r_{i+\frac{1}{2},j}} \left(\frac{U_{i+\frac{1}{2},j}^r - U_{i+\frac{1}{2},j-1}^r}{h_\theta} - U_{i+\frac{1}{2},j}^\theta \right) + \frac{\min(U_{i+\frac{1}{2},j}^\theta, 0)}{r_{i+\frac{1}{2},j}} \left(\frac{U_{i+\frac{1}{2},j+1}^r - U_{i+\frac{1}{2},j}^r}{h_\theta} - U_{i+\frac{1}{2},j}^\theta \right)$$

$$F_{i+\frac{1}{2},j}^2 \approx \frac{2}{3} \frac{k_{i+1,j} - k_{i,j}}{h_r}$$

$$F_{i+\frac{1}{2},j}^3 \approx \frac{2}{h_r^2} \left(\nu_{i+1,j}^* (U_{i+\frac{3}{2},j}^r - U_{i+\frac{1}{2},j}^r) - \nu_{i,j}^* (U_{i+\frac{1}{2},j}^r - U_{i-\frac{1}{2},j}^r) \right) + \frac{2\nu_{i+\frac{1}{2},j}^*}{r_{i+\frac{1}{2},j}} \left(\frac{U_{i+\frac{3}{2},j}^r - U_{i-\frac{1}{2},j}^r}{2h_r} \right) \\ + \frac{1}{r_{i+\frac{1}{2},j}} \left(\frac{1}{h_\theta^2} \left(\frac{\nu_{i+\frac{1}{2},j+\frac{1}{2}}^*}{r_{i+\frac{1}{2},j+\frac{1}{2}}} (U_{i+\frac{1}{2},j+1}^r - U_{i+\frac{1}{2},j}^r) - \frac{\nu_{i+\frac{1}{2},j-\frac{1}{2}}^*}{r_{i+\frac{1}{2},j-\frac{1}{2}}} (U_{i+\frac{1}{2},j}^r - U_{i+\frac{1}{2},j-1}^r) \right) \right) \\ + \frac{1}{r_{i+\frac{1}{2},j}} \left(\frac{1}{h_\theta} \left(\nu_{i+\frac{1}{2},j+\frac{1}{2}}^* \left(\frac{U_{i+1,j+\frac{1}{2}}^\theta - U_{i,j+\frac{1}{2}}^\theta}{h_r} - \frac{U_{i+\frac{1}{2},j+\frac{1}{2}}^\theta}{r_{i+\frac{1}{2},j+\frac{1}{2}}} \right) - \nu_{i+\frac{1}{2},j-\frac{1}{2}}^* \left(\frac{U_{i+1,j-\frac{1}{2}}^\theta - U_{i,j-\frac{1}{2}}^\theta}{h_r} - \frac{U_{i+\frac{1}{2},j-\frac{1}{2}}^\theta}{r_{i+\frac{1}{2},j-\frac{1}{2}}} \right) \right) \right)$$

$$G_{i,j+\frac{1}{2}}^1 \approx \max(U_{i,j+\frac{1}{2}}^r, 0) \cdot \frac{U_{i,j+\frac{1}{2}}^\theta - U_{i-1,j+\frac{1}{2}}^\theta}{h_r} + \min(U_{i,j+\frac{1}{2}}^r, 0) \cdot \frac{U_{i+1,j+\frac{1}{2}}^\theta - U_{i,j+\frac{1}{2}}^\theta}{h_r} \\ + \frac{\max(U_{i,j+\frac{1}{2}}^\theta, 0)}{r_{i,j+\frac{1}{2}}} \left(\frac{U_{i,j+\frac{1}{2}}^\theta - U_{i,j-\frac{1}{2}}^\theta}{h_\theta} + U_{i,j+\frac{1}{2}}^r \right) + \frac{\min(U_{i,j+\frac{1}{2}}^\theta, 0)}{r_{i,j+\frac{1}{2}}} \left(\frac{U_{i,j+\frac{3}{2}}^\theta - U_{i,j+\frac{1}{2}}^\theta}{h_\theta} + U_{i,j+\frac{1}{2}}^r \right)$$

$$G_{i,j+\frac{1}{2}}^2 \approx \frac{2}{3} \frac{1}{r_{i,j+\frac{1}{2}}} \frac{k_{i,j+1} - k_{i,j}}{h_\theta}$$

$$G_{i,j+\frac{1}{2}}^3 \approx \frac{\nu_{i,j+\frac{1}{2}}^*}{r_{i,j+\frac{1}{2}}} \left(\frac{1}{r_{i,j+\frac{1}{2}}} \left(\frac{U_{i,j+1}^r - U_{i,j}^r}{h_\theta} - U_{i,j+\frac{1}{2}}^\theta \right) + \frac{U_{i+1,j+\frac{1}{2}}^\theta - U_{i-1,j+\frac{1}{2}}^\theta}{2h_r} \right) \\ + \frac{1}{h_r} \left(\frac{\nu_{i+\frac{1}{2},j+\frac{1}{2}}^*}{r_{i+\frac{1}{2},j+\frac{1}{2}}} \left(\frac{U_{i+\frac{1}{2},j+1}^r - U_{i+\frac{1}{2},j}^r}{h_\theta} - U_{i+\frac{1}{2},j+\frac{1}{2}}^\theta \right) - \frac{\nu_{i-\frac{1}{2},j+\frac{1}{2}}^*}{r_{i-\frac{1}{2},j+\frac{1}{2}}} \left(\frac{U_{i-\frac{1}{2},j+1}^r - U_{i-\frac{1}{2},j}^r}{h_\theta} - U_{i-\frac{1}{2},j+\frac{1}{2}}^\theta \right) \right) \\ + \frac{1}{h_r^2} \left(\nu_{i+\frac{1}{2},j+\frac{1}{2}}^* (U_{i+1,j+\frac{1}{2}}^\theta - U_{i,j+\frac{1}{2}}^\theta) - \nu_{i-\frac{1}{2},j+\frac{1}{2}}^* (U_{i,j+\frac{1}{2}}^\theta - U_{i-1,j+\frac{1}{2}}^\theta) \right) \\ + \frac{2}{r_{i,j+\frac{1}{2}} h_\theta^2} \left(\frac{\nu_{i,j+1}^*}{r_{i,j+1}} (U_{i,j+\frac{3}{2}}^\theta - U_{i,j+\frac{1}{2}}^\theta) - \frac{\nu_{i,j}^*}{r_{i,j}} (U_{i,j+\frac{1}{2}}^\theta - U_{i,j-\frac{1}{2}}^\theta) \right) + \frac{2}{r_{i,j+\frac{1}{2}} h_\theta} \left(\frac{\nu_{i,j+1}^* U_{i,j+1}^r}{r_{i,j+1}} - \frac{\nu_{i,j}^* U_{i,j}^r}{r_{i,j}} \right)$$

Diese Diskretisierung ist in dieser Form nur für randferne Punkte gültig. Sobald man in die Nähe der Rührerblätter oder des Rührergehäuses kommt, ist mit entsprechenden Randwerten zu arbeiten, die in Kapitel 3.3 angegeben sind.

3.2.2 Diskretisierung der skalaren Transportgleichungen

Die Gleichungen (8),(9) und (12) haben eine ähnliche mathematische Struktur und können daher in der verallgemeinerten skalaren Funktion φ geschrieben werden als:

$$\frac{\partial}{\partial t}\varphi + \overbrace{\vec{U} \cdot \nabla \varphi - \operatorname{div}(\xi_1 \nabla \varphi)}^{=: R^\varphi} - \xi_2 \overbrace{\|\nabla \vec{U} + \nabla \vec{U}^T\|_F^2}^{=: \tilde{T}_F(\vec{U})} = \eta^\varphi. \quad (22)$$

Hierbei wurden die Abkürzungen

$$\xi_1 := \begin{cases} \nu_T & \text{für } \varphi = k \\ \frac{c_\epsilon}{c_\mu} \nu_T & \text{für } \varphi = \epsilon \\ \nu_T & \text{für } \varphi = m^{(l)} \end{cases}, \quad \xi_2 := \begin{cases} \frac{\nu_T}{2} & \text{für } \varphi = k \\ \frac{c_1}{2} k & \text{für } \varphi = \epsilon \\ 0 & \text{für } \varphi = m^{(l)} \end{cases} \quad \text{und} \quad \eta^\varphi := \begin{cases} -\epsilon & \text{für } \varphi = k \\ -c_2 \frac{\epsilon^2}{k} & \text{für } \varphi = \epsilon \\ S^{(l)} & \text{für } \varphi = m^{(l)} \end{cases}$$

eingeführt.

Da die Quellterme $S^{(l)}$ bereits in Kapitel 2.3.2 ortsabhängig bestimmt und diskretisiert wurden, ist die rechte Seite η^φ von Gleichung (22) bekannt und kann leicht im Ort diskretisiert werden.

Im Weiteren soll eine vollständige Ortsdiskretisierung der Frobenius Norm des Reynolds'schen Spannungstensors in Polarkoordinaten sowie des Differentialoperators R^φ diskutiert werden.

Reynolds'scher Spannungstensor in Polarkoordinaten:

Im Folgenden soll das Quadrat der Frobenius Norm des Reynolds'schen Spannungstensors

$$\tilde{T}_F(\vec{U}) = \|\nabla \vec{U} + \nabla \vec{U}^T\|_F^2$$

diskretisiert werden, wobei $\vec{U} = U^r \vec{e}_r + U^\theta \vec{e}_\theta$ in Zylinderkoordinaten gegeben ist und $\|\cdot\|_F$ die Frobenius Norm einer Matrix ist.

Schreibe dazu das totale Differential von \vec{U} und bestimme $\nabla \vec{U}$ später durch Koeffizientenvergleich:

$$d\vec{U} = \frac{\partial}{\partial r} \vec{U} dr + \frac{1}{r} \frac{\partial}{\partial \theta} \vec{U} r d\theta \stackrel{!}{=} [dr \ r d\theta] \cdot \nabla \vec{U}.$$

Die partiellen Ableitungen sind dabei:

$$\begin{aligned} \frac{\partial}{\partial r} \vec{U} &= \frac{\partial}{\partial r} (U^r \vec{e}_r + U^\theta \vec{e}_\theta) = \frac{\partial U^r}{\partial r} \vec{e}_r + \frac{\partial U^\theta}{\partial r} \vec{e}_\theta, \\ \frac{1}{r} \frac{\partial}{\partial \theta} \vec{U} &= \frac{1}{r} \frac{\partial}{\partial \theta} (U^r \vec{e}_r + U^\theta \vec{e}_\theta) = \frac{1}{r} \left(\frac{\partial U^r}{\partial \theta} \vec{e}_r + U^r \vec{e}_\theta + \frac{\partial U^\theta}{\partial \theta} \vec{e}_\theta - U^\theta \vec{e}_r \right). \end{aligned}$$

Koeffizientenvergleich in \vec{e}_r - und \vec{e}_θ -Richtung liefert schließlich:

$$\nabla \vec{U} = \begin{bmatrix} \frac{\partial U^r}{\partial r} & \frac{\partial U^\theta}{\partial r} \\ \frac{1}{r} \frac{\partial U^r}{\partial \theta} - \frac{U^\theta}{r} & \frac{1}{r} \left(\frac{\partial U^\theta}{\partial \theta} + U^r \right) \end{bmatrix}.$$

Somit ist der Spannungstensor eine symmetrische Matrix, deren quadratische Frobenius Norm sich zu

$$\begin{aligned}\tilde{T}_F(\vec{U}) &= \left\| \begin{bmatrix} 2\frac{\partial U^r}{\partial r} & \frac{1}{r}\frac{\partial U^r}{\partial \theta} + \frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r} \\ \frac{1}{r}\frac{\partial U^r}{\partial \theta} + \frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r} & \frac{2}{r}\left(\frac{\partial U^\theta}{\partial \theta} + U^r\right) \end{bmatrix} \right\|_F^2 \\ &= \left(2\frac{\partial U^r}{\partial r}\right)^2 + 2\left(\frac{1}{r}\frac{\partial U^r}{\partial \theta} + \frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r}\right)^2 + \left(\frac{2}{r}\frac{\partial U^\theta}{\partial \theta} + U^r\right)^2\end{aligned}\quad (23)$$

ergibt.

Die in (23) auftretenden partiellen Ableitungen können erneut mit der FDM diskretisiert werden:

$$\begin{aligned}\tilde{T}_F^{i,j} &\approx \left(\frac{2(U_{i+\frac{1}{2},j}^r - U_{i-\frac{1}{2},j}^r)}{h_r}\right)^2 + 2\left(\frac{1}{r_{i,j}}\frac{U_{i,j+\frac{1}{2}}^r - U_{i,j-\frac{1}{2}}^r}{h_\theta} + \frac{U_{i+\frac{1}{2},j}^\theta - U_{i-\frac{1}{2},j}^\theta}{h_r} - \frac{U_{i,j}^\theta}{r_{i,j}}\right)^2 \\ &\quad + \left(\frac{2}{r_{i,j}}\frac{U_{i,j+\frac{1}{2}}^\theta - U_{i,j-\frac{1}{2}}^\theta}{h_\theta} + U_{i,j}^r\right)^2.\end{aligned}$$

Der in (23) angegeben Spannungstensor ist in [21] zu finden.

Ortsdiskretisierung:

Für die Diskretisierung von R^φ schreibe zunächst die Differentialoperatoren nach [37] aus:

$$\begin{aligned}R^\varphi &= \vec{U} \cdot \nabla \varphi - \text{div}(\xi_1 \nabla \varphi) \\ &= \begin{bmatrix} U^r \\ U^\theta \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \varphi}{\partial r} \\ \frac{1}{r} \frac{\partial \varphi}{\partial \theta} \end{bmatrix} - \text{div} \begin{bmatrix} \xi_1 \frac{\partial \varphi}{\partial r} \\ \xi_1 \frac{1}{r} \frac{\partial \varphi}{\partial \theta} \end{bmatrix} \\ &= \left(U^r \frac{\partial \varphi}{\partial r} + U^\theta \frac{1}{r} \frac{\partial \varphi}{\partial \theta} \right) - \frac{1}{r} \frac{\partial}{\partial r} \left(r \cdot \xi_1 \frac{\partial \varphi}{\partial r} \right) - \frac{1}{r} \frac{\partial}{\partial \theta} \left(\xi_1 \frac{1}{r} \frac{\partial \varphi}{\partial \theta} \right).\end{aligned}\quad (24)$$

Anwendung der FDM und Upwind-Diskretisierung des Konvektionsterms führt auf:

$$\begin{aligned}R_{i,j}^\varphi &\approx \min(U_{i,j}^r, 0) \cdot \frac{\varphi_{i+1,j} - \varphi_{i,j}}{h_r} + \max(U_{i,j}^r, 0) \cdot \frac{\varphi_{i,j} - \varphi_{i-1,j}}{h_r} \\ &\quad + \min(U_{i,j}^\theta, 0) \cdot \frac{1}{r_{i,j}} \frac{\varphi_{i,j+1} - \varphi_{i,j}}{h_\theta} + \max(U_{i,j}^\theta, 0) \cdot \frac{1}{r_{i,j}} \frac{\varphi_{i,j} - \varphi_{i,j-1}}{h_\theta} \\ &\quad - \frac{1}{r_{i,j} \cdot h_r^2} \left(r_{i+\frac{1}{2},j} \cdot \xi_1^{i+\frac{1}{2},j} \cdot (\varphi_{i+1,j} - \varphi_{i,j}) - r_{i-\frac{1}{2},j} \cdot \xi_1^{i-\frac{1}{2},j} \cdot (\varphi_{i,j} - \varphi_{i-1,j}) \right) \\ &\quad - \frac{1}{r_{i,j} \cdot h_\theta^2} \left(\xi_1^{i,j+\frac{1}{2}} (\varphi_{i,j+1} - \varphi_{i,j}) - \xi_1^{i,j-\frac{1}{2}} (\varphi_{i,j} - \varphi_{i,j-1}) \right),\end{aligned}$$

wobei teilweise zwischen den Netzen geeignet interpoliert werden muss.

3.3 Einbau von Randbedingungen

Nachdem das System (17) bisher im Inneren des Diskretisierungsgebiets betrachtet wurde, sollen an dieser Stelle Randbedingungen für den Rührer besprochen werden. Dazu soll lediglich ein Sechstel des rotationssymmetrischen Rührers betrachtet werden. Der Rand dieses Sechstels teilt sich dann auf in $\partial\Omega =: \Gamma = \Gamma_{wall}^i \cup \Gamma_{wall}^a \cup \Gamma_{stirrer} \cup \Gamma_{per}^1 \cup \Gamma_{per}^2$ (siehe Abbildung 9), wobei zwischen periodischem und physikalischem Rand unterschieden wird. Durch die Vorgabe periodischer Randbedingungen auf $\Gamma_{per}^1 \cup \Gamma_{per}^2$ kann die Simulation leicht auf einen vollen Rührer erweitert werden.

Um Ableitungen in \vec{e}_θ Richtung am Rührerblatt zu realisieren, wird mit so genannten Geisterzellen (engl. *ghost nodes*), wie in [8] vorgeschlagen, gearbeitet. Somit kann unterschieden werden, ob man sich vor oder hinter dem Rührerblatt befindet.

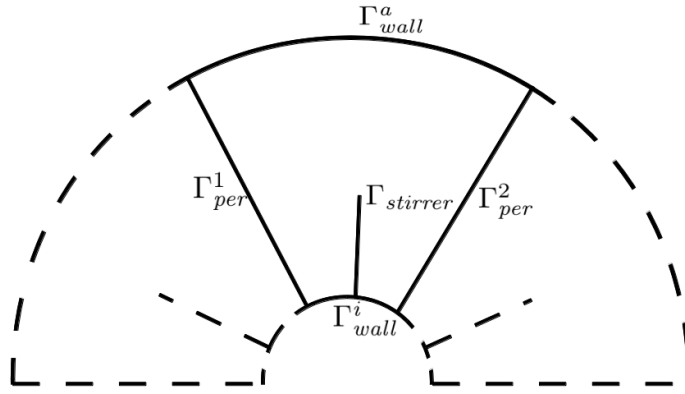


Abbildung 9: Der Rand eines Rührersegments setzt sich zusammen aus physikalischem und periodischem Rand.

3.3.1 Randbedingungen für die Geschwindigkeit

Die Randbedingungen für die Geschwindigkeit sind aus der physikalischen Anschauung bekannt. Auf Grund der Wandhaftbedingung am Rührergehäuse sowie der Drehung des Rührstabs in positive \vec{e}_θ -Richtung gilt für die Komponenten der Geschwindigkeit:

$$U^r = \begin{cases} 0 & \text{auf } \Gamma_{wall}^i, \\ 0 & \text{auf } \Gamma_{wall}^a, \\ 0 & \text{auf } \Gamma_{stirrer}, \end{cases} \quad U^\theta = \begin{cases} \omega r_1 & \text{auf } \Gamma_{wall}^i, \\ 0 & \text{auf } \Gamma_{wall}^a, \\ \omega r & \text{auf } \Gamma_{stirrer}. \end{cases}$$

An dieser Stelle sei bereits darauf hingewiesen, dass auf Grund der speziellen Realisierung der Drehbewegung in Kapitel 3.5 die Randbedingungen für U^θ modifiziert werden müssen.

Die periodischen Randbedingungen auf $\Gamma_{per}^1 \cup \Gamma_{per}^2$ werden durch Kopieren der Geschwindigkeitswerte des jeweils anderen Rands realisiert:

$$\begin{bmatrix} U^r \\ U^\theta \end{bmatrix} \Big|_{\Gamma_{per}^1} = \begin{bmatrix} U^r \\ U^\theta \end{bmatrix} \Big|_{\Gamma_{per}^2}$$

3.3.2 Wandfunktion des k - ϵ Modells

Die Vorgabe von Randbedingungen für das k - ϵ Modell ist schwierig, da das Modell selbst in randnahen Gebieten nicht mehr gültig ist. Auf Grund von kleinen Geschwindigkeiten am Rand fällt die Reynolds-Zahl hier unter den Gültigkeitsbereich des k - ϵ Modells. Die Reynolds-Zahl Re ist eine dimensionslose Größe, die über eine charakteristische Länge und eine charakteristische Geschwindigkeit Aussagen über das Turbulenzverhalten einer Strömung macht. Für den hier betrachteten Versuchsaufbau eines zweidimensionalen Rührers ist in Kapitel 4 die Reynolds-Zahl definiert.

Um Randbedingungen für das k - ϵ Modell vorzugeben wird berücksichtigt, dass in der Nähe von starren Wänden eine so genannte *Grenzschicht* vorliegt, in der die Strömung als laminar angenommen wird. In der Grenzschicht wird die Gültigkeit des k - ϵ Modells durch eine Wandfunktion ersetzt.

In der hier vorgestellten Implementierung wird die Wandfunktion verwendet, um Dirichlet-Randbedingungen für das k - ϵ Modell vorzugeben. Die hier verwendete Wandfunktion wird zum Beispiel in [8, Kap. 10.2.4] eingeführt.

Berechne hierzu zunächst die Schubspannungs-Geschwindigkeit u^* , die in [27] definiert wird via

$$u^* := \sqrt{\nu \left. \frac{\partial \vec{U}_t}{\partial \vec{n}} \right|_{\Gamma}}, \quad (25)$$

wobei \vec{U}_t die Tangentialkomponente der Geschwindigkeit entlang der Wand ist und \vec{n} der Normalenvektor.

Für die Spannungen gilt somit:

$$\left. \frac{\partial \vec{U}_t}{\partial \vec{n}} \right|_{\Gamma} = \begin{cases} \frac{1}{r} \frac{\partial U^r}{\partial \theta}, & \text{falls } \Gamma \text{ in Richtung } \vec{e}_r \text{ zeigt} \\ \frac{\partial U^\theta}{\partial r} - \frac{U^\theta}{r}, & \text{falls } \Gamma \text{ in Richtung } \vec{e}_\theta \text{ zeigt} \end{cases}$$

Berechnet man die Schubspannungs-Geschwindigkeit u^* nach (25), so kann via

$$\delta^* := \frac{\nu}{u^*}$$

ein kritischer Abstand von der Wand berechnet werden.

Empirische und analytische Untersuchungen von [27] haben ergeben, dass die Dirichlet-Randwerte für k und ϵ bestimmt werden können nach:

$$k = (u^*)^2 c_\mu^{-1/2}, \quad \epsilon = \frac{(u^*)^3}{0.41 \delta},$$

wobei $\delta \in [20\delta^*, 100\delta^*]$ gewählt wird, vgl. [8, Kap. 10.2.4].

Aus Randbedingungen für k und ϵ können auch entsprechende Randwerte für $\nu_T = c_\mu \frac{k^2}{\epsilon}$ berechnet werden.

Die Implementierung von Randbedingungen durch Vorgabe einer Wandfunktion ist zum Teil sehr schwierig, weshalb häufig das sogenannte *Niedrige-Reynolds-Zahl* Modell verwendet wird. Bei diesem Ansatz sind die Koeffizienten c_1 , c_2 und c_μ nicht mehr wie in Kapitel 2.2.2 angegeben, als Konstanten zu wählen sondern werden mit ortsabhängigen Funktionen multipliziert. Diese Funktionen berücksichtigen unter anderem den Wandabstand und können [27] entnommen werden. Als Randbedingung kann im Falle des *Niedrige-Reynolds-Zahl* Modells

$$k = 0, \quad \frac{\partial \epsilon}{\partial \vec{n}} = 0$$

entlang starrer Wände gewählt werden, vgl. [8, S. 163].

3.3.3 Randbedingungen für die Momente

Die Vorgabe von Randbedingungen für die Momente ist schwer, da keine physikalische Entsprechung gegeben ist. Da das Verhalten der Momente im Inneren des Diskretisierungsgebiets durch die Differentialgleichung bekannt ist, werden am Rand Null-Neumann Randbedingungen vorgegeben um das Verhalten der Momente möglichst wenig zu beeinflussen. Damit die Randbedingungen zu einer eindeutigen Lösung führen, wird an einem der periodischen Ränder eine Einstrombedingung vorgegeben. Diese wird durch Dirichlet-Randbedingungen mit Werten aus dem vorigen Zeitschritt realisiert. Wählt man am linken Rand in Abbildung 9 eine Einstrombedingung, so kann diese durch

$$m^{(l)} \Big|_{\Gamma_{per}^1}^{(n)} = m^{(l)} \Big|_{\Gamma_{per}^2}^{(n-1)}, \quad l \in \{0, \dots, 2N - 1\}$$

mathematisch umgesetzt werden. Hierbei steht der hochgestellte Index analog zur verwendeten Notation in Kapitel 3.4 für den aktuellen Zeitschritt. Man wählt somit Werte aus dem vorigen als Dirichlet-Randbedingung.

3.4 Zeitintegration

Nachdem in Kapitel 3.2 die Differentialoperatoren des Systems (17) vollständig im Ort diskretisiert wurden, soll an dieser Stelle die Zeitdiskretisierung der RANS-Gleichung sowie der skalaren Transportgleichungen diskutiert werden.

Um eine Funktion $f_{Eul} : \Omega \times [0, T_{end}] \rightarrow \Omega$ in der Zeit zu diskretisieren, wird das Zeitintervall $[0, T_{end}]$ in gleichlange Subintervalle der Form $[n \delta t, (n+1)\delta t]$, $n = 0, \dots, T_{end}/(\delta t) - 1$ unterteilt. Die Euler-Methode approximiert eine zeitliche Ableitung dann durch:

$$\left[\frac{\partial f_{Eul}}{\partial t} \right]^{(n+1)} \approx \frac{f_{Eul}^{(n+1)} - f_{Eul}^{(n)}}{\delta t}.$$

Die hochgestellten Indizes bedeuten im Folgenden also, dass die Funktion f_{Eul} zum Zeitpunkt $t_n = n \cdot \delta t$ ausgewertet wird: $f_{Eul}^{(n)}(\vec{x}) := f_{Eul}(\vec{x}, t = t_n)$. Wenn die restlichen Terme zur Zeit t_n gegeben sind, so kann $f_{Eul}^{(n+1)}$ direkt bestimmt werden. Man spricht von einer expliziten Euler-Zeitintegration (andernfalls: implizit).

3.4.1 Lösen der Druck-Poissongleichung (PPE)

Die RANS-Gleichung kann unter Verwendung der algebraischen Nebenbedingung $\text{div } \vec{U} = 0$ zunächst in eine Poissongleichung für den Druck $P^{(n+1)}$ zum Zeitpunkt t_{n+1} umgeschrieben werden. Nach Lösung der Poissongleichung werden dann Geschwindigkeiten zum Zeitpunkt t_{n+1} aus dem *neuen* Druck berechnet. Dies ist ein gängiger Ansatz zur Zeitintegration der RANS-Gleichung und ist beispielsweise in [8, Kap. 3.2.3] zu finden.

Herleitung der PPE:

Anwendung der expliziten Euler-Methode auf die diskretisierte RANS-Gleichung (19) ergibt:

$$U^{r(n+1)} = \overbrace{U^{r(n)} + \delta t F^{(n)}}^{=: \tilde{F}^{(n)}} - \delta t \frac{\partial P^{(n+1)}}{\partial r}, \quad (26)$$

$$U^{\theta(n+1)} = \overbrace{U^{\theta(n)} + \delta t G^{(n)}}^{=: \tilde{G}^{(n)}} - \delta t \frac{1}{r} \frac{\partial P^{(n+1)}}{\partial \theta}, \quad (27)$$

wobei an dieser Stelle der Druckgradient in Polarkoordinaten ausgeschrieben wird. Fordert man jetzt divergenzfrequ Geschwindigkeit, so ergibt sich:

$$\begin{aligned} 0 &\stackrel{!}{=} \text{div } \vec{U} = \text{div} \begin{bmatrix} \tilde{F}^{(n)} - \delta t \frac{\partial P^{(n+1)}}{\partial r} \\ \tilde{G}^{(n)} - \delta t \frac{1}{r} \frac{\partial P^{(n+1)}}{\partial \theta} \end{bmatrix} \\ &= \frac{1}{r} \left[\frac{\partial}{\partial r} \left(r \cdot \left(\tilde{F}^{(n)} - \delta t \frac{\partial P^{(n+1)}}{\partial r} \right) \right) + \frac{\partial}{\partial \theta} \left(\tilde{G}^{(n)} - \delta t \frac{1}{r} \frac{\partial P^{(n+1)}}{\partial \theta} \right) \right] \\ &= \frac{1}{r} \left[r \cdot \left(\frac{\partial \tilde{F}^{(n)}}{\partial r} - \delta t \frac{\partial^2 P^{(n+1)}}{\partial r^2} \right) + \left(\tilde{F}^{(n)} - \delta t \frac{\partial P^{(n+1)}}{\partial r} \right) + \frac{\partial \tilde{G}^{(n)}}{\partial \theta} - \delta t \frac{1}{r} \frac{\partial^2 P^{(n+1)}}{\partial \theta^2} \right] \\ &= \frac{\partial \tilde{F}^{(n)}}{\partial r} - \delta t \frac{\partial^2 P^{(n+1)}}{\partial r^2} + \frac{1}{r} \tilde{F}^{(n)} - \delta t \frac{1}{r} \frac{\partial P^{(n+1)}}{\partial r} + \frac{1}{r} \frac{\partial \tilde{G}^{(n)}}{\partial \theta} - \delta t \frac{1}{r^2} \frac{\partial^2 P^{(n+1)}}{\partial \theta^2}. \end{aligned}$$

Umsortierten dieser Gleichung ergibt schließlich:

$$\Delta P^{(n+1)} = \frac{1}{\delta t \cdot r} \left(\frac{\partial}{\partial r} (r \cdot \tilde{F}^{(n)}) + \frac{\partial \tilde{G}^{(n)}}{\partial \theta} \right) =: \eta^P. \quad (28)$$

Hierbei ist die rechte Seite explizit gegeben, so dass $P^{(n+1)}$ durch Lösen der Poissonsgleichung $\Delta P = \eta^P$ in jedem Zeitschritt berechnet werden kann. Das Lösen einer Poissonsgleichung ist ein numerisches Standardproblem und wird beispielsweise in [8, 20] besprochen. Zur Lösung der Poissonsgleichung ist noch die Vorgabe von Randbedingungen erforderlich.

Randbedingungen für die PPE:

Nach Multiplikation mit dem äußeren Normalenvektor $\vec{n} = [n_1 n_2]^T$ ergeben sich Neumann-Randbedingungen für den Druck:

$$\begin{aligned} \nabla P^{(n+1)} \cdot \vec{n} &= \frac{\partial P^{(n+1)}}{\partial r} n_1 + \frac{1}{r} \frac{\partial P^{(n+1)}}{\partial \theta} n_2 \\ &\stackrel{(26),(27)}{=} -\frac{1}{\delta t} \left((U^{r(n+1)} - \tilde{F}^{(n)}) n_1 + (U^{\theta(n+1)} - \tilde{G}^{(n)}) n_2 \right). \end{aligned}$$

Es können also Randbedingungen für den Druck direkt aus den Dirichlet-Randbedingungen für die Geschwindigkeit (vgl. Abschnitt 3.3.1) berechnet werden. Die hochgestellte Indizes sind hier nur aus Konsistenzgründen geschrieben. Die Randbedingungen gelten auf dem gesamten Zeitintervall $[0, T_{end}]$.

Nachdem der Druck $P^{(n+1)}$ durch Lösen der Poissonsgleichung (28) bestimmt wird, kann der Gradient des Drucks berechnet werden und nach (26) bzw. (27) eine Aufdatierung der Geschwindigkeiten berechnet werden.

3.4.2 Zeitintegration der skalaren Transportgleichungen

Die skalare Transportgleichung (22) lautet in diskretisierter Form:

$$\frac{\partial}{\partial t} \varphi_{i,j} = -R_{i,j}^\varphi + \xi_2 \cdot \tilde{T}_F^{i,j} + \eta_{i,j}^\varphi.$$

Anwendung der expliziten Euler-Methode ergibt:

$$\varphi_{i,j} \big|_{t=t_{n+1}} = \varphi_{i,j} \big|_{t=t_n} + \delta t \cdot \left(-R_{i,j}^\varphi \big|_{t=t_n} + \xi_2 \cdot \tilde{T}_F^{i,j} \big|_{t=t_{n+1}} + \eta_{i,j}^\varphi \big|_{t=t_n} \right),$$

wobei es aus numerischer Sicht günstig ist, alle Terme, die im Algorithmus schon zum Zeitpunkt t_{n+1} bekannt sind, in impliziter Form in die rechte Seite zu schreiben.

3.5 Realisierung in rotierenden Polarkoordinaten

Um die Drehung des Rührstabs zu simulieren, ist es aus programmiertechnischer Sicht günstiger, den Rührstab als starre Wand zu interpretieren und stattdessen das Koordinatensystem in entgegengesetzte Richtung drehen zu lassen (siehe Abbildung 10). Eine weitere Möglichkeit, das Drehen des Rührstabs algorithmisch umzusetzen ist das in [32] beschriebene *clicking-grids* Verfahren, bei dem ein inneres und ein äußeres Netz gegeneinander verdreht werden. Dieses Verfahren stellt allerdings einen deutlich höheren Programmieraufwand dar.

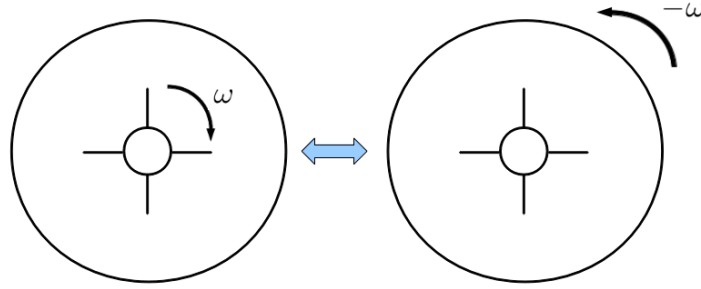


Abbildung 10: Das Drehen des Rührers kann durch ein rotierendes Koordinatensystem modelliert werden.

Das Drehen des Koordinatensystems kann aus mathematischer Sicht durch einer Drehung um die \vec{e}_z -Achse realisiert werden. Dies führt zu einer weiteren Transformation der Geschwindigkeit:

$$\vec{U}_{rot} = \vec{U} - \vec{\omega} \times \vec{r}, \quad \text{mit } \vec{\omega} \times \vec{r} = \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \times \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \omega r \\ 0 \end{bmatrix}, \quad (29)$$

wobei die Auswertung des Kreuzprodukts zeigt, dass eine Drehung um die \vec{e}_z -Achse in Zylinderkoordinaten einer Translation in \vec{e}_θ -Richtung entspricht.

In einem gedrehten Koordinatensystem muss die vektorwertige RANS-Gleichung um drei Terme erweitert werden [40]:

$$\frac{\partial}{\partial t} \vec{U} = - \underbrace{(\vec{U} \cdot \nabla) \vec{U} - \frac{2}{3} \nabla k + \text{div}(\nu^* (\nabla \vec{U} + \nabla \vec{U}^T))}_{=[F, G]^T} - \nabla P - (2 \vec{\omega} \times \vec{U}) - (\vec{\omega} \times (\vec{\omega} \times \vec{r})) - (\dot{\vec{\omega}} \times \vec{r}),$$

wobei der letzte Term bei konstanter Drehgeschwindigkeit verschwindet.

Die neu auftretenden Terme sind zum einen die **Corioliskraft**, die in einem rotierenden Koordinatensystem mit berücksichtigt werden muss. Die Corioliskraft steht sowohl zur Bewegungsrichtung als auch zur Rotationsachse senkrecht und ist gegeben durch:

$$-2 \vec{\omega} \times \vec{U} = -2 \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \times \begin{bmatrix} U^r \\ U^\theta \\ U^z \end{bmatrix} = \begin{bmatrix} 2\omega U^\theta \\ -2\omega U^r \\ 0 \end{bmatrix}.$$

Diese zusätzlichen Terme werden in der Ortsdiskretisierung von F und G mit berücksichtigt.

Eine weitere Kraft, die in einem rotierenden Bezugssystem auftritt, ist die **Zentrifugalkraft**. Die Zentrifugalkraft steht senkrecht auf der Rotationsachse \vec{e}_z und zeigt nach außen. Sie ist gegeben durch:

$$-\vec{\omega} \times (\vec{\omega} \times \vec{r}) = - \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \times \left(\begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \times \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} \right) = \begin{bmatrix} \omega^2 r \\ 0 \\ 0 \end{bmatrix}.$$

Auf Grund der Beziehung $-\nabla(-\frac{1}{2}\omega^2 r^2) = [\omega^2 r, 0, 0]^T$ kann der Anteil der Zentrifugalkraft formal im negativen Druckgradient berücksichtigt werden. Insgesamt erhält man somit:

$$\frac{\partial}{\partial t} \vec{U} = \begin{bmatrix} F_{rot} \\ G_{rot} \end{bmatrix} - \nabla \underbrace{\left(P - \frac{1}{2}\omega^2 r^2 \right)}_{=: P_{rot}}, \quad \text{mit} \quad \begin{bmatrix} F_{rot} \\ G_{rot} \end{bmatrix} := \begin{bmatrix} F + 2\omega U^\theta \\ G - 2\omega U^r \end{bmatrix}.$$

Analog zu der in Abschnitt 3.1.1 beschriebenen Transformation zwischen kartesischen Koordinaten und Polarkoordinaten, ist am Ende der Berechnungen nun eine weitere Rücktransformation notwendig, die das Rotieren des Koordinatensystems berücksichtigt.

Hierfür werden sowohl die θ -Komponenten als auch der Druck via

$$\begin{aligned} U^\theta &= U_{rot}^\theta + \omega r, \\ P &= P_{rot} + \frac{1}{2}\omega^2 r^2, \end{aligned} \tag{30}$$

transformiert.

Auf Grund der Transformation (29) müssen außerdem die in Kapitel 3.3.1 beschriebenen Randbedingungen für die Komponenten der Geschwindigkeit entsprechend angepasst werden:

$$U_{rot}^r = \begin{cases} 0 & \text{auf } \Gamma_{wall}^i, \\ 0 & \text{auf } \Gamma_{wall}^a, \\ 0 & \text{auf } \Gamma_{stirrer}, \end{cases} \quad U_{rot}^\theta = \begin{cases} 0 & \text{auf } \Gamma_{wall}^i, \\ -\omega R & \text{auf } \Gamma_{wall}^a, \\ 0 & \text{auf } \Gamma_{stirrer}. \end{cases}$$

3.6 Algorithmus

Zusammenfassend sei hier der prinzipielle Ablauf der Matlab-Funktion `main.m` in Pseudocode gegeben, wobei die Diskretisierung der Differentialoperatoren in Subfunktionen geschieht.

Algorithmus 1 Ablauf des Programms `main.m` (Pseudocode)

Input: Setzen der Simulationsparameter in `inputfile.m`

Output: Simulationsergebnis im Ordner `output/`

Setze $t := 0, n := 0, \delta t, N$

Initialisiere $U_{rot}^r, U_{rot}^\theta, P_{rot}, k, \epsilon, m_l$

Berechne Geometrie wie in 3.1 beschrieben

while $t < T_{end}$ **do**

 //RANS:

 Setzen der RB für $U_{rot}^{r(n)}$ und $U_{rot}^{\theta(n)}$ wie in Kapitel 3.3.1 bzw. 3.5

 Berechne $\tilde{F}^{(n)}$ und $\tilde{G}^{(n)}$ nach Gleichung (20) bzw. (21)

 Aufstellen der rechten Seite η^P nach (28)

 Löse die PPE: $\Delta P_{rot}^{(n+1)} = \eta^P$

 Berechne $\nabla P_{rot}^{(n+1)}$

 Berechne $U_{rot}^{r(n+1)}$ und $U_{rot}^{\theta(n+1)}$ nach (26) bzw. (27)

 //k-eps Modell:

 Setzen der RB für $k^{(n)}$ und $\epsilon^{(n)}$ durch Auswertung der Wandfunktion nach 3.3.2

 Berechne Frobenius Norm des Spannungstensors $\tilde{T}_F^{(n+1)}$ nach (23)

$rhs^k := -R^k + \frac{\nu_T^{(n)}}{2} \cdot \tilde{T}_F^{(n+1)} - \epsilon^{(n)}$, mit R^k nach (24)

$rhs^\epsilon := -R^\epsilon + \frac{c_1}{2} k^{(n)} \cdot \tilde{T}_F^{(n+1)} - c_2 \frac{\epsilon^{(n)} \epsilon^{(n)}}{k^{(n)}}$, mit R^ϵ nach (24)

$k^{(n+1)} := k^{(n)} + \delta t \cdot rhs^k$

$\epsilon^{(n+1)} := \epsilon^{(n)} + \delta t \cdot rhs^\epsilon$

 //Momentengleichung:

 Setzen der RB für $m_l^{(n)}$ wie in Kapitel 3.3.3

 Berechnung der ortsabhängigen Quellterme $S^{(l)}$ nach (15)

for $l=1:2N$ **do**

$rhs^{m_l} := -R^{m_l} + S^{(l)}$, mit R^{m_l} nach (24)

end for

for $l=1:2N$ **do**

$m_l^{(n+1)} := m_l^{(n)} + \delta t \cdot rhs^{m_l}$

end for

$t := t + \delta t$

$n := n + 1$

end while

Rücktransformation nach (18) und (30) sowie Berechnung von d_{32} und σ nach (16)

Postprocessing, Visualisierung

4 Numerische Tests

Zur Validierung der Implementierung sollen an dieser Stelle einige numerische Testrechnungen vorgestellt und physikalisch interpretiert werden. Hierfür ist es zunächst notwendig, verschiedene Parameter für die Simulation zu setzen. Die Abmaße eines typischen Rührers sowie Stoffparameter können Tabelle 1 entnommen werden. Diese werden im `inputfile` zu Beginn einer Simulation gesetzt (siehe Anhang C.1).

Parameter	Wert
Radius des Rührergehäuses	$R = 0.075m$
Innenradius	$r_1 = 0.02m$
Radius der Rührblätter	$r_2 = 0.04m$
Drehgeschwindigkeit	$\omega = 0.5 \frac{1}{s}$
Geschwindigkeit an Spitze des Rührblatts	$u_{tip} = 0.02 \frac{m}{s}$
Kinematische Viskosität	$\nu_{Wasser} = 10^{-6} \frac{m^2}{s}$
Zeitschrittweite	$\delta t = 10^{-6}s$
Simulationszeit	$T_{end} = 0.0015s$
Diskretisierung eines Rührersegments	100 x 150

Tabelle 1: Setzen verschiedener Parameter für die Simulation.

Um eine Vergleichbarkeit der Ergebnisse unabhängig von der Dimension des Rührers zu erhalten, wird üblicherweise die Reynolds-Zahl definiert. Hierfür wird eine charakteristische Länge sowie eine charakteristische Geschwindigkeit benötigt. Für das Problem des zweidimensionalen Rührers wird in der Literatur einheitlich folgende Reynolds-Zahl verwendet:

$$Re := \frac{\omega \cdot D^2}{\nu}, \quad \text{mit } D := 2r_2,$$

so dass als charakteristische Geschwindigkeit die Drehgeschwindigkeit des Rührers gewählt wurde und als charakteristische Länge der Durchmesser des Rührstabs. Die Reynolds-Zahl ist somit dimensionslos. Mit den in Tabelle 1 gewählten Parametern ergibt sich somit $Re = 3.200$ für den ersten Testfall bei einer maximalen Geschwindigkeit an der Spitze der Rührblätter von $u_{tip} = \omega \cdot r_2 = 0.02 \frac{m}{s}$.

Für die nachfolgende Visualisierung wurde PARAVIEW [13] verwendet.

4.1 Simulationsergebnisse für $Re = 3.200$

An dieser Stelle werden Simulationsergebnisse im niedrigen Reynoldszahlbereich zu verschiedenen Zeitpunkten präsentiert. Hierfür wurde die Simulation mit konstanter Drehgeschwindigkeit $\omega = 0.5 \frac{1}{s}$ durchgeführt, was 30 Umdrehungen pro Minute entspricht. Auf Grund des sehr kleinen Zeitschritts wurde ein Simulationsdauer von $0.0015s$ gewählt.

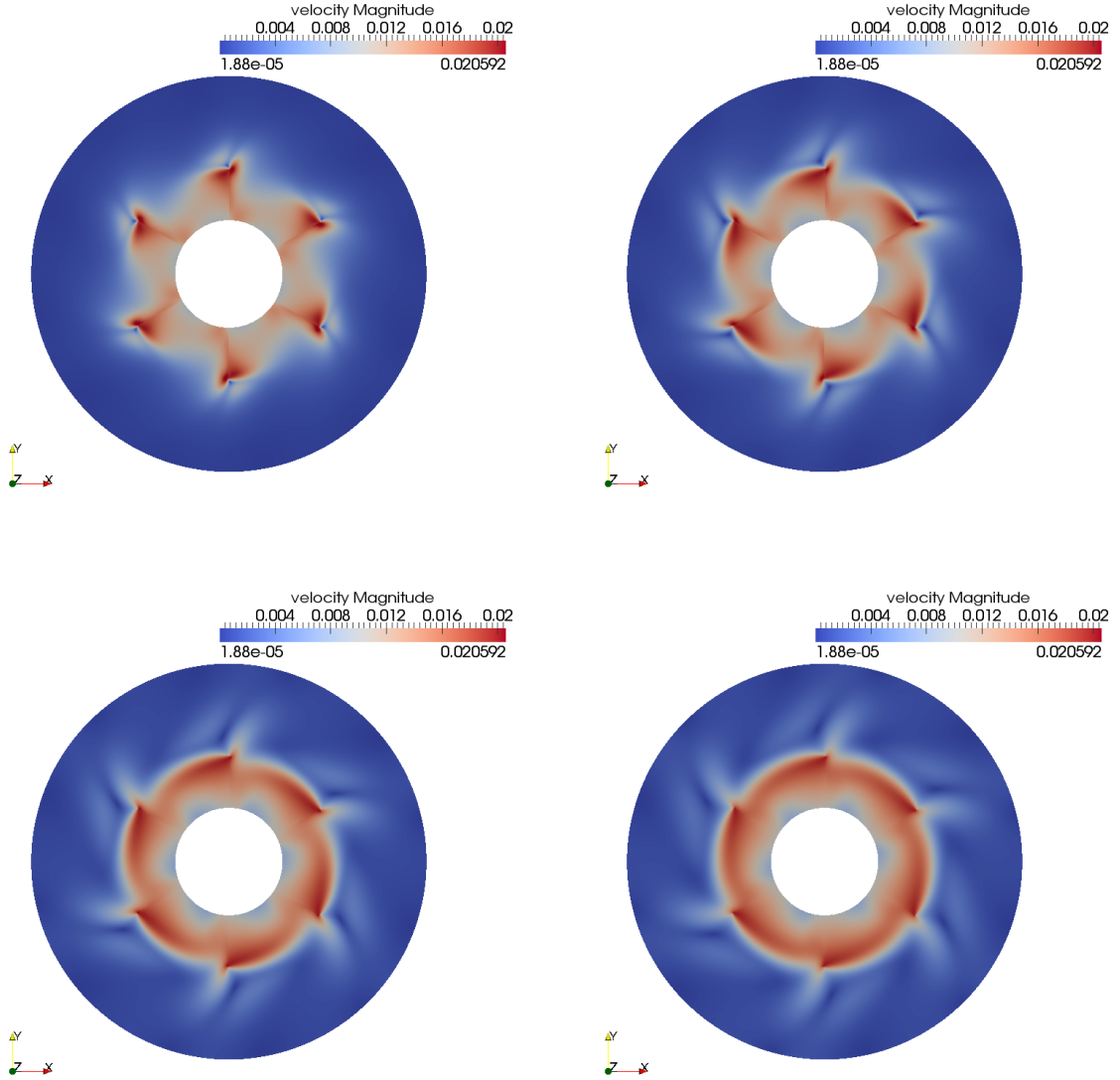


Abbildung 11: Betrag der Geschwindigkeit zu den Zeitpunkten $t_1 = 0.0003s$, $t_2 = 0.0006s$, $t_3 = 0.00105s$ und $t_4 = 0.0015s$ bei einer konstanten Drehgeschwindigkeit von $\omega = 0.5 \frac{1}{s}$

Die numerischen Ergebnisse in Abbildung 11 zeigen zunächst, dass die für die Geschwindigkeit vorgegebenen Dirichlet-Randbedingungen korrekt eingehalten werden: Am Rührergehäuse liegt wegen der Wandhaftbedingung keine Geschwindigkeit vor und auf Grund der Drehgeschwindigkeit von $\omega = 0.5$ Umdrehungen pro Sekunde liegt an der Spitze der Rührblätter eine maximale Geschwindigkeit von $u_{tip} = 0.02 \frac{m}{s}$ vor.

Darüber hinaus ist das zeitliche Verhalten der Strömung innerhalb des Rührbehälters gut zu erkennen. Zu Beginn liegt mit Ausnahme der direkten Umgebung der Rührblätter keine Geschwindigkeit vor. Nach einiger Simulationszeit induziert die Bewegung des Rührstabs eine Bewegung des Fluids. Beim Auftreffen der Rührblätter auf das Fluid sind im Wesentlichen zwei Effekte zu beobachten: Zum einen entsteht durch das Verdrängen des Fluids vor dem

Rührblatt eine Geschwindigkeit in radiale Richtung und zum anderen ist eine Rückströmung hinter dem Rührblatt zu erkennen. Insbesondere die radiale Geschwindigkeit ist für die Vermischung während des Rührvorgangs verantwortlich.

4.2 Simulationsergebnisse für $Re = 12.800$

Um in den Gültigkeitsbereich des k - ϵ Modells zu kommen, aber auch aus Sicht der Anwendung, ist man an deutlich höheren Reynolds-Zahlen interessiert. Dies führt allerdings auch zu numerischen Instabilitäten, weshalb es sich als notwendig erwiesen hat, die Drehgeschwindigkeit des Rührstabs langsam *hochzufahren*. Wie Abbildung 12 zu entnehmen ist, wurde die Simulation in den hier präsentierten Testfällen mit einer Drehgeschwindigkeit von $\omega = 0.5 \frac{1}{s}$ gestartet und linear auf $\omega = 2 \frac{1}{s}$ erhöht. Dies entspricht einer maximalen Drehgeschwindigkeit von 120 Umdrehungen pro Minute. Dabei ist insbesondere zu beachten, dass sich die RANS-Gleichung im gedrehten Koordinatensystem um einen nicht verschwindenden Term $\vec{\omega} \times \vec{r}$ erweitert (vgl. Kapitel 3.5). Die restlichen Parameter blieben im Vergleich zu 4.1 unverändert.

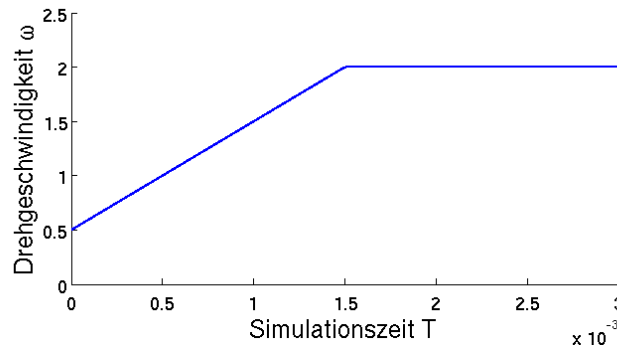


Abbildung 12: Lineares Hochfahren der Drehgeschwindigkeit ω auf maximal $\omega = 2 \frac{m}{s}$

In Abbildung 13 ist erneut das Geschwindigkeitsfeld im Rührer zu verschiedenen Zeitpunkten geplottet. Im Vergleich zu den numerischen Ergebnissen bei $Re = 3.200$ ist festzustellen, dass

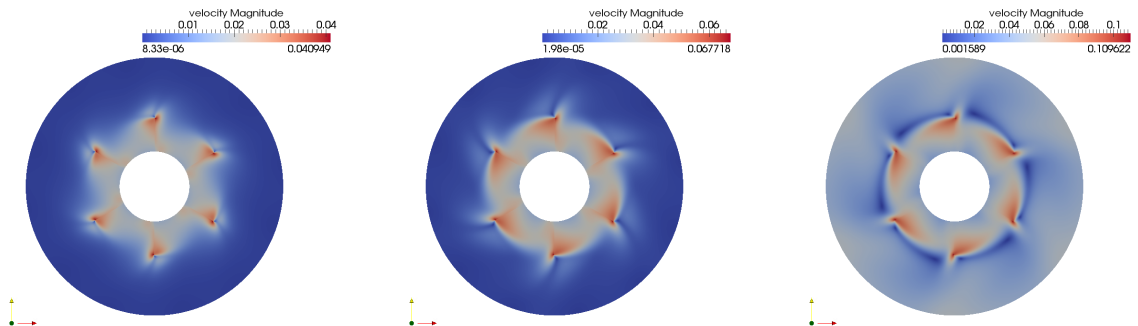


Abbildung 13: Betrag der Geschwindigkeit zu den Zeitpunkten $t_1 = 0.0003s$, $t_2 = 0.0006s$ und $t_3 = 0.0021s$, wobei die Drehgeschwindigkeit auf maximal $\omega = 2 \frac{1}{s}$ erhöht wurde.

sich bei höheren Reynolds-Zahlen insbesondere größere Radialgeschwindigkeiten am Rührstab entwickeln, die durch Konvektion zu höheren Geschwindigkeiten im Bereich des Rührergehäuses führen.

In Abbildung 14 sind außerdem Ergebnisse für k und ϵ , die durch Lösen der Transportgleichungen des k - ϵ Modells berechnet wurden, grafisch dargestellt. Es ist zu erkennen, dass sich beide Größen erst im Bereich höherer Reynolds-Zahlen ausbreiten.

Da beide Größen zunächst keine physikalische Entsprechung haben, ist hier die weitere Validierung durch Vergleich mit Messwerten erforderlich. Es ist dennoch festzustellen, dass sich Turbulenz offenbar zuerst im Bereich der Rührblätter ausbreitet.

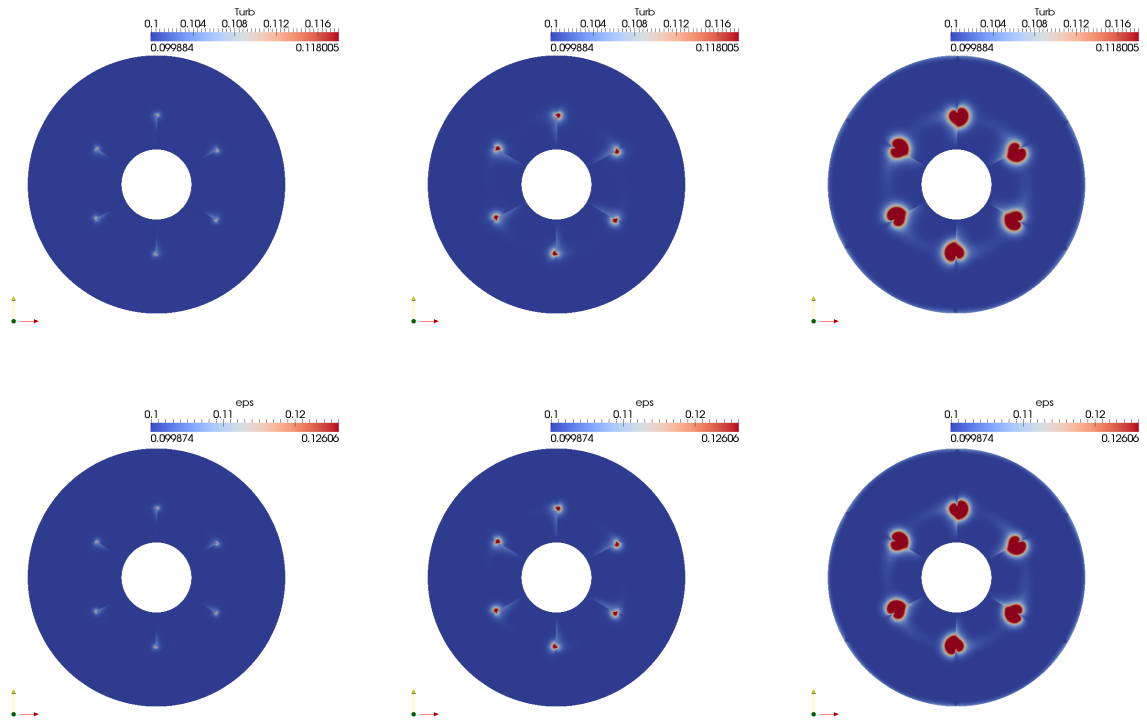


Abbildung 14: Werte der turbulenten kinetischen Energie k (oben) sowie der Dissipationsrate ϵ (unten) zu den Zeitpunkten $t_1 = 0.0003s$, $t_2 = 0.0006s$ und $t_3 = 0.0021s$

5 Zusammenfassung und Ausblick: Steuerung der Drehgeschwindigkeit ω

In dieser Bachelorarbeit wurde zunächst in Kapitel 2 eine mathematisch vollständige Beschreibung des Strömungsverhaltens eines Gemischs aus einer dispersen und einer kontinuierlichen Phase in turbulenter Strömung hergeleitet. Hierzu wurde die Navier-Stokes Gleichung hergeleitet und durch Anwendung eines Filter-Operators zur Reynolds-Averaged-Navier-Stokes (RANS) Gleichung erweitert, die das Strömungsverhalten in turbulenter Strömung modelliert. Um die RANS-Gleichung abzuschließen wurde das k - ϵ Turbulenzmodell eingeführt, das aus zwei skalaren Transportgleichungen für die turbulente kinetische Energie k und dessen Dissipationsrate ϵ besteht.

Zur mathematischen Beschreibung der dispersen Phase wurde ebenfalls eine Populations-Bilanz Gleichung (PBE) für die sogenannte Anzahl-Dichtefunktion q entwickelt. Diese beschreibt die mittlere Partikelgröße zu festem Ort und Zeit und gibt somit die Verteilung der Tropfen der dispersen Phase innerhalb des Gemischs an. Zur Lösung der PBE wurde die Quadratur-Momenten-Methode (QMOM) vorgestellt und implementiert. Dabei wird die PBE in ihren Momenten gelöst und in der rechten Seite ein Tropfenmodell nach Coulaloglou und Tavlarides verwendet, das den Zerfall und die Koaleszenz von Tropfen während des Mischvorgangs beschreibt. Diese Quellterme wurden in Abschnitt 2.3.2 diskretisiert.

Insgesamt wurde ein System aus fünf gekoppelten partiellen Differentialgleichungen hergeleitet, dass sowohl das turbulente Strömungsverhalten des Gemischs als auch die Verteilung der dispersen Phase modelliert.

Darauf aufbauend wurde ein Algorithmus entwickelt, der die zuvor hergeleiteten Gleichungen, angewandt auf einen vereinfachten zweidimensionalen Rührer, numerisch löst. Dazu wurde zunächst ein Gittergenerierer entwickelt und die verwendeten Gleichungen in Polarkoordinaten formuliert. In Abschnitt 3.2 ist mit Hilfe der Finite-Differenzen Methode eine vollständige Ortsdiskretisierung der RANS-Gleichung sowie der skalaren Transportgleichungen in Polarkoordinaten gegeben. Für die Zeitintegration beider Gleichungen wurde im Falle der RANS-Gleichung die sogenannte Druck-Poissongleichung hergeleitet und numerisch gelöst und im Falle der skalaren Transportgleichungen die explizite EULER-Methode verwendet.

Darüber hinaus wurden in Kapitel 3 Randbedingungen für das spezielle Problem des zweidimensionalen Rührers vorgegeben und eine Methode präsentiert, mit deren Hilfe das Drehen des Rührstabs algorithmisch realisiert werden kann.

Die numerischen Tests in Kapitel 4 zeigen für das Strömungsverhalten des Gemischs sehr gute Ergebnisse für Simulationen im kleinen Reynoldszahlbereich. Diese wurden in dem präsentierten Testfall bei einer konstanten Drehgeschwindigkeit von $\omega = 0.5 \frac{1}{s}$ und $Re = 3.200$ erreicht.

Um größere Reynolds-Zahlen zu erreichen ist es notwendig die Drehgeschwindigkeit des Rührers langsam zu steigern. Dennoch treten numerische Instabilitäten bei der Berechnung auf. Diese machen es notwendig mit zum Teil sehr kleinen Zeitschritten und einer feineren Ortsdiskretisierung zu arbeiten.

In anschließenden Arbeiten wäre daher zunächst zu evaluieren, in wie fern beispielsweise eine implizite Zeitintegration oder der Einsatz von Mehrschritt-Verfahren zu einem stabileren Al-

gorithmus führt und gleichzeitig die Wahl eines größeren Zeitschritts ermöglicht. Dies ist aus Sicht der Anwendung wünschenswert, da sich so effizienter längere Simulationszeiten erreichen lassen, so dass mehrere Umdrehungen des Rührstabs simuliert werden können. Hierfür sind dann auch erste Ergebnisse für die Verteilung des Sauter-Durchmessers und dessen Varianz zu erwarten, die im Rahmen dieser Bachelorarbeit zwar berechnet wurden, aber keine zeitliche Entwicklung zeigten.

Eine weitere Anwendung des entstandenen Strömungslösers, die auch Gegenstand aktueller Forschung ist, liegt in der Kontrolltheorie [26]. Aus ingenieurwissenschaftlicher Sicht ist es in vielen Anwendungen wünschenswert, einen bestimmten Sauter-Durchmesser \hat{d}_{32} bei kleiner Varianz σ im Rührer zu erhalten. Fasst man diese beiden Größen als Ausgangsgrößen des Rührsystems auf, so kann man analog die Drehgeschwindigkeit ω als Eingangsgröße verstehen. Gesucht ist also eine Steuerung für die Drehgeschwindigkeit ω , so dass im Grenzfall folgendes gilt:

$$\begin{bmatrix} d_{32} \\ \sigma \end{bmatrix} \rightarrow \begin{bmatrix} \hat{d}_{32} \\ 0 \end{bmatrix}.$$

Das soeben beschriebene Kontrollproblem ist in Abbildung 15 grafisch dargestellt. In einer weiterführenden Arbeit ist also die Entwicklung effizienter Kontrollalgorithmen sowie der Entwurf eines Reglers zur Kontrolle des Sauter-Durchmessers durch Vorgabe der Drehgeschwindigkeit zu untersuchen. Die Arbeit von [10] hat hierbei gezeigt, dass die Kopplung eines Strömungslösers mit einer Kontroll-Einheit möglich ist.

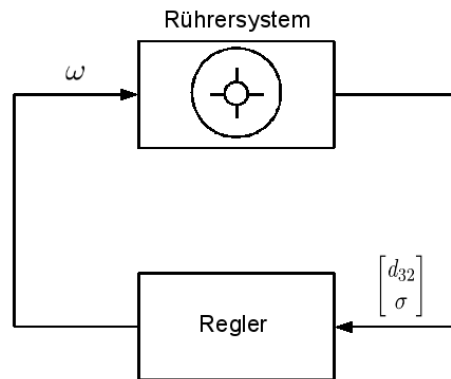


Abbildung 15: Steuerung der Drehgeschwindigkeit ω

A Mathematischer Anhang

An dieser Stelle seien aus Gründen der Vollständigkeit zwei zentrale Sätze der mehrdimensionalen Analysis angegeben, die für die Herleitung in Kapitel 2 verwendet werden. Die Sätze sind den Standardwerken [15, 42] entnommen.

A.1 Der Gauß'sche Integralsatz

Es sei $\Omega \subset \mathbb{R}^3$ kompakt mit abschnittsweise glattem Rand $\partial\Omega$, der Rand sei orientiert durch ein äußeres Normalen-Einheitsfeld \vec{n} . Ferner sei die vektorwertige Funktion $\vec{f} : \Omega \rightarrow \Omega, \vec{x} \mapsto \vec{f}(\vec{x})$ stetig differenzierbar. Nach [15, Kap. 12, Satz 12.4] gilt dann die folgende Beziehung:

$$\int_{\Omega} \operatorname{div} \vec{f} \, dV = \int_{\partial\Omega} \vec{f} \cdot \vec{n} \, dS.$$

A.2 Das Reynolds'sche Transporttheorem

Das Reynolds'sche Transporttheorem ist zum Beispiel in [42, Satz 1.3.1] zu finden und besagt, dass für eine differenzierbare skalare Funktion $f : \Omega \times [0, T_{end}] \rightarrow \mathbb{R}, (\vec{x}, t) \mapsto f(\vec{x}, t)$, die sowohl vom Ort als auch von der Zeit abhängt, gilt:

$$\frac{d}{dt} \int_{\Omega} f(\vec{x}, t) \, dV = \int_{\Omega} \left(\frac{\partial}{\partial t} f + \operatorname{div}(f \vec{u}) \right) (\vec{x}, t) \, dV.$$

B Der PD-Algorithmus

Der in Kapitel 2.3.1 erwähnte PD-Algorithmus zur Bestimmung von Abszissen und Momenten aus vorgegebenen Momenten ist der Arbeit [24] entnommen und wird hier als Pseudocode zur Vervollständigung angegeben.

Algorithmus 2 Der PD-Algorithmus (Pseudocode)

Input: Momente $m^{(1)}, \dots, m^{(2N)}$

Output: Gewichte $\omega_1, \dots, \omega_N$ und Abszissen ξ_1, \dots, ξ_N

1.Schritt: Konstruktion der Matrix $P \in \mathbb{R}^{2N+1, 2N+1}$

Erste Spalte: $P[i, 1] := \delta_{i,1}, \quad i \in 1, \dots, 2N+1$

Zweite Spalte: $P[1, 2] := m^{(1)}$ und $P[i, 2] := (-1)^{i-1} m^{(i)}, \quad i \in 2, \dots, 2N+1$

for $j=3:2N+1$ **do**

for $i=1:2N+2-j$ **do**

$P[i, j] := P[1, j-1] * P[i+1, j-2] - P[1, j-2] * P[i+1, j-1]$

end for

end for

2.Schritt: Berechnung der Koeffizienten α_i

Setze $\alpha_0 := 0$

Berechne $\alpha_i := \frac{P[1, i+1]}{P[1, i] \cdot P[1, j-1]}, \quad i \in 2, \dots, 2N+1$

3.Schritt: Konstruktion der Tridiagonalmatrix $T \in \mathbb{R}^{N, N}$

for $i=1:N$ **do**

$a[i] := \alpha_{2i-1} \cdot \alpha_{2i}$

end for

for $i=1:N-1$ **do**

$b[i] := \sqrt{\alpha_{2i} \cdot \alpha_{2i+1}}$

end for

$T := \text{diag}(a, 0) + \text{diag}(b, 1) + \text{diag}(b, -1)$

4.Schritt: Berechnung von ω_i und ξ_i aus Eigenpaaren von T

Berechne alle Paare $(\lambda_i, v_i) \in \mathbb{R} \times \mathbb{R}^N : Tv_i = \lambda_i v_i, \quad i \in 1, \dots, N$

return $\xi_i := \lambda_i, \quad \omega_i := m^{(1)} \cdot v_i[1]^2$

Die Korrektheit des Algorithmus' kann leicht überprüft werden, indem man aus vorgegebenen Momenten die Abszissen und Gewichte nach dem PD-Algorithmus berechnen und anschließend, gemäß Formel (14), die vorgegebenen Momente reproduziert.

C Kleines Solver-Handbuch

Dieser Bachelorarbeit ist eine CD mit dem kompletten Quellcode in MATLAB R2010B beigelegt. Um die Benutzung des Programms besser zu verdeutlichen, ist an dieser Stelle das Inputfile sowie die Datei `main.m` angehängen. In dem Inputfile können alle Parameter gesetzt werden, die die Simulation beeinflussen. Dazu gehören sowohl Stoffwerte der simulierten Flüssigkeit als auch Geometriedaten und Netzparameter des Rührers sowie verschiedene Simulationsparameter. Nachdem im Inputfile alle Werte gesetzt sind, kann eine Simulation durch Eingabe von `»main` in der Matlab-Konsole gestartet werden. Die Datei `main.m` ist im Wesentlichen so aufgebaut wie in Kapitel 3.6 beschrieben. Es ist zu beachten, dass in der Datei `gohome.m` der aktuell Verzeichnispfad gesetzt werden muss. Der entstandene Output liegt im Verzeichnis `/output` und kann mit dem Programm PARAVIEW visualisiert werden. Auf der CD ist das SetUp aus Kapitel 4.1 eingestellt.

C.1 inputfile.m

```
1  %% simulation
2  restart = 0;           % new calculation (0) or restart with saved data (1)
3
4  %% turbulent modelling
5  cmu=0.09;              % parameter of the k-eps model
6  ceps=0.07;             % parameter of the k-eps model
7  c1=0.126;              % parameter of the k-eps model
8  c2=1.92;               % parameter of the k-eps model
9
10 %% geometry params
11 R=0.15/2;               % radius of the tank
12 r2=0.08/2;              % radius of the stirrer
13 r1=0.04/2;              % radius of the hub
14
15 %% spatial discretization
16 ZS=6;                   % # Segments (1-6)
17 Nr = 100;                % # cells in r-direction
18 Nphi=150;                % # cells/segment in theta direction
19 plotOn = 0;              % plot or no plot
20 invTrafo = 1;            % visualization in cartesian (1) or polar (0) coordinates
21
22 %% temporal discretization
23 tE = 0.0015;             % end time
24 kEnd=150;                % # inner iterations
25 nClk = 1;                % # of clicks per timestep
26 small=10^-15;            % to avoid deviding by zero
27
28 %% pressure iteration
29 tolPPE = 1e-3;           % tolerance for the relative error in the PPE iteration
30 itPPE = 20000;            % max # iterations in the PPE
31 relPar = 1.5;             % relaxation parameter
32
33 %% flow properties
34 nu = 1e-6;                % kin. viscosity of the mixture (water)
35 phi = 0.1;                % disperse phase rate
36
```

```

37 %% drop size simulation
38 QMOMon = 0; % drop size simul on (1) or off (0)
39 phid=0.1; % disperse phase rate
40 dpmax=10^-3; % maximum drop diameter
41 Nmom=2; % compute first 2*Nmom moments
42 rho=1000; % density (of water)
43 sigg=32e-3; % interfacial tension
44 mue=1e-3; % dyn. viscosity of the continious phase
45 clb=6.14e-4; % free params of the
46 c2b=5.70e-2; % drop breakage and coalescence
47 clc=1.50e-4; % model by Coualaloglou
48 c2c=2.56e12; % and Tavlarides
49 saut0=200*10^-6; % initial value for Sauter diameter
50 sig0=100*10^-6; % initial value for sigma
51
52
53 %% derived quantities
54 Lphi = pi/3*ZS; % simulation length
55
56 gohome, cd data
57
58 save meshParams.mat ZS Lphi R r2 r1 Nr Nphi plotOn invTrafo
59 save ppeParams.mat tolPPE itPPE relPar
60 save timestepParams.mat tE kEnd nClk small Nmom QMOMon
61 save simParams.mat restart cmu ceps c1 c2
62 save flowParams.mat nu phi
63 save dsdParams.mat phid dpmax Nmom rho sigg mue clb c2b clc c2c saut0 sig0

```

C.2 main.m

```

1 gohome;
2 clear all;
3 setpath;
4
5 %% Init
6 fprintf('Reading input data... \n');
7 w0 = 0.5; % the stirrer velocity
8 dt = 0.0000001; % defines the timestep
9 wDot = 1000; % accelaration of the stirrer
10 inputfile; % reads input params
11 initGeo; % mesh generation
12 [uR,uTh,p,tke,tDiss,saut0,sig0,rhsMom,mom,t,swt,RES] = rotInitSim(w0);
13 setGeo(swt);
14
15 %% outsourcing for speed-up
16 fprintf('Build constant matrices for speed-up... \n');
17 APpe = getAPpe; % constant Laplace matrix
18 SbreakData=InitSbreak(); % create lookup-table for integral term in S_break
19
20 fprintf('\n\nStart time loop \n');
21 load timestepParams
22 for part = 1:10
23     for k = 1:kEnd
24         fprintf('Zeitschritt: t=%4d s \n',t);
25         [w,Re] = stirrerSpeed(w0,t,wDot,r2,R,nu);

```

```

26
27 wAcc = wAccUTh(wDot); %acceleration due to changes in rotational speed
28 [uR,uRGB,uRGU,uTh] = rotSetVelBcs(uR,uTh,w);
29 [uRHat,uThHat,~,~]=rotCompUHat(uR,uTh);
30
31 %% RANS:
32 [kF,kG]=upwindKonvVel(uR,uRHat,uTh,uThHat,uR,uRGU,uRGB,uRHat,uTh,uThHat);
33 [gTkeF,gTkeG]=gradk(tke);
34 [ddF,ddG]=turbDiffVel(uR,uTh,tke,tDiss);
35
36 F=uR+dt*((-2/3)*gTkeF+ddF+2*w*uThHat-kF);
37 G=uTh+dt*((-2/3)*gTkeG+ddG-wAcc-2*w*uRHat-kG);
38
39 [Fb,Gb]=setBcsFG(F,G,uR,uTh);
40
41 % Solve pressure Poisson equation
42 rhsPP = rhsPPE(Fb,Gb);
43 [p,res] = solPpe(p,APpe,1/dt*rhsPP);
44 RES=[RES res];
45 [gPR,gPTh] = gradP(p);
46
47 uR = F-dt*gPR;
48 uTh = G-dt*gPTh;
49
50 [uR,uRGB,uRGU,uTh] = rotSetVelBcs(uR,uTh,w);
51
52 %% k-eps turbulence model:
53 [uRHat,uThHat,uThHatGB,uThHatGU]=rotCompUHat(uR,uTh); %update: [uR uTh]^k+1
54
55 tDiss(tDiss<small)=small;
56 nuT=cmu*tke.^2./tDiss;
57 tke(tke<small)=small;
58
59 % boundary conditions
60 [tkeR1,tkeR2,tkeGB,tkeGU,tDissR1,tDissR2,...
61 tDissGB,tDissGU,nuTR1,nuTR2,nuTGB,nuTGU]=wallFunction(uR,uTh,nu,cmu,tke,tDiss);
62
63 strTens=stressTensor(uR,uTh,uRHat,uThHat);
64 % spatial discretisation
65 rhsk = -KonvScal_keps(tke,uR,uTh,tkeR1,tkeR2,tkeGB,tkeGU)+(nuT/2).*strTens+...
66 DivGrad_keps(nuT,nuTR1,nuTR2,nuTGB,nuTGU,tke,tkeR1,tkeR2,tkeGB,tkeGU)-...
67 tDiss;
68 rhstDiss = -KonvScal_keps(tDiss,uR,uTh,tDissR1,tDissR2,tDissGB,tDissGU)+...
69 (c1/2)*tke.*strTens+DivGrad_keps(ceps/cmu*nuT,ceps/cmu*nuTR1,...
70 ceps/cmu*nuTR2,ceps/cmu*nuTGB,ceps/cmu*nuTGU,tDiss,tDissR1,tDissR2,tDissGB,tDissGU)-
71 c2*tDiss.^2./tke;
72
73 % explicite Euler method
74 tke = tke + dt*rhsk;
75 tDiss = tDiss + dt*rhstDiss;
76
77 %% QMOM:
78 if QMOMon == 1 && k==kEnd
79 [xi,omega]=PDalgoInSpace(mom); % mom --> xi, omega
80
81 S=SourceTerms(xi,omega,SbreakData,tDiss);
82

```

```

83         % spatial discretisation
84         tDiss(tDiss<small)=small;
85         nuT=cmu*tke.^2./tDiss; %update: nuT^k+1
86         for kkk=1:2*Nmom
87             rhsMom(:, :, kkk)=S(:, :, kkk)-DivScalMom(uR,uTh,mom(:, :, kkk))+...
88                 DivGradMom(nuT,nuTR1,nuTR2,nuTGB,nuTGU,mom(:, :, kkk));
89         end
90
91         % explicite Euler method
92         for kkk=1:2*Nmom
93             mom(:, :, kkk)=mom(:, :, kkk)+kEnd*dt*rhsMom(:, :, kkk);
94         end
95     end
96
97     t = t+dt;
98 end
99
100 %% Postprocessing + plots
101 [uThInit,pInit] = invRotVar(uTh,p,w);
102 writeOutput(uR,uThInit,pInit,part);
103 addScal2plot(tke,part,0);
104 addScal2plot(tDiss,part,1);
105 if QMOMon == 1
106     [saut,sig]=invMomToSautSig(mom);
107     addScal2plot(saut,part,2);
108     addScal2plot(sig,part,3);
109 end
110
111 gohome;
112 cd data
113 save restart.mat uR uTh p t w tke tDiss swt RES saut sig mom rhsMom
114 fprintf('%4d %% of interations completed \n', 10*part);
115 fprintf('Re number: Re=%i \n\n',Re);
116 end
117 gohome;

```

Literatur

- [1] G. Bärwolff. *Numerik für Ingenieure, Physiker und Informatiker*. Spektrum Akademischer Verlag, Heidelberg, 2006.
- [2] A. J. Chorin und J. E. Marsden. *A Mathematical Introduction to Fluid Mechanics*. Springer-Verlag, New York/Berlin/Heidelberg/Tokyo, 1979.
- [3] C. A. Coulaloglou und L. L. Tavlarides. Description of Interaction Processes in Agitated Liquid/Liquid Dispersions. *Chem. Eng. Sci.*, 32:1289–1297, 1977.
- [4] C. Drumm, M. Attarakih, M.W. Hlawitschka und H.-J. Bart. One-Group Reduced Population Balance Model for CFD Simulation of a Pilot-Plant Extraction Column. *Ind. Eng. Chem. Prod. Rd.*, 49(7):3442–3451, 2010.
- [5] J.H. Ferziger und M. Perić. *Numerische Strömungsmechanik*. Springer-Verlag, Berlin/Heidelberg, 2008.
- [6] A. Gäbler, M. Wegener, A.R. Paschedag und M. Kraume. The effect of pH on experimental and simulation results of transient drop size distributions in stirred liquid-liquid dispersions. *Chem. Eng. Sci.*, 61:3018–3024, 2006.
- [7] A. Gerslauer. *Herleitung und Reduktion populationsdynamischer Modelle am Beispiel der Flüssig-Flüssig-Extraktion*. VDI Verlage, Düsseldorf, 1999.
- [8] M. Griebler, T. Dornseifer und T. Neunhoffer. *Numerical Simulation in Fluid Dynamics - A Practical Introduction*. SIAM, Philadelphia, 1998.
- [9] Z. Harth. *Automated Numerical Shape Optimization of 3-dimensional Flow Geometry Configurations*. Dissertation, TU Darmstadt, Fachgebiet für numerische Berechnungsverfahren im Maschinenbau, 2008.
- [10] J. Heiland, A. Walle, V. Mehrmann und M. Schäfer. Simulation and control of drop size distributions in stirred liquid/liquid systems. In *4th International Conference on Population Balance Modelling*, Berlin, Germany, 2010. Herausgegeben als USB-Stick.
- [11] H.M. Hulburt und S. Katz. Some problems in particle technology: A statistical mechanical formulation. *Chem. Eng. Sci.*, 19(9):555–574, 1964.
- [12] V. John, I. Angelov, A.A. Öncül und D. Thévenin. Techniques for the reconstruction of a distribution from a finite number of its moments. *Chem. Eng. Sci.*, 62:2890–2904, 2007.
- [13] Kitware. *ParaView Version 3.6.1*. www.paraview.org.
- [14] A.N. Kolmogorov. The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers. *Proc. R. Soc. Lond. A*, 434:9–13, 1991.
- [15] K. Königsberger. *Analysis 2*. Springer-Verlag, Berlin/Heidelberg, 2000.
- [16] S. Kumar und D. Ramkrishna. On the solution of population balance equations by discretization-II. A moving pivot technique. *Chem. Eng. Sci.*, 51(8):1333–1342, 1996.

- [17] B.E. Lauder und D.B. Spalding. The numerical computation of turbulent flows. *Comp. Meth. in Appl. Mech. and Eng.*, 3:269–289, 1974.
- [18] Y. Liao und D. Lucas. A literature review of theoretical models for drop and bubble breakup in turbulent dispersions. *Chem. Eng. Sci.*, 62:3389–3406, 2009.
- [19] Y.X. Liao und D. Lucas. A literature review on mechanisms and models for the coalescence process of fluid particles. *Chem. Eng. Sci.*, 65(10):2851–2864, 2010.
- [20] J. Liesen. *Numerik 2 für Ingenieure*. TU Berlin, Vorlesungsskript, WS 2009/2010.
- [21] P.A. Longwell. *Mechanics of Fluid Flow*. McGraw-Hill Book Company, New York, 1996.
- [22] S. Maaß, S. Wollny, R. Sperling und M. Kraume. Numerical and experimental analysis of particle strain and breakage in turbulent dispersions. *Chem. Eng. Res. Des.*, 87:565–572, 2009.
- [23] S. Maaß, S. Hermann und M. Kraume. Determination of breakage rates with single drop experiments. In *4th International Conference on Population Balance Modelling*, Berlin, Germany, 2010. Herausgegeben als USB-Stick.
- [24] D. Marchisio. *Precipitation in Turbulent Fluids*. Dissertation, Politecnico di Torino, 2001.
- [25] R. McGraw. Description of aerosol dynamics by the quadrature method of moments. *Aerosol. Sci. Tech.*, 27(2):255–265, 1997.
- [26] V. Mehrmann. *Kontrolltheorie*. TU Berlin, Vorlesungsskript, SS 2004.
- [27] B. Mohammadi und O. Pironneau. *Analysis of the K-Epsilon Turbulence Model*. Wiley, Chichester, UK, 1993.
- [28] A. Paschedag. *Computational Fluid Dynamics in der Verfahrenstechnik*. Wiley-VHC-Verlag GmbH & Co, Weinheim, Bergstr., 2004.
- [29] D. Ramkrishna. *Population Balances. Theory and Applications to Particulate Systems in Engineering*. Academic Press, San Diego, 2000.
- [30] J. Ritter. *Dispergierung und Phasentrennung in gerührten Flüssig/flüssig-Systemen*. Dissertation, TU Berlin, 2002.
- [31] E. Schade und E. Kunz. *Strömungslehre*. de Gruyter, Berlin, 2007.
- [32] S. Schlauch. *Modeling and Simulation of Drop Size Distributions in Stirred Liquid-Liquid Systems*. Dissertation, TU Berlin, 2006.
- [33] J. Sesterhenn. *Skript zur Vorlesung CFD 1*. TU Berlin, SS 2010.
- [34] J. Sesterhenn. *Skript zur Vorlesung CFD 2*. TU Berlin, WS 2010/2011.
- [35] R. Siebel, M. Schäfer, G. Lauschke und W. Schierholz. Strömungssimulation in Wendel- und Dispersionsrührwerken. *Chemie Ingenieur Technik*, 71(10):1159–1163, 1999.

- [36] L.F.L.R. Silva, R.B. Damian und P.L.C. Lage. Implementation and analysis of numerical solution of the population balance equation in CFD packages. *Comp. Chem. Eng.*, 32:2933–2945, 2008.
- [37] E. Specht. *Einführung in die Vektoranalysis*. www.matheplanet.com, 2005.
- [38] M. Stieß. *Mechanische Verfahrenstechnik - Partikeltechnologie 1*. Springer-Verlag, Berlin/Heidelberg, 2009.
- [39] The MathWorks, Inc., Cochituate Place, 24 Prime Park Way, Natick, MA. MATLAB, Matlab Version 7.11.0.584 (R2010b), 2010.
- [40] A. Thess. *Geophysikalische Fluidodynamik*. TU Dresden, Online-Skriptum, <http://www.tu-dresden.de/mw/ilr/geo/vor6/vor6.html>, 1996.
- [41] B.G.M. van Wachem und A.E. Almstedt. Methods for multiphase computational fluid dynamics. *Chem. Eng. J.*, 96:1385–8947, 2003.
- [42] P. Wesseling. *Principles of Computational Fluid Dynamics*. Springer-Verlag, Berlin/Heidelberg/New York, 2001.
- [43] D. Wilcox. *Turbulence Modeling for CFD*. DCW Industries, Inc., La Canada, California, 1994.