Technische Universität München
Institut für Energietechnik

Lehrstuhl für Fluidmechanik

Ein modulares Verfahren zur Simulation von Mehrfeldproblemen in Strömungsmaschinen

Benedikt Flurl

Vollständiger Abdruck der von der Fakultät für Maschinenwesen der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr.-Ing. T. Sattelmayer
Prüfer der Dissertation:	
1.	Univ.-Prof. Dr.-Ing. Dr. h.c. R. Schilling
2.	Univ.-Prof. Dr.-Ing. H. Baier

Die Dissertation wurde am 12. 04. 2011 bei der Technischen Universität München eingereicht und durch die Fakultät für Maschinenwesen am 02. 08. 2011 angenommen.

Vorwort

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Fluidmechanik, Abteilung hydraulische Strömungsmaschinen und Anlagen, der Technischen Universität München.

Mein besonderer Dank gilt Herrn Prof. Dr.-Ing. Dr. h.c. Rudolf Schilling, der mir diese Arbeit ermöglichte und mit zahlreichen interessanten und lehrreichen Hinweisen wesentlich zum Gelingen beigetragen hat.

Herrn Prof. Dr.-Ing. Horst Baier danke ich für die Übernahme des Koreferates und das Interesse, das er meiner Arbeit entgegenbrachte. Für den Vorsitz der Prüfungskommission darf ich Herrn Prof. Dr.-Ing. Thomas Sattelmayer freundlich danken.

Für die außerordentlich gute Zusammenarbeit gebührt meinen Kollegen, die mich in einer ausgezeichneten Arbeitsatmosphäre begleitet und unterstützt haben, mein aufrichtiger Dank. Hervorzuheben sind hier besonders meine langjährigen Büropartner Mathias Bogner und Joseph Kiermeir, mit denen ich stets konstruktiv diskutieren und produktiv zusammenarbeiten konnte.

Herzlicher Dank geht auch an meine Eltern, die mich beim Erreichen meiner beruflichen Ziele immer bestärkt und nach Kräften unterstützt haben.

Größter Dank gebührt meiner lieben Ehefrau Christina, welche mich in meiner anstrengenden Ausbildungszeit mit Geduld, Rücksichtnahme und Ermutigungen unendlich gestützt hat. Ihr und unserer schnell wachsenden Familie ist diese Arbeit gewidmet.

München, August 2011

Benedikt Flurl

Inhaltsverzeichnis

Verwendete Formelzeichen und Abkürzungen **X**

Zusammenfassung **XIII**

1 Einleitung **1**

1.1 Problemstellung 1

1.2 Stand des Wissens 2

1.2.1 Simulations-Verfahren 2

1.2.1.1 Finite-Elemente-Methode 2

1.2.1.2 Finite-Volumen-Methode 4

1.2.1.3 Fluid-Struktur-Interaktion 8

1.2.1.4 Einseitige FSI 8

1.2.1.5 Zweiseitige FSI 8

1.2.2 Verfahrensentwicklung am FLM 9

1.2.2.1 Integriertes-Design-System 9

1.2.2.2 Simulations-Verfahren 10

1.3 Zielsetzung und Aufbau der Arbeit 11

2 Strömungsmechanik **13**

2.1 Massenerhaltung 13

2.2 Impulserhaltung 14

2.3 Turbulenzmodellierung 14

2.4 Rotierendes Bezugssystem 15

3 Strukturmechanik **17**

3.1 Konfiguration eines Körpers 17

3.2 Zusammenhang von Dehnung und Verschiebung 19

3.3 Gleichgewichtsbeziehungen 22

3.4 Materialgesetz 23
3.4.1 Linear-elastisches und kompressibles Material 23
3.4.2 Linear-elastisches Material 24
3.5 Impulserhaltung 25
3.5.1 Cauchy-Formulierung 25
3.5.2 Totale-Lagrange-Formulierung 26
3.5.3 Fortschreitende-Lagrange-Formulierung 27
3.6 Rotierendes Bezugssystem 28

4 Numerische Verfahren 29
4.1 Allgemeine Transportgleichung 29
4.2 Räumliche Diskretisierung 31
4.2.1 Interpolation an die Zellflächen 31
4.2.2 Gradientenberechnung 32
4.2.3 Diskretisierung des konvektiven Terms 33
4.2.3.1 Stromauf-Interpolations-Verfahren 33
4.2.3.2 Zentrales-Differenzen-Schema 33
4.2.3.3 Hochauflösende Verfahren 33
4.2.4 Diffusiver Term 34
4.2.5 Quellterme 36
4.2.5.1 Volumen-Quellterme 36
4.2.5.2 Oberflächen-Quellterme 36
4.3 Zeitliche Diskretisierung 36
4.3.1 Rückwärts-Euler-Verfahren erster Ordnung 36
4.3.2 Rückwärts-Euler-Verfahren zweiter Ordnung 37
4.3.3 Raum-Erhaltungs-Gesetz 38
4.4 Randbedingungen 40
4.4.1 Dirichlet-Randbedingung 40
4.4.2 Neumann-Randbedingung 40
4.5 Gleichungslöser 41
4.5.1 Unvollständige LU-Zerlegung (ILU) 42
4.5.2 Konjugierte Gradientenverfahren (CG) 43
4.5.3 Mehrgitterverfahren (MG) 44
4.5.3.1 Algebraisches Mehrgitterverfahren (AMG) 45

4.5.3.2 Full Approximation Scheme (FAS) 49
4.5.4 Anmerkungen zur Parallelisierung 51
4.6 Sequentielle Lösungsverfahren 53
4.6.1 Strömungsmechanik 53
4.6.1.1 SIMPLE-Algorithmus 54
4.6.1.2 PISO-Algorithmus 56
4.6.1.3 Diskretisierung der Druckkorrekturgleichungen 57
4.6.2 Strukturmechanik 59
4.6.2.1 Einheitliche Formulierung und Implementierung 59
4.6.2.2 Inkrementelles Rechenverfahren 61
4.6.2.3 Inkompressibles Materialverhalten 61
4.6.2.4 Schwächen des sequentiellen Lösungsverfahrens 62
4.6.3 Fluid-Struktur-Interaktion 64
4.6.3.1 Fluid-Struktur-Interface 64
4.6.3.2 Netzverformung 65
4.6.4 Sequentielles Lösungsverfahren für Fluid-Struktur-Interaktionen 67
4.7 Erhöhung der Stabilität 68
4.7.1 Explizite Unterrelaxation 68
4.7.2 Implizite Unterrelaxation 68
4.7.3 Unterrelaxation Fluid-Struktur-Interface 69
4.7.4 Limitierungen 70
4.8 Parallelisierung und Partitionierung 70
4.8.1 Allgemeines Interface 70
4.8.2 Partitionierung und Parallelisierung 72
4.8.3 Periodizitäten 72

5 Verfahrensentwicklung 75
5.1 Anforderungen an das Simulationswerkzeug 75
5.2 Entwicklungsziele *solver3d* 77
5.3 Komponenten numerischer Anwendungen 78
5.4 Flexibilität durch Konfigurationen 81
5.4.1 Gleichungs-Objekte 82
5.4.2 Diskretisierungs-Objekte 84
5.4.3 Löser-Objekte 84

5.4.4 Das Verfahren *solver3d* 85
5.5 Finite-Elemente-Verfahren fem3d 87

6 Verifikation und Validierung 89
6.1 Fluidmechanik 89
6.1.1 Stationäre Testfälle 90
6.1.1.1 Lid-Driven-Cavity 90
6.1.1.2 Konvektionsgetriebene Strömungen 93
6.1.2 Simulation der Strömung in radialen Kreiselpumpen 99
6.1.2.1 Quasi-Navier-Stokes- bzw. Quasi-Euler-Verfahren . . . 99
6.1.2.2 Auswertegrößen 100
6.1.2.3 Simulationsergebnisse 101
6.1.3 Instationärer Testfall 106
6.1.3.1 Karmansche Wirbelstraße 106
6.2 Strukturmechanik 109
6.2.1 Stationäre linear-elastische Testfälle 109
6.2.1.1 Prismatischer Stab unter Eigengewicht 109
6.2.1.2 Biegung eines beidseitig eingespannten Balkens 111
6.2.1.3 Biegung eines einseitig eingespannten Balkens 112
6.2.1.4 Dickwandiges Rohr unter Innendruck 113
6.2.1.5 Dickwandiges Rohr unter Torsionslast 115
6.2.1.6 Gelochte Platte unter Zug 116
6.2.1.7 Effizienzsteigerung durch Mehrgitterverfahren 120
6.2.2 Stationäre geometrisch-nichtlineare Testfälle 122
6.2.2.1 Nichtlineare Verformung eines eingespannten Balkens . 122
6.2.3 Instationäre linear-elastische Testfälle 124
6.2.3.1 Eingespannter Balken unter plötzlicher Belastung . . . 124
6.2.4 Instationäre geometrisch-nichtlineare Testfälle: 126
6.2.4.1 Eingespannter Balken unter plötzlicher Belastung . . . 126
6.2.5 Kritische Diskussion der Finite-Volumen-Methode 128
6.3 Fluid-Struktur-Interaktion 130
6.3.1 Kaplan-Turbine KT192 130

7 Bewertung und Ausblick 139

A Kompressible Medien **141**

A.1 Energieerhaltung 141

A.1.1 Energieerhaltung im ein rotierenden Bezugssystem 141

A.2 Zustandsgesetze 142

A.2.1 Dichteänderung von Gasen 142

A.2.2 Dichteänderung von Flüssigkeiten 142

A.2.3 Viskositätsänderung von Gasen 143

B Statistische Turbulenzmodellierung **145**

B.0.4 Zwei-Gleichungs-Modelle 146

B.0.4.1 k-ϵ-Turbulenzmodell 146

B.0.4.2 k-ω-Turbulenzmodell 147

B.0.4.3 Shear-Stress-Transport(SST)-Turbulenzmodell 147

B.0.5 Wandbehandlung 149

B.0.5.1 Low-Reynolds-Ansatz 149

B.0.5.2 Wandfunktions-Ansatz 149

B.0.6 Automatische Wandbehandlung 152

C Randbedingungen **153**

C.0.7 Randbedingungen der Strömungsmechanik 153

C.0.7.1 Einlassrandbedingungen 153

C.0.7.2 Auslassrandbedingungen 155

C.0.7.3 Wandrandbedingungen 155

C.0.7.4 Symmetrierandbedingungen 156

C.0.8 Randbedingungen der Strukturmechanik 157

C.0.8.1 Verschiebungs -und Temperaturrandbedingungen . . . 157

C.0.8.2 Spannungsrandbedingungen 157

C.0.8.3 Symmetrierandbedingungen 157

D Lineare Löser **159**

Literaturverzeichnis **160**

Verwendete Formelzeichen und Abkürzungen

Abkürzungen

AMG	Algebraisches Mehrgitterverfahren
ALE	Arbitrary-Lagrange-Euler
ATG	Allgemeine Transportgleichung
CAD	Computer Aided Design
CAA	Computational Aero-Acoustics
CDS	Central Differencing Scheme
CF	Cauchy Formulierung
CFD	Computational Fluid Dynamics
CHH	Computational Hydro-Acoustics
CSA	Computational Structural Analysis
CST	Cauchy-Spannungstensor
DGT	Deformations-Gradienten-Tensor
DNS	Direkte numerische Simulation
DVT	Dehnung-Verzerrungs-Tensor
FEM	Finite-Elemente-Methode
FEV	Finite-Elemente-Verfahren
FLF	Fortschreitende-Lagrange-Formulierung
FLM	Lehrstuhl für Fluidmechanik
FMG	Full-Multigrid
FVM	Finite-Volumen-Methode
FVV	Finite-Volumen-Verfahren
FSI	Fluid-Struktur-Interaktion
Gl.	Gleichung
GLDVT	Green-Lagrange-Dehnungs-Verzerrungs-Tensor
IDS	Integrated-Design-System
NS3D	Navier-Stokes 3D
LF	Lagrange-Formulierung
LR	Links-Rechts
MPI	Message Passing Interface
PKST	Piola-Kirchhoff-Spannungs-Tensor
QEV	Quasi-Euler-Verfahren
QNS	Quasi-Navier-Stokes

RANS	Reynolds-Averaged Navier-Stokes
SCL	Space-Conservation-Law
SIP	Strongly-Implicit-Procedure
SST	Shear-Stress-Transport
TLF	Totale Lagrange Formulierung
TUM	Technische Universität München
ULF	Updated Lagrange Formulierung
URANS	Unsteady Reynolds-Averaged Navier-Stokes

Lateinische Zeichen

$\mathbf{a}$	$[m/s^2]$	Beschleunigungsvektor
A	$[m^2]$	Fläche
$\mathbf{c}$	$[m/s]$	Vektor der Absolutgeschwindigkeit
c		Beiwert
$\mathbf{c}$	$[m/s]$	Vektor der Führungsgeschwindigkeit bei Rotation
D	$[m]$	Durchmesser
$\mathbf{f}$	$[N/m^3]$	Kraftdichte
$\mathbf{F}$		Deformationstensor
F	$[N]$	Kraft
$\mathbf{g}$	$[m/s^2]$	Erdbeschleunigung
h	$[m]$	Höhe
H	$[m]$	Höhe, Förderhöhe
i		Zählvariable
$\mathbf{I}$		Einheitstensor
J		Jacobi-Determinante
k	$[m^2/s^2]$	Turbulente Kinetische Energie
$\dot{m}$	$[kg/s]$	Massenstrom
n	$[s^{-1}]$	Drehzahl
$\mathbf{n}$		Normalenvektor
nb		Nachbar
p	$[N/m^2]$	Druck
P		Zellmittelpunkt
P_k	$[m^2/s^3]$	Quellterm in k-Gleichung
q	$[[\Phi]/s]$	Quellterm allg. Transportgleichung
r	$[m]$	Radius
$\mathbf{r}$	$[m]$	Ortsvektor
t	$[s]$	Zeit
$\mathbf{t}$	$[N]$	Schnittkraft
u	$[m/s]$	Geschwindigkeitskomponente in x-Richtung
$\mathbf{u}$	$[m/s]$	Verschiebungsvektor

$\mathbf{v}$	$[m/s]$	Geschwindigkeitsvektor
v	$[m/s]$	Geschwindigkeitskomponente in y-Richtung
V	$[m^3]$	Volumen
w	$[m/s]$	Geschwindigkeitskomponente in z-Richtung
$\mathbf{w}$	$[m/s]$	Vektor der Relativgeschwindigkeit bei Rotation
$\mathbf{x}$	$[m]$	Ortsvektor

Griechische Zeichen

δ	$[m]$	Stromröhrendicke
δ_{ij}	$[1]$	Kronecker-Symbol $i = j : 1 \quad i \neq j : 0$
Δ		Differenzwert; normierte Stromröhrendicke
ϵ	$[m^2/s^3]$	Dissipationsrate
$\boldsymbol{E}$	$[m^2/s^3]$	Green-Lagrange-Tensor
Γ	$[...]$	Allgemeiner Diffusionskoeffizient
λ	$[1/s]$	Erste Lamésche Konstante
μ	$[kg/(m\,s)]$	Dynamische Viskosität, Zweite Lamesche Konstante
ν	$[m^2/s]$	Kinematische Viskosität, Poisson-Zahl
ϱ	$[kg/m^3]$	Dichte
$\boldsymbol{\tau}$	$[N/m^2]$	Schubspannungstensor
ϕ	$[...]$	Transportgröße
Ψ	$[m^3/s; 1]$	Druckzahl
Θ	$[m^2/s]$	Drall
$\boldsymbol{\sigma}$	$[N/m^2]$	Spannungstensor
$\boldsymbol{\Sigma}$	$[N/m^2]$	Spannungstensor
$\boldsymbol{\Omega}$	$[1/s]$	Winkelgeschwindigkeit

Zusammenfassung

Die numerische Simulation von Mehrfeldproblemen, wie sie in zahlreichen wissenschaftlichen und technischen Problemstellungen häufig auftreten, stellen sowohl an die mathematische als auch an die physikalische sowie an die numerische Modellierung und an das Simulationswerkzeug höchste Anforderungen. Es beeinflussen sich bei diesen Mehrfeldproblemen gleichzeitig unterschiedliche physikalische Phänomene, wie beispielsweise bei Fluid-Struktur-Interaktionen (FSI) ein Fluid eine angrenzende Struktur und umgekehrt.
Mehrfeldprobleme wie FSI treten auch in Strömungsmaschinen auf; das strömende Fluid regt die Gehäusestrukturen, Leit- und Laufräder zu Schwingungen an, und die schwingenden Strukturen werden stark belastet oder erzeugen Lärm. Derartige Effekte mit Hilfe eines leistungsfähigen Simulationswerkzeugs vorausberechnen oder nachvollziehen zu können ist, vor dem Hintergrund der Kosten oder der begrenzten Möglichkeit von Experimenten sehr erstrebenswert.

Für die Simulation derartiger und weiterer Problemstellungen wurde im Rahmen dieser Arbeit ein modulares Lösungsverfahren entworfen, umgesetzt und angewendet. Eine Besonderheit stellt dabei dar, dass dieses Verfahren durch die Definitionen von Konfigurationen und mit Hilfe geeigneter Diskretisierungs- und Gleichungs-Objekte für neue Anwendungen effizient erweiterbar und anpasspar ist. Die Leistungsfähigkeit des Verfahrens und die hohe Vielseitigkeit werden an unterschiedlichen Fluid-, Struktur- und FSI-Simulationen nachgewiesen.

Die Diskretisierung erfolgt auf beliebig polyedrischen Gittern mit einem Finite-Volumen-Verfahren (FVV) zweiter Ordnung. Auch die strukturmechanischen Berechnungnen können mit diesem FVV durchgeführt werden. Das FVV für strukturmechanische Berechnungen wird im Rahmen dieser Arbeit umfassend für unterschiedliche Materialgesetze und anhand zahlreicher Testfälle vorgestellt, und die Leistungsfähigkeit und Schwächen dieser Modellierung für Struktursimulationen werden ausführlich diskutiert. Die Anwendbarkeit des modularen Verfahrens auf die Simulation einer FSI in einer Strömungsmaschine rundet die Arbeit ab. Neben Modulen zur Gleichungsdiskretisierung und -lösung kommen Module für die Behandlung der Fluid-Struktur-Interfaces und zur Netzverformung zur Anwendung.

Mit dem neu entwickelten modularen Lösungsverfahren steht für zukünftige Forschungs- und Entwicklungsprojekte eine leistungsfähige und ausbaufähige Basis für komplexe ingenieurwissenschaftliche Problemstellungen zur Verfügung und soll in Zukunft als Grundbaustein eines integrierten Entwurfs- und Optimierungssystems für Strömungsmaschinen eingesetzt werden.

Kapitel 1

Einleitung

1.1 Problemstellung

Numerische Verfahren zur Simulationen von Problemstellungen der Strömungs- und Strukturmechanik haben in vielen Bereichen der Ingenieurwissenschaften gleichermaßen im akademischen wie im industriellen Bereich breite Anwendung gefunden. Für viele Problemstellungen aus diesen Bereichen können die die Strukturen oder Strömungen beschreibenden Gleichungen in der Regel nicht mehr analytisch gelöst werden, sondern nach sinnvoller Modellierung und Abstrahierung nur noch numerisch.

Weiterhin sind experimentelle Untersuchungen im Gegensatz zu einer numerischen Simulation häufig zeit- und kostenintensiv oder aber in vielen Fällen nicht durchführbar. Ferner kann beim Entwurf oder bei der Optimierung von technischen Systemen die numerische Simulation sehr vorteilhaft eingesetzt werden. Durch Variation der untersuchten Geometrien können Parameterstudien durchgeführt und wiederum einfache neue analytische Ersatzmodelle entwickelt oder bekannte Ersatzmodelle kalibriert werden. Die Entwicklungszeiten technischer Systeme können so verkürzt, der erforderliche Prototypenbau minimiert und dadurch die Entwicklungskosten reduziert werden.

Durch die immer weiter fortschreitende Entwicklung der Digitalrechner werden diese numerischen Verfahren zunehmend effizienter sowie immer häufiger und selbstverständlicher eingesetzt. Je effizienter jedoch die Digitalrechner werden, desto komplexer werden auch die zu untersuchenden Phänomene: Zum einen werden bisher stark abstrahierte und modellierte Probleme detaillierter untersucht, zum anderen werden neue Arten von Problemstellungen gewählt.

Eine besondere Herausforderung stellen hierbei Simulationen aus dem Bereich der Mehrfeldprobleme dar, bei denen unterschiedliche physikalische Phänomene gleichzeitig wirken und sich gegenseitig beeinflussen. Ein Beispiel für diese Mehrfeldproblematik ist die Simulation von Fluid-Struktur-Interaktionen (FSI), bei der Simulationsverfahren zur Behandlung von Strömung und Struktur miteinander so gekoppelt werden, dass sowohl die Strömung die Struktur als auch die Struktur die Strömung anregt und damit beide Felder miteinander interagieren.

Neben der Genauigkeit solcher Verfahren zur Simulation von Fluid-Struktur-Interaktionen spielt die Effizienz eine wichtige Rolle. Nicht nur das Verfahren zur Simulation der Strömung, sondern auch das Verfahren zur Simulation der Struktur muss effizient arbeiten können. Weiterhin müssen Daten an den Schnittstellen der Felder schnell aufbereitet und untereinander ausgetauscht werden, zudem sind Simulationen dieser Art meist instationärer Natur, so dass auch das Speichern und das Nachbearbeiten der Daten pro Zeitschritt effizient erfolgen müssen.

Weiterhin sollte ein Verfahren so flexibel sein, dass es in einfacher Weise zur Untersuchung neuer Problemstellungen angepasst oder erweitert werden kann.

Für eine genaue numerische Simulation von realen Strömungen müssen nach WALTERS [106] folgende Hauptkriterien eingehalten werden: Die

- genaue Geometrieapproximation,
- hohe Qualität der Netzgenerierung,
- genaue Approximation der konvektiven Terme der Transportgleichungen und
- physikalische sinnvolle Modellierung der Turbulenz.

Für eine genaue numerische Simulation realer Strukturen gelten ebenfalls die ersten beiden Kriterien, weiterhin müssen die diffusiven Terme der Transportgleichungen genau modelliert werden. Der vierte Punkte entfällt für Struktur-Simulationen, es treten jedoch andere Schwierigkeiten auf, wie etwa die Formulierung geeigneter Materialgesetze oder die genaue Modellierung geometrischer oder physikalischer Nichtlinearitäten.

1.2 Stand des Wissens

Numerische Simulationsmethoden werden in den unterschiedlichsten wissenschaftlichen und industriellen Bereichen erfolgreich angewendet. In den Ingenieurswissenschaften sind dies hauptsächlich die Simulation von Wärmeleitungs- und Strömungsvorgängen sowie die Strukturanalyse. Zur numerischen Behandlung der zugrunde liegenden partiellen Differentialgleichungen werden unterschiedliche Finite-Diskretisierungs-Verfahren verwendet und in den folgenden Abschnitten kurz beschrieben. Ein Hauptaugenmerk liegt hierbei auf der Finite-Volumen-Methode, die einen Schwerpunkt dieser Arbeit bildet.

1.2.1 Simulations-Verfahren

1.2.1.1 Finite-Elemente-Methode

Die Finite-Elemente-Methode (FEM) zur Diskretisierung und Lösung partieller Differentialgleichungen wurde in den 50er Jahren eingeführt, s. AGYRIS [11], ZIENKIEWICZ [111] und BATHE [15], und hat sich seither besonders in der

Strukturmechanik als effizientes numerisches Werkzeug etabliert. Die FEM basiert auf den Näherungsverfahren nach Galerkin und Ritz, s. PFEIFFER [82].
Die Idee hinter diesen Näherungsverfahren steht besteht darin, die gesuchte kontinuierliche Lösungsgröße, wie z. B. bei strukturmechanischen Problemstellungen das Verschiebungsfeld, durch eine endliche Linearkombination mit Hilfe von sog. Basisfunktionen zu approximieren. Die Basisfunktionen müssen lediglich die kinematischen Randbedingungen erfüllen und können ansonsten beliebig gewählt werden.

Anstatt das gesamte Verschiebungsfeld zu approximieren, werden innerhalb der sog. Finiten Elemente gebietsweise Näherungen ermittelt, welche durch Knoten definiert werden. Das Verschiebungsfeld innerhalb eines Elements wird dann mit Hilfe der Knotenverschiebungen bestimmt. Durch lokal gültige Ansatzfunktionen werden jedem Element Eigenschaften wie Masse und Steifigkeit zugeordnet. Durch Superposition der einzelnen Elemente zur Gesamtstruktur gewinnt man schließlich das problembeschreibende Gleichungssystem, welches mit direkten oder iterativen Verfahren gelöst wird.

Die geometrische Form der Elemente kann beliebig gewählt werden. Man unterscheidet allgemein gültige Kontinuumselemente und spezialisierte Elemente wie Stab-, Balken oder Schalenelemente, welche sich spezieller Modellvorstellungen der Kontinuumsmechanik bedienen.

Mit Hilfe der FEM lassen sich heute stationäre, instationäre, geometrisch oder physikalisch lineare und nichtlineare Problemstellungen der Struktur- und der Strömungsmechanik erfolgreich untersucht.

Vor allem die folgenden genannten Gründe sprechen für eine Verwendung der FEM:

- Die FEM wurde in den 50er Jahren und in den folgenden Jahren bis heute stetig weiterentwickelt und sie ist deswegen für unterschiedlichste Bereiche der numerischen Simulation etabliert und wird in diesen zuverlässig angewendet.
- Die Diskretisierungsordnung und damit die Genauigkeit der numerischen Simulation kann durch die gewählten Ansatzfunktionen in einfacher Weise lokal auf Elemente bezogen erhöht werden.

Diesen positiven Eigenschaften stehen u. a. folgende negative Merkmale gegenüber:

- Eine Formulierung für allgemeine Element- und Netztopologien ist schwierig in eine Programmstruktur umsetzbar, s. BATHE [15].
- Bei den verschiebungsbasierten FEM können sog. Locking-Phänomene auftreten, d.h. eine durch die Diskretisierung künstlich und damit unphysikalische erhöhte Steifigkeit.
- Bei der Diskretisierung von sog. Erhaltungsgleichungen mit der FEM können die Erhaltungseigenschaften nur schwer beibehalten werden. Die FEM garantiert zunächst keine Konservativität, weswegen sie sich auch bis heute nicht zur Simulation strömungsmechanischer Problemstellungen durchsetzen konnte.

1.2.1.2 Finite-Volumen-Methode

Die Finite-Volumen-Methode (FVM) wurde in den 60er Jahren von PATANKAR [79] für Wärmetransportvorgänge und Strömungsmechanische Problemstellungen eingeführt und hat sich vor allem durch die Arbeiten von PERIC [81] und vielen weiteren in den 90er Jahren für die numerische Simulation realer Strömungen durchgesetzt und wird für unterschiedlichste Problemstellungen erfolgreich angewendet.
Bei den Finite-Volumen-Verfahren (FVV) werden nicht die originären, problembeschreibenden partiellen Differentialgleichungen diskretisiert. Zunächst wird das Simulationsgebiet in nichtüberlappende Finite Volumen unterteilt. Schließlich werden die partiellen Differentialgleichung über ein repräsentatives Volumen integriert. Mit Hilfe des Integralsatzes von Gauss werden dann die auftretenden Volumenintegrale in Oberflächenintegrale umgewandelt. Somit erhält man Bilanz- oder Flussgleichungen über die Oberflächen der einzelnen Finiten Volumina.

Das durch diese Diskretisierungstechnik entstandene, problembeschreibende Gleichungssystem kann dann mit direkten oder iterativen Verfahren gelöst werden.

Vor allem die folgenden Eigenschaften der FVM sprechen für eine Verwendung dieses Verfahrens:

- Durch die Flussformulierung der partiellen Differentialgleichung werden die Erhaltungseigenschaften dieser Gleichungen automatisch erfüllt, dann spricht man von einem konservativen Verfahren.
- Die Finite-Volumen-Formulierung ist besonders einfach auf beliebige Volumen- und Netztopologien anwendbar und deswegen für geometrisch hochkomplexe Problemstellungen besonders geeignet.

Ursprünglich zur Simulation strömungsmechanischer Problemstellungen entwickelt, kann das Finite-Volumen-Verfahren grundsätzlich auch zur Lösung strukturmechanischer Problemstellungen angewendet werden und ist überall dort eine sinnvolle Diskretisierung- und Lösungsmethode, wo die Erhaltung von physikalischen Größen eine wichtige Rolle spielt.
Im Rahmen dieser Arbeit wird ein modulares Lösungsverfahren entwickelt, welches für Problemstellungen der Strömungs- und Strukturmechanik sowie zur Simulation von Fluid-Struktur-Interaktionen die FVM als Diskretisierungsverfahren anwendet. Deswegen werden kurz der bisherige Stand der Technik der FVM und für diese Arbeit wichtige Meilensteine bei der Entwicklung der FVM in diesen Teilgebieten der numerischen Simulation zusammengefasst.

Finite-Volumen-Methode in der Strömungsmechanik

Die FVM zur Simulation realer Strömungen wurde durch die grundlegenden Arbeiten von PATANKAR [79] etabliert. Patankar führte die sequentiellen Lösungsverfahren SIMPLE und SIMPLER zur Behandlung gekoppelter nichtlinearer Gleichungssysteme ein, welche heute beinahe ausschließlich in universitären und kommerziellen Verfahren Verwendung finden. Bei diesem sequentiellen Lösungsverfahren können in eleganter Weise unterschiedliche Kombinationen von Gleichungssystemen behandelt werden, so dass heute laminare und turbulente, kavitierende und kondensierende, mehrphasige, reaktive u. v. a. Strömungsphänomene numerisch untersucht werden können.
Die sequentiellen Lösungsverfahren existieren heute in zahlreichen Varianten; zu nennen sind hier z. B. das SIMPLEC-Verfahren nach DOORMAL [100], welches jedoch nur in speziellen Fällen wesentliche Verbesserungen gegenüber dem originären SIMPLE-Verfahren liefert. Ein umfassender Vergleich dieser und der weiteren Varianten SIMPLEX, SIMPLEST und PRIME findet sich in DARWISH [28].
Patankar verwendet zur Lösung noch versetzte Rechennetze. Hier werden das Geschwindigkeits- und Druckfeld auf unterschiedlichen Netzen diskretisiert und bestimmt, woraus ein hoher Speicheraufwand und eine komplizierte Datenhaltung resultieren. RHIE [86] führt die nichtversetzte Variablenanordnung für das Geschwindigkeits- und das Druckfeld ein, welche auch in dieser Arbeit Verwendung findet. Durch diese wird der Speicheraufwand gering gehalten und die Datenhaltung wesentlich vereinfacht.

Instationäre Strömungsvorgänge, die im Rahmen dieser Arbeit eine wesentliche Rolle spielen, können wie stationäre Strömungsvorgänge mit den von PATANKAR [79] eingeführten sequentiellen Lösungsverfahren behandelt werden. Eine entscheidende Beschleunigung bei Simulationen mit kleinen Zeitschritten wurde durch das neue PISO-Druckkorrektur-Verfahren nach ISSA [56] erreicht, das auch im Rahmen dieser Arbeit zur Anwendung kommt.

Die Einfachheit und Vielseitigkeit der FVM haben diese zu dem heute allgemein verwendeten Werkzeug zur numerischen Simulation realer Strömungsvorgänge gemacht.

Finite-Volumen-Methode in der Strukturmechanik

Die FVM hat sich zur Simulation realer Strömungsvorgänge durchgesetzt. Sie wird heute im universitären und industriellen Umfeld erfolgreich als Simulationswerkzeug angewendet und für immer neue Problemstellungen angepasst, verbessert und erweitert. Im Gegensatz dazu ist die FVM zur Simulation strukturmechanischer Problemstellungen heute noch wenig bekannt und wenig verbreitet.

In der Strukturmechanik wurde die FVM in den 90er Jahren durch eine Arbeit von DEMIRDZIC [30] eingeführt. In diesem Artikel wird die Anwendung der FVM für die thermo-elasto-plastische Spannungsanalyse entwickelt. Neben den Lamé-Gleichungen zur Lösung des Verschiebungsfeldes wird außerdem das Fourier-Gesetz als zusätzliche Erhaltungsgleichung diskretisiert und gelöst, um neben den Spannungen, resultierend

aus dem Verschiebungsfeld, Wärmespannungen berücksichtigen zu können. Es folgen einfache, zweidimensionale und rotationssymmetrische dreidimensionale Simulationsrechnungen, welche mit analytischen und experimentellen Lösungen verglichen werden.

In einer weiteren Arbeit von DEMIRDZIC [31] wird die FVM ausschließlich auf die Spannungsanalyse in komplexen Rechengebieten erfolgreich angewendet. Die Lamé-Gleichungen werden diskretisiert und liefern nach Lösung das dreidimensionale Verschiebungsfeld, mit welchem dann über das Hooke-Gesetz das Spannungsfeld ermittelt wird. Die Diskretisierung wird auf polyedrisch, unstrukturierten Rechennetzen durchgeführt. Die Diskretisierung lässt eine sequentielle Lösungstechnik, wie schon in PATANKAR [79] zur Lösung der Fluid-Gleichungen beschrieben, zu. Es wird auf die Konservativität und die aus dieser Eigenschaft resultierende mögliche Anwendbarkeit und physikalische Korrektheit des Verfahrens auch bei der Diskretisierung auf sehr groben Netzen hingewiesen. Zudem kann das Verfahren einfach erweitert werden.
Das vorgestellte Verfahren wird auf zwei einfache Testfälle, einen dreidimensionalen Balken unter Eigengewicht und eine gelochte Platte unter Zug, angewendet und zeigt hierbei hervorragende Übereinstimmung mit analytisch gewonnenen Vergleichslösungen.
Ein großer Nachteil bei der angewendeten Diskretisierungstechnik liegt in ihrer geometrischen Richtungsabhängigkeit begründet, wodurch die Koeffizientenmatrizen für jede der drei gelösten Erhaltungsgleichungen gespeichert werden müssen. Außerdem ist eine starke Unterrelaxation der diskreten Gleichungen für eine stabile Lösung notwendig, und das Konvergenzverhalten des iterativen Lösungsverfahrens ist mäßig.

Eine Abhilfe gegen diese erwähnten Nachteile liefert DEMIRDZIC [32] in einer weiteren Arbeit, in welcher er das iterative Verfahren unter Verwendung eines geometrischen Mehrgitterverfahrens stabilisiert und stark beschleunigt. Der Nachteil des hohen Speicherbedarfs aufgrund der unterschiedlichen Koeffizientenmatrizen für die diskreten Erhaltungsgleichungen konnte jedoch noch nicht behoben werden. Es werden drei Testfälle auf unterschiedlich feinen Netzen gerechnet und als Benchmark-Ergebnisse vorgestellt. Zu erwähnen ist hier vor allem der letzte Testfall, welcher von der NAFEMS [5] (*N*ational *A*gency for *F*inite *E*lement *M*ethods and *S*tructures), als Validierungstestfall vorgeschlagen wurde. Dieser Testfall stellt eine elliptische, dicke und gelochte Platte dar, welche am Außenrand eingespannt und normal zu diesem aus einer Richtung belastet wird. Das vorgestellte FVV liefert hervorragende Übereinstimmungen mit Lösungen, die mit einem FEV gewonnen wurden.

Dass die FVM auch zur Simulation dünnwandiger Strukturen wie Platten oder Schalen verwendet werden kann, zeigt WHEEL [107]. In seinem Artikel untersucht er die Durchbiegung am Rand eingespannter Platten mit quadratischer und runder Form und erzielt schon bei sehr groben Netzen gute Ergebnisse. Das Verfahren zeigt kein Locking-Verhalten, wie es bei strukturmechanischen Simulationen mit der FEM auftreten kann.

Ein wichtiger Schritt in Richtung der Erhöhung der Stabilität und der Verbesserung des Konvergenzverhaltens der FVM in der Strukturmechanik stellt die Arbeit von JASAK [60] dar: In dieser Publikation wird die Diskretisierung so modifiziert, dass für alle Erhaltungsgleichungen zur Berechnung der Verschiebungen dieselbe

Koeffizientenmatrix erhalten wird. Dadurch wird der Speicherbedarf des Verfahrens im Gegensatz zu seinen Vorgänger-Varianten signifikant reduziert. Weiterhin wird durch Einführung einer Überrelaxierung des Gleichungssystems eine maximale implizite Diskretisierung angewendet und das Verfahren auch ohne Anwendung eines Mehrgitterverfahrens wie in DEMIRDZIC [32] extrem stabilisiert und beschleunigt. Darin werden drei unterschiedliche Testfälle vorgestellt: Zunächst wiederum die gelochte Platte unter Zug, schließlich die instationäre Wellenausbreitung in einem plötzlich belasteten prismatischen Stab und die instationäre Spannungsanalyse in einem Diesel-Einspritzsystem. Es werden hervorragende Ergebnisse erzielt. Mit dem letzten Testfall wird außerdem die einfache Parallelisierbarkeit des Verfahrens gezeigt.

In einem späteren Forschungsbericht zeigt JASAK [61], wie in einfacher Weise gemischte Randbedingungen in einer FVM in der Strukturmechanik zu implementieren sind. Solche Randbedingungen treten unter anderem in Kontaktproblemen zwischen zwei Strukturoberflächen auf.

Schließlich zeigt erstmals MANEERATANA [69] in seiner Dissertation, wie die FVM auch für geometrische Nichtlinearitäten angewendet werden kann. Er untersucht fast ausschließlich stationäre Belastungsvorgänge und vergleicht die Simulationen sowohl mit analytischen Ergebnissen als auch experimentellen Daten. Die Übereinstimmung ist in beiden Fällen sehr gut.
Diese Arbeit wird von TUKOVIC [98] aufgegriffen. Dabei wird die stationäre und instationäre, extreme Deformation eines eingespannten Balkens unter Endlast untersucht. Weiterhin wird die Schwingung einer idealisierten Turbinenschaufel simuliert. Ein besonderes Augenmerk dieser Arbeit liegt in der Verwendung einer robusten und nicht-dissipativen Diskretisierung des Zeitterms.

Auch BIJELONJA [19] wendet die FVM für Simulationen mit großen Verschiebungen und Verzerrungen an, beschränkt sich aber auf inkompressible Materialien, die dem Mooney-Rivlin-Materialgesetz gehorchen.

Die bisher aufgeführten Arbeiten liefern die Basis für die numerische Behandlung der Strukturmechanik des in dieser Arbeit entwickelten Verfahrens. Die vielseitige Anwendbarkeit der FVM zur Simulation strukturmechanischer Problemstellungen ist weiteren Arbeiten zu entnehmen.
Hier sind die Veröffentlichungen von IVANKOVIC [57] zu nennen, der die FVM zur Simulation der instationären Rissausbreitung in Materialien verwendet, die er in einem späteren Artikel, s. (IVANKOVIC [58]), unter Verwendung eines Mehrgitter-Lösers hinsichtlich Robustheit und Schnelligkeit verbessert. Desweiteren ist eine Arbeit von DEMIRDZIC [29] zu erwähnen, in der die FVM zur Simulation des Spannungszustand und der Deformation in thermo-elastischen orthotropen Materialien untersucht wird, sowie eine Abhandlung von BIJELONJA [19], in der die FVM zur Simulation inkompressibler linear elastischer Körper untersucht wird.

Zusammenfassend lassen sich in allen Arbeiten eine große Vielseitigkeit in der Anwendbarkeit und eine hohe Qualität erzielter Simulationsergebnisse unter Verwendung der FVM erkennen, weswegen diese eine attraktive Alternative und somit eine zukunftsfähige Methode für Forschungen und Anwendungen im universitären wie im

industriellen Umfeld darstellt.

1.2.1.3 Fluid-Struktur-Interaktion

Die Arbeit befasst sich mit der Entwicklung eines modularen Lösungsverfahrens, welches die FVM im Speziellen zur Simulation von Mehrfeldproblemen anwendet. Zu den Mehrfeldproblemen gehören auch die Problemstellungen der Fluid-Struktur-Interaktion.
Bei diesen Simulationen können zwei Klassen unterschieden werden: die sog. einseitige und die zweiseitige FSI. Beide Varianten können mit dem entwickelten Verfahren realisiert werden, weswegen eine kurze Beschreibung dieser Methoden erfolgt.

1.2.1.4 Einseitige FSI

Bei der einseitigen FSI werden Strömungsfeld und Strukturfeld getrennt simuliert. Dadurch tritt keine Rückkopplung vom Strukturfeld ins Strömungsfeld auf. Ein Vorteil dieses Verfahrens ist, dass bei dieser Methode beliebige, voneinander unabhängige Programme miteinander kombiniert werden können.
Ein Nachteil ergibt sich aus der Tatsache, dass dieses Vorgehen nur beim Auftreten von kleinen Verformungen sinnvoll anwendbar ist.

1.2.1.5 Zweiseitige FSI

Bei der zweiseitigen FSI kann die Strukturantwort aufgrund der Fluidbelastung nicht mehr vernachlässigt werden.
Man unterteilt weiterhin in die externe und die interne zweiseitige FSI.

Bei der externen FSI werden, wie bei der einseitigen FSI, beliebige Programme miteinander gekoppelt, weshalb ein eigenständiges, weiteres Programmmodul zur Datenaufbereitung und zum Datenaustausch benötigt wird. Müssen die Daten häufig ausgetauscht werden, wie das in iterativen Lösungsverfahren der Fall ist, können diese Methoden in der Anwendung aufwändig werden.
EINZINGER [34] verwendet ein derartiges Verfahren zur Untersuchung der FSI in Turbomaschinen und koppelt einen Finite-Volumen-Strömungslöser mit einem Finite-Elemente-Strukturlöser. Auch kommerzielle Programme wie z. B. ANSYS-CFX [2] verfolgen oftmals diese Lösungsstrategie.

Der genannte Nachteil der externen, zweiseitigen Kopplung wird durch die Verwendung interner FSI-Verfahren umgangen, wobei ein eigenes Programm die Berechnung des Strömungs- und des Strukturfeldes sowie die Datenaufbereitung und den Austausch übernimmt. Der bei diesen Verfahren zeitlich vernachlässigbare Datenaustausch kann so oft wie nötig effizient durchgeführt werden. WALL [105] oder HRON [55] verwenden durchgängige FEV mit dieser Lösungsstrategie. Auch die Open-Source-Bibliothek OPENFOAM [6] arbeitet an einem derartigen Verfahren, jedoch unter ausschließlicher Verwendung der FVM.

Zu erwähnen sind weiterhin Verfahren, bei denen Fluid- und Strukturfeld als Teile eines einzigen zusammenhängenden Kontinuums betrachtet werden. Datenaufbereitung und -austausch an Schnittstellen zwischen den Feldern existieren dann nicht mehr. Diese Verfahren können daher sehr effizient sein. Die Unterscheidung zwischen Fluid- und Strukturbereich erfolgt dann einfach über bereichsweise definierte Zustandsgesetze.
Verfahren dieser Art, sog. Monolithische Verfahren, werden z. B. von GREENSHIELDS [42] und PAPADAKIS [78] beschrieben, bleiben aber noch auf spezielle Anwendungen beschränkt.

1.2.2 Verfahrensentwicklung am FLM

1.2.2.1 Integriertes-Design-System

Seit etwa zwei Jahrzenten wird am Lehrstuhl für Fluidmechanik (FLM) der TUM ein integriertes Entwurfssystem (*I*ntegrated *D*esign *S*ystem) für den Erstentwurf, die Nachrechnung und die Optimierung von Strömungsmaschinen entwickelt. Das IDS wird in der Industrie bereits erfolgreich eingesetzt und im Rahmen von Forschungsprojekten stetig weiterentwickelt, s. SCHILLING [88] u. SCHMALHORST [91].
Prinzipiell können mehrstufige Arbeits- und Kraftmaschinen in radialer und axialer Bauweise untersucht werden. Mit Hilfe des IDS werden sowohl einfache stationäre Einkanalrechnungen als auch aufwändige instationäre Rechnungen ganzer Maschinen durchgeführt. Die Vielseitigkeit und Komplexität der zu untersuchenden Problemstellungen machen eine hochmodularisierte und somit gut erweiterbare Programmstruktur unbedingt erforderlich. Man unterscheidet drei Hauptmodule des IDS:

- Das *Preprocessing-Modul* beinhaltet die Spezifizierung des zu untersuchenden Maschinentyps hinsichtlich des Auslegungspunkt und der physikalischen Eigenschaften des strömenden Fluids.
  Weiterhin beinhaltet dieses Modul einen einfachen, aber leistungsfähigen Netzgenerator. In der zweidimensionalen Meridianansicht wird mit Hilfe geladener CAD-Daten der rotationssymmetrische Teil des Rechengebiets in Bereiche aufgeteilt und anschließend blockstrukturiert, zweidimensional vernetzt. Die zweidimensionalen Netze werden dann rotiert, um den realen dreidimensionalen Rechenraum abzubilden.
  Unsymmetrische Maschinenbereiche, wie etwa das Spiralgehäuse, werden in einem eigenen Modul mit Hilfe der geladenen CAD-Daten dreidimensional vernetzt.

- Das *Solver-Modul* beinhaltet den dreidimensionalen Löser für die durchzuführenden stationären und instationären Simulationen. Das Modul liefert den Input für die Visualisierung und das Postprocessing der ermittelten Daten.

- Mit Hilfe des *Postprocessing-Moduls* können die ermittelten Daten visualisiert und ausgewertet werden. Schnittstellen zu externen kommerziellen oder Open-Source-Postprocessing-Programmen werden ebenfalls bereitgestellt. Ferner können die Daten für eine Optimierungsrechnung aufbereitet werden.

Im Rahmen dieser Arbeit wurde das *Solver-Modul* durch ein neues, modulares und leistungsfähigeres Verfahren ersetzt. Dieser Schritt wurde vor allem aufgrund der veralteten Programm- und hinterlegten Datenstruktur und der daraus resultierenden schlechten Wartbarkeit und kaum möglichen Erweiterbarkeit dieses Moduls notwendig.

1.2.2.2 Simulations-Verfahren

Das *Solver-Modul* des integrierten Entwurfssystems unterliegt einem ständigen Prozess aus Weiter- und Neuentwicklung aufgrund der stetig steigenden Anforderungen hinsichtlich der Komplexität der zu untersuchenden Phänomene. Die zunächst am Lehrstuhl entwickelten Simulationsverfahren waren einfache, hochspezialisierte Programme mit einfacher strukturierter Datenbasis und einem sehr statischen nicht-modularen Programmaufbau.

Die ersten modernen Simulationsverfahren, basierend auf einer Finite-Volumen-Diskretisierung am FLM, wurden von SCHUSTER [92] und BADER [13] entwickelt. Schuster entwickelte ein parallelisiertes Programm zur Simulation der turbulenten, instationären Strömungsphänomene in einem Voith-Schneider-Propeller auf bewegten Rechennetzen.
Bader entwickelte ein einfaches Simulationsverfahren für die turbulente, stationäre, schwachkompressible Durchströmung von Axialventilatoren und beschränkte sich auf Einkanalrechnungen und Simulationen auf strukturierten Gittern.

MÜLLER [74] und REINELT [84] unternahmen erste erfolgreiche Versuche zur Modularisierung und Objektorientierung des Simulationsverfahrens. Dadurch konnten erste einfache Simulationen akademischer Testfälle auf unstrukturierten Rechennetzen durchgeführt werden.

SKODA [93] vollzog schließlich den Umstieg auf die Programmiersprache C und verwendete in einigen Programmteilen bereits objektorientierte Ansätze der Sprache C++. Dieses Verfahren arbeitet mit blockstrukturierten Rechennetzen, was die Simulation komplexer Problemstellungen und ganzer Maschinen erstmals in der Geschichte des Lehrstuhls möglich machte. Die Parallelisierung wird mit der Bibliothek MPI [4] durchgeführt. SKODA [93] implementierte eine umfangreiche Bibliothek von Turbulenzmodellen und eine Methoden zur Diskretisierung der konvektiven Terme. Es können somit stationäre und instationäre Problemstellungen untersucht werden. Ferner wurde ein nichtlineares Mehrgitterverfahren, welches den SIMPLE-Algorithmus als Glätter verwendet, integriert. Hierbei ergaben sich jedoch Schwierigkeiten hinsichtlich Stabilität und Konvergenzverhalten bei der Simulation turbulenter Strömungsvorgänge.

EINZINGER [34] erweiterte das Verfahren von Skoda auf kompressible Strömungen beliebiger Machzahl-Bereiche. Ferner entwickelt er ein Modul zur Simulation von strukturmechanischen Problemen, basierend auf der Finite-Elemente-Methode, und machte damit einfache Simulationen von stationären und instationären Fluid-Struktur-Interaktionen möglich.

FROBENIUS [39] machte erste Versuche zur stationären und instationären Modellierung und Simulation von Kavitationsvorgängen mit dem inkompressiblen SIMPLE-Druckkorrekturverfahren.

Zur Simulation von Strömungsvorgängen in Schraubenspindelpumpen wird das Verfahren um ein allgemeines nichtpassendes Interface und die Chimera-Technik hinsichtlich Praxistauglichkeit untersucht, s. HERMANN [50] u. [51].

BOGNER [20] erweiterte die Kavitationsmodellierung und überarbeitete diese grundlegend. Das Verfahren wurde um neue, stabilere und schnellere lineare Gleichungslöser erweitert. Die zunehmend komplexeren Problemstellungen setzen dem kontinuierlich weiterentwickelten Verfahren unüberwindbare Grenzen. Bogner schlug deswegen eine grundlegende Überarbeitung der Programmstruktur vor. Diese Überarbeitung umfasst den Umstieg auf eine unstrukturierte, polyedrische Datenstruktur, die Aufteilung des Programms in einzelne dynamisch verlinkte Bibliotheken und die Verwendung von moderner Templateprogrammierung der Programmiersprache C++.

1.3 Zielsetzung und Aufbau der Arbeit

Im Rahmen dieser Arbeit wird ein modulares Lösungverfahren zur Simulation von Mehrfeldproblemen entwickelt, welches sowohl als zentraler Baustein eines numerischen Entwicklungssystems für Strömungsmaschinen im industriellen Entwurfs- und Optimierungsprozess eingesetzt werden kann als auch für universitäre Forschungsarbeiten eine leicht adaptierbare und erweiterbare Programm- und Programmierumgebung für allgemeinere Problemstellungen darstellt.

Das entwickelte Verfahren wird im Rahmen dieser Arbeit zur Lösung sowohl strömungs- als auch strukturmechanischer Problemstellungen wie auch gekoppelter Fluid-Struktur-Interaktionen verwendet. Die Fluid-Gleichungen werden mit einem Finite-Volumen-Verfahren diskretisiert. Zur Lösung der Gleichungen wird eine Programmstruktur entwickelt, welche eine maximale Flexibilität hinsichtlich Erweiterbarkeit auf neue Problemstellungen zulässt. Die Struktur-Gleichungen können entweder mit einem neu entwickelten Finite-Volumen-Modul oder mit einem bereits bestehenden, aber überarbeiteten Finite-Elemente-Modul berechnet werden.

Die mit dem entwickelten Verfahren untersuchten Problemstellungen sind stationärer wie instationärer Natur. Die Berechnung voll-turbulenter Strömungen erfolgt durch Lösung der Reynolds- bzw. Favré-gemittelten Navier-Stokes-Gleichungen in Verbindung mit einem Wirbelviskositätsansatz von BOUSSINESQ [22].
Bei den strukturmechanischen Problemstellungen können geometrische und physikalische Nichtlinearitäten in einfacher Weise berücksichtigt werden. Für die Fluid-Struktur-Interaktion können unterschiedliche Kopplungsalgorithmen verwendet werden. Ferner kann die Struktur-Berechnung entweder mit einem Finite-Volumen- oder einem Finite-Elemente-Verfahren durchgeführt werden.

Um eine problemlose Wiederverwendbarkeit, einfache Wartbarkeit und Erweiterbarkeit zu ermöglichen, ist der Kern des Verfahrens in der objektorientierten Programmiersprache C/C++ entwickelt und unterteilt in verschiedene speziell entwickelte, unabhängige Bibliotheken; sie greift aber auch auf frei verfügbare und eingebundene Bibliotheken zu.

Im ersten Kapitel erfolgte nach einer kurzen Darlegung der Problemstellung ein Abriss des momentanen Wissensstandes.

Die beiden folgenden Kapitel fassen die wesentlichen physikalischen Grundlagen der zu beschreibenden Phänomene zusammen. Ein besonderer Schwerpunkt liegt hierbei in der Beschreibung der strukturmechanischen Gleichungen und den verschiedenen Ansätzen zur Betrachtung instationärer strukturmechanischer Problemstellungen.

Das vierte Kapitel erläutert dann ausführlich die in dieser Arbeit angewendeten numerischen Verfahren und die ihm zugrunde liegende Diskretisierungstechnik. Die angewandten Techniken zur Lösung linearer Gleichungssyteme werden im Einzelnen vorgestellt. Ferner werden die Besonderheiten bei der Anwendung von sequentiellen Lösungsverfahren auf fluidmechanische und strukturmechanische Problemstellungen aufgezeigt.

Im fünften und sechsten Kapitel wird das Verfahren sowohl auf akademische wie auch praktische Problemstellungen aus der Fluid- und Strukturmechanik angewendet. Es soll dabei verdeutlicht werden, wie durch die Modularität des Verfahrens in einfacher Weise neue und unterschiedlich komplexe Problemstellungen untersucht werden können.
Ein besonderes Augenmerk liegt dabei auf der Validierung des neu entwickelten Finite-Volumen-Struktur-Moduls. Dabei werden Stärken, Grenzen und Verbesserungsmöglichkeiten des hinterlegten Moduls aufgezeigt. Ferner werden das Finite-Volumen- und das Finite-Elemente-Modul hinsichtlich der Eignung zur Simulation von Fluid-Struktur-Interaktionen untersucht und schließlich ein Testfall mit diesen unterschiedlichen Methoden untersucht sowie die Ergebnisse verglichen und diskutiert.

Alle neu gewonnenen Erkenntnisse dieser Arbeit werden im letzten Kapitel zusammengefasst.

Kapitel 2

Strömungsmechanik

Strömungen allgemeiner Fluide werden vollständig durch die Erhaltungsgleichungen für Masse, Impuls und Energie sowie durch die Anwendung geeigneter Zustandsgesetze beschrieben. Diese Gleichungen liefern die Lösungsgrößen des Geschwindigkeitsvektors $\mathbf{v}$, des Drucks p, der Temperatur T und der Dichte ρ.
Die die Strömungsmechanik beschreibenden Gleichungen werden im Folgenden in der Eulerschen Betrachtungsweise geschrieben. Bei dieser Formulierung wird der Fluss durch ein feststehendes Kontrollvolumen betrachtet. Wird der Strömungsraum während einer Simulation verformt, wie beispielweise bei der Simulation von rotierenden Laufrädern in Turbomaschinen oder bei FSI-Simulationen, wird das Raum-Erhaltungs-Gesetz (Space-Conservation-Law, SCL) angewandt, wodurch man eine gemischte Lagrange-Euler-Beschreibung (Arbitrary-Lagrange-Euler, ALE) erhält.
Für die später verwendete Finite-Volumen-Diskretisierung erweist sich die vektorielle, integrale Schreibweise der Gleichungen als besonders vorteilhaft. Die auftretenden Integrale werden für ein beliebiges Kontrollvolumen dV gebildet und - sofern möglich - mit Hilfe des Integralsatzes von Gauss in Oberflächenintegrale umgewandelt.

2.1 Massenerhaltung

In Vektorschreibweise lautet die Erhaltungsgleichung für die Masse:

$$\int_V \frac{\partial \rho}{\partial t} dV + \int_S \rho \mathbf{v}\mathbf{n} dS = 0. \tag{2.1}$$

Dabei stehen auf der linken Seite der Zeitterm und der konvektive Term. Quellen oder Senken auf der rechten Seite der Massenerhaltungsgleichung werden hier nicht berücksichtigt. Im inkompressiblen Fall lässt sich vereinfachen:

$$\int_S \mathbf{v}\mathbf{n} dS = 0. \tag{2.2}$$

2.2 Impulserhaltung

Die Impulserhaltungsgleichungen in integraler Vektorschreibweise lauten:

$$\int_V \frac{\partial \rho \mathbf{v}}{\partial t} dV + \int_S \rho \mathbf{v}\mathbf{v}\mathbf{n} dS = \int_S \boldsymbol{\tau}\mathbf{n} dS - \int_S \mathbf{I} p \mathbf{n} dS + \int_V \rho \mathbf{g} dV. \tag{2.3}$$

Auf der linken Seite stehen die zeitliche Änderung und der konvektive Transport des Impulses, auf der rechten Seite der Schubspannungs-, der Druckkraft- und der Gewichtskraftterm mit dem Ortsfaktor $\mathbf{g}$. $\mathbf{I}$ bezeichnet den Einheitstensor.
Diese Gleichungen sind prinzipiell für jedes Fluid anwendbar, eine Spezialisierung ergibt sich erst durch die Modellierung des Schubspannungsterms. Im Rahmen dieser Arbeit wird die Hypothese nach Stokes verwendet, die für Newtonsche Fluide lautet:

$$\boldsymbol{\tau} = \mu \left[(\nabla \mathbf{v} + \nabla \mathbf{v}^T) - \frac{2}{3} \nabla \mathbf{v} \mathbf{I} \right]. \tag{2.4}$$

Darin verschwindet der zweite Term in den eckigen Klammern bei der Betrachtung eines inkompressiblen Fluids aufgrund der Massenerhaltung, s. Gl. 2.2.
Man fasst die Massen- und Impulserhaltung unter Verwendung der Stokes-Hypothese unter dem Namen *Navier-Stokes-Gleichungen* oder unter der Bezeichnung *Hydrodynamisches Grundsystem* zusammen.

Für die im Rahmen dieser Arbeit betrachteten inkompressiblen und isothermen Strömungen genügt das *Hydrodynamische Grundsystem* zur Bestimmung des unbekannten Geschwindigkeits- und Druckfeldes. Es können mit dem Verfahren jedoch auch kompressible Flüssigkeiten und Gasströmungen untersucht werden. In diesen Fällen müssen zusätzlich die Energieerhaltungsgleichung und geeignete Zustands- und Stoffgesetze zur Bestimmung der Temperaturverteilung und der nicht konstanten Dichte und Viskosität gelöst werden, welche im Anhang C aufgeführt werden.

2.3 Turbulenzmodellierung

Die bisher vorgestellten Gleichungen gelten für laminare wie turbulente Strömungen. Ohne zusätzliche Annahmen stellt dieses Gleichungssystem die Basis für die direkte numerische Simulation (DNS) dar. Bei diesem Verfahren müssen die dreidimensionalen, instationären Gleichungen gelöst werden, um das chaotische und zeitlich komplexe Verhalten von Strömungen korrekt wiedergeben zu können. Für praktische Problemstellungen ist dieser Ansatz aufgrund enormer Rechenzeiten noch nicht verwendbar, weswegen im Rahmen dieser Arbeit sog. statistische Turbulenzmodelle zur Anwendung kommen. Eine kurze Herleitungsskizze zur Gewinnung derartiger Turbulenzmodelle findet sich im Anhang B.

Die später vorgestellten Simulationen werden mit dem $k\epsilon$-Turbulenzmodell unter Verwendung einer Standardwandfunktion durchgeführt. Implementierungsspezifische Details zu diesem Turbulenzmodell und den zugehörigen Randbedingungen finden sich ebenfalls im Anhang B.0.4.1 und B.0.5.

Ferner wurden im Verfahren noch das $k\omega$-Turbulenzmodell und das SST-Turbulenzmodell im modularen Lösungsverfahren hinterlegt. Auch diese Turbulenzmodelle werden mit ausführlichen Details zur Implementierung der zugehörigen Randbedingungen im Anhang beschrieben, s. B.0.4.2, B.0.4.3 und B.0.5. .

2.4 Rotierendes Bezugssystem

Gerade bei Strömungen in Turbomaschinen, wie sie in den abschließenden Kapiteln vorgestellt werden, spielen Trägheitskräfte, die durch Stomlinienkrümmung und die Rotation des Bezugssystems hervorgerufen werden, eine große Rolle. Die Stromlinienkrümmung ist implizit in den Impulsgleichungen enthalten, die Rotation des Koordinatensystems muss durch eine geeignete Transformation der Gleichungen berücksichtigt werden. Durch diese Transformation in ein rotierendes, körperfestes Bezugssystem lässt sich die eigentlich instationäre Strömung z. B. durch ein rotierendes Laufrad als stationär betrachten und damit auch mit einer stationären Simulation erfassen. Der Rechenaufwand sinkt damit erheblich.
Transformiert man mit Hilfe der Beziehung

$$\mathbf{v} = \mathbf{w} + \mathbf{c}, \tag{2.5}$$

wobei $\mathbf{v}$ die Absolut-, $\mathbf{w}$ die Relativ- und $\mathbf{c} = \mathbf{\Omega}^F \times \mathbf{r}$ die Führungsgeschwindigkeit mit der Führungs-Winkelgeschwindigkeit $\mathbf{\Omega}^F$ darstellt, entstehen zusätzlich die auf die Strömung wirkende Zentrifugal- und Coriolisbeschleunigung:

$$\mathbf{a}^Z = \mathbf{\Omega}^F \times \left(\mathbf{\Omega}^F \times \mathbf{r}\right), \tag{2.6}$$

$$\mathbf{a}^C = 2\mathbf{\Omega}^F \times \mathbf{w}. \tag{2.7}$$

Diese Kräfte müssen in den Impulsgleichungen durch zusätzliche Quellterme berücksichtigt werden. Die transformierte Massen- und Impulserhaltung lauten nun:

$$\int_V \frac{\partial \rho}{\partial t} dV + \int_S \rho \mathbf{w}\mathbf{n} dS = 0, \tag{2.8}$$

$$\int_V \frac{\partial \rho \mathbf{w}}{\partial t} dV + \int_S \rho \mathbf{w}\mathbf{w}\mathbf{n} dS = \int_S \boldsymbol{\tau}\mathbf{n} dS - \int_S \mathbf{I} p \mathbf{n} dS + \int_V \rho \mathbf{g} dV + \int_V \mathbf{q}^{rot} dV, \tag{2.9}$$

$$\mathbf{q}^{rot} = -\rho \mathbf{\Omega}^F \times \left(\mathbf{\Omega}^F \times \mathbf{x}\right) - 2\rho \mathbf{\Omega}^F \times \mathbf{w}. \tag{2.10}$$

Eine numerische Behandlung der Impulsgleichungen neigt gerade bei hohen Corioliskräften zu Instabilitäten. Weiterhin ist der Übergang von einem rotierenden in ein stehendes Bezugssystem schwierig zu implementieren.
Nach KROLL [64] können die Gleichungen aber so transformiert werden, dass die Lösungsvariable weiterhin die Absolutgeschwindigkeiten bleiben, die durch die relativen Massenströme transportiert werden. Durch diese Umformung können Teile der entstandenen Quellterme implizit verarbeitet werden, wodurch die numerische Stabilität beim Lösen stark erhöht wird. Die Massen- und die Impulserhaltung mit den modifizierten Quelltermen $\mathbf{S}$ lauten nun:

$$\int_V \frac{\partial \rho}{\partial t} dV + \int_S \rho \mathbf{v}\mathbf{n} dS = 0, \tag{2.11}$$

$$\int_V \frac{\partial \rho \mathbf{v}}{\partial t} dV + \int_S \rho \mathbf{v}\mathbf{w}\mathbf{n} dS = \int_S \boldsymbol{\tau}\mathbf{n} dS - \int_S \mathbf{I} p \mathbf{n} dS + \int_V \rho \mathbf{g} dV + \int_V \mathbf{q}^{mod} dV, \tag{2.12}$$

$$\mathbf{q}^{mod} = -2\rho \boldsymbol{\Omega}^F \times \mathbf{v}. \tag{2.13}$$

Details zur Transformation der Energieerhaltung in ein rotierendes Bezugssystem finden sich im Anhang A.1.1.

Auch in allen weiteren Transportgleichungen, wie z. B. in der Gleichung für die turbulente kinetische Energie oder für die turbulente Dissipation, wird der konvektive Transport durch die relativen Massenströme ausgedrückt. Weitere Modifikationen, welche rotationsbedingte, turbulente Effekte berücksichtigen, werden im Rahmen dieser Arbeit nicht untersucht. Hierfür sei auf die Arbeiten von BARDINA [14], HOWARD [54] und KLEMENT [63] verwiesen.

Kapitel 3

Strukturmechanik

Einer der Schwerpunkte dieser Arbeit liegt in der Implementierung eines Moduls zur numerischen Behandlung der Grundgleichungen der Strukturmechanik. Deswegen werden diese in diesem Kapitel in verschiedenen im Verfahren hinterlegten Formulierungen ausführlich hergeleitet.
Die Herleitung vollzieht sich in drei Schritten.

- Zunächst wird der Zusammenhang zwischen Dehnung und Verschiebung formuliert.
- Es folgt die Beschreibung des Zusammenhangs zwischen Dehnung und Spannung, des so genannten Materialgesetzes.
- Schließlich muss das Kräftegleichgewicht aufgestellt werden, um die Impulserhaltungsgleichungen für die Strukturmechanik zu erhalten.

Diesen Punkten vorangestellt ist noch die exakte Beschreibung der Konfiguration eines Körpers.
Für eine einfache Umsetzbarkeit in eine FV-Formulierung wird im Folgenden die Vektor- und Tensorschreibweise verwendet.

3.1 Konfiguration eines Körpers

Dieser Abschnitt erläutert die Beschreibung der allgemeinen Bewegung eines Körpers. Die momentane Lage des Körpers wird als Konfiguration bezeichnet. Bei großen Verschiebungen ändert sich diese Konfiguration ständig. Eine effiziente und elegante mathematische Behandlung erfolgt durch eine inkrementelle Zerlegung des Bewegungs- oder Deformationsvorgangs. Damit lassen sich dann auch die Dehnungen und die Spannungen inkrementell zerlegen.

Die Beschreibung der kontinuierlichen Änderung der Konfiguration eines Körpers wird in einem feststehenden kartesischen Koordinatensystem betrachtet. Die Lage eines Punktes P in der undeformierten Anfangs- oder Initial-Konfiguration wird

durch den Ortsvektor $\mathbf{x}^0$ beschrieben. Zu einem neuen Zeitpunkt oder Zeitschritt n befindet sich der Punkt am neuen Ort $\mathbf{x}^n$.
In Abb. 3.1 ist ein Körper mit einem Punkt P zu diesen beiden Zeitpunkten und zu einem weiteren Zeitpunkt $n+1$ dargestellt. Im Verlauf der Berechnung folgt man allen Punkten des Körpers von der Initial- zur Endkonfiguration. Man nennt diese Vorgehensweise die *Lagrangesche Betrachtungsweise* im Gegensatz zur in der Strömungsmechanik hauptsächlich *verwendeten Eulerschen Betrachtungsweise.*

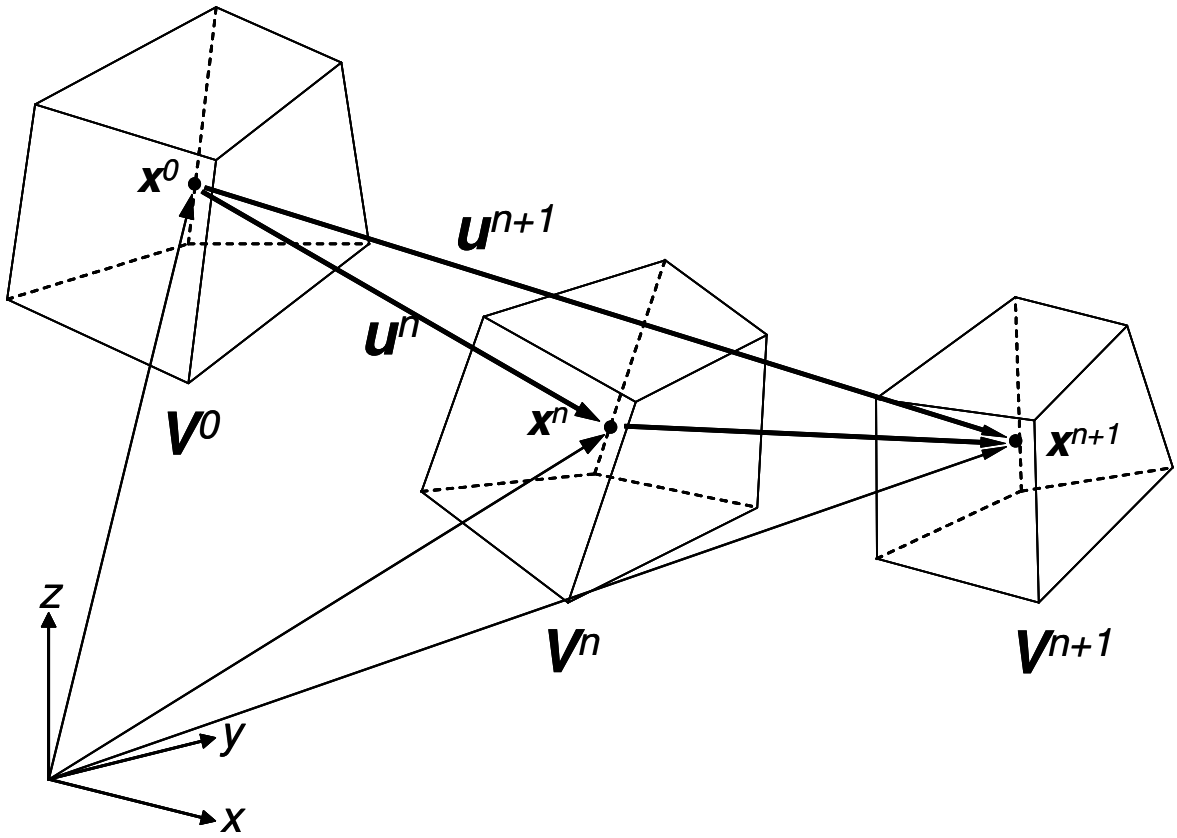

Abbildung 3.1: Konfiguration eines Körpers in den Zeitpunkten t_0, t_n und t_{n+1}

Die Ortsvektoren $\mathbf{x}$ des Punktes P zum Zeitpunkt 0 und zum Zeitpunkt n wird durch die Verschiebung $\mathbf{u}^n$ verknüpft. Analog geschieht das mit der Lage zum Zeitpunkt n und zum Zeitpunkt $n+1$ durch die Verschiebung $\mathbf{u}^{n+1}$:

$$\mathbf{x}^n = \mathbf{x}^0 + \mathbf{u}^n \tag{3.1}$$

bzw.

$$\mathbf{x}^{n+1} = \mathbf{x}^0 + \mathbf{u}^{n+1}. \tag{3.2}$$

Der Zuwachs der Verschiebung vom Zeitpunkt n zum Zeitpunkt n+1 lautet dann:

$$\mathbf{u} = \mathbf{u}^{n+1} - \mathbf{u}^n. \tag{3.3}$$

Nun betrachtet man die Konfiguration des Punktes zum Zeitpunkt n. Ein infinitesimal kleines, materielles Linienelement $d\mathbf{x}^0$ wurde deformiert und wird zum Zeitpunkt n

mit $d\mathbf{x}^n$ bezeichnet. Mit Hilfe der Kettenregel erhält man die folgende Verknüpfung zwischen diesen beiden Größen:

$$d\mathbf{x}^n = \frac{\partial \mathbf{x}^n}{\partial \mathbf{x}^0} d\mathbf{x}^0. \tag{3.4}$$

Mit dieser Gleichung hat man nun eine Beziehung zwischen undeformierter und deformierter Lage gewonnen. Die Abbildung selbst ist durch den sog. Deformations-Gradienten-Tensor (DGT) gegeben:

$$\mathbf{F}_0^n = \mathbf{F} = \frac{\partial \mathbf{x}^n}{\partial \mathbf{x}^0} = \nabla \mathbf{x}. \tag{3.5}$$

Er stellt einen nicht-symmetrischen Tensor zweiter Stufe dar und beschreibt die Dehnung und Drehung des betrachteten Körpers. Der Tensor wird zum Zeitpunkt 0 aufgestellt und bezieht sich somit auf die Initialkonfiguration.
Eine alternative Darstellung für den DGT erhält man durch Ableitung von Gl. 3.1:

$$\frac{\partial \mathbf{x}^n}{\partial \mathbf{x}^0} = \frac{\partial \mathbf{x}^0}{\partial \mathbf{x}^0} + \frac{\partial \mathbf{u}^n}{\partial \mathbf{x}^0}. \tag{3.6}$$

In dieser Gleichung stellt der erste Ausdruck der rechten Seite den Einheitstensor $\mathbf{I}$ dar, und man erhält:

$$\mathbf{F}_0^n = \mathbf{F} = \mathbf{I} + \frac{\partial \mathbf{u}^n}{\partial \mathbf{x}^0} = \mathbf{I} + \nabla \mathbf{u}. \tag{3.7}$$

Da der DGT eine Abbildung von der undeformierten zur deformierten Lage darstellt, muss ebenfalls eine Abbildung von der deformierten zur undeformierten Lage der Form $\mathbf{F}^{-1}$ existieren.
Bei der späteren Verwendung werden die Indizes der Tensoren der Übersichtlichkeit halber oft weggelassen. Die verwendeten Tensoren beziehen sich deshalb im Folgenden immer auf eine definierte Referenzkonfiguration.

3.2 Zusammenhang von Dehnung und Verschiebung

Mit den nun vorhandenen Verformungsmaßen des Deformations- und des Verschiebungsgradienten wird in diesem Abschnitt ein Dehnungsmaß eingeführt, welches eine effiziente Simulationsmethode ermöglicht. Dazu betrachtet man zunächst die Länge eines materiellen Linienelements $d\mathbf{l} = d\mathbf{x}$ in der undeformierten und die Länge eines materiellen Linienelements in der deformierten Lage eines Körpers. Mit der Differenz dieser Längen definiert man den Green-Lagrange-Dehnungs-Verzerrungs-Tensor (GLDVT) $\mathbf{E}$:

$$\frac{1}{2}\left((d\mathbf{l}^n)^T d\mathbf{l}^n - \left(d\mathbf{l}^0\right)^T d\mathbf{l}^0\right) = \frac{1}{2}\left((d\mathbf{x}^n)^T d\mathbf{x}^n - \left(d\mathbf{x}^0\right)^T d\mathbf{x}^0\right) = (d\mathbf{x}^n)^T \mathbf{E} d\mathbf{x}^n. \tag{3.8}$$

Mit Hilfe des DGT, s. Gl. 3.7, ersetzt man die Terme in der neuen Konfiguration:

$$\left(d\mathbf{x}^0\right)^T \mathbf{F}^T \mathbf{F} d\mathbf{x}^0 - \left(d\mathbf{x}^0\right)^T \mathbf{I} d\mathbf{x}^0 = 2\left(d\mathbf{x}^0\right)^T \mathbf{E} d\mathbf{x}^n \tag{3.9}$$

und fasst zusammen:

$$\left(d\mathbf{x}^0\right)^T \left(\mathbf{F}^T\mathbf{F} - \mathbf{I}\right) d\mathbf{x}^0 = 2\left(d\mathbf{x}^0\right)^T \mathbf{E} d\mathbf{x}^n. \tag{3.10}$$

Somit erhält man für den GLDVT

$$\mathbf{E} = \frac{1}{2}\left(\mathbf{F}^T\mathbf{F} - \mathbf{I}\right), \tag{3.11}$$

bzw. gewinnt man mit der Beziehung für den DGT Gl. 3.5 folgenden Ausdruck in Abhängigkeit der Verschiebungen:

$$\mathbf{E} = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T + \nabla\mathbf{u}\nabla\mathbf{u}^T\right). \tag{3.12}$$

Man kann zeigen, dass dieser Tensor unabhängig von einer Starrköper-Drehung ist, s. LIPPMANN [68].

Ferner erkennt man einen linearen und einen nichtlinearen Anteil und schreibt:

$$\mathbf{E} = \mathbf{E}_{lin} + \mathbf{E}_{nlin} \tag{3.13}$$

mit

$$\mathbf{E}_{lin} = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right) \tag{3.14}$$

und

$$\mathbf{E}_{nlin} = \frac{1}{2}\left(\nabla\mathbf{u}\nabla\mathbf{u}^T\right). \tag{3.15}$$

Der lineare Anteil wird als infinitesimaler Dehnungs-Verzerrungs-Tensor (DVT) bezeichnet. Der nichtlineare Anteil enthält die geometrischen Nichtlinearitäten. Bei der Berechnung mit kleinen Verschiebungen kann der nichtlineare Anteil vernachlässigt werden, da dann auch der Verschiebungsgradient klein ist.
Die Komponenten der Hauptdiagonalen des GLDVT beschreiben somit die Längenänderung in die jeweilige Raumrichtung, die Nebendiagonalelemente die Winkeländerungen bzw. die Verzerrungen des betrachteten Elements. Dies wird in der folgenden Abb. 3.2 für den zweidimensionalen Fall veranschaulicht.

Die wichtigen Erkenntnisse bei der Herleitung des GLDVT werden im Folgenden zusammengefasst:

- Bei der Herleitung des GLDVT wurde keine Annahme bezüglich der Größe der auftretenden Verformungen gemacht, er ist für jeden beliebig großen Verformungszustand gültig und unabhängig von Starrkörperrotationen.
- Der hergeleitete GLDVT besteht aus einem linearen und einem nichtlinearen Anteil. Alle Berechnungen mit großen Verformungen sind nichtlinear.
- Bei Berechnung kleiner Verformungszustände kann der nichtlineare Anteil vernachlässigt werden.
- Die auftretenden Gradienten sind alle auf die Ausgangs- oder Initialkonfiguration bezogen. Dehnungen und Verzerrungen werden deswegen auf diese Initialkonfiguration bezogen berechnet.

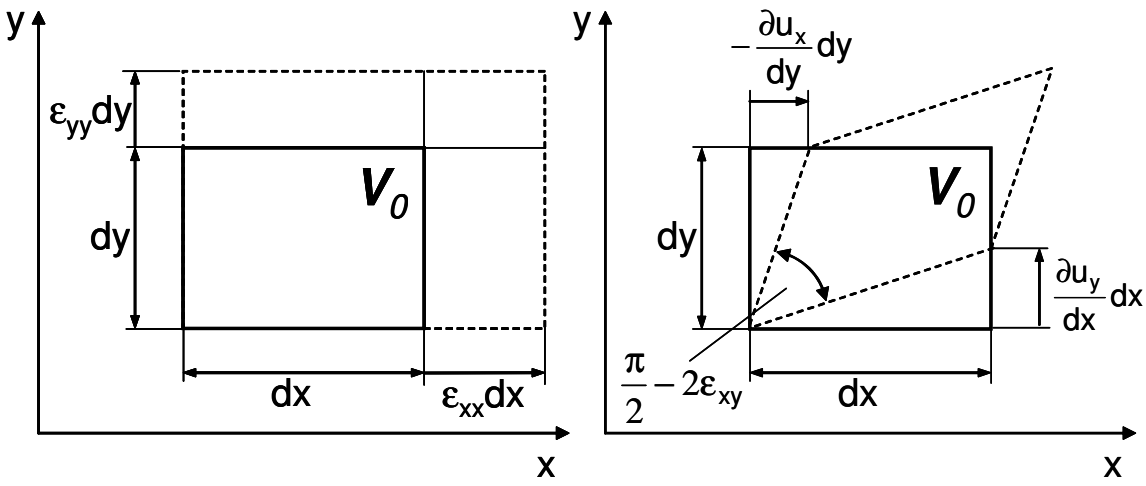

Abbildung 3.2: Anschauliche Herleitung von Dehnungen und Verzerrungen an einem ebenen, zweidimensionalen Element

Genauso wie man einen geeigneten Dehnungstensor zur Beschreibung beliebiger Verformung eines Körpers gefunden hat, muss nun noch zur Beschreibung des Spannungszustands in einem Körper ein geeigneter Spannungstensor gefunden werden.
Der Cauchy-Spannungstensor $\boldsymbol{\sigma}$ wird bei linear-elastischen Verformungen verwendet. Das geeignete Dehnungs-Verzerrungs-Maß ist der lineare Anteil des GLDVT, siehe Gl. 3.3. Die Spannung wird als Kraft auf eine unverformte Fläche definiert.

Werden die Verformungen aber groß, sind die Flächen der verformten Konfiguration nicht mehr bekannt. Man leitet sich deswegen einen Spannungstensor her, der in der Referenzkonfiguration mit den aktuell unverformten Flächen verwendet werden kann.
Dazu werden zunächst alle auftretenden Längen, Flächen und Volumina aus Ausgangskonfiguration 0 und Endkonfiguration 1 in Beziehung zueinander gesetzt. Für ein Längenelement des betrachteten Körpers erhält man:

$$d\mathbf{x}^1 = \mathbf{F}d\mathbf{x}^0. \tag{3.16}$$

Flächenelemente werden mit der *Nansonschen Formel* transformiert:

$$\mathbf{n}^1 dA^1 = J\mathbf{F}^{-1}\mathbf{n}^0 dA^0. \tag{3.17}$$

Der Jakobische Operator J ist die Determinante des Deformationstensors und liefert die Transformationsbeziehung für die Volumina:

$$dV^1 = det\mathbf{F}dV^0 = JdV^0. \tag{3.18}$$

Im Sonderfall des inkompressiblen Materials gilt $J = 1$.
Mit Hilfe dieser Beziehungen können die sog. *Piola-Kirchhoff-Spannungs-Tensoren* (PKST) abgeleitet werden. Der erste PKST ist aufgrund seiner Nichtsymmetrie und in Bezug auf die Ableitung geeigneter Materialgesetze schwierig zu handhaben und wird im Rahmen dieser Arbeit nicht weiter verwendet.

Der zweite PKST $\mathbf{\Sigma}$ ist so definiert, dass er, multipliziert mit der Fläche in der Ausgangskonfiguration, die in die Ausgangskonfiguration transformierte Kraft F ergibt:

$$\mathbf{dF}^0 = \mathbf{\Sigma n} dA^1. \tag{3.19}$$

Mit dem Deformationsgradienten wird nun eine direkte Beziehung zwischen dem zweiten PKST und dem Cauchy-Spannungs-Tensor formuliert:

$$\mathbf{dF}^0 = \mathbf{F}^{-1}\boldsymbol{\sigma}\mathbf{n} dA^1. \tag{3.20}$$

Wird nun noch die Fläche in der Ausgangskonfiguration mit Hilfe der Nansonschen Formel transformiert, ergibt sich:

$$\mathbf{\Sigma n} dA^0 = \mathbf{F}^{-1}\boldsymbol{\sigma} J \left(\mathbf{F}^T\right)^{-1} \mathbf{n} dA^0 \tag{3.21}$$

und

$$\mathbf{\Sigma} = J\mathbf{F}^{-1}\boldsymbol{\sigma} \left(\mathbf{F}^T\right)^{-1}. \tag{3.22}$$

Dieser Spannungstensor ist symmetrisch und mit dem GLDVT energetisch konsistent. Das bedeutet, dass die Verformungsenergie, die sich aus der Multiplikation des 2. PKSTs mit dem vollständigen GLDVT ergibt, dieselbe ist, welche aus der Multiplikation des Cauchy-Spannungs-Tensors und dem linearen Anteil des GLDVT resultiert:

$$\boldsymbol{\sigma\epsilon} = \mathbf{\Sigma E}. \tag{3.23}$$

Der Vorteil ist, dass die linke Seite der Gleichung in der unbekannten verformten Konfiguration, die rechte Seite dagegen in der Momentan- oder Referenzkonfiguration bestimmt werden kann.

3.3 Gleichgewichtsbeziehungen

Nun müssen noch Gleichgewichtsbeziehungen und ein geeignetes Materialgesetz gefunden werden. Für erstere benötigt man geeignete Spannungstensoren. Infolge einer allgemeinen Belastung erfährt ein Körper beliebig große Verschiebungen und befinde sich zum Zeitpunkt n im Gleichgewicht.

In einem solchen Zustand ist der Spannungszustand an einem beliebigen Punkt P des Körpers durch drei zueinander orthogonale Schnitte mit den in ihnen wirkenden Schnittspannungen bestimmt. Diese drei Schnittspannungsvektoren werden nun noch in ihre Komponenten parallel zu den Achsenrichtungen des kartesischen Koordinatensystems zerlegt und man erhält damit die neun Komponenten des Spannungstensors:

$$\boldsymbol{\sigma} = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{pmatrix}. \tag{3.24}$$

Verwendet man die Indexschreibweise, signalisiert der erste Index die Wirkebene, charakterisiert durch deren Normalenvektor, der zweite dagegen kennzeichnet die

Wirkrichtung der Spannung in dieser Ebene. Dieser Spannungstensor wird Cauchy-Spannungs-Tensor (CST) genannt.

Nun kann das Kräftegleichgewicht über den Spannungszustand an einem Punkt P, welcher in einer beliebig orientierten Schnittebene mit der Oberflächennormale $\mathbf{n}$ liegt, gebildet werden. Dieses Kräftegleichgewicht liefert dann den gesuchten Schnittspannungsvektor $\mathbf{t}$:

$$\mathbf{t} = \boldsymbol{\sigma}\mathbf{n}. \tag{3.25}$$

Da Spannungen in der Regel in einem Körper stetig verteilt sind und eben nicht nur punktuell wirken, werden die Gleichgewichtsbedingungen für ein beliebiges, aus einem Körper geschnittenes Volumenelement um einen Punkt aufgestellt: An dessen Oberflächen wirken die Spannungsvektoren $\mathbf{t}$ und in dessen Schwerpunkt greift die Volumenkraft $\mathbf{f}$ an. Für eine statische Gleichgewichtslage muss die Summe aller Kräfte verschwinden:

$$\int_A \mathbf{t} dA + \int_V \mathbf{f} dV = \mathbf{0}. \tag{3.26}$$

Mit Gl. 3.25 ersetzt man den Spannungsvektor und erhält:

$$\int_A \boldsymbol{\sigma}\mathbf{n} dA + \int_V \mathbf{f} dV = \mathbf{0}. \tag{3.27}$$

Schließlich erhäjt man mit dem Integralsatz von Gauss:

$$\int_V \nabla\boldsymbol{\sigma}\mathbf{n} dV + \int_V \mathbf{f} dV = \mathbf{0}. \tag{3.28}$$

Mit einer Erweiterung um einen instationären Term wird diese Formulierung zur späteren numerischen Diskretisierung verwendet.
Neben dieser Beziehung werden die weiteren wichtigen Erkenntnisse dieses Abschnitts wie folgt zusammengefasst:

- Aus einer Momentenbetrachtung an einem Volumenelement erkennt man die Symmetrie des CST.
- Da dieser sich auf die aktuelle Konfiguration bezieht, sind dessen Komponenten bei einer Starrkörperdrehung nicht konstant.
- Das Kräftegleichgewicht wurde am verformten Körper aufgestellt, die Differentiale beziehen sich folglich auf die verformte Konfiguration.

3.4 Materialgesetz

3.4.1 Linear-elastisches und kompressibles Material

Die abgeleiteten Dehnungs- und Spannungstensoren müssen durch ein geeignetes Materialgesetz verknüpft werden. Es beschreibt, wie ein Körper auf eine äußere

Belastung reagiert. Eine äußere Belastung induziert einerseits Verformungen und andererseits Spannungen.
Das Materialgesetz kann nur experimentell gewonnen werden. Bei einem in allen Punkten eines Körpers gleichen Materialverhalten spricht man von einem *homogenen*, ansonsten von einem *inhomogenen* Materialverhalten. Weiterhin heißt ein Werkstoff *isotrop*, wenn dessen Eigenschaften von der Richtung unabhängig sind, anderenfalls *anisotrop*.

Bei vielen Werkstoffen stellt man eine eindeutige Zuordnung von Dehnung und Spannung zu einem Zeitpunkt n fest, was eine Unabhängigkeit von der Deformationsgeschichte bedeutet. Nimmt der deformierte Körper nach Entlastung wieder seine ursprüngliche Gestalt an, nennt man das Verhalten elastisch. Im Falle eines linear-elastischen Materialverhaltens lässt sich formulieren:

$$\boldsymbol{\sigma} = \mathbf{CE}, \tag{3.29}$$

was eine lineare Abbildung zwischen zwei Tensoren zweiter Stufe darstellt. Der Elastizitätstensor $\mathbf{C}$ ist demzufolge ein Tensor vierter Stufe mit 81 Komponenten, siehe GROSS [44]. Durch die Symmetrie des Spannungstensors und des Dehnungstensors unterliegt dieser ebenfalls gewissen Symmetrieeigenschaften.
Im Rahmen dieser Arbeit wird im Folgenden nur linear-elastisches, homogenes und isotropes Materialverhalten vorausgesetzt. Damit ergibt sich für den Elastizitätstensor:

$$\mathbf{C} = \frac{E}{1+\nu} \begin{pmatrix} \frac{1-\nu}{1-2\nu} & \frac{\nu}{1-2\nu} & \frac{\nu}{1-2\nu} & 0 & 0 & 0 \\ \frac{\nu}{1-2\nu} & \frac{1-\nu}{1-2\nu} & \frac{1-\nu}{1-2\nu} & 0 & 0 & 0 \\ \frac{\nu}{1-2\nu} & \frac{1-\nu}{1-2\nu} & \frac{1-\nu}{1-2\nu} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & & \frac{1}{2} & 0 \\ 0 & 0 & 0 & & & \frac{1}{2} \end{pmatrix}. \tag{3.30}$$

Ferner werden die Lamé-Konstanten λ und μ eingeführt, welche auch als Kompressions- und Schub- oder Querkontraktionsmodul bezeichnet werden. Sie lassen sich mit Hilfe des Elastizitätsmoduls E und der Querkontraktions- oder Poisson-Zahl ν im allgemeinen dreidimensionalen Fall durch

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \tag{3.31}$$

und

$$\mu = \frac{E}{2(1+\nu)} \tag{3.32}$$

ausdrücken.

3.4.2 Linear-elastisches Material

Ein elastischer Werkstoff besitzt demnach zwei unabhängige Materialkonstanten. Für den Spannungstensor $\boldsymbol{\sigma}$ schreibt man mit diesen Konstanten und Beziehung Gl. 3.30:

$$\boldsymbol{\sigma} = 2\mu\mathbf{E} + \lambda tr\,(\mathbf{E})\,\mathbf{I}. \tag{3.33}$$

Für zweidimensionale, ebene Spannungszustände muss die erste Lamé-Konstante λ modifiziert werden:

$$\lambda = \frac{E\nu}{(1+\nu)(1-\nu)}. \tag{3.34}$$

Je nachdem, ob im Dehnungs-Verzerrungs-Tensor $\mathbf{E}$ nur lineare Anteile oder auch der nichtlineare Anteil berücksichtigt wird, siehe Gl. 3.12, erhält man das *Hooke*-Materialgesetz zur Beschreibung eines linear elastischen Materials oder das *Saint-Venant*-Materialgesetz zur Beschreibung eines geometrisch-nichtlinearen, hyperelastischen Materials. Das betrachtete Material ist kompressibel.

Für die Beschreibung eines inkompressiblen Materials müssen wegen $\nu = 0,5$ und der damit nicht definierten zweiten Lamé-Konstante andere Materialgesetze gefunden werden.
Beispielsweise kann der hydrostatische Druck als zusätzliche Lösungsvariable eingeführt werden:

$$\boldsymbol{\sigma} = 2\mu\mathbf{E} - p\mathbf{I}, \tag{3.35}$$

s. z. B. LIPPMANN [68]. Als zusätzliche Bestimmungsgleichung muss dann eine kinematische Bedingung gewählt werden:

$$\nabla\mathbf{u} = \mathbf{0}, \tag{3.36}$$

welche die Volumenerhaltung beschreibt.

3.5 Impulserhaltung

Die Impulserhaltung für die Strukturberechnung wird je nach Anwendungsbereich in verschiedenen Formulierungen verwendet. Bei der Berechnung eines geometrisch linearen und elastischen Verformungszustandes verwendet man die sog. Cauchy-Formulierung (CF), unter der Annahme kleiner Verformungen, dass keine geometrische Unterscheidung zwischen unverformtem und verformtem Körper gemacht werden muss.

Wird dagegen eine Festigkeitsberechnung mit großen, geometrisch-nichtlinearen Verformungen untersucht, verwendet man die Lagrange-Formulierung (LF) der Impulserhaltung, welche weiter in die Totale-Lagrange-Formulierung (TLF) und die Fortschreitende-Lagrange-Formulierung (FLF) unterteilt wird.
Diese unterschiedlichen Ansätze werden in den folgenden Abschnitten erklärt und mit deren Hilfe die Impulserhaltung in eine zur Diskretisierung geeignete Form gebracht.

3.5.1 Cauchy-Formulierung

Bei der Cauchy-Formulierung sind die Verformungen so klein, dass keine geometrische Unterscheidung zwischen unverformter und verformter Geometrie gemacht wird. Damit entsprechen bei dieser Formulierung die Inkremente der Verschiebungen den Absolut-Verschiebungen:

$$\mathbf{u} := \delta\mathbf{u}. \tag{3.37}$$

Die Gleichgewichtslage eines durch eine Volumenkraft $\mathbf{f}$ belasteten Körpers mit dem beliebigen Volumen V und mit der Dichte ρ lautet in integraler Form:

$$\int_V \frac{\partial^2 (\rho \mathbf{u})}{\partial t^2} dV - \int_V \nabla \boldsymbol{\sigma} dV = \int_V \rho \mathbf{f} dV, \tag{3.38}$$

wobei im Gegensatz zu Gl. 3.28 nun auch der Zeitterm auf der linken Seite mit berücksichtigt wurde.
Mit Hilfe des Satzes von Gauss wird für die spätere Diskretisierung der zweite Term der linken Seite in ein Oberflächenintegral umgewandelt:

$$\int_V \frac{\partial^2 (\rho \mathbf{u})}{\partial t^2} dV - \int_S \boldsymbol{\sigma} \mathbf{n} dS = \int_V \rho \mathbf{f} dV. \tag{3.39}$$

Die Dehnungen und Verzerrungen $\mathbf{E}$ werden durch den linearen Anteil des Green-Lagrange-Verzerrungstensors ausgedrückt:

$$\mathbf{E} = \frac{1}{2} \left[\nabla \mathbf{u} + \nabla \mathbf{u}^T \right]. \tag{3.40}$$

Mit dem Hooke-Materialgesetz Gl. 3.33 werden die Verformungen mit dem Spannungstensor $\boldsymbol{\sigma}$ verknüpft:

$$\boldsymbol{\sigma} = 2\mu \mathbf{E} + \lambda tr(\mathbf{E}) \mathbf{I}. \tag{3.41}$$

Man erhält somit die integrale Form der Impulserhaltungsgleichungen in Cauchy-Formulierung:

$$\int_V \frac{\partial^2 (\rho \mathbf{u})}{\partial t^2} dV - \int_S \nabla \left[\mu \nabla \mathbf{u} + \mu \nabla \mathbf{u}^T + \lambda \mathbf{I} tr(\nabla \mathbf{u}) \right] \mathbf{n} dS = \int_V \rho \mathbf{f} dV. \tag{3.42}$$

3.5.2 Totale-Lagrange-Formulierung

Mit der Totalen-Lagrange-Formulierung (TLF) können geometrisch-nichtlineare Verformungen in einfacher Weise untersucht werden. Dazu wird die Lösung in Teilschritte zerlegt. In jedem Teilschritt oder Zeitschritt werden Zuwächse oder Inkremente der Verschiebungen $\delta \mathbf{u}$ und Spannungen $\delta \boldsymbol{\sigma}$ bestimmt.

Die Impulserhaltung wird auf eine physikalische Initialkonfiguration 0 bezogen:

$$\int_{V_0} \frac{\partial^2 (\rho_0 \mathbf{u})}{\partial t^2} dV_0 - \int_{S_0} \boldsymbol{\Sigma} \mathbf{F}^T \mathbf{n}_0 dS_0 = \int_{V_0} \rho_0 \mathbf{f} dV_0. \tag{3.43}$$

An die Stelle des Cauchy-Spannungstensors tritt bei dieser Formulierung der mit dem DGT $\mathbf{F}$ transfomierte zweite PKST $\boldsymbol{\Sigma}$.
Um die inkrementellen Verschiebungen als Lösungsvariablen zu erhalten, wird nun die Impulserhaltung in inkrementeller Form angeschrieben:

$$\int_{V_0} \frac{\partial^2 (\rho_0 \delta \mathbf{u})}{\partial t^2} dV_0 - \int_{S_0} \left(\delta \boldsymbol{\Sigma} \mathbf{F}^T + \boldsymbol{\Sigma} \delta \mathbf{F}^T + \delta \boldsymbol{\Sigma} \delta \mathbf{F}^T \right) \mathbf{n}_0 dS_0 = \int_{V_0} (\rho_0 \delta \mathbf{f}) \, dV_0. \tag{3.44}$$

Das Inkrement des DGT ergibt sich mit:

$$\delta\mathbf{F} = \delta\nabla\mathbf{u}. \tag{3.45}$$

Das Inkrement des zweiten PKST lautet unter Verwendung des Saint-Venant-Materialgesetzes:

$$\delta\mathbf{\Sigma} = 2\mu\delta\mathbf{E}_0 + \lambda tr(\delta\mathbf{E}_0)\mathbf{I}, \tag{3.46}$$

wobei das Inkrement des auf die Initialkonfiguration bezogenen GLDVT $\mathbf{E}_0$ mit

$$\delta\mathbf{E}_0 = \frac{1}{2}\left[\nabla\delta\mathbf{u} + (\nabla\delta\mathbf{u})^T + \nabla\delta\mathbf{u}(\nabla\mathbf{u})^T + \nabla\mathbf{u}(\nabla\delta\mathbf{u})^T + \nabla\delta\mathbf{u}(\nabla\delta\mathbf{u})^T\right] \tag{3.47}$$

bestimmt wird.

Nach jedem inkrementellen Berechnungsschritt werden die neu bestimmten Inkremente der Verschiebungen und Spannungen dem alten Gesamtverschiebungs- bzw. Gesamtspannungstensor hinzuaddiert:

$$\mathbf{\Sigma} := \mathbf{\Sigma}^0 + \delta\mathbf{\Sigma} \tag{3.48}$$

und

$$\mathbf{u} := \mathbf{u}^0 + \delta\mathbf{u}. \tag{3.49}$$

Da stets die Initialkonfiguration als Referenz verwendet wird, muss das Rechennetz während der Simulation nicht verformt werden. Dies spart Rechenzeit, und die Formulierung eignet sich für stationäre Simulationen großer Verformungszustände bei reinen Strukturberechnungen.

3.5.3 Fortschreitende-Lagrange-Formulierung

Bei der Fortschreitenden-Lagrange-Formulierung (FLF) wird jeweils die zuletzt bestimmte Lösung während der ebenfalls inkrementellen Lösungsprozedur als Referenzkonfiguration u verwendet.

Man erhält die Impulsgleichungen in dieser inkrementellen Formulierung, indem man in den mit der TLF ausgedrückten Impulsgleichungen $\mathbf{F} = \mathbf{I}$ setzt, s. Gl. 3.44:

$$\int_{V_u} \frac{\partial^2 (\rho_u \delta\mathbf{u})}{\partial t^2} dV_u - \int_{S_u} \left(\delta\mathbf{\Sigma}_u + \mathbf{\Sigma}_u \delta\mathbf{F}_u^T + \delta\mathbf{\Sigma}_u \delta\mathbf{F}_u^T\right) \mathbf{n}_u dS_0 = \int_{V_u} (\rho_u \delta\mathbf{f})\, dV_u. \tag{3.50}$$

Für das Inkrement des GLDVT gilt nun:

$$\delta\mathbf{E}_u = \frac{1}{2}\left[\nabla\delta\mathbf{u} + (\nabla\delta\mathbf{u})^T + \nabla\delta\mathbf{u}(\nabla\delta\mathbf{u})^T\right]. \tag{3.51}$$

Das Inkrement des zweiten PKST lautet in der Fortschreitenden-Lagrange-Formulierung unter Anwendung des Saint-Venant-Materialgesetzes:

$$\delta\mathbf{\Sigma}_u = 2\mu\delta\mathbf{E}_u + \lambda tr(\delta\mathbf{E})\mathbf{I}. \tag{3.52}$$

Nach jedem Teilschritt werden wie bei der Totalen-Lagrange-Formulierung die neu bestimmten Zuwächse oder Inkremente der Verschiebungen und Spannungen zum Gesamtverschiebungs- bzw. Gesamtspannungstensor hinzuaddiert, bzw. kumuliert:

$$\mathbf{\Sigma} := \mathbf{\Sigma}^0 + \delta\mathbf{\Sigma} \tag{3.53}$$

und

$$\mathbf{u} := \mathbf{u}^0 + \delta\mathbf{u}. \tag{3.54}$$

Zusätzlich wird am Ende eines Zeitschritts das Rechengebiet mit den vorliegenden Verschiebungsinkrementen verformt und mit dem verformten Rechengebiet wird die neue Referenzkonfiguration definiert. Der zweite PKST wird in diese neue deformierte Referenzkonfiguration transformiert:

$$\mathbf{\Sigma}_u := \frac{1}{\det \mathbf{F}} \mathbf{F} \mathbf{\Sigma}^0 \mathbf{F}^T. \tag{3.55}$$

Diese Formulierung wird immer dann verwendet, wenn eine Netzverformung während des Simulationsprozesses notwendig ist, z. B. bei gekoppelten Fluid-Struktur-Interaktionen.

3.6 Rotierendes Bezugssystem

Werden stationäre Simulationen in einem Relativsystem durchgeführt, wie z. B. bei der Strukturanalyse eines rotierenden Laufrads in einer Turbomaschine, so müssen die auftretenden Fliehkräfte in allen Formulierungen als zusätzliche Volumenkraft $\mathbf{f}_V$ berücksichtigt werden:

$$\mathbf{f}_V = \mathbf{\Omega} \times (\mathbf{\Omega} \times \mathbf{x}) \,. \tag{3.56}$$

Ein zusätzlicher Quellterm durch die Coriolis-Kraft wie bei den Impulsgleichungen der Strömungsmechanik treten aufgrund des fehlenden konvektiven Terms nicht auf.

Kapitel 4

Numerische Verfahren

In den folgenden Abschnitten wird das im entwickelten Lösungsverfahren hinterlegte Finite-Volumen-Verfahren zur Diskretisierung partieller Differentialgleichungen ausführlich vorgestellt. Bei der Diskretisierung werden folgende Ziele verfolgt:

- Diskretisierungsgenauigkeit zweiter Ordnung sowohl für zeitliche als auch für räumliche Terme,
- die Anwendbarkeit auf komplexeste geometrische Strukturen und
- ein speicherschonendes, einfach zu parallelisierendes Verfahren.

4.1 Allgemeine Transportgleichung

Die in den Kapiteln 2 und 3 vorgestellten Erhaltungsgleichungen weisen alle eine ähnliche Gestalt auf, so dass sich eine allgemeine Transportgleichung (ATG) formulieren lässt:

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla(\rho\mathbf{u}\phi) = \nabla\Gamma\nabla\phi + q. \tag{4.1}$$

Diese besteht aus einem Zeit-, einem Konvektions-, einem Diffusions- und einem Quellterm.
Γ bezeichnet die sogenannte Diffusionskonstante und ϕ die transportierte Größe.
Je nachdem wie Γ und ϕ gewählt werden, erhält man die Erhaltungsgleichungen für Masse, Impuls, Energie und Turbulenz. Terme dieser zuletzt genannten Gleichungen, welche nicht in der ATG auftreten, werden im Quellterm zusammengefasst.
Für die später folgende Finite-Volumen-Diskretisierung wird die ATG über ein betrachtetes finites Volumen V integriert:

$$\int_V \left(\frac{\partial(\rho\phi)}{\partial t}\right) dV + \int_V (\nabla(\rho\mathbf{u}\phi))\, dV = \int_V (\nabla\Gamma\nabla\phi)\, dV + \int_V q dV. \tag{4.2}$$

Im Anschluss daran werden die Volumenintegrale des Konvektions- und des Diffusionsterms mit Hilfe des Satzes von Gauss in Oberflächenintegrale umgewandelt:

$$\int_V \left(\frac{\partial(\rho\phi)}{\partial t}\right) dV + \int_S (\rho\mathbf{u}\phi)\, \mathbf{n} dS = \int_S (\Gamma\nabla\phi)\, \mathbf{n} dS + \int_V q dV. \tag{4.3}$$

Die Fläche S beschreibt die Oberfläche des betrachteten finiten Volumens V.

Es wurden keinerlei Annahmen hinsichtlich der Gestalt des betrachteten Zellvolumens und seiner Oberfläche gemacht, so dass beliebig geformte, polyedrische Zellvolumina mit polygonalen Oberflächenteilen, s. Abb. 4.1, verwendet werden können. Hierdurch ist das Verfahren einfach auf sehr komplexe Geometrien anwendbar. Eine derartige polyedrische Volumenzelle ist in Abb. 4.1 dargestellt.

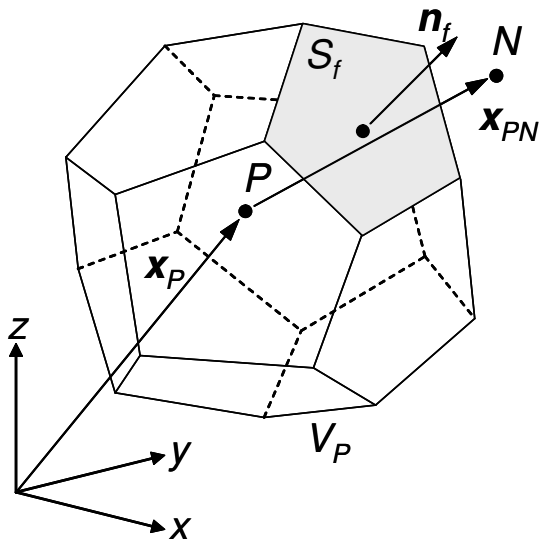

Abbildung 4.1: Polyedrische dreidimensionale finite Volumenzelle

Der Zellmittelpunktwert einer betrachteten Größe ϕ wird mit P gekennzeichnet, der unmittelbar benachbarte Wert mit N. Der Verbindungsvektor zwischen diesen beiden Punkten wird mit $\mathbf{x}_{PN}$ bezeichnet, der normierte Verbindungsvektor mit $\mathbf{d}$. Jede Zellfläche ist beschrieben durch ihren Mittelpunkt c, den Einheits-Normalenvektor $\mathbf{n}$ und den Zellflächeninhalt S.

Bei der vorgestellten Diskretisierungsmethode wird ein kompaktes Rechenmolekül angestrebt. Das bedeutet, dass bei der Diskretisierung einer Gleichung am Zellmittelpunkt P nur die unmittelbaren Nachbarn der Zelle in die implizite Diskretisierung der iterativen Lösungsprozedur miteinbezogen werden. Weiter entfernte Nachbarn werden aus Werten der alten Iteration berechnet und explizit in die iterative Lösungsprozedur miteinbezogen. Die so entstehenden Koeffizientenmatrizen der linearen Gleichungssysteme bleiben dann dünnbesetzt und das Lösungsverfahren speicherschonend.

Der Zellfächeninhalt S einer polygonalen Fläche wird mit Hilfe des Satzes von Gauss, den zugehörigen Kantenlängen $L_{x,i}$ und Kantenschwerpunkten $x_{c,i}$ bestimmt:

$$S = \int_S dS = \int_S \nabla \begin{pmatrix} x \\ 0 \\ 0 \end{pmatrix} dS = \int_L \begin{pmatrix} x \\ 0 \\ 0 \end{pmatrix} \mathbf{n} dL \approx \sum_i x_{c,i} L_i^x. \tag{4.4}$$

Der Zellflächenmittelpunkt $x_{c,S}$ ist der Schwerpunkt der betrachteten Zellfläche:

$$x_{c,S} = \frac{\sum_i x_{c,i} L_i}{\sum_i L_i}. \tag{4.5}$$

In analoger Weise wird das Volumen einer polyedrischen Zelle mit Hilfe des Satzes von Gauss, den zugehörigen Flächen und den Flächenschwerpunkten bestimmt:

$$V = \int_V dV = \int_V \nabla \begin{pmatrix} x \\ 0 \\ 0 \end{pmatrix} dV = \int_S \begin{pmatrix} x \\ 0 \\ 0 \end{pmatrix} \mathbf{n} dS \approx \sum_i x_{c,i} S_i^x. \tag{4.6}$$

Der Zellmittelpunkt $x_{c,V}$ ist der Schwerpunkt der betrachteten Zelle:

$$x_{c,V} = \frac{\sum_i x_{c,S,i} S_i}{\sum_i S_i}. \tag{4.7}$$

Dadurch dass die Summe der Teilvolumina das betrachtete Gesamtvolumen wiedergibt sowie die Zellfächen zwischen zwei Zellen identisch sind, ergibt sich eine konsistente und konservative Diskretisierung. Für ein Einzelvolumen wie in Abb. 4.1 werden nun alle Terme der allgemeinen Transportgleichung 4.3 diskretisiert.

4.2 Räumliche Diskretisierung

4.2.1 Interpolation an die Zellflächen

Werden bei der Diskretisierung Werte an den Zellflächen f benötigt, erhält man diese durch lineare Interpolation aus den der Zellfläche benachbarten Zellmittelpunktswerten:

$$\phi_f = \phi_P(1 - \lambda_f) + \phi_N \lambda_f. \tag{4.8}$$

Der Interpolationskoeffizient λ_f wird mit Hilfe der Ortsvektoren $\mathbf{x}$ berechnet:

$$\lambda_f = \frac{\mathbf{x}_c - \mathbf{x}_P}{|\mathbf{x}_N - \mathbf{x}_P|}. \tag{4.9}$$

Man spricht hier auch von einem Zentrale-Differenzen-Schema.
Bei stark nichtorthogonalen, verzerrten oder gescherten Volumen-Zellen stabilisiert eine sogenannte *Skewness-Correction*, welche zusätzlich den Gradienten der betrachteten Größe in die Interpolation miteinbezieht:

$$\phi_f = \frac{1}{2}(\phi_P + \phi_N) + \frac{1}{2}(\nabla\phi_P + \nabla\phi_P)\left[\mathbf{x}_f - \frac{1}{2}(\mathbf{x}_P + \mathbf{x}_N)\right]. \tag{4.10}$$

Bei der Diskretisierung des konvektiven Terms werden spezielle Interpolationen an die Zellflächen verwendet, s. Abschnitt 4.2.3.

4.2.2 Gradientenberechnung

Gradienten können entweder mit Hilfe des Satzes von Gauss oder mit Hilfe der Methode der kleinsten Quadrate ermittelt werden.
Bei der ersten Methode lautet die Approximation:

$$\nabla\phi \approx \frac{\sum_{iN} (\phi_f S_f)_{iN}}{\Delta V}. \tag{4.11}$$

Die auftretende Summe erstreckt sich dabei auf alle Teilfächen der Gesamtoberfläche des betrachteten Volumens.

Bei der zweiten Methode nimmt man eine lineare Variation der Lösungsvariable ϕ an und erhält als Diskretisierungsfehler e_N:

$$e_N = \phi_N - (\phi_P + \nabla\phi_P \mathbf{d}_{NP}) \,. \tag{4.12}$$

Wird nun der Ausdruck am betrachteten Zellmittelpunkt e_P minimiert, wobei $w_N = 1/|\mathbf{d}_{PN}|$ geometrische Gewichtungsfaktoren darstellen, erhält man für den Gradienten an einem Zellmittelpunkt P:

$$\nabla\phi_P = \sum_{iN} \left(w_N^2 \mathbf{G}^{-1} \mathbf{d}_{PN} (\phi_N - \phi_P) \right)_{iN} . \tag{4.13}$$

Dabei ist $\mathbf{G}$ eine symmetrische 3×3-Matrix, welche mit Hilfe von Abstandsvektoren und den Gewichtungsfaktoren bestimmt wird:

$$\mathbf{G} = \sum_{iN} \left(w_N^2 \mathbf{d}_{PN} \mathbf{d}_{PN} \right)_{iN} . \tag{4.14}$$

Der Vorteil der ersten Methode ist die Einfachheit der Implementierung und eine sehr schnelle und speicherschonende Berechnung der Gradienten. Bei stark verzerrten oder nichtorthogonalen Zellen ist aber eine Genauigkeit zweiter Ordnung bei der Diskretisierung nicht mehr gewährleistet.
Die zweite Methode liefert auch bei stark verzerrten oder nichtorthogonalen Zellen die geforderte Genauigkeit der Diskretisierung. Die Berechnung ist allerdings aufwändiger, sowohl die Geschwindigkeit als auch den Speicherbedarf betreffend. Dies liegt an der Bestimmung der Hilfsmatrizen $\mathbf{G}$. Zu Beginn jedes Zeitschritts müssen die Koeffizienten aus der aktuell vorliegenden Geometrie neu bestimmt und zusätzlich gespeichert werden.
Da sich die im Rahmen dieser Arbeit implementierte zweite Methode als sehr stabil erwiesen hat, wurde sie in allen vorgestellten Simulationen, soweit nicht anders vermerkt, verwendet.
Bei sehr stark verzerrten Netzen schlägt ZWART [112] eine Relaxation der Gradienten vor, welche ebenfalls implementiert wurde. Bei nicht ausreichender Konvergenz der Lösung können aber starke Abweichungen vom erwarteten Ergebnis auftreten, weswegen diese bei den vorgestellten Simulationen nicht verwendet wurde.

4.2.3 Diskretisierung des konvektiven Terms

Der konvektive Fluss F über eine repräsentative Zellfläche S_f wird mit Hilfe des Massenstroms $\dot{m}$ über diese folgendermaßen genähert:

$$F_f^c = \int_{S_f} (\nabla(\rho \mathbf{u} \phi))\, dS_f \approx \dot{m}_f \phi_f. \tag{4.15}$$

Im verwendeten iterativen Lösungsschema wird der Massenstrom während einer Iteration als konstant betrachtet, wodurch dann auch die Impulsgleichungen ein lineares Gleichungssystem bilden.

Es fehlt jedoch noch eine geeignete Bestimmung der Lösungsgröße ϕ an der Zellfläche. Die Schwierigkeit besteht darin, einen möglichst guten Kompromiss zwischen zu erreichender Genauigkeit und Stabilität des Lösungsverfahrens zu finden.
Die beiden gängigsten Verfahren zur Approximation werden kurz vorgestellt und dann ein Ausblick auf die sogenannten hochauflösenden Fluss-Begrenzungs- oder Flux-Limiter-Verfahren gegeben.

4.2.3.1 Stromauf-Interpolations-Verfahren

Bei der Stromauf-Interpolation (Upwind-Differencing-Scheme, UDS) wird der gesuchte Zellflächenwert ϕ_f in Abhängigkeit des vorliegenden Massenstroms $\dot{m}$ durch eben diese Zellfläche bestimmt:

$$\phi_f = \begin{cases} \phi_P, & \text{wenn } \dot{m}_f \geq 0, \\ \phi_N, & \text{wenn } \dot{m}_f < 0. \end{cases} \tag{4.16}$$

Dieses Verfahren ist uneingeschränkt stabil, besitzt jedoch nur eine Genauigkeit erster Ordnung. Die dem Verfahren inhärente numerische Diffusion führt zu einem Verschmieren der Lösung, s. PERIC [81].

4.2.3.2 Zentrales-Differenzen-Schema

Beim Zentralen-Differenzen-Schema (Central-Differencing-Scheme, CDS) wird der Zellflächenwert ϕ_f mit Hilfe von Gl. 4.9 linear an die betrachtete Zellfläche interpoliert:

$$\phi_f = \lambda_f \phi_N + (1 - \lambda_f)\phi_N. \tag{4.17}$$

Dieses Verfahren besitzt eine Genauigkeit zweiter Ordnung, die Stabilität des Verfahrens nimmt aber ab. Dies resultiert aus Oszillationen im Lösungsfeld, die während des Iterationsprozesses bei Verwendung dieses Schemas auftreten können.

4.2.3.3 Hochauflösende Verfahren

Die genaue Approximation des konvektiven Terms stellt ein fundamentales Problem dar. Das Erreichen einer möglichst hohen Genauigkeit und das Erreichen einer hohen numerischen Stabilität sind dabei konträre Optimierungsziele, s. PATANKAR [79].

Inzwischen werden sogenannte hochauflösende, limitierende Flux-Limiter-Verfahren erfolgreich angewendet, siehe SKODA [93], die auch im vorgestellten Verfahren hinterlegt sind. Diese Flux-Limiter-Verfahren stellen eine Kombination aus verschiedenen Basisverfahren dar. Im Rahmen dieser Arbeit wurde sehr erfolgreich das MINMOD-Schema nach HARTEN [49] verwendet. Weitere Schemata zur Diskretisierung des konvektiven Term sind im Verfahren hinterlegt und werden in FIEREDER [37] ausführlich besprochen.

Obwohl diese Flux-Limiter-Verfahren ihre Genauigkeit und Stabilität automatisch dem lokalen Strömungszustand anpassen, zeigte sich bei deren Anwendung in komplexen Problemstellungen immer noch ein instabiles numerisches Verhalten. Deswegen werden die hochauflösenden Interpolationsschemata nicht direkt implizit diskretisiert, wie ursprünglich vorgeschlagen, sondern mit einer *nachgezogenenen Korrektur* berücksichtigt. Diese Methode, erstmals von KHOSLA [62] vorgeschlagen, stabilisiert das Verfahren. Die Berechnung des Wertes ϕ_f an einer Zellfläche f erfolgt nun über die folgende Beziehung:

$$\phi_f^m = \left(\phi_f^{UDS}\right)^m + \alpha(\phi_f^{HOS} - \phi_f^{UDS})^{m-1}, \tag{4.18}$$

wobei Werte zum aktuellen Iterationsschritt mit m, Werte aus dem vorhergegangenen Iterationsschritt mit $m-1$ gekennzeichnet werden.
HOS bezeichnet dabei den mit einem "High-Order-Scheme", z. B. CDS oder MINMOD, diskretisierten Wert, UDS den mit dem Stromauf-Interpolations-Verfahren diskretisierten Wert.
Durch den Blendingfaktor α wird zusätzlich der Anteil des höherwertigen Schemas eingestellt. Bei $\alpha = 0.0$ erhielte man also das reine Stromauf-Interpolations-Schema, bei $\alpha = 1.0$ das reine höherwertige Interpolations-Schema. In komplexen Anwendungen zeigte sich die Notwendigkeit eines Wertes $\alpha < 1.0$, was die Genauigkeit des Verfahrens prinzipiell herabsetzt, aber stets einen guten Kompromiss zwischen Genauigkeit und Stabilität gewährleistete.
Die Konvergenzeigenschaften des iterativen Lösungsprozesses verschlechtern sich gegenüber der rein impliziten Betrachtung in der Regel nur wenig, da der explizit nachgeführte Anteil meistens klein gegenüber dem impliziten Anteil ist.

4.2.4 Diffusiver Term

Bei der Näherung des diffusiven Flusses F^d an einer Zellfläche S_f

$$F_f^d = \Gamma_f \left(\nabla\phi\right)_f \mathbf{n}_f S_f \tag{4.19}$$

wird der Gradient der Lösungsgröße ϕ an dieser Zellfläche benötigt.
Bei linearer Interpolation der Gradienten der Zellflächen-Nachbarn an die Zellfläche entstünde ein Ausdruck in Abhängigkeit nicht nur der unmittelbaren Nachbarwerte der betrachteten Fläche, sondern auch von Werten der jeweiligen übernächsten Zellen. Die Implementierung wird im allgemein polyedrischen Fall komplex und zu speicheraufwändig, weswegen MUZAFERIJA [76] eine Näherung des diffusiven Flusses in Richtung des normierten Verbindungsvektors $\mathbf{d}$ von der betrachteten Zellfläche zur benachbarten Zellen vorschlägt. Bei wenig verzerrten Netzen ist $\mathbf{n} \approx \mathbf{d}$, die Richtung

des Verbindungsvektors zweier Nachbarzellen entspricht also in etwa der Richtung des Normalenvektors der betrachteten Zellfläche.
Über eine zentrale Differenz erhält man dann den folgenden leicht implementierbaren, impliziten Ausdruck:

$$F_f^d = \Gamma_f \left(\nabla\phi\right)_f \mathbf{d} S_f = \Gamma_f S_f \frac{\phi_N - \phi_P}{|\mathbf{x}_N - \mathbf{x}_P|}. \tag{4.20}$$

Die Abweichung der Richtung des Normalenvektors $\mathbf{n}$ und des Verbindungsvektors $\mathbf{d}$ zieht man durch eine explizite Korrektur nach. Mit dem aktuellen Iterationsschritt m und dem vorangegangenen Iterationsschritt $m-1$ erhält nun:

$$F_f^d = \left(\Gamma_f S_f \left(\nabla\phi\right)_f \mathbf{d}\right)^m + \left(\Gamma_f S_f \left[\nabla\phi\mathbf{n} - \nabla\phi\mathbf{d}\right]\right)^{m-1}. \tag{4.21}$$

Kombiniert man nun diese beiden Ausdrücke, erhält man die einfach zu implementierende und stabile Form des diskreten diffusiven Flusses:

$$F_f^D = \Gamma_f S_f \frac{\phi_N - \phi_P}{|\mathbf{x}_N - \mathbf{x}_P|} + \Gamma_f S_f \left(\nabla\phi\right)_f \left(\mathbf{n} - \mathbf{d}\right). \tag{4.22}$$

Da dieser Korrekturterm die Nichtorthogonalität im Sinne einer Finite-Volumen-Diskretisierung behandelt, wird dieser auch als *Nichtorthogonale Korrektur* bezeichnet. Da der explizit nachgeführte Korrekturterm bei gewissenhaft erstellten Rechennetzen klein ist, verschlechtert sich das Konvergenzverhalten durch diese Behandlung kaum.
Diese Diskretisierung wurde auch von SKODA [93] im Vorgängerverfahren hinterlegt. ZWART [112] zeigt jedoch, dass eine bessere Approximation erreicht wird, wenn man einen Korrekturwert α in 4.22 einführt

$$F_f^D = \Gamma_f \alpha S_f \frac{\phi_N - \phi_P}{|\mathbf{x}_N - \mathbf{x}_P|} + \Gamma_f S_f \left(\nabla\phi\right)_f \left(\mathbf{n} - \alpha\mathbf{d}\right) \tag{4.23}$$

und diesen mit

$$\alpha = \mathbf{n}\mathbf{d} \tag{4.24}$$

bestimmt.
Durch diese Modifikation wird die Differenz $\mathbf{n} - \mathbf{d}$ auf $\mathbf{d}$ projiziert und somit vom ersten Term aus 4.22 mitbehandelt. Der Gradient aus der alten Iteration wird jetzt nur noch zur Bestimmung der nichtorthogonalen Korrektur verwendet.

Bei sehr stark verzerrten und nichtorthogonalen Netzen können die explizit nachgezogenen Korrekturen sehr groß werden und das Konvergenzverhalten verschlechtern oder sogar zur Divergenz führen. Der ursprüngliche Koeffizienten-Anteil wird deswegen unter Berücksichtigung der Nichtorthogonalität der betrachteten Zellfläche modifiziert:

$$\frac{\Gamma_f \alpha S_f}{|\mathbf{x}_N - \mathbf{x}_P|} \Rightarrow \frac{\Gamma_f S_f}{|\mathbf{x}_N - \mathbf{x}_P|} \left(\alpha + \frac{1-\alpha}{\lambda}\right). \tag{4.25}$$

Der Wert λ ist ein beliebiger, empirisch gefundener Faktor, welcher typischerweise um $\lambda = 0,8$ liegt. Ein entsprechender Term muss auf der rechten Seite der diskreten Gleichung berücksichtigt werden. Diese Modifikation greift nur in problematischen Regionen des Rechennetzes und ermöglicht auch die Lösung bei akzeptablem Konvergenzverhalten auf sehr schlechten Netzen.

4.2.5 Quellterme

4.2.5.1 Volumen-Quellterme

Ein Volumen-Quellterm Q_P in einer Zelle wird über das Volumenintegral über diese Zelle bestimmt. Hierfür setzt man die Existenz eines repräsentativen Mittelwerts q_P von q im Zellmittelpunkt voraus und erhält:

$$Q_P = \int_V q dV \approx q_P \Delta V_P. \tag{4.26}$$

Die Zellmittelpunktswerte sind unmittelbar bekannt. Die Approximation liefert eine Genauigkeit zweiter Ordnung.

4.2.5.2 Oberflächen-Quellterme

Zur Bestimmung von Oberflächen-Quelltermen müssen die geometrischen Flächenverläufe sowie die Verteilung des Flusses über die jeweilige betrachtete Fläche bekannt sein. Um stets ebene Teilflächen zu erhalten, wird eine beliebige polyedrische Fläche zunächst trianguliert. An jedem Teilflächenstück exisitiert ein repräsentativer Integrand des Flusses. Mit dem integralen Fluss $F_i = f_i S_i$ durch eine Zellfäche S erhält man dann für das Oberflächenintegral über ein Teilvolumen:

$$\int_S f dS = \sum_i F_i = \sum i \int_S f dS \approx \sum_i f_i S_i. \tag{4.27}$$

Die Approximation liefert eine Genauigkeit zweiter Ordnung.

4.3 Zeitliche Diskretisierung

Für eine genaue zeitliche Auflösung einer Simulation muss eine geeignete zeitliche Diskretisierung für jede Rechenzelle P gewählt werden. In den Fluidgleichungen tritt nur die erste zeitliche Ableitung nach der jeweiligen betrachteten Lösungsgröße auf, bei den in den Verschiebungen formulierten Impulsgleichungen der Strukturmechanik jedoch auch die zweite zeitliche Ableitung.
Für Fluid- und Strukturgleichungen wird dieselbe Genauigkeit der Diskretisierung angestrebt, man erhält diese mit den im folgenden vorgestellten Euler-Verfahren.

4.3.1 Rückwärts-Euler-Verfahren erster Ordnung

Beim impliziten Eulerverfahren erhält man eine zeitliche Diskretisierungsgenauigkeit erster Ordnung und ein theoretisch für beliebig große Zeitschritte anwendbares Verfahren:

$$\int_V \phi dV \approx \frac{\phi_P - \phi_P^o}{\Delta t} \Delta V. \tag{4.28}$$

Zusätzlich zur Lösungsgröße ϕ^n des aktuellen oder des neuen Zeitschritts müssen nun auch Werte des alten Zeitschritts o verwendet und abgespeichert werden.

4.3.2 Rückwärts-Euler-Verfahren zweiter Ordnung

Beim Rückwärts-Euler-Verfahren werden zusätzlich Werte aus dem vorletzten Zeitschritt *oo* in die Diskretisierung miteinbezogen, weswegen man auch vom Drei-Zeitebenen-Verfahren spricht und die Werte aus dem letzten und dem vorletzten Zeitschritt im Speicher gehalten werden müssen.

Für die erste zeitliche Ableitung lautet die Diskretisierung:

$$\left(\frac{\partial \phi}{\partial t}\right)^n = \frac{3\phi^n - 4\phi_P^o + \phi_P^{oo}}{2\delta t}. \tag{4.29}$$

Für die Impulsgleichungen der Struktur benötigt man eine konsistente Diskretisierung der zweiten zeitlichen Ableitung der Verschiebungen. Man geht von der Formulierung der ersten zeitlichen Ableitung der Strukturgeschwindigkeiten aus

$$\left(\frac{\partial \delta \mathbf{v}_P}{\partial t}\right)^n = \frac{3\delta \mathbf{v}_P^n - 4\delta \mathbf{v}_P^o + \delta \mathbf{v}_P^{oo}}{2\delta t} \tag{4.30}$$

und ersetzt die Geschwindigkeit des neuen Zeitschritts $\mathbf{v}^n$ durch einen Drei-Zeitebenen-Ausdruck mit den Verschiebungen:

$$\delta \mathbf{v}_P^n = \frac{3\delta \mathbf{u}_P^n - 4\delta \mathbf{u}_P^o + \delta \mathbf{u}_p^{oo}}{2\delta t} \tag{4.31}$$

und erhält schließlich die folgende Diskretisierung:

$$\left(\frac{\partial^2 \mathbf{u}_P}{\partial t^2}\right)^n = \left(\frac{\partial \delta \mathbf{v}_P}{\partial t}\right)^n = \frac{3\left(\delta \mathbf{u}_P^n - 4\delta \mathbf{u}_P^o\right)}{4\delta t^2} - \frac{4\delta \mathbf{v}_P^o - \delta \mathbf{v}_P^{oo}}{2\delta t}. \tag{4.32}$$

TUKOVIC [98] und die vorliegende Arbeit zeigen ein sehr gutes und wenig dissipatives Lösungsverhalten bei instationären Simulationen. Zusätzlich zu den Werten der Verschiebungen aus den letzten und vorletzten Zeitschritten müssen nun auch noch die Geschwindigkeiten aus diesen beiden Zeitschritten im Speicher gehalten werden.
Eine direkte Diskretisierung der zweiten zeitlichen Ableitung unter Verwendung der Werte aus der Zeitebene *ooo*

$$\left(\frac{\partial^2 \mathbf{u}_P}{\partial t^2}\right)^n = \left(\frac{\partial \delta \mathbf{v}_P}{\partial t}\right)^n = \frac{2\delta \mathbf{u}_P^n - 5\delta \mathbf{u}_P^o + 4\delta \mathbf{u}_P^{oo} - \delta \mathbf{u}_P^{ooo}}{2\delta t} \tag{4.33}$$

führt zu Instabilitäten im Lösungsverhalten und wurde deswegen nicht verwendet.

4.3.3 Raum-Erhaltungs-Gesetz

Das Raum-Erhaltungs-Gesetz berücksichtigt Netzbewegungen oder Netzverformungen des Fluidgebiets im Lösungsprozess. Netzbewegungen treten zum Beispiel bei Simulationen rotierender Laufräder oder verschiebbarer Kolben in Strömungsmaschinen auf. Netzverformungen ergeben sich ebenfalls bei der Simulation von Fluid-Struktur-Interaktionen auf. Bei der Bewegung oder Verformung von Netzen entstehen zusätzliche sog. Netz-Massenflüsse, durch die die konvektive Terme der Erhaltungsgleichungen modifiziert werden müssen.

Bei der Herleitung dieser Netz-Massenflüsse geht man von der Volumen-Kontinuitätsgleichung aus

$$\frac{dV}{dt}\int_V dV - \int_S \mathbf{v}_g \mathbf{n} dS = 0, \tag{4.34}$$

aus, welche die Raumerhaltung für eine bewegte oder verformte Volumenzelle mit dem Volumen V beschreibt. Der zweite Term auf der linken Seite stellt den Netz-Massenfluss dar, welcher durch die Netzgeometrie und die Netzgeschwindigkeit $\mathbf{v}_g$ bestimmt wird. Durch die Diskretisierung erhält man

$$\frac{V^n - V^o}{\Delta t} = \sum_i \left(\mathbf{v}_g \mathbf{n}\right)_i S_i, \tag{4.35}$$

wobei für die zeitliche Ableitung hier beispielhaft das bereits vorgestellte Rückwärts-Euler-Verfahren verwendet wurde. Die Anwendung anderer zeitlicher Diskretisierungen erfolgt analog.

Die zeitliche Ableitung wird nun mit dem durch die Zellbewegung oder Zellverformung überstrichenen Volumen δV ausgedrückt:

$$\frac{V^n - V^o}{\Delta t} = \frac{\sum_i \partial V_i}{\Delta t}, \tag{4.36}$$

bzw. für ein Zellfläche f:

$$\frac{\partial V_f}{\Delta t} = \dot{V}_f. \tag{4.37}$$

Zur Berechnung des überstrichenen Zellvolumens wird die jeweilig betrachtete Zellfläche zunächst trianguliert. Durch die Bewegung oder Verformung wird durch ein betrachtetes Dreieck dieser Zellfläche ein Prisma überstrichen, dessen Zellvolumen über Tetraedrisierung berechnet wird, s. Abb. 4.2.

Hierbei ist zu beachten, dass die Seitenflächen des jeweilig betrachteten Prismas Vierecke sind, welche verwunden sein können. Der entstehende Fehler wird durch zweimalige Tetraedrisierung mit unterschiedlicher Ausrichtung der Tetraeder und anschließender Mittelung der unterschiedlich berechneten überstrichenen Volumina vermieden.

Die Netzbewegung oder Netzverformung beeinflusst nur die Massenflüsse $\dot{m}_f$, welche nun um die Netz-Massenflüsse $\dot{m}_{f,g}$ vermindert werden müssen:

$$\dot{m}_{f,res} = \int_{S,f} \rho \left(\mathbf{v} - \mathbf{v}_g\right) \mathbf{n} dS \approx \left(\rho \mathbf{v} \mathbf{n} S - \rho V\right)_f = \dot{m}_f - \dot{m}_{f,g}. \tag{4.38}$$

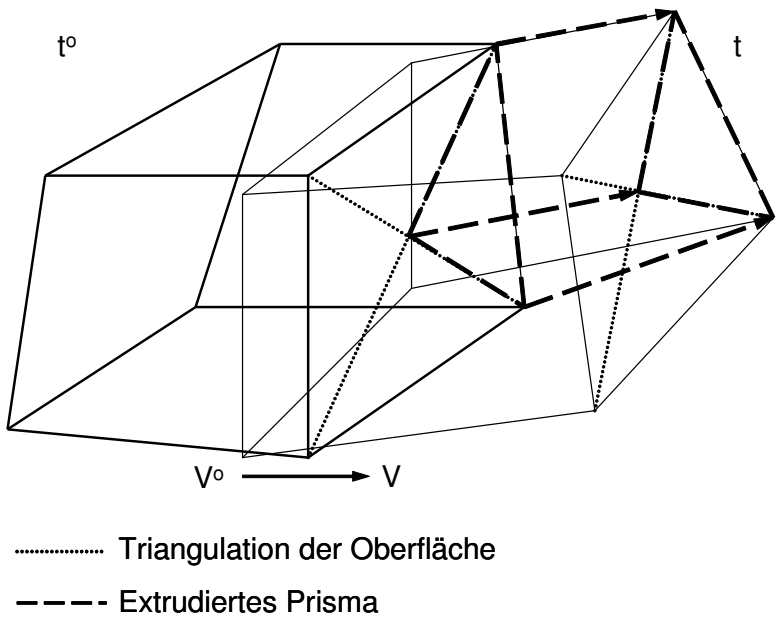

Abbildung 4.2: Raum-Erhaltungs-Gesetz

Diese Netz-Massenflüsse müssen deswegen jedes Mal neu berechnet werden, wenn das Netz eine Bewegung oder Verformung erfährt, in der Regel also zu Beginn jedes neuen Zeitschritts. Mit deren Hilfe werden dann die resultierenden Massenströme $\dot{m}_{f,res}$ ermittelt.

SKODA [93] implementierte das Raum-Erhaltungs-Gesetz noch in einer für Starrkörperrotationen spezialisierten Form. Die neue polyedrische Datenstruktur und die beliebigen Netzdeformationen, wie sie bei FSI-Simulationen auftreten können, machten obige Implementierung, siehe FIEREDER [37], notwendig.

4.4 Randbedingungen

Eine genaue Festlegung, Diskretisierung und Implementierung der Randbedingungen ist Grundlage für ein stabiles und schnelles Konvergenzverhalten. Prinzipiell unterscheidet man Dirichlet- und Neumann-Randbedingungen, die für das Fluid- und das Strukturgebiet an unterschiedlichen Rändern angewandt werden.

4.4.1 Dirichlet-Randbedingung

Bei einer Dirichlet-Randbedingung wird der Wert einer Größe ϕ am Rand vorgegeben. Für Gleichungen, die am randnächsten Mittelpunkt M diskretisiert wurden, muss der Randwert ϕ_R zusammen mit dem korrespondierenden Koeffizienten der diskreten Gleichung auf die rechte Seite $\mathbf{b}$ aufgeschlagen werden:

$$\mathbf{b}_M := \mathbf{b}_M - A_{MR}\phi_R. \tag{4.39}$$

4.4.2 Neumann-Randbedingung

Bei einer Neumann-Randbedingung wird der Gradient der Größe ϕ vorgegeben.
Für Gleichungen, die am randnächsten Mittelpunkt M diskretisiert wurden und für deren Diskretisierung der Gradient am Rand R benötigt wird, muss der zum Gradienten $g_R = (\nabla\phi_R)\mathbf{n}$ korrespondierende Fluss direkt auf die rechte Seite $\mathbf{b}$ hinzuaddiert werden.
Dies geschieht beispielsweise beim diffusiven Term der allgemeinen Transportgleichung:

$$\begin{aligned}\int_V \nabla\nabla\phi dV = \int_S \nabla\phi\mathbf{n}dS = \sum_i \left(\int_S \nabla\phi\mathbf{n}dS\right)_i \\ = \sum_{i\setminus R} \left(\int_S \nabla\phi\mathbf{n}dS\right)_{i\setminus R} + g_R|\mathbf{S}|.\end{aligned} \tag{4.40}$$

Der Vektor der rechten Seite wird dann modifiziert nach der Beziehung:

$$\mathbf{b}_M := \mathbf{b}_M - g_R\,|\mathbf{S}|\,. \tag{4.41}$$

Die Größe ϕ_R am Rand wird dann über Extrapolation vom Zellmittelpunktswert bestimmt:

$$\phi_R = \phi_M + g_R|\mathbf{d}_{MR}|, \tag{4.42}$$

wobei $\mathbf{d}_{MR}$ den Verbindungsvektor vom Zellmittelpunkt zum Punkt der Randfläche bezeichnet.

Die speziellen Ausprägungen dieser Randbedingungstypen und Details der Implementierung werden sowohl für die fluidmechanischen als auch die strukturmechanischen Gleichungen im Anhang C beschrieben.

4.5 Gleichungslöser

Die effiziente Lösung der durch die Diskretisierung gewonnenen linearen Gleichungssysteme stellt eine zentrale Problemstellung bei der Entwicklung des neuen Lösungsverfahrens dar. Im Rahmen dieser Arbeit wurde eine mit MPI [99] parallelisierte Bibliothek zur Lösung linearer Gleichungssysteme implementiert, welche verschiedene Varianten von iterativen Lösungsverfahren enthält, deren Funktionsweise in den folgenden Abschnitten näher beschrieben wird.

Nach der vollständigen Diskretisierung und der Einarbeitung der Randbedingungen erhält man für jede Erhaltungsgleichung ein lineares Gleichungssystem:

$$\mathbf{A}\boldsymbol{\phi} = \mathbf{b}. \tag{4.43}$$

Die $N \times N$ Koeffizientenmatrix $\mathbf{A}$ weißt dabei eine symmetrische oder unsymmetrische, dünnbesetzte Struktur auf, wobei N der Anzahl der verwendeten Finite-Volumen Zellen entspricht. Jede Zeile der Koeffizientenmatrix enthält dabei ein Diagonalelement und so viele Nebendiagonalelemente, wie die korrespondierende Finite-Volumen Zelle unmittelbare Nachbarn besitzt. $\boldsymbol{\phi}$ stellt den Lösungsvektor und $\mathbf{b}$ die rechte Seite des Gleichungssystems dar.

Zur Lösung eignen sich iterative Lösungstechniken aus den folgenden Gründen:

- Bei den später vorgestellten iterativen Lösungsverfahren müssen nur wenige Zusatzinformationen im Lösungsprozess gespeichert werden und die Verfahren bleiben somit speicherschonend.
- Sowohl die Koeffizientenmatrix als auch die rechte Seite der zu lösenden Gleichungssysteme stellen innerhalb der äußeren Iterationsschleife des sequentiellen Lösungsverfahrens nur momentane Näherungen dar und müssen zu Beginn jedes äußeren Iterationsschritts aktualisiert werden. Deswegen müssen die Gleichungssysteme innerhalb eines äußeren Iterationsschrittes nur näherungsweise mit sog. inneren Iterationsschritte gelöst werden.

Die entkoppelt diskretisierten Gleichungssysteme werden iterativ bis zu einem vernünftig gewählten Konvergenzkriterium gelöst. Konvergenz ist dann erreicht, wenn die Norm des Fehlers $\mathbf{e}$ der Lösung gegen Null geht. Da der Fehler $\mathbf{e}$ der vorliegenden Lösung nur mit Hilfe der momentanen Lösung ϕ^* und der exakten Lösung ϕ bestimmt werden kann

$$\mathbf{e} = \boldsymbol{\phi}^* - \boldsymbol{\phi}, \tag{4.44}$$

die exakte Lösung jedoch unbekannt ist, wird das Residuum $\mathbf{r}$ oft aus pragmatischen Gründen als Hilfsmaß zur Bewertung des Konvergenzstatus verwendet:

$$\mathbf{r} = \mathbf{b} - \mathbf{A}\boldsymbol{\phi}^*. \tag{4.45}$$

Im Lösungsprozess wird versucht, eine geeignete Norm des Residuums um einige Größenordnungen zu reduzieren. Dazu wird die Betragssummenorm $||\mathbf{r}||_1$ zur Bewertung des Lösungsfortschritts der inneren Iterationsschritte verwendet. In jeder Iteration

wird diese neu bestimmt und mit dem Wert der Betragssummennorm des ersten inneren Iterationsschritts entdimensioniert.

Die Bewertung des Lösungsfortschrittes der äußeren Iterationsschleife erfolgt mit Hilfe einer Matrix-basierten Entdimensionierung nach JASAK [59], welche zwar einen relativ hohen Berechnungsaufwand verlangt, gleichzeitig aber für unterschiedlichste Problemstellungen zuverlässig anwendbar ist.
Hierzu wird zunächst der arithmetische Mittelwert $\overline{\phi}$ des betrachteten Lösungsvektors $\boldsymbol{\phi}$ und damit ein Hilfsvektor $\boldsymbol{\phi}^* = \boldsymbol{\phi} - \overline{\boldsymbol{\phi}}\mathbf{I}$ gebildet. Mit diesem Ausdruck und dem Residuenvektor $\mathbf{r}$ wird dann der entdimensionerte Residuenvektor $\mathbf{r}_N$ bestimmt:

$$\mathbf{r}_N = \frac{\mathbf{r}}{|\mathbf{A}\boldsymbol{\phi}^*| - |\mathbf{A}\boldsymbol{\phi}|}. \tag{4.46}$$

Beim Erreichen eines Residuenabfalls von vier Größenordnungen während der äußeren Iterationsschleife des sequentiellen Lösungsverfahrens 4.6 wird im Rahmen der in dieser Arbeit vorgestellten Simulationen von Konvergenz gesprochen. Für die inneren Iterationen reicht in der praktischen Anwendung ein Residuenabfall von einer Größenordnung, da bei den untersuchten Gleichungssystemen sowohl die Koeffizientenmatrizen als auch die Vektoren der rechten Seite nur momentane Näherungen darstellen.

Für die verschiedenen auftretenden Gleichungstypen eignen sich unterschiedliche iterative Lösungsverfahren, welche in den folgenden Abschnitt kurz vorgestellt werden.

4.5.1 Unvollständige LU-Zerlegung (ILU)

Die unvollständige LU-Zerlegung (*I*ncomplete *L*ower *U*pper decomposition, ILU) liefert die näherungsweise Zerlegung einer Matrix $\mathbf{A}$ in eine linke, untere (*L*ower) Dreiecksmatrix und eine rechte, obere (*U*pper) Dreiecksmatrix:

$$\mathbf{A} \approx \mathbf{LU}. \tag{4.47}$$

Für die Zerlegungsmatrizen werden nur Elemente für eine vorgegebene Besetzungsstruktur ermittelt. Mit Hilfe dieser Zerlegung können Gleichungssysteme $\mathbf{A}\boldsymbol{\phi} = \mathbf{b}$ durch wiederholte *Vorwärtssubstitutions*-Schritte:

$$\boldsymbol{\psi} = \mathbf{L}^{-1}\mathbf{b}, \tag{4.48}$$

und *Rückwärtssubstitutions*-Schritte:

$$\mathbf{x} = \mathbf{U}^{-1}\boldsymbol{\psi}, \tag{4.49}$$

effizient vorkonditioniert oder gelöst werden.
Erstmals erwähnt wurde die ILU-Methode in VARGA [102]. Im Rahmen dieser Arbeit wurde eine sehr einfache Variante der sog. ILU(0)-Zerlegung verwendet. Die 0 bezieht sich auf den Level der Besetzungsstruktur, s. MEISTER [72]. Bei der verwendeten Variante werden nur die Diagonalelemente der ursprünglichen Koeffizientenmatrix $\mathbf{A}$ modifiziert. Diese Variante wurde aus den folgenden Gründen gewählt:

- Für die Modifikation der Diagonalelemente muss nur wenig zusätzlicher Speicher allokiert werden.
- Die Variante kann effizient parallelisiert werden, s. Abs. 4.5.4.
- Sie kann als Präkonditionierer für die später beschriebenen Gradientenverfahren verwendet werden.
- Als Teil des implementierten Mehrgitterverfahrens, s. Abs. 4.5.3 wird sie als Relaxationsverfahren verwendet.
- Sie kann auch auf Blockgleichungssysteme angewendet werden, wie sie bei der gekoppelten Diskretisierung von Gleichungssystemen auftreten, s. ZWART [112].

Die implementierte Variante der ILU(0) wird im sequentiellen Lösungsverfahren zur iterativen Lösung aller auftretenden Konvektions-Diffusions-Gleichungen effizient angewendet wie sie bei der Diskretisierung der allgemeinen Transportgleichung entstehen. Sie löst den SIP-Algorithmus nach STONE [95] ab, eine für strukturierte Verfahren spezialisierte ILU-Zerlegung, der im Vorgängerverfahren von SKODA [93] verwendet wurde.

4.5.2 Konjugierte Gradientenverfahren (CG)

Konjugierte Gradientenverfahren (*C*onjugate-*G*radient-Verfahren), welche erstmals von HESTENES [52] vorgestellt wurden, sind Abstiegsverfahren zur Minimierung quadratischer Funktionen.
Die Grundidee der Konjugierten Gradientenverfahren besteht darin, die quadratische Form des ursprünglichen Gleichungssystems $\mathbf{Ax} = \mathbf{b}$ zu verwenden: $\mathbf{F}$ angeschrieben:

$$\mathbf{F} = \frac{1}{2}\mathbf{x}^T\mathbf{Ax} - \mathbf{x}^T\mathbf{b}, \tag{4.50}$$

welche als einziges Minimum die Lösung des ursprünglichen Gleichungssystems besitzt. $\mathbf{F}$ wird beim CG-Verfahren nicht in die Richtung des Residuums $\mathbf{r} = -\nabla\mathbf{F}$ minimiert, sondern mit Hilfe von Suchrichtungsvektoren $\mathbf{d}_i$, die zur Koeffizientenmatrix $\mathbf{A}$ konjugiert sind:

$$\mathbf{d}_i^T\mathbf{Ad}_{i+1} = \mathbf{0}. \tag{4.51}$$

Durch diese Bestimmung der Minimierungsrichtung gelangt das konjugierte Gradientenverfahren theoretisch in N Schritten zur Lösung des Gleichungssystems mit der $N \times N$ Koeffizientenmatrix $\mathbf{A}$.

Das konjugierte Gradientenverfahren in seiner Grundform ist nur auf Gleichungssysteme mit symmetrischer und positiv definiter Koeffizientenmatrix $\mathbf{A}$ anwendbar. Symmetrische und positiv definite Matrizen treten bei der Diskretisierung der Druckkorrekturgleichungen sowie der Diskretisierung der Struktur-Impulsgleichungen auf. Mit entsprechender Präkonditionierung, beispielsweise mit der vorgeschlagenen ILU(0)-Methode oder *der unvollständigen Cholesky-Zerlegung* (*I*ncomplete *C*holesky *C*onjugate *G*radient, ICCG-Verfahren, s. MEIJERINK [71]) können derartige Gleichungssysteme sehr effizient gelöst werden.

Zur effizienten Lösung von unsymmetrischen Gleichungssystemen sei noch das Bi-CG-Verfahren (*Bi-C*onjugate-*G*radient) zu erwähnen, welches bereits von LANCZOS [65] vorgeschlagen wurde und seine Verbreitung durch die Formulierung von FLETCHER [38] fand. Dieses Verfahren weist drei entscheidende Nachteile auf, s. MEISTER [72]: Zunächst kann es zu einem Verfahrensabbruch kommen, obwohl die Lösung des Gleichungssystems noch nicht ermittelt wurde. Dann werden aufwändige Matrix-Vektor-Multiplikationen mit $\mathbf{A}$ und $\mathbf{A}^T$ benötigt. Schließlich können sich Oszillationen im Konvergenzverhalten zeigen. VAN DER VORST [103] hat deswegen die BiCGSTAB-Methode (BiCG *Stab*ilized) vorgestellt, welche ein wesentlich glatteres Konvergenzverhalten aufweist.
Das BiCGSTAB(l)-Verfahren stellt eine Verallgemeinerung der BiCGSTAB-Methode dar und verwendet anstelle einer eindimensionalen Minimierung eine l-dimensionale Minimierung. Es wurde von SLEIJPEN [94] hergeleitet und wird im Lösungsverfahren sehr effizient und robust zur Lösung der Druckkkorrekturgleichung, Abs. 4.6.1.1, angewendet.

4.5.3 Mehrgitterverfahren (MG)

Mehrgitterverfahren zählen zu den effizientesten iterativen Lösungsverfahren für lineare Gleichungssysteme, wobei im Optimalfall der Lösungsaufwand proportional zur Anzahl der Unbekannten ist. Der Grundgedanke dieser Verfahren besteht darin, Fehler mit Hilfe einer Hierarchie von Leveln unterschiedlicher Auflösung der Problemstellung iterativ zu lösen. Eine kompakte Einführung in die Mehrgitterverfahren findet sich z.B. in BRIGGS [26]. Viele einfache iterative Lösungsverfahren wie das bekannte Gauss-Seidel- oder das Jakobi-Verfahren, s. MEISTER [72], besitzen die Eigenschaft, kurzwellige oder stark oszillatorische Fehler besser als langwellige Fehlerkomponenten glätten zu können.

Bei *geometrischen Mehrgitterverfahren*, s. HACKBUSCH [46], [47] u. [48], wird deswegen eine Hierarchie von geometrisch immer gröber werdenden Gittern erstellt. Die mögliche Anzahl von Ebenen ist dabei durch die Größe des ursprünglichen Problems bestimmt. Langwellige Fehlerkomponenten auf einem Feingitter erscheinen auf einem Grobgitter kurzwellig und können dort mit einfachen Glättern effizient gedämpft werden.
Durch die geeignete Definition von Operatoren zur *Restriktion* der Feingitterlösung auf das Grobgitter sowie zur *Prolongation* von Korrekturwerten vom Grobgitter auf das Feingitter sowie durch die Wahl eines geeigneten *Glätters* und einer sinnvollen Netzhierarchie kann dann mit iterativ ausgeführten *Zyklen* optimal schnell die Lösung erreicht werden.

Es ist in vielen Fällen komplizierter Geometrien und komplexer unstrukturierter Vernetzungen schwierig und oftmals unmöglich eine wirkliche Hierarchie von geometrisch repräsentierten Netzen zu erstellen.
Algebraische Mehrgitterverfahren, s. BRANDT [24], BRANDT [25] und RUGE [87], verwenden keine geometrische Hierarchie von Rechennetzen, es wird eine Hierarchie von immer kleiner werdenden Gleichungssystemen mit Hilfe des Gleichungssystems der

ursprünglichen Problemstellung gebildet. Wiederum müssen geeignete Operatoren zur Restriktion, Prolongation und Glättung sowie ein geeigneter Zyklus zur Lösung der Problemstellung definiert werden. Mit der geeigneten Wahl von Restriktions- und Prolongationsoperatoren ergeben sich die Gleichungssysteme aller Level dann automatisch.

Im Rahmen dieser Arbeit kamen sowohl ein algebraisches Mehrgitterverfahren, bei welchem die Vergröberung auf Finite-Volumen-Aggregationen basiert, sowie ein geometrisches Mehrgitterverfahren, welches lediglich die Vergröberungsstrategie unstrukturiert, algbraisch durchführt, zum Einsatz. Diese Verfahren werden in den nächsten Abschnitten kurz beschrieben.

4.5.3.1 Algebraisches Mehrgitterverfahren (AMG)

Vergröberungsstrategie

Die algebraische Vergröberungsstrategie basiert rein auf den Matrixeigenschaften des ursprünglichen Problemstellung und folgt einem sog. Aggregations- bzw. Agglomerationsansatz, s. VANEK [101], bei welchem Finite-Volumen-Zellen zu Blöcken aggregiert werden. Diese Blöcke bilden die Zellen des gröberen Levels. Bei der implementierten Variante nach RAW [83] werden die minimalen und maximalen Blockgrößen durch empirische gefundene Werte festgelegt.

Während der Vergröberung kann festgestellt werden:

- Die Vergröberungsrichtung kann wechseln.
- Ein ursprünglich strukturiert vernetztes Rechengebiet kann auf dem nächsten, gröberen Level unstrukturiert sein.
- Eine unstrukturierte Datenbasis ist bei Verwendung dieses AMG deswegen zwingend erforderlich und macht eine Implementierung in das strukturierteVorgängerverfahren von SKODA [93] unmöglich.

Weitere bezeichnende Eigenschaften für das Verfahren von RAW [83] sind:

- Es ist speziell für die Finite-Volumen-Diskretisierung entworfen worden und behält den konservativen Charakter der Gleichungen über alle Levels bei.
- Es ist einfach parallelisierbar, s. 4.5.4.
- Es ist speicherschonend.
- Es kann auch für Blockgleichungssysteme angewendet werden, wie sie z. B. bei der gekoppelten Diskretisierung von Gleichungssystemen auftreten können, s. ZWART [112].

Der Vergröberungsalgorithmus lässt sich in die folgenden Schritte zusammenfassen:

1. Finde zunächst für jede Rechenzelle den betragsmäßig maximalen Nachbarkoeffizienten A_{ij}^{max}.

2. Gehe über alle Gleichungen, deren Finite-Volumen-Zelle noch nicht einem Block zugewiesen wurde und beginne mit einem neuen Block für die nächste nicht zugewiesene Zelle.

3. Bestimme für den aktuellen Block die Gleichungen, welche stark mit diesem in Verbindungen stehen. Eine starke Verbindung liefert die empirische gefundene Beziehung $A_{ij} > \theta A_{ij}^{max}$, wobei im Rahmen dieser Arbeit sehr erfolgreich mit dem Wert $\theta = 0,25$ gearbeitet wurde. Die Zelle, die mit dieser Gleichung korrespondiert, wird zum aktuellen Block hinzugefügt, falls dessen Größe noch nicht eine festgelegte Maximalgröße überschreitet.

4. Mit einem zusätzlichen Aggregationsschritt werden die gebildeten Blöcke so rund wie möglich gemacht. Hierbei werden Zellen zum aktuellen Block hinzugefügt, falls diese zwei oder mehr starke Verbindungen zu diesem besitzen. Auch dieser Schritt wird bis zum Erreichen einer festgelegten Maximal-Blockgröße durchgeführt.

5. Falls die Größe des aktuellen Blocks kleiner als eine festgelegte Minimalgröße ist, gehe zu Schritt 2 bis zumindest diese Minimalgröße erreicht ist.

Als Beispiel für den Vergröberungsalgorithmus werden in Abb. 4.3 die Rechennetze, welche ein Quadrat kartesisch und polyedrisch diskret abbilden, auf dem feinsten und den folgenden gröberen Leveln dargestellt. Zu lösen sei auf dem feinsten Rechennetz eine reine Diffusionsgleichung.

Bei entsprechender Nummerierung der ursprünglichen Netzzellen wird durch den algebraischen Vergröbergungsalgorithmus eine geometrische Vergröberung erreicht. Die minimale Blockgröße wurde auf vier, die maximale Blockgröße auf acht Rechenzellen festgelegt. Der implementierte algebraische Vergröberungsalgorithmus zeigt bei allgemeiner Nummerierung der Netzzellen eine deutlich unregelmäßigere Struktur der gebildeten Rechenzellen als bei quasi-geometrischer Vergröberung. Die minimale Blockgröße kann nicht immer eingehalten werden. Die Auswirkung dieser unregelmäßigen Rechenzellen auf das Konvergenzverhalten des Lösungsverfahrens wird in Kapitel 6.2.1.7 diskutiert.

Restriktion, Prolongation, Zyklus und Relaxation

Nach der Festlegung der Vergröberungsstrategie, bleibt noch zu klären, wie die Grobgitter-Gleichungen aufgestellt werden können, wie die Lösung von einem feinen zu einem groben Netz durch die sog. *Restriktion* und wie umgekehrt die Lösung von einem groben zu einem feinen Netz durch die sog. *Prolongation* übertragen werden kann. Zusätzlich fehlt noch die Festlegung eines Glättungs- oder Relaxationsverfahrens auf den einzelnen Netzebenen.
Im Folgenden werden Größen und Indizes, die sich auf die Feingitter-Zellen beziehen klein, Größen und Indizes, die sich auf Grobgitter-Zellen beziehen, groß geschrieben.

Die Gleichung einer Feingitter-Zelle i lautet:

$$\sum_{j \in N_i} a_{ij} \boldsymbol{\phi}_j = b_i, \tag{4.52}$$

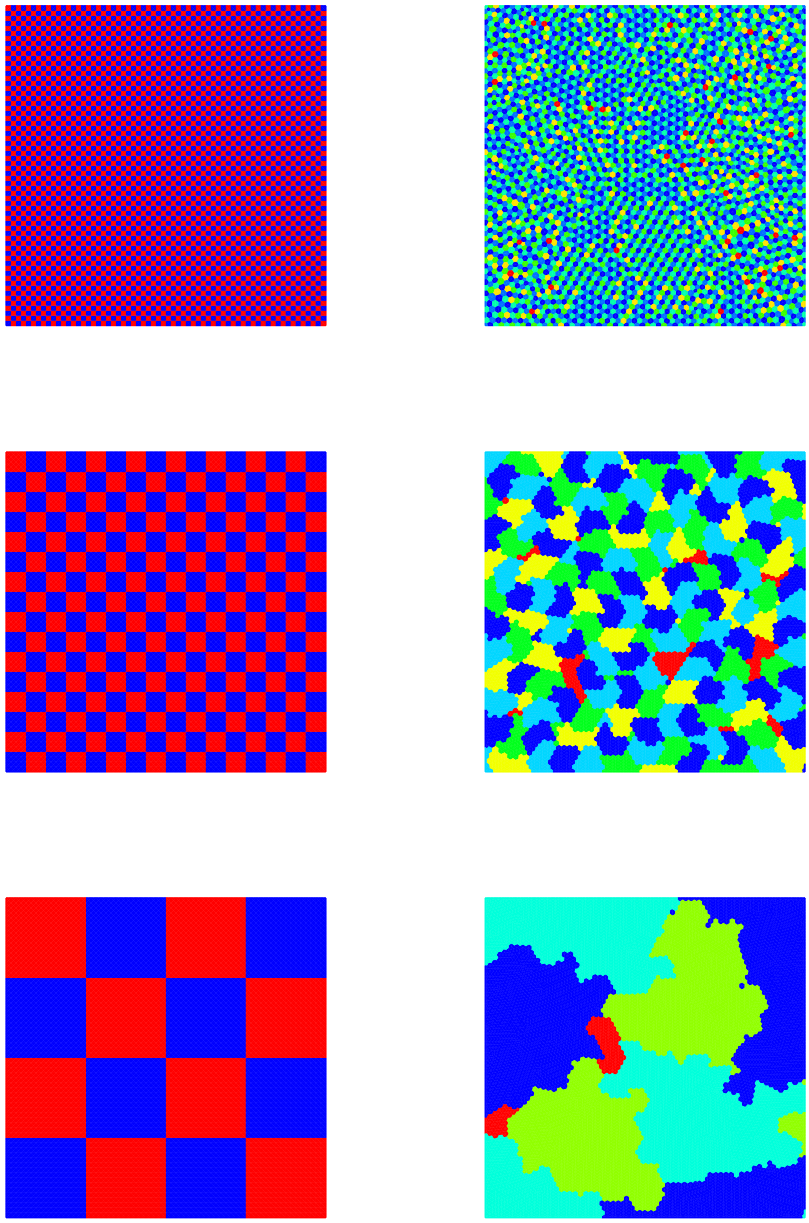

Abbildung 4.3: Vergröberungsalgorithmus für verschiedene Vernetzungstopologien, links für ein kartesisches, rechts für ein polyedrisches Rechennetz.

wobei sich die Summe über j auf alle Einträge N_i der betreffenden Zeile i der betrachteten Koeffizientenmatrix bezieht.
Die Lösung innerhalb einer Feingitterzelle ϕ_i wird mit Hilfe des Wertes des Grobgitterblockes $\Phi_{I/i}$, welcher die Feingitterzelle i enthält, korrigiert:

$$\tilde{\phi}_i = \phi_i + \Phi_{I/i} \tag{4.53}$$

Hiermit ist der Prolongationsoperator festgelegt.
Es wird nun gefordert, dass die korrigierte Summe der Feingitter-Residuen i innerhalb

eines Grobgitterblocks I verschwindet:

$$\sum_{i \in I} \tilde{r_i} = 0. \tag{4.54}$$

$\tilde{r_i}$ folgt mit Gl.4.52:

$$\tilde{r_i} = b_i - \sum_{j \in N_i} a_{ij} \tilde{\phi}_{ij}. \tag{4.55}$$

Mit dieser Beziehung und Gl. 4.53 folgt:

$$\tilde{r_i} = r_i - \sum_{j \in N_i} a_{ij} \phi_{J/j}. \tag{4.56}$$

Mit Hilfe von Gl.4.54 und einigen Umformungen folgt schließlich eine Beziehung für die Grobgitter-Gleichung:

$$\sum_{i \in I} \sum_{j \in N_i} a_{ij} \boldsymbol{\phi}_{J/j} = \sum_{i \in I} r_i, \tag{4.57}$$

wobei sich die erste und die letzte Summe auf alle Feingitterzellen i innerhalb der Grobitterzelle I erstrecken, die zweite Summe auf alle Einträge j der jeweils betrachteten Zeile i der Koeffizientenmatrix des Feingitter-Gleichungssystems.
Diese Beziehung wird in Analogie zu den Gleichungen des Feingitters umgeschrieben:

$$\sum_{J \in N_I} A_{IJ} \boldsymbol{\phi}_J = B_I. \tag{4.58}$$

Die Koeffizienten der Grobgitter-Gleichungen erhält man mit Gl.4.57 durch einfache Summation von Feingitter-Koeffizienten, s. Abb. 4.4. Die rechte Seite enthält die

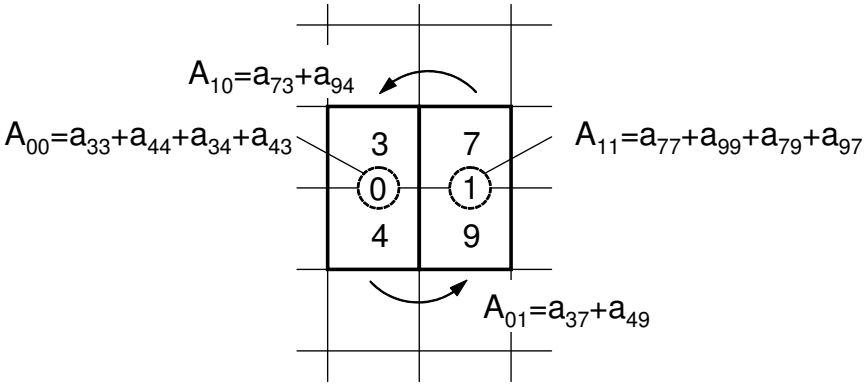

Abbildung 4.4: Beispiel zur Bestimmung der Grobgitter-Koeffizienten bei einem zweidimensionalen Feingitter mit zwei Grobgitterzellen.

Summe aller Feingitterresiduen innerhalb eines Grobitterblocks und legt den Restriktionsoperator fest. Die Gleichungssysteme aller Level können nun bestimmt werden.

Zuletzt bleibt noch die Frage zu klären, nach welchem Schema sich Feingitter- und Grobgitter-Lösungen abwechseln. Hierfür stehen verschiedene Arten von Zyklen zur Verfügung wie der *V-Zyklus*, der *W-Zyklus* und der *FMG-Zyklus*, s. Abb. 4.5. FMG steht hier für Full-Multigrid. Alle drei Typen wurden implementiert und können verwendet werden.

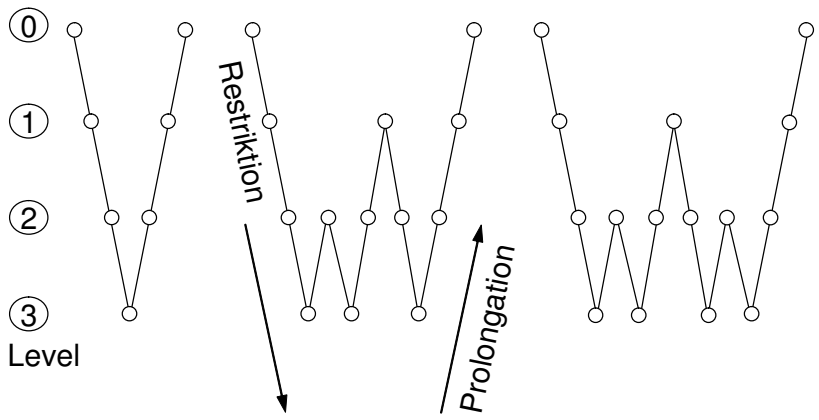

Abbildung 4.5: Verschiedene Zyklus-Typen bei Verwendung eines Mehrgitterverfahrens

Auf allen Ebenen des verwendeten Zyklus werden einige Iterationsschritte zur *Glättung* der Lösung der gebildeten Gleichungssysteme durchgeführt. Dazu wurde die schon vorgestellte ILU(0) verwendet, s. Abschnitt 4.5.1.
Durch zahlreiche Testläufe konnte ermittelt werden, dass ein Abfall der Residuennorm während eines Restriktions- oder Prolongationsschrittes von einer Größenordnung für ein sehr schnelles Konvergenzverhalten für viele Problemstellungen ausreichend ist. Allgemeine Regeln oder Aussagen zur Anzahl der zu verwendenden Iterationen können hier jedoch nicht getroffen werden, denn die Anzahl der zu verwendenden Glättungsschritte während der Restriktions- oder Prolongationsphase kann je nach Problemstellung stark variieren.

Mit den vorgangegangenen Abschnitten sind alle benötigten Bausteine für ein einfaches Aggregations-basiertes algebraisches Mehrgitterverfahren definiert worden. Es wurde effizient zur Lösung der auftretenden Druckkorrekturgleichung, s. Abs. 4.6.1.1 bei großen Problemstellungen angewendet.

4.5.3.2 Full Approximation Scheme (FAS)

Das Full Approximation Scheme (FAS) ist eine Mehrgitterstrategie zur Lösung von nichtlinearen Gleichungen oder gekoppelten Gleichungssystemen, s. BRANDT [23] und BRIGGS [26]. Im Falle linearer Gleichungen reduziert es sich auf ein lineares Mehrgitterverfahren. Zusätzlich zu den Operatoren zur Restriktion, Prolongation und Glättung muss noch ein geeignetes Schema zur nichtlinearen Glättung definiert werden.

PERIC [81] schlägt beispielsweise ein geometrisches FAS zur effizienten Lösung des hydrodynamischen Grundsystems vor, die nichtlineare Glättung erfolgt hier durch die äußere Iterationsschleife des SIMPLE-Algorithmus, s. Abs. 4.6.1. Ein ähnliches Verfahren wird in dieser Arbeit zur Lösung der Struktur-Impulsgleichungen verwendet und im Folgenden kurz beschrieben.

Feingitterwerte werden im Folgenden wiederum klein geschrieben, Grobgitterwerte groß.

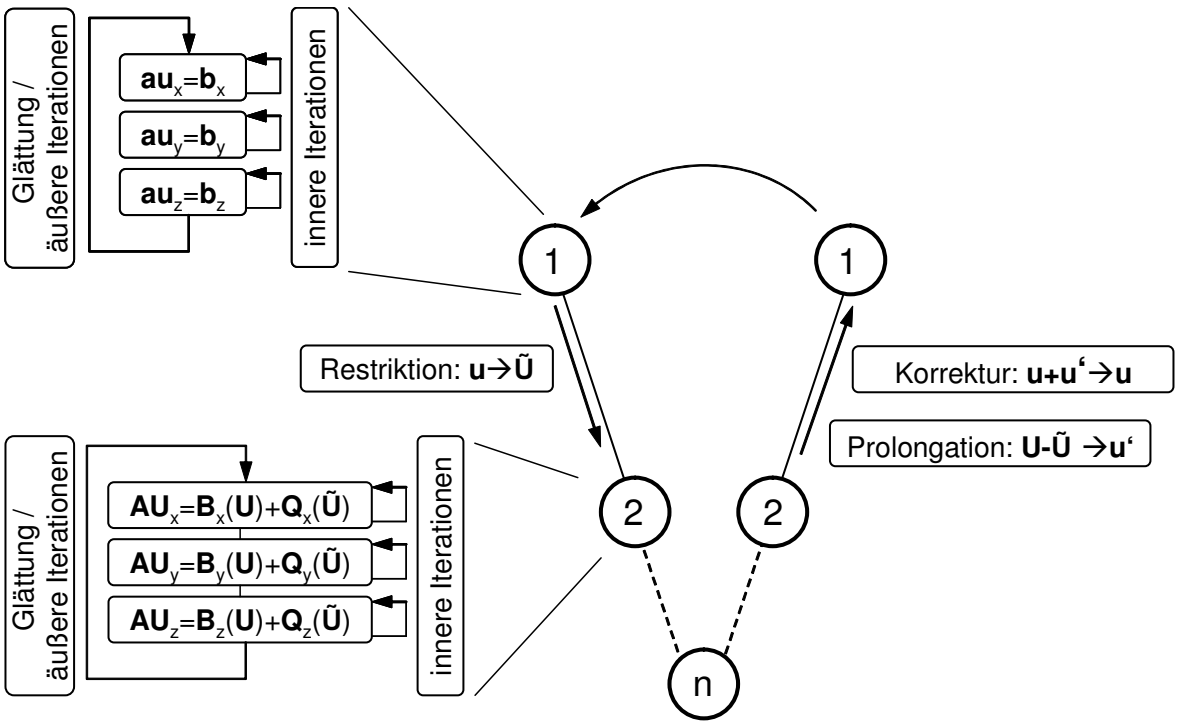

Abbildung 4.6: Schematische Darstellung des Mehrgitterverfahrens zur Lösung der Impulsgleichungen der Strukturmechnik bei Anwendung eines V-Zyklus und n Netzebenen. Die Impulsgleichungen sind entkoppelt diskretisiert, die resultierenden Koeffizientenmatrizen sind für alle Komponenten identisch. Durch Anwendung der äußeren Iterationen als Glätter wird die Kopplung der Gleichungen auf jedem Level wieder hergestellt.

Für die hier betrachteten Gleichungssysteme gilt:

$$\mathbf{a}\left(\boldsymbol{\phi}\right)\boldsymbol{\phi} = \mathbf{b}\left(\boldsymbol{\phi}\right), \tag{4.59}$$

wobei die Abhängigkeit der Koeffizientenmatrix und der rechten Seite von $\boldsymbol{\phi}$ der Übersichtlichkeit halber in den folgenden Ableitungen nicht mehr explizit vermerkt wird. Zunächst werden einige Glättungsschritte auf dem Feingitter durchgeführt und mit der momentanen Lösung $\boldsymbol{\phi}^*$ und dem Feingitterresiduum $\mathbf{r}$ wird geschrieben:

$$\mathbf{a}\boldsymbol{\phi} = \mathbf{b} + \mathbf{a}^*\boldsymbol{\phi}^* - \mathbf{b}^* + \mathbf{r} \tag{4.60}$$

Diese Gleichung wird nun auf das Grobgitter übertragen:

$$\mathbf{A}\boldsymbol{\Phi} = \mathbf{B} + \tilde{\mathbf{A}}\tilde{\boldsymbol{\Phi}} - \tilde{\mathbf{B}} + \mathbf{R} = \mathbf{B} + \mathbf{Q} \tag{4.61}$$

Die Grobgittergleichung entspricht somit der Feingittergleichung bis auf einen zusätzlichen Quellterm, der ein Maß für den auf dem Feingitter verbleibenden langwelligen Fehler darstellt.
Dieser zusätzliche Quellterm wird mit geeigneten Restriktionsvorschriften ermittelt. Die Lösungsgröße $\boldsymbol{\phi}$ wird im implementierten Verfahren nach der Vorschrift

$$\tilde{\Phi}_I = \frac{1}{N_i} \sum_{i \,\in\, I} \left[\boldsymbol{\phi}_i + \nabla\boldsymbol{\phi}_i \left(\mathbf{x}_i - \mathbf{x}_I\right)\right], \tag{4.62}$$

auf eine Grobgitterzelle I restringiert, wobei N_i der Anzahl der Feingitterzellen i entspricht, welche in der Grobgitterzelle I enthalten sind.

Mit der restringierten Lösungsgröße $\tilde{\boldsymbol{\phi}}$ werden die restlichen Terme $\tilde{\mathbf{A}}(\tilde{\boldsymbol{\Phi}})$ und $\tilde{\mathbf{B}}(\tilde{\boldsymbol{\phi}})$ bestimmt.
Die Restriktion der Feingitterresiduen $\mathbf{r}$ erfolgt mit Gl. 4.54.

Der zusätzliche Quellterm der Grobgittergleichung bleibt während der nichtlinearen Glättungsschritte konstant.

Nach diesen Glättungsschritten wird mit der momentanen Grobgitterlösung Φ_I eine Korrektur bestimmt

$$\Phi'_I = \Phi_I - \tilde{\Phi}_I. \tag{4.63}$$

welche auf das Feingitter mit der Prolongationsvorschrift

$$\boldsymbol{\phi}'_{i\in I} = \Phi'_I + \nabla_I \Phi'_I \left(\mathbf{x}_{i\in I} - \mathbf{x}_I\right). \tag{4.64}$$

übertragen wird. Die prolongierten Werten werden auf die Feingitterlösung aufgeschlagen.
Das Verfahren kann problemlos auf einen beliebigen Zyklus nach Abs. 4.5.3.1 erweitert werden.

Im Rahmen dieser Arbeit werden die Impulsgleichungen der Strukturmechanik mit diesem geometrischen FAS gelöst, wobei lediglich die Netzhierarchie mit Hilfe der algebraischen Vergröberungstrategie nach 4.5.3.1 erfolgt.
Alle zur Diskretisierung benötigten geometrischen Werte wie Zellmittelpunkte, Flächennormalen und Flächenmittelpunkte und Verbindungsvektoren von Nachbarzellen werden mit Hilfe der aggregierten Finite-Volumen-Zellen ermittelt.
Die Koeffizientenmatrizen enthalten in diesem speziellen Fall zwar konstante Werte und es liegt kein nichtlineares Problem vor, die einzelnen Gleichungen wurden jedoch derart diskretisiert, dass die Kopplung zwischen diesen lediglich durch die rechten Seiten $\mathbf{b}$ der diskreten Impulsgleichungen hergestellt wird.
Die vorgestellte sequentielle Lösungsmethode nach Abschnitt 4.6.2.1 versagt bei der entkoppelten Diskretisierung der Impulsgleichungen bei vielen Simulationsaufgaben. Mit Hilfe des FAS wird die Kopplung zwischen den Gleichungen mit Hilfe durchgeführter *äußeren Iterationsschritte* als *Glättungsoperationen* auf allen Netzebenen wieder hergestellt. Die *inneren Iterationsschritte* werden mit Hilfe der vorgestellten ILU(0) durchgeführt.
Die Feingitterlösung und die Feingitterresiduen werden mit Hilfe von Gl. 4.62 auf das Grobgitter restringiert und mit den restringierten Werte werden die Vektoren der rechten Seiten $\mathbf{b}$ der Impulsgleichungen gebildet. Diese enthalten alle explizit nachgeführten Korrekturen und alle Kopplungsterme. Die Koeffizientenmatrizen müssen nur einmal zu Beginn der Lösung für alle Level bestimmt werden. Das Verfahren ist im Überblick in Abb. 4.6 dargestellt.
Das Konvergenzverhalten wird mit diesem FAS deutlich verbessert, s. Abs. 6.2.1.7.

4.5.4 Anmerkungen zur Parallelisierung

Zur parallelen Simulation wird das Rechengebiet in Teilgebiete zerteilt. Ein einfaches zweidimensionales Beispiel ist mit der resultierenden Koeffizientenmatrix bei der vorgestellten Diskretisierungstechnik in Abb. 4.7 schematisch dargestellt. Von Null ver-

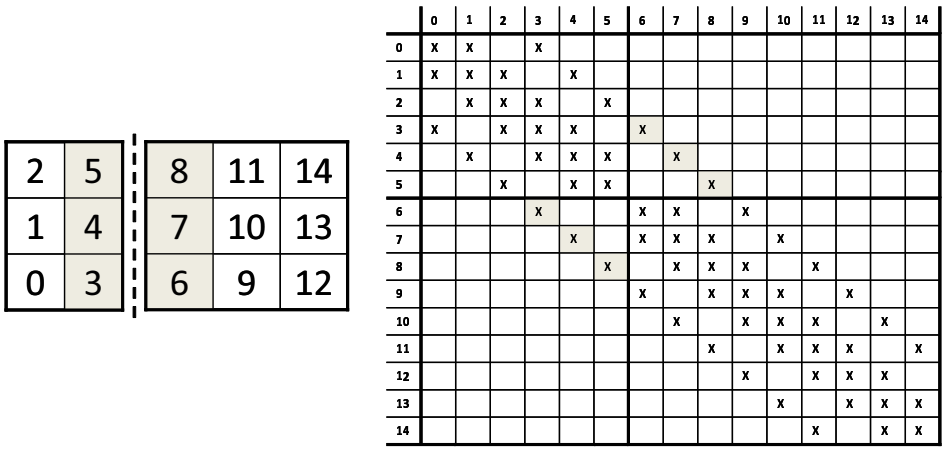

Abbildung 4.7: Einfaches zweidimensionales, partitioniertes Rechengebiet und Besetzungsstruktur der zugehörige Koeffizientenmatrix bei der angewendeten Diskretisierungstechnik. Grau hinterlegte Werte und Matrixeinträge werden in eigenen Datenfeldern gespeichert.

schiedene Matrixeinträge sind nur an den gekennzeichneten Positionen vorhanden. Im Diskretisierungs- und Lösungsprozess werden benötigte Werte an Gebietsgrenzen in zusätzlichen Feldern abgelegt. Die Matrixkoeffizienten, welche die einzelnen Gebiete mit einander koppeln, werden analog wie im Feldinneren gebildet und ebenfalls in zusätzlichen Feldern gespeichert.
Alle vorgestellten Lösungsverfahren sind mit den folgenden Vereinfachungen nach LILEK [67] effizient parallelisierbar:

- Beim Bilden der ILU- und der IC-Zerlegung werden lediglich die Diagonalkoeffizienten der Koeffizientenmatrizen modifiziert. Diese Modifikationen werden lokal auf den Teilgebieten durchgeführt und die Kopplungskoeffizienten werden nicht miteinbezogen.

- Die Vorkonditionierungsschritte und die inneren Iterationen werden ebenfalls lediglich lokal auf den Teilgebieten durchgeführt. Nach jedem Vorkonditionierungsschritt oder nach jeder innerer Iteration werden die Lösungsgrößen zwischen den Teilgebieten ausgetauscht.

- In den Mehrgitterverfahren werden die Vergröberungen lokal in den Teilgebieten vorgenommen. Die Kopplungskoeffizienten zwischen den Teilgebieten werden auf allen Grobleveln gebildet und in zusätzlichen Feldern gespeichert. Auf allen Leveln erfolgt nach jedem Relaxationsschritt ein Datenaustausch der aktuellen Werte zwischen den Teilgebieten.

Die vorgestellten Lösungsverfahren wurden in einer Programm-Bibliothek hinterlegt, auf welche das modulare Simulationswerkzeug zugreifen kann. Alle Verfahren sind mit Hilfe der MPI-Bibliothek, MPI [4], parallelisiert. Ferner sind alle Lösungsverfahren für einfache lineare Gleichungssysteme und für Blockgleichungssysteme anwendbar. Solche Gleichungssysteme entstehen, wenn mehrere Erhaltungsgleichungen in einem Diskretisierungsschritt zusammengefasst oder gekoppelt gelöst werden. Es bietet sich zum Beispiel an, das hydrodynamsiche Grundsystem gekoppelt zu lösen, s. ZWART [112]. Eine Gegenüberstellung der vorgestellten Lösungsverfahren findet sich in Anhang D.

4.6 Sequentielle Lösungsverfahren

Bei den implementierten Lösungsverfahren werden die die Problemstellungen beschreibenden diskretisierten Gleichungen nacheinander, bzw. sequentiell abgearbeitet. Alle diskretisierten Gleichungen stellen für die hinterlegten iterativen Lösungsverfahren nun lineare und voneinander entkoppelte Gleichungssysteme dar, s. 4.5.

Um der Kopplung und der Nichtlinearität Rechnung zu tragen, sind diese linearen Gleichungssysteme in eine äußere Iterationsschleife eingebettet. Während eines äußeren Iterationsschrittes müssen die linearen Gleichungssysteme nicht sehr genau gelöst werden, da die momentan vorliegenden Gleichungssysteme nur eine Näherung der untersuchten Problemstellung darstellen. Ferner werden einige Terme, wie nichtorthogonale Korrekturen, Skewness-Korrekturen oder nachgeführte Korrekturterme für die konvektiven Flüsse der Fluidgleichungen explizit mit Hilfe von Werten aus alten Iterationsschritten berechnet und müssen deswegen ebenfalls aktualisiert werden, s. Abschnitt 4.2.4 und 4.2.3.3.

Falls nach einer gewissen Anzahl von äußeren Iterationen alle gelösten Gleichungen ein vernünftig gewähltes Konvergenzkriterium erfüllen ist die Lösung erreicht, s. Kap. 4.5. Im Falle einer instationären Rechnung wird dann der nächste Zeitschritt begonnen.

Ein großer Vorteil dieser sequentiellen Lösungsweise ist die einfache Erweiterbarkeit auf unterschiedlichste Problemstellungen. Neu zu untersuchende Gleichungen und damit neue physikalische Modelle können einfach in die äußere Iterationsschleife eingebunden werden, wodurch bei entsprechender Programmstruktur eine gute Modularisierbarkeit erreicht wird, s. Kap.5. Ein weiterer Vorteil ist, dass in der Regel nur Speicher für eine Koeffizientenmatrix für alle linearen Gleichungssysteme bereitgestellt werden muss.

Eine gekoppelte Lösung des gesamten diskreten Gleichungssystems ist aus speichertechnischen Gründen für praktische Problemstellungen heute noch nicht möglich und auch nicht sinnvoll. Teile des Gleichungssystems, wie das hydrodynamische Grundsystem, bestehend aus Impulsgleichungen und Massenerhaltung, gekoppelt zu lösen, ist bei entsprechender Diskretisierungstechnik und entsprechenden Gleichungslösern, s. Abschnitt 4.5 heute möglich, s. ZWART [112], und wird heutzutage in kommerziellen Programmen wie ANSYS-CFX bereits erfolgreich angewendet.

Im Folgenden werden die hinterlegten sequentiellen Lösungsverfahren für fluid- und strukturmechanische Problemstellungen näher beschrieben und bei deren Anwendung auftretende Besonderheiten erläutert.

4.6.1 Strömungsmechanik

Mit Hilfe der vorgestellten Finite-Volumen-Diskretisierungstechnik erhält man für jede Transportgleichung ein lineares Gleichungssystem. In den nichtlinearen Impulserhaltungsgleichungen, den Turbulenzgleichungen und der Energiegleichung wurden die Koeffizientenbeiträge des konvektiven Terms mit Hilfe des a priori unbe-

kannten Geschwindigkeitsfeldes bestimmt. Nachgeführte Korrekturen werden für eine explizite Behandlung auf im Quellterm berücksichtigt.

Dies erfordert einen iterativen Lösungsalgorithmus notwendig, bei dem die Erhaltungsgleichungen innerhalb einer nichtlinearen oder äußeren Iterationsschleife sequentiell gelöst werden. Mit dem jeweils vorliegenden Geschwindigkeitsfeld werden die konvektiven Anteile der Koeffizienten aktualisiert. Mit den aktuellen Lösungsgrößen werden auch die explizit nachgeführten Korrekturterme neu berechnet.

Die linearen Gleichungssysteme müssen deswegen für die aktuelle äußere Iteration nicht sehr genau gelöst werden, typischerweise genügt ein Residuenabfall für die betrachtete Lösungsgröße um eine Größenordnung.

Bei einer instationären Berechnung ist zusätzlich die äußere Iterationsschleife in eine Zeitschleife eingebettet, welche die Simulation in die einzelnen Zeitschritte unterteilt. Die instationäre Berechnung wird somit in einzelne stationäre Teillösungen aufgeteilt.

Um die voneinander abhängigen Gleichungen bei dieser sequentiellen Lösung miteinander zu koppeln und ein schnelles und robustes Konvergenzverhalten zu gewinnen, wurden von PATANKAR [80] das SIMPLE-Verfahren (*S*emi-*I*mplict *M*omentum for *P*ressure-*L*inked *E*quations), hauptsächlich zur Lösung stationärer Problemstellungen, und von ISSA [56] das PISO-Verfahren (*P*ressure *I*mplicit with *S*plittung of *O*perators) für instationäre Problemstellungen entwickelt. Beide wurden im entwickelten Programmsystem hinterlegt und werden deswegen im Folgenden kurz zur Bestimmung eines inkompressiblen Strömungsfeldes beschrieben. Eine Erweiterung auf beliebige Machzahl-Bereiche ist einfach möglich und wird beispielsweise in DEMIRDZIC [30] vorgestellt.

4.6.1.1 SIMPLE-Algorithmus

Zu Beginn eines äußeren Iterationsschrittes werden die drei Impulsgleichungen gelöst. Man erhält mit Hilfe des momentan vorliegenden Druckfeldes, zu Beginn der Simulation mit Hilfe einer vernünftigen Schätzung des Drucks die neuen Geschwindigkeiten. Mit diesem nun vorliegenden Geschwindigkeitsfeld und mit Hilfe der Massenerhaltung wird dann ein neues Druckfeld bestimmt. Dies ist zunächst problematisch, da die Massenerhaltung den Druck nicht enthält. Weiterhin kann das aktuelle Geschwindigkeitsfeld aufgrund des geschätzten Druckfeldes die Massenerhaltung nicht erfüllen.

Ziel ist es nun, das vorliegende Druckfeld und das mit den Impulsgleichungen ermittelte Geschwindigkeitsfeld so zu korrigieren, dass die Massenerhaltung erfüllt werden kann:

$$\mathbf{v} = \tilde{\mathbf{v}}^* + \mathbf{v}' \tag{4.65}$$

bzw.:

$$p = p^{n-1} + p'. \tag{4.66}$$

Der Exponent n kennzeichnet dabei den aktuellen Iterationsschritt. Die gestrichenen Größen sind die Korrekturwerte. $\mathbf{v}^*$ ist das aus den Impulsgleichungen gewonnene Ge-

schwindigkeitsfeld.
Formal wurden die Geschwindigkeiten am betrachteten Punkt P mit den diskretisierten Impulsgleichungen berechnet:

$$\mathbf{v}^{n*} = \frac{\mathbf{b}^{n-1} - \sum_i A_i \mathbf{v}_i^{n*}}{A_P^{\mathbf{v}}} - \frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p^{n-1}), \tag{4.67}$$

welche umgeschrieben werden in:

$$\mathbf{v}^{n*} = \tilde{\mathbf{v}}^{n*} - \frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p^{n-1}). \tag{4.68}$$

Mit den korrigierten Werten aus Gl.4.65 und Gl.4.66 erhält man:

$$\mathbf{v}^{n} = \tilde{\mathbf{v}}^{n*} - \frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p^{n}). \tag{4.69}$$

Der erste Term auf der rechten Seite wurde beibehalten, denn die Geschwindigkeiten der Nachbarpunkte der aktuell betrachteten Zelle P werden nicht korrigiert.
Durch diese Vernachlässigung erhält man durch Substraktion der Gl.4.69 von Gl.4.68 eine einfache Beziehung zwischen den Geschwindigkeits- und den Druckkorrekturen:

$$\mathbf{v}' = -\frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p'). \tag{4.70}$$

Mit dieser Beziehung und mit Hilfe der inkompressiblen Massenerhaltung, s. Kap.2:

$$\nabla \mathbf{v} = \nabla \mathbf{v}^* + \nabla \mathbf{v}' = 0 \tag{4.71}$$

gewinnt man somit eine Gleichung für die gesuchten Druckkorrekturen:

$$\nabla \left[\frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p') \right]_P = \left[\nabla \mathbf{v}^* \right]_P . \tag{4.72}$$

Diese Poisson-Gleichung für die Druckkorrekturen an den Zellmittelpunkten P wird mit einem Standardverfahren gelöst, s. Abschnitt 4.5. Mit den Druckkorrekturwerten werden die Geschwindigkeitskorrekturen aus Gl.4.66 bestimmt und sowohl das Druck- als auch das Geschwindigkeitsfeld korrigiert. Die Massenerhaltung ist nun erfüllt, die Impulsgleichungen wegen ihrer Nichtlinearität jedoch noch nicht. Somit erfolgt der nächste äußere Iterationsschritt.

Die oben erwähnte Vernachlässigung der Geschwindigkeitskorrekturen der Nachbarwerte, die zum benötigten Zeitpunkt noch nicht bekannt sind, ist das Charakteristikum des SIMPLE-Verfahrens. Stabile und schnelle Konvergenz ist bei dieser Vorgehensweise nur durch starke Unterrelaxation der Druckkorrekturwerte und für komplexe Problemstellungen auch der Geschwindigkeitskorrekturwerte zu erhalten, s. SKODA [93]. Es wurden zahlreiche Versuche unternommen, diese Vernachlässigung zu kompensieren. Zu erwähnen sind das SIMPLER-Verfahren nach PATANKAR [79] und das SIMPLEC-Verfahren nach DOORMAL [100]. Bei praktischen Problemstellungen wurden jedoch keine grundlegenden Verbesserungen im Konvergenzverhalten gegenüber dem ursprünglichen SIMPLE-Verfahren erreicht.

Ein sehr effizientes, sequentielles Verfahren stellt das PISO-Druckkorrekturverfahren für instationäre Berechnungen dar, welches im folgenden Abschnitt erläutert wird.

4.6.1.2 PISO-Algorithmus

Das PISO-Druckkorrekturverfahren wurde von ISSA [56] für die Simulation instationärer Problemstellungen entwickelt.
Das prinzipielle Vorgehen bei der Herleitung folgt dem des SIMPLE-Verfahrens, und es werden Korrekturwerte für das Geschwindigkeitsfeld und das Druckfeld gesucht. Die Korrekturen der Geschwindigkeiten der Nachbarwerte in 4.68 werden hier nicht mehr vernachlässigt, und die korrigierten Impulsgleichungen lauten daher:

$$\mathbf{v}^n = \frac{\mathbf{b}^{n-1} - \sum_i A_i \mathbf{v}_i^n}{A_P^{\mathbf{v}}} - \frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p^n). \tag{4.73}$$

Zieht man Gl. 4.68 von Gl. 4.73 ab, erhält man nun für die Geschwindigkeitskorrekturen:

$$\mathbf{v}' = -\frac{\sum_i A_i \mathbf{v}_i'}{A_P^{\mathbf{v}}} - \frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p'). \tag{4.74}$$

Diese Korrekturwerte werden nun nochmals korrigiert:

$$\mathbf{v}' + \mathbf{v}'' = -\frac{\sum_i A_i \mathbf{v}_i'}{A_P^{\mathbf{v}}} - \frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p' + p''). \tag{4.75}$$

Zieht man von dieser Gl.4.75 Gl.4.70 ab, erhält man einen Ausdruck für die zweiten Geschwindigkeitskorrekturen in Abhängigkeit der ersten Korrekturwerte und der zweiten Druckkorrekturen:

$$\mathbf{v}'' = -\frac{\sum_i A_i \mathbf{v}_i'}{A_P^{\mathbf{v}}} - \frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p''). \tag{4.76}$$

Da die Kontinuitätsgleichung nach dem ersten Korrekturschritt bereits erfüllt war und mit der Forderung, dass diese auch nach der zweiten Korrektur erfüllt bleibt, lautet die nun zu lösende Gleichung:

$$\nabla \mathbf{v}'' = 0, \tag{4.77}$$

welche mit Gl.4.76 eine Poisson-Gleichung für die zweiten Druckkorrekturen liefert:

$$\nabla \left[\frac{\Delta V}{A_P^{\mathbf{v}}} \nabla(p'') \right]_P = \left[\frac{\sum_i A_i \mathbf{v}_i'}{A_P^{\mathbf{v}}} \right]_P. \tag{4.78}$$

Es lassen sich analog zu obigem Vorgehen weitere Korrekturgleichungen herleiten. Bei einem orthogonalen Rechengitter sind zwei Korrekturschritte innerhalb eines Zeitschritts ausreichend, s. ISSA [56]. Nichtorthogonale Korrekturen, wie in Abschnitt 4.2.4 beschrieben, können direkt in diese Korrekturschritte mit eingearbeitet werden.

Bei Einhaltung eines kleinen Zeitschritts, s. Kap. 4.7, sind bei diesem Verfahren keine äußeren Iterationsschritte notwendig, und das Verfahren ist für diese Art der Problemstellungen dem SIMPLE-Verfahren hinsichtlich des zu betreibenden Rechenaufwandes deutlich überlegen.

Grundsätzlich könnten diese nachgezogenen Korrekturen auch in einem stationären Simulationsprozess durchgeführt werden. Das Konvergenzverhalten konnte aber bei den untersuchten komplexen Problemstellungen dadurch nicht verbessert werden.

Dadurch, dass die Korrekturwerte der Nachbarpunkte in die Berechnung mit einbezogen werden, müssen die Matrixkoeffizienten der Impulsgleichungen zusätzlich abgespeichert werden.

Der große Vorteil der sequentiellen Verfahren ist zum einen der niedrige Speicherbedarf, da nur eine Koeffizientenmatrix, bzw. im PISO-Verfahren zwei gespeichert werden muss, zum anderen, dass zusätzliche abzuarbeitende Gleichungen einfach hinzugefügt und mitgelöst werden können.

4.6.1.3 Diskretisierung der Druckkorrekturgleichungen

Alle Gleichungen wurden an den Zellmittelpunkten diskretisiert. PATANKAR[79] stellte jedoch fest, dass bei dieser nicht versetzten Variablenanordnung Geschwindigkeiten und Druck von einander gekoppelt sind und unphysikalische Oszillationen in der numerischen Lösung auftreten können und es in den meisten Fällen überhaupt nicht möglich ist, eine Lösung zu generieren.
PATANKAR [79] schlägt zur Vermeidung dieser Oszillationen eine Variablenanordnung vor, bei der Geschwindigkeiten und Druck auf zueinander versetzten Rechennetzen gelöst werden. Für konturangepasste und unstrukturierte Rechennetze ist dies jedoch schwierig zu realisieren, und die Datenhaltung wird sehr kompliziert. Deswegen wurde ein Korrekturterm nach RHIE[86] in die Diskretisierung miteinbezogen, der die Kopplung der Gleichungen sicherstellt und durch den die Oszillationen vermieden werden können.
Die Diskretisierung der Druckkorrekturgleichung geht von der korrigierten Massenerhaltungsgleichung aus:

$$\sum_i \dot{m}_i' + \sum_i \dot{m}_i^* = 0. \tag{4.79}$$

Der erste Term enthält die Korrekturmassenströme, welche mit Gl. 4.65 für jede Zellfläche f mit den Druckkorrekturen ausgedrückt werden:

$$\dot{m}_f' = \rho_f u_{n,f}' S_f = -S_f \overline{\frac{\Delta V}{A_P^{\mathbf{v}}}}_f \left(\frac{\partial p'}{\partial n}\right)_f . \tag{4.80}$$

Das Zellvolumen ΔV und die Zentralkoeffizienten der Impulsgleichungen $A_P^{\mathbf{v}}$ werden dabei zentral an die betrachtete Zellfläche interpoliert.
Der zweite Term wird nun mit dem von RHIE [86] vorgeschlagenen Korrekturterm und den Werten aus der alten Iteration diskretisiert:

$$\dot{m}^* = \rho_f u_{n,f}^* S_f = \rho_f \left(u^* S_f^x + v^* S_f^y + w^* S_f^z\right) - \overline{\frac{\rho_\Delta V}{A_P^{\mathbf{v}}}}_f \left[\left(\frac{\partial p}{\partial n}\right)_f - \overline{\left(\frac{\partial p}{\partial n}\right)}_f\right]^{n-1} . \tag{4.81}$$

Dabei werden die verwendeten Druckgradienten nach PERIC [81] mit Hilfe von

$$\left(\frac{\partial p}{\partial n}\right)_f = \frac{\overline{(\Delta V)}_f}{(\mathbf{r}_N - \mathbf{r}_P)\,\mathbf{n}_f} (p_N - p_P) , \tag{4.82}$$

bzw. mit

$$\overline{\left(\frac{\partial p}{\partial n}\right)}_f = \frac{\overline{(\Delta V)}_f}{(\mathbf{r}_N - \mathbf{r}_P)\,\mathbf{n}_f} \overline{\left(\frac{\partial p}{\partial n}\right)}_f (\mathbf{r}_N - \mathbf{r}_P) \tag{4.83}$$

bestimmt.

Bei der Herleitung der Druckkorrekturgleichung werden die reziproken Zentralkoeffizienten der Impulsgleichungen verwendet. Das ist nur aufgrund einer speziellen Diskretisierung der Wandrandbedingungen der Impulsgleichungen mit einem Nachgezogenen-Korrektur-Ansatz nach PERIC [81] möglich, aus dessen Anwendung identische Koeffizienten für alle drei Komponenten der Impulsgleichungen resultieren.
Die verwendeten Zentralkoeffizienten der Impulsgleichungen enthalten bei instationären Rechnungen den Reziprokwert des Zeitschrittes. Deswegen kann die verwendete Diskretisierung nach RHIE [86] bei Simulationen mit sehr kleinem Zeitschritt zu Instabilitäten führen, da der Korrekturterm aus Gl. 4.81 seine Wirkung verliert. Nach PERIC [81] treten derartige Probleme allerdings erst bei einer Courant-Zahl $Co < 0.01$ auf und waren bei Simulationen im Rahmen dieser Arbeit nicht relevant.
Die Druckkorrekturgleichungen wurden derart diskretisiert, dass keine Abhängigkeit der Größe des Korrekturterms aus Gl. 4.81 vom Unterrelaxationsfaktor α der Impulsgleichungen vorliegt, s. Abschnitt 4.7.2.

Sowohl der SIMPLE- als auch der PISO-Algorithmus können direkt mit der Lösungsvariable p formuliert werden, s. JASAK [59]. Dadurch entsteht ein physikalisch anschaulicheres Verfahren, welches keine künstlichen, zusätzlichen Größen wie die Druckkorrektur benötigt und darüber hinaus hinsichtlich Stabilität und Konvergenzeigenschaften der hier vorgestellten Druckkkorrekturvariante überlegen sein kann, s. FIEREDER [37]. Die grundsätzlich auftretenden Probleme bei der Druck-Geschwindigkeitskopplung bleiben erhalten und die daraus resultierenden Diskretisierungstechniken müssen weiterhin angewendet werden.
Der sequentielle Lösungsverlauf für den beschriebenenen SIMPLE-Algorithmus der Fluid-Gleichungen ist in Abb. 4.8 dargestellt.

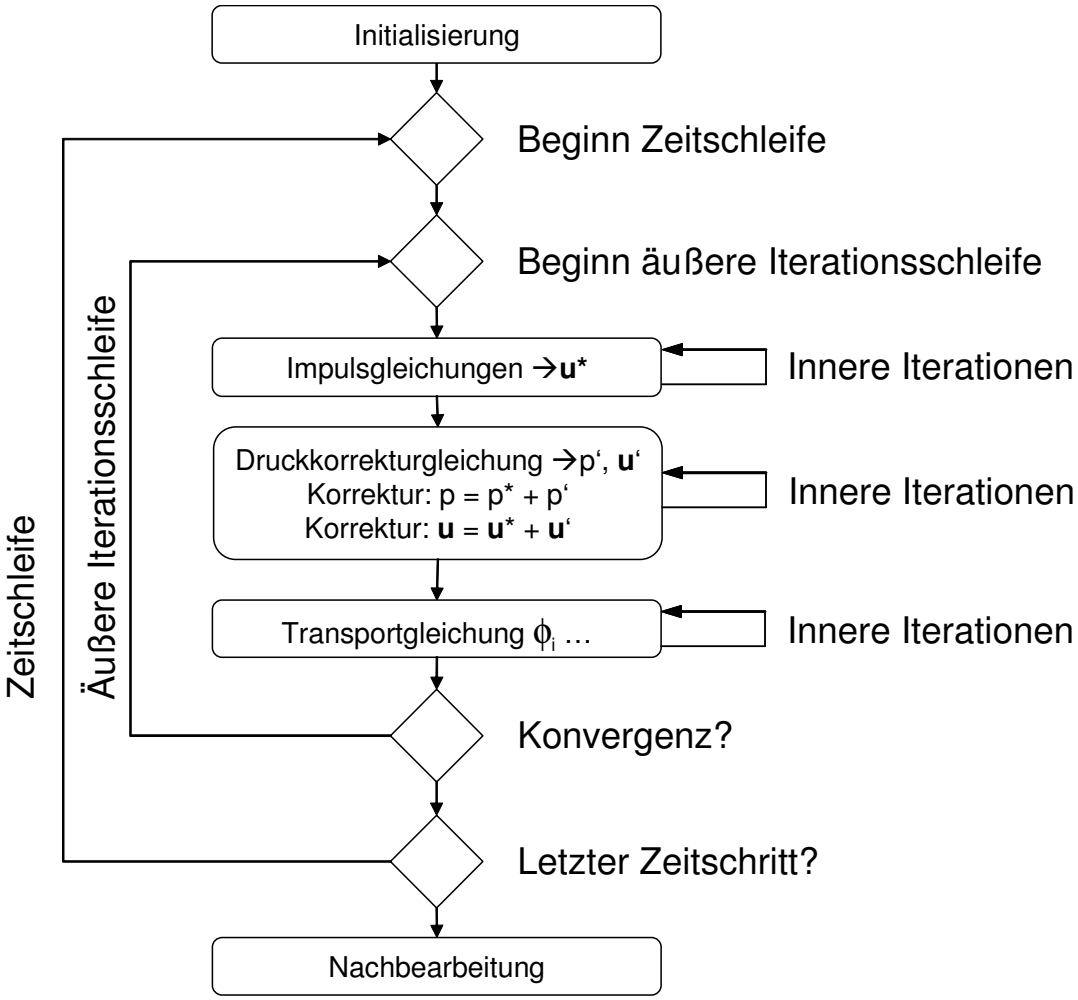

Abbildung 4.8: Ablaufdiagramm des sequentiellen SIMPLE-Algorithmus.

4.6.2 Strukturmechanik

4.6.2.1 Einheitliche Formulierung und Implementierung

Die Gleichungen zur Simulation strukturmechanischer Problemstellungen werden sequentiell abgearbeitet. Die Impulsgleichungen der Strukturmechanik enthalten in der CF, der LF und der TLF keine konvektiven Terme, so dass auf der linken Seite der Gleichungen nur der Zeitterm und der diffusive Term der jeweils betrachteten Verschiebung auftreten. Alle anderen Terme, die durch Anwendung auf ein spezielles Materialgesetz auftreten, werden explizit auf der rechten Seite der diskretisierten Gleichungen behandelt. Im Rahmen dieser Arbeit wurde nur das Saint-Venant-Materialgesetz bzw. das Hooke-Gesetz für elastische Verformungen angewendet.
Von JASAK [60] wurde bei Anwendung dieser Materialgesetze vorgeschlagen, als Diffusionskoeffizienten $\Gamma = 2\mu + \lambda$ zu verwenden, wodurch eine identische Koeffizientenmatrix für alle drei Impulsgleichungen resultiert und das Konvergenzverhalten des Verfahrens gegenüber Verfahren mit anderer Diskretisierungstechnik stark verbessert wird. Man spricht hier von einer Überrelaxierung des Gleichungssystems.
Mit dieser Überrelaxierung werden nun die Impulsgleichungen für die CF, die TLF und

die ULF in inkrementeller Schreibweise formuliert:

$$\int_{dV} \frac{\partial^2 (\rho \mathbf{u})}{\partial t^2} dV - \int_{dS} (\delta \mathbf{\Sigma}) \mathbf{n} dS = \int_{dV} (\rho \mathbf{f}) dV \tag{4.84}$$

bzw.:

$$\int_{dV_0} \frac{\partial^2 (\rho_0 \delta \mathbf{u})}{\partial t^2} dV_0 - \int_{dS_0} (\delta \mathbf{\Sigma}) \mathbf{n}_0 dS_0 = \int_{dV_0} (\rho_0 \delta \mathbf{f}) dV_0 + \int_{dS_0} \left[\nabla \mathbf{u}^T + (\mathbf{\Sigma} + \delta \mathbf{\Sigma}) \delta \mathbf{F}^T \right] \mathbf{n}_0 dS_0 \tag{4.85}$$

bzw.:

$$\int_{dV_u} \frac{\partial^2 (\rho_u \delta \mathbf{u})}{\partial t^2} dV_u - \int_{dS_u} (\delta \mathbf{\Sigma}_u) \mathbf{n}_u dS_u = \int_{dV_u} (\rho_u \delta \mathbf{f}) dV_u + \int_{dS_u} \left[(\mathbf{\Sigma}_u + \delta \mathbf{\Sigma}_u) \delta \mathbf{F}_u^T \right] \mathbf{n}_u dS_u. \tag{4.86}$$

Für das Inkrement des Spannungstensors Σ gilt mit der Anwendung des Materialgesetzes für alle Formulierungen:

$$\delta \mathbf{\Sigma} = 2\mu \delta \mathbf{E} + \lambda tr (\delta \mathbf{E}) \mathbf{I}, \tag{4.87}$$

wobei der Ausdruck des Dehnungs-Verzerrungs-Tensors $\mathbf{E}$ an die jeweilige Formulierung angepasst werden muss. Unabhängig von der Formulierung kann dieser Tensor in einen linearen und einen nichtlinearen Anteil aufgespalten werden:

$$\mathbf{E}^F = \mathbf{E}_{lin} + \mathbf{E}^F_{nlin}, \tag{4.88}$$

wobei F für die jeweils angewendete Formulierung steht. Der lineare Anteil ist in allen Formulierungen identisch, nur der nichtlineare Anteil wird angepasst. Man erhält nun für das Inkrement des Spannungstensors:

$$\delta \mathbf{\Sigma} = 2\mu \delta \mathbf{E}_{lin} + \lambda tr (\delta \mathbf{E}_{lin}) \mathbf{I} + 2\mu \delta \mathbf{E}^F_{nlin} + \lambda tr \left(\delta \mathbf{E}^F_{nlin} \right) \mathbf{I}. \tag{4.89}$$

Mit Gl. 4.89 und dem bereits erwähnten Vorschlag von JASAK [60] für eine Überrelaxierung erhält man den für die Diskretisierung fertigen Ausdruck des Inkrements des Spannungstensors:

$$\delta \mathbf{\Sigma} = (2\mu + \lambda) \nabla \delta \mathbf{u} + \mu (\nabla \delta \mathbf{u})^T - (\mu + \lambda) \nabla \delta \mathbf{u} + \lambda \mathbf{I} tr (\nabla \delta \mathbf{u}) + 2\mu \delta \mathbf{E}^F_{nlin} + \lambda tr \left(\delta \mathbf{E}^F_{nlin} \right) \mathbf{I}, \tag{4.90}$$

wobei nur der erste Term der rechten Seite implizit im Gleichungssystem verarbeitet wird.
Für die drei Impulsgleichungen ergibt sich deswegen für alle vorgestellten Formulierungen eine einheitliche Diskretisierung, und nur die rechte Seite muss jeweils bei der Diskretisierung speziell behandelt werden. Der implizite diffusive Term der Impulsgleichungen wird mit einer nachgezogenen Korrektur für die diffusiven Flüsse implementiert, s. 4.23.

4.6.2.2 Inkrementelles Rechenverfahren

Die Impulsgleichungen wurden im vorhergehenden Abschnitt einheitlich formuliert und können nach der durchgeführten Diskretisierung nun iterativ gelöst werden.
In einem instationären, inkrementellen Verfahren müssen die Verschiebungs- und die Spannungsinkremente zur Gesamtverschiebung und zur Gesamtspannung kumuliert werden:

$$\mathbf{u} = \mathbf{u}^o + \mathbf{u} \tag{4.91}$$

bzw.:

$$\mathbf{\Sigma} = \mathbf{\Sigma}^o + \delta\mathbf{\Sigma}, \tag{4.92}$$

wobei o den verfügbaren Wert aus dem letzten, alten Zeitschritt bezeichnet.
Wird während des Berechnungsvorgangs das Strukturnetz verformt, werden die Knotenwerte der Verschiebungen benötigt, und die an den Zellmittelpunkten ermittelten Lösungen müssen an die Netzknoten interpoliert werden, s. 4.6.3.2. Durch die Netzverformung wird eine neue Referenzkonfiguration generiert. Bei der ULF muss der 2. Piola-Kirchhoff-Spannungstensor in diese neue Referenzkonfiguration transformiert werden:

$$\mathbf{\Sigma} := \frac{1}{\det \mathbf{F}} \mathbf{F} \mathbf{\Sigma} \mathbf{F}^T. \tag{4.93}$$

Alle zur Diskretisierung benötigten geometrischen Größen wie Zellvolumina, Zellflächen, Zellzentren, Flächenzentren und Flächennormalen müssen ebenfalls neu berechnet werden.
Der sequentielle Lösungsverlauf für die Struktur-Gleichungen ist in Abb. 4.9 dargestellt.

4.6.2.3 Inkompressibles Materialverhalten

Unter Anwendung geeigneter Materialgesetze können mit dem Finite-Volumen-Modul auch inkompressible Materialien untersucht werden, s. Kap. 3.4.2.
Da für Problemstellungen dieser Art der Druck als zusätzliche unbekannte Größe auftritt, können analog zu den sequentiellen Verfahren für fluidmechanische Problemstellungen nun auch Druck- bzw. Druckkkorrekturverfahren für die gekoppelten strukturmechanischen Problemstellungen angewendet werden, s. Abschnitt 4.6.1. Die Herleitung verläuft analog, wenn man in den entsprechenden Gleichungen für die Massenerhaltung und den aus dieser resultiernden Druck- bzw. Druckkorrekturgleichungen, anstelle der Geschwindigkeiten $\mathbf{v}$ die Verschiebungen $\mathbf{u}$ verwendet.
Im Konvergenzverhalten und in der Stabilität des Verfahrens treten dann die gleichen Vorzüge und Nachteile der Druckkorrekturverfahren auch für strukturmechanische Problemstellungen auf wie z. B. eine starke Unterrelaxation der Druckkorrekturen, s. Abschnitt 4.7.1.
Ferner können die Erhaltungsgleichungen bei der Untersuchung dieses Materialverhaltens in einfacher Weise gekoppelt diskretisiert und gelöst werden, wie es für die fluidmechanischen Gleichungen von üblich ist, s. z. B. ZWART [112].

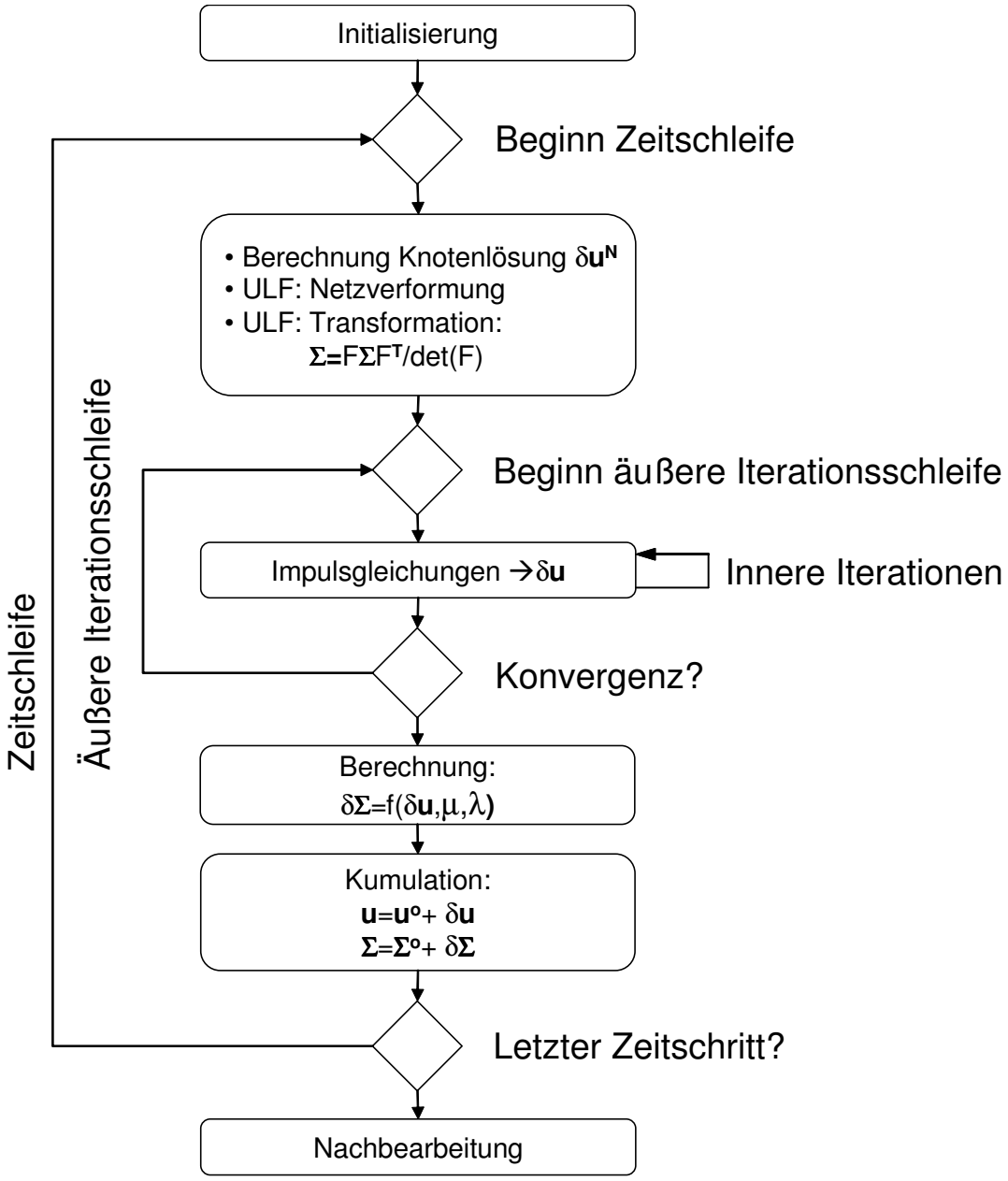

Abbildung 4.9: Sequentielle Lösung der Strukturgleichungen.

4.6.2.4 Schwächen des sequentiellen Lösungsverfahrens

Durch die Überrelaxierung der Impulsgleichungen nach JASAK [60] wird für viele unterschiedliche Problemstellungen ein schnelles und stabiles Konvergenzverhalten erzielt. Eine Unterrelaxation, wie sie in der Regel zur Lösung der Fluidgleichungen angewendet werden muss, ist nur in wenigen Fällen notwendig, s. Kap. 6.2. Trotzdem wird die sequentielle Methode für einige spezielle Problemstellungen sehr zeitaufwändig und damit ineffizient. Dieses Versagen tritt besonders auf bei Simulationen, bei welchen

- die stationäre Berechnung
- eines kompressiblen Materials unter

- linear-elastischen Verformungszuständen
- bei dünnwandigen oder langgestreckten Bauteilen
- unter Querbelastung

untersucht werden soll. Bei instationären, inkompressiblen Problemstellungen oder Problemstellungen mit geometrischen Nichtlinearitäten tritt dieses Verhalten in der Regel nicht auf.
Gründe für dieses Versagen und Möglichkeiten mögliche Abhilfen werden in den folgenden Abschnitten diskutiert.

Bei den sequentiellen Lösungsverfahren werden große Teile der Erhaltungsgleichungen explizit aus der vorangegangenen Iteration berechnet und damit auf den Quellterm der rechten Seite der zu lösenden Gleichungen geschrieben. Unter Verwendung des Hooke-Materialgesetzes lautet der Quellterm $\mathbf{q}$ der rechten Seite der Impulsgleichungen:

$$\mathbf{q} = (2\mu + \lambda)\,\nabla\delta\mathbf{u} + \mu\delta\mathbf{u}^T - (\mu + \lambda)\,\nabla\delta\mathbf{u} + \lambda tr\,(\nabla\delta\mathbf{u})\,\mathbf{I}. \tag{4.94}$$

Die expliziten Korrekturterme für die Nichtorthogonalitäten treten erst in der für die Diskretisierung umgeformten Darstellung auf und stellen bei qualitativ guten Netzen kein Problem dar, s. Abschnitt 4.2.4. Die Problemstellung tritt aber auch bei vollkommen orthogonalen Netzen auf, weswegen dieser Term in der folgenden Betrachtung ausgeschlossen werden kann. Die anderen Terme werden im Folgenden diskutiert:

- $\lambda tr(\nabla\delta\mathbf{u})\mathbf{I}$: Dieser Term beschreibt die Kompressibilität des Materials und kann bei der dargestellten Problemstellung als vernachlässigbar klein angenommen werden.
- $-(\mu + \lambda)\nabla\delta\mathbf{u}$: Dieser Term kompensiert die implementierte Überrelaxation und dient in Kombination mit dieser der Beschleunigung des Lösungsverhaltens, s. JASAK [60].
- $\mu(\nabla\delta\mathbf{u})^T$: Dieser Term, der den transponierten Gradienten der Verschiebungen beinhaltet, verursacht die Problematik der implementierten sequentiellen Lösungsstrategie, da die beiden vorangegangenen Terme keine Verschlechterung des Lösungsverhaltens bewirken können,

Wie eben gezeigt wurde, stellt der transponierte Gradient der Verschiebungen einen limitierenden Faktor für die Effizienz des sequentiellen Lösungsverfahrens dar. Der Term ist - anders als der entsprechende Term im Schubspannungstensor der Impulsgleichungen der Fluidmechanik - in vielen Problemstellungen sehr groß und eignet sich oft nicht zur expliziten Behandlung. Er enthält wichtige koppelnde Terme zwischen den Komponenten der Impulsgleichungen und müsste zur Verbesserung des sequentiellen Lösungsverfahrens implizit behandelt werden.

Bei der impliziten Diskretisierung dieses Terms entstünden jedoch große Rechenmoleküle, in die nicht nur unmittelbare Nachbarn der betrachteten Rechenzelle,

sondern auch Nachbarn der Nachbarzellen miteinbezogen werden müssten. Dadurch stiege der Speicheraufwand zur Lösung der strukturmechanischen Gleichung stark an. Somit müssten die Diskretisierungstechniken stark angepasst, bzw. spezialisiert werden, und zudem wäre eine gekoppelte Lösung mit daraus resultierendem erhöhtem Speicheraufwand der Gleichungen notwendig.

Ein weiterer Ansatz, um eine bessere Kopplung zwischen den Impulsgleichungen zu erreichen und damit das Lösungsverfahren für diese Art der Problemstellungen zu verbessern, bestünde darin, die betrachteten Strukturen als inkompressibel zu betrachten, s. 3.4.2. Wie in Abschnitt 4.6.2.3 gezeigt, könnte dann, ähnlich wie bei der Lösung der Fluidgleichungen und unter Anwendung eines geeigneten Materialgesetzes, eine Druckkorrekturgleichung abgeleitet werden, die die Komponenten der Impulslgeichungen dann über den Druck koppelt und damit das Konvergenzverhalten verbessert. Es wird hierbei allerdings ein Modellierungsfehler in Kauf genommen, welcher bei vielen Problemstellungen nicht akzeptiert werden kann, s. BATHE [15].

Ein dritter Ansatz, um eine bessere Kopplung zwischen den Impulsgleichungen zu erreichen und damit das Lösungsverfahren für diese Art der Problemstellungen zu verbessern, wäre die Lösung des Impulsgeichungssystems mit Hilfe eines FAS-Mehrgitterverfahrens. Dieser Ansatz wurde im entwickelten Verfahren umgesetzt und wurde eingehend was in Abs. 4.5.3.2 beschrieben.

4.6.3 Fluid-Struktur-Interaktion

4.6.3.1 Fluid-Struktur-Interface

Für eine physikalisch korrekte und stabile numerische Simulation ist ein konservativer Datenaustausch am Fluid-Struktur-Interface zwingend notwendig. Einerseits müssen vom Fluid-Gebiet die Spannungen und Drücke auf die angrenzenden Strukturen übertragen werden, andererseits die durch Verformung der Struktur neuen Berandungen des Fluids vom Struktur- zum Fluidgebiet hin. Weiterhin erhalten die Fluidgebiete die Geschwindigkeiten der angrenzenden, verformten Strukturgebiete als zusätzliche Randbedingungen. Für einen konservativen Datenaustausch gelten dann am Fluid-Struktur-Interface IF die Beziehungen:

$$\int_S \sigma_{Fluid} \mathbf{n} dS_{IF} = \int_S \sigma_{Struktur} \mathbf{n} dS_{IF} \tag{4.95}$$

bzw.:

$$\mathbf{v}_{IF,Fluid} = \mathbf{v}_{IF,Struktur}. \tag{4.96}$$

Dies wird einfach an topologisch passenden Interfaces erreicht, wo die Netzknoten übereinstimmen. Dies stellt bei einem polyedrischen Finite-Volumen-Verfahren keine Einschränkung hinsichtlich dessen Flexibilität dar, da jedes Interface durch einen implementierten Verschneidungsalgorithmus und das allgemeine Netzinterface, s. 4.8.1, passend gemacht werden kann. Ein derartiges Interface ist für den ebenen Fall in Abb. 4.10 unter Verwendung eines Finite-Volumen-Strukturmoduls dargestellt. Bei FE-Strukturberechnungen werden die Verformungen der Strukturen an den Knoten

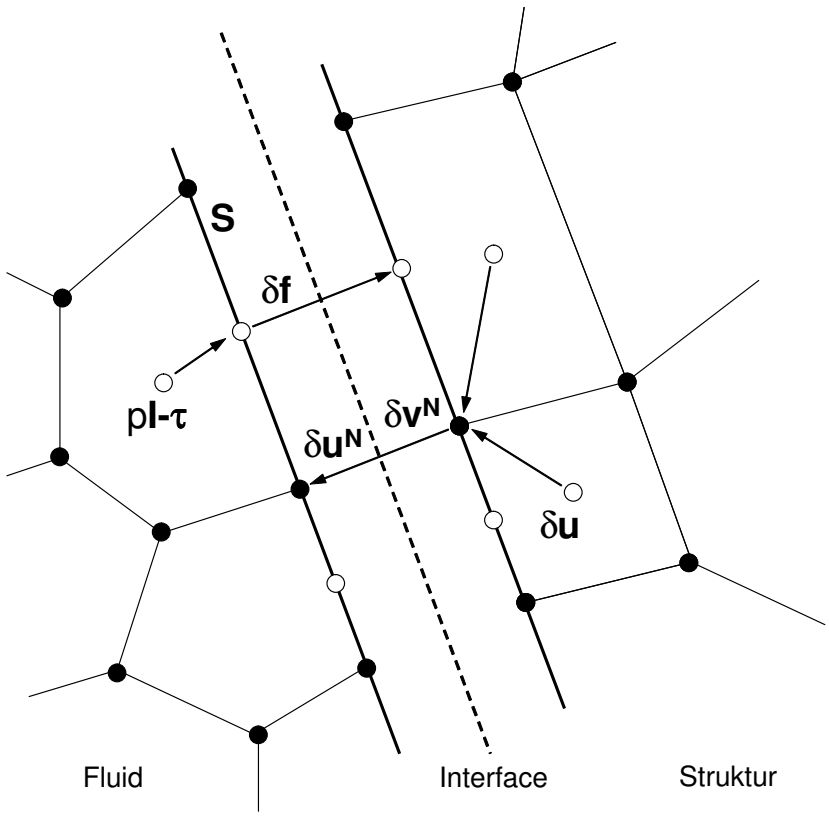

Abbildung 4.10: Datenaustausch am Fluid-Struktur-Interface bei Finite-Volumen-Strukturberechnung.

der Strukturnetze bestimmt. Deswegen müssen die Fluidkräfte in diesem Fall an die Netzknoten interpoliert werden. Die Verformungen der Strukturen können in diesem Fall direkt übergeben werden.

4.6.3.2 Netzverformung

Bei der Struktursimulation mit großen Verformungen und der Modellierung mit Hilfe der FLF müssen die Strukturnetze verformt werden. Da Verschiebungen $\mathbf{u}$ an den Zellmittelpunkten Z der finiten Volumina bestimmt werden, müssen diese in geeigneter Weise an die Netzknoten N interpoliert werden. Eine einfache Bestimmung der Knotenverschiebung über eine Volumen-gewichtete Mittelwertbildung aus den an den betreffenden Knoten angrenzenden Zellwerten i

$$\mathbf{u}^N = \frac{\sum_i \left(\Delta V_i^Z \mathbf{u}^Z\right)}{\sum_i \left(\Delta V_i^Z\right)} \tag{4.97}$$

ist einfach zu implementieren und schnell in der Ausführung, reicht aber für genaue Simulationen mit großen Deformationen nicht aus.
Deswegen werden im entwickelten Verfahren die Knoten-Verschiebungen mit Hilfe der über die Abstände gewichteten umliegenden Zellwerte und den umliegenden Gradienten bestimmt:

$$\mathbf{u}^N = \frac{\sum_i \frac{1}{|\Delta \mathbf{x}|}\left(\mathbf{u}^Z + \nabla \mathbf{u}^Z \Delta \mathbf{x}\right)}{\sum_i \frac{1}{|\Delta \mathbf{x}|}}. \tag{4.98}$$

Der Berechnungsaufwand ist größer als bei der Methode nach 4.97, die Simulationsergebnisse werden jedoch deutlich besser. TUKOVIC [98] geht sogar noch einen Schritt weiter und bezieht die zweiten Ableitungen der Verschiebungen in die Berechnung mit ein

$$\mathbf{u}^N = \frac{\sum_i \frac{1}{|\Delta\mathbf{x}|}\left(\mathbf{u}^Z + \nabla\mathbf{u}^Z\Delta\mathbf{x} + \frac{1}{2}\Delta\mathbf{x}^T\nabla\nabla\mathbf{u}^Z\Delta\mathbf{x}\right)}{\sum_i \frac{1}{|\Delta\mathbf{x}|}}, \tag{4.99}$$

wodurch der Rechenaufwand zur Netzverformung stark ansteigt, jedoch in den erreichten Lösungen keine Verbesserungen erzielt werden konnten.

Bei FSI-Simulationen müssen nicht nur die Strukturnetze, sondern auch die Netze des Fluidgebiets verformt werden, wenn diese an eine verformte Struktur angrenzen. Hierfür wurden unterschiedliche Methoden implementiert, welche sich hinsichtlich Genauigkeit, Schnelligkeit und Robustheit unterscheiden, sich prinzipiell in zwei Klassen von Algorithmen unterteilen lassen und im folgenden Abschnitt ausführlich beschrieben werden.

Differentielle Methoden

Bei den differentiellen Methoden zur Netzverformung werden die Verschiebungen der Netzknoten innerhalb des Fluidgebiets mit Hilfe partieller Differentialgleichungen ermittelt. Die Randbedingungen oder Randverschiebungen werden direkt aus der Verformung des Strukturgebietes gewonnen.

Bei den Federanalogie-Modellen, s. z. B. BATINA [16], werden die Kanten der Rechenzellen als Federn aufgefasst, die an den Netzknoten miteinander verbunden sind. Vor der Netzverformung verschwinden alle Federkräfte. Bei Verschiebungen der Berandungen werden die Federn dann gestreckt oder gestaucht um eine Länge Δl, die Federkräfte sind zu dieser Länge proportional. Über die Formulierung des Kräftegleichgewichts an jedem inneren Netzknoten gewinnt man ein lineares Gleichungssystem zur Bestimmung der Knotenverschiebungen. Für die Federn sind noch geeignete Federkonstanten zu ermitteln. Die einfachste Methode ist, diese umgekehrt proportional zur Kantenlänge zu setzen. Über einen zusätzlichen Proportionalitätsfaktor kann die Federkonstante lokal angepasst werden, so dass zum Beispiel sehr feine Zellen in Wandnähe weniger verformt werden als große Zellen innerhalb des Rechennetzes.

Bei moderaten Verformungen liefern Methoden dieser Art gute Ergebnisse. Bei großen Verformungen und Verdrehungen können extreme Scherungen der Netzzellen auftreten. WUNDERER [110] schlägt deswegen die Einführung zusätzlicher Drehfedern zwischen jeweils zwei Kanten vor, welche diesen Scherungen entgegenwirken. Das Problem wird jedoch nichtlinear, der Aufwand zur Lösung steigt stark an.

Bei den linear-elastischen Modellen wird das Rechennetz als linear-elastisches Kontinuum aufgefasst. Eine Bewegung der Berandung bewirkt eine Deformation des Kontinuums. Diese Deformation wird mit den linear-elastischen Lamé-Gleichungen für isotrope Materialien berechnet:

$$\mu\nabla\nabla\mathbf{u} + (\mu+\lambda)\,\nabla\nabla\mathbf{u}^T = 0. \tag{4.100}$$

Eine Vereinfachung erhält man, indem man $\lambda = 0$ setzt. Dadurch unterbindet man die Querkontraktion, erhält aber immer noch eine hinreichende Beschreibung der Netzdeformation, s. WICK [108]:

$$\mu \nabla \left(\mathbf{u} + \mathbf{u}^T \right) = 0. \tag{4.101}$$

Diese Gleichung wird mit den Techniken aus Kap. 4 diskretisiert und mit einem Standardlösungsverfahren aus 4.5 iterativ gelöst. Bei stark nichtorthogonalen Netzen kann dies sehr aufwändig werden.
Die Lösungsgrößen an den Zellmittelpunkten werden an die Netzknoten interpoliert, was bei stark inhomogenen Netzen problematisch sein kann. Bei homogenen Netzen erhält man gute bis sehr gute Ergebnisse für die Netzdeformation, s. WUNDERER [110]. Die Stärke der Verformung kann lokal über μ eingestellt werden.

Algebraische Methoden
Die oben beschriebenen Methoden zur Netzdeformation wurden implementiert und ausführlich getestet. Die Lösung der zusätzlich auftretenden partiellen Differentialgleichung kann sehr zeitaufwändig werden. Aus diesem Grund wurde auch eine sehr einfache algebraische Methode implementiert, welche eine direkte Berechnung der neuen Positionen der inneren Netzknoten ermöglicht. Hierfür werden für jeden Netzknoten die Distanz zum nächsten bewegten $\mathbf{x}_b$ und die Distanz zum nächsten fixierten Netzknoten $\mathbf{x}_f$ bestimmt und mit diesen Werten ein Wichtungsfaktor α bestimmt:

$$\alpha = \frac{|\mathbf{x}_f|}{|\mathbf{x}_f| + |\mathbf{x}_b|}. \tag{4.102}$$

Die neue Position $\mathbf{x}_n$ des betrachteten inneren Netzknotens wird dann mit

$$\mathbf{x}_n = \alpha \mathbf{x}_a \tag{4.103}$$

bestimmt, wobei $\mathbf{x}_a$ den Verbindungsvektor von alter zu neuer Position des nächsten bewegten Berandungsknoten darstellt.
Die neuen Positionen der Netzknoten können also direkt berechnet werden. Es müssen jedoch die Distanzen $|\mathbf{x}_b|$ und $|\mathbf{x}_f|$ effizient bestimmt werden. Hierzu wurde ein einfacher Suchbaum, s. BENTLEY [17], verwendet. Diese Bestimmung muss nur einmal pro berechneten Zeitschritt erfolgen, wodurch die Berechnung der Netzdeformation an Schnelligkeit die differentiellen Methoden übertrifft. Die in der Regel weniger gute Qualität der deformierten Netze stellte für die im Rahmen dieser Arbeit durchgeführten Simulationen kein Problem dar.

4.6.4 Sequentielles Lösungsverfahren für Fluid-Struktur-Interaktionen

Die sequentiellen Lösungsverfahren für fluidmechanische und strukturmechanische Simulationen können einfach für die Fluid-Struktur-Interaktion miteinander kombiniert werden. Je nachdem wie oft der Datenaustausch zwischen den Fluid- und den Strukturbereichen erfolgt, spricht man von einer schwachen oder starken Kopplung.

Bei der schwachen Kopplung werden die Daten nur zum Ende des betrachteten Zeitschrittes ausgetauscht, anschließend Fluid- und Strukturnetze verformt, notwendige Transformationen durchgeführt und der neue Zeitschritt begonnen.
Bei der starken Kopplung hingegen werden der Datenaustausch, die Netzverformung und die Transformationen mehrere Male pro Zeitschritt durchgeführt. Dies geschieht so lange, bis ein geeignetes Konvergenzkriterium für den Datenaustausch erreicht ist. Zum Beispiel kann überprüft werden, ob sich die am Fluid-Struktur-Interface berechneten Kräfte nicht mehr ändern.

4.7 Erhöhung der Stabilität

Zur Erhöhung der Stabilität und zur Beschleunigung des Lösungsverfahrens werden zum einen Relaxationen, zum anderen Limitierungen der Lösungsgrößen verwendet. Diese Methoden bewähren sich gerade bei praktischen Problemstellungen sehr, und oft ist eine Lösung erst mit ihrer Hilfe möglich.

4.7.1 Explizite Unterrelaxation

Bei der expliziten Unterrelaxation wird die neu erhaltene Lösungsgröße ϕ_n mit der Lösung aus dem vorangegangenen Iterationsschritt ϕ_{n-1} mit einem Unterrelaxationsfaktor α gemischt:

$$\phi_n = \alpha\phi_n + (1-\alpha)\phi_{n-1}. \tag{4.104}$$

Der Unterrelaxationsfaktor α ist zwischen $0.0 < \alpha \leq 1.0$ zu wählen. Diese Unterrelaxation ist beim angewandten SIMPLE-Druckkorrekturverfahren für die neu berechneten Druckkorrekturwerte zwingend notwendig. Auch die ermittelten Geschwindigkeitskorrekturen müssen im entwickelten Verfahren bei komplexen Problemstellungen so relaxiert werden.
Von ZWART [112] wird vorgeschlagen, auch die neu berechneten Gradienten zu unterrelaxieren. Dies wurde im Verfahren ebenfalls implementiert, jedoch konnte hierdurch keine Erhöhung der Lösungsstabilität oder eine Beschleunigung des Verfahrens erhalten werden.

4.7.2 Implizite Unterrelaxation

Die implizite Unterrelaxation wurde von PATANKAR [79] vorgeschlagen und greift direkt in das betrachtete Gleichungssystem ein. Mit einem Unterrelaxationsfaktor α, der zwischen $0.0 < \alpha \leq 1.0$ gewählt wird, erhöht man durch Division die Diagonaldominanz des Gleichungssystems:

$$\frac{A_P}{\alpha}\phi_P^n + \sum_i A_i\phi_i^n = b + \left(\frac{A_P}{\alpha} - A_P\right)\phi_P^{n-1}. \tag{4.105}$$

Bei Konvergenz des iterativen Lösungsprozesses ist die Lösungsgröße ϕ^n zum Iterationsschritt n gleich der Lösungsgröße ϕ^{n-1} zum vorhergegangenen Iterationsschritt $n-1$,

die neu hinzugefügten Terme verschwinden, und man erhält das originäre Gleichungssystem. Für die verwendeten iterativen Lösungsverfahren spielt die Diagonaldominanz eine wichtige Rolle, bzw. macht eine iterative Lösung des Gleichungssystems erst möglich.
Diese Art der Unterrelaxation wurde stets auf die Impulsgleichungen und die Turbulenzgleichungen mit einem typischen Wert des Unterrelaxationsfaktors $0.2 < \alpha \leq 0.5$ angewendet. Die Druckkorrekturgleichungen und die Strukturgleichungen sind als reine Diffusionsgleichungen diskretisiert, somit diagonaldominant und müssen nicht unterrelaxiert werden.

4.7.3 Unterrelaxation Fluid-Struktur-Interface

Bei der Simulation von Fluid-Struktur-Interaktionen mit großen Verformungen kann es zu Instabilitäten des Verfahrens kommen. Diese Instabilitäten sind stark vom Dichte-Verhältnis zwischen Fluid und Struktur, Fluid-Viskosität, Steifigkeit des Materials und Zeitschrittgröße abhängig, s. TUKOVIC [98], WALL [105]. Durch Unterrelaxation der von der Struktur an das Fluid übergebenen Verschiebungen wird das Verfahren auch für diese Problemstellungen stabilisiert. Hierzu werden in der Literatur verschiedene Varianten angegeben. All diesen Methoden ist gemeinsam, dass die Lösung der Struktur- und Fluid-Gleichungen in einer weiteren FSI-Schleife, hier mit Iterationszähler k symbolisiert, erfolgt, bis keine Änderung der übertragenen Verschiebungen mehr auftritt.
In der einfachsten Methode werden die zu übertragenden Verschiebungen $\delta\mathbf{u}^*$ mit einem konstanten Unterrelaxationsfaktor $0,0 < \alpha <= 1,0$ mit Hilfe von aktuellen Verschiebungen und Verschiebungen aus dem alten FSI-Iterationsschritt übergeben:

$$\delta\mathbf{u}_k^* := \alpha\delta\mathbf{u}_k + (1-\alpha)\delta\mathbf{u}_{k-1}^*. \tag{4.106}$$

Bei der Methode nach Aitken, s. z. B. MOK [75], wird der Unterrelaxationsfaktor α adaptiv aus alten Verschiebungswerten und den aktuellen Verschiebungen ermittelt:

$$\alpha_k = 1 - \gamma_k, \tag{4.107}$$

wobei

$$\gamma_k = \gamma_{k-1} + (\gamma_{k-1} - 1)\frac{(\Delta\delta\mathbf{u}_{k-1} - \Delta\delta\mathbf{u}_k)\,\Delta\delta\mathbf{u}_k}{(\Delta\delta\mathbf{u}_{k-1} - \Delta\delta\mathbf{u}_k)^2} \tag{4.108}$$

und die Beziehungen

$$\Delta\delta\mathbf{u}_k = \delta\mathbf{u}_{k-1}^* - \delta\mathbf{u}_k \tag{4.109}$$

und

$$\Delta\delta\mathbf{u}_{k-1} = \delta\mathbf{u}_{k-2}^* - \delta\mathbf{u}_{k-1} \tag{4.110}$$

gelten.
Die zweite Methode fand im Rahmen dieser Arbeit noch keine Anwendung, kann aber eine deutliche Beschleunigung des Lösungsverfahrens ermöglichen.

4.7.4 Limitierungen

Innerhalb des iterativen Lösungsverfahrens können die gesuchten Größen unphysikalische Werte annehmen. Durch Limitierungen nach jedem äußeren Iterationsschritt kann eine Destabilisierung des Verfahrens vermieden werden.

So wird verhindert, dass Drücke, Dichte und Temperatur während des Rechnungsverlaufs kleiner Null werden. Auch die turbulente kinetische Energie und die Dissipation werden auf Null bzw. einen sehr kleinen Wert begrenzt. Die turbulente Viskosität wird je nach Problemstellungen auf ein Vielfaches der laminaren Viskosität begrenzt, wobei typischerweise als Grenze $\mu_{turb} \leq 1000\mu_{lam}$ gesetzt wurde. Hieraus resultiert dann automatisch eine Begrenzung der turbulenten Dissipation nach oben.

4.8 Parallelisierung und Partitionierung

Durch die Partitionierung wird das Rechengebiet in einzelne Bereiche aufgeteilt. Diese Bereiche werden über sogenannte *Interfaces* miteinander verbunden. Die Interfaces können innerhalb eines Rechenprozesses liegen, wie es beispielsweise häufig bei Berücksichtigung von Periodizitäten vorkommt, können aber auch prozessübergreifend eine Verbindung zwischen Rechenbereichen herstellen. Die Topologie der Rechennetze an den Schnittstellen kann unterschiedlich sein, weshalb in einem Vorbereitungsschritt die aneinandergrenzenden Rechennetze verschnitten und damit passend gemacht werden müssen.

4.8.1 Allgemeines Interface

Um einen konservativen Datenaustausch zu gewährleisten und die Flexibilität des Verfahrens zu gewährleisten, wurde im Rahmen dieser Arbeit ein Verschneidungsalgorithmus implementiert, der sowohl sehr schnell als auch robust ist und auch innerhalb der Fluid- und Strukturgebiete als allgemeines nichtpassendes oder *Non-Matching-Interface* verwendet werden kann. Das Verschneiden vollzieht sich dabei in den folgenden Schritten:

- Zunächst werden an den sich berührenden Oberflächen die beliebig polygonalen Oberflächennetze extrahiert.
- Die extrahierten Oberflächennetze werden nun trianguliert.
- Ein Verschneidungs-*Master* wird festgelegt und das zugehörige Oberflächennetz extrudiert. Die Extrusion erfolgt in Richtung der über die angrenzenden Flächenmaßzahlen gewichteten Knoten-Normalvektoren. Die Extrusionslänge wird als Teil der kleinsten Kantenlänge des Master-Netzes bestimmt. Der Master stellt somit wieder ein Volumennetz mit einer Zellschicht dar. Die extrudierten Volumenzellen sind Prismen mit Grundfläche eines Dreiecks und werden als *Master-Cells* bezeichnet. Der Verschneidungspartner wird als *Slave* bezeichnet und wird durch ein Oberflächennetz repräsentiert. Die Teilflächen des *Slave* werden als *Slave-Faces* bezeichnet.

- Für jede entstandene Zelle des extrudierten Master-Netzes werden zunächst über einfache *Bounding-Box-Kriterien* mögliche Verschneidungsnachbarn gesucht.
- Nun werden Verschneidungspunkte von Zellen des Master-Netzes, welche aus extrudierten Dreiecksflächen bestehen, mit den Dreicksflächen des Oberflächen-Slave-Netzes bestimmt. Dies geschieht in den folgenden Schritten:
  - Liegt der Punkt eines *Slave-Faces* innerhalb oder auf der betrachteten *Master-Cell*, ist dieser ein Verschneidungspunkt. Liegen alle Punkte eines *Slave-Faces* innerhalb oder auf der *Master-Cell*, so ist die aktuelle Verschneideoperation beendet.
  - Sonst werden nun die Kanten des *Slave-Faces* mit den Seitenflächen der *Master-Cell* verschnitten.
  - Nun werden noch Verschneidungspunkte von Extrusion-Kanten der *Master-Cell* mit dem betrachteten *Slave-Face* gesucht.

Durch die durchgeführten Verschneideoperationen wurden Verschneidungspunkte gefunden, welche die neuen Teilflächen des Verschneidungsnetzes bilden und als *Intersection-Faces* bezeichnet werden. Ein *Intersection-Face* kann durch die Triangulation der betrachteten Oberflächennetze aus minimal drei und maximal sechs Punkten aufgebaut sein. Die Verschneidungspunkte müssen nach den Verschneideoperationen noch in eine umlaufende Reihenfolge gebracht werden. Die mit den Verschneidungspunkten gebildeten *Intersection-Faces* sind in jedem Fall konvex.
Die Verschneideoperationen selbst bestehen aus elementaren Gerade-Ebene- und Gerade-Gerade-Verschneideoperationen und Tests, ob Punkte innerhalb oder außerhalb der extrudierten *Master-Cells* liegen. Die zugehörigen notwendigen geometrischen Genauigkeiten werden lokal für die einzelnen Verschneidungspartner über einen Teil der kleinsten auftretenden Kantenlänge bestimmt.
Durch die der Verschneidung vorangegangene Triangulierung ist der Verschneidealgorithmus nicht auf Verschneidungsnetze mit konvexen Elementen beschränkt.
Zu kleine Verschneidungsflächen, typischerweise kleiner als 0.1 Prozent des Flächeninhalts des zugehörigen Master-Face, oder Verschneidungsflächen mit sehr hohen Streckungsverhältnissen l, typischerweise $l > 1000$, werden aussortiert.
Nach den Verschneideoperationen ist bekannt, welche *Intersection-Faces* zu welchem *Master-Face* und zu welchem *Slave-Face* gehören, und das Verschneidungsnetz könnte idealerweise zurück in die Volumennetze eingefügt werden. Bei räumlich gekrümmten Verschneidungsnetzen wird dies aufgrund dreidimensionaler Effekte zunehmend unmöglich, bzw. durch Vorbereitungsoperationen zu aufwändig; dieser Schritt wird deswegen nicht vorgenommen. Das Verfahren rechnet deswegen nur intern mit diesem Verschneidungsnetz, welches als Grundlage der impliziten Diskretisierung über ein Interface dient. Das ursprüngliche Volumennetz bleibt erhalten. Hierdurch bleibt das Verfahren sehr schnell, was gerade bei instationären Rechnungen wichtig ist, denn oftmals müssen die Verschneidungsnetze der *Non-Matching-Interfaces* in jedem Zeitschritt neu bestimmt werden.

Weiterhin ist anzumerken, dass sich bei dreidimensional gekrümmten Interfaces die Fläche des Verschneidungsnetzes betragsmäßig von den Oberflächennetzen des Masters und des Slaves unterscheiden kann. Das bedeutet, dass ein ursprüngliches

Master- oder Slave-Face betragsmäßig eine andere Fläche besitzt als die Summe der Flächen der zugehörigen Verschneidungs-Faces. Hierdurch wäre keine geschlossene Finite-Volumen-Zelle mehr vorhanden und die konservative Diskretisierung eines Oberflächenintegrals ist nicht mehr möglich. Aus diesem Grund werden die Flächeninhalte der *Intersection-Faces* skaliert. Nun besteht aber das Problem, dass Master und Slave flächenmäßig unterschiedlich große Verschneidungsnetze sehen und die berechneten Massenströme über das allgemeine Interface unterschiedlich würden und hierdurch wiederum die Konservativität des Verfahrens verletzt würde. Deswegen werden die Massenströme nur vom Master berechnet und dem Slave zur Verfügung gestellt, wodurch dann auch Massenkonservativität gewährleistet ist.

Durch das allgemeine nichtpassende Interface wird dem Lösungsverfahren ein passendes Interface suggeriert, und die Diskretisierungsverfahren ändern sich über diese Interfaces nicht. Die Diskretisierung bleibt voll implizit.

4.8.2 Partitionierung und Parallelisierung

Ein großer Nachteil des alten blockstrukturierten Verfahrens *ns3d* war, dass es bei einer stark inhomogenen Größe der einzelnen Rechenblöcke und der Verteilung dieser auf einzelne Rechenprozesse kaum möglich war, eine hohe und damit effiziente Auslastung dieser Prozesse zu erreichen. Dieser Nachteil entfällt bei dem vorgestellten Verfahren fast vollständig, da das betrachtete Rechengebiet nun von der frei zur Verfügung stehenden Software-Bibliothek METIS optimal aufgeteilt, bzw. partioniert wird. Lediglich Struktur- und Fluidgebiete und stehende sowie rotierende Bereiche müssen im implementierten Verfahren noch auf unterschiedlichen Prozessen behandelt werden.
Die durch diese Partitionierung gebildeten Blöcke des Rechengebiets müssen auf Daten der angrenzenden Rechenzellen in Nachbarblöcken zugreifen können. Diese Daten werden in sog. *Shadow*-Zellen gespeichert. Durch das implementierte allgemeine Netz-Interface konnte die hinterlegte Datenstruktur im Vergleich zum Vorgängerverfahren wesentlich vereinfacht werden.
Der Datentransfer über ein Interface zwischen zwei unterschiedlichen Prozessen wird durch die frei verfügbare Software-Bibliothek MPI realisiert.

4.8.3 Periodizitäten

Periodische Interfaces unterscheiden sich hinsichtlich der Datenhaltung nicht von gewöhnlichen Interfaces. Sie können passend und nichtpassend sein und werden in translatorische und rotatorische Periodizitäten unterteilt.
Bei translatorischen Interfaces erfolgt keine Modifikation der auszutauschenden Daten. Lediglich beim Datenaustausch über rotationsperiodische Interfaces muss beachtet werden, dass skalare Größen ϕ zwar ohne Modifikation punktweise übergeben werden können, vektorielle Größen $\boldsymbol{\phi}$ und tensorielle Größen $\boldsymbol{\Phi}$ jedoch mit einer Drehmatrix D

modifiziert werden müssen:

$$\phi_N = \phi \tag{4.111}$$
$$\boldsymbol{\phi}_N = \mathbf{D}\boldsymbol{\phi} \tag{4.112}$$
$$\boldsymbol{\Phi}_N = D\boldsymbol{\Phi}D^{-1}. \tag{4.113}$$

Rotationsperiodische Interfaces sind im implementierten Verfahren auf Rotationen um die z-Achse beschränkt, und die Drehmatrix lautet für eine Periodizität mit dem Winkel γ deswegen:

$$\mathbf{D} = \begin{pmatrix} cos\gamma & -sin\gamma & 0 \\ sin\gamma & cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.114}$$

Eine Erweiterung für Periodizitäten um eine allgemeine Rotationsachse ist einfach möglich, s. HERMANN [50].

Kapitel 5

Verfahrensentwicklung

Das vorgestellte Verfahren stellt eine grundlegende Neuentwicklung dar, welche auf den Erfahrungen der Verfahrensentwicklung von SKODA [93], EINZINGER [34] und BOGNER [20] aufbaut, die sinnvollen Kernideen konsequent wiederverwendet bzw. erweitert sowie einen einfachen neuen, modularen Ansatz zur Lösung kontinuumsmechanischer Problemstellungen aufzeigt und anwendet.

Das physikalische und mathematische Modell, die angewendeten Diskretisierungstechniken und die Lösungmethoden wurden in den bisherigen Kapiteln ausführlich beschrieben. In diesem Kapitel werden auftretende Problemstellungen und Zielsetzungen beim Softwareentwurf für ein anwendbares Programmpaket diskutiert und die reale Umsetzung des Programmkonzepts vorgestellt.

5.1 Anforderungen an das Simulationswerkzeug

Nach BOGNER [20] ergeben sich für ein Simulationswerkzeug sowohl im universitären als auch im industriellen Umfeld drei wesentliche Anwenderrollen:

- die des Simulationsingenieurs,
- die des Forschers und
- die des Programmierers.

Der Letztgenannte ist zwar kein Anwender im herkömmlichen Sinne, stellt aber ebenfalls Anforderungen an die Software. Diesen Sachverhalt veranschaulicht auch die folgende Abb. 5.1.

Der Simulationsingenieur ist primär an den hinterlegten physikalischen Modellen, der Programmstabilität und Geschwindigkeit interessiert. Für den Programmierer hingegen stehen die Erweiterbarkeit und Wartbarkeit des Programmcodes im Vordergrund. Der Forscher muss gleichzeitig neue Modelle entwickeln, implementieren, validieren sowie anwenden und stellt also eine Mischung aus den beiden vorher genannten Anwendergruppen dar. Dieser Sachverhalt wird von BOGNER [20] in

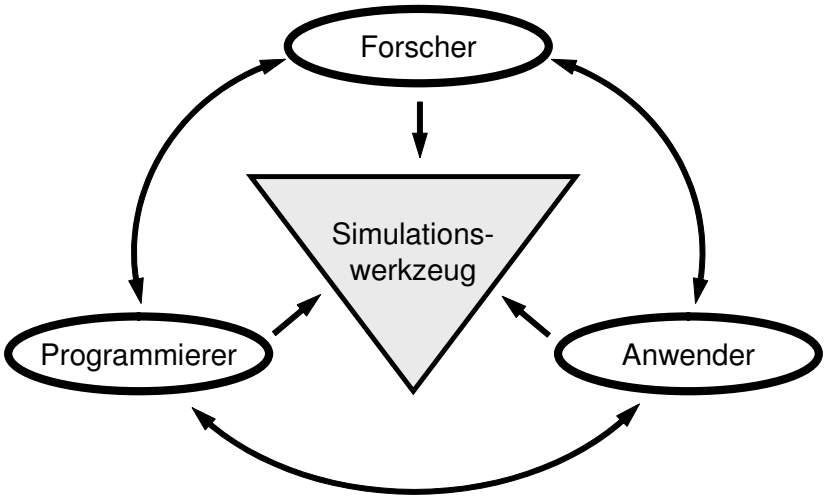

Abbildung 5.1: Anwenderrollen bei Simulationsaufgaben.

der folgenden Abb. 5.2 veranschaulicht. Sie besagt auch, dass die Abgrenzung dieser Anwendergruppen oft nicht strikt möglich ist.
Ebenfalls wird anhand dieser Graphik die Möglichkeit erkannt, dass Angehörige einer Gruppe durch Rollentausch ganz neue Anforderungen an die Software stellen können. Dieser Sachverhalt ist typisch für den Forscher, welcher oftmals nach der Modelldefinition ein Modell implementieren, validieren und anwenden muss bzw. für andere Forscher oder Anwender anwendbar machen muss.

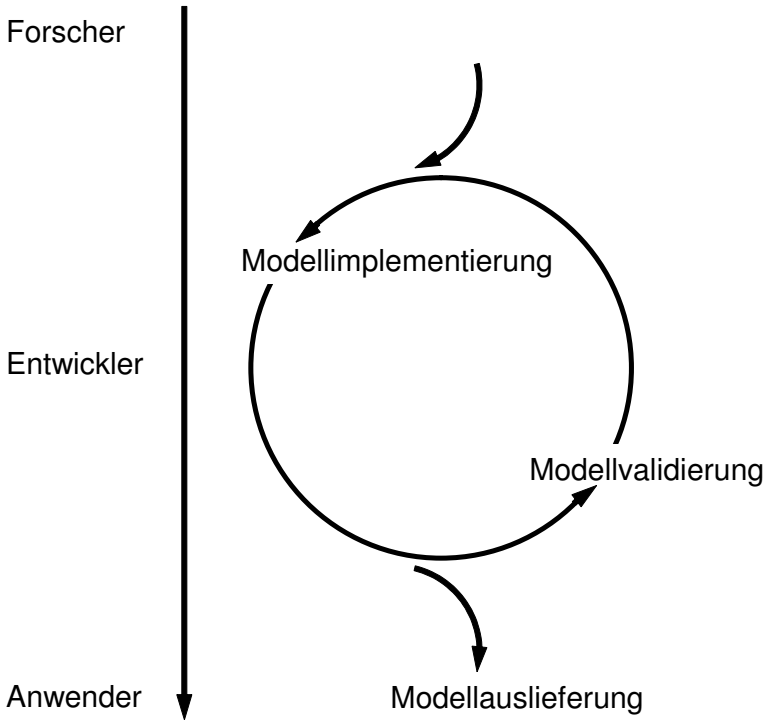

Abbildung 5.2: Rollentausch bei Anwendung des Simulationswerkzeugs.

Die Anwendergruppen stehen somit in einem ständigen Kommunikations- und Interaktionsprozess. Das Simulationswerkzeug muss ständig angepasst oder erweitert

werden.

Die Anforderungen an ein verwendetes, numerisches Simulationswerkzeug können somit nur durch einen nichtstarren und hochdynamischen Software-Entwurf und eine ebenso flexible Software-Umsetzung befriedigt werden. Die Ziele für die Entwicklung eines derartigen, neuen Simulationswerkzeuges können vor dem Hintergrund der bisherigen Ausführungen nun klar definiert werden.

5.2 Entwicklungsziele *solver3d*

Die Implementierung und programmiertechnische Umsetzung eines numerischen Simulationswerkzeugs stellt einen enormen programmiertechnischen und -organisatorischen Aufwand dar. Eine anwendbare und leistungsfähige Software kann somit nur entstehen, wenn Anforderungen und Ziele für das numerische Simulationswerkzeug klar definiert werden:

- Allen anderen Zielen übergeordnet muss das neue Verfahren physikalisch korrekte und sinnvolle Ergebnisse liefern.

- Das Verfahren muss für neue Problemstellungen wiederverwendbar sein, deswegen muss das Verfahrens einfach und effizient erweiterbar und wartbar sein.

- Die einfache und effiziente Erweiterbarkeit und Wartbarkeit setzt eine hohe Modularität des Verfahrens voraus. Die einzelnen Module müssen in einfacher Weise unabhängig voneinander getestet und angewendet werden können.

- Durch diese hohe Modularität des Verfahrens kann dann sichergestellt werden, dass einzelne Module schnell ausgetauscht, angepasst oder erweitert werden können. Durch diesen Punkt wird eine schnelle Adaption des Verfahrens an neue Problemstellungen erreicht, wodurch ein schnelles Experimentieren jederzeit möglich ist.

- Das Verfahren sollte einfach im Kontext mit anderen bestehenden Programmsystemen verwendet werden können. Hierzu ist es notwendig, Schnittstellen an verfügbare Netzgeneratoren sowie an Programme zum Pre- und Postprocessing oder andere Simulationsverfahren bereitzustellen. Nur dadurch können Ergebnisse eines Simulationswerkzeugs einfach und mit notwendiger Sicherheit verglichen, verifiziert und validiert werden.

Für *physikalisch korrekte und sinnvolle Ergebnisse* ist ein mathematisch korrektes und sinnvolles Modell notwendig. Die im Simulationswerkzeug umgesetzte Numerik zur Beschreibung dieser Modelle muss konsistent und genau sein. Dass das hier vorgestellte Verfahren physikalisch sinnvolle Ergebnisse liefern kann, wird in Kap. 6 gezeigt. Eine wesentliche Verbesserung zu den Vorgängerverfahren wurde durch die überarbeitete Diskretisierungstechnik von Zeit-, Konvektions- und Diffusionstermen erreicht.

Das *Programm ist einfach wartbar* aufgrund seiner übersichtlichen und klar gegliederten Programmstruktur. Überflüssige oder veraltete Routinen aus den Vorgängerverfahren wurden entfernt.

Die *hohe Modularität des Verfahrens* wird primär erreicht durch die konsequente Anwendung der Programmiersprache *C++*. Mit dieser Programmiersprache ist es möglich, höchste Komplexität zu beherrschen, indem man klar hierarchisch gliedert und organisiert, s. STROUSTRUP [96]. Die Vorgängerverfahren sind noch in originärem *C* programmiert und verwenden noch kaum höhere Datenstrukturen. Hierdurch wurde die Software zunehmend unübersichtlich und nur mit enormem zeitlichem Aufwand wartbar.

Das vorliegende Verfahren ist *einfach anpassbar*. Es verwendet konsequent objektorientierte Elemente wie Namensbereiche, Strukturen und Klassen, wodurch das Verfahren übersichtlich strukturiert wird. Weiterhin ist das Verfahren unterteilt in verschiedene Bibliotheken, die effizient miteinander interagieren. So existieren beispielsweise Bibliotheken zur Lösung linearer Gleichungssysteme, s. Kap. 4.5, oder zur geometrischen Bearbeitung von Rechennetzen, s. Kap. 4.8.1, welche auch in einem anderen Kontext als mit dem Lösungsverfahren *solver3d* verwendet werden können.

Eine einfache Verwendbarkeit im *Kontext mit anderen Programmsystemen* wird durch die Verwendung von offenen Dateitypen erreicht. Ferner ist es einfach möglich, Eingabedateien für eine Simulation in unterschiedlichste Dateitypen zu konvertieren, um z. B. Vergleichsrechnungen mit dem Simulationsprogramm ANSYS [2] oder OPENFOAM [6] zu ermöglichen.

5.3 Komponenten numerischer Anwendungen

BOGNER [20] adaptiert in seinem Entwurf für sein numerisches Werkzeug das Schichtenmodell von ACOSTA [7]. Dieses Schichtenmodell für numerische Software ist ähnlich einem Eisberg aufgebaut, welcher zum größten Teil unter der Wasseroberfläche liegt, s. Abb. 5.3. Nur die sichtbare Spitze ist für den CFD-Anwender zu sehen.

Die unterste Schicht des ACOSTA [7]-Modells bildet die *Kommunikations-Schicht*, welche die funktionalen Komponenten zur Parallelisierung und zur Datenstruktur enthält.

Die darüberliegende *Algebra-Schicht* enthält Algorithmen zur Diskretisierung von Gleichungen. Kenntnisse über Programmierungs-Details der darunterliegenden Kommunikationsschicht sind für die Algebra-Schicht nicht notwendig, es werden lediglich geeignete Schnittstellen der Kommunikationsschicht verwendet.

Nun folgt über dieser Algebra-Schicht die *Löser-Schicht* (Solver), welche Lösungsalgorithmen für die diskreten Gleichungssysteme enthält. Diese Löser-Schicht arbeitet wiederum ohne Kenntnisse der Algebra-Schicht und die Löser-Schicht greift über bereitgestellte Schnittstellen der Algebra-Schicht auf selbige zu.

Die oberste *Benutzer-Schicht* (User) stellt lediglich die zum User offenen Schnittstellen zur Ansteuerung der Software zur Verfügung. Sie ist nach außen sichtbar und repräsentiert die numerische Software als Ganzes.

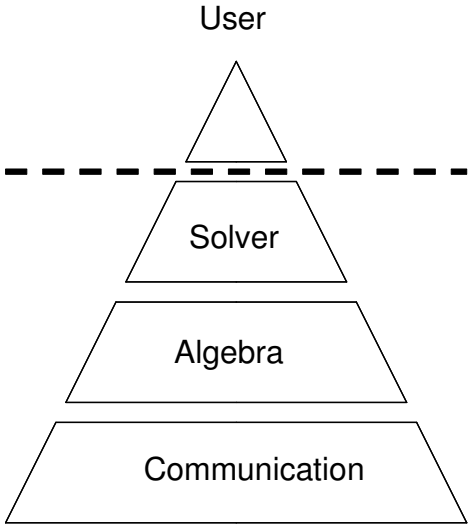

Abbildung 5.3: Schichtenmodell nach Acosta.

Das ACOSTA [7]-Modell ist in seiner Grundform stark hierarchisch aufgebaut, und die Interaktion zwischen den Schichten kann nur in vertikaler Richtung erfolgen. Dieses grundlegende Prinzip ist sowohl im Entwurf des CFD-Tools von BOGNER [20] als auch im hier vorgestellten Verfahren abgebildet. BOGNER [20] kritisiert aber am originären Schichtenmodell, dass keine horizontalen Schnittstellen für die funktionalen Komponenten existieren, wodurch eine bessere Interaktion zwischen Benutzer und Software möglich wäre und die Nutzbarkeit des Werkzeugs erheblich erweitert würde. Für das neu entwickelte Verfahren *solver3d* wurde nach BOGNER [20] eine Anpassung des ACOSTA [7]-Modell vorgenommen.

Die oberste Schicht bildet die *Anwenderschicht* und bietet für den Standardnutzer Ansteuerungsmöglichkeiten für das Verfahren *solver3d*. Der Anwender konfiguriert lediglich über Parameterdateien das Verfahren für seine jeweilige Problemstellung. Er benötigt keine tiefer gehenden Kenntnisse über die zugrunde liegende physikalische und numerische Modellierung.

Darauf folgt die Schicht des *Modellierungsexperten*. Dieser stellt sich bei der Anwendung des Verfahrens neue Konfigurationen zusammen oder setzt neue physikalische Modelle in Numerik um. Er benötigt keine Kenntnisse über die zugrundeliegenden Lösungsverfahren für lineare Gleichungssysteme oder diskrete Geometriebearbeitung oder die zugrundeliegenden Datenstruktur.

Über diese Kenntnisse verfügt der *Numerikexperte*, welcher die Module zur Lösung linearer Gleichungssysteme oder zur diskreten Geometriebearbeitung anpasst, erweitert oder verbessert.

Die darunter liegende Schicht ist für den *Programmierexperten* bestimmt, welcher neue externe Module einbindet oder anpasst. Er benötigt keinerlei Kenntnisse über die darüberliegenden Schichten oder die spätere Anwendung.

Durch diese vier Schichten ergibt sich auch die Grobunterteilung des neu entwickelten numerischen Verfahrens. Die obersten beiden Schichten werden durch den Kerncode des *solver3d* gebildet, welcher als reines Anwendungsobjekt fungieren oder eben durch den Modellierungsexperten angepasst und erweitert werden kann.
Die dritte Schicht wird repräsentiert durch entwickelte Bibliotheken zur Lösung linearer Gleichungssysteme und zur Bearbeitung von Geometrie und Rechennetzen.
Die unterste Schicht beinhaltet alle externen eingebundenen Bibliotheken wie die Partitionierungs-Software METIS [1] oder die Parallelisierungs-Software MPI [4], s. Kap 4.8.
Dieses neue und für das entwickelte Verfahren angepasste Schichtenmodell ist in Abb. 5.4 dargestellt.

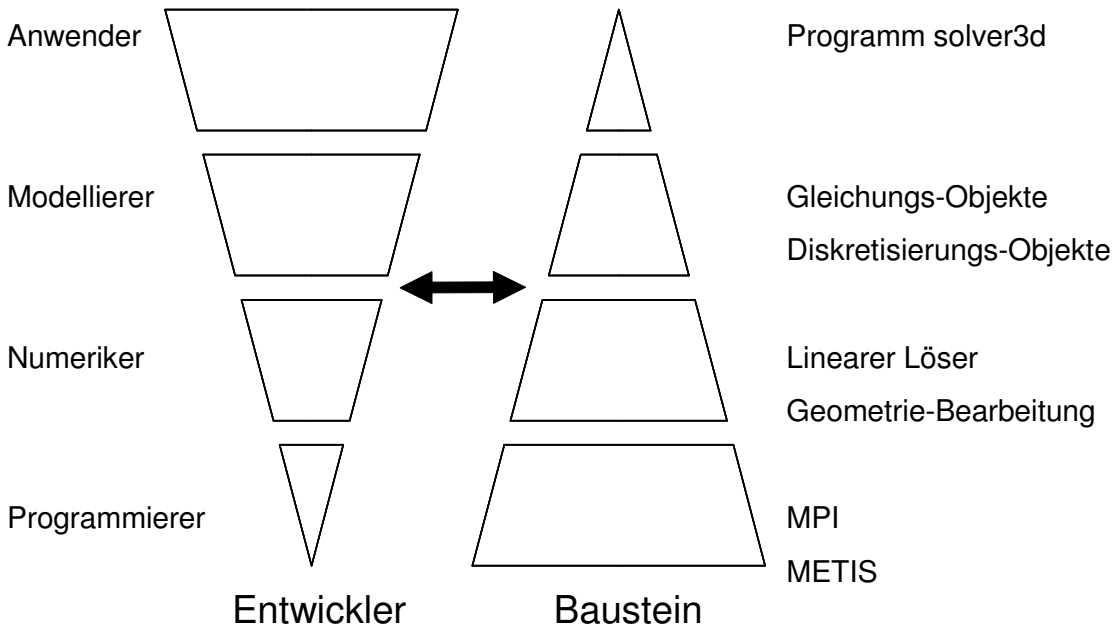

Abbildung 5.4: Das für das neue Verfahren angepasste Acosta-Schichtenmodell.

Man erkennt an den obigen Ausführungen, dass eine strikte Trennung der Anwendergruppen, wie von BOGNER [20] vorgeschlagen, nicht immer möglich ist. Gerade zwischen den ersten beiden Schichten muss oftmals durch neue Anforderungen an die Software eine starke Interaktion erfolgen können. Die beiden untersten Schichten sind in der Regel jedoch für die Anwender der oberen beiden Schichten nur in sehr seltenen Fällen von Interesse.

Eine wesentliche Grundlage zur Modellierung komplexer Mehrfeldprobleme ist eine hohe Flexibilität des Verfahrens. Diese Flexibilität wird durch die Definition von Konfigurationen erreicht, deren Grundkonzept und Aufbau im Folgenden beschrieben werden.

5.4 Flexibilität durch Konfigurationen

Eine hohe Mächtigkeit erreicht das neue Verfahren durch die Ermöglichung der Zusammenstellung von Konfigurationen. Durch diese Konfigurationen ist das Verfahren sehr einfach auf unterschiedlichste Problemstellungen anpassbar und erweiterbar und erreicht damit die angestrebte hohe Flexibilität.

Die Konfigurationen werden somit individuell gestaltet. Festgesetzt wird lediglich, dass die Lösung innerhalb einer Zeitschleife geschieht, welche wiederum eine äußere Iterationsschleife umschließt, s. Kap. 4.6. Innerhalb der äußeren Iterationsschleife können dann z. B. die Nichtlinearitäten einer Problemstellung oder nachgezogene Korrekturen, s. Kap. Kap. 4, berücksichtigt werden. Durch diese Festsetzung des Lösungsablaufs entstehen keinerlei Einschränkungen hinsichtlich zu behandelnder Problemstellungen.

Die Konfiguration wird in einer Initialisierungsphase des Programms festgelegt. Zunächst wird die Netzdatei geladen. Es folgt die Randbedingungsdatei, welche zusätzlich zu den spezifizierten Randbedingungen die Informationen enthält, zu welcher *Domain* das betreffende Rechennetz gehört. Außerdem wird in dieser Datei die Kinematik des Rechengebietes festgelegt. Eine Domain könnte z. B. einen zusammengehörigen Struktur- oder Fluidbereich darstellen. Durch das Einlesen einer Parameterdatei pro Domain wird dann festgelegt, welche Gleichungen in diesem Rechengebiet gelöst werden sollen.

Mit Hilfe dieser Informationen über die zu lösenden Gleichungen wird nun der pro Domain definierte Equationpool befüllt. Dieser Pool stellt einen einfachen Container dar, welcher im Wesentlichen einen $std :: vector < equationBase* >$ enthält, s. z. B. STROUSTRUP [96].
Die $equationBase*$ repräsentiert dabei ein Objekt einer Basisklasse für alle während der Lösung durchzuführenden Operationen bzw. zu lösenden Gleichungen. Diese Basisklasse enthält rein virtuelle Funktionen für

- die *Initialisierung*, welche für die Gleichung spezifische Parameter festsetzt,
- das *Preprocessing*, welches jeweils zu Beginn eines Zeitschritts aufgerufen wird,
- das *Lösen*, welches innerhalb der äußeren Iterationsschleife aufgerufen wird,
- das *Postprocessing*, welches jeweils zum Ende eines Zeitschritts aufgerufen wird,
- den *Output*, mit dem der aktuelle Lösungsstand der Gleichung angezeigt wird und
- den *State*, mit dem der Status der Gleichung abgefragt und mit dessen Hilfe entschieden wird, ob der Iterationsprozess abgebrochen oder fortgesetzt werden muss.

In der *Initialisierung* eines Gleichungs-Objektes werden beispielsweise Unterrelaxationsfaktoren für die betreffende Lösungsgröße gelesen und gesetzt.

Ein konkretes Beispiel für die Verwendung für das *Preprocessing* einer Gleichung zu Beginn eines Zeitschritts ist die Akkumulation des Verschiebungsvektors und des Spannungstensors im Falle der Strukturgleichungen bei einer inkrementellen Lösung, s. Abschnitt 4.6.

Im Schritt *Lösen* erfolgt das eigentliche Update der betrachteten Lösungsvariable.

Ein konkretes Beispiel für das *Postprocessing* einer Gleichung am Ende eines Zeitschritts ist die Berechnung des aktuellen Spannungstensorinkrements im Falle der Strukturgleichungen, welches während der inkrementellen Lösung nicht benötigt wird.

Diese virtuellen Funktionen werden in den abgeleiteten konkreten Gleichungs-Objekten - falls benötigt - ausimplementiert, und der *Equationpool* stellt zentrale Abarbeitungsroutinen für diese Funktionalitäten zur Verfügung. Der Lösungsablauf mit Hilfe der vorgestellten Programmstrukturen wird in Abb. 5.5 dargestellt.

5.4.1 Gleichungs-Objekte

Die Gleichungs-Objekte stellen die zentralen *Operationsbausteine* des Lösungsverfahrens dar und existieren in unterschiedlichen Ausprägungen, welche jedoch alle von einer gemeinsamen Basisklasse *equationBase* abgeleitet werden. Durch diese Basisklasse werden zentrale Zugriffspunkte und Schnittstellen bereitsgestellt. Gleichungs-Objekte werden einer Domain zugeordnet. Eine Domain beschreibt ein Kontinuum einer physikalischen Ausprägung. Beispielsweise unterscheidet man Fluid- und Struktur-Domain. Fluid-spezifische Gleichungen können natürlich nur der Fluid-Domain und dem zugehörigen Equationpool zugeordnet werden. Gleiches gilt für Struktur-spezifische Gleichungen.

Im Folgenden werden vier unterschiedliche Beispiele für Gleichungstypen näher beschrieben, durch welche eine sehr universelle Anwendbarkeit des Verfahrens ermöglicht wird. Eine weiterreichende, sinnvolle und mögliche Ableitungshierarchie der Gleichungs-Objekte ist in Abb. 5.6 dargestellt.

Durch ein Objekt der Gleichungs-Klasse *EquationOfState* wird eine Zustandsgleichung repräsentiert. Dieses einfache Gleichungs-Objekt enthält in der Lösungsfunktion die Ausimplementierung des angewendeten Zustandsgesetzes, z. B. des allgemeinen Gasgesetzes, s. Anhang A. Als zusätzlicher Parameter wird ein expliziter Unterrelaxationsfaktor zur Verfügung gestellt, welcher in der Parameterdatei festgelegt wird, und den Lösungsprozess bei komplexen Problemstellungen stabilisiert.

Mit einem Objekt der Gleichungs-Klasse *PureDiffusionEquation* wird eine reine Diffusionsgleichung repräsentiert, welche z. B. zur Lösung von Wärmeleitungsproblemen in Strukturen, s. Kap. 3, oder zur Lösung der Druck- oder Druckkorrekturgleichungen in sequentiellen Lösungsverfahren für strömungstechnische Anwendungen verwendet wird. Es wird ein impliziter Unterrelaxationsfaktor zur Verfügung gestellt, welcher in der Parameterdatei festgelegt wird und der der Stabilisierung des Lösungsprozesses

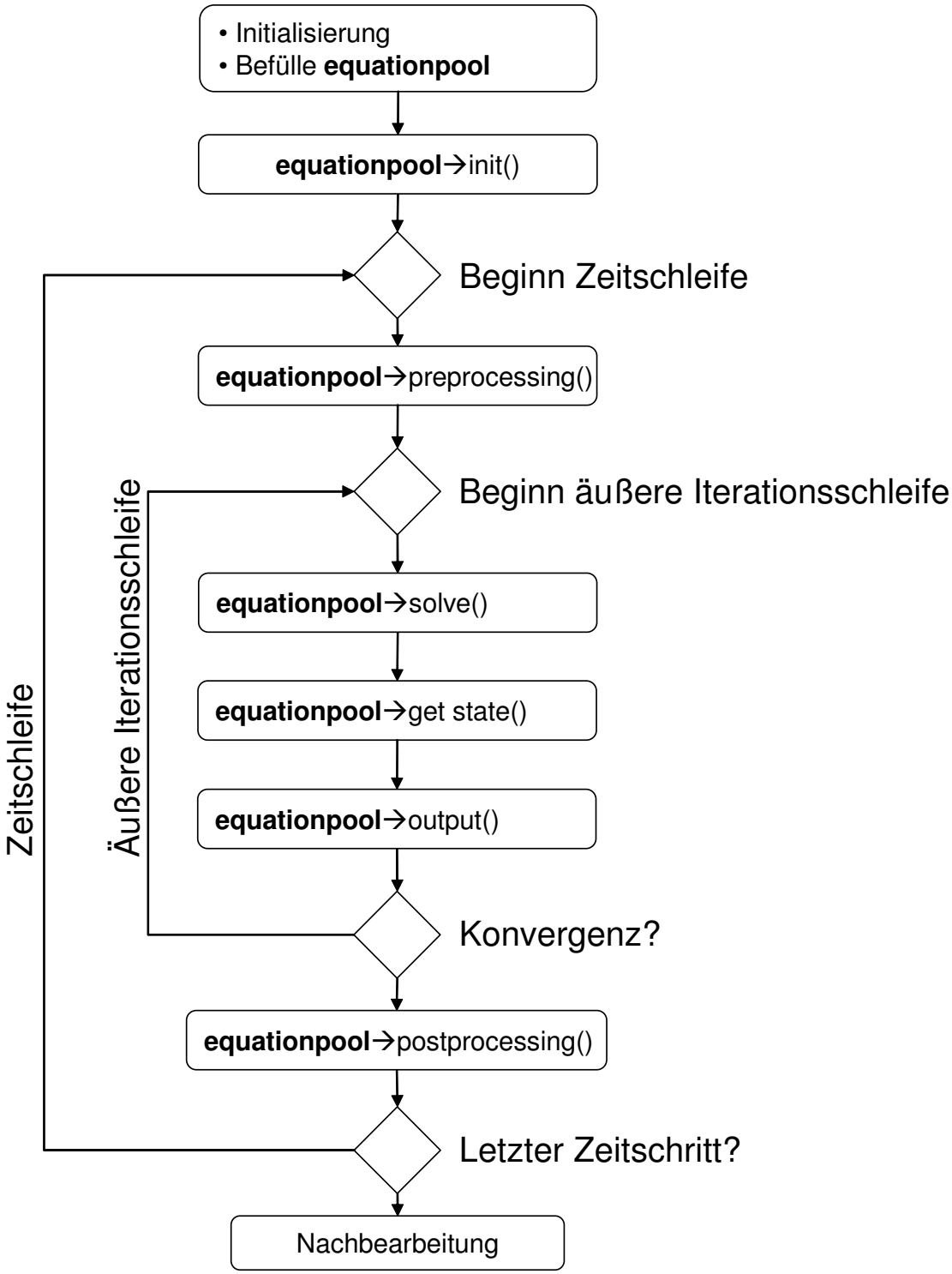

Abbildung 5.5: Sequentielles Lösungsverfahren im objektorientierten Kontext.

dient. Ferner steht dem konkreten Gleichungs-Objekt ein Diskretisierungsobjekt zur Verfügung, welches eine Diskretisierung des Diffusionsterms ermöglicht. Im Falle einer Finite-Volumen-Diskretisierung könnte über die Parameterdatei die konkret angewendete nichtorthogonale Korrektur oder ein Relaxationsfaktor für die Skewness-Correction, s. Kap. 4, festgelegt werden. Im Falle einer instationären Berechnung ist zusätzlich das zeitliche Diskretisierungsschema festzulegen. Zur Lösung der implizit

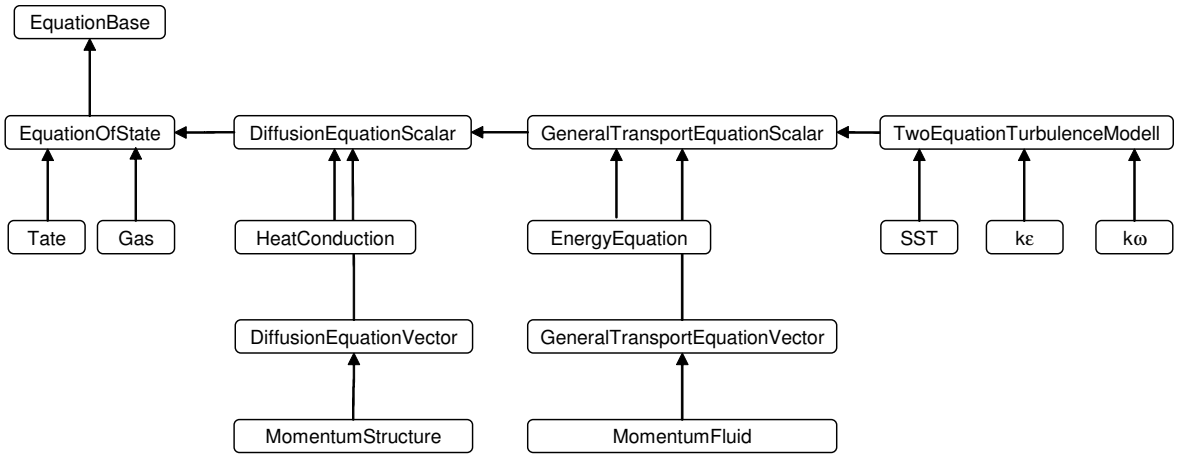

Abbildung 5.6: Mögliche Ableitungshierarchie für Gleichungs-Objekte.

diskretisierten Gleichungs-Objekte wird weiterhin ein Objekt der Klasse LinearEquationSystemSolver zur Verfügung gestellt. Die konkrete Ausprägung dieses Objekts, ob z. B. eine ILU oder ein AMG, s. Kap. 4.5, angewendet werden soll, geschieht wiederum über die Parameterdatei.

Mit einem Objekt der Gleichungs-Klasse *GeneralTransportEquation* wird eine allgemeine Transportgleichung repräsentiert. Sie enthält prinzipiell alle Eigenschaften und Funktionalitäten der PureDiffusionEquation, wird aber ergänzt durch Einstellungsmöglichkeiten zur Diskretisierung des konvektiven Terms, s. Kap. 4.

Eine interessante Gleichungs-Klasse stellt das *FSI-Interface* dar, welches sowohl dem EquationPool für die zu untersuchende Fluid-Domain als auch der zu untersuchenden Struktur-Domain hinzugefügt wird. In der Initialisierungsfunktion für diese Gleichung werden zunächst Verbindungen zwischen Fluid- und Strukturbereichen hergestellt. Für die Strukturbereiche müssen Verbindungen von flächenzentrierten Fluid-Berandungen, für die Fluidbereiche Verbindungen von Netzknoten-Koordinaten von Strukturrändern hergestellt werden. In den ausimplementierten Lösungsfunktionen werden dann die Kräfte der Strömung an die Strukturen übergeben, bzw. die Geschwindigkeiten der Strukturen an die Strömung und die Netzdeformationen übergeben, s. Abschnitt 4.6.

5.4.2 Diskretisierungs-Objekte

Bei der Diskretisierung der zu lösenden Gleichungen hilft eine Instanz der Klasse *Discretizer*, welcher alle dazu notwendigen Funktionalitäten enthält. Hierzu zählen Methoden zur Diskretisierung der Zeit-, Konvektions-, Diffusions- und Quellterme sowie zur geeigneten Einarbeitung von Randbedingungen.

5.4.3 Löser-Objekte

Zur Lösung der durch die Diskretisierung entstehenden linearen Gleichungssysteme werden den betreffenden Gleichungen Löser-Objekte zur Verfügung gestellt. Diese

Löser-Objekte greifen dann auf Funktionalitäten einer eingebundenen Bibliothek zur Lösung linearer Gleichungssysteme zu, welche für unterschiedliche Anwendungen verschiedene Gleichungslöser zur Verfügung stellt, s. Abschnitt 4.5.

Man erkennt, dass schon durch die Definition dieser einfachen Gleichungs-Objekte eine einfache Anpassbarkeit und sehr hohe Flexibilität des Verfahrens erreicht wird. Die programmiertechnische Umsetzung wurde bewusst sehr einfach gehalten, um ein schnelles Einarbeiten in die bestehende Programmstruktur zu ermöglichen. Auf Techniken wie Template-Programmierung oder generisches Programmieren, s. BOGNER [20] und ALEXANDRESCU [9], wurde bewusst im Rahmen dieser Arbeit verzichtet. Es ist jedoch problemlos möglich, diese Techniken in weitergehenden Arbeiten für dieses Verfahren sinnvoll anzuwenden. Durch die Festlegung der Konfigurationen können mehrere Entwickler unabhängig voneinander programmieren, experimentieren und anwenden, ohne den Kern des Lösers, wie Objekte zur Datenhaltung, zur Geometrieaufbereitung, zur Diskretisierung oder zur Lösung linearer Gleichungssysteme modifizieren zu müssen.

Weiterhin erkennt man, dass bei diesem Konzept keinerlei Einschränkung hinsichtlich der verwendeten Diskretisierungstechnik gemacht wurde. Ein Gleichungs-Objekt, welches im Lösungsprozess aufgerufen wird und durch die Bereitstellung der Schnittstellen wie Initialisierung, Preprocessing und Postprocessing effizient angesprochen werden kann, kann seinerseits in der Solve-Methode externe Bibliotheken aufrufen und mit den im Verfahren hinterlegten Modulen koppeln. Beispielsweise wurde im Verfahren ein Gleichungs-Objekt entwickelt, welches die von EINZINGER [34] entwickelte Finite-Elemente-Bibliothek ansteuert. Ferner konnte für ein laufendes Forschungsprojekt, s. ESTORFF [35] , ein externes Akustik-Modul erfolgreich in den Lösungsprozess eingebunden werden.

5.4.4 Das Verfahren *solver3d*

Abschließend wird in Abb. 5.7 der modulare Aufbau des Lösungsverfahrens dargestellt. Zentral steht das numerische Simulationswerkzeug *solver3d*, der Diskretisierungs- und Gleichungs-Objekte und die Gleichungs-Pools zur Verfügung stellt. Zur Lösung der Gleichungssysteme wird auf die externe Bibliothek *linearSolver* zugegriffen. Die Parallelisierung übernimmt die Open-Source-Bibliothek *MPI*. Rechennetze können über das *Integrierte Design System* oder kommerzielle Programme wie *Tecplot* oder *ICEM* erzeugt werden. Die Netze werden mit Hilfe einer *converter*-Bibliothek in das passende Löser-Format konvertiert. Die *converter*-Bibliothek greift einerseits auf METIS zu, um das Rechengebiet zu partitionieren, andererseits auf die Bibliothek *polytool*, die Bearbeitungsfunktionen für polyedrische Rechennetze zur Verfügung stellt. Die Bibliothek *polytool* greift auf eine Bibliothek *geometry* zu, welche einfache Operationen zur Bearbeitung und Modifikation von Geometrien zur Verfügung stellt und dient wiederum als Basis für eine Bibliothek *GGI-intersection*, die spezielle Verschneidungs-Operationen für das General-Grid-Interface durchführt. Der solver3d stellt schließlich die problemangepassten Konfigurationen zur Verfügung, von denen beispielhaft diejenigen aufgeführt wurden, welche im Rahmen dieser Arbeit eingesetzt wurden. Für Fluid-Struktur-Interaktionen kann auf die Finite-Elemente-Bibliothek fem3 zugegrif-

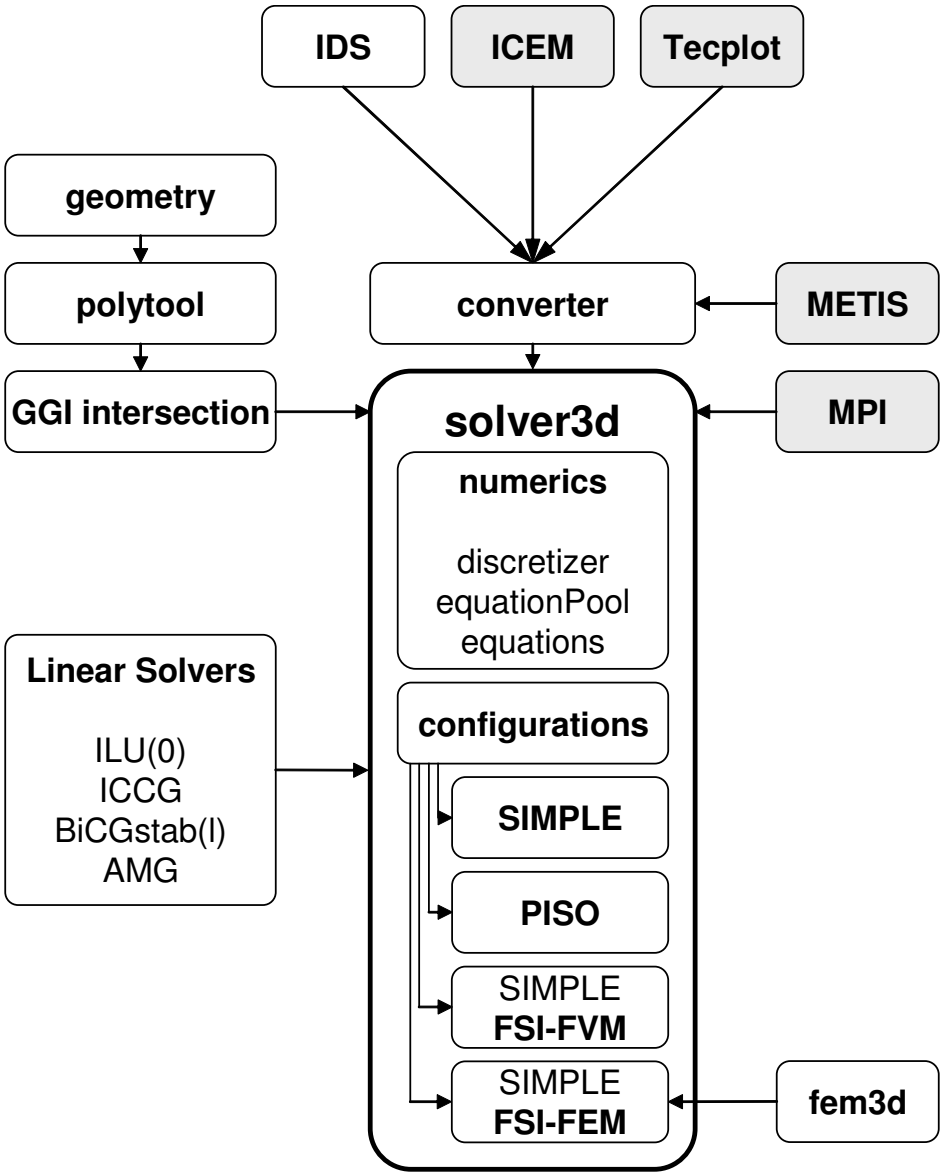

Abbildung 5.7: Modularer Aufbau des Verfahrens *solver3d*. Kommerzielle bzw. Open-Source Bibliotheken sind grau hinterlegt.

fen werden.
Das Diagramm stellt den derzeitigen Stand des Verfahrens dar.

5.5 Finite-Elemente-Verfahren fem3d

Im abschlienden Kapitel dieser Arbeit, s. Kap. 6.3, werden auch Simulationsergebnisse vorgestellt, welche mit der Finite-Elemente-Methode ermittelt wurden. Das dazu verwendete externe Modul wird als eigenständiges Gleichungsobjekt, s. Abschnitt 5.4.1, in das Lösungsverfahren eingebettet und angesteuert. Dieses Modul $fem3d$ wurde von EINZINGER [34] entwickelt, implementiert, verifiziert und validiert.

Das Verfahren ist verschiebungsbasiert implementiert und enthält eine umfangreiche Bibliothek von isoparametrischen Volumenelementen sowie generalisierten Elementen wie Stab, Balken und Schale und kann gleichermaßen für ebene wie räumliche Simulationen angewendet werden. Es können geometrische, jedoch keine physikalischen Nichtlinearitäten behandelt werden. Es wurde das linear-elastische Hooke-Materialgesetz hinterlegt.

Eine ausführliche Beschreibung der Implementierung des Verfahrens findet sich in EINZINGER [34] und ist stark an die Strategie nach BATHE [15] angelehnt. Die hinterlegten, verschiebungsbasierten Elemente neigen jedoch zu sog. Locking-Phänomenen, also künstlichen und nicht-physikalischen Versteifungseffekten, zu deren Vermeidung EINZINGER [34] die Verwendung der *Enhanced-Assumed-Strain-Method* nach ANDELFINGER [10] vorschlägt.

Die Ergebnisse von Simulationen, welche mit diesem Lösungsverfahren durchgeführt wurden, zeigen hervorragende Übereinstimmung entweder mit analytischen Referenzlösungen oder mit numerischen Lösungen, die mit der kommerziellen Finite-Elemente-Software ANSYS [2].

Kapitel 6

Verifikation und Validierung

In diesem Kapitel werden ausführlich Simulationsergebnisse, welche mit dem neuen modularen Lösungsverfahren erhalten wurden, vorgestellt und diskutiert. Das Kapitel gliedert sich in drei Teile: Zunächst werden Simulation fluidmechanischer Problemstellungen, dann strukturmechanischer Problemstellungen und zuletzt die Simulation einer Fluid-Struktur-Interaktionen vorgestellt. Die Ergebnisse werden entweder mit analytischen Lösungen oder verfügbaren Benchmark-Lösungen verglichen.

6.1 Fluidmechanik

Die folgenden Beispiele für Problemstellungen aus der Fluidmechanik gliedern sich in zwei Teile: Zunächst werden stationäre, dann instationäre Testbeispiele gezeigt und die Ergebnisse diskutiert.

Sofern nicht anders vermerkt, gelten für die vorgestellten Simulationen:

- Alle Testfälle aus der Fluidmechanik wurden mit dem neu entwickelten modularen Lösungsverfahren simuliert.
- Für die stationären Simulationen wurde stets das SIMPLE-Druckkorrekturverfahren angewendet. Bei den instationären Simulationen erfolgen Vergleiche des SIMPLE- und des PISO-Verfahrens.
- Das Konvergenzkriterium k der inneren Iterationen für die Impulsgleichungen und die skalaren Transportgleichungen wurde stets $k < 0,1$ gewählt, für die Druckkorrekturgleichung $k < 0,01$.
- Das Konvergenzkriterium k der äußeren Iterationen wurde $k < 10^{-4}$ gesetzt.
- Zur Lösung der Impulsgleichungen und der skalaren Transportgleichungen wurde die iterative Form der unvollständigen LU-Zerlegung verwendet, zur Lösung der Druckkorrekturgleichung das BiCGstab(l)-Verfahren, s. Abschnitt 4.5.
- Zellflächenwerte werden über die Beziehung 4.9 bestimmt, lediglich die Massenströme in der Druckkorrektur werden mit Hilfe der Skewness-Correction nach Beziehung 4.10 ermittelt.

- Die Gradientenberechnung erfolgt über Gl. 4.11.
- Die konvektiven Flüsse werden mit dem UDS- dem CDS-, MINMOD- oder SMART-Schema, s. Abschnitt 4.2.3, approximiert.
- Die diffusiven Flüsse werden mit den nichtorthogonalen Korrekturen nach Gl. 4.23 diskretisiert.
- Für die stationären Simulationen müssen für alle Gleichungen implizite Unterrelaxationen angewendet werden, s. Abschnitt 4.7.2. Die Unterrelaxation für die Druckkorrekturgleichung α_p wurde stets stärker als für die Impulsgleichungen eingestellt und zwar im Bereich $0,2 < \alpha_p < 0,4$. Für die Impulsgleichungen wurde die Unterrelaxation α_I im Bereich $0,5 < \alpha_I < 0,7$ gewählt. Für die restlichen Transportgleichungen wurde die Unterrelaxation α stets mit $\alpha_p = 0,2$ gewählt.

6.1.1 Stationäre Testfälle

6.1.1.1 Lid-Driven-Cavity

Als erstes akademisches Beispiel wird die zweidimensionale Simulation der Lid-Driven-Cavity diskutiert. Hierbei handelt es sich um eine quadratische, mit Flüssigkeit befüllte Kavität der Kantenlänge l, welche an den Seitenwänden und der Bodenseite von stationären Wänden und an der Oberseite von einer mit konstanter Geschwindigkeit u_0 bewegten Wand begrenzt wird. Durch die Wandbewegung wird ein Wirbelsystem in der Kavität induziert.

Dieses Testbeispiel hat sich in zahlreichen Studien als Benchmark für neue Lösungsverfahren bewährt. Es liegen ausführliche Vergleichsdaten bei verschiedenen Reynoldszahlen vor, von denen die von GHIA [40] hier zum Vergleich herangezogen werden. Diese Ergebnisse wurden von der NASA aufbereitet und zur Verfügung gestellt, s. NASA [3]. Das Rechengebiet wurde mit drei unterschiedlichen kartesischen Rechennetzen mit $16 \times 16 \times 1, 32 \times 32 \times 1$ und $64 \times 64 \times 1$, sowie einem extrem verzerrten Rechennetz mit $64 \times 64 \times 1$ Zellen, einem unstrukturierten Dreiecksnetz mit 3112 und einem Polyedernetz Zellen mit 1627 Zellen modelliert, s. Abb. 6.1. Das polyedrische Netz wurde auf Basis des Voronoi-Diagramms, s. AURENHAMMER [12] bzw. BROWN [27], des unstrukturierten Dreiecksnetzes generiert.

Da das entwickelte Simulationsverfahren nur dreidimensionale Gebiete behandeln kann, wird ein Rechennetz in der x-y-Ebene mit Einheitstiefe in z-Richtung und nur einer Zelle in Tiefenrichtung verwendet. Vorder- und Hinterseite werden mit reibungsfreien Wänden modelliert, so dass man eine wirkliche zweidimensionale Strömung erhält.

Beispielhaft werden nur Simulationsergebnisse und Vergleiche für den Strömungszustand bei einer Reynolds-Zahl $Re = 100$ besprochen. Die Aussagen, die hier getroffen werden, lassen sich auch auf die Simulationen, welche für größere Reynolds-Zahlen $Re = 400$ und $Re = 3200$, durchgeführt wurden, übertragen.

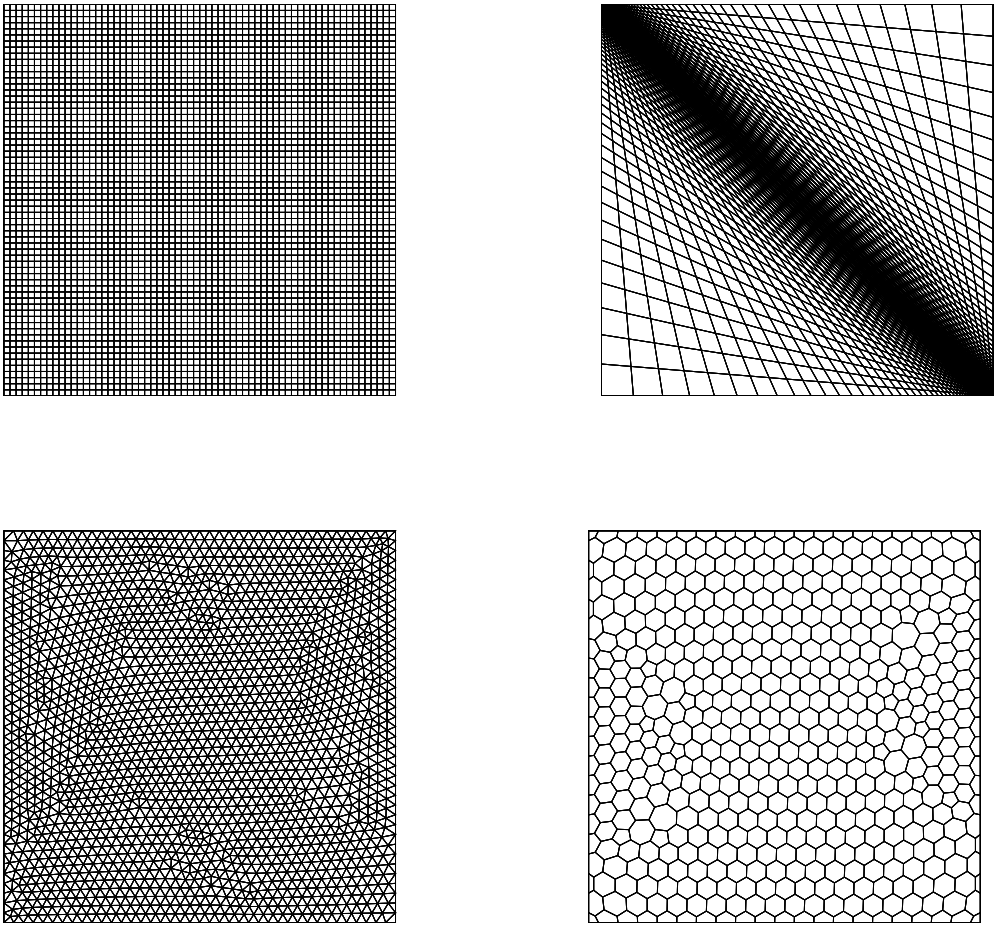

Abbildung 6.1: Unterschiedliche Vernetzungsstrategien.

In allen Abbildungen werden die Profile die normierten Geschwindigkeitskomponenten u/u_0 und v/u_0 an den Positionen $y = 0,5l$ und $x = 0,5l$ mit den oben erwähnten Referenzdaten verglichen. In Abb. 6.3 sind die Ergebnisse für die sukzessive verfeinerten kartesischen Netze dargestellt. Die konvektiven Terme wurden mit dem UDS-Verfahren diskretisiert. Man erkennt deutlich die Netzkonvergenz bei feiner werdenden Netzen.

Ferner wurde der Einfluss der unterschiedlichen Diskretisierungsschemata für die konvektiven Flüsse auf dem feinsten kartesischen Netz untersucht. Hier wurden das Schema erster Ordnung UDS und die Schemen zweiter Ordnung CDS und MINMOD verwendet. Die nachgeführte Korrektur wurde mit einem Blending-Faktor $\alpha = 1,0$ realisiert, s. 4.2.3.3.

Dass bei der Diskretisierungstechnik wirklich eine Genauigkeit zweiter Ordnung erreicht wird, zeigen die in den Abb. 6.3 dargestellten Verläufe der Fehlerreduktion. Ausgewertet werden hier der Minimalwert der Geschwindigkeitskomponente u sowie die Minimal- und Maximalwerte der Geschwindigkeitskomponente v für die oben

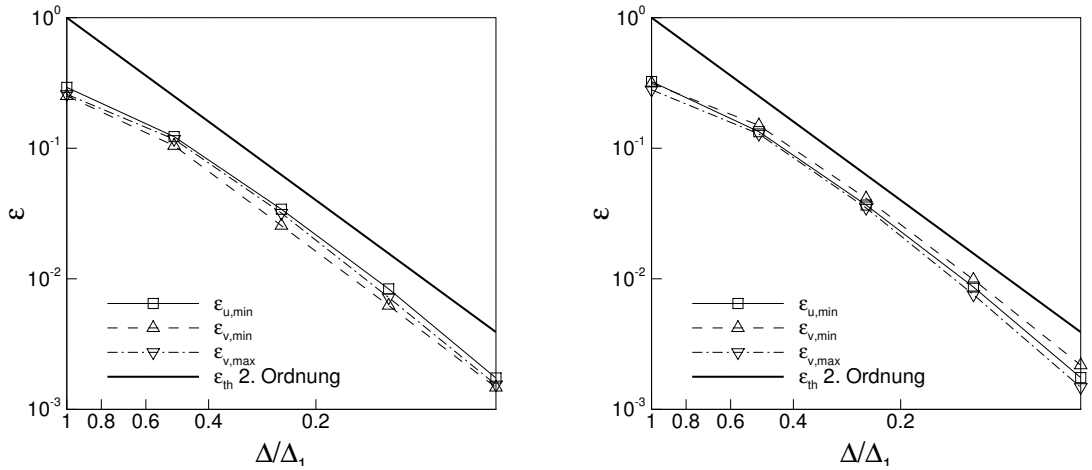

Abbildung 6.2: Fehlerreduktionen ϵ über der relativen Netzfeinheit Δ/Δ_1 unter Verwendung der konvektiven Schemata CDS (links) und MINMOD (rechts).

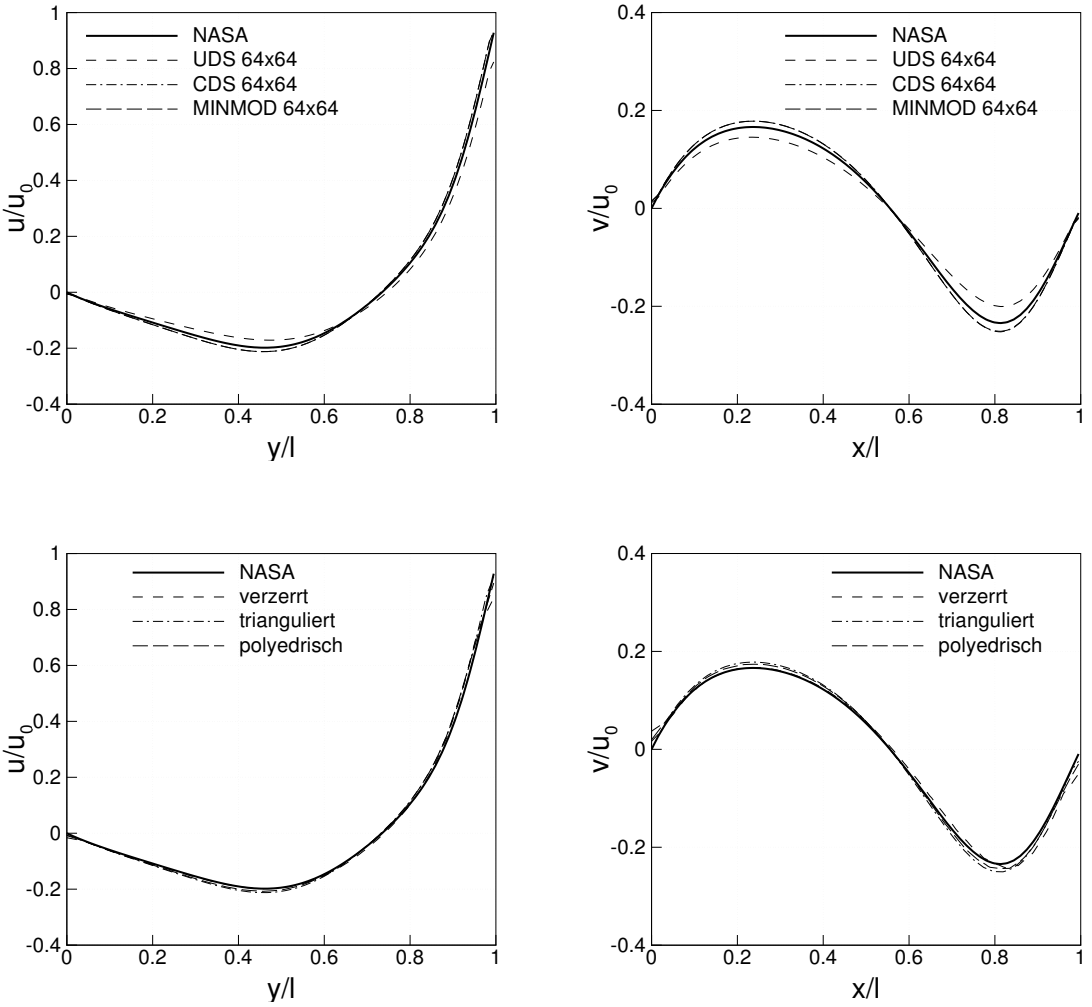

Abbildung 6.3: Verlauf von $u(y)$ an der Stelle $x = 0,5$ und $v(x)$ an der Stelle $y = 0,5$ im Vergleich mit Referenzdaten, Studie zur Netzkonvergenz (oben) und zur Netztopologie (unten).

bereits besprochenen dargestellen Verläufe.

Die Ergebnisse zeigen, wie zu erwarten, die größten Abweichungen für das UDS-Schema. Auch das MINMOD-Schema zeigt noch geringe Abweichungen von den Referenzdaten, das CDS-Schema liefert im Vergleich zur Referenz nahezu identische Werte. Das Konvergenzverhalten ist beim UDS-Schema am besten, gefolgt vom MINMOD- und vom CDS-Schema. Das MINMOD-Schema stellt damit einen guten Kompromiss zwischen erzielter Genauigkeit und Stabilität des Verfahrens dar.

Die Ergebnisse, welche auf den beiden unstrukturierten Rechennetzen unter Anwendung der CDS-Diskretisierung gewonnen wurden, zeigen ebenfalls sehr gute Übereinstimmungen mit den Referenzdaten, s. Abb. 6.3. Das triangulierte Rechennetz besitzt dabei etwa so viele Rechenzellen wie das kartesische Rechennetz mit $64 \times 64 \times 1$ Zellen, weswegen die Ergebnisse, die auf diesem Netz ermittelt wurden, als Vergleich ebenfalls dargestellt sind. Das polyedrische Rechennetz wurde auf Basis des triangulierten Netzes mit Hilfe des Voronoi-Diagramms gebildet und das Rechennetz hat deutlich weniger Rechenzellen als das triangulierte Rechennetz, dafür besitzt eine Zelle aber nun wesentlich mehr Nachbarn. Dies wirkt sich besonders positiv auf eine glatte numerische Berechnung von Gradienten aus. Die Qualität des Ergebnisses bleibt trotz des relativ groben Netzes erhalten.

Die Untersuchung auf dem extrem verzerrten, strukturierten Rechennetz mit $64 \times 64 \times 1$ Volumina zeigt ebenfalls eine sehr gute Übereinstimmung mit den Referenzdaten und ist ebenfalls in Abb. 6.3 dargestellt. Eine akzeptale Konvergenz konnte bei der verwendeten CDS-Diskretisierung auf diesem Netz nur durch die Verwendung der neu implementierten nichtorthogonalen Korrekturen für die diffusiven Flüsse und durch Anpassung der Unterrelaxationen für Impulsgleichungen und Druckkorrekturgleichungen erreicht werden. Eine weitere Stabilisierung erhält man durch die Verwendung der Gradientenberechnung nach der Methode der kleinsten Quadrate, s. Abs. 4.13.

6.1.1.2 Konvektionsgetriebene Strömungen

Die Strömung des vorhergenden Abschnittes war aufgrund der niedrigen betrachteten Reynoldszahl hauptsächlich durch Diffusionseffekte geprägt. Mit dem Testfall konnte die stabile und genaue Diskretisierungstechnik für die diffusiven Terme auch für unstrukturierte und qualitativ schlechte Rechennetze gezeigt werden.

Bei konvektionsdominierten Strömungen, welche hohe Reynolds-Zahlen aufweisen, treten diffusive Effekte in den Hintergrund, und eine genaue Approximation der konvektiven Terme bestimmt die Güte des numerischen Verfahrens in entscheidendem Maße.

Im Folgenden wird an zwei kurzen Beispielen, welche für die Anwendung des Verfahrens zur Simulation von Strömungsmaschinen besonders wichtig sind, auch die Güte der Diskretisierungstechnik der konvektiven Terme gezeigt.
Zunächst wird eine Staupunktströmung untersucht, für die eine analytische Lösung existiert. Staupunktströmungen treten bei jeder Profil- oder Schaufelumströmung auf

und sind deswegen von elementarer Bedeutung bei der Untersuchung von Strömungsmaschinen.
Hierauf wird noch der sog. Gostelow-Testfall untersucht, der eine zweidimensionale Strömung durch ein Schaufelgitter darstellt. Auch zu diesem Fall existiert eine geschlossene analytische Lösung.

Staupunkt-Strömung

Die untersuchte Staupunktströmung erfolgt in einem quadratischen zweimdimensionalen Gebiet vom oberen Rand zu den Seitenrändern und verlässt dort das untersuchte Gebiet. Dieser Testfall wird auch von SKODA [93] untersucht. Die neuen hinterlegten unstrukturierten Diskretisierungstechniken und numerischen Eigenschaften des Verfahrens lassen sich an diesem Testfall in geeigneter Weise untersuchen.
Die untere Wand wird reibungsfrei modelliert, damit ein Vergleich mit der analytische, potentialtheoretischen Lösung möglich ist. Diese feldweise Lösung für das Geschwindigkeits- und das Druckfeld ist durch die folgenden Gleichungen gegeben:

$$v_x = a\left(x^2 - y^2\right) \tag{6.1}$$

$$v_y = -2axy, \tag{6.2}$$

$$p = -\frac{1}{2}\rho\left(v_x^2 + v_y^2\right) + C. \tag{6.3}$$

Die Geschwindigkeiten werden an den Ein-und Ausstromrändern als Dirichlet-Randbedingungen vorgegeben. Für das Druckfeld ergeben sich Neumann-Randbedingungen an allen Rändern des Strömungsgebietes, weswegen in der Beziehung für den Druck auch eine beliebige Konstante C auftritt. Die Geschwindigkeitsvektoren werden durch Hyperbeln beschrieben, die Druck-Isolinien durch konzentrische Kreise.

Durch den Vorfaktor a kann die Reynolds-Zahl der Strömung definiert werden. Es wurden Simulationen für $a = 1$, $a = 10$ und $a = 100$ durchgeführt, wobei die dynamische Viskosität stets zu $\mu = 0.0001 kg/ms$ gewählt wurde. Für die unterschiedlichen Werte für a ergeben sich keine grundlegend unterschiedlichen Aussagen, so dass nur Ergebnisse aus Rechnungen mit $a = 1$ im Folgenden besprochen werden.

Für die numerischen Simulationen wurden sechs sukzessive verfeinerte kartesische und sechs verfeinerte polyedrische Rechennetze analog zu Abb. 6.1 gewählt. Mit der angestrebten Vervierfachung der Rechenzellen auf den einzelnen Netzebenen ergeben sich die Netzgrößen in nachstehender Tab. 6.1.

In den numerischen Simulationen wurden die konvektiven Terme mit dem UDS-, dem CDS- und dem MINMOD- und dem SMART-Verfahren diskretisiert. Um Konvergenz auch für das CDS-Verfahren auf allen Netzebenen erreichen zu können, wurde das CDS mit einem zehn-prozentigen UDS-Anteil gemischt. Mit dieser Modifikation wurden dieselben stabilen und schnellen Konvergenzeigenschaften für alle unterschiedlichen Verfahren erreicht.

Netz	kartesisch	N_i/N_{i-1}	polyedrisch	N_i/N_{i-1}
g1	16	-	16	-
g2	64	4,0	68	4,25
g3	256	4,0	249	3,66
g4	1024	4,0	1036	4,16
g5	4096	4,0	4127	3,98
g6	16384	4,0	16245	3,94

Tabelle 6.1: Netzgrößen und Verfeinerungsgrade für den Testfall der Staupunktströmung.

In Abb. 6.4 sind die Fehlerreduktionsraten ϵ für die einzelnen Diskretisierungsschemata dargestellt. Die Fehler werden mit dem UDS-Verfahren mit der erwarteten ersten Ordnung, mit den höherwertigen Verfahren mit der erwarteten annähernd zweiten Ordnung reduziert. Lediglich für das CDS-Verfahren ergibt sich ein nicht absolut stetiges Abklingverhalten des Fehlers. Dies lässt sich zum einen mit dem oszillatorischen Verhalten des Verfahrens und zum anderen mit der Beimischung des UDS-Anteils erklären, welches das Verfahren in der Genauigkeit herabsetzt. Bemerkenswert ist hier jedoch, dass auch mit dem CDS-Verfahren für alle Gitter eine konvergierende Lösung erzielt werden konnte.

	kartesisch						polyedrisch					
	CDS(0,9)		MINMOD		SMART		CDS(0,9)		MINMOD		SMART	
Netz	u	v	u	v	u	v	u	v	u	v	u	v
g1	-	-	-	-	-	-	-	-	-	-	-	-
g2	1,0	0,9	2,6	2,2	2,8	2,2	1,4	1,5	2,4	2,3	2,1	1,4
g3	1,2	1,2	2,4	2,0	1,7	1,9	1,7	3,2	1,8	1,6	3,0	3,1
g4	3,3	3,8	1,8	1,8	1,8	2,1	2,0	3,4	2,0	1,9	1,4	1,9
g5	1,7	2,3	1,8	1,9	1,9	2,2	1,8	1,4	1,5	1,3	1,0	1,9
g6	1,4	1,5	1,9	1,9	1,9	2,2	1,5	1,4	1,4	1,5	1,1	1,0

Tabelle 6.2: Erreichte Ordnungen für den Testfall Staupunktströmung.

Auch bei der Verwendung der polyedrischen Rechennetze ist kein wesentlicher Unterschied im Konvergenzverhalten hinsichtlich Iterationszahl und Stabilität festzustellen. Die Ergebnisse sind den erzielten Ergebnissen auf den kartesischen Rechennetzen gleichwertig; die Fehlerreduktion erfolgt für alle Größen und die höheren Konvektionsschemata mit annähernd zweiter Ordnung, s. Tab. 6.2. Eine Ausnahme bilden lediglich die Lösungen, die mit dem SMART-Verfahren ermittelt wurden, wo die Fehlerreduktionsrate vom vorletzten auf das feinste Netz nahe den Wert 1.0 geht.
Die erreichten Ordnungen für die mit den Impulsgleichungen ermittelten Geschwindigkeitskomponenten sind in Tab. 6.2 für die höherwertigen Konvektionsschemata zusammengefasst.

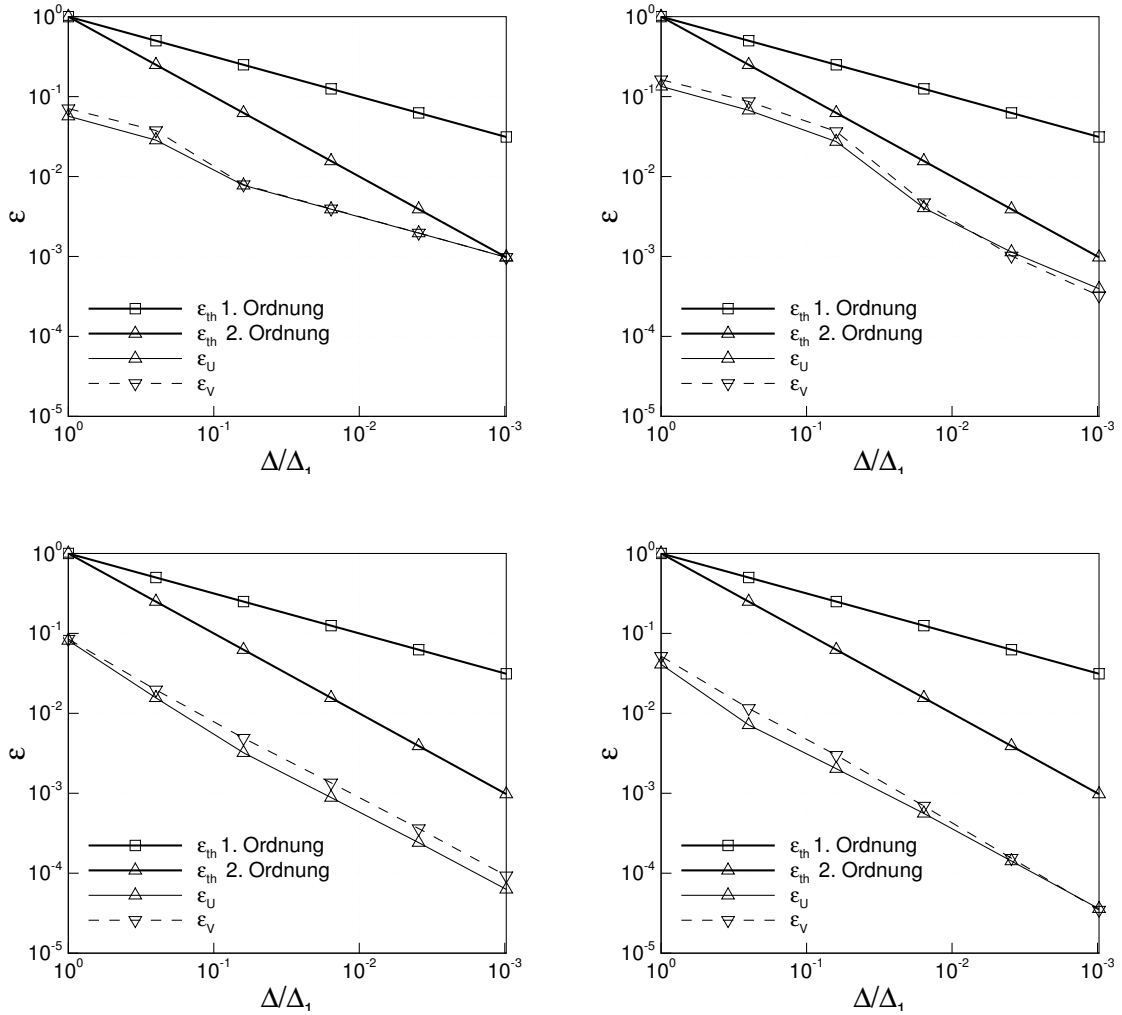

Abbildung 6.4: Fehlerreduktionen ϵ über der relativen Netzfeinheit Δ/Δ_1 für den Testfall der Staupunktströmung unter Verwendung der konvektiven Schemata UDS (links oben), CDS (rechts oben), MINMOD (links unten) und SMART (rechts unten) unter Verwendung kartesischer, sukzessive verfeinerter Rechennetze.

Gitterströmung nach Gostelow

Bei der reibungsfreien Gitterströmung nach Gostelow treten zusätzliche Anforderungen an das numerische Verfahren im Gegensatz zum vorhergehenden, vorgestellten Testfall auf. Zum einen die Dirichlet-Randbedinung für den Druck am Auslass und zum anderen die translatorisch periodischen Ränder des Rechengebietes.

Die analytische Lösung für den Druckverlauf um die Schaufel gilt für eine abgewickelte, unendlich ausgedehnte Schaufelkaskade. Die analytische Lösung wurde von GOSTELOW [41] ermittelt, von MÜLLER [74] und REINELT [84] korrigiert. Sie wird im Vergleich mit numerischen Simulationsergebnissen auf unterschiedlich feinen Netzen in Abb. 6.6 dargestellt und auf Strichstärke genau getroffen.

Die Profilkontur wird mit einer reibungsfreien Wand modelliert. Die Vernetzungsstrategie wird für die ersten drei Netzstufen in Abb. 6.5 dargestellt. Zusätzlich sind in der

Abbildung die Druckisolinien dargestellt. Die Rechenzellen sind in Bereichen des Gitters geschert und nichtorthogonal, so dass eine volle zweite Ordnung bei Verwendung hochauflösender Konvektions-Diskretisierungen nicht zu erwarten ist. Auch müssen sich hierdurch immer Abweichung zur analytischen Lösung ergeben, die aber natürlich zu minimieren sind.

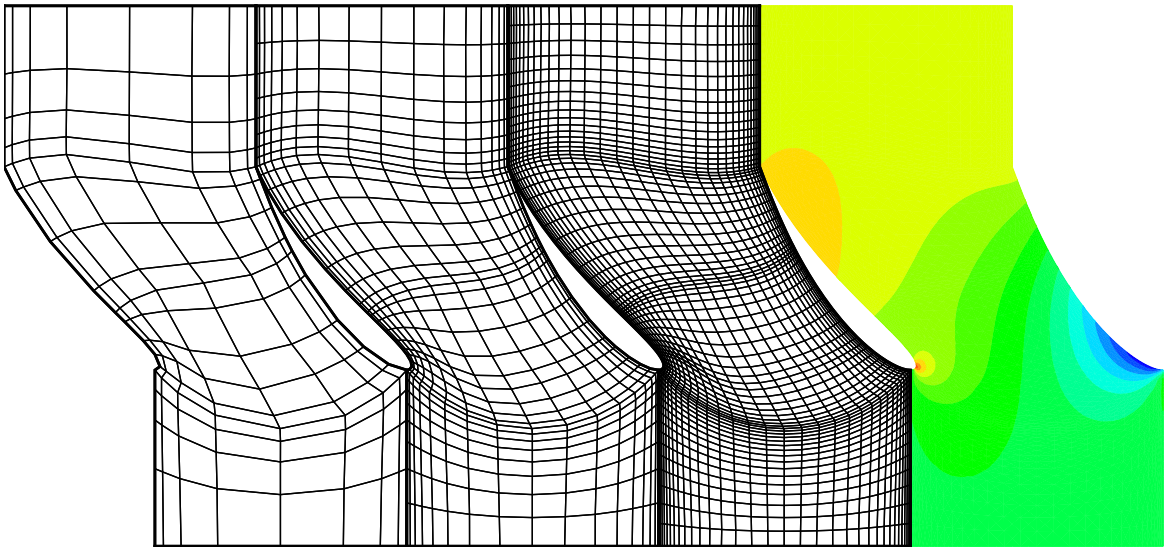

Abbildung 6.5: Vernetzungsstrategie für den Testfall des Gostelow-Gitters für die Netze g1, g2 und g3, sowie Konturplot des Druckverlaufs auf dem feinsten Rechengitter.

Es wurden Simulationen auf fünf sukzessive verfeinerten Netzen durchgeführt. Bei der numerischen Betrachtung der Fehlerreduktion wird die auf dem feinsten Rechennetz ermittelte Lösung als Referenz herangezogen. Die Netzgrößen sind in Tab. 6.3 zusammengefasst. Es werden wiederum Simulationsergebnisse, welche mit den UDS-, dem

Netz	Knoten	N_i/N_{i-1}
g1	136	-
g2	544	4,0
g3	2176	4,0
g4	8704	4,0
g5	34816	4,0

Tabelle 6.3: Netzgrößen und Verfeinerungsgrade für den Testfall des Gostelow-Gitters.

CDS-, dem MINMOD- und dem SMART-Verfahren ermittelt wurden, vorgestellt.
Für eine stabile Verwendung des CDS-Verfahrens musste ein 10-prozentiger UDS-Anteil beigemischt werden. Die ermittelten Fehlerreduktionsraten für die höherwertigen Schemata sind in Abb. 6.6 und Tab. 6.4 zusammengefasst. Das CDS-Verfahren weist hierbei zu den feineren Netzen hin das ungenaueste Verhalten auf. Dies kann mit dem oszillatorischen Charakter des Verfahrens und dem beigemischten UDS-Anteil begründet werden. Das SMART- und das MINMOD-Verfahren lassen zu den feineren Gittern

hin eine höhere Fehlerreduktionsrate erkennen. Das SMART-Verfahren liefert im Vergleich zum MINMOD-Verfahren geringere absolute Fehler bei sehr ähnlichen, stabilen Konvergenzeigenschaften.

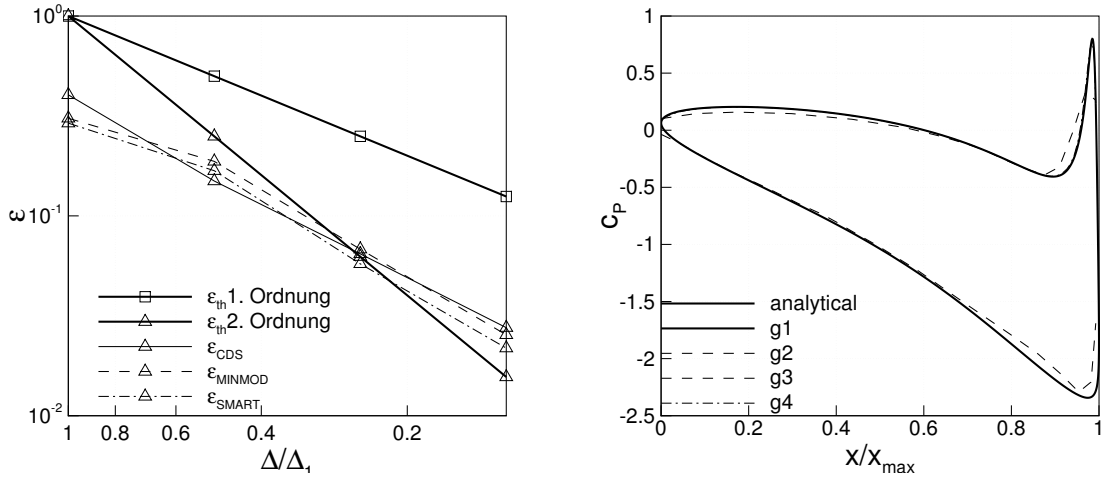

Abbildung 6.6: Links: Fehlerreduktion ϵ über der relativen Netzfeinheit Δ/Δ_1 für den Testfall Gostelow-Gitter unter Verwendung der konvektiven Schemen CDS, MINMOD und SMART. Rechts: Analytische Lösung des normierten Druckverlaufs im Vergleich mit numerisch ermittelten Lösungen auf den Netzen g1, g3 und g5.

Für die folgenden Lösungen auf den feiner werdenden Netzen erkennt man durchschnittliche Werte für die ermittelten Ordnungen im Bereich zwischen $0,9$ und $1,7$, s. Tab. 6.4. Die ermittelten Ordnungen nach Tab. 6.4 fallen bei Verwendung

	CDS	MINMOD	SMART
g1	-	-	-
g2	1,4	1,7	1,7
g3	1,3	0,9	0,9
g4	1,1	1,3	1,5
g5	1,2	1,3	1,3

Tabelle 6.4: Erreichte Ordnungen für den Testfall des Gostelow-Gitters.

des MINMOD- oder SMART-Verfahrens zunächst auf einen Wert unter Eins beim Übergang von Netz G1 auf G2 auf. Dies ist durch die noch unzureichende Abbildung der Profilkontur bei der gröbsten Vernetzung zu erklären, s. Abb. 6.5. Dieses Verhalten deckt sich mit Erkenntnissen aus der Literatur, s. PERIC [81], wonach sich das erwartete asymptotische Abklingverhalten des Fehlers erst bei Verwendung hinreichend feiner Netzen ergibt.

Auch für Konvektions-dominierte Strömungen arbeitet das numerische Verfahren stabil und genau. Unter idealen Netzbedingungen werden für die implementierten konvektiven Schemata Ordnungen nahe 2.0 erreicht. In der praktischen Anwendung mit nichtorthogonalen oder verzerrten Netzen muss mit Ordnungen zwischen $1,1 < O < 1,5$ gerechnet werden.

6.1.2 Simulation der Strömung in radialen Kreiselpumpen

In den folgenden Abschnitten wird das neue Lösungsverfahren hinsichtlich der Eignung zur Simulation realer Anwendungen untersucht. Dabei wird eine Pumpenreihe radialer Bauweise untersucht. Die Betriebspunkte der unterschiedlichen Pumpen werden durch die dimensionslose spezifische Drehzahl n_q beschrieben:

$$n_q = n\frac{Q^{\frac{1}{2}}}{H^{\frac{3}{4}}}; \tag{6.4}$$

wobei n die Drehzahl darstellt, H die Förderhöhe und Q den geförderten Volumenstrom. Es werden Simulationen für Maschinen spezifischer Drehzahl $n_q = 30$, 40, 50, 70 und 90 $1/min$ jeweils für den Optimal- bzw. Auslegungspunkt, welcher durch den geförderten Volumenstrom $Q = Q_{opt}$ definiert ist, durchgeführt.

Repräsentativ wird nur ein Sektor bzw. ein Schaufelkanal der Radialmaschinen betrachtet. Der rotierende Bereich wird im Relativsystem modelliert, s. 2.4, wodurch eine stationäre Simulation möglich wird. Die Vernetzung erfolgt auf unstrukturierten Rechennetzen, s. Abb. 6.7 dargestellt. Die Rechennetze wurden mit dem Integrierten Design-System (IDS) des Lehrstuhl zunächst als blockstrukturierte Netze erstellt und dann zu einem unstrukturierten rotierenden und einem unstrukturierten stationärem Netzteil zusammengefasst. Die einzelnen Netzbereiche müssen dabei keine zusammenhängenden Gebiete darstellen.

Es werden laminare und turbulente Strömungssimulationen durchgeführt. Bei den laminaren Strömungen erfolgt die Modellierung mit Hilfe des Quasi-Navier-Stokes- bzw. des Quasi-Euler-Verfahren, welche im folgenden Abschnitt beschrieben werden. Bei den turbulenten Strömungen wird das Standard-$k\epsilon$-Turbulenzmodell mit Standard-Wandfunktion verwendet, Kap. B und Abs. B.0.5.

Verglichen werden die Ergebnisse im Falle der laminaren Strömungen mit Ergebnissen, die mit dem Vorgängerverfahren $ns3d$, s. SCHMALHORST [90] ermittelt wurden. Im Falle der turbulenten Strömungen werden Vergleiche mit dem Vorgängerverfahren $ns3d$ und der kommerziellen CFD-Software ANSYS-CFX durchgeführt.

6.1.2.1 Quasi-Navier-Stokes- bzw. Quasi-Euler-Verfahren

Die Berechnungen der laminaren Strömungen wurden mit dem Quasi-Navier-Stokes (QNS)- bzw. mit dem Quasi-Euler-Verfahren (QE) durchgeführt. Diesem Verfahren liegen folgende Annahmen zugrunde:

- Die effektive dynamische Viskosität ist konstant, womit bei konstanter laminarer Viskosität μ_l die Konstanz der turbulenten Viskosität μ_t folgt.
- Die turbulente Viskosität wird über eine Turbulenzkonstante C_t und die laminare Viskosität berechnet: $\mu_t = C_t\mu_l$.
- Die Turbulenzkonstante C_t liegt dabei im Bereich $10 < C_t \leq 1000$, womit die turbulente Viskosität stets deutlich höher als die laminare Viskosität ist.

Die Vorteile des Verfahrens liegen darin begründet, dass nun keine Transportgleichungen für die turbulente kinetische Energie und die turbulente Dissipation gelöst werden müssen. Zusätzlich können die QNS- bzw. QE-Simulationen im Vergleich zu RANS-Berechnungen, auf deutlich gröberen Netzen ausgeführt werden. Damit ergeben sich für QNS/QE-Simulationen im Vergleich zu RANS-Berechnungen deutlich geringere Simulationszeiten, weswegen dieses Verfahren für Voroptimierungen und Vorauslegungen von Strömungsmaschinen in vielen Fällen sehr gut geeignet ist, wie z. B. von SCHILLING [89] und SCHMALHORST [90] gezeigt wird.

6.1.2.2 Auswertegrößen

Zum detaillierten Vergleich der beiden Simulationsverfahren wird die dimensionslose Druckverteilung an verschiedenen Positionen der Laufschaufeloberfläche betrachtet:

$$c_p = \frac{p}{\frac{1}{2}\rho u_{ref}^2}. \tag{6.5}$$

Der dimensionslose Druck wird über der dimensionslosen Lauflänge s entlang der Profiloberfläche auf der Druck- und Saugseite aufgetragen. Als Profilpositionen wurden Positionen in Nähe der Deckscheibe (shroud), der Nabe (hub) und etwa in Profilmitte (mean) gewählt.

Als integrale Auswertegrößen werden die Totaldruckerhöhung Ψ_{rcu} und der Wirkungsgrad η_h herangezogen. Alle Größen werden dimensionslos dargestellt, um eine Vergleichbarkeit von Strömungsmaschinen unterschiedlicher Baugröße zu gewährleisten. Als Entdimensionierungsparameter werden der Durchmesser des Laufrades D sowie die Winkelgeschwindigkeit ω herangezogen.

$$r_{ref} = \frac{D}{2}, \tag{6.6}$$

$$u_{ref} = \omega r_{ref}, \tag{6.7}$$

$$P = \frac{2p}{\rho r_{ref}^2 \omega^2}. \tag{6.8}$$

Die Totaldruckerhöhung wird als Differenz der integralen Totaldrücke P bestimmt:

$$\Psi_t = P_{T,out} - P_{T,in}. \tag{6.9}$$

Die theoretische Totaldruckerhöhung ergibt sich aus der Dralländerung der Strömung:

$$\Psi_{t,th} = \Psi_{rcu} = \frac{2}{u_{ref}^2}\left(c_{u2}u_2 - c_{u1}u_1\right), \tag{6.10}$$

wobei c_u die Umfangskomponenten der Absolutgeschwindigkeit und u die Umfangsgeschwindikeit darstellt. Beide Größen werden über Bilanzebenen unmittelbar vor bzw. hinter dem Laufrad bestimmt.
Der hydraulische Wirkungsgrad wird dann mit Hilfe von

$$\eta_h = \frac{\Psi_t}{\Psi_{t,th}} \tag{6.11}$$

ermittelt.

6.1.2.3 Simulationsergebnisse

Die Simulationen für die unterschiedlichen spezifischen Drehzahlen $n_q = 30, 40, 50, 70, 901/min$ wurden auf unstrukturierten Rechennetzen durchgeführt, welche einen Schaufelkanal der untersuchten Pumpen in einen stehenden und einen rotierenden Bereich unterteilen. Die Netztopologien sind in Abb. 6.7 für die unterschiedlichen Maschinentypen dargestellt. Für die QNS-Rechnungen lassen sich deutlich gröbere Netze als für die RANS-Rechnungen verwenden. Die einzelnen Netzgrößen sind in Tab. 6.5 zusammengefasst. An den Seitenrändern der untersuchten Rechengebiete werden periodische Randbedingungen nach Kap. 4.8.3 gesetzt.

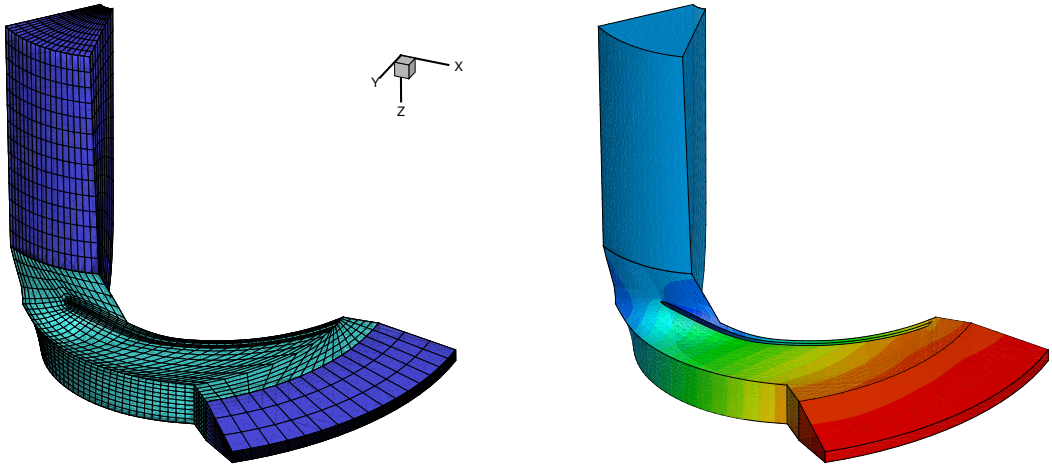

Abbildung 6.7: Einkanalsimulation einer Pumpe $n_q = 30$: Vernetzungsstrategie mit rotierendem und stehendem Bereich (links) und Druckkonturen.

$n_q[1/min]$	N_{QNS}	N_{RANS}	N_{QNS}/N_{RANS}
30	8280	74034	0,11
40	7524	39780	0,19
50	11328	61776	0,19
70	8875	91200	0,10
90	9112	64056	0,14

Tabelle 6.5: Netzgrößen für die QNS- und RANS-Berechnungen der Pumpenreihe.

Wie bereits erwähnt, werden die Ergebnisse, die das neue Verfahren liefert, mit Ergebnissen des Vorgängerverfahrens $ns3d$ und der kommerziellen Software $ANSYS-CFX$ verglichen.
Bei den zunächst vorgestellten RANS-Simulationen wird das Standard-$k\epsilon$-Turbulenzmodell mit Standardwandfunktion verwendet.
Die Diskretisierung der konvektiven Terme erfolgt in den Simulationsläufen zunächst mit UDS, dann mit $MINMOD$ und CDS, s. Abschnitt 4.2.3. In der kommerziellen Software ANSYS-CFX werden nur Rechnungen mit dem sog. *First-Order*- und dem *High-Resolution*-Schema durchgeführt. Das *High-Resolution*-Schema liefert hierbei wie MINMOD und CDS im Idealfall optimaler Netze eine Genauigkeit zweiter Ordnung.

RANS-Simulationen

Zunächst werden die ermittelten integralen Werte der RANS-Simulationen untereinander verglichen und diskutiert. In den Abb. 6.8 sind die Werte für Druckzahlen Ψ_{rcu} und die Werte für die hydraulischen Wirkungsgrade η_h für Ergebnisse aus UDS-Simulationen dargestellt.

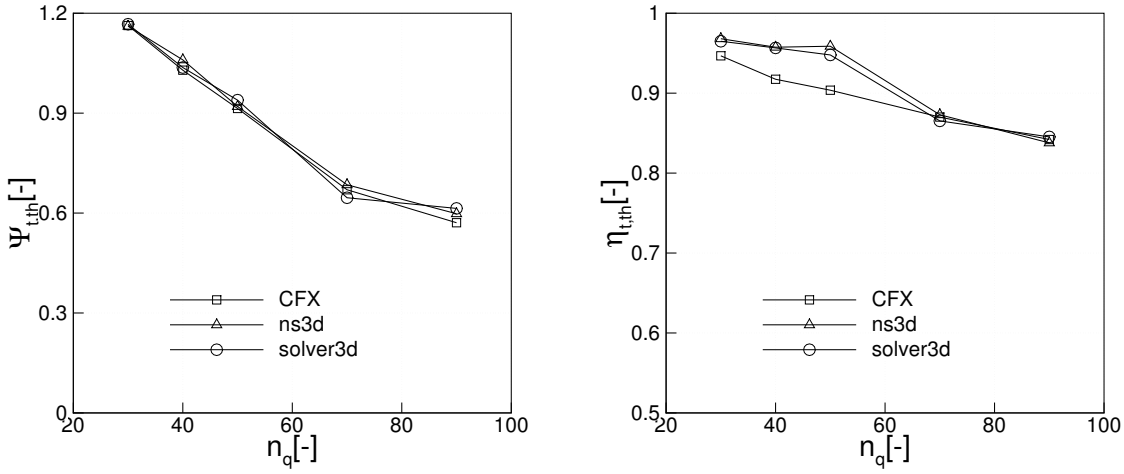

Abbildung 6.8: Ingetrale Werte Druckzahl Ψ_{rcu} (links) und des hydraulischen Wirkungsgrades η_h über der spezifischen Drehzahl n_q und für UDS-Simulationen.

Man erkennt zunächst für alle Kurven die gleichen Tendenzen in den Verläufen der untersuchten Größen.
Bei den Verläufen für die dimensionslosen Druckzahlen Ψ_{rcu} werden für alle Verfahren und Drehzahlbereiche sehr ähnliche Werte ermittelt.

Für die ermittelten hydraulischen Wirkungsgrade η_h liefert das neue Verfahren *solver3d* sehr ähnliche Werte wie *CFX* im Bereich der hohen Drehzahlen und größere Abweichungen im Bereich der niedrigen Drehzahlen.

Simulationen mit dem UDS-Diskretisierungsschema sollten nur für erste Abschätzungen durchgeführt werden. Zur realistischen Systemabbildung sollte ein Diskretisierungsschema zweiter Ordnung für die konvektiven Terme verwendet werden. Vergleiche von derartig ermittelten Simulationsergebnissen werden in Abb. 6.9 dargestellt. Die Simulationsergebnisse, welche mit *ns3d* und *solver3d* ermittelt wurden, wurden mit MINMOD-Verfahren zur Diskretisierung der konvektiven Terme ermittelt, wobei kein Blending-Faktor angewendet wurde, s. Abs. 4.2.3 und Abs. 4.2.3.3. Für die Simulationen mit *CFX* wurde das High-Resolution-Schema zur Diskretisierung der konvektiven Terme verwendet, welches wie MINMOD im Idealfall eine Diskretisierungsgenauigkeit zweiter Ordnung liefert.
Für alle dargestellten Simulationsergebnisse sind sehr gute Übereinstimmungen für die ermittelten Verläufe für Druckzahlen und Wirkungsgrade erzielt worden. Sowohl *ns3d* als auch das Verfahren *solver3d* liefern im Vergleich mit *CFX* erhöhte Werte in den Druckzahlen und den Wirkungsgraden. Das Verfahren *solver3d* bringt hierbei noch etwas höhere Werte als *ns3d*. Diese kann mit der geringen numerischen Diffusivität des

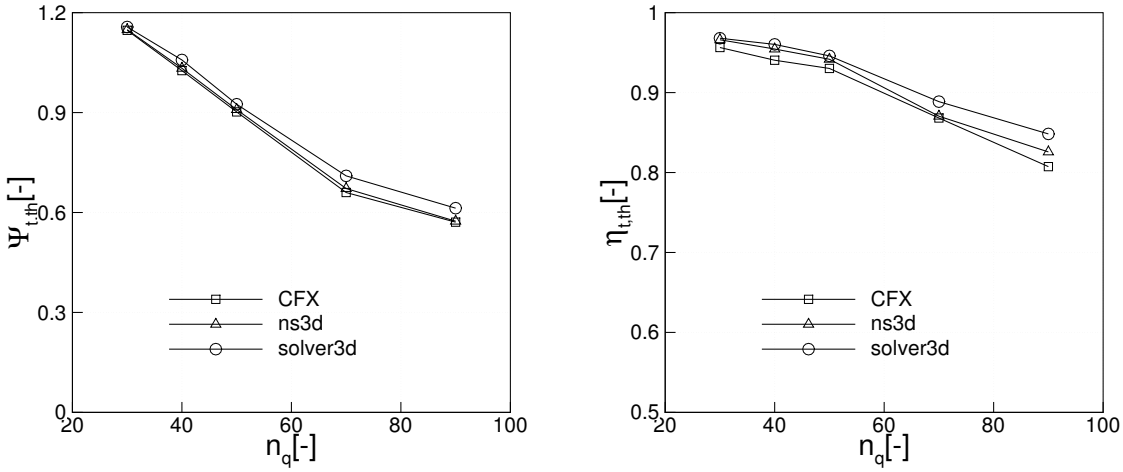

Abbildung 6.9: Ingetrale Werte Druckzahl Ψ_{rcu} (links) und des hydraulischen Wirkungsgrades η_h über der spezifischen Drehzahl n_q und für HOS-Simulationen.

Verfahrens erklärt werden, welche durch die überarbeiteten Diskretisierungstechniken für die konvektiven Terme und die neue Diskretisierungstechnik für die diffusiven Terme erklärt werden kann.

Einen detaillierteren Blick auf die Qualität der erzielten Simulationsergebnisse erhält man mit der Untersuchung der dimensionslosen Druckverläufe c_p an ausgewählten Positionen der Laufschaufeln. Derartige Ergebnisse für Simulationen mit dem UDS- und dem MINMOD- bzw. HIGH-RESOLUTION-Schema für die unterschiedlichen Verfahren sind in die Abb. 6.10 zu sehen. Die dimensionslosen Druckverläufe sind hierbei repräsentativ für die Kreiselpumpe der spezifischen Drehzahl $n_q = 501/min$ der dimensionslosen Schaufelkoordinate s an Schaufelober- und unterseite für Positionen an einer mittleren Position zwischen Nabe und Deckscheibe dargestellt.

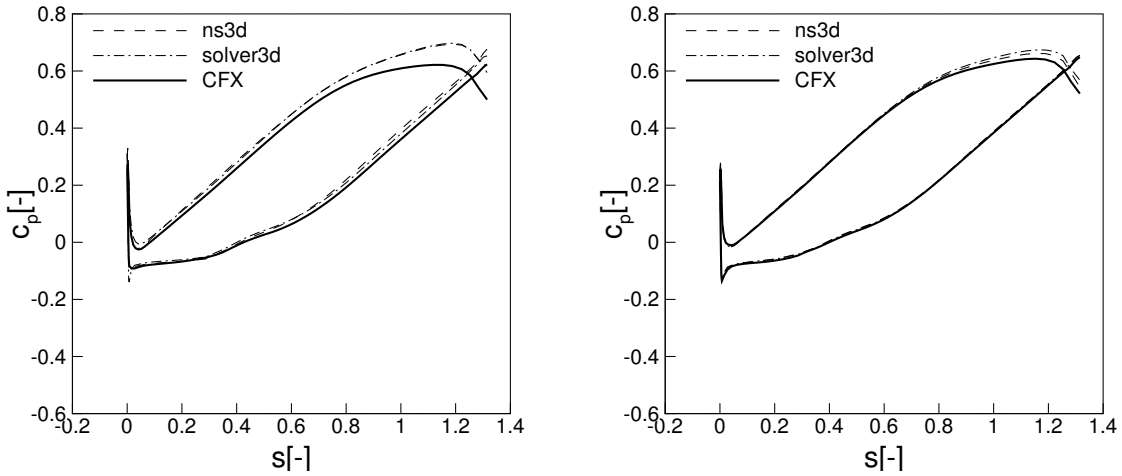

Abbildung 6.10: Dimensionslose Druckverläufe bei Anwendung des RANS-Verfahrens an einem repräsentativen Mittelschnitt der Laufschaufel für UDS- und HOS-Diskretisierung.

Die beiden Verfahren $ns3d$ und $solver3d$ liefern bis auf Strichstärke identische Verläufe. Im Gegensatz zu CFX fällt eine Abweichung im Bereich der Schaufelhinterkante und Schaufeloberseite bei Verwendung des UDS-Diskretisierungsschema auf. Diese Abweichungen verschwinden fast vollständig bei Verwendung der Diskretisierungsschemata zweiter Ordnung. Die Abbildung von Druck- und Saugspitzen im Bereich der Schaufelvorderkanten, welche für eine genaue Voraussage von Kavitationsgebieten besonders wichtig sind, erfolgt von allen Verfahren in gleicher Quantität und Qualität, s. BOGNER [20] und BOGNER [21].

Stabilität

Das neuen Verfahrens $solver3d$ zeigt im Gegensatz zum Vorgängerverfahren $ns3d$ eine höhere numerische Stabilität.
Diese resultiert zunächst aus der polyedrisch unstrukturierten Datenstruktur, wodurch die Rechengebiete nicht mehr in viele Einzelbereiche zerlegt werden müssen wie beim blockstrukturierten Verfahren $ns3d$. Dadurch wird die Präkonditionierung für die linearen Gleichungslöser, s. Kap. 4.5, verbessert, und die linearen Gleichungslöser können die geforderten Konvergenzkriterien für die inneren Iterationen schneller erreichen. Dies kann vor allem bei der Lösung der Druckkorrekturgleichungen, welche eine Poisson-Gleichung darstellt, s. Abschnitt 4.6, erhebliche zeitliche Vorteile bringen.

Ferner wurde die Diskretisierungstechnik vollständig überarbeitet. Gerade die neue Diskretisierungstechnik für die nichtorthogonalen Korrekturen in den diffusiven Flüssen, s. Abs. 4.2.4, sowie die Gradientenberechnung mit der Methode der kleinesten Quadrate, s. Abs. 4.2.2, bringen für die numerische Stabilität erhebliche Vorteile.
Diese Verbesserungen hinsichtlich der numerischen Stabilität werden in der folgenden Tab. 6.6 dargestellt. Für die durchgeführten Simulationen der untersuchten Pumpenreihe mit den unterschiedlichen Diskretisierungsschemata UDS, $MINMOD$ und CDS für die konvektiven Terme ist jeweils eingetragen, ob mit dem Verfahren Konvergenz K, Divergenz D oder ein eingeschwungener Zustand E erreicht wurde.

Für das UDS-Schema wurde für beide Verfahren stets Konvergenz erreicht. Bei den reinen MINMOD-Simulationen konnte mit dem $ns3d$-Verfahren meistens nur ein eingeschwungener Zustand erreicht werden. Bei Verwendung des CDS-Schema divergierte das $ns3d$-Verfahren in drei untersuchten Punkten, in den übrigen zwei konnte keine Konvergenz erreicht werden. Das neue Verfahren liefert auch bei den CDS-Simulationen stets ein verwendbares Ergebnis.

QNS-Simulationen

Im Folgenden wird noch das neue Verfahren hinsichtlich seiner Eignung für QNS-Simulationen getestet. Diese Art der Simulationen werden sehr effizient zur Voroptimierung und für das Initialdesign von Strömungsmaschinen eingesetzt, s. SCHMALHORST [90]. Dabei werden möglichst kurze Rechenzeiten angestrebt. Dies wird erreicht durch Verwendung von sehr groben Rechennetzen, welche für die eigentlich laminaren Rechnungen mit erhöhter effektiver Viskosität geeignet sind,

	UDS		MINMOD		CDS	
n_q	ns3d	s3d	ns3d	s3d	ns3d	s3d
30	K	K	E	K	E	K
40	K	K	E	K	D	K
50	K	K	K	K	D	K
70	K	K	E	K	E	K
90	K	K	E	K	D	E

Tabelle 6.6: Vergleich der numerischen Stabilität der beiden Verfahren *ns3d* und *solver3d* hinsichtlich (K)onvergenz, (E)ingeschwungen und (D)ivergenz.

s. Abs. 6.1.2.1. Da mit diesen Rechnungen nur Abschätzungen für das Initialdesign getroffen werden und um das Verfahren sehr stabil und schnell zu machen, werden die konvektiven Terme mit UDS diskretisiert.

Verglichen werden wiederum die dimensionslosen Druckverläufe an Schaufelpositionen in Nabennähe, in der Mitte und in Deckscheibennähe. Es werden lediglich repräsentative Ergebnisse für den mittleren spezifischen Drehzahlbereich $n_q = 501/min$ dargestellt. Die QNS-UDS-Ergebnisse für verschiedene Turbulenzkonstanten C_T, welche mit den Verfahren *ns3d* und *solver3d* ermittelt wurden, sind im Vergleich mit den ermittelten RANS-UDS-Ergebnissen dargestellt, s. Abb. 6.11.

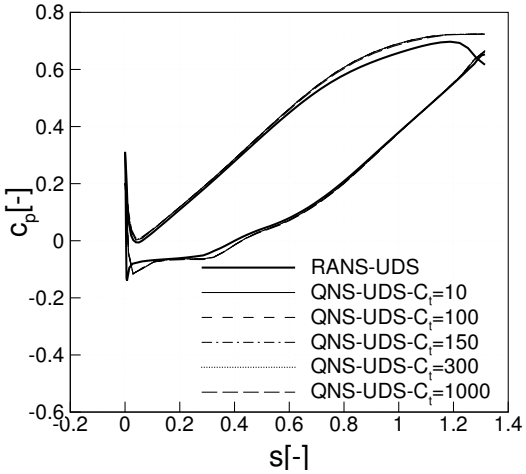

Abbildung 6.11: Dimensionslose Druckverläufe bei Anwendung des QNS-Verfahrens an einem repräsentativen Mittelschnitt der Laufschaufel.

Alle QNS-Verläufe zeigen eine sehr gute Übereinstimmung mit den RANS-Ergebnissen. Es zeigt sich kaum ein Einfluss für die unterschiedlichen Turbulenzkonstanten C_t. Die numerische Stabilität ist für die höheren Turbulenzkonstanten jedoch als deutlich höher zu bewerten. Lediglich im Bereich der Schaufelvorderkante werden die Druck- und Saugspitzen nicht so ausgeprägt wie bei den RANS-Rechnungen abgebildet, was aber auf die verwendeten groben Rechennetze zurückzuführen ist.

Das neue entwickelte Verfahren *solver3d* ist wie sein Vorgängerverfahren für QNS-Rechnungen für den Initialdesign- und Optimierungsprozess geeignet.

6.1.3 Instationärer Testfall

6.1.3.1 Karmansche Wirbelstraße

Zur Untersuchung des instationären Verhaltens des Lösungsverfahrens wurde das Phänomen einer ebenen laminaren Karmanschen Wirbelstraße numerisch untersucht. Dieses Beispiel ist in der Literatur als numerischer Benchmark gut dokumentiert, und die Ergebnisse von PERIC [81] werden als Vergleich herangezogen. Die Karmansche Wirbelstraße bildet sich hierbei durch die Umströmung eines Zylinders in einem Kanal aus. Der Zylinder ist leicht außermittig positioniert, s. Abb. 6.12. Am Einlass wird ein parabolisches Geschwindigkeitsprofil vorgegeben.

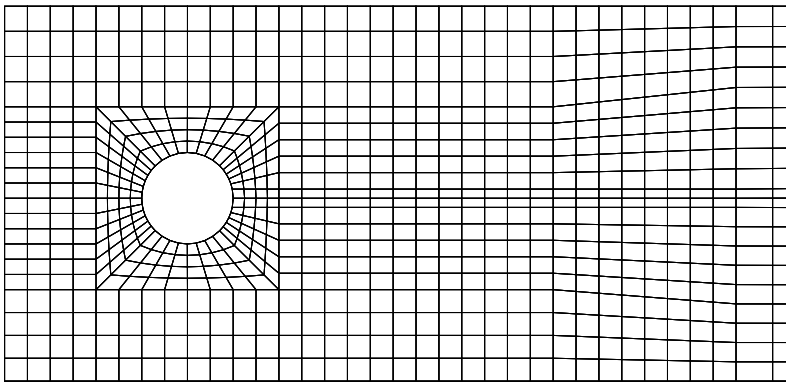

Abbildung 6.12: Vorderer Teil des gröbsten Rechennetzes für den Testfall der Karmanschen Wirbelstraße.

Die auf den Zylinderdurchmesser D und die mittlere Einlassgeschwindigkeit U bezogene Reynolds-Zahl beträgt $Re = 100$. Die numerischen Simulationen wurden auf vier sukzessive verfeinerten, unstrukturierten Netzen mit 1024, 4096, 16384 und 48256 Rechenzellen durchgeführt.

Zur Simulation werden die sequentiellen SIMPLE- und das PISO-Druckkorrekturverfahren verwendet und hinsichtlich ihrer Eigenschaften verglichen. Ferner wurde der Einfluss der zeitlichen Diskretisierungsordnung sowie der Einfluss der Zeitschrittgröße untersucht. Um eine Ablösung und damit die Ausbildung einer Wirbelstraße erhalten zu können, muss räumlich mit einer Genauigkeit zweiter Ordnung diskretisiert werden. Dazu wurde das CDS für die konvektiven Flüsse verwendet. Beim iterativen SIMPLE-Algorithmus wurde dieses Schema mit einer nachgezogenen Korrektur, s. Abs. 4.2.3.3, und einem Blendingfaktor $\alpha = 1,0$ angewendet. Das PISO-Verfahren verwendet keine nachgezogenen Korrekturen, s. FIEREDER [36].

Als Vergleichsgrößen werden die Verläufe des oszillierenden Auftriebsbeiwertes c_A und des Widerstandsbeiwertes c_W herangezogen, die mittels der angreifenden Druck- und Schubspannungen, welche aus den Kräften f_x und f_y resultieren, an den

Zylinderoberfläche ermittelt werden:

$$c_A = \frac{f_y}{\frac{1}{2}\rho U^2}, \tag{6.12}$$

$$c_W = \frac{f_x}{\frac{1}{2}\rho U^2}. \tag{6.13}$$

Charakteristisch für die Problemstellung sind die sich einstellende Ablösefrequenz, die Amplituden und die entsprechenden Mittelwerte. Die Ablösefrequenz f bzw. Umlaufsdauer T ist durch die dimensionslose Strouhal-Zahl beschrieben:

$$Sr = \frac{D}{Tu}, \tag{6.14}$$

welche für diese Problemstellung nach PERIC [81] zum Wert $Sr = 0,3018$ konvergiert. Dieser Wert wird in den durchgeführten Simulationen mit dem $Sr = 0,3125$ auf dem feinsten Netz nur leicht überschätzt.

Die Netzkonvergenz für Auftriebs- und Widerstandsbeiwerte stellt sich ist für die SIMPLE- und die PISO-Berechnung ein, s. Abb. 6.13 .

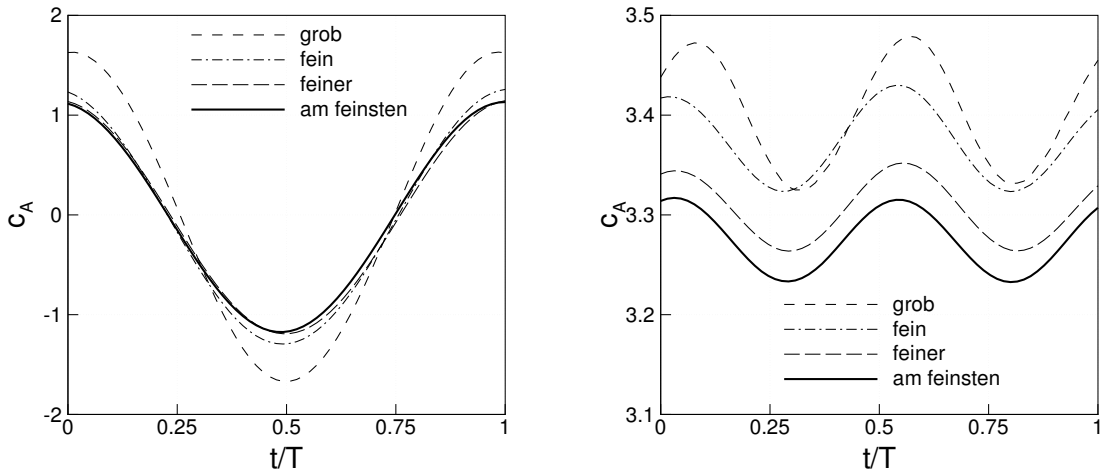

Abbildung 6.13: Netzkonvergenz für Auftriebs- und Widerstandsbeiwerte bei Simulation mit dem PISO-Druckkorrekturverfahren.

Die Reduktion des Fehlers für die ermittelten Auftriebs- und Widerstandsbeiwertie im Vergleich mit der idealen Reduktionsrate ist in Abb. 6.14 dargestellt. Für die Auftriebsbeiwerte ergibt sich fast der für ein Schema zweiter Ordnung ideale Verlauf, für den Widerstandsbeiwert ist der Verlauf bei den gewählten Netzen noch nicht gleichmäßig, im Steigungsverlauf dem idealen Verlauf jedoch sehr ähnlich.

Die Verläufe der Auftriebs- und Widerstandsbeiwerte, welche auf dem feinsten Netz mit dem SIMPLE- und dem PISO-Druckkorrekturverfahren ermittelt wurden, sind vergleichsweise in Abb. 6.15 gegenübergestellt. Der ermittelte Auftriebsbeiwert zeigt nahezu einen identischen Verlauf, der Widerstandsbeiwert wird vom PISO-Verfahren etwas überschätzt.

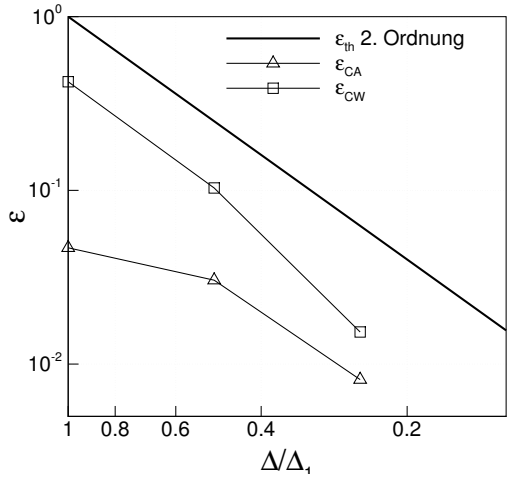

Abbildung 6.14: Fehlerreduktion ϵ über der relativen Netzfeinheit Δ/Δ_1 für die ermittelten Auftriebs- und Widerstandsbeiwerte.

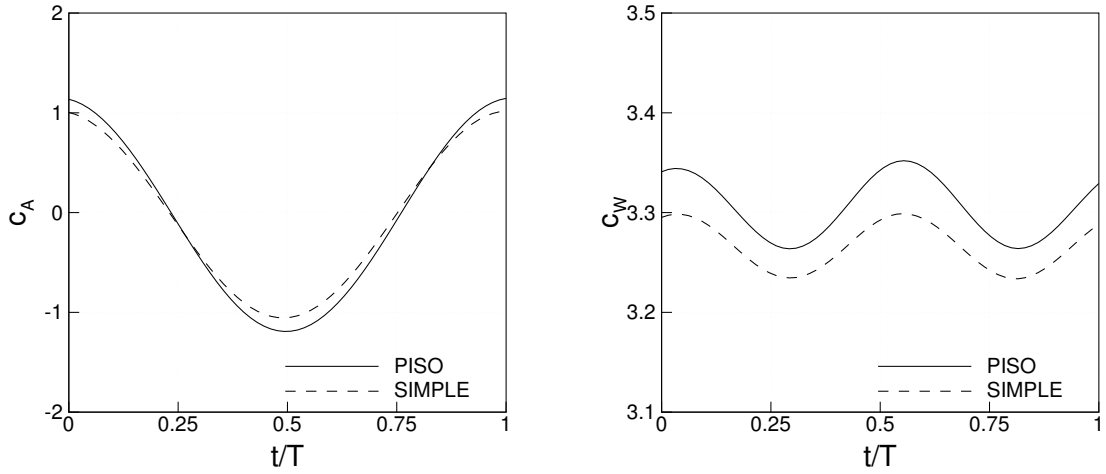

Abbildung 6.15: Ermittelte Auftriebs- und Widerstandsbeiwerte auf dem feinsten Rechennetz für SIMPLE- und PISO-Druckkorrekturverfahren über der normierten Zeit.

Das Verfahren zeigt hinsichtlich instationärer Berechnungen sowohl für zeitliche Diskretisierung erster als auch zweiter Ordnung sehr gute Konvergenzeigenschaften bei sehr guten Ergebnissen. Bei Verwendung kleiner Zeitschritte kann das PISO-Verfahren verwendet werden, welches eine deutliche Beschleunigung des Verfahrens bei instationären Problemstellungen bewirkt. Im Falle des feinsten hier untersuchten Rechennetzes konnte mit dem PISO-Verfahren die Simulation in einem Sechstel der mit dem SIMPLE-Algorithmus benötigten Zeit durchgeführt werden. Effiziente instationäre Strömungssimulationen sind eine wichtige Grundvoraussetzung für die erfolgreiche Simulation von Fluid-Struktur-Interaktionen.

6.2 Strukturmechanik

In den nächsten Abschnitten erfolgt nun die Validierung des implementierten Finite-Volumen-Struktur-Moduls. Es werden Vergleiche mit geeigneten Referenzlösungen dargestellt und werden wiederum stationäre und instationäre Testfälle untersucht. Es wird nur physikalisch lineares Materialverhalten vorrausgesetzt, geometrische Nichtlinearitäten werden in eigenen Abschnitten behandelt. Das Material wird stets als kompressibel angenommen.

Sofern nicht anders vermerkt werden für das numerische Simulationsverfahren die folgenden Einstellungen vorgenommen:

- Alle Testfälle aus der Strukturmechanik wurden mit dem neu entwickelten modularen Lösungsverfahren simuliert.
- Das Konvergenzkriterium k der inneren Iterationen für die Impulsgleichungen wurde stets mit $k < 0,01$ gewählt.
- Das Konvergenzkriterium k der äußeren Iterationen wurde stets mit $k < 10^{-4}$ gewählt.
- Zur Lösung der Impulsgleichungen wurde stets das ICCG-Verfahren verwendet, s. Abs. 4.5.
- Zellflächenwerte werden mit Hilfe der Skewness-Correction nach Gl. 4.10 ermittelt.
- Die Gradientenberechnung erfolgt stets über die Methode der kleinsten Quadrate, s. Gl. 4.13.
- Die diffusiven Flüsse werden mit den nichtorthogonalen Korrekturen nach Gl. 4.23 diskretisiert.
- Für die Simulationen müssen in der Regel keine impliziten Unterrelaxationen angewendet werden.

6.2.1 Stationäre linear-elastische Testfälle

6.2.1.1 Prismatischer Stab unter Eigengewicht

Im ersten vorgestellten Validierungsbeispiel wird ein prismatischer Stab bzw. Balken mit quadratischem Querschnitt und der Länge l unter Eigengewicht untersucht. Für diesen Testfall gibt es analytische Lösungen, s. TIMOSHENKO [97], welche hier zum Vergleich herangezogen werden. Der betrachtete Balken wird an einem Ende fixiert und ist sonst an allen anderen Seitenflächen frei. Der Balken wird lediglich durch den Ortsfaktor g und sein daraus resultierendes Eigengewicht belastet. Für diesen Belastungszustand erhält man für die Verschiebungen die folgenden analytischen Lösungen:

$$u = -\frac{\nu \rho g}{E} xz, \tag{6.15}$$

$$v = -\frac{\nu \rho g}{E} yz, \tag{6.16}$$

$$w = \frac{\rho g}{E} \left[\frac{1}{2} \left(z^2 - l^2 \right) + \nu \left(x^2 + y^2 \right) \right]. \tag{6.17}$$

Bis auf die Spannungskomponente in z-Richtung

$$\sigma_{zz} = \rho g z \tag{6.18}$$

ist der Balken spannungsfrei.

Zur numerischen Untersuchung wurde ein Rechennetz mit $3 \times 3 \times 25$ Volumenzellen verwendet. Die Abweichung der Lösungsgrößen von den analytischen Lösungen liegt bei diesem Beispiel im Bereich des Konvergenzkriteriums, was bei einem Diskretisierungsverfahren zweiter Ordnung und den quadratischen Verläufen der Lösungsgrößen zu erwarten ist.

DEMIRDZIC [31] benötigt zum Erreichen eines Residuenabfalls von drei Größenordnungen noch 35 Iterationen. Mit der implementierten sequentiellen Lösungsmethode und der Verwendung des ICCG-Verfahrens wurde dieses Konvergenzkriterium bereits nach elf Iterationen erreicht.

Repräsentativ wird die Verschiebungsverteilung in Balken-Längsrichtung und die zugehörige Spannungsverteilung in Abb. 6.16 dargestellt.

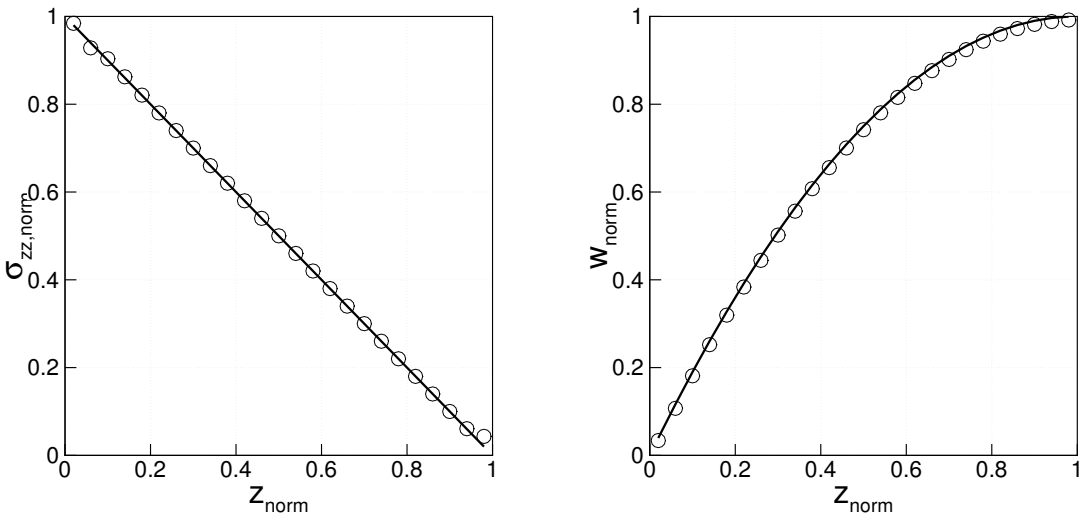

Abbildung 6.16: Simulierte Verschiebungs- und Spannungsverläufe entlang der Balkenachse im Vergleich mit den analytischen Lösungen.

6.2.1.2 Biegung eines beidseitig eingespannten Balkens

Beim zweiten Testfall wird die Biegung eines beidseitig eingespannten Balkens der Länge l unter Gewichtskraft durch den Ortsfaktor g untersucht. Die Belastung erfolgt nun quer zur Balkenlängsachse und wird als Streckenlast $q = \frac{mg}{l}$ modelliert.

Die numerische Simulation wird auf drei sukzessive verfeinerten Netzen durchgeführt. Die Ergebnisse werden mit der analytischen eindimensionalen Balkentheorie verglichen. Die eindimensionale Balkentheorie kann keine dreidimensionalen Effekte wie den Querschub berücksichtigen, so dass Abweichungen der Ergebnisse von der Referenz zu erwarten sind.
Die Biegelinie aus der eindimensionalen Theorie für diesen Belastungsfall lautet:

$$w(x) = \frac{ql^4}{24EI}\left[\left(\frac{x}{l}\right)^2 - 2\left(\frac{x}{l}\right)^3 + \left(\frac{x}{l}\right)^4\right], \tag{6.19}$$

worin I das Flächenträgheitsmoment darstellt, s. z. B. GROSS [43].

Bei der Betrachtung der Lösungskurven, welche für ein Konvergenzkriterium $k < 10^{-4}$ ermittelt wurden, stellt man eine sehr gute Übereinstimmung mit dem analytischen Verlauf fest. Ferner ist die Netzkonvergenz gut erkennbar. Auf dem feinsten Rechennetz wird die analytische Lösung leicht überschätzt, was sich durch numerische Berücksichtigung von dreidimensionalen Effekten erklären lässt.

Das Konvergenzverhalten ist als sehr gut einzustufen. Obwohl das feinste Rechennetz 64-mal mehr Zellen als das gröbste Rechennetz beinhaltet, werden nicht einmal doppelt so viele äußere Iterationen zur Lösung der Problemstellung auf dem feinsten Rechennetz benötigt.

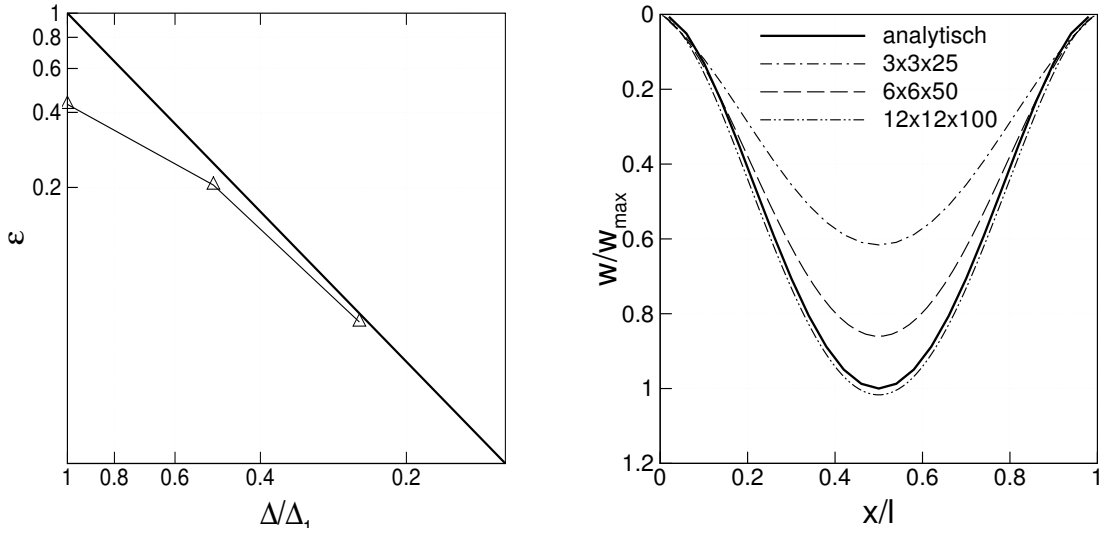

Abbildung 6.17: Simulation der Durchbiegung eines beidseitig eingespannten Balkens: Fehlerreduktionen ϵ über der relativen Netzfeinheit Δ/Δ_1 und Vergleich mit der analytischen Lösung.

6.2.1.3 Biegung eines einseitig eingespannten Balkens

Das dritte Beispiel behandelt einen einseitg eingespannten Balken unter dem Einfluss des Eigengewichtes. Die Belastung erfolgt quer zur Balkenlängsachse. Es wurden die gleichen Netze wie im vorhergehenden Beispiel verwendet. Die numerisch ermittelten Lösungen werden mit der analytischen Lösung aus der eindimensionalen Balkentheorie verglichen:

$$w(x) = \frac{ql^4}{24EI}\left[6\left(\frac{x}{l}\right)^2 - 4\left(\frac{x}{l}\right)^3 + \left(\frac{x}{l}\right)^4\right], \tag{6.20}$$

s. z. B. GROSS [43].

Zunächst wurde für alle drei Netze untersucht, ab welchem Konvergenzkriterium keine Änderung in der Lösung mehr zu erwarten ist. Dieses ist nach Abb. 6.18 für alle Netze bei einem Residuenabfall von 5 Größenordnungen zu erwarten. Der große Unterschied in den Lösungen bei den unterschiedlichen Konvergenzkriterien von 10^{-3} und 10^{-4} zeigt, dass ein Residuenabfall von 3 Größenordnungen für den sequentiellen Lösungsprozess bei der Untersuchung von linear-elastischen Problemstellungen zu niedrig ist. Das vorhergehende Beispiel in Abschnitt 6.2.1.1 stellt eine Ausnahme dar. Netzkonvergenz stellt sich ein, auf dem feinsten Netz erhält man ein gute Übereinstimmung mit der analytischen Lösung. Das Konvergenzverhalten bei dieser Belastungsart

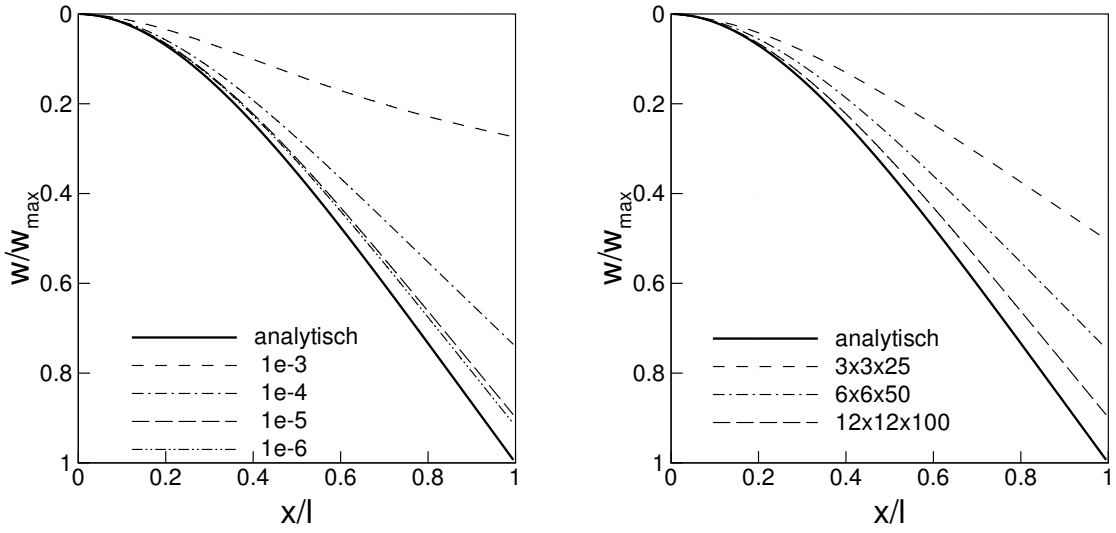

Abbildung 6.18: Simulation der normierten Durchbiegung eines einseitig eingespannten Balkens, links die numerisch ermittelten Lösungen auf dem feinsten Netz für verschiedene Konvergenzkriterien, rechts Darstellung der Netzkonvergenz für das Konvergenzkriterium $k < 10^{-5}$ im Vergleich mit der analytischen Lösung.

als mäßig einzustufen. Mit der sequentiellen Lösungsmethode werden sehr viele äußere Iterationsschritte zum Erreichen des Konvergenzkriterium benötigt. Dieses Verhalten wurde auch bei weiteren Testfällen, z.B. dünnwandigen Strukturen unter Querbelastung, festgestellt. In Abschnitt 6.2.1.7 wird eine Methode zur Verbesserung des Konvergenzverhaltens vorgeschlagen, besprochen und angewendet.

6.2.1.4 Dickwandiges Rohr unter Innendruck

Im Folgenden wird ein dickwandiges Rohr unter Innendruck p_{p0} simuliert. Das Rohr ist beschrieben durch seinen Innen- und Außenradius r_i und r_a, wobei das Verhältnis $\frac{r_a}{r_i} = 2$ gewählt wurde. Die Ausdehnung in Achsrichtung des Rohres sei unendlich, so dass eine zweidimensionale Betrachtung genügt.

Aufgrund der Rotationssymmetrie der Problemstellung wurde nur ein Sektor mit Winkel $\alpha = 60°$ vernetzt, dessen begrenzende Berandungen eine reibungsfreie Einspannung repräsentieren. Diese Einspannung wird in der implementieren FVM einfach durch eine Symmetrie-Randbedingungen, s. Abb. 6.19, modelliert, wobei lediglich Verschiebungskomponenten in tangentialer Richtung zugelassen werden.

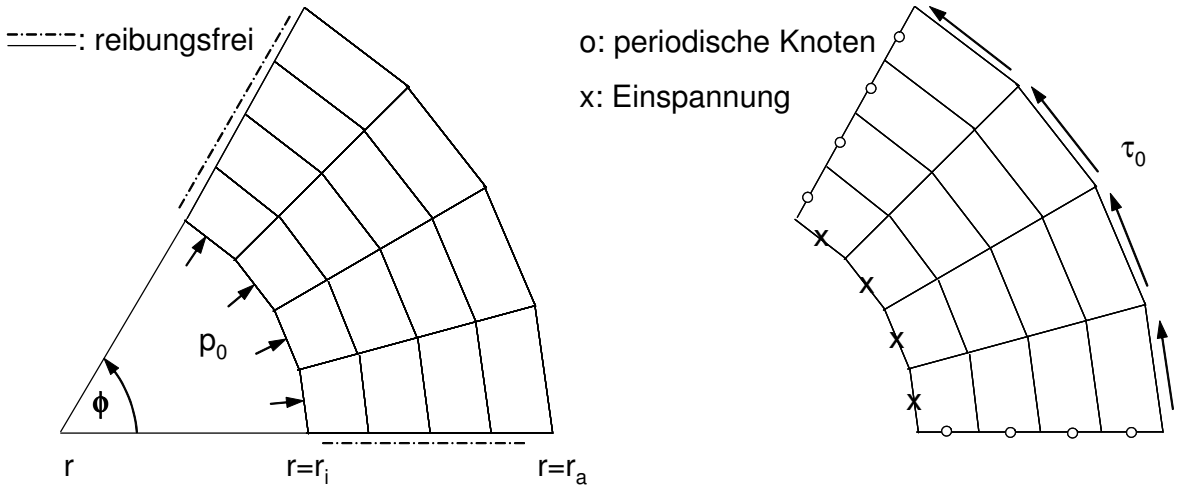

Abbildung 6.19: Geometrie, Rechennetz und Randbedingungen zur numerischen Simulation eines dickwandigen Rohres unter Innendruck (links) sowie unter Torsionslast (rechts).

Die numerischen Simulationen wurden auf drei sukzessive verfeinerten Netzen mit $8 \times 8 \times 1$, $16 \times 16 \times 1$ und $32 \times 32 \times 1$ Zellen durchgeführt. Das gröbste Rechennetz und die Randbedingungen für die Simulation sind in Abb. 6.19 dargestellt.

Auch für diesen Testfall existieren analytische Lösungen sowohl für die radiale Verschiebungsverteilung

$$u_r(r) = \frac{p_0}{E} r \frac{r_i^2}{r_a^2 - r_i^2} \left[(1-\nu) + \left(\frac{r_a}{r}\right)^2 (1+\nu) \right] \tag{6.21}$$

als auch für die radiale Spannnungsverteilung σ_{rr} und die azimutale Spannnungsverteilung $\sigma_{\phi\phi}$

$$\sigma_{rr}(r) = -p_0 r \frac{r_i^2}{r_a^2 - r_i^2} \left[\left(\frac{r_a}{r}\right)^2 - 1 \right], \tag{6.22}$$

bzw:

$$\sigma_{\phi\phi}(r) = +p_0 r \frac{r_i^2}{r_a^2 - r_i^2} \left[\left(\frac{r_a}{r}\right)^2 + 1 \right], \tag{6.23}$$

welche zum Vergleich herangezogen werden, s. GROSS [44].

Die Vergleiche der mit $u_0 = \frac{p_0}{E} r \frac{r_i^2}{r_a^2 - r_i^2}$ normierten Verschiebungen und der mit $\sigma_0 = u_0 \frac{E}{r}$ Spannungen sind für einen Schnitt $\phi = konst.$ in den folgenden Abb. 6.20 und 6.21 dargestellt. Die Ergebnisse zeigen die erwartete Netzkonvergenz. Ferner erkennt man, dass die analytische Lösung schon für das gröbste verwendete Gitter sehr gut getroffen wird. Zwischen mittlerem und feinstem Netz sind nur Unterschiede

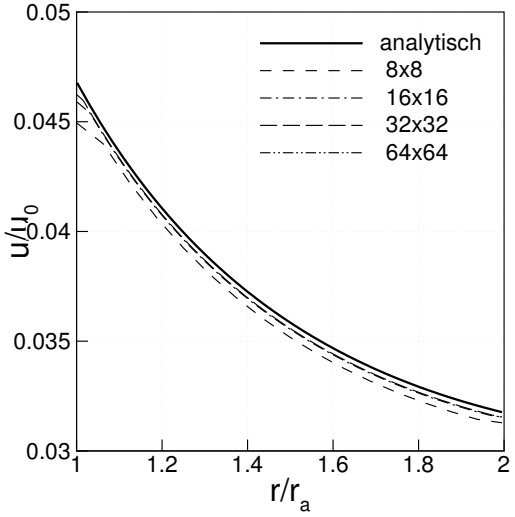

Abbildung 6.20: Vergleich von Verschiebungen $u_r(r)$ an einem Schnitt $\phi = konst$ mit der analytischen Lösung.

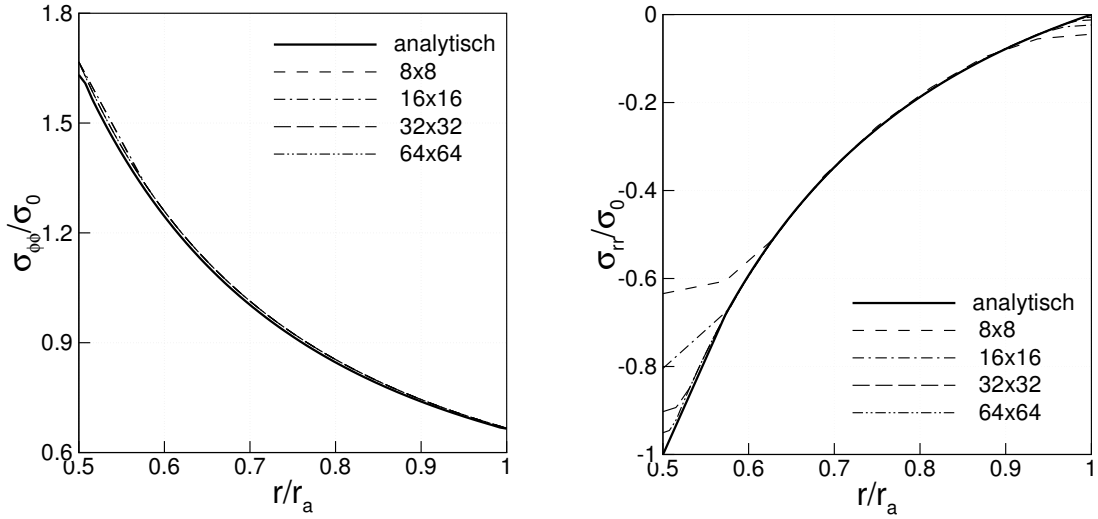

Abbildung 6.21: Vergleich von Spannungen $\sigma_{\phi\phi}$ und σ_{rr} an einem Schnitt $\phi = konst.$ mit der analytischen Lösung.

in Strichstärke erkennbar. Ein kritischer Punkt sind die Spannungen an Innen- und Außenrand, wo die Übereinstimmung mit der analytischen Lösung erst auf dem feinsten Netz gut getroffen wird. Dies könnte aber in einfacher Weise mit Hilfe eines zu den Rändern verfeinerten Netzes erreicht werden.

6.2.1.5 Dickwandiges Rohr unter Torsionslast

In diesem Testfall wird ein dickwandiges Rohr mit identischen Abmessungen wie beim Testfall im vorherigen Abschnitt - doch nun unter Torsionslast - untersucht. Das Rohr ist an seinem Innenrand fixiert. Am Außenrand wird die Torsionslast τ_0 in Form einer tangential wirkenden Kraft aufgebracht. Wiederum wird nur ein Sektor mit $\alpha = 60°$ untersucht. Die Berandungen werden als periodisches Interface modelliert, s. Abb. 6.19.

Die numerischen Simulationen wurden auf drei sukzessive verfeinerten Netzen mit $8 \times 8 \times 1$, $16 \times 16 \times 1$ und $32 \times 32 \times 1$ Zellen durchgeführt. Das gröbste Rechennetz und die Randbedingungen für die Simulation sind in Abb. 6.19 dargestellt.

Für diesen Testfall existieren analytische Lösungen sowohl für die azimutalen Verschiebungen

$$u_\phi(r) = \frac{\tau_0 r_a^2}{2Gr_i}\left[\frac{r}{r_i} - \frac{r_i}{r}\right] \tag{6.24}$$

als auch für die Schubspannungen

$$\sigma_{r\phi}(r) = \tau_0 \left(\frac{r_a}{r_i}\right)^2, \tag{6.25}$$

s. z. B. GROSS [44].

Die numerisch simulierten und mit $u_0 = \frac{\tau_0 r_a^2}{2Gr_i}$ normierten azimutalen Verschiebungen und die mit τ_0 normierten Schubspannungen sind für einen Schnitt $\phi = konst.$ in Abb. 6.22 dargestellt. Die Ergebnisse zeigen die erwartete Netzkonvergenz. Ferner erkennt man, dass die analytische Lösung schon für das gröbste verwendete Gitter sehr gut

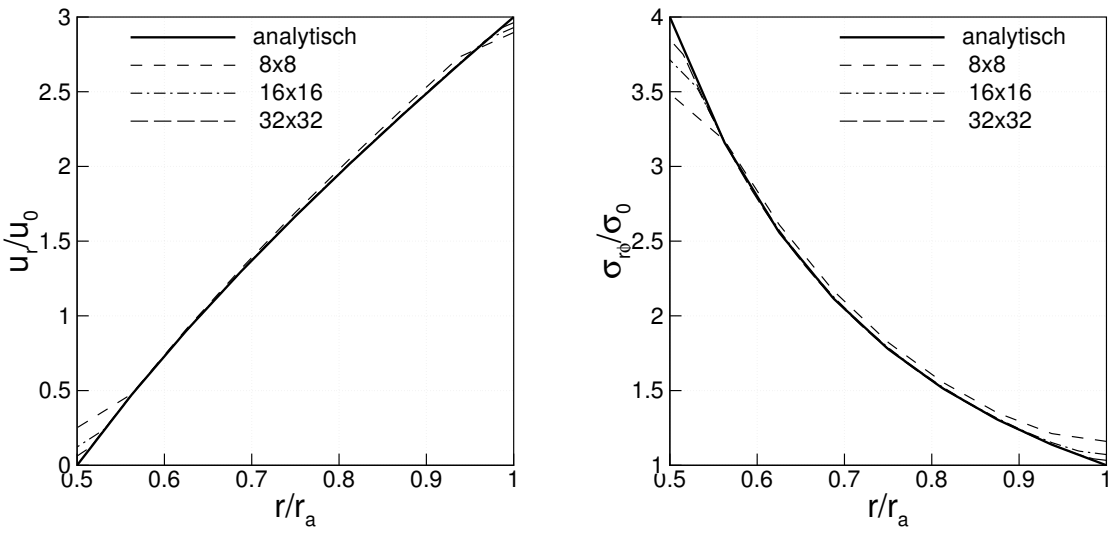

Abbildung 6.22: Vergleich von Verschiebungen $u_\phi(r)$ und Spannungen $\sigma_{r\phi}$ an einem Schnitt $\phi = konst.$ mit der analytischen Lösung.

getroffen wird.

Zwischen mittlerem und feinstem Netz sind nur noch Unterschiede in Strichstärke

erkennbar. Kritische Punkte sind die Spannungsränder an Innen- und Außenrand, wo die Übereinstimmung mit der analytischen Lösung erst auf dem feinsten Netz gut getroffen wird. Dies könnte in einfacher Weise mit Hilfe eines zu den Rändern hin verfeinerten Netzes erreicht werden. Mit einem so verfeinerten Netz reicht dann eine Simulation schon für die gröbste Auflösung aus.

Auch EINZINGER [34] untersucht diesen Fall mit einem Finite-Elemente-Verfahren und verwendet zur Modellierung lineare und quadratische, verschiebungsbasierte Elemente. In seinen Ergebnissen erkennt man große Abweichungen in Verschiebungen und Spannungen zum Außenrand hin. Dieses Phänomen ist in den Finite-Elemente-Verfahren als *Locking* bekannt und tritt bei dem vorgestellten Finite-Volumen-Verfahren nicht auf.

6.2.1.6 Gelochte Platte unter Zug

Der folgende Testfall zeigt die Spannungsverteilung in einer gelochten Platte unter Zug. Die theoretisch unendlich ausgedehnte Platte wird im Unendlichen mit einer Zugspannung f_x an ihren Enden beaufschlagt. Der Lochradius wird mit a bezeichnet. In der numerischen Simulation wird nur ein Ausschnitt der Platte untersucht und gleichzeitig die Symmetrie der Problemstellung ausgenutzt, s. Abb. 6.23.

Die numerischen Simulationen wurden auf drei sukzessive verfeinerten Netzen mit $16 \times 16 \times 1$, $32 \times 32 \times 1$ und $64 \times 64 \times 1$ Zellen durchgeführt. Zusätzlich wurden Lösungen auf einem triangulierten und einem polyedrischen Netz berechnet. Das triangulierte Netz ist in etwa so fein wie das feinste strukturierte Netz gewählt. Das polyedrische Netz wurde auf der Basis des triangulierten Netzes generiert und besitzt nur noch etwa ein Viertel der Anzahl der Zellen des triangulierten Netzes. Die beiden unstrukturierten Netze sind in Abb. 6.24 dargestellt.
Für diesen Testfall existieren analytische Lösungen für die Spannungsverläufe in radialer Richtung σ_{rr}, azimutaler Richtung $\sigma_{\phi\phi}$ sowie für die Schubspannungen $\sigma_{r\phi}$, s. GROSS [44], welche auch in kartesischen Koordinaten dargestellt werden können:

$$\sigma_{xx} = f_x \left[1 - \frac{a^2}{r^2} \left(\frac{3}{2} cos2\phi + cos4\phi \right) + \frac{3}{2} \frac{a^4}{r^4} cos\phi \right] \tag{6.26}$$

bzw.:

$$\sigma_{xy} = f_x \left[-\frac{a^2}{r^2} \left(\frac{1}{2} cos2\phi - cos4\phi \right) - \frac{3}{2} \frac{a^4}{r^4} cos4\phi \right] \tag{6.27}$$

bzw.:

$$\sigma_{yy} = f_x \left[-\frac{a^2}{r^2} \left(\frac{1}{2} sin2\phi + sin4\phi \right) + \frac{3}{2} \frac{a^4}{r^4} sin4\phi \right]. \tag{6.28}$$

Für alle mit f_x normierten Spannungskomponenten σ_{xx}, σ_{xy} und σ_{yy} zeigen sich gute bis sehr gute Übereinstimmungen der numerisch ermittelten Verläufe mit den analytisch berechneten. Auf den sukzessive verfeinerten Netzen ist für alle Spannungskomponenten Netzkonvergenz eindeutig erkennbar. Die Ordnung des Verfahrens wird beim feldweisen Vergleich mit den analytischen Lösungen für alle Spannungskomponenten bestätigt und die Kurven der Fehlerreduktion gehen asymptotisch gegen den erwarteten idealen Verlauf, s. Abb. 6.25. rößere Abweichungen von der Referenzlösung ergeben sich

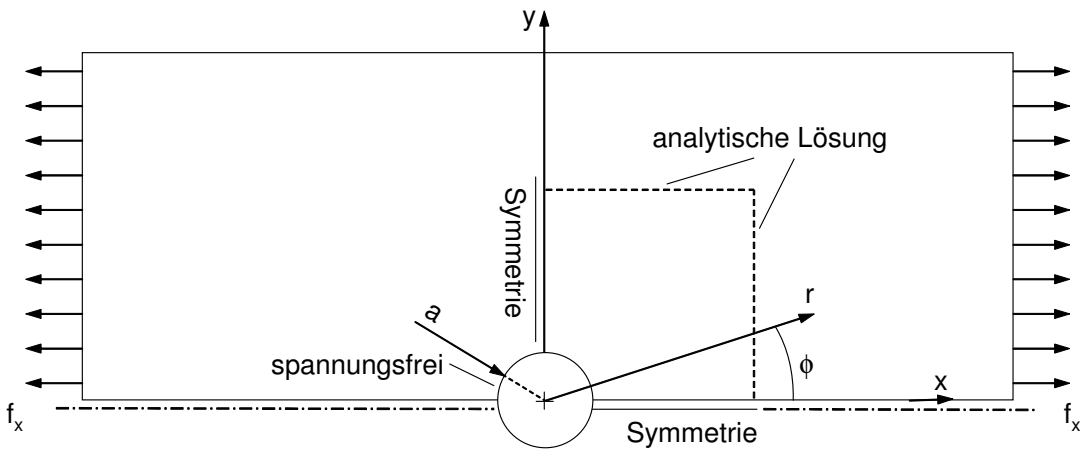

Abbildung 6.23: Geometrie der untersuchten Lochplatte mit Randbedingungen für die Simulation einer gelochten Platte unter Zug.

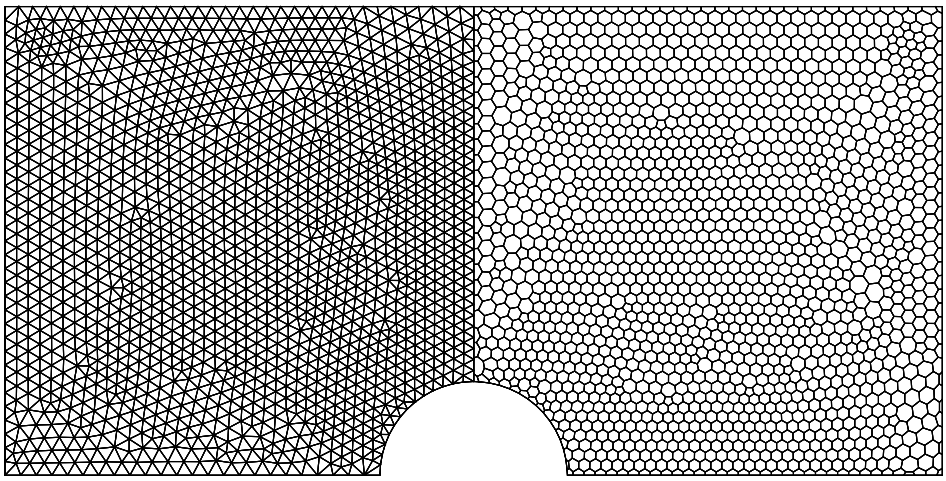

Abbildung 6.24: Unstrukturierte Vernetzungsstrategien für die Simulation einer gelochten Platte unter Zug.

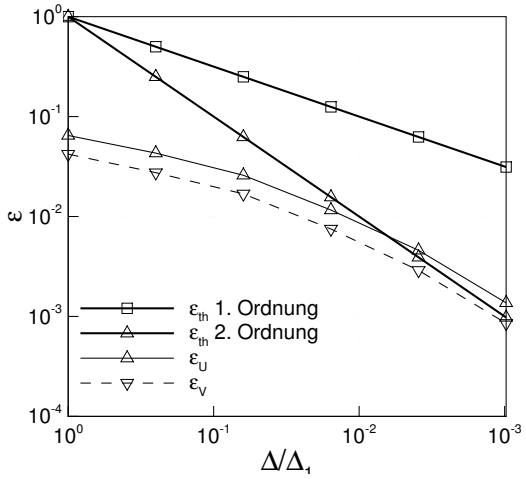

Abbildung 6.25: Fehlerreduktionen ϵ über der relativen Netzfeinheit Δ/Δ_1 für die Spannungskomponenten σ_{xx}, σ_{xy} und σ_{xz} bei sukzessive verfeinerten Rechennetzen.

nur in Nähe des Plattenlochs. Diese Abweichungen können in einfacher Weise durch eine adaptive Netzverfeinerung zum Rechengebietsrand vermieden werden, welche im Rahmen dieser Arbeit nicht durchgeführt wurde.

Auch bei den auf den beiden unstrukturierten Netzen berechneten Lösungen gelten die oben angeführten Aussagen. Die repräsentativ dargestellt Verläufe der Spannungskomponenten zeigen untereinander und im Vergleich mit den Kurven, die auf dem strukturierten Rechennetz ermittelt wurden, keine nennenswerte Unterschiede.

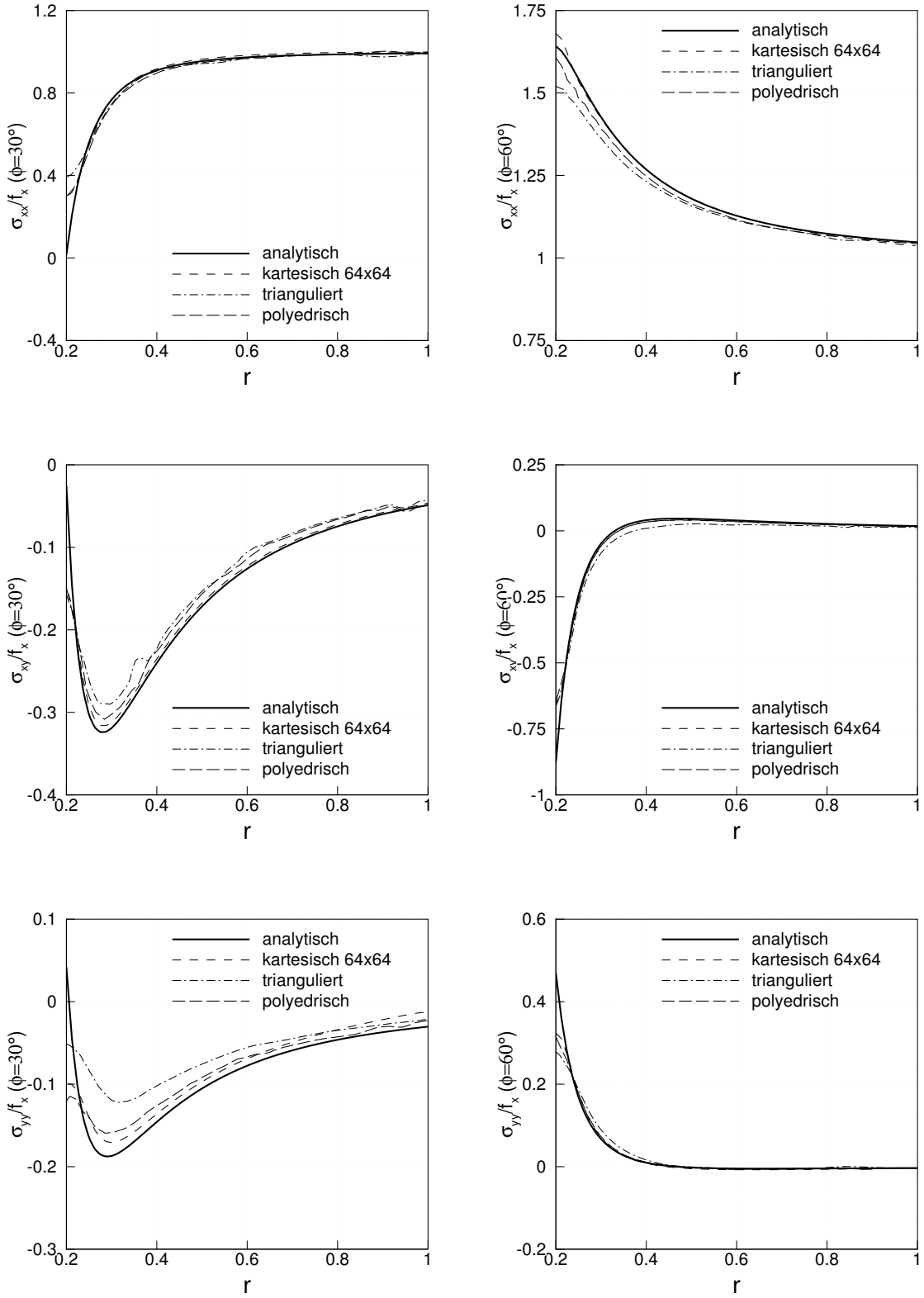

Abbildung 6.26: Verlauf der normierten Spannungskomponente σ_{xx} (oben), σ_{xy} (mitte), σ_{xx} (unten), an Schnitten $\phi = \frac{\pi}{6}$ (links) und $\phi = \frac{\pi}{3}$ (rechts). Die dargestellten Ergebnisse wurden bei Simulationen auf strukturierten und unstrukturierten Rechennetzen durchgeführt und werden hier mit den analytischen Lösungen verglichen.

6.2.1.7 Effizienzsteigerung durch Mehrgitterverfahren

In Abschnitt 4.6.2.4 wurden Schwächen im Konvergenzverhalten des sequentiellen Lösungsverfahrens für die Strukturgleichungen diskutiert und Möglichkeiten zur Behebung aufgezeigt. Obwohl bei allen bisher vorgestellten Testbeispielen Konvergenz ohne Unterrelaxation oder andere Maßnahmen zur Stabilisierungs erreicht wurde, wird der sequentielle Lösungsprozess bei speziellen Problemstellungen wie der Balkenbiegung, welche in Abschnitt 6.2.1.3 vorgestellt wurde, sehr langwierig.

Bei diesen Problemstellungen wird durch die Verwendung des implementierten Mehrgitterverfahrens nach Abs. 4.5.3.2 das Konvergenzverhalten stark beschleunigt. Dieses Verfahren wird nun beispielhaft auf das Balkenproblem aus Abs. 6.2.1.3 angewendet.

Es wurden Simulationen für drei Feingitterdiskretisierungen mit $3 \times 3 \times 25$, $6 \times 6 \times 50$ und $12 \times 12 \times 100$ Zellen durchgeführt. Die Ergebnisse der Simulationen decken sich mit Abschnitt 6.2.1.3 und werden nicht mehr gezeigt. Zur Lösung wurde das Mehrgitterverfahren mit einem V-Zyklus eingesetzt. Auf dem feinsten Rechennetz wird zunächst eine äußere Iteration durchgeführt. In der Restriktionsphase werden jeweils drei (äußere) Iterationen, während der Prolongationsphase vier (äußere) Iterationen vorgenommen.

Es wurden Rechnungen, die auf geometrischen und algebraisch unstrukturierten Vergröberungen nach Abschnitt Abs. 4.5.3.1 basieren, durchgeführt. Die geometrische Vergröberung ist bei unstrukturierten Netzen nur möglich, falls diese topologisch auf ein strukturiertes Netz zurückgeführt werden können und die Volumenzellen entsprechend nummeriert wurden. Deswegen ist diese Art der Vergröberung für die spätere praktische Anwendungen nur bedingt möglich. Bei der algebraischen Vergröberung wurde die minimale Blockgröße auf vier, die maximale auf acht Rechenzellen festgelegt.

Der Unterschied im Konvergenzverhalten bei Anwendung dieser unterschiedlichen Vergröberungen zeigt Abb. 6.27. Repräsentativ werden nur die Residuenabfälle der Verschiebungen, welche sich aus Rechnungen auf den feinsten Gittern ergaben, dargestellt. Man erkennt hier für beide Strategien einen glatten und gleichmäßigen Abfall der Residuen, welcher bis zu einer Größe von 10^{-8} abgebildet wird. Gute Ergebnisse sind nach Abs.6.2.1.3 schon bei einer Größe von 10^{-5} zu erwarten, weswegen der Iterationsprozess deutlich früher abgebrochen werden kann. Durch die gleichmäßige und regelmäßige geometrische Vergröberung kann die Lösung schneller erreicht werden als bei Verwendung des unstrukturierten Vergröberungsalgorithmus. Der Unterschied im Konvergenzverhalten von der herkömmlichen sequentiellen Lösungsmethode und der Mehrgitter-Lösungsmethode unter Verwendung des unstrukturierten Vergröberungsalgorithmus ist ebenfalls in Abb. 6.27 dargestellt. Das Mehrgitterverfahren erreicht hier - die Anzahl der Iterationen betreffend - um Größenordnungen schneller das Konvergenzkriterium. Da eine äußere Iteration in den Mehrgitterverfahren aufgrund der Restriktions-, Prolongations- und Relaxationsoperationen aufwändiger ist als im sequentiellen Lösungsverfahren, werden die normierten Lösungszeiten für die unterschiedlichen Lösungsverfahren in Tabelle 6.7 zusammengestellt. Zeitlich sind die Mehrgitterverfahren dem herkömmlichen Lösungsverfahren deutlich überlegen.

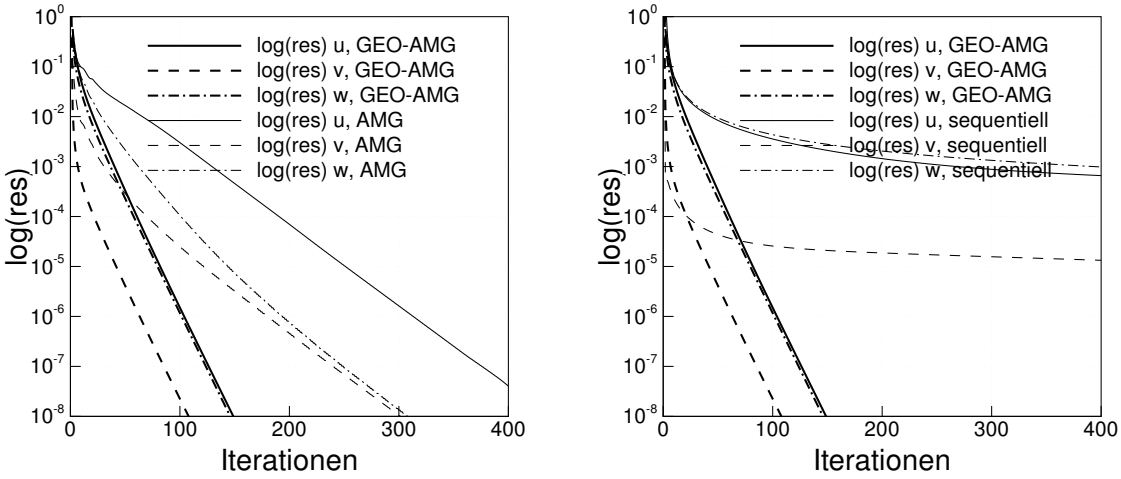

Abbildung 6.27: Vergleich der Residuenabfälle für die Verschiebungskomponenten u, v und w für das Balkenproblem. Links das sequentielle Lösungsverfahren im Vergleich mit dem Mehrgitterverfahren bei idealer geometrischer Vergröberungsstrategie. Rechts die Verläufe bei Anwendung des Mehrgitterverfahrens unter Verwendung der idealen und der algebraischen Vegröberungsstrategie.

N	N/N_1	i_S	i_G	i_A	$t_G/t_{G,1}$	$t_A/t_{A,1}$	t_A/t_G	t_S/t_G	t_S/t_A
225	1	9848	63	94	$1,0$	$1,0$	$2,0$	$26,0$	$13,0$
1800	8	> 10000	84	190	$6,0$	$3,1$	$1,5$	$39,8$	$26,6$
14400	64	> 10000	149	289	$55,0$	$42,0$	$2,2$	$52,8$	$54,4$

Tabelle 6.7: Vergleich der sequentiellen (S) Lösungsmethode mit dem Mehrgitterverfahren mit geometrischer Vergröberung (G) und algebraisch unstrukturierter Vergröberung (A) hinsichtlich Iterationen i und Lösungszeiten t bei der Simulation eines einseitig eingespannten Balkens mit drei unterschiedlichen Feingitterdiskretisierungen.

Für die Berechnungen wurden noch keinerlei Optimierungen hinsichtlich der unstrukturierten Vergröberungsstrategie, des Relaxationsschemas, der Anzahl der Glätteriterationen oder der Art des Zyklus durchgeführt, wodurch mit Sicherheit die Leistungsfähigkeit des Verfahrens noch deutlich gesteigert werden kann.

In diesem Abschnitt konnte gezeigt werden, dass die implentierte FVM auch für stationäre, linear-elastische Problemstellungen mit dünnwandigen, langgestreckten Konstruktionselemente, welche mit den Finite-Volumen-Zellen sehr fein diskretisiert werden müssen, effizient angewendet werden kann. Dies ist eine Grundvoraussetzung, um spezialisierte Elementtypen wie Stäbe, Balken oder Schalen vermeiden zu können.

6.2.2 Stationäre geometrisch-nichtlineare Testfälle

6.2.2.1 Nichtlineare Verformung eines eingespannten Balkens

Bei diesem Testfall wird die geometrisch nichtlineare Verformung eines einseitig eingespannten Balkens der Länge l und der Querschnittsfläche A unter Endlast F untersucht. Dabei werden die inkrementellen Lagrange-Formulierungen, die TLF und die ULF, angewendet und die ermittelte Lösungen verglichen. Analytische Lösungen für die Verschiebungen der Balkenspitze existieren für diesen Testfall nur in Form elliptischer Integrale, welche numerisch ausgewertet werden müssen, s. MATTIASON [70] und als Vergleichswerte herangezogen werden. Die geometrischen Abmessungen und Materialdaten sind in der folgenden Tabelle 6.8 zusammengefasst.

$l\ [m]$	$2,0$
$A\ [m^2]$	$0,04$
$E\ [N/m^2]$	$15,293 \cdot 10^6$
$\rho\ [kg/m^3]$	$1000,0$
$\nu\ [-]$	$0,3$

Tabelle 6.8: Geometrie und Materialeigenschaften für nichtlinear verformten Balken

Die Werte wurden so gewählt, dass die erste natürliche Eigenfrequenz $1Hz$ beträgt.

Die numerische Simulation erfolgt wiederum auf drei sukzessive verfeinerten Rechennetzen mit $3 \times 3 \times 30$, $6 \times 6 \times 60$ und $12 \times 12 \times 120$ Volumenzellen. Die inkrementelle Lösungsweise wird instationär durchgeführt werden. Um schnell zu einem stationären Endzustand zu gelangen, wurde das implizite Eulerverfahren erster Ordnung zur zeitlichen Diskretisierung verwendet. TUKOVIC [98] schlägt zur Beschleunigung des Verfahrens die Einführung einer künstlichen Dämpfung D in Form eines zusätzlichen Zeitterms vor

$$D = \int_{dV} K \frac{\partial \mathbf{v}}{\partial t} dV, \tag{6.29}$$

wobei die Dämpfungskonstante K prinzipiell beliebig hoch gewählt werden kann, da der Einfluss der Dämpfung bei Erreichen des stationären Endzustandes verschwindet. Um keine Instabilität des Verfahrens herbeizuführen, wurde diese etwa in der Größenordnung des Elastizitätsmoduls gewählt.

Der untersuchte und verformte Balken mit den verformten Rechennetzen ist in Abb. 6.28 für verschiedene Belastungszustände dargestellt.

Die ermittelten Absolutbeträge der Maximalauslenkungen der Balkenspitze sind in den Diagrammen 6.29 für verschiedene Belastungen F_{norm} dargestellt, wobei die Normierung der aufgebrachten Last nach $F_{norm} = \frac{Fl^2}{EI}$ erfolgt.

Man erkennt selbst bei extremen Verformungszuständen gute bis sehr gute Übereinstimmung mit den Referenzlösungen. Stabilität kann jedoch bei großen Verformungen und unter Anwendung der Fortschreitenden-Lagrange-Formulierung nur durch eine

genauen Interpolation der Lösungsgrößen an die Netzknoten und anschließende Netzdeformation erreicht werden, s. Abs. 4.6.3.2. Beim Vergleich der beiden implementierten Lagrange-Formulierungen erkennt man, dass Lösungen für die TLF generell niedriger als für die FLF, bzw. ULF ausfallen. Ursachen hierfür wurden im Rahmen dieser Arbeit nicht gefunden.

Bei den TLF-Simulationen mussten die Impulsgleichungen mit einem Faktor $\alpha = 0,8$ unterrelaxiert werden, um für alle Belastungsfälle Lösungen erhalten zu können. Zusätzliche Anmerkungen zur inkrementellen, instationären Lösungsprozedur finden sich in Abschnitt 6.2.4.1.

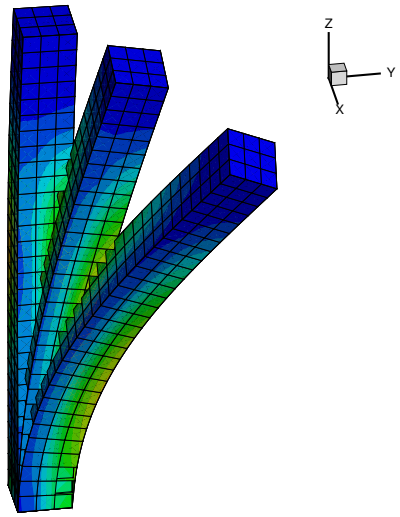

Abbildung 6.28: Auslenkungen für die nichtlineare Verformung mit den zugehörigen Rechennetzen eines eingespannten Balkens für verschiedene Belastungszustände.

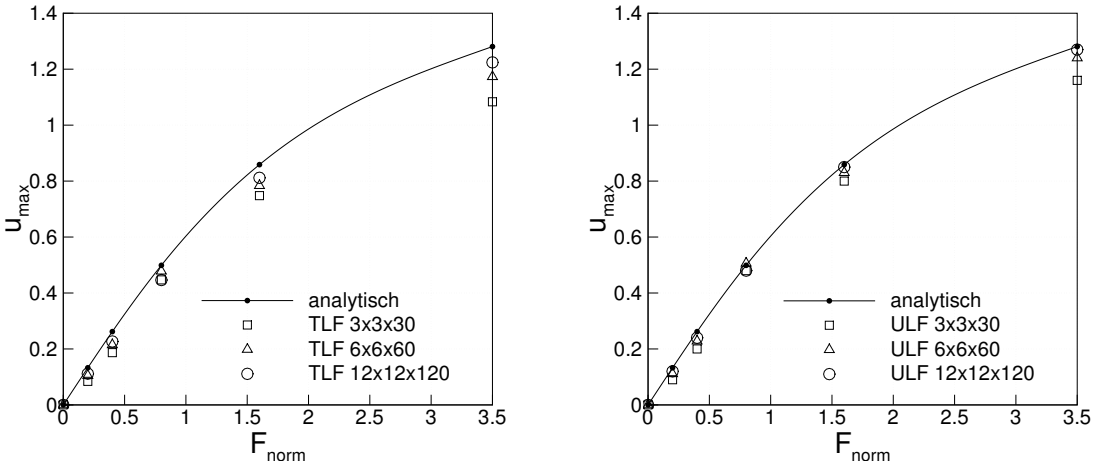

Abbildung 6.29: Auslenkungen für die nichtlineare Verformung eines eingespannten Balkens für verschiedene normierte Belastungszustände.

6.2.3 Instationäre linear-elastische Testfälle

6.2.3.1 Eingespannter Balken unter plötzlicher Belastung

In diesem Abschnitt wird die eindimensionale und zweidimensionale instationäre Wellenausbreitung in einem einseitig eingespannten Balken untersucht. Der Balken ist $10m$ lang und $1m$ hoch. Die Materialeigenschaften des Balkens sind in Tab. 6.9 zusammengefasst.

$E\ [N/m^2]$	$2,1 \cdot 10^{11}$
$\rho\ [kg/m^3]$	$7850,0$
$\nu\ [-]$	$0,3$

Tabelle 6.9: Materialeigenschaften des untersuchten, eingespannten Balkens.

Zum Zeitpunkt $t = 0$ wird am freien Ende eine Verschiebung $u_0 = 10^{-3}m$ aufgebracht. Das Rechengebiet wird dazu mit Rechennetzen mit $100 \times 1 \times 1$ bzw. $100 \times 9 \times 1$ Volumenzellen diskretisiert. Die Ausbreitungsgeschwindigkeit der Wellen im Material wird mit

$$c = \sqrt{\frac{E}{\rho}} \tag{6.30}$$

bestimmt, s. z. B. GROSS [44]. Mit dieser Größe kann einfach ermittelt werden, wann die Welle einen bestimmten Punkt im Material erreicht haben muss und wie groß der Zeitschritt für die instationäre Berechnung zu wählen ist. Über die mit dieser Beziehung und für die Simulationen verwendeten Courant-Zahlen $Co = 0,5$ und $Co = 0,05$ wurden die Zeitschrittgrößen mit $\delta t = 10^{-5}s$ und $\delta t = 10^{-6}s$ bestimmt. Für die Simulationen mit dem kleinen Zeitschritt wurde das Zeitdiskretisierungsverfahren nach 4.3.1 mit einer Genauigkeit erster Ordnung verwendet, für die Simulationen mit dem großen Zeitschritt das Zeitdiskretisierungsverfahren nach 4.3.1 mit einer Genauigkeit zweiter Ordnung. Die Unterschiede in den resultierenden Ergebnissen werden im Folgenden diskutiert.

Zunächst erfolgt die Diskussion der rein eindimensionalen Simulationen. Für eine Stelle $x = 5m$ in der Mitte des Rechengebiets wird in der folgenden Abbildung 6.30 die normierte Verschiebung $u_{norm} = u/u_0$ über der Zeit dargestellt.

Man erkennt bei den Simulationsergebnissen mit den kleinen Zeitschritten für das Zeitdiskretisierungsverfahren erster Ordnung sehr gut den erwarteten Verlauf, kann aber am rechten Abbildungsrand schon eine Dämpfung der Lösung durch numerische Diffusion des Verfahrens erkennen. Für das Zeitdiskretisierungsverfahren zweiter Ordnung erscheint diese Dämpfung geringer, dafür zeigen sich aber stärkere unphysikalische Überschwinger im Verlauf der dargestellten Verschiebungen.

Bei der Verwendung des großen Zeitschritts werden die unphysikalischen Überschwinger für das Zeitdiskretisierungsverfahren erster Ordnung vollkommen vermieden und auch beim Zeitdiskretisierungsverfahren 2. Ordnung fallen diese wesentlich schwächer aus. Beim Zeitdiskretisierungsverfahren erster Ordnung erkennt man jedoch nun eine viel stärkere Dämpfung im globalen Verlauf.

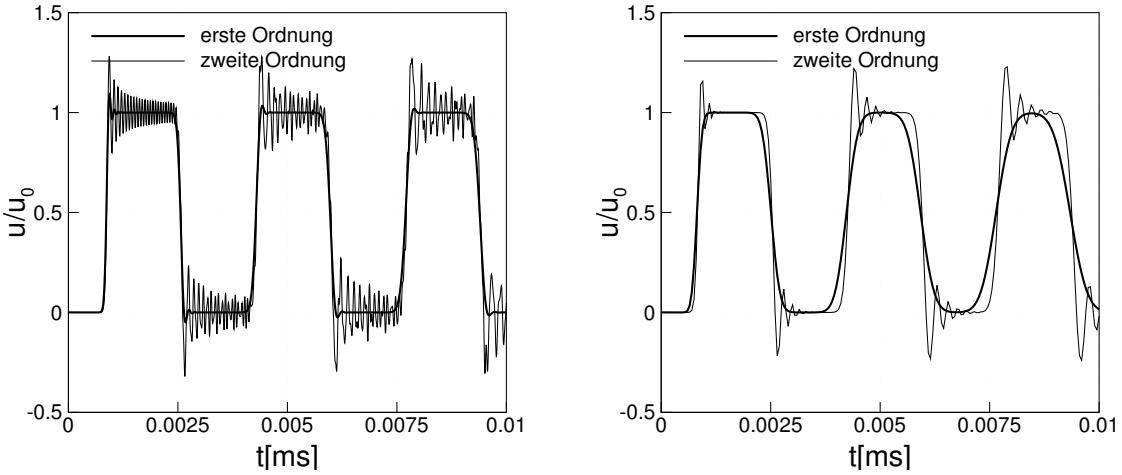

Abbildung 6.30: Eindimensionale, instationäre Wellenausbreitung in einem sprunghaft angeregten, eingespannten Balken: links für eine Courant-Zahl $Co = 0,5$, rechts für $Co = 0,05$.

Die eben getroffenen Aussagen gelten auch für die zweidimensionalen Simulationen, s. Abb. 6.31. Durch die Querkontraktion des Materials treten nun auch Verschiebungen quer zur Hauptausbreitungsrichtung der Materialwelle auf, und die Struktur des zeitlichen Verlaufs der Verschiebungen ist nun deutlich komplexer. Bei Verwendung

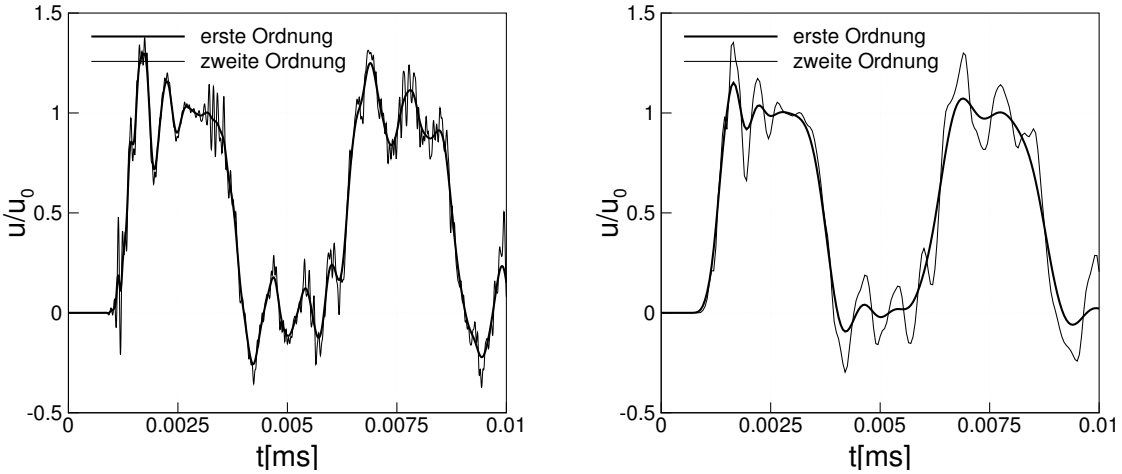

Abbildung 6.31: Zweidimensionale, instationäre Wellenausbreitung in einem sprunghaft angeregten, eingespanntem Balken: links für eine Courant-Zahl $Co = 0,5$, rechts für $Co = 0,05$.

beider Zeitschrittgrößen zeigt sich beim Zeitdiskretisierungsverfahren erster Ordnung wieder die starke Dämpfung der Lösungen zum Abbildungsrand. Dieser Effekt fällt für die Simulationen unter Verwendung des Zeitdiskretisierungsverfahrens zweiter Ordnung zwar schwächer aus, jedoch zeigen sich wiederum typische Überschwinger in den Lösungen. Der deutlich komplexere, zeitliche Verlauf resultiert aus diesen Überschwingern und den deutlich komplexeren physikalischen Effekten, die mit diesen zweidimensionalen Rechnungen abgebildet werden. Neben den Querkontraktionen

und der Wellenreflexion am Balkenende werden die Materialwellen auch an den Seitenrändern der Geometrie gestreut und reflektiert.

Alle durchgeführten Simulationen liefen ohne Anwendung von Unterrelaxationen stabil mit etwa $2 - 5$ Iterationen innerhalb eines Zeitschritts. Die Ergebnisse und die numerischen Eigenschaften des Lösungsverfahrens zeigen plausible Ergebnisse und auch sehr gute Übereinstimmung mit den Ergebnissen von JASAK [60] und MANEERATANA [69].

6.2.4 Instationäre geometrisch-nichtlineare Testfälle:

6.2.4.1 Eingespannter Balken unter plötzlicher Belastung

Im Folgenden wird noch die instationäre Schwingung des einseitg eingespannten Balkens aus Kap. 6.2.2 simuliert. Repräsentativ für alle durchgeführten Simulationen wird nur die Simulation für den größten aufgebrachten normierten Kraftbetrag $F_{norm} = 1.6$ und für die FLF diskutiert. Für alle durchgeführten Simulationen entsprechen die Maximalamplituden den stationär ermittelten Werten.

Bei den durchgeführten Simulationen wurde untersucht, welchen Einfluss das Konvergenzkriterium der äußeren Iterationen innerhalb eines Zeitschritts oder Inkrements auf die Stabilität des Lösungsverhaltens und die Plausibilität der Lösung hat. Für verschiedene Abbruchkriterien ist der Schwingungsverlauf der Balkenspitze über der Zeit in Abb. 6.32 dargestellt. Man erkennt, dass für das Konvergenzkriterium $k < 10^{-3}$ keine stabile Lösung erhalten werden kann. Bei einem Konvergenzkriterium $k < 10^{-4}$ ergibt sich eine stationäre Lösung mit deutlich zu geringer Maximalamplitude. Erst ab einem Konvergenzkriterium $k < 10^{-5}$ erhält man die erwartete Lösung mit einer gleichmäßig ausgeprägten Schwingung. Es ist jedoch am rechten Abbildungsrand noch leicht zu erkennen, dass das inkrementelle Verfahren für dieses Abbruchkriterium numerische Diffusivität zeigt und die Amplitude gedämpft wird. Ab einem Konvergenzkriterium $k < 10^{-6}$ stellt sich schließlich ein absolut gleichmäßiges und ungedämpftes Schwingungsverhalten ein. Wichtig für eine realistische Abbildung des Schwingungsvorgangs ist auch beim inkrementellen Lösungsverfahren die geeignete Wahl der Zeitdiskretisierung. Der Vergleich im Schwingungsverlauf für Simulationen mit einem Zeitdiskretisierungsverfahren erster und zweiter Ordnung ist im Vergleich in Abb. 6.33 dargestellt. Man erkennt beim Zeitdiskretisierungsverfahren 1. Ordnung eine starke Dämpfung durch künstliche numerische Diffusion bereits nach wenigen Schwingungen. Beim Zeitdiskretisierungsverfahren zweiter Ordnung konnte eine derartige Dämpfung auch bei sehr langen Simulationen mit mehreren tausend Zeitschritten nicht festgestellt werden.

Ferner wurde noch der Einfluss der Größe des Zeitschritts beim Konvergenzkriterium $k < 10^{-6}$ untersucht, s. Abb. 6.34. Bei den gewählten Zeitschrittgrößen, welche maximalen Courant-Zahlen, bezogen auf die undeformierte Ausgangsgeometrie, $Co = 0,5$, $Co = 1$, $Co = 2$, $Co = 4$, $Co = 8$ entsprechen, erkennt man beim verwendeten Zeitdiskretisierungsschema zweiter Ordnung nur einen marginalen Einfluss. Bei größeren Zeitschritten konnte keine Konvergenz mehr erreicht werden.

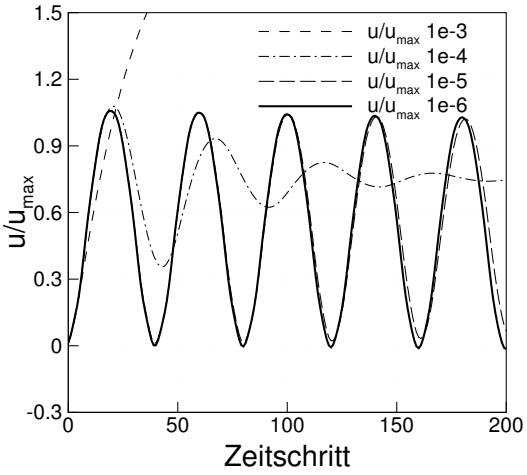

Abbildung 6.32: Schwingung eines sprunghaft angeregten, eingespannten Balkens für verschiedene Konvergenzkriterien.

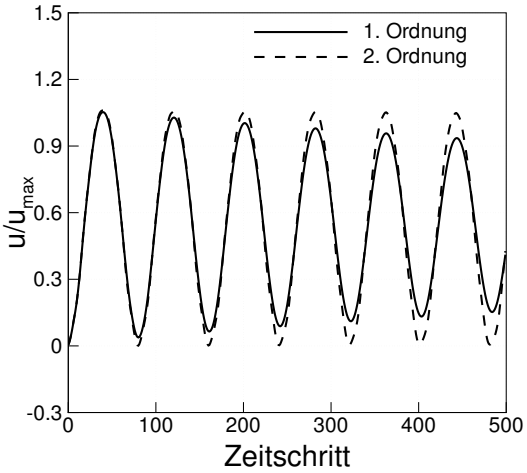

Abbildung 6.33: Schwingung eines sprunghaft angeregten, eingespannten Balkens für unterschiedliche Zeitdiskretisierungen.

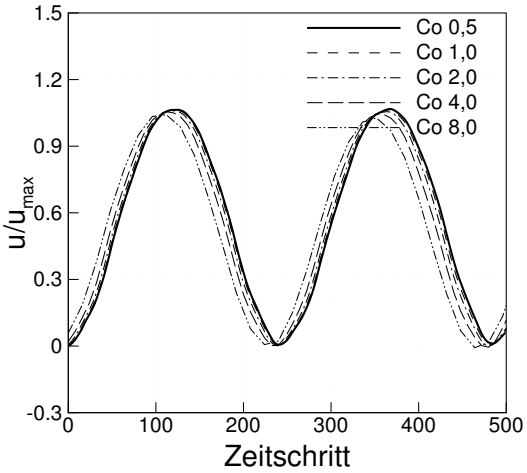

Abbildung 6.34: Schwingung eines sprunghaft angeregten, eingespannten Balkens für unterschiedliche Zeitschrittgrößen.

Es zeigt sich, dass bei den implementierten inkrementellen Lösungsverfahren für die Simulation großer Verformungszustände noch mehr auf eine sinnvolle Wahl von Konvergenzkriterium, Zeitschrittgröße und Zeitdiskretisierungsverfahren geachtet werden muss als bei Simulationen linear-elastischer Problemstellungen. Besonders wichtig für eine stabile Lösungsprozedur ist ferner bei der FLF eine möglichst genaue Interpolation der Lösungsgrößen an die Netzknoten, um eine genaue Netzdeformation auch bei Simulationen über eine lange Zeitdauer zu erhalten.

Mit diesen instationären Testfällen wurde die Eignung des FV-Struktur-Moduls für instationäre Simulationen gezeigt. Das Konvergenzverhalten war bei allen Simulationen als stabil und effizient einzustufen. Instationäre Struktursimulationen sind eine wichtige Grundvoraussetzung für eine erfolgreiche Durchführung von Fluid-Struktur-Interaktionen.

6.2.5 Kritische Diskussion der Finite-Volumen-Methode

Die bisher vorgestellten Beispiele strukturmechanischer Simulationen mit dem entwickelten Finite-Volumen-Verfahren zeigen gute bis sehr gute Übereinstimmungen mit den jeweiligen Referenzdaten. Trotzdem weist die FVM für strukturmechanische Simulationen einige signifikante Nachteile gegenüber der etablierten FEM auf, die sich wie folgt zusammenfassen lassen:

- Die Finite-Elemente-Methode weist einen großen Entwicklungsvorsprung auf und wird deswegen für unterschiedlichste Anwendungsbereiche erfolgreich und zuverlässig angewendet.
- Im Gegensatz zu Finite-Elemente-Methoden können generalisierte Elemente wie Stab, Balken, Platte und Schale nicht ohne erheblichen Aufwand in die vorhandene Datenstruktur und das vorhandene Programmsystem eingearbeitet werden.
- Die Diskretisierungsgenauigkeit kann bei der hinterlegten Diskretisierungstechnik nicht ohne erheblichen Aufwand erhöht werden, und man bleibt auf eine Genauigkeit zweiter Ordnung beschränkt. Hierdurch müssen relativ feine Rechennetze für strukturmechanische Berechnungen mit der FVM verwendet werden.
- Bei der hinterlegten sequentiellen und damit speicherarmen Lösungsmethode können bei stark nichtorthogonalen und verzerrten Rechennetze Instabilitäten festgestellt werden, und es ist oft nur mit Unterrelaxationen Konvergenz erreichbar. Durch die Unterrelaxation wird das Lösungsverfahren langsam.

Neben diesen genannten Nachteilen sind aber auch die folgenden großen Vorteile der FVM gegenüber der FEM für strukturmechanische Simulationen im Kontext des entwickelten modularen Simulationsverfahrens unbedingt zu nennen:

- Das Finite-Volumen-Verfahren zur Strukturberechnung wurde in analoger Weise wie das Verfahren zur Fluidberechnung integriert. Module zur Diskretisierung und zur Lösung der linearen Gleichungssysteme sowie zur übrigen numerischen

Behandlung können für beide Module gleichermaßen verwendet werden. Dadurch entsteht ein sehr kompaktes, effizient wartbares und erweiterbares Programmsystem.

- Die vorgestellte Diskretisierungstechnik bleibt auf ein Genauigkeit zweiter Ordnung beschränkt. Dieser Nachteil kann durch die Verwendung sehr feiner Rechennetze problemlos ausgeglichen werden. Die sequentielle Lösungsmethode ist sehr speicherarm und sehr einfach parallelisierbar, so dass die resultierenden großen Simulationsaufgaben problemlos gelöst werden können.

- Das FVV kann einfach auf unterschiedlichste Materialgesetze, physikalische und geometrische Nichtlinearitäten erweitert und somit einfach auf unterschiedlichste Anwendungsbereiche zugeschnitten werden.

Die FVM kann somit für strukturmechanische Simulationen als sinnvolle Alternative zur FEM gesehen werden. Aufgrund des großen Entwicklungsvorsprungs der FEM wird diese jedoch aller Wahrscheinlichkeit nach weiterhin die primär angewendete Methode bleiben, bzw. es ist zu erwarten, dass die FVM bereichsweise für spezielle Anwendungen der Struktursimulation, bzw. der FSI in Zukunft stärker zum Einsatz kommen wird.

6.3 Fluid-Struktur-Interaktion

6.3.1 Kaplan-Turbine KT192

Im letzten vorgestellten Testfall wird die instationäre Fluid-Struktur-Interaktion in einer Kaplan-Turbine mit der spezifischen Drehzahl $n_{q,opt} = 192,01/min$ beschrieben. Es handelt sich bei der untersuchten Turbine um eine Stufe des Kraftwerks "Dettelbach", welche strömungstechnisch von BÖHM [18] ausführlich untersucht wurde. Für den betrachteten Optimalpunkt sind die Betriebsdaten in der nachfolgenden Tabelle 6.10 zusammengefasst.

Fallhöhe H $[m]$	4,7
Volumenstrom im Optimalpunkt Q_{opt} $[m^3/s]$	37,5
Drehzahl n $[1/min]$	100,0
Laufradaußenradius D/2 $[m]$	1,77
Anzahl Laufschaufeln $[-]$	4
Anzahl Leitschaufeln $[-]$	24

Tabelle 6.10: Betriebsdaten der Kaplan-Turbine KT192.

Mit diesem Testfall wird die hohe Modularität des Verfahrens durch verschiedene Konfigurationen für die Simulationsläufe dargestellt. Die Strukturberechnungen erfolgen zum einen mit dem von EINZINGER [34] entwickelten und validierten Finite-Elemente-Modul, zum anderen mit dem in dieser Arbeit vorgestellten Finite-Volumen-Modul. Es werden grundsätzliche Unterschiede in den Lösungsverhalten und Anwendungsregeln für die unterschiedlichen Strukturmodule abgeleitet. Da für diese Simulation keine experimentellen Vergleichsdaten vorliegen, werden die im Rahmen dieser Arbeit ermittelten Ergebnisse mit den Ergebnissen der Arbeit von EINZINGER [34] verglichen. Dieser verwendet das blockstrukturierte Vorgängerverfahren $ns3d$ zur Simulation des Fluids und das bereits erwähnte Finite-Elemente-Verfahren $fem3D$ zur Simulation der Stahl-Struktur.

Insgesamt werden Ergebnisse aus vier Simulationsläufen mit unterschiedlichen Verfahrenskonfigurationen vorgestellt, diskutiert und untereinander verglichen:

- Der *erste* Simulationslauf stellt eine instationäre Strukturnachrechnung im Anschluss an eine zuvor durchgeführte Fluidrechnung dar. Die instationäre Fluidrechnung erfolgt mit dem neu entwickelten Modul zur Lösung fluidmechanischer Gleichungen. Die Strukturnachrechnung wird durchgeführt mit dem von EINZINGER [34] entwickelten Finite-Elemente-Verfahren $fem3d$. Das Finite-Elemente-Modul wird hierbei als eigene Gleichung, welche die externe erstellte Bibliothek ansteuert, in das neue modulare Lösungsverfahren eingebunden.

- Der *zweite* Simulationslauf ist ebenfalls eine instationäre Strukturnachrechnung im Anschluss an eine zuvor durchgeführte Fluidrechnung. Die instationäre Fluidrechnung erfolgt wiederum mit dem neu entwickelten Modul zur Lösung fluid-

mechanischer Gleichungen. Die Strukturnachrechnung wird mit dem neu entwickelten Finite-Volumen-Modul zur Lösung strukturmechanischer Gleichungen durchgeführt.

- Der *dritte* Simulationslauf stellt eine zweiseitige instationäre FSI dar. Die instationäre Fluidrechnung erfolgt mit dem neu entwickelten Modul zur Lösung fluidmechanischer Gleichungen. Die Strukturnachrechnung erfolgt mit dem von EINZINGER [34] entwickelten FE-Verfahren $fem3d$. Das FE-Modul wird hierbei als eigene Gleichung, welche die externe erstellte Bibliothek ansteuert, in das neue modulare Lösungsverfahren eingebunden. Die Kopplung erfolgt über ein $FSI - Interface$ nach Kap. 4.6.3, welches den Datenaustausch und die Netzverformungen steuert.

- Der *vierte* Simulationslauf ist eine zweiseitige instationäre FSI. Die instationäre Fluidrechnung erfolgt mit dem neu entwickelten Modul zur Lösung fluidmechanischer Gleichungen und die Strukturnachrechnung mit dem neu entwickleten FV-Modul zur Lösung strukturmechanischer Gleichungen. Die Kopplung erfolgt über ein $FSI - Interface$ nach Kap. 4.6.3 dar, welche den Datenaustausch und die Netzverformungen steuert.

Die Berechnung wird auf einem Rechennetz durchgeführt, welches in drei Bereiche unterteilt ist, den stehenden Fluidbereich mit Zulauf und Leitschaufeln, den rotierenden Fluidbereich mit Laufschaufeln, den Auslassbereich sowie die rotierenden Laufschaufel-Strukturen. Der rotierende und der stehende Fluidbereich sind über ein GGI verbunden, s. Kap. 4.8.1. In jedem Zeitschritt werden die rotierenden Netzbereiche weiterrotiert und die Verschneidungen am GGI neu durchgeführt. Die Netzrotation und die Netzdeformation und die damit einhergehenden Netzmassflüsse werden durch Anwendung des Space-Conservation-Law berücksichtigt, s. Abschnitt 4.3.3.

Im Gegensatz zur Berechnung von EINZINGER [34] wurde das gesamte Laufrad der Maschine simuliert, um Einflüsse der periodischen Ränder auf die Simulationsergebnisse ausschließen zu können. EINZINGER [34] modellierte die Maschine nur mit einem Viertel des gesamten Umfangs. Die Rechennetze für Fluid- und Strukturbereich sind in Abb. 6.35 und Abb. 6.36 dargestellt. Die Netzgrößen sind in der nachfolgenden Tab. 6.11 zusammengefasst. Die Simulation erfolgte parallel auf vier Prozessen für die Fluidberechnung und einem Prozessor für die Strukturberechnung. Es wurden identische Rechennetze für die Strukturberechnungen mit dem FE- und dem FV-Modul verwendet, um einen direkten Vergleich beider Methoden zu erhalten. Für die FE-Berechnung wurden verschiebungsbasierte lineare Elemente verwendet.

Netzelemente Fluid Stationär $[-]$	383616
Netzelemente Fluid Rotierend $[-]$	192000
Netzelemente Struktur $[-]$	13056
Netzelemente gesamt $[m]$	588672

Tabelle 6.11: Netzgrößen der untersuchten Kaplan-Turbine.

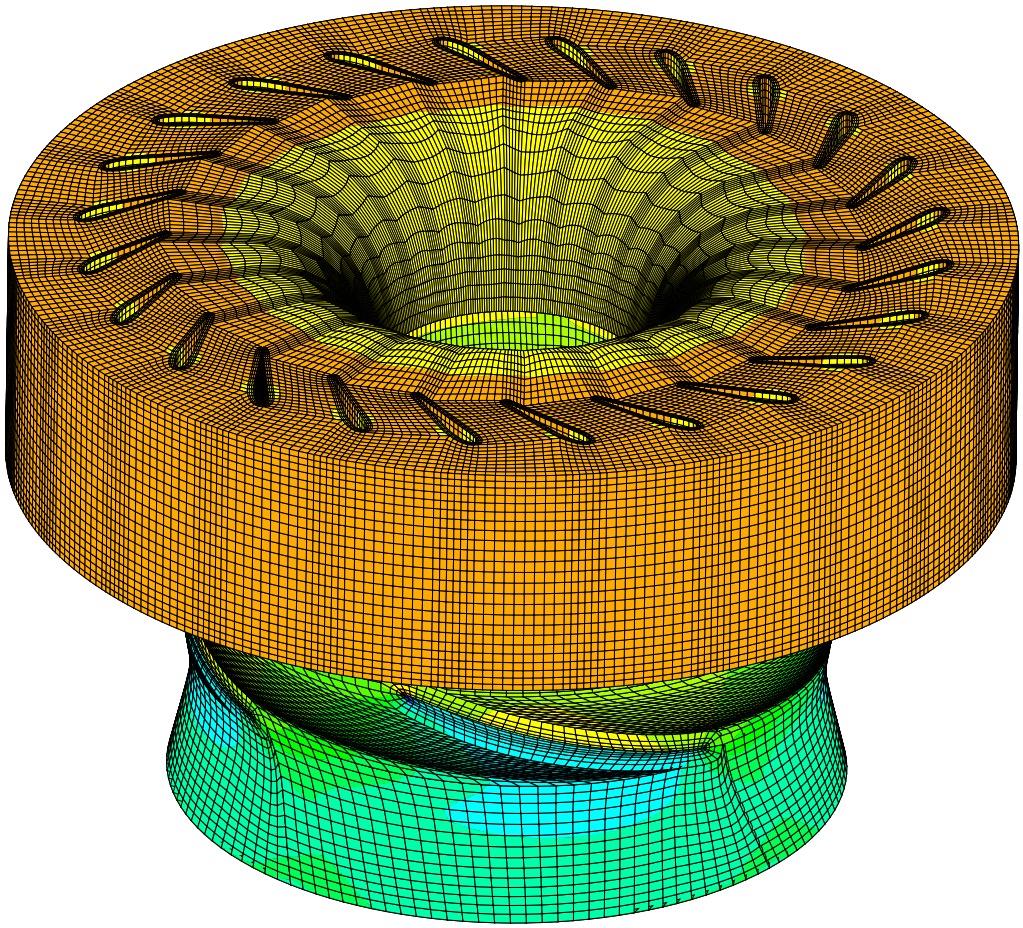

Abbildung 6.35: Vernetzung des Strömungsraums der untersuchten Kaplan-Turbine, Konturplot des statischen Drucks.

Die zeitlichen Verläufe in der Druckzahl $\psi_{t,th}$ und dem hydraulischen Wirkungsgrad η_h liefern nur sehr geringe Schwankungsbreiten in der Größe der auftretenden Residuen, weswegen keine eindeutigen Tendenzen bei unterschiedlicher Modellierung mit einseitiger oder zweiseitiger FSI-Simulation abgeleitet werden können.

Ausgewertet und verglichen werden deswegen bei den Berechnungen die Strukturantworten an vier ausgewählten Knoten an den Laufschaufeln. Zum einen sind das die sich ergebenden Absolutverschiebungen an der Vorderkante (A), der Hinterkante (C) und der Mitte (B) am Außenrand der Laufschaufeln, zum anderen die ermittelten Von-Mises-Vergleichsspannungen in der Mitte der Einspannstelle (D). Für alle durchgeführten Simulationsrechnungen lässt sich erkennen, dass die Strukturantworten sowohl hinsichtlich der untersuchten Verschiebungen als auch der untersuchten Spannungen bei den zweiseitig gekoppelten Rechnungen größer ausfallen als bei den reinen Strukturnachrechnungen. Dies ist vor allem deshalb interessant, da man bei den insgesamt sehr kleinen ermittelten Verschiebungen keinen Einfluss auf den Strömungsverlauf erwartet, dieser jedoch durch resultierende größere Drücke

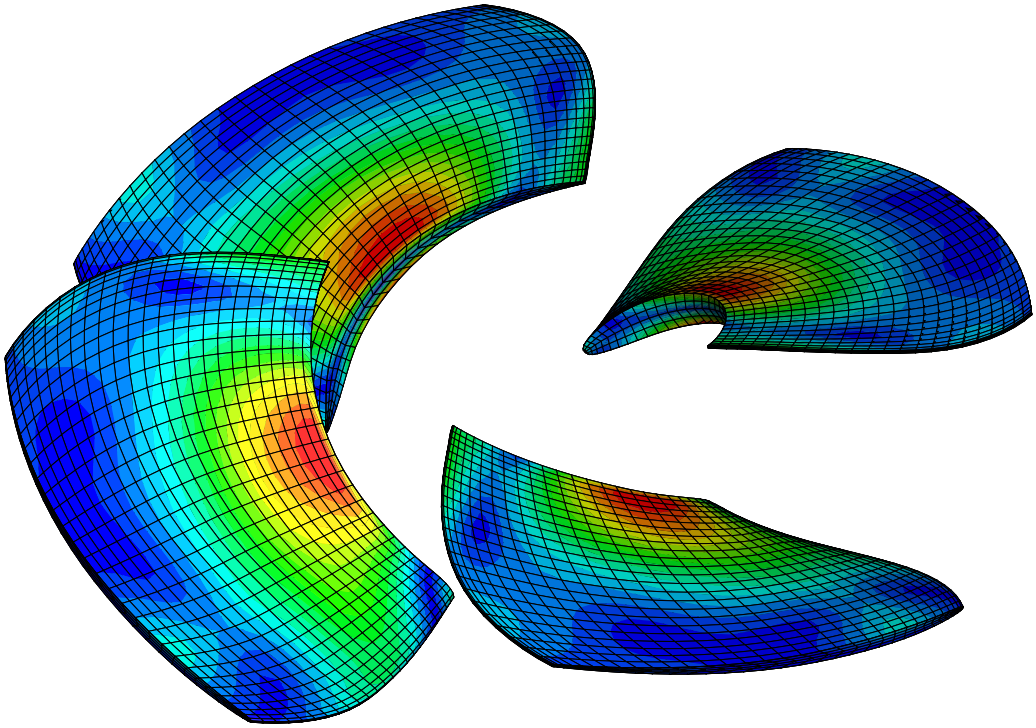

Abbildung 6.36: Vernetzung der Laufradstrukturen der untersuchten Kaplan-Turbine, Konturplot der v. Mises-Vergleichsspannung.

eindeutig gegeben ist.

Bei den Simulationen nach EINZINGER [34], welcher zur Verwendung der Fluidrechnung das Vorgängerverfahren $ns3d$ verwendete, erkennt man zum Ende der dargestellten Laufradumdrehung eine deutliche Aufweitung der Schwingungsamplituden, welche entweder auf einen noch nicht eingeschwungenen Strömungszustand oder auf numerische Instabilitäten im Lösungsverfahren zurückgeführt werden können. In der vorgestellten numerischen Simulation mit der Kombination aus dem neuen Lösungsverfahren $solver3d$ und dem von EINZINGER [34] entwickelten FE-Verfahren erkennt man einen deutlich gleichmäßigeren Verlauf in den Strukturantworten. Dies gilt für alle in diesem Abschnitt mit dem neuen Verfahren vorgestellten Simulationen. Trotzdem erkennt man im Gegensatz zu einer oszillatorisch vollkommen gleichmäßigen numerischen Lösung immer noch leichte Abweichungen vom erwarteten glatten Verlauf. Es kann aber trotzdem auf eine Verbesserung der numerischen Stabilität des neuen modularen Lösungsverfahrens geschlossen werden.

Beim Vergleich der Strukturantworten, welche mit dem FE- und dem FV-Modul ermittelt wurden, stellt man bei Anwendung des Finite-Volumen-Verfahrens teilweise deutlich geringere Amplituden fest. Dies gilt für alle Simulationsläufe bei Verwendung des FV-Moduls. Zur Vergleichbarkeit der Verfahren wurde für die FV- und die FE-

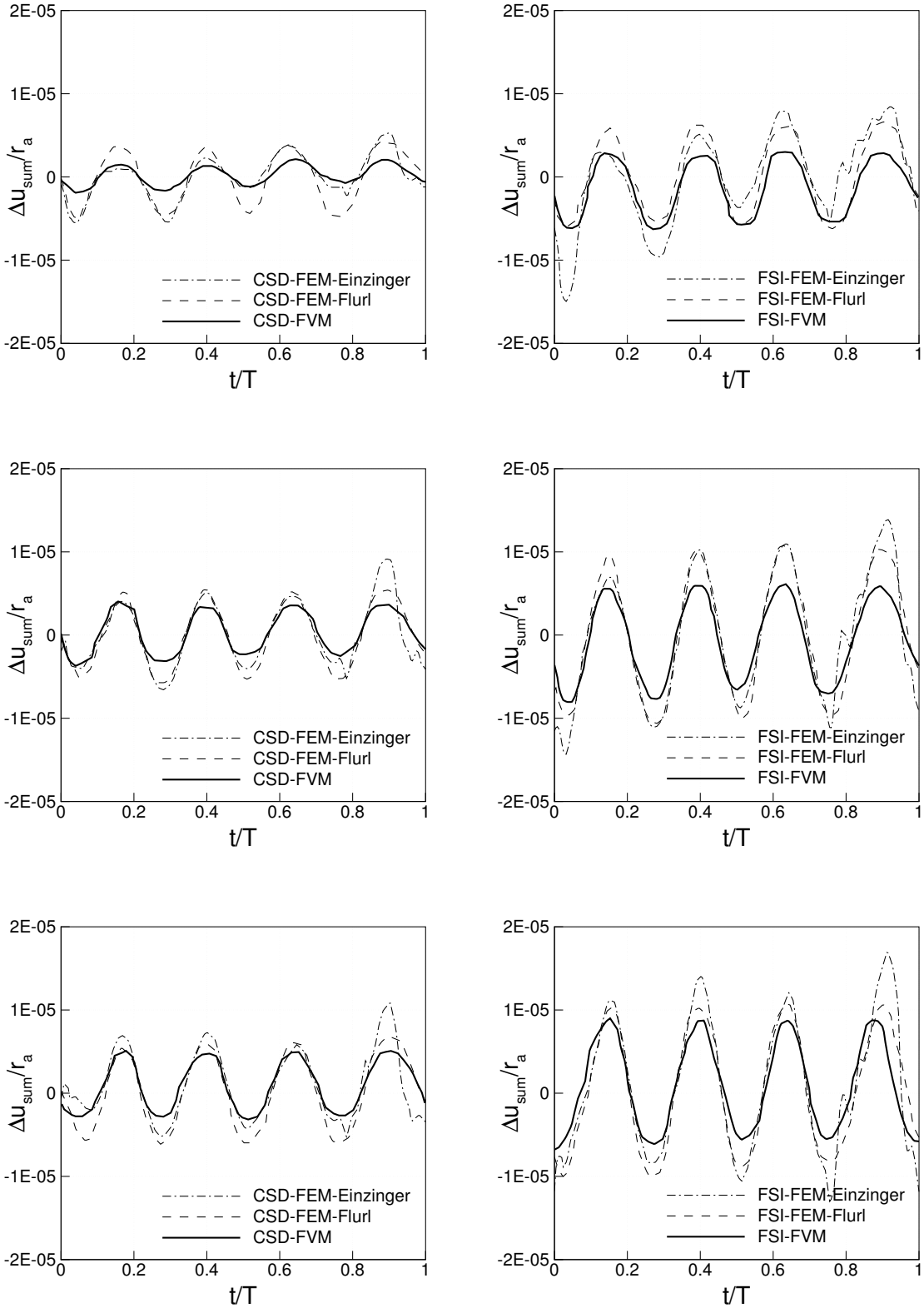

Abbildung 6.37: Zeitlicher Verlauf der Verschiebungsantworten an den Knoten A (oben), B (mitte) und C (unten).

Simulationen identische Strukturnetze verwendet. Man erkennt anhand der durchge-

führten Simulationen jedoch, dass die Anforderungen an ein FV-Netz hinsichtlich der Netzqualität und der Netzfeinheit deutlich höher sind als bei der Durchführung einer FE-Simulation. Das vorgestellte Rechennetz weist relativ starke Nichtorthogonalitäten und Scherungen auf, durch die das FV-Verfahren stark an numerischer Stabilität und numerischer Genauigkeit verliert. Für genaue Verschiebungs- und Spannungsanalysen müssen somit qualitativ hochwertige und deutlich feinere Netze als bei der Untersuchung mit einem FE-Verfahren gewählt werden.

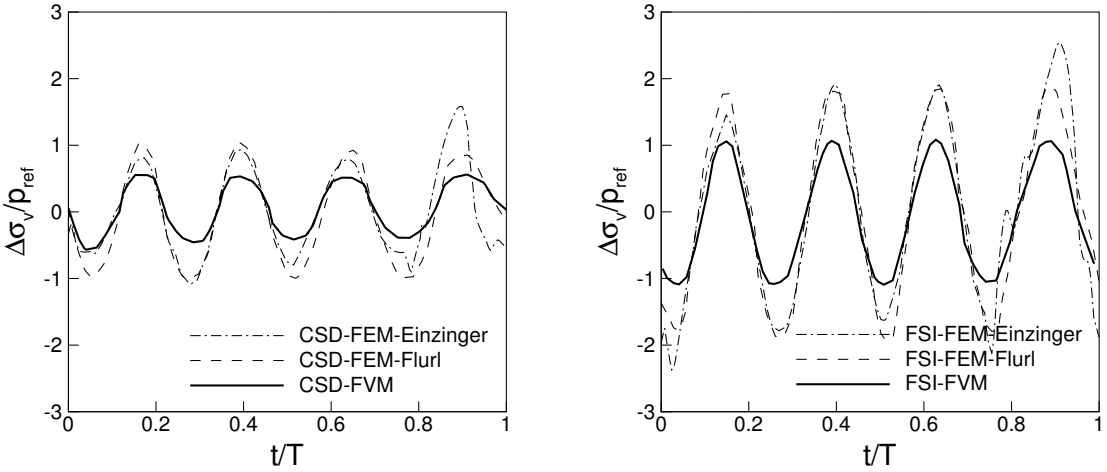

Abbildung 6.38: Zeitlicher Verlauf der Vergleichsspannungen nach v. Mises am Knoten D.

Neben den eben zusammengefassten Problemstellungen, welche bei derartigen Simulationen mit einer FV-Strukturberechnung auftreten, werden nun noch weitere Aussagen zusammengefasst, die aus den Simulationsergebnissen gleichermaßen für alle Konfigurationen abgeleitet werden können.

- Die Laufradschaufel ist an der Vorderkante (Knoten A) im Mittel größeren Verschiebungen als an der Hinterkante unterworfen.
- Die höchsten Amplituden und damit auch die höchsten Mittelwerte der Verschiebungen ergeben sich im Bereich der Schaufelmitte.
- Am Knoten D am Schaufelfuß ergibt der höchste Betrag der Vergleichsspannung. Wieder zeigen sich für die zweiseitigen Berechnungen deutlich höhere Amplituden als bei der einseitigen Berechnung.

Instationäre FSI-Simulationen können wichtige Aussagen hinsichtlich des Festigkeitsnachweises von Bauteilen von Strömungsmaschinen liefern. Dabei darf der stationäre mittlere Anteil der Vergleichsspannung die Streckgrenze des Werkstoffs nicht überschreitten. Die maximalen Amplituden der Schwankungsanteile der Vergleichsspannung müssen dagegen geringer als die Dauerfestigkeit des Werkstoffs sein. Mit Hilfe der Schwingungsfrequenzen, bzw. der Lastwechsel und mit dem jeweiligen Wöhlerdiagramm des betrachteten Werkstoffs kann dann die Lebensdauer der Schaufel abgeschätzt werden, s. NIEMANN [77]. Dieses Beispiel zeigt, dass das neu entwickelte modulare Verfahren auch für komplexe Mehrfeldprobleme in realen

Anwendungen eingesetzt werden kann. Die Modularität ermöglicht dem Forscher und dem Anwender größte Freiheiten hinsichtlich der gewünschten physikalischen und numerischen Modellierung.

Bei einer zweiseitigen FSI-Simulation, wie sie hier vorgestellt wurde, kommen alle in dieser Arbeit implementierten Module des Verfahrens zur Anwendung. Eine Übersicht über den angewendeten FSI-Kopplungs-Algorithmus ist in 6.39 zusammengefasst. Die Abbildung zeigt die ausgeführten Operationen während der Zeitschleife. Zu Beginn eines Zeitschrittes werden die Netze für die rotierenden Bereiche weiterbewegt und an den Schnittstellen zwischen rotierenden und stehenden Bereichen die Interfaces neu verschnitten. Innerhalb der Schleife werden die Gleichung für Fluid-

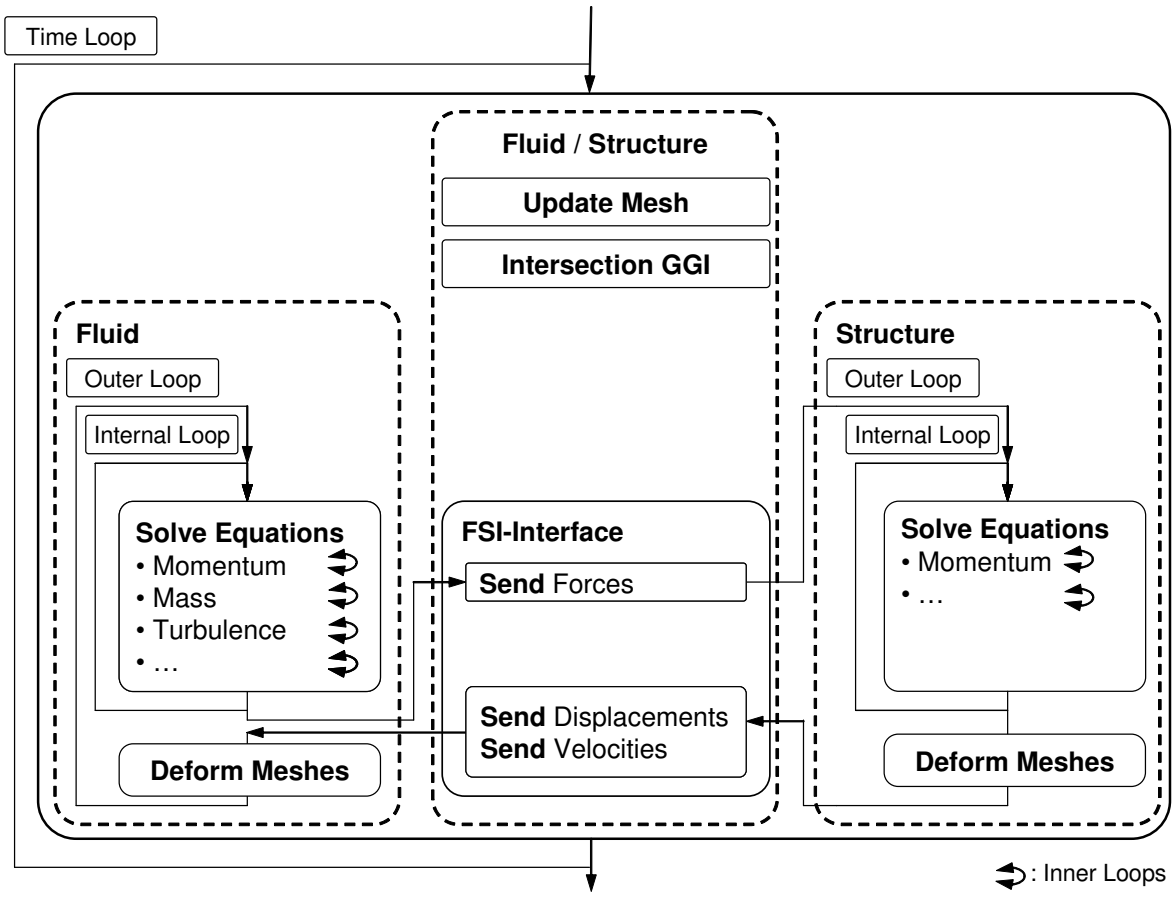

Abbildung 6.39: Das modulare Verfahren bei der Durchführung einer FSI-Simulation.

und Strukturraum auf verschiedenen Prozessen gelöst.
Die Verbindung erfolgt über das FSI-Interface, welches sowohl auf Fluid- als auch auf Struktur-Prozessen arbeitet. Über dieses Interface werden Fluid-Kräfte und Struktur-Verschiebungen und -Geschwindigkeiten an den entsprechenden Rändern gesendet. Die Fluid- und Struktur-Prozesse steuern die Netzverformung. Das FSI-Interface stellt sicher, dass die Fluid-und Strukturbereiche während des Simulationslaufes miteinander verbunden bleiben.
Die äußere Iterationsschleife stellt nun die stabile FSI-Kopplung sicher, der hier bezeichnete Internal-Loop behandelt die Nichtlinearitäten des Systems. Bei komplizierten FSI-Simulationen können mehrere äußere Iterationsschritte, auch als *staggered*

iterations bezichnet, s. HRON [55], notwendig werden. In der vorgestellten Simulation waren zwischen 1 und 3 derartige Iterationen notwendig. Die Internal-Loop-Schleife wurde stets bis zum Konvergenzkriterium des Zeitschritts ausgeführt.

Das neu entwickelte modulare Verfahren eignet sich somit auch zur Simulation komplexer Mehrfeldprobleme mit rotierenden und stehenden Fluid- und Strukturbereichen. Über die Definition entsprechender Gleichungsobjekte können unterschiedliche Module zur Strukturberechnung verwendet werden. Bei der Verwendung des im Rahmen dieser Arbeit entwickelten und implementierten FV-Moduls ist jedoch große Vorsicht hinsichtlich der Qualität und der Feinheit der verwendeten Rechennetze geboten, um gleichwertige Aussagen zu erhalten wie sie mit FE-Verfahren auf identischen Rechennetzen ermittelt werden.

Kapitel 7

Bewertung und Ausblick

Die Entwicklung des in dieser Arbeit vorgestellten modularen Lösungsverfahrens verfolgte zwei Ziele:

- Die Bereitstellung eines effizient erweiterbaren und effizient wartbaren numerischen Lösungsverfahrens zur Simulation geometrisch und physikalisch komplexer Mehrfeldprobleme für den universitären und industriellen Einsatz.
  Hierfür wurden das am Lehrstuhl entwickelte Vorgängerverfahren auf ein neues Fundament hinsichtlich des Programmdesigns, der Programmstruktur und der Modularität gestellt und das numerische Verfahren grundsätzlich überarbeitet.

- Die Implementierung und Validierung der strukturmechanischen Erhaltungsgleichungen in einer Finite-Volumen-Formulierung, um die Eignung dieses Verfahrens einerseits hinsichtlich strukturmechanischer Berechnungen und andererseits hinsichtlich der Eignung zur Simulation von Fluid-Struktur-Interaktionen zu untersuchen sowie zusätzlich Vorteile und Nachteile gegenüber den etablierten Finite-Elemente-Verfahren herauszuarbeiten.

Für die spätere Anwendung dient das modulare Lösungsverfahren als Grundbaustein in einem Integrierten-Design-System für die Erstauslegung und Optimierung von Strömungsmaschinen.

Entwicklung des Modularen Verfahrens

Das Verfahren wurde in *C++* implementiert und greift auf verschiedene Bibliotheken zur Geometriebearbeitung, Lösung linearer Gleichungssysteme, Partitionierung und Parallelisierung zu.
Durch die Verwendung objektorientierter Techniken können sog. Konfigurationen für verschiedene Anwender und unterschiedlichste Problemstellungen wie mit einem Baukasten neu erstellt oder bereits bestehende Konfigurationen erweitert werden. Das Programm ist somit einerseits effizient wartbar und erweiterbar, andererseits ist es einfach, sich in die objektorientierten Strukturen einzuarbeiten und so das Verfahren effizient für Forschungs- und Industrieprojekte anzuwenden.

Finite-Volumen-Methode

Im Rahmen dieser Arbeit wurde die FVM zur Diskretisierung partieller Differentialgleichungen für strömungs- und strukturmechanische Problemstellungen angewendet. Im Gegensatz zu den von SKODA [93] und EINZINGER [34] entwickelten Vorgängerverfahren arbeitet das neue Verfahren auf einer polyedrisch unstrukturierten Datenstruktur, wodurch die Behandlung beliebig komplexer Geometrien ermöglicht wird. Aufgrund der neuen Datenstruktur mussten neue Module zur Diskretisierung, zum Datenaustausch an allgemeinen Domain-Interfaces und zur Lösung linearer Gleichungssysteme entwickelt werden.

Die Leistungsfähigkeit der hinterlegten Diskretisierungstechnik wird sowohl für fluid- als auch strukturmechanische Problemstellungen an repräsentativen Testfällen aufgezeigt. Hauptaugenmerk liegt auf den strukturmechanischen Testfällen, an denen Stärken, aber auch Schwächen bei Anwendung der FVM dargelegt werden.

Eine FSI-Simulation einer Kaplanturbine zeigt die Anwendbarkeit des modularen Lösungsverfahrens für komplexe Mehrfeldprobleme, bei der alle im Rahmen der Arbeit entwickelten Module miteinander interagieren. Mit dem entwickelten Modul FSI-Interface können sowohl einseitig als auch zweiseitig gekoppelte FSI-Simulationen durchgeführt werden. Durch die Strukturberechnung sowohl mit dem FE-Modul von EINZINGER [34] als auch dem neu entwickelten FV-Modul wird die Vielseitigkeit des modularen Konzepts unterstrichen. Die mit dem FV-Modul entwickelten Ergebnisse zeigen teilweise starke Abweichungen von den mit dem FE-Modul ermittelten Lösungen. Für die erfolgreiche Anwendung des FV-Modul zur Strukturberechnung müssen deutlich feinere und qualitativ bessere Rechennetze bzw. Techniken wie adaptive Netzverfeinerung eingesetzt werden als bei herkömmlichen FE-Verfahren.

Fazit

Die Enwicklungszeit des Verfahrens erscheint vor dem Hintergrund der Verfügbarkeit des Quellcodes gerechtfertigt.
Durch den modularen und objektorientierten Aufbau ist das Verfahren effizient wartbar und einfach erweiterbar. Eine Einarbeitung in die Programmstruktur ist einfach und schnell möglich.

Die implementierten Programmteile zur Struktursimulation und zur FSI mit dem FVV wurden umfangreich validiert und getestet. Für die Simulation stationärer und instationärer strukturmechanischer Problemstellungen mit kleinen und großen Verformungen erweist sich das FVV als effizient, stabil und - unter Verwendung sehr feiner und qualitativ hochwertiger Rechennetze - genau.

Anhang A

Kompressible Medien

Zur Beschreibung kompressibler Strömungsvorgänge müssen zusätzliche Gleichungen gelöst werden, welche im Folgenden ausführlich in der implementierten Formulierung aufgeführt werden.

A.1 Energieerhaltung

Die Erhaltungsgleichung für die Totalenergie lautet in Vektorschreibweise:

$$\frac{\partial(\rho e_0)}{\partial t} + \nabla(\rho \mathbf{v} h_0) = -\nabla \mathbf{q} + \nabla(\mathbf{v}\boldsymbol{\tau}). \tag{A.1}$$

Auf der linken Seite stehen die zeitliche Änderung und der konvektive Transport der Totalenergie, auf der rechten Seite der Energieeintrag durch den Wärmestromdichtevektor und die Änderung der Totalenergie durch Reibungsarbeit. e_0 bezeichnet die spezifische totale Energie, h_0 die spezifische totale Enthalpie. Für diese Größen gilt mit der spezifischen statischen Enthalpie h folgender Zusammenhang:

$$h = e + \frac{p}{\rho}, \tag{A.2}$$

$$e_0 = e + \frac{1}{2}|\mathbf{v}|, \tag{A.3}$$

$$h_0 = h + \frac{1}{2}|\mathbf{v}|. \tag{A.4}$$

Der Wärmestromdichtevektor wird mit Hilfe des Fourier-Gesetzes ausgedrückt:

$$\mathbf{q} = -\lambda \nabla T. \tag{A.5}$$

Darin bezeichnet λ die Wärmeleitfähigkeit des Fluids.

A.1.1 Energieerhaltung im ein rotierenden Bezugssystem

Bei der Transformation der Energiegleichung muss zusätzlich die Arbeit der Zentrifugalkraft berücksichtigt werden, die Corioliskraft verrichtet nach HIRSCH [53] keine

Arbeit. Die transformierte Energiegleichung lautet dann:

$$\frac{\partial(\rho e_0^{rel})}{\partial t} + \nabla(\rho \mathbf{w} h_0^{rel}) = \nabla[(\mu + \frac{\mu_t}{Pr}\nabla h] + \nabla(\mathbf{w}\boldsymbol{\tau}). \tag{A.6}$$

Die spezifische totale innere Energie und die spezifische totale Enthalphie im Relativsystem sind durch die Beziehungen

$$e_0^{rel} = e + \frac{1}{2}\mathbf{ww} - \frac{1}{2}\mathbf{vv} + k \tag{A.7}$$

und

$$h_0^{rel} = h + \frac{1}{2}\mathbf{ww} - \frac{1}{2}\mathbf{vv} + k \tag{A.8}$$

gegeben.

A.2 Zustandsgesetze

Können die Stoffwerte Dichte ρ und Viskosität μ nicht als konstant angenommen werden, benötigt man neben den bisher vorgestellten Erhaltungsgleichungen geeignete Zustandsgesetze, um diese korrekt berechnen zu können und damit Temperatur, Dichte, Viskosität und Druck zu verknüpfen.

A.2.1 Dichteänderung von Gasen

Bei Gasströmungen wird im Rahmen dieser Arbeit das ideale Gasgesetz verwendet:

$$p = \rho R T, \tag{A.9}$$

wobei R die spezifische Gaskonstante darstellt.

A.2.2 Dichteänderung von Flüssigkeiten

Flüssigkeiten werden im Rahmen dieser Arbeit als inkompressibel oder schwach kompressibel und isotherm betrachtet. Die Dichte ist deswegen nur eine Funktion des Drucks und kann mit der Zustandgleichung nach Tate bestimmt werden:

$$p - p_0 = B\left(\frac{\rho}{\rho_0}\right)^{m-1}. \tag{A.10}$$

Die auftretende Konstante B wird folgendermaßen bestimmt:

$$B = \frac{\rho_0 c_0^2 c_v}{c_p}. \tag{A.11}$$

ρ_0 bezeichnet die Initial-Dichte, ρ die Dichte des komprimierten Mediums, c_0 die Schallgeschwindigkeit bei Initial-Druck und c_v und c_p die spezifischen Wärmekapazitäten bei konstantem Volumen und bei konstantem Druck. m stellt den Adiabatenexponenten dar. Die angegebenen Werte gelten für Wasser und müssen je nach Fluid angepasst werden.
Eine schwach kompressible Betrachtung z. B. von Wasserströmungen kann die numerische Stabilität signifikant erhöhen und wurde im Rahmen dieser Arbeit sehr erfolgreich angewandt.

A.2.3 Viskositätsänderung von Gasen

Die Änderung der Viskosität mit der Temperatur in strömenden Gasen wird durch das Sutherland-Gesetz bestimmt:

$$\mu = \mu_0 \frac{S + T_0}{S + T} \left(\frac{T}{T_0} \right)^{\frac{3}{2}} . \tag{A.12}$$

S und μ sind dabei fluidabhängige Konstanten, T_0 bezeichnet eine Bezugstemperatur.

Anhang B

Statistische Turbulenzmodellierung

Die statistische Turbulenzmodellierung geht auf einen Separationsansatz von REYNOLDS [85] zurück, mit dem man die betrachtete Strömungsgröße ϕ in einen zeitlichen Mittelwert $\bar{\phi}$ und einen Schwankungswert ϕ' aufteilt. Dieser Ansatz lässt sich neben stationären auch auf instationäre Problemstellungen anwenden, wenn die globale Änderung der Strömung deutlich langsamer verläuft als die turbulenten Schwankungen.
Wird der Separationsansatz nach Reynolds auf die inkompressiblen Navier-Stokes-Gleichungen angewendet, erhält man die Reynolds-gemittelten Navier-Stokes-Gleichungen, welche im Gegensatz zu den ursprünglichen Gleichungen den Transport der zeitlich gemittelten Strömungsgrößen beschreiben. Durch die Mittelung der konvektiven Terme tritt auf der rechten Seite der Reynolds-Spannungstensor $\mathbf{R}$ auf, der die zeitlich gemittelte Wirkung der turbulenten Konvektion darstellt:

$$\mathbf{R} = -\overline{\mathbf{v}'\mathbf{v}'}. \tag{B.1}$$

Dieser neue Term ist zunächst unbekannt und muss zur Schließung des Gleichungssystems durch zusätzliche Transportgleichungen ermittelt werden.
In der Ingenieurspraxis haben sich aufgrund ihrer Einfachheit die sog. linearen Zwei-Gleichungs-Modelle durchgesetzt und bewährt. Bei diesen Modellen wird der Reynolds-Spannungstensor über die Wirbelviskositätsannahme direkt modelliert:

$$\mathbf{R} = \mu_t \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T \right) - \frac{2}{3} \rho k \mathbf{I}. \tag{B.2}$$

In allen Erhaltungsgleichungen wird dann die Viskosität durch eine effektive Gesamtviskosität ersetzt, welche sich aus der laminaren und der turbulenten Viskosität zusammensetzt:

$$\mu_{eff} = \mu_{lam} + \mu_{turb}. \tag{B.3}$$

Zur Schließung des Gleichungssystems wird bei diesen Zwei-Gleichungs-Modellen eine Transportgleichung für die turbulente kinetische Energie und die turbulente Dissipation abgeleitet. Die turbulente kinetische Energie wird mit dem Reynolds-Spannungstensor folgendermaßen definiert:

$$k = -\frac{1}{2}\mathbf{R}. \tag{B.4}$$

Die Dissipation beschreibt, wie turbulente kinetische Energie in Wärme dissipiert wird.

Die vorgestellte Mittelungsvorschrift nach Reynolds kann prinzipiell auch für kompressible Fluide angewendet werden, stellt sich hierfür allerdings als unvorteilhaft heraus. Bei kompressiblen Fluiden wendet man deswegen einen massengewichteten Mittelungs-Ansatz nach FAVRE [8] an, wobei dieselbe Problematik wie bei der Reynoldsmittelung für inkompressible Fluide entsteht und eine Modellierung des Reynolds-Spannungstensor notwendig macht. Eine ausführliche Beschreibung dieses Ansatzes findet sich beispielsweise in WILCOX [109].
EINZINGER [34] entwickelte ein Verfahren, welches für alle Machzahl-Bereiche verwendet werden kann. Dies funktioniert, da man die Favré-gemittelten Gleichungen auch für inkompressible Fluide anwenden kann, wenn die Reynolds-gemittelten Größen durch die Favré-gemittelten Größen ersetzt, den Reynolds-Spannungstensor anpasst und entsprechende Definitionen für Dissipation und Produktionsterm der turbulenten kinetischen Energie verwendet. Man erhält hierdurch ein sehr kompaktes, universell anwendbares numerisches Simulations-Werkzeug.
Die im Rahmen dieser Arbeit verwendeten Zwei-Gleichungs-Modelle werden in den folgenden Abschnitten vorgestellt. Alle diese Zwei-Gleichungs-Modelle existieren in unterschiedlichen Varianten, so dass eine Zusammenfassung aller im Verfahren auftretenden und hinterlegten Konstanten jeweils tabellarisch aufgeführt wird.

B.0.4 Zwei-Gleichungs-Modelle

B.0.4.1 k-ϵ-Turbulenzmodell

Beim k-ϵ-Turbulenzmodell nach LAUNDER [66] wird die turbulente Viskosität mit Hilfe der turbulenten kinetischen Energie und der Dissipation bestimmt:

$$\mu_t = C_\mu \rho \frac{k^2}{\epsilon}. \tag{B.5}$$

C_μ ist eine Modell-spezifische Konstante.
Die Erhaltungsgleichung für die turbulente kinetische Energie lautet:

$$\frac{(\partial \rho k)}{\partial t} + \nabla(\rho \mathbf{v} k) = P_k - \rho\epsilon + \nabla(\mu + \frac{\mu_t}{\sigma_k})\nabla k. \tag{B.6}$$

Die Dissipation wird bestimmt mit:

$$\frac{(\partial \rho \epsilon)}{\partial t} + \nabla(\rho \mathbf{v} \epsilon) = C_{\epsilon 1}\frac{\epsilon}{k}P_k - C_{\epsilon 2}\frac{\epsilon^2}{k}P_k - \nabla(\mu + \frac{\mu_t}{\sigma_\epsilon})\nabla k. \tag{B.7}$$

P_k bezeichnet den Produktionsterm für die turbulente kinetische Energie:

$$P_k = \mu_t \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T\right) \nabla \mathbf{v} + \frac{2}{3}\rho k \mathbf{I} \nabla \mathbf{v}. \tag{B.8}$$

Das ursprüngliche Modell ist nur für hohe Reynoldszahlen geeignet; durch die Dämpfungsfunktionen $C_{\epsilon 1}$ und $C_{\epsilon 2}$ auf der rechten Seite der ϵ-Gleichung wird ein korrektes Verhalten an Wänden ermöglicht. Diese Dämpfungsfunktionen werden für jede Variante des Turbulenzmodells speziell definiert. Die Modellkonstanten und die Dämpfungsfunktionen für die verwendeten Varianten des k-ϵ-Modells sind in nachstehener Tabelle angegeben.

MODELL	σ_k	σ_ϵ	C_{mu}	$C_{\epsilon 1}$	$C_{\epsilon 2}$
k-ϵ-STANDARD	1.0	1.3	0.09	1.0	1.0

Tabelle B.1: Konstanten des k-ϵ-Turbulenzmodells

B.0.4.2 k-ω-Turbulenzmodell

Beim k-ω-Turbulenzmodell wird zusätzlich zur Erhaltungsgleichung für die turbulente kinetische Energie B.6 eine Erhaltungsgleichung für die spezifische Dissipation ω verwendet:

$$\frac{(\partial \rho \omega)}{\partial t} + \nabla(\rho \mathbf{v} \omega) = \alpha \frac{P_k}{k} \omega - \beta \omega^2 + \nabla(\mu + \frac{\mu_t}{\sigma_\omega}) \nabla \omega. \tag{B.9}$$

Die spezifische Dissipation wird mit Hilfe der turbulenten kinetischen Energie und der Dissipation definiert:

$$\omega = \frac{\epsilon}{C_\mu k} \tag{B.10}$$

und man erhält wegen dieser Beziehung und B.5 die turbulente Viskosität mit:

$$\mu_t = \rho \frac{k}{\omega}. \tag{B.11}$$

Der Vorteil dieses Modells gegenüber dem k-ϵ-Modell ist einerseits die korrekte Voraussage der turbulenten Größen in Wandnähe ohne Dämpfungsfunktionen. Andererseits liefert das k-ω-Modell weniger gute Voraussagen für freie Strömungen weitab von Wänden. Die Modellkonstanten für die verwendeten Variante des k-ω-Modells sind in nachstehender Tabelle angegeben.

MODELL	σ_k	σ_ω	C_{mu}/β^*	α	β
k-ω-STANDARD	0.5	0.53	0.09	$\frac{5}{9}$	$\frac{3}{40}$

Tabelle B.2: Konstanten des k-ω-Turbulenzmodells

Durch die Verwendung der Wirbelviskositätsannahme wird bei Verwendung der beiden vorgestellten Turbulenzmodelle in Staupunktströmungen eine überhöhte Produktion von turbulenter kinetischer Energie vorausgesagt. Diese Überproduktion beeinflusst die turbulente Grenzschicht massiv, verfälscht die Ergebnisse und muss deswegen vermieden werden.
Hierfür wurden von zahlreichen Autoren verschiedene Limitierungs-Maßnahmen vorgeschlagen. Im Rahmen dieser Arbeit werden die Korrekturen nach KATOLAUNDER (KL), MOOREMOORE (MM) and DURBIN (D) angewendet, welche teilweise standardmäßig verwendet werden oder optional zugeschaltet werden können.

B.0.4.3 Shear-Stress-Transport(SST)-Turbulenzmodell

Die Vorteile der beiden bisher vorgestellten Turbulenzmodelle werden im Shear-Stress-Transport-Modell nach MENTER [73] vereint. Durch eine Mischungs- oder Blending-

Funktion F_1

$$F_1 = tanh(arg_1^4), \tag{B.12}$$

$$arg_1 = min\left(max\left(\frac{\sqrt{k}}{C_\mu \omega y}; \frac{500\nu}{y^2\omega}\right); \frac{4\rho\sigma_{\omega 2}k}{CD_{k\omega}y^2}\right), \tag{B.13}$$

$$CD_{k\omega} = max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}; 1.0e^{-10}\right) \tag{B.14}$$

wird erreicht, dass in Wandnähe das k-ω-Modell, in freien Scher-Strömungen das k-ϵ-Modell verwendet wird. Die verwendeten Erhaltungsgleichungen für die turbulente kinetische Energie und die spezifische Dissipation ω lauten bei diesem Modell:

$$\frac{(\partial \rho k)}{\partial t} + \nabla(\rho \mathbf{v} k) = \tilde{P}_k - C_\mu \rho \omega k + \nabla(\mu + \frac{\mu_t}{\sigma_k})\nabla k \tag{B.15}$$

$$\frac{(\partial \rho \omega)}{\partial t} + \nabla(\rho \mathbf{v} \omega) = \frac{\gamma}{\nu_t}P_k - \beta\rho\omega^2 + \nabla(\mu + \frac{\mu_t}{\sigma_\omega})\nabla\omega + (1 - F_1)\, 2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}. \tag{B.16}$$

Der Produktionsterm wird nur in der k-Gleichung mit der folgenden Beziehung limitiert:

$$\tilde{P}_k = min\left(P_k; c_l \epsilon\right). \tag{B.17}$$

Die auftretenden Konstanten ϕ des Modells werden mit Hilfe der Mischungs-Funktion F_1 berechnet:

$$\phi = F_1\phi_1 + (1 - F_1)\phi_2. \tag{B.18}$$

Die turbulente Viskosität wird mit Hilfe einer weiteren Blending-Funktion F_2:

$$F_2 = tanh(max\left(\frac{2\sqrt{k}}{C_\mu \omega y}; \frac{500\nu}{y^2\omega}\right)) \tag{B.19}$$

begrenzt:

$$\mu_t = \rho\frac{a_1 k}{max\left(a_1\omega; \sqrt{2}SF_2\right)}. \tag{B.20}$$

Im Rahmen dieser Arbeit wird die SST-Variante nach VIESER [104] implementiert, bei welcher zur Berechnung der turbulenten Viskosität der Betrag der Dehnungs-Rate S anstelle der Wirbelstärke Ω verwendet wird:

$$S = \sqrt{S_{ij}S_{ij}} = \sqrt{\frac{1}{4}\left(\nabla\mathbf{v} + \nabla\mathbf{v}^T\right)^2}. \tag{B.21}$$

Diese verhindert zusätzlich zum Produktionslimiter B.6 eine überhöhte Produktion von turbulenter kinetischer Energie in der Nähe von Staupunkten.
Das SST-Modell vermag im Gegensatz zu den originären k-ϵ- und k-ω-Modellen auch Strömungen mit negativem Druckgradienten zu simulieren, wie sie in Ablösegebieten oder Transitionsgebieten auftreten können.
Die auftretenden Konstanten des hinterlegten SST-Turbulenzmodells sind in nachstehender Tabelle zusammengefasst.

MODELL	σ_{k1}/σ_{k2}	$\sigma_{\omega1}/\sigma_{\omega2}$	γ_1/γ_2	β_1/β_2	c_l	a_1	β^*
SST-Vieser	2.0/1.0	2.0/1.168	0.5532/0.4403	0.075/0.0828	10.0	0.31	0.09

Tabelle B.3: Konstanten des SST-Turbulenzmodells nach Vieser

B.0.5 Wandbehandlung

Bei praktischen Anwendungen ist es sehr schwierig, das Verhalten einer turbulenten Strömung in der Nähe von festen Wänden korrekt zu simulieren. Die zu bestimmenden Wandschubspannungen oder Transitions- und Ablösegebiete der turbulenten Grenzschicht hängen in starkem Maß von einer korrekten Wandbehandlung ab.
Die turbulente Grenzschicht an Wänden wird in drei Bereiche unterteilt, die viskose Unterschicht, den logarithmischen Bereich und die turbulente Kernströmung.

Für eine korrekte Simulation des Einflusses der Grenzschicht auf die Strömung stehen prinzipiell zwei unterschiedliche Ansätze der Wandbehandlung zur Verfügung:

- der Ansatz der niedrigen Reynoldszahlen bzw. der Low-Reynolds-Ansatz und der
- Wandfunktions-Ansatz,

welche beide im entwickelten Verfahren hinterlegt sind und deswegen im Folgenden kurz beschrieben werden.

B.0.5.1 Low-Reynolds-Ansatz

Die Anwendung dieses Ansatzes erfordert ein sehr feines Rechennetz in Wandnähe, da die viskose Unterschicht mit einer ausreichenden Anzahl von Gitter-Zellen aufgelöst werden muss. Der Vorteil dieses Vorgehens liegt in einer detaillierten Abbildung der Strömung auch in Wandnähe. Das Rechennetz muss mit großer Sorgfalt erstellt werden, und der Rechenaufwand ist wegen der hohen Netzauflösung sehr groß.

B.0.5.2 Wandfunktions-Ansatz

Den genannten Nachteil des Low-Reynolds-Ansatzes versucht der Wandfunktions-Ansatz zu umgehen, indem die viskose Unterschicht durch empirisch gefundene Formeln so überbrückt wird, dass geeignete Randbedingungen für alle Gleichungen in Wandnähe gefunden werden können.
Dazu müssen die Wand-nächsten Rechenpunkte, an denen mit diesen Formeln Werte für die turbulente kinetische Energie, die Dissipation, die Produktion und die Wandschubspannung festgelegt werden, im logarithmischen Bereich der turbulenten Grenzschicht liegen. Für eine praktische Anwendung in geometrisch komplexen Strömungsbauteilen ist das kaum für alle Wandbereiche einzuhalten und eine Netzgenerierung erweist sich auch bei diesem Ansatz als schwierig.
Der Vorteil dieses Ansatzes liegt darin, dass das verwendete Rechennetz im Vergleich zum Low-Reynolds-Ansatz sehr grob gewählt werden kann und der Rechenaufwand deutlich sinkt.

Bei der Verwendung von Wandfunktionen unterscheidet man die Standard-Wandfunktion und die skalierbare Wandfunktion.

Standard-Wandfunktion

Bei der Standard-Wandfunktion wird davon ausgegangen, dass das Rechengebiet erst im logarithmischen Bereich der Grenzschicht beginnt. In diesem Bereich gilt für den Verlauf der dimensionslosen Geschwindigkeit das universelle logarithmische Wandgesetz:

$$v_\tau^+ = \frac{1}{\kappa} \ln y^+ + C. \tag{B.22}$$

Die Konstante C wird mit $C = 5.2$ angesetzt, die Konstante κ mit $\kappa = 0.41$.
Die dimensionslose Geschwindigkeit u^+ und der dimensionslose Wandabstand y^+ sind dabei wie folgt definiert:

$$v_\tau^+ = \frac{\bar{v}}{v_\tau}, \tag{B.23}$$

bzw.:

$$y^+ = \frac{y_P v_\tau}{\nu}. \tag{B.24}$$

$\bar{u}$ bedeutet hierbei die wandtangentiale Geschwindigkeitskomponente. y_P stellt den dimensionsbehafteten Wandabstand dar.
Schubspannungsgeschwindigkeit v_τ und Wandschubspannung τ_w stehen in folgendem Zusammenhang:

$$v_\tau = \sqrt{\frac{\tau_w}{\rho}}. \tag{B.25}$$

Unter der Annahme, dass im logarithmischen Bereich der Grenzschicht Produktion und Dissipation im Gleichgewicht stehen, erhält man nach DURBIN [33] eine weitere Beziehung für u_τ:

$$v_\tau = C_\mu^{1/4} \sqrt{k} \tag{B.26}$$

und gewinnt mit dieser Beziehung einen Wert für die Dissipation am Wand-nächsten Rechenpunkt:

$$\epsilon = \frac{C_\mu^{3/4} k^{3/2}}{\kappa y_P}. \tag{B.27}$$

Dieser wird dann im diskreten Gleichungssystem direkt gesetzt. Die Dissipations-Gleichungen werden also am Wand-nächsten Rechenpunkt nicht gelöst. Für die turbulente kinetische Energie wird eine Null-Gradienten-Randbedingung verwendet.

$$\frac{\partial k}{\partial n} = 0. \tag{B.28}$$

Bei der Verwendung der Standard-Wandfunktion wird empfohlen, den ersten Rechenpunkt in einem dimensionslosen Abstand $y^+ \approx 30$ von der Wand zu platzieren, was in vielen praktischen Problemstellungen schwierig oder sogar unmöglich ist.
Bei Verwendung dieser Art der Wandfunktion können deswegen Netzunabhängigkeitsstudien nur schwer durchgeführt werden.

Skalierbare Wandfunktion
Die Nachteile, welche sich bei der Verwendung der Standard-Wandfunktion ergeben, werden durch die skalierbare Wandfunktion nach GROTJANS [45] vermieden. Bei diesem Ansatz werden die Gleichungen so diskretisiert, als ob das Rechennetz auf der viskosen Unterschicht platziert wäre. Das Rechennetz wird aber weiterhin direkt auf der Wand platziert, in realen Strömungen ist aber die Dicke der viskosen Unterschicht im Vergleich zur Ausdehnung der restlichen Strömung zu vernachlässigen.
Der erste Rechenpunkt muss dann wie bei der Standard-Wandfunktion nicht speziell behandelt werden und ist damit unabhängig von der Netzgenerierung.
Ferner ist es nun möglich, eine Netzverfeinerung ohne Verletzung des y^+-Kriteriums durchzuführen.
Das Ende der viskosen Unterschicht wird als Schnittpunkt des linearen und logarithmischen Verlaufs des dimensionslosen Geschwindigkeitsprofils ermittelt:

$$\tilde{y}^+ = \frac{1}{\kappa} \ln y^+ + C. \tag{B.29}$$

Die iterative Lösung dieser Gleichung liefert einen Wert von $\tilde{y}^+ \approx 11.06$.
Die Werte der Geschwindigkeit, der turbulenten kinetischen Energie, der Dissipation und der Viskosität an der Stelle $\tilde{y}^+$ können dann direkt bestimmt werden, denn es gelten die folgenden Beziehungen:

$$v_y = v_\tau \left(\frac{1}{\kappa} \ln y^+ + C \right), \tag{B.30}$$

$$k(y) = \frac{v_\tau^2}{\sqrt{C_\mu}}, \tag{B.31}$$

$$\epsilon(y) = \frac{v_\tau^3}{\kappa y} = \rho \kappa v_\tau y. \tag{B.32}$$

Die Wandschubspannung wird mit Hilfe der Wandschubgeschwindigkeit bestimmt:

$$\tau_w = -\rho v_\tau^2. \tag{B.33}$$

Dabei wird die Geschwindigkeit v_τ mit

$$v_\tau = max \left(u^*; \frac{|\mathbf{v}|}{\tilde{y}^+} \right) \tag{B.34}$$

berechnet. Die verwendete Geschwindigkeit u^*

$$v^* = C_\mu^{1/4} \sqrt{k} \tag{B.35}$$

verhindert, dass die Schubspannung an Ablösepunkten der Strömung verschwindet.
Für die turbulente kinetische Energie wird wiederum ein Nullgradient in Wandnormalenrichtung als Randbedingung angewendet.
Für die turbulente Dissipation wird eine Flussrandbedingung aus Gleichung B.32 abgeleitet:

$$\frac{\partial \epsilon}{\partial n} = -\frac{\rho^2 v^{*5}}{\tilde{y}^+ \mu \mu_\tau}. \tag{B.36}$$

B.0.6 Automatische Wandbehandlung

Eine weitere Möglichkeit zur Vermeidung der Problematik bei Verwendung der Standard-Wandfunktion wird mit der automatischen Wandbehandlung nach MENTER [73] umgangen. Hierbei werden die Werte ϕ_P für Produktion, Dissipation und Wandschubspannung am Wand-nächsten Punkt P sowohl für die viskose Unterschicht als auch den logarithmischen Teil der turbulenten Grenzschicht berechnet und mit einer einfachen Mischungsfunktion verknüpft:

$$\phi_P = \sqrt{\phi_{vis}^2 + \phi_{log}^2}. \tag{B.37}$$

Dabei stellt ϕ_P den effektiven Wert am Wand-nächsten Zellmittelpunkt dar, ϕ_{vis} den Wert in der viskosen Unterschicht und ϕ_{log} im logarithmischen Bereich der Grenzschicht.
Mit dieser Beziehung werden die effektiven Werte der Dissipation ω und des Produktionsterms P folgendermaßen bestimmt:

$$\omega_P = \sqrt{\omega_{P,vis}^2 + \omega_{P,log}^2} = \sqrt{\left[\frac{6\nu}{0.075 y_P^2}\right]^2 + \left[\frac{u_\tau}{0.3\kappa y}\right]^2}, \tag{B.38}$$

$$P_P = \sqrt{P_{vis}^2 + P_{log}^2} = \sqrt{\left[\nu_t \left(\frac{\partial v}{\partial y}\right)^2\right]^2 + \left[\frac{C_\mu^{3/4} k^{3/2}}{\kappa y_P}\right]^2}. \tag{B.39}$$

Die effektive Wandschubspannung τ_w

$$\tau_w = -\rho v_\tau^2 \tag{B.40}$$

wird mit Hilfe der in ähnlicher Weise gemischten Wandschubgeschwindigkeiten v_τ ermittelt:

$$v_\tau = \left(v_{\tau,vis}^4 + v_{\tau,log}^4\right)^{1/4}. \tag{B.41}$$

Die Wandschubgeschwindigkeiten in der Unterschicht und im logarithmischen Bereich werden wie folgt bestimmt:

$$v_{\tau,vis} = \frac{|v|}{y^+}, \tag{B.42}$$

$$v_{\tau,log} = \frac{|v|}{\frac{1}{\kappa}\ln y^+ + C}. \tag{B.43}$$

Für die Gleichung der turbulenten kinetischen Energie wird wieder eine Nullgradienten-Randbedingung in Wandnormalen-Richtung gesetzt, welche im Bereich der viskosen Unterschicht und im logarithmischen Bereich der Grenzschicht Gültigkeit bewahrt.
Die Dissipation am Wand-nächsten Rechenpunkt wird direkt mit Beziehung B.38 gesetzt.
Diese Mischung stabilisiert das numerische Verfahren und macht die Lösung im Wandnahen Bereich gitterunabhängig. Der Ansatz ist jedoch numerischer Natur und deswegen physikalisch nur begrenzt korrekt, weswegen ein sorgfältig erstelltes Netz in Wandnähe immer noch notwendig bleibt.
Bei sehr groben Rechennetzen muss wiederum die Standard-Wandfunktion verwendet werden.

Anhang C

Randbedingungen

C.0.7 Randbedingungen der Strömungsmechanik

Die geeignete Wahl und Einarbeitung der physikalischen Randbedingungen filtert aus der unendlichen Zahl der möglichen Lösungen eine eindeutige Lösung heraus. Es müssen geeignete Randbedingungen sowohl für Fluid- als auch Strukturbereiche bereichsweise gewählt werden und werden in den folgenden Abschnitten zusammengefasst.

C.0.7.1 Einlassrandbedingungen

Geschwindigkeiten
Die Geschwindigkeiten am Einlass *in* können direkt über einen konstanten Wert vorgegeben werden.

Ferner besteht die Möglichkeit, den Geschwindigkeitsvektor aus einem vorgegebenen konstanten Totaldruck p_0 und einer konstanten, vorgegebenen Geschwindigkeitsrichtung zu berechnen. Dazu wird der Betrag der Geschwindigkeit im inkompressiblen Fall mit der folgenden Beziehung ermittelt:

$$|\mathbf{v}_{in}| = \sqrt{\frac{2}{\rho}(p_0 - p)}, \tag{C.1}$$

mit der die Geschwindigkeitskomponenten mit Hilfe der gegebenen Geschwindigkeitsrichtung ermittelt werden. Diese Beziehung kann auch im Falle schwach kompressibler, isothermer Medien verwendet werden.

Bei der Untersuchung kompressibler Gase benötigt man zusätzlich zu Totaldruck und zu statischem Druck noch die statische Temperatur, um den benötigten Geschwindigkeitsbetrag zu ermitteln:

$$|\mathbf{v}_{in}| = \sqrt{2\frac{\kappa - 1}{\kappa} R T_{in} \left[\left(\frac{p_{0,in}}{p_{in}} \right)^{\frac{\kappa-1}{\kappa}} - 1 \right]}. \tag{C.2}$$

Diese Beziehung gilt prinzipiell nur für isentrope Strömungen, wird aber auch bei verlustbehafteten Strömungsvorgängen erfolgreich angewendet.

Statischer Druck
Bei inkompressiblen und schwach kompressiblen Strömungen oder bei kompressiblen Unterschallströmungen wird der statische Druck am Einlass durch Extrapolation von der Einlass-nächsten Rechenzelle bestimmt.

Bei kompressiblen Strömungen kann der statische Druck mit Hilfe des Totaldrucks und der Machzahl nach der isentropen Beziehung

$$p_{in} = p_{0,in} \left(1 + \frac{\kappa - 1}{\kappa} Ma_{in}^2\right)^{\frac{\kappa}{1-\kappa}} \tag{C.3}$$

bestimmt werden. Die Machzahl am Einlass $Main$ wird mit der aktuell vorliegenden Geschwindigkeit und statischen Temperatur bestimmt.

Temperatur
Die Festlegung der Temperatur am Einlass erfolgt durch direkte Festlegung der statischen Temperatur oder bei Vorgabe einer konstanten Totaltemperatur mit Hilfe der isentropen Beziehung:

$$T_{in} = T_{0,in} \left(1 + \frac{\kappa - 1}{\kappa} Ma_{in}^2\right)^{-1}. \tag{C.4}$$

Dichte
Die Dichte ergibt sich im kompressiblen Fall automatisch mit Hilfe eines geeigneten Zustandsgesetzes.

Turbulenzgrößen
Die turbulente kinetische Energie am Einlass wird aus dem Geschwindigkeitsbetrag am Einlass und dem Turbulenzgrad Tu ermittelt:

$$k_{in} = \frac{3}{2} Tu^2 |\mathbf{v}_{in}|^2. \tag{C.5}$$

Mit diesem Wert und einem charakteristischen Längenmaß L wird die turbulente Dissipation mit Hilfe von

$$\epsilon_{in} = \frac{k_{in}^{\frac{3}{2}}}{L} \tag{C.6}$$

bestimmt, bzw. die turbulente Frequenz mit

$$\omega_{in} = \frac{k_{in}^{\frac{1}{2}}}{C_\mu L}. \tag{C.7}$$

Von zahlreichen Autoren werden Werte für den Turbulenzgrad von $0.01 \leq Tu < 0.1$ vorgeschlagen. Der Wert für das charakteristische Längenmaß liegt zwischen $0.1 \leq L_{Tu} < 1.0$ einer für das untersuchte Problem charakteristischen Länge. Im Turbomaschinenbau wird als charakteristische Länge häufig der Durchmesser des Laufrades gewählt.

C.0.7.2 Auslassrandbedingungen

Im Fall einer inkompressiblen Strömung oder im Fall einer kompressiblen Unterschallabströmung wird der statische Druck am Auslass auf einen konstanten oder einen flächengemittelten Wert festgesetzt. Mit einem Mischungsfaktor α können diese beiden Zustände gemischt werden:

$$p = p_M + (1 - \alpha)p_C. \tag{C.8}$$

Hierin bezeichnet p_M den am Auslass über extrapolierte Druckwerte gemittelten Wert und p_C den konstanten vorgeschriebenen Wert am Auslass.

Lediglich im Fall einer kompressiblen Überschallabströmung wird auch der statische Druck am Auslass extrapoliert.

Die Geschwindigkeiten und alle weiteren Transportgrößen werden aus dem Inneren des Strömungsraum an den Auslass extrapoliert. Die Dichte ergibt sich automatisch mit Hilfe eines geeigneten Zustandsgesetzes.

C.0.7.3 Wandrandbedingungen

Bei undurchlässigen Wänden verschwindet der konvektive Fluss der angewendeten Transportgleichungen durch die entsprechenden Zellflächen. Trotzdem findet ein Impulsaustausch über den diffusiven Fluss statt.

Geschwindigkeiten

Durch die Haftbedingung nimmt die Relativgeschwindigkeit an der Wand den Wert Null an, $w_R = 0$. Der diffusive Fluss an der Wand wird mit Hilfe der Wandschubspannung τ_w ermittelt und diese in einem lokal aufgespannten Koordinatensystem aufgestellt. Das Koordinatensystem wird festgesetzt durch den Vektor in Strömungsrichtung, den Richtungsvektor normal zur Wand und den zu diesen beiden Vektoren orthogonalen Vektor.

Der tangentiale Richtungsvektor im wandnächsten Punkt P wird dabei über die wandnächste Relativgeschwindigkeit $\mathbf{w}_P$ und die wandnächste Relativgeschwindigkeit in Normalenrichtung $\mathbf{w}_{P,nor}$ ermittelt:

$$\mathbf{w}_{P,tan} = \mathbf{w}_P - \mathbf{w}_{P,nor}. \tag{C.9}$$

Dieses Vorgehen basiert auf der Annahme, das kein Richtungswechsel der Geschwindigkeit zwischen den Punkten P und R stattfindet, was nicht realistisch sein muss und von FIEREDER [36] adaptiert wurde.

Die Tangentialrichtung $\mathbf{t}$ ist nun gegeben durch:

$$\mathbf{t} = \frac{\mathbf{w}_{P,tan}}{|\mathbf{w}_{P,tan}|}. \tag{C.10}$$

Mit Hilfe der Approximation für die wirkende Schubspannung

$$\tau_w \approx \mu \frac{\mathbf{t}(\mathbf{w}_P - \mathbf{w}_R)}{y_P} \tag{C.11}$$

und den ermittelten Tangentenvektor wird nun der diffusive Fluss an der Wand ermittelt:

$$\mathbf{F}_W^d = \int_{A_w} \tau_w \mathbf{t} dS \approx \tau_w \mathbf{t} S_w. \tag{C.12}$$

Dieser Fluss wird mit einer nachgezogenen Korrektur in den Impulsgleichungen auf die Quellterme geschrieben:

$$\mathbf{F}_W^d \approx \left(\mathbf{u}\frac{S_W}{y_P}\right)^m + \left[-\mathbf{u}\frac{S_W}{y_P} + \mu\mathbf{t}\frac{\mathbf{t}(\mathbf{w}_P - \mathbf{w}_W)}{y_P}\right]^{m-1}, \tag{C.13}$$

wodurch die Koeffizientenmatrizen für alle drei Komponenten der Impulsgleichungen identisch bleiben.

Druck

Der Druck und die Druckkorektur an den Wänden werden durch Extrapolation bestimmt.

Temperatur

Die Temperatur wir mit einem Nullgradienten an die Wand extrapoliert. Dadurch wird eine adiabate Wand impliziert. Eine Erweiterung auf Wände mit Wärmeleitung ist in einfacher Weise möglich, s. EINZINGER [34].

Dichte

Die Dichte ergibt sich im kompressiblen Fall mit Hilfe eines geeigneten Zustandsgesetzes.

Turbulenzgrößen

Die speziellen Wandrandbedingungen für die turbulenten Größen werden in Abschnitt B.0.5 ausführlich dargestellt.

C.0.7.4 Symmetrierandbedingungen

Geschwindigkeiten

Die zu einer Symmetrierandbedingung normalen Geschwindigkeiten verschwinden. Die Schubspannungen normal zu den Symmetrierändern lassen sich berechnen mit:

$$\tau_n = 2\mu\frac{\partial w_n}{\partial n} \tag{C.14}$$

und werden dann als zusätzliche diffusive Flüsse ähnlich wie bei reibungsbehafteten Wänden den Impulsgleichungen aufgeschlagen. Durch Anwendung einer nachgezogenen Korrektur wird weiterhin eine identische Koeffizientenmatrix der Komponenten der Impulsgleichungen beibehalten:

$$\mathbf{F}_S^d \approx \left(\mathbf{u}\frac{S_W}{y_P}\right)^m + \left[-\mathbf{u}\frac{S_W}{y_P} + \mu\mathbf{n}\frac{\mathbf{n}(\mathbf{w}_P - \mathbf{w}_W)}{y_P}\right]^{m-1}, \tag{C.15}$$

Druck, Temperatur, Turbulenzgrößen
Druck, Temperatur und Turbulenzgrößen sowie alle weiteren skalaren Transportgrößen werden mit einem Nullgradienten extrapoliert.

Dichte
Die Dichte ergibt sich im kompressiblen Fall mit Hilfe eines geeigneten Zustandsgesetzes.

C.0.8 Randbedingungen der Strukturmechanik

C.0.8.1 Verschiebungs -und Temperaturrandbedingungen

Die Vorgabe des Verschiebungsvektors oder der Temperatur an einem Strukturrand wird über die Implementierung einer Dirichlet-Randbedingung für diese Größen realisiert.
Der Verschiebungsgradient an diesen Ränder wird aus dem Feldinneren extrapoliert, wodurch die Spannung berechnet werden kann.

C.0.8.2 Spannungsrandbedingungen

Spannungen, Kräfte und freie Oberflächen
Die Vorgabe von Spannungen oder Kräften wird über Neumann-Randbedingungen realisiert. Es können alle sechs Komponenten des symmetrischen Spannungstensors oder der Druck vorgegeben werden. Der resultierende Kraftvektor $\mathbf{f}$ wird dann mit Hilfe des Rand-Normalenvektors und der Rand-Zellfläche ermittelt

$$\mathbf{f} = \boldsymbol{\sigma}\mathbf{n}A - p\mathbf{I}A \tag{C.16}$$

und wird direkt auf die Quellterme der entsprechenden Impulsgleichungen aufgeschlagen.

Die Verschiebungen an Spannnungsrändern werden über eine lineare Extrapolation bestimmt.

Freie Oberflächen sind ein Sonderfall der Spannungs- und Kraftrandbedingung und werden analog behandelt.

Temperatur
Die Temperatur an einem Strukturrand kann direkt vorgegeben und über eine Dirichlet-Randbedingung implementiert werden oder der Temperaturfluss wird vorgegeben und über eine Neumann-Randbedingung eingearbeitet.

C.0.8.3 Symmetrierandbedingungen

Verschiebungen
Der Verschiebungsvektor wird in tangentiale Richtung zum Symmetrierand gedreht

und extrapoliert und dann für die aktuelle äußere Iteration als vorgegeben betrachtet und mit einer Dirichlet-Randbedingung in das Gleichungssystem eingearbeitet.

Temperatur

Die Temperatur wird über einen Nullgradienten an den Symmetrierand extrapoliert.

Anhang D

Lineare Löser

Mit dem folgenden akademischen Testfall werden die unterschiedlichen implementierten linearen Löser ILU(0), ICCG und das Mehrgitterverfahren nach Abs. 4.5.3.2 miteinander verglichen.
Dazu wurde die Laplacegleichung $\Delta\boldsymbol{\phi} = \mathbf{0}$ mit der vorgestellten Finite-Volumen-Methode sowohl auf kartesisch als auch auf polyedrisch vernetzten, zweidimensionalen Gebieten, s. Abs. 6.1.1.2, diskretisiert und gelöst. An den Rändern werden als Dirichlet-Randbedingung die Werte $\phi_{Rand} = xy$ vorgegeben, als Initiallösung im Feldinneren die Initial-Werte $\phi_0 = 0$.

Das angestrebte Konvergenzkriterium ist eine Reduktion der Summenbetragsnorm des Residuum um acht Größenordnungen. Beim Mehrgitterverfahren wurde ein V-Zyklus angewendet und die Glättungsschritte wurden mit der ILU(0) durchgeführt.

Die Residuenverläufe für die Lösungen auf dem feinsten kartesischen und dem feinsten polyedrischen Netz sind für die unterschiedlichen Verfahren in Abb. D.1

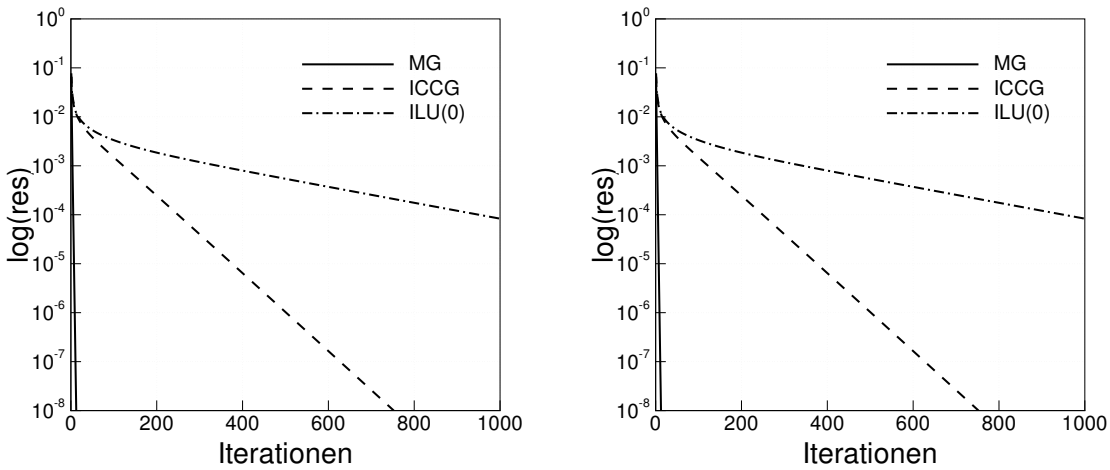

Abbildung D.1: Residuenverläufe der implementierten Gleichungslöser bei Lösung einer Laplacegleichung, welche auf kartesischen und polyedrischen Rechennetzen mit 65536 und 64376 Finite-Volumen-Zellen diskretisierte wurde.

abgebildet.

Tabelle D.1 und Tabelle D.2 zeigen alle durchgeführten Rechenläufe im Vergleich hinsichtlich Iteration bzw. Zyklen-Anzahl und Zeit. Das Konvergenzverhalten wird bei allen Verfahren bei der polyedrischen Vernetzung schlechter. Dies liegt an den nachgeführten Korrekturen für die Nichtorthogonalitäten in der vorgestellten Diskretisierungstechnik, s. Abs. 4.2.4.
Das Mehrgitterverfahren zeigt hinsichtlich der Anzahl der verwendeten Zyklen ein von der Netzfeinheit unabhängiges Konvergenzverhalten. Auf den feinsten Rechennetzen ist es den anderen iterativen Lösungsverfahren deutlich überlegen.

N	N/N_1	i_{MG}	i_{ICCG}	i_{ILU}	$t_{MG}/t_{MG,1}$	$t_{ICCG}/t_{MG,1}$	$t_{ILU}/t_{MG,1}$
4096	$1,0$	13	55	236	$1,0$	$1,4$	$3,2$
16384	$4,0$	13	200	906	$3,4$	$7,2$	$24,6$
65536	$16,0$	13	753	3433	$14,4$	$76,0$	$318,4$

Tabelle D.1: Verlgeich der Gleichungslöser bei der Lösung einer Laplacegleichung auf *kartesischen* Rechennetzen hinsichtlich Iterationszahlen und Lösungszeit.

N	N/N_1	i_{MG}	i_{ICCG}	i_{ILU}	$t_{MG}/t_{MG,1}$	$t_{ICCG}/t_{MG,1}$	$t_{ILU}/t_{MG,1}$
4127	1	42	70	317	$1,0$	$0,7$	$2,1$
16245	$3,9$	42	254	1181	$3,42$	$4,5$	$16,5$
64376	$15,6$	43	968	4509	$17,5$	$55,66$	$238,6$

Tabelle D.2: Verlgeich der Gleichungslöser bei der Lösung einer Laplacegleichung auf *polyedrischen* Rechennetzen hinsichtlich Iterationszahlen und Lösungszeit.

Literaturverzeichnis

[1] http://glaros.dtc.umn.edu/gkhome/views/metis. (Stand 09.2011)

[2] http://www.cfx-germany.com. (Stand 09.2011)

[3] http://www.grc.nasa.gov/WWW/wind/valid. (Stand 09.2011)

[4] http://www.mcs.anl.gov/research/projects/mpi. (Stand 09.2011)

[5] http://www.nafems.org/regional/dach/. (Stand 09.2011)

[6] http://www.openfoam.com. (Stand 09.2011)

[7] Acosta, J. M.: *Numerical Algorithms for three Dimensional Computational Fluid Dynamics*. Universidad Politecnica de Catalunya, Terrassa, tesis doctoral, 2001

[8] A.J.A., Favre: *Formulation of the Statistical Equations of Turbulent Flows with Variable Density*. 1. Auflage. Springer, Berlin, 2002

[9] Alexandrescu, Andrei: *Modern C++ Design: Generic Programming and Design Patterns Applied*. Addison-Wesley Professional, München, 2001

[10] Andelfinger, U. ; Ramm, E.: EAS-Elements for two-dimensional, three-dimensional, Plate and Shell Structures and their equivalence to HR-Elements. In: *International Journal for Numerical Methods in Engineering* 36 (1993), S. 1311–1337

[11] Argyris, J. H. ; Mlejnek, H.-P.: *Die Methode der Finiten Elemente in der Elementaren Strukturmechanik. Band 1: Verschieburgsmethode in der Statik*. Vieweg, Braunschweig, Wiesbaden, 1986

[12] Aurenhammer, F. ; Klein, R.: *Handbook of Computational Geometry*. Elsevier Science Publishing, 1999

[13] Bader, R.: *Simulation kompressibler und inkompressibler Strömungen in Turbomachinen*. Technische Universität München, Lehrstuhl und Laboratorium für Hydraulische Maschinen und Anlagen, Dissertation, 2000

[14] Bardina, J. ; Ferziger, J.H. ; Rogallo, R.S.: Effect of Rotation on isotropic Turbulence. In: *Journal of Fluid Mechanics* 154 (1985), S. 321–336

[15] Bathe, K.-J.: *Finite-Elemente-Methoden: Matrizen und lineare Algebra, die Methode der finiten Elemente, Lösung von Gleichgewichtsbedingungen und Bewegungsgleichungen.* Springer, Berlin, 1986

[16] Batina, J.T.: Unsteady Euler algorithm with unstructured dynamic mesh for complex aircraft aeroelastic analysis. In: *AIAA* 89-1189 (1989)

[17] Bentley, J. L.: Multidimensional binary search trees used for associative searching. In: *Communications of the ACM* 18(9) (1975), S. 509–517

[18] Böhm, C.: *Numerische Simulation des Fischdurchgangs durch Wasserturbinen*, Technische Universität München, Dissertation, 2004

[19] Bijelonja, I. ; Demirdzic, I. ; Muzaferija, S.: Finite volume method for contact problems of linear elastic solids. In: *Computational Methods Applied Mechanical Engineering* 195 (2006), S. 6378–6390

[20] Bogner, M.: *Modulare und objektorientierte Implementierung der Finite-Volumen-Methode (FVM) zur Simulation kavitierender Strömungen in Turbomaschinen.* Technische Universität München, Lehrstuhl für Fluidmechanik, Dissertation, 2010

[21] Bogner, M. ; Flurl, B. ; Schilling, R.: *Simulation of Cavitating Flows in Hydraulic Turbomachinery.* CMFF09, Budapest, Hungary, 2010

[22] Boussinesq, J.: Essai sur la théorie des eaux courantes. In: *Mémoires Acad. des Sciences* 23 (1872), Nr. 1

[23] Brandt, A.: Multi-Level Adaptive Solutions to Boundary Value Problems. In: *Math. Comp.* 31 (1977), S. 333–390

[24] Brandt, A.: Algebraic multigrid theory: The symmetric case. In: *Appl. Math. Comp.* 19 (1986), S. 23–56

[25] Brandt, A. ; McCormick, S.F. ; Ruge, J.: Algebraic multigrid (AMG) for sparse matrix equations, in Sparsity and its Applications, D.J. Evans, ed. In: *Cambridge University Press, Cambridge, UK* (1984), S. 257–284

[26] Briggs, W.L. ; Henson, V.E. ; McCormick, S. F.: *A Multigrid Tutorial.* SIAM, Philadelphia, 2000

[27] Brown, K. Q.: Voronoi Diagrams from Convex Hulls. In: *Inf. Process. Lett. 9* 5 (1979), S. 223–228

[28] Darwish, M. ; Asmar, D. ; Moukalled, F.: A Comparative Assessment within a Multigrid Environment of Segregated Pressure Based Algorithms for Fluid Flow at All Speeds. In: *Numerical Heat Transfer* 45 (2004), S. 49–74

[29] Demirdzic, I. ; Hormann, I. ; Martinovic, D.: Finite volume analysis of stress and deformation in hygro-thermo-elastic orthotropic body. In: *Computational Methods Applied Mechanical Engineering* 190 (2000), S. 1221–1232

[30] Demirdzic, I. ; Martinovic, D.: Finite volume method for thermo-elasto-plastic stress analyis. In: *Computational Methods Applied Mechanical Engineering* 109 (1993), S. 331–349

[31] Demirdzic, I. ; Muzaferija, S.: Finite volume method for stress analysis in complex domains. In: *International Journal for numerical methods in engineering* 37 (1994), S. 3751–3766

[32] Demirdzic, I. ; Muzaferija, S. ; Peric, M.: Benchmark solutions of some structural analysis problems using finite-volume method and multigrid acceleration. In: *International Journal for numerical methods in engineering* 40 (1997), S. 1893–1908

[33] Durbin, P. A. ; Pettersson Reif, B. A.: *Statistical theory and modelling for turbulent flows.* Wiley, New York, 2000

[34] Einzinger, J.: *Numerische Simulation der Fluid-Struktur Interaktion in Turbomaschinen.* Technische Universität München, Lehrstuhl und Laboratorium für Hydraulische Maschinen und Anlagen, Dissertation, 2006

[35] Estorff, O. ; Schilling, R. ; Flurl, B. ; Michels, T.: Numerische Simulation der Schallentstehung und Schallausbreitung in Kreiselpumpen II. In: *Abschlussbericht zum AIF-Vorhaben 14566N* (2010)

[36] Fiereder, R.: *Mündliche Mitteilungen*, Technische Universität München, Diss.

[37] Fiereder, R.: *Numerische Simulation einer Pelton-Turbine*, Technische Universität München, Dissertation, 2011

[38] Fletcher, R.W.: Conjugate Gradients Methods for Indefinite Systems with Complex Systems. In: *In Dundee Biennial Conference on Numerical Analysis, G.A. Watson, ed.* (1975), S. 73–89

[39] Frobenius, Moritz: *Numerische Simulation kavitierender Strömungen in hydraulischen Strömungsmaschinen.* Technische Universität München, Lehrstuhl für Fluidmechanik, Dissertation, 2003

[40] Ghia, Ghia ; Shin: High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method. In: *Journal of Computational Physics* 48 (1982), S. 387–411

[41] Gostelow, J. P. ; Horlock, J. H.: Potential Flow through Cascades - A Comparison between Exact and Approximate Solutions / Ministry of Aviation, Aeronautical Research Council. 1964 (C.P. No. 807). – Current Papers. – UB/TIB Hannover 112 607 950

[42] Greenshields, C.J. ; Weller, H.G.: A unified formulation for continuum mechanics applied to fluid-structure interaction in flexible tubes. In: *International Journal for numerical methods in engineering* 64 (2005), S. 1575–1593

[43] Gross, D. ; Hauger, W. ; Schnell, W. ; Wriggers, P.: *Technische Mechanik 2.* 1. Auflage. Springer, Berlin, 1995

[44] Gross, D. ; Hauger, W. ; Schnell, W. ; Wriggers, P.: *Technische Mechanik 4*. 1. Auflage. Springer, Berlin, 1995

[45] Grotjans, H.: *Turbulenzmodelle höherer Ordnung für komplexe Anwendungen*. Technische Universität München, Lehrstuhl für Strömungsmechanik, Dissertation, 2006

[46] Hackbusch, W.: *Multi-Grid Methods and Applications*. Springer Verlag, Berlin, 1985

[47] Hackbusch, W.: *Iterative Lösung großer schwachbesetzter Gleichungssysteme*. Teubner Studienbücher, 1991

[48] Hackbusch, W.: Iterative Lösung großer schwach besetzter Gleichungssysteme. In: *Leitfäden der angewandten Mathematik und Mechanik*. Stuttgart : Teubner-Studienbücher, 1993 (Mathematik)

[49] Harten, A. ; Lax, P. D.: On A Class Of High Resolution Total-Variation-Stable Finite-Difference Schemes. In: *SIAM J. NUMER. ANAL* 21 (1984), February, Nr. 1

[50] Hermann, S.: *N.N.* Technische Universität München, Lehrstuhl für Fluidmechanik, Dissertation, 2011

[51] Hermann, S. ; Flurl, B. ; Schilling, R.: *Ein Beitrag zur numerischen Simulation von Schraubspindelpumpen mit überlappenden Gittern*. VDI-Berichte / Tagungsbeiträge Energietechnik, Dortmund, 2010

[52] Hestenes, M.R. ; Stiefel, E.: Methods of conjugate gradients for solving linear systems. In: *Nat. Bur. Standards, J. of Res.* 49 (1952), S. 409–436

[53] Hirsch, C.: *Fundamentals of Numerical Discretization*. Chichster New York Brisbane : John Wiley & Sons, 1988 (Numerical Computation of Internal and External Flows)

[54] Howard, J.H. ; Patankar, S.V. ; Bordynuik, R.M.: Flow prediction in rotating ducts using Coriolis-modified turbulence models. In: *Journal of Fluids Engineering* 102 (1980), S. 456–461

[55] Hron, J. ; Turek, S.: A monolithic FEM/multigrid solver for ALE formulation of fluid structure interaction with application in biomechanics. In: *Fluid-Structure Interaction: Modelling, Simulation, Optimisation* (2006)

[56] Issa, R.I.: Solution of implicitly discretized fluid flow equations by operator-splitting. In: *Journal of Computational Physics* 62 (1986), S. 40–65

[57] Ivankovic, A. ; Demirdzic, I. ; Williams, J.G. ; Leevers, P.S.: Application of the finite volume method to the analysis of dynamic fracture problems. In: *International Journal of fracture* 66 (1994), S. 357–371

[58] Ivankovic, A. ; Demirdzic, I. ; Williams, J.G. ; Leevers, P.S.: Finite volume method and multigrid acceleration in modelling of rapid crack propagation in full-scale pipe test. In: *Computational Mechanics* 20 (1997), S. 46–52

[59] Jasak, H.: *Error analysis and estimation in the Finite Volume method with applications to fluid*, Imperial College, University of London, PhD thesis, 1996

[60] Jasak, H. ; Weller, H.G.: Application of the Finite volume method and unstructured meshes to linear elasticity. In: *International Journal for numerical methods in engineering* 48 (2000), S. 267–287

[61] Jasak, H. ; Weller, H.G.: *Finite Volume Methodology for Contact Problems of Linear Elastic Solids.* 2001

[62] Khosla, P.K. ; Rubin, S.G.: A Diagonal Dominant Second-Order Accurate Implicit Scheme. In: *Computers & Fluids* 2 (1974), S. 207–209

[63] Klement, P.: *Vergleich verschiedener Turbulenzmodelle zur Berechnung realer Strömungen.* Technische Universität München, Dissertation, 1998

[64] Kroll, N.: Berechnung von Strömungsfeldern um Propeller und Rotoren im Schwebeflug durch die Lösung der Euler-Gleichungen / Deutsche Forschungsanstalt für Luft- und Raumfahrt, Institut für Entwurfsaerodynamik. Braunschweig, Juni 1989 (89-37). – Forschungsbericht

[65] Lanczos, C.: Solution of systems of linear equations by minimized iterations. In: *Journal of research of the National Bureau of Standards* 49 (1952), S. 33–52

[66] Launder, B.P. ; Spalding, D.B.: The numerical computation of turbulent flows. In: *Computer Methods in applied mechanics and engineering* 3 (1974), S. 269–289

[67] Lilek, Z. ; Muzaferija, S. ; Perić, M.: An implicit finite-volume method using nonmatching blocks of structured grid. In: *Numerical Heat Transfer, Part B* 32 (1997), S. 385–401

[68] Lippmann, H. ; Mannl, V.: *Einführung in die Kontinuumsmechanik.* Wintersemester 2002/2003. – Skriptum zur Vorlesung Grundlagenfach Technische Mechanik, Lehrstuhl für Werkstoffkunde und Werkstoffmechanik

[69] Maneeratana, K. ; A., Ivankovic: *A Finite Volume method for structural applications involving material and geometrical non-linearities.* 1999

[70] Mattiason, K.: Numerical Results from large deflection beam and frame problems analysed by means of elliptic integrals. In: *International journal for numerical methods in engineering* 17(1) (1981), S. 145–153

[71] Meijerink, J.A. ; Vorst, H.A. v. d.: An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix is a Symmetric M-Matrix. In: *Mathematics of Computation*

[72] Meister, A.: *Numerik linearer Gleichungssysteme.* 1. Auflage. Vieweg u. Sohn, Braunschweig, 1999

[73] Menter, F.R.: Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications. In: *AIAA Journal* 32 (1994), S. 1598–1605

[74] Müller, A.: *Objektorientierte Strukturen für Adaptive Multilevelverfahren zur Strömungssimulation*, Technische Universität München, Dissertation, 2000

[75] Mok, D.P. ; Wall, W.A. ; Ramm, E.: Accelerated iterative substructuring schemes for instationary fluid-structure interaction. In: *Computational Fluid and Solid Mechanics, K.J. Bathe, Ed., Elsevier Science Publishing* (2001), S. 1325–1328

[76] Muzaferija, S.: *Adaptive finite volume method for flow predictions using unstructured meshes and multigrid approach*, University of London, PhD Thesis, 1994

[77] Niemann, G.: *Maschinenelemente*. 1. Auflage. Springer, Berlin, 1981

[78] Papadakis, G.: A novel pressure-velocity formulation and solution method for fluid-structure interaction problems. In: *Journal of computational physics* 227 (2008), S. 3382–3404

[79] Patankar, S. V.: *Numerical Heat Transfer and Fluid Flow*. 1. Washington, London, New York : Hemisphere Publishing Corporation, 1980

[80] Patankar, S. V. ; Spalding, D. B.: A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows. In: *Int. Journal of Heat and Mass Transfer* 15 (1972), S. 1787–1806

[81] Peric, M. ; Ferzifer, J.H.: *Computational methods for fluid dynamics*. 3. Auflage. Springer, Berlin, 2002

[82] Pfeiffer, F.: *Einführung in die Dynamik*. Teubner Studienbücher, 1992 (Leitfäden der angewandten Mathematik und Mechanik 65)

[83] Raw, M.J.: A coupled algebraic multigrid method for the 3D Navier-Stokes equations. 49 (1995), S. 204–215

[84] Reinelt, R.: *Ein unstrukturiertes Finite-Volumen-Verfahren für hydraulische Strömungsmaschinen*, Technische Universität München, Dissertation, 2001

[85] Reynolds, O.: On the Dynamical Theory of Incompressible Viscous Fluids and the Determination of the Criterion. In: *Philosophical Transactions of the Royal Society of London, Series A* 186 (1895), S. 123

[86] Rhie, C.M. ; Chow, W.L.: A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation. In: *AIAA Journal* 21 (1983), S. 1525–1532

[87] Ruge, J. ; Stüben, K.: Algebraic multigrid. In: *Multigrid Methods, S.F. McCormick, ed., SIAM, Philadelphia* 3 (1987), S. 73–130

[88] Schilling, R. ; Bogner, M. ; Flurl, B. ; A., Jantzen ; Schmalhorst, C.: Development of a Numerical Test Rig for Hydraulic Turbomachinery. In: *International Rotating Equipment Conference 08* (2008)

[89] Schilling, R. ; Fernandez, A. ; Aschenbrenner, Th. ; Riedel, N. ; Bader, R.: Echtzeit-Entwurf von Beschaufelungen - Möglichkeiten und Grenzen. In: *Pumpentagung* (1996)

[90] Schmalhorst, C.: *Ein modulares System zur Nachrechnung und Optimierung hydraulischer Turbomaschinen radialer Bauweise*, Technische Universität München, Dissertation, 2010

[91] Schmalhorst, C. ; Flurl, B. ; Schilling, R.: Development of an Integrated Design System (IDS) for Hydraulic Machinery. In: *ESDA08* (2008)

[92] Schuster, M.: *Simulation gehäuseloser, hydraulischer Strömungsmachinen.* Technische Universität München, Lehrstuhl und Laboratorium für Hydraulische Maschinen und Anlagen, Dissertation, 2000

[93] Skoda, R.: *Numerische Simulation abgelöster und transitionaler Strömungen in Turbomaschinen.* Technische Universität München, Lehrstuhl für Fluidmechanik, Dissertation, 2003

[94] Sleijpen, G.L.G. ; Fokkema, D.R.: BICGSTAB(l) for Linear Systems involving Unsymmetric Matrices with Complex Spectrum. In: *Electronic Transactions on Numerical Analysis* 1 (1993), S. 11–32

[95] Stone, H.L.: Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations. In: *SIAM J. Numer. Anal.* 5 (1968), Nr. 3, S. 530–558

[96] Stroustrup, B.: *Die C++-Programmiersprache.* 3. Auflage. Addison-Wesley-Longman, 1998

[97] Timoshenko, S.P. ; Gere, J.M.: *Theory of Elastic Stability.* McGraw-Hill, 1961

[98] Tukovic, Z. ; Jasak, H.: Updated Lagrangian Finite volume Solver for large deformation dynamic response of elastic body. In: *Transactions of FAMENA* XXX-1 (2007), S. S. 1–16

[99] University of Tennessee (Hrsg.): *MPI: A message passing interface.* Knoxville, Tennessee: University of Tennessee, 1994

[100] Van Doormal, J.P. ; Raithby, G.D.: Enhancements of the SIMPLE method for predicting incompressible fluid flows. In: *Journal of Numerical Heat Transfer* 7 (1984), S. 147–163

[101] Vaněk, P. ; Mandel, J. ; Brezina, M.: Algebraic Multigrid based on smoothed aggregation for second and fourth order problems. In: *Computing* 56 (1996), S. 179–196

[102] Varga, R.S.: Factorization and normalized iterative methods. In: *In: Boundary problems in differential equations, R.E. Langer, ed.* 5 (1960), S. 121–142

[103] Vorst, H.A. v. d.: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems. In: *SIAM Journal on Scientific and Statistical Computing* 13 (1992), S. 631–644

[104] W., Vieser ; T., Esch ; F., Menter: Heat Transfer Predictions using Advanced Two-Equation Turbulence Models. In: *CFX Technical Memorandum* (2002)

[105] Wall, W.A.: *Fluid-Struktur-Interaktion mit stabilisierten Finiten Elementen.* Universität Stuttgart, Dissertation, 1999

[106] Walters, D.K. ; Leylek, J.H.: A systematic computational methodology applied to a three-dimensional film cooling flowfield. In: *ASME* 96 (1996), S. S. 351ff

[107] Wheel, M.A.: A finite volume method for analysing the bending deformation of thick and thin plates. In: *Computational Methods Applied Mechanical Engineering* 147 (1997), S. 199–208

[108] Wick, A.: *Ein numerisches Verfahren zur Strömungssimulation in zeitveränderlichen Gebieten mit integriertem Modell zur Gitternachführung*, Technische Universität Berlin, Dissertation, 2003

[109] Wilcox, D.C.: *Turbulence Modeling for CFD.* DCW Industries, La Canada, California., 1998

[110] Wunderer, R.: *Numerische Simulation der aktiven Strömungsbeeinflussung in Turbomaschinen*, Technische Universität München, Dissertation, 2009

[111] Zienkiewicz, O. C.: The Finite Element Method in Engineering Science. In: *3rd Edition, McGraw Hill.* London, 1977

[112] Zwart, P.J.: *The Integrated Space-Time Finite Volume Method*, University of Waterloo, Dissertation, 1999