

Lehrstuhl für Statik
der Technischen Universität München

**Effiziente Algorithmen zur partitionierten Lösung stark
gekoppelter Probleme der Fluid-Struktur-Wechselwirkung**

Thomas Georg Gallinger

Vollständiger Abdruck der von der Fakultät für Bauingenieur- und Vermessungswesen
der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender:

Univ.-Prof. Dr. rer. nat. Ernst Rank

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing. Kai-Uwe Bletzinger

2. Univ.-Prof. Dr.-Ing. habil. Michael Manhart

3. Prof. Dr.-Ing. Kurt Maute,
University of Colorado, Boulder, Colorado/USA

Die Dissertation wurde am 23.06.2010 bei der Technischen Universität München eingereicht und durch die Fakultät für Bauingenieur- und Vermessungswesen am 15.11.2010 angenommen.

Zusammenfassung

Diese Arbeit beschäftigt sich mit dem Aspekt der Effizienz bei der partitionierten Simulation von Problemen des Typs Fluid-Struktur-Wechselwirkung. Dabei werden eine starke Kopplung, große Verformungen und instationäres Verhalten angenommen. Es werden unterschiedliche Bereiche identifiziert, die auf die Effizienz des Gesamtsimulationskonzeptes Einfluss haben. Diese sind sowohl kopplungsmethodischer als auch programmtechnischer Art.

Die kopplungsmethodischen Aspekte beinhalten die Algorithmen zur Lösung des gekoppelten nichtlinearen Problems. Innerhalb einer impliziten Kopplung sind dies die Feldlöseransätze, die Bestimmung der Initialposition des Interfaces innerhalb eines Zeitschrittes und der Kopplungsalgorithmus. Bei der iterativen Lösung der nichtlinearen Feldgleichungen wird in dieser Arbeit das Konzept des angepassten Feldlösers entwickelt. Angepasste Feldlöser nutzen die Konvergenzeigenschaften des übergeordneten Kopplungsproblems zur Beschleunigung der Feldkonvergenz. Im numerischen Beispiel zeigt sich, dass dieser Ansatz im Vergleich zu einer konventionellen Methode die Gesamtrechenzeit um bis zu 23,7 % verringert. Die Bestimmung der Initialposition des Interfaces in der ersten Kopplungsiteration innerhalb eines Zeitschritts stellt einen weiteren wichtigen kopplungsmethodischen Aspekt dar. Hierbei werden diverse bereits bestehende Methoden hergeleitet, neue Methoden entwickelt und diese miteinander verglichen. Es zeigt sich, dass eine erweiterte Betrachtung der physikalischen Verhältnisse am Interface die prädiktive Qualität deutlich erhöht. Der direkte Vergleich zwischen der besten und der schlechtesten Methode ergibt in der Gesamtrechenzeit einen Unterschied von 59,2 %. Der dritte wichtige Aspekt beinhaltet die Wahl des Kopplungsalgorithmus als Kern einer gekoppelten Berechnung. Hierbei werden die Methoden relaxationsbasierte Fixpunktiteration nach Aitken, Vektorextrapolation, Quasi-Newton-Verfahren und inexakte Newton-Krylov-Verfahren mit einer Krylovraumevaluierung über finite Differenzen oder ein linearisiertes Ersatzproblem im Detail hergeleitet. Im numerischen Beispiel wird eine große Zahl an Einzelberechnungen durchgeführt, um Aussagen über Effizienz und Stabilität zu treffen. Vor allem in Bezug auf Effizienz treten sehr hohe Unterschiede auf. Zwischen dem langsamsten und dem schnellsten Verfahren liegt im Extremfall ein Faktor 11 in der Gesamtrechenzeit. Nach einer getrennten Untersuchung der Einzelaspekte werden diese miteinander kombiniert und so eine möglichst effiziente Methodenkombination festgelegt.

Die programmtechnischen Aspekte beinhalten die Softwareumgebung und das Parallelisierungskonzept. In dieser Arbeit wird eine Softwareumgebung entwickelt, die drei Programme für die Bereiche Struktur, Wechselwirkung und Fluid verwendet. Diese Methode setzt die Idee des partitionierten Ansatzes konsequent um und zeichnet sich durch eine besonders hohe Flexibilität aus, da jedes der Programme nur für einen spezifischen Bereich zuständig ist. Das Parallelisierungskonzept erlaubt die Aufteilung des Kopplungsinterfaces auf eine beliebige Anzahl an Prozessoren und ermöglicht so die massiv parallele Berechnung großer Probleme, ohne eine Beschränkung des Parallelisierungskonzeptes der Teilfeldlöser. Im numerischen Beispiel wird gezeigt, dass dadurch die parallele Skalierung und Effizienz der Teilfeldlöser auch bei einer gekoppelten und partitionierten Berechnung erhalten bleibt.

Abstract

This thesis deals with the aspect of efficiency in the partitioned simulation of fluid-structure interaction problems. Strong coupling, large deformations, and instationary behavior are assumed. Different areas are identified that play a major role in the efficiency of the whole simulation concept. These areas cover coupling as well as programming aspects.

Coupling aspects involve all algorithms used in the solution of the coupled nonlinear problem. Within implicit coupling, this includes the field solver approaches, the determination of the initial interface position within a time step, and the coupling algorithm itself. The so-called adaptive field solver-approach is developed within this thesis based on the iterative solution of the nonlinear field equations. Adaptive field solvers use the convergence properties of the overall coupled problem to speed up the field convergence. The numerical example shows that this approach can reduce the overall computation time by up to 23.7 % compared to conventional methods. Another important coupling aspect is the determination of the initial interface position in the first coupling iteration of a time step. Here, multiple methods are derived, developed and compared. It is seen that an enhanced view of the physical properties of the interface leads to a considerable improvement of the predictor quality. A direct comparison between the best and the worst method gives a difference of 59.2 % in the overall computation time. The third important aspect covers the choice of the coupling algorithm, which is the core of a coupled simulation. A detailed derivation is given for the following approaches: relaxation-based fixed-point iteration with Aitken factor, vector extrapolation, quasi-Newton method, and inexact Newton-Krylov methods with Krylov evaluation by finite differences or a linearized problem. Various computations are carried out to compare these approaches with respect to efficiency and stability. Especially regarding efficiency very high differences occur. Comparing the overall computation time of the slowest and the fastest approach, a maximal difference of a factor of 11 is observed. After a separate analysis of these coupling aspects, different combinations are examined. To conclude, a very efficient method combination is proposed.

Programming aspects involve the computational framework and the parallelization strategy. In this work a computational framework is developed based on three programs for the structure, coupling, and fluid field. This approach is a consistent realization of the partitioned idea. It offers high flexibility, because every program is only responsible for one specific area. The parallelization concept allows the decomposition of the coupled interface onto an unlimited number of processors. This enables the massively parallel computation of large problems without restrictions of the parallelization concept of the single field solvers. The numerical example shows that this approach maintains the parallel scaling and efficiency of the single field solvers used within a coupled and partitioned simulation.

Vorwort

Die vorliegende Arbeit entstand in den Jahren 2006 bis 2010 während meiner Zeit als wissenschaftlicher Assistent am Lehrstuhl für Statik der Technischen Universität München. Dabei waren viele Menschen beteiligt, denen ich hier meinen Dank aussprechen möchte.

An erster Stelle möchte ich meinem Doktorvater und Inhaber des Lehrstuhls, Herrn Univ.-Prof. Dr.-Ing. Kai-Uwe Bletzinger, danken. Er gab mir großen wissenschaftlichen Freiraum und ermöglichte mir so ein selbstbestimmtes und eigenverantwortliches Arbeiten, ohne welchen diese Arbeit nicht möglich gewesen wäre. Die wissenschaftlichen Diskussionen mit ihm haben diese Arbeit entscheidend geprägt und zum erfolgreichen Abschluss beigetragen.

Des Weiteren möchte ich Herrn Univ.-Prof. Dr. rer. nat. Ernst Rank für die Übernahme des Vorsitzes der Prüfungskommission danken. Bei Herrn Univ.-Prof. Dr.-Ing. habil. Michael Manhart und Herrn Prof. Dr.-Ing. Kurt Maute möchte ich mich für die Übernahme der weiteren Gutachten und Teilnahme an der Prüfungskommission bedanken. Insbesondere Herrn Maute gebührt Dank, da er die weite Anreise aus Amerika auf sich genommen hat.

Bei meinen Kollegen am Lehrstuhl, ehemaligen und aktuellen, und allen externen Partnern möchte ich mich ganz herzlich für die gemeinsame Zeit und die dabei entstandenen Freundschaften bedanken. Dies wird mir in bester Erinnerung bleiben.

Meiner Familie, meiner Mutter Maria und meinem Vater Georg, möchte ich danken, da sie mich immer unterstützt und an mich geglaubt haben.

Meiner baldigen Frau Regina gebührt der größte Dank. Sie hat mir immer Kraft gegeben und mich stets unterstützt - in guten wie in schlechten Zeiten. Und nicht zuletzt: Sie hat diese Arbeit in Bezug auf Rechtschreibung und Ausdruck korrigiert, was für jemand fachfremden eine nicht nur angenehme Aufgabe ist. Dafür und für alles andere bedanke ich mich bei Ihr.

München im November 2010
Thomas Georg Gallinger

Inhaltsverzeichnis

1. Einleitung	1
1.1. Motivation	1
1.2. Zielsetzung und Überblick	2
1.3. Kapitelübersicht	3
2. Strukturfeld	5
2.1. Herleitung der kontinuumsmechanischen Grundgleichungen	6
2.1.1. Kinematik	6
2.1.2. Impulsbilanz	7
2.1.3. Konstitutive Gleichungen	7
2.1.4. Schwache Form	8
2.2. Diskretisierung im Raum	10
2.3. Diskretisierung in der Zeit	12
2.3.1. Zeitintegrationsverfahren Generalized- α	13
2.3.2. Effektive Strukturgleichung	15
2.3.3. Parameterwahl	15
2.4. Lösung der effektiven Strukturgleichung	17
2.5. Strukturlösungsalgorithmus für implizite Kopplungsalgorithmen	19
2.6. Zusammenfassung	23
3. Fluidfeld	24
3.1. Herleitung der kontinuumsmechanischen Grundgleichungen	25
3.1.1. Massenerhaltung	25
3.1.2. Impulserhaltung	25
3.1.3. Konstitutive Gleichungen	26
3.1.4. Inkompressible Navier-Stokes-Gleichungen	26
3.2. Diskretisierung im Raum	29
3.3. Diskretisierung in der Zeit	31
3.4. Druckkorrekturverfahren PISO	32
3.5. Raum- und zeitveränderliches Fluidgebiet	35
3.6. Fluidlösungsalgorithmus für implizite Kopplungsalgorithmen	38
3.7. Zusammenfassung	40
4. Fluid-Struktur-Wechselwirkung im partitionierten Ansatz	41
4.1. Definition des gekoppelten Problems	42
4.1.1. Geometrie	42
4.1.2. Grundgleichungen der Teilfelder	42

4.1.3.	Kopplungsbedingungen	43
4.1.4.	Formulierung in Operatorschreibweise	44
4.1.5.	Einflussgrößen auf die Nichtlinearität des gekoppelten Problems	44
4.2.	Grundform eines impliziten Kopplungsalgorithmus	46
4.3.	Konvergenzkriterien	49
4.4.	Interfaceprädiktor	51
4.4.1.	Polynombasierte Prädiktoren	51
4.4.2.	Newmarkbasierte Prädiktoren	52
4.4.3.	Adaptive Prädiktoren	52
4.5.	Verfahrensüberblick zu Kopplungsalgorithmen	54
4.6.	Fixpunktiteration mit Relaxation	59
4.7.	Vektorextrapolation	61
4.8.	Newton-Verfahren	64
4.8.1.	Jakobi-freie Newton-Krylov-Verfahren	68
4.8.1.1.	Krylov-Unterraum-Verfahren GMRES	68
4.8.1.2.	Einbettung in den Newton-Kopplungsalgorithmus	70
4.8.1.3.	Wiederverwendung von Krylovrauminformationen	71
4.8.1.4.	Krylovraumevaluierung	73
4.8.1.5.	Methoden zur Krylovraumevaluierung	74
4.8.2.	Quasi-Newton-Verfahren	81
4.8.2.1.	Startprozedur	81
4.8.2.2.	Algorithmusbeschreibung	81
4.8.2.3.	Vergleich mit Jakobi-freien Newton-Krylov-Verfahren	82
4.8.2.4.	Vergleich mit weiteren Quasi-Newton-Verfahren	84
4.8.2.5.	Wiederverwendung von Informationen	84
4.8.2.6.	Verzicht auf den Relaxationsschritt	85
4.9.	Zusammenfassung	86
5.	Softwareumgebung	87
5.1.	Strukturfeldlöser Carat++	89
5.1.1.	Verwendung geometrisch reduzierter Strukturmodelle	89
5.2.	Fluidfeldlöser OpenFOAM	91
5.3.	Kopplungssoftware CoMA	93
5.3.1.	Parallelisierungsstrategie	96
5.3.2.	Datentransfer zwischen Oberflächennetzen	98
5.3.2.1.	Netz- und Datenbeschreibung	99
5.3.2.2.	Datentransfermethoden	101
5.3.3.	Simulationssteuerung und Kopplungsalgorithmen	104
6.	Numerische Beispiele	108
6.1.	Numerischer DFG-Benchmark	109
6.1.1.	Problemdefinition	110
6.1.1.1.	Geometrie und Physik	110
6.1.1.2.	Numerisches Modell	111

6.1.2.	Simulationsergebnisse	113
6.1.2.1.	Fall FSI2	114
6.1.2.2.	Fall FSI3	115
6.1.3.	Konstante versus angepasste Feldlöser	117
6.1.3.1.	Fall FSI2	119
6.1.3.2.	Fall FSI3	120
6.1.3.3.	Schlussfolgerung	120
6.1.4.	Einfluss des Interfaceprädiktors	120
6.1.4.1.	Fall FSI2	121
6.1.4.2.	Fall FSI3	122
6.1.4.3.	Schlussfolgerung	123
6.1.5.	Einfluss des Kopplungsalgorithmus	124
6.1.5.1.	Fall FSI2	125
6.1.5.2.	Fall FSI3	127
6.1.5.3.	Schlussfolgerung	129
6.1.6.	Kombination der Methoden	130
6.1.7.	Schlussfolgerung	131
6.2.	Bogenhalle	133
6.2.1.	Problemdefinition	134
6.2.1.1.	Geometrie und Physik	134
6.2.1.2.	Numerisches Modell	136
6.2.2.	Vergleich von Kopplungsmethoden	141
6.2.3.	Parallele Skalierung	142
6.2.4.	Simulationsergebnisse	143
6.2.5.	Schlussfolgerung	149
6.3.	Zusammenfassung	150
7. Zusammenfassung und Ausblick		151
7.1.	Zusammenfassung	151
7.2.	Ausblick	153
A. Anhang		155
A.1.	Numerischer DFG-Benchmark - Vergleichsergebnisse	155
Literaturverzeichnis		158

Abbildungsverzeichnis

4.1. Geometriedefinition bei oberflächengekoppelten Problemen	42
4.2. Explizites (links) und implizites (rechts) Kopplungsschema	46
4.3. Überblick über partitionierte und implizite Kopplungsalgorithmen	55
5.1. Geometrische Reduktion durch Diskretisierung - Schnitt durch einen Zylinder	90
5.2. Erzeugung des Oberflächennetzes mittels Projektion	91
5.3. Softwareumgebung	93
5.4. Klassenstruktur des Prozessgruppenkonzepts in CoMA	96
5.5. Kommunikatorkonzept in CoMA	97
5.6. Netzkonzept in CoMA	100
5.7. Projektionsbasierter Transfer von Verschiebungen	103
5.8. Projektionsbasierter Transfer von Lasten	104
5.9. Kopplungsarten in CoMA: Einseitige, explizit zweiseitige und implizit zweiseitige Kopplung	105
6.1. Geometrie und Randbedingungen des Fluidfeldes	110
6.2. Geometrie und Randbedingungen des Strukturfeldes	110
6.3. Fluidnetzdetail um den Kopplungsbereich	111
6.4. Randbedingungen am Strukturmodell	112
6.5. FSI2 - Verschiebungsverlauf am Punkt A in x- and y-Richtung	114
6.6. FSI2 - Verlauf der summierten Kräfte am Interface in x- und y-Richtung	115
6.7. FSI2 - Geschwindigkeitsvisualisierung des Strömungsfeldes an drei Zeitpunkten	116
6.8. FSI3 - Verschiebungsverlauf am Punkt A in x- and y-Richtung	117
6.9. FSI3 - Verlauf der summierten Kräfte am Interface in x- und y-Richtung	117
6.10. FSI3 - Geschwindigkeitsvisualisierung des Strömungsfeldes an drei Zeitpunkten	118
6.11. Tragwerksentwurf der Bogenhalle	133
6.12. Geometrie und Abmessungen des vereinfachten Modells	134
6.13. Strukturmodell - Formfindungsprozess	137
6.14. Fluidmodell - Simulationsgebiet	138
6.15. Fluidmodell - Netzdetail im Bereich der Bogenhalle (Schnitt in der x-z-Ebene, links) und Oberflächennetz der Bogenhalle (rechts)	139
6.16. Windeinlass - Vergleich von mittleren und generierten Geschwindigkeiten in drei Höhen	140
6.17. Geschwindigkeitsverteilung (Schnittnormal zur y-Achse)	144

6.18. Stromlinien (Erzeugende entlang y -Achse)	145
6.19. Stromlinien (Erzeugende entlang z -Achse)	145
6.20. Druckverteilung an der Membranoberfläche (links: Innenseite, rechts: Aus- senseite)	146
6.21. Verformungen der Membranoberfläche bei $t = 0,8 s$ (links: x -Komponente, rechts: z -Komponente)	146
6.22. Isogeometrische Darstellung der Verformungsamplitude bei $t = 0,8 s$. . .	147
6.23. Position der Auswertungspunkte auf der Hallenoberfläche	148
6.24. Zeit-Verschiebungsdiagramm der Punkte A, B und C	148

Tabellenverzeichnis

2.1. Parameteranforderungen des Generalized- α Verfahrens	16
2.2. Parameterkombinationen für verschiedene Spektralradien	16
5.1. Verwendung von CoMA	94
6.1. Physikalische Parameter der Fälle FSI2 and FSI3	111
6.2. FSI2 - eigene Ergebnisse und Referenzwerte	114
6.3. FSI3 - eigene Ergebnisse und Referenzwerte	115
6.4. Konstante versus angepasste Feldlöser - FSI2	119
6.5. Konstante versus angepasste Feldlöser - FSI3	120
6.6. Prädiktoreinfluss FSI2 - $\Delta t = 5,0e^{-04} s$	122
6.7. Prädiktoreinfluss FSI2 - $\Delta t = 20,0e^{-04} s$	122
6.8. Prädiktoreinfluss FSI3 - $\Delta t = 2,5e^{-04} s$	123
6.9. Prädiktoreinfluss FSI3 - $\Delta t = 10,0e^{-04} s$	123
6.10. Algorithmuseinfluss FSI2 - $\Delta t = 5,0e^{-04} s$	125
6.11. Algorithmuseinfluss FSI2 - $\Delta t = 20,0e^{-04} s$	126
6.12. Algorithmuseinfluss FSI3 - $\Delta t = 2,5e^{-04} s$	127
6.13. Algorithmuseinfluss FSI3 - $\Delta t = 10,0e^{-04} s$	128
6.14. Effizienzsteigerung durch Kombination aller Methoden - Vergleich der re- lativen Rechenzeit	131
6.15. Physikalische Parameter der Bogenhalle	135
6.16. Physikalische Parameter der Windströmung	136
6.17. Abmessungen des numerischen Windkanals	138
6.18. Vergleich der Effizienz von Standard- und effizienter Kombination	142
6.19. Vergleich der parallelen Beschleunigung von Einzelfeld- und gekoppelter Simulation	143
A.1. FSI2 - eigene Ergebnisse und Bandbreite der Ergebnisse weiterer Gruppen	155
A.2. FSI3 - eigene Ergebnisse (Gruppe 7) und Ergebnisse weiterer Gruppen ([THR ⁺ 10]) - Teil 1	156
A.3. FSI3 - eigene Ergebnisse (Gruppe 7) und Ergebnisse weiterer Gruppen ([THR ⁺ 10]) - Teil2	157

Algorithmenverzeichnis

1.	Strukturlösungsalgorithmus für implizite Kopplungsalgorithmen	21
2.	Generalized- α Zeitintegrationsalgorithmus in der nichtlinearen Struktur- dynamik - Einzelfeldberechnung	22
3.	PISO-Druckkorrekturverfahren	34
4.	Fluidlösungsalgorithmus für implizite Kopplungsalgorithmen	39
5.	Grundform eines impliziten Dirichlet-Neumann Kopplungsalgorithmus . .	47
6.	Kopplungsalgorithmus basierend auf Vektorextrapolation	63
7.	Grundform eines Kopplungsalgorithmus auf Basis eines Newton-Verfahrens	66
8.	Newton-Kopplungsalgorithmus unter Verwendung des GMRES-Verfahrens	72
9.	Fluidlösungsalgorithmus für Newton-Krylov-Kopplungsalgorithmen	74
10.	Strukturlösungsalgorithmus für Newton-Krylov-Kopplungsalgorithmen . .	75
11.	Quasi-Newton-Kopplungsalgorithmus	83
12.	Datenzugriff bei partitionsbasierter Netzbeschreibung	100
13.	Ablaufalgorithmus in CoMA bei zweiseitiger impliziter Kopplung einer Fluid-Struktur-Wechselwirkung	107

1. Einleitung

1.1. Motivation

Die numerische Simulation eines gekoppelten Problems des Typs Fluid-Struktur-Wechselwirkung hat das Ziel, die Interaktion zwischen einer Fluidströmung und einer elastischen oder inelastischen Struktur abzubilden. Diese Art von Problem findet sich in vielen Bereichen von Natur und Technik.

Ein Beispiel aus dem Alltag stellt das Flattern einer Fahne im Wind dar. Bei Windstille hängt die Fahne durch das Fehlen jeglicher Biegesteifigkeit schlaff herab. Kommt Wind auf, beginnt die Fahne zu flattern. Dieser Flattervorgang ist das Ergebnis einer Wechselwirkung zwischen Struktur und Fluid. Durch die großen Verformungen verändert sich die Umströmung. Dies führt zu einer Änderung der Druckverteilung auf die Oberfläche der Fahne. Dies wiederum führt zu einer Änderung der Verformung der Fahne. Und so weiter. Dabei befindet sich das gekoppelte System zu jedem Zeitpunkt im Gleichgewicht. Innerhalb des Ingenieurwesens treten gekoppelte Probleme beispielsweise in der Biomechanik bei der Blutströmung in elastischen Arterien, der Aeroelastik bei der Verformung von Flugzeugtragflügeln in einer Luftströmung, oder in der Gebäudeaerodynamik bei der Umströmung leichter und biegeweicher Flächentragwerke auf.

Die Abbildung dieser Phänomene innerhalb eines Computermodells ist komplex und aufwendig. Bei vielen Fragestellungen ist der Einfluss der Wechselwirkung auch relativ gering und kann deshalb vernachlässigt werden. Bei einigen Fragestellungen jedoch ist die Wechselwirkung ein maßgebender Faktor, und muss daher für ein vertieftes Verständnis des Verhaltens abgebildet werden.

Bei der numerischen Simulation der Fluid-Struktur-Wechselwirkung wird oftmals ein partitionierter Ansatz verwendet. Dies bedeutet, dass die einzelnen Partitionen des gekoppelten Problems, das Strukturfeld und das Fluidfeld, mittels eigenständiger Programme und damit Ansätze abgebildet werden. Die Wechselwirkung zwischen beiden wird über Vorgabe von Randbedingungen abgebildet. Der größte Vorteil des partitionierten Ansatzes ist die Möglichkeit, bereits existierende Software wiederverwenden zu können. Diese Feldlöser für Struktur- und Fluidfeld beinhalten bereits eine große Anzahl an Lösungsmethoden und -techniken zur Abbildung des spezifischen Feldverhaltens. Deren Wiederverwendung stellt eine große Erleichterung und Arbeitersparnis dar.

Besteht allerdings eine starke Kopplung zwischen den beiden Feldern, wie im Beispiel der Fahne im Wind, ist die Simulation mittels eines partitionierten Ansatzes von sehr hohem numerischen Aufwand und mit extrem langen Rechenzeiten verbunden. Damit ist die Anwendung dieser Methode auf reale Problemstellungen der Ingenieurspraxis bislang sehr schwierig.

1.2. Zielsetzung und Überblick

In dieser Arbeit wird ein partitionierter Ansatz für die Simulation von Problemen der Fluid-Struktur-Wechselwirkung verwendet. Dabei werden stark gekoppelte und transiente Problemstellungen untersucht. Das Ziel ist es, diese Art von Simulation möglichst effizient durchzuführen und durch die Reduktion der Rechenzeit die Simulation realistischer Problemstellungen aus der Praxis zu ermöglichen. Effizienz bezieht sich hierbei sowohl auf ein effizientes Softwarekonzept, als auch auf effiziente Algorithmen zur Lösung des nichtlinearen gekoppelten Problems.

Die Ausgangsbasis stellen zwei Feldlöser mit standardisierten Lösungsansätzen dar. Darauf aufbauend wird eine Softwareumgebung entwickelt, welche sich durch zwei Aspekte auszeichnet.

Der erste Aspekt wird als Drei-Programme-Ansatz bezeichnet und stellt die Weiterentwicklung des klassischen partitionierten Ansatzes dar. Üblicherweise wird im partitionierten Ansatz die physikalische Aufteilung des Mehrfeldproblems auf die Software übertragen, indem Struktur und Fluid über zwei eigenständige Programme abgebildet werden. Aufgaben, welche aus der Wechselwirkung resultieren (z.B. Kopplungsalgorithmen oder Datentransfer zwischen Oberflächennetzen), werden oftmals einem der Feldlöser zusätzlich zugeordnet. In dieser Arbeit wird die Wechselwirkung als eigenständiges Problem identifiziert und alle Aufgaben, welche damit in Verbindung stehen, in einem dritten eigenständigen Programm gelöst. Jedes Programm behandelt somit nur ein spezifisches Teilproblem. Dieser Ansatz führt zu einer hohen Flexibilität der Softwareumgebung. Durch klar definierte Schnittstellen ist eine einfache Austauschbarkeit einzelner Komponenten gegeben.

Der zweite Aspekt behandelt das Parallelisierungskonzept. Die hohe Anzahl an Unbekannten bei realistischen Problemstellungen erfordert eine parallele Simulation auf Hochleistungsrechnern. Die Teilfeldlöser verwenden in der Regel bereits eine eigene Parallelisierungsstrategie. Für die Verwendung innerhalb eines partitionierten Ansatzes ist es von besonderer Wichtigkeit, die parallele Skalierung der Teilfelder nicht durch Vorgabe ungünstiger Randbedingungen zu reduzieren. Das in dieser Arbeit entwickelte Parallelisierungskonzept führt zu einer nahezu optimalen parallelen Skalierung bei der partitionierten Simulation des gekoppelten Problems.

Nach der Entwicklung des Softwarekonzeptes werden im nächsten Schritt die algorithmischen Aspekte bei der impliziten Lösung des nichtlinearen gekoppelten Problems behandelt. Hierbei werden drei Bereiche identifiziert, die auf die Effektivität der gesamten Lösung besonders hohen Einfluss haben.

Der erste Bereich stellt den Lösungsansatz innerhalb der verwendeten Feldlöser dar. Bei der üblichen iterativen Vorgehensweise kann die Effizienz der Feldlöser durch Verwendung spezifischer Initialwerte für die Feldunbekannten innerhalb einer Kopplungsiteration deutlich verbessert werden.

Der zweite Bereich beinhaltet die Bestimmung einer Initialposition für das Interface in der ersten Kopplungsiteration. Hierzu wird ein Interfaceprädiktor verwendet. Je näher die Initialposition an der endgültigen Position liegt, desto geringer ist das Startresiduum und desto weniger Kopplungsiterationen werden benötigt.

Der dritte Bereich beinhaltet den Kopplungsalgorithmus als Kernbestandteil der partitionierten Lösung. Kopplungsalgorithmen können hierbei in Bezug auf Konvergenz- und Stabilitätseigenschaften und hinsichtlich ihrer Anforderungen an die Teilfeldlöser beurteilt werden. Ein effizienter Kopplungsalgorithmus sollte auch bei großen Zeitschrittweiten innerhalb weniger Iterationen konvergieren und die Verwendung von Standardfeldlöseransätzen ermöglichen.

Die Kernthese dieser Arbeit lässt sich also folgendermaßen formulieren:

Für die effiziente Simulation gekoppelter Probleme sind mehrere Teilbereiche von Wichtigkeit. Diese sind sowohl programmtechnischer als auch kopplungsmethodischer Art. Die Betrachtung der Effektivität nur eines einzelnen Bereiches stellt eine unnötige Einschränkung dar. Nur aus der Kombination möglichst effizienter Teilbereiche ergibt sich ein leistungsfähiges Gesamtsimulationskonzept.

1.3. Kapitelübersicht

Diese Arbeit besteht aus insgesamt sieben Kapiteln.

Das erste Kapitel bietet einen knappen Überblick des zu behandelnden Themas.

Das zweite Kapitel behandelt das Strukturfeld als Teilfeld des gekoppelten Problems. Nach einer Herleitung der kontinuumsmechanischen Grundgleichungen wird die Diskretisierung im Raum und in der Zeit besprochen. Dazu wird die Finite-Elemente-Methode und das Generalized- α Zeitintegrationsverfahren verwendet. Das dadurch entstehende nichtlineare diskrete Problems wird mittels des Newton-Verfahrens gelöst. Für die Verwendung innerhalb eines gekoppelten Problems wird der Lösungsansatz zusätzlich mit einer spezifischen Erweiterung versehen. Das Kapitel endet mit einem detaillierten Lösungsalgorithmus.

Das dritte Kapitel beinhaltet das Fluidfeld als Teilfeld des gekoppelten Problems. Deswegen Aufbau entspricht der Vorgehensweise des Strukturfeldes. Nach einer Herleitung der kontinuumsmechanischen Grundgleichungen wird die Diskretisierung in Raum und besprochen. Dazu wird die Finite-Volumen-Methode und das BDF-Verfahren verwendet. Das dadurch entstehende nichtlineare diskrete Problems wird mittels eines Druckkorrekturverfahrens gelöst. Auch der Fluidlösungsalgorithmus wird mit einer spezifischen Erweiterung für die Verwendung innerhalb eines gekoppelten Problems versehen. Das Kapitel endet mit einem detaillierten Lösungsalgorithmus.

Das vierte Kapitel beinhaltet die Fluid-Struktur-Wechselwirkung. Nach der Definition des gekoppelten Problems wird die Grundform eines impliziten Kopplungsverfahrens vorgestellt. Darauf aufbauend werden die Teilaspekte Konvergenzkriterien, Interfaceprädiktor und Kopplungsalgorithmus besprochen. Bei den Interfaceprädiktoren werden verschiedene Möglichkeiten mit ansteigender Komplexität vorgestellt, mit denen die Initialposition des Interfaces bestimmt werden kann. Bei den Kopplungsalgorithmen wird als erstes ein Verfahrensblick zu aktuell gängigen Methoden gegeben. Aus diesen wird eine Auswahl getroffen, welche im Detail vorgestellt wird.

Das fünfte Kapitel beinhaltet die Softwareumgebung mit dem verwendeten Drei-

Programme-Ansatz. Die verwendeten Teilfeldlöser für Struktur und Fluid werden kurz angesprochen. Der Schwerpunkt liegt in der Vorstellung des entwickelten Kopplungsprogrammes CoMA. Dabei wird auf die Parallelisierungsstrategie, die Datentransfermethoden und die Aufgabe der Simulationssteuerung eingegangen.

Das sechste Kapitel beinhaltet zwei numerische Beispiele. Das erste Beispiel dient zur Validierung der entwickelten Software und wird benutzt, um die in den vorherigen Kapiteln vorgestellten Kopplungsmethoden detailliert in Bezug auf Effizienz und Stabilität miteinander zu vergleichen. Das Ziel dieses Beispiels ist die Empfehlung einer Methodenkombination, die möglichst hohe Effizienz und Stabilität miteinander vereint. Das zweite Beispiel beinhaltet eine realistische Problemstellung. Zur Lösung wird die empfohlene Methodenkombination verwendet. Durch die hohe Anzahl an Unbekannten eignet sich dieses Beispiel, um die entwickelte Parallelisierungsstrategie auf ihre Effektivität zu überprüfen. Auch werden die physikalisch interessanten Simulationsergebnisse vorgestellt.

Das siebte und letzte Kapitel beinhaltet eine kurze Zusammenfassung dieser Arbeit und der darin erzielten Ergebnisse und endet mit einem Ausblick.

2. Strukturfeld

In diesem Kapitel wird das Strukturfeld als Teilfeld des gekoppelten Problems der Fluid-Struktur-Wechselwirkung behandelt. Da die in dieser Arbeit untersuchten Problemstellungen von transienter Natur sind und große Verformungen aufweisen, wird auf diese Aspekte im Folgenden besonders eingegangen. Das Ziel dieses Kapitels ist es, die algorithmische Behandlung des Strukturfeldes für die Verwendung innerhalb der Lösungsalgorithmen des gekoppelten Problems herzuleiten. Dazu wird basierend auf den kontinuumsmechanischen Grundgleichungen die Diskretisierung in Raum und Zeit gezeigt und die Lösung des dabei entstehenden diskreten und nichtlinearen Problems mittels eines iterativen Verfahrens. Für die Verwendung innerhalb einer impliziten Kopplung wird das iterative Verfahren erweitert mit einem als “angepasster Feldlöser” benannten Ansatz.

Das Kapitel ist wie folgt aufgebaut: In Abschnitt 2.1 werden die kontinuumsmechanischen Grundgleichungen hergeleitet. Darauf aufbauend wird in Abschnitt 2.2 die Diskretisierung im Raum und in den Abschnitten 2.3 und 2.3.1 die Diskretisierung in der Zeit behandelt. Das diskrete und nichtlineare Problem wird mittels der in den Abschnitten 2.4 und 2.5 vorgestellten Methoden gelöst, wobei bereits die Einbettung in einen impliziten Kopplungsalgorithmus vorgenommen wird. Das Kapitel endet mit einer Zusammenfassung dieser Vorgänge.

2.1. Herleitung der kontinuumsmechanischen Grundgleichungen

Bei der Herleitung der kontinuumsmechanischen Grundgleichungen des Festkörpers wird von großen Verformungen und kleinen Verzerrungen ausgegangen. Das Material wird als homogenes und isotropes Kontinuum betrachtet. Das zugehörige Strukturmodell ist die geometrisch nichtlineare Elastodynamik. Für die Beschreibung im Raum wird die Lagrange- bzw. materielle Betrachtungsweise gewählt.

Für die Herleitung existiert eine große Anzahl an Veröffentlichungen. Um nur einige wenige aufzuführen, sei hier auf [BLM99, Hol00, Bat01, Wri01] verwiesen. In den folgenden Abschnitten wird dieses Thema deshalb nur kurz angesprochen. Sie orientieren sich an den Herleitungen in [Kue09, Sch09].

2.1.1. Kinematik

Die Kinematik dient dazu, die Deformation eines Körpers zu beschreiben. Als Folge der materiellen Betrachtungsweise wird in Referenz- und Momentankonfiguration unterschieden. Die Referenzkonfiguration beschreibt die Initial- bzw. Ausgangsposition eines Materiepunktes im Raum. Diese wird über den Ortsvektor \mathbf{X} dargestellt. Die Momentankonfiguration ist die Position nach einer Bewegung und wird über den Ortsvektor \mathbf{x} dargestellt. Das Koordinatensystem ist dabei raumfest, orthonormiert und kartesisch. Die Verschiebung \mathbf{d} des Materiepunktes im Raum ist damit:

$$\mathbf{d} = \mathbf{x} - \mathbf{X}. \quad (2.1)$$

Ein materielles Linienelement habe die Länge $d\mathbf{X}$ in der Referenz- und $d\mathbf{x}$ in der Momentankonfiguration. Eine Beziehung zwischen beiden kann mittels des Deformationsgradienten \mathbf{F} hergestellt werden:

$$d\mathbf{x} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} \cdot d\mathbf{X} = \mathbf{F} \cdot d\mathbf{X} \quad (2.2)$$

und

$$d\mathbf{X} = \frac{\partial \mathbf{X}}{\partial \mathbf{x}} \cdot d\mathbf{x} = \mathbf{F}^{-1} \cdot d\mathbf{x}, \quad (2.3)$$

welcher eine lineare Abbildung zwischen den Konfigurationen beschreibt. Da der Deformationsgradient die Deformation vollständig beschreibt, enthält er auch Starrkörperanteile und ist damit als Maß für die Verzerrungen ungeeignet. Deshalb wird als Verzerrungsmaß der Green-Lagrange-Verzerrungstensor \mathbf{E} eingeführt:

$$\mathbf{E} = \frac{1}{2} (\mathbf{F}^T \mathbf{F} - \mathbf{I}). \quad (2.4)$$

Dieser stellt eine Beziehung zwischen den Verschiebungen und den Verzerrungen dar und da der Deformationsvorgang dadurch vollständig beschrieben wird, ist er auch für große Deformationen geeignet.

2.1.2. Impulsbilanz

Die Impulsbilanz stellt die dynamische Form des Gleichgewichts eines Körpers dar. In der undeformierten Referenzkonfiguration ist sie gegeben über:

$$\rho^S \frac{d^2 \mathbf{d}}{dt^2} = \nabla \cdot (\mathbf{P}) + \rho^S \mathbf{b}. \quad (2.5)$$

Hierbei ist ρ^S die Dichte der Struktur, $\frac{d^2}{dt^2}$ die zweite Zeitableitung (im Folgenden abgekürzt mit $\ddot{}$) und \mathbf{b} der Volumenkraftvektor pro Masseinheit. \mathbf{P} ist der Piola-Kirchhoff-Spannungstensor erster Ordnung (PK1), der die aktuelle Kraft auf ein Flächenelement der Referenzkonfiguration bezieht. Da dieser asymmetrisch ist, führt seine numerische Behandlung zu Nachteilen. Deshalb wird er durch den symmetrischen Piola-Kirchhoff-Spannungstensor zweiter Ordnung (PK2) ersetzt. Dieser ist das zum Green-Lagrange-Verzerrungstensor energetisch konjugierte Spannungsmaß und steht mit dem Piola-Kirchhoff-Spannungstensor erster Ordnung über den Deformationsgradienten in Beziehung:

$$\mathbf{S} = \mathbf{F}^{-1} \cdot \mathbf{P}, \quad \mathbf{P} = \mathbf{F} \cdot \mathbf{S}. \quad (2.6)$$

Damit lässt sich die Impulsbilanz angeben zu:

$$\rho^S \frac{d^2 \mathbf{d}}{dt^2} = \nabla \cdot (\mathbf{F} \cdot \mathbf{S}) + \rho^S \mathbf{b}. \quad (2.7)$$

Diese muss noch um Anfangs- und Randbedingungen ergänzt werden. Die Anfangsbedingungen sind hierbei Anfangsverschiebungen \mathbf{d}_0 und -geschwindigkeiten $\dot{\mathbf{d}}_0$:

$$\mathbf{d}(t=0) = \mathbf{d}_0 \quad \text{und} \quad \dot{\mathbf{d}}(t=0) = \dot{\mathbf{d}}_0. \quad (2.8)$$

Die Randbedingungen bestehen aus den Dirichlet-Randbedingungen, welche Verschiebungen $\bar{\mathbf{d}}$ vorschreiben:

$$\mathbf{d}(t) = \bar{\mathbf{d}} \quad \text{auf} \quad \Gamma^{S,D} \quad (2.9)$$

und Neumann-Randbedingungen, welche Randlasten \mathbf{t}^S in der Referenzkonfiguration vorschreiben:

$$\mathbf{N}\mathbf{S} = \mathbf{t}^S \quad \text{auf} \quad \Gamma^{S,N}, \quad (2.10)$$

mit \mathbf{N} als dem Normalenvektor am Rand. Die Ränder $\Gamma^{S,D}$ und $\Gamma^{S,N}$ bezeichnen die Teile des Randes, auf denen Dirichlet- bzw. Neumann-Bedingungen vorgeschrieben werden und überlappen sich nicht.

2.1.3. Konstitutive Gleichungen

Die konstitutiven Gleichungen stellen den Zusammenhang zwischen den kinematischen und den statischen Feldgrößen her, also zwischen Spannungen und Dehnungen. In der hier gewählten Betrachtungsweise, in der sich alle Größen auf die undeformierte Referenzkonfiguration beziehen, ist dies der Zusammenhang zwischen dem Piola-Kirchhoff-Spannungstensor zweiter Ordnung \mathbf{S} und dem Green-Lagrange-Verzerrungstensor \mathbf{E} .

Die echten physikalischen Spannungen sind die Cauchy-Spannungen σ , da sie in der Momentankonfiguration wirken. Der in der Referenzkonfiguration definierte PK2 lässt sich aus diesen mittels des Deformationsgradienten über eine Rücktransformation gewinnen:

$$\mathbf{S} = \det \mathbf{F} \mathbf{F}^{-1} \cdot \sigma \cdot \mathbf{F}^{-T}. \quad (2.11)$$

Als Materialmodell wird in dieser Arbeit das lineare Saint-Venant-Kirchhoff-Material verwendet, da es sich für die Beschreibung großer Verschiebungen und kleiner Dehnungen eignet. Die Beziehung zwischen dem Piola-Kirchhoff-Spannungstensor zweiter Ordnung \mathbf{S} und dem Green-Lagrange-Verzerrungstensor \mathbf{E} ist damit gegeben als:

$$\mathbf{S} = \lambda^S \operatorname{tr} \mathbf{E} \mathbf{I} + 2\mu^S \mathbf{E}. \quad (2.12)$$

Die Lamé-Konstanten λ^S und μ^S können aus dem Elastizitätsmodul E und der Querdehnzahl ν bestimmt werden über:

$$\lambda^S = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad \text{und} \quad \mu^S = \frac{E}{2(1+\nu)}. \quad (2.13)$$

2.1.4. Schwache Form

Die vollständige Beschreibung der starken Form des Anfangs-Randwertproblems der nichtlinearen Elastodynamik besteht aus den Gleichungen 2.4, 2.7 und 2.12, ergänzt durch Rand- und Initialbedingungen. Für eine weitergehende Diskretisierung mittels der Finite-Elemente-Methode ist eine Umformung von der starken in die schwache Form notwendig. Als Primärvariablen werden hierzu die Verschiebungen \mathbf{d} gewählt. Mittels der Methode der gewichteten Residuen kann die schwache Form aufgestellt werden. Dazu wird über das Strukturgebiet Ω^S integriert und die Einträge mit Testfunktionen $\delta \mathbf{d}$ und $\delta \mathbf{E}$ multipliziert. Die Impulsbilanz in der schwachen Form lautet damit:

$$\int_{\Omega^S} \rho^S \ddot{\mathbf{d}} \cdot \delta \mathbf{d} d\Omega^S + \int_{\Omega^S} \mathbf{S} : \delta \mathbf{E} d\Omega^S = \int_{\Omega^S} \rho_S \mathbf{b} \cdot \delta \mathbf{d} d\Omega^S + \int_{\Gamma^{S,N}} \mathbf{t}^S \cdot \delta \mathbf{d} d\Gamma^{S,N}. \quad (2.14)$$

Während in der starken Form das Gleichgewicht und die statischen Randbedingungen an jedem Punkt des Gebiets erfüllt sein müssen, ist dies in der schwachen Form nur noch im integralen Sinne der Fall. Die kinematischen und konstitutiven Beziehungen sind weiterhin in jedem Punkt erfüllt.

Werden die Testfunktionen $\delta \mathbf{d}$ als virtuelle Verschiebungen interpretiert, entspricht dieses Ergebnis dem Prinzip der virtuellen Verschiebungen. Dieses lautet:

$$\delta W = \delta W_{dyn} + \delta W_{int} + \delta W_{ext} = 0 \quad (2.15)$$

und besagt, dass sich die virtuelle Arbeit δW als Summe aus der virtuellen Arbeit der Trägheitskräfte δW_{dyn} , der inneren Arbeit δW_{int} und der äußeren Arbeit δW_{ext} zu Null ergibt. Die einzelnen Anteile korrelieren hierbei zu:

$$-\delta W_{dyn} = \int_{\Omega^S} \rho_S \ddot{\mathbf{d}} \cdot \delta \mathbf{d} d\Omega^S, \quad (2.16)$$

$$-\delta W_{int} = \int_{\Omega^S} \mathbf{S} : \delta \mathbf{E} d\Omega^S \quad \text{und} \quad (2.17)$$

$$\delta W_{ext} = \int_{\Omega^S} \rho_S \mathbf{b} \cdot \delta \mathbf{d} d\Omega^S + \int_{\Gamma^{S,N}} \mathbf{t} \cdot \delta \mathbf{d} d\Gamma^{S,N}. \quad (2.18)$$

2.2. Diskretisierung im Raum

Die Diskretisierung im Raum mittels der Finite-Elemente-Methode wird in einer großen Anzahl von Veröffentlichungen behandelt. Um nur einige wenige zu nennen, wird hier auf [BLM99, Bat01, Hug00, ZT05] verwiesen.

Das durch die schwache Form (Gl. 2.14) definierte Problem wird durch das unbekannte Verschiebungsfeld charakterisiert, welches eine räumliche und zeitliche Dimension besitzt:

$$\mathbf{d} = \mathbf{d}(\mathbf{x}, t). \quad (2.19)$$

Dieses Verschiebungsfeld ist kontinuierlich und weist unendlich viele Freiheitsgrade auf. Durch die Diskretisierung im Raum wird das Verschiebungsfeld $\mathbf{d}(\mathbf{x}, t)$ in einen Verschiebungsvektor $\mathbf{d}(t)$ überführt, welcher eine endliche Zahl an Freiheitsgraden besitzt. Als Diskretisierungsmethode wird die Finite-Elemente-Methode (FEM) gewählt.

Die Finite-Elemente-Methode unterteilt das zu untersuchende Strukturgebiet Ω^S in eine endliche Zahl n_{ele} an Elementen Ω^e :

$$\Omega^S \approx \Omega_h^S = \bigcup_{e=1}^{n_{ele}} \Omega^e. \quad (2.20)$$

Der Index e bezeichnet im Folgenden Größen, welche sich auf ein Element beziehen, der Index h bezeichnet diskretisierte Größen. Das Verschiebungsfeld pro Element wird approximiert, indem das Verschiebungsfeld an diskreten Knoten bestimmt wird (\mathbf{d}^e) und der Verlauf zwischen den Knoten mittels polynomischen Ansatzfunktionen \mathbf{N} interpoliert wird. Diese Diskretisierung im Raum kann pro Element angegeben werden als:

$$\mathbf{d}^e(\mathbf{x}, t) \approx \mathbf{d}_h^e = \mathbf{N}(\mathbf{x})\mathbf{d}^e(t). \quad (2.21)$$

Wird die virtuelle Verschiebung $\delta\mathbf{d}$ analog zu \mathbf{d} diskretisiert, spricht man von einer Bubnov-Galerkin-Diskretisierung. Durch Einsetzen von Gl. 2.21 in die schwache Form (Gl. 2.14) folgt:

$$\delta W = \bigcup_{e=1}^{n_{ele}} (\delta\mathbf{d}^e)^T \left[\mathbf{M}^e \ddot{\mathbf{d}}^e + \mathbf{f}^{e,int}(\mathbf{d}^e) - \mathbf{f}^{e,ext} \right] = 0, \quad (2.22)$$

wobei \bigcup die Assemblierung der Anteile der einzelnen Elemente zu einem globalen System unter der Berücksichtigung der kinematischen Verträglichkeiten und des Knotengleichgewichts symbolisiert. Die Integration über das Elementvolumen wird in der Regel mittels einer Gauss-Quadratur ausgeführt.

Durch Anwendung des Fundamentallemmas der Variationsrechnung wird Gl. 2.22 in die semidiskrete Form der allgemeinen nichtlinearen Bewegungsgleichung überführt:

$$\mathbf{M}\ddot{\mathbf{d}} + \mathbf{f}^{int}(\mathbf{d}) = \mathbf{f}^{ext}, \quad (2.23)$$

mit \mathbf{M} als der Massenmatrix, $\ddot{\mathbf{d}}$ als der an den diskreten Knoten definierten Beschleunigung, $\mathbf{f}^{int}(\mathbf{d})$ als dem nichtlinearen Vektor der inneren Kräfte und \mathbf{f}^{ext} als dem Vektor

der externen Lasten, welcher die diskretisierten Integrale über Rand- und Volumenlasten enthält und hier als verschiebungsunabhängig angenommen wird.

Anmerkung 1: Bei Verwendung von Polynomen niedriger Ansatzordnung kann das sogenannte „Locking“-Phänomen auftreten. Aus diesem folgen unphysikalische Versteifungseffekte, welche die Qualität der Strukturantwort negativ beeinflussen. In dieser Arbeit werden Methoden zur Reduzierung dieser Versteifungseffekte angewandt. Ein Überblick zu diesen Methoden findet sich z.B. in [BWBR04, Har07b].

Anmerkung 2: Bei den in dieser Arbeit untersuchten numerischen Beispielen werden vorwiegend Schalen- und Membranelemente benutzt. Das 7-Parameter-Schalenelement wird z.B. in [BR92, Bis99, Gee04] vorgestellt. Das Membranelement wird z.B. in [Lin09, Die09] näher erläutert.

Anmerkung 3: Bei der Herleitung der allgemeinen nichtlinearen Bewegungsgleichung (Gl. 2.25) wird die innere Reibung von sich deformierenden Festkörpern vernachlässigt. Bei der numerischen Behandlung ist es möglich, diesen Anteil über die zusätzliche Aufnahme eines Dämpfungsterms zu berücksichtigen. Dieser ist geschwindigkeitsproportional und besteht aus dem Vektor der Geschwindigkeiten $\dot{\mathbf{d}}$ und der Dämpfungsmatrix \mathbf{C} . Häufig wird für die Bestimmung der Dämpfungsmatrix die sogenannte Rayleigh-Dämpfung herangezogen ([CP93, LG95]). Dabei wird die Dämpfungsmatrix als Linearkombination aus Massen- und linearer Steifigkeitsmatrix gebildet:

$$\mathbf{C} = a_1 \cdot \mathbf{M} + a_2 \cdot \mathbf{K}. \quad (2.24)$$

Die Linearkoeffizienten a_1 und a_2 sind problemabhängig und experimentell zu bestimmen. Sie ergeben sich aus den an zwei Eigenschwingungen auftretenden Dämpfungsdekrementen.

Die allgemeine nichtlineare Bewegungsgleichung geht damit über in:

$$\mathbf{M}\ddot{\mathbf{d}} + \mathbf{C}\dot{\mathbf{d}} + \mathbf{f}^{int}(\mathbf{d}) = \mathbf{f}^{ext}. \quad (2.25)$$

2.3. Diskretisierung in der Zeit

Nach der Diskretisierung im Raum mittels der Finite-Elemente-Methode entsteht die allgemeine nichtlineare Bewegungsgleichung 2.25. Diese enthält als Unbekannte die Beschleunigungen $\ddot{\mathbf{d}}$, die Geschwindigkeiten $\dot{\mathbf{d}}$ und die Verschiebungen \mathbf{d} , welche eine kontinuierliche zeitliche Abhängigkeit besitzen. Um dieses Problem lösen zu können, wird eine Diskretisierung in der Zeit mittels eines Zeitintegrationsverfahrens durchgeführt.

Dazu wird der Untersuchungszeitraum $t = [t_0, t_{ges}]$ in diskrete Zeitintervalle $[t_n, t_{n+1}]$ bzw. Zeitschritte der Länge $\Delta t_n = t_{n+1} - t_n$ unterteilt, und es gilt: $t_{ges} = t_0 + \sum_{n=1}^m \Delta t_n$. Die Lösung des Problems ist nun nicht mehr kontinuierlich, sondern wird nur an diskreten Zeitpunkten berechnet. Zwischen diesen Zeitpunkten werden Annahmen über den zeitlichen Verlauf der Unbekannten getroffen, um von den am Anfang des Intervalls als bekannt vorausgesetzten Systemvektoren den Zustand am Ende des Intervalls zu bestimmen. Der prinzipielle Lösungsansatz und der Verlauf zwischen den Zeitpunkten wird durch die Wahl eines Zeitintegrationsalgorithmus bestimmt.

Zu den in der Strukturodynamik gebräuchlichsten Ansätzen zählen die expliziten und impliziten direkten Einschrittverfahren ([Kuh96]). Direkte Lösungsverfahren lösen die Bewegungsgleichung in ihrer ursprünglichen Form, während indirekte die Gleichung z.B. mittels Modalanalyse in eine andere Form transferieren. Einschrittverfahren verwenden zur Lösung bekannte Feldvariablen eines einzelnen Zeitpunktes, während Mehrschrittverfahren die Zustände mehrerer Zeitpunkte zur Bestimmung am Ende eines Intervalls benötigen.

Die Unterscheidung in explizite und implizite Verfahren ergibt sich aus dem Zeitpunkt, zu dem der Gleichgewichtszustand der Bewegungsgleichung betrachtet wird. Sollen die Feldvariablen zum Zeitpunkt t_{n+1} auf Basis einer Gleichgewichtsbetrachtung zum Zeitpunkt t_n bestimmt werden, spricht man von einem expliziten Verfahren. Implizite Verfahren hingegen basieren auf einer Gleichgewichtsbetrachtung zum Zeitpunkt $t_{n+\frac{1}{2}}$ oder t_{n+1} .

Explizite Verfahren haben den Vorteil, dass bei Vernachlässigung von Dämpfungstermen und Verwendung einer Hauptdiagonal-Massenmatrix kein Gleichungssystem in jedem Zeitschritt gelöst werden muss, sondern nur Vektor-Matrix Operationen ausgeführt werden müssen. Dadurch sind explizite Verfahren numerisch hoch effizient und robust. Als großer Nachteil expliziter Methoden ist aber deren bedingte Stabilität zu nennen. Dies bedeutet, dass die Lösung bei Wahl eines Zeitschrittes $\Delta t > \Delta t_{crit}$ divergiert und damit ein Zeitschrittlimit existiert. Der kritische Zeitschritt Δt_{crit} ist hierbei abhängig von der innerhalb des Modells auftretenden höchsten Frequenz und der kleinsten charakteristischen Elementlänge. Als Folge dieser Einschränkungen werden explizite Methoden in der Strukturodynamik vor allem bei sehr großen Modellen und kleinen Untersuchungszeiträumen genutzt, wie sie z.B. bei Crashuntersuchungen vorkommen.

Implizite Methoden hingegen zeigen unbedingte Stabilität für lineare Probleme bei entsprechender Wahl der Integrationsparameter. Es existiert also im Sinne der numerischen Stabilität keine Limitierung der Zeitschrittweite. Um die relevanten Frequenzen der Strukturantwort adäquat abzubilden, ist natürlich eine entsprechende Zeitschrittweite nötig. Diese ist aber in der Regel deutlich größer als bei expliziten Verfahren. Aller-

dings sind in jedem Zeitschritt mehrere Gleichungssysteme zu lösen. Da aber bei den im Rahmen dieser Arbeit untersuchten Problemen der Fluid-Struktur Wechselwirkung die Anzahl an Strukturfreiheitsgraden deutlich geringer als die Anzahl an Fluidfreiheitsgraden ist, ist dieser Aufwand nur von geringer Relevanz. Damit sind für die Untersuchung von langen Zeiträumen implizite Methoden vorzuziehen, da sie den großen Vorteil der freien Zeitschrittwahl besitzen.

Hier wird für die Diskretisierung des Strukturfeldes im Raum das Generalized- α Zeitintegrationsverfahren ([CH93, HHT77, Hug00, Woo90]) verwendet. Es zeichnet sich durch die folgenden Vorteile aus:

- Das Verfahren ist ein voll implizites und damit unbedingt stabiles Einschrittverfahren.
- Es besitzt eine Genauigkeit zweiter Ordnung.
- Die numerische Dissipation nieder- und hochfrequenter Moden lässt sich ohne Verlust der Genauigkeit kontrollieren.

In den folgenden Abschnitten werden die Grundformeln des Generalized- α Zeitintegrationsverfahrens vorgestellt und die damit mögliche Diskretisierung der allgemeinen nicht-linearen Bewegungsgleichung.

2.3.1. Zeitintegrationsverfahren Generalized- α

Als Basis für das Generalized- α Verfahren dienen die Ansätze von Newmark bzw. das zugehörige Newmark- β Verfahren ([New59, BLM00]). Hier werden die folgenden Ansätze für den Zusammenhang der Strukturvariablen zwischen Anfang und Ende eines Zeitschritts angenommen:

$$\mathbf{d}_{n+1} = \tilde{\mathbf{d}}_{n+1} + \beta \Delta t^2 \ddot{\mathbf{d}}_{n+1} \quad (2.26)$$

$$\text{mit } \tilde{\mathbf{d}}_{n+1} = \mathbf{d}_n + \Delta t \dot{\mathbf{d}}_n + \frac{\Delta t^2}{2} (1 - 2\beta) \ddot{\mathbf{d}}_n,$$

$$\dot{\mathbf{d}}_{n+1} = \dot{\tilde{\mathbf{d}}}_{n+1} + \gamma \Delta t \ddot{\mathbf{d}}_{n+1} \quad (2.27)$$

$$\text{mit } \dot{\tilde{\mathbf{d}}}_{n+1} = \dot{\mathbf{d}}_n + \Delta t (1 - \gamma) \ddot{\mathbf{d}}_n.$$

Diese basieren auf der Annahme eines linearen Beschleunigungsverlaufes innerhalb des Zeitschrittes. Somit ergibt sich durch Integration über den Zeitschritt ein quadratischer Verlauf der Geschwindigkeit und ein kubischer Verlauf der Verschiebungen. Die mit einer Tilde gekennzeichneten Vektoren bestehen aus bereits bekannten Anteilen vom Anfang des Zeitschritts. Eine anschauliche Erklärung dieser Ansätze und eine Erläuterung der Parameter β und γ ist z.B. in [AM86] gegeben.

Die allgemeine Bewegungsgleichung 2.25 enthält die drei unbekannt Systemvektoren der Beschleunigung, Geschwindigkeit und Verschiebung. Um dieses Gleichungssystem lösen zu können, werden obige Ansätze benutzt, damit dieses System auf einen einzelnen

unbekannten Systemvektor reduziert werden kann. Dafür werden die Verschiebungen gewählt.

Für $\beta > 0$ lässt sich Gleichung 2.26 nach $\ddot{\mathbf{d}}_{n+1}$ auflösen, um die Beschleunigung als Funktion der Verschiebung zu erhalten:

$$\ddot{\mathbf{d}}_{n+1} = \frac{1}{\beta \Delta t^2} (\mathbf{d}_{n+1} - \tilde{\mathbf{d}}_{n+1}). \quad (2.28)$$

Gleichung 2.28 lässt sich in Gleichung 2.27 einsetzen, um auch die Geschwindigkeit als Funktion der Verschiebung zu erhalten:

$$\dot{\mathbf{d}}_{n+1} = \dot{\tilde{\mathbf{d}}}_{n+1} + \gamma \Delta t \frac{1}{\beta \Delta t^2} (\mathbf{d}_{n+1} - \tilde{\mathbf{d}}_{n+1}). \quad (2.29)$$

Durch Verwendung von Gleichung 2.28 und Gleichung 2.29 ist es möglich, die Bewegungsgleichung zum Zeitpunkt t_{n+1} nur in Abhängigkeit der unbekanntes Verschiebungen \mathbf{d}_{n+1} zu formulieren.

Das Generalized- α Verfahren führt nun zusätzlich die Shift-Parameter α_m und α_f ein und führt die Bewegungsgleichung in eine modifizierte Form über:

$$\mathbf{M} \ddot{\mathbf{d}}_\alpha + \mathbf{C} \dot{\mathbf{d}}_\alpha + \mathbf{f}^{int}(\mathbf{d}_\alpha) = \mathbf{f}_\alpha^{ext}, \quad (2.30)$$

mit

$$\ddot{\mathbf{d}}_\alpha = (1 - \alpha_m) \ddot{\mathbf{d}}_{n+1} + \alpha_m \ddot{\mathbf{d}}_n, \quad (2.31)$$

$$\dot{\mathbf{d}}_\alpha = (1 - \alpha_f) \dot{\mathbf{d}}_{n+1} + \alpha_f \dot{\mathbf{d}}_n, \quad (2.32)$$

$$\mathbf{d}_\alpha = (1 - \alpha_f) \mathbf{d}_{n+1} + \alpha_f \mathbf{d}_n \quad \text{und} \quad (2.33)$$

$$\mathbf{f}_\alpha^{ext} = (1 - \alpha_f) \mathbf{f}_{n+1}^{ext} + \alpha_f \mathbf{f}_n^{ext}. \quad (2.34)$$

Der nichtlineare Vektor der inneren Kräfte $\mathbf{f}^{int}(\mathbf{d}_\alpha)$ an der Stelle \mathbf{d}_α innerhalb des Zeitintervalls lässt sich vereinfacht als Linearkombination der inneren Kräfte an den Intervallgrenzen ausdrücken ([Kuh96]):

$$\begin{aligned} \mathbf{f}^{int}(\mathbf{d}_\alpha) &= \mathbf{f}^{int}((1 - \alpha_f) \mathbf{d}_{n+1} + \alpha_f \mathbf{d}_n) \\ &\approx (1 - \alpha_f) \mathbf{f}^{int}(\mathbf{d}_{n+1}) + \alpha_f \mathbf{f}^{int}(\mathbf{d}_n). \end{aligned} \quad (2.35)$$

Dies wird als Linearisierung der inneren Kräfte bezeichnet. Damit ergibt sich die semidiskrete modifizierte Bewegungsgleichung:

$$\begin{aligned} &\mathbf{M} \cdot \left[(1 - \alpha_m) \cdot \ddot{\mathbf{d}}_{n+1} + \alpha_m \cdot \ddot{\mathbf{d}}_n \right] \\ &+ \mathbf{C} \cdot \left[(1 - \alpha_f) \cdot \dot{\mathbf{d}}_{n+1} + \alpha_f \cdot \dot{\mathbf{d}}_n \right] \\ &+ (1 - \alpha_f) \cdot \mathbf{f}_{n+1}^{int}(\mathbf{d}_{n+1}) + \alpha_f \cdot \mathbf{f}_n^{int}(\mathbf{d}_n) \\ &= (1 - \alpha_f) \cdot \mathbf{f}_{n+1}^{ext} + \alpha_f \cdot \mathbf{f}_n^{ext}. \end{aligned} \quad (2.36)$$

Bei Verwendung des Generalized- α Zeitintegrationsverfahrens wird die Bewegungsgleichung also zu einem Zeitpunkt innerhalb der Zeitintervallgrenzen $[t_n, t_{n+1}]$ gelöst. Dieser wird über eine Linearkombination der Grenzzustände beschrieben. Dabei werden die Anteile aus der Beschleunigung unabhängig von den restlichen Anteilen behandelt.

2.3.2. Effektive Strukturgleichung

Unter Verwendung der Ansätze von Newmark (Gleichungen 2.28 und 2.28) kann nun die semidiskrete Bewegungsgleichung (2.36) in ein System überführt werden, welches nur den Systemvektor der Verschiebungen \mathbf{d}_{n+1} als einzige Unbekannte enthält:

$$\begin{aligned}
 & \mathbf{M} \cdot \left[(1 - \alpha_m) \cdot \frac{1}{\beta \Delta t^2} \left(\mathbf{d}_{n+1} - \tilde{\mathbf{d}}_{n+1} \right) + \alpha_m \cdot \ddot{\mathbf{d}}_n \right] \\
 & + \mathbf{C} \cdot \left[(1 - \alpha_f) \cdot \left(\dot{\tilde{\mathbf{d}}}_{n+1} + \gamma \Delta t \frac{1}{\beta \Delta t^2} \left(\mathbf{d}_{n+1} - \tilde{\mathbf{d}}_{n+1} \right) \right) + \alpha_f \cdot \dot{\mathbf{d}}_n \right] \\
 & + (1 - \alpha_f) \cdot \mathbf{f}_{n+1}^{int}(\mathbf{d}_{n+1}) + \alpha_f \cdot \mathbf{f}_n^{int}(\mathbf{d}_n) \\
 & = (1 - \alpha_f) \cdot \mathbf{f}_{n+1}^{ext} + \alpha_f \cdot \mathbf{f}_n^{ext}. \tag{2.37}
 \end{aligned}$$

Durch Sortieren aller unbekanntenen Terme auf die linke Seite und aller bekannten Terme auf die rechte Seite ergibt sich die nun vollständig diskretisierte effektive Strukturgleichung zu:

$$\begin{aligned}
 & \mathbf{M} \cdot (1 - \alpha_m) \cdot \frac{1}{\beta \Delta t^2} \mathbf{d}_{n+1} + \mathbf{C} \cdot (1 - \alpha_f) \gamma \Delta t \frac{1}{\beta \Delta t^2} \mathbf{d}_{n+1} + (1 - \alpha_f) \cdot \mathbf{f}_{n+1}^{int}(\mathbf{d}_{n+1}) \\
 & = \mathbf{M} \cdot \left[(1 - \alpha_m) \cdot \frac{1}{\beta \Delta t^2} \tilde{\mathbf{d}}_{n+1} - \alpha_m \cdot \ddot{\mathbf{d}}_n \right] \\
 & + \mathbf{C} \cdot \left[(1 - \alpha_f) \cdot \left(-\dot{\tilde{\mathbf{d}}}_{n+1} + \gamma \Delta t \frac{1}{\beta \Delta t^2} \tilde{\mathbf{d}}_{n+1} \right) - \alpha_f \cdot \dot{\mathbf{d}}_n \right] \\
 & - \alpha_f \cdot \mathbf{f}_n^{int}(\mathbf{d}_n) + (1 - \alpha_f) \cdot \mathbf{f}_{n+1}^{ext} + \alpha_f \cdot \mathbf{f}_n^{ext}, \tag{2.38}
 \end{aligned}$$

welche in Folge des Anteils der inneren Kräfte $\mathbf{f}_{n+1}^{int}(\mathbf{d}_{n+1})$ ein nichtlineares Gleichungssystem darstellt.

2.3.3. Parameterwahl

Die unbekanntenen Parameter des Generalized- α Verfahrens in der effektiven Strukturgleichung sind β , γ , α_m und α_f . Um das gewünschte Verhalten des Verfahrens zu gewährleisten, ergeben sich entsprechende Anforderungen an diese Parameter, die in Anlehnung an [Kuh96] in Tabelle 2.1 aufgeführt sind. Ein zusätzlicher Parameter, welcher zur Beurteilung von Zeitintegrationsverfahren benutzt wird, ist der Spektralradius ρ_∞ mit $\rho_\infty \in [0, 1]$, welcher sich aus dem größten Eigenwert der Vergrößerungsmatrix \mathbf{A} ableitet. Der Spektralradius gibt an, in wie weit höherfrequente Moden der Strukturantwort

gedämpft werden. Für $\rho_\infty = 1,0$ findet keine numerische Dämpfung statt, für $\rho_\infty < 1$ eine zunehmende numerische Dämpfung. Durch die entsprechenden Beziehungen zwischen den Parametern ist es möglich, alle Parameter in Abhängigkeit des Spektralradius zu setzen. In Tabelle 2.2 sind für Werte des Spektralradius von $\rho_\infty = 1,0, 0,9$ und $0,8$ die Parameter exemplarisch angegeben.

Bedingung	
unbedingte Stabilität (bei linearen Problemen)	$\alpha_m \leq \alpha_f \leq \frac{1}{2}$ $\beta \geq \frac{1}{4} + \frac{1}{2}(\alpha_f - \alpha_m)$
Genauigkeit 2. Ordnung	$\gamma = \frac{1}{2} - \alpha_m + \alpha_f$
benutzerdefinierter Spektralradius ρ_∞	$\alpha_m = \frac{2\rho_\infty - 1}{\rho_\infty + 1}$
minimale Dissipation hochfrequenter Moden	$\alpha_f = \frac{\rho_\infty}{\rho_\infty + 1}$
maximale Dissipation hochfrequenter Moden	$\beta = \frac{1}{4}(1 - \alpha_m + \alpha_f)^2$

Tabelle 2.1.: Parameteranforderungen des Generalized- α Verfahrens

Spektralradius ρ_∞	β	γ	α_m	α_f
1,00	0,250	0,500	0,500	0,500
0,90	0,277	0,553	0,421	0,474
0,80	0,309	0,611	0,333	0,444

Tabelle 2.2.: Parameterkombinationen für verschiedene Spektralradien

2.4. Lösung der effektiven Strukturgleichung

Zur Lösung des nichtlinearen Gleichungssystems der effektiven Strukturgleichung wird ein Newtonverfahren verwendet. Das Newtonverfahren, auch Newton-Raphson Methode genannt, ist ein Näherungsverfahren zur Lösung nichtlinearer Gleichungen $f(x) = 0$. Dazu wird an einem Startpunkt x_0 die Tangente des Problems $\frac{df}{dx}(x_0)$ bestimmt (Linearisierung) und die Nullstelle der Tangente mittels $f(x_0) + \frac{df}{dx}(x_0)\Delta x$ berechnet. Dies liefert eine bessere Näherung der Nullstelle und dient als Ausgangspunkt der nächsten Iteration. Diese Abfolge wird wiederholt (Iterationen), bis die Nullstelle mit ausreichender Genauigkeit gefunden ist. Die Konvergenzordnung ist maximal quadratisch. Wichtig für das Konvergenzverhalten ist ein Startwert x_0 , der möglichst nahe an der Lösung des Problems liegt. Ist der Startwert zu weit von der Lösung entfernt, kann das Verfahren divergieren.

Übertragen auf die Problemstellung der Strukturdynamik ergibt sich als nichtlineare Gleichung die nach Null aufgelöste effektive Strukturgleichung (2.38), welche damit auch in die Residuumsform überführt wird. Die zu bestimmende Unbekannte ist der Systemvektor der Verschiebungen zum Ende des Zeitschritts \mathbf{d}_{n+1} . Damit ergibt sich pro Newtoniteration j die Rechenvorschrift zu:

$$\frac{\partial \mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j)}{\partial \mathbf{d}_{n+1}} \Delta \mathbf{d}_{n+1}^j = -\mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j), \quad (2.39)$$

welche nach dem neuen Verschiebungsincrement gelöst wird. Die zugehörige Updateregeln lautet:

$$\mathbf{d}_{n+1}^{j+1} = \mathbf{d}_{n+1}^j + \Delta \mathbf{d}_{n+1}^j. \quad (2.40)$$

Die Newtoniterationen werden abgebrochen, sobald ein Verschiebungsvektor \mathbf{d}_{n+1}^{j+1} gefunden ist, für den der zugehörige Residuumsvektor $\mathbf{R}_{n+1}^{S,j+1}(\mathbf{d}_{n+1}^{j+1})$ eine entsprechende Konvergenzbedingung erfüllt. Als Konvergenzbedingung wird i.d.R. das Unterschreiten einer gewissen Schranke gesetzt. Diese ergibt sich z.B. über die Anwendung der L2-Norm auf den Residuenvektor, wobei diese skalare Größe mit einem entsprechenden Toleranzwert für das Strukturproblem ϵ^S verglichen werden kann:

$$\left| \mathbf{R}_{n+1}^{S,j+1}(\mathbf{d}_{n+1}^{j+1}) \right|_2 \leq \epsilon^S. \quad (2.41)$$

Dies stellt ein absolutes Kriterium dar. Es sind auch relative Kriterien üblich, in denen vergleichende Aussagen verwendet werden.

Zur Umsetzung des Newtonverfahrens benötigt man also in Iteration j das Residuum der Strukturproblems in Abhängigkeit der Verschiebungen \mathbf{d}_{n+1}^j :

$$\begin{aligned}
 \mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j) = & \mathbf{M} \cdot \left[(1 - \alpha_m) \cdot \frac{1}{\beta \Delta t^2} \left(\mathbf{d}_{n+1}^j - \tilde{\mathbf{d}}_{n+1} \right) + \alpha_m \cdot \ddot{\mathbf{d}}_n \right] \\
 & + \mathbf{C} \cdot \left[(1 - \alpha_f) \cdot \left(\dot{\tilde{\mathbf{d}}}_{n+1} + \gamma \Delta t \frac{1}{\beta \Delta t^2} \left(\mathbf{d}_{n+1}^j - \tilde{\mathbf{d}}_{n+1} \right) \right) + \alpha_f \cdot \dot{\mathbf{d}}_n \right] \\
 & + (1 - \alpha_f) \cdot \mathbf{f}_{n+1}^{int,j} \left(\mathbf{d}_{n+1}^j \right) + \alpha_f \cdot \mathbf{f}_n^{int} \left(\mathbf{d}_n \right) \\
 & - (1 - \alpha_f) \cdot \mathbf{f}_{n+1}^{ext} - \alpha_f \cdot \mathbf{f}_n^{ext}
 \end{aligned} \tag{2.42}$$

und die Matrix der Ableitungen des Residuums nach den Verschiebungen:

$$\begin{aligned}
 \frac{\partial \mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j)}{\partial \mathbf{d}_{n+1}} = & \mathbf{M} \cdot (1 - \alpha_m) \cdot \frac{1}{\beta \Delta t^2} \\
 & + \mathbf{C} \cdot (1 - \alpha_f) \gamma \Delta t \frac{1}{\beta \Delta t^2} \\
 & + (1 - \alpha_f) \cdot \frac{\partial \mathbf{f}_{n+1}^{int} \left(\mathbf{d}_{n+1}^j \right)}{\partial \mathbf{d}_{n+1}}.
 \end{aligned}$$

2.5. Strukturlösungsalgorithmus für implizite Kopplungsalgorithmen

Innerhalb dieses Abschnitts wird der Strukturlösungsalgorithmus vorgestellt, wie er innerhalb eines impliziten Kopplungsverfahrens verwendet wird. Die impliziten Kopplungsalgorithmen werden zwar erst im Kapitel 4 vorgestellt, die Details für das Verständnis des Strukturlösungsalgorithmus werden deshalb hier kurz wiedergegeben.

Das Strukturfeld fungiert innerhalb der in dieser Arbeit untersuchten Problem des Typs Fluid-Struktur-Wechselwirkung als Neumann-Partition. Damit werden dem Strukturfeld als Randbedingung externe Lasten vorgegeben. Diese Lasten setzen sich hierbei zusammen aus Lasten $\mathbf{f}^{ext,S}$ aus dem Strukturmodell selbst (z.B. Eigengewicht) und Lasten $\mathbf{f}^{ext,F}$, welche aus dem umgebenden Fluid auf die Struktur einwirken. Das Ziel ist es, das zu den vorgegebenen Lasten zugehörige Verschiebungsfeld zu bestimmen.

Der gesamte Lösungsalgorithmus besteht hierbei aus drei ineinander geschachtelten Schleifen bzw. Iterationen. Allem übergeordnet ist die Zeitschrittsschleife (Iterationsindex n). Pro Zeitschritt ergibt sich die Schleife über die Kopplungsiterationen (Iterationsindex k). In jeder Kopplungsiteration wird durch den Kopplungsalgorithmus ein neuer Fluidlastvektor $\mathbf{f}^{ext,F}$ vorgegeben. Das Strukturproblem wird mittels der Finite-Elemente-Methode und des vorgestellten Generalized- α Zeitintegrationsalgorithmus diskretisiert, um die zugehörigen Strukturverschiebungen zu berechnen. Zur Lösung dieses nichtlinearen Problems wird das Newtonverfahren verwendet und damit ergibt sich als innerste Schleife die Schleife über die einzelnen Newtoniterationen (Iterationsindex j).

Ein detaillierter Algorithmus zur Lösung des nichtlinearen dynamischen Strukturproblems mittels des Generalized- α Zeitintegrationsalgorithmus und der Newton-Methode innerhalb einer impliziten Kopplung ist in Algorithmus 1 gegeben.

Anmerkung 1: Als Startwert der unbekanntenen Verschiebungen des Newtonverfahrens wird in der ersten Kopplungsiteration pro Zeitschritt der durch das Zeitintegrationsverfahren bestimmte Vektor $\tilde{\mathbf{d}}_{n+1}$ benutzt. Unter der Annahme einer stetigen Konvergenz des gekoppelten Problems folgt, dass die aufzubringenden Fluidlasten ebenfalls konvergieren. Damit ist es sinnvoll, ab der zweiten Kopplungsiteration die Verschiebungen der vorherigen Iteration \mathbf{d}_{n+1}^k jeweils als Startwert zu benutzen, da diese mit zunehmender Konvergenz näher am Ergebnis liegen als $\tilde{\mathbf{d}}_{n+1}$. Dies stellt sicher, dass die notwendige Anzahl an Newtoniterationen pro Kopplungsiteration möglichst gering ist und die Lösung des Strukturfeldes effizient erlangt werden kann. Dieser Ansatz wird im weiteren Verlauf als „angepasster Feldlöser“ bezeichnet. Er stellt die Erweiterung eines Standardansatzes zur Einzelfeldberechnung für eine gekoppelte Simulation dar. Eine Untersuchung des Einflusses auf die Effizienz des Lösungsalgorithmus wird anhand eines numerischen Beispiels in Abschnitt 6.1.3 durchgeführt.

Anmerkung 2: Der Vollständigkeit wegen wird in Algorithmus 2 auch der Lösungsansatz für eine reine Strukturfeldberechnung ohne Wechselwirkung angegeben. Damit kann auch die Anpassung eines Standardfeldlöseransatzes für eine implizite Kopplung besser

nachvollzogen werden.

Algorithmus 1 Strukturlösungsalgorithmus für implizite Kopplungsalgorithmen

```

1:  $n = 1, \quad t_n = t_0$ 
2: initialisiere  $\mathbf{d}_n^0, \dot{\mathbf{d}}_n^0, \ddot{\mathbf{d}}_n^0, \mathbf{f}_n^{ext,0}$  und  $\mathbf{f}_n^{int,0}(\mathbf{d}_n^0)$ 
3: Wenn  $\mathbf{f}_n^{ext,0} \neq 0$ : Berechne statische Initallösung für  $\mathbf{d}_n^0$ 
4: Iterationen über Zeitschritte:
5: while ( $t_n \leq t_{ges}$ ) do
6:   Berechne Tilde-Vektoren  $\tilde{\mathbf{d}}_{n+1}, \dot{\tilde{\mathbf{d}}}_{n+1}$ 
7:   Verschiebungsprädiktor:  $\mathbf{d}_{n+1}^{k=1,j=1} = \tilde{\mathbf{d}}_{n+1}$ 
8:    $k = 1$ 
9:   Iterationen über Kopplungsschritte:
10:  while (keine Kopplungskonvergenz) do
11:    Erstelle  $\mathbf{f}_{n+1}^{ext,k+1} = \mathbf{f}_{n+1}^S + \mathbf{f}_{n+1}^{F,k+1}$ 
12:     $j = 1$ 
13:    Iterationen über Newtonschritte:
14:    while (keine Strukturkonvergenz) do
15:      Berechne aktuelles Residuum  $\mathbf{R}_{n+1}^{S,k+1,j}(\mathbf{d}_{n+1}^{k+1,j})$ 
16:      if ( $\left| \mathbf{R}_{n+1}^{S,k+1,j}(\mathbf{d}_{n+1}^{k+1,j}) \right|_L \geq \epsilon^S$ ) then
17:        Berechne Ableitung des Residuums  $\frac{\partial \mathbf{R}_{n+1}^{S,k+1,j}(\mathbf{d}_{n+1}^{k+1,j})}{\partial \mathbf{d}_{n+1}}$ 
18:        Löse  $\frac{\partial \mathbf{R}_{n+1}^{S,k+1,j}(\mathbf{d}_{n+1}^{k+1,j})}{\partial \mathbf{d}_{n+1}} \Delta \mathbf{d}_{n+1}^{k+1,j} = -\mathbf{R}_{n+1}^{S,k+1,j}(\mathbf{d}_{n+1}^{k+1,j})$ 
19:        Verschiebungsupdate  $\mathbf{d}_{n+1}^{k+1,j+1} = \mathbf{d}_{n+1}^{k+1,j} + \Delta \mathbf{d}_{n+1}^{k+1,j}$ 
20:         $j \leftarrow j + 1$ 
21:      else
22:        Abbruch Strukturiterationen, gehe zu nächster Kopplungsiteration
23:         $\mathbf{d}_{n+1}^{k+1} = \mathbf{d}_{n+1}^{k+1,j}$ 
24:      end if
25:    end while
26:    Überprüfe Kopplungskonvergenz
27:    if (Kopplung konvergiert) then
28:      Abbruch Kopplungsiterationen, gehe zu nächstem Zeitschritt
29:       $\mathbf{d}_{n+1} = \mathbf{d}_{n+1}^k$ 
30:    else
31:      Gehe zu nächster Kopplungsiteration
32:       $k \leftarrow k + 1$ 
33:    end if
34:  end while
35:   $t_{n+1} = t_n + \Delta t_n$ 
36:   $n \leftarrow n + 1$ 
37: end while

```

Algorithmus 2 Generalized- α Zeitintegrationsalgorithmus in der nichtlinearen Strukturdynamik - Einzelfeldberechnung

```

1:  $t_n = t_0$ 
2: initialisiere  $\mathbf{d}_n^0, \dot{\mathbf{d}}_n^0, \ddot{\mathbf{d}}_n^0, \mathbf{f}_n^{ext,0}$  und  $\mathbf{f}_n^{int,0}(\mathbf{d}_n^0)$ 
3: Wenn  $\mathbf{f}_n^{ext,0} \neq 0$ : Berechne statische Initallösung für  $\mathbf{d}_n^0$ 
4:  $n = 1$ 
5: Iterationen über Zeitschritte:
6: while ( $t_n < t_{ges}$ ) do
7:   Erstelle  $\mathbf{f}_{n+1}^{ext,j}$ 
8:   Berechne Tilde-Vektoren  $\tilde{\mathbf{d}}_{n+1}, \dot{\tilde{\mathbf{d}}}_{n+1}$ 
9:   Verschiebungsprädiktor:  $\mathbf{d}_{n+1}^1 = \tilde{\mathbf{d}}_{n+1}$ 
10:   $j = 1$ 
11:  Iterationen über Newtonschritte:
12:  while (keine Konvergenz) do
13:    Berechne aktuelles Residuum  $\mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j)$ 
14:    if  $\left| \mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j) \right|_2 < \epsilon^S$  then
15:      konvergiert
16:    else
17:      Berechne Ableitung des Residuums  $\frac{\partial \mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j)}{\partial \mathbf{d}_{n+1}}$ 
18:      Löse  $\frac{\partial \mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j)}{\partial \mathbf{d}_{n+1}} \Delta \mathbf{d}_{n+1}^j = -\mathbf{R}_{n+1}^{S,j}(\mathbf{d}_{n+1}^j)$ 
19:      Verschiebungsupdate  $\mathbf{d}_{n+1}^{j+1} = \mathbf{d}_{n+1}^j + \Delta \mathbf{d}_{n+1}^j$ 
20:       $k \leftarrow j + 1$ 
21:    end if
22:  end while
23:   $t_{n+1} = t_n + \Delta t_n$ 
24:   $n \leftarrow n + 1$ 
25: end while

```

2.6. Zusammenfassung

In diesem Kapitel wird das Strukturfeld als Teilfeld des gekoppelten Problems der Fluid-Struktur-Wechselwirkung behandelt. Das zugehörige Strukturmodell ist die geometrisch nichtlineare Elastodynamik. Nach einer Beschreibung der kontinuumsmechanischen Grundgleichungen und der schwachen Form wird das Problem im Raum mittels der Finite-Elemente-Methode und in der Zeit mittels des Generalized- α Zeitintegrationsverfahrens diskretisiert. Das dadurch entstehende nichtlineare Gleichungssystem wird iterativ unter Verwendung der Newton-Methode gelöst. Das Kapitel endet mit der detaillierten Vorstellung des Strukturfeldlösungsalgorithmus innerhalb einer impliziten Kopplung. Dabei wird der Term „angepasster Feldlöser“ eingeführt, welcher eine spezifische Anpassung des Löseralgorithmus an die implizite Kopplung bedeutet. Ein angepasster Feldlöser macht sich die Konvergenzeigenschaften des übergeordneten gekoppelten Problems zu Nutze, um die benötigte Anzahl an Newton-Iterationen in der Strukturfeldlösung pro Kopplungsiteration zu verringern.

3. Fluidfeld

In diesem Kapitel wird das Fluidfeld als Teilfeld des gekoppelten Problems der Fluid-Struktur-Wechselwirkung behandelt. Das Ziel dieses Kapitels ist es, die algorithmische Behandlung des Fluidfeldes für die Verwendung innerhalb der Lösungsalgorithmen des gekoppelten Problems herzuleiten. Das Fluid wird hierbei als inkompressibel und viskos angenommen, und, da in dieser Arbeit zeitabhängiges Verhalten untersucht wird, als instationär. Für die Herleitung der Grundgleichungen wird die Eulersche Betrachtungsweise gewählt. Da aber das Fluidfeld von einem elastischen Körper begrenzt wird, verändert sich die Feldgeometrie. Diesem Aspekt wird durch die spätere Erweiterung des Ansatzes auf eine ALE-Betrachtungsweise Rechnung getragen. Der numerische Simulationsansatz besteht aus einer Diskretisierung mittels der Finite-Volumen-Methode und einer Lösung des dadurch entstehenden nichtlinearen Problems mittels des PISO-Druckkorrekturverfahrens. Für die Verwendung innerhalb einer impliziten Kopplung wird das PISO-Verfahren erweitert mit einem als „angepasster Feldlöser“ benannten Ansatz.

Das Kapitel ist wie folgt aufgebaut: In Abschnitt 3.1 werden die kontinuumsmechanischen Grundgleichungen hergeleitet. Darauf aufbauend wird in Abschnitt 3.2 die Diskretisierung im Raum und in Abschnitt 3.3 die Diskretisierung in der Zeit behandelt. Das diskrete und nichtlineare Problem wird mittels der in den Abschnitten 3.4, 3.5 und 3.6 vorgestellten Methoden gelöst, wobei bereits die Einbettung in einen impliziten Kopplungsalgorithmus vorgenommen wird. Das Kapitel endet mit einer Zusammenfassung dieser Vorgänge.

3.1. Herleitung der kontinuumsmechanischen Grundgleichungen

Bei der Herleitung der kontinuumsmechanischen Grundgleichungen des Fluidfeldes wird ein instationäres Newtonsches Fluid mit konstanter Dichte angenommen. Das zugehörige Fluidmodell wird über die inkompressiblen Navier-Stokes-Gleichungen beschrieben. Für die Beschreibung im Raum wird die Eulersche Betrachtungsweise gewählt.

Es existiert eine große Anzahl an Veröffentlichungen zu diesem Thema. Um nur einige wenige aufzuführen, sei hier auf [FP99, FP08, GDN95, Sch99, Bre01, Fro06, Wil06] verwiesen. Die Herleitung wird deshalb nur kurz angesprochen.

Die Navier-Stokes-Gleichungen basieren auf zwei Prinzipien von fundamentaler Bedeutung: Die Erhaltung der Masse und die Erhaltung des Impulses eines Körpers. Diese werden im Folgenden vorgestellt. Die Betrachtung erfolgt dabei anhand eines Kontrollvolumens (KV), also mittels eines vorgegebenen Volumens im Raum. V bezeichnet das Volumen und S die Oberfläche eines Kontrollvolumens.

3.1.1. Massenerhaltung

Die Integralform der Erhaltungsgleichung für die Masse (auch als Kontinuitätsgleichung bekannt) lautet:

$$\frac{\partial}{\partial t} \int_V \rho^F dV + \int_S \rho^F \mathbf{u} \cdot \mathbf{n} dS = 0. \quad (3.1)$$

Hierin bezeichnet ρ^F die Fluidichte, \mathbf{u} die Geschwindigkeit und \mathbf{n} den Normalenvektor auf der Oberfläche des Kontrollvolumens. Die Erhaltungsgleichung für die Masse besagt, dass die zeitliche Änderungsrate der Masse des Kontrollvolumens gleich der Masse sein muss, die über die Oberfläche zu- und abgeführt wird.

Durch Anwendung des Gauss-Theorems auf den konvektiven Term wird das Oberflächenintegral in ein Volumenintegral umgewandelt und es folgt:

$$\frac{\partial}{\partial t} \int_V \rho^F dV + \int_V \nabla \cdot (\rho^F \mathbf{u}) dV = 0. \quad (3.2)$$

Da Gl. (3.2) auch für infinitesimal kleine Kontrollvolumen erfüllt sein muss, ergibt sich daraus die Differentialform der Kontinuitätsgleichung:

$$\frac{\partial \rho^F}{\partial t} + \nabla \cdot (\rho^F \mathbf{u}) = 0. \quad (3.3)$$

3.1.2. Impulserhaltung

Die Integralform der Impulserhaltung ist gegeben über:

$$\frac{\partial}{\partial t} \int_V \rho^F \mathbf{u} dV + \int_S \rho^F \mathbf{u} \mathbf{u} \cdot \mathbf{n} dS = \int_S \boldsymbol{\sigma}^F \cdot \mathbf{n} dS + \int_V \rho^F \mathbf{b} dV. \quad (3.4)$$

Die rechte Seite von Gl. (3.4) gibt den Einfluss von Kräften wieder, die auf ein Kontrollvolumen einwirken. Dies sind Oberflächenkräfte, wie z.B. aus Druck oder Scherspannungen, und Volumenkräfte, wie z.B. aus Gravitation. Volumenkräfte werden mittels \mathbf{b} gekennzeichnet. Mittels des Gauss-Theorems, angewandt auf die Terme mit Oberflächenintegral, werden diese transformiert in:

$$\frac{\partial}{\partial t} \int_V \rho^F \mathbf{u} dV + \int_V \nabla \cdot (\rho^F \mathbf{u} \mathbf{u}) dV = \int_V \nabla \cdot \sigma^F dV + \int_V \rho^F \mathbf{b} dV. \quad (3.5)$$

Und da Gl. (3.5) auch für infinitesimal kleine Kontrollvolumen erfüllt sein muss, ergibt sich daraus die Differentialform der Impulserhaltungsgleichung:

$$\frac{\partial \rho^F \mathbf{u}}{\partial t} + \nabla \cdot (\rho^F \mathbf{u} \mathbf{u}) = \nabla \cdot \sigma^F + \rho^F \mathbf{b}. \quad (3.6)$$

3.1.3. Konstitutive Gleichungen

Da in die Erhaltungsgleichung für den Impuls der Einfluss äußerer Kräfte auf das Kontrollvolumen eingeht, ist es notwendig, den Spannungszustand im Fluid abbilden zu können. Über die Annahme eines Newtonschen Fluides ist die Schubspannung proportional zum Geschwindigkeitsgradienten. Damit wird der Spannungstensor σ^F folgendermaßen beschrieben:

$$\sigma^F = - \left(p + \frac{2}{3} \mu \nabla \cdot \mathbf{u} \right) \mathbf{I} + 2\mu \mathbf{D}, \quad (3.7)$$

mit p als dem Druck, μ als der dynamischen Viskosität, \mathbf{I} als Einheitstensor und \mathbf{D} als dem Tensor der Deformationsgeschwindigkeit. Dieser ist gegeben über:

$$\mathbf{D} = \frac{1}{2} \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right]. \quad (3.8)$$

Damit ist der Zusammenhang zwischen den Feldvariablen Geschwindigkeit und Druck und der Spannung am Kontrollvolumen hergestellt.

3.1.4. Inkompressible Navier-Stokes-Gleichungen

Bei den in dieser Arbeit untersuchten Strömungsproblemen kann das Fluid als volumenbeständig betrachtet werden, da für die auftretenden Mach-Zahlen gilt:

$$Ma = \frac{u_{max}}{c} < 0,3, \quad (3.9)$$

mit c als der Schallgeschwindigkeit im betrachteten Medium und u_{max} als der maximal auftretenden Geschwindigkeit. Damit ist die Fluidichte konstant und das Fluid verhält sich inkompressibel. Es gilt: $\rho = const.$ und $\frac{\partial \rho}{\partial t} = 0$.

Damit geht die Massenerhaltungsgleichung (3.3) über in:

$$\nabla \cdot \mathbf{u} = 0, \quad (3.10)$$

die Impulserhaltungsgleichung (3.6) in:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) = \frac{1}{\rho^F} \nabla \cdot \sigma^F + \mathbf{b}, \quad (3.11)$$

und die konstitutive Gleichung (3.7) vereinfacht sich zu:

$$\sigma^F = -p\mathbf{I} + 2\mu\mathbf{D}. \quad (3.12)$$

Damit folgt:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) = \nabla \cdot (\nu \nabla \mathbf{u}) - \frac{1}{\rho^F} \nabla p + \mathbf{b}, \quad (3.13)$$

mit $\nu = \frac{\mu}{\rho^F}$ als der kinematischen Viskosität.

Die Gleichungen 3.10 und 3.13 werden als die inkompressiblen Navier-Stokes-Gleichungen bezeichnet. Die darin enthaltenen unbekanntenen Variablen sind die Geschwindigkeit \mathbf{u} als vektorielle Größe und der Druck p als skalare Größe.

Diese Gleichungen müssen noch um Anfangs- und Randbedingungen ergänzt werden. Die Anfangsbedingungen sind hierbei Anfangsgeschwindigkeiten \mathbf{u}_0 und -druck p_0 :

$$\mathbf{u}(t=0) = \mathbf{u}_0 \quad \text{und} \quad p(t=0) = p_0. \quad (3.14)$$

Die Randbedingungen bestehen aus den Dirichlet-Randbedingungen, welche Geschwindigkeiten $\bar{\mathbf{u}}$ vorschreiben:

$$\mathbf{u}(t) = \bar{\mathbf{u}} \quad \text{auf} \quad \Gamma^{F,D} \quad (3.15)$$

und Neumann-Randbedingungen, welche die Normalkomponenten der Spannungen am Rand vorschreiben:

$$\mathbf{N}\sigma^F = \mathbf{t}^F \quad \text{auf} \quad \Gamma^{F,N}, \quad (3.16)$$

mit \mathbf{N} als dem Normalenvektor am Rand. Die Ränder $\Gamma^{F,D}$ und $\Gamma^{F,N}$ bezeichnen die Teile des Randes, auf denen Dirichlet- bzw. Neumann-Bedingungen vorgeschrieben werden und überlappen sich nicht.

Werden für die allgemein definierten Dirichlet- und Neumann-Bedingungen bestimmte Werte vorgegeben, entstehen die in der Fluidmechanik gebräuchlichen Randbedingungen. Diese sind z.B. eine Einströmbedingung, welche als Dirichlet-Bedingung am Einströmrand eine bestimmte Geschwindigkeitsverteilung vorschreibt, eine „No-Slip“-Bedingung, welche durch Vorgabe von Geschwindigkeiten zu Null eine Wand abbildet, oder eine Ausströmbedingung, welche z.B. über eine Neumann-Bedingung für den Druck abgebildet werden kann.

Anmerkung: Bei der Betrachtung von Fluidströmungen sind zwei Typen zu unterscheiden. Der erste Typus ist die laminare Strömung, welche als eine geordnete Strömung von sich nicht vermischenden Schichten beschrieben werden kann. Der zweite Typus ist die turbulente Strömung, welche durch das Auftreten von Verwirbelungen charakterisiert ist. Turbulente Strömungen sind stark instationär, dreidimensional und scheinbar chaotisch. Das Strömungsbild ist also charakterisiert durch starke Schwankungen sowohl

in der Längen- als auch in der Zeitskala. Die Navier-Stokes-Gleichungen sind sowohl für laminare als auch turbulente Strömungen gültig.

Eine Kenngröße zur Beurteilung einer Strömung ist die Reynolds-Zahl, definiert über:

$$Re = \frac{u_{rep} \cdot l}{\nu}. \quad (3.17)$$

Sie ist abhängig von einer repräsentativen Geschwindigkeit u_{rep} , der kinematischen Viskosität ν , einem Materialparameter, und einer charakteristischen Länge l im Strömungsfeld, wie z.B. des Zylinderdurchmessers bei einer Zylinderumströmung. Der Umschlag von einer laminaren zu einer turbulenten Strömung, die Transition, wird durch die Reynolds-Zahl charakterisiert. Dieser Umschlag ist problemabhängig. Bei einer Rohrströmung findet er z.B. bei einer Reynolds-Zahl von circa 2.300 statt.

Die numerische Simulation turbulenter Strömungen stellt immer noch eines der größten Probleme der modernen Physik dar. Dies liegt zum einen an den starken Schwankungen der abzubildenden Größen, zum anderen erstrecken sich die auftretenden Längenskalen über einen sehr großen Bereich. Da auch Größen mit kleinen Längenskalen für den energetischen Kaskadenprozess maßgeblich sind, muss das numerische Gitter diese abbilden, woraus sich die Forderung nach einem extrem feinen Netz ergibt.

Wird ein numerisches Gitter verwendet, welches auch die kleinskaligen maßgeblichen Längen abbilden kann, spricht man von einer Direkten Numerischen Simulation (DNS). Da dies mit der momentan zur Verfügung stehenden Rechnerleistung auch auf Supercomputern nur für relativ geringe Reynolds-Zahlen und kleine Rechengebiete möglich ist, sind für die numerische Simulation turbulenter Strömungen zwei weitere Ansätze üblich, welche den Turbulenzeinfluss modellieren:

1. Die RANS-Methode, welche die Mittelung der Bewegungsgleichungen über die Zeit vornimmt und damit die Reynolds-gemittelten Navier-Stokes Gleichungen löst.
2. Die LES-Methode (Grobstruktursimulation), welche die großskaligen Bewegungen auflöst und den Einfluss kleinskaliger Anteile über Modelle abbildet.

In dieser Arbeit werden teils laminare, teils turbulente Strömungen abgebildet. Für die Simulation turbulenter Strömungen wird die RANS-Methode mit dem $k - \omega - SST$ Turbulenzmodell verwendet. Die Abbildung turbulenter Strömungen soll hier aber nicht weiter verfolgt werden, es wird auf die am Anfang des Kapitels angegebene Literatur verwiesen. Wichtig ist die Feststellung, dass sich die im Folgenden vorgestellten Lösungsalgorithmen für laminare Strömungen nur in wenigen Details von der turbulenten Simulation mittels der RANS-Methode unterscheiden und damit die Allgemeingültigkeit der folgenden Abschnitte nicht verletzt wird.

3.2. Diskretisierung im Raum

Die inkompressiblen Navier-Stokes-Gleichungen stellen ein gekoppeltes System nichtlinearer partieller Differentialgleichungen dar. Für eine Lösung mittels numerischer Methoden sind die gebräuchlichsten Ansätze die Finite-Differenzen-Methode (FDM), die Finite-Volumen-Methode (FVM) und die Finite-Elemente-Methode (FEM). Während sich in der Strukturmechanik eindeutig die Finite-Elemente-Methode als Standard etabliert hat, lässt sich in der Fluidmechanik kein überwiegend benutztes Diskretisierungsverfahren festlegen. Im Rahmen dieser Arbeit wird die Finite-Volumen-Methode benutzt, welche auch in der überwiegenden Zahl kommerzieller Produkte Anwendung findet.

Bei der Finite-Volumen-Methode wird das Simulationsgebiet in einzelne endliche Volumen (Kontrollvolumen, KV) unterteilt. Bei der hier verwendeten zellenbasierten FVM definiert das Zentrum der Kontrollvolumina den Punkt, an dem die unbekannt Variablen diskret beschrieben werden. In jedem Kontrollvolumen werden die Erhaltungsgleichungen in Integralform formuliert. Diese erhält man durch Integration der Differentialgleichung über ein einzelnes Kontrollvolumen. Daraus ergibt sich ein großer Vorteil der FVM, die Erhaltungsgleichungen sind nach der Lösung des Problems automatisch erfüllt.

Die einzelnen Terme der Navier-Stokes-Gleichungen werden unabhängig voneinander diskretisiert. Für den Fall der Impulserhaltungsgleichung (3.13) ergeben sich also nach Formulierung in Integralform die folgenden vier Terme:

1. Der Term der Zeitableitung:

$$\int_V \frac{\partial \mathbf{u}}{\partial t} dV. \quad (3.18)$$

Dieser Term wird im folgenden Abschnitt 3.3 näher besprochen.

2. Der konvektive Term:

$$\int_V \nabla \cdot (\mathbf{u}\mathbf{u}) dV. \quad (3.19)$$

Der konvektive Term wird mittels des Gauss-Theorems in ein Integral über die Oberflächen des KV transformiert. Das Integral entspricht der Summe über die Werte pro Oberfläche. Für die Approximation der Werte pro Oberfläche existiert eine große Anzahl an Ansätzen, welche sich bezüglich Stabilität und Genauigkeit unterscheiden. Erwähnt werden soll hier exemplarisch das Upwind-Interpolation-Schema und das Zentraldifferenzen-Verfahren.

3. Der diffusive Term:

$$\nu \int_V \nabla \cdot (\nabla \mathbf{u}) dV \quad (3.20)$$

Der diffusive Term wird ebenfalls mittels des Gauss-Theorems in ein Integral über die Oberflächen des KVs transformiert. Die Approximation des Geschwindigkeitsgradienten kann z.B. über lineare Interpolation der Werte aus Nachbarvolumina geschehen.

4. Der Term des Druckgradienten:

$$-\frac{1}{\rho^E} \int_V \nabla p dV. \quad (3.21)$$

Auch dieser Term wird mittels des Gauss-Theorems in ein Integral über die Oberflächen des KVs transformiert und z.B. mittels linearer Interpolation approximiert.

5. Der Term der Volumenkräfte (Quellterm):

$$\int_V \mathbf{b} dV. \quad (3.22)$$

Dieser wird durch eine Integration über das gesamte Kontrollvolumen bestimmt, wobei ein konstanter oder linearer Verlauf zwischen den Zentren der Kontrollvolumina angenommen werden kann.

3.3. Diskretisierung in der Zeit

Für die Diskretisierung des Problems in der Zeit ist der Term der Zeitableitung zu behandeln:

$$\int_V \frac{\partial \mathbf{u}}{\partial t} dV. \quad (3.23)$$

Bei der Diskretisierung dieses Terms ist, ähnlich zur Vorgehensweise in der Strukturmechanik, in explizite und implizite Verfahren zu unterscheiden. Explizite Verfahren bestimmen den Zustand einer Variablen zum nächsten Zeitpunkt t_{n+1} über bereits bekannte Zustände am Zeitpunkt t_n , während implizite Verfahren eine Bedingungsgleichung zum Zeitpunkt t_{n+1} aufstellen, zu deren Erfüllung die Lösung eines Gleichungssystems notwendig ist. Explizite Verfahren sind daher numerisch deutlich weniger aufwendig. Allerdings sind explizite Verfahren nur bedingt stabil und werden bei Überschreiten einer kritischen Zeitschrittweite $\Delta t > \Delta t_{krit}$ instabil. In dieser Arbeit werden implizite Zeitdiskretisierungsmethoden verwendet. Bei entsprechender Wahl der Integrationsparameter tritt hierbei keine Beschränkung der Zeitschrittweite auf.

Durch eine Mittelung über das Kontrollvolumen wird das Integral in eine diskrete Gleichung mit einer im Zentrum des Volumens definierten Variablen \mathbf{u}_P überführt:

$$\int_V \frac{\partial \mathbf{u}}{\partial t} dV \approx \frac{\partial \mathbf{u}_P}{\partial t} V. \quad (3.24)$$

Im Folgenden wird eine Diskretisierung mittels des BDF-Verfahrens („Backward Differencing Formula“) zweiter Ordnung verwendet:

$$\frac{\partial \mathbf{u}_P}{\partial t} = \frac{3\mathbf{u}_P^{n+1} - 4\mathbf{u}_P^n + \mathbf{u}_P^{n-1}}{2\Delta t}. \quad (3.25)$$

Dieses zählt zu den impliziten Mehrschrittverfahren, da es zur Bestimmung der Werte zum Zeitpunkt t_{n+1} Werte von zwei vorhergehenden Schritten benötigt: t_n und t_{n-1} . Es ist unbedingt stabil und von einer Genauigkeit zweiter Ordnung.

3.4. Druckkorrekturverfahren PISO

Die Wahl der Finiten-Volumen-Methode erlaubt die räumliche und zeitliche Diskretisierung der einzelnen Terme der inkompressiblen Navier-Stokes-Gleichungen. Für die Lösung des dabei entstehenden diskreten und nichtlinearen Problems existieren eine Vielzahl von Ansätzen. Eine besondere Schwierigkeit stellt hierbei das Fehlen einer eigenen Gleichung für den Druck dar, der nur über seinen Gradienten in den drei Impulsgleichungen vorkommt.

Die beiden gängigsten Ansätze zur Bestimmung von Geschwindigkeit \mathbf{u} und Druck p sind gekoppelte Verfahren und Druckkorrekturverfahren. Gekoppelte Verfahren lösen die beschreibenden Gleichungen für Massen- und Impulserhalt simultan und mit Hilfe eines einzelnen, globalen Gleichungssystems. Druckkorrekturverfahren hingegen lösen die Gleichungen sequentiell. Gekoppelte Verfahren sind i.d.R. robuster und erlauben dadurch größere Zeitschritte, benötigen aber pro Zeitschritt mehr Rechenzeit und insgesamt auch mehr Speicher als Druckkorrekturverfahren. Innerhalb dieser Arbeit wird ein Druckkorrekturverfahren verwendet und im Folgenden genauer beschrieben. Bei der Lösung der gekoppelten Gleichungen muss ein Druck- und Geschwindigkeitsfeld gefunden werden, welches sowohl die Impuls- als auch die Kontinuitätsgleichung erfüllt ([Sch99, FP99, Jas96]). Die Vorgehensweise von Druckkorrekturverfahren ist es, zuerst aus der Impulsgleichung ein Geschwindigkeitsfeld zu berechnen (Prädiktorschritt). Mit diesem Geschwindigkeitsfeld wird die Druckgleichung aufgestellt und nach dem Druck gelöst. Das Geschwindigkeitsfeld wird korrigiert, um die Kontinuitätsgleichung zu erfüllen. Dieses Vorgehen wird iterativ ausgeführt, bis sowohl Impuls- als auch Kontinuitätsgleichung gleichzeitig erfüllt sind (PISO-Iterationen).

Der gesamte Fluidlösungsalgorithmus besteht aus drei aufeinander aufbauenden Teilen:

- Als Kern fungiert das PISO Druckkorrekturverfahren (*Pressure Implicit with Splitting of Operators*) nach Issa ([Iss85, IGW86]). Mit diesem Verfahren werden die Drücke und Geschwindigkeiten der transienten Problemstellung bestimmt.
- Zur Abbildung der Randbewegung des Fluidfeldes wird das PISO-Verfahren in einen ALE-Ansatz eingebettet und die Netzverformung und damit die relative Netzgeschwindigkeit berechnet.
- Um innerhalb eines impliziten Kopplungsalgorithmus eine effiziente Fluidlösung zu erhalten, wird der PISO-Algorithmus derart erweitert, dass in jeder Kopplungsiteration die Ergebnisse der vorherigen Iteration als Initiallösung verwendet werden. Dadurch wird eine schnellere Konvergenz erreicht. Dabei ist vor allem auf die richtige Behandlung der zeitlichen Ableitung und der relativen und absoluten Geschwindigkeiten zu achten. Dieser Ansatz wird im Folgendem als „angepasster Feldlöser“ bezeichnet und stellt eine Erweiterung eines Standardlöseransatzes für eine gekoppelte Berechnung dar.

Bei der folgenden Beschreibung des PISO-Druckkorrekturverfahrens werden die Terme „Fluss“ und „linearisierte Momentengleichung“ verwendet. Als Fluss wird das Integral

einer Geschwindigkeit über eine Fläche bezeichnet. Im Kontext der Finiten Volumen Methode sind die Flächen die Begrenzungen der einzelnen Kontrollvolumen. Aus einem bekannten Geschwindigkeitsfeld können die Flüsse also über eine Interpolation bestimmt werden. Die linearisierte Momentengleichung ergibt sich, indem in Gl. 3.13 der nichtlineare konvektive Term $\nabla \cdot (\mathbf{u}\mathbf{u})$ linearisiert und der erste Geschwindigkeitsterm in einen Fluss transferiert wird: $\phi \nabla \mathbf{u}$.

Das PISO-Druckkorrekturverfahren lässt sich folgendermaßen beschreiben (Iterationsindex n in der Zeit, Iterationsindex j für PISO-Druckkorrekturen):

1. **Prädiktorschritt:** Berechnung eines Prädiktors für die Geschwindigkeiten mittels der linearisierten Momentengleichung. Dazu werden die Flüsse ϕ_n und Drücke p_n des letzten Zeitschritts benutzt:

$$\frac{\partial \mathbf{u}_{n+1}^P}{\partial t} + \phi_n \nabla \mathbf{u}_{n+1}^P - \nabla \cdot (\nu \nabla \mathbf{u}_{n+1}^P) = -\frac{1}{\rho} \nabla p_n. \quad (3.26)$$

Die Lösung dieser Gleichung ergibt eine Approximation \mathbf{u}_{n+1}^P des neuen Geschwindigkeitsfeldes, welches nicht divergenzfrei ist.

2. **Drucklösung:** Mittels der approximierten Geschwindigkeiten \mathbf{u}_{n+1}^P können die temporären Flüsse $\phi^{*,j}$ bestimmt und die Druckgleichung aufgestellt werden:

$$\nabla \cdot \left(\frac{1}{a_p^j} \nabla p_{n+1}^{j+1} \right) = \nabla \cdot \phi^{*,j}. \quad (3.27)$$

Hierbei werden mit a_p^j Matrixkoeffizienten aus dem gewähltem Diskretisierungsverfahren bezeichnet. Diese Schreibweise orientiert sich an der in [Jas96] gewählten Methode, worauf für weitere Details verwiesen wird.

Die Lösung der Druckgleichung ergibt eine Approximation p_{n+1}^{j+1} des neuen Druckfeldes. Bei nicht-orthogonalen Netzen wird die Druckgleichung mehrfach gelöst. Dabei wird in der ersten Lösung angenommen, dass die Oberflächen zwischen den Zellen senkrecht zur Verbindung zwischen den Zellzentren stehen, also ein orthogonales Netz vorliegt. Über die Iterationen wird diese Annahme korrigiert, bis der tatsächliche Zustand hergestellt ist.

3. **Explizite Geschwindigkeitskorrektur:** Mittels des neuen Druckfeldes p_{n+1}^{j+1} lässt sich durch die Druck-Geschwindigkeitskopplung ein neues Geschwindigkeitsfeld \mathbf{u}_{n+1}^{j+1} explizit, also ohne Lösung eines Gleichungssystems, bestimmen. Damit werden auch die temporären Flüsse korrigiert.
4. **PISO-Iterationen:** Mit dem neuen Geschwindigkeitsfeld \mathbf{u}_{n+1}^{j+1} als Eingabeparameter werden die Schritte 2. Drucklösung und 3. Geschwindigkeitskorrektur solange wiederholt, bis Konvergenz erreicht ist.

Ein detaillierter Algorithmus zur Lösung des nichtlinearen transienten Fluidproblems unter Verwendung des PISO-Druckkorrekturverfahrens ist in Algorithmus 3 gegeben.

Algorithmus 3 PISO-Druckkorrekturverfahren

```

1:  $n = 1, \quad t_n = t_0$ 
2: initialisiere  $\mathbf{u}_n^0, \phi_n^0$  und  $p_n^0$ 
3: Iterationen über Zeitschritte:
4: while ( $t_n \leq t_{ges}$ ) do
5:   Prädiktorschritt:
6:   Löse  $\frac{\partial \mathbf{u}_{n+1}^P}{\partial t} + \phi_n \nabla \mathbf{u}_{n+1}^P - \nabla \cdot (\nu \nabla \mathbf{u}_{n+1}^P) = -\frac{1}{\rho} \nabla p_n$ 
7:      $\rightarrow \mathbf{u}_{n+1}^P$  mit  $\nabla \cdot \mathbf{u}_{n+1}^P \neq 0$ 
8:    $j = 1$ 
9:   PISO-Iterationen:
10:  while (keine Fluidkonvergenz) do
11:    if (Fluidlösung nicht konvergiert) then
12:      Drucklösung:
13:      Löse  $\nabla \cdot \left( \frac{1}{a_p^j} \nabla p_{n+1}^{j+1} \right) = \nabla \cdot \phi^{*,j}$ 
14:      Korrektur der Geschwindigkeit und temporären Flüsse:
15:       $\mathbf{u}_{n+1}^{j+1} = \mathbf{u}_{n+1}^{j+1}(p_{n+1}^{j+1})$ 
16:       $\phi^{*,j+1} = \phi^{*,j+1}(\mathbf{u}_{n+1}^{j+1})$ 
17:       $j \leftarrow j + 1$ 
18:    else
19:      Abbruch Fluiditerationen, gehe zu nächstem Zeitschritt
20:       $\mathbf{u}_{n+1} = \mathbf{u}_{n+1}^j$ 
21:       $p_{n+1} = p_{n+1}^j$ 
22:    end if
23:  end while
24:   $t_{n+1} = t_n + \Delta t_n$ 
25:   $n \leftarrow n + 1$ 
26: end while

```

3.5. Raum- und zeitveränderliches Fluidgebiet

Aus der Problemstellung der Fluid-Struktur-Wechselwirkung ergibt sich, dass das Koppungsinterface als Berandung des Fluidgebietes seine Position im Raum während der Simulationszeit verändert. Die Interfacebewegung wird durch den Kopplungsalgorithmus vorgegeben. Die damit einhergehende zeitliche und räumliche Veränderung des Fluidgebietes muss innerhalb des Fluidlösungsalgorithmus abgebildet werden. Dazu wird der sogenannte ALE-Ansatz (*Arbitrary Lagrangian Eulerian*) gewählt ([HAC74, DHPRF04, Wal99, WGG⁺06]). Dieser Ansatz erlaubt es, die Gebietsverformung und das Lösen der Fluidgleichungen separat zu behandeln. Die Gebietsverformung wird über eine Anpassung des Fluidgitters an die vorgegebene Randverschiebung abgebildet. Hierzu ist ein Netzverformungsalgorithmus notwendig, welcher die zusätzliche Lösung eines Gleichungssystems erfordert. Nach Berechnung der Netzverformung ist die zugehörige Netzgeschwindigkeit bekannt, welche für das Lösen der an den ALE-Ansatz angepassten Fluidgleichungen benötigt wird. Diese Anpassung besteht darin, die Fluidgleichungen relativ zur Netzverformung als Kombination von Lagrangescher und Eulerscher Betrachtungsweise zu beschreiben. Da nur der Term der zeitlichen Ableitung innerhalb der Navier-Stokes-Gleichungen von der Zeit abhängt, wirkt sich die zeitliche Veränderung der Elemente und damit der Kontrollvolumen nur innerhalb dieses Terms aus. Mit Hilfe der Leibniz-Regel wird der Term der Zeitableitung in folgende zwei Terme aufgespalten ([FP08]):

$$\frac{\partial \mathbf{u}}{\partial t} = \frac{\partial \mathbf{u}^{abs}}{\partial t} - \frac{\partial \mathbf{u}^g}{\partial t}. \quad (3.28)$$

Der Anteil aus der Netzgeschwindigkeit wird in einen Fluss transformiert und mit dem konvektiven Term der Momentengleichung zusammengefasst. Bei der Lösung des Prädiktorschrittes (Gl. 3.26) werden also die absoluten Geschwindigkeiten verwendet und der Term der Flüsse ϕ_n mit den Flüssen aus der Netzgeschwindigkeit erweitert. Damit folgt für den Prädiktorschritt innerhalb des PISO-Algorithmus:

$$\frac{\partial \mathbf{u}_{n+1}^P}{\partial t} + (\phi_n^{abs} - \phi_n^g) \nabla \mathbf{u}_{n+1}^P - \nabla \cdot (\nu \nabla \mathbf{u}_{n+1}^P) = -\frac{1}{\rho} \nabla p_n. \quad (3.29)$$

Hierbei sind ϕ_n^{abs} die aus dem letzten Zeitschritt bekannten absoluten Flüsse. ϕ_n^g sind die Netzflüsse, welche sich aus (Netz-)Verformung mal Fläche eines Kontrollvolumenrandes ergeben.

Für die Erweiterung des PISO-Lösungsalgorithmus ist es vorteilhaft, bei den zu behandelnden Flüssen zu unterscheiden in Flüsse als absolute Werte ϕ^{abs} , Flüsse relativ zur Netzgeschwindigkeit ϕ^{rel} und Flüsse aus der Netzgeschwindigkeit ϕ^g . Damit gilt:

$$\phi^{abs} = \phi^{rel} + \phi^g. \quad (3.30)$$

Ein wichtiger Aspekt bei der Behandlung bewegter Fluidnetze ist die Einhaltung der geometrischen Erhaltungsgleichungen („geometric conservation laws“, GCL, oder auch „space conservation laws“, SCL, siehe z.B. [FP08, DP88, LF96]). Bei deren Verletzung ergeben sich Fehler in Form von künstlichen zusätzlichen Massenquellen am Interface.

Um dies zu vermeiden, muss die Berechnung der Flüsse in Abhängigkeit des verwendeten Zeitintegrationsverfahrens erfolgen (siehe z.B. [TJ08]).

Zur Bestimmung der Flüsse ϕ aus der Netzgeschwindigkeit \mathbf{u}^g ist es notwendig, die Netzverformung zu bestimmen. Hierbei wird unter Vorgabe der Randverformungen \mathbf{d}^Γ am Kopplungsinterface Γ ein Gleichungssystem mit den Knotenverschiebungen des Fluidgitters als Unbekannte gelöst. Sind die Verschiebungen bekannt, lassen sich daraus die Netzgeschwindigkeiten bestimmen. Für die Berechnung der Netzverformung existieren eine Vielzahl von Ansätzen. Dabei sind vor allem zwei Ziele von Wichtigkeit, die i.d.R. leider konträr sind:

1. Hohe numerische Effizienz, um den zusätzlichen Aufwand in Folge der Netzverformung gering zu halten.
2. Erhaltung der Elementqualität in Bezug auf ihre Geometrie auch bei großen Randverformungen, da dies einen hohen Einfluss sowohl auf das Konvergenzverhalten als auch auf die Ergebnisqualität des Fluidlösungsalgorithmus hat.

An vorhandenen Ansätzen sind unter anderen zu nennen: Einfache Dreh- und Senkfedermodelle, die jedem Fluidknoten eine Federsteifigkeit zuordnen ([Bat90, Blo00, DF02]), Pseudo-Strukturmodelle, die jedem Knoten eine künstliche Steifigkeit auf Basis eines einfachen linearen Pseudostrukturmodells zuordnen ([JT94]), Methoden die eine Laplace-Gleichung mit zusätzlicher Diffusion auf dem Fluidgebiet lösen ([LY96]) und Methoden welche auf radialen Basisfunktionen beruhen ([BLVOB07, BMT⁺08]).

Innerhalb dieser Arbeit wird zur Berechnung der Netzverformung die Laplace-Methode mit variablem Diffusionskoeffizienten benutzt ([JT07, TJ08]). Dazu wird auf dem Fluidgebiet eine Laplacegleichung mit den Knotenverschiebungen \mathbf{d}^F als Unbekannte gelöst:

$$\nabla \cdot (\gamma \nabla \mathbf{d}^F) = 0 \quad \text{in} \quad \Omega^F, \quad (3.31)$$

und den Randbedingungen:

$$\mathbf{d}^F = \mathbf{d}^\Gamma \quad \text{auf} \quad \Gamma^{FSI}, \quad (3.32)$$

$$\mathbf{d}^F = 0 \quad \text{auf} \quad \Gamma^F \quad \text{ohne} \quad \Gamma^{FSI}. \quad (3.33)$$

γ ist hierbei ein zellenweise variabler Diffusionskoeffizient, dessen Wahl maßgeblich die Qualität der Netzverformung beeinflusst. Je höher der Diffusionskoeffizient γ_i einer Zelle, desto steifer verhält es sich in Bezug auf seine Relativverformung. Um die relativen Verformungen in Interfacenähe gering zu halten (und damit die Netzqualität zu erhalten), muss sich die am Kopplungsinterface vorgegebene Randverformung auf einen möglichst großen Bereich des Fluidgebietes verteilen. Dazu ist es notwendig, möglichst hohe Diffusionskoeffizienten und damit Zellensteifigkeiten in der Umgebung des Kopplungsinterfaces zu erhalten. Hierzu wird der Diffusionskoeffizient in Abhängigkeit des geringsten Abstandes zwischen Zellenzentrum und Kopplungsinterface r_i bestimmt. Zur Bestimmung eignet sich z.B. der Kehrwert des Abstandes zum Quadrat:

$$\gamma_i(r_i) = \frac{1}{r_i^2}. \quad (3.34)$$

Diese Methode stellt einen guten Kompromiss zwischen numerischem Aufwand und Qualitätserhalt der Fluidelemente dar. Pro Knoten ergeben sich drei Verschiebungsfreiheitsgrade. Da sich die Verschiebungen in den drei Raumrichtungen voneinander unabhängig formulieren lassen, müssen drei lineare Gleichungssysteme mit der Anzahl an Knoten als Unbekannte bei jeder Netzverformung gelöst werden. Als Nachteil dieser Methode ist zu nennen, dass die Knoten keine Verdrehsteifigkeit aufweisen und sich dadurch bei einer Verformung des Netzes parallel zu einem Rand (Schubverformung) eine geringe Netzqualität ergibt. Innerhalb der in dieser Arbeit untersuchten Beispiele treten Verformungen überwiegend senkrecht zum Rand auf, so dass dieser Nachteil nicht zum Tragen kommt.

3.6. Fluidlösungsalgorithmus für implizite Kopplungsalgorithmen

Innerhalb dieses Abschnitts wird der Fluidlösungsalgorithmus vorgestellt, wie er innerhalb eines impliziten Kopplungsverfahrens verwendet wird. Die impliziten Kopplungsalgorithmen werden zwar erst im nachfolgenden Kapitel 4 vorgestellt, aber einige der zum Verständnis notwendigen Details werden bereits an dieser Stelle wiedergegeben.

Aufgabe des Fluidfeldes als Dirichlet-Partition innerhalb des gekoppelten Problems ist es, unter Vorgabe von Interfaceverschiebungen die zugehörigen Interfacebelastungen aus Druck und Geschwindigkeit zu berechnen.

Der gesamte Lösungsalgorithmus besteht hierbei aus drei ineinander geschachtelten Schleifen bzw. Iterationen. Allem übergeordnet ist die Zeitschrittsschleife (Iterationsindex n). Pro Zeitschritt ergibt sich die Schleife über die Kopplungsiterationen (Iterationsindex k). In jeder Kopplungsiteration wird durch den Kopplungsalgorithmus eine neue Interfaceverschiebung \mathbf{d}^Γ vorgegeben. Das Fluidgebiet wird mittels der Laplace-Netzverformungsmethode angepasst. Die innerhalb eines ALE-Ansatzes formulierten Fluidgleichungen werden mittels des PISO-Druckkorrekturverfahrens gelöst, woraus sich als innerste Schleife die PISO-Iterationen ergeben (Iterationsindex j).

Ein detaillierter Algorithmus zur Lösung des nichtlinearen transienten Fluidproblems innerhalb einer impliziten Kopplung ist in Algorithmus 4 gegeben.

Wie aus dem Algorithmus ersichtlich, müssen bei einem dreidimensionalen Problem in jeder Kopplungsiteration die folgenden linearen Gleichungssysteme der Größe n_{Ele}^F gelöst werden:

1. Drei Gleichungssysteme bei Berechnung der Netzverformung (die Laplacegleichung wird für jede Raumrichtung aufgestellt).
2. Drei Gleichungssysteme im Prädiktorschritt (die Impulsgleichung wird für jede Raumrichtung aufgestellt).
3. Je PISO-Iteration Lösung der skalaren Druckgleichung. Bei nicht-orthogonalen Netzen mehrfaches Lösen der Druckgleichung mit Anzahl gleich Korrekturschritte je PISO-Iteration.

Anmerkung: Um eine möglichst effiziente Lösung des Fluidproblems innerhalb einer impliziten Kopplung zu gewährleisten, wird der PISO-Lösungsalgorithmus angepasst. Aus der Annahme eines konvergierenden gekoppelten Problems folgt, dass auch die erhaltenen Interfaceverschiebungen \mathbf{d}^Γ eine konvergierende Reihe darstellen. Dies bedeutet, dass die in den einzelnen Kopplungsiterationen berechneten Ergebnisse für \mathbf{u} und p dem konvergierten Ergebnis näher sind als die am Anfang des Zeitschritts benutzten Initialwerte. Damit werden im Prädiktorschritt nicht die Flüsse ϕ_n und Drücke p_n des letzten Zeitschritts verwendet, um eine Näherung für das Geschwindigkeitsfeld zu erhalten, sondern die Ergebnisse des vorherigen Kopplungsschrittes ϕ_{n+1}^k und p_{n+1}^k . Dieser Ansatz wird im weiteren Verlauf als „angepasster Feldlöser“ bezeichnet.

Algorithmus 4 Fluidlösungsalgorithmus für implizite Kopplungsalgorithmen

```

1:  $n = 1, \quad t_n = t_0$ 
2: Initialisiere  $\mathbf{u}_n^0, \phi_n^0$  und  $p_n^0$ 
3: Iterationen über Zeitschritte:
4: while ( $t_n \leq t_{ges}$ ) do
5:    $k = 1$ 
6:   Iterationen über Kopplungsschritte:
7:   while (keine Kopplungskonvergenz) do
8:     Netzverformung:
9:     Aufbringen der Interfaceverschiebung  $\mathbf{d}_{n+1}^{\Gamma, k+1}$ 
10:    Löse  $\nabla \cdot (\gamma \nabla \mathbf{d}^F) = 0 \quad \rightarrow \quad \phi_{n+1}^{g, k+1}$ 
11:    Prädiktorschritt:
12:    Löse  $\frac{\partial \mathbf{u}_{n+1}^{k+1, P}}{\partial t} + \phi_{n+1}^{rel, k} \nabla \mathbf{u}_{n+1}^{k+1, P} - \nabla \cdot (\nu \nabla \mathbf{u}_{n+1}^{k+1, P}) = -\frac{1}{\rho} \nabla p_{n+1}^k \quad \rightarrow \quad \mathbf{u}_{n+1}^{k+1, P}$ 
13:     $j = 1$ 
14:    PISO-Iterationen:
15:    while (keine Fluidkonvergenz) do
16:      if (Fluidlösung nicht konvergiert) then
17:        Drucklösung:
18:        Löse  $\nabla \cdot \left( \frac{1}{a_p^{k+1, j}} \nabla p_{n+1}^{k+1, j+1} \right) = \nabla \cdot \phi^{*, k+1, j}$ 
19:        Geschwindigkeitskorrektur:
20:         $\mathbf{u}_{n+1}^{k+1, j+1} = \mathbf{u}_{n+1}^{k+1, j+1}(p_{n+1}^{k+1, j+1})$ 
21:         $j \leftarrow j + 1$ 
22:      else
23:        Abbruch Fluiditerationen, gehe zu nächster Kopplungsiteration
24:         $\mathbf{u}_{n+1}^{k+1} = \mathbf{u}_{n+1}^{k+1, j}, \quad p_{n+1}^{k+1} = p_{n+1}^{k+1, j}$ 
25:      end if
26:    end while
27:    if (Kopplung konvergiert) then
28:      Abbruch Kopplungsiterationen, gehe zu nächstem Zeitschritt
29:       $\mathbf{u}_{n+1} = \mathbf{u}_{n+1}^k, \quad p_{n+1} = p_{n+1}^k$ 
30:    else
31:      Gehe zu nächster Kopplungsiteration
32:       $k \leftarrow k + 1$ 
33:    end if
34:  end while
35:   $t_{n+1} = t_n + \Delta t_{n+1}$ 
36:   $n \leftarrow n + 1$ 
37: end while

```

3.7. Zusammenfassung

In diesem Kapitel wird das Fluidfeld als Teilfeld des gekoppelten Problems der Fluid-Struktur-Wechselwirkung behandelt. Das zugehörige Fluidmodell besteht aus den inkompressiblen Navier-Stokes-Gleichungen mit ALE-Ansatz. Das Fluidfeld wird mittels der Finiten-Volumen-Methode diskretisiert und das diskrete nichtlineare Problem mittels des PISO-Druckkorrekturverfahrens gelöst. Die Netzverformung wird mittels der Laplace-Methode mit variablem Diffusionskoeffizienten abgebildet. Das Kapitel endet mit der detaillierten Vorstellung des Fluidfeldlösungsalgorithmus innerhalb einer impliziten Kopplung. Dabei wird der Term „angepasster Feldlöser“ eingeführt, welcher eine spezifische Anpassung des Löseralgorithmus an die implizite Kopplung bedeutet. Ein angepasster Feldlöser macht sich die Konvergenzeigenschaften des übergeordneten gekoppelten Problems zu Nutze, um die benötigte Anzahl an PISO-Iterationen in der Fluidfeldlösung pro Kopplungsiteration zu verringern.

4. Fluid-Struktur-Wechselwirkung im partitionierten Ansatz

In diesem Kapitel wird die numerische Simulation der Fluid-Struktur-Wechselwirkung behandelt. Das durch die Wechselwirkung von Fluid und Struktur entstehende Problem wird hierbei auch Mehrfeldproblem, gekoppeltes Problem oder oberflächengekoppeltes Problem genannt. Als Basis dienen hierbei die in den Kapiteln 2 und 3 vorgestellten Methoden zur Lösung der Teilfelder Struktur und Fluid.

Den methodischen Schwerpunkt stellt die Simulation von stark gekoppelten und transienten Problemstellungen dar. Die daraus resultierenden hohen Nichtlinearitäten und langen Simulationszeiträume führen zu einem extrem hohen numerischen Lösungsaufwand, weswegen effiziente und stabile Lösungsansätze von besonders großer Bedeutung sind. Als Basis wird ein partitionierter Ansatz gewählt. Dabei werden die Teilfelder getrennt voneinander behandelt und deren Wechselwirkung über einen impliziten Kopplungsalgorithmus abgebildet.

Das Ziel dieses Kapitels ist es, einen Überblick zu den im wissenschaftlichen Bereich momentan gängigsten Lösungsansätzen, welche zur Simulation stark gekoppelter und transientser Probleme geeignet sind, zu geben. Darauf beruhend werden einige dieser Verfahren im Detail vorgestellt und innerhalb eines numerischen Beispiels in Abschnitt 6.1 in Bezug auf Effizienz und Stabilität miteinander verglichen und beurteilt.

Eine wichtige Eigenschaft der gekoppelten Lösungsansätze stellt das Anforderungsprofil an die Teilfeldlöser dar. Können für die Lösung der Teilfelder weiterhin standardisierte Ansätze verwendet werden, ist dies ein großer Vorteil für die gesamte Simulationsmethode, da es die Wiederverwendung existierender Software erlaubt. Deshalb wird hier jeder Lösungsansatz auch auf diesen Aspekt hin untersucht.

Dieses Kapitel ist wie folgt aufgebaut: In Abschnitt 4.1 wird das gekoppelte Problem definiert. Dazu zählen die Beschreibung der Geometrie, ein Überblick zu den Grundgleichungen der Teilfelder und die Operatorschreibweise für den partitionierten Ansatz. In Abschnitt 4.2 wird die Grundform eines impliziten Kopplungsalgorithmus dargestellt und in Abschnitt 4.3 die verwendeten Konvergenzkriterien besprochen. In Abschnitt 4.4 werden über den Interfaceprädiktor Möglichkeiten zur Bestimmung einer Initiallösung für den nichtlinearen Lösungsverlauf dargestellt. In Abschnitt 4.5 wird ein Überblick über die im wissenschaftlichen Bereich derzeit verwendeten Verfahren gegeben. Einige von diesen werden ausgewählt und im Detail in den Abschnitten 4.6 bis 4.8 vorgestellt. Diese Verfahren werden auch in den numerischen Beispielen verwendet. Das Kapitel endet mit einer Zusammenfassung in 4.9.

4.1. Definition des gekoppelten Problems

Zur Definition des gekoppelten Problems innerhalb eines partitionierten Ansatzes ist es notwendig, die folgenden Bereiche zu betrachten: Die geometrische Beschreibung des Simulationsgebietes, die Grundgleichungen der Teilfelder Struktur und Fluid und die Kopplungsbedingungen am gemeinsamen Interface. Darauf basierend kann das gekoppelte Problem in der sogenannten Operatorschreibweise formuliert werden.

4.1.1. Geometrie

Betrachtet wird das gesamte Simulationsgebiet Ω . Dieses besteht aus zwei Teilgebieten, dem Strukturfeld im Gebiet Ω^S und seiner Berandung Γ^S und dem Fluidfeld im Gebiet Ω^F mit seiner Berandung Γ^F . Es gilt: $\Omega = \Omega^S + \Omega^F$. Die Teilgebiete besitzen einen gemeinsamen Rand Γ , definiert über $\Gamma = \Omega^S \cap \Omega^F$ und es gilt $\Gamma \subset \Gamma^F$ und $\Gamma \subset \Gamma^S$. Der gemeinsame Rand Γ wird im Folgenden als Kopplungsinterface bzw. nur Interface bezeichnet. Eine graphische Darstellung der Geometriedefinition ist in Abbildung 4.1 gegeben.

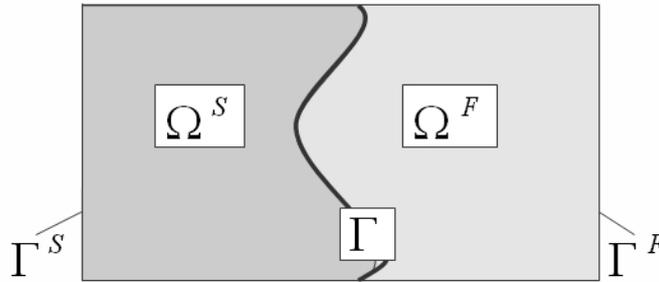


Abbildung 4.1.: Geometriedefinition bei oberflächengekoppelten Problemen

4.1.2. Grundgleichungen der Teilfelder

Das **Strukturfeld** wird durch die Grundgleichungen der Elastodynamik beschrieben, wie sie in Kapitel 2 hergeleitet werden. Diese bestehen aus dem Impulsgleichgewicht und der Zustandsgleichung:

$$\begin{aligned} \rho^S \frac{d^2 \mathbf{d}}{dt^2} - \nabla \cdot (\mathbf{F} \cdot \mathbf{S}) &= \mathbf{f}, \\ \mathbf{S} &= \mathbf{C} : \mathbf{E} \quad \text{in } \Omega^S, \end{aligned} \quad (4.1)$$

mit \mathbf{d} als Verschiebungen und \mathbf{f} als Lasten. Die zugehörigen Rand- und Initialbedingungen lauten:

$$\mathbf{d}(x, t) = \mathbf{d}^{\Gamma^S}(t), \quad \mathbf{f}(x, t) = \mathbf{f}^{\Gamma^S}(t) \quad \text{auf } \Gamma^S \quad (4.2)$$

und

$$\mathbf{d}(x, 0) = \mathbf{d}_0(x), \quad \mathbf{f}(x, 0) = \mathbf{f}_0(t) \quad \text{in } \Omega^S. \quad (4.3)$$

Für die Diskretisierung im Raum wird die Finite-Elemente-Methode benutzt, für die Diskretisierung in der Zeit das Generalized- α Zeitintegrationsverfahren. Zur Lösung des diskretisierten nichtlinearen Problems wird das Newton-Verfahren angewendet. Der Strukturlösungsalgorithmus innerhalb einer impliziten Kopplung ist in Algorithmus 1 angegeben.

Das **Fluidfeld** wird durch die inkompressiblen Navier-Stokes-Gleichungen beschrieben, wie sie in Kapitel 3 hergeleitet werden. Diese bestehen in der ALE-Form aus dem Massen- und Impulserhalt:

$$\begin{aligned} \nabla \cdot \mathbf{u} &= 0 \\ \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} - \mathbf{u}_G) \nabla \mathbf{u} - \nabla \cdot (\nu \nabla \mathbf{u}) + \frac{1}{\rho} \nabla p &= 0 \quad \text{in } \Omega^F \end{aligned} \quad (4.4)$$

mit \mathbf{u} als absoluter Geschwindigkeit, \mathbf{u}_G als Gittergeschwindigkeit und p als Druck. Die zugehörigen Rand- und Initialbedingungen lauten:

$$\begin{aligned} \mathbf{u}(x, t) &= \mathbf{u}^{\Gamma^F}(t), \quad \mathbf{u}_G(x, t) = \mathbf{u}_G^{\Gamma^F}(t), \quad p(x, t) = p^{\Gamma^F}(t) \quad \text{auf } \Gamma^F, \\ \mathbf{u}(x, 0) &= \mathbf{u}_0(x), \quad \mathbf{u}_G(x, 0) = \mathbf{u}_{G,0}(x), \quad p(x, 0) = p_0(x) \quad \text{in } \Omega^F. \end{aligned} \quad (4.5)$$

Für die Diskretisierung im Raum wird die Finite-Volumen-Methode benutzt, für die Diskretisierung in der Zeit das BDF2-Verfahren. Zur Lösung des diskretisierten nichtlinearen Problems wird das PISO-Druckkorrekturverfahren angewendet. Der Fluidlösungsalgorithmus innerhalb einer impliziten Kopplung ist in Algorithmus 4 angegeben.

4.1.3. Kopplungsbedingungen

Am gemeinsamen Rand Γ zwischen den Teilfeldern sind für den Erhalt des Gleichgewichts drei Bedingungen zu erfüllen. Dies sind die Kontinuität der Verschiebungen:

$$\mathbf{d}^{\Gamma^S} = \mathbf{d}^{\Gamma^F} \quad \text{auf } \Gamma, \quad (4.6)$$

die Kontinuität der Geschwindigkeiten:

$$\dot{\mathbf{d}}^{\Gamma^S} = \frac{\partial \mathbf{d}^{\Gamma^S}}{\partial t} = \mathbf{u}_G^{\Gamma^F} \quad \text{auf } \Gamma \quad (4.7)$$

und die Kontinuität der Oberflächenkräfte:

$$n^{\Gamma^S} \cdot \tau^{\Gamma^S} = n^{\Gamma^F} \cdot \tau^{\Gamma^F} \quad \text{auf } \Gamma, \quad (4.8)$$

mit n als Normalenvektor auf dem gemeinsamen Rand Γ . Damit ergibt sich die Position des Interfaces im Simulationsgebiet zu:

$$\mathbf{x}^\Gamma(t) = \mathbf{x}_0^\Gamma + \mathbf{d}^\Gamma(t). \quad (4.9)$$

4.1.4. Formulierung in Operatorschreibweise

Zur Lösung des gekoppelten Problems wird ein partitionierter Dirichlet-Neumann-Ansatz verwendet. Das bedeutet, dass Fluid- und Strukturfeld als separate Partitionen gelöst werden und die gegenseitige Wechselwirkung über die Beeinflussung entsprechender Randbedingungen am Interface abgebildet wird. Um eine einfache Formulierung des gekoppelten Problems zu erreichen, wird eine Operatorschreibweise benutzt. Die Anwendung eines Operators beinhaltet das Lösen des zugehörigen Teilfeldes unter Vorgabe einer entsprechenden Randbedingung bis zur Konvergenz.

Das Fluidfeld wird nach den unbekanntem Netzverschiebungen \mathbf{d}^F , den Geschwindigkeiten \mathbf{u} und Drücken p mit einer vorgeschriebenen Randverschiebung \mathbf{d}^Γ gelöst und ist damit die Dirichlet-Partition. Für die Lösung des Fluidfeldes wird der Algorithmus 4 verwendet. Die vom Fluid auf das Interface wirkenden Kräfte \mathbf{f}^Γ werden aus der Fluidlösung bestimmt. Der Operator wird mittels \mathcal{F} gekennzeichnet und seine Anwendung ergibt:

$$\mathbf{f}^\Gamma = \mathcal{F}(\mathbf{d}^\Gamma). \quad (4.10)$$

Das Strukturfeld wird nach den unbekanntem Verschiebungen \mathbf{d} mit einer vorgegebenen Randkraft \mathbf{f}^Γ gelöst und ist damit die Neumann-Partition. Für die Lösung des Strukturfeldes wird der Algorithmus 1 verwendet. Die relaxierten Interfaceverschiebungen $\tilde{\mathbf{d}}^\Gamma$ werden aus der Strukturlösung bestimmt. Der Operator wird mittels \mathcal{S} gekennzeichnet und seine Anwendung ergibt:

$$\tilde{\mathbf{d}}^\Gamma = \mathcal{S}(\mathbf{f}^\Gamma) = \mathcal{S}(\mathcal{F}(\mathbf{d}^\Gamma)) = \mathcal{S} \circ \mathcal{F}(\mathbf{d}^\Gamma). \quad (4.11)$$

Die Interfaceverschiebungen \mathbf{d}^Γ sind damit die Primärvariablen eines gekoppelten Problems vom Typ Fluid-Struktur-Wechselwirkung, da die Interfacekräfte \mathbf{f}^Γ implizit von diesen abhängen.

Das gekoppelte Problem ist konvergiert, wenn innerhalb eines Zyklus die auf die Dirichlet-Partition vorgegebenen Verschiebungen \mathbf{d}^Γ und die von der Neumann-Partition berechneten Verschiebungen $\tilde{\mathbf{d}}^\Gamma$ übereinstimmen. Das Residuum am Interface berechnet sich daher über:

$$\mathbf{R}^\Gamma(\mathbf{d}^\Gamma, \mathbf{f}^\Gamma) = \tilde{\mathbf{d}}^\Gamma - \mathbf{d}^\Gamma = \mathcal{S} \circ \mathcal{F}(\mathbf{d}^\Gamma) - \mathbf{d}^\Gamma. \quad (4.12)$$

Die Lösung des gekoppelten Problems ist gefunden, wenn die Residuen der zwei Teilfelder und das Residuum am Interface unter einer gewissen Schranke liegen, z.B.:

$$|\mathbf{R}^S(\mathbf{d})|_2 \leq \epsilon^S \quad , \quad |\mathbf{R}^F(\mathbf{d}^F, \mathbf{u}, p)|_2 \leq \epsilon^F \quad , \quad |\mathbf{R}^\Gamma(\mathbf{d}^\Gamma, \mathbf{f}^\Gamma)|_2 \leq \epsilon^\Gamma. \quad (4.13)$$

4.1.5. Einflussgrößen auf die Nichtlinearität des gekoppelten Problems

Um die Lösungsansätze für gekoppelte Probleme, welche in den folgenden Abschnitten vorgestellt werden, miteinander vergleichen zu können, werden in Kapitel 6 diverse numerische Beispiele verwendet. Dabei ist es wichtig, auch den Einfluss von Modellparametern

auf das Verhalten eines Lösungsansatzes zu untersuchen. Hierzu eignen sich vor allem Parameter, welche die Nichtlinearität und damit die Lösungsschwierigkeit des gekoppelten Problems beeinflussen.

Der Grad an Nichtlinearität eines Problems kann leider nicht exakt bestimmt werden. Während bei einem linearen Problem ein proportionaler Zusammenhang zwischen der Änderung einer Einflussgröße und einer Ergebnisgröße besteht, ist dies bei einem nicht-linearen Problem gerade nicht der Fall. Eine Möglichkeit zur Quantifizierung des Grades der Nichtlinearität besteht darin, die Abweichung zu einem linearem Problem zu untersuchen. Eine weitere Möglichkeit besteht darin, den Änderungsgrad einer Ergebnisgröße zwischen zwei Zeitschritten zu untersuchen.

Bei der numerischen Lösung von oberflächengekoppelten Problemen des Typs Fluid-Struktur-Wechselwirkung existiert eine große Anzahl an Einflussparametern auf den Grad der Nichtlinearität. Im Rahmen dieser Arbeit werden drei besonders exponierte Parameter im Detail untersucht. Diese sind:

1. **Die Zeitschrittweite:** Die Zeitschrittweite bestimmt den zeitlichen Abstand zwischen den diskreten Zeitpunkten, an welchen die Lösung des Problems berechnet wird. Bei einer sehr geringen Zeitschrittweite ist die Veränderung der Ergebnisgrößen zwischen den Zeitpunkten gering, während eine Erhöhung in der Regel zu einer größeren Änderung aller Werte führt. Eine größere Zeitschrittweite erhöht damit die Nichtlinearität des gekoppelten Problems.
2. **Die Verformungshöhe:** Die Höhe der auftretenden Strukturverformungen ist, ebenfalls wie die Zeitschrittweite, ein relatives Kriterium. Während bei kleinen Verformungen auch die Änderungen zwischen den Zeitschritten klein sind, treten bei großen Verformungen auch größere Änderungen zwischen den Zeitschritten auf. Damit erhöhen große Verformungen die Nichtlinearität des gekoppelten Problems.
3. **Die Dichteverhältnisse:** Wird ein Körper in einem Fluid beschleunigt, muss dabei auch ein Teil des umliegenden Fluids mitbewegt werden. Die dadurch zusätzlich auftretende Trägheit wird „Added Mass Effect“ genannt, da sie wie eine zusätzliche Masse auf den Körper wirkt. Die Höhe dieses Effektes ergibt sich aus den Dichteverhältnissen von Körper und Fluid. Ist die Fluiddichte vergleichsweise gering, ist auch die zusätzlich zu bewegende Masse gering. Übertragen auf Probleme der Fluid-Struktur-Wechselwirkung spricht man vom „Artificial Added Mass Effect“ ([FWR07, CGN05]), da hier das umgebende Fluid als zusätzliche Masse auf die Strukturfreiheitsgrade am Interface wirkt. Nähern sich die Dichten von Fluid und Struktur an, so zeigen sich bei expliziten Kopplungsschemen (siehe 4.2) Instabilitäten, bei impliziten Kopplungsschemen verschlechtert sich das Konvergenzverhalten.

Der Einfluss dieser drei Parameter auf das Verhalten der Lösungsansätze wird in den numerischen Beispielen in Kapitel 6 besonders untersucht.

4.2. Grundform eines impliziten Kopplungsalgorithmus

Prinzipiell ist bei der partitionierten Analyse von transienten Problemen zwischen expliziten und impliziten Kopplungsalgorithmen zu unterscheiden (siehe Abb. 4.2). Explizite Kopplungsalgorithmen tauschen die Informationen zwischen den Feldern über Verschiebungen und Kräfte pro Zeitschritt nur einmal aus und werden damit auch als einfach gestaffelte Verfahren bezeichnet (siehe z.B. [FL98, FL00, FPF01, Mok01, FZG06]). Dies ist zwar von geringem numerischem Aufwand, es ist aber in der Regel nicht möglich, die Kopplungsbedingungen (siehe 4.1.3) am gemeinsamen Rand zum betrachteten Zeitpunkt exakt zu erfüllen. Damit geben diese Verfahren nur die Möglichkeit, eine schwache algorithmische Kopplung abzubilden. Bei einer Kopplung elastischer Strukturen mit inkompressiblen Fluiden weisen diese Verfahren zusätzlich schlechte Stabilitätseigenschaften auf (siehe z.B. [ZH04, CGN05, FWR07, WKB07])

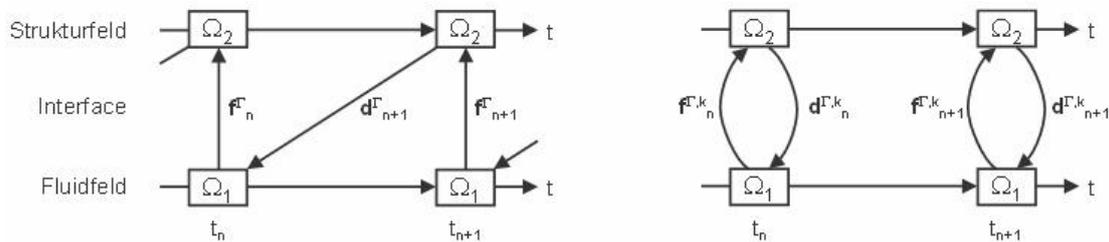


Abbildung 4.2.: Explizites (links) und implizites (rechts) Kopplungsschema

In dieser Arbeit werden deshalb implizite Kopplungsalgorithmen verwendet. Bei impliziten Kopplungsalgorithmen wird die Gleichgewichtsbedingung zum neuen Zeitpunkt t_{n+1} formuliert. Diese wird über ein iteratives Vorgehen erfüllt, indem Kopplungsiterationen zwischen den Teilfelder ausgeführt und Verschiebungen und Kräfte pro Zeitschritt mehrfach ausgetauscht werden (mehrfach gestaffelte Verfahren). Bei der hier verwendeten Dirichlet-Neumann-Kopplung wird dazu in jeder Kopplungsiteration als erstes die Dirichlet-Partition unter Vorgabe eines Verschiebungsfeldes gelöst und als zweites die Neumann-Partition unter Vorgabe des soeben berechneten Kraftfeldes. Die prinzipielle Form eines Dirichlet-Neumann-Kopplungsalgorithmus ist in Algorithmus 5 dargestellt.

Die Zeitschritte werden mit Iterationsindex n gekennzeichnet, die Kopplungsiterationen mit Iterationsindex k . Der neue Gleichgewichtszustand wird für den Zeitpunkt t_{n+1} gesucht unter der Vorgabe, dass am Zeitpunkt t_n ein Gleichgewichtszustand vorliegt, der die Bedingungen in 4.13 erfüllt. Damit lässt sich das Vorgehen in jedem Zeitschritt folgendermaßen beschreiben:

1. **Bestimmung der neuen Interfaceverschiebungen $\mathbf{d}_{n+1}^{\Gamma,k}$:**

In der ersten Kopplungsiteration ($k = 1$) Bestimmung über einen Prädiktor: $\mathbf{d}_{n+1}^{\Gamma,P}$ (siehe Abschnitt 4.4), für $k > 1$: Bestimmung über einen Kopplungsalgorithmus (siehe Abschnitt 4.5).

2. **Lösung Fluidfeld nach Algorithmus 4:**

Aufbringen der Interfaceverschiebung $\mathbf{d}_{n+1}^{\Gamma,k}$ auf den Fluidrand Γ^F . Anpassung des

Algorithmus 5 Grundform eines impliziten Dirichlet-Neumann Kopplungsalgorithmus

```

1: for  $t = 0$  bis  $t = t_{tot}$  do
2:    $k=1$ 
3:   while (keine Konvergenz) do
4:     If( $k=1$ ) Prädiktorschritt:  $\mathbf{d}_{n+1}^{\Gamma,1} = \mathbf{d}_{n+1}^{\Gamma,P}$ 
5:      $\tilde{\mathbf{d}}_{n+1}^{\Gamma,k} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,k})$ 
6:      $\mathbf{R}_{n+1}^{\Gamma,k} = \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k}$ 
7:     if  $\left( \left| \mathbf{R}_{n+1}^{\Gamma,k} \right|_2 < \epsilon^\Gamma \right)$  then
8:        $\mathbf{d}_{n+1}^\Gamma = \tilde{\mathbf{d}}_{n+1}^{\Gamma,k}$ 
9:       konvergiert, gehe zu nächstem Zeitschritt
10:    else
11:       $\Delta \mathbf{d}_{n+1}^{\Gamma,k} = f(\mathbf{R}_{n+1}^{\Gamma,k})$ 
12:       $\mathbf{d}_{n+1}^{\Gamma,k+1} = \mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k}$ 
13:    end if
14:     $k \leftarrow k + 1$ 
15:  end while
16:   $t_{n+1} = t_n + \Delta t_n$ 
17:   $n \leftarrow n + 1$ 
18: end for

```

Fluidgebietes an die neue Randgeometrie durch Berechnung der Netzverformung. Berechnung von \mathbf{u}_{n+1}^k und p_{n+1}^k mittels PISO-Algorithmus. Bestimmung der Oberflächenkräfte am Interface $\mathbf{f}_{n+1}^{\Gamma,k}$.

Abgekürzt in Operatorschreibweise: $\mathbf{f}_{n+1}^{\Gamma,k} = \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,k})$

3. Lösung Strukturfeld nach Algorithmus 1:

Aufbringen der Interfacekräfte $\mathbf{f}_{n+1}^{\Gamma,k}$ auf das Strukturfeld. Berechnung der Strukturverschiebung \mathbf{d}_{n+1}^k mittels Newton-Algorithmus. Bestimmung der relaxierten Verschiebung am Interface $\tilde{\mathbf{d}}_{n+1}^{\Gamma,k}$.

Abgekürzt in Operatorschreibweise: $\tilde{\mathbf{d}}_{n+1}^{\Gamma,k} = \mathcal{S}(\mathbf{f}_{n+1}^{\Gamma,k}) = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,k})$

4. Konvergenzüberprüfung:

Berechnung des Residuums des gekoppelten Problems über $\mathbf{R}_{n+1}^{\Gamma,k} = \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k}$ und Überprüfung der Konvergenz nach Abschnitt 4.3, z.B. mittels L2-Norm: $\left| \mathbf{R}_{n+1}^{\Gamma,k} \right|_2 < \epsilon^\Gamma$.

Konvergenz: Gleichgewichtszustand erreicht, gehe zu nächstem Zeitschritt.

Keine Konvergenz: Gehe zu Punkt 1 und beginne nächste Kopplungsiteration.

Anmerkung 1: Zu beachten ist, dass in jeder Kopplungsiteration die Teilfelder immer bis zur Konvergenz gelöst werden. Daraus resultiert auch der hohe numerische Aufwand bei

der impliziten Lösung von gekoppelten Problemen. Werden pro Zeitschritt beispielsweise 5 Kopplungsiterationen benötigt, ist der numerische Aufwand stark vereinfacht in etwa das Fünffache der separaten Einzelfeldlösungen.

Anmerkung 2: Die für die Effizienz besonders wichtigen Bereiche sind aus Algorithmus 5 leicht ersichtlich. Diese sind:

1. Der Interfaceprädiktor: In der ersten Kopplungsiteration wird über einen Prädiktor ein Startfeld für die Interfaceverschiebung $\mathbf{d}_{n+1}^{\Gamma,1}$ bestimmt. Je näher dieses Startfeld an der endgültigen Lösung ist, desto geringer ist das Residuum nach dem ersten Kopplungszyklus und desto weniger Iterationen werden bis zur Konvergenz benötigt. Möglichkeiten zur Bestimmung der Initialposition werden in Abschnitt 4.4 besprochen, deren Auswirkung innerhalb eines numerischen Beispiels wird in Abschnitt 6.1.4 ermittelt.
2. Der Kopplungsalgorithmus: Während der Kopplungsiterationen wird durch den Kopplungsalgorithmus die neue Interfaceverschiebung $\mathbf{d}_{n+1}^{\Gamma,k+1}$ bestimmt. Je höher die Konvergenzrate des Algorithmus, desto weniger Iterationen werden benötigt. Die unterschiedlichen Kopplungsalgorithmen werden in Abschnitt 4.5 hergeleitet und deren Anwendung innerhalb eines numerischen Beispiels wird in Abschnitt 6.1.5 gezeigt.
3. Die Teilfeldlöser: In jeder Kopplungsiteration werden beide Teilfelder bis zur Feldkonvergenz gelöst. Durch die Nichtlinearität der Feldprobleme wird jeweils ein iteratives Vorgehen genutzt, welches ebenfalls entsprechende Initialwerte für die unbekanntes Feldvariablen benötigt. Diese Initialwerte können aus den Werten des letzten Zeitschrittes bestimmt werden und sind damit während der Kopplungsiterationen konstant („konstante Feldlöser“). Oder sie werden aus den Werten der letzten Kopplungsiteration bestimmt und sind damit angepasst („angepasster Feldlöser“). Unter der Annahme einer Konvergenz des gekoppelten Problems benötigen angepasste Feldlöser weniger Teilfelditerationen pro Kopplungsiteration. Die Umsetzung dieses Vorgehens wurde für das Strukturfeld in Abschnitt 2.5 und für das Fluidfeld in Abschnitt 3.6 gezeigt, die numerischen Auswirkungen werden in Abschnitt 6.1.3 untersucht.

4.3. Konvergenzkriterien

Das gekoppelte Problem ist konvergiert, wenn sowohl die Residuen der zwei Teilfelder als auch das Interfaceresiduum eine gewisse Konvergenzbedingung erfüllen (siehe Gleichung 4.13). Die Konvergenz der Teilfelder ist in jeder Kopplungsiteration garantiert, da diese jeweils vollständig gelöst werden. Der Vektor des Interfaceresiduums bestimmt sich pro Zeitschritt und Kopplungsiteration über:

$$\mathbf{R}_{n+1}^{\Gamma,k} = \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k}. \quad (4.14)$$

Für eine Konvergenzkontrolle wird dieser Vektor in eine skalare Größe transformiert, welche dann mit einer gewissen skalaren Toleranzschranke ϵ^Γ verglichen werden kann. Für die Bestimmung der skalaren Größe kann in absolute und relative Ansätze unterschieden werden. Absolute Ansätze betrachten nur den absoluten Wert des aktuellen Residuums und sind damit dimensionsabhängig. Relative Ansätze hingegen sind nicht dimensionsabhängig, da sie die relative Änderungsrate des Residuums bewerten.

Nach dem Vorschlag von [KW08, KW09] ergibt sich ein absolutes Kriterium über:

$$\frac{1}{\sqrt{n_{dof}^\Gamma}} \left| \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k} \right|_2 < \epsilon^\Gamma. \quad (4.15)$$

Dabei wird die mittels der L2-Norm bestimmte Länge des Residuenvektors noch mit der Anzahl der Freiheitsgrade am Interface skaliert, um eine gemittelte Änderung pro Freiheitsgrad zu erhalten. Der Nachteil eines absoluten Kriteriums ist die Abhängigkeit von der Dimension der Verschiebungen, womit problemabhängige Toleranzschranken gewählt werden müssen.

Relative Kriterien sind dimensionsunabhängig, indem ein absoluter Wert durch einen Referenzwert geteilt und damit skaliert und relativiert wird. Zur Bestimmung des Skalierungsfaktors kann z.B. die L2-Norm auf den Vergleich zwischen aktuellen Verschiebungen und Verschiebungen des letzten Zeitschritts angewendet werden:

$$\frac{\left| \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k} \right|_2}{\left| \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_n^\Gamma \right|_2} < \epsilon^\Gamma, \quad (4.16)$$

oder mit der maximalen Differenz skaliert werden:

$$\frac{\left| \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k} \right|_2}{\left| \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_n^\Gamma \right|_\infty} < \epsilon^\Gamma. \quad (4.17)$$

In dieser Arbeit wird das Konvergenzkriterium nach Gleichung 4.16 verwendet. Als relatives Kriterium kann es auf alle Arten von Problemen angewendet werden.

Ein weitere Möglichkeit, die Konvergenz zu überprüfen, ist, wie in [Kup09] vorgeschlagen, die am Interface transferierten Kräfte zusätzlich auf Konvergenz zu überprüfen. Damit wird sowohl die Primär- als auch die Sekundärvariable auf Konvergenz überprüft. Da

die Kräfte aber direkt von den Verschiebungen abhängen, sind bei einer Konvergenz der Verschiebungen auch die Kräfte konvergiert. Damit ist eine doppelte Überprüfung nicht nötig. Der Einfluss unterschiedlicher Konvergenzraten kann relativiert werden, indem die Toleranzschranke des überprüften Wertes entsprechend angepasst wird.

4.4. Interfaceprädiktor

Wie bei jeder Lösung eines nichtlinearen Problems, sind auch bei impliziten Kopplungsalgorithmen Startwerte für die zu behandelnden Variablen notwendig. Der Interfaceprädiktor dient dazu, in der ersten Kopplungsiteration pro Zeitschritt einen Initialwert $\mathbf{d}_{n+1}^{\Gamma,1} = \mathbf{d}_{n+1}^{\Gamma,P}$ für den Verschiebungsvektor zu bestimmen. Je näher der durch den Prädiktor bestimmte Initialwert an der tatsächlichen Lösung des Problems liegt, desto schneller konvergiert das gekoppelte Problem. Des Weiteren ist zu beachten, dass stark gekoppelte Probleme durch ihre hochgradige Nichtlinearität nur lokale Konvergenzeigenschaften aufweisen. Es tritt also Divergenz auf, falls der Startwert zu weit von der Lösung entfernt ist. Für ein effizientes und stabiles Kopplungsverfahren ist deshalb die Verwendung eines Prädiktors, der eine möglichst gute Abschätzung für das Interfaceverschiebungsfeld im neuen Zeitschritt vornimmt, von großer Bedeutung.

Die innerhalb dieser Arbeit verwendeten Prädiktoren basieren auf der Annahme, dass sich aus bekannten Lösungen an vorherigen Zeitschritten eine Näherung zum nächsten Zeitschritt extrapolieren lässt. Die Ansätze lassen sich in drei Bereiche gruppieren: Polynombasierte Prädiktoren (4.4.1), Newmarkbasierte Prädiktoren (4.4.2) und adaptive Prädiktoren (4.4.3). Sie werden in den drei folgenden Abschnitten besprochen. Deren Einfluss auf das Lösungsverhalten wird innerhalb eines numerischen Beispiels in Abschnitt 6.1.4 näher untersucht. Für den Vergleich werden zwei Kriterien herangezogen:

1. Als absolutes Kriterium die Anzahl an benötigten Kopplungsiterationen bei Verwendung eines fixen Kopplungsalgorithmus.
2. Die Höhe des Interfaceresiduums nach dem ersten Austauschzyklus, welcher auf der vom Prädiktor bestimmten Startposition basiert.

4.4.1. Polynombasierte Prädiktoren

Polynombasierte Prädiktoren beruhen auf der Annahme, dass sich der zeitliche Verschiebungsverlauf eines Knotens, der ja nur an diskreten Zeitpunkten bekannt ist, über ein Polynom näherungsweise beschreiben lässt. Unter Annahme einer konstanten Zeitschrittweite sind die Koeffizienten des Polynoms bekannt und es lässt sich eine Lösung für den nächsten Zeitschritt extrapolieren. Für ein Polynom der Ordnung $\mathbf{O} = 0$ ergibt sich damit die Lösung zum nächsten Zeitschritt zu:

$$\mathbf{d}_{n+1}^{\Gamma,P} = \mathbf{d}_n^{\Gamma}. \quad (4.18)$$

Dies ist auch als konstanter Prädiktor bekannt, da einfach das Ergebniss des letzten Zeitschritts als Initialwert benutzt wird. Für Polynome der Ordnung $\mathbf{O} = 1$ ergibt sich:

$$\mathbf{d}_{n+1}^{\Gamma,P} = 2\mathbf{d}_n^{\Gamma} - \mathbf{d}_{n-1}^{\Gamma}. \quad (4.19)$$

Für Polynome der Ordnung $\mathbf{O} = 2$ ergibt sich:

$$\mathbf{d}_{n+1}^{\Gamma,P} = 3\mathbf{d}_n^{\Gamma} - 3\mathbf{d}_{n-1}^{\Gamma} + \mathbf{d}_{n-2}^{\Gamma}. \quad (4.20)$$

Und für Polynome der Ordnung $\mathbf{O} = 3$ ergibt sich:

$$\mathbf{d}_{n+1}^{\Gamma,P} = 4 \mathbf{d}_n^\Gamma - 6 \mathbf{d}_{n-1}^\Gamma - \frac{20}{3} \mathbf{d}_{n-2}^\Gamma - 1 \mathbf{d}_{n-3}^\Gamma. \quad (4.21)$$

Polynombasierte Prädiktoren sind einfach zu erstellen und zu implementieren. Je nach gewähltem Polynomgrad ergibt sich die Anzahl an Interfaceverschiebungsvektoren, die vorgehalten werden müssen.

4.4.2. Newmarkbasierte Prädiktoren

Newmarkbasierte Prädiktoren ziehen erweiterte physikalische Eigenschaften des betrachteten Problems hinzu. Das Interface als Berandung des Strukturfeldes weist zusätzlich zur Verschiebung noch eine Geschwindigkeit und eine Beschleunigung auf. Diese können mittels der auch im Strukturfeld benutzten Ansätze bestimmt werden. Bei Verwendung des Generalized- α Zeitintegrationsverfahrens werden die Newmark-Ansätze zur Bestimmung von Geschwindigkeit und Beschleunigung benutzt (siehe Abschnitt 2.3.1). Damit ergibt sich für einen Prädiktor der Ordnung $\mathbf{O} = 1$ ([Pip97]):

$$\mathbf{d}_{n+1}^{\Gamma,P} = \mathbf{d}_n^\Gamma + \Delta t \dot{\mathbf{d}}_n^\Gamma. \quad (4.22)$$

Und für einen Prädiktor der Ordnung $\mathbf{O} = 2$ ([PF01]):

$$\mathbf{d}_{n+1}^{\Gamma,P} = \mathbf{d}_n^\Gamma + \Delta t \left(\frac{3}{2} \dot{\mathbf{d}}_n^\Gamma - \frac{1}{2} \dot{\mathbf{d}}_{n-1}^\Gamma \right). \quad (4.23)$$

Für diese Art von Prädiktor muss neben der Verschiebung zusätzlich die Geschwindigkeit am Interface mit den zum Strukturfeld kompatiblen Ansätzen bestimmt werden. Damit erhält man ein Verfahren, welches über die Geschwindigkeiten auch höherwertige Informationen als nur das Verschiebungsfeld nutzt.

4.4.3. Adaptive Prädiktoren

Die bisher vorgestellten Prädiktoren haben gemeinsam, dass deren Annahmen über den Zustand zum nächsten Zeitschritt zum Beginn des Verfahrens gesetzt werden und über den gesamten betrachteten Zeitraum als konstant angenommen werden. Die Prädiktoren sind also nicht in der Lage, eine zeitliche Veränderung des Systemverhaltens abzubilden. Um diese zeitliche Veränderlichkeit abbilden zu können, wird in dieser Arbeit ein selbstentwickelter adaptiver Prädiktor vorgestellt. Dabei werden in jedem Zeitschritt die einzelnen Anteile des Prädiktors mit Korrekturkoeffizienten skaliert, um diese gewichten zu können. Als Basis kann prinzipiell jedes der bisher vorgestellten Verfahren verwendet werden, hier wird ein Prädiktor basierend auf den Newmark-Ansätzen gewählt:

$$\mathbf{d}_{n+1}^{\Gamma,P} = C_1 \cdot \mathbf{d}_n^\Gamma + C_2 \cdot \Delta t \dot{\mathbf{d}}_n^\Gamma + C_3 \cdot \frac{\Delta t^2}{2} (1 - 2\beta) \ddot{\mathbf{d}}_n^\Gamma \quad (4.24)$$

Um die Korrekturkoeffizienten C_1 bis C_3 bestimmen zu können, wird angenommen, dass sich deren Größe über die Zeit zwar ändert, aber nur minimal von Zeitschritt zu

Zeitschritt. Damit kann ein Minimierungsproblem gelöst werden, indem obiger Ansatz auf den vorherigen Zeitschritt angewendet wird, dessen Ergebniss ja bekannt ist:

$$C_i = \arg \min \left| C_1 \cdot \mathbf{d}_{n-1}^\Gamma + C_2 \cdot \Delta t \dot{\mathbf{d}}_{n-1}^\Gamma + C_3 \cdot \frac{\Delta t^2}{2} (1 - 2\beta) \ddot{\mathbf{d}}_{n-1}^\Gamma - \mathbf{d}_n^\Gamma \right|_2. \quad (4.25)$$

Dieser Prädiktor benötigt zum Start die angegebenen Variablen von vorhergehenden Zeitschritten am Interface. Solange diese noch nicht zur Verfügung stehen, müssen die Korrekturkoeffizienten mit 1, 0 als Initialwerte angenommen werden. Außerdem ist es bei jeder Verwendung notwendig, ein Gleichungssystem zu lösen. Dieses besitzt allerdings als Unbekannte nur die drei Korrekturkoeffizienten, was die Lösung natürlich wenig aufwendig macht.

Anmerkung 1: Eine weitere Gruppe von Prädiktoren stellen die von Scheven ([STRB07, Sch09]) eingeführten „Grobgitterprädiktoren“ dar. Hierbei wird in Anlehnung an Mehrgitterverfahren zur Lösung linearer Gleichungen das gekoppelte Problem auf einer größeren Gitterebene gelöst, diese Lösung auf das feinere Gitter übertragen und dort als Initiallösung genutzt. Dies stellt einen sehr interessanten Ansatz dar, welcher unter Umständen einen qualitativ hochwertigen Prädiktor produziert. Als Nachteile sind allerdings der hohe Speicher- und Programmieraufwand zu nennen. Zum einen muss zusätzlich die Diskretisierung der größeren Gitterebene im Speicher vorgehalten werden, zum anderen muss in den Feldlösern noch eine zusätzliche Kopplungsebene eingeführt werden, was i.d.R. große Veränderungen an den Programmcodes erfordert. Diese Prädiktormethode wird innerhalb dieser Arbeit nicht weiter untersucht.

Anmerkung 2: In vielen Arbeiten wird auf die Art des verwendeten Prädiktors nur sehr am Rande (z.B. [Kup09, Mic05, VLDV07, Kol10]) und teilweise gar nicht eingegangen ([Kue09]). In der Regel wird ein bestimmter Prädiktor ohne nähere Erläuterung verwendet. Da die Bestimmung einer adäquaten Startposition aber auf die Effizienz des gesamten gekoppelten Verfahrens gerade bei starker Kopplung einen hohen Einfluss hat, wird in dieser Arbeit der Einfluss der Prädiktormethode fundiert untersucht. Dazu ist es notwendig, das Verhalten der unterschiedlichen Ansätze in verschiedenen Szenarien zu testen. Dies geschieht im numerischen Beispiel in Kapitel 6 in Abschnitt 6.1.4.

4.5. Verfahrensüberblick zu Kopplungsalgorithmen

Als Kopplungsalgorithmus wird in dieser Arbeit das Verfahren bezeichnet, mit welchem während der Kopplungsiterationen das neue Interfaceverschiebungsinkrement $\Delta \mathbf{d}_{n+1}^{\Gamma,k}$ bzw. die neue Interfaceverschiebung $\mathbf{d}_{n+1}^{\Gamma,k+1} = \mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k}$ bestimmt wird. Der Kopplungsalgorithmus ist der maßgebliche Einflussparameter auf Stabilität und Effizienz der gesamten gekoppelten Simulation. Algorithmen zur Lösung des gekoppelten Problems sind seit vielen Jahren ein aktiver Forschungsbereich, woraus eine große Anzahl an verschiedensten Ansätzen resultiert.

Das Ziel dieses und der folgenden Abschnitte ist es, für den Fall stark gekoppelter und transienter Probleme der Fluid-Struktur-Wechselwirkung einen Überblick über die im wissenschaftlichen Bereich derzeit verwendeten Verfahren zu geben (siehe dieser Abschnitt), einige davon nach bestimmten Kriterien auszuwählen und diese gesondert zu betrachten (siehe Abschnitte 4.6 bis 4.8). Diese Verfahren werden innerhalb eines numerischen Beispiels in Bezug auf Effizienz und Stabilität miteinander verglichen (siehe Abschnitt 6.1.5).

Durch die rege Forschungstätigkeit in diesem Bereich entstand im Verlauf der letzten Jahre eine große Anzahl an unterschiedlichsten Lösungsansätzen. Um diese Anzahl ein wenig einzuschränken, werden in dieser Arbeit nur Verfahren betrachtet, welche als Grundlage einen partitionierten und impliziten Ansatz wählen. Eine mögliche Gruppierung der Verfahren findet typischerweise statt, indem strikt partitionierte und monolithische Ansätze als Extreme definiert werden und die Verfahren zwischen diesen Extremen einsortiert werden. Solche Übersichten finden sich beispielsweise in [Mok01, OF01, OK06, BS06, Kue09, Kup09, Sch09] und in der darin angegebenen Literatur.

Im Folgenden wird ein Ansatz gewählt, welcher eine Gruppierung basierend auf den Anforderungen des Kopplungsalgorithmus an die verwendeten Feldlöser vornimmt. Ein großer, wenn nicht sogar der größte Vorteil eines partitionierten Ansatzes ist die Wiederverwendbarkeit von existierenden Teilfeldlösern. Dies ist der Fall bei einem Kopplungsalgorithmus, der die Verwendung von Standardlösungsansätzen in den Teilfeldlösern ermöglicht. Ein Kopplungsalgorithmus hingegen, welcher spezifische Anpassungen und Erweiterungen der verwendeten Teilfeldlöser benötigt, vermindert diesen Vorteil und stellt die Wahl eines partitionierten Vorgehens zumindest teilweise in Frage. Dieser Nachteil kann aber durch besonders hohe Effizienz und Stabilität wieder ausgeglichen werden.

Ein graphischer Überblick ist in Abbildung 4.3 gegeben. Die hier aufgeführten Ansätze erfüllen zwar nicht den Anspruch an Vollständigkeit, geben aber einen Großteil der aktuell im wissenschaftlichen Bereich diskutierten Ansätze wieder. Die einzelnen Methoden werden im Folgenden kurz angesprochen, um darauf basierend eine Auswahl im Detail in den folgenden Abschnitten vorzustellen.

Zu den Ansätzen, welche die Verwendung von Standardlösungsansätzen in den Teilfeldlösern ermöglichen, zählen die Fixpunktiteration, Vektorextrapolation und eine besondere Form einer Quasi-Newton-Methode. Die Methoden im Einzelnen:

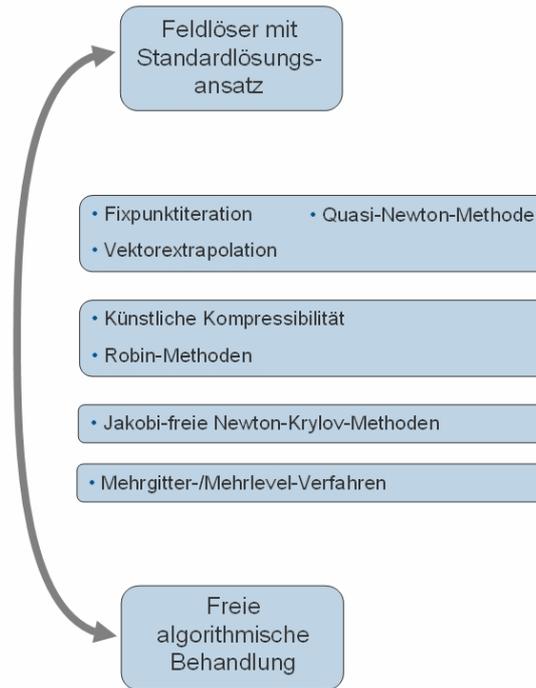


Abbildung 4.3.: Überblick über partitionierte und implizite Kopplungsalgorithmen

- **Fixpunktiteration:** Die Fixpunktiteration ist ein Überbegriff für Verfahren, welche eine Iterationsvorschrift konstruieren und damit aus einer alten eine neue Lösung bestimmen. Die Iterationsvorschrift wird so lange angewandt, bis die Lösung der vorhergehenden Iteration auf sich selbst abgebildet wird. Kern des Verfahrens ist die Iterationsvorschrift, für welche eine ganze Bandbreite an Methoden existiert. In der Fluid-Struktur-Wechselwirkung wird über eine Relaxation das neue Verschiebungsincrement bestimmt. Der Relaxationsparameter kann hierbei konstant oder dynamisch angenommen werden. Die Aitken-Methode, die in vielen wissenschaftlichen Arbeiten Verwendung findet (z.B. [Kup09, Sch09, Kue09]), ist eine Fixpunktiteration mit dynamischer Relaxation.
- **Vektorextrapolation:** Diese Methode basiert auf einer konvergierenden Reihe an Vektoren, die mittels Extrapolation in eine neue Reihe überführt wird, welche schneller als die vorherige konvergiert. Dies ist ein in der numerischen Mathematik gebräuchliches Verfahren zur Konvergenzbeschleunigung (z.B. [JS00, Sid03]), welches in [KW09] auf Probleme der Fluid-Struktur-Wechselwirkung angewendet wurde.
- **Quasi-Newton-Methode nach [Vie06]:** Diese Methode zählt zu den Quasi-Newton-Verfahren, wobei die Inverse der Jakobimatrix über eine Linearkombination angenähert wird. Als Grundlage für die Näherung sind aber nur Informationen notwendig, welche durch Standardfeldlöser bestimmt werden können. Die-

ser Ansatz wurde speziell für Probleme der Fluid-Struktur-Wechselwirkung entwickelt und wird in mehreren aktuellen Arbeiten mit großem Erfolg verwendet (z.B. [DBV09, DAV10, GB10]).

Diese drei Verfahren werden alle in den folgenden Abschnitten näher untersucht (siehe Abschnitte 4.6, 4.7 und 4.8.2), da sie weit verbreitet sind und die grundlegende Idee des partitionierten Ansatzes erhalten.

Zu den Methoden, welche geringe Modifikationen der Teilfeldlöser benötigen, zählen die künstliche Kompressibilität und Robin-Methoden:

- **Künstliche Kompressibilität:** Bei vielen Anwendungen der Fluid-Struktur-Wechselwirkung wird das Fluidfeld als inkompressibel angenommen, da die auftretenden Fluidgeschwindigkeiten kleiner als das 0,3-fache der Mach-Zahl des entsprechenden Mediums sind ([Sch99]). Bei gekoppelten Problemen jedoch, insbesondere bei nahezu geschlossenen Untersuchungsgebieten, ergeben sich dadurch Druckoszillationen, welche die Konvergenz des Problems negativ beeinflussen bzw. sogar zur Divergenz führen können.

Um dieses Problem zu verringern, wird das Fluidfeld künstlich kompressibel gerechnet und die Kompressibilität in Abhängigkeit der Kopplungskonvergenz verringert. Dies wird erreicht, indem der kompressible Anteil in Abhängigkeit der Druckveränderung formuliert und die Kontinuitätsgleichung entsprechend angepasst wird:

$$\frac{c}{\Delta t} \left(p_{n+1}^k - p_{n+1}^{k-1} \right) + \nabla \cdot \mathbf{u}_{n+1}^k = 0. \quad (4.26)$$

Bei Erreichen der Konvergenz ist der inkompressible Zustand wieder hergestellt, da dann gilt: $p_{n+1}^{k-1} \rightarrow p_{n+1}^k$. Dieses Verfahren wird z.B. von [RVD00, RRLJ01] verwendet. Es kann aber nur in Kombination mit einem weiteren Kopplungsalgorithmus angewendet werden, wie z.B. einer Methode aus der vorherigen Gruppe, die Standardlösungsansätze in den Feldlösern verwendet. Damit ist die Methode der künstlichen Kompressibilität streng genommen auch kein eigenständiger Kopplungsalgorithmus, sondern nur die Kombination aus künstlicher Kompressibilität und z.B. einer relaxationsbasierten Fixpunktiteration. Des Weiteren wird dieser Ansatz zur Lösung reiner Fluidprobleme verwendet, um die Konvergenzeigenschaften zu verbessern ([RKK87, CB91, Cho97]).

- **Robin-Methoden:** Der Dirichlet-Neumann Ansatz basiert auf der Vorgabe von Randverschiebungen auf das Dirichlet-Feld und Randkräften auf das Neumann-Feld. Im Gegensatz dazu werden bei Robin-Methoden auf eines (oder beide) der Felder Robin-Randbedingungen vorgeschrieben. Robin-Randbedingungen sind eine Linearkombination von Dirichlet- und Neumannbedingungen. Bei Anwendung der Robin-Bedingung auf beide Felder (Robin-Robin Ansatz) werden also auf Struktur- und Fluidfeld sowohl Verschiebungen als auch Kräfte vorgegeben. Ein wichtiger Punkt ist hierbei die Bestimmung der Linearkoeffizien-

ten, wobei diese in der Regel anhand vereinfachter Feldmodelle hergeleitet werden (siehe z.B. [BNV08a, BNV08b]). Der Vorteil von Robin-Methoden ist unter gewissen Umständen eine Unabhängigkeit vom “Artificial Added Mass Effect” ([FWR06, FWR07, LTM01, Nob01, CGN05]), wodurch diese Algorithmen für eine große Bandbreite an Dichteverhältnissen ein identisches Konvergenzverhalten zeigen. Als Nachteil dieses Ansatzes ist zu nennen, dass Robin-Randbedingungen in regulären Feldlösern nicht vorhanden sind, was deren aufwendige Anpassung notwendig macht. Robin-Methoden werden z.B. in den Arbeiten [Nob01, CGN05, FVCJ⁺06, BNV08a, BNV08b, NV08, BNV09] verwendet.

Diese beiden Verfahren werden im Folgenden nicht weiter untersucht.

Eine weitere große Verfahrensgruppe sind die **Jakobi-freien Newton-Krylov-Verfahren**. Klassische Newton-Methoden benötigen zur Lösung die Jakobi-Matrix des betrachteten Problems. Dazu sind Ableitungen der Teilfeldlöser sowohl nach den eigenen Unbekannten als auch nach den Interface-Unbekannten notwendig. Die exakte Bestimmung der Ableitungen ist bisher nur für einen sehr spezifischen Fall gelungen (siehe [FM03, FM05]) und scheint als allgemeine Methode nicht möglich zu sein. Eine Alternative hierzu stellen die Jakobi-freien Verfahren dar. Bei diesen wird die Jakobi-Matrix nicht exakt, sondern nur in genäherter Form benötigt. Die Näherung wird hierbei durch verschiedenste Ansätze erzeugt.

Ein Überblick über allgemeine Ansätze zur Jakobi-freien Lösung nichtlinearer Probleme findet sich z.B. in [KK04] und deren Anwendung auf Fluidprobleme z.B. in [KM00, PT02]. Eine Übertragung dieser Methoden auf Probleme der Fluid-Struktur-Wechselwirkung ist nicht ohne Weiteres möglich. Der grundlegende Unterschied besteht in der Größe und der Verfügbarkeit von beschreibenden Matrizen. Bei Einzelfeldproblemen skaliert der Aufwand zur Erstellung z.B. der Steifigkeitsmatrix linear, während der Aufwand zum Lösen der linearen Gleichungssysteme quadratisch skaliert. Deshalb wird versucht, möglichst wenig Gleichungssysteme lösen zu müssen, aber einen hohen Aufwand bei der Erstellung zu betreiben. Bei Problemen der Fluid-Struktur-Wechselwirkung hingegen ist die Anzahl an Freiheitsgraden am Interface in der Regel deutlich geringer als in den Teilfeldproblemen. Damit ist auch der Aufwand zur Lösung des Gleichungssystems deutlich geringer. Der Aufwand zur Erstellung von Vektoren oder Matrizen ist jedoch sehr hoch, da hierzu im partitionierten Kontext immer die Teilfeldlöser gerufen werden müssen. Deshalb ist bei der Anwendung von Jakobi-freien Newton-Krylov Methoden auf Probleme der Fluid-Struktur-Wechselwirkung zu beachten, dass die Teilfeldlöser möglichst wenig gerufen werden müssen.

Eine Möglichkeit besteht in der Verwendung Newton-basierter Teilfeldlöser und der vereinfachten Behandlung der Jakobi-Ableitungen in Bezug auf die Interface-Freiheitsgrade. Dieser Ansatz wird z.B. in [Men98, MMV99, Men99, Ste02, MS02, MS03, MNS06, BBM06] verwendet und oft als Block-Newton-Methode bezeichnet. Dies stellt in der Regel eine Kombination aus drei Newton-Lösungsalgorithmen dar.

Ein weitere Möglichkeit besteht darin, nur am Interface ein Newton-basiertes Schema zu verwenden. Zur Lösung des Gleichungssystems pro Newton-Schritt werden Krylov-

Löser verwendet, da diese die Jakobi-Matrix nicht explizit benötigen. Für die Evaluation des Krylovraumes müssen die Teilfeldlöser aber in der Regel erweitert werden, was keine Wiederverwendung von Standardlösungsalgorithmen ermöglicht. Diese Verfahren werden z.B. in [GV03, GVF05, Dep04, DGV04, BMB05, Mic05] verwendet. Für die Evaluation des Krylovraumes können hierbei auch Nicht-Newton-Teilfeldlöser verwendet werden.

In dieser Arbeit wird im weiteren Verlauf nur die Interface-Newton-Methode weiter untersucht (4.8.1), da in dem verwendeten Fluidfeldlöser kein Newton-Schema, sondern ein Druckkorrekturverfahren zum Einsatz kommt.

Die **Mehrgitter- oder Mehrlevel-Verfahren** sind aus einer Verfahrensgruppe, welche erst vor Kurzem entstanden ist. Hierbei wird die Idee der Mehrgitter-Löser aufgegriffen und auf partitionierte Probleme der Fluid-Struktur-Wechselwirkung übertragen. Dies bedeutet, dass auf einem oder beiden der Felder mehrere Gitterebenen vorliegen und das gekoppelte Problem auf verschiedenen Ebenen formuliert und gelöst wird. Eine Möglichkeit besteht z.B. darin, das Problem auf einem groben Gitter bis zur Konvergenz zu lösen und diese Lösung als Initialwert für die Lösung auf dem feinen Gitter zu verwenden (siehe [STRB07, Sch09, RSFW09]). Eine weitere Möglichkeit besteht darin, die W- oder V-Zyklen von Mehrgitterlösern nachzubilden und das Problem alternierend auf verschiedenen Ebenen zu lösen ([ZBB07, ZB10]). Diese Verfahren müssen noch mit einem weiteren Kopplungsalgorithmus, wie z.B. der Aitken-Methode kombiniert werden. Eine wichtige Frage ist hierbei, wie die einzelnen Gitterebenen erzeugt und die Informationen zwischen diesen transformiert werden. In [ZBB07] werden z.B. die Methoden eines geometrischen Mehrgitter-Lösers verwendet. Die Implementierung in existierende Programme ist relativ aufwendig, da mehrere Gitter- und Problem-Ebenen erstellt, vorgehalten und gleichzeitig behandelt werden müssen.

Diese Verfahren werden im Folgenden nicht mehr weiter untersucht.

Der hier gegebene Überblick zu partitionierten Lösungsansätzen ist keineswegs vollständig. Auch ist dies immer noch ein aktives Forschungsgebiet, wodurch noch eine ganze Reihe an weiteren Verfahren existieren und entwickelt werden. Der Überblick zeigt aber die Bandbreite und vor allem die große Unterschiedlichkeit der Ansätze.

In den folgenden Abschnitten werden einige dieser Verfahren im Detail dargestellt. Dazu zählen alle Methoden, die Standardlösungsalgorithmen in den Teilfeldlösern verwenden: Die relaxationsbasierte Fixpunktiteration (4.6), die Vektorextrapolation (4.7) und die Quasi-Newton-Methode (4.8.2). Als Vertreter für Verfahren, welche eine Anpassung der Teilfeldlöser benötigen, werden diverse Newton-Krylov-Verfahren untersucht (4.8.1).

4.6. Fixpunktiteration mit Relaxation

Die allgemeine Form jedes Fixpunktverfahrens ist gegeben durch die Iterationsvorschrift ([DR06]):

$$x^{k+1} = \Phi(x^k), \quad k = 1, \dots, n, \quad (4.27)$$

mit der in jeder Iteration k die neue Lösung x^{k+1} aus der alten Lösung x^k mittels der Iterationsvorschrift Φ bestimmt wird. Die Iteration wird abgebrochen, sobald die exakte Lösung x^* erreicht ist, die, unter der Voraussetzung einer konvergenten Folge, die Bedingung:

$$x^* = \Phi(x^*) \quad (4.28)$$

erfüllt.

Bezogen auf ein oberflächengekoppeltes Problem mit den unbekanntem Interfacefreiheitsgraden \mathbf{d}_{n+1}^Γ in Zeitschritt $n + 1$ und Relaxation als Iterationsvorschrift ergibt sich Φ zu :

$$\Phi(\mathbf{d}_{n+1}^{\Gamma,k}) = \mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k} \quad (4.29)$$

und damit die Grundgleichungen der Fixpunktiteration zu:

$$\begin{aligned} \mathbf{d}_{n+1}^{\Gamma,k+1} &= \mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k} \\ &= \mathbf{d}_{n+1}^{\Gamma,k} + \omega^k \mathbf{R}_{n+1}^{\Gamma,k} \\ &= \mathbf{d}_{n+1}^{\Gamma,k} + \omega^k (\tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k}) \\ &= (1 - \omega^k) \mathbf{d}_{n+1}^{\Gamma,k} + \omega^k \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} \end{aligned}$$

mit k als Kenner der Kopplungsiteration, n als Kenner des Zeitschritts, ω als Relaxationsparameter und \mathbf{R} als das Interfaceresiduum.

Zur Bestimmung des Relaxationsparameters ω^k existieren u.a. die Möglichkeiten:

- **Konstanter Relaxationsparameter ω :** Dieser wird über alle Kopplungsiterationen und Zeitschritte als konstant angenommen, was eine sehr einfache Methode darstellt. Allerdings muss der Relaxationsparameter so gewählt werden, dass das gekoppelte Problem in jedem Zeitschritt konvergiert. Da sich der Kopplungsgrad und die Nichtlinearität des Problems sowohl über die Zeit als auch über die Kopplungsiterationen verändert, muss sich die Wahl am kleinsten notwendigen Parameter orientieren. Damit ist eine effiziente Wahl des Parameters nicht möglich.
- **Dynamischer Relaxationsparameter über Atiken's Δ^2 -Methode:** In jeder Kopplungsiteration wird ein neuer Relaxationsparameter bestimmt, der sich aus dem aktuellen Zustand des gekoppelten Problems ergibt. Da die Bestimmung des Parameters auf der Veränderung des Verschiebungs- und Residuenvektors am Interface basiert, ist eine Startprozedur notwendig. Dafür wird nach dem Prädiktorschritt z.B. ein konstanter Relaxationsschritt ausgeführt. Damit ist ein erstes

Inkrement vorhanden und der Aitken-Faktor μ_{n+1} kann über die Aitken-Formel bestimmt werden ([IT69]):

$$\mu_{n+1}^k = \mu_{n+1}^{k-1} + \left(\mu_{n+1}^{k-1} - 1 \right) \frac{\left(\Delta \mathbf{d}_{n+1}^{\Gamma, k-1} - \Delta \mathbf{d}_{n+1}^{\Gamma, k} \right)^T \cdot \Delta \mathbf{d}_{n+1}^{\Gamma, k}}{\left(\Delta \mathbf{d}_{n+1}^{\Gamma, k-1} - \Delta \mathbf{d}_{n+1}^{\Gamma, k} \right)^2} \quad (4.30)$$

mit

$$\begin{aligned} \Delta \mathbf{d}_{n+1}^{\Gamma, k-1} &= \tilde{\mathbf{d}}_{n+1}^{\Gamma, k-1} - \mathbf{d}_{n+1}^{\Gamma, k-1}, \\ \Delta \mathbf{d}_{n+1}^{\Gamma, k} &= \tilde{\mathbf{d}}_{n+1}^{\Gamma, k} - \mathbf{d}_{n+1}^{\Gamma, k}. \end{aligned}$$

Damit ergibt sich der dynamisch bestimmte Relaxationsparameter zu:

$$\omega^k = 1 - \mu_{n+1}^k. \quad (4.31)$$

Eine Interpretation obiger Formel findet sich z.B. in [DGV04, Dep04].

Dieses Verfahren wird in sehr vielen Arbeiten überwiegend eingesetzt (u.a. [Mok01, KW08, Kue09, JDP08, Kup09, Sch09]) oder für Vergleiche mit anderen Verfahren genutzt. Es ist robust und einfach zu implementieren, da es nur die Evaluation einiger Vektorprodukte erfordert. Allerdings besitzt es keine garantierte Stabilität. Damit kann es als Quasi-Standard angesehen werden.

- **Dynamischer Relaxationsparameter über die Methode des steilsten Abstiegs:** Mit dieser Methode (besser unter „Steepest Descent“ bekannt), wird ebenfalls ein dynamischer Relaxationsparameter bestimmt. Hierbei wird der Parameter basierend auf dem aktuellen Interfaceresiduum und der Jakobi-Matrix berechnet ([LTM01, Mok01, KW08]):

$$\omega_{n+1}^k = - \frac{\left(\mathbf{R}_{n+1}^{\Gamma, k} \right)^T \cdot \mathbf{R}_{n+1}^{\Gamma, k}}{\left(\mathbf{R}_{n+1}^{\Gamma, k} \right)^T \mathbf{J}_{n+1}^{\Gamma, k} \mathbf{R}_{n+1}^{\Gamma, k}}. \quad (4.32)$$

Da die Jakobi-Matrix bei den hier untersuchten Problemen nicht verfügbar ist, muss diese angenähert werden. Dies kann z.B. über die Verwendung finiter Differenzen oder über die Evaluierung eines reduzierten Fluidmodells geschehen. Diese Näherung ist allerdings sehr aufwendig und erfordert zusätzliche Löser-Evaluationen pro Kopplungssiteration. Des Weiteren wird in [KW08] gezeigt, dass dieser Ansatz deutlich weniger effizient als z.B. die Aitken-Methode ist.

Im weiteren Verlauf der Arbeit wird die dynamische Relaxation über Aitken's Δ^2 -Methode verwendet, da die beiden anderen Varianten eine deutlich geringere Effizienz aufweisen.

4.7. Vektorextrapolation

Die grundlegende Idee von Methoden, die auf Vektorextrapolation basieren, ist es, eine bereits konvergierende Serie von Vektoren in eine neue Serie zu transformieren, welche schneller als die vorhergehende konvergiert. Ein bereits konvergierendes Verfahren kann durch die Anwendung von Vektorextrapolation also beschleunigt werden. Diese Methode wird in der Regel bei der allgemeinen Lösung großer linearer Gleichungssysteme eingesetzt ([JS00, Sid91, Sid03]), bei der Lösung oberflächengekoppelter Probleme ([KW09]) und bei der Simulation turbulenter inkompressibler Strömungen zur Konvergenzbeschleunigung von Mehrgitterverfahren ([JJM07, JM07, JM07]).

Ist \mathbf{d}_{n+1}^Γ die Lösung des oberflächengekoppelten Problems zum Zeitpunkt t_{n+1} und existiert eine konvergierende Serie von k Interfaceverschiebungsvektoren $\mathbf{d}_{n+1}^{\Gamma,1}, \mathbf{d}_{n+1}^{\Gamma,2}, \dots, \mathbf{d}_{n+1}^{\Gamma,k}$, dann lässt sich die Lösung als Linearkombination mit den unbekanntem Extrapolationsfaktoren α_j dieser Serie approximieren:

$$\begin{aligned}\hat{\mathbf{d}}_{n+1}^\Gamma &= \mathbf{d}_{n+1}^{\Gamma,1} + \sum_{j=2}^k \alpha_j \left(\mathbf{d}_{n+1}^{\Gamma,j} - \mathbf{d}_{n+1}^{\Gamma,j-1} \right) \\ &= \mathbf{d}_{n+1}^{\Gamma,1} + \sum_{j=2}^k \alpha_j \Delta \mathbf{d}_{n+1}^{\Gamma,j}.\end{aligned}\quad (4.33)$$

Für den Fall einer konvergierenden Serie tendieren die Differenzen zwischen den Verschiebungsvektoren zu Null und dieses Inkrement kann als Verschiebungsresiduum $\bar{\mathbf{r}}$ innerhalb der Linearkombination betrachtet werden:

$$\lim_{j \rightarrow \infty} \left(\mathbf{d}_{n+1}^{\Gamma,j} - \mathbf{d}_{n+1}^{\Gamma,j-1} \right) = \lim_{j \rightarrow \infty} \bar{\mathbf{r}}^j \rightarrow 0, \quad (4.34)$$

mit $j \geq 2$. Die Extrapolation der Verschiebungen lässt sich auf das Residuum übertragen:

$$\begin{aligned}\hat{\mathbf{r}}_{n+1}^\Gamma &= \bar{\mathbf{r}}_{n+1}^{\Gamma,2} + \sum_{j=3}^k \alpha_j \left(\bar{\mathbf{r}}_{n+1}^{\Gamma,j} - \bar{\mathbf{r}}_{n+1}^{\Gamma,j-1} \right) \\ &= \bar{\mathbf{r}}_{n+1}^{\Gamma,2} + \sum_{j=3}^k \alpha_j \Delta \bar{\mathbf{r}}_{n+1}^{\Gamma,j}.\end{aligned}\quad (4.35)$$

Mit diesem Ansatz ergeben sich aber numerische Probleme, da bei fortschreitender Iteration und Konvergenz die Differenzvektoren $\Delta \bar{\mathbf{r}}$ gegen Null tendieren. Deshalb ist es numerisch sinnvoll, obigen Ansatz umzuformen, um Differenzvektoren zu verhindern:

$$\hat{\mathbf{r}}_{n+1}^\Gamma = \bar{\mathbf{r}}_{n+1}^{\Gamma,2} + \sum_{j=3}^k \alpha_j \left(\bar{\mathbf{r}}_{n+1}^{\Gamma,j} - \bar{\mathbf{r}}_{n+1}^{\Gamma,j-1} \right) = \sum_{j=1}^{k-1} \gamma_j \bar{\mathbf{r}}_{n+1}^{\Gamma,j} \quad (4.36)$$

mit $\gamma_j = \alpha_j - \alpha_{j+1}$ und $\alpha_1 = 1.0$.

Da das extrapolierte Residuum gegen Null konvergieren soll ($\hat{\mathbf{r}}_{n+1}^\Gamma \rightarrow 0$), lassen sich die Extrapolationsfaktoren γ_j bzw. α_j über eine Minimierung bestimmen:

$$\gamma_j = \arg \min \left| \sum_{j=1}^{k-1} \gamma_j \bar{\mathbf{r}}_{n+1}^{\Gamma,j} \right|. \quad (4.37)$$

Nach der notwendigen Normalisierung der Extrapolationsfaktoren:

$$\gamma_j \leftarrow \gamma_j / \sum_{j=1}^{k-1} \gamma_j \quad (4.38)$$

lassen sich auch die Extrapolationsfaktoren α_j bestimmen und damit ergibt sich die extrapolierte Interfaceverschiebung zu:

$$\hat{\mathbf{d}}_{n+1}^\Gamma = \sum_{j=1}^k \alpha_j \mathbf{d}_{n+1}^{\Gamma,j}. \quad (4.39)$$

Mit diesen Verschiebungen wird ein weiterer Löserzyklus durchlaufen und auf Konvergenz überprüft. Für den Fall keiner Konvergenz wird auf Grundlage dieser Verschiebungen eine weitere Serie von k Interfaceverschiebungen bestimmt, an dessen Ende wieder ein Extrapolationsschritt steht.

Das Minimierungsproblem 4.37 lässt sich allgemein als lineares und in der Regel überbestimmtes Gleichungssystem mit den $k - 1$ Gleichungen

$$\sum_{j=1}^{k-1} \gamma_j \beta_{i,j} = 0 \quad (4.40)$$

mit $i = 1$ bis $i = k - 1$ formulieren. Die Vektorextrapolationsverfahren werden nach der Art, in welcher die Koeffizienten $\beta_{i,j}$ bestimmt werden, unterschieden. In der Literatur werden verschiedene Verfahren diskutiert ([Sid91]). In den numerischen Beispielen werden in Anlehnung an [KW09] zwei Verfahren näher untersucht, dies sind die *Minimal Polynomial Extrapolation (MPE)* und die *Reduced Rank Extrapolation (RRE)*. Folgende Ansätze werden dabei gewählt:

- *Minimal Polynomial Extrapolation (MPE)*:

$$\beta_{i,j} = \bar{\mathbf{r}}_{n+1}^{\Gamma,i} \cdot \bar{\mathbf{r}}_{n+1}^{\Gamma,j}. \quad (4.41)$$

- *Reduced Rank Extrapolation (RRE)*:

$$\beta_{i,j} = \left(\bar{\mathbf{r}}_{n+1}^{\Gamma,i} \right)^2 \cdot \bar{\mathbf{r}}_{n+1}^{\Gamma,j}. \quad (4.42)$$

Bisher noch nicht angesprochen wurde, wie sich eine konvergierende Reihe von k Interfaceverschiebungen erzeugen lässt. In [KW09] wird dazu eine Fixpunktiteration mit konstanter Unterrelaxation vorgeschlagen. Prinzipiell sind aber alle Verfahren, die eine konvergierende Vektorsequenz erzeugen, geeignet, also z.B. auch Aitken's Δ^2 -Methode.

Die Erweiterung der Grundform eines impliziten Kopplungsalgorithmus für die Vektorextrapolation ist in Algorithmus 6 gegeben. Dabei wird pro Zeitschritt erst eine Vektorsequenz durch Ausführung von k konstanten Relaxationsschritten erzeugt. Darauf aufbauend wird ein Vektorextrapolationsschritt durchgeführt, dessen Ergebnis als Startwert für eine neue Sequenz dient. Die Konvergenz wird sowohl während des Erzeugens der Sequenz als auch nach dem Extrapolationsschritt überprüft.

Algorithmus 6 Kopplungsalgorithmus basierend auf Vektorextrapolation

```

1: for  $t = 0$  bis  $t = t_{tot}$  do
2:   while (keine Konvergenz) do
3:     Erzeugen der Vektorsequenz:
4:     for  $j = 1$  to  $j = k$  do
5:        $\tilde{\mathbf{d}}_{n+1}^{\Gamma,j} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,j})$ 
6:        $\mathbf{R}_{n+1}^{\Gamma,j} = \tilde{\mathbf{d}}_{n+1}^{\Gamma,j} - \mathbf{d}_{n+1}^{\Gamma,j}$ 
7:       if  $(\|\mathbf{R}_{n+1}^{\Gamma,j}\|_2 < \epsilon^\Gamma)$  then
8:          $\mathbf{d}_{n+1}^\Gamma = \mathbf{d}_{n+1}^{\Gamma,j}$ 
9:         konvergiert, gehe zu nächstem Zeitschritt
10:      else
11:        Relaxationsschritt:
12:         $\mathbf{d}_{n+1}^{\Gamma,j+1} = \mathbf{d}_{n+1}^{\Gamma,j} + \omega \mathbf{R}_{n+1}^{\Gamma,j}$ 
13:         $\bar{\mathbf{r}}_{n+1}^{\Gamma,j+1} = \mathbf{d}_{n+1}^{\Gamma,j+1} - \mathbf{d}_{n+1}^{\Gamma,j}$ 
14:      end if
15:       $j \leftarrow j + 1$ 
16:    end for
17:     $j = k$ 
18:    Vektorextrapolationsschritt:
19:    Löse  $0 \approx \sum_{j=1}^{k-1} \gamma_j \bar{\mathbf{r}}_{n+1}^{\Gamma,j} \rightarrow \gamma_j$ 
20:    Normalisiere  $\gamma_j$ :  $\gamma_j \leftarrow \gamma_j / (\sum_{j=1}^{k-1} \gamma_j)$ 
21:    Berechne extrapolierte Interfaceverschiebung:
22:     $\hat{\mathbf{d}}_{n+1}^{\Gamma,k} = \sum_{j=1}^k \alpha_j \mathbf{d}_{n+1}^{\Gamma,j}$ 
23:     $\mathbf{d}_{n+1}^{\Gamma,k} = \hat{\mathbf{d}}_{n+1}^{\Gamma,k}$ 
24:     $k = 1$ 
25:  end while
26:   $t_{n+1} = t_n + \Delta t_n$ 
27:   $n \leftarrow n + 1$ 
28: end for

```

4.8. Newton-Verfahren

Newton-Verfahren stellen einen sehr gebräuchlichen Ansatz zur Lösung nichtlinearer Probleme dar, wobei der Begriff Newton-Verfahren ein Überbegriff für eine ganze Reihe an Lösungsmethoden ist. Ein Überblick findet sich zum Beispiel in [Kel95, Kel03] oder auch [Deu04].

In seinem Grundsatz ist das Newton-Verfahren, auch Newton-Raphson Methode genannt, ein Näherungsverfahren zur Lösung nichtlinearer Gleichungen $f(x) = 0$. Dazu wird an einem Startpunkt x_0 die Tangente des Problems $\frac{df}{dx}(x_0)$ bestimmt (Linearisierung) und die Nullstelle der Tangente mittels $f(x_0) + \frac{df}{dx}(x_0)\Delta x$ berechnet. Dies liefert eine bessere Näherung der Nullstelle und dient als Ausgangspunkt für die nächste Iteration. Diese Abfolge wird wiederholt (Newton-Iterationen), bis die Nullstelle mit ausreichender Genauigkeit gefunden ist. Die Matrix der ersten Ableitungen des Problems wird u.a. auch Jakobi-Matrix genannt.

Das Newton-Verfahren ist ein lokal konvergentes Verfahren. Dies bedeutet, dass Konvergenz nur für einen Startwert x_0 erzielt wird, der genügend nahe an der tatsächlichen lokalen Lösung liegt. Ist der Startwert zu weit entfernt, kann das Verfahren divergieren oder gegen eine andere Nullstelle als die Gewünschte konvergieren. Deshalb ist die Wahl eines guten Startwertes von großer Bedeutung.

Die Konvergenzordnung von Newton-Verfahren ist unter Voraussetzung einer vollständigen Linearisierung des Problems maximal quadratisch. Dies bedeutet, dass sich die Anzahl der korrekten Dezimalstellen in jedem Schritt verdoppelt.

Bezogen auf das gekoppelte Problem der Fluid-Struktur-Wechselwirkung im partitionierten Ansatz ergibt sich als Funktion, deren Nullstelle gesucht wird, das Residuum am Interface \mathbf{R}^Γ und als Lösungsvariable, deren Wert zu bestimmen ist, der Verschiebungsvektor am Interface \mathbf{d}^Γ . In Operatorschreibweise bestimmt sich das Residuum über:

$$\mathbf{R}^\Gamma(\mathbf{d}^\Gamma) = \mathcal{S} \circ \mathcal{F}(\mathbf{d}^\Gamma) - \mathbf{d}^\Gamma = \tilde{\mathbf{d}}^\Gamma - \mathbf{d}^\Gamma. \quad (4.43)$$

Die Ableitung des Residuums, die Jacobimatrix, ist die Ableitung in Bezug auf die Interfacefreiheitsgrade und stellt die Linearisierung des nichtlinearen Problems an der aktuellen Stelle \mathbf{d}^Γ dar:

$$\frac{\partial \mathbf{R}^\Gamma}{\partial \mathbf{d}^\Gamma} = \mathbf{J}^\Gamma = \frac{\partial (\mathcal{S} \circ \mathcal{F}(\mathbf{d}^\Gamma) - \mathbf{d}^\Gamma)}{\partial \mathbf{d}^\Gamma}. \quad (4.44)$$

Innerhalb einer Newtoniteration mit Iterationsindex k ergibt sich damit folgendes lineares Gleichungssystem:

$$\frac{\partial \mathbf{R}^{\Gamma,k}}{\partial \mathbf{d}^\Gamma} \Delta \mathbf{d}^{\Gamma,k} = -\mathbf{R}^{\Gamma,k} \quad (4.45)$$

und dessen Lösung zu:

$$\Delta \mathbf{d}^{\Gamma,k} = \left(\frac{\partial \mathbf{R}^{\Gamma,k}}{\partial \mathbf{d}^\Gamma} \right)^{-1} \left(-\mathbf{R}^{\Gamma,k} \right). \quad (4.46)$$

Zur Bestimmung der Interfaceverschiebungen in der nächsten Newtoniteration wird die zugehörige Updateregeln verwendet:

$$\mathbf{d}^{\Gamma,k+1} = \mathbf{d}^{\Gamma,k} + \Delta \mathbf{d}^{\Gamma,k}. \quad (4.47)$$

Damit ist das Newtonverfahren ein inkrementelles Verfahren, da sich der Lösung schrittweise angenähert wird. Die Newtoniterationen werden solange ausgeführt, bis das Interfaceresiduum eine der in Abschnitt 4.3 vorgestellten Konvergenzschranken unterschreitet. Zur Bestimmung eines Startwertes $\mathbf{d}^{\Gamma,1}$ kann einer der in Abschnitt 4.4 vorgestellten Methoden verwendet werden.

Die Grundform eines Kopplungsalgorithmus auf Basis eines Newton-Verfahrens innerhalb einer transienten Problemstellung ist in Algorithmus 7 angegeben. In Bezug auf den numerischen Aufwand ist hierbei zu bemerken, dass innerhalb jedes Zeitschrittes an folgenden Stellen Gleichungssysteme zu lösen sind:

1. Während des Prädiktorschrittes bei der Evaluierung der beiden Teilfeldlöser.
2. Während jedes Newtonschrittes das Tangentenproblem am Interface.
3. Während jedes Newtonschrittes bei der Evaluierung der beiden Teilfeldlöser zur Berechnung des Residuums.

Die Schwierigkeit bei der Verwendung eines Newton-Verfahrens zur Lösung nichtlinearer Probleme liegt vor allem in der Evaluierung der Jacobimatrix $\mathbf{J}^{\Gamma} = \frac{\partial \mathbf{R}}{\partial \mathbf{d}^{\Gamma}}$. Während zum Beispiel in der nichtlinearen Strukturmechanik die exakte Bestimmung der Jacobimatrix keine Probleme bereitet (siehe Abschnitt 2.4), ist dies bei vielen anderen nichtlinearen Problemen und insbesondere bei gekoppelten Problemen der Fluid-Struktur-Wechselwirkung der Fall. Die Definition des gekoppelten Problems geschieht über die Interfaceoperatoren als Synonym für die dahinterliegenden nichtlinearen Feldoperatoren. Die Erstellung einer exakten Jakobi-Matrix erfordert nicht nur die Ableitung der Feldoperatoren nach den Feldvariablen, sondern auch nach den Interfacevariablen („Cross Derivatives“). Diese Informationen sind aber z.B. bei Verwendung von Druckkorrektur-Verfahren im Fluidfeld nicht verfügbar.

Die mögliche quadratische Konvergenz von Newton-Verfahren gilt nur bei einer vollständigen Linearisierung des gekoppelten Problems. Mit den hier vorgestellten Lösungsmöglichkeiten kann aber keine quadratische Konvergenz erreicht werden, da einzelne Terme immer angenähert werden müssen. Aussagen über die erzielten Konvergenzordnungen werden in Abschnitt 6.1 basierend auf der Auswertung numerischer Berechnungen gemacht.

Bei der Anwendung von allgemeinen Methoden zur Lösung nichtlinearer Probleme auf den spezifischen Fall des gekoppelten Problems der Fluid-Struktur-Wechselwirkung sind einige Besonderheiten zu beachten, um die einzelnen Verfahren bewerten zu können.

Anmerkung 1: Die Anzahl an Freiheitsgraden am Interface ist i.d.R. deutlich geringer als die Anzahl an Freiheitsgraden der Teilfeldlöser. Daraus folgt, dass das Lösen des

Algorithmus 7 Grundform eines Kopplungsalgorithmus auf Basis eines Newton-Verfahrens

```

1:  $n = 1, \quad t_n = t_0$ 
2: Iterationen über Zeitschritte:
3: while ( $t_n \leq t_{ges}$ ) do
4:    $k = 1$ 
5:   Prädiktorschritt nach Abschnitt 4.4:
6:      $\mathbf{d}_{n+1}^{\Gamma,1} = \mathbf{d}_{n+1}^{\Gamma,P}$ 
7:      $\tilde{\mathbf{d}}_{n+1}^{\Gamma,1} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,1})$ 
8:      $\mathbf{R}_{n+1}^{\Gamma,1}(\mathbf{d}_{n+1}^{\Gamma,1}) = \tilde{\mathbf{d}}_{n+1}^{\Gamma,1} - \mathbf{d}_{n+1}^{\Gamma,1}$ 
9:   Iterationen über Newtonschritte (Kopplungsschritte):
10:  while (keine Kopplungskonvergenz) do
11:    Löse  $\frac{\partial \mathbf{R}_{n+1}^{\Gamma,k}}{\partial \mathbf{d}^{\Gamma}} \Delta \mathbf{d}_{n+1}^{\Gamma,k} = -\mathbf{R}_{n+1}^{\Gamma,k} \rightarrow \Delta \mathbf{d}_{n+1}^{\Gamma,k}$ 
12:    Update Interfaceverschiebungen:
13:     $\mathbf{d}_{n+1}^{\Gamma,k+1} = \mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k}$ 
14:    Update Feldlöser:
15:     $\tilde{\mathbf{d}}_{n+1}^{\Gamma,k+1} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,k+1})$ 
16:     $\mathbf{R}_{n+1}^{\Gamma,k+1}(\mathbf{d}_{n+1}^{\Gamma,k+1}) = \tilde{\mathbf{d}}_{n+1}^{\Gamma,k+1} - \mathbf{d}_{n+1}^{\Gamma,k+1}$ 
17:    if  $\left| \mathbf{R}_{n+1}^{\Gamma,k+1}(\mathbf{d}_{n+1}^{\Gamma,k+1}) \right|_2 < \epsilon_{\Gamma}$  then
18:      Abbruch Newtonschritte, gehe zu nächstem Zeitschritt
19:       $\mathbf{d}_{n+1}^{\Gamma} = \mathbf{d}_{n+1}^{\Gamma,k+1}$ 
20:    else
21:      Gehe zu nächstem Newtonschritt
22:       $k \leftarrow k + 1$ 
23:    end if
24:  end while
25:   $t_{n+1} = t_n + \Delta t_n$ 
26:   $n \leftarrow n + 1$ 
27: end while

```

Interfacetangentenproblems numerisch deutlich weniger aufwendig ist als eine Evaluierung der Feldoperatoren. Damit ist bei allen Varianten von Newton-Verfahren eines zu bevorzugen, welches möglichst wenig Newtonschritte benötigt, auch wenn der Aufwand pro Newtonschritt größer ist.

Anmerkung 2: Die Bestimmung des Residuumvektors erfolgt bei allgemeinen Newton-Verfahren über eine Auswertung des nichtlinearen Problems an der aktuellen Position. Bei gekoppelten Problemen ist hierzu eine vollständige Evaluierung der Feldoperatoren notwendig. Dies ist von hohem Aufwand. Deshalb ist ein Newton-Verfahren zu bevorzugen, welches eine möglichst geringe Anzahl an Residuumsevaluierungen benötigt.

Als Lösungsmöglichkeit existiert eine ganze Bandbreite an Möglichkeiten, die auf einer Näherung der Jacobimatrix oder ihrer Inversen basiert. Wie in Abschnitt 4.5 bereits angesprochen, werden im Folgenden die Jakobi-freien Newton-Krylov-Verfahren vorgestellt, welche in jeder Newton-Iteration eine Evaluierung des Krylovraumes durchführen (siehe Abschnitt 4.8.1), und eine spezielle Variante eines Quasi-Newton-Verfahrens, welches keine zusätzliche Informationen von den Feldlösern benötigt (siehe Abschnitt 4.8.2).

4.8.1. Jakobi-freie Newton-Krylov-Verfahren

Jakobi-freie Newton-Krylov-Verfahren sind eine Kombination aus Krylov-Unterraum-Verfahren zur Lösung des linearen Gleichungssystems in jeder Newton-Iteration mit einer Näherung der zugehörigen Jakobi-Matrix. Diese Methoden finden bei der Lösung allgemeiner nichtlinearer Problemstellungen vielfach Anwendung. Bei der Anwendung auf partitionierte Kopplungsprobleme kann die normale Vorgehensweise aber nicht ohne Weiteres verwendet werden, sondern muss an die spezifische Problemstellung angepasst werden.

4.8.1.1. Krylov-Unterraum-Verfahren GMRES

Die große Schwierigkeit bei Verwendung von Newton-Verfahren zur Lösung gekoppelter Probleme stellt das Fehlen der Interface-Jakobi-Matrix bei der notwendigen Lösung des linearen Tangentenproblems dar. Bei Verwendung des Krylov-Unterraum-Verfahrens GMRES allerdings wird die Jacobimatrix nicht explizit benötigt - dies zeigt den großen Vorteil dieses Verfahrens. Die Methode erfordert zwar die Evaluierung diverser Matrix-Vektor-Produkte, diese können aber durch eine Evaluierung der Feldoperatoren angenähert werden, worauf im Weiteren noch eingegangen wird.

Auf Krylov-Verfahren im Allgemeinen und GMRES im Besonderen soll hier nur kurz eingegangen werden. Als Literatur zu diesem Themen eignen sich zum Beispiel die Bücher von Meister [Mei05] und Saad [Saa03].

Wir betrachten ein allgemeines lineares Gleichungssystem der Form:

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \quad (4.48)$$

mit der regulären Matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, der rechten Seite $\mathbf{b} \in \mathbb{R}^n$ und dem Vektor der Unbekannten $\mathbf{x} \in \mathbb{R}^n$. Als Krylov-Unterraum K_m bezeichnet man den von den Vektoren $\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \dots, \mathbf{A}^{m-1}\mathbf{r}_0$ aufgespannten Untervektorraum:

$$K_m = K_m(\mathbf{A}, \mathbf{r}_0) = \text{span} \{ \mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \dots, \mathbf{A}^{m-1}\mathbf{r}_0 \}, \quad (4.49)$$

mit dem Residuum $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ und \mathbf{x}_0 als Startwert für die Unbekannten. Krylov-Verfahren finden eine bessere Näherungslösung $\mathbf{x}_m \in \mathbf{x}_0 + K_m$ der gesuchten Lösung $\mathbf{A}^{-1}\mathbf{b}$ im Sinne einer Orthogonalitätsbedingung:

$$(\mathbf{b} - \mathbf{A}\mathbf{x}_m) \perp L_m, \quad (4.50)$$

mit dem Unterraum L_m . Die Dimension des Unterraums wird hierbei bei jeder Iteration um eins erhöht. Die Unterscheidung in die einzelnen speziellen Varianten geschieht basierend auf den Ansätzen zur Wahl des Unterraumes L_m . CG-Verfahren (Conjugate Gradient) wählen z.B. den Krylov-Raum als Unterraum: $L_m = K_m$. Krylov-Verfahren erreichen nach spätestens n Iterationen eine exakte Lösung. Von größerer Bedeutung ist aber, dass sie bereits nach wenigen Iterationen eine sehr gute Näherung der Lösung finden.

Das GMRES-Verfahren (*Generalized Minimal Residual Method*) als spezielle Variante der Krylov-Verfahren wurde von Saad und Schultz ([SS86]) 1986 vorgestellt. Es basiert auf einer aus dem Krylovraum K_m berechneten Orthonormalbasis \mathbf{V} mit den orthonormalen Basisvektoren \mathbf{v}_i , welche z.B. mittels der Arnoldi-Prozedur bestimmt werden. Die Basisvektoren werden auch Krylovvektoren genannt. Mit $\mathbf{V}_m = (\mathbf{v}_1, \dots, \mathbf{v}_m) \in \mathbb{R}^{n \times m}$ lässt sich jede Lösung \mathbf{x}_m mit $\mathbf{x}_m \in \mathbf{x}_0 + K_m$ darstellen als:

$$\mathbf{x}_m = \mathbf{x}_0 + \mathbf{V}_m \alpha_m \quad \text{mit} \quad \alpha_m \in \mathbb{R}^m. \quad (4.51)$$

Der Unterraum L_m ist bei GMRES also durch die Petrov-Galerkin-Bedingung $L_m = \mathbf{V}_m = \mathbf{A}K_m$ gegeben. Der zentrale Bestandteil von GMRES ist hierbei die Bestimmung der Faktoren α_m , welche aus einer Minimierung der Residuumsnorm folgt:

$$\alpha_m = \arg \min |\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{V}_m \alpha_m)|_2. \quad (4.52)$$

Die Lösung des Minimierungsproblems wird i.d.R. mittels Givens-Rotationen erzeugt, kann aber auch durch eine Householder-Transformation erfolgen. Für weitere Details wird an dieser Stelle aber auf die entsprechende Literatur verwiesen.

Vorlagen für die Programmierung eines GMRES-Algorithmus finden sich z.B. in [Mei05] oder [Ste05]. Im Hinblick auf die Matrix \mathbf{A} wird auch ersichtlich, dass der GMRES-Algorithmus diese Matrix nicht explizit benötigt, sondern das Matrix-Vektor-Produkt der Matrix mit dem aktuellen Krylovvektor $\mathbf{A}\mathbf{v}_m$ in jeder GMRES-Iteration.

Anmerkung 1: Krylov-Unterraum-Verfahren zeichnen sich generell dadurch aus, dass die Matrix der linken Seite des Gleichungssystems nicht explizit benötigt wird. Während die GMRES-Variante aber nur ein Matrix-Vektor-Produkt in der Form $\mathbf{A}\mathbf{v}_m$ in Kryloviteration m erfordert, benötigen andere Verfahren dieser Gruppe pro Kryloviteration z.B. die Auswertung mehrerer Matrix-Vektorprodukte (Conjugate-Gradient-Algorithmus) oder das Berechnen der Transponierten (Bi-Lanczos-Algorithmus). Damit ist das GMRES-Verfahren am Besten geeignet für eine Verwendung innerhalb der Lösung gekoppelter Probleme, da hier die Evaluation der linken Seite von hohem numerischen Aufwand ist.

Anmerkung 2: Ein Nachteil des GMRES-Verfahrens ist der prinzipiell hohe Speicherbedarf für die Basisvektoren in Abhängigkeit der Iterationszahl. Dabei kann es dazu kommen, dass im Extremfall bei einem Gleichungssystem mit n Unbekannten die Orthonormalbasis $n \times n$ Einträge hat. Bei der Anwendung zur Lösung eines gekoppelten Problems ist dies aber von untergeordneter Bedeutung. Die Anzahl an Unbekannten im GMRES-System entspricht der Anzahl an Interfacefreiheitsgraden. Im Vergleich mit der Anzahl an Freiheitsgraden der Einzelfeldprobleme ist diese in der Regel deutlich geringer, womit sich der zusätzlich notwendige Speicheraufwand am Interface relativiert.

Anmerkung 3: Das Minimierungsproblem wird in dieser Arbeit mittels der Householder-Transformation gelöst. Dies ist zwar numerisch aufwendiger als eine Givens-Rotation,

erlaubt es aber, Krylov-Informationen aus vorherigen Kopplungsiterationen in das Minimierungsproblem aufzunehmen, worauf in Abschnitt 4.8.1.3 noch näher eingegangen wird.

4.8.1.2. Einbettung in den Newton-Kopplungsalgorithmus

In diesem Kapitel wird die Einbettung des GMRES-Verfahrens in den Newton-Kopplungsalgorithmus besprochen. Betrachtet man die Grundform des Newton-Kopplungsalgorithmus (siehe Algorithmus 7), wird der GMRES-Algorithmus zur Lösung des linearen Tangentenproblems am Interface in jedem Kopplungsschritt benutzt (vgl. Gl. 4.45):

$$\frac{\partial \mathbf{R}_{n+1}^{\Gamma,k}}{\partial \mathbf{d}^{\Gamma}} \Delta \mathbf{d}_{n+1}^{\Gamma,k} = -\mathbf{R}_{n+1}^{\Gamma,k}. \quad (4.53)$$

Damit entspricht die Matrix \mathbf{A} aus der Herleitung des GMRES-Verfahrens der Jakobi-Matrix $\frac{\partial \mathbf{R}_{n+1}^{\Gamma,k}}{\partial \mathbf{d}^{\Gamma}}$, die Unbekannte \mathbf{x} der inkrementellen Interfaceverschiebung $\Delta \mathbf{d}_{n+1}^{\Gamma,k}$ und die rechte Seite \mathbf{b} dem negativen Residuumsvektor $-\mathbf{R}_{n+1}^{\Gamma,k}$. Die Jakobi-Matrix wird im Folgenden abgekürzt mit $\frac{\partial \mathbf{R}_{n+1}^{\Gamma,k}}{\partial \mathbf{d}^{\Gamma}} = \left(\mathbf{R}_{n+1}^{\Gamma,k} \right)'$.

Wie bereits erwähnt, besteht der Vorteil der GMRES-Methode darin, dass die Jakobi-Matrix nicht explizit benötigt wird. Bei der Lösung des Tangentenproblems wird allerdings in jeder Kryloviteration (Iterationsindex m) das Matrix-Vektor-Produkt aus Jakobi-Matrix und Krylovvektor $\left(\mathbf{R}_{n+1}^{\Gamma,k} \right)' \mathbf{v}_m$ zur Evaluierung des Krylovraumes benötigt.

Anschaulich gesehen entsprechen die Krylovvektoren \mathbf{v}_m inkrementellen Interfaceverschiebungen $\Delta \mathbf{d}$. Das nichtlineare gekoppelte Problem wird ausgehend von der aktuellen Position des Interfaces $\mathbf{d}_{n+1}^{\Gamma,k}$ an verschiedenen, durch die Krylovvektoren \mathbf{v}_m vorgegebenen, Positionen evaluiert:

$$\mathbf{d}_{n+1}^{\Gamma,k,m} = \mathbf{d}_{n+1}^{\Gamma,k} + \mathbf{v}_m. \quad (4.54)$$

Nach Konvergenz des GMRES-Lösers bestimmt sich das neue Verschiebungsinkrement $\Delta \mathbf{d}_{n+1}^{\Gamma,k}$ als Kombination der Krylovvektoren:

$$\Delta \mathbf{d}_{n+1}^{\Gamma,k} = \sum_{m=1}^n \mathbf{v}_m \alpha_m, \quad (4.55)$$

wobei sich die Faktoren α_m aus der Lösung des zugehörigen Minimierungsproblems ergeben:

$$\alpha_m = \arg \min \left| \mathbf{R}_{n+1}^{\Gamma,k} - \sum_{j=1}^m \alpha_j \left(\mathbf{R}_{n+1}^{\Gamma,k} \right)' \mathbf{v}_j \right|_2. \quad (4.56)$$

Angewandt auf die Grundform des Newton-Kopplungsalgorithmus (siehe Algorithmus 7) ergibt sich also zur Lösung des Tangentenproblems noch eine zusätzliche Schleife über

die GMRES-Iterationen. Die algorithmische Darstellung eines Newton-Kopplungsalgorithmus unter Verwendung des GMRES-Verfahrens ist in Algorithmus 8 gegeben. Zur Evaluierung des Krylovraumes werden spezifische Teilfeldlöser verwendet. Diese sind durch einen Stern gekennzeichnet: Für das Strukturfeld \mathcal{S}^* und für das Fluidfeld \mathcal{F}^* .

Anmerkung 1: Des Weiteren ist zu beachten, dass bei Verwendung von Newton-Kopplungsmethoden in Kombination mit Krylov-Verfahren in der Regel keine Standard-Feldlöseransätze für die Einzelfelder verwendet werden können. In jeder Kopplungsiteration ist die zusätzliche Evaluierung des zugehörigen Krylovunterraumes innerhalb der GMRES-Iterationen nötig. Diese Evaluierung ist mit Standard-Lösungsmethoden nur sehr eingeschränkt möglich. Auf diesen Punkt wird in den weiteren Abschnitten näher eingegangen.

Anmerkung 2: Bei der Lösung des Tangentenproblems mittels GMRES wird das zugehörige Gleichungsresiduum als Teil des GMRES-Lösers mitberechnet. Ein wichtiger Aspekt ist hierbei die festzulegende Genauigkeit der Lösung. Bei zu geringer Genauigkeit wird im folgendem Newton-Schritt das Interfaceresiduum nicht effektiv verringert, bei zu hoher Genauigkeit müssen die spezifischen Feldlöser unnötig oft gerufen werden. Aus numerischen Beispielen ist ersichtlich, dass am Anfang der Kopplungsiterationen bei relativ hohem Interfaceresiduum bereits eine Lösung mit geringer Genauigkeit das Interfaceresiduum effektiv verringert. Ist das gekoppelte Problem nahezu konvergiert und das Interfaceresiduum bereits stark reduziert, ist eine hohe Genauigkeit notwendig, um eine weitere Reduktion zu ermöglichen.

In dieser Arbeit wird eine adaptive Toleranzschranke für den GMRES-Löser verwendet, welche in jeder Kopplungsiteration neu bestimmt wird. Dazu wird ein Initialwert vorgeschrieben, welcher sicherstellt, dass in der ersten Kopplungsiteration nach dem Prädiktorschritt das Interfaceresiduum verringert wird, und dieser mit der aktuellen Norm des Interfaceresiduum multipliziert. Startet man also mit einer GMRES-Toleranz von $\epsilon_{GMRES}^{init} = 1,0e^{-6}$ und einer Interfacetoleranz von $\epsilon^{\Gamma} = 1,0e^{-5}$, ergibt sich für die letzte Kopplungsiteration eine Toleranzschranke für den GMRES-Löser in Höhe von $\epsilon_{GMRES} = 1,0e^{-11}$.

4.8.1.3. Wiederverwendung von Krylovrauminformationen

Unter der Annahme, dass ein instationäres Problem betrachtet wird und sich die Jakobi-Matrix über die Zeitschritte und Kopplungsiterationen nur bedingt verändert, können in das Minimierungsproblem (Gleichung 4.56) auch Matrix-Vektor-Produkte aus vorherigen Kopplungsiterationen und Zeitschritten aufgenommen werden. Dies ist eine Methode, welche oft angewendet wird (z.B. [BMB05, Mic05, MVBDB05]). Ist die Information des Matrix-Vektor-Produktes nicht mehr relevant, wird das Residuum dadurch nicht minimiert und der zugehörige Faktor α_m geht gegen Null. Eine wichtige Frage ist hierbei, wie viele Produkte zusätzlich mit aufgenommen werden sollen. Dies wird im numerischen Beispiel in Abschnitt 6.1.5 näher untersucht. Die Wiederverwendung ist unabhängig von der Art der Krylovraumevaluierung. Die notwendige Anpassung des Kopplungsalgorithmus

Algorithmus 8 Newton-Kopplungsalgorithmus unter Verwendung des GMRES-Verfahrens

```

1:  $n = 1, \quad t_n = t_0$ 
2: Iterationen über Zeitschritte:
3: while ( $t_n \leq t_{ges}$ ) do
4:    $k = 1$ 
5:   Prädiktorschritt:
6:      $\mathbf{d}_{n+1}^{\Gamma,1} = \mathbf{d}_{n+1}^{\Gamma,P}$ 
7:      $\tilde{\mathbf{d}}_{n+1}^{\Gamma,1} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,1})$ 
8:      $\mathbf{R}_{n+1}^{\Gamma,1}(\mathbf{d}_{n+1}^{\Gamma,1}) = \tilde{\mathbf{d}}_{n+1}^{\Gamma,1} - \mathbf{d}_{n+1}^{\Gamma,1}$ 
9:   Iterationen über Newtonschritte (Kopplungsschritte):
10:  while (keine Kopplungskonvergenz) do
11:     $m = 1, \quad \mathbf{v}_1 = \mathbf{R}_{n+1}^{\Gamma,k} / \left| \mathbf{R}_{n+1}^{\Gamma,k} \right|_2$ 
12:    Iterationen über GMRES-Schritte:
13:    while (keine GMRES-Konvergenz) do
14:      Evaluation Krylovraum:  $(\mathbf{R}_{n+1}^{\Gamma,k})' \mathbf{v}_m = f(\mathcal{S}^* \circ \mathcal{F}^*(\mathbf{v}_m))$ 
15:      GMRES-Löser:  $\mathbf{v}_{m+1} = f((\mathbf{R}_{n+1}^{\Gamma,k})' \mathbf{v}_m)$ 
16:      if  $\left( \left| \mathbf{R}_{n+1}^{\Gamma,k} - \sum_{n=1}^m \alpha_n (\mathbf{R}_{n+1}^{\Gamma,k})' \mathbf{v}_n \right|_2 \leq \epsilon_{GMRES}^{\Gamma} \right)$  then
17:        Abbruch GMRES-Schritte, gehe zu nächstem Kopplungsschritt
18:         $\Delta \mathbf{d}_{n+1}^{\Gamma,k} = \sum_{n=1}^{m+1} \mathbf{v}_n \alpha_n$ 
19:      else
20:        Gehe zu nächstem GMRES-Schritt
21:         $m \leftarrow m + 1$ 
22:      end if
23:    end while
24:    Update Interfaceverschiebungen:
25:     $\mathbf{d}_{n+1}^{\Gamma,k+1} = \mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k}$ 
26:    Update Feldlöser:
27:     $\tilde{\mathbf{d}}_{n+1}^{\Gamma,k+1} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,k+1})$ 
28:     $\mathbf{R}_{n+1}^{\Gamma,k+1}(\mathbf{d}_{n+1}^{\Gamma,k+1}) = \tilde{\mathbf{d}}_{n+1}^{\Gamma,k+1} - \mathbf{d}_{n+1}^{\Gamma,k+1}$ 
29:    if  $\left( \left| \mathbf{R}_{n+1}^{\Gamma,k+1}(\mathbf{d}_{n+1}^{\Gamma,k+1}) \right|_2 < \epsilon^{\Gamma} \right)$  then
30:      Abbruch Newtonschritte, gehe zu nächstem Zeitschritt
31:       $\mathbf{d}_{n+1}^{\Gamma} = \mathbf{d}_{n+1}^{\Gamma,k+1}$ 
32:    else
33:      Gehe zu nächstem Newtonschritt
34:       $k \leftarrow k + 1$ 
35:    end if
36:  end while
37:   $t_{n+1} = t_n + \Delta t_n$ 
38:   $n \leftarrow n + 1$ 
39: end while

```

ist nur geringfügig.

4.8.1.4. Krylovraumevaluierung

Ein maßgeblicher Punkt bei Verwendung des GMRES-Verfahrens stellt die Evaluierung des Krylovraumes dar. Bei Verwendung von GMRES zur Lösung eines linearen Gleichungssystems ist diese Evaluierung eine einfache Vektor-Matrix-Multiplikation mit der gegebenen Matrix \mathbf{A} . Nachdem diese Matrix bei einem gekoppelten Problem nicht existiert, muss dieses Produkt angenähert werden. Zur Beurteilung des Näherungsverfahrens sind vor allem zwei Punkte von Bedeutung:

- Qualität der Näherung: Je besser die Näherung, desto größer ist die durch den GMRES-Algorithmus erzielte Residuumsreduktion am Interface je Kopplungsschritt.
- Numerischer Aufwand: Da die Näherung in jedem Krylovschritt zu bestimmen ist, ist der zugehörige numerische Aufwand von großem Einfluss auf die numerische Effizienz des gesamten Verfahrens.

Für ein effizientes Kopplungsverfahren ist es also das Ziel der Näherung, einen möglichst guten Kompromiss aus Aufwand und Qualität zu finden.

Der prinzipielle Ablauf zur Evaluierung des Krylovraums gliedert sich, ähnlich des Ablaufs eines Zyklus innerhalb der impliziten Kopplung, in die folgenden Teile:

1. Berechnung des Krylovvektors \mathbf{v}_m , der anschaulich gesehen einer zusätzlichen inkrementellen Interfaceverschiebung entspricht.
2. Evaluierung des spezifischen Fluidfeldlösers \mathcal{F}^* und Berechnung der spezifischen Fluidlasten:

$$\mathbf{f}_{m+1}^* = \mathcal{F}^*(\mathbf{d}_{n+1}^{\Gamma,k} + \mathbf{v}_m). \quad (4.57)$$

3. Evaluierung des spezifischen Strukturfeldlösers \mathcal{S}^* und Berechnung der spezifischen Strukturverschiebungen:

$$\tilde{\mathbf{v}}_{m+1} = \mathcal{S}^*(\mathbf{f}_{n+1}^{\Gamma,k} + \mathbf{f}_{m+1}^*). \quad (4.58)$$

4. Basierend auf der Berechnung des Residuums des gekoppelten Problems:

$$\left(\mathbf{R}_{n+1}^{\Gamma,k}\right)' \Delta \mathbf{d}_{n+1}^{\Gamma,k} = -\mathbf{R}_{n+1}^{\Gamma,k}, \quad (4.59)$$

$$\begin{aligned} -(\mathbf{R}_{n+1}^{\Gamma,k}) &= -(\tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k}) \\ &= \mathbf{d}_{n+1}^{\Gamma,k} - \tilde{\mathbf{d}}_{n+1}^{\Gamma,k}, \end{aligned} \quad (4.60)$$

ergibt sich analog die Auswertung des Matrix-Vektor-Produktes aus Tangentematrix und Krylovvektor zu:

$$\left(\mathbf{R}_{n+1}^{\Gamma,k}\right)' \mathbf{v}_m = \mathbf{v}_m - \tilde{\mathbf{v}}_{m+1}. \quad (4.61)$$

Wie hieraus ersichtlich wird, ist die Verwendung von Black-Box-Teilfeldlösern nicht mehr möglich. Die notwendige Anpassung der Teilfeldlöser wird in Algorithmus 9 für das Fluidfeld und in Algorithmus 10 für das Strukturfeld vereinfacht angegeben. Dazu wird wieder die Operatorschreibweise benutzt, um auf die Details der Einzelfeldlösungsalgorithmen verzichten zu können (diese sind in Algorithmus 1 für das Strukturfeld und in Algorithmus 4 für das Fluidfeld angegeben).

Algorithmus 9 Fluidlösungsalgorithmus für Newton-Krylov-Kopplungsalgorithmen

```

1:  $n = 1, \quad t_n = t_0$ 
2: Iterationen über Zeitschritte:
3: while ( $t_n \leq t_{ges}$ ) do
4:    $k = 1$ 
5:   Prädiktorschritt Kopplung:
6:      $\mathbf{f}_{n+1}^{\Gamma,1} = \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,1})$ 
7:   Iterationen über Kopplungsschritte:
8:   while (keine Kopplungskonvergenz) do
9:      $m = 1$ 
10:    Iterationen über GMRES-Schritte:
11:    while (keine GMRES-Konvergenz) do
12:      Evaluation Krylovraum:
13:       $\mathbf{f}_{m+1}^* = \mathcal{F}^*(\mathbf{d}_{n+1}^{\Gamma,k} + \mathbf{v}_m)$ 
14:      Gehe zu nächstem GMRES-Schritt
15:       $m \leftarrow m + 1$ 
16:    end while
17:    Update Feldlöser:
18:     $\mathbf{f}_{n+1}^{\Gamma,k+1} = \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k})$ 
19:     $\mathbf{d}_{n+1}^{\Gamma,k+1} = \mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k}$ 
20:    Gehe zu nächster Kopplungsiteration
21:     $k \leftarrow k + 1$ 
22:  end while
23:  Gehe zu nächstem Zeitschritt
24:   $t_{n+1} = t_n + \Delta t_{n+1}$ 
25:   $n \leftarrow n + 1$ 
26: end while

```

4.8.1.5. Methoden zur Krylovraumevaluierung

Bevor die möglichen Ansätze zur Evaluierung des Krylovraumes vorgestellt werden, sollen an dieser Stelle noch einige grundlegende Überlegungen dargestellt werden. Die Evaluierung findet an einem Iterationspunkt statt, an dem sich die Teilfeldlöser im Gleich-

Algorithmus 10 Strukturlösungsalgorithmus für Newton-Krylov-Kopplungsalgorithmen

```

1:  $n = 1, \quad t_n = t_0$ 
2: Iterationen über Zeitschritte:
3: while ( $t_n \leq t_{ges}$ ) do
4:    $k = 1$ 
5:   Prädiktorschritt Kopplung:
6:      $\tilde{\mathbf{d}}_{n+1}^{\Gamma,1} = \mathcal{S}(\mathbf{f}_{n+1}^{\Gamma,1})$ 
7:   Iterationen über Kopplungsschritte:
8:   while (keine Kopplungskonvergenz) do
9:      $m = 1$ 
10:    Iterationen über GMRES-Schritte:
11:    while (keine GMRES-Konvergenz) do
12:      Evaluation Krylovraum:
13:       $\tilde{\mathbf{v}}_{m+1} = \mathcal{S}^*(\mathbf{f}_{n+1}^{\Gamma,k} + \mathbf{f}_{m+1}^*)$ 
14:      Gehe zu nächstem GMRES-Schritt
15:       $m \leftarrow m + 1$ 
16:    end while
17:    Update Feldlöser:
18:     $\tilde{\mathbf{d}}_{n+1}^{\Gamma,k+1} = \mathcal{S}(\mathbf{f}_{n+1}^{\Gamma,k}, \mathbf{f}_{n+1}^{\Gamma,k+1})$ 
19:    Gehe zu nächster Kopplungsiteration
20:     $k \leftarrow k + 1$ 
21:  end while
22:  Gehe zu nächstem Zeitschritt
23:   $t_{n+1} = t_n + \Delta t_{n+1}$ 
24:   $n \leftarrow n + 1$ 
25: end while

```

gewicht befinden, das gekoppelte Problem jedoch nicht. Dieser Punkt ist charakterisiert durch die aktuelle Interfaceverschiebung. Die Krylovvektoren sind anschaulich gesehen inkrementelle Interfaceverschiebungen. Bei der Evaluierung des Krylovraumes wird die Sensitivität des Residuums des gekoppelten Problems in Bezug auf eine Veränderung der Interfaceposition bestimmt. Auf diese Art wird der Krylovraum aufgespannt, in dem sich das neue Verschiebungsinkrement befindet.

Die Möglichkeiten zur Bestimmung der Sensitivität sind vielfältig, wobei vor allem das Verhältnis von Evaluierungsqualität zu numerischem Aufwand maßgebend ist. Prinzipiell können an allen Punkten, die in die Erstellung des numerischen Modells eingehen, Aufwand und Qualität beeinflusst werden, indem zur Erstellung des approximativen Tangentenoperators reduzierte Modelle betrachtet werden. Dazu zählen unter anderem folgende Punkte, welche natürlich teilweise ineinander übergehen:

- Reduzierung der abzubildenden Physik, indem Terme in der Differentialgleichung vernachlässigt werden.
- Lösung eines Ersatzproblems, welches einfacher zu lösen ist, und dabei die wesentlichen Eigenschaften des ursprünglichen Problems beinhaltet.
- Vernachlässigung der Nichtlinearität der Teilfelder durch Linearisierung.
- Maßgebend für die Rechenzeit bei der Lösung des gekoppelten Problems sind die Lösung von Netzbewegung, Druckgleichung und effektiver Strukturgleichung. Vernachlässigung eines oder mehrerer dieser Bereiche durch Wahl entsprechender Randbedingungen eines anderen Bereichs.

Die Entwicklung und Verwendung reduzierter Modelle sind allerdings sehr schwierig. Um eine Reduzierung vornehmen zu können, müssen Punkte im numerischen Modell gefunden werden, deren Behandlung zwar sehr aufwendig ist, aber gleichzeitig nur geringen Einfluss auf das Ergebniss haben. Dies stellt i.d.R. aber einen Widerspruch dar, da vorhandene komplexe Modelle nur deswegen entwickelt werden, weil einfachere Modelle die Gegebenheiten nicht adäquat abbilden können. Auch ist eine Modellreduzierung stark problemabhängig, was das Finden von allgemeinen Ansätzen erschwert.

Krylovraumevaluierung über Finite Differenzen: Als einfachste Methode zur Krylovraumevaluierung ergibt sich die Verwendung **Finiter Differenzen**. Dieser Ansatz wird z.B. in den Arbeiten von Heil ([Hei04]) oder Matthies ([MS02, MS03]) genutzt. Dabei werden die nicht-reduzierten Modelle von Fluid- und Strukturfeld evaluiert, um über globale Finite Differenzen das Matrix-Vektor-Produkt bestimmen zu können. Damit ergibt sich bei Verwendung beispielsweise eines vorwärts gerichteten Differenzenquotienten:

$$\left(\mathbf{R}_{n+1}^{\Gamma,k}\right)' \left(\mathbf{d}_{n+1}^{\Gamma,k}\right) \mathbf{v}_m \approx \frac{\mathbf{R}_{n+1}^{\Gamma,k}(\mathbf{d}_{n+1}^{\Gamma,k} + \alpha \mathbf{v}_m) - \mathbf{R}_{n+1}^{\Gamma,k}(\mathbf{d}_{n+1}^{\Gamma,k})}{\alpha}, \quad (4.62)$$

und bei Verwendung eines zentralen Differenzenquotienten:

$$\left(\mathbf{R}_{n+1}^{\Gamma,k}\right)'(\mathbf{d}_{n+1}^{\Gamma,k})\mathbf{v}_m \approx \frac{\mathbf{R}_{n+1}^{\Gamma,k}(\mathbf{d}_{n+1}^{\Gamma,k} + \alpha\mathbf{v}_m) - \mathbf{R}_{n+1}^{\Gamma,k}(\mathbf{d}_{n+1}^{\Gamma,k} - \alpha\mathbf{v}_m)}{2\alpha}, \quad (4.63)$$

unter Verwendung des skalaren Wertes α , welcher die Schrittweite des Differenzenschrittes bestimmt.

Bei Verwendung des vorwärts gerichteten Differenzenquotienten ist das Interfaceresiduum $\mathbf{R}_{n+1}^{\Gamma,k}(\mathbf{d}_{n+1}^{\Gamma,k})$ bereits bekannt. Das Residuum $\mathbf{R}_{n+1}^{\Gamma,k}(\mathbf{d}_{n+1}^{\Gamma,k} + \alpha\mathbf{v}_m)$ wird über die Standard-Feldoperatoren ausgewertet. Die spezifischen Teilfeldlöser \mathcal{S}^* und \mathcal{F}^* sind also identisch mit den Standard-Feldoperatoren, die an der Position $\mathbf{d}_{n+1}^{\Gamma,k} + \alpha\mathbf{v}_m$ ausgewertet werden müssen. Dies führt zu:

$$\mathbf{R}_{n+1}^{\Gamma,k}(\mathbf{d}_{n+1}^{\Gamma,k} + \alpha\mathbf{v}_m) = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,k} + \alpha\mathbf{v}_m) - (\mathbf{d}_{n+1}^{\Gamma,k} + \alpha\mathbf{v}_m). \quad (4.64)$$

Bewertung: Da hier die nicht-reduzierten Teilfeldprobleme evaluiert werden und damit keine Effekte vernachlässigt werden, kann die Qualität der Ergebnisse prinzipiell sehr hoch sein. Die Skalierung der Krylovvektoren mittels α ist notwendig, da diese per Definition die Länge eins besitzen.

Dieser Ansatz hat aber drei gravierende Nachteile:

1. Der Einfluss von α auf die Evaluierungsqualität ist sehr hoch, da damit der Radius für die sinnvolle Evaluierung des nichtlinearen Problems bestimmt wird. Vorgaben für dessen Bestimmung sind aber im Kontext der Fluid-Struktur-Wechselwirkung unbekannt. Ist α zu groß, konvergieren die Teilfeldlöser nicht mehr, ist α zu klein, haben numerische Rundungsfehler einen großen Lösungseinfluss.
2. In jeder Kopplungsiteration ist zusätzlich die mehrfache Lösung des gesamten Problems notwendig. Dies führt zu hohem numerischen Aufwand.
3. Da die Krylovvektoren orthogonal zueinander sind, ist es bei der Krylovraumevaluierung für iterative Feldlöser schwer, effizient zu arbeiten. Damit ist die Evaluierung des Krylovraumes eventuell sogar aufwendiger als ein Standardruf der Feldlöser.

Damit stellt die Evaluierung über Finite Differenzen ein zwar qualitativ hochwertiges, aber numerisch sehr aufwendiges Vorgehen dar.

Krylovraumevaluierung über ein Ersatzproblem: Eine weitere Methode stellt die Verwendung eines **Ersatzproblems** dar. Hier sind vor allem die Arbeiten von Deparis und Gerbeau zu nennen ([DGV04, Dep04, GV03, GVF05]). Diese betrachten als Problemstellung die Arterienströmung mit einer Interaktion von Blutfluss und elastischer Arterie. Diese Probleme sind bekannt für die starke Kopplung zwischen den Feldern. Als maßgeblicher Effekt stellt sich hierbei der sogenannte „Added Mass Effect“ innerhalb des Fluidfeldes heraus. Das Ersatzproblem versucht, das ursprüngliche Problem durch ein einfacheres Problem zu ersetzen, welches die maßgeblichen Eigenschaften abbilden kann.

Laut Gerbeau ([GV03]) lässt sich dieser Mechanismus durch das einfache lineare Fluidmodell einer reibungsfreien Strömung hinreichend beschreiben. Dieses Modell entspricht damit einer Potentialströmung.

Das Fluidgebiet wird hierbei im aktuellen Iterationspunkt fixiert ($\Omega_{n+1}^{F,k}$) und das Ersatzproblem mit den Variablen p^* und \mathbf{u}^* wird durch folgende Gleichungen beschrieben:

$$\rho_F \frac{\partial \mathbf{u}^*}{\partial t} + \nabla p^* = 0 \quad \text{in } \Omega_{n+1}^{F,k}, \quad (4.65)$$

$$\mathbf{u}^* = \frac{\partial \mathbf{v}}{\partial t} \quad \text{auf } \Gamma_{n+1}^{FSI,k}, \quad (4.66)$$

$$p^* = 0 \quad \text{auf } \Gamma_{n+1}^{F,k}. \quad (4.67)$$

Die Eliminierung von \mathbf{u}^* und das Einführen der entsprechenden Iterationsindizes führt zu:

$$\Delta p_{m+1}^* = 0 \quad \text{in } \Omega_{n+1}^{F,k}, \quad (4.68)$$

$$\frac{\partial p_{m+1}^*}{\partial \mathbf{n}} = -\frac{\rho_F}{\Delta t^2} \mathbf{v}_m \cdot \mathbf{n} \quad \text{auf } \Gamma_{n+1}^{FSI,k}, \quad (4.69)$$

$$p_{m+1}^* = 0 \quad \text{auf } \Gamma_{n+1}^{F,k}, \quad (4.70)$$

mit \mathbf{n} als Normalenvektor auf dem Fluidinterface. Dies entspricht einem skalaren Poisson-Problem.

Aus dem damit berechneten Druck am Interface p_{m+1}^* wird der Lastvektor \mathbf{f}_{m+1}^* bestimmt. Die Lösung des spezifischen Fluidproblems in Operatorschreibweise ergibt sich zu $\mathbf{f}_{m+1}^* = \mathcal{F}^*(\mathbf{d}_{n+1}^{\Gamma,k} + \mathbf{v}_m)$.

Für das spezifische Strukturproblem wird die am aktuellen Iterationspunkt vorhandene Tangentensteifigkeit \mathbf{K}_T^S verwendet:

$$\mathbf{K}_{T,n+1}^{S,k} \tilde{\mathbf{v}}_{m+1} = -\mathbf{f}_{m+1}^* \quad \text{in } \Omega_{n+1}^{S,k}. \quad (4.71)$$

Die Lösung des spezifischen Strukturproblems in Operatorschreibweise ergibt sich zu $\tilde{\mathbf{v}}_{m+1} = \mathcal{S}^*(\mathbf{f}_{n+1}^{\Gamma,k} + \mathbf{f}_{m+1}^*)$. Bei Verwendung des Generalized- α Zeitintegrationsverfahrens muss bei Lösung des spezifischen Strukturproblems beachtet werden, dass dieses Zeitintegrationsverfahren eine Gewichtung der Anteile von Anfang und Ende des Zeitschritts über die Shift-Parameter α_m und α_f einführt (siehe Abschnitt 2.3.1). Damit kann entweder auf die Skalierung verzichtet werden, womit sich eine Tangentensteifigkeit unter Verwendung des Newmark- β Zeitintegrationsverfahrens ergibt, oder eine Skalierung auch der rechten Seite vorgenommen werden. Diese zweite Möglichkeit hat den Vorteil, dass die Tangentensteifigkeit des letzten Iterationspunktes weiter verwendet werden kann und damit nicht neu aufgestellt werden muss. Daraus folgt:

$$\begin{aligned}\mathbf{K}_{T,n+1}^{S,k} &= \frac{\partial \mathbf{R}_{n+1}^{S,k}(\mathbf{d}_{n+1}^k)}{\partial \mathbf{d}_{n+1}} \\ &= (1 - \alpha_m) \mathbf{M} \cdot \frac{1}{\beta \Delta t^2} + (1 - \alpha_f) \mathbf{C} \cdot \frac{\gamma}{\beta \Delta t} + (1 - \alpha_f) \mathbf{f}_{n+1}^{int}(\mathbf{d}_{n+1}^k),\end{aligned}\quad (4.72)$$

$$\mathbf{f}_{m+1}^* \leftarrow (1 - \alpha_f) \mathbf{f}_{m+1}^*. \quad (4.73)$$

Anmerkung: Betrachtet man die ursprüngliche Form des Strukturproblems (Gleichungen 2.39 und 2.42) erkennt man, dass auf der rechten Seite der Gleichung noch zusätzliche Terme aus dem am aktuellen Iterationspunkt vorhandenen Residuum $\mathbf{R}_{n+1}^{S,k}(\mathbf{d}_{n+1}^k)$ vorhanden sind. Da aber der aktuelle Iterationspunkt für das Strukturproblem einen Gleichgewichtszustand darstellt, ergibt sich die Lösung des spezifischen Strukturproblems für diese zusätzlichen Terme zu Null. Somit ist auch der Anteil dieser Terme an $\tilde{\mathbf{v}}_{m+1}$ Null und kann damit vernachlässigt werden.

Bewertung: Diese Methode verwendet zur Evaluierung des Krylovraumes für Fluid- als auch Strukturfeld lineare Ersatzprobleme. Durch die Linearisierung ist eine Skalierung von \mathbf{v} nicht nötig. Die Krylovraumevaluierung ist damit auch von geringem numerischem Aufwand, da in jedem Krylovschritt nur zwei lineare Gleichungssysteme zu lösen sind. Die Netzbewegung im Fluidfeld wird umgangen, indem die Randverschiebung in eine Randgeschwindigkeit umgewandelt wird. Während auf der Strukturseite durch Verwendung der aktuellen Tangentensteifigkeit ein klarer Bezug zwischen dem aktuellen Zustand des Feldes und der Sensitivität besteht, ist die Evaluierung der Fluidsensitivität nur von der aktuellen Geometrie des Feldes, aber nicht von den aktuellen Zustandsvariablen abhängig. Damit wird natürlich auch nur der „Added Mass Effect“ abgebildet.

Der Implementierungsaufwand ist stark von den verwendeten Teilfeldlösern abhängig. Während die Tangentensteifigkeit auf Strukturseite bereits zur Verfügung steht, muss auf der Fluidseite das zusätzliche und unabhängige Problem einer Potentialströmung gelöst werden. Dies führt zu einem erhöhten Speicherbedarf und stellt u.U. einen hohen zusätzlichen Aufwand dar.

Der Nachteil dieser Methode ist die hohe Problemabhängigkeit bzw. Abhängigkeit von den spezifischen Eigenschaften des untersuchten Strömungsfeldes. Das spezifische Fluidproblem basiert auf der Annahme, dass der „Added Mass Effect“ den maßgebenden Effekt im Fluidfeld darstellt. Für Strömungen, bei denen dies nicht der Fall ist, folgt damit auch eine geringe Qualität der Krylovraumevaluierung.

Krylovraumevaluierung über ein linearisiertes Modell: Eine weitere Methode für die Krylovraumevaluierung ist die Verwendung eines Modells, welches die Physik des Problems erhält, aber die Nichtlinearitäten der beiden Felder und die Gebietsanpassung

des Fluidfeldes vernachlässigt (In [Dep04] wird dieser Ansatz unter der Bezeichnung FSI-QN 2 eingeführt). Dies wird linearisiertes Modell genannt und basiert auf der Annahme, dass auch das am aktuellen Punkt linearisierte Problem genug Informationen für eine qualitativ hochwertige Evaluierung besitzt.

Das Fluidgebiet wird im aktuellen Zustand $\Omega_{n+1}^{F,k}$ eingefroren und eine reine Euler-Beschreibung benutzt. Die Krylovvektoren werden in eine Randgeschwindigkeit transformiert und dienen damit als Randbedingung für die Geschwindigkeiten und Flüsse am Interface. Alle anderen Randbedingungen werden vom Standard-Feldmodell übernommen. Der Prädiktorschritt für die Geschwindigkeiten wird ausgelassen und eine einzelne PISO-Iteration ausgeführt. Der spezifische Fluidfeldoperator $\mathbf{f}_{m+1}^* = \mathcal{F}^*(\mathbf{v}_m)$ ergibt sich damit zu:

$$\begin{aligned} \nabla \cdot \left(\frac{1}{a_j^*} \nabla p_{n+1}^{*,m+1} \right) &= \nabla \cdot \tilde{\phi}^{*,k} && \text{in } \Omega_{n+1}^{F,k}, \\ p_{n+1}^{*,m+1} &= p_{n+1}^k && \text{auf } \Gamma_{n+1}^{F,k}, \\ \mathbf{u}_{n+1}^{*,m+1} &= \mathbf{u}_{n+1}^k && \text{auf } \Gamma_{n+1}^{F,k}, \\ \mathbf{u}_{n+1}^{*,m+1} &= \frac{\mathbf{v}_m}{\Delta t} && \text{auf } \Gamma_{n+1}^{FSI,k}. \end{aligned} \quad (4.74)$$

Im Strukturfeld wird die aktuelle Tangentensteifigkeit ausgewertet. Der spezifische Strukturfeldoperator $\tilde{\mathbf{v}}_{m+1} = \mathcal{S}^*(\mathbf{f}_{m+1}^*)$ ergibt sich damit zu:

$$\frac{\partial \mathbf{R}_{n+1}^{S,k}(\mathbf{d}_{n+1}^k)}{\partial \mathbf{d}_{n+1}} \tilde{\mathbf{v}}_{m+1} = \mathbf{f}_{m+1}^* \quad \text{in } \Omega_{n+1}^{S,k} \quad (4.75)$$

und bestimmt sich wie im vorherigen Abschnitt. Das Matrix-Vektor-Produkt wird damit angenähert über:

$$\frac{\partial \mathbf{R}_{n+1}^{\Gamma,k}(\mathbf{d}_{n+1}^{\Gamma,k})}{\partial \mathbf{d}^{\Gamma}} \mathbf{v}_m = \mathbf{v}_m - \tilde{\mathbf{v}}_{m+1}. \quad (4.76)$$

Bewertung: Diese Methode erhält im Ansatz alle relevanten physikalischen Eigenschaften des Strömungs- und Strukturfeldes und ist von geringem numerischen Aufwand. Bei jeder Krylovraumevaluation sind nur zwei lineare Gleichungssysteme zu lösen. Allerdings führt die Vernachlässigung der Netzbewegung und des ALE-Ansatzes zu falschen Matrix-Koeffizienten im Fluidfeld und das Interface wird damit nicht mehr als Wand behandelt. Der Einfluss dieser Fehler ist stark problemabhängig und schwer abzuschätzen. Des Weiteren ist die Evaluierungsqualität über eine Linearisierung nur hoch, wenn die nichtlinearen Effekte vergleichsweise gering sind. Daher haben Zeitschrittweite und Kopplungsgrad großen Einfluss auf das Ergebnis.

4.8.2. Quasi-Newton-Verfahren

Das hier vorgestellte Quasi-Newton-Verfahren ist ein relativ neu entwickeltes Verfahren ([Vie06, VLDV07, DBV09]). Es basiert auf der Veränderung der Residuenvektoren am Interface und löst ein Minimierungsproblem, mit welchem die Inverse der Jakobi-Matrix approximiert wird. Ein besonderer Vorteil ist die Möglichkeit, Standard-Feldlöser zu verwenden.

4.8.2.1. Startprozedur

Der Kern des Algorithmus basiert auf der Behandlung von Veränderungen der Residuenvektoren am Interface. Für seine Anwendung ist eine Startprozedur notwendig, welche das erste Inkrement bereitstellt. Diese Startprozedur besteht aus einem Prädiktorschritt gefolgt von einem konstanten Relaxationsschritt. Der Prädiktorschritt kann mittels einer der in Abschnitt 4.4 vorgestellten Methoden ausgeführt werden. Der konstante Relaxationsschritt wird mittels des in Abschnitt 4.6 vorgestellten Verfahrens bestimmt. Nach diesen zwei Schritten ist das erste Inkrement des Interface-Residuumsvektors im aktuellen Zeitschritt bekannt: $\Delta \mathbf{R}_{n+1}^{\Gamma,1} = \mathbf{R}_{n+1}^{\Gamma,2} - \mathbf{R}_{n+1}^{\Gamma,1}$.

4.8.2.2. Algorithmusbeschreibung

Nach der Startprozedur werden die Quasi-Newton-Iterationen ausgeführt. Die grundlegende Idee des Verfahrens ist es, dass sich das Interfaceresiduum in der nächsten Iteration zu Null ergibt:

$$\mathbf{R}_{n+1}^{\Gamma,k+1} = 0. \quad (4.77)$$

Die gewünschte Veränderung des Residuums ist also:

$$\Delta \mathbf{R}_{n+1}^{\Gamma,k} = 0 - \mathbf{R}_{n+1}^{\Gamma,k}. \quad (4.78)$$

Diese Veränderung wird approximiert als Linearkombination der bekannten Residueninkremente:

$$\Delta \mathbf{R}_{n+1}^{\Gamma,k} = 0 - \mathbf{R}_{n+1}^{\Gamma,k} \approx \sum_{i=1}^{k-1} \alpha_i^k \Delta \mathbf{R}_{n+1}^{\Gamma,i}. \quad (4.79)$$

Dieses Gleichungssystem ist für $k < n_{\text{dof}}^{\Gamma}$ überbestimmt, was in der Regel immer der Fall ist. Die Linearkoeffizienten α_i^k lassen sich über das Lösen eines Minimierungsproblems bestimmen:

$$\alpha^k = \arg \min \left| \mathbf{R}_{n+1}^{\Gamma,k} + \sum_{i=1}^{k-1} \alpha_i^k \Delta \mathbf{R}_{n+1}^{\Gamma,i} \right|_2, \quad (4.80)$$

womit Gleichung (4.79) approximativ erfüllt wird. Für die Bestimmung des Verschiebungsinkrements werden als erstes die Koeffizienten auf die zugehörigen relaxierten Verschiebungsinkremente übertragen:

$$\Delta \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} = \sum_{i=1}^{k-1} \alpha_i^k \Delta \tilde{\mathbf{d}}_{n+1}^{\Gamma,i}, \quad (4.81)$$

mit $\Delta \tilde{\mathbf{d}}_{n+1}^{\Gamma,i} = \tilde{\mathbf{d}}_{n+1}^{\Gamma,i} - \tilde{\mathbf{d}}_{n+1}^{\Gamma,i-1}$. Die Beziehung zwischen \mathbf{d}^Γ und $\tilde{\mathbf{d}}^\Gamma$ ist gegeben durch $\mathbf{R}^\Gamma = \tilde{\mathbf{d}}^\Gamma - \mathbf{d}^\Gamma$ und damit: $\mathbf{d}^\Gamma = \tilde{\mathbf{d}}^\Gamma - \mathbf{R}^\Gamma$. Für die zugehörige inkrementelle Betrachtung folgt daraus:

$$\Delta \mathbf{d}_{n+1}^{\Gamma,k} = \Delta \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \Delta \mathbf{R}_{n+1}^{\Gamma,k} \quad (4.82)$$

als Regel für die Bestimmung des neuen Verschiebungsinkrementes am Interface.

Der detaillierte Ablauf des Verfahrens ist in Algorithmus 11 gegeben.

Dabei ist zu beachten, dass die Matrizen V^k und W^k benutzt werden, um die Residuen- und relaxierten Verschiebungsinkremente zu speichern:

$$V^k = \begin{bmatrix} \Delta \mathbf{R}^{\Gamma,k} & \Delta \mathbf{R}^{\Gamma,k-1} & \dots & \Delta \mathbf{R}^{\Gamma,1} \end{bmatrix}, \quad (4.83)$$

$$W^k = \begin{bmatrix} \Delta \tilde{\mathbf{d}}^{\Gamma,k} & \Delta \tilde{\mathbf{d}}^{\Gamma,k-1} & \dots & \Delta \tilde{\mathbf{d}}^{\Gamma,1} \end{bmatrix}. \quad (4.84)$$

Die Anzahl an Spalten q ist gegeben durch: k , die Anzahl Zeilen p entspricht der Anzahl an Interfacefreiheitsgraden n_{dof}^Γ .

Um das Minimierungsproblem auf effiziente Art und Weise zu lösen, wird mittels einer Householder-Transformation eine QR-Zerlegung von V^k bestimmt, welche die Rechtecksmatrix $V^k \in \mathbb{R}^{p \times q}$ in eine orthogonale Matrix $Q \in \mathbb{R}^{p \times q}$ und eine obere Dreiecksmatrix $R \in \mathbb{R}^{q \times q}$ aufteilt:

$$V^k = Q^k R^k. \quad (4.85)$$

Damit gilt:

$$-\mathbf{R}^{\Gamma,k} = V^k \alpha^k = Q^k R^k \alpha^k \quad (4.86)$$

$$R^k \alpha^k = Q^{k-1} \left(-\mathbf{R}^{\Gamma,k} \right) \quad (4.87)$$

$$= Q^{kT} \left(-\mathbf{R}^{\Gamma,k} \right). \quad (4.88)$$

Da R^k eine obere Dreiecksmatrix ist, lassen sich die Koeffizienten α_i^k durch einfaches Rückwärtseinsetzen mit sehr geringem Aufwand bestimmen.

4.8.2.3. Vergleich mit Jakobi-freien Newton-Krylov-Verfahren

Bei der Betrachtung der Vorgehensweise des Quasi-Newton-Verfahrens ergeben sich einige Ähnlichkeiten mit den im vorherigen Abschnitt 4.8.1 vorgestellten Jakobi-freien Newton-Krylov-Verfahren. So wird in beiden Verfahren das neue Verschiebungsinkrement als

Algorithmus 11 Quasi-Newton-Kopplungsalgorithmus

```

1: for  $t = 0$  bis  $t = t_{tot}$  do
2:    $k=1$ 
3:   Prädiktorschritt:
4:    $\mathbf{d}_{n+1}^{\Gamma,1} = \mathbf{d}_{n+1}^{\Gamma,P}$ 
5:    $\tilde{\mathbf{d}}_{n+1}^{\Gamma,1} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,1})$ 
6:    $\mathbf{R}_{n+1}^{\Gamma,1} = \tilde{\mathbf{d}}_{n+1}^{\Gamma,1} - \mathbf{d}_{n+1}^{\Gamma,1}$ 
7:   konstanter Relaxationsschritt:
8:    $\mathbf{d}_{n+1}^{\Gamma,2} = \mathbf{d}_{n+1}^{\Gamma,1} + \omega \mathbf{R}_{n+1}^{\Gamma,1}$ 
9:   Quasi-Newton-Iterationen:
10:  while (keine Konvergenz) do
11:     $\tilde{\mathbf{d}}_{n+1}^{\Gamma,k} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}_{n+1}^{\Gamma,k})$ 
12:     $\mathbf{R}_{n+1}^{\Gamma,k} = \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \mathbf{d}_{n+1}^{\Gamma,k}$ 
13:    if  $\left( \left\| \mathbf{R}_{n+1}^{\Gamma,k} \right\|_2 < \epsilon^\Gamma \right)$  then
14:       $\mathbf{d}_{n+1}^\Gamma = \mathbf{d}_{n+1}^{\Gamma,k}$ 
15:      konvergiert, gehe zu nächstem Zeitschritt
16:    else
17:       $\Delta \mathbf{R}_{n+1}^{\Gamma,k} = \mathbf{R}_{n+1}^{\Gamma,k} - \mathbf{R}_{n+1}^{\Gamma,k-1}$ 
18:       $\Delta \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} = \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} - \tilde{\mathbf{d}}_{n+1}^{\Gamma,k-1}$ 
19:       $V^k = \begin{bmatrix} \Delta \mathbf{R}_{n+1}^{\Gamma,k} & \Delta \mathbf{R}_{n+1}^{\Gamma,k-1} & \dots & \Delta \mathbf{R}_{n+1}^{\Gamma,1} \end{bmatrix}$ 
20:       $W^k = \begin{bmatrix} \Delta \tilde{\mathbf{d}}_{n+1}^{\Gamma,k} & \Delta \tilde{\mathbf{d}}_{n+1}^{\Gamma,k-1} & \dots & \Delta \tilde{\mathbf{d}}_{n+1}^{\Gamma,1} \end{bmatrix}$ 
21:      QR-Zerlegung:  $V^k = Q^k R^k$ 
22:       $R^k \alpha^k = Q^{kT} (-\mathbf{R}_{n+1}^{\Gamma,k}) \rightarrow \alpha^k$ 
23:       $\Delta \mathbf{d}_{n+1}^{\Gamma,k} = W^k \alpha^k + \mathbf{R}_{n+1}^{\Gamma,k}$ 
24:       $\mathbf{d}_{n+1}^{\Gamma,k+1} = \mathbf{d}_{n+1}^{\Gamma,k} + \Delta \mathbf{d}_{n+1}^{\Gamma,k}$ 
25:    end if
26:     $k \leftarrow k + 1$ 
27:  end while
28:   $t_{n+1} = t_n + \Delta t_n$ 
29:   $n \leftarrow n + 1$ 
30: end for

```

Linearkombination bestimmt (Gleichungen (4.82) und (4.81) bzw. (4.55)) und die Linearkoeffizienten ergeben sich aus der Lösung eines Minimierungsproblems (Gleichung (4.80) bzw. (4.56)).

Um Unterschiede festzustellen, ist es notwendig, noch einmal den grundsätzlichen Ansatz der beiden Verfahren darzustellen. Das Newton-Krylov-Verfahren evaluiert die Antwort des gekoppelten Problems am aktuellen Iterationspunkt in einem durch die Krylovvektoren aufgespannten Raum. Das neue Verschiebungsinkrement wird dann als Linearkombination aus den lokal erzeugten Antworten gebildet. Das Quasi-Newton-Verfahren verwendet die am aktuellen und an den vorherigen Iterationspunkten vorhandene Information über das Residuumsinkrement. Während das Newton-Krylov-Verfahren also zur Formulierung des Minimierungsproblems nur am aktuellen Iterationspunkt erzeugte Informationen verwendet, kombiniert das Quasi-Newton-Verfahren die Informationen aus mehreren Iterationspunkten.

Die beiden Ansätze weisen also gewisse Ähnlichkeiten auf, sind aber bezogen auf die Grundidee verschieden.

4.8.2.4. Vergleich mit weiteren Quasi-Newton-Verfahren

Das hier vorgestellte Verfahren wird von den Erfindern als Quasi-Newton-Verfahren bezeichnet. Von den üblicherweise als Quasi-Newton-Verfahren bezeichneten Methoden unterscheidet es sich aber. Diese werden vornehmlich in der Strukturoptimierung eingesetzt und basieren auf einer Approximation der Inversen der Jakobi-Matrix ([LY08]). Die üblichen Verfahren beginnen mit einer Startapproximation der Jakobi-Matrix, wie z.B. über die Einheitsmatrix, welche sukzessive über die Iterationen verbessert wird. Die Verfahren unterscheiden sich hierbei nach der Art der Verbesserung. Die bekanntesten Verfahren sind die Methoden nach Davidon-Fletcher-Powell (DFP) und nach Broyden-Fletcher-Goldfarb-Shanno (BFGS). Diese Verfahren approximieren die vollständige Inverse der Jakobi-Matrix. Das hier im Rahmen der Fluid-Struktur-Wechselwirkung verwendete Quasi-Newton-Verfahren basiert ebenfalls auch auf einer sukzessiven Verbesserung einer Startapproximation. Allerdings wird an keiner Stelle die Jakobi-Matrix direkt verwendet, sondern das neue Verschiebungsinkrement aus einer Linearkombination von Vektoren bestimmt. Damit ist auch der Speicheraufwand dieser Methode deutlich geringer, da keine globalen Matrizen vorgehalten werden müssen.

4.8.2.5. Wiederverwendung von Informationen

Der Algorithmus erlaubt die Wiedererwendung von Informationen aus vorherigen Zeitschritten. Dies geschieht unter der Annahme, dass die Informationen über die Kombination aus Residuumsverringern und Verschiebungsänderung auch zu späteren Zeitpunkten relevant sind. Dazu werden die Matrizen V^k und W^k noch mit Vektoren aus vorherigen Zeitschritten angefüllt und das Minimierungsproblem (4.80) entsprechend erweitert. Eine wichtige Frage ist hierbei, wie viele Level wiederverwendet werden sollen. Da sich für Vektoren, die ohne Relevanz sind, die zugehörigen Linearkoeffizienten zu nahezu Null ergeben, können prinzipiell beliebig viele Level aufgenommen werden. Dabei sind drei

Einschränkungen zu beachten:

1. Durch die Nichtlinearität des gekoppelten Problems verringert sich die Relevanz von Vektoren, je weiter sie zurückliegen.
2. Die einzelnen Vektoren müssen voneinander linear unabhängig sein, da sich ansonsten keine eindeutige Lösung bestimmen lässt.
3. Der Speicheraufwand steigt für jeden zusätzlichen Vektor, welcher aufgenommen wird. Daraus folgt, dass nicht beliebig viele Vektoren hinzugenommen werden sollen und diese auf lineare Unabhängigkeit überprüft werden müssen. Dies wird im numerischen Beispiel in Abschnitt 6.1.5 näher untersucht.

4.8.2.6. Verzicht auf den Relaxationsschritt

Durch die Startprozedur sind in jedem Zeitschritt zwei Kopplungsiterationen notwendig, bevor mit den Quasi-Newton-Iterationen begonnen werden kann. Der Relaxationsfaktor muss dabei sehr konservativ gewählt werden, damit in diesem Schritt keine Divergenz auftritt und das gesamte Verfahren zum Abbruch bringt. Dies ist wenig effizient, da so im Relaxationsschritt das Residuum in der Regel nur geringfügig minimiert wird. Sind insgesamt wenig Kopplungsiterationen notwendig, fällt dieser Effekt um so stärker auf.

Eine Möglichkeit, dies zu umgehen, ist es, nach dem zwingend notwendigen Prädiktor-schritt auf den konstanten Relaxationsschritt zu verzichten und das Minimierungsproblem mit Residueninkrementen aus Iterationen des vorherigen Zeitschritts aufzustellen. Sind diese Inkremente noch für den aktuellen Zustand relevant, kann das Residuum möglicherweise deutlich stärker reduziert werden. Dieses Vorgehen ist ab dem zweiten Zeitschritt möglich. Allerdings wird dadurch die Basis an verfügbaren Inkrementen nicht automatisch erweitert, was zu Konvergenzproblemen führen kann. Diese Option wird im numerischen Beispiel in Abschnitt 6.1.5 näher untersucht.

Bewertung: Das von [Vie06] 2006 eingeführte Quasi-Newton-Verfahren verwendet den einfachen Ansatz, basierend auf der Veränderung von Residuum und Verschiebung am Interface, eine neue Lösung zu approximieren. Dazu muss ein Minimierungsproblem gelöst werden, was über die QR-Zerlegung durch einfaches Rückwärtseinsetzen geschehen kann. Auch die Wiederverwendung von Informationen aus vorherigen Iterationen ist ohne großen Aufwand möglich. Ein großer Vorteil des Verfahrens besteht darin, dass Standardlösungsansätze in den Teilfeldern verwendet werden können.

4.9. Zusammenfassung

In diesem Kapitel wird die Fluid-Struktur-Wechselwirkung als gekoppeltes nichtlineares Problem behandelt. Nach einer Problemdefinition und -beschreibung wird die Grundform eines Algorithmus zur partitionierten und impliziten Lösung gezeigt. Darauf aufbauend werden drei wichtige Bereiche vorgestellt. Als Erstes die Konvergenzkriterien, um eine Lösung des gekoppelten Problems zu identifizieren. Als Zweites die Interfaceprädiktoren, um eine Initiallösung bestimmen zu können. Als Drittes werden Kopplungsalgorithmen vorgestellt. Nach einem Überblick der aktuell gängigsten verwendeten Methoden werden einzelne Methoden ausgewählt und in den folgenden Abschnitten im Detail vorgestellt. Als Vertreter von Methoden, welche Standardlösungsansätze in den Teilfeldern verwenden, werden die Fixpunktiteration mit dynamischer Relaxation, die Vektorextrapolation und das Quasi-Newton-Verfahren ausgewählt. Als Vertreter von Methoden, welche eine Anpassung der Feldlöser benötigen, werden verschiedene Jakobi-freie Newton-Krylov-Verfahren ausgewählt. Die Interfaceprädiktoren und Kopplungsalgorithmen werden im numerischen Beispiel in Abschnitt 6.1 auf Effizienz und Stabilität untersucht.

5. Softwareumgebung

In Kapitel 5 wird die im Rahmen dieser Arbeit entstandene Softwareumgebung („Computational Framework“) näher vorgestellt. Für die effiziente Simulation großer Probleme der Fluid-Struktur-Wechselwirkung sind effiziente Lösungsfeldalgorithmen und Kopplungsalgorithmen nur ein Baustein. Für ein leistungsfähiges und effizientes Gesamtsimulationskonzept sind auch Softwareaspekte bzw. die softwaremäßige Umsetzung von großer Wichtigkeit. Zu diesen Aspekten zählen vor allem:

- **Massiv parallele Simulation:** Insbesondere die Lösung des Fluidfeldes ist numerisch sehr aufwendig. Die komplexe Physik, v.a. im turbulenten Fall, muss adäquat abgebildet werden, um aussagekräftige Ergebnisse zu erzielen. Dazu müssen Gleichungssysteme mit vielen Millionen Unbekannten gelöst werden, was mittels massiv paralleler Simulation auf Hochleistungsrechnern geschieht.
- **Abbildung geometrisch komplexer Modelle:** Zur Simulation praxisorientierter Fragestellungen muss die Software in der Lage sein, auch dreidimensionale Modelle mit komplexer Geometrie abbilden zu können.
- **Modularität und Adaptivität:** Um einen möglichst breiten Simulationsbereich abdecken zu können, muss die Software hochmodular gestaltet sein. Für die Optimierung eines gekoppelten Problems sind andere Aspekte von Bedeutung als für die Untersuchung langzeitdynamischen Verhaltens. Auch muss die Möglichkeit bestehen, Neuentwicklungen bei Feldlösern und Kopplungsalgorithmen mit geringem Aufwand integrieren zu können.

Als Ausgangspunkt für die Softwareentwicklung dient die Wahl eines partitionierten Lösungsansatzes. Damit werden die Teilfelder durch separate Programme abgebildet. Die Verwendung eigenständiger Programme erfordert die Definition einer Schnittstelle für den Informationsaustausch. Wird die Schnittstelle möglichst allgemein definiert, erfüllt der partitionierte Ansatz per se die Forderung nach Modularität und Adaptivität, da die einfache Austauschbarkeit von Programmen gewährleistet ist. Sollten sich für einzelne Bereiche neue Entwicklungen ergeben, die einen leistungsfähigeren Ansatz verwenden, so ist in einer partitionierten Umgebung die Anpassung mit relativ wenig Aufwand möglich. Ein weiteres Vorteil des partitionierten Ansatzes ist es, existierende Software wiederzuverwenden. Damit kann auf bereits geleistete Entwicklungsarbeit zurückgegriffen werden.

Für die softwaremäßige Umsetzung wird ein Drei-Programme-Ansatz gewählt, welcher auch die Aufspaltung in die Teildisziplinen gekoppelter Berechnungen widerspiegelt. Zwei der Programme sind die beiden Teilfeldlöser für Fluid und Struktur, das dritte

Programm ist für die Abbildung der Wechselwirkung verantwortlich. Durch diese Aufspaltung sind die methodischen Blöcke gekapselt und es ergibt sich eine besonders hohe Modularität. In einem ersten Schritt sind entsprechend leistungsfähige Teilfeldlöser zu wählen bzw. vorhandene weiterzuentwickeln. Im einem zweiten Schritt werden die Teilfeldlöser in das Gesamtkonzept für die gekoppelte Berechnung eingebunden. Hierbei muss sichergestellt werden, dass deren Leistungsfähigkeit nicht eingeschränkt wird.

Als gutes Beispiel für die Effektivität des gewählten Ansatzes dient der Lehrstuhl für Statik, an dem diese Arbeit entstand. Zum Zeitpunkt der Fertigstellung dieser Arbeit wurden durch Austausch von Feldlösern innerhalb verschiedener Projekte bereits vier verschiedene Fluidlöser verwendet (die kommerzielle Software CFX, die Open Source-Software OpenFOAM und die universitären Codes MGLET und FASTEST). Jeder dieser Codes behandelt einen spezifischen Aspekt besonders effizient und es wurden verschiedenste Simulationsarten durchgeführt (Fluid-Struktur-Wechselwirkung, Fluid-Fluid-Wechselwirkung, Optimierung oberflächengekoppelter Probleme).

Für die in dieser Arbeit vorgestellten Probleme der Fluid-Struktur-Wechselwirkung wird für die Simulation des Strukturfeldes das institutseigene Programm *Carat++* verwendet, die Wechselwirkung wird über die institutseigene Kopplungssoftware *CoMA* abgebildet und das Fluidfeld wird mittels der Open Source-Software *OpenFOAM* simuliert, welche angepasst und erweitert wurde. Die Entwicklung und Anwendung dieser Softwareumgebung wird u.A. auch in [GKWB07b, GKWB07a, GKI⁺09, GKUW09, GKBW09] besprochen.

In den folgenden Abschnitten werden die Teilfeldlöser *Carat++* und *OpenFOAM* kurz angesprochen, den Schwerpunkt bildet die Kopplungssoftware *CoMA*, da deren Konzeption und Entwicklung ein zentraler Bestandteil dieser Arbeit ist.

5.1. Strukturfeldlöser Carat++

Der Strukturfeldlöser *Carat++* wird am Lehrstuhl für Statik seit 2008 entwickelt. Zentrales Designkonzept der Simulationssoftware bildet die Objektorientierung in C++. Sie ermöglicht die Entwicklung generischer, wiederverwendbarer und einfach erweiterbarer Software, was gerade die Entwicklung fortgeschrittener Algorithmen für komplexe, nicht-lineare Probleme deutlich vereinfacht. Des Weiteren zeichnet sich die Software durch Plattformunabhängigkeit, numerische Effizienz, sowie Eignung für große und komplexe Problemstellungen, z.B. für industrielle Anwendungen [FFMB10, FFB10], aus.

Wie seit Anfang der 90er-Jahre bekannt ist, sind außerdem gerade objektorientierte Programme besonderes geeignet für parallele Programmierung [Mil91]. So sind für eine optimale Skalierung alle Softwarekomponenten von Carat++ parallelisiert: Einlesen der Modelldaten, Gebietszerlegung (basierend auf ParMETIS [KK97]), Simulationsberechnung inklusive Gleichungslösung, sowie Postprocessing. Dabei wird die Methode der geometrischen Parallelisierung gewählt (Domain Decomposition), mit knotenweiser Partitionierung und Interfaceelementreihen (überlappende Zerlegung).

Innerhalb dieser Arbeit werden mit Carat++ strukturdynamische Vorgänge mit großen Verformungen simuliert. Für die Abbildung des dynamischen Verhaltens wird das implizite Zeitintegrationsverfahren Generalized- α benutzt und mit dem als „angepasster Feldlöser“ benannten Verfahren erweitert. Die großen Verformungen werden mit einer geometrisch nichtlinearen Formulierung abgebildet. Für die Diskretisierung des in Abschnitt 6.1 gezeigten numerischen Beispiels werden Schalenelemente benutzt, für das Beispiel in Abschnitt 6.2 Schalen-, Membran- und Seilelemente. Das zweite Beispiel stellt ein vorgespanntes Membrantragwerk dar, dessen Gleichgewichtsform über einen Formfindungsalgorithmus bestimmt wird.

5.1.1. Verwendung geometrisch reduzierter Strukturmodelle

Die verwendeten Schalen- und Membranelemente weisen als Folge der Diskretisierung eine Geometriereduzierung der Dickendimension auf. Die Geometrie des numerischen Modells ist damit eine Abbildung der Mittelfläche der realen Struktur. Für oberflächengekoppelte Simulationen wird aber eine geometrische Beschreibung der Oberfläche in Form eines diskreten Netzes benötigt, um die Umströmung und damit die zugehörigen Variablen physikalisch korrekt darzustellen. Die Dimensionsreduzierung bei Diskretisierung des Feldmodells muss also für eine gekoppelte Berechnung zumindest teilweise wieder rückgängig gemacht werden.

Der in Carat++ verwendete Ansatz soll hier anhand eines Beispiels kurz dargestellt werden. Für weitere Details wird auf die entsprechende Literatur [Fis09] hingewiesen. Als Beispiel wird das geometrische Modell eines Zylinders gewählt, der eine gewisse Dicke t aufweist (siehe linkes Bild in Abbildung 5.1). Werden für eine Diskretisierung 4-knotige Schalenelemente gewählt, so wird die Mittelfläche des Zylinders diskretisiert. Dazu werden Knoten entlang der Mittelfläche angeordnet und über direkte Verbindungen Flächen erzeugt (siehe rechtes Bild in Abbildung 5.1). Eine vierseitige Fläche mit vier Knoten bildet die geometrische Repräsentation eines Schalenelementes. Hierbei spricht man von

einer geometrischen Reduzierung, da die diskrete Geometrie die Information über die Zylinderdicke als auch die gekrümmte Oberfläche nicht mehr enthält.

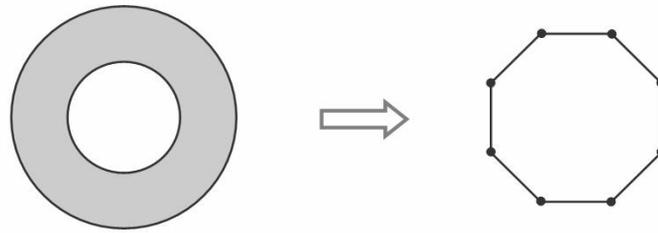


Abbildung 5.1.: Geometrische Reduktion durch Diskretisierung - Schnitt durch einen Zylinder

Um ein solches Modell innerhalb einer oberflächengekoppelten Simulation einsetzen zu können, müssen diese zwei Aspekte näher betrachtet werden:

1. Das in dieser Arbeit verwendete diskrete Fluidmodell besitzt ebenfalls keine gekrümmte Oberfläche. Der Verlust der Krümmungsinformation durch die Facettierung tritt also in beiden Feldern auf und lässt sich auch nicht umgehen.
2. Fehlt in der geometrischen Repräsentation des Strukturmodells die Dicke, kann die Umströmung und damit auch alle oberflächenrelevanten Strömungsvorgänge nicht korrekt dargestellt werden. Das Oberflächenmodell des Strukturfeldes muss also im allgemeinen Fall die Information über die Strukturdicke enthalten.

Der in Carat++ verwendete Ansatz zur Erstellung eines Oberflächenmodells basiert auf einer Projektion der diskretisierten Mittelfläche in Richtung Ober- und Unterseite der Fläche. Hierzu wird die Topologie des Mittelflächenetzes beibehalten, es werden nur die Knotenkoordinaten angepasst. Dazu wird an jedem Knoten ein Vektor \mathbf{n}_m normal zur Oberfläche gebildet, und der Knoten in Richtung dieses Normalenvektors um $\pm t/2$ verschoben. Der Knotennormalenvektor soll an diesem Punkt eine gemittelte Normale der facettierten Oberfläche darstellen. Dazu wird eine Mittelung aller an den Knoten angrenzende Flächennormalen durchgeführt. Dies ist in Abbildung 5.2 dargestellt. Hierbei wird am Punkt P_1 die gemittelte Normale \mathbf{n}_m aus den angrenzenden Flächennormalen \mathbf{n}_1 und \mathbf{n}_2 gebildet und daraus die Punkte an der oberen K_{11} und an der unteren Oberfläche K_{12} erstellt.

Die Verwendung des Projektionsansatzes hat den Vorteil, das bereits vorhandene Strukturmodelle aus Einzelfeldsimulationen ohne größere Anpassung für eine oberflächengekoppelte Simulation verwendet werden können. Auch kann das so erzeugte Oberflächennetz des Strukturfeldes zur Netzerzeugung im Fluidfeld verwendet werden.

Nach der Erstellung des Oberflächennetzes stellt sich die Frage, wie die an der Oberfläche auftretenden physikalischen Felder behandelt werden. Bei einer Simulation vom Typ Fluid-Struktur-Wechselwirkung sind dies Verschiebungen und Kräfte:

- Nach der Lösung des Strukturfeldes sind an den Knoten der Mittelfläche Verschiebungen und Verdrehungen bekannt. Beim Transfer von der Mittelfläche zur

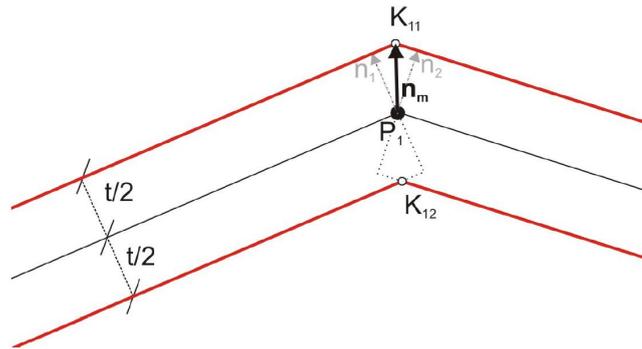


Abbildung 5.2.: Erzeugung des Oberflächennetzes mittels Projektion

Oberfläche werden nur die Verschiebungen der Mittelfläche berücksichtigt, die Verdrehungen werden vernachlässigt. Unter der Annahme einer dünnen Schale ist der dadurch entstehende Fehler vernachlässigbar.

- Die an der Oberfläche wirkenden Kräfte sind nach dem Lösen des Fluidfeldes bekannt. Für den Transfer der Kräfte von der Oberfläche auf die Knoten der Mittelfläche werden die Kräfte von Ober- und Unterseite aufsummiert. Eine aus der Dicke resultierende Momentenwirkung der Kräfte wird vernachlässigt. Unter der Annahme einer dünnen Schale ist der dadurch entstehende Fehler vernachlässigbar.

5.2. Fluidfeldlöser OpenFOAM

Der Fluidfeldlöser *OpenFOAM* ist eine frei verfügbare und quellcodeoffene Software. Sie wurde ursprünglich in den 90er Jahren am Imperial College in London entwickelt ([WTJF98]) und wird mittlerweile von der Firma OpenCFD Ltd. weiter entwickelt und unter der GPL Version 2-Lizenz veröffentlicht ([ope]).

Die Software ist objektorientiert und in C++ implementiert. Das Grundkonzept von OpenFOAM ist die Bereitstellung einer großen Anzahl von Bibliotheken, welche bestimmte Funktionalitäten und Methoden bieten. Darauf basierend werden Top-Level-Löser entwickelt, die für einzelne spezifische Problemstellungen geeignet sind und nur auf den dazu benötigten Teil der Bibliotheken zugreifen. In den Lösern selbst sind nur die Steuerungsalgorithmen der obersten Ebene abgebildet. Damit sind diese sehr übersichtlich und leicht zu verstehen.

Innerhalb dieser Arbeit werden mit OpenFOAM inkompressible fluiddynamische Vorgänge auf bewegten Netzen simuliert. Dafür wurden eigene Top-Level-Löser entwickelt, welche die folgenden Eigenschaften aufweisen: Als Diskretisierungsverfahren wird die Finite-Volumen-Methode benutzt. Alle Terme werden in Raum und Zeit mit Operatoren abgebildet, die eine Genauigkeit von zweiter Ordnung aufweisen. Zur Lösung der nicht-linearen Navier-Stokes-Gleichungen wird das PISO-Druckkorrekturverfahren eingesetzt und mit dem als „angepasster Feldlöser“ benannten Verfahren erweitert. Die Netzbewegung wird über eine Laplace-Methode mit variablem Diffusionskoeffizient berechnet. Tur-

bulente Strömungen werden mit dem RANS-Ansatz unter Verwendung eines $k-\omega-SST$ Turbulenzmodells und zonalen Wandfunktionen simuliert. Das Parallelisierungskonzept besteht aus einer geometrischen Gebietszerlegung, wodurch massiv parallele Simulationen möglich sind. Für die Verwendung innerhalb des „Computational Framework“ wurde das parallele Kommunikationskonzept von OpenFOAM erweitert, um mit CoMA Informationen austauschen zu können.

5.3. Kopplungssoftware CoMA

Der Name *CoMA* bedeutet „Coupling for Multiphysics Analysis“. CoMA ist eine eigenständige Software, die in der partitionierten Simulation oberflächengekoppelter Probleme eingesetzt wird. Typischerweise sind als Ausgangspunkt entsprechende Einzelfeldlöser vorhanden. Das Ziel von CoMA ist es, alle Erweiterungen und Methoden, die für eine oberflächengekoppelte Simulation notwendig sind, auf Basis der Einzelfeldlöser bereitzustellen. CoMA ist dabei der zentrale Prozess, der die komplette Simulation steuert, und an den die Teilfeldlöser angeschlossen sind. Anzumerken ist, dass die Teilfeldprozesse nur Informationen mit CoMA austauschen, aber nicht untereinander, was in Bild 5.3 veranschaulicht wird. Als zu Grunde liegende Hardwareplattform werden Parallelrechner mit verteiltem Speicher („Distributed Memory Parallel Computer“) vorausgesetzt, da diese sich im Bereich der numerischen Simulation großer Probleme als Standard etabliert haben.

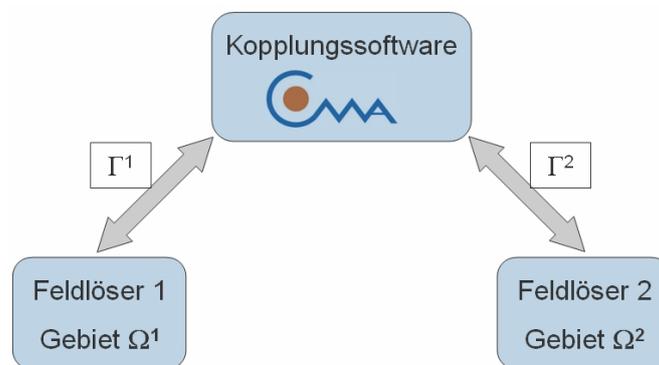


Abbildung 5.3.: Softwareumgebung

CoMA wird seit mehreren Jahren am Lehrstuhl für Statik entwickelt. Es existieren zwei Varianten, die sich in einigen Aspekten unterscheiden. Eine Variante wird beispielsweise in den Arbeiten [IKL⁺07, Kup09] verwendet, die weitere Variante wird in dieser Arbeit verwendet und soll im Folgenden detailliert vorgestellt werden.

Um dem Anspruch einer möglichst allgemeinen Verwendbarkeit gerecht zu werden, erfüllt die Software die folgenden Eigenschaften:

- Keine Beschränkung auf ein spezifisch abzubildendes, gekoppeltes Problem, wie z.B. Fluid-Struktur-Wechselwirkung, oder einen Kopplungsalgorithmus, wie z.B. implizite Kopplung, sondern Eignung für allgemeine Oberflächenkopplung.
- Eignung für die Kopplung von kommerzieller sowie quellcodeoffener Software.
- Verschiedene Kommunikations- bzw. Datenaustauschkonzepte, um den Möglichkeiten der Teilfeldlöser und der zu Grunde liegenden Hardwareplattform zu entsprechen.

- Möglichst geringe Eingriffe in die Teilfeldlöser, um den Adaptierungsaufwand gering zu halten.
- Keine Einschränkung der Parallelisierungskonzepte der Teilfeldlöser.
- Optionalität aller Steuerungsmöglichkeiten, also Simulationskontrolle durch CoMA und Steuerung von Black-Box-Feldlösern oder Simulationskontrolle durch Feldlöser und CoMA als reine Schnittstelle für den Datentransfer.
- Wiederverwendbarkeit existierender Teilfeldlöser, ohne für die Kopplung spezifische und aufwendige Erweiterungen notwendig zu machen.

Aus programmiertechnischer Sicht ergeben sich damit hohe Ansprüche an die Modularität und Erweiterbarkeit der Software, welche am Besten durch eine objektorientierte Programmiersprache erfüllt werden kann. Die Wahl fiel auf C++, da:

1. die in diesem Projekt benutzten Teilfeldlöser ebenfalls auf C++ basieren und sich damit eine einheitliche Programmierumgebung bietet,
2. Compiler für alle in Frage kommenden Hardwareplattformen existieren und
3. bei entsprechender Programmierung C++ hocheffizienten und schnellen Code ermöglicht.

Im Verlaufe dieser Arbeit wurde CoMA innerhalb verschiedenster Projekte eingesetzt. Dies unterstreicht, dass das Ziel der Entwicklung einer allgemein verwendbaren Kopplungssoftware erreicht wurde. In Tabelle 5.1 sind die bisher unter Verwendung von CoMA bearbeiteten Projekte aufgeführt.

Simulationstyp	Feldlöser 1	Feldlöser 2
Fluid-Struktur-Wechselwirkung	<i>FCarat</i>	CFX
Fluid-Struktur-Wechselwirkung	<i>Carat</i> ++	OpenFOAM
Fluid-Struktur-Wechselwirkung	<i>Carat</i> ++	FASTEST
Fluid-Fluid-Wechselwirkung	OpenFOAM	MGLET
Optimierung gekoppelter Probleme	<i>Carat</i> ++	OpenFOAM

Tabelle 5.1.: Verwendung von CoMA

In den folgenden Unterkapiteln werden die wichtigsten Teilaspekte und Programmstrategien von CoMA näher vorgestellt. Dazu zählen:

- Parallelisierungsstrategie,
- Datentransfer zwischen Oberflächennetzen,
- Simulationssteuerung und Kopplungsalgorithmen.

Anmerkung: Warum ist CoMA eine eigenständige Software und keine Bibliothek?

Gerade in der objektorientierten Programmierung werden Funktionalitätserweiterungen für Programme gerne über das Anbinden externer Bibliotheken realisiert. Diese Möglichkeit bestünde auch für die Funktionalitäten, die CoMA bietet. Innerhalb einer gekoppelten Berechnung zusätzlich zu den Feldlösern noch eine weitere Software einzuführen, erschwert auf den ersten Blick die Handhabung. Dennoch wurde diese Umsetzung hier ganz bewusst gewählt, wofür vor allem zwei Gründe maßgeblich sind:

1. Der Grundgedanke von CoMA ist die Simulationskontrolle allgemeiner gekoppelter Berechnungen. Die Umsetzung als eigenständiges Tool unterstützt diesen Gedanken, da sie eine Unabhängigkeit von Teilfeldlösern erfordert. Eine Ankopplung an einen der beiden Teilfeldlöser wertet dessen Position aber auf. Im weiteren Entwicklungsprozess besteht die Gefahr, Methoden des Teilfeldprozesses so zu benutzen, dass aus einer Ankopplung eine zwingende Bindung entsteht. Damit ist eine freie Austauschbarkeit der Teilfeldlöser nicht mehr möglich und der Anwendungsbereich von CoMA wird unnötig beschränkt.
2. Oberflächengekoppelte Probleme sind numerisch oft sehr aufwendig. Damit ist die massiv parallele Simulation der Teilfelder keine Kür, sondern eine Notwendigkeit. Die parallele Berechnung erfolgt auf Basis einer Gebietszerlegung. Die erzielbare Geschwindigkeitssteigerung in Folge der Parallelisierung hängt maßgeblich von der Art der Gebietszerlegung ab. Als relevante Faktoren ergeben sich die Anzahl an Freiheitsgraden pro Gebiet (daraus folgt die Größe der zu lösenden Gleichungssysteme) und die Anzahl an Freiheitsgraden am Gebietsrand, welche dem aktuellen und einem Nachbarschaftsgebiet zugeordnet sind (daraus folgt der notwendige Kommunikationsaufwand mit den Nachbargebieten). Hierbei ist das Ziel der Gebietszerlegung, ein Optimum unter Berücksichtigung dieser beiden Faktoren zu erreichen. Als Quasi-Standard hat sich hierbei die Methode *METIS* etabliert ([KK95, KK99, Gup97]). Bei Verwendung von *METIS* lässt sich aber die Zuordnung von Gebietsrändern, wie z.B. dem Kopplungsinterface, zu Teilgebieten nicht beeinflussen. Daraus folgt, dass das Kopplungsinterface nach der Gebietszerlegung auf mehrere Teilgebiete verteilt sein kann. In einer gekoppelten Berechnung müssen Operationen ausgeführt werden, für welche das gesamte Interface zur Verfügung steht. Sind auf beiden Feldern die gekoppelten Oberflächen unabhängig voneinander zerlegt, ergeben sich geometrisch komplexe und aufwendige Beziehungen, welche ein schwieriges Programmsetup nach sich ziehen. Für CoMA als zentralen Prozess allerdings folgt, dass es die beiden Oberflächennetze zentral vorhalten und darauf arbeiten kann, was den Entwicklungsprozess innerhalb CoMA stark vereinfacht.

Natürlich gibt es die Möglichkeit den Gebietszerlegungsprozess soweit zu beeinflussen, dass das Kopplungsinterface nur einem Teilgebiet zugeordnet ist. Beachtet man aber, dass z.B. im Falle turbulenter Strömungen gerade in Nähe des Kopplungsinterface ein besonders feines Netz vorliegt, folgt daraus eine hoch ineffiziente Gebietszerlegung, da damit das Teilgebiet mit dem Kopplungsinterface eine deut-

lich höhere Anzahl an Freiheitsgraden innehat als die anderen Teilgebiete.

5.3.1. Parallelisierungsstrategie

Die Parallelisierungsstrategie innerhalb der Feldlöser besteht aus den Konzepten zur Gebietszerlegung und zur Kommunikation zwischen den Gebieten. Um eine effiziente Kopplung zu gewährleisten, dürfen die bestehenden Feldkonzepte durch CoMA nicht eingeschränkt werden.

Aus der Gebietszerlegung der Feldlöser folgt, dass deren Berechnung durch mehrere Prozesse mit zugeordnetem Teilgebiet (Feldprozesse) erfolgt und dass das Oberflächennetz pro Feld (Kopplungsinterface) durch eine Untermenge der Feldprozesse (gekoppelte Feldprozesse) beschrieben wird. Um diese Aufsplittung innerhalb von CoMA abbilden zu können, wird ein Prozessgruppenkonzept verwendet. Nach Identifizierung der Anzahl der gekoppelten Feldprozesse wird für jeden Prozess ein Prozessobjekt erzeugt, welches die für die Kommunikation notwendigen Informationen enthält. Die einzelnen Prozesse werden zu einem Prozessgruppenobjekt zusammengefügt. Jeder Prozessgruppe ist ein eigenes Kommunikationsobjekt zugeordnet, auf welches die einzelnen Prozesse zugreifen, um mit ihrem Äquivalent innerhalb der Teilfelder zu kommunizieren. Innerhalb CoMA existieren immer zwei Prozessgruppen, welche das Äquivalent zu den Kopplungsinterfaces der zwei Teilfelder darstellen. Durch dieses Gruppierungskonzept lassen sich alle Gebietszerlegungsmethoden der Teilfelder innerhalb CoMA adäquat abbilden. Eine anschauliche Darstellung der entsprechenden Klassenstruktur ist in Abbildung 5.4 gegeben.

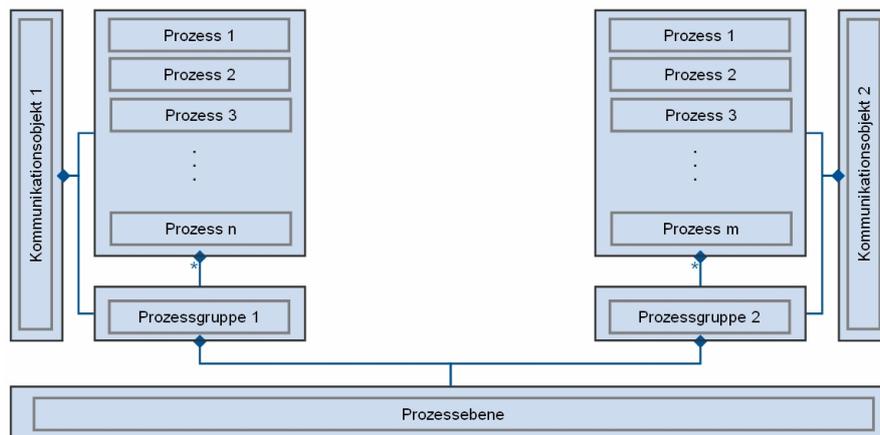


Abbildung 5.4.: Klassenstruktur des Prozessgruppenkonzepts in CoMA

Für die Kommunikation und damit den Datenaustausch zwischen CoMA und den Feldlösern kommen in Abhängigkeit der Möglichkeiten verschiedene Ansätze in Betracht, welche innerhalb des Feldlöser realisierbar sind. Als Standard zur Kommunikation innerhalb der Feldprozesse auf Hochleistungsrechnern hat sich das *Message Passing Interface* (MPI) etabliert ([GLDS96, GLS99, Sni98]), welches klar definierte Funktionen für den Datenaustausch zwischen Prozessen, die einem sogenannten Kommunikator zugeordnet sind,

ermöglicht. Des Weiteren ist die Kommunikation mittels MPI sehr effizient, da entsprechende Hochgeschwindigkeitsnetzwerkverbindungen unterstützt werden. Dies ermöglicht den schnellen Austausch auch großer Datenmengen.

Für das in CoMA verwendete Kommunikationskonzept bietet es sich an, diese in den meisten Feldlösern bereits vorhandene Methode ebenfalls zu nutzen. Um den notwendigen Eingriff in das bestehende Kommunikationskonzept der Feldlöser möglichst gering zu halten, wird ein Subkommunikatorkonzept verwendet. Dadurch sind nur minimale Änderungen notwendig und das vorhandene Konzept kann ohne Einschränkung weiter verwendet werden. Grundlegendes Element für eine Kommunikation mittels MPI sind die sogenannten Kommunikatoren. Ein Kommunikator besteht aus einer Gruppierung von Prozessen, zwischen denen Daten transferiert werden können. Teilfeldlöser verwenden i.d.R. einen Kommunikator, in dem alle beim Programmstart vorhandenen Prozesse eingebunden sind, da diese ja bei standardgemäßer Verwendung dem Feldlöser zuzuordnen sind. Wird für die Kommunikation zwischen CoMA und den Feldlösern MPI verwendet, werden CoMA und die zwei parallelen Teilfeldlöser gleichzeitig gestartet und damit standardmäßig in einen gemeinsamen globalen Kommunikator eingeordnet. Danach werden die Teilfeldprozesse innerhalb der globalen Prozessliste identifiziert und in eigenständige Kommunikatoren eingruppiert (Subkommunikatoren). Wird nun in den Funktionen der Feldlöser zur Kommunikation innerhalb des Feldes der bisherige Kommunikator durch den neu gebildeten Subkommunikator ersetzt, können diese ohne weitere Änderung weiter verwendet werden. Für die Kommunikation mit CoMA wird der globale Kommunikator verwendet und es sind einige zusätzliche Funktionen notwendig. Dieses Subkommunikatorkonzept wird in Abb. 5.5 anschaulich dargestellt. Die Parallelisierungsstrategie wird als *MPMD* (Multiple Program Multiple Data) bezeichnet, da verschiedene Prozesse verschiedene Programme ausführen.

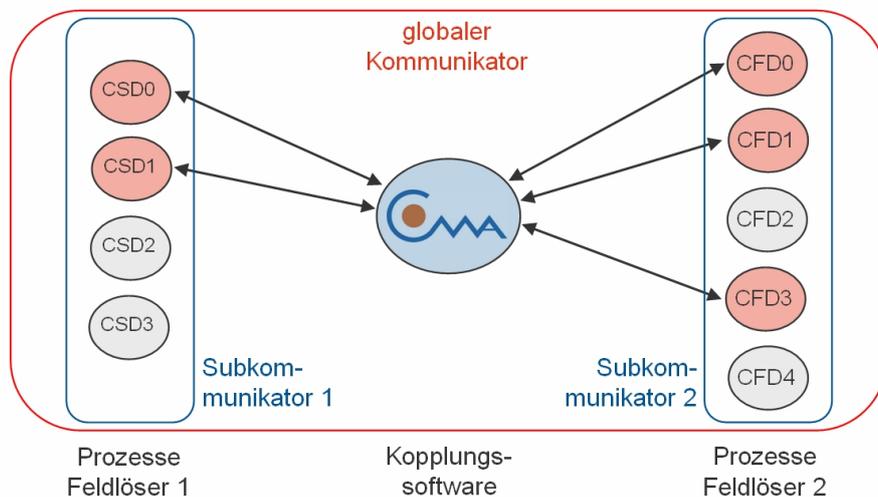


Abbildung 5.5.: Kommunikatorkonzept in CoMA

Anmerkung 1: Die durch den MPI-Standard definierten Funktionen sind zwei Versionen zugeteilt. Die grundlegenden Funktionen aus Version 1 sind in allen MPI-Bibliotheken implementiert, die Funktionen aus Version 2 sind (noch) nicht in allen Bibliotheken verfügbar. Um eine Kompatibilität von CoMA mit möglichst vielen MPI-Bibliotheken zu gewährleisten, werden nur Funktionen aus Version 1 verwendet.

Anmerkung 2: Da für eine auf MPI basierende Kommunikation Änderungen am Quellcode und am Kommunikationskonzept der Feldlöser notwendig sind, kann es natürlich nur eingesetzt werden, wenn diese Möglichkeiten gegeben sind. Als Alternative ist es möglich, den Datenaustausch über das Schreiben und Lesen von Dateien zu realisieren. Die notwendigen Eingriffe in die Teilfeldlöser sind damit deutlich geringer, allerdings ist diese Methode auch deutlich langsamer. Diese Methode ist nur in der anfangs erwähnten zweiten Variante von CoMA vorhanden.

Anmerkung 3: Jede Prozessgruppe in CoMA verfügt über ein eigenes Kommunikationsmodell. Damit ist es CoMA möglich, mit einem Feldlöser mittels MPI, mit dem anderen mittels Dateien zu kommunizieren. Dies unterstützt das Ziel der allgemeinen Verwendbarkeit von CoMA.

Anmerkung 4: Da das Kommunikationsmodell innerhalb CoMA gekapselt ist und über eine standardisierte Schnittstelle angesprochen wird, ist die Implementierung von weiteren Modellen mit geringem Aufwand möglich. Zum Beispiel könnte auch eine Kommunikation, die auf dem TCP/IP-Level basiert, realisiert werden. Damit ist eine einfache Anpassung an weitere Feldlöser gegeben.

5.3.2. Datentransfer zwischen Oberflächennetzen

In einer oberflächengekoppelten Simulation wird das Kopplungsinterface durch zwei Oberflächennetze beschrieben, welche den beiden Teilfeldnetzen zugeordnet sind. Diese sind das Ergebnis der in den Feldlösern angewandten Vernetzungsstrategie des Simulationsgebietes. Die für eine Kopplung relevanten Daten werden auf dem Kopplungsinterface beschrieben. Wesentliche Bestandteile der Kopplungssoftware CoMA sind daher:

- Beschreibung der zwei Oberflächennetze unter Berücksichtigung der Gebietszerlegung der Feldlöser.
- Beschreibung unterschiedlicher Datenverläufe auf den Oberflächennetzen.
- Möglichkeit zum Datentransfer zwischen den Oberflächennetzen unter Berücksichtigung nicht-koinzidenter Netze.

Für den Fall koinzidenter Oberflächennetze, in denen die Netze identisch sind, ist der Datentransfer einfach zu realisieren. Für den Fall nicht-koinzidenter Oberflächennetze ergeben sich aber komplexere Probleme, deren Lösungsansätze in CoMA im Folgenden beschrieben werden. Für die Behandlung nicht-koinzidenter Oberflächen sprechen mehrere Gründe:

- Bereits existierende Feldnetze aus Einzelfeldsimulationen können innerhalb einer gekoppelten Simulation weiter verwendet werden, ohne das Interfacenetz und damit das gesamte Feldmodell anpassen zu müssen.
- Durch die im Regelfall unterschiedliche Physik der Felder werden verschiedene Vernetzungsrichtlinien angewandt, z.B. sehr feine und strukturierte Fluidnetze in Wandnähe zur Auflösung der Grenzschicht oder unstrukturierte Netze bei Vernetzung komplexer Geometrien.
- Durch unterschiedliche Diskretisierungsansätze der Felder, wie z.B. mittels der Finite-Elemente-Methode oder der Finite-Volumen-Methode, ergeben sich unterschiedliche Ansätze zur Beschreibung der Feldverläufe.
- Bei Forderung koinzidenter Oberflächenetze müssen sich die Feldnetze an der höheren Einzelvorgabe orientieren. Damit muss z.B. das Strukturnetz genauso fein wie das Fluidnetz aufgelöst sein, obwohl auch ein gröberes Strukturnetz ausreichende Genauigkeit liefert. Damit werden mehr Freiheitsgrade als nötig eingeführt, was die Effizienz des Ansatzes deutlich verringert.

Um die allgemeine Verwendbarkeit und Modularität von CoMA zu gewährleisten, ist es also nötig, nicht-koinzidente Oberflächenetze abbilden und behandeln zu können.

5.3.2.1. Netz- und Datenbeschreibung

Grundlegend für die Aufgabe des Datentransfers ist eine effiziente Beschreibung der Oberflächenetze und der darauf zu speichernden Daten. Diese muss vornehmlich zwei Vorgaben erfüllen:

1. Abbildung der Aufspaltung des Oberflächenetzes in Prozessen zugeordneten Partitionen durch die Gebietszerlegung der Feldlöser.
2. Eine programmiertechnisch effiziente Datenhaltung und -speicherung, da die meisten Operationen in CoMA auf die Netzdaten zugreifen und damit die Gesamtperformance stark beeinflussen.

Basierend auf dem Prozessgruppenkonzept (Abbildung 5.4) wird jeder Prozessgruppe eine Netzgruppe mit Oberflächenetzen zugeordnet. Für die Beschreibung eines Oberflächenetzes wird ein partitionsweises Konzept benutzt. Damit ist das gesamte Oberflächenetz in einzelne Partitionen unterteilt. Jede Partition ist einem Prozess der Prozessgruppe (des zugehörigen Feldlösers) zugeordnet; jeder Prozess kann aber auch mehrere Partitionen besitzen. Pro Partition werden die zugehörigen Knoten und Elemente gespeichert, welche damit die tatsächliche Geometriedefinition darstellen. Damit sind alle Operationen, die auf Netzdaten zugreifen, über einen einfachen Algorithmus abbildbar (siehe Algorithmus 12). Eine anschauliche Darstellung der entsprechenden Klassenstruktur ist in Abbildung 5.6 gegeben.

Da die Anzahl an Prozessen und Partitionen pro Prozess in der Regel gering ist, ist für einen effizienten Datenzugriff vor allem die Speichermethode pro Partition maßgebend.

Algorithmus 12 Datenzugriff bei partitionsbasierter Netzbeschreibung

```

1: for Prozesse pro Prozessgruppe do
2:   for Partitionen pro Prozess do
3:     for Elemente/Knoten pro Partitionen do
4:       Führe Operation aus
5:     end for
6:   end for
7: end for

```

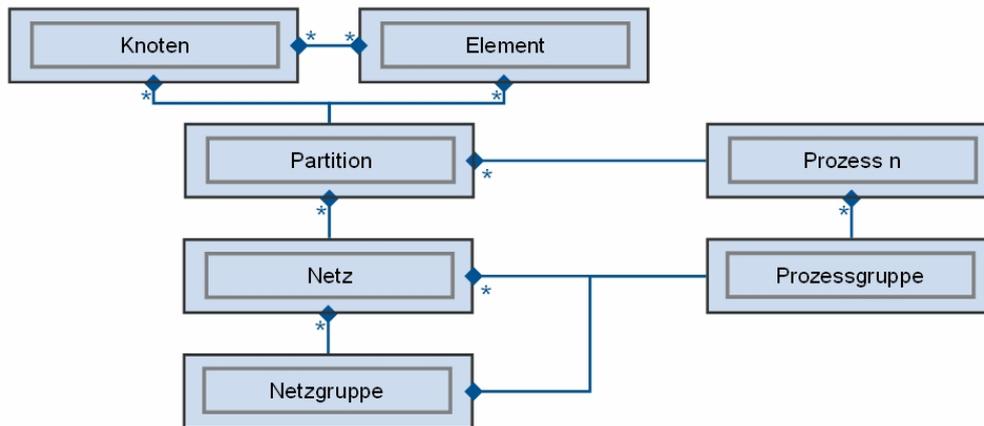


Abbildung 5.6.: Netzkonzept in CoMA

Deshalb wird in CoMA der Speicher für Netz und Daten immer blockweise pro Partition allokiert und damit sichergestellt, dass die Daten kontinuierlich im Speicher liegen.

Für die Beschreibung von Datenverläufen auf dem Oberflächennetz wird ein FE-basierter Ansatz gewählt. Damit werden die Daten an diskreten Punkten, den Knoten, vorgegeben und die Verläufe zwischen den Knoten elementweise über bilineare Formfunktionen angenähert. Damit ist auch der Datenaustausch zwischen CoMA und den Feldlösern charakterisiert durch den Austausch knotenbasierter Werte.

Anmerkung: Es ist in CoMA möglich, innerhalb einer Netzgruppe mehrere Oberflächennetze gleichzeitig zu behandeln. Dazu ist jede Partition mit einem Kenner versehen. Nur Partitionen mit identischem Kenner werden zu einem Oberflächennetz zusammengefasst. Dadurch kann eine gekoppelte Simulation mit mehreren Kopplungsinterfaces durchgeführt werden, die unterschiedlich behandelt werden können. Daraus ergeben sich folgende Vorteile:

1. Es können unterschiedliche Kopplungsmethoden gleichzeitig abgebildet werden. So kann z.B. an einem Interface innerhalb des Gesamtgebietes die Simulation einer Fluid-Struktur-Wechselwirkung mit Austausch von Verschiebungen und Kräften stattfinden und an einem weiteren Interface eine einseitige Kopplung mit Tempe-

raturaustausch.

2. Für den Fall der Umströmung extrem dünner Flächentragwerke, z.B. Membranstrukturen, liegen die Oberflächennetze an Ober- und Unterseite des Tragwerks nahezu aufeinander. Dadurch ist es möglich, dass in der für den Datentransfer notwendigen Nachbarschaftssuche fehlerhafte Beziehungen zwischen Ober- und Unterseite hergestellt werden. Dieses Problem wird durch eine getrennte Behandlung von Ober- und Unterseite umgangen.

5.3.2.2. Datentransfermethoden

Mittels Datentransfermethoden werden die auf einem Oberflächennetz am Interface vorgegebenen Daten auf das andere Oberflächennetz transferiert. Die verwendete Methode kann dabei nach folgenden Kriterien beurteilt werden [BZB07, BZB08]:

- Globale Energiekonservativität am Interface,
- globale Lastkonservativität am Interface,
- Genauigkeitsordnung und Erhalt der Genauigkeitsordnung der Teilfeldlöser und
- Effizienz, definiert als Verhältnis zwischen Genauigkeit und numerischem Aufwand.

Für den Fall der Fluid-Struktur-Wechselwirkung sind am Interface Γ die Bedingungen der Kontinuität der Verschiebungen (kinematische Randbedingung) und der Oberflächenkräfte (dynamische Randbedingung) zu erfüllen (siehe Abschnitt 4.1.3). In der kontinuierlichen Form sind diese gegeben durch:

$$\begin{aligned} d^{\Gamma^S} &= d^{\Gamma^F}, \\ f^{\Gamma^S} &= f^{\Gamma^F}. \end{aligned} \quad (5.1)$$

Γ^S bezeichnet hierbei die Strukturseite und Γ^F die Fluidseite des Interfaces. Unabhängig von der Datentransfermethode lässt sich die diskrete Form dieser Bedingungen allgemein angeben als:

$$\begin{aligned} \mathbf{d}^{\Gamma^F} &= \mathbf{T}_{SF} \mathbf{d}^{\Gamma^S}, \\ \mathbf{f}^{\Gamma^S} &= \mathbf{T}_{FS} \mathbf{f}^{\Gamma^F} \end{aligned} \quad (5.2)$$

unter Verwendung der Transfermatrizen \mathbf{T}_{SF} und \mathbf{T}_{FS} . Mit $n_{dof}^{\Gamma^F}$ als Anzahl an Freiheitsgraden am Fluidinterface und $n_{dof}^{\Gamma^S}$ als Anzahl an Freiheitsgraden am Strukturinterface ergibt sich für \mathbf{T}_{SF} die Dimension $n_{dof}^{\Gamma^F} \times n_{dof}^{\Gamma^S}$ und für \mathbf{T}_{FS} die Dimension $n_{dof}^{\Gamma^S} \times n_{dof}^{\Gamma^F}$.

Die auf dem Fluidinterface verrichtete virtuelle Arbeit der Fluids δW_F ergibt sich zu:

$$\delta W_F = \left(\mathbf{f}^{\Gamma^F} \right)^T \cdot \delta \mathbf{d}^{\Gamma^F} \quad (5.3)$$

und die von der Struktur verrichtete virtuelle Arbeit δW_S auf das Strukturinterface zu:

$$\delta W_S = \left(\mathbf{f}^{\Gamma^S} \right)^T \cdot \delta \mathbf{d}^{\Gamma^S}. \quad (5.4)$$

Für eine Energiekonservativität gilt: $\delta W_F = \delta W_S$, und damit folgt:

$$\mathbf{f}^{\Gamma^S} = (\mathbf{T}_{SF})^T \mathbf{f}^{\Gamma^F}, \quad (5.5)$$

als Bedingung für die globale Energiekonservativität am Interface. Damit kann also die Transponierte der für den Verschiebungstransfer von Fluid nach Struktur benutzten Transfermatrix für den Lasttransfer in die Gegenrichtung verwendet werden:

$$\mathbf{T}_{FS} = (\mathbf{T}_{SF})^T. \quad (5.6)$$

Damit lässt sich eine allgemeine Bedingung für die diskrete Erfüllung der globalen Lastkonservativität am Interface aufstellen. Konservativität ist gegeben, wenn sich die Spaltensummen von $(\mathbf{T}_{SF})^T$ zu 1 ergeben:

$$\sum_i (\mathbf{T}_{SF})_{ij}^T = 1, \quad (5.7)$$

denn damit gilt:

$$\sum_i \mathbf{f}_i^{\Gamma^S} = \sum_i \sum_j (\mathbf{T}_{SF})_{ij}^T \mathbf{f}_j^{\Gamma^F} = \sum_j \mathbf{f}_j^{\Gamma^F} \sum_i (\mathbf{T}_{SF})_{ij}^T = \sum_j \mathbf{f}_j^{\Gamma^F}. \quad (5.8)$$

Im Folgenden werden die in CoMA vorhandenen Datentransfermethoden vorgestellt und gezeigt, wie die zugehörigen Transfermatrizen erstellt werden:

- **„Nearest Neighbour“-Methode**

Bei dieser Methode ([TBU00]) wird zu jedem Knoten von Netz A der nächstgelegene Nachbarknoten von Netz B gesucht und als Projektionspunkt verwendet. Für den Datentransfer werden die entsprechenden Knotendaten einfach auf den zugehörigen Projektionspunkt übertragen. Die Transfermatrix T_{SF} ergibt sich damit als Boolesche Matrix mit je einem Eintrag ungleich 0 pro Spalte. Existiert für jeden Punkt auch ein zugehöriger Projektionspunkt, ist sie auch global last- und energiekonservativ. Die Genauigkeitsordnung bleibt aber nur für den Fall koinzidenter Netze erhalten. Dies stellt auch den großen Nachteil dieser zwar sehr einfachen, aber insgesamt wenig effizienten Methode dar. Für den Fall nicht-koinzidenter Netze bleibt die Genauigkeitsordnung nicht erhalten.

- **Projektionsbasierte Methode**

Projektionsbasierte Methoden beruhen auf einer orthogonalen Projektion der Knoten von Netz A auf die Elemente von Netz B mittels einer Nachbarschaftssuche in Kombination mit einer Interpolation über Formfunktionen ([FLL98, LYC⁺95, CL97]). Unter Verwendung entsprechender Formfunktionen ist die Position der Projektionspunkte mittels lokaler Koordinaten ξ und η innerhalb der Elemente

von Netz B bekannt. Damit ergibt sich die Position des Projektionspunktes x_B^i in Element j von Netz A zu:

$$\mathbf{x}_B^i \approx \sum_{k=1}^{n\text{nod}} N_k(\xi_i, \eta_i) \mathbf{x}_A^{j,k}. \quad (5.9)$$

Für den Transfer von Verschiebungen vom Struktur- auf das Fluidnetz werden diese auf die gleiche Art wie die Geometrie approximiert, indem die Formfunktionen an der Stelle des Projektionspunktes ausgewertet werden:

$$\mathbf{d}_F^i \approx \sum_{k=1}^{n\text{nod}^S} N_k(\xi_i, \eta_i) \mathbf{d}_S^{j,k}, \quad (5.10)$$

mit $\mathbf{d}_S^{j,k}$ als vorgegebene Knotenverschiebung von Knoten k in Strukturelement j und $n\text{nod}^S$ als Anzahl an Knoten des Strukturelementes, in dem der Projektionspunkt liegt. In Abbildung 5.7 ist dies vereinfacht für den eindimensionalen Fall dargestellt.

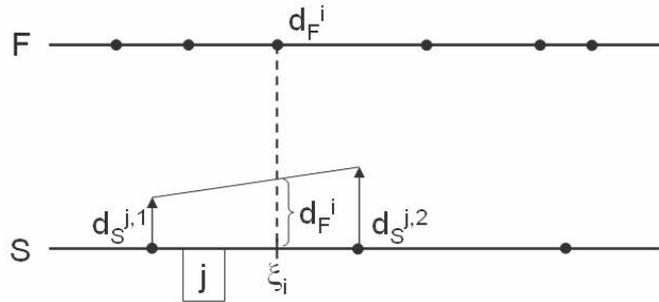


Abbildung 5.7.: Projektionsbasierter Transfer von Verschiebungen

Der Transfer von Lasten vom Fluid- zum Strukturnetz geschieht über eine gewichtete Summation. Für die Kraft \mathbf{f}_S^j im Strukturknoten j werden aus allen an den Knoten angrenzenden Strukturelementen j von den darin befindlichen Projektionspunkten der Fluidknoten k die Fluidkräfte \mathbf{f}_F^k über Formfunktionen skaliert und aufsummiert:

$$\mathbf{f}_S^j \approx \sum_{j=1}^{nele^S} \sum_{k=1}^{n\text{nod}^F} N_j(\xi_k, \eta_k) \mathbf{f}_F^k, \quad (5.11)$$

mit $nele^S$ als Anzahl der angrenzenden Strukturelemente und $n\text{nod}^F$ als Anzahl der in Strukturelement j vorhandenen Fluidprojektionspunkte. In Abbildung 5.8 ist dies vereinfacht für den eindimensionalen Fall dargestellt. Da die Summe über die Formfunktionen sich zu 1 ergibt, erfüllt dieses Verfahren die globale Lastkonservativität.

Die Einträge der Transformationsmatrizen T_{SF} und T_{FS} ergeben sich über die Auswertung der Formfunktionen $N_k(\xi_i, \eta_i)$ bzw. $N_j(\xi_k, \eta_k)$ an den Projektionspunkten der Fluidknoten in den Strukturelementen.

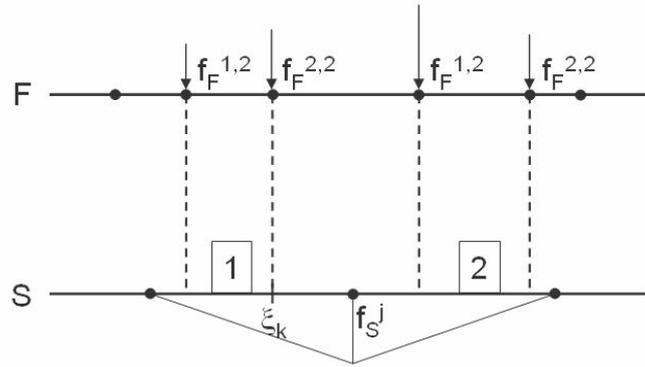


Abbildung 5.8.: Projektionsbasierter Transfer von Lasten

Diese Methode ist global last- und energiekonservativ und erhält die Genauigkeitsordnung der Feldlöser, wenn auf beiden Seiten die gleichen Formfunktionen verwendet werden. Dies ist aber nur selten der Fall. Werden z.B. für die Struktur bilineare Ansätze in Folge einer FE-Diskretisierung und für das Fluid FV-basierte elementweise konstante Ansätze verwendet, ist ein Erhalt der Genauigkeitsordnung nicht möglich. Die erzielte Genauigkeitsordnung ist also das Minimum aus den Ansätzen der Feldlöser und den in CoMA verwendeten Ansätzen. Damit ist diese Methode vergleichsweise effizient, da sie numerisch wenig aufwendig ist (die Transformationsmatrizen sind konstant und müssen deshalb nur einmal erstellt werden) und eine akzeptable Genauigkeit liefert. Ein Vergleich dieser mit anderen Derivaten projektionsbasierter Methoden wird in [JJGL05] gegeben.

Ein Vergleich verschiedener Methoden ist z.B. in [BZB07] aufgeführt, wo die Erfüllung der Lastkonservativität über die Abbildung von Starrkörperbewegungen überprüft wird.

5.3.3. Simulationssteuerung und Kopplungsalgorithmen

Eine der wichtigsten Aufgaben von CoMA als allgemeinem Kopplungstool ist die Steuerung der gekoppelten Berechnung. Hierbei ist das Ziel, dass die angeschlossenen Feldlöser primär nur die Methoden zur Simulation des Einzelfeldes aufweisen müssen und alle zusätzlich für eine gekoppelte Simulation benötigten Methoden durch CoMA realisiert werden.

Dabei ist zwischen den folgenden Kopplungsarten zu unterscheiden, da sich aus diesen das zugehörige algorithmische Konzept ergibt:

- Einseitige Kopplung, wobei nur ein Informationsaustausch von Feld 1 zu Feld 2 stattfindet und keine wirkliche Wechselwirkung.
- Zweiseitige Kopplung, wobei ein Informationsaustausch von Feld 1 zu Feld 2 und zurück stattfindet, und damit eine Wechselwirkung mit gegenseitiger Beeinflussung der beiden Felder.

Für die zweiseitige Kopplung ist des Weiteren noch zu unterscheiden in (siehe auch Abschnitt 4.2):

- Explizite zweiseitige Kopplung, wobei pro Zeitschritt nur ein Austauschzyklus stattfindet.
- Implizite zweiseitige Kopplung, wobei pro Zeitschritt mehrere Austauschzyklen stattfinden, i.d.R. bis der Gleichgewichtszustand des gekoppelten Systems erreicht ist.

Eine graphische Darstellung dieser Kopplungsarten ist in Abb. 5.9 gegeben.

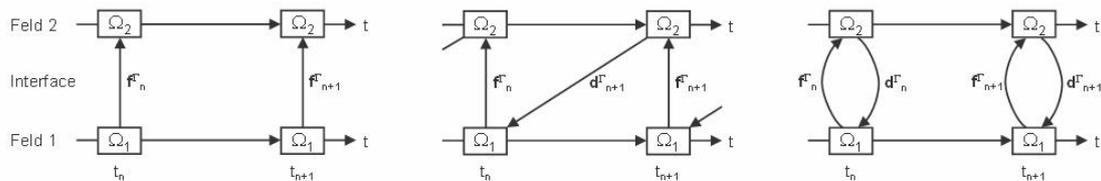


Abbildung 5.9.: Kopplungsarten in CoMA: Einseitige, explizit zweiseitige und implizit zweiseitige Kopplung

Um diese Methoden in CoMA abbilden zu können, wird ein sogenanntes Austauschpunkt-konzept verwendet. Ein Austauschpunkt („exchange point“) ist dabei eine Funktionseinheit, welche innerhalb des zeitlichen Ablaufs eines Programms Daten von Prozessgruppe 1 nach Prozessgruppe 2 transferiert, also austauscht. Dabei findet der Datentransfer zwischen den nicht-koinzidenten Oberflächennetzen statt und optional werden die Daten dabei auch verändert. Einzelne Austauschpunkte werden zu einer Gruppe zusammengefasst („exchange point group“). Bei einer einseitigen Kopplung beinhaltet die Gruppe nur einen Austauschpunkt, bei einer zweiseitigen Kopplung zwei Austauschpunkte. Es können auch mehr als zwei Austauschpunkte eingeführt werden, um z.B. in einer Fluid-Struktur-Wechselwirkung zusätzlich zu Kräften und Verschiebungen auch Geschwindigkeiten oder Temperaturen am gleichen Interface zu transferieren. Die Behandlung von Zyklen pro Zeitschritt wird durch die übergeordnete Wahl einer Kopplungsart vorgegeben.

Um für den Fall einer zweiseitigen impliziten Kopplung eine entsprechende Simulation durchzuführen, werden einem Austauschpunkt zusätzlich zum Datenaustausch zwischen den Oberflächennetzen weitere Eigenschaften zugeordnet. Dazu zählen:

- **Prädiktor**

Ein Prädiktor wird benutzt, um in der ersten Subiteration einen Startwert für die zu transferierende Variable zu erhalten. Je näher der Startwert an der endgültigen Lösung liegt, desto effizienter ist der Prädiktor und desto weniger Subiterationen werden benötigt. Alle der in Abschnitt 4.4 behandelten Prädiktoren sind in CoMA vorhanden. Dabei sind die polynombasierten Methoden allgemein verwendbar, die Newmarkbasierten und der adaptive Prädiktor nur für Probleme mit strukturdynamischer Komponente.

- **Kopplungsalgorithmus**

Der Kopplungsalgorithmus ist die Methode, mit der das nichtlineare gekoppelte Problem gelöst wird. Dieser wird hinsichtlich Effizienz und Stabilität beurteilt. Alle in den Abschnitten 4.6, 4.7, 4.8.2 und 4.8.1 behandelten Kopplungsalgorithmen sind in CoMA vorhanden. Methoden, die in Kombination mit Black-Box-Feldlösern verwendet werden, können für alle gekoppelten Probleme eingesetzt werden, da sie unabhängig von der verwendeten Physik sind. Methoden, welche eine algorithmisch freie Behandlung des Problems verwenden, sind auf Probleme der Fluid-Struktur Wechselwirkung beschränkt.

- **Konvergenzüberprüfung**

Die Lösung des nichtlinearen gekoppelten Problems ist erreicht, sobald eine gewisse Konvergenz erfüllt ist. In CoMA sind die in Abschnitt 4.3 vorgestellten relativen und absoluten Konvergenzkriterien vorhanden. Alle diese Kriterien sind unabhängig von der behandelten Physik des gekoppelten Problems.

Ein exemplarisches Ablaufdiagramm für eine zweiseitige implizite Kopplung einer Fluid-Struktur-Wechselwirkung in CoMA ist in Algorithmus 13 gegeben. Hierbei ist dem Strukturfeld die Prozessgruppe 1 und dem Fluidfeld die Prozessgruppe 2 zugeordnet. Die Bestimmung der neuen Interfaceverschiebungen basiert auf der Diskretisierung des Netzes der Prozessgruppe 1.

Dieses Grundkonzept ist sehr flexibel und leicht erweiterbar, womit auch weitere Kopplungsmethoden mit wenig Aufwand implementiert werden können. Durch die Optionalität aller Eigenschaften ist es auch möglich, dass die Simulationsteuerung bzw. Teile davon durch einen der Feldlöser übernommen wird und CoMA ausschließlich zum Transfer der Daten zwischen den Lösern benutzt wird.

Algorithmus 13 Ablaufalgorithmus in CoMA bei zweiseitiger impliziter Kopplung einer Fluid-Struktur-Wechselwirkung

```

1: for Schleife über Zeitschritte  $n$  do
2:   for Schleife über Subiterationen  $k$  do
3:     Austauschpunkt 1:
4:     if ( $k = 1$ ) then
5:       Prädiktorschritt:  $\mathbf{d}_{n+1}^{\Gamma,k} = f(\mathbf{d}_n^{\Gamma,k}, \mathbf{d}_{n-1}^{\Gamma,k}, \dots)$ 
6:       Transferiere  $\mathbf{d}_{n+1}^{\Gamma,k}$  von Netz 1 auf Netz 2
7:       Sende  $\mathbf{d}_{n+1}^{\Gamma,k}$  an Prozessgruppe 2
8:     else
9:       Empfange  $\tilde{\mathbf{d}}_{n+1}^{\Gamma,k}$  von Prozessgruppe 1
10:      Konvergenz:  $f(\mathbf{d}_{n+1}^{\Gamma,k-1}, \tilde{\mathbf{d}}_{n+1}^{\Gamma,k}, \dots)$ 
11:      if (keine Konvergenz) then
12:        Kopplungsalgorithmus:  $\mathbf{d}_{n+1}^{\Gamma,k+1} = f(\dots)$ 
13:        Transferiere  $\mathbf{d}_{n+1}^{\Gamma,k+1}$  von Netz 1 auf Netz 2
14:        Sende  $\mathbf{d}_{n+1}^{\Gamma,k+1}$  an Prozessgruppe 2
15:      else
16:        Gehe zu nächstem Zeitschritt
17:      end if
18:    end if
19:    Austauschpunkt 2:
20:    Empfange  $\mathbf{f}_{n+1}^{\Gamma,k}$  von Prozessgruppe 2
21:    Transferiere  $\mathbf{f}_{n+1}^{\Gamma,k}$  von Netz 2 auf Netz 1
22:    Sende  $\mathbf{f}_{n+1}^{\Gamma,k}$  an Prozessgruppe 1
23:  end for
24: end for

```

6. Numerische Beispiele

In diesem Kapitel werden mehrere numerische Beispiele behandelt. Sie dienen dazu, die in den vorherigen Kapiteln theoretisch hergeleiteten Lösungsmethoden und -ansätze hinsichtlich Anwendbarkeit, Effizienz und Stabilität miteinander zu vergleichen. Hierzu werden zwei anspruchsvolle Beispiele verwendet.

Das erste Beispiel (Abschnitt 6.1), der sogenannte „DFG-Benchmark“ ([BS10]), beinhaltet ein rein numerisches Modellproblem. Es weist eine einfache zweidimensionale Geometrie auf. Die Schwierigkeit liegt in der starken Kopplung zwischen den Teilfeldern und den auftretenden großen Verformungen. Dieses Beispiel dient dazu, die in den Kapiteln 2 bis 4 vorgestellten algorithmischen Ansätze miteinander zu vergleichen. Dazu zählen der Feldlöseransatz, der Interfaceprädiktor und der Kopplungsalgorithmus. In einem ersten Schritt werden die Ansätze getrennt voneinander untersucht, in einem zweiten Schritt werden diese miteinander kombiniert und die Kombinationen verglichen. Das Beispiel endet mit der Empfehlung für eine besonders effiziente und stabile Methodenkombination.

Im zweiten Beispiel (Abschnitt 6.2) wird die realistische und dreidimensionale Problemstellung der Wechselwirkung zwischen einer atmosphärischen Grenzschichtströmung und einem leichten Flächentragwerk abgebildet. Dieses Beispiel dient dazu, die empfohlene Methodenkombination als Ergebnis des ersten Beispiels auf ein komplexes Problem anzuwenden, und die Effizienz der entwickelten Parallelisierungsstrategie für den partitionierten Ansatz (Kapitel 5) zu untersuchen. Auch werden die physikalisch interessanten Simulationsergebnisse dargestellt.

Das Kapitel endet mit einer kurzen Zusammenfassung.

6.1. Numerischer DFG-Benchmark

In diesem Beispiel werden die in den Kapiteln 2, 3 und 4 vorgestellten Methoden angewendet und hinsichtlich Effizienz und Stabilität miteinander verglichen. Dabei ist es das Ziel, allgemeine Empfehlungen für zukünftige Beispiele geben zu können. Als Beispiel werden dabei die Fälle FSI2 und FSI3 des von Turek und Hron ([STD⁺96, TH06]) vorgestellten numerischen Modellproblems genutzt. Dieses Beispiel wurde innerhalb der von der DFG geförderten Forschergruppe FOR 493 entwickelt und von allen beteiligten Gruppen mittels unterschiedlichster numerischer Ansätze berechnet ([BS10]).

Dieses Modellproblem wurde gewählt, da es die folgenden Eigenschaften besitzt:

- Die Fälle FSI2 und FSI3 sind transiente Probleme mit großen Verformungen und starker Kopplung. Damit sind die Voraussetzungen für die in dieser Arbeit untersuchten Problemstellungen erfüllt.
- Das Beispiel ist zweidimensional, Geometrie und physikalische Eigenschaften sind klar definiert und die Einzelfelder sind einfach zu modellieren. Damit eignet es sich besonders, um methodische und algorithmische Untersuchungen durchzuführen.
- Es existieren Referenzergebnisse für alle in der Kopplung relevanten Ergebnisse. Deren Richtigkeit kann angenommen werden, da sie von sieben verschiedenen Gruppen bestätigt wurden ([THR⁺10]). Damit dient dieses Beispiel auch dazu, die eigene Softwareumgebung zu validieren und verifizieren.

Die folgenden Methoden werden in diesem Beispiel untersucht und miteinander verglichen:

- Konstante versus angepasste Feldlöser (siehe Abschnitt 2.5 und Abschnitt 3.6),
- Interfaceprädiktoren (siehe Abschnitt 4.4) und
- Kopplungsalgorithmen (siehe Abschnitte 4.5 bis 4.8).

Dieses Kapitel ist wie folgt aufgebaut: In Abschnitt 6.1.1.1 wird die Geometrie und Physik des Problems und in Abschnitt 6.1.1.2 die verwendeten numerischen Modelle vorgestellt. In Abschnitt 6.1.2 werden die Simulationsergebnisse der zwei Fälle dargestellt und mit den Referenzwerten verglichen. In Abschnitt 6.1.3 wird die durch Verwendung inkrementeller Feldlöser erzielbare Effizienzsteigerung untersucht. In Abschnitt 6.1.4 werden unterschiedliche Prädiktoren für die Interfacepositionsbestimmung in der ersten Kopplungsiteration miteinander verglichen. In Abschnitt 6.1.5 werden die vorgestellten Kopplungsalgorithmen verwendet und verglichen. In Abschnitt 6.1.6 werden die einzelnen Methoden kombiniert, um Aussagen über eine insgesamt mögliche Effizienzsteigerung treffen zu können. Das Kapitel endet mit einer Zusammenfassung und einer Empfehlung für die Wahl einer besonders effizienten Methodenkombination.

6.1.1. Problemdefinition

6.1.1.1. Geometrie und Physik

Fluidfeld: Das Fluidfeld wird als newtonsches und inkompressibles Fluid angenommen. Die Geometrie und Randbedingungen des Fluidfeldes sind in Abbildung 6.1 angegeben. Das rechteckige Fluidgebiet hat die Länge $L = 2,5 \text{ m}$ und die Höhe $H = 0,41 \text{ m}$. Innerhalb des Gebietes befindet sich ein Zylinder mit Radius $r = 0,05 \text{ m}$ und Mittelpunkt $(0,2; 0,2 \text{ m})$, an den ein Balken der Länge $l = 0,35 \text{ m}$ und der Höhe $h = 0,02 \text{ m}$ anschließt. Zylinder und Balken sind also leicht asymmetrisch platziert. Das Fluidfeld besitzt ein parabolisches Einströmprofil an der linken Begrenzung, gegeben zu:

$$u_x = 1,5 U^* \frac{4,0}{0,1681} y (0,41 - y). \quad (6.1)$$

Die rechte Begrenzung ist der Ausströmrand und alle übrigen Ränder sind Wände, an denen die Geschwindigkeit Null beträgt.

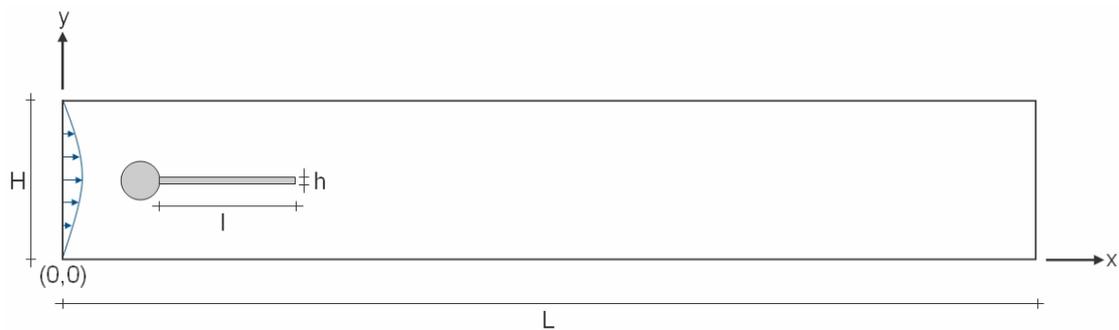


Abbildung 6.1.: Geometrie und Randbedingungen des Fluidfeldes

Strukturfeld: Das Strukturfeld besteht aus dem Zylinder und dem anschließenden Balken, siehe Abbildung 6.2. Der Zylinder wird als starr angenommen. Der Balken ist elastisch und erfüllt das St. Venant-Kirchhoff Materialgesetz.

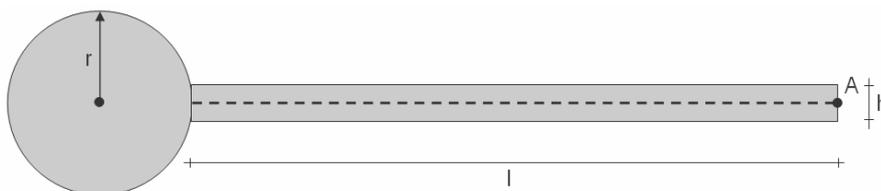


Abbildung 6.2.: Geometrie und Randbedingungen des Strukturfeldes

Die physikalischen Parameter der Fälle FSI2 und FSI3 sind in Tabelle 6.1 zusammengefasst. Eine detailliertere Beschreibung findet sich in [TH06].

	$\frac{\rho^S}{\rho^F}$	$\rho^S \frac{kg}{m^3}$	$\nu^S [-]$	$E \frac{kg}{m^3}$	$\rho^F \frac{kg}{m^3}$	$\nu^F \frac{m^2}{s}$	$U^* \left[\frac{m}{s}\right]$	$Re [-]$
FSI2	10, 0	10.000	0, 4	$1, 4e^{06}$	1.000	$1e^{-03}$	1, 0	100
FSI3	1, 0	1.000	0, 4	$5, 6e^{06}$	1.000	$1e^{-03}$	2, 0	200

Tabelle 6.1.: Physikalische Parameter der Fälle FSI2 and FSI3

6.1.1.2. Numerisches Modell

Fluidfeld: Das laminare Fluidfeld wird mittels der Finiten-Volumen-Methode im Raum und mittels des impliziten BDF-Verfahrens zweiter Ordnung in der Zeit diskretisiert. Es wird ein blockstrukturiertes Gitter verwendet. Da der verwendete Fluidlöser OpenFOAM eine zweidimensionale Problemstellung nicht abbilden kann, wird ein dreidimensionales Gitter mit einem einzelnen Element der Dicke $1,0 \text{ m}$ in die z -Richtung verwendet und es werden keine Variablen in die dritte Richtung gelöst. Das Fluidnetz besteht aus 54.300 Elementen, damit ergeben sich 270.000 Freiheitsgrade für die Unbekannten \mathbf{d}_x^F , \mathbf{d}_y^F , p , \mathbf{u}_x und \mathbf{u}_y pro Element. Eine Detailansicht des Fluidnetzes aus einem Bereich um die Struktur ist in Abb. 6.3 gegeben.

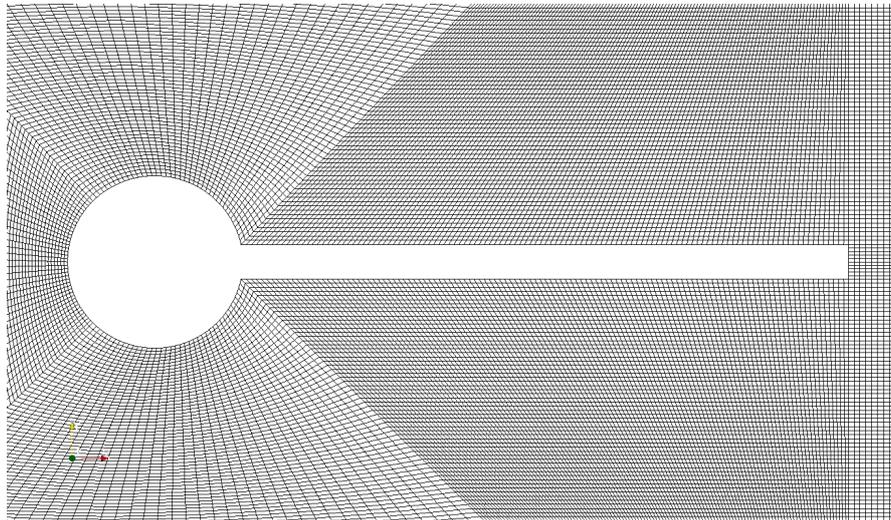


Abbildung 6.3.: Fluidnetzdetail um den Kopplungsbereich

Strukturfeld: Das Strukturfeld wird mittels der Finite-Elemente-Methode im Raum und mittels des Generalized- α Zeitintegrationsverfahrens in der Zeit diskretisiert. Zur Abbildung der großen Verformungen wird eine geometrisch nichtlineare Formulierung benutzt. Da der Zylinder starr ist, muss das Strukturfeld nur den am Zylinder angeschlossenen elastischen Balken abbilden. Dies entspricht dem Strukturmodell eines Kragarms. Um die durch den Fluidlöser vorgegebene Pseudodreidimensionalität abbilden zu können, wird ein dreidimensionales Strukturmodell benutzt und über die Wahl der Randbedingungen das gewünschte zweidimensionale Verhalten erzwungen. Dazu wird die Mittelfläche des Kragarms (siehe gestrichelte Linie in Abb. 6.2) mit vierknotigen Schalenelementen

vom Typ Reissner-Mindlin in der x - z -Ebene diskretisiert. Die Dicke der Schale entspricht damit der Höhe h des Balkens.

Durch diese Dimensionsreduzierung auf die Mittelfläche entspricht die gekoppelte Oberfläche nicht mehr der geometrischen Darstellung des FE-Strukturmodells. Deshalb ist die Einführung eines Transferoperators notwendig, welcher die in der Kopplung auszutauschenden Informationen zwischen der gekoppelten Oberfläche und der Mittelfläche transformiert. Im Falle der Fluid-Struktur-Wechselwirkung ergeben sich Verschiebungen und Kräfte als auszutauschende Informationen. Das hier verwendete Schalenelement basiert auf einer Diskretisierung der Mittelfläche. Es verwendet 6 Freiheitsgrade pro Knoten, davon 3 Verschiebungen der Mittelfläche und 3 Verschiebungen an der Oberfläche. Auf die Eigenheiten der verwendeten Schalenformulierung soll hier aber nicht weiter eingegangen werden. Als weitergehende Literatur empfiehlt sich [Bis99].

Aus der Zweidimensionalität des Problems folgt ein ebener Spannungszustand in der x - y -Ebene. Die Spannungen des Strukturmodells in z -Richtung müssen also Null sein. Daraus folgen die in Abbildung 6.4 angegebenen Randbedingungen des Strukturmodells. Die durch den Querdehneffekt auftretenden Dehnungen in z -Richtung können so spannungsfrei abgetragen werden. Das Strukturnetz besteht aus 30 Elementen und 62 Knoten, damit ergeben sich 366 Freiheitsgrade für das Strukturfeldmodell.



Abbildung 6.4.: Randbedingungen am Strukturmodell

Anmerkung 1: Für dieses Problem wurde eine Netzstudie erstellt, deren Ergebnisse hier aber nicht dargestellt werden. Die hier verwendeten Netze und Elementanzahlen werden gewählt, weil sich dabei ein guter Kompromiss aus Genauigkeit der Ergebnisse und geringem numerischem Aufwand ergeben hat.

Anmerkung 2: Als Initialposition für die gekoppelte Simulation wird für das Strukturfeld die unverformte und ruhende Ausgangslage benutzt. Für das Fluidfeld werden die konvergierten Ergebnisse einer Einzelfeldberechnung mit starren Rändern verwendet. Im Fall FSI2 führt dies zu einem stationären Zustand, im Fall FSI3 zu einer transienten Strömung mit periodisch ablösenden Wirbeln im Zylindernachlauf (von Karmann-Straße). Hier wird als Startwert der Zeitpunkt des Nulldurchgangs der in y -Richtung auf den Balken wirkenden aufsummierten Kräfte benutzt.

Anmerkung 3: Ein Zeitschritt ist konvergiert, wenn das auf die Interfaceverschiebung angewendete relative Konvergenzkriterium, definiert nach Abschnitt 4.3, die Bedingung

$\epsilon_T \leq 10^{-5}$ erfüllt. Dies stellt ein relativ strenges Konvergenzkriterium dar und ist bei Vergleichen mit Werten aus anderen Arbeiten zu beachten.

Anmerkung 4: Die Courant-Zahl c des Fluides (auch CFL-Zahl genannt) ist ein Maß für die Nichtlinearität des Fluidfeldes in Abhängigkeit der Zeitschrittweite und bestimmt sich über:

$$c = \frac{u\Delta t}{\Delta x}, \quad (6.2)$$

mit der charakteristischen Geschwindigkeit u und der diskreten Gitterweite Δx . Für $c = 1$ wird eine Information also pro Zeitschritt um genau eine Gitterzelle weiter transportiert. Für $c > 1$ werden pro Zeitschritt Zellen übersprungen. Bei expliziten Zeitintegrationsverfahren führt dies zu Stabilitätsproblemen. Die hier verwendeten impliziten Zeitintegrationsverfahren sind zwar unbedingt stabil, die Änderungsrate der Fluidvariablen pro Zeitschritt erhöht sich aber. Daraus folgt auch eine höhere Nichtlinearität in der Kopplung. Um diesen Einfluss untersuchen zu können, werden in den beiden Fällen jeweils zwei Zeitschrittweiten untersucht. Die untersuchten Zeitschrittweiten sind für den Fall FSI2 $\Delta t = 5,0e^{-04} s$ und $\Delta t = 20,0e^{-04} s$. Dies korreliert mit einer maximalen Courantzahl von 0,8 und 3,2 für die unverformte Fluidkonfiguration. Für den Fall FSI3 werden $\Delta t = 2,5e^{-04} s$ und $\Delta t = 10,0e^{-04} s$. Dies korreliert ebenfalls mit einer maximalen Courantzahl von 0,8 und 3,2 für die unverformte Fluidkonfiguration.

Anmerkung 5: Alle Berechnungen wurden auf einem Hochleistungsrechner durchgeführt. Dazu wurde der Fluidlöser parallel auf 3 Prozessoren ausgeführt, Strukturlöser und Kopplungsprozess seriell auf jeweils einem eigenen Prozessor. Es wurde sichergestellt, dass die während einer Simulation benutzten Knoten nur der Simulation zugehörige Prozesse ausführen. Damit sind die benötigten Rechenzeiten zwischen den einzelnen Simulationen vergleichbar. Bei allen Simulationen werden die benötigten Rechenzeiten nur relativ zueinander verglichen. Absolute Werte werden nicht angegeben, da sie weniger Aussagekraft besitzen.

Anmerkung 6: Für die abschließende Beurteilung der möglichen Effizienzsteigerung mittels einer Kombination der hier vorgestellten Methoden ist es notwendig, eine Basis zu definieren. Diese wird, ohne jeglichen Anspruch auf allgemeine Gültigkeit, folgendermaßen festgelegt: Die Feldlöser verwenden als Startwert in jeder Kopplungsiteration die am Ende des letzten Zeitschritts bestimmten Werte („konstanter Feldlöser“). Als Prädiktor der Interfaceposition wird ein Polynomprädiktor erster Ordnung verwendet. Als Kopplungsalgorithmus dient eine Fixpunktiteration mit adaptiver Unterrelaxation nach Aitken.

6.1.2. Simulationsergebnisse

Durch die asymmetrische Geometrie und Ablöseerscheinungen am Zylinder entstehen Wirbel im Zylindernachlauf, die auf den elastischen Balken treffen und ihn nach einer Einschwingphase zu periodischen Schwingungen mit konstanter Amplitude und Frequenz

anregen. Dabei ist die prinzipielle Schwingungscharakteristik in den Fällen FSI2 und FSI3 gleich, sie unterscheiden sich jedoch in Amplituden- und Frequenzhöhe. Im Folgenden werden die Ergebnisse für den zeitlichen Verlauf der Verschiebungen am Punkt **A** (siehe Abb. 6.2) und der Summe der auf Zylinder und Balken einwirkenden Kräfte dargestellt. Die Darstellung der Simulationsergebnisse ist dem Vergleich der einzelnen Methoden vorangestellt, da diese unabhängig von den verwendeten Methoden sind.

6.1.2.1. Fall FSI2

Der zeitliche Verlauf der Verschiebungen in x- und y-Richtung am Punkt **A** ist in Abb. 6.5 dargestellt. Nach einer Einschwingphase mit stetig zunehmender Amplitude stellt sich ab dem Zeitpunkt $t = 6,1 \text{ s}$ eine harmonische Schwingung mit konstanter Frequenz und Amplitude ein. Die Auswertung der auf die Struktur wirkenden Kräfte ist in Abb. 6.6 dargestellt. Sie verhalten sich analog zu den Schwingungen. Ein Überblick zu den Ergebnissen und ein Vergleich mit den in [TH06] vorgestellten Referenzwerten ist in Tabelle 6.2 gegeben. Eine Visualisierung der Geschwindigkeiten des Strömungsfeldes an drei verschiedenen Zeitpunkten ist in Abb. 6.7 gegeben.

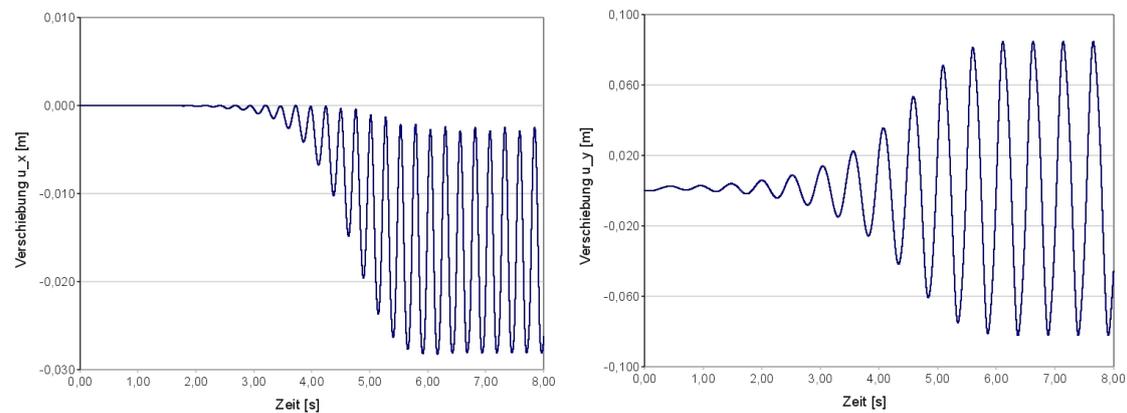


Abbildung 6.5.: FSI2 - Verschiebungsverlauf am Punkt A in x- und y-Richtung

	eigene Ergebnisse	Referenzwerte ([TH06])
$\mathbf{d}_x(\mathbf{A}) [m]$	$-0,01538 \pm 0,0127$	$-0,01458 \pm 0,0124$
$\mathbf{d}_y(\mathbf{A}) [m]$	$0,00123 \pm 0,0834$	$0,00123 \pm 0,0806$
$\sum_x \mathbf{f}^\Gamma [N]$	$224,63 \pm 85,96$	$208,83 \pm 73,75$
$\sum_y \mathbf{f}^\Gamma [N]$	$-0,474 \pm 268,08$	$-0,88 \pm 234,2$
$F_x [s^{-1}]$	3,9	3,8
$F_y [s^{-1}]$	1,94	2,0

Tabelle 6.2.: FSI2 - eigene Ergebnisse und Referenzwerte

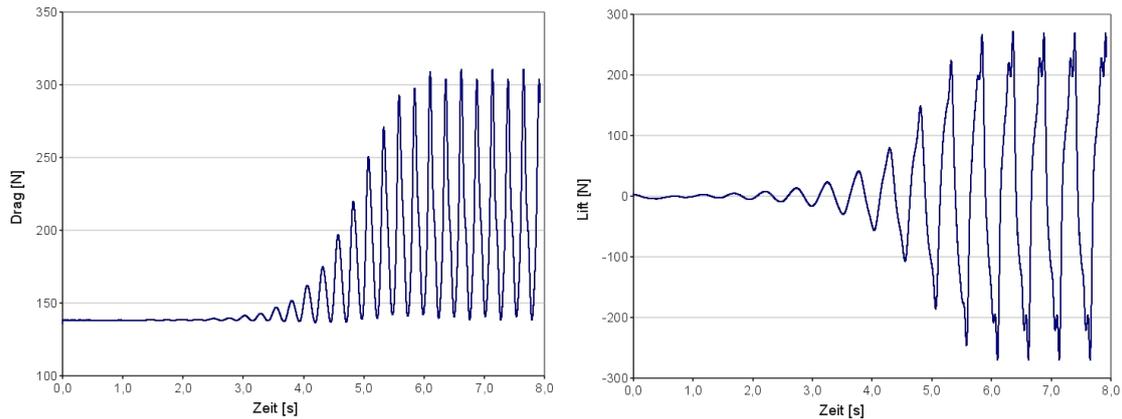


Abbildung 6.6.: FSI2 - Verlauf der summierten Kräfte am Interface in x- und y-Richtung

6.1.2.2. Fall FSI3

Der zeitliche Verlauf der Verschiebungen in x- und y-Richtung am Punkt **A** ist in Abb. 6.8 dargestellt. Nach einer kurzen Einschwingphase stellt sich ab dem Zeitpunkt $t = 0,7$ s eine harmonische Schwingung mit konstanter Frequenz und Amplitude ein. Im Vergleich mit FSI2 ist die Einschwingphase deutlich kürzer und die Schwingungsamplitude deutlich kleiner. Die Auswertung der auf die Struktur wirkenden Kräfte ist in Abb. 6.9 dargestellt. Sie verhalten sich analog zu den Schwingungen. Ein Überblick zu den Ergebnissen und ein Vergleich mit den in [TH06] vorgestellten Referenzwerten ist in Tabelle 6.3 gegeben. Eine Visualisierung der Geschwindigkeiten des Strömungsfeldes an drei verschiedenen Zeitpunkten ist in Abb. 6.10 gegeben.

	eigene Ergebnisse	Referenzwerte ([TH06])
$\mathbf{d}_x(\mathbf{A})$ [m]	$-0,00304 \pm 0,00287$	$-0,002885 \pm 0,00272$
$\mathbf{d}_y(\mathbf{A})$ [m]	$0,00155 \pm 0,0366$	$0,00147 \pm 0,0350$
$\sum_x \mathbf{f}^I$ [N]	$474,9 \pm 28,12$	$460,5 \pm 27,74$
$\sum_y \mathbf{f}^I$ [N]	$3,86 \pm 165,9$	$2,50 \pm 153,9$
F_x [s^{-1}]	10,99	10,93
F_y [s^{-1}]	5,51	5,46

Tabelle 6.3.: FSI3 - eigene Ergebnisse und Referenzwerte

Anmerkung: Bei dem in Tabelle 6.2 und 6.3 angegebenen Referenzwerten ist zu beachten, dass diese die Ergebnisse einer einzelnen Gruppe darstellen. Betrachtet man die Ergebnisse aller an dieser Berechnung teilnehmenden Gruppen (siehe A.1 und für den Fall FSI3 auch [THR⁺10]), erkennt man die relativ hohe Streuung der einzelnen Ergebniswerte. Für diese hochgradig nichtlineare und komplexe Simulation sind diese Abweichungen nicht überraschend, da jede Gruppe unterschiedliche numerische Modelle verwendet (zeitliche und räumliche Diskretisierungsverfahren, numerische Netze, Konvergenzkriterien, etc.). Die hier vorgestellten eigenen Ergebnisse befinden sich klar innerhalb

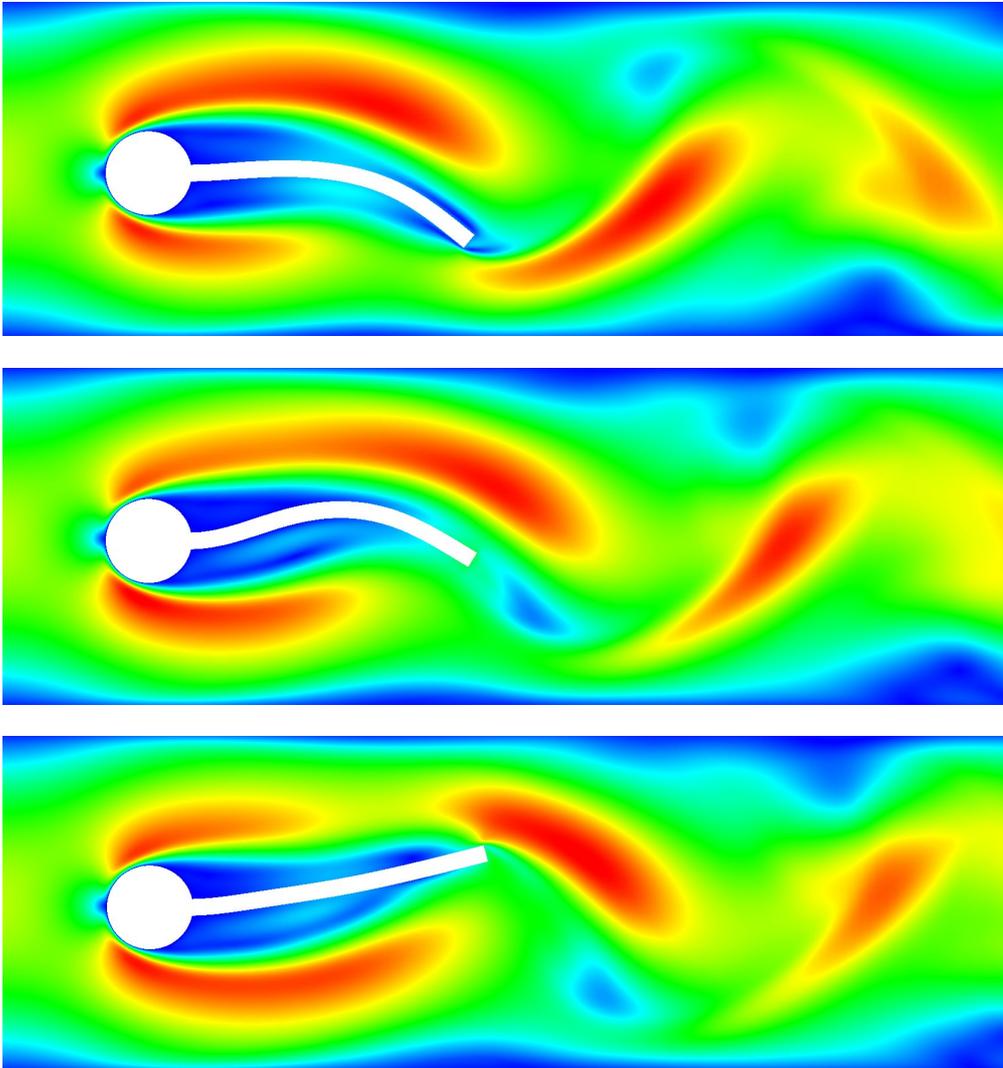


Abbildung 6.7.: FSI2 - Geschwindigkeitsvisualisierung des Strömungsfeldes an drei Zeitpunkten

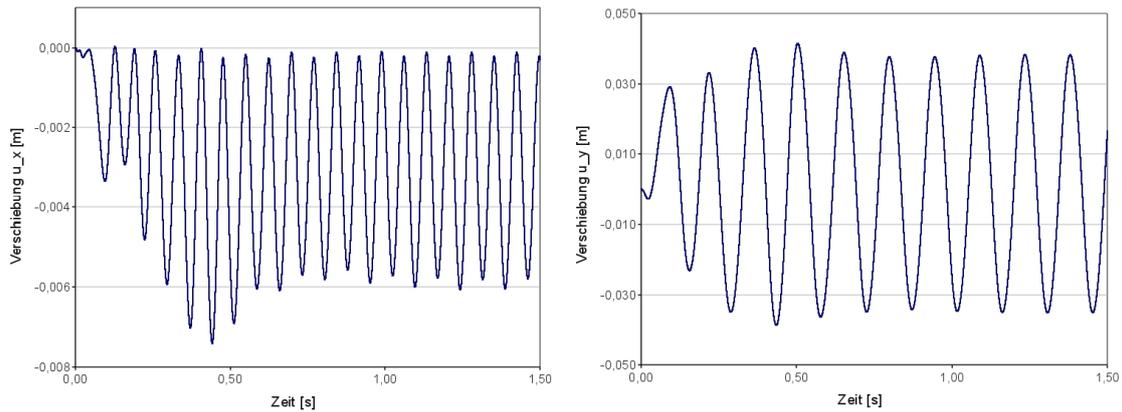


Abbildung 6.8.: FSI3 - Verschiebungsverlauf am Punkt A in x- und y-Richtung

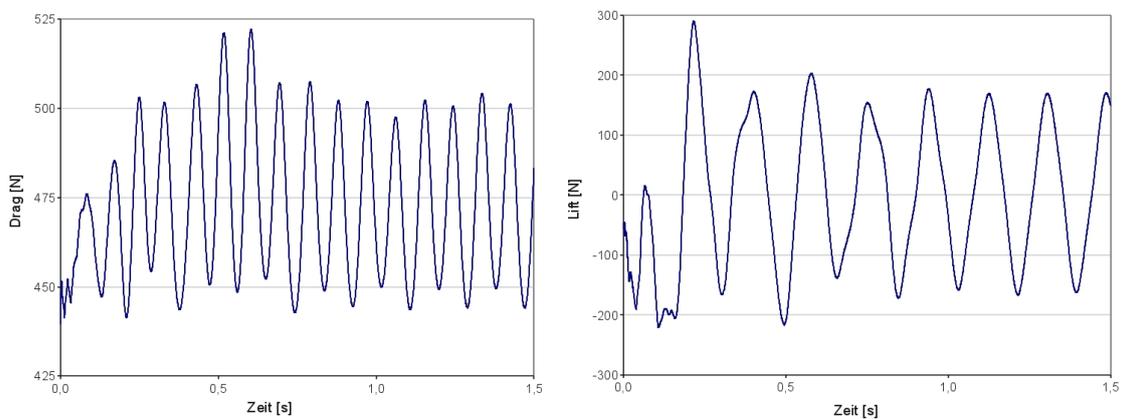


Abbildung 6.9.: FSI3 - Verlauf der summierten Kräfte am Interface in x- und y-Richtung

des Ergebnishorizonts der anderen Gruppen. Damit werden die erzielten Ergebnisse als qualitativ hochwertig erachtet und die verwendete Softwareumgebung gilt als validiert.

6.1.3. Konstante versus angepasste Feldlöser

In diesem Beispiel wird der in den Feldlösern genutzte Ansatz zur Abbildung einer impliziten Kopplung näher betrachtet. Während jeder Kopplungsiteration benutzen die nichtlinearen Feldlöser einen Initialwert als Startwert für die Felditerationen. Für diesen Initialwert können die Ergebnisse des letzten Zeitschritts angenommen werden. Die Initialwerte sind damit über die Kopplungsiterationen konstant („konstanter Feldlöser“). Des Weiteren besteht die Möglichkeit, die Ergebnisse der letzten Kopplungsiteration als Initialwert zu verwenden, die Feldlöser also an den Kopplungsverlauf zu adaptieren („angepasste Feldlöser“). Unter Annahme stetig konvergierender Interfacevariablen stellen die Ergebnisse der letzten Kopplungsiteration einen Initialwert dar, der näher an der Lösung liegt als die Ergebnisse des letzten Zeitschritts. Daraus folgt eine geringere Anzahl an

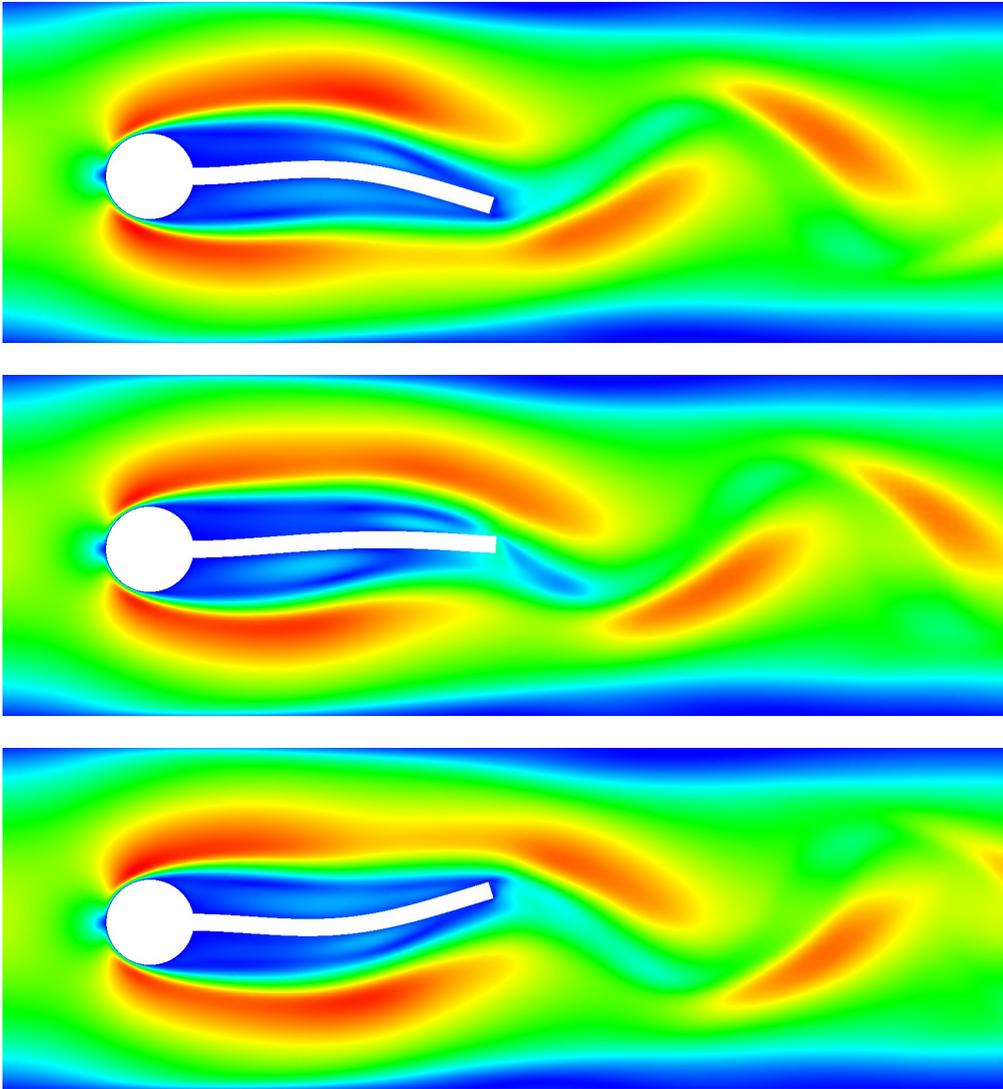


Abbildung 6.10.: FSI3 - Geschwindigkeitsvisualisierung des Strömungsfeldes an drei Zeitpunkten

notwendigen Feldlöseriterationen und damit eine Steigerung der Effizienz. Details für die algorithmische Umsetzung innerhalb der Feldlöser finden sich in den Kapiteln 2.5 und 3.6. Dieser Aspekt wird in diesem Abschnitt näher untersucht.

Um einen Vergleich durchführen zu können, werden Feldlöser verwendet, die sich nur in der Wahl der Initialwerte unterscheiden, ansonsten aber identisch sind. Für die Auswertung ist es notwendig, auf die spezifischen Eigenschaften der Feldlöser einzugehen. Als absolutes Kriterium dient die benötigte CPU-Zeit. Damit ist eine eindeutige Aussage über die erzielbare Effizienzsteigerung möglich. Für eine differenziertere Betrachtung werden zusätzlich die maßgebenden Iterationen angegeben. Hierbei wird jeweils über den gesamten Berechnungszeitraum gemittelt. Für das Strukturfeld sind die benötigten Newtoniterationen wesentlich. Da der im Fluidfeld verwendete PISO-Algorithmus eine fixe Anzahl an Druckiterationen ausführt, ist diese Zahl nicht aussagekräftig. Der zum Lösen der Druckgleichung verwendete Mehrgitterlöser nutzt die Ergebnisse der letzten Iteration als Initialwerte. Damit zeigen sich die Unterschiede zwischen den beiden Ansätzen in den benötigten Iterationen des Gleichungslösers. Deshalb werden die Iterationszahlen für die Auswertung genutzt. Für die Kopplung maßgebend sind die benötigten Kopplungsiterationen. Für den Fall, dass beide Feldlöser in jeder Kopplungsiteration voll auskonvergiert sind, müssen die Kopplungsiterationen bei konstanten und angepassten Ansätzen identisch sein. Durch die Verwendung des PISO-Ansatzes ist die vollständige Konvergenz des Fluidfeldes aber nur durch Vorgabe einer hohen Anzahl an PISO-Iterationen sichergestellt, was aber im direkten Vergleich den konstanten Ansatz benachteiligt. Die PISO-Iterationen werden deshalb so vorgegeben, dass die Interfaceverschiebungen innerhalb der Interfacetoleranz bei konstantem und angepasstem Ansatz identisch sind.

6.1.3.1. Fall FSI2

Im Fall FSI2 werden die ersten 500 Zeitschritte mit einer Zeitschrittweite von $\Delta t = 5,0e^{-4}$ s berechnet. Der Interfaceprädiktor ist vom Typ Newmark 1. Ordnung. Als Kopplungsalgorithmus wird eine Fixpunktiteration mit adaptiver Aitken-Unterrelaxation benutzt. Die vergleichenden Berechnungsergebnisse sind in Tabelle 6.4 dargestellt.

	relative Rechenzeit	Kopplungs- iterationen	Newton- iterationen	Drucklöser- iterationen
konstant	1,000	10,414	933	89.388
angepasst	0,855	11,071	720	72.373
Vergleich [%]	-14,5	+6,3	-22,8	-19,1

Tabelle 6.4.: Konstante versus angepasste Feldlöser - FSI2

Durch die Verwendung des angepassten Ansatzes sinkt die Berechnungszeit um 14,5 %. Die Anzahl an Kopplungsiterationen steigt leicht um 6,3 %. Dieser erhöhte Aufwand wird aber durch den erzielten Effizienzgewinn als Folge der Verringerung der notwendigen Newton- und Drucklöseriterationen deutlich relativiert.

6.1.3.2. Fall FSI3

Im Fall FSI3 werden die ersten 300 Zeitschritte mit einer Zeitschrittweite von $\Delta t = 2,5e^{-4}$ s berechnet. Der Interfaceprädiktor ist vom Typ Newmark 1. Ordnung. Als Kopplungsalgorithmus wird eine Fixpunktiteration mit adaptiver Aitken-Unterrelaxation benutzt. Die vergleichenden Berechnungsergebnisse sind in Tabelle 6.5 dargestellt.

	relative Rechenzeit	Kopplungsiterationen	Newtoniterationen	Drucklöseriterationen
konstant	1,000	21,036	1.881	270.843
angepasst	0,763	24,310	1.856	157.312
Vergleich [%]	-23,7	+15,6	-0,9	-41,9

Tabelle 6.5.: Konstante versus angepasste Feldlöser - FSI3

Durch die Verwendung des angepassten Ansatzes sinkt die Berechnungszeit signifikant um 23,7 %. Die Anzahl an Kopplungsiterationen steigt hingegen um 15,6 %. Dieser erhöhte Aufwand wird aber durch den erzielten Effizienzgewinn über eine Verringerung der notwendigen Newton- und Drucklöseriterationen deutlich relativiert. Hier ist vor allem zu beachten, dass die Anzahl an notwendigen Drucklöseriterationen um 41,9 % sinkt.

6.1.3.3. Schlussfolgerung

Insgesamt ist durch die Verwendung angepasster Feldlöser eine klar sichtbare Effizienzsteigerung zu erzielen. Wie zu erwarten, ist die erzielbare Steigerung um so höher, je mehr Kopplungsiterationen benötigt werden bzw. je stärker der Kopplungsgrad ist. Der dazu notwendige Aufwand ist gering, da nur die Initialwerte der Felder angepasst werden müssen. Bei Black-Box-Feldlösern ist die verwendete Methodik nicht zu beeinflussen, bei offenen Lösern folgt daraus eine klare Empfehlung für eine entsprechende Anpassung der Einzelfeldlösungsalgorithmen.

Als Nachteil ist zu bemerken, dass die Anzahl an Kopplungsiterationen durch angepasste Feldlöser leicht ansteigt. Wie bereits in Abschnitt 6.1.3 ausgeführt, liegt dies an der fest vorgegebenen Anzahl an PISO-Iterationen im Fluidfeldlöser. Wird eine sehr hohe Anzahl vorgegeben, ist der Fluidlöser in jeder Kopplungsiteration vollständig konvergiert und die Anzahl an Kopplungsiterationen ist bei konstantem und angepasstem Ansatz identisch. Im direkten Vergleich wird dadurch aber der konstante Ansatz benachteiligt, da bereits bei einer deutlich geringeren Anzahl an PISO-Iterationen eine ausreichende Konvergenz vorliegt und das gekoppelte Problem gelöst werden kann.

6.1.4. Einfluss des Interfaceprädiktors

Ein wesentlicher Bestandteil einer impliziten Kopplung ist die Bestimmung einer Startposition des Interfaces in der ersten Kopplungsiteration. Je näher die Startposition an der bei Konvergenz bekannten endgültigen Position des Interfaces liegt, desto schneller konvergiert das gekoppelte Problem. Ist die Startposition zu weit von der endgültigen

Position entfernt, besteht auch die Möglichkeit einer Divergenz (vergleiche auch lokale Konvergenzeigenschaften von Newtonverfahren).

Da die endgültige Interfaceposition nicht bekannt ist, kommen für die Prädiktion extrapolative Verfahren in Betracht, welche auf Grundlage bekannter Ergebnisse aus vorherigen Zeitschritten eine Startposition bestimmen. Die hier untersuchten Verfahren werden in Kapitel 4.4 im Detail vorgestellt.

Dazu zählen:

- Polynombasierte Prädiktoren nullter („P0“), erster („P1“), zweiter („P2“) und dritter („P3“) Ordnung, die auf Basis vorheriger Positionen eine rein geometrische Extrapolation vornehmen.
- Newmarkbasierte Prädiktoren erster („N1“) und zweiter („N2“) Ordnung, welche durch die Bestimmung von Interfacegeschwindigkeit und -beschleunigung die Physik des gekoppelten Problems mit höherer Genauigkeit abbilden.
- Adaptive Newmarkprädiktoren zweiter Ordnung („AN“), welche zusätzlich die zeitliche Veränderlichkeit des Problems abbilden, indem ein Minimierungsproblem gelöst wird und entsprechende Parameter angepasst werden.

Es werden die Fälle FSI2 und FSI3 untersucht, wobei angepasste Feldlöser und Fixpunktiteration mit adaptiver Aitken-Unterrelaxation angewendet werden. Da eine anwachsende Zeitschrittweite die Prädiktion schwieriger macht, werden in beiden Fällen zwei unterschiedliche Zeitschrittweiten untersucht. Für den Vergleich werden über einen bestimmten Zeitraum gemittelte Werte angegeben. Dazu zählen das mittels des Prädiktors erzielte Startresiduum am Interface (definiert nach Abschnitt 4.3), welches sich nach der ersten Kopplungsiteration bestimmen lässt, die Anzahl benötigter Kopplungsiterationen und die relative CPU-Zeit als die zur Beurteilung der Effizienzsteigerung am besten geeigneten Kriterien.

6.1.4.1. Fall FSI2

Die untersuchten Zeitschrittweiten sind $\Delta t = 5,0e^{-4}$ s mit einer maximalen Courant-Zahl von 0,8 und $\Delta t = 20,0e^{-4}$ s mit einer maximalen Courant-Zahl von 3,2. Es werden jeweils die ersten 500 Zeitschritte simuliert. Die Ergebnisse sind in den Tabellen 6.6 und 6.7 dargestellt.

Generell lässt sich bei beiden Zeitschrittweiten ein hoher Einfluss des Prädiktors auf die Lösungseffizienz feststellen. Bei den polynombasierten Prädiktoren ergibt sich im Vergleich von P2 zu P0 eine Rechenzeitverringerung um 53,4 % bzw. 49,4 %. Bei Verwendung von P3 ergibt sich ein starker Anstieg des Startresiduums. Dies erklärt sich durch den sogenannten „Artificial Added Mass Effect“ ([FWR06, FWR07, LTM01, Nob01, CGN05]). Dieser besagt, dass bei einer expliziten Kopplung die Verwendung von Prädiktoren höherer Ordnung zu einem größeren Fehler in der Kopplung führen, da durch hochfrequente Lösungsanteile zusätzliche „künstliche“ Massen am Interface erzeugt werden. Da der Prädiktorschritt der impliziten Kopplung einem Zeitschritt der expliziten Kopplung

	relatives Startresiduum	Kopplungs- iterationen	relative Rechenzeit
P0	1,0000	12,588	1,000
P1	0,0357	9,500	0,641
P2	0,0120	7,292	0,466
P3	2,9495	12,538	0,981
N1	0,0193	8,984	0,603
N2	0,0082	6,844	0,428
AN	0,0063	7,548	0,487

Tabelle 6.6.: Prädiktoreinfluss FSI2 - $\Delta t = 5,0e^{-04} s$

	relatives Startresiduum	Kopplungs- iterationen	relative Rechenzeit
P0	1,0000	12,748	1,000
P1	0,0650	10,558	0,736
P2	0,0191	7,716	0,506
P3	2,6026	12,674	0,990
N1	0,0345	9,948	0,670
N2	0,0142	7,222	0,466
AN	0,0148	7,958	0,524

Tabelle 6.7.: Prädiktoreinfluss FSI2 - $\Delta t = 20,0e^{-04} s$

entspricht, sind diese Effekte hier besonders schön zu sehen. Die Anzahl an Kopplungsiterationen und Rechenzeit bleibt im Vergleich mit P0 aber nahezu unverändert, was an der spezifischen Charakteristik der Aitken-basierten Kopplung liegt. Die Methoden N1 und N2 führen beide zu geringen Startresiduen und damit auch Kopplungsiterationen und Rechenzeit, wobei im Vergleich von N2 mit P0 eine Rechenzeitreduktion um 57.2 % bzw. 53.4 % auftritt. Die Verwendung des adaptiven Prädiktors AN ist qualitativ im Bereich von N2 anzusiedeln, vermag aber keine weitere Rechenzeitreduktion zu erreichen. Der Einfluss der unterschiedlichen Zeitschrittweite ist wie erwartet. Bei größerem Zeitschritt sind die Startresiduen prinzipiell höher, der Startwert weicht also stärker von der Lösung ab. Das relative Verhalten der Prädiktoren zueinander bleibt aber nahezu unverändert.

6.1.4.2. Fall FSI3

Die untersuchten Zeitschrittweiten sind $\Delta t = 2,5e^{-4} s$ mit einer maximalen Courant-Zahl von 0,8 und $\Delta t = 10,0e^{-4} s$ mit einer maximalen Courant-Zahl von 3,2. Es werden jeweils die ersten 300 Zeitschritte simuliert. Die Ergebnisse sind in den Tabellen 6.8 und 6.9 dargestellt.

Generell ist das Verhalten der Prädiktoren ähnlich zum Fall FSI2. Die erzielbaren Reduktionen bewegen sich in der selben Größenordnung, die Unterschiede sind minimal. Methode P3 führt wiederum zu einem starken Anstieg des Startresiduums, wodurch aber Kopplungsiterationen und CPU-Zeit kaum beeinflusst werden. Dies ist wiederum auf den

	relatives Startresiduum	Kopplungs- iterationen	relative Rechenzeit
P0	1,0000	30,673	1,000
P1	0,0909	21,483	0,611
P2	0,0105	16,423	0,437
P3	1,3678	30,283	0,969
N1	0,0498	20,083	0,569
N2	0,0083	15,413	0,408
AN	0,0155	16,393	0,452

Tabelle 6.8.: Prädiktoreinfluss FSI3 - $\Delta t = 2,5e^{-04} s$

	relatives Startresiduum	Kopplungs- iterationen	relative Rechenzeit
P0	1,0000	42,850	1,000
P1	0,2730	34,389	0,799
P2	0,0204	25,083	0,523
P3	1,4179	41,817	0,970
N1	0,1489	32,173	0,722
N2	0,0127	23,641	0,479
AN	0,0292	28,365	0,583

Tabelle 6.9.: Prädiktoreinfluss FSI3 - $\Delta t = 10,0e^{-04} s$

„Artificial Added Mass Effect“ zurückzuführen. Der adaptive Newmarkprädiktor AN ist qualitativ im Bereich des N2 angesiedelt, führt also zu keiner weiteren Effizienzsteigerung. Das Verhalten der Methoden relativ zueinander ist bei unterschiedlichen Zeitschrittweiten gleich, bei größerem Zeitschritt steigt das Startresiduum aber bei allen Methoden an. Bei Verwendung des Prädiktors N2 kann im Vergleich mit P0 die Rechenzeit um 59,2 % bzw. 52,1 % verringert werden.

6.1.4.3. Schlussfolgerung

Die Wahl des Interfaceprädiktors beeinflusst den Kopplungsverlauf bei stark gekoppelten Problemen massiv. Durch die Wahl eines Newmarkprädiktors zweiter Ordnung (N2) kann im Vergleich mit einem konstanten Prädiktor (P0) die Rechenzeit um bis zu 59,2 % verringert werden. Auch im Vergleich zu einem Prädiktor erster Ordnung (P1), welcher in vielen Arbeiten verwendet wird, ergeben sich im Fall FSI3 mit $\Delta t = 10,0e^{-04} s$ Rechenzeitverringern um 40,0 %. Der Implementierungsaufwand für einen Newmarkprädiktor ist gering, auch der zusätzliche Speicheraufwand ist vernachlässigbar. Der adaptive Newmarkprädiktor AN verhält sich in der Gesamtbetrachtung ebenfalls sehr gut, aber geringfügig schlechter als N2. Da AN zusätzlich die Lösung eines Minimierungsproblems nötig macht, ist dieser Aufwand nicht gerechtfertigt. Zusammenfassend ist N2 die am besten geeignete Methode, da sie hohe Effizienz mit geringem Aufwand kombiniert.

6.1.5. Einfluss des Kopplungsalgorithmus

Der zweite wesentliche Bestandteil einer impliziten Kopplung ist neben der Bestimmung der Interfacestartposition der verwendete Kopplungsalgorithmus. Kopplungsalgorithmen können hierbei nach drei Gesichtspunkten beurteilt werden:

1. Effizienz: Die Effizienz eines Kopplungsalgorithmus beeinflusst die Anzahl der notwendigen Kopplungsiterationen pro Zeitschritt.
2. Stabilität: Die Stabilität eines Kopplungsalgorithmus beeinflusst die mögliche Zeitschrittweite.
3. Implementierungsaspekte: Bei Kopplungsalgorithmen ist zwischen Methoden zu unterscheiden, bei denen die Feldlöser Standardlösungsverfahren benutzen können, und Methoden, welche eine spezifische Anpassung der Feldlöseralgorithmen erfordern.

In diesem Beispiel werden alle der im Kapitel 3 in den Abschnitten 4.6 bis 4.8 vorgestellten Kopplungsalgorithmen miteinander verglichen. Dazu zählen folgende Methoden:

- FPA: Fixpunktiteration mit adaptiver Aitken-Relaxation (siehe 4.6).
- VE-MPE und VE-RRE: Vektorextrapolation mit MPE- oder RRE-Methodik (siehe 4.7).
- QN und QN-NC: Quasi-Newton-Verfahren mit und ohne konstanten Relaxationsschritt (-NC, siehe 4.8.2).
- NKFD und NKLP: Inexakte Newton-Krylov-Verfahren mit Krylovraumevaluierung über finite Differenzen oder linearisiertes Problem (siehe 4.8.1).

Bei den QN- und NK-Verfahren gibt es noch die Möglichkeit der Wiederverwendung von Informationen aus n Schritten, dies wird über ein „R(n)“ gekennzeichnet. Bei den NK-Verfahren ist in Klammern die durchschnittliche Anzahl an GMRES-Iterationen angegeben. Es werden wieder die Fälle FSI2 und FSI3 untersucht. Ausgewertet werden die über einen gewissen Zeitraum gemittelten Kopplungsiterationen und die benötigte Rechenzeit. Die Effizienz leitet sich ab aus einem Vergleich der Kopplungsiterationen bzw. der Rechenzeiten. Die Stabilität ergibt sich aus dem Verhalten bei verschiedenen Zeitschrittweiten. Die Implementierungsaspekte werden bei der Herleitung der Verfahren betrachtet.

Anmerkung: Das Newton-Krylov-Verfahren mit Krylovraumevaluierung durch ein Ersatzproblem (siehe Abschnitt 4.8.1.5) divergiert in allen hier vorgestellten Fällen nach mehr oder weniger Zeitschritten. Wie bereits erwähnt, ist die Evaluationsqualität eines Ersatzproblems stark problemspezifisch, was sich an diesen Beispielen deutlich zeigt. Dieses Verfahren wird deswegen in den Vergleichen nicht weiter berücksichtigt.

6.1.5.1. Fall FSI2

Die untersuchten Zeitschrittweiten sind $\Delta t = 5,0e^{-4}$ s mit einer maximalen Courant-Zahl von 0,8 und $\Delta t = 20,0e^{-4}$ s mit einer maximalen Courant-Zahl von 3,2. Es werden jeweils die ersten 500 Zeitschritte simuliert. Die Ergebnisse sind in den Tabellen 6.10 und 6.11 dargestellt.

	Kopplungsiterationen	relative Rechenzeit
FPA	8,984	1,000
VE-MPE(5)	9,008	0,959
VE-MPE(10)	12,464	1,291
VE-RRE(5)	8,92	0,952
VE-RRE(10)	11,618	1,230
QN	5,356	0,618
QN-R(10)	3,562	0,401
QN-R(20)	3,470	0,391
QN-NC-R(10)	3,650	0,410
QN-NC-R(20)	3,486	0,392
NKFD	5,126 (11,146)	2,231
NKFD-R(10)	5,006 (6,404)	1,553
NKFD-R(20)	5,012 (5,116)	1,342
NKLP	5,612 (12,320)	0,942
NKLP-R(10)	5,604 (6,182)	0,735
NKLP-R(20)	5,604 (5,450)	0,711

Tabelle 6.10.: Algorithmeinfluss FSI2 - $\Delta t = 5,0e^{-04}$ s

Aitken's Methode (**FPA**) als Standardansatz benötigt 8,984 Iterationen für $\Delta t = 5,0e^{-4}$ s und 9,948 Iterationen für $\Delta t = 20,0e^{-4}$ s, der Einfluss der Zeitschrittweite ist also eher gering.

Die Vektorextrapolationsmethoden (**VE-MPE** und **VE-RRE**) zeigen bei beiden Zeitschrittweiten eine geringere Effizienz als FPA, unabhängig von der Größe der jeweiligen Vektorbasis. Durch manuelles Optimieren ist es möglich, die Effizienz der VE-Methoden geringfügig zu steigern. Allerdings ist dies stark problemabhängig und wird hier nicht weiter verfolgt.

Die Quasi-Newton-Methode in ihrer Basisversion (**QN**) benötigt 5,356 Iterationen für $\Delta t = 5,0e^{-4}$ s und 6,162 Iterationen für $\Delta t = 20,0e^{-4}$ s, damit ergibt sich im Vergleich zu FPA eine Verringerung der Kopplungsiterationen um 40,4 % bzw. 38,1 % und der Rechenzeit um 38,2 % bzw. 37,2 %. Wiederum ist der Einfluss der größeren Zeitschrittweite gering. Aus dem festgelegten Konvergenzkriterium von $\epsilon^{\Gamma} \leq 10^{-5}$ folgt für QN eine in etwa lineare Konvergenzrate.

Die Wiederverwendung von Informationen aus den Kopplungsiterationen vorheriger Zeitschritte (**QN-R**) führt zu einer deutlichen Verringerung der Kopplungsiterationen und damit zu einer Steigerung der Effizienz. Die Wiederverwendung von 20 Leveln führt

	Kopplungs- iterationen	relative Rechenzeit
FPA	9,948	1,000
VE-MPE(5)	12,156	1,142
VE-MPE(10)	13,132	1,262
VE-RRE(5)	11,298	1,068
VE-RRE(10)	12,116	1,178
QN	6,162	0,628
QN-R(10)	6,372	0,628
QN-R(20)	5,966	0,599
QN-NC-R(10)	keine Konvergenz	
QN-NC-R(20)	keine Konvergenz	
NKFD	7,528 (22,762)	3,431
NKFD-R(10)	7,526 (12,358)	2,375
NKFD-R(20)	7,514 (10,148)	2,031
NKLP	5,012 (15,478)	0,829
NKLP-R(10)	5,016 (8,268)	0,660
NKLP-R(20)	5,022 (5,580)	0,593

Tabelle 6.11.: Algorithmuseinfluss FSI2 - $\Delta t = 20,0e^{-04} s$

im Vergleich mit QN für $\Delta t = 5,0e^{-4} s$ zu 3,47 Iterationen und einer Reduktion um zusätzliche 36,7 % der Rechenzeit und für $\Delta t = 20,0e^{-4} s$ zu 5,966 Iterationen und einer Reduktion um zusätzliche 4,6 % der Rechenzeit. Hierbei wird ein großer Einfluß der Zeitschrittweite festgestellt. Dies entspricht der Erwartung, da die Aussagequalität von Informationen aus vorhergehenden Zeitschritten zum aktuellen Zeitpunkt bei steigender Zeitschrittweite abnimmt.

Der Verzicht auf den konstanten Relaxationsschritt (**QN-NC-R**) führt für $\Delta t = 5,0e^{-4} s$ zu keiner weiteren Effizienzsteigerung, es ist sogar ein leichter Anstieg an Iterationen und Rechenzeit festzustellen. Für $\Delta t = 20,0e^{-4} s$ führt der Verzicht sogar zu einer Divergenz der gekoppelten Simulation nach einigen Zeitschritten. Durch den Verzicht auf den konstanten Relaxationsschritt startet ein Zeitschritt, ohne dass eine aktuelle Residuumsinformation gebildet wird. Dies kann bei einer geringen Anzahl an durchschnittlich benötigten Kopplungsiterationen und damit einer geringen Basis an Residuumsinformationen zu Konvergenzproblemen und damit zum Abbruch der Berechnung führen.

Das Newton-Krylov-Verfahren mit finiten Differenzen (**NKFD**) benötigt 5,126 Iterationen für $\Delta t = 5,0e^{-4} s$. Betrachtet man nur die Kopplungsiterationen ist dieses Verfahren etwas besser als QN und ergibt damit eine näherungsweise lineare Konvergenzrate. Für $\Delta t = 20,0e^{-4} s$ benötigt NKFD 7,528 Iterationen. Die Wiederverwendung von Krylov-Rauminformationen (**NKFD-R**) führt zu einer deutlichen Reduktion der benötigten GMRES-Iterationen: für $\Delta t = 5,0e^{-4} s$ um 54,1 % und für $\Delta t = 20,0e^{-4} s$ um 55,4 %. Dabei ist allerdings zu beachten, dass die Krylov-Raumevaluation über finite Differenzen pro Iteration in etwa die gleichen Kosten verursacht wie ein Newton-Updateschritt. Für einen Vergleich müssen die beiden Zahlen also addiert oder die

Rechenzeit betrachtet werden. Hierbei ist klar ersichtlich, wie ineffizient die Finite-Differenzen-Evaluation ist. NKFD benötigt das 3,431-fache von FPA und selbst bei Wiederverwendung von 20 Leveln sind die Rechenzeiten 2,031-fach höher.

Die Verwendung eines linearisierten Modells zur Krylov-Raumevaluation (**NKLP**) reduziert diese Kosten signifikant, da nur zwei lineare Modelle pro GMRES-Iteration ausgewertet werden müssen. Die Linearisierungsqualität ist allerdings stark problemspezifisch. Für $\Delta t = 5,0e^{-4}$ s verschlechtert sich die Konvergenzrate leicht (von 5,126 auf 5,612 Iterationen), für $\Delta t = 20,0e^{-4}$ s verbessert sie sich (von 7,528 auf 5,012 Iterationen). Wiederum verringert die Wiederverwendung von Krylov-Raumevaluationen die Anzahl an benötigten GMRES-Iterationen signifikant (um 55,8 % für $\Delta t = 5,0e^{-4}$ s und um 63,9 % für $\Delta t = 20,0e^{-4}$ s). Die geringen Kosten der Krylovraumevaluation zeigen sich vor allem bei einer Betrachtung der Rechenzeit. Im Vergleich zur Evaluation über finite Differenzen sinken die Rechenzeiten deutlich. Die NKLP-Versionen sind damit effizienter als FPA und VE, aber immer noch deutlich weniger effizient als QN. Der Einfluss der Zeitschrittweite ist sehr gering. Im Fall FSI2 ist das linearisierte Problem in der Lage, den Krylovraum adäquat abzubilden.

6.1.5.2. Fall FSI3

Die untersuchten Zeitschrittweiten sind $\Delta t = 2,5e^{-4}$ s mit einer maximalen Courant-Zahl von 0,8 und $\Delta t = 10,0e^{-4}$ s mit einer maximalen Courant-Zahl von 3,2. Es werden jeweils die ersten 300 Zeitschritte simuliert. Die Ergebnisse sind in den Tabellen 6.12 und 6.13 dargestellt.

	Kopplungs-iterationen	relative Rechenzeit
FPA	20,083	1,000
VE-MPE(5)	23,090	1,249
VE-MPE(10)	keine Konvergenz	
VE-RRE(5)	23,107	1,253
VE-RRE(10)	keine Konvergenz	
QN	10,253	0,540
QN-R(10)	6,567	0,338
QN-R(20)	6,460	0,335
QN-NC-R(10)	5,44	0,271
QN-NC-R(20)	5,067	0,254
NKFD	5,253 (33,76)	2,503
NKFD-R(10)	5,296 (24,62)	1,910
NKFD-R(20)	6,097 (18,990)	1,590
NKLP	11,927 (65,633)	1,331
NKLP-R(10)	11,953 (33,200)	0,963
NKLP-R(20)	11,857 (24,057)	0,878

Tabelle 6.12.: Algorithmuseinfluss FSI3 - $\Delta t = 2,5e^{-04}$ s

	Kopplungs- iterationen	relative Rechenzeit
FPA	32,177	1,000
VE-MPE(5)	keine Konvergenz	
VE-MPE(10)	keine Konvergenz	
VE-RRE(5)	keine Konvergenz	
VE-RRE(10)	keine Konvergenz	
QN	12,983	0,406
QN-R(10)	7,690	0,263
QN-R(20)	8,257	0,281
QN-NC-R(10)	6,623	0,216
QN-NC-R(20)	7,953	0,265
NKFD	9,493 (67,64)	2,385
NKFD-R(10)	9,477 (45,333)	1,721
NKFD-R(20)	10,650 (36,473)	1,507
NKLP	12,917 (88,533)	0,841
NKLP-R(10)	13,023 (47,375)	0,603
NKLP-R(20)	13,010 (34,220)	0,536

Tabelle 6.13.: Algorithmuseinfluss FSI3 - $\Delta t = 10,0e^{-04} s$

Aitken's Methode (**FPA**) als Standardansatz benötigt 20,083 Iterationen für $\Delta t = 2,5e^{-4} s$ und 32,177 Iterationen für $\Delta t = 10,0e^{-4} s$. Da der Fall FSI3 stärker gekoppelt ist als FSI2, ist hier auch der Einfluss der Zeitschrittweite deutlich größer (+60,2 % Iterationen).

Die Vektorextrapolationsmethoden (**VE-MPE** und **VE-RRE**) zeigen für $\Delta t = 2,5e^{-4} s$ und eine Verwendung von 5 Basisvektoren eine etwas geringere Effizienz als FPA. Für die Verwendung von 10 Basisvektoren ergibt sich divergentes Verhalten nach wenigen Zeitschritten. Für $\Delta t = 10,0e^{-4} s$ und damit einer weiteren Erhöhung des nichtlinearen Charakters konvergiert keine der VE-Methoden.

Die Quasi-Newton-Methode in ihrer Basisversion (**QN**) benötigt 10,253 für $\Delta t = 2,5e^{-4} s$ und 12,983 für $\Delta t = 10,0e^{-4} s$, damit ergibt sich im Vergleich zu FPA eine Verringerung der Kopplungsiterationen um 48,9 % bzw. 40,3 % und Rechenzeit um 46,0 % bzw. 49,4 %. Der Zeitschrittweiteinfluss ist bei QN damit deutlich geringer als bei FPA (+26,6 % Iterationen).

Die Wiederverwendung von Informationen aus den Kopplungsiterationen der letzten Zeitschritte (**QN-R**) führt auch hier zu einer deutlichen Verringerung der Kopplungsiterationen und damit zu einer Steigerung der Effizienz. Die Wiederverwendung von 20 Leveln führt im Vergleich mit QN zu 6,460 Iterationen für $\Delta t = 2,5e^{-4} s$ und einer Reduktion um zusätzliche 37,9 % der Rechenzeit und 8,257 Iterationen für $\Delta t = 10,0e^{-4} s$ und einer Reduktion um zusätzliche 30,8 % der Rechenzeit. Der Einfluss der Zeitschrittweite bleibt gering.

Der Verzicht auf den konstanten Relaxationsschritt (**QN-NC-R**) führt zu einer weiteren deutlichen Effizienzsteigerung mit 5,067 Iterationen für $\Delta t = 2,5e^{-4} s$ und 7,953

Iterationen für $\Delta t = 10,0e^{-4}$ s. Im Fall FSI3 gibt es durch den Verzicht keine Konvergenzprobleme, da durch die durchschnittlich höhere Anzahl an Kopplungsiterationen genug Informationen vom aktuellen Zeitschritt vorliegen.

Das Newton-Krylov-Verfahren mit Finiten Differenzen (**NKFD**) benötigt 5,253 Iterationen für $\Delta t = 2,5e^{-4}$ s und 9,493 Iterationen für $t = 10,0e^{-4}$ s. Betrachtet man nur die Kopplungsiterationen ist NKFD deutlich effizienter als QN und FPA. Die aufwendige Krylov-Raumevaluation jedoch zeigt sich in der Rechenzeit. Zwar kann durch Wiederverwendung von Krylovvektoren die Anzahl an GMRES-Iterationen deutlich verringert werden (um 43,8 % für $\Delta t = 2,5e^{-4}$ s und 46,1 % für $\Delta t = 10,0e^{-4}$ s), die Rechenzeiten sind aber weiterhin deutlich höher als bei allen bisherigen Methoden. Der Einfluss der Zeitschrittweite ist bei NKFD deutlich höher als bei QN (+80,7 % versus 26,6 %).

Die Verwendung eines linearisierten Modells zur Krylov-Raumevaluation (**NKLP**) reduziert diese Kosten signifikant, da nur zwei lineare Modelle pro GMRES-Iteration ausgewertet werden müssen. Die Linearisierungsqualität ist im Fall FSI3 aber geringer als im Fall FSI2, da das Problem deutlich stärker nichtlinear ist und damit im Vergleich zu FD deutlich mehr Iterationen benötigt werden. Für $\Delta t = 2,5e^{-4}$ s steigen die Iterationen um 127,1 %, für $\Delta t = 10,0e^{-4}$ s um 36,1 %. Die Wiederverwendung von Krylov-Rauminformationen senkt die Anzahl an benötigten GMRES-Iterationen deutlich. Betrachtet man die Rechenzeiten, verringern sich diese im Vergleich zu NKFD deutlich. Obwohl die Evaluierungsqualität von LP im Vergleich mit FD gering ist, sinken die Rechenzeiten so stark, dass die NKFD Methoden sogar effizienter als FPA sind. Der Einfluss der Zeitschrittweite ist sehr gering. Im Fall FSI3 ist das linearisierte Problem also ebenfalls in der Lage, den Krylovraum adäquat abzubilden.

6.1.5.3. Schlussfolgerung

Bei den hier gezeigten Beispielen zeigt sich die enorme Ergebnisbandbreite, welche sich aus einem Vergleich diverser Kopplungsalgorithmen ergibt. Dies betrifft sowohl die Effizienz als auch die Stabilität der Methoden. Zusammenfassend ergeben sich für die einzelnen Methoden die folgenden Schlussfolgerungen:

- Im Vergleich zur Standardmethode Aitken können deutliche Effizienzgewinne erzielt werden.
- Die Vektorextrapolationsmethode ist weniger effizient als die Aitken-Methode und wird bei stark nichtlinearen Problemen instabil.
- Die Quasi-Newton-Methode ist am effizientesten und auch bei großen Zeitschrittweiten stabil. Der Verzicht auf den konstanten Relaxationsschritt kann zu einer weiteren Effizienzsteigerung führen, birgt aber die Gefahr der Instabilität bei mittel bis schwach gekoppelten Problemen.
- Bei den Newton-Krylov-Verfahren ist der maßgebende Parameter die Methode zur Krylov-Raumevaluierung. Diesem Aspekt kommt in der entsprechenden Literatur generell wenig Bedeutung zu.

- Mit der Evaluierung über ein reduziertes Ersatzproblem konnte in keinem der Beispiele eine durchgehende Konvergenz erzielt werden.
- Bei einer Krylov-Raumevaluierung über Finite-Differenzen sind zwar wenige Kopplungsiterationen notwendig, die Methode ist aber numerisch extrem aufwendig. Diese Beobachtung zeigt sich auch in [Kol10]. Die Evaluierung durch ein linearisiertes Problem ist im Vergleich deutlich effizienter, aber stark problemabhängig.

Die allgemeinen Schlussfolgerungen lauten:

- Die Wiederverwendung bereits bekannter Informationen kann zu einer deutlichen Effizienzsteigerung führen.
- Mit Quasi-Newton- und Newton-Krylov-Methoden lassen sich lineare Konvergenzraten erzielen, die Quasi-Newton-Methode mit Wiederverwendung kann zu superlinearen Konvergenzraten führen.
- Nur die Vektorextrapolationsmethoden zeigen bei großen Zeitschrittweiten divergentes Verhalten.
- Das QN-R Verfahren ist in den hier vorgestellten Berechnungen das effizienteste und erlaubt die Verwendung von Black-Box-Feldlösern. Damit kombiniert diese Methode hohe Effizienz, Stabilität und geringen Implementierungsaufwand.
- Der Vergleich der Rechenzeiten der einzelnen Methode zeigt, dass der numerische Aufwand des Kopplungsalgorithmus zur Bestimmung der entsprechenden Interfacevariablen (z.B. Notwendigkeit zur Lösung eines Minimierungsproblems in QN oder GMRES-Löseriterationen in NKFD) vernachlässigbar ist. Dies ist auch zu erwarten, da die Anzahl an Interfacefreiheitsgraden um mehrere Größenordnungen geringer ist als im Fluidfeld. Daraus folgt als wichtigstes Ziel eines Kopplungsalgorithmus für den Fall der Fluid-Struktur-Wechselwirkung mit großem Fluidfeld eine möglichst geringe Anzahl an Feldevaluationen, unabhängig vom Aufwand des Kopplungsalgorithmus selbst.

6.1.6. Kombination der Methoden

In den bisherigen Abschnitten werden erzielbare Effizienzsteigerungen in den drei Bereichen Feldlöseransatz, Interfaceprädiktor und Kopplungsalgorithmus getrennt voneinander betrachtet. In diesem Abschnitt werden Kombinationen dieser Methoden gegenübergestellt. Damit wird untersucht, ob sich die Steigerungen summieren oder in wie weit sich die einzelnen Methoden durch die starke Nichtlinearität des gekoppelten Problems gegenseitig negativ beeinflussen. Auch ist damit eine Aussage über die gesamt mögliche Effizienzsteigerung möglich.

Für den Vergleich werden zwei extreme Kombinationen gewählt. Wie bereits in 6.1.1.2 erwähnt, besteht die Standardkombination aus konstanten Feldlösern, einem Polynom-Prädiktor erster Ordnung und einer Fixpunktiteration mit adaptiver Aitken-Relaxation.

Die möglichst effiziente Kombination wird basierend auf den Ergebnissen der vorherigen Kapitel gewählt und besteht aus angepassten Feldlösern, einem Newmark-Prädiktor zweiter Ordnung und einem Quasi-Newton-Verfahren mit Wiederverwendung von 20 Leveln.

Wie in den Abschnitten zuvor werden die Fälle FSI2 und FSI3 mit jeweils zwei unterschiedlichen Zeitschrittweiten untersucht. Für den Vergleich werden die relativierten Berechnungszeiten angegeben, da diese eine absolute Aussage über Effizienzsteigerungen ermöglichen. Die Ergebnisse sind in Tabelle 6.14 dargestellt.

FSI2	$\Delta t = 5,0e^{-4} s$	$\Delta t = 20,0e^{-4} s$
Standardkombination	1,000	1,000
Effiziente Kombination	0,235	0,296
Vergleich [%]	-76,5	-70,4
FSI3	$\Delta t = 2,5e^{-4} s$	$\Delta t = 10,0e^{-4} s$
Standardkombination	1,000	1,000
Effiziente Kombination	0,197	0,259
Vergleich [%]	-80,7	-74,1

Tabelle 6.14.: Effizienzsteigerung durch Kombination aller Methoden - Vergleich der relativen Rechenzeit

Wie aus den Ergebnissen ersichtlich, ergibt sich aus der Kombination möglichst effizienter Komponenten eine Verringerung der Rechenzeit um bis zu 80,7 %. Der Mittelwert der vier untersuchten Fälle liegt bei 75,43 %.

Basierend auf den Einzelergebnissen ergibt sich exemplarisch für den Fall FSI3 mit $\Delta t = 2,5e^{-4} s$ folgende theoretische Verringerung der Rechenzeit: Die Wahl eines angepassten Feldlösers verringert die Rechenzeit um 23,7 % auf relative 0,763, die Wahl eines Newmark-Prädiktors zweiter Ordnung um weitere 33,2 % auf 0,509, und die Wahl eines Quasi-Newton Kopplungsalgorithmus mit Wiederverwendung von 20 Leveln um weitere 66,5 % auf ein relative Rechenzeit von 0,171. Damit ergibt sich eine theoretische Verringerung der Rechenzeit um 82,9 %. Die tatsächliche Verringerung beträgt 80,7 %. Daraus folgt, dass die erzielbaren Effizienzsteigerungen in den einzelnen Komponenten auch in der Kombination erhalten bleiben und summiert werden können.

6.1.7. Schlussfolgerung

In diesem Kapitel werden drei unterschiedliche Bereiche für Effizienzsteigerungen für den Fall stark gekoppelter transienter Probleme vorgestellt. Diese Bereiche sind der Feldlöseransatz, der Interfaceprädiktor und der Kopplungsalgorithmus. Es zeigt sich, dass jeder dieser Bereiche starken Einfluss auf die Effizienz und damit auch die Rechenzeit der Simulation hat.

Im vorherigen Abschnitt werden die einzelnen Verfahren miteinander kombiniert und es zeigt sich, dass die in den einzelnen Bereichen erzielbaren Effizienzsteigerungen auch in der Kombination erhalten bleiben und aufsummiert werden können. Im Vergleich mit

einer als Standard definierten Methodenkombination lässt sich in vier untersuchten Fällen die Rechenzeit um durchschnittlich 75,43 %, also auf circa ein Viertel, verringern.

Zusammenfassend lässt sich also feststellen, dass hohe Effizienz aus der Kombination verschiedener Methoden resultiert. Die reine Reduktion auf den Kopplungsalgorithmus, die in der Literatur größtenteils stattfindet, stellt eine starke Einschränkung dar, mit der nicht das ganze Potential an Möglichkeiten ausgeschöpft wird. Die Effizienzsteigerung in Folge angepasster Feldlöser und Newmark-Prädiktor ist hoch und nur mit geringem Implementierungsaufwand verbunden. Eine äußerst effiziente Kombination für die Simulation transienter und stark gekoppelter Probleme der Fluid-Struktur-Wechselwirkung besteht also aus angepassten Feldlösern, Newmark-Prädiktor zweiter Ordnung und Quasi-Newton-Kopplungsalgorithmus mit Wiederverwendung von 20 Leveln.

6.2. Bogenhalle

Da das vorherige Beispiel von rein numerischer Natur ist, wird in diesem Beispiel eine realistische Problemstellung simuliert. Als Anwendung wird die Wechselwirkung zwischen einer atmosphärischen Grenzschichtströmung und einem leichten Flächentragwerk gewählt. Hierbei werden drei Zielsetzungen verfolgt:

1. Anwendung der im vorherigen Beispiel getesteten Kopplungsmethoden auf ein komplexes dreidimensionales Beispiel.
2. Demonstration der Funktionsfähigkeit der entwickelten Softwareumgebung bei Behandlung komplexer dreidimensionaler Probleme.
3. Hohe parallele Skalierung durch effiziente und uneingeschränkte Gebietszerlegung im partitionierten Ansatz.

Die Idee für dieses Beispiel ergab sich aus einem Architektenentwurf, der am Lehrstuhl für Tragwerksplanung der TU München entstand. Dieser beinhaltet eine Bogenhalle, die besonders leichte und damit effiziente Tragstrukturen verwendet. Eine Visualisierung des Entwurfs ist in Abbildung 6.11 gegeben.

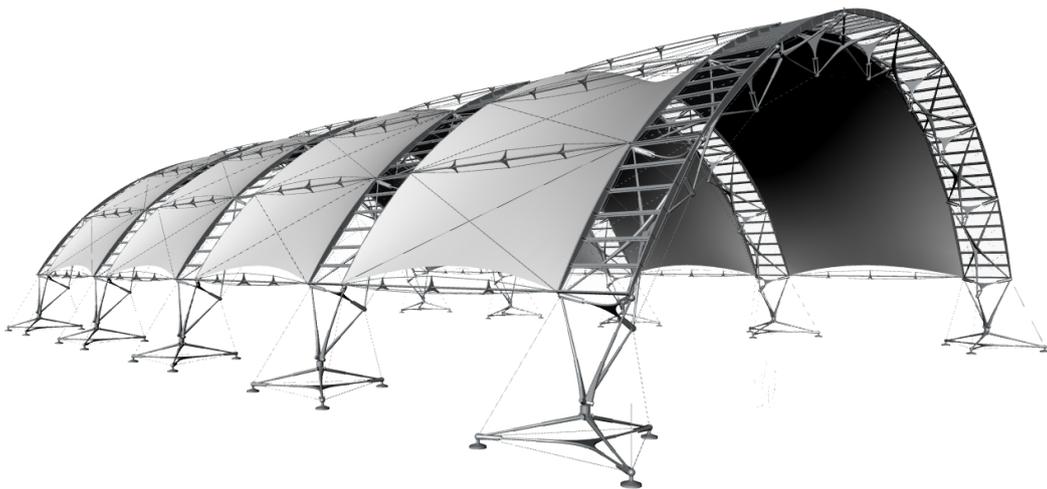


Abbildung 6.11.: Tragwerksentwurf der Bogenhalle

Die Bogenhalle besitzt ein segmentweises Konstruktionsprinzip. Jedes Segment besteht aus einer Dachkonstruktion, die sich über vier Gitterrohrträger auf den Boden stützt. Einzelne Segmente lassen sich nach Belieben miteinander kombinieren, in der Visualisierung werden vier Segmente aneinandergesetzt. Jedes Segment besteht aus zwei steifen Bögen und zwischen den Bögen ist eine Membran gespannt. Die Konstruktion ist nach allen vier Seiten offen.

Diese Kombination aus Offenheit, weit überspannter Fläche und geringer Biegesteifigkeit des Membrandaches führt zu einer hohen Anfälligkeit des Tragwerks gegenüber

Windlasten. Damit ist Windeinfluss der maßgebende Parameter für die Bemessung des Systems. Da die Membran für diesen Fall große Verformungen zeigt und extrem leicht ist, liegt eine starke Wechselwirkung zwischen Fluid und Struktur vor, die für adäquate Ergebnisse abgebildet werden muss. Um das Strukturverhalten unter Windeinfluss numerisch abbilden zu können, wird deshalb eine gekoppelte Simulation durchgeführt.

Dieses Kapitel ist wie folgt aufgebaut: In Abschnitt 6.2.1 werden Geometrie, Physik und die numerischen Modelle des Problems dargestellt. Hierbei wird auf die Aspekte numerischer Windkanal und Fluidnetzerstellung besonders eingegangen. In Abschnitt 6.2.4 werden die Simulationsergebnisse dargestellt. Dies beinhaltet zum einen die parallelen Skalierungstudien und zum anderen die Auswertung des Strukturverhaltens.

6.2.1. Problemdefinition

Die strukturelle und geometrische Komplexität des Tragwerkentwurfs wird für die numerische Simulation mit der Absicht verringert, um an einem vereinfachten Modell die wesentlichen auftretenden Effekte abbilden zu können. Relevant sind unter Windeinfluss alle großen Flächen des Tragwerks. Dazu zählen die Membran und die geschlossene Oberfläche der Bögen. Die Gitterrohrträger und die Aussteifungen haben durch ihre kleine Oberfläche nur einen sehr geringen Einfluss auf die Windströmung und damit auch Lastenaufnahme infolge des Winddruckes. Innerhalb des vereinfachten Modells werden diese vernachlässigt. Des Weiteren wird angenommen, dass die Gitterrohrträger eine sehr hohe Steifigkeit besitzen. Die Aussteifungen werden durch Abspannungen der Bögen auf dem Boden ersetzt. Es werden zwei Segmente untersucht.

Die Problemdefinition ist zweigeteilt: Im ersten Abschnitt werden die realen Eigenschaften Geometrie und Physik, im zweiten Abschnitt die numerische Umsetzung vorgestellt.

6.2.1.1. Geometrie und Physik

Struktur: Eine Skizze mit Geometrie und Abmessungen des vereinfachten Modells ist in Abbildung 6.12 gegeben.

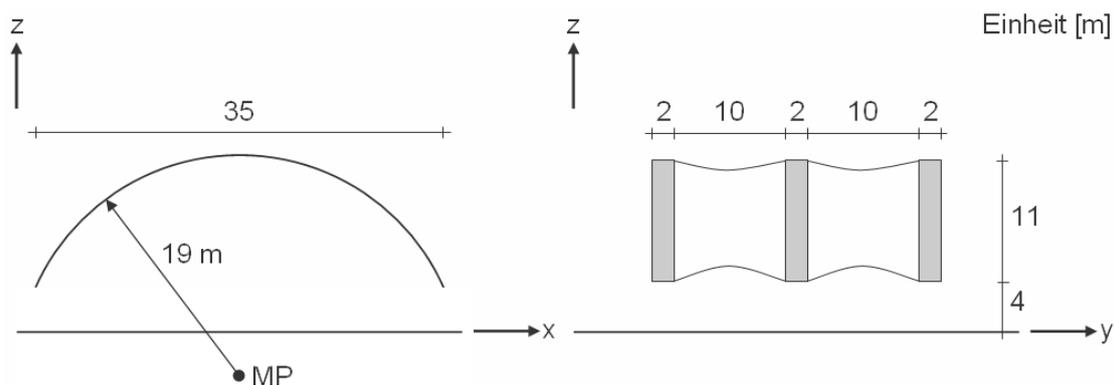


Abbildung 6.12.: Geometrie und Abmessungen des vereinfachten Modells

Die Abmessungen der Halle betragen $b \times t \times h = 35 \times 26 \times 11 \text{ m}^3$. Die Unterkante der Dachkonstruktion befindet sich in 4 m Höhe, die Oberkante in 15 m Höhe. Die Halle überdeckt eine Fläche von 910 m^2 . Die beiden maßgeblichen Tragelemente sind die steifen Bögen in Form eines Zylindersegments und die zwischen den Bögen gespannte und doppelt gekrümmte Membran, welche den flächenmäßig größten Anteil an der Dachkonstruktion hat. Die steifen Bögen werden als massive Schalen mit Rechteckquerschnitt angenommen, die Membran als isotrop vorgespannte Fläche ohne Biegesteifigkeit. Die nicht über Bögen abgedeckten Ränder der Membran sind über vorgespannte Randseile ausgebildet. Die Enden der beiden äußeren Bögen werden als am Boden fixiert angenommen. Eine Auflistung aller Materialparameter ist in Tabelle 6.15 gegeben.

Parameter	E [N/m ²]	ν [–]	ρ [kg/m ³]	t [m]	σ^{Pre} [N/m ²]
Membran	$9 \cdot 10^7$	0,3	1.700	0,001	$2 \cdot 10^6$
Randseil	$1,9 \cdot 10^{11}$	0,3	7.800	0,01	$2 \cdot 10^8$
Bögen	$2,1 \cdot 10^{11}$	0,3	7.800	0,5	–

Tabelle 6.15.: Physikalische Parameter der Bogenhalle

Fluid: Um bodennahen Windeinfluss auf eine Struktur abbilden zu können, muss das Strömungsfeld die Eigenschaften einer atmosphärischen Grenzschichtströmung aufweisen. Für die Charakterisierung einer atmosphärischen Grenzschichtströmung an einem bestimmten Ort sind mehrere Parameter wichtig:

- Die Umgebungsbeschaffenheit hat Einfluss auf die Bodenrauigkeit. Daraus folgt der Parameter der Rauigkeitslänge z_0 . Dieser gibt an, in welcher Höhe über der Geländeoberkante die Windgeschwindigkeit theoretisch gleich Null ist.
- In Abhängigkeit der meteorologischen Gegebenheiten wird z.B. Deutschland in vier verschiedene Windzonen unterteilt, in denen unterschiedliche Geschwindigkeitsverteilungen auftreten. Daraus folgen die Parameter Referenzwindgeschwindigkeit \mathbf{u}_{ref} und Referenzhöhe z_{ref} . Diese geben die Windgeschwindigkeit in einer bestimmten Höhe an, welche bei einem Windereignis mit einer gewissen statistischen Wahrscheinlichkeit auftritt.
- Wird eine zeitliche Mittelung durchgeführt, kann ein Profil für die mittlere Windgeschwindigkeit $\bar{\mathbf{u}}$ in Abhängigkeit der Höhe über Gelände z angegeben werden („logarithmisches Gesetz“):

$$\bar{\mathbf{u}}(z) = \frac{\mathbf{u}_\tau}{\kappa} \ln \left(\frac{z}{z_0} \right), \quad (6.3)$$

mit κ als Karmánn-Konstante und \mathbf{u}_τ als Schubspannungsgeschwindigkeit. Eine übliche Annahme für die Karmánn-Konstante lautet $\kappa \approx 0,4$. Die Schubspannungsgeschwindigkeit \mathbf{u}_τ ist abhängig von der Bodenrauigkeit und lässt sich nach [RH93, RH07, Har07a] bestimmen über:

$$\mathbf{u}_\tau = \frac{\kappa \mathbf{u}_{ref}}{\ln \frac{z_{ref}}{z_0}}. \quad (6.4)$$

In diesem Beispiel wird eine Referenzwindgeschwindigkeit von $\mathbf{u}_{ref} = 25 \text{ m/s}$ in einer Höhe von $z_{ref} = 10 \text{ m}$ angenommen. In Windzone II wird diese Windgeschwindigkeit mit einer Wahrscheinlichkeit von 0,02 pro Jahr überschritten (50-Jahres-Wahrscheinlichkeit, [DIN05]). Die Umgebung wird mittels der Rauigkeitsklasse 2,5 beschrieben. Dies entspricht einer freien Ackerfläche mit einer durchschnittlichen Ackerschollenhöhe von 0,4 m, vereinzelter Bebauung in mäßiger Entfernung und führt zu einer Rauiglänge von $z_0 = 0,2 \text{ m}$ ([SS96]). Eine Auflistung der Strömungsparameter ist in Tabelle 6.16 gegeben.

Zur Vereinfachung wird in der numerischen Simulation nur eine einzelne Hauptanströmrichtung untersucht. Da die Steifigkeit der Membrane quer zu ihrer Fläche gering ist, wird eine Anströmung aus der x-Richtung vorgenommen, um möglichst hohe Oberflächendrücke in diese Richtung zu generieren. Bei Annahme der logarithmischen Geschwindigkeitsverteilung nach Gl. 6.3 folgt an der Oberkante der Halle eine Windgeschwindigkeit von $\bar{\mathbf{u}}(z = 15) = 27,59 \text{ m/s}$. Die Breite der Halle quer zur Anströmrichtung beträgt 26 m. Daraus folgt eine Reynoldszahl von $Re = 47,82 \cdot 10^6$.

\mathbf{u}_{ref} [m/s]	z_{ref} [m]	z_0 [m]	ρ [kg/m ³]	ν [m ² /s]	κ [-]	\mathbf{u}_τ [m/s]	Re [-]
25	10	0,2	1,184	$1,5 \cdot 10^{-5}$	0,4	2,556	$47,82 \cdot 10^6$

Tabelle 6.16.: Physikalische Parameter der Windströmung

6.2.1.2. Numerisches Modell

Die Umsetzung in ein numerisches Modell besteht aus den Teilmodellen für Struktur und Fluid und dem Kopplungsansatz zur Abbildung der Wechselwirkung. Diese werden in den folgenden drei Abschnitten behandelt.

Anmerkung: Ein klassischer Ansatz zur Reduzierung des numerischen Aufwands ist die Ausnutzung von Symmetrieeigenschaften des Problems. Die untersuchte Bogenhalle weist eine Symmetrieebene auf, wodurch die Geometrie die Symmetriebedingung erfüllt. Die Anströmung und damit das Strömungsfeld sind aber asymmetrisch (worauf im Weiteren noch eingegangen wird), woraus sich asymmetrische Verformungen ergeben. Die Ausnutzung einer Symmetrieeigenschaft ist hier also nicht möglich.

Struktur: Für die Diskretisierung im Raum wird die geometrisch nichtlineare Finite-Elemente-Methode benutzt. Die steifen Bögen werden durch Schalenelemente mit einer 7-Parameter-Formulierung modelliert ([Bis99]). Pro Bogen werden 2×40 Elemente verwendet, an jedem Knoten existieren 6 Freiheitsgrade. Als Randbedingung werden die Verschiebungen an den Enden der beiden äußeren Bögen fixiert. Die Membrane werden

über verschiebungsformulierte dreidimensionale Scheibenelemente und isotrope Vorspannung modelliert. Pro Segment werden 15×40 Elemente verwendet, an jedem Knoten existieren 3 Freiheitsgrade. Die Randseile werden über vorgespannte Fachwerkstäbe modelliert. Pro Segment werden 2×15 Elemente verwendet, an jedem Knoten existieren 3 Freiheitsgrade. Damit besteht das gesamte Strukturmodell aus 1500 Elementen und 3144 Freiheitsgraden. Für die Diskretisierung in der Zeit wird das Generalized- α Zeitintegrationsverfahren verwendet und über die Wahl des Spektralradius zu $\rho_\infty = 0,9$ eine leichte numerische Dämpfung hochfrequenter Moden eingeführt.

Ein Teil der Dachstruktur wird mittels einer Membran ausgeführt. Die Membran ist eine vorgespannte und doppelt gekrümmte Fläche, deren Position und Form einen Gleichgewichtszustand darstellt. Dieser Gleichgewichtszustand ist das Resultat von Randbedingungen und Vorspannung. Für das numerische Modell wird der Gleichgewichtszustand und die zugehörige Form mittels eines Formfindungsprozesses bestimmt. Dazu wird die iterative URS-Strategie („Updated Reference Strategy“, [BR99, BFLW10]) verwendet. Als Ausgangspunkt für die Formfindung wird ein Strukturmodell benötigt, welches die gewünschten Randbedingungen und Vorspannungen aufweist und über eine Form verfügt, die bereits nahe der endgültigen Form liegt. Für die Bogenhalle wird die Ausgangsform eines Zylindersegmentes gewählt. Eine graphische Darstellung der Ausgangsform und das Ergebnis des Formfindungsprozesses ist in Abbildung 6.13 gegeben.

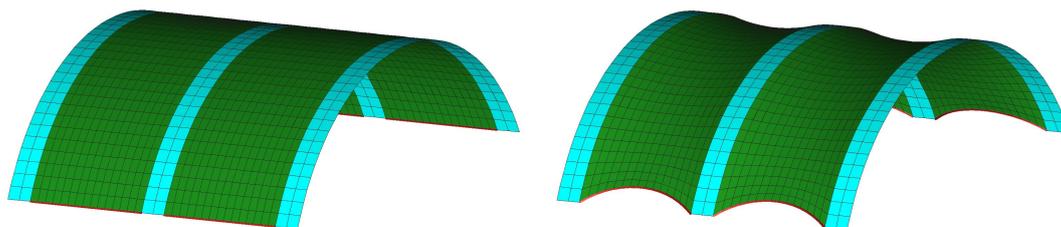


Abbildung 6.13.: Strukturmodell - Formfindungsprozess

Fluid: Zur Abbildung einer atmosphärischen Grenzschicht wird eine numerische Windkanalsimulation durchgeführt ([FHJ⁺04, Mic10, Kup09, Kna07, BSC07]). Dazu wird ein quaderförmiges Simulationsgebiet gewählt. In diesem wird ein Ausschnitt der atmosphärischen Grenzschicht abgebildet, indem durch entsprechende Randbedingungen die spezifische Charakteristik nachgeahmt wird. Das zu untersuchende Objekt, in diesem Fall die Bogenhalle, wird innerhalb des Simulationsgebietes platziert. Im Folgenden werden die wichtigsten Aspekte des numerischen Fluidmodells besprochen. Dies sind die Simulationsgebietsabmessungen, das Berechnungsnetz, die Turbulenzmodellierung und die Methode zur Erzeugung einer Windströmung.

Der erste wichtige Aspekt betrifft die Wahl der Gebietsabmessungen. Für einen geringen numerischen Aufwand muss das Gebiet möglichst klein sein. Um aber eine von den Rändern unbeeinflusste Gebäudeumströmung zu gewährleisten, müssen die Ränder möglichst weit entfernt sein. In [FHJ⁺04] werden basierend auf der Auswertung diverser Studien Anforderungen an notwendige Gebietsabmessungen in Abhängigkeit der Höhe

des umströmten Objektes gestellt. Mit diesen Angaben als grobe Richtlinie werden die in Tabelle 6.17 angegebenen Gebietsabmessungen gewählt. *BR* bezeichnet hierbei die „Blocking Ratio“, also den Versperrungsgrad des Gesamtquerschnitts infolge des Gebäudes. Das Koordinatensystem bezieht sich auf die in Abb. 6.12 bereits angegebene Definition. Eine isometrische Darstellung des quaderförmigen Simulationsgebietes und der darin enthaltenen Dachstruktur ist in Abb. 6.14 gegeben.

L_x [m]	L_y [m]	L_z [m]	V [m ³]	BR [%]
200	76	70	$1,064 \cdot 10^6$	5,38

Tabelle 6.17.: Abmessungen des numerischen Windkanals

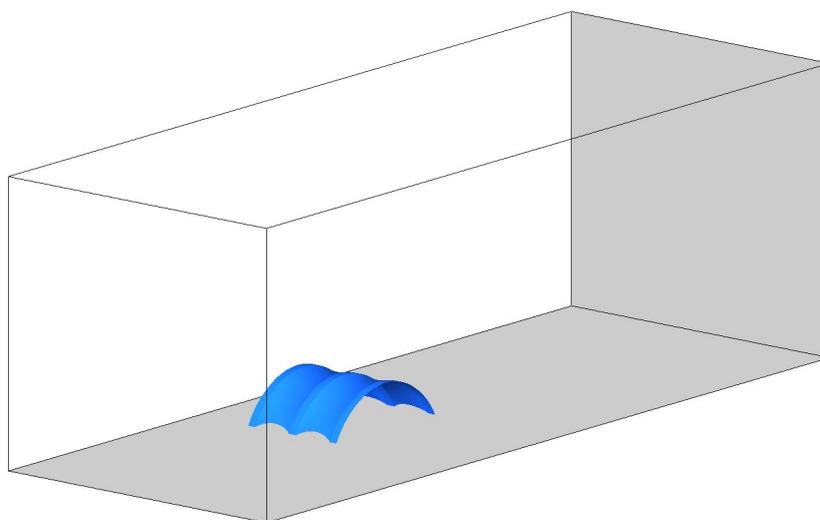


Abbildung 6.14.: Fluidmodell - Simulationsgebiet

Nach Festlegung der Gebietsabmessungen sind die Eingabeparamter für die Erstellung des Berechnungsnetzes bekannt. In diesem Beispiel wird für die Netzerzeugung eine eigens entwickelte Methode verwendet, welche einige der Vorteile strukturierter und unstrukturierter Netze vereint. Strukturierte Netze weisen vor allem im Finite-Volumen-Kontext sehr gute Konvergenzeigenschaften auf, während unstrukturierte Netze die Anpassung an komplexe Oberflächengeometrien erlauben.

Hierzu wird als erstes ein strukturiertes Netz erzeugt, welches an die Ausgangsform des Strukturmodells vor der Formfindung angepasst ist (siehe linkes Bild in Abb. 6.13). Dies ist möglich, da die Ausgangsform ein Zylindersegment ist. Danach wird eine Formfindungsanalyse des Strukturmodells durchgeführt mit dem Resultat der doppelt gekrümmten und nicht analytisch beschreibbaren Membrangeometrie (siehe rechtes Bild in Abb. 6.13). Das zugehörige Verschiebungsfeld wird als Randverschiebung auf das Fluidnetz aufgebracht und dieses strukturierte Netz mittels der Netzverformungsmethoden (siehe Abschnitt 3.5) an die veränderte Randgeometrie angepasst. Durch diese Herangehens-

weise wird eine hohe Elementqualität der Fluidelemente erreicht und es ist eine schnelle Anpassung des Fluidnetzes bei Veränderung von Strukturparametern möglich, da die Netzanpassungsmethode automatisiert ist.

Das gesamte Fluidgebiet wird mit 1.030.200 Elementen diskretisiert. Da pro Element 9 Freiheitsgrade zu lösen sind, ergeben sich insgesamt 9.271.800 Freiheitsgrade. Die Darstellung eines Netzdetails im Bereich der Bogenhalle mittels eines Schnittes in der x - z -Ebene ist in Abb. 6.15 links gegeben, das Fluidnetz an der Oberfläche der Bogenhalle ist in Abb. 6.15 rechts dargestellt (Anmerkung: Im Schnitt werden die Elemente als Dreiecke dargestellt. Dies ist eine Eigenschaft der verwendeten Visualisierungssoftware, im Simulationsmodell sind die Elemente quaderförmig).

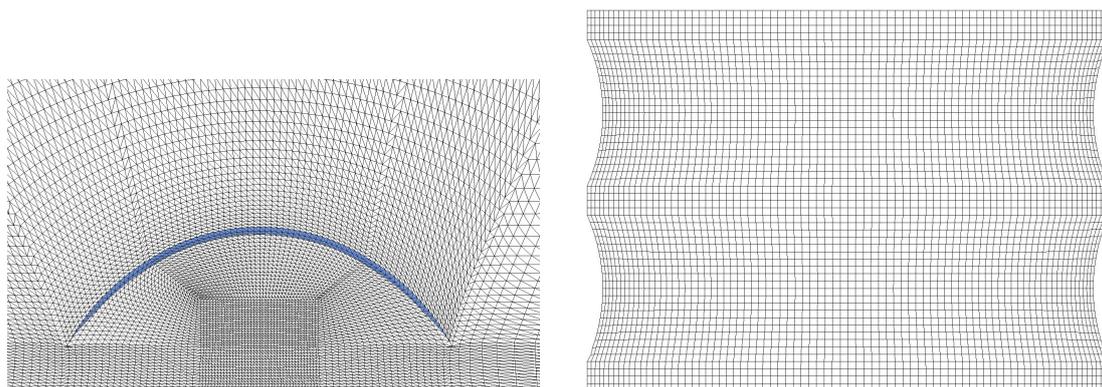


Abbildung 6.15.: Fluidmodell - Netzdetail im Bereich der Bogenhalle (Schnitt in der x - z -Ebene, links) und Oberflächennetz der Bogenhalle (rechts)

Aus der angenommenen Referenzwindgeschwindigkeit folgt eine Reynoldszahl von $Re = 47,82 \cdot 10^6$. Um die turbulenten Eigenschaften des Strömungsfeldes abzubilden, wird die RANS-Methode mit einem $k - \omega - SST$ Turbulenzmodell verwendet ([ME01, MKL03]). Hierzu ist am Einlass die turbulente kinetische Energie k und die Dissipationsfrequenz ω zu spezifizieren. Diese werden als konstant über die Höhe angenommen mit dem in der Referenzhöhe bestimmten Wert. Damit folgt nach [Har07a]:

$$k = \frac{\mathbf{u}_\tau^2}{\sqrt{C_\mu}} = 21,78 \quad \text{und} \quad (6.5)$$

$$\omega = \frac{\kappa(z_{ref} + z_0)}{\sqrt{C_\mu} \mathbf{u}_\tau} = 5,32, \quad (6.6)$$

mit der üblichen Annahme von $C_\mu = 0,09$. Um die hohen Gradienten in Wandnähe ohne eine entsprechende Netzauflösung abbilden zu können wird in der ersten Reihe an Elementen, die an eine Wand (Boden oder Bogenhalle) angrenzen, ein zonales Wandmodell verwendet.

Um eine Windströmung im Fluidmodell abbilden zu können, wird am Einlassrand eine entsprechende Geschwindigkeitsverteilung vorgeschrieben. Dafür existieren zwei Möglichkeiten. Einerseits die Vorgabe eines zeitlich konstanten mittleren Einströmprofils nach

Gl. 6.3. Damit wird eine stationäre Anströmung erreicht, welche aber die zeitlichen und räumlichen Schwankungen einer realen Windströmung vernachlässigt und damit auch z.B. Spitzenlasten nicht abbilden kann. Andererseits kann das mittlere Einströmprofil mit einem räumlich und zeitlich schwankenden Verlauf überlagert werden. Dieser Ansatz wird z.B. in [Man98, Mic10] angewandt. Die statistischen Kenngrößen des Einströmprofils entsprechen dabei den in realen Windströmungen auftretenden Verhältnissen. Damit entsteht eine instationäre Anströmung, durch welche auch z.B. Böen im Windfeld abgebildet werden.

In diesem Beispiel wird eine instationäre Anströmung gewählt, wie sie in [Mic10] hergeleitet und verwendet wird. Dazu werden mit einer Hilfssimulation im Vorfeld Zeitreihen der Geschwindigkeitsverteilung am Einströmrand generiert und als Randbedingung auf das Fluidmodell aufgebracht. Die so generierten Zeitreihen weisen Schwankungen in z-Richtung und y-Richtung auf. Um einen Eindruck über die so generierte Einströmbedingung zu bekommen, wird in Abb. 6.16 der zeitliche Verlauf der Geschwindigkeitsamplitude an drei Punkten am Einlass angegeben. Die Punkte befinden sich in der Achse des mittleren Bogens in den drei Höhen $z = 5, 15, \text{ und } 60 \text{ m}$. Zum Vergleich ist für jeden der Punkte die Geschwindigkeit aus dem mittleren Profil nach Gl. 6.3 angegeben. Der Gesamtzeitraum beträgt 60 s . Die instationären Schwankungen um den Mittelwert sind in allen drei Höhen klar ersichtlich.

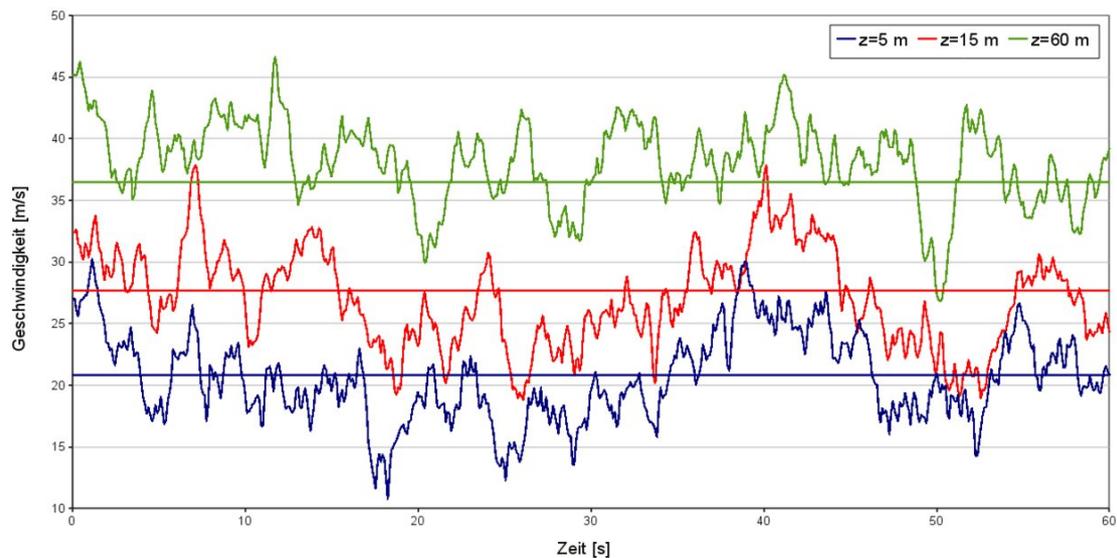


Abbildung 6.16.: Windeinlass - Vergleich von mittleren und generierten Geschwindigkeiten in drei Höhen

Anmerkung: Beim Start einer Simulation ist es notwendig, für alle Fluidvariablen eine Initiallösung im Feld anzugeben. Typischerweise werden Geschwindigkeit, Druck, etc. zu Null vorgegeben. Dies führt i.d.R. zu unphysikalischen Druckschwankungen in den ersten Zeitschritten. Da daraus in einer gekoppelten Berechnung Konvergenzprobleme

resultieren, werden besser geeignete und physikalisch stimmige Initiallösungen verwendet. Dazu wird das zum Zeitpunkt $t = 0$ s generierte Windprofil verwendet und eine reine CFD-Berechnung mit diesem Profil über einen Zeitraum von $t = 5$ s durchgeführt. Die am Ende des Zeitraumes vorliegenden Lösungen werden als Initiallösung für die gekoppelte Berechnung verwendet.

Kopplung: Es wird eine implizit gekoppelte Simulation durchgeführt. Basierend auf den Ergebnissen des ersten numerischen Beispiels in Abschnitt 6.1 wird eine Kombination aus angepassten Feldlösern, Newmark-Prädiktor zweiter Ordnung und Quasi-Newton Kopplungsalgorithmus mit Wiederverwendung von 20 Leveln verwendet. Damit ist eine stabile und hoch effiziente Simulation gewährleistet.

In der Simulation wird ein Gesamtzeitraum von 50 s untersucht. Die Zeitschrittweite beträgt $\Delta t = 0,01$ s. Dafür ergibt sich im Fluid eine maximale Courant-Zahl von $\max Co \approx 1,6$.

Anmerkung: Das Auftreten einer starken oder schwachen Kopplung wird üblicherweise aus den Dichteverhältnissen von Struktur und Fluid abgeleitet. Im Fall der Umströmung einer sehr dünnen Fläche ist dieses Kriterium aber nur bedingt anwendbar. In diesem Beispiel ist die Membrandichte um den Faktor 1.436 höher als die Fluidichte, wodurch eigentlich eine schwache Kopplung vorliegt. Da die Membran aber nur eine Dicke von 1 mm aufweist, ist die bewegte Strukturmasse äußerst gering. Die auftretende starke Kopplung zeigt sich also bei einer Bewertung der bewegten Massen.

6.2.2. Vergleich von Kopplungsmethoden

Eines der Ergebnisse des ersten Beispiels ist die Empfehlung für eine besonders effiziente Kombination an Kopplungsmethoden. In diesem Abschnitt wird untersucht, in wie weit diese an einem geometrisch sehr einfachen Beispiel erzielte Effizienzsteigerung auch innerhalb einer komplexen und realistischen Problemstellung erzielt werden kann.

Dazu werden Berechnungen mit zwei Methodenkombinationen durchgeführt, wie sie bereits in Abschnitt 6.1.6 verwendet wurden. Die erste Methodenkombination ist die Standardkombination. Sie besteht aus konstanten Feldlösern, einem Polynom-Prädiktor erster Ordnung und einer Fixpunktiteration mit adaptiver Aitken-Relaxation. Die zweite Methodenkombination ist die möglichst effiziente Kombination. Sie besteht aus angepassten Feldlösern, einem Newmark-Prädiktor zweiter Ordnung und einem Quasi-Newton-Verfahren mit Wiederverwendung von 20 Leveln.

In den zwei Berechnungen werden für einen Vergleich nur die ersten 100 Zeitschritte der gekoppelten Gesamtsimulation durchgeführt. Für die Auswertung wird der Durchschnittswert des Interfaceresiduums nach Anwendung des Prädiktors, die durchschnittliche Anzahl an Kopplungsiterationen, und die relative Rechenzeit angegeben. Die Berechnung wird auf einem Hochleistungsrechner mit 18 Fluidprozessen, einem Kopplungs- und einem Strukturprozess durchgeführt. Die Ergebnisse sind in Tabelle 6.18 angegeben.

Das durchschnittliche Startresiduum verringert sich durch Verwendung des Newmark-Prädiktors zweiter Ordnung um 86,2 %, die Verwendung des Quasi-Newton-Verfahrens

	relatives Startresiduum	Kopplungs- iterationen	relative Rechenzeit
Standardkombination	0,251	14,051	1,000
Effiziente Kombination	0,035	6,575	0,314

Tabelle 6.18.: Vergleich der Effizienz von Standard- und effizienter Kombination

mit Wiederverwendung von 20 Leveln reduziert die Anzahl an Kopplungsiterationen um 53,2 %. Die gesamte Rechenzeit verringert sich dadurch um 68,6 %. Wie aus den Ergebnissen klar ersichtlich, ist auch in diesem komplexem Beispiel die Effizienzsteigerung durch Verwendung der effizienten Methodenkombination sehr hoch. Damit lässt sich eine gewisse allgemeine Gültigkeit der möglichen Effizienzsteigerung durch Verwendung der effizienten Methodenkombination feststellen.

6.2.3. Parallele Skalierung

Gerade bei der Simulation großskaliger Probleme ist ein paralleler Lösungsansatz notwendig, um die Gesamtrechenzeit in einem akzeptablen Rahmen zu halten. Da eine gekoppelte Simulation deutlich aufwendiger als eine vergleichbare Einzelfeldsimulation ist, folgt daraus die besonders hohe Bedeutung einer effizienten Parallelisierungsstrategie. Bei Einzelfeldproblemen wird üblicherweise eine geometrische Parallelsierung mit einer Gebietszerlegung verwendet. Damit ergibt sich ab einer gewissen Gebietsanzahl die Aufteilung des Interfacenetzes auf mehrere Gebiete. Mit der in dieser Arbeit entwickelten Softwareumgebung kann eine unbegrenzte Aufteilung des Interfacenetzes behandelt werden (siehe Abschnitt 5.3.1). Das Ziel ist es hierbei, die Effektivität der in den Einzelfeldlösern verwendeten Strategie nicht einzuschränken.

Um die Effektivität dieser Parallelisierungsstrategie nachzuweisen, werden in diesem Abschnitt die Ergebnisse einer parallelen Skalierungsstudie vorgestellt. Dazu werden in einem ersten Schritt parallele Einzelfeldsimulationen des Fluidfeldes durchgeführt. Daraus ergeben sich die möglichen Skalierungsfaktoren des Fluidlösers bei einer Einzelfeldsimulation. Im nächsten Schritt werden mit den identischen Gebietszerlegungen gekoppelte Simulationen durchgeführt. Im Vergleich der sich hierbei ergebenden Skalierungsfaktoren mit den vorherigen spiegelt sich der Einfluss und eine mögliche Einschränkung durch den partitionierten Ansatz wider.

Da das Strukturfeld eine um den Faktor 2.300 geringere Anzahl an Freiheitsgraden aufweist, ist der zeitliche Anteil an der gesamten Berechnungszeit sehr gering. Deshalb wird das Strukturfeld seriell gelöst, da eine parallele Lösung nur eine unbedeutende Geschwindigkeitssteigerung zur Folge hätte.

Im numerischen Beispiel werden die ersten 100 Zeitschritte der Simulation durchgeführt. Es wird eine Zerlegung des Fluidfeldes auf $n = 2, 4, 8, 12$ und 16 Gebiete untersucht. Bei der Einzelfeldlösung des Fluidfeldes wird ein Lösungsansatz verwendet, der dem gekoppelten Fluidlöser ohne Netzverformung und mit einer Kopplungsiteration entspricht. Damit ist eine Vergleichbarkeit gewährleistet. Bei der gekoppelten Berechnung werden zusätzlich zwei Prozesse für Kopplung und Strukturfeldlösung

ausgeführt. Die Berechnung der parallelen Beschleunigung basiert auf der notwendigen Rechenzeit. Da eine gekoppelte Simulation mit nur einem Fluidgebiet durch die verwendete Fluidsoftware nicht möglich ist, wird als Basis die mit zwei Fluidgebieten benötigte Rechenzeit $T_{nprocs=2}$ gewählt. Die Beschleunigung bestimmt sich dann über $S(nprocs) = 0,5 \cdot T_{nprocs=2}/T_{nprocs}$. Für eine ideale lineare Beschleunigung folgt also $S = nprocs$. Die Ergebnisse sind in Tabelle 6.19 dargestellt. Zusätzlich wird die bei der gekoppelten Berechnung auftretende Untermenge an Prozessen angegeben, welche einen Teil des gekoppelten Interfacenetzes besitzt und deshalb an der Kommunikation mit der Kopplungssoftware teilnimmt.

Anzahl Teilfelder	Parallele Beschleunigung S		Anzahl Teilnetze Interface
	Fluid-Einzelfeldsimulation	gekoppelte Simulation	
2	2,00	2,00	2
4	3,41	3,30	4
8	6,72	6,44	8
12	7,88	7,56	10
16	10,10	9,76	10

Tabelle 6.19.: Vergleich der parallelen Beschleunigung von Einzelfeld- und gekoppelter Simulation

Wie aus den Ergebnissen klar ersichtlich wird, ist die parallele Skalierung des CFD-Lösers bis zu einer Aufteilung auf 8 Teilfelder nahezu linear. Bei einer weiteren Aufteilung des Fluidgebietes folgt aber nur noch eine geringe Verkürzung der Rechenzeit. Dies liegt an der zu geringen Anzahl an Elementen im Fluidfeld für eine Aufteilung auf 12 und 16 Teilfelder und stellt eine Eigenschaft des verwendeten Fluidlösers OpenFOAM dar. Wichtiger ist jedoch die Skalierung der gekoppelten Berechnung. Die einzelnen Faktoren sind nur geringfügig kleiner als bei einer Einzelfeldsimulation. Die Unterschiede nehmen bei ansteigender Teilfeldzahl leicht zu, da der Strukturlöser und das Kopplungsprogramm nur seriell arbeiten und deren Rechenzeit damit konstant bleibt. Insgesamt lässt sich aber bei der gekoppelten Berechnung nahezu die gleiche Beschleunigung erzielen wie bei einer Einzelfeldberechnung. Damit ist die Effizienz der entwickelten Parallelisierungsstrategie für die gekoppelte Simulation im Fall einer mäßigen Anzahl an Prozessoren nachgewiesen.

6.2.4. Simulationsergebnisse

In diesem Abschnitt werden die Ergebnisse der Simulation vorgestellt. Für die Charakterisierung des Fluidfeldes wird das Strömungsverhalten im Bereich der Bogenhalle präsentiert, für die Charakterisierung des Strukturverhaltens werden die Oberflächendrucke am Interface und das Verformungsverhalten der Bogenhalle dargestellt. Für die Beschreibung des jeweils untersuchten Bereiches der Bogenhalle wird mit „vorderer Bereich“ der strömungszugewandte Teil der Halle, mit „hinterer Bereich“ der strömungsabgewandte Teil, und mit „oberer Bereich“ der Firstbereich bezeichnet.

Fluid: Das Fluidfeldverhalten wird exemplarisch zum Zeitpunkt $t = 0,8$ s untersucht,

da hier die größten Verformungen der Struktur auftreten. In den Abbildungen 6.17 bis 6.20 sind vier Visualisierungen der Strömungsergebnisse dargestellt.

Abbildung 6.17 zeigt die Geschwindigkeitsamplitude in einem Schnitt normal zur y -Achse in Höhe des mittleren Bogens der Bogenhalle. Deutlich ist der Einfluss des generierten instationären Windfeldes am Einlass zu erkennen. Im Vorlauf der Bogenhalle ist das gemittelte logarithmische Profil mit stark instationären Wirbeln überlagert, wodurch sich die dargestellte Geschwindigkeitsverteilung einstellt.

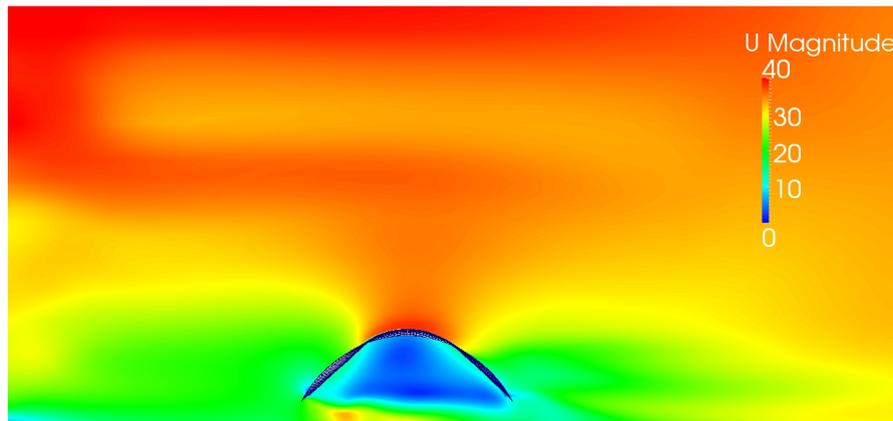


Abbildung 6.17.: Geschwindigkeitsverteilung (Schnittnormal zur y -Achse)

In Abbildung 6.18 sind Stromlinien visualisiert, die auf einer parallel zur y -Achse liegenden Geraden in 10 m Abstand zum vorderen Bereich der Halle erzeugt werden. Hier zeigt sich die Entstehung von zwei im Nachlauf der Halle befindlichen Kopfwirbeln, welche eine gegensinnige Drehrichtung aufweisen und durch eine Ablösung an der Kante der seitlich offenen Bogenhalle entstehen.

In Abbildung 6.19 sind Stromlinien dargestellt, die auf einer parallel zur z -Achse liegenden Geraden in 10 m Abstand zum vorderen Bereich der Halle erzeugt werden. Hier zeigt sich die Erhöhung der Geschwindigkeiten am oberen Punkt der Halle aus der Umströmung des Objektes. Des weiteren sind die starken Verwirbelungen innerhalb der Halle als Folge der seitlichen Offenheit zu erkennen.

Abbildung 6.20 zeigt die Druckverteilung auf der Innenseite (links) und auf der Außenseite (rechts) der Membran. Deutlich ist zu erkennen, dass die Drücke außen um einen Faktor von $\approx 2,5$ höher sind als innen. Maßgebend für die Membranstruktur ist damit der Außendruck. Entlang des Bogens stellt sich im vorderen Bereich auf der strömungszugewandten Seite ein Überdruck ein, im mittleren Bereich Unterdruck, welcher bis zum Ende des Bogens auf der strömungsabgewandten Seite im hinteren Bereich wieder in schwachen Überdruck übergeht.

Struktur: Das Strukturverhalten wird über die Verformungen des Systems charakterisiert. Hierzu wird als erstes das Verformungsverhalten des gesamten Systems zu einem exemplarischen Zeitpunkt betrachtet, als zweites wird die zeitliche Veränderung der Ver-

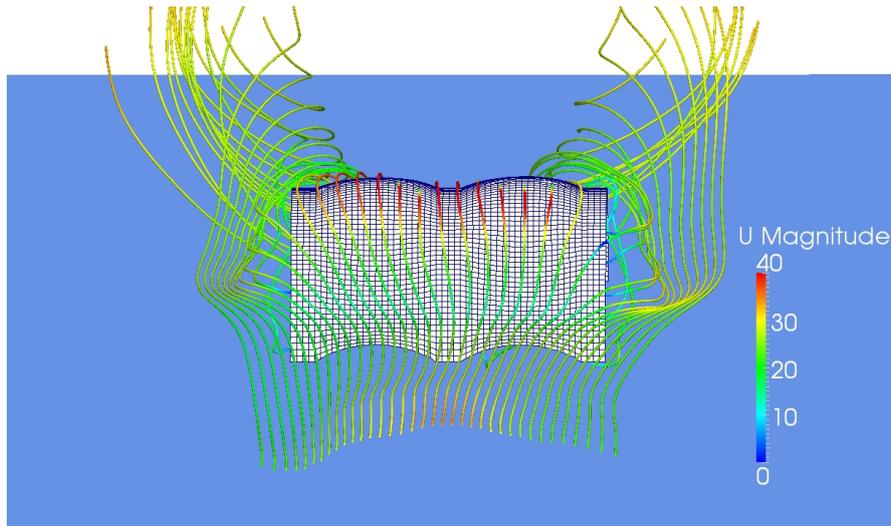


Abbildung 6.18.: Stromlinien (Erzeugende entlang y-Achse)

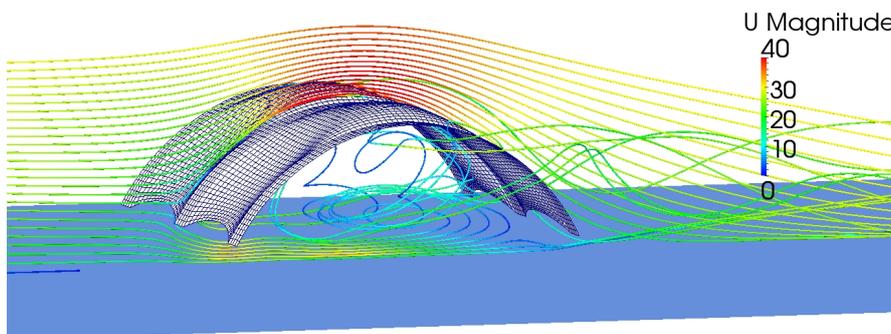


Abbildung 6.19.: Stromlinien (Erzeugende entlang z-Achse)

fomung an einzelnen Punkten des Tragwerks gezeigt.

Für die exemplarische Betrachtung wird, äquivalent zum Fluidfeld, der Zeitpunkt $t = 0,8 \text{ s}$ gewählt. In Abbildung 6.21 sind die Verformungen in x- (links) und z-Richtung (rechts) als Projektion in die x-y-Ebene dargestellt, in Abbildung 6.22 ist eine isogeometrische Ansicht der verformten Struktur mit Visualisierung der Verformungsamplitude dargestellt.

Das Verformungsbild folgt hierbei der in Abbildung 6.20 dargestellten Oberflächen-druckverteilung auf der Außenseite der Membran. Im vorderen Bereich des Halle treten als Folge des Überdrucks Verformungen auf, welche die in der Initialgeometrie bereits vorhandene Krümmung noch verstärken. Die maximale Amplitude beträgt in diesem Bereich ca. $0,8 \text{ m}$. Im oberen Bereich treten starke Sogkräfte auf. Diese führen zu einem Durchschlagen der Membran, die Krümmung kehrt sich also um. Während des Durchschlagens treten in diesem Bereich starke Steifigkeitsverluste auf, da im Durchschlagpunkt

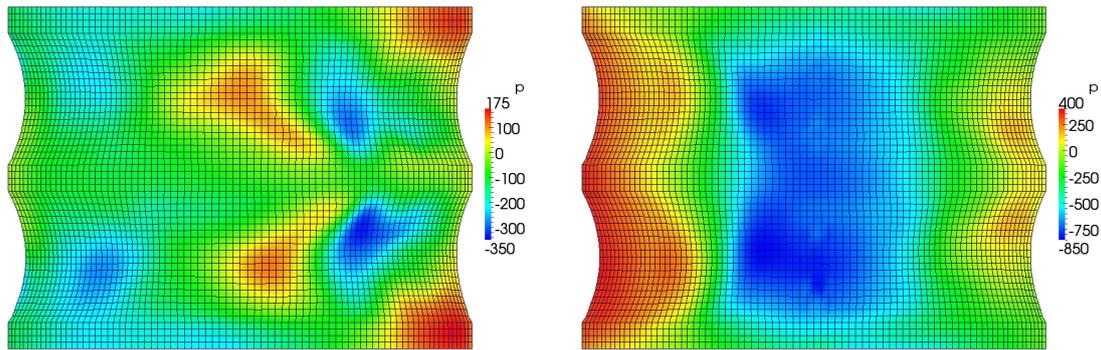


Abbildung 6.20.: Druckverteilung an der Membranoberfläche (links: Innenseite, rechts: Außenseite)

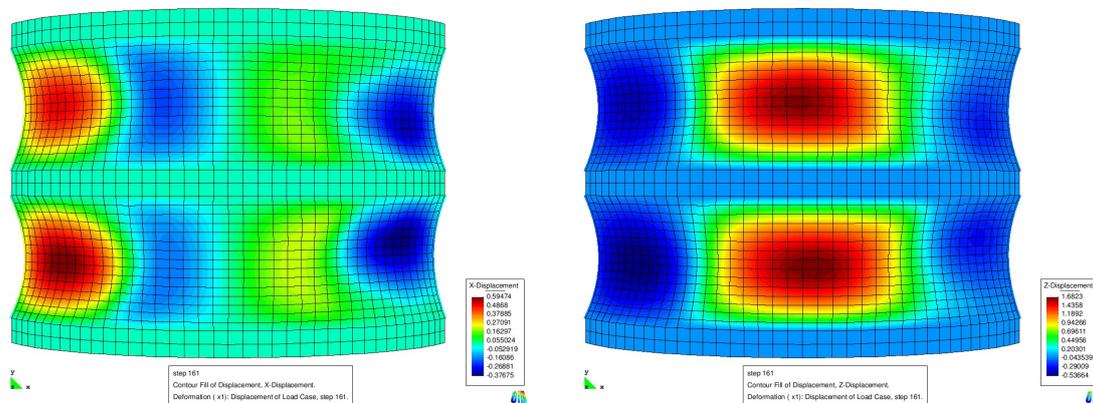


Abbildung 6.21.: Verformungen der Membranoberfläche bei $t = 0,8$ s (links: x-Komponente, rechts: z-Komponente)

die geometrischen Steifigkeiten aus der Vorspannung auf nahezu Null reduziert werden. Die auftretende maximale Amplitude beträgt ca. $1,8$ m. Die maximalen Verformungen im hinteren Bereich betragen ca. $0,4$ m und sind damit deutlich geringer.

Um die transiente Charakteristik des Simulationsbeispiels zu veranschaulichen, wird im Folgenden die zeitliche Veränderung der Verformung in z-Richtung an einzelnen Punkten dargestellt. Hierzu werden die drei Punkte A, B und C gewählt, deren Position in Abbildung 6.23 angegeben ist. Über die Punkte wird exemplarisch das Strukturverhalten im vorderen, oberen und hinteren Bereich abgebildet. Das Zeit-Verschiebungsdiagramm über den untersuchten Zeitraum von 50 s ist in Abbildung 6.24 dargestellt.

Das Zeit-Verschiebungsdiagramm zeigt die stark instationäre Charakteristik des untersuchten Beispiels. Die Bogenhalle befindet sich zum Zeitpunkt $t = 0$ s in der Ruhelage, so dass sich innerhalb der ersten ca. 4 s ein Einschwingvorgang mit höheren Amplituden als im Rest des untersuchten Zeitraums einstellt. Nach dem Einschwingvorgang zeigen alle drei Punkte Schwingungen um einen gewissen Mittelwert. Über eine zeitliche Mittelung ergibt für den Punkt A dessen Höhe zu $-0,389$ m, für den Punkt B zu $0,908$ m, und für

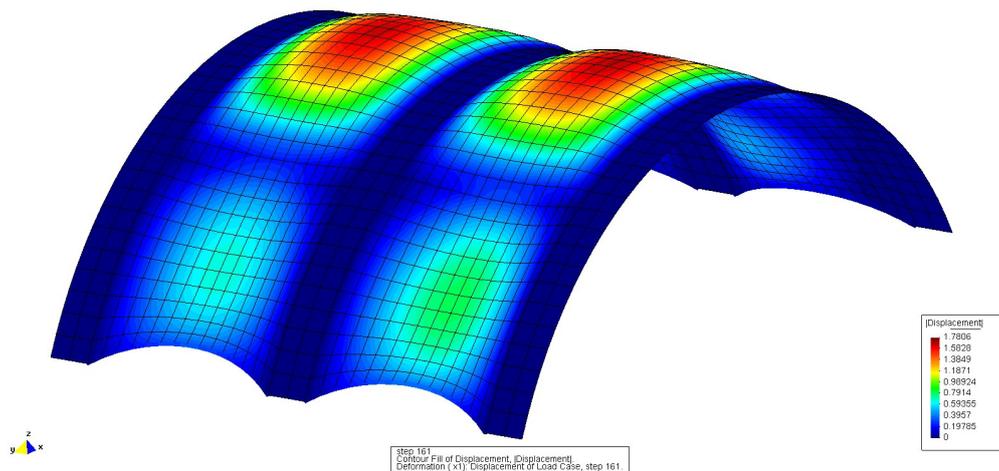


Abbildung 6.22.: Isogeometrische Darstellung der Verformungsamplitude bei $t = 0,8 \text{ s}$

den Punkt C zu $0,118 \text{ m}$. Diese Mittelwerte stellen das Äquivalent zu einer stationären Anströmung dar. Die Höhe der Verformungen entspricht den zum Zeitpunkt $t = 0,8 \text{ s}$ getroffenen Beobachtungen. Durch die starke Sogwirkung im oberen Bereich der Bogenhalle findet ein Durchschlagen der Membran statt (sichtbar in Abb. 6.22), wodurch sich am Punkt B die größten Verformungen einstellen. Durch die Reduktion der lokalen Steifigkeiten ergeben sich hier auch die größten Schwingungsamplituden. Es werden mehrere Eigenfrequenzen und Eigenformen der Struktur angeregt.

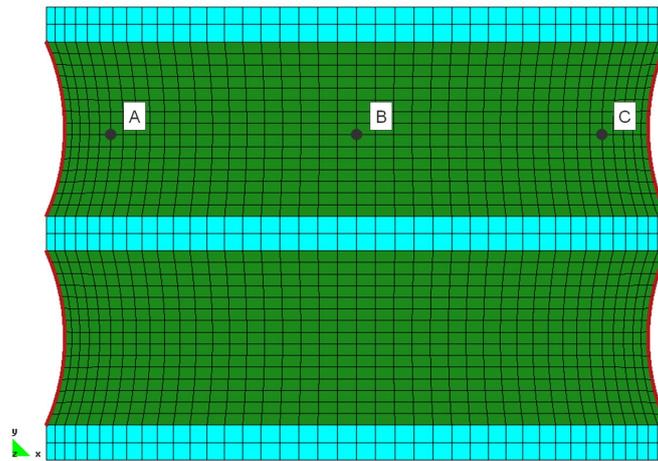


Abbildung 6.23.: Position der Auswertungspunkte auf der Hallenoberfläche

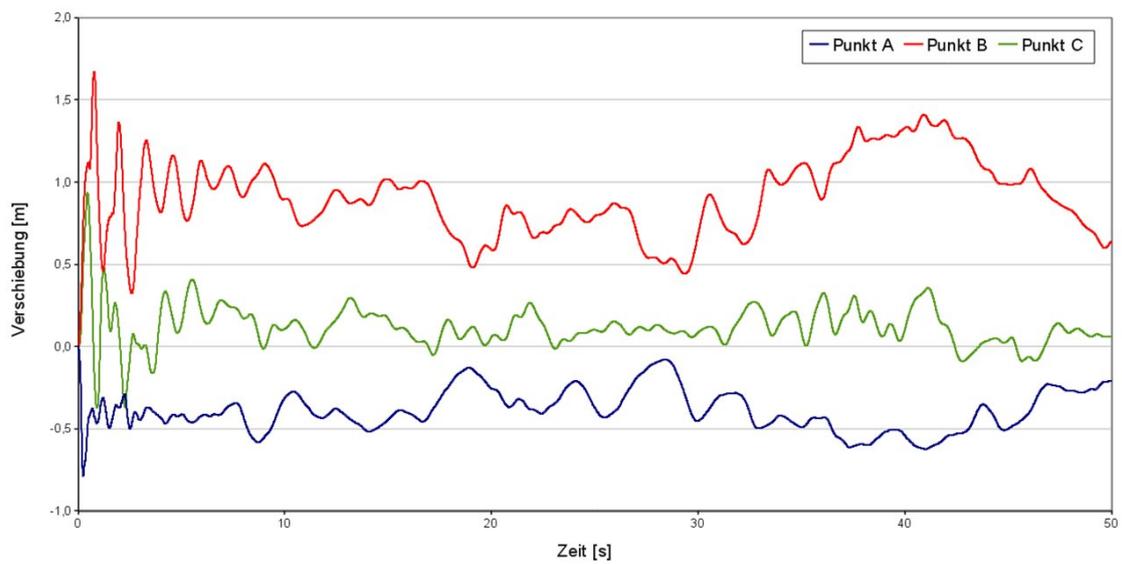


Abbildung 6.24.: Zeit-Verschiebungsdiagramm der Punkte A, B und C

6.2.5. Schlussfolgerung

Im Beispiel der windumströmten Bogenhalle werden drei Zielsetzungen verfolgt. Das Ergebnis des numerischen DFG-Benchmark stellt eine Empfehlung für eine bestimmte Kombination aus Kopplungsmethoden dar, um eine besonders effiziente und stabile gekoppelte Simulation durchzuführen. Diese Kombination wird innerhalb dieses komplexen Beispiels erfolgreich angewendet und dadurch validiert. Eine hohe Steigerung der Effizienz ist auch hier gegeben und damit ist die erste Zielsetzung erreicht. Die zweite Zielsetzung beinhaltet die Demonstration der Funktionsfähigkeit der entwickelten Softwareumgebung bei Behandlung komplexer dreidimensionaler Probleme. Da das behandelte Beispiel die geforderte Komplexität aufweist, ist auch diese Zielsetzung erreicht. Als drittes wird die Effizienz der entwickelten Parallelsierungsstrategie untersucht. Da die parallele Skalierung der gekoppelten Simulation nur geringfügig kleiner als die Skalierung einer Einzelfeldsimulation ist, gilt auch diese Zielsetzung als erfüllt.

6.3. Zusammenfassung

In diesem Kapitel werden anhand von zwei numerischen Beispielen einzelne Aspekte eines effizienten Gesamtlösungsansatzes für die Simulation stark gekoppelter Probleme der Fluid-Struktur-Wechselwirkung untersucht.

Der numerische DFG-Benchmark stellt ein geometrisch einfaches Beispiel dar, dessen Ergebnisse bekannt sind. Darin werden die in den Kapiteln 2, 3 und 4 vorgestellten Methoden miteinander verglichen, um eine Empfehlung für eine möglichst effiziente Methodenkombination geben zu können. Die untersuchten Methoden sind der Feldlöseransatz, der Interfaceprädiktor, und der Kopplungsalgorithmus. Basierend auf einer großen Anzahl an Einzelergebnissen durch Variation diverser Parameter wird die Kombination aus angepasstem Feldlöser, Newmark-Prädiktor zweiter Ordnung und Quasi-Newton-Kopplungsalgorithmus mit Wiederverwendung bekannter Information als äußerst effizient und stabil identifiziert. Des Weiteren wird festgestellt, dass der Feldlöseransatz und der Interfaceprädiktor einen ebenso hohen Einfluss auf das Ergebnis haben können wie der Kopplungsalgorithmus. Dieser Aspekt wird in vielen Arbeiten nicht beachtet.

Die Bogenhalle ist ein deutlich komplexeres Beispiel mit einer realistischen Problemstellung, der Simulation einer Wechselwirkung zwischen einer atmosphärischen Grenzschichtströmung und einem leichten Flächentragwerk. Dabei wird die hohe Effektivität des in Kapitel 5 vorgestellten Parallelisierungskonzeptes gezeigt. Ebenfalls wird die effiziente Methodenkombination des ersten Beispiels durch Anwendung auf dieses komplexe Problem weiter validiert. Im direkten Vergleich von Standardkombination und effizienter Kombination zeigt sich, dass die im ersten Beispiel erzielte Effizienzsteigerung auch in diesem komplexen Beispiel erreicht wird. Durch die Verwendung eines Windeinlassprofils ergibt sich ein stark instationäres Verhalten und es wird ein Zeitraum von 50 s untersucht.

7. Zusammenfassung und Ausblick

7.1. Zusammenfassung

Diese Arbeit beschäftigt sich mit dem Aspekt der Effizienz im Kontext einer partitionierten Simulation des Typs Fluid-Struktur-Wechselwirkung. Dabei werden stark gekoppelte und transiente Probleme mit großen Verformungen betrachtet. Die Kernthese lautet hierbei, dass sich eine hohe Gesamteffizienz nur aus der Kombination möglichst effizienter Teilaspekte ergibt und eine Beschränkung auf einen einzelnen Aspekt, z.B. ausschließlich auf den Kopplungsalgorithmus, eine unnötige Einschränkung darstellt. Deshalb werden in dieser Arbeit sowohl programmtechnische als auch kopplungsmethodische Aspekte untersucht.

Die kopplungsmethodischen Aspekte beinhalten die Algorithmen zur Lösung des gekoppelten nichtlinearen Problems. Innerhalb einer impliziten Kopplung sind dies die Feldlösungsansätze, die Bestimmung der Initialposition des Interfaces innerhalb eines Zeitschrittes und der Kopplungsalgorithmus. In den Kapiteln 2 und 3 werden die Feldlösungsalgorithmen für Struktur und Fluid hergeleitet. In Kapitel 4 wird die Fluid-Struktur-Wechselwirkung mit den Kopplungsalgorithmen und der Bestimmung der Initialposition dargestellt. Kapitel 5 beinhaltet ein numerisches Beispiel, bei welchem die kopplungsmethodischen Aspekte miteinander in Bezug auf Effizienz und Stabilität verglichen werden. Die Effizienz der Verfahren wird hierbei über einen Vergleich der Rechenzeit festgestellt, die Stabilität über eine Variation der Zeitschrittweite. Die einzelnen Methoden werden als erstes getrennt voneinander untersucht, danach werden auch Kombinationen erprobt.

Bei den Feldlösern findet eine Unterscheidung in konstante und angepasste Ansätze statt. In jeder Kopplungsiteration sind bei der iterativen Lösung der nichtlinearen Feldgleichungen Startwerte für die Feldvariablen anzunehmen. Konstante Feldlöser bestimmen diese mittels der Ergebnisse des letzten Zeitschritts. Dies ist das Standardvorgehen. Angepasste Feldlöser bestimmen diese basierend auf den Ergebnissen der letzten Kopplungsiteration, es wird also eine spezifische Erweiterung des Feldlösungsansatzes für die Verwendung innerhalb einer impliziten Kopplung vorgenommen. Im numerischen Beispiel zeigt sich beim direkten Vergleich der beiden Methoden, dass sich die Anzahl der benötigten Feldlösungsiterationen durch einen angepassten Feldlösungsansatz deutlich reduzieren lässt und damit die benötigte Rechenzeit um bis zu 23,7 % verringert wird.

Der zweite wichtige Aspekt behandelt die Bestimmung der Initialposition des Interfaces in der ersten Kopplungsiteration innerhalb eines Zeitschritts. Je näher die geschätzte Initialposition an der endgültigen Position liegt, desto geringer ist das Startresiduum am Interface und desto weniger Kopplungsiterationen werden benötigt. Die Bestimmung der Initialposition geschieht über einen Interfaceprädiktor, der aus bereits bekannten Werten vorheriger Zeitschritte eine neue Position extrapoliert. Es werden insgesamt sieben ver-

schiedende Prädiktoren angewendet, welche sich drei Gruppen zuordnen lassen. Die erste Gruppe sind polynombasierte Prädiktoren mit unterschiedlichem Polynomgrad, welche die neue Position basierend auf bekannten Verschiebungen bestimmen. Die zweite Gruppe stellen die Newmarkbasierten Prädiktoren dar, welche über die Newmarkansätze zusätzlich zu den Verschiebungen auch Geschwindigkeiten und Beschleunigungen des Interfaces bestimmen, und somit physikalisch höherwertige Informationen über das Interface enthalten. Die dritte Gruppe besteht aus einem adaptiven Prädiktor, der zusätzlich die zeitliche Veränderlichkeit des gekoppelten Problems durch eine Anpassung entsprechender Parameter abbildet. Im numerischen Beispiel zeigt sich, dass der Prädiktortyp gerade bei stark gekoppelten Problemen sehr hohen Einfluss auf die Rechenzeit hat. Zwischen dem besten und dem schlechtesten Prädiktor zeigt sich ein Unterschied in der Rechenzeit von 59,2 %. Als Empfehlung ergibt sich der Newmark-Prädiktor zweiter Ordnung, da er hohe Effizienz mit geringem Implementierungsaufwand kombiniert.

Der dritte wichtige Aspekt beinhaltet die Wahl des Kopplungsalgorithmus als Kern einer gekoppelten Berechnung. Da eine sehr hohe Anzahl an Kopplungsalgorithmen existiert, kann nur eine beschränkte Auswahl näher untersucht werden. Dazu wird eine Unterscheidung vorgenommen in Ansätze, welche Standardlösungsverfahren in den Teilfeldlösern verwenden, und Ansätze, welche eine Anpassung der Teilfeldlöser benötigen. In der ersten Gruppe werden die relaxationsbasierte Fixpunktiteration nach Aitken, die Vektorextrapolation und die Quasi-Newton-Methode untersucht. In der zweiten Gruppe werden inexakte Newton-Krylov-Verfahren untersucht mit einer Krylovraumevaluierung über finite Differenzen oder ein linearisiertes Ersatzproblem. Im numerischen Beispiel werden fünf Kopplungsalgorithmen in insgesamt 16 Ausprägungen untersucht und bei vier unterschiedlichen Fällen angewandt. Für einen Vergleich in Bezug auf Stabilität wird die Zeitschrittweite variiert. Es zeigt sich, dass bei großen Zeitschritten die Vektorextrapolationsverfahren und das Quasi-Newton-Verfahren ohne konstanten Relaxationsschritt divergieren. Alle anderen Verfahren bleiben weiterhin stabil. Für den Vergleich der Effizienz wird als absolutes Kriterium die benötigte Rechenzeit ausgewertet. Als Empfehlung ergibt sich das Quasi-Newton-Verfahren mit Wiederverwendung von Information vorheriger Zeitschritte, da es hohe Effizienz mit geringem Implementierungsaufwand kombiniert. Im direkten Vergleich mit der Aitken-Methode ergibt sich eine Reduktion der Rechenzeit um 40,1 % bis 71,9 %. Enttäuschend ist das Verhalten der Newton-Krylov-Varianten, welche eine Anpassung der Feldlöser benötigen. Diese benötigen zwar sehr wenige Kopplungsiterationen, die Kosten pro Iteration sind aber äußerst hoch. Dies zeigt sich bei einer Auswertung der Rechenzeit, welche sich im Vergleich mit Aitken im Maximum um das bis zu 3,43-fache erhöhen kann.

Nach der getrennten Untersuchung der einzelnen Methoden werden abschließend Kombinationen gegenübergestellt. Die erste Kombination stellt eine Standardkombination dar, da sie Methoden verwendet, die in sehr vielen Arbeiten zum Einsatz kommen. Sie besteht aus einem konstanten Feldlöser, einem Polynomprädiktor erster Ordnung und einer relaxationsbasierten Fixpunktiteration nach Aitken. Die zweite Kombination stellt eine besonders effiziente Methodenkombination dar. Basierend auf den bisherigen Ergebnissen besteht diese aus angepassten Feldlösern, einem Newmark-Prädiktor zweiter Ordnung und dem Quasi-Newton-Verfahren mit Wiederverwendung von 20 Leveln. In

den vier untersuchten Fällen ergibt die effiziente Kombination eine Reduzierung der Rechenzeit von 70,4 % bis zu 80,7 %. Hier zeigt sich also sehr deutlich, dass durch die Kombination der Methoden eine deutlich höhere Effizienzsteigerung erzielt werden kann, als es innerhalb eines einzelnen Bereiches möglich ist.

Die programmtechnischen Aspekte beinhalten die Softwareumgebung und das Parallelisierungskonzept. In dieser Arbeit wird eine Softwareumgebung entwickelt, die einen Drei-Programme-Ansatz verwendet. Dabei werden die drei Bereiche der Fluid-Struktur-Wechselwirkung, Fluid, Struktur und Wechselwirkung, jeweils in einem eigenen Programm behandelt. Diese Methode setzt die Idee des partitionierten Ansatzes konsequent um und zeichnet sich durch eine besonders hohe Flexibilität aus. Jedes der Programme ist nur für einen einzelnen spezifischen Aspekt zuständig und kann diesen möglichst effizient behandeln. Durch die Verwendung standardisierter Schnittstellen zwischen den Programmen ist die leichte Austauschbarkeit einzelner Teile gewährleistet. Wie im Kapitel fünf gezeigt, werden die in dieser Arbeit entwickelten Programme innerhalb vieler verschiedener Projekte mit verschiedensten Konstellationen verwendet. Dies zeigt die Leistungsfähigkeit der entwickelten Softwareumgebung und des Drei-Programme-Ansatzes. Einen weiteren wichtigen programmtechnischen Aspekt stellt das Parallelisierungskonzept dar. Im partitionierten Ansatz werden bereits vorhandene Teilfeldlöser miteinander gekoppelt. Diese weisen in der Regel bereits ein eigenes Parallelisierungskonzept auf. In der gekoppelten Simulation ist es das Ziel, die parallele Effizienz der Teilfeldlöser in keiner Weise zu beschränken, indem z.B. nur bestimmte Gebietszerlegungsmethoden verwendet werden dürfen oder das Kopplungsinterface nur auf einem einzelnen Prozess liegen darf. Das hier entwickelte Parallelisierungskonzept ermöglicht die uneingeschränkte Aufteilung des Kopplungsinterfaces und erlaubt damit die Verwendung jeglicher geometrischer Gebietszerlegungsmethoden.

Die Leistungsfähigkeit der Softwareumgebung und des Parallelisierungskonzeptes wird im zweiten numerischen Beispiel in Kapitel sechs gezeigt. Dieses Beispiel beinhaltet die realistische Problemstellung der Simulation einer Wechselwirkung zwischen einem biegeweichen Membrantragwerk und einer hochturbulenten Windströmung. Hier wird die effiziente Methodenkombination des ersten Beispiels angewendet und so an einem dreidimensionalen und komplexen Problem validiert. Des weiteren wird das Parallelisierungskonzept untersucht, indem die parallele Skalierung der gekoppelten Berechnung mit der Skalierung einer Einzelfeldberechnung verglichen wird. Es zeigt sich, dass die parallele Skalierung des gekoppelten Problems nur unwesentlich geringer als die des Einzelfeldproblems ist. Damit ist die Effizienz des Parallelisierungskonzeptes nachgewiesen.

7.2. Ausblick

Das in dieser Arbeit entwickelte Gesamtkonzept erlaubt es, realistische Problemstellungen des Typs Fluid-Struktur-Wechselwirkung hocheffizient zu simulieren. Damit ist auch die Behandlung komplexer Probleme möglich, welche eine gekoppelte Analyse als zentralen Bestandteil und die hier entwickelten Methoden als Werkzeug verwenden.

Die Software kann beispielsweise als numerischer Windkanal eingesetzt werden, der

im Vergleich mit einem realen Windkanal auch elastische Verformungen des umströmten Objektes abbilden kann. Damit werden auch Phänomene erfasst, die im realen Windkanal nicht sichtbar werden. Der hier gezeigte Fall einer Membran-Wind-Interaktion ist dabei nur ein Beispiel, in dem die Wechselwirkung maßgeblichen Einfluss auf das Strukturverhalten hat. Mit einer gekoppelten Berechnung als Kern ist möglicherweise auch die Erstellung eines erweiterten Bemessungskonzepts für Membranbauwerke unter Windeinfluss möglich.

Ein weiteres interessantes Gebiet ist die Formoptimierung oberflächengekoppelter Probleme. Damit kann beispielsweise die Querschnittsform eines Tragflügels in einem bestimmtem Flugzustand optimiert werden. Hierzu muss die gekoppelte Simulation um eine übergeordnete Optimierungsschleife erweitert werden. Durch die objektorientierte Programmierung der Software ist dies mit geringem Aufwand möglich und wird aktuell bereits in ersten Ansätzen umgesetzt ([BGKW06, HSG⁺10]).

Die Verwendung einer NURBS-basierten Diskretisierung in der Strukturanalyse als Alternative zur klassischen FEM ist eine interessante und relativ neue Entwicklung. NURBS ist ein mathematisches Modell zur analytischen Beschreibung von Kurven und Flächen. Damit können sowohl die Geometrie als auch die Verformungen einer Struktur exakt beschrieben werden. Nach einer Einzelfeldanalyse stellt die Verwendung von NURBS innerhalb einer gekoppelten Berechnung den logischen nächsten Schritt dar. Da ein Großteil der hier vorgestellten Lösungsmethoden unabhängig vom Diskretisierungsverfahren ist, können diese ohne Einschränkung weiter verwendet werden.

„Smart Structures“ sind Strukturen, die aktiv auf die Veränderung von Zuständen reagieren. Dies kann z.B. ein Flügel sein, der ab einer bestimmten Verformung oder beim Eintreten einer dynamischen Resonanz lokale Steifigkeiten verändert. Auch in diesem Fall können Phänomene interessant sein, die nur durch die Abbildung einer Wechselwirkung sichtbar werden. Ähnlich zur Formoptimierung muss dazu die gekoppelte Simulation einer übergeordneten Schleife zugeordnet werden. Auch dies ist mit geringem Aufwand möglich.

Die hier aufgezeigten Beispiele stellen natürlich nur einen kleinen Teil der zahlreichen Möglichkeiten dar. Diese Liste lässt sich beliebig erweitern.

A. Anhang

A.1. Numerischer DFG-Benchmark - Vergleichsergebnisse

In diesem Abschnitt werden die Ergebnisse des numerischen DFG-Benchmarks dargestellt, welche von den anderen an der DFG-Forschergruppe FOR493 beteiligten Gruppen angegeben werden.

Für den Fall FSI2 existiert keine offizielle Veröffentlichung. Deshalb wird auf die auf einer Forschergruppen-Internen Internetseite gesammelten Daten zurückgegriffen (<http://fsw.in.tum.de/intern/wiki/index.php/Benchmark>). Hier sind die Ergebnisse von sechs Gruppen dargestellt, inklusive der eigenen Gruppe des Lehrstuhl für Statik. Um der Veröffentlichung nicht vorzugreifen, wird für die einzelnen Ergebnisse nur die Bandbreite der berechneten Werte angegeben. Die Übersicht ist in Tabelle A.1 angegeben.

	eigene Ergebnisse	Bandbreite der weiteren Gruppen
$\mathbf{d}_x(\mathbf{A}) [m]$	$-0,01538 \pm 0,0127$	$[-0,0139 \text{ bis } -0,0151] \pm [0,0117 \text{ bis } 0,0129]$
$\mathbf{d}_y(\mathbf{A}) [m]$	$0,00123 \pm 0,0834$	$[0,00120 \text{ bis } 0,0049] \pm [0,0817 \text{ bis } 0,0876]$
$\sum_x \mathbf{f}^T [N]$	$224,63 \pm 85,96$	$[215,0 \text{ bis } 220,0] \pm [51,6 \text{ bis } 89,3]$
$\sum_y \mathbf{f}^T [N]$	$-0,474 \pm 268,08$	$[-12,12 \text{ bis } 0,61] \pm [237 \text{ bis } 382]$
$F_x [s^{-1}]$	$3,9$	$[3,16 \text{ bis } 3,86]$
$F_y [s^{-1}]$	$1,94$	$[1,9 \text{ bis } 1,949]$

Tabelle A.1.: FSI2 - eigene Ergebnisse und Bandbreite der Ergebnisse weiterer Gruppen

Für den Fall FSI3 wird auf die in [THR⁺10] veröffentlichten Ergebnisse der Forschergruppe zurückgegriffen. Die Übersicht ist in den Tabellen A.2 und A.3 angegeben. Aus Platzgründen werden die Einheiten nicht angegeben, sie sind identisch mit denen aus der Tabelle FSI2. Die Spaltenbezeichnung *FG* bedeutet *Freiheitsgrad*.

Gruppe	FG	$\Delta t [10^{-3}]$	$\mathbf{d}_x(\mathbf{A}) [10^{-3}]$	$\mathbf{d}_y(\mathbf{A}) [10^{-3}]$	$\sum_x \mathbf{f}^I$
1	61318	1,0	$-2,54 \pm 2,41$	$1,45 \pm 32,80$	$450,3 \pm 23,51$
	237286	2,0	$-2,88 \pm 2,73$	$1,53 \pm 34,94$	$458,6 \pm 27,18$
	237286	1,0	$-2,87 \pm 2,73$	$1,54 \pm 34,94$	$458,6 \pm 27,31$
	237286	0,5	$-2,86 \pm 2,72$	$1,53 \pm 34,90$	$458,6 \pm 27,27$
	941158	1,0	$-2,91 \pm 2,77$	$1,47 \pm 35,26$	$459,9 \pm 27,92$
2a	11250	5,0	$-2,48 \pm 2,24$	$1,27 \pm 36,50$	–
2b	7176	5,0	$-2,44 \pm 2,32$	$1,02 \pm 31,82$	$473,5 \pm 56,97$
	7176	2,0	$-2,48 \pm 2,39$	$0,92 \pm 32,81$	$471,3 \pm 62,28$
	7176	1,0	$-2,58 \pm 2,49$	$0,94 \pm 33,19$	$470,4 \pm 64,02$
	27744	5,0	$-2,43 \pm 2,27$	$1,41 \pm 31,73$	$483,7 \pm 22,31$
	27744	2,0	$-2,63 \pm 2,61$	$1,46 \pm 33,46$	$483,3 \pm 24,48$
	27744	1,0	$-2,80 \pm 2,64$	$1,45 \pm 34,12$	$483,0 \pm 25,67$
	42024	2,5	$-2,40 \pm 2,26$	$1,39 \pm 31,71$	$448,7 \pm 21,16$
	42024	1,0	$-2,53 \pm 2,38$	$1,40 \pm 32,49$	$449,7 \pm 22,24$
	42024	0,5	$-2,57 \pm 2,42$	$1,42 \pm 32,81$	$450,1 \pm 22,49$
	72696	2,5	$-2,64 \pm 2,48$	$1,38 \pm 33,25$	$451,1 \pm 24,57$
	72696	1,0	$-2,79 \pm 2,62$	$1,28 \pm 34,61$	$452,0 \pm 25,78$
	72696	0,5	$-2,84 \pm 2,67$	$1,28 \pm 34,61$	$452,4 \pm 26,19$
	3	19488	1,0	$-3,02 \pm 2,83$	$1,41 \pm 35,47$
19488		0,5	$-3,02 \pm 2,85$	$1,42 \pm 35,63$	$458,7 \pm 28,78$
19488		0,25	$-3,02 \pm 2,85$	$1,32 \pm 35,73$	$458,7 \pm 28,80$
76672		1,0	$-2,78 \pm 2,62$	$1,44 \pm 34,36$	$459,1 \pm 26,63$
76672		0,5	$-2,78 \pm 2,62$	$1,44 \pm 34,35$	$459,1 \pm 26,62$
76672		0,25	$-2,77 \pm 2,61$	$1,43 \pm 34,43$	$459,1 \pm 26,50$
304128		1,0	$-2,86 \pm 2,70$	$1,45 \pm 34,93$	$460,2 \pm 27,65$
304128		0,5	$-2,86 \pm 2,70$	$1,45 \pm 34,90$	$460,2 \pm 27,47$
304128		0,25	$-2,88 \pm 2,72$	$1,47 \pm 34,99$	$460,5 \pm 27,74$
4	81120	0,09	$-5,18 \pm 5,04$	$1,12 \pm 45,10$	$477,0 \pm 48,00$
	324480	0,02	$-4,54 \pm 4,34$	$1,50 \pm 42,50$	$467,5 \pm 39,50$
5	2480814	0,051	$-2,88 \pm 2,71$	$1,48 \pm 35,10$	$463,0 \pm 31,30$
6	7059	0,5	$-1,60 \pm 1,60$	$1,50 \pm 25,90$	$525,0 \pm 22,50$
	27147	0,5	$-2,00 \pm 1,89$	$1,45 \pm 29,00$	$434,0 \pm 17,50$
7	271740	0,5	$-3,04 \pm 2,87$	$1,55 \pm 36,63$	$474,9 \pm 28,12$

Tabelle A.2.: FSI3 - eigene Ergebnisse (Gruppe 7) und Ergebnisse weiterer Gruppen ([THR+10]) - Teil 1

Gruppe	FG	Δt	$\sum_y \mathbf{f}^I$	F_x	F_y
1	61318	1,0	$-0,10 \pm 143,0$	10,90	5,13
	237286	2,0	$2,08 \pm 153,1$	10,60	5,30
	237286	1,0	$2,00 \pm 153,3$	10,34	5,91
	237286	0,5	$2,01 \pm 153,4$	12,16	6,08
	941158	1,0	$1,84 \pm 157,7$	11,63	4,98
2a	11250	5,0	–	10,10	5,10
2b	7176	5,0	$8,08 \pm 283,8$	11,07	5,29
	7176	2,0	$6,11 \pm 298,6$	10,73	5,35
	7176	1,0	$4,65 \pm 300,3$	10,69	5,36
	27744	5,0	$2,21 \pm 149,0$	10,53	5,37
	27744	2,0	$2,08 \pm 161,2$	10,66	5,43
	27744	1,0	$2,21 \pm 165,3$	10,75	5,41
	42024	2,5	$1,84 \pm 141,3$	10,72	5,42
	42024	1,0	$1,61 \pm 142,8$	10,77	5,44
	42024	0,5	$1,49 \pm 143,7$	10,79	5,42
	72696	2,5	$2,04 \pm 150,6$	10,73	5,38
	72696	1,0	$1,91 \pm 152,7$	10,78	5,42
	72696	0,5	$2,36 \pm 152,7$	10,84	5,42
	3	19488	1,0	$2,41 \pm 145,6$	10,75
19488		0,5	$2,23 \pm 146,0$	10,75	5,37
19488		0,25	$2,23 \pm 146,0$	10,74	5,33
76672		1,0	$2,41 \pm 151,3$	10,93	5,46
76672		0,5	$2,39 \pm 150,7$	10,92	5,46
76672		0,25	$2,36 \pm 149,9$	10,93	5,46
304128		1,0	$2,47 \pm 154,9$	10,95	5,47
304128		0,5	$2,37 \pm 153,8$	10,92	5,46
304128		0,25	$2,50 \pm 153,9$	10,93	5,46
4	81120	0,09	$7,00 \pm 223,0$	10,14	4,99
	324480	0,02	$16,20 \pm 188,7$	10,12	5,05
5	2480814	0,051	$1,81 \pm 154,0$	11,00	5,50
6	7059	0,5	$-0,55 \pm 106,0$	10,90	5,45
	27147	0,5	$2,53 \pm 88,6$	10,60	5,30
7	271740	0,5	$3,86 \pm 165,9$	10,99	5,51

Tabelle A.3.: FSI3 - eigene Ergebnisse (Gruppe 7) und Ergebnisse weiterer Gruppen ([THR+10]) - Teil2

Literaturverzeichnis

- [AM86] ARGYRIS, J.H. ; MLEJNEK, H.P.: *Die Methode der Finiten Elemente in der elementaren Strukturmechanik. Band III: Einführung in die Dynamik.* Braunschweig; Wiesbaden: Friedr. Vieweg & Sohn, 1986
- [Bat90] BATINA, J.: Unsteady Euler airfoil solutions using unstructured dynamic meshes. In: *AIAA journal* 28 (1990), Nr. 8, S. 1381–1388
- [Bat01] BATHE, K.J.: *Finite-Elemente-Methoden.* Springer, 2001
- [BBM06] BARCELOS, M. ; BAVESTRELLO, H. ; MAUTE, K.: A Schur-Newton-Krylov solver for steady-state aeroelastic analysis and design sensitivity analysis. In: *Computer Methods in Applied Mechanics and Engineering* 195 (2006), Nr. 17-18, S. 2050–2069
- [BFLW10] BLETZINGER, K.-U. ; FIRL, M. ; LINHARD, J. ; WÜCHNER, R.: Optimal shapes of mechanically motivated surfaces. In: *Computer Methods in Applied Mechanics and Engineering* 199 (2010), Nr. 5-8, S. 324–333. <http://dx.doi.org/10.1016/j.cma.2008.09.009>. – DOI 10.1016/j.cma.2008.09.009
- [BGKW06] BLETZINGER, K.-U. ; GALLINGER, T. ; KUPZOK, A. ; WÜCHNER, R.: Partitioned Strategies for Optimization and FSI. In: *Proceedings of the EC-COMAS CFD 2006.* Egmond aan Zee, Netherlands, 2006
- [Bis99] BISCHOFF, M.: *Theorie und Numerik einer dreidimensionalen Schalenformulierung,* Institut für Baustatik, Universität Stuttgart, Diss., 1999
- [BLM99] BELYTSCHKO, T. ; LIU, W.K. ; MORAN, B.: *Nonlinear finite element analysis for continua and structures.* Wiley New York, 1999
- [BLM00] BELYTSCHKO, T. ; LIU, W.K. ; MORAN, B.: *Nonlinear finite elements for continua and structures.* Wiley New York, 2000
- [Blo00] BLOM, F.J.: Considerations on the spring analogy. In: *International journal for numerical methods in fluids* 32 (2000), Nr. 6, S. 647–668
- [BLVOB07] BOS, F.M. ; LENTINK, D. ; VAN OUDHEUSDEN, B.W. ; BIJL, H.: Influence of wing kinematics on aerodynamic performance in hovering insect flight. In: *Journal of Fluid Mechanics* 594 (2007), S. 341–368

- [BMB05] BRUMMELEN, E.H. van ; MICHLER, C. ; BORST, R. de: Interface-GMRES (R) acceleration of subiteration for fluid-structure-interaction problems / DACS-05-001, Delft Aerospace Computational Science. 2005. – Forschungsbericht
- [BMT⁺08] BOS, F.M. ; MATIJAŠEVIC, D. ; TERZE, Z. ; VAN OUDHEUSDEN, B.W. ; BIJL, H.: OpenFOAM mesh motion using Radial Basis Function interpolation. In: *Proceedings of the ECCOMAS CFD 2008* (2008)
- [BNV08a] BADIA, S. ; NOBILE, F. ; VERGARA, C.: Fluid-structure partitioned procedures based on Robin transmission conditions. In: *Journal of Computational Physics* 227 (2008), S. 7027–7051
- [BNV08b] BADIA, S. ; NOBILE, F. ; VERGARA, C.: The Robin-Robin/GMRES Algorithm for Fluid-Structure Interaction: Large Added-Mass Effect and Balloon-Type Problems. In: *Proceedings of the 8th World Congress on Computational Mechanics*, 2008
- [BNV09] BADIA, S. ; NOBILE, F. ; VERGARA, C.: Robin-Robin preconditioned Krylov methods for fluid-structure interaction problems. In: *Computer Methods in Applied Mechanics and Engineering* 198 (2009), Nr. 33-36, S. 2768–2784
- [BR92] BÜCHTER, N. ; RAMM, E.: 3d-extension of nonlinear shell equations based on the enhanced assumed strain concept. In: *Computational Methods in Applied Sciences* (1992), S. 55–62
- [BR99] BLETZINGER, K.U. ; RAMM, E.: A general finite element approach to the form finding of tensile structures by the updated reference strategy. In: *International Journal of Space Structures* 14 (1999), Nr. 2, S. 131–145
- [Bre01] BREUER, M.: *Direkte Numerische Simulation und Large-Eddy Simulation turbulenter Strömungen auf Hochleistungsrechnern*. Shaker Verlag Aachen, 2001
- [BS06] BUNGARTZ, H.-J. (Hrsg.) ; SCHÄFER, M. (Hrsg.): *Fluid-Structure Interaction. Modelling, Simulation, Optimization*. Springer, 2006
- [BS10] BUNGARTZ, Hans-Joachim (Hrsg.) ; SCHÄFER, Michael (Hrsg.): *Fluid-Structure Interaction. Part II*. Springer, 2010 (Lecture Notes in Computational Science and Engineering 53)
- [BSC07] BLOCKEN, B. ; STATHOPOULOS, T. ; CARMELIET, J.: CFD simulation of the atmospheric boundary layer - wall function problems. In: *Atmospheric Environment* 41(2) (2007), S. 238–252
- [BWBR04] BISCHOFF, M. ; WALL, W.A. ; BLETZINGER, K.U. ; RAMM, E.: Models and finite elements for thin-walled structures. In: *Encyclopedia of computational mechanics* 2 (2004), S. 59–137

- [BZB07] BOER, A. de ; ZUIJLEN, A.H. van ; BIJL, H.: Review of coupling methods for non-matching meshes. In: *Computer Methods in Applied Mechanics and Engineering* 196 (2007), S. 1515–1525
- [BZB08] BOER, A. de ; ZUIJLEN, A.H. van ; BIJL, H.: Comparison of conservative and consistent approaches for the coupling of non-matching meshes. In: *Computer Methods in Applied Mechanics and Engineering* 197 (2008), Nr. 49-50, S. 4284–4297
- [CB91] CARTER, E.D. ; BAKER, A.J.: Accuracy and stability of a finite element pseudo-compressibility CFD algorithm for incompressible thermal flows. In: *Numerical Heat Transfer, Part B: Fundamentals* 20 (1991), Nr. 1, S. 1–23
- [CGN05] CAUSIN, P. ; GERBEAU, J.F. ; NOBILE, F.: Added-mass effect in the design of partitioned algorithms for fluid–structure problems. In: *Computer Methods in Applied Mechanics and Engineering* 194 (2005), Nr. 42-44, S. 4506–4527
- [CH93] CHUNG, J. ; HULBERT, G.M.: A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-alpha Method. In: *Journal of Applied Mechanics* 60 (1993), S. 371–375
- [Cho97] CHORIN, A.J.: A numerical method for solving incompressible viscous flow problems. In: *Journal of Computational Physics* 135 (1997), S. 118–125
- [CL97] CEBRAL, J.R. ; LÖHNER, R.: Conservative Load Projection and Tracking for Fluid-Structure Problems. In: *AIAA Journal of Aircraft* 35(4) (1997), S. 687–692
- [CP93] CLOUGH, R. ; PENZIEN, J.: *Dynamics of structures*. McGraw Hill, 1993
- [DAV10] DEGROOTE, J. ; ANNEREL, S. ; VIERENDEELS, J.: Stability analysis of Gauss–Seidel iterations in a partitioned simulation of fluid-structure interaction. In: *Computers and Structures* 88(5-6) (2010), S. 263–271. <http://dx.doi.org/10.1016/j.compstruc.2009.09.003>. – DOI 10.1016/j.compstruc.2009.09.003
- [DBV09] DEGROOTE, J. ; BATHE, K.-J. ; VIERENDEELS, J.: Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction. In: *Computers and Structures* 87, 11-12 (2009), S. 193–801. <http://dx.doi.org/10.1016/j.compstruc.2008.11.013>. – DOI 10.1016/j.compstruc.2008.11.013
- [Dep04] DEPARIS, S.: *Numerical analysis of axisymmetric flows and methods for fluid-structure interaction arising in blood flow simulation*, ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, Diss., 2004

- [Deu04] DEUFLHARD, P.: *Newton methods for nonlinear problems: affine invariance and adaptive algorithms*. Springer Verlag, 2004
- [DF02] DEGAND, C. ; FARHAT, C.: A three-dimensional torsional spring analogy method for unstructured dynamic meshes. In: *Computers and Structures* 80 (2002), Nr. 3-4, S. 305–316
- [DGV04] DEPARIS, S. ; GERBEAU, J.-F. ; VASSEUR, X.: A Dynamic Preconditioner for Newton-Krylov Algorithms. Application to Fluid-Structure Interaction. / INRIA. 2004. – Forschungsbericht
- [DHPRF04] *Kapitel Arbitrary Lagrangian-Eulerian Methods*. In: DONEA, J. ; HUERTA, A. ; PONTHOT, J.P. ; RODRIGUEZ-FERRAN, A.: *Encyclopedia of Computational Mechanics, Volume 2: Solids and Structures*. John Wiley & Sons, Ltd, 2004, S. 413–438
- [Die09] DIERINGER, F.: *Implementierung eines geometrisch nichtlinearen Membranelements in einer objektorientierten Programmierumgebung*, Lehrstuhl für Statik , Technische Universität München, Diplomarbeit, 2009
- [DIN05] DIN, Normenausschuss B.: DIN 1055-4:2005-3. Einwirkung auf Tragwerke - Teil4: Windlasten / DIN Deutsches Institut für Normung e.V. 2005. – Forschungsbericht
- [DP88] DEMIRDŪIĆ, I. ; PERIĆ, M.: Space conservation law in finite volume calculations of fluid flow. In: *International Journal for Numerical Methods in Fluids* 8 (1988), S. 1037–1050
- [DR06] DAHMEN, W. ; REUSKEN, A.: *Numerik für Ingenieure und Naturwissenschaftler*. Springer, 2006
- [FFB10] FISCHER, M. ; FIRL, M. ; BLETZINGER, K.-U.: Nichtlineare Strukturoptimierung und Simulation basierend auf objektorientierter und paralleler Programmierung. In: *Fraunhofer-Symposium: Simulation für robuste Produkte und Prozesse*. Bremen, Germany, 2010
- [FFMB10] FISCHER, M. ; FIRL, M. ; MASCHING, H. ; BLETZINGER, K.-U.: Optimization of Nonlinear Structures based on Object-Oriented Parallel Programming. In: *ECT2010: The Seventh International Conference on Engineering Computational Technology*. Valencia, Spain, 2010
- [FHJ⁺04] FRANKE, J. ; HIRSCH, C. ; JENSEN, A. G. ; KRŪS, H.W. ; SCHATZMANN, M. ; WESTBURY, P.S. ; MILES, S.D. ; WISSE, J.A. ; WRIGHT, N.G.: Recommendations on the use of CFD in Wind Engineering. In: *Proceedings of the International Conference on Urban Wind Engineering and Building Aerodynamics, (Ed. van Beeck, JPAJ), COST Action C 14* (2004)

- [Fis09] FISCH, R.: *Fluid-Struktur Interaktion geometrisch reduzierter Strukturmodelle - Konzeption und Realisierung eines Interface*, Lehrstuhl für Statik, Technische Universität München, Diplomarbeit, 2009
- [FL98] FARHAT, C. ; LESOINNE, M.: Enhanced Partitioned Procedures for Solving Nonlinear Transient Aeroelastic Problems. In: *AIAA* (1998)
- [FL00] FARHAT, C. ; LESOINNE, M.: Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems. In: *Computer Methods in Applied Mechanics and Engineering* 182 (2000), S. 499–515
- [FLL98] FARHAT, C. ; LESOINNE, M. ; LETALLEC, P.: Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. In: *Computer Methods in Applied Mechanics and Engineering* 157 (1998), S. 95–114
- [FM03] FERNÁNDEZ, M.Á. ; MOUBACHIR, M.: An exact Block–Newton algorithm for solving fluid–structure interaction problems. In: *Comptes rendus-Mathématique* 336 (2003), Nr. 8, S. 681–686
- [FM05] FERNÁNDEZ, M.Á. ; MOUBACHIR, M.: A Newton method using exact Jacobians for solving fluid-structure coupling. In: *Computers and Structures* 83 (2005), Nr. 2-3, S. 127–142
- [FP99] FERZIGER, J.H. ; PERIĆ, M.: *Computational Methods for Fluid Dynamics*. Springer New York, 1999
- [FP08] FERZIGER, J.H. ; PERIĆ, M.: *Numerische Strömungsmechanik*. Springer Berlin Heidelberg, 2008
- [FPF01] FELIPPA, C. A. ; PARK, K. C. ; FARHAT, C.: Partitioned analysis of coupled mechanical systems. In: *Computer Methods in Applied Mechanics and Engineering* 190 (2001), S. 3247–3270
- [Fro06] FROELICH, J.: *Large eddy simulation turbulenter Strömungen*. Vieweg+Teubner Verlag, 2006
- [FVCJ⁺06] FIGUEROA, C.A. ; VIGNON-CLEMENTEL, I.E. ; JANSEN, K.E. ; HUGHES, T.J.R. ; TAYLOR, C.A.: A coupled momentum method for modeling blood flow in three-dimensional deformable arteries. In: *Computer Methods in Applied Mechanics and Engineering* 195 (2006), Nr. 41-43, S. 5685–5706
- [FWR06] FÖRSTER, C. ; WALL, W.A. ; ; RAMM, E.: On the geometric conservation law in transient flow calculations on deforming domains. In: *International Journal for Numerical Methods in Fluids* 50 (2006), S. 1369–1379

- [FWR07] FÖRSTER, C. ; WALL, W.A. ; RAMM, E.: Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows. In: *Computer Methods in Applied Mechanics and Engineering* 196 (2007), Nr. 7, S. 1278–1293
- [FZG06] FARHAT, C. ; ZEE, K. G. d. ; GEUZAIN, P.: Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity. In: *Computer Methods in Applied Mechanics and Engineering* 195 (2006), S. 1973–2001. <http://dx.doi.org/10.1016/j.cma.2004.11.031>. – DOI 10.1016/j.cma.2004.11.031
- [GB10] GALLINGER, T. ; BLETZINGER, K.-U.: Comparison of Algorithms for strongly coupled partitioned Fluid-Structure Interaction - Efficiency versus Simplicity. In: PEREIRA, J. C. F. (Hrsg.) ; SEQUEIRA, A. (Hrsg.): *Proceedings of the ECCOMAS CFD 2010*. Lisbon, 2010
- [GDN95] GRIEBEL, M. ; DORNSEIFER, T. ; NEUNHOEFFER, T.: *Numerische Simulation in der Strömungsmechanik*. Vieweg Braunschweig, Wiesbaden, 1995
- [Gee04] GEE, M. W.: *Effiziente Lösungsstrategien in der nichtlinearen Schalenmechanik*, Universität Stuttgart, Diss., 2004
- [GKBW09] GALLINGER, T. ; KUPZOK, A. ; BLETZINGER, K.-U. ; WÜCHNER, R.: Modeling and Simulation of Light-Weight Membrane Structures under Wind Influence. In: *Proceedings of the ECCOMAS Coupled Problems 2009*. Ischia, Italy, 2009
- [GKI⁺09] *Kapitel A Computational Environment for Membrane-Wind Interaction*. In: GALLINGER, T. ; KUPZOK, A. ; ISRAEL, U. ; WUECHNER, R. ; BLETZINGER, K.-U.: *Fluid-Structure Interaction. Theory, Numerics and Applications*. kassel university press, 2009, S. 97–114
- [GKUW09] GALLINGER, T. ; K.-U., Bletzinger ; WÜCHNER, R.: Light-Weight Structure Design with regard to Wind-Induced Effects. In: *Proceedings of the Forschungskolloquium Baustatik-Baupraxis 2009*. Falkenstein, Harz, 2009
- [GKWB07a] GALLINGER, T. ; KUPZOK, A. ; WÜCHNER, R. ; BLETZINGER, K.-U.: Aspects and Development of a Flexible and Parallel Toolkit for FSI-Simulations in Civil Engineering. In: *Proceedings of the 2nd GACM Colloquium on Computational Mechanics*. Munich, Germany, 2007
- [GKWB07b] GALLINGER, T. ; KUPZOK, A. ; WÜCHNER, R. ; BLETZINGER, K.-U.: Massiv parallele Simulation gekoppelter Probleme der Aeroelastik. In: *Proceedings of the 19. Forum Bauinformatik*. Graz, Austria, 2007
- [GLDS96] GROPP, W. ; LUSK, E. ; DOSS, N. ; SKJELLUM, A.: A high-performance, portable implementation of the MPI message passing interface standard. In: *Parallel Computing* 22 (1996), Nr. 6, S. 789–828

- [GLS99] GROPP, W. ; LUSK, E. ; SKJELLUM, A.: *Using MPI: portable parallel programming with the message-passing interface*. MIT press Massachusetts USA, 1999
- [Gup97] GUPTA, A.: Fast and effective algorithms for graph partitioning and sparse-matrix ordering. In: *IBM Journal of Research and Development* 41 (1997), Nr. 1, S. 171–183
- [GV03] GERBEAU, J.-F. ; VIDRASCU, M.: A Quasi-Newton Algorithm Based on a Reduced Model for Fluid-Structure Interaction Problems in Blood Flows / INRIA. 2003. – Forschungsbericht
- [GVF05] GERBEAU, J.F. ; VIDRASCU, M. ; FREY, P.: Fluid–structure interaction in blood flows on geometries based on medical imaging. In: *Computers and structures* 83 (2005), Nr. 2-3, S. 155–165
- [HAC74] HIRT, C.W. ; AMSDEN, A.A. ; COOK, J.L.: An arbitrary Lagrangian-Eulerian computing method for all flow speeds. In: *Journal of Computational Physics* 14 (1974), Nr. 3, S. 227–253
- [Har07a] HARGREAVES, Wright N. D.M.: On the use of the k- ϵ model in commercial CFD software to model the neutral atmospheric boundary layer. In: *Journal of Wind Engineering and Industrial Aerodynamics* 95(5) (2007), S. 355–369. <http://dx.doi.org/10.1016/j.jweia.2006.08.002>. – DOI 10.1016/j.jweia.2006.08.002
- [Har07b] HARTMANN, S.: *Kontaktanalyse dünnwandiger Strukturen bei grossen Deformationen*, Institut für Baustatik und Baudynamik der Universität Stuttgart, Diss., 2007
- [Hei04] HEIL, M.: An efficient solver for the fully coupled solution of large-displacement fluid-structure interaction problems. In: *Computer Methods in Applied Mechanics and Engineering* 193 (2004), Nr. 1-2, S. 1–23
- [HHT77] HILBER, H.M. ; HUGHES, T.J.R. ; TAYLOR, R.L.: Improved numerical dissipation for time integration algorithms in structural dynamics. In: *Earthquake engineering and structural dynamics* 5 (1977), Nr. 3, S. 283–292
- [Hol00] HOLZAPFEL, G. A.: *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. Chichester : John Wiley and Sons, 2000
- [HSG⁺10] HOJJAT, M. ; STAVROPOULOU, E. ; GALLINGER, T. ; ISRAEL, U. ; WÜCHNER, R. ; K.-U., Bletzinger: Fluid-Structure Interaction in the Context of Shape Optimization and Computational Wind Engineering. In: [BS10]
- [Hug00] HUGHES, T.J.R.: *The Finite Element Method*. Dover publications, 2000

- [IGW86] ISSA, R.I. ; GOSMAN, A.D. ; WATKINS, A.P.: The Computation of Compressible and Incompressible Recirculating Flows by a Non-iterative Implicit Scheme. In: *Journal of Computational Physics* 62 (1986), S. 66–82
- [IKL⁺07] ISRAEL, U. ; KUPZOK, A. ; LAEHR, A. ; GALLINGER, T. ; WUECHNER, R. ; K.-U., Bletzinger: A flexible interface program for code coupling in partitioned multiphysics analysis. In: *Proceedings of the 2nd GACM Colloquium on Computational Mechanics*, 2007
- [Iss85] ISSA, R.I.: Solution of the Implicitly Discretised Fluid Flow Equations by Operator-Splitting. In: *Journal of Computational Physics* 62 (1985), S. 40–65
- [IT69] IRONS, B.M. ; TUCK, R.C.: A version of the Aitken accelerator for computer iteration. In: *International Journal for Numerical Methods in Engineering* 1 (1969), Nr. 3
- [Jas96] JASAK, H.: *Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows*, Imperial College London, Diss., 1996
- [JDP08] JOOSTEN, M.M. ; DETTMER, W.G. ; PERIC, D.: Analysis of the block Gauss–Seidel solution procedure for a strongly coupled model problem with reference to fluid–structure interaction. In: *Int. J. Numer. Meth. Engng.* 78 (2008), S. 757–778. <http://dx.doi.org/10.1002/nme.2503>. – DOI 10.1002/nme.2503
- [JJGL05] JAIMAN, R.K. ; JIAO, X. ; GEUBELLE, P.H. ; LOTH, E.: Assessment of conservative load transfer for fluid-solid interface with non-matching meshes. In: *Int. J. Numer. Meth. Engng* 64 (2005), S. 2014–2038
- [JJM07] JASAK, H. ; JEMCOV, A. ; MARUSZEWSKI, J.P.: Preconditioned Linear Solvers for Large Eddy Simulation. In: *CFD 2007 Conference, CFD Society of Canada*, 2007
- [JM07] JEMCOV, A. ; MARUSZEWSKI, J. P.: Acceleration and Stabilization of Algebraic Multigrid Solver applied to incompressible flow problems. In: *AIAA* (2007)
- [JMJ07] JEMCOV, A. ; MARUSZEWSKI, J.P. ; JASAK, H.: Performance Improvement of Algebraic Multigrid Solver by Vector Sequence Extrapolation. In: *CFD 2007 Conference, CFD Society of Canada*, 2007
- [JS00] JBILOU, K. ; SADOK, H.: Vector extrapolation methods. Applications and numerical comparison. In: *Journal of computational and applied mathematics* 122 (2000), Nr. 1-2, S. 149–165
- [JT94] JOHNSON, A.A. ; TEZDUYAR, T.E.: Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and

- interfaces. In: *Computer Methods in Applied Mechanics and Engineering* 119 (1994), Nr. 1, S. 73–94
- [JT07] JASAK, H. ; TUKOVIC, Z.: Automatic Mesh Motion for the Unstructured Finite Volume Method. In: *Transactions of FAMENA* 30 (2007), S. 1–18
- [Kel95] KELLEY, C.T.: *Iterative methods for linear and nonlinear equations*. Society for Industrial Mathematics, 1995
- [Kel03] KELLEY, C.T.: *Solving nonlinear equations with Newton's method*. Society for Industrial Mathematics, 2003
- [KK95] KARYPIS, G. ; KUMAR, V.: METIS - Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 2.0. In: *The University of Minnesota* 2 (1995)
- [KK97] KARYPIS, G. ; KUMAR, V.: ParMETIS: Parallel Graph Partitioning and Sparse Matrix Ordering Library / Department of Computer Science, Univ. of Minnesota. 1997 (97-060). – Forschungsbericht
- [KK99] KARYPIS, G. ; KUMAR, V.: A fast and high quality multilevel scheme for partitioning irregular graphs. In: *SIAM Journal on Scientific Computing* 20 (1999), Nr. 1, S. 359
- [KK04] KNOLL, D.A. ; KEYES, D.E.: Jacobian-free Newton-Krylov methods: a survey of approaches and applications. In: *Journal of Computational Physics* 193 (2004), Nr. 2, S. 357–397
- [KM00] KNOLL, D.A. ; MOUSSEAU, V.A.: On Newton-Krylov multigrid methods for the incompressible Navier-Stokes equations. In: *Journal of Computational Physics* 163 (2000), Nr. 1, S. 262–267
- [Kna07] KNAPP, G.A.: *Improved Methods for Structural Wind Engineering*, University of Nottingham, Diss., 2007
- [Kol10] KOLLMANNBERGER, S.: *ALE-type and fixed grid fluid-structure interaction involving the p-version of the Finite Element Method*, Technische Universität München, Lehrstuhl Computation in Engineering, Diss., 2010
- [Kue09] KUETTLER, U.: *Effiziente Lösungsverfahren für Fluid-Struktur-Interaktions-Probleme*, Technische Universität München, Lehrstuhl für numerische Mechanik, Diss., 2009
- [Kuh96] KUHLE, D.: *Stabile Zeitintegrationsalgorithmen in der nichtlinearen Elastodynamik dünnwandiger Tragwerke*, Institut für Baustatik, Universität Stuttgart, Diss., 1996

- [Kup09] KUPZOK, A.: *Modeling the Interaction of Wind and Membrane Structures by Numerical Simulation*, Lehrstuhl für Statik der Technischen Universität München, Diss., 2009
- [KW08] KÜTTLER, U. ; WALL, W. A.: Fixed-point fluid-structure interaction solvers with dynamic relaxation. In: *Computational Mechanics* 43(1) (2008), S. 61–72. <http://dx.doi.org/10.1007/s00466-008-0255-5>. – DOI 10.1007/s00466-008-0255-5
- [KW09] KÜTTLER, U. ; WALL, W.: Vector extrapolation for strong coupling fluid-structure interaction solvers. In: *Journal of Applied Mechanics*. 76 (2009). <http://dx.doi.org/10.1115/1.3057468>. – DOI 10.1115/1.3057468
- [LF96] LESOINNE, M. ; FARHAT, C.: Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. In: *Computer Methods in Applied Mechanics and Engineering* 134 (1996), Nr. 1-2, S. 71–90
- [LG95] LIU, M. ; GORMAN, D.G.: Formulation of Rayleigh damping and its extensions. In: *Computers and Structures* 57 (1995), Nr. 2, S. 277–285
- [Lin09] LINHARD, J.: *Numerisch-mechanische Betrachtung des Entwurfsprozesses von Membrantragwerken*, Lehrstuhl für Statik der Technischen Universität München, Diss., 2009
- [LTM01] LE TALLEC, P. ; MOURO, J.: Fluid structure interaction with large structural displacements. In: *Computer Methods in Applied Mechanics and Engineering* 190 (2001), Nr. 24-25, S. 3039–3067
- [LY96] LÖHNER, R. ; YANG, C.: Improved ALE mesh velocities for moving bodies. In: *Communications in Numerical Methods in Engineering* 12 (1996), Nr. 10, S. 599–608
- [LY08] LUENBERGER, D.G. ; YE, Y.: *Linear and nonlinear programming*. Springer Verlag, 2008
- [LYC+95] LÖHNER, R. ; YANG, C. ; CEBRAL, J. ; BAUM, J.D. ; LUO, H. ; PELESSONE, D. ; CHARMAN, C.: Fluid-structure interaction using a loose coupling algorithm and adaptive unstructured grids. In: *AIAA Paper* 95 (1995), S. 2259
- [Man98] MANN, Jakob: Wind field simulation. In: *Prob. Engng. Mech.* 13/4 (1998), S. 269–282
- [ME01] MENTER, F. ; ESCH, T.: Elements of Industrial Heat Transfer Predictions. In: *16th Brazilian Congress of Mechanical Engineering (COBEM)*, 2001, S. 26–30

- [Mei05] MEISTER, A.: *Numerik linearer Gleichungssysteme: Eine Einführung in moderne Verfahren*. Vieweg, 2005
- [Men98] MENCK, J.: An Approximate Newton-Like Coupling of Subsystems / Technische Universität Hamburg-Harburg. 1998. – Forschungsbericht
- [Men99] MENCK, J.: Work Control for Newton Type Coupling / Technische Universität Hamburg-Harburg. 1999. – Forschungsbericht
- [Mic05] MICHLER, C.: *Efficient Numerical Methods for Fluid-Structure Interaction*, Technische Universität Delft, Diss., 2005
- [Mic10] MICHALSKI, A.: *Modellierung atmosphärischer Turbulenzen zur numerischen Simulation leichter Flächentragwerke*, Lehrstuhl für Statik der TU München, Diss., 2010
- [Mil91] MILLER, G.R.: An Object-Oriented Approach to Structural Analysis and Design. In: *Computers and Structures* 40 (1991), S. 75–82
- [MKL03] MENTER, F.R. ; KUNTZ, M. ; LANGTRY, R.: Ten Years of Industrial Experience with the SST Turbulence Model. In: *Turbulence, Heat and Mass Transfer* 4 (2003), S. 625–632
- [MMV99] MACKENS, W. ; MENCK, J. ; VOSS, H.: Coupling Iterative Subsystem Solvers / Technische Universität Hamburg-Harburg. 1999. – Forschungsbericht
- [MNS06] MATTHIES, H. G. ; NIEKAMP, R. ; STEINDORF, J.: Algorithms for strong coupling procedures. 195 (2006), S. 2028–2049
- [Mok01] MOK, D. P.: *Partitionierte Lösungsansätze in der Strukturdynamik und der Fluid-Struktur-Interaktion*, Institut für Baustatik der Universität Stuttgart, Diss., 2001
- [MS02] MATTHIES, H.G. ; STEINDORF, J.: Partitioned but strongly coupled iteration schemes for nonlinear fluid–structure interaction. In: *Computers and Structures* 80 (2002), Nr. 27-30, S. 1991–1999
- [MS03] MATTHIES, H.G. ; STEINDORF, J.: Partitioned strong coupling algorithms for fluid-structure interaction. In: *Computers and Structures* 81 (2003), Nr. 8-11, S. 805–812
- [MVBDB05] MICHLER, C. ; VAN BRUMMELEN, E.H. ; DE BORST, R.: An interface Newton-Krylov solver for fluid-structure interaction. In: *International Journal for Numerical Methods in Fluids* 47 (2005), Nr. 10-11, S. 1189–1195
- [New59] NEWMARK, N.M.: A method of computation for structural dynamics. In: *Journal of the Engineering Mechanics Division* 85 (1959), Nr. 67-94, S. 100

- [Nob01] NOBILE, F.: *Numerical approximation of fluid-structure interaction problems with application to haemodynamics*, Ecole Polytechnique Fédérale de Lausanne, Switzerland, Diss., 2001
- [NV08] NOBILE, F. ; VERGARA, C.: An effective fluid-structure interaction formulation for vascular dynamics by generalized Robin conditions. In: *SIAM Journal on Scientific Computing* 30 (2008), Nr. 2, S. 731–763
- [OF01] OHAYON, R. ; FELIPPA, C.: Special Issue on Fluid-Structure Interactions. In: *Comput. Methods Appl. Mech. Engrg* 190 (2001), S. 2977–3292
- [OK06] OHAYON, R. ; KVAMSDAL, T.: Special Issue on Fluid-Structure Interactions. In: *Comput. Methods Appl. Mech. Engrg* 195 (2006)
- [ope] www.openfoam.org
- [PF01] PIPERNO, S. ; FARHAT, C.: Partitioned procedures for the transient solution of coupled aeroelastic problems—Part II: energy transfer analysis and three-dimensional applications. In: *Computer methods in applied mechanics and engineering* 190 (2001), Nr. 24-25, S. 3147–3170
- [Pip97] PIPERNO, S.: Explicit/implicit fluid/structure staggered procedures with a structural predictor and fluid subcycling for 2D inviscid aeroelastic simulations. In: *International journal for numerical methods in fluids* 25 (1997), Nr. 10, S. 1207–1226
- [PT02] PERNICE, M. ; TOCCI, M.D.: A multigrid-preconditioned Newton-Krylov method for the incompressible Navier-Stokes equations. In: *SIAM Journal on Scientific Computing* 23 (2002), Nr. 2, S. 398–418
- [RH93] RICHARDS, P.J. ; HOXEY, R.P.: Appropriate boundary conditions for computational wind engineering models using the k-epsilon turbulence model. In: *Journal of Wind Engineering and Industrial Aerodynamics* 46 (1993), S. 145–153
- [RH07] RICHARDS, P. J. ; HOXEY, R. P.: Wind loads on the roof of a 6m cube. In: *Journal of Wind Engineering and Industrial Aerodynamics* 96 (2007), S. 984–993
- [RKK87] ROGERS, S.E. ; KWAK, D. ; KAUL, U.: On the accuracy of the pseudocompressibility method in solving the incompressible Navier-Stokes equations. In: *Applied Mathematical Modelling* 11 (1987), Nr. 1, S. 35–44
- [RRLJ01] RABACK, P. ; RUOKOLAINEN, J. ; LYLY, M. ; JARVINEN, E.: Fluid-structure interaction boundary conditions by artificial compressibility. In: *ECCOMAS Computational Fluid Dynamics Conference, Swansea, UK*, 2001

- [RSFW09] *Kapitel Interaction of Incompressible Flows and Thin-Walled Structures.* In: RAMM, E. ; SCHEVEN, M. von ; FÖRSTER, C. ; WALL, W. A.: *ECCOMAS Multidisciplinary Jubilee Symposium. New Computational Challenges in Materials, Structures, and Fluids.* Springer, 2009, S. 219–233
- [RVD00] RIEMSLAGH, K. ; VIERENDEELS, J. ; DICK, E.: An efficient coupling procedure for flexible wall fluid-structure interaction. In: *Eccomas Congress on Comp. Meth. in Appl. Sci. and Eng.* Barcelona, 2000
- [Saa03] SAAD, Y.: *Iterative Methods for Sparse Linear Systems.* Society for Industrial Mathematics, 2003
- [Sch99] SCHÄFER, M.: *Numerik im Maschinenbau.* Springer, 1999
- [Sch09] SCHEVEN, v.M.: *Effiziente Algorithmen für die Fluid-Struktur-Wechselwirkung,* Institut für Baustatik und Baudynamik der Universität Stuttgart, Diss., 2009
- [Sid91] SIDI, A.: Efficient implementation of minimal polynomial and reduced rank extrapolation methods. In: *Journal of Computational and Applied Mathematics* 36 (1991), Nr. 3, S. 305–337
- [Sid03] SIDI, A.: *Practical extrapolation methods: theory and applications.* Cambridge University Press, 2003
- [Sni98] SNIR, M.: *MPI—the complete reference.* The MIT Press Massachusetts, 1998
- [SS86] SAAD, Y. ; SCHULTZ, M.H.: GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. In: *SIAM J. Sci. Stat. Comput.* 7 (1986), Nr. 3, S. 856–869
- [SS96] SIMIU, E. ; SCANLAN, R.H.: *Wind effects on structures: fundamentals and applications to design.* John Wiley, New York, 1996
- [STD⁺96] SCHÄFER, M. ; TUREK, S. ; DURST, F. ; KRAUSE, E. ; RANNACHER, R.: Benchmark computations of laminar flow around a cylinder. In: *Notes on numerical fluid mechanics* 52 (1996), S. 547–566
- [Ste02] STEINDORF, J.: *Partitionierte Verfahren für Probleme der Fluid-Struktur Wechselwirkung,* Technische Universität Braunschweig, Diss., 2002
- [Ste05] STEINBACH, O.: *Lösungsverfahren für lineare Gleichungssysteme: Algorithmen und Anwendungen.* Vieweg+Teubner Verlag, 2005
- [STRB07] SCHEVEN, v.M. ; TIYYAGURA, S.R. ; RAMM, E. ; BISCHOFF, M.: Efficiency Issues of Partitioned Solution in Fluid-Structure Interaction. In: *Proc. of International Conf. on Computational Methods for Coupled Problems in Science and Engineering,* 2007

- [TBU00] THÉVENAZ, P. ; BLU, T. ; UNSER, M.: Interpolation revisited. In: *IEEE Trans. Med. Imaging* 19 (2000), Nr. 7, S. 739–758
- [TH06] TUREK, S. ; HRON, J.: Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow. In: *Lecture Notes in Computational Science and Engineering* 53 (2006), S. 371–385
- [THR⁺10] TUREK, S. ; HRON, J. ; RAZZAQ, M. ; WOBKER, H. ; SCHÄFER, M.: Numerical Benchmarking of Fluid-Structure Interaction: A comparison of different discretization and solution approaches. In: [BS10]
- [TJ08] TUKOVIC, Z. ; JASAK, H.: Simulation of Free-Rising Bubble with Soluble Surfactant using Moving Mesh Finite Volume/Area Method. In: SINTEF/NTNU (Hrsg.): *6th International Conference on CFD in Oil & Gas, Metallurgical and Process Industries*. Trondheim, Norway, 2008
- [Vie06] VIERENDEELS, J.: Implicit coupling of partitioned fluid-structure interaction solvers using reduced-order models. In: *Lecture Notes in Computational Science and Engineering* 53 (2006), S. 1
- [VLDV07] VIERENDEELS, J. ; LANOYE, L. ; DEGROOTE, J. ; VERDONCK, P.: Implicit coupling of partitioned fluid-structure interaction problems with reduced order models. In: *Computers and Structures* 85 (2007), Nr. 11-14, S. 970–976
- [Wal99] WALL, W.A.: *Fluid-Struktur-Interaktion mit stabilisierten Finiten Elementen*, Institut für Baustatik der Universität Stuttgart, Diss., 1999
- [WGG⁺06] WALL, W.A. ; GERSTENBERGER, A. ; GAMNITZER, P. ; FÖRSTER, C. ; RAMM, E.: *Fluid-Structure Interaction: Modelling, Simulation, Optimisation. Lecture Notes in Computational Science and Engineering (53)*. Springer Berlin, 2006
- [Wil06] WILCOX, D.C.: *Turbulence modeling for CFD*. La Canada, California : DCW industries, 2006
- [WKB07] WÜCHNER, R. ; KUPZOK, A. ; BLETZINGER, K.-U.: A framework for stabilized partitioned analysis of thin membrane- wind interaction. In: *International Journal for Numerical Methods in Fluids* 54 (2007), S. 945–963
- [Woo90] WOOD, W.L.: *Practical time-stepping schemes*. Oxford University Press, USA, 1990
- [Wri01] WRIGGERS, P.: *Nichtlineare Finite-Element-Methoden*. Birkhäuser, 2001
- [WTJF98] WELLER, H.G. ; TABOR, G. ; JASAK, H. ; FUREBY, C.: A Tensorial Approach to CFD using Object Oriented Techniques. In: *Computers in Physics* 12(6) (1998), S. 620–631

- [ZB10] ZUIJLEN, A.H. van ; BIJL, H.: Multi-level accelerated sub-iterations for fluid-structure interaction. In: [BS10]
- [ZBB07] ZUIJLEN, A.H. van ; BOSSCHER, S. ; BIJL, H.: Two level algorithms for partitioned fluid-structure interaction computations. In: *Computer Methods in Applied Mechanics and Engineering* 196 (2007), Nr. 8, S. 1458–1470
- [ZH04] ZHANG, Q. ; HISADA, T.: Studies of the strong coupling and weak coupling methods in FSI analysis. In: *International Journal for Numerical Methods in Engineering* 60 (2004), Nr. 12, S. 2013–2029
- [ZT05] ZIENKIEWICZ, O.C. ; TAYLOR, R.L.: *The finite element method for solid and structural mechanics*. Butterworth-Heinemann, 2005