Ein Splitting-Algorithmus höherer Ordnung für die Navier-Stokes-Gleichung auf der Basis der Finite-Element-Methode

Dissertation

zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt beim

Fachbereich Mathematik

der Universität Duisburg-Essen

von

Jörg Frochte
aus Essen

Tag der mündlichen Prüfung: 07.12.2005

Vorsitzender:	Prof. Dr. rer. nat. Karl-Josef Witsch
Erster Gutachter:	Prof. Dr. rer. nat. Wilhelm Heinrichs
Zweiter Gutachter:	Prof. Dr. rer. nat. Gerhard Starke (Universität Hannover)

Inhaltsverzeichnis

Bezeichnungen

Ω	offene Menge im $\mathbb{R}^d$
ϱ	Dichte eines Fluids
$\hat{p}$	Druck
p	$= \frac{1}{\varrho}\hat{p}$
∇^2	Laplace-Operator
v	das Vektorfeld, $\Omega \to \mathbb{R}^d$ der Geschwindigkeit eines Fluids
η	Viskositätskoeffizient eines Fluids
ν	spezifische Viskosität
$L^n_{t,k}$	Operator, der mit Hilfe eines BDF-Schemas $\frac{\partial}{\partial t}$ approximiert
$\triangle t$	$\in \mathbb{R}^+$ die Schrittweite in der Zeit t
$L^2(\Omega)$	Raum der über Ω quadrat-integrierbaren Funktionen
$L_{2,0}(\Omega)$	$:= \{q \in L_2(\Omega); \int_\Omega q\ dx = 0\}$
$H^m(\Omega)$	Sobolev-Raum von L^2-Funktionen mit quadrat-integrierbaren Ableitungen bis zur Ordnung m
$H^m_0(\Omega)$	Untermenge von $H^m(\Omega)$ der Funktionen mit verallgemeinerten Nullrandbedingungen
$C^m(\Omega)$	Menge der Funktionen mit stetigen Ableitungen der Ordnung m über Ω
V	Hilbert-Raum
V_h	endlich dimensionaler Finite-Element-Raum mit $V_h \subset V$
P	Prolongationsoperator
P^t	Restriktionsoperator
R_h	Ritz-Projektion
I_h	Interpolationsoperator nach V_h
I	Identität
G	Gradientenrekonstruktion
$\mathcal{T}_h$	Triangulierung
$\mathcal{I}_h$	Menge der Knoten einer Triangulierung $\mathcal{T}_h$
$\Vert.\Vert_{\max}$	knotenbezogene Maximumsnorm ; $\Vert u\Vert_{\max} = \max_{i\in\mathcal{I}_h}\{\ \vert u(x_i,y_i)\vert\ \}$

1 Einleitung

Richard P. Feynman, Nobelpreisträger für Physik des Jahres 1965, schreibt in seinem Buch [14], S.786:

> *Das Thema der Flüssigkeitsströmung, besonders der von Wasser, fasziniert jeden. Wir können uns alle daran erinnern, daß wir als Kinder in der Badewanne oder in Schmutzpfützen mit diesem merkwürdigen Stoff gespielt haben. Als wir dann älter wurden, haben wir Flüsse, Wasserfälle und Strudel beobachtet und waren von dieser Substanz fasziniert, die im Vergleich zu Festkörpern fast lebendig zu sein schien. Das Verhalten der Flüssigkeiten ist in vieler Hinsicht unerwartet und interessant [...]*

Dieses Verhalten wird in der Strömungsmechanik untersucht, in der eine der wichtigsten Gleichungen die Navier-Stokes-Gleichung ist. Sie beschreibt das Verhalten von Strömungen in Flüssigkeiten und Gasgemischen. Bei der Navier-Stokes-Gleichung handelt es sich um ein System von nichtlinearen partiellen Differentialgleichungen zweiter Ordnung. Der Franzose Claude Louis Marie Henri Navier und der Brite George Gabriel Stokes entdeckten unabhängig voneinander in der ersten Hälfte des 19. Jahrhunderts den der Gleichung zu Grunde liegenden Impulssatz für Newton'sche Fluide, wie z.B. Wasser, Honig oder Luft.

Die analytische Durchdringung der Navier-Stokes-Gleichung ist noch nicht abgeschlossen und gehört laut dem Clay Mathematics Institute zu den wichtigsten ungelösten mathematischen Problemen dieses Jahrhunderts.

Zur Lösung der Navier-Stokes-Gleichung stützt man sich daher fast ausschließlich auf numerische Näherungsverfahren, welche auf verschiedene Weise ihre Anwendung in den Natur- und Ingenieurwissenschaften finden. In Kapitel 8.2 wird als Beispiel für eine solche anwendungsbezogene Fragestellung der „Fluss um einen Zylinder“ untersucht. Hier werden Fragestellungen behandelt, welche zum Beispiel bei Experimenten im Windkanal auftreten.

Um Fragestellungen aus den Ingenieurwissenschaften numerisch lösen zu können, sind dringend effiziente numerische Algorithmen erforderlich. Wichtig ist dabei die Robustheit der Algorithmen und die Möglichkeit, die exakte Lösung von solch komplexen Anwendungsproblemen hinreichend genau zu approximieren.

Als Methode zur Lösung der auftretenden partiellen Differentialgleichungen werden in dieser Arbeit die finiten Elemente verwendet. Dabei handelt es sich um eine so genannte lokale Methode, welcher eine weit entwickelte mathematische Theorie zugrunde liegt.
Im Gegensatz zur Methode der Spektralen Verfahren eignet sich die Finite-Element-Methode besonders gut für Problemstellungen auf komplexen Geometrien. Die Spektralen Verfahren verwenden zur Lösung der partiellen Differentialgleichungen globale Ansatzfunktionen. Auf einer einfachen Geometrie wie dem Quadrat erreicht man mit diesem Ansatz sehr hohe

Genauigkeiten, falls die Lösung sehr glatt ist.

Mit Hilfe dieser Spektralen Verfahren wurde von Haschke und Heinrichs in den Veröffentlichungen [21] und [22] ein Operatorsplitting zweiter Ordnung in der Zeit für die Navier-Stokes-Gleichung vorgestellt. Bei einem solchen Operatorsplitting ist das Ziel die Entkopplung von Druck und Geschwindigkeit durch aufeinander folgende Lösung zweier oder mehrerer Gleichungen. Die Rechenkosten zur Lösung der so entstandenen kleineren Systeme sind geringer als diejenigen, die zur Lösung des gekoppelten Problems nötig wären.

Gegenstand der vorliegenden Arbeit ist die Entwicklung eines solchen Operatorsplittings für die Finite-Element-Methode ausgehend von den Erfahrungen, die bereits bei der Arbeit mit Spektralen Verfahren gemacht wurden.

In Kapitel 2 wird sowohl der physikalische als auch der mathematische Hintergrund der Navier-Stokes-Gleichung dargestellt. Hiermit werden Anforderungen anwendungsbezogener Fragestellungen an die numerischen Verfahren deutlich und die Wahl von Randbedingungen wird hierdurch motiviert.

Kapitel 3 stellt verschiedene Splittingverfahren für die Navier-Stokes-Gleichung vor, so auch das von Haschke und Heinrichs.

In Kapitel 4 wird eine Einführung in die Finite-Element-Methode zur Lösung partieller Differentialgleichungen gegeben. Dabei wird auf die Stromliniendiffusionsmethode als Stabilisierungstechnik für die finiten Elemente eingegangen. Die dargelegte Theorie der Sattelpunktprobleme motiviert die Wahl der für das Splitting verwendeten Taylor-Hood-Elemente. Zur Zeitdiskretisierung werden BDF-Verfahren (Backward Differentiation Formulas) verwendet und für konstante Zeitschrittweiten in Abschnitt 4.3 vorgestellt. Auf ihre Anpassung an adaptiv gewählte Zeitschrittweiten und die dabei auftretenden Stabilitätsbeschränkungen geht in Kapitel 7 der Abschnitt 7.1.1 ein.

Das grundlegende Operatorsplitting auf der Basis der Finite-Element-Methode wird in Kapitel 5 entwickelt. Hierzu wird zunächst in Abschnitt 5.1 ein neues Verfahren zur Gradientenrekonstruktion vorgestellt. Mit Hilfe eines solchen Verfahrens kann der Gradient einer im Finite-Element-Raum dargestellten Funktion in höherer Ordnung approximiert werden. Ein bekanntes Verfahren zur Gradientenrekonstruktion ist zum Beispiel das so genannte Z^2-Verfahren von Zienkiewicz und Zhu aus dem Artikel [45]. Diese Verfahren verlieren in der Regel am Rand eines Gebietes stark an Genauigkeit. Jedoch ist auch dort eine hohe Genauigkeit zur Gewinnung von Randwerten für das Splitting nötig. In Abschnitt 5.1.3 wird ein neues Verfahren zur Gradientenrekonstruktion vorgestellt, das in dieser Hinsicht den bekannten Verfahren überlegen ist. Auch konnte mit Hilfe dieses Verfahrens die Ordnung der Gradientenrekonstruktion weiter erhöht werden.

Anschließend wird in Abschnitt 5.2 der Splittingalgorithmus für die Finite-Element-Methode vorgestellt. Um einen stabilen Splittingalgorithmus zu erhalten, mussten einige Modifikationen an dem ursprünglichen Algorithmus von Heinrichs und Haschke vorgenommen werden. Dem schließt sich in Abschnitt 5.4 eine Analyse des Splittingalgorithmus an. Das dort postulierte Verfahren zweiter Ordnung in der Zeit konnte in den numerischen Tests

in Abschnitt 5.5 bestätigt werden. Wesentlich hierfür war unter anderem die Bestimmung geeigneter Neumann-Randwerte für den Druck.
Kapitel 5 zeigt Möglichkeiten zur Behandlung des konvektiven Anteils $(v \cdot \nabla)v$. Zunächst sind zwei Strategien denkbar, eine explizite und eine implizite Behandlung. Bei der expliziten wurde in dieser Arbeit auf eine Picard-Iteration statt der sonst oft üblichen Approximation mit einer Adams-Bashforth-Formel gesetzt, was allerdings weniger kostengünstig ist. Demgegenüber steht jedoch die Hoffnung, dass auf diese Weise größere Zeitschrittweiten möglich werden. Mit Hilfe der expliziten Strategie ist es möglich, nichtsymmetrische Gleichungssysteme zu vermeiden, wodurch die Lösung der auftretenden großen dünnbesetzten Gleichungssysteme wesentlich erleichtert wird. Auch ist im Fall einer konstanten Zeitschrittweite nur ein einmaliges Assemblieren der Galerkinmatrix nötig. Da sich diese Matrix im Laufe des Splittings nicht ändert, werden auch teurere Vorkonditionierer hier sehr interessant. Für niedrige Reynoldszahlen steht damit ein sehr günstiges Verfahren zur Verfügung.
Bei höheren Reynoldszahlen ist eine implizite Behandlung des nichtlinearen Terms erforderlich. Diese implizite Strategie hat sich als äußerst robust sowohl für weniger glatte Lösungen als auch für hohe Reynoldszahlen herausgestellt. Zusammen mit der in Kapitel 7.1 vorgestellten adaptiven Wahl der Zeitschrittweite hat diese Variante das Potenzial zu einem sehr robusten Black-Box-Verfahren.

Die adaptive Wahl der Zeitschrittweite kann zusammen mit den Strategien zum Postprocessing bzw. Preconditioning aus Kapitel 6 besonders günstig realisiert werden. Hierzu wird die adaptive Wahl der Zeitschrittweite mittels des kleineren Finite-Element-Ansatzraumes vorgenommen und führt so zu geringen zusätzlichen Rechenkosten.

Mit Hilfe der Strategien aus Kapitel 6 verbessert sich die Ordnung des Verfahrens. Hiermit erreicht das Splitting höhere Reduktionsraten als die zweite Ordnung in der Zeit. Gleichzeitig dient das Verfahren als Preconditioning und führt zu einer signifikanten Beschleunigung des Splittings. Die in Tabelle 6.4 aufgezeigte Wirkung des Preconditioning würde sich beim Einsatz iterativer Löser ggf. weiter steigern lassen.

Eine adaptiv angepasste Wahl der Zeitschrittweite ist wesentlich, um auf der Basis des Splittings ein robustes Verfahren zur Lösung der Navier-Stokes-Gleichung zu erhalten. Mit Hilfe der adaptiv gewählten Zeitschrittweiten ist das Splittingverfahren in der Lage, auch Unstetigkeiten in der ersten zeitlichen Ableitung der Lösung zu meistern.

Die adaptive Anpassung des Gitters, welche in 7.2 vorgestellt wird, ermöglicht es, Layer aufzulösen und problematische Bereiche eines Gitters zu verfeinern. Es hat sich jedoch herausgestellt, dass Gitteranpassungen negative Konsequenzen auf den approximierten Druck haben, die in Abschnitt 7.2.3 untersucht werden.

In Kapitel 8.1 wird das Splitting an Hand des Driven Cavity Problems getestet. Es hat sich gezeigt, dass das Splitting den dabei auftretenden Singularitäten gewachsen ist. Hierbei konnte auf ein Herausrechnen der Singularitäten, wie es beim Einsatz Spektraler Verfahren üblich ist, verzichtet werden.

Zuletzt wird in Kapitel 8.2 das aus dem DFG-Forschungsprogramm “Flow Simulation on High Performance Computers” stammende Problem „Fluss um einen Zylinder“ behandelt.

Das Problem ist von besonderem Interesse, da es Fragen aus den Ingenieurwissenschaften aufgreift. Hier kommt besonders die Flexibilität der Finite-Element-Methode in Hinblick auf die zu approximierende Geometrie zum Tragen.
Wie in Kapitel 8 deutlich wird, kann das vorgestellte Splittingverfahren auf Standardproblemen der numerischen Strömungsmechanik effektiv und mit hoher Genauigkeit eingesetzt werden.

Danksagung

An dieser Stelle möchte ich mich bei den Menschen bedanken, die mir während meiner Zeit als Doktorand, insbesondere beim Erstellen dieser Arbeit hilfreich zur Seite gestanden haben.

Für die Anregung zu dieser Doktorarbeit möchte ich mich zu allererst bei meinem Doktorvater Prof. Dr. Wilhelm Heinrichs bedanken. In den zurückliegenden Jahren habe ich als Mitglied der Arbeitsgruppe Ingenieurmathematik von ihm jede mögliche Förderung erfahren, insbesondere viel Freiheit zum Ausprobieren meiner Ideen.

Erwähnen möchte ich auch Prof. Dr. Gerhard Starke, der mir in seinen Vorlesungen die numerische Mathematik näher gebracht hat und sich nun bereiterklärt hat, das Koreferat für diese Arbeit zu übernehmen.

Kollegiale Hilfe, die man nicht für selbstverständlich halten darf, habe ich unter anderem von Seiten der Arbeitsgruppe Klawonn erfahren, die mir den Opteron-Cluster für meine Berechnungen überlassen hat. Prof. Dr. John von der Universität des Saarlandes war so freundlich, mir die Dateien mit seinen Benchmarkdaten zum „Fluss um einen Zylinder“-Problem zu überlassen, wodurch ein besserer Vergleich der erzielten Ergebnisse möglich wurde. Vielen Dank auch an Dr. Heike Haschke, die mir zu Beginn meiner Arbeit eine Einführung in das Gebiet gab.

Ein Dankeschön ebenfalls an viele Mitglieder des Fachbereichs Mathematik am Campus Essen für anregende Diskussionen, wertvolle Tipps und geduldiges Zuhören.

Dasselbe gilt für Dipl.-Phys. Jan van Heys und Dipl.-Phys. Marc Stroisch, mit denen ich oft über die angewandten Seiten meiner Arbeit diskutiert habe. Ihnen ist es neben Dr. Sebastian Bauer und Barbara Fröhlich, welche die Hauptlast des Korrekturlesens auf sich genommen hat, zu verdanken, dass einige Kommata, Betragsstriche und Ähnliches doch noch den ihnen gebührenden Platz gefunden haben.

Für den menschlichen Rückhalt während der gesamten Promotion möchte ich mich bei meinen Eltern und meiner Freundin Barbara bedanken, ohne die vieles sicher bedeutend schwieriger geworden wäre.

2 Der Hintergrund der Navier-Stokes-Gleichung

Zur physikalischen Herleitung der Navier-Stokes-Gleichung betrachten wir das Verhalten eines Fluids, das durch ein Gebiet $\Omega \subset \mathbb{R}^d$ mit $d = 1, 2$ oder 3 strömt. Fluid ist ein Oberbegriff, der sowohl Flüssigkeiten als auch Gase umfasst.

2.1 Charakteristische Größen eines Fluids

Das Verhalten eines Fluids wird von mehreren Größen beeinflusst, wird die Navier-Stokes-Gleichung als Modell verwendet, so ist es zum Beispiel unabhängig von der Temperatur. Die zwei skalaren Größen, welche im Modell der Navier-Stokes-Gleichung ein Fluid charakterisieren, sind zum einen die *Viskosität* und zum anderen die *Dichte*.

Viskosität

Die Viskosität einer Flüssigkeit charakterisiert Richard P. Feynman in [14], S. 786 wie folgt:

> *Eine Flüssigkeit unterscheidet sich von einem Festkörper hauptsächlich darin, daß eine Flüssigkeit eine Scherspannung nicht aufrechterhalten kann, und sei es für noch so kurze Zeiten. Übt man eine Scherung auf eine Flüssigkeit aus, so bewegt sie sich unter ihrem Einfluß. Dickere Flüssigkeiten wie Honig bewegen sich weniger leicht als Flüssigkeiten wie Luft oder Wasser. Das Maß der Leichtigkeit, mit der eine Flüssigkeit nachgibt, ist ihre Viskosität.*

Der Begriff Scherspannung wird z.B. in [44], S.93 und S.131 weiter erläutert.

Dichte

Die Dichte oder spezifische Masse eines Stoffes ist, wie man zum Beispiel in [44], S.13 nachlesen kann, durch

$$\varrho = \frac{\text{Masse}}{\text{Volumen}}$$

gegeben und wird in kg/m^3 oder g/cm^3 ausgedrückt.

Druck

Eine weitere wesentliche physikalische Größe, die für das Verstehen des Verhaltens notwendig ist, ist der *Druck*, welcher in $\frac{\text{Newton}}{\text{m}^2}$[1] angegeben wird. Der Druck in einem Fluid ist eine orts- und ggf. zeitabhängige skalare Größe $\hat{p}(x, y, z, t)$. Auf das Fluid wirkt durch den Druck laut [14], S.787 eine Kraft von $-\nabla\hat{p}$ ($\frac{\text{Newton}}{\text{m}^3}$).

---

[1] Andere Maßeinheiten sind 1 Pascal = 10^{-5} bar = $1\frac{\text{Newton}}{\text{m}^2}$.

2.2 Die Bewegungsgleichung

Beim Aufstellen der Bewegungsgleichung werden wir zunächst die Viskosität der Flüssigkeit vernachlässigen. Um die Bewegung einer Flüssigkeit beschreiben zu können, müssen wir ihre Eigenschaften in jedem Punkt angeben können. Zum Beispiel bewegt sich die Flüssigkeit abhängig vom Ort mit unterschiedlicher *Geschwindigkeit.* Die Geschwindigkeit ist eine vektorielle Größe $v : \Omega \to \mathbb{R}^d$, deren d Komponenten wir jeweils bestimmen müssen. Das physikalische Modell, welches wir betrachten werden, wird dadurch weiter vereinfacht, dass man die Dichte als Konstante auffassen kann. Dadurch wird das Fluid als inkompressibel angenommen. Diese Näherung ist oft sehr gut, besonders wenn man davon ausgehen kann, dass die Strömungsgeschwindigkeit sehr viel kleiner ist als die Schallgeschwindigkeit. Vergleiche dazu auch [14], S.789 und S.805.
Damit folgt aus der hydrodynamischen *Kontinuitätsgleichung*

$$\nabla \cdot (\varrho v) = -\frac{\partial \varrho}{\partial t}$$

direkt

$$\nabla \cdot v = 0 \ .$$

Um nun die Bewegungsgleichung herzuleiten, betrachtet man zunächst das Newton'sche Gesetz (Kraft = Masse · Beschleunigung) für die Änderung einer Geschwindigkeit durch einwirkende Kräfte

$$\varrho \cdot \text{Beschleunigung} = \hat{f} \ .$$

$\hat{f}$ ist dabei die Summe der Kräfte pro Volumen (Newton pro m^3), dazu gehören u.a. auch die Kraft, die aus einer Druckdifferenz $-\nabla \hat{p}$ entsteht, sonstige äußere Kräfte f_{ext} und die von uns bis jetzt vernachlässigte Kraft auf Grund der Viskosität der Flüssigkeit f_{visk}. Damit erhält man

$$\varrho \cdot \text{Beschleunigung} = -\nabla \hat{p} + f_{ext} + f_{visk} \ .$$

Da der Einfluss von f_{visk} zunächst nicht betrachtet werden soll, bleibt noch die Beschleunigung zu bestimmen. Um diesen Ausdruck zu beschreiben, muss man sich zunächst verdeutlichen, dass die Beschleunigung nicht, wie man annehmen könnte, einfach $\frac{\partial v}{\partial t}$ ist. Die Ableitung $\frac{\partial v}{\partial t}$ ist die Rate, mit der sich die Geschwindigkeit $v(x, y, z, t)$ an einem festen Ort in Ω ändert. Was wir jedoch brauchen, ist die Geschwindigkeit für einen bestimmten Teil der Flüssigkeit. Man stelle sich hier einfach einen gefärbten Tropfen Wasser vor, dessen Bewegung wir verfolgen.

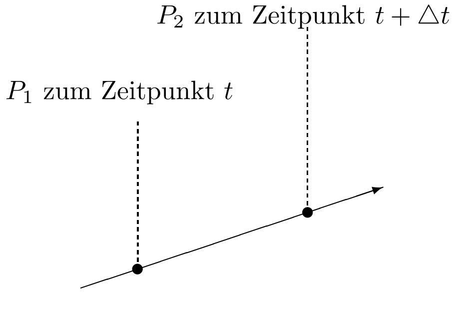


Abbildung 2.1: Skizze der Bewegung eines Flüssigkeitsvolumens in der Strömung

Dieser Tropfen wird in einem Zeitintervall Δt an einen anderen Ort transportiert, z.B. vom Punkt P_1 zum Punkt P_2. Er legt also in den einzelnen Dimensionen den Weg $v_1\Delta t$, $v_2\Delta t$ und $v_3\Delta t$ zurück. Sei $v(x,y,z,t)$ die Geschwindigkeit des Tropfens, der sich zur Zeit t am Punkt (x,y,z) befindet, so wird klar, dass die Geschwindigkeit desselben Teilchens dann zur Zeit $t+\Delta t$ durch $v(x+v_1\Delta t, y+v_2\Delta t, z+v_3\Delta t, t+\Delta t)$ gegeben ist. Damit gilt in erster Ordnung

$$\begin{aligned}&v(x+v_1\Delta t, y+v_2\Delta t, z+v_3\Delta t, t+\Delta t)\\ &= v(x,y,z,t) + \frac{\partial v}{\partial x}v_1\Delta t + \frac{\partial v}{\partial y}v_2\Delta t + \frac{\partial v}{\partial z}v_3\Delta t + \frac{\partial v}{\partial t}\Delta t \ .\end{aligned}$$

Die Beschleunigung $\frac{\Delta v}{\Delta t}$ ist folglich gleich

$$v_1\frac{\partial v}{\partial x} + v_2\frac{\partial v}{\partial y} + v_3\frac{\partial v}{\partial z} + \frac{\partial v}{\partial t} \ ,$$

was sich mit Hilfe des Nabla-Operators wie folgt notieren lässt:

$$(v\cdot\nabla)v + \frac{\partial v}{\partial t} \ .$$

Die Bewegungsgleichung lässt sich nun folgendermaßen angeben:

$$(v\cdot\nabla)v + \frac{\partial v}{\partial t} = \frac{1}{\varrho}(-\nabla\hat{p} + f_{ext} + f_{visk}) \ .$$

Für f_{visk} gilt laut [14], S.805-810 für inkompressible Fluide

$$f_{visk} = \eta\nabla^2 v \ .$$

η wird dabei als *Viskositätskoeffizient* [2] bezeichnet. Die Einheit von η ist Newton · Sek/m^2. Es ist jedoch oft sinnvoller, die *spezifische Viskosität* (ν) oder kinematische Viskosität zu verwenden, die gleich $\frac{\eta}{\varrho}$ ist. Dadurch werden die Werte für verschiedene Flüssigkeiten vergleichbar. Einige typische Viskositätswerte können der folgenden Tabelle aus [28], S.55 entnommen werden:

	$\eta\ \frac{\text{kg}}{\text{m s}}$	$\varrho\ \frac{\text{kg}}{\text{m}^3}$	$\frac{\eta}{\varrho} = \nu\ \frac{\text{m}^2}{\text{s}}$
Wasser	$1.00\ 10^{-3}$	$1.00\ 10^{3}$	$1.0\ 10^{-6}$
Luft	$1.80\ 10^{-5}$	$1.20\ 10^{0}$	$1.5\ 10^{-5}$
Alkohol	$1.80\ 10^{-3}$	$0.81\ 10^{3}$	$2.2\ 10^{-6}$
Glyzerin	$8.50\ 10^{-1}$	$1.25\ 10^{3}$	$6.8\ 10^{-4}$
Quecksilber	$1.56\ 10^{-3}$	$1.18\ 10^{4}$	$1.3\ 10^{-7}$

Tabelle 2.1: Einige Stoffwerte bei 20°C im MKS-System

Die Viskosität hängt von der Temperatur ab, z.B. ist ν für Wasser knapp über dem

---

[2] Manchmal wird η auch als *erster Viskositätskoeffizient* bezeichnet, da f_{visk} für nicht-inkompressible Fluide noch einen Term mit einem weiteren Viskositätskoeffizient enthält.

Gefrierpunkt $1,8$-mal größer als bei 20°C. Die Werte aus Tabelle 2.1 machen deutlich, dass spezifische Viskositäten, für die $\nu < 10^{-3}$ gilt, besonders interessant sind.

Nun sind wir in der Lage, die vollständige Bewegungsgleichung zu notieren:

$$\frac{\partial v}{\partial t} + (v \cdot \nabla)v - \nu\nabla^2 v + \frac{1}{\varrho}\nabla\hat{p} = \frac{1}{\varrho} f_{ext} \, , \tag{2.1}$$

Dies ist die so genannte *Navier-Stokes-Gleichung*.

2.3 Klassifizierung von Strömungen

Als *laminare Strömung* bezeichnet man die Bewegung von Flüssigkeiten, bei der keine Turbulenzen auftreten. Die Flüssigkeit strömt in Schichten, die sich nicht vermischen. Im lateinischen Wort für Platte, lamina, liegt der Ursprung des Ausdrucks laminare Strömung.
Im Jahr 1883 unternahm der Physiker Osborne Reynolds einen Versuch zur Untersuchung von laminarer und turbulenter Strömung. Dabei stellte er fest, dass sich Turbulenzen in der von ihm verwendeten Rohrleitung erst ab einer gewissen Grenzgeschwindigkeit bildeten. Als Beurteilungskriterium wird hierzu die dimensionslose Größe

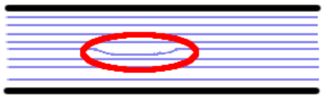


Abbildung 2.2: Skizze

$$Re = \frac{\|v\| d \varrho}{\eta}$$

angewandt, die Reynoldszahl oder auch Reynolds'sche Zahl genannt wird. Hierbei wird mit d die charakteristische Länge, im Falle des Rohres ist das der Durchmesser, bezeichnet. Über einem kritischen Wert Re_{krit} wird die laminare Strömung instabil gegenüber kleinen Störungen. Dieser Wert liegt z.B. bei der Rohrströmung laut [44], Seite 124 bei etwa 2000. In diesem Fall geht die geordnete Bewegung schließlich in die ungeordnete turbulente Strömung über. Die Skizze (2.2) stellt dieses Verhalten anschaulich dar. Man kann sich vorstellen, dass ein Fluid ein Rohr zunächst laminar mit parallelen Stromlinien durchfließt. Irgendwo treten dann kleine Störungen auf, die eine Stromlinie etwas nach oben oder unten verbiegen. Dadurch entstehen, wie in [44] auf S.117 beschrieben wird, Quellen für ein turbulentes Verhalten der Strömung. In einem dickeren Rohr geschieht dies statistisch häufiger als in einem dünnen.
Beispiele für laminare Strömungen findet man bei der Betrachtung des Grundwassers oder eines Blutkreislaufs. Weitere Ausführungen zum Thema laminare Strömungen, Reynoldszahl etc. findet sich z.B. in [17], [28] und [14].

2.4 Das Verhalten der physikalischen Größen am Rand

Um eindeutige Lösungen der Navier-Stokes-Gleichung berechnen zu können, müssen wir etwas über das Verhalten von v und p am Rand wissen.

An der Oberfläche eines unbewegten Festkörpers ist die Flüssigkeitsgeschwindigkeit gleich Null, vergleiche zum Beispiel [14], S.805f. Aus diesem Grund verwendet man dort homogene Null-Dirichlet-Randbedingungen für v. Diesen Effekt kann man im Alltag z.B. bei einem Ventilator beobachten, auf dessen Rotorblättern sich eine dünne Staubschicht ansammeln kann, die auch nach der Bewegung der Rotorblätter nicht verschwindet. Obwohl sich das Ventilatorblatt mit großer Geschwindigkeit durch die Luft bewegt, fällt die Luftgeschwindigkeit relativ zum Ventilatorblatt an der Oberfläche auf Null ab. Daher werden die kleinsten Staubteilchen nicht mit der Luft wegtransportiert. Erst, wenn ein Objekt groß genug ist, reicht es bis in die Luftströmung.

Wir können also für viele Modelle folgende Randbedingung an v stellen:

$$v = 0 \text{ auf } \partial\Omega\ .$$

Diese homogenen Dirichlet-Randbedingungen können variiert werden, um z.B. an einer Stelle des Randes eine einströmende Flüssigkeit zu modellieren.

Für den Druck p ist die Angabe von Neumann-Randbedingungen möglich. Nehmen wir an, die Viskositäts- und äußere Kraft seien am Rand vernachlässigbar, so gilt

$$\frac{\partial v}{\partial t} + (v \cdot \nabla)v + \frac{1}{\varrho}\nabla\hat{p} = 0 \text{ auf } \partial\Omega\ .$$

Setzt man voraus, dass wie oben geschrieben v am Rand konstant Null ist, verschwindet sowohl $\frac{\partial v}{\partial t}$ als auch der nichtlineare Term am Rand.

Damit ist

$$\frac{1}{\varrho}\nabla\hat{p} = 0 \text{ auf } \partial\Omega$$

und somit gelten mit dem Normalenvektor n die homogenen Randbedingungen

$$n \cdot \nabla\hat{p} = 0 \text{ auf } \partial\Omega\ .$$

Sollte man die Viskositäts- und äußere Kraft am Rand nicht vernachlässigen können, so erhält man entsprechend

$$n \cdot \nabla\hat{p} = n \cdot (f_{ext} + \varrho\nu\nabla^2 v) \text{ auf } \partial\Omega\ .$$

2.5 Existenz und Eindeutigkeit der Lösung der Navier-Stokes-Gleichung

Abschließend bleibt die Frage nach der Existenz und Eindeutigkeit der Lösung der Navier-Stokes-Gleichung zu klären. Es wurden mehrere Wege gefunden, die Existenz einer schwachen Lösung sowohl in zwei als auch in drei Raumdimensionen zu beweisen; so zum Beispiel mit dem Beweis aus den 80er Jahren von Sohr in [38].

Für die schwache Lösung in zwei Raumdimensionen konnte die so genannte Serrin's Bedingung aus [37] gezeigt werden, was deren Eindeutigkeit garantiert. In drei Raumdimensionen konnte die Eindeutigkeit noch nicht gezeigt werden.

Sowohl in zwei- als auch in drei Raumdimensionen gilt, dass existierende klassische Lösungen auch eindeutig sind.

2.6 Zusammenfassung

Fassen wir nun noch einmal die Erkenntnisse dieses Kapitels kurz zusammen. In der Bewegungsgleichung (2.1) kann man die Größen $\frac{1}{\varrho}\hat{p}$ und $\frac{1}{\varrho}f_{ext}$ durch entsprechend normierte Größen p und f ersetzen. Das Verhalten des Fluids und des Drucks am Rand des untersuchten Gebietes Ω wird durch geeignete Funktionen h und g modelliert. In den folgenden Kapiteln wird ein Verfahren entwickelt, mit dem die Lösung der *Navier-Stokes-Gleichung* für inkompressible Fluide, gegeben durch

$$\begin{aligned}
\frac{\partial v}{\partial t} + (v \cdot \nabla)v - \nu \nabla^2 v + \nabla p &= f \quad \text{in } \Omega,\ t \in [0, t_{end}] && (2.2)\\
\nabla \cdot v &= 0 \quad \text{in } \Omega,\ t \in [0, t_{end}] && (2.3)\\
v &= h \quad \text{auf } \partial\Omega,\ t \in [0, t_{end}] && (2.4)\\
v &= v_0 \quad \text{für } t = 0, \quad \text{in } \Omega && (2.5)\\
n \cdot \nabla p &= n \cdot g \quad \text{auf } \partial\Omega,\ t \in [0, t_{end}]\ , && (2.6)
\end{aligned}$$

numerisch approximiert werden kann. Dabei dienen die Funktionen h und g zur Modellierung des Verhaltens der Geschwindigkeit und des Drucks am Rand.

Da in der Navier-Stokes-Gleichung nur der Gradient des Drucks auftritt, ist p durch (2.2) - (2.6) nur bis auf eine Konstante eindeutig bestimmt. Daher fordert man zur Normierung

$$\int_\Omega p \, d\Omega = 0 \quad \text{zu allen Zeiten } t.$$

In der Literatur findet man die Navier-Stokes-Gleichung häufig mit $\frac{1}{Re}$ statt ν wie in Gleichung (2.2) angegeben. Das liegt daran, dass man, falls die charakteristische Geschwindigkeit bekannt ist, durch eine geeignete Koordinatentransformation, wie sie im Kapitel 41.3 von [14] durchgeführt wird, die Gleichung auf diese Form bringen kann, so dass hier gerade die Reynoldszahl auftaucht.

In der Praxis soll natürlich durch die numerische Simulation erst die Geschwindigkeit bestimmt werden, weshalb wir wegen der größeren Praxisnähe der obigen Form den Vorzug geben werden.

Kann man in der Gleichung (2.2) den nichtlinearen Term vernachlässigen, so erhält man die nicht-stationäre *Stokes-Gleichung*:

$$\begin{aligned}
\frac{\partial v}{\partial t} - \nu \nabla^2 v + \nabla p &= f \text{ in } \Omega, && (2.7)\\
\nabla \cdot v &= 0 \text{ in } \Omega, && (2.8)\\
v &= 0 \text{ auf } \partial\Omega. && (2.9)
\end{aligned}$$

Häufig versteht man unter Stokes-Gleichung nur den Spezialfall $\nu = 1$. Wir werden hier jedoch die allgemeinere Variante verwenden. Des Weiteren existiert zur numerischen Behandlung der Stokes-Gleichung eine weiterentwickeltere, fundiertere Theorie als zur Navier-Stokes-Gleichung. Somit bietet es sich an, ein neues Verfahren zunächst für die Stokes-Gleichung zu testen und die dort vorhandene Theorie zu nutzen.

3 Splittingtechniken für die Navier-Stokes-Gleichung

Eine der Schwierigkeiten bei der numerischen Behandlung der Navier-Stokes-Gleichung besteht darin, dass Druck und Geschwindigkeit miteinander gekoppelt sind und es darum aufwendig ist, das Problem direkt zu lösen. Alternativ ist es möglich, mit Hilfe eines Splittingverfahrens (kurz: Splitting) separate Gleichungen für Druck und Geschwindigkeit zu gewinnen.

Dieser Ansatz zur Behandlung der Stokes- und Navier-Stokes-Gleichung hat eine lange Tradition. Splittingverfahren werden bereits seit Ende der sechziger Jahre eingesetzt und kommen mittlerweile in einer Reihe von unterschiedlichen Varianten bei der Lösung der Navier-Stokes-Gleichung zum Einsatz. In der Literatur wird die Klasse der Splittingverfahren daher mitunter noch weiter eingeteilt, z.B. in die Druckkorrektur- (pressure correction schemes) oder die Projektions-Verfahren (projection schemes).

Einen Überblick über die Klasse der Druckkorrektur- und Projektions-Verfahren gibt das Buch [32] von Andreas Prohl "*Projection and Quasi-Compressibility Methods for Solving the Incompressible Navier-Stokes Equations*".

Wir werden im Folgenden zwei bekannte Splittingverfahren kurz vorstellen, nämlich die Projektionsverfahren nach Chorin und nach Van Kan. Anschließend wird das Verfahren eingeführt, welches der Ausgangspunkt für das in dieser Arbeit entwickelte Splittingverfahren ist. Die dort verwendete Splittingtechnik wurde bereits mit Erfolg im Zusammenhang mit Spektralen Verfahren von Heinrichs und Haschke in [23], [24], [21] und [22] angewandt.

Anhand dieser drei unterschiedlichen Splittingverfahren können dann grundsätzliche Gemeinsamkeiten all dieser Verfahren gezeigt und abschließend auf die Stärken und Schwächen von Splittingverfahren im Allgemeinen sowie einzelner Verfahren eingegangen werden.
Darüber hinaus werden wir natürlich die mit Hilfe der verschiedenen Splittingverfahren erreichten numerischen Ergebnisse später vergleichen.

3.1 Das Projektionsverfahren nach Chorin

Das bereits 1968 von Chorin entwickelte Projektionsverfahren verwendet zur Zeitdiskretisierung der Gleichungen (2.2) das implizite Euler-Verfahren.

Der Splittingalgorithmus lässt sich wie folgt notieren, wobei mit $\triangle t \in \mathbb{R}^+$ die Schrittweite in der Zeit t bezeichnet wird. Der Index m gibt an, dass die Funktion für $m = 0, 1, 2, \ldots$ für $t^m \in [0, t_{end}]$ ausgewertet wird. Im Falle von äquidistanten Zeitschritten, die wir zunächst nur betrachten werden, gilt also $t^m = m\triangle t$.

Projektionsverfahren nach Chorin:

1. Starte mit einer Funktion v^0.
2. Für $m \geq 0$, mit gegebenem v^m, berechne zunächst $\tilde{v}^{m+1}$ aus der Gleichung

$$\frac{1}{\triangle t}\left(\tilde{v}^{m+1} - v^m\right) - \nu\nabla^2\tilde{v}^{m+1} + \left(v^m \cdot \nabla\right)\tilde{v}^{m+1} = f^{m+1} \tag{3.1}$$

$$\tilde{v}^{m+1} = 0 \text{ auf } \partial\Omega\ . \tag{3.2}$$

3. Mit dem zuvor berechneten $\tilde{v}^{m+1}$ werden $\left\{v^{m+1}, p^{m+1}\right\}$ durch Lösen der Gleichung

$$\frac{1}{\triangle t}\left(v^{m+1} - \tilde{v}^{m+1}\right) + \nabla p^{m+1} = 0 \tag{3.3}$$

$$\nabla \cdot v^{m+1} = 0\ ;\ v^{m+1} \cdot n = 0 \text{ auf } \partial\Omega \tag{3.4}$$

bestimmt.

$\tilde{v}^{m+1}$ bezeichnet eine „Zwischengeschwindigkeit“ (engl. intermediate velocity), welche von allen Splittingverfahren verwendet wird. Im Fall des Chorin-Verfahrens stellt die Zwischengeschwindigkeit die Geschwindigkeit zum Zeitpunkt t^{m+1} unter Vernachlässigung des Drucks dar.

Mit Hilfe des dritten Schritts wird der aktuelle Druck berechnet und zudem die Inkompressibilität des errechneten Vektorfeldes gesichert. Um dies zu erreichen, wird der dritte Schritt des Verfahrens mit Hilfe der Divergenzfreiheit von v^{m+1} und dem Divergenzoperator $(\nabla\cdot)$ wie folgt umformuliert:

$$-\nabla^2 p^{m+1} = \frac{-1}{\triangle t}\nabla \cdot \tilde{v}^{m+1}\ ,\ n \cdot \nabla p^{m+1} = 0 \text{ auf } \partial\Omega \tag{3.5}$$

Um die Divergenzfreiheit herzustellen und damit auch den Einfluss des Drucks wieder einzubringen, wird nun v^{m+1} einfach mittels

$$v^{m+1} = \tilde{v}^{m+1} - \triangle t\, \nabla p^{m+1} \tag{3.6}$$

berechnet.

Wie man sieht, muss innerhalb eines Zeitschrittes eine Konvektions-Diffusions-Reaktions-Gleichung mit Dirichlet-Randwerten und eine Laplace-Gleichung mit Neumann-Randwerten gelöst werden. Die Behandlung dieser beiden Gleichungstypen mit Hilfe der von uns und ebenso in [32] eingesetzten Finite-Element-Methode werden in Kapitel 4 beschrieben.

Problematisch sind die unphysikalischen Randwerte, welche in (3.5) für den Druck angesetzt werden. Wie wir in Kapitel 2.4 diskutiert haben, entspricht dies nur unter sehr speziellen Bedingungen dem tatsächlichen Randwertverhalten des Drucks. Abhängig von der Problemstellung können die homogenen Neuman-Randwerte zu einem stark verfälschten Druck, besonders in Randnähe, führen. Eine weitergehende Diskussion der Frage, in welchen Fällen der mit Hilfe des Projektionsverfahrens nach Chorin berechnete Druck eine geeignete Approximation ist, findet man bei [32], S.106ff.
Wegen des eingesetzten Zeitschrittverfahrens ist das Verfahren höchstens von erster Ordnung, siehe hierzu [32], S.7 und 107.

3.2 Das Projektionsverfahren nach Van Kan

Auf Grund seiner niedrigen Ordnung ist das Projektionsverfahren nach Chorin für viele Anwendungen nicht sehr attraktiv. Es wurden und werden Verfahren von höherer Ordnung benötigt, um Probleme effizient und zugleich mit hinreichend großer Genauigkeit zu lösen.

Ein Verfahren, das 1986 entworfen wurde, um diesem Anspruch gerecht zu werden, ist das Projektionsverfahren nach Van Kan. Dieses kann mit Hilfe des Crank-Nicolson-Verfahrens zur Zeitdiskretisierung ein Splittingverfahren zweiter Ordnung realisieren. Die hier vorgestellte Version des Verfahrens entstammt ebenfalls [32], S.5 bzw. 141ff.

Projektionsverfahren nach Van Kan:

1. Starte mit einer Funktion v^0 und $p^0 = p(0)$.
2. Für $m \geq 0$, mit gegebenen $v^m, \tilde{v}^m, p^m$, berechne zunächst $\tilde{v}^{m+1}$ aus der Gleichung

$$\frac{1}{\triangle t}\left(\tilde{v}^{m+1} - v^m\right) - \frac{\nu}{2}\nabla^2(\tilde{v}^{m+1} + \tilde{v}^m) + \frac{3}{2}\left(v^m \cdot \nabla\right)\tilde{v}^m \tag{3.7}$$

$$= \bar{f}^{m+1/2} + \frac{1}{2}\left(v^{m-1} \cdot \nabla\right)\tilde{v}^{m-1} - \nabla p^m \tag{3.8}$$

$$\tilde{v}^{m+1} = 0 \text{ auf } \partial\Omega\,. \tag{3.9}$$

3. Mit dem zuvor berechneten $\tilde{v}^{m+1}$ werden $\left\{v^{m+1}, p^{m+1}\right\}$ durch Lösen der Gleichung

$$\frac{1}{\triangle t}\left(v^{m+1} - \tilde{v}^{m+1}\right) + \frac{1}{2}\nabla(p^{m+1} - p^m) = 0 \tag{3.10}$$

$$\nabla \cdot v^{m+1} = 0\ ;\ v^{m+1} \cdot n = 0 \text{ auf } \partial\Omega \tag{3.11}$$

bestimmt.

Bei diesem Vorgehen muss der erste Schritt gesondert behandelt werden, da hier noch kein $\tilde{v}^m$ vorliegt. Häufig wird daher der erste Schritt laut [32], S.5 implizit durchgeführt.

Im Gegensatz zum ersten vorgestellten Splittingverfahren wird hier der Druck zur Berechnung der Zwischengeschwindigkeit nicht völlig vernachlässigt, sondern der Druck des letzten Zeitschritts als Approximation an den Druck zum Zeitpunkt t^{m+1} verwendet.

Wie schon bei dem Verfahren nach Chorin wird der dritte Schritt in eine Laplace-Gleichung mit homogenen Neumann-Randbedingungen umformuliert.

Das in [32], S.22/23 vorgestellte Benchmarkproblem für die Stokes-Gleichung ist von uns in die Reihe der in dieser Arbeit untersuchten Testprobleme aufgenommen worden.

3.3 Das Splitting nach Haschke und Heinrichs

Wie schon erwähnt, war das folgende Splitting Ausgangspunkt der Überlegungen und Tests für das von uns entwickelte adaptive Splittingverfahren auf der Basis der Finite-Element-Methode.

In seinen Artikeln [23] und [24] beschäftigte sich W. Heinrichs mit der Anwendung auf die Stokes-Gleichung unter Verwendung der Spektralen Verfahren. Dabei konstruierte er ein Verfahren dritter Ordnung für die Stokes-Gleichung, das er in seinem Artikel von 1998 vorstellt. Auf die Navier-Stokes-Gleichung wurde das Splitting dann schließlich in [21] und [22] von Haschke und Heinrichs übertragen und erreichte Reduktionsraten zweiter Ordnung für die Navier-Stokes-Gleichung.

Dieses Splitting werden wir zunächst für die Stokes- und anschließend seine Erweiterung für die Navier-Stokes-Gleichung in der von Haschke und Heinrichs verwendeten Form vorstellen.
Es hat sich im Laufe der Arbeit gezeigt, dass für dieses Splitting in Kombination mit der Finite-Elemente-Methode einige Modifikationen notwendig sind, die in Kapitel 5 präsentiert werden.

3.3.1 Das Splitting der Stokes-Gleichung

Zur Zeitdiskretisierung der Gleichungen (2.7) - (2.9) werden die bekannten BDF-Verfahren eingesetzt, auf die wir im Kapitel 4.3 noch detaillierter eingehen werden. Das BDF-Schema wird durch den Operator

$$L^m_{t,l} v^{m+1} = \frac{1}{\triangle t} \sum_{j=0}^{l} \beta_j v^{m+1-j} \tag{3.12}$$

dargestellt und approximiert den Operator $\frac{\partial}{\partial t}$. Dabei gibt l die Ordnung des Verfahrens an. Mit dem Operator $L^m_{t,l}$ führt dies auf folgendes Schema:

$$\begin{aligned} L^m_{t,l} v^{m+1} - \nu \nabla^2 v^{m+1} + \nabla p^{m+1} &= f^{m+1} \text{ in } \Omega \, , && (3.13)\\ \nabla \cdot v^{m+1} &= 0 \text{ in } \Omega \, , && (3.14)\\ v^{m+1} &= 0 \text{ auf } \partial\Omega \, . && (3.15) \end{aligned}$$

Unter Verwendung eines Druckextrapolanten $\bar{p}_l^{m+1}$ erhält man die Gleichung für die Zwischengeschwindigkeit $\tilde{v}^{m+1}$

$$\begin{aligned} L_{t,l}^m \tilde{v}^{m+1} - \nu\nabla^2\tilde{v}^{m+1} + \nabla\bar{p}_l^{m+1} &= f^{m+1} \text{ in } \Omega, && (3.16)\\ \tilde{v}^{m+1} &= 0 \text{ auf } \partial\Omega && (3.17)\end{aligned}$$

und die Druckkorrektur mittels

$$\begin{aligned} \beta_0 \frac{v^{m+1} - \tilde{v}^{m+1}}{\triangle t} + \nabla(p^{m+1} - \bar{p}_l^{m+1}) &= 0 \text{ in } \Omega, && (3.18)\\ \nabla \cdot v^{m+1} &= 0 \text{ in } \Omega, && (3.19)\\ v^{m+1} \cdot n &= 0 \text{ auf } \partial\Omega. && (3.20)\end{aligned}$$

n ist wieder die äußere Einheitsnormale. Der Extrapolant des Drucks berechnet sich aus den vorherigen s Schritten, ist also maximal von der Ordnung s. Damit hängt die Konvergenzordnung von der Ordnung l des BDF-Schemas und von der Ordnung s des Extrapolanten ab, wobei gilt:

$$\bar{p}_s^{m+1} = \sum_{j=0}^{s-1} \gamma_j p^{n-j}. \qquad (3.21)$$

Die Wahl der Koeffizienten γ_m $(m = 0, \dots, s-1)$ orientiert sich an der Taylorentwicklung von p nach der Zeit. Damit ergibt sich für $s = 1$ $\gamma_0 = 1$ und für $s = 2$ $\gamma_0 = 2$ und $\gamma_1 = -1$. Mit

$$\tilde{f}^{m+1} = f^{m+1} - \frac{1}{\triangle t}\sum_{j=1}^{l} \beta_j v^{m+1-j} - \nabla\bar{p}_l^{m+1}$$

lassen sich die Gleichungen (3.16) und (3.17) zu folgendem Helmholtz-Problem umformulieren:

$$\begin{aligned} \left(-\nu\nabla^2 + \frac{\beta_0}{\triangle t} I\right)\tilde{v}^{m+1} &= \tilde{f}^{m+1} \text{ in } \Omega, && (3.22)\\ \tilde{v}^{m+1} &= 0 \text{ auf } \partial\Omega. && (3.23)\end{aligned}$$

Mit I wird dabei der Identitätsoperator bezeichnet. Die Gleichungen (3.18) - (3.20) lassen sich nun zu

$$\begin{aligned} \frac{\beta_0}{\triangle t} v^{m+1} + \nabla p^{m+1} &= g^{m+1} \text{ in } \Omega, && (3.24)\\ \nabla \cdot v^{m+1} &= 0 \text{ in } \Omega, && (3.25)\\ v^{m+1} \cdot n &= 0 \text{ auf } \partial\Omega, && (3.26)\end{aligned}$$

umformen, wobei $g^{m+1} = \frac{\beta_0}{\triangle t}\tilde{u}^{m+1} + \nabla\bar{p}_l^{m+1}$.

Mit Hilfe der Gleichungen (3.22) und (3.23) kann man nun die Zwischengeschwindigkeit $\tilde{v}^{m+1}$ berechnen. Wir werden also die Gleichungen (3.24)-(3.26) wie schon bei dem Verfahren von Chorin weiter umformen müssen. Dabei wird erneut die Divergenzfreiheit der Geschwindigkeit ausgenutzt, um die Gleichungen zu entkoppeln. Wendet man die Divergenz $(\nabla\cdot)$ auf die Gleichung (3.24) an, führt dies zu einer Gleichung, in der nur noch der Druck als Unbekannte

enthalten ist.
Auch hier tritt die Schwierigkeit auf, Randbedingungen für dieses Problem zu formulieren. Heinrichs setzt in [24] bei der Behandlung des Stokes-Problems auf angepasste Neumann-Randbedingungen, damit verwendet er die Gleichung

$$\nabla p = \nabla^2 v + f$$

zu ihrer Konstruktion. Beim Navier-Stokes-Problem hingegen beschränken sich Haschke und Heinrichs in [21] und [22] auf die Verwendung homogener Randbedingungen. Damit wird das Vektorfeld des Drucks korrekt dargestellt, für den Druck sind in Randnähe hingegen Fehler auf Grund der homogenen Randwerte zu erwarten. Eine Analyse von Rannacher [34] für die auch in der vorliegenden Arbeit verwendete Finite-Element-Methode deutet jedoch darauf hin, dass eine sinnvolle Approximation des Drucks trotz der falschen Neumann-Randbedingungen möglich ist. Der durch die Randbedingungen hervorgerufene Fehler verringert sich exponentiell in Abhängigkeit von

$$\frac{\operatorname{dist}(x, \partial\Omega)}{\sqrt{\triangle t}}\,, \tag{3.27}$$

mit $x \in \Omega$.

3.3.2 Erweiterung für die Navier-Stokes-Gleichung

Das für die Stokes-Gleichung erarbeitete Splitting soll nun auf die Navier-Stokes-Gleichungen (2.2) - (2.5) übertragen werden. Hierfür gilt es zunächst zu klären, wie der neu hinzugekommene konvektive Term $(v \cdot \nabla)v$ behandelt werden soll. Dazu stehen im Wesentlichen zwei Methoden zur Verfügung. Der Term kann explizit wie beim Van-Kan-Splitting oder implizit wie beim Chorin-Splitting behandelt werden.
Während bei expliziten Methoden oft kleine Zeitschrittweiten benötigt werden, erhofft man sich, dass die implizite Behandlung eine größere Zeitschrittweite zulässt. Allerdings ist die in jedem Schritt zu lösende Konvektions-Diffusions-Reaktions-Gleichung wesentlich aufwendiger als die einfachere Diffusions-Reaktions-Gleichung. Diese tritt bei einer expliziten Behandlung auf, so dass der Rechenaufwand bei impliziten Verfahren höher ist.

Der explizite Ansatz

Zur expliziten Behandlung des konvektiven Terms gilt es zunächst eine geeignete Approximation v_e^{m+1} für v^{m+1} zu finden, d.h. man versucht, eine Näherung aus den letzten l Zeitschritten zu berechnen, wobei l der Ordnung des BDF-Verfahrens entspricht:

$$(v_e^{m+1} \cdot \nabla) v_e^{m+1} = \sum_{j=0}^{l} \alpha_j \left(v^{m-l+j} \cdot \nabla \right) v^{m-l+j}. \tag{3.28}$$

Bei der Wahl der α_j orientieren sich Haschke und Heinrichs zunächst an der Approximationsordnung von (3.21) und wählen sie entsprechend der Ordnung der Zeitdiskretisierung von v. Hat man also in der Zeit ein Verfahren erster Ordnung, so wird $\alpha_0 = 1$ gewählt und damit die Geschwindigkeit des letzten Zeitschritts als Näherung verwendet. Bei einem Verfahren zweiter Ordnung werden die α_j hingegen wie folgt gewählt (siehe dazu auch [11]):

$$\alpha_0 = -1, \qquad \alpha_1 = 2.$$

Liegt eine Zeitdiskretisierung 3. Ordnung vor, so gilt:

$$\alpha_0 = 1, \quad \alpha_1 = -3, \quad \alpha_2 = 3.$$

Analog zur Stokes-Gleichung führt die Zeitdiskretisierung der Gleichungen (2.2) - (2.4) dann zu folgendem Schema:

$$\begin{aligned} L_{t,l}^m v^{m+1} - \nu \nabla^2 v^{m+1} + \nabla p^{m+1} + (v_e^{m+1} \cdot \nabla) v_e^{m+1} &= f^{m+1} \text{ in } \Omega, && (3.29)\\ \nabla \cdot v^{m+1} &= 0 \text{ in } \Omega, && (3.30)\\ v^{m+1} &= h^{m+1} \text{ auf } \partial\Omega. && (3.31) \end{aligned}$$

Nun bringt man den konvektiven Anteil auf die rechte Seite von (3.29) und erhält eine Gleichung der Art (3.13) - (3.15), für die wir das Splitting bereits besprochen haben.
Für die vorliegende Navier-Stokes-Gleichung bedeutet das nun, dass nach der Wahl der α_i folgende Gleichungssysteme zu lösen sind:

$$\begin{aligned} \left(-\nu\nabla^2 + \frac{\beta_0}{\triangle t} I\right) \tilde{v}_i^{m+1} &= \tilde{f}_i^{m+1} - (v_e^{m+1} \cdot \nabla) v_e^{m+1} \quad \text{in } \Omega, && (3.32)\\ \tilde{v}_i^{m+1} &= h_i^{m+1} \text{ auf } \partial\Omega, \quad i = 1, 2. && (3.33) \end{aligned}$$

Der implizite Ansatz

Die implizite Behandlung erfolgt entsprechend der folgenden Gleichung:

$$\left(-\nu\nabla^2 + \frac{\beta_0}{\triangle t} I\right) \tilde{v}_i^{m+1} + \left(v_e^{m+1} \cdot \nabla\right) \tilde{v}_i^{m+1} = \tilde{f}_i^{m+1}, \quad i = 1, 2. \qquad (3.34)$$

Dabei ist v_e^{m+1} wieder eine Näherung an die aktuelle Geschwindigkeit. Als Startnäherung nimmt man die Geschwindigkeit des letzten Zeitschritts oder einen analog zum Vorgehen beim expliziten Ansatz gewählten Extrapolanten.

Im Gegensatz zum Vorgehen in der Arbeit von Haschke und Heinrichs, bei dem das explizite Verfahren ohne Picard-Iteration durchgeführt wurde, benutzen wir für das Splitting mit finiten Elementen sowohl bei der expliziten als auch bei der impliziten Behandlung eine Picard-Iteration mit der Startnäherung $v_e^{m+1} = v^m$.
Hierdurch ist der Extrapolant lediglich von erster Ordnung, doch hat sich diese Wahl als die robusteste herausgestellt. In [22] wird die Picard-Iteration ausschließlich für den impliziten Ansatz verwendet. Hierzu wird v_e^{m+1} durch die Lösung von 3.34 bzw. 3.32 ersetzt, $v_e^{m+1} = \tilde{v}^{m+1}$, und so erneut die Gleichung 3.34 bzw. 3.32 gelöst.
Diese Picard-Iteration wird so lange durchgeführt, bis

$$\|\tilde{v}_e^{m+1} - \tilde{v}_e^m\| < \varepsilon_{Pic} \qquad (3.35)$$

ist, dabei ist $\varepsilon_{Pic} > 0$ geeignet zu wählen. Alle in der Arbeit angestellten Berechnungen wurden mit $\varepsilon_{Pic} = 1 \cdot 10^{-3}$ durchgeführt.

3.4 Stärken und Schwächen von Splittingverfahren

Zu den Vorteilen der Splittingverfahren gehört der geringere Aufwand für die Lösung der Gleichungssysteme, die häufig einen Großteil der Rechenlast des Verfahrens ausmachen. Dazu kommt die einfache Implementierung, da die für die Teilprobleme benötigten Komponenten häufig bereits in hochoptimierter Form vorliegen. Die numerische Theorie der Teilprobleme ist in der Regel ebenfalls weiter fortgeschritten als die der Navier-Stokes-Gleichung als Gesamtproblem.

Viele Splittingverfahren sind hingegen für sehr große Zeitschrittweiten nicht gut geeignet. Zusätzlich kommt es laut [42], S.21f bei Splittingverfahren für große Viskositätskoeffizienten zu einem schlechteren Konvergenzverhalten. Da wie oben erwähnt kleine Viskositäten physikalisch interessant sind, ist dies oft von geringer Bedeutung.

Problematisch ist hingegen die Konstruktion eines echten Fehlerschätzers in der Zeit, sowohl für Splittingverfahren als auch für die Behandlung des ungesplitteten Problems. Die häufig gut funktionierende und auch in dieser Arbeit verwendete Variante, zwei Zeitschritte durchzuführen, einen mit $\triangle t$ und den zweiten in zwei Schritten mit einer Schrittweite von $\triangle t/2$, und anschließend die Ergebnisse zu vergleichen, erlaubt leider keine theoretisch fundierte und präzise Abschätzung des Fehlers des Gesamtproblems. Als Fehlerindikator erweist sie jedoch, wie wir sehen werden, gute Dienste und erlaubt so eine adaptive Zeitschrittweitensteuerung.

Weitere Schwierigkeiten treten ebenfalls laut [42], S.21f bei der Veränderung des räumlichen Gitters auf. Das bedeutet, dass das Zeitschrittverfahren in seiner Schrittweite unter Umständen angepasst werden muss, falls die räumliche Zerlegung des Gebietes verändert wird. Ein solches Verhalten ist besonders deshalb unerwünscht, weil es keine physikalischen Gründe für eine kleine Zeitschrittweite gibt.

Alle vorgestellten Projektionsverfahren haben laut [32], S.7 zusätzlich mit dem Auftreten von Fehlerlayern am Rand zu kämpfen, welche aus den falschen Neumann-Randbedingungen des Drucks resultieren. Daneben benötigen beide sehr glatte Lösungen, um das optimale Konvergenzverhalten zu erreichen.

Inwieweit die für Splittingverfahren typischen Schwierigkeiten bei dem in den folgenden Kapiteln entwickelten Verfahren auftreten und ob es gelingt, diese ggf. abzumildern, wird im Folgenden aufgezeigt.

4 Die Finite-Element-Methode in der Strömungsmechanik

Zur numerischen Lösung der Navier-Stokes-Gleichung mit Hilfe des Splittingverfahrens müssen eine Reihe von partiellen Differentialgleichungen gelöst werden. Hierzu wird die Finite-Element-Methode verwendet, welche sich u.a. durch eine hohe Flexibilität in Bezug auf die Gebietsgeometrie auszeichnet. Diese Eigenschaft ist sehr hilfreich bei der Modellierung von Problemen aus Natur und Technik, wie z.B. eines Flussverlaufes. Des Weiteren zeichnet sich die Finite-Element-Methode durch eine weit fortgeschrittene Theorie aus, die dazu dient, Eigenschaften wie Stabilität und Konvergenz zu untersuchen.

Um Indizes zu vermeiden und damit die Lesbarkeit zu verbessern, wird in diesem Kapitel eine Abbildung von $\Omega \to \mathbb{R}$ mit v statt mit v_i , $i = 1, 2$ bezeichnet.

4.1 Grundlagen der Finite-Element-Methode

Dieser Abschnitt soll einen kurzen Einblick in die Teile der Finite-Element-Methode geben, welche für die Arbeit von Bedeutung sind und einige wichtige Resultate und Abschätzungen zusammenfassen. Für einen detaillierteren Einblick in die Materie kann z.B. auf [6], [7] oder [27] zurückgegriffen werden.

Es sei im Folgenden $\Omega \subset \mathbb{R}^2$ und darüber hinaus soll Ω ein polygonales Gebiet sein, das heißt $\partial\Omega$ besteht aus einem geschlossenen Polygonzug.

Zur Lösung der Navier-Stokes-Gleichung mit Hilfe eines Splittingalgorithmus sind für uns lineare partielle Differentialgleichungen zweiter Ordnung vom Typ

$$-\nu\Delta v + k \cdot \nabla v + rv = f \tag{4.1}$$

$$v = 0 \text{ auf } \partial\Omega \tag{4.2}$$

von besonderer Bedeutung.

Sind alle Koeffizienten $\nu \in \mathbb{R}$ (Diffusionskoeffizient), $k : \Omega \to \mathbb{R}^2$ (Konvektionsfeld) und $r \in \mathbb{R}$ (Reaktionskoeffizient) ungleich Null bzw. der Nullabbildung, so handelt es sich um eine Konvektions-Diffusions-Reaktions-Gleichung, welche bei der numerischen Behandlung die größte Herausforderung darstellt. Sie tritt bei der impliziten Behandlung des nichtlinearen Terms mit $k = v_e$ auf. Des Weiteren werden die Helmholtz-Gleichung ($k = 0$) und die Laplace-Gleichung ($k, r = 0$) im Laufe des Splittingverfahrens benötigt.

Die Variationsformulierung

Finite-Element-Methoden basieren im Gegensatz zu Differenzen- oder Spektralenverfahren auf der Variationsformulierung der partiellen Differentialgleichung.

Die Lösung dieser Variationsformulierung wird, wie z.B. in [27] Kapitel 3.1 ausgeführt wird, innerhalb der Sobolev-Räume $H^m(\Omega)$ gesucht, welche Vervollständigungen von $C^m(\bar{\Omega})$ bzgl. der H^m-Norm $\|.\|_{H^m}$ sind. Des Weiteren wird vorausgesetzt, dass k und $f \in L^2(\Omega)$ sind.

Um die Variationsformulierung zu erhalten, wird (4.1) zunächst mit einer Testfunktion $u \in H_0^1(\Omega)$ multipliziert. Anschließend werden beide Seiten über Ω integriert und homogene Dirichlet-Randbedingungen angenommen. Dies geschieht o.B.d.A., da eine Reduktion auf homogene Randbedingungen möglich ist, vergleiche hierzu [6], S.35. Somit erhält man mit Hilfe der partiellen Integration im ersten Term der linken Seite

$$\int_\Omega \nu \nabla v \cdot \nabla u + k \cdot \nabla v u + r v u \, dx = \int_\Omega f u \, dx \; . \tag{4.3}$$

Es ist klar, dass ein v, das (4.1) erfüllt, ebenfalls Lösung von (4.3) ist. Da diese für ein beliebiges $u \in H_0^1(\Omega)$ gilt, wird nun mit den Bezeichnungen

$$a(v,u) := \int_\Omega \nu \nabla v \cdot \nabla u + k \cdot \nabla v u + r v u \, dx \tag{4.4}$$

$$(f,u)_{L^2} := \int_\Omega f u \, dx$$

gefordert, dass

$$a(v,u) = (f,u)_{L^2} \; \forall u \in H_0^1(\Omega) \tag{4.5}$$

gilt. Eine Funktion v, welche (4.5) erfüllt, nennt man eine Lösung des Variationsproblems. Bei $a(.,.)$ handelt es sich um eine für $k \neq 0$ nicht symmetrische Bilinearform. $b := (f,.)_{L^2}$ kann als lineares Funktional aufgefasst werden. Jede Lösung von (4.1) ist eine Lösung von (4.5), nicht jedoch umgekehrt.

Die Eindeutigkeit der Lösung garantiert der Satz von Lax-Milgram. Die hier angegebene Version für nicht-symmetrische Bilinearformen stammt aus [27], S.86.

Satz 4.1 (Lax-Milgram)
V sei ein Hilbert-Raum mit Skalarprodukt $(.\,,\,.)$ und einer davon erzeugten Norm $\|.\|$.
$a : V \times V \to \mathbb{R}$ sei eine (nicht notwendigerweise symmetrische) Bilinearform und $b : V \to \mathbb{R}$ ein lineares Funktional. Es gelte:

- *a ist stetig, das heißt, es existiert eine Konstante $C_S > 0$ mit*

$$|a(v,u)| \leq C_S \|v\| \|u\| \; \forall v,u \in V$$

- *a ist V-elliptisch, das heißt, es existiert eine Konstante $C_E > 0$ mit*

$$|a(v,v)| \geq C_E \|v\|^2 \; \forall v \in V$$

- *b ist stetig, das heißt, es existiert eine Konstante $C > 0$ mit*

$$|b(v)| \leq C\|v\| \; \forall v \in V$$

Dann besitzt die Variationsgleichung (4.5) genau eine Lösung.

Beweis: Siehe [2], S.147.

Falls (4.1) eine klassische Lösung ($u \in C^2(\bar{\Omega})$) besitzt, handelt es sich folglich um dieselbe Funktion, die auch (4.5) löst. Hier sind also schwache und klassische Lösung identisch. Anderenfalls existiert für (4.1) nur eine schwache Lösung. Setzt man voraus, dass

- $V := H_0^1(\Omega)$
- $\nu > 0$, $k \in C^1(\bar{\Omega}, \mathbb{R})$, $r \in C(\bar{\Omega})$, $f \in L^2(\Omega)$
- eine Konstante $r_0 > 0$ existiert, mit der in Ω $r - \frac{1}{2}\nabla \cdot k \geq r_0$ gilt,

dann greift der Satz von Lax-Milgram für das Variationsproblem (4.5). Die Herleitung für den Fall $k \neq 0$ erfolgt über $\nu-$gewichtete Normen und kann in [27], Kapitel 9 nachgelesen werden.

Wichtig ist, dass die Stetigkeitskonstante C_S von der Größe aller Koeffizienten abhängt. Bei der Konstante C_E lässt sich lediglich eine Abhängigkeit vom Minimum aus ν und r zeigen. Bevor wir uns einem numerischen Verfahren zur Lösung des Variationsproblems zuwenden, werden wir noch kurz einige Betrachtungen zur Regularität von Variationsproblemen anstellen, welche für spätere Fehlerabschätzungen benötigt werden.

Definition 4.2
Sei $m \geq 1$, $H_0^m(\Omega) \supset V \subset H^m(\Omega)$ und $a(.,.)$ eine V-elliptische Bilinearform. Das Variationsproblem

$$a(v, u) = (f, u) \text{ für alle } u \in V$$

heißt H^s-regulär, wenn es zu jedem $f \in H^{s-2m}(\Omega)$ eine Lösung $v \in H^s(\Omega)$ gibt und mit einer Zahl $c = c(\Omega, a, s)$:

$$\|v\|_s \leq c\|f\|_{s-2m} \tag{4.6}$$

gilt.

Satz 4.3 (Regularitätssatz)
Sei a eine H_0^1-elliptische Bilinearform mit hinreichend glatten Koeffizientenfunktionen.

1. *Wenn Ω konvex ist, ist das Dirichlet-Problem H^2-regulär.*
2. *Sei $s \geq 2$. Wenn Ω einen C^s-Rand besitzt, ist das Dirichlet-Problem H^s-regulär.*

Beweis: vgl. [6], S.83f.

Das Galerkin-Verfahren

Das Variationsproblem ist zunächst über einem unendlich-dimensionalen Raum definiert. Um dieses Problem numerisch behandeln zu können, muss man sich bei dem Versuch, die Lösung des Variationsproblems zu berechnen, auf einen geeigneten endlich-dimensionalen Raum beschränken.
Dieser endlich-dimensionale Teilraum V_h von H^m heißt Finite-Element-Raum. Er entsteht

1. durch eine Zerlegung $\mathcal{T}_h$ des Gebietes Ω in endlich viele finite Elemente, wobei es sich wie in unserem Fall um Dreiecke handeln kann.

2. durch die Wahl der auf den einzelnen Elementen definierten Ansatzfunktionen, im Folgenden Polynome ersten Grades.

Der Index h des Teilraums V_h gibt die Feinheit der Zerlegung an. Er wird definiert als

$$h := \max_{T \in \mathcal{T}_h} \{h_T\} ,$$

wobei h_T den Durchmesser eines Elements $T \in \mathcal{T}_h$ bezeichnet. Die Angabe dieses Wertes in der Literatur ist nicht einheitlich. Oft wird mit h_T auch der halbe Durchmesser bezeichnet. Eine solche Zerlegung von Ω soll einigen Forderungen genügen.

Definition 4.4
Eine Zerlegung $\mathcal{T} = \{T_1, T_2, T_3, ..., T_m\}$ von Ω in Dreiecke heißt zulässig, wenn folgende Eigenschaften erfüllt sind:

1. $\bar{\Omega} = \cup_{i=1}^{m} T_i$

2. *Besteht $T \cap T'$ genau aus einem Punkt, so handelt es sich um einen Eckpunkt beider Elemente.*

3. *Besteht $T \cap T'$ aus mehr als einem Punkt, so handelt es sich um eine gemeinsame Kante von T und T'.*

Diese Art von Zerlegung in Dreiecke wird auch als Triangulierung bezeichnet. In dieser Arbeit werden ausschließlich solche zulässigen Triangulierungen verwendet. Als Knoten der Triangulierung bezeichnet man die Ecken der Dreiecke und die Menge all dieser Knoten einer Triangulierung mit $\mathcal{I}$. Wenn $K \in \mathcal{I}$ ein solcher Knoten ist, nennen wir die Knoten, die durch eine Kante mit ihm verbunden sind, seine Nachbarn $\mathcal{N}(K)$. Durch die Beschränkung auf zulässige Triangulierungen werden so genannte hängende Knoten vermieden.

Definition 4.5
Eine Familie von Zerlegungen $\{\mathcal{T}\}$ heißt quasiuniform, wenn es eine Zahl $\kappa > 0$ gibt, so dass jedes $T \in \mathcal{T}$ einen Kreis mit dem Radius $\rho_t \leq \frac{h_T}{\kappa}$ enthält.

Definition 4.6
Eine Familie von Zerlegungen $\{\mathcal{T}\}$ heißt uniform, wenn es eine Zahl $\kappa > 0$ gibt, so dass jedes $T \in \mathcal{T}$ einen Kreis mit dem Radius $\rho_t \leq \frac{h}{\kappa}$ enthält.

Werden auf jedem Element lineare Ansatzfunktionen verwendet, erhält man die *nodale* Basis, also die knoten-bezogene, indem man Basisfunktionen ϕ mit der folgenden Eigenschaft wählt:

$$\phi_K(x) = \begin{cases} 1, \text{ beim Knoten } K \\ 0, \text{ an allen anderen Knoten aus } \mathcal{I} \end{cases} .$$

Diese Basis spannt nun einen Raum stückweise linearer Funktionen auf, der in $H^m(\Omega)$ liegt.

Das Ziel ist es also, das Variationsproblem über einen solchen n-dimensionalen Finite-Element-Raum zu lösen. Wir suchen daher ein $v_h \in V_h$, für das gilt:

$$a(v_h, u_h) = (f, u_h)_{L^2} \; \forall u_h \in V_h(\Omega). \tag{4.7}$$

Da sich v_h als Linearkombination der Basisfunktionen ϕ_i schreiben lässt

$$v_h = \sum_{i=1}^{n} v_i \phi_i(x) ,$$

erhält man aus (4.7)

$$a(v_h, u) = \sum_{i=1}^{n} u_i a(\phi_i, u) .$$

Da (4.7) für alle $u_h \in V_h(\Omega)$ gelten soll, gilt es also insbesondere auch für die n Basisfunktionen. Folglich erhalten wir

$$a(v_h, \phi_j) = \sum_{i=1}^{n} v_i a(\phi_i, \phi_j) = (f, \phi_j)_{L^2} \; \forall \; j = 1...n .$$

Damit reduziert sich das Problem auf das Lösen eines Gleichungssystems

$$Au = f$$

mit $A := [a(\phi_i, \phi_j)]_{i,j=1}^{n}$, $f := [(f, \phi_i)_{L^2}]_{i=1}^{n}$ und $u := [u_h(K_i)]_{i=1}^{n}$. Die Gleichungssysteme, die dabei entstehen, sind in der Regel sehr groß und dünn besetzt, da die Bilinearform nur für Basisfunktionen von benachbarten Knoten einen von Null verschiedenen Eintrag erzeugt.

Man kann also die Arbeitsschritte im Lösungsprozess des Galerkin-Verfahrens wie folgt zusammenfassen:

1. **Gitter erzeugen**
   Ein Gittergenerator bildet auf Grund einer übergebenen Geometrie eine Triangulierung.
2. **Assemblieren von A und f** nach dem Schema aus [27], Kapitel 2.4.2
   In diesem Schritt wird die Matrix A sowie die vektorielle Form der rechten Seite f berechnet.
3. **Das entstandene Gleichungssystem lösen**

Es bleibt die Frage, wie genau mit Hilfe des Galerkin-Verfahrens die Lösung des unendlich-dimensionalen Variationsproblems in dem gewählten endlich-dimensionalen Unterraum approximiert werden kann. Eine generelle Aussage über die Qualität der erhaltenen Lösung liefert das Cèa-Lemma.

Lemma 4.7 (Cèa-Lemma)
Die Bilinearform a sei V-elliptisch, mit $H_0^m(\Omega) \subset V \subset H^m(\Omega)$. Ferner seien v bzw. v_h die Lösungen der Variationsaufgabe in V bzw. in $V_h \subset V$. Dann gilt

$$\|v - v_h\|_V \leq \left(\frac{C_S}{C_E}\right)^{1/2} \inf_{u_h \in V_h} \|v - u_h\|_V \ . \tag{4.8}$$

Beweis: Siehe [6], S.53.

Das Lemma zeigt, dass die Genauigkeit der berechneten Lösung v_h im Wesentlichen davon abhängt, dass ein Funktionsraum V_h gewählt wird, in dem die Lösung v gut approximierbar ist. Je glatter die Lösung ist, desto besser lässt sie sich auf jedem Element mit Polynomen annähern. Dies wird auch aus dem folgenden Satz deutlich, für dessen Formulierung wir noch eine weitere Definition benötigen.

Definition 4.8 (Interpolationsoperator)
Mit I_h bezeichnen wir den Interpolationsoperator von $V \to V_h$, für den

$$I_h(v) = \sum_{i=1}^{m} v(K_i)\phi_i \ \in V_h$$

gilt, wobei $K \in \mathcal{I}$ und ϕ_i der zugehörige Basisvektor ist. Mit m wird die Dimension von V_h bezeichnet.

Satz 4.9
Sei $t \leq 2$ und $\mathcal{T}_h$ eine quasiuniforme Triangulierung von Ω. Dann gilt für die Interpolation durch stückweise Polynome vom Grad $t-1$ mit einer Konstanten $c = c(\Omega, \kappa, t)$:

$$\|v - I_h v\|_{m,h} \leq c \cdot h^{t-m} |v|_{t,\Omega} \text{ für } v \in H^t(\Omega), \ 0 \leq m \leq t \ .$$

Beweis: vgl. [6], S.73.

Wird in einem Teil des Gebietes Ω der Fehler zu groß, so kann die Genauigkeit dort dadurch erhöht werden, dass man die Dreiecke dort verkleinert. Darüber hinaus existiert die Möglichkeit, den Grad der Polynome in dem entsprechenden Gebiet zu erhöhen. Um herauszufinden, wo solche Maßnahmen nötig sind, wird ein Fehlerschätzer, wie wir ihn in Kapitel 7.2.2 entwickeln werden, benötigt.
Für die Abschätzung des Interpolationsfehlers in der L^2-Norm sagt Satz 4.9 eine Fehlerreduktion mit h^2 voraus. Laut [6], S.85f lässt sich mit Hilfe des Aubin-Nitsche-Lemmas zeigen, dass sich diese Eigenschaft auch auf die durch das Galerkin-Verfahren erhaltene Näherung überträgt.

4.2 Die Stromliniendiffusionsmethode

Erinnern wir uns daran, welche Größen aus der Gleichung 4.1 in die Konstanten C_S und C_E eingehen, so wird klar, dass das Cèa-Lemma für den Fall eines im Vergleich zu ν dominanten k keine gute Approximation durch die Finite-Element-Lösung garantiert.

Eine Möglichkeit, die Qualität der Finite-Element-Lösung für konvektionsdominante Konvektions-Diffusions-Reaktions-Gleichungen zu verbessern, ist die Stromliniendiffusionsmethode.

Betrachten wir hierzu erneut die Gleichung (4.1)

$$-\nu \Delta v + k \cdot \nabla v + rv = f$$

und zwar für den Fall, dass alle Koeffizienten ungleich Null sind.
Abhängig von der Größe dieser Koeffizienten hat die Gleichung einen unterschiedlichen Charakter. Mit Hilfe einer dimensionslosen Größe, der *Péclet-Zahl*, lässt sich dieser bestimmen. Die globale Péclet-Zahl ist definiert durch

$$Pe := \frac{\|k\|_{0,\infty} \operatorname{diam}(\Omega)}{|\nu|}.$$

Sie wird als global bezeichnet, da hier der Durchmesser des gesamten Gebietes Ω verwendet wird.

Wenn die Péclet-Zahl wesentlich größer als 1 ist, handelt es sich um ein konvektionsdominiertes Problem, ansonsten um ein diffusionsdominiertes.
Zwar sind auch konvektionsdominierte Probleme elliptische partielle Differentialgleichungen, jedoch ist ihre Elliptizität unter Umständen so schwach, dass sie nahezu einen elliptisch-hyperbolischen Grenzfall darstellen. Bei ihrer numerischen Behandlung versagen daher viele Methoden, die für andere elliptische partielle Differentialgleichungen gute Resultate liefern.

Zunächst gilt nach dem Satz von Lax-Milgram, dass eine eindeutige Lösung existiert und der Approximationsfehler der mit Hilfe des Galerkinverfahrens berechneten Lösung sich durch das Cèa-Lemma abschätzen lässt. Dieses stellt jedoch auf Grund der großen Konstante $\frac{C_S}{C_E}$ eine gute Approximation erst für unverhältnismäßig kleine h sicher.

Die entsprechenden Ungenauigkeiten müssen zwar nicht auftreten, jedoch hat es sich gezeigt, dass die berechneten Lösungen v_h zu Oszillationen neigen, die bei der exakten Lösung v nicht auftreten. Die Oszillationen verschwinden erst auf sehr feinen Zerlegungen, bei denen die Anzahl der Unbekannten in der Regel schon unvertretbar groß wird.
Um diesem Verhalten vorzubeugen, werden wir die Stromliniendiffusionsmethode als Stabilisierungstechnik einsetzen.

Die grundlegende Idee, die aus einem Artikel von Hughes und Brooks aus dem Jahre 1979 stammt, besteht darin, die Elliptizität durch Addition geeigneter Residuen zu verbessern und damit die Konstante $(\frac{C_S}{C_E})$ im Cèa-Lemma (4.7) zu verringern.
Dadurch wird ein mit der Stromliniendiffusion stabilisiertes Galerkin- zu einem Petrov-Galerkin-Verfahren, vergleiche hierzu [31], Kapitel 5.
Wir betrachten nun erneut die Gleichung (4.1)

$$-\nu \Delta v + k \cdot \nabla v + rv = f \text{ in } T \quad \forall T \in \mathcal{T}_h \tag{4.9}$$

für jedes einzelne Element der Triangulierung. Für die Koeffizienten gelten die am Ende des Kapitels 4.1 genannten Bedingungen.

Nehmen wir nun an, v sei die Lösung von (4.9) auf $T \in \mathcal{T}_h$ und beschränken wir uns erneut an dieser Stelle auf lineare Ansatzfunktionen. Die allgemeine Herleitung findet man in [35], Kapitel 3.2.1.

Multipliziert man (4.9) mit $k \cdot \nabla u$, $u \in H_0^1(\Omega)$ und einem Skalar δ_T, ist v ebenfalls Lösung dieser modifizierten Gleichung. Es existieren auch Alternativen zur Wahl der Funktion $k \cdot \nabla u$, z.B. eine in der Strömungsmechanik populäre Wahl ist $-\nu\Delta u + k \cdot \nabla u + ru$, welche auf die Galerkin/least-squares-FEM führt. Durch Integration der so entstandenen Gleichung erhält man

$$\int_T \delta_T(-\nu\Delta v(k\cdot\nabla u)+(k\cdot\nabla v)(k\cdot\nabla u)+rv(k\cdot\nabla u))\,dx = \int_T \delta_T(f(k\cdot\nabla u))\,dx\ \forall T \in \mathcal{T}_h\ . \quad (4.10)$$

Für die verwendeten linearen Ansatzfunktionen ist $\Delta v|_T = 0$, wodurch der erste Term der linken Seite entfällt und sich (4.10) zu

$$\delta_T \int_T (\nabla u)^t(kk^t)\nabla v + ru(k\cdot\nabla v)\,dx = \delta_T \int_T f(k\cdot\nabla v)\,dx \quad \forall T \in \mathcal{T}_h \quad \forall v \in V_h \quad (4.11)$$

vereinfacht. Addiert man nun (4.11) zur variationellen Formulierung

$$a(u,v) = (f,v)_{L^2} \quad \forall v \in V_h$$

von (4.1), erhält man mit den Bezeichnungen

$$a_\delta(v,u) = \sum_{T\in\mathcal{T}_h} \int_T (\nabla v)^t(\nu I + \delta_T kk^t)\nabla u + (1-\delta_T r)(k\cdot\nabla v)u + r(1-\delta_T\nabla\cdot k)vu\,dx \quad (4.12)$$

und

$$f_\delta := (f,u)_{L^2} + \sum_{T\in\mathcal{T}_h} \delta_T(f, k\cdot\nabla v)_{L^2} \quad (4.13)$$

das variationelle Problem

$$a_\delta(v,u) = f_\delta \quad \forall u \in V_h\ . \quad (4.14)$$

Es ist also klar, dass falls

$$v \text{ Lösung von (4.1)} \Rightarrow v \text{ Lösung von (4.14).}$$

Auf welche Weise das Verfahren hierdurch stabilisiert wird, erkennt man schnell, wenn man annimmt, dass der Reaktionskoeffizient $r = 0$ sei und dann erneut (4.12) betrachtet:

$$a_\delta(v,u) = \sum_{T\in\mathcal{T}_h} \int_T (\nabla v)^t(\nu I + \delta_T kk^t)\nabla u + k\cdot\nabla vu\ . \quad (4.15)$$

Wenn man $(\nabla v)^t kk^t \nabla u$ ein wenig umschreibt

$$(\nabla v)^t kk^t \nabla u = (k\cdot\nabla v)(k\cdot\nabla u) = \cos\alpha_1 \cos\alpha_2 (|k|)^2 |\nabla v||\nabla u|\ ,$$

wobei α_1, α_2 die Winkel zwischen k und u bzw. v sind, erkennt man leicht, warum (4.9) mit $k \cdot \nabla u$, $u \in H_0^1(\Omega)$ multipliziert wurde. Der Diffusionsterm wird durch den neuen

Summanden, hauptsächlich in Richtung der Konvektion, vergrößert. Man addiert also eine künstliche Diffusion zur bereits vorhandenen und vergrößert somit die für die Stabilität wichtige Konstante C_E in der entscheidenden Raumrichtung.

Um erneut mit Hilfe des Satzes 4.1 (Lax-Milgram) die Eindeutigkeit der Lösung von (4.14) zeigen zu können, muss noch die Bilinearform (4.12) auf V-Elliptizität überprüft werden. Mit

$$r_T := \max_{x \in T} |r(x)| \text{ für jedes } T \in \mathcal{T}_h$$

und erneut

$$r - \frac{1}{2}\nabla \cdot k \geq r_0 > 0$$

gelangt man, wie in [35], S.231 für den allgemeineren Fall bewiesen wird, zu folgendem Lemma.

Lemma 4.10
Für die Wahl

$$0 < \delta_T \leq \frac{1}{2}\frac{r_0}{r_T^2}$$

auf jedem $T \in \mathcal{T}_h$ ist die Bilinearform (4.12) V-elliptisch.

Eine Hilfestellung bei der Wahl dieses Parameters liefert der folgende Satz, zu dessen Hilfe wir noch die

Definition 4.11 (Stromliniendiffusionsnorm)

$$\|v\|_{sd} := \left(\nu |v|_1^2 + \frac{r_0}{2}\|v\|_0^2 + \sum_{K \in \mathcal{T}_h} \delta_T \|k \cdot \nabla v\|_{0,K}^2 \right)^{1/2} \quad , \; v \in V \tag{4.16}$$

benötigen.

Satz 4.12
Die bestmögliche Stabilisierung erhält man für die Wahl

$$\delta_T = \begin{cases} \delta_0 h_T, & \text{für } Pe_T > 1 \text{ (konvektionsdominierter Fall)} \\ \delta_1 \frac{h_T^2}{\nu} & \text{für } Pe_T \leq 1 \text{ (diffusionsdominierter Fall)} \end{cases}$$

mit der lokalen Péclet-Zahl

$$Pe_T := \frac{\|k\|_{0,\infty,T} h_T}{2\nu} \; ,$$

wobei δ_0 und δ_1 unter der Einschränkung von (4.10) zu wählen sind. Wird δ_T entsprechend der obigen Einschränkungen gewählt und liegt die Lösung von 4.14 in $H^{k+1}(\Omega)$, so gilt

$$\|v - v_h\|_{sd} \leq C(\sqrt{\nu} + \sqrt{h}) h^k |v|_{k+1} \; . \tag{4.17}$$

Beweis und Herleitung: Siehe [35], S.231-233 oder [27], S.325-329.

Trotz dieser Einschränkung bleibt ein großer Spielraum für die Wahl des Stromliniendiffusionsparameters δ_T.

Die Wahl des Stromliniendiffusionsparameters

Die Frage nach der Wahl des optimalen Stromliniendiffusionsparameters δ_T ist bis heute nicht abgeschlossen. Für eindimensionale Probleme ist laut [35], Remark 3.34 bei der Verwendung von linearen Testfunktionen die Berechnung eines optimalen δ_T möglich.

Für mehrdimensionale Probleme wie in unserem Fall ist

$$\hat{\delta}_T := \begin{cases} \frac{h_T}{\|k\|_{0,\infty,\Omega}}, & \text{für } Pe_T > 1 \text{ (konvektionsdominierter Fall)} \\ 0, & \text{für } Pe_T \leq 1 \text{ (diffusionsdominierter Fall)} \end{cases}$$

eine sinnvolle und häufige Wahl für den Stromliniendiffusionsparameter, wie z.B. in [39] oder [13], S.463.
Die Wahl von $\delta_1 = 0$ kann dadurch gerechtfertigt werden, dass für den diffusionsdominierten Fall die lokale Péclet-Zahl bereits kleiner als eins ist und somit kein Handlungsbedarf zur Stabilisierung auf diesem Element besteht. Da δ_T außerdem im diffusionsdominierten Fall für ein sehr kleines δ_1 näherungsweise Null ist, bietet sich diese Wahl an, um die Kosten für eine Assemblierung auf diesen Elementen zu sparen. Eine solche Wahl von δ_1 ist im Sinne von Lemma 4.10 ein theoretischer Grenzfall.

Die Gründe für die Wahl von δ_0 werden deutlich, wenn man für den konvektionsdominierten Fall die lokale Péclet-Zahl der Bilinearform (4.12) in Richtung der Konvektion betrachtet:

$$\frac{\|k\|_{0,\infty,T} h_T}{2(\nu + \delta_T \|k\|^2_{0,\infty,T})} = \frac{\|k\|_{0,\infty,T} h_T}{2(\nu + \frac{h_T}{\|k\|_{0,\infty,\Omega}} \|k\|^2_{0,\infty,T})} \leq \frac{1}{2} \frac{\|k\|_{0,\infty,T} h_T}{\|k\|_{0,\infty,T} h_T} < 1 \; .$$

Für die Wahl $\delta_T = \hat{\delta}_T$ ist damit sichergestellt, dass die lokale Péclet-Zahl auf dem Element $T \in \mathcal{T}_h$ ausreichend reduziert wird.

Wie wir im nächsten Kapitel sehen werden, ist bei den für das Splitting zu lösenden Gleichungen oft der Reaktionskoeffizient recht groß, daher sind diese weniger problematisch als reine Konvektions-Diffusions-Gleichungen, welche keinen Reaktionsterm enthalten. Dies macht sich auch bei dem Verhalten der Löser positiv bemerkbar. Diesem Umstand haben wir auch bei der Wahl von δ_T Rechnung getragen und hier lediglich $\delta_T = \frac{1}{2}\hat{\delta}_T$ zur Stabilisierung verwendet.

4.3 Parabolische Probleme

Durch das Splitting der Navier-Stokes-Gleichung entstehen parabolische Teilprobleme vom Typ

$$\frac{\partial v}{\partial t} - \nu \Delta v + k \cdot \nabla v = f \quad (i = 1, 2) \; , \tag{4.18}$$

zu deren numerischer Behandlung wir die in [27], S.257f beschriebenen volldiskreten Schemata einsetzen werden. Dabei wird zunächst der Operator $\frac{\partial}{\partial t}$ in geeigneter Weise diskretisiert.
Zu diesem Zwecke nutzen wir Verfahren, die aus der Behandlung gewöhnlicher Differentialgleichungen bekannt sind. Wir werden uns dabei in dieser Arbeit auf lineare Mehrschrittverfahren

und zwar die BDF-Verfahren (Backwards Differentiation Formulas) bis einschließlich der Ordnung 3 konzentrieren.
Dazu betrachten wir den Operator aus (3.12)

$$L_{t,l}^{m} v^{m+1} = \frac{1}{\triangle t} \sum_{j=0}^{l} \beta_j v^{m+1-j} .$$

Für Verfahren mit einer äquidistanten Schrittweite können die Koeffizienten β_j der Tabelle 4.1 aus [11] entnommen werden. Ersetzen wir nun den Operator $\frac{\partial}{\partial t}$ in (4.18) durch seine Approximation, so erhalten wir

Ordnung	β_0	β_1	β_2	β_3
1	1	-1	-	-
2	$\frac{3}{2}$	-2	$\frac{1}{2}$	-
3	$\frac{11}{6}$	-3	$\frac{3}{2}$	$-\frac{1}{3}$

Tabelle 4.1: BDF-Koeffizienten

$$-\nu \Delta v^{m+1} + k \cdot \nabla v^{m+1} + \frac{\beta_0}{\triangle t} v^{m+1} = f - \frac{1}{\triangle t} \sum_{j=1}^{l} \beta_j v^{m+1-j} . \tag{4.19}$$

Hiermit zerfällt das zeitabhängige Problem nun in eine nacheinander zu lösende Folge von Gleichungen des Typs (4.1) mit $r = \frac{\beta_0}{\triangle t}$ und $k = v_e$.

Fehlerabschätzungen für das implizite Euler-Verfahren

Mit Hilfe von Fehlerabschätzungen für parabolische Probleme ist es uns möglich, Rückschlüsse auf das Fehlerverhalten des Splittings im günstigsten Fall zu ziehen. Betrachten wir dazu erneut (2.7)-(2.9)

$$\begin{aligned} \frac{\partial v}{\partial t} - \nabla^2 v + \nabla p &= f \text{ in } \Omega, \\ \nabla \cdot v &= 0 \text{ in } \Omega, \\ v &= 0 \text{ auf } \partial\Omega . \end{aligned}$$

Wäre p für alle $t \in \mathbb{R}^+$ bekannt, so würde das Problem (2.7)-(2.9) in zwei ungekoppelte parabolische Standardprobleme zerfallen. Um diesen Gedanken weiter zu verfolgen, nehmen wir o.B.d.A. an, für das bekannte p gelte p=0 und es würde als Zeitschrittverfahren das implizite Euler-Verfahren eingesetzt.

Definition 4.13 (Ritz-Projektion)
Die elliptische oder Ritz-Projektion R_h von $V \to V_h$ ist für eine V-elliptische, stetige Bilinearform $a : V \times V \to \mathbb{R}$ definiert durch:

$$v \mapsto R_h v \Leftrightarrow a(R_h v - v, v_h) = 0 \text{ für alle } v_h \in V_h ,$$

wobei $K \in \mathcal{I}$ und ϕ_i der zugehörige Basisvektor ist.

Unter den obigen Annahmen gilt mit der Definition 4.13 die Abschätzung aus dem folgenden Satz.

Satz 4.14
Es sei a eine V-elliptische, stetige Bilinearform, $v_h^0 \in V$, $v^0 \in V$. *Ist* $v \in C^2([0,T],V)$, *so gilt die Abschätzung*

$$\begin{aligned} \|v(t_n) - v^n\|_0 &\leq \|v_h^0 - R_h v^0\|_0 + \|(I - R_h)v(t_n)\|_0 && (4.20) \\ &+ \int_0^{t_n} \|(I - R_h)v'(s)\|_0 \, ds + \triangle t \int_0^{t_n} \|v''(s)\|_0 \, ds \; . && (4.21) \end{aligned}$$

Beweis: Siehe [27], S.260.

Wie in [27], S.260 nachgelesen werden kann, folgt unter weiteren Voraussetzungen, die da wären $v^0 \in V \cup H^2(\Omega)$ und $\|v_h^0 - v^0\|_0 \leq Ch^2\|v^0\|_2$, dass sich die Abschätzung aus Satz 4.14 zu

$$\|v(t_n) - v^n\|_0 \leq C(u)\ (h^2 + \triangle t) \qquad (4.22)$$

vereinfacht. Für ein Zeitschrittverfahren höherer Ordnung käme unter ähnlichen Voraussetzungen $\triangle t$ in der entsprechenden Potenz vor.
Aus dieser Abschätzung wird klar, dass zur Bestimmung der Ordnung eines Splittingverfahrens in der Regel sowohl die Zeitschrittweite $\triangle t$ als auch die Diskretisierung h reduziert werden muss.

Stabilitätseigenschaften der BDF-Verfahren

Für den Einsatz der BDF-Verfahren zur Zeitdiskretisierung sprechen ihre guten Stabilitätseigenschaften. Das BDF-Verfahren erster und zweiter Ordnung besitzt die zur Behandlung steifer Probleme nötige $A-$Stabilität, während das Verfahren dritter Ordnung fast $A-$stabil ist, nämlich $A(\alpha)$-stabil mit $\alpha = 86.03°$ ($A(\alpha)$-stabil mit 90° entspricht A-stabil). Eine ausführliche Analyse der Stabilitätseigenschaften von BDF-Verfahren findet man bei [40] in den Kapiteln 7.2 bzw. 6.2.

4.4 Sattelpunkt-Probleme

Bei der Wahl der Finite-Element-Räume für die Geschwindigkeit v und den Druck p orientieren wir uns an der Sattelpunkt-Theorie der Stokes-Gleichung. Eine ausführliche Darstellung der Theorie der Sattelpunktprobleme findet man z.B. in [19], Kapitel 12 oder [6], S.139ff. Dazu betrachten wir die zeitunabhängige Stokes-Gleichung mit dem Koeffizienten $\nu = 1$. Die Wahl von $\nu = 1$ erfolgt o.B.d.A., da die Gleichung theoretisch geeignet skaliert werden kann.

Betrachten wir zunächst die schwache Formulierung der nicht gesplitteten Stokes-Gleichung:

$$\begin{aligned} X &= H_0^1(\Omega)^2 \ , \ M = L_{2,0}(\Omega) := \{q \in L_2(\Omega); \int_\Omega q \, dx = 0\} \ , && (4.23) \\ a(v,u) &= \int_\Omega \nabla v \cdot \nabla u \, dx \ , && (4.24) \\ b(u,q) &= \int_\Omega (\nabla \cdot u) q \, dx \ . && (4.25) \end{aligned}$$

Gesucht werden $(v,p) \in X \times M$, so dass

$$\begin{aligned} a(v,u) + b(u,p) &= (f,u)_0 \text{ für alle } u \in X \ , \qquad (4.26)\\ b(v,q) &= 0 \text{ für alle } q \in M \ . \qquad (4.27) \end{aligned}$$

Jede klassische Lösung von (4.23) - (4.27) ist auch Lösung der Stokes-Gleichung.[1] Setzt man $W = X \times M$ und $c(v,p) := a(v,u) + b(u,p) + b(v,q)$, erkennt man, dass die Frage nach der Existenz und Eindeutigkeit dieses Problems nicht analog zur Theorie für elliptische, partielle Differentialgleichungen behandelt werden kann, da c über W nicht elliptisch ist. Es ist also im Gegensatz zu einem elliptischen Problem hier nicht das Ziel, das Minimum eines Funktionals wie beim Satz von Lax-Milgram zu bestimmen, sondern einen Sattelpunkt.
Um die Existenz einer Lösung von Sattelpunktproblemen nachweisen zu können, benötigt man einen abstrakteren Existenzsatz als den Satz von Lax-Milgram. Die funktionalanalytische Theorie dieses Satzes wird z.B. in [6], Kapitel III §3 beschrieben.
Mit Hilfe dieser theoretischen Grundlagen sagt der in [6], S.124, Satz 4.3 formulierte Hauptsatz für Sattelpunktprobleme aus, dass, falls die Bilinearform $a(.,.)$ über

$$V := \{u \in X \ ; \ (\nabla \cdot u \ , \ q)_{0,\Omega} = 0 \text{ für } q \in V\}$$

elliptisch ist und die Bilinearform b die inf-sup- oder Babuska-Brezzi-Bedingung

$$\inf_{\mu \in M} \sup_{u \in X} \frac{b(u,\mu)}{\|u\|\|\mu\|} \geq \alpha > 0$$

erfüllt, die Existenz und Eindeutigkeit des Sattelpunktproblems (4.23) - (4.27) garantiert ist. Die Bilinearform a ist über ganz H_0^1 elliptisch, daher ist laut [6], S.140 diese Bedingung erfüllt, ebenso wie die inf-sup-Bedingung für die oben angegebenen Räume.

Bei den Konvergenzbetrachtungen der Finite-Element-Methode für Sattelpunktprobleme wird vorausgesetzt, dass auch die Ansatzräume X_h und M_h diese Bedingungen erfüllen.

Die Elliptizität gilt trivialerweise für konforme Finite-Element-Räume, da die Bilinearform $a(.,.)$ über $H_0^1(\Omega)$ elliptisch ist, wodurch sich diese Eigenschaft auf die Unterräume vererbt. Um die Babuska-Brezzi-Bedingung zu erfüllen, müssen hingegen X_h und M_h geeignet gewählt werden.

Es hat sich herausgestellt, dass diese Bedingung nicht nur von theoretischer Bedeutung ist. Innerhalb der Strömungsmechanik wurde die praktische Erfahrung gemacht, dass die Verwendung von finiten Elementen, welche die Babuska-Brezzi-Bedingung nicht erfüllen, oft zu Instabilitäten in der Lösung führt. Ein populäres Beispiel für solche Elemente ist das $Q_1 - P_0$ Element, vergleiche hierzu [6], S.143f. Um die Einhaltung der Babuska-Brezzi-Bedingung zu garantieren, werden wir in dieser Arbeit das Taylor-Hood-Element verwenden. Es gibt zwei Varianten des Taylor-Hood-Elements. Für die erste, mit den Finite-Element-Räumen

$$\begin{aligned} X_h &= \{v_h \in C(\bar{\Omega})^2 \cap H_0^1(\Omega)^2 \ ; v_{h|T} \in \mathcal{P}_2 \text{ für } T \in \mathcal{T}_h\} \ , \qquad (4.28)\\ M_h &= \{q_h \in C(\Omega) \cap L_{2,0}(\Omega) \ ; q_{h|T} \in \mathcal{P}_1 \text{ für } T \in \mathcal{T}_h\} \qquad (4.29) \end{aligned}$$

---

[1] Siehe Bemerkung 5.2 in [6] oder Bemerkung 12.2.1 in [19]

Abbildung 4.1: Zwei Varianten des Taylor-Hood-Elements. v ist an den Knoten ($\bullet$) und p an den Knoten ($\circ$) gegeben.

werden für die Geschwindigkeit Polynome höheren Grades als für den Druck herangezogen. Die zweite stabile Variante verwendet für die Geschwindigkeit stückweise lineare Funktionen. Um mit diesen ein stabiles Element zu erhalten, wird für die Geschwindigkeit die Triangulierung verwendet, die sich durch die globale reguläre Verfeinerung des Gitters ergibt, welches für den Druck verwendet wird. Hieraus entstehen folgende Finite-Element-Räume:

$$\begin{aligned} X_h &= \{v_h \in C(\bar{\Omega})^2 \cap H_0^1(\Omega)^2 \,; v_{h|T} \in \mathcal{P}_1 \text{ für } T \in \mathcal{T}_{h/2}\} \,, && (4.30) \\ M_h &= \{q_h \in C(\Omega) \cap L_{2,0}(\Omega) \,; q_{h|T} \in \mathcal{P}_1 \text{ für } T \in \mathcal{T}_h\} \,. && (4.31) \end{aligned}$$

Damit ist Folgendes klar: Falls für die Unterräume X_h, M_h bekannt ist, dass sie die Babuska-Brezzi-Bedingung erfüllen, so erfüllen $X_h \subset \hat{X}_h$, M_h sie ebenfalls. Diesen Umstand werden wir uns bei der Entwicklung eines adaptiven Algorithmus zur Verfeinerung im Ort zunutze machen.

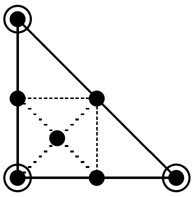


Abbildung 4.2: Taylor-Hood Typ2 mit $X_h \subset \hat{X}_h$, M_h

4.5 Verfahren zur Lösung linearer Gleichungssysteme

Wie schon in den Kapiteln 4.1 und 4.3 erwähnt, werden im Laufe des Splitting-Algorithmus viele große, dünnbesetzte lineare Gleichungssysteme zu lösen sein, die von einer Finite-Element-Diskretisierung herrühren. Es ist daher für einen effektiven Splitting-Algorithmus wesentlich, geeignete Löser zu benutzen. Von einem solchen Löser erwartet man im Wesentlichen zwei Eigenschaften.

Die wichtigste ist sicherlich die Robustheit. Im Fall von reinen Diffusions-Reaktions-Gleichungen existieren eine Reihe von Lösern, die dieses Kriterium erfüllen, nicht so, wenn es sich um eine Konvektions-Diffusions-Reaktions-Gleichung handelt.

Man spricht von einem robusten Verfahren, wenn dessen Konvergenzfaktor nicht nur - wie im Fall reiner Diffusionsgleichungen - unabhängig von h, sondern auch unabhängig von der Péclet-Zahl und von der Art des Konvektionsfeldes k von eins weg beschränkt bleibt, vergleiche auch Jürgen Bey in [3], S.238 ff.

Bei einem robusten Verfahren kann man also davon ausgehen, nach einer endlichen Anzahl von Iterationsschritten die Lösung zu erreichen. Über die Geschwindigkeit dieses Vorgangs sagt der Begriff robust nichts aus. Jürgen Bey schreibt in seinem Buch [3], S.239 hierzu:

> „Für konvektionsdominierte Probleme im $\mathbb{R}^2$, $\mathbb{R}^3$, mit variablen Koeffizienten existiert unseres Wissens nach bisher kein Verfahren, dessen Robustheit wirklich bewiesen wäre - jedenfalls keines mit optimaler oder zumindest quasioptimaler Komplexität.“

Damit kommen wir zur zweiten wesentlichen Eigenschaft der (optimalen) Komplexität. Das bedeutet, dass

1. der Rechenaufwand pro Iterationsschritt höchstens linear mit der Anzahl der Unbekannten wächst und
2. die Konvergenzrate unabhängig von der verwendeten Gitterweite h ist.

Bei einem Verfahren mit optimaler Komplexität steigt entsprechend der Rechenaufwand, der benötigt wird, um den Anfangsfehler in einer geeigneten Norm um einen konstanten Faktor zu reduzieren, nur linear an. Von einem quasioptimalen Verfahren spricht man, falls hier ein logarithmischer Faktor auftritt.

Im Laufe der numerischen Tests wurde mit einer Reihe von Lösern experimentiert. Die Ergebnisse sollen hier kurz zusammengefasst werden. Für die abschließende Version des Splittings wurde dann eine *sparse*-LU-Zerlegung mit UMFPACK zur Lösung der Gleichungssysteme benutzt.

4.5.1 Iterative Löser

Im Folgenden werden zunächst die Erfahrungen mit einigen Verfahren beschrieben, welche sehr robust sind. Im Anschluss wenden wir uns den Multilevel-Verfahren zu, welche bei Problemen ohne Konvektion die oben erwähnte optimale bzw. zumindest quasi optimale Komplexität besitzen, wie in [3], Kapitel 5.2.2 nachzulesen ist. Diese verlieren sie jedoch bei Konvektions-Diffusions-Reaktions-Gleichungen. Die bei Multilevel-Verfahren üblichen Prolongations- und Restriktionsoperatoren werden unabhängig vom Löser innerhalb des Splittingalgorithmus verwendet.

Vorab kann zu allen iterativen Verfahren angegeben werden, dass ihr Konvergenzverhalten deutlich besser war als es die Literatur zu Konvektions-Diffusions-Gleichungen vermuten lässt. Der Grund ist der insbesondere bei kleinen Zeitschrittweiten bedeutende Reaktions-Term.

MINRES, GMRES und restarted GMRES

GMRES Das GMRES-Verfahren (siehe dazu [16]) ist ein Krylov-Raum-Verfahren zur Lösung linearer Gleichungssysteme und kann für beliebige reguläre Matrizen eingesetzt werden. Dazu wird zur Konstruktion einer Orthonormalbasis des Krylov-Raumes die so genannte

Arnoldi-Methode verwendet, welche auf einem Gram-Schmidt-Verfahren über diesem Raum basiert. Die Art und Größe des Krylov-Raumes hängt von der Näherung ab, mit der der Algorithmus gestartet wird. Je besser die Startnäherung ist, desto schneller konvergiert das GMRES-Verfahren. Im Anschluss an die Konstruktion einer Orthonormalbasis des Krylov-Raumes muss dann ein Least-Square Problem gelöst werden.

restarted GMRES Der GMRES-Algorithmus ist robust, jedoch wachsen die Rekursionen beim Arnoldi-Verfahren linear mit der Dimension des Krylov-Raums. Im schlechtesten Fall kann die Dimension des Krylov-Raums die Größe des Finite-Element-Problems erreichen, wodurch schnell die Grenzen des Speichers erreicht sind und die Lösung rechnerisch zu aufwendig wird. Daher verwendet man in der Regel eine *restarted* Version des GMRES-Algorithmus. Dazu wird der GMRES-Algorithmus mit einer festen Krylov-Raum-Dimension gestartet und nur über diesem Raum wird das Least-Square-Problem gelöst. Ein solches Verfahren wird oft in der Form GMRES(Krylov-Raum-Dimension) angegeben. Mit der daraus gewonnenen Näherung wird das GMRES erneut gestartet.
Wir haben mit der Variante GMRES(60) experimentiert, welche auf unseren Testproblemen robust war. Jedoch benötigte der GMRES(60) mit steigender Péclet-Zahl wesentlich mehr Iterationen und war als Löser für Péclet-Zahl zwischen 10^3 und 10^5 zu aufwendig. Der Wert der Péclet-Zahl, ab dem das Verfahren in Hinblick auf die CPU-Kosten unbrauchbar wird, variiert, da auch die Natur des Testproblems von Bedeutung ist.

MINRES Ist klar, dass es sich bei der Matrix A um eine symmetrische handelt, kommt man mit kurzen Rekursionen, d.h. nicht mit der Dimension des Krylov-Raums wachsenden Dimensionen, und folglich mit einem wesentlich geringeren Speicherbedarf aus. Die für symmetrische Matrizen spezialisierte Variante des Arnoldi-Algorithmus heißt Lanczos-Verfahren und die daraus resultierende Variante des GMRES-Verfahrens heißt MINRES. Eine detaillierte Darstellung des MINRES-Verfahrens findet sich in [16], Kapitel 2.5.
Interessant für die Lösung der beim Splitting auftretenden Druckaktualisierungsgleichung ist die Fähigkeit des MINRES-Algorithmus, indefinite Probleme zu lösen. Ebenso wie der GMRES-Algorithmus ist das MINRES-Verfahren eine robuste Methode zur Lösung von linearen Gleichungssystemen.
Der MINRES-Algorithmus hätte aus numerischer Sicht weiter bei der Implementierung zur Lösung der Druckkorrekturgleichung verwendet werden können. Da er jedoch bei der von uns verwendeten Anzahl von Freiheitsgraden keine wesentlichen Vorteile gegenüber einer Behandlung mit einer *sparse*-LU-Zerlegung mit UMFPACK aufwies, wurde auf die Verwendung des MINRES-Algorithmus im Hinblick auf eine einfachere Wartung des Programm-Codes verzichtet.

Mehrgitter-Verfahren mit Gauß-Seidel-Glätter Für Probleme ohne Konvektion haben sich hier die so genannten Mehrgitter-Verfahren (engl. multigrid) als Löser mit einer optimalen Komplexität bewährt.
Dabei machen sich die Mehrgitter-Verfahren die in [27], Kapitel 5.1.2 oder [6], S.201f beschriebenen Glättungseigenschaften der klassischen Iterationsverfahren, wie zum Beispiel des Gauß-Seidel-Verfahrens, zunutze. Damit das Gauß-Seidel-Verfahren konvergiert, muss die Matrix A einige Bedingungen erfüllen. Ein hinreichendes Kriterium für die Konvergenz des Gauß-Seidel-Verfahrens liefert der folgende Satz.

Satz 4.15
Das Gauß-Seidel-Verfahren konvergiert global und monoton bzgl. $\|.\|_\infty$, wenn das starke Zeilensummenkriterium erfüllt ist:

$$\sum_{j=1;j\neq i}^{m} |a_{ij}| < |a_{ii}| \; \forall \; i = 1, ..., m$$

Beweis: Siehe [27], S.197, Satz 5.3.

Weitere Details zu diesem Themenkomplex findet man in [27], Kapitel 5.1.2.

Für Probleme mit verschwindendem Konvektionsanteil ist das Gauß-Seidel-Verfahren für die aus dem Galerkin-Verfahren resultierenden Matrizen folglich als Glätter geeignet. Grundlage für die Mehrgitter-Verfahren ist eine Menge von Ansatzräumen

$$V_0 \subset V_1 \subset V_2 ... V_n \subset H_0^1(\Omega) \; ,$$

die ineinander verschachtelt sind und z.B. durch Verfeinerungsprozesse, wie sie in Kapitel 7.2 beschrieben werden, entstehen.
Zum Übergang zwischen den Gittern benötigt man den Prolongations- sowie den Restriktionsoperator. Die Prolongation bildet von V_i nach V_{i-1} ab, die Restriktion umgekehrt von V_{i-1} nach V_i. Die Knoten des gröberen Gitters werden im Folgenden mit $\mathcal{I}_H$ und die des verfeinerten Gitters mit $\mathcal{I}_h$ bezeichnet. Die neu hinzugekommenen Knoten erhält man durch $\mathcal{J} = \mathcal{I}_h \backslash \mathcal{I}_H$.

Ein geeigneter Prolongations- und Restriktionsoperator entsteht, indem wir die Darstellung der Basisvektoren $\phi_{H,i}$ in der Basis $\Phi_h := \{\phi_{h,k} \; : \; k \in \mathcal{I}_h\}$ des Raumes V_h benutzen. Dies ist möglich, da V_h durch eine Verfeinerung aus V_H hervorgegangen ist, also $V_H \subset V_h$ ist.

Bezeichnen wir nun mit $\mathcal{N}_k \subset \mathcal{J}$ die benachbarten Knoten von $k \in \mathcal{I}_H$ in $\mathcal{I}_h$. Mit diesen Bezeichnungen erhält man für die Basisfunktion der gröberen Triangulierung folgende Darstellung:

$$\phi_{H,k} = \sum_{m \in \{k\} \cup \mathcal{N}_k} \phi_{H,k}(m) \phi_{h,m} \; .$$

Daraus ergibt sich für die Prolongationsmatrix P

$$P_{m,k} = \phi_{H,k}(m), \quad m \in \mathcal{I}_h \; , k \in \mathcal{I}_H \tag{4.32}$$

und entsprechend wählen wir den Restriktionsoperator nach [27], S.238 gleich P^t.

Beim Splitting ohne Postprocessing verwenden wir lineare finite Elemente in der Variante des in Kapitel 4.4 beschriebenen Taylor-Hood-Elements, so dass wir schon zum Startzeitpunkt des Splittings über zwei Gitter verfügen, wobei das zweite durch eine reguläre Verfeinerung des ersten entstanden ist. Hier bietet sich die Verwendung eines Mehrgitter-Verfahrens an. Entstand das Druckgitter seinerseits durch mehrfache Verfeinerungen, so können auch tiefere V-Zyklen benutzt werden.

Mit zunehmender Bedeutung des Konvektionsanteils verliert das Gauß-Seidel-Verfahren seine Glättungseigenschaft, bis schließlich das gesamte Mehrgitter-Verfahren nicht mehr konvergiert. Das Verfahren konvergiert folglich bei einer impliziten Behandlung des nichtlinearen Terms abhängig von dem Testproblem und der Zeitschrittweite für Péclet-Zahlen größer 10^2 nicht mehr.
Der Grund für das gute Konvergenzverhalten auch bei recht großen Péclet-Zahlen liegt in der Bedeutung des Reaktionsterms, insbesondere bei kleinen Zeitschrittweiten. Dadurch werden die Glättungseigenschaften im Sinne von Satz 4.15 verbessert.

Eine Alternative zum traditionellen Multilevel-Verfahren mit Gauß-Seidel-Glätter besteht im Einsatz von einem speziell für konvektionsdominante Probleme entwickelten Multilevel-Verfahren, wie z.B. dem im Artikel [39] von Gerhard Starke vorgestellten.

4.5.2 *sparse*-LU-Zerlegung mit UMFPACK

Die Verwendung des Programmpaketes UMFPACK erlaubt die direkte Lösung von linearen Gleichungssystemen $Ax = b$ mit einer dünnbesetzten und unsymmetrischen Matrix A. Das Paket wird von Tim Davis von der University of Florida entwickelt und in MATLAB[2] eingesetzt, kann jedoch auch separat von diversen Programmiersprachen aus genutzt werden.

Die von UMFPACK zur LU-Faktorisierung von Matrizen eingesetzte Methode versucht, geeignete Permutationen zu finden, um bei der Zerlegung neue Nicht-Null-Einträge zu vermeiden.

Eine optimale Sortierung hierfür zu finden, die auch noch die numerische Stabilität garantiert, ist ein NP-vollständiges Problem. Daher verwendet UMFPACK heuristische Ansätze zur Lösung dieses Problems. Die Matrix A wird analysiert und eine von drei Strategien *unsymmetrisch, 2x2* und *symmetrisch* zum Sortieren der Zeilen und Spalten ausgewählt. Dabei ist symmetrisch nicht direkt mit der entsprechenden Matrixeigenschaft gleichzusetzen. Die symmetrische Strategie wird auch bei näherungsweise symmetrischen Matrizen verwendet.
Weitere technische Details zum Vorgehen des UMFPACK-Programmpaketes findet man im „User Guide UMFPACK Version 4.4“ [9] und in den Veröffentlichungen von Tim Davis zu diesem Algorithmus, so z.B. in [10].

Selbst einem sehr intelligenten direkten Löser wie UMFPACK sind natürlich im Gegensatz zu iterativen Verfahren durch den benötigten Speicher und die mangelnde quasi optimale Komplexität engere Grenzen für die Lösung sehr großer Gleichungssysteme gesetzt.

Die quadratischen finiten Elemente zeichnen sich gegenüber den linearen bei gleicher Anzahl an Freiheitsgraden durch eine dichtere Besetzung der Matrix A aus. Allerdings war bei beiden der UMFPACK-Löser bei den von uns behandelten Problemen mit weniger als 600.000 Freiheitsgraden allen anderen Verfahren in zweierlei Hinsicht überlegen:
Erstens in Hinblick auf die Robustheit und zweitens auf die Lösungsgeschwindigkeit, beson-

---

[2] Version 4.3 ist als eingebaute Routine in MATLAB 7.1. enthalten, die Version 4.0 UMFPACK war in MATLAB 6.5 integriert. Die in dieser Arbeit präsentierten numerischen Ergebnisse wurden mit Hilfe der UMFPACK Version 4.4 vom 28 Januar 2005 erzeugt.

ders in den konvektionsdominierten Fällen. Bei den im Fall einer expliziten Behandlung des nichtlinearen Terms auftretenden Helmholtz-Gleichungen war der Unterschied zwischen den iterativen Verfahren und dem UMFPACK-Löser geringer, jedoch immer noch erkennbar. Der Grund hierfür ist sicherlich u.a. in der optimierten Implementierung und der Verwendung einer ebenfalls optimierten BLAS-Variante, ATLAS (Automatically Tuned Linear Algebra Software), zu suchen. Daher haben wir uns zur Verwendung von UMFPACK als Löser für alle im Splitting auftretenden dünnbesetzten Gleichungssysteme entschlossen.

Durch die Verwendung von UMFPACK als Löser für die entstehenden Gleichungssysteme besteht die Gefahr, dass die Geschwindigkeitsvorteile einer expliziten Behandlung verschleiert werden. Deshalb sollte festgehalten werden, dass mit einer wachsenden Zahl an Unbekannten sowie für eine dichtere Besetzung der Matrix A, wie sie zum Beispiel im dreidimensionalen Fall auftritt, die Kosten zur Lösung bei der impliziten Behandlung gegenüber der expliziten stärker ansteigen werden.

5 Entwicklung einer Splitting-Methode für die finiten Elemente

5.1 Verfahren zur Gradientenrekonstruktion

An vielen Stellen des Splittingalgorithmus muss der Gradient einer Funktion u_h aus dem Finite-Element-Raum gebildet werden. Auf Grund der Natur dieses Raumes

$$V_h = \{u_h \in C(\bar{\Omega})\ ; u_{h|T} \in \mathcal{P}_1 \text{ für } T \in \mathcal{T}_h\}$$

ist die Bildung des Gradienten an den Knoten nicht direkt möglich.
Die Idee, einen Raum so zu wählen, dass $u_h \in C^1(\bar{\Omega})$ ist, also die in [6], S.62f beschriebenen C^1-finiten Elemente zu verwenden, erweist sich als problematisch, da in diesem Fall zusätzlich Randbedingungen benötigt werden.
Wir werden daher auf ein Verfahren, das durch den Operator $R_h u_h \approx \nabla u$, $R_h : V_h \to V_h \times V_h$ dargestellt wird, zurückgreifen. Ein solches Verfahren wird als Gradientenrekonstruktion bezeichnet. Für viele Aufgaben hat sich ein recht einfaches Verfahren bewährt, die Rekonstruktion über Mittelung, die z.B. in [27], Kapitel 4.2, S.182f vorgestellt wird.

5.1.1 Rekonstruktion über Mittelung

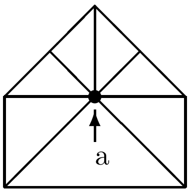


Abbildung 5.1: Dreiecksumgebung von a

Für einen Knoten a sei die Menge $\triangle(a) := \bigcup_{K\,:\,a\in\partial K} K$ die Dreiecksumgebung von a (siehe Abbildung 5.1). Die Menge entspricht dem Träger der Basisfunktion am Knoten a. Um den Gradienten einer Funktion $u_h \in V_h$ an jedem Knoten a der Triangulierung $\mathcal{I}_h$ zu konstruieren, bildet man eine Mittelung aus den Werten der Gradienten aller Elemente aus $\triangle(a)$. Weniger einfach ist die Frage nach einer geeigneten Mittelung zu beantworten. In [27] wird mit Hilfe der Größe von $\triangle(a)$ gemittelt:

$$R_h u_h(a) := \frac{1}{|\triangle(a)|} \sum_{K\subset\triangle(a)} |K| \nabla u_h|_K \ .$$

Eine Alternative besteht darin, über die Anzahl der Elemente n in $\triangle(a)$ zu mitteln:

$$R_h u_h(a) := \frac{1}{n} \sum_{K \subset \triangle(a)} \nabla u_h|_K \ .$$

An Abbildung 5.1 sieht man die Schwäche beider Verfahren. Sei a z.B. der Punkt (0,0) und die Funktion, deren Gradienten es zu rekonstruieren gilt, lautet $f(x,y) := y^2$. Wird die erste Form der Mittelung verwendet, wird sich für f_y in $(0,0)$ eine negative Zahl ergeben, da die drei Dreiecke unterhalb von a eine größere Fläche aufweisen. Mittelt man hingegen auf der Basis der Anzahl der Dreiecke, so ist es ebenfalls unmöglich, das korrekte Ergebnis Null zu erhalten, da sich oberhalb von a vier Dreiecke befinden, unterhalb jedoch nur drei.

5.1.2 Rekonstruktion nach Zienkiewicz und Zhu

Eine Variante, die Frage nach der geeigneten Mittelung zu lösen, besteht in der Umformulierung in ein Minimierungsproblem, wie es in dem Artikel [45] von Zienkiewicz/Zhu in dem so genannten Z^2-Verfahren geschieht. Das Ziel ist es, eine lineare Ausgleichsfunktion für jede Komponente von ∇u_h bzgl. der Werte von ∇u_h an den Schwerpunkten x_S der Dreiecke zu bilden. Mit $p(x) = [1, x_1, x_2]^T$ muss nun für die Komponenten von ∇u_h ein Vektor $c^{(i)} = [c_0^{(i)}, c_1^{(i)}, c_2^{(i)}]$ $i = (1,2)$ bestimmt werden, so dass

$$c^{(i)} = \min_{\hat{c}^{(i)} \in \mathbb{R}^3} \sum_{K \subset \triangle(a)} \left(\frac{\partial u_h(x_S)}{\partial x_i} - p(x_S) \cdot \hat{c}^{(i)} \right)^2 \ .$$

Der gemittelte Gradient in a ergibt sich damit durch $R_h u_h(a) = p(a)^T [a^{(1)}, a^{(2)}]$. Das Z^2-Verfahren erfordert folglich wie der vorangegangene Mittelungsoperator die Bildung des Gradienten auf jedem Element und darüber hinaus die Lösung eines Ausgleichsproblems, z.B. mittels einer QR-Zerlegung für jeden Knoten.

5.1.3 Rekonstruktion mit Taylor-Ansatz

Das Z^2-Verfahren ist zunächst auf lineare finite Elemente beschränkt und verliert für die Randpunkte des Gebietes Ω stark an Genauigkeit. Um den Gradienten am Rand besser zu erfassen und um unabhängig von der Höhe des Polynomgrades über den finiten Elementen und der Verwendung von entweder Dreiecken oder Quadraten zu sein, wurde für diese Arbeit ein neuer Mittelungsoperator entwickelt, der auf einem Taylor-Ansatz im Punkt a basiert. Hierzu nehmen wir an, dass es sich bei der Funktion u, welche durch u_h approximiert wird, lokal um eine C^2-Funktion handelt. (x_j, y_j) $(j = 1...m)$ $m \geq 5$ seien Punkte in der Umgebung von a, an denen Näherungswerte für u durch die Funktion u_h vorhanden sind. Die Menge all dieser Punkte bezeichnen wir mit $\hat{U}$. Eine mögliche Wahl sind alle Nachbarn von a und wiederum deren Nachbarn. Bei quadratischen Ansatzfunktionen kann man in der Regel auf die Nachbarn der Nachbarn verzichten.
Damit lässt sich in a die folgende Gleichung mit Hilfe des Taylorpolynoms aufstellen:

$$\begin{aligned} T_2(x_j, y_j) - u(x_a, y_a) &\approx u_h(x_j, y_j) - u_h(x_a, y_a) = \mathbf{u_x}(\mathbf{x_a}, \mathbf{y_a})(x_j - x_a) + \mathbf{u_y}(\mathbf{x_a}, \mathbf{y_a})(y_j - y_a) \\ &+0.5[\mathbf{u_{xx}}(\mathbf{x_a}, \mathbf{y_a})(x_j - x_a)^2 + \mathbf{u_{xy}}(\mathbf{x_a}, \mathbf{y_a})(x_j - x_a)(y_j - y_a) + \mathbf{u_{yy}}(\mathbf{x_a}, \mathbf{y_a})(y_j - y_a)^2] \end{aligned}$$

Die fett gedruckten Terme werden über die Lösung eines Least-Squares-Problems mit einer $(m \times 5)$-Matrix ermittelt.

Bemerkung 5.1
Wird diese Gradientenrekonstruktion für jeden Freiheitsgrad eines Elementes u_h *des* n *dimensionalen Finite Element Raums* V_h *durchgeführt, so notieren wir dies als* $G_T u_h$*.* G_T *als Operator hat die folgenden Eigenschaften:*

- G_T *ist eine Abbildung von* $V_h \to V_h \times V_h$
- G_T *ist durch seine Konstruktion über* n *lineare Ausgleichsprobleme linear*
- G_T *ist als linearer Operator über endlichdimensionalen Vektorräumen Lipschitz-stetig*

Es ist wichtig, mehr über die Abhängigkeit der Lipschitzkonstante von h zu erfahren. Dazu betrachten wir einen Ausschnitt einer so genannten Friedrichs-Keller-Triangulierung des Einheitsquadrates, vgl. [27], S.57. Eine solche Triangulierung wird im weiteren auch als kanonische Zerlegung oder kanonische Triangulierung eines Rechtecks bezeichnet. Die Abbildungen 5.2 skizziert den Fall der Rekonstruktion des Gradienten im Inneren mit Hilfe der Nachbarknoten.

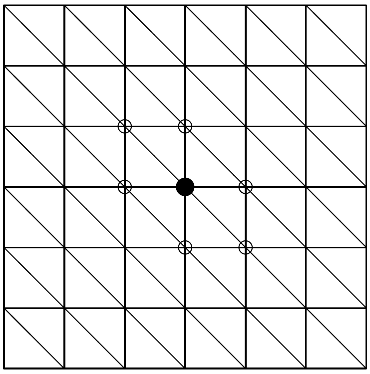


Abbildung 5.2: Bildung von G_T im Inneren von Ω mit Hilfe der Nachbarknoten

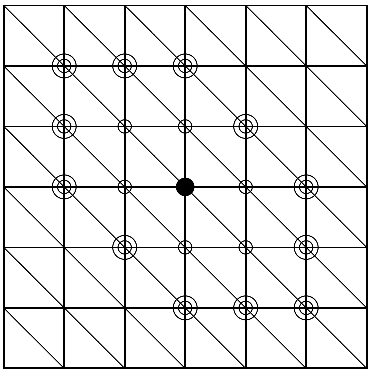


Abbildung 5.3: Bildung von G_T im Inneren von Ω mit Hilfe der Nachbarknoten und deren Nachbarknoten

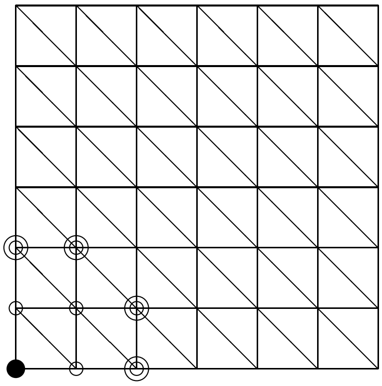


Abbildung 5.4: Bildung von G_T auf $\partial\Omega$ mit Hilfe der Nachbarknoten und deren Nachbarknoten

In Abbildung 5.3 werden zusätzlich noch die Nachbarn der Nachbarknoten verwendet. Die Abbildung 5.4 veranschaulicht den schwierigsten in der Ecke des Quadrats. Bezeichet man nun mit h die Länge der Kathetenseiten der Dreiecke, so ergeben sich für die zu lösenden Ausgleichsprobleme die folgenden Matrizen:

$$A_{5.2} = \begin{pmatrix} h & 0 & \frac{h^2}{2} & 0 & 0 \\ -h & 0 & \frac{h^2}{2} & 0 & 0 \\ 0 & h & 0 & 0 & \frac{h^2}{2} \\ 0 & -h & 0 & 0 & \frac{h^2}{2} \\ -h & h & \frac{h^2}{2} & -\frac{h^2}{2} & \frac{h^2}{2} \\ h & -h & \frac{h^2}{2} & -\frac{h^2}{2} & \frac{h^2}{2} \end{pmatrix}$$

$$A_{5.4} = \begin{pmatrix} h & 0 & \frac{h^2}{2} & 0 & 0 \\ 0 & h & 0 & 0 & \frac{h^2}{2} \\ 2h & 0 & 2h^2 & 0 & 0 \\ 0 & 2h & 0 & 0 & 2h^2 \\ h & h & 2h^2 & 2h^2 & 2h^2 \\ 2h & h & 2h^2 & h^2 & \frac{h^2}{2} \\ h & 2h & \frac{h^2}{2} & h^2 & 2h^2 \end{pmatrix}$$

$$A_{5.3} = \begin{pmatrix} h & 0 & \frac{h^2}{2} & 0 & 0 \\ -h & 0 & \frac{h^2}{2} & 0 & 0 \\ 0 & h & 0 & 0 & \frac{h^2}{2} \\ 0 & -h & 0 & 0 & \frac{h^2}{2} \\ -h & h & \frac{h^2}{2} & -\frac{h^2}{2} & \frac{h^2}{2} \\ h & -h & \frac{h^2}{2} & -\frac{h^2}{2} & \frac{h^2}{2} \\ h & h & \frac{h^2}{2} & \frac{h^2}{2} & \frac{h^2}{2} \\ -h & -h & \frac{h^2}{2} & \frac{h^2}{2} & \frac{h^2}{2} \\ -2h & 2h & 2h^2 & -2h^2 & 2h^2 \\ 2h & -2h & 2h^2 & -2h^2 & 2h^2 \\ 2h & 0 & 2h^2 & 0 & 0 \\ 0 & 2h & 0 & 0 & 2h^2 \\ -2h & 0 & 2h^2 & 0 & 0 \\ 0 & -2h & 0 & 0 & 2h^2 \\ 2h & -h & 2h^2 & -h^2 & \frac{h^2}{2} \\ -2h & h & 2h^2 & -h^2 & \frac{h^2}{2} \\ h & -2h & \frac{h^2}{2} & -h^2 & 2h^2 \\ -h & 2h & \frac{h^2}{2} & -h^2 & 2h^2 \end{pmatrix}$$

Lemma 5.2

Sei $a \in \mathcal{T}_h$ und $v_h, w_h \in V_h$. Des Weiteren sei $\mathcal{I}_a$ eine Indexmenge mit der Mächtigkeit m, durch welche alle $x_i \in \mathcal{T}_h$, die zur Berechnung der Gradientenrekonstruktion in a benutzt werden, bestimmt sind. $[G_T x_h(a)]_i$ ist die i-te Komponente des rekonstruierten Gradienten. Mit diesen Bezeichnungen gilt:

$$|\,[G_T u_h(a)]_i - [G_T w_h(a)]_i\,| \leq 2m \frac{\mathcal{K}_2(A)}{\|A\|_2} |u_h(\mathcal{I}_a \cup \{a\}) - w_h(\mathcal{I}_a \cup \{a\})|_{\max} \quad (i = 1, 2) \qquad (5.1)$$

Beweis :

Wir verwenden die folgenden Bezeichnungen; $x := G_T u_h(a)$ und $\tilde{x} := G_T w_h(a)$. Sei $e = (1, ..., 1)$ ein Vektor der Länge m. Zur Berechnung der rekonstruierten Gradienten wird ein Ausgleichsproblem $Ax = u_h(\mathcal{I}_a) - u_h(a)e$ bzw. $A\tilde{x} = w_h(\mathcal{I}_a) - w_h(a)e$ gelöst. Mit Hilfe der Pseudoinversen erhält man : $x = (A^tA)^{-1}A^t(u_h(\mathcal{I}_a) - u_h(a)e)$ und $\tilde{x} = (A^tA)^{-1}A^t(w_h(\mathcal{I}_a) - w_h(a)e)$. Mit der bekannten Identität, s. z.B. [12] S.74, $\frac{\mathcal{K}_2(A)}{\|A\|_2} = \|(A^tA)^{-1}A^t\|_2$ folgt nun:

$$\begin{aligned} |\,[G_T u_h(a)]_i - [G_T w_h(a)]_i\,| &= |x_i - \tilde{x}_i| \leq \|x - \tilde{x}\|_{\max} \leq \|x - \tilde{x}\|_2 \\ &= \|(A^tA)^{-1}A^t(u_h(\mathcal{I}_a) - w_h(\mathcal{I}_a) + (w_h(a) - u_h(a))e\|_2 \\ &\leq \|(A^tA)^{-1}A^t\|_2(\|u_h(\mathcal{I}_a) - w_h(\mathcal{I}_a)\|_2 + m|w_h(a) - u_h(a)|) \\ &\leq \frac{\mathcal{K}_2(A)}{\|A\|_2} 2m|u_h(\mathcal{I}_a \cup \{a\}) - w_h(\mathcal{I}_a \cup \{a\})|_{\max} \end{aligned}$$

□

Das Lemma 5.2 macht klar, dass der Faktor $\frac{\mathcal{K}_2(A)}{\|A\|_2}$ wichtig für die Libschitzstetigkeit von G_T ist und damit, wie wir später sehen werden, auch für die Qualität der Gradientenrekonstruktion ingesamt.

h	$A_{5.2}$	$A_{5.3}$	$A_{5.4}$
1/2	9.5	2.0	6.3
1/4	38.0	8.3	23.3
1/8	152.0	33.5	91.2
1/16	608.2	134.3	362.7
1/32	2433.1	537.5	1448.9

Tabelle 5.1: $\frac{\mathcal{K}_2(A)}{\|A\|_2}$

Bemerkung 5.3
Für Friedrichs-Keller-Triangulierung ist klar, dass $\frac{\mathcal{K}_2(A)}{\|A\|_2} = C_1(a)O(h^{-2})$ *gilt. Die Entwicklung von* $\frac{\mathcal{K}_2(A)}{\|A\|_2}$ *für die verschiedenen Fälle kann der Tabelle 5.1 entnommen werden.*

Bemerkung 5.4
Die Aussage $\frac{\mathcal{K}_2(A)}{\|A\|_2} = C_1(a)O(h^{-2})$ *lässt sich auf andere Triangulierungen übertragen. Die Konstanten* $C_1(a)$ *können auf Grund der Struktur von* $\mathcal{T}_h$ *stark variieren.*

Für den praktischen Einsatz ist die Größe der Konstanten wichtig. An Hand der Tabelle 5.1 erkennt man, dass eine breitere Datenbasis zu einer Verringerung von $\frac{\mathcal{K}_2(A)}{\|A\|_2}$ führt. Dies führt auch unter Berücksichtigung von m zu seiner besseren Abschätzung in Lemma 5.2.

Darüber hinaus ist die Abschätzung in Lemma 5.2 pessimistisch, unter anderem in Hinblick auf die Mächtigkeit der Menge m und damit auf die Größe des zur Gradientenrekonstruktion verwendeten Patches. Im praktischen Einsatz verbessert sich das Konvergenzverhalten durch die zusätzliche Verwendung der Nachbarn der Nachbarn weit mehr als Bemerkung 5.3 bzw. Tabelle 5.1 vermuten lassen.

Mit Hilfe von Lemma 5.2 kann nun der folgende Satz bewiesen werden.

Satz 5.5
Auf einer Friedrichs-Keller-Triangulierung gilt für die Libschitzstetigkeit von G_T

$$\|[G_T u_h]_i - [G_T w_h]_i\|_{\max} \leq \frac{\tilde{C}}{h^2}\|u_h - w_h\|_{\max} \quad (i = 1, 2) \tag{5.2}$$

$$\|G_T u_h - G_T w_h\|_0 \leq \frac{\hat{C}}{h^2}\|u_h - w_h\|_0 \tag{5.3}$$

Beweis :
Seien A_i die Matrix zum Ausgleichsproblem am i-ten Knoten und m_i die entsprechende Mächtigkeit. Dann sei

$$\frac{\tilde{C}}{h^2} = 2 \max_{j=1..n} \frac{\mathcal{K}_2(A_i)}{\|A_i\|_2} \cdot \max_{j=1..n} m_i \ ,$$

so dass $\tilde{C}$ unabhängig von h ist. Dann gilt mit Lemma 5.2

$$\begin{aligned} \|[G_T u_h]_i - [G_T w_h]_i\|_{\max} &= \max_{i=1..n} \mid [G_T u_h(a)]_i - [G_T w_h(a)]_i \mid \\ &\leq \frac{\tilde{C}}{h^2} \|u_h - w_h\|_{\max} \; . \end{aligned}$$

Durch die Äquivalenz der Normen in dem endlich dimensionalen Vektorraum V_h gilt damit auch nach wenigen Umformungen die Abschätzung 5.3.

□

Der folgende Satz ist bei der Analyse der auf das Gradientenrekonstruktionsverfahren aufbauenden Algorithmen hilfreich.

Satz 5.6
Sei $u_h \in V_h$ eine Näherung an eine Funktion $u \in V$. Diese Näherung sei mit Fehlern behaftet, die unabhängig von der Triangulierung sind, in diesem Fall auf Grund einer Zeitdiskretisierung mit der Zeitschrittweite $\triangle t$. Für diese Näherug gelte nun $\|Iu - u_h\|_0 \leq O(h^s) + O(\triangle t^k)$. Des Weiteren gelte für G_T $\|\nabla u - G(Iu)\|_0 \leq C_1 O(h^l)$, wobei $I : V \to V_h$ hier wieder den Interpolationsoperator bezeichnet.
Dann gilt:

$$\|\nabla u - Gu_h\|_0 \leq C_1 O(h^l) + C_2 O(h^{s-2}) + C_3 O(\triangle t^k) \cdot O(h^{-2}) \; . \tag{5.4}$$

Beweis:
Mit den Eigenschaften von G_T aus Bemerkung 5.1 und der Abschätzung aus Satz 5.5 erhält man direkt:

$$\begin{aligned} \|\nabla u - Gu_h\|_0 &= \|\nabla u - G(Iu) + G(Iu) - Gu_h\|_0 \leq \|\nabla u - G(Iu)\|_0 + \|G(Iu) - Gu_h\|_0 \\ &\leq C_1 O(h^l) + \|G(Iu - u_h)\|_0 \leq C_1 O(h^l) + \frac{\tilde{C}}{h^2} \|Iu - u_h\|_0 \\ &\leq C_1 O(h^l) + C_2 O(h^{s-2}) + C_3 O(\triangle t^k) \cdot O(h^{-2}) \; . \end{aligned}$$

□

Bemerkung 5.7 (Satz 5.4 für lineare und quadratische finite Elemente)
Für lineare finite Elemente gilt $s = 2$, für quadratische bei hinreichend großer Regularität $s = 3$. Für zeitunabhängige Probleme ergibt sich damit aus der Abschätzung 5.4 für die linearen finiten Elemente

$$\|\nabla u - Gu_h\|_0 \leq C_1 O(h^l) + C_2$$

und für die quadratischen finiten Elemente bei ausreichender Regularität

$$\|\nabla u - Gu_h\|_0 \leq C_1 O(h^l) + C_2 O(h) \; .$$

Für Näherungen mit einem durch eine Zeitdiskretisierung verursachten Fehler erhalten wir hingegen für lineare Ansätze

$$\|\nabla u - Gu_h\|_0 \leq C_1 O(h^l) + C_2 + C_3 O(\triangle t^k) \cdot O(h^{-2})$$

und für quadratische

$$\|\nabla u - Gu_h\|_0 \leq C_1 O(h^l) + C_2 O(h) + C_3 O(\triangle t^k) \cdot O(h^{-2}) \; .$$

Bemerkung 5.8 (Größenordnung von C_2)
Im folgenden Kapitel wird klar, dass C_2 in der Regel sehr klein ist, so dass man im Allgemeinen von $C_2 \approx 0$ ausgehen kann.

Bemerkung 5.9 (Praktische Konsequenzen von Satz 5.4)
Der Term $C_3 O(\triangle t^k) \cdot O(h^{-2})$ in der obigen Abschätzung bedarf einer weiteren Erörterung. Zum einen ist klar, dass bei einer festen Triangulierung dieser Term von der Ordnung k gegen Null geht.
Werden hingegen sowohl h als auch $\triangle t$ verfeinert, kommt es laut der obigen Abschätzung darauf an, dies zumindestens gleichmäßig zu tun. Eine Reduktion dieses Terms ist in der Theorie erst dann sichergestellt, wenn $\triangle t$ stärker reduziert wird als h.

Im praktischen Gebrauch kommt es primär auf die Natur des durch die Zeitdiskretisierung erzeugten Fehlers an. Ist der Fehler langwellig, so ist der Term unabhängig von h von geringer Bedeutung. Ist der Fehler hingegen kurzwellig, so wird es bei einer alleinigen Verfeinerung der Triangulierung zu einer Verschlechterung der genäherten Gradienten kommen. Dies ist jedoch für den praktischen Gebrauch keine Einschränkung, da es bei großen durch die Zeitdiskretisierung verursachten Fehlern nicht sinnvoll ist, die Triangulierung zu verfeinern.

Anpassung der Gradientenrekonstruktion G_T auf adaptiv verfeinerte Gitter Mit Hilfe einer im Folgenden beschriebenen Gewichtung sollen die oben erwähnten Ausgleichsprobleme so angepasst werden, dass sie auf adaptiv verfeinerten Gittern bessere Resultate liefern. Zunächst bewirkt eine Gewichtung, dass die errechneten Werte am Rand sowie an Übergangsbereichen, die durch lokale Verfeinerung entstehen, verbessert werden. Hierzu werden die Anzahl der Punkte danach unterschieden, ob sie oberhalb $U_1 := \{x_j, y_j | y_j > y_a\}$ oder unterhalb $U_2 := \{x_j, y_j | y_j < y_a\}$ bzw. entsprechend rechts $U_3 := \{x_j, y_j | x_j > x_a\}$ oder links $U_4 := \{x_j, y_j | x_j < x_a\}$ von a liegen. Es ist klar, dass

$$\hat{U} = \bigcup_{i=1}^{4} U_i$$

ist, mit $U_1 \bigcap U_2 = \emptyset$ und $U_3 \bigcap U_4 = \emptyset$. Des Weiteren benötigen wir noch die folgenden Bezeichnungen:

$$h_{1j} = x_j - x_a \qquad h_{2j} = y_j - y_a \qquad d_j = \frac{1}{\sqrt{h_{1j}^2 + h_{2j}^2}}\ , \ (j = 1...m)$$

$$g_{1j} = \frac{h_{1j}}{d_j} \qquad g_{2j} = \frac{h_{2j}}{d_j} \qquad \alpha_i = \begin{cases} 1\ , \text{ falls } (x_j, y_j) \in U_i \\ 0\ , \text{ falls } (x_j, y_j) \notin U_i \end{cases}$$

$$G_1 = \sum_{(x_j,y_j)\in U_1} |g_{1j}| \quad G_2 = \sum_{(x_j,y_j)\in U_2} |g_{1j}| \quad G_3 = \sum_{(x_j,y_j)\in U_3} |g_{2j}| \quad G_4 = \sum_{(x_j,y_j)\in U_4} |g_{2j}|\ .$$

Außerdem werden die Gleichungen jeweils mit $\frac{1}{d_j}$ skaliert, um Punkten, die sich näher an a befinden, innerhalb der Ausgleichsrechnung ein größeres Gewicht zu geben. Dies ist auch auf sehr regelmäßigen Gittern wie bei der Friedrichs-Keller-Triangulierung von Vorteil. Damit ergibt sich schließlich als Gesamtgewicht für die Gleichung j:

$$s_j = \sum_{i=1}^{4} \alpha_i \frac{|h_{1j}| + |h_{2j}|}{d_j G_i} \ , \ (j = 1 ... m) \ .$$

Superkonvergente Gradientenrekonstruktion Ein großer Vorteil vieler Verfahren zur Gradientenrekonstruktion, die wir im Folgenden mit den Operatoren G_M (Rekonstruktion über Mittelung), G_Z (Rekonstruktion nach Zienkiewicz und Zhu) und G_T (Rekonstruktion mit Taylor-Ansatz) bezeichnen werden, sind ihre Superkonvergenzeigenschaften. Von Superkonvergenz spricht man, wenn die mit Hilfe des rekonstruierten Gradienten Gu_h erzeugte Funktion asymptotisch schneller gegen ∇u konvergiert als ∇u_h. Die Superkonvergenzeigenschaft konnte für den Operator G_Z unter recht eingeschränkten Voraussetzungen an die Zerlegung $\mathcal{T}_h$ von Ω und an die Glattheit der Lösung v von Zienkiewicz und Zhu in dem Artikel [45] gezeigt werden.

Freiheitsgrade	Fehler in der L_2-Norm $\|\nabla u - \nabla u_h\|_{L^2}$	Quotient
243	6.39e-01	-
858	3.27e-01	1.95
3233	1.65e-01	1.98
12649	8.27e-02	1.99
50053	4.14e-02	1.99

Tabelle 5.2: Konvergenzverhalten des Gradienten

5.1.4 Vergleich der Verfahren

Um die drei Verfahren zur Gradientenrekonstruktion vergleichen zu können, lösen wir eine Laplace-Gleichung, deren exakte Lösung

$$u = \sin(\pi(x-1)/2)\sin(\pi(y-1)/2)$$

lautet, mit Hilfe der Finite-Element-Methode. Somit wird die Gradientenrekonstruktion basierend auf einer Funktion u_h durchgeführt, welche einige Näherungsfehler enthält. Darüber hinaus wurde ein kreisförmiger Teil des Gitters lokal verfeinert, um die Verfahren auf Gittern zu testen, wie sie durch adaptive Verfeinerungsprozesse entstehen.
Alle drei Verfahren rekonstruierten die partielle Ableitung

$$u_x = \frac{\pi}{2}\cos(\pi(x-1)/2)\sin(\pi(y-1)/2)$$

mit den in der Tabelle 5.3 angegebenen Fehlern.
Der Vergleich der Tabellen 5.2 und 5.3 zeigt die Vorteile der Gradientenrekonstruktion gegenüber der Verwendung des unbehandelten Gradienten der Funktion u_h aus dem Finite-Element-Raum. Ohne ein solches Verfahren sind die linearen finiten Elemente hinsichtlich des Gradienten lediglich ein Verfahren erster Ordnung.

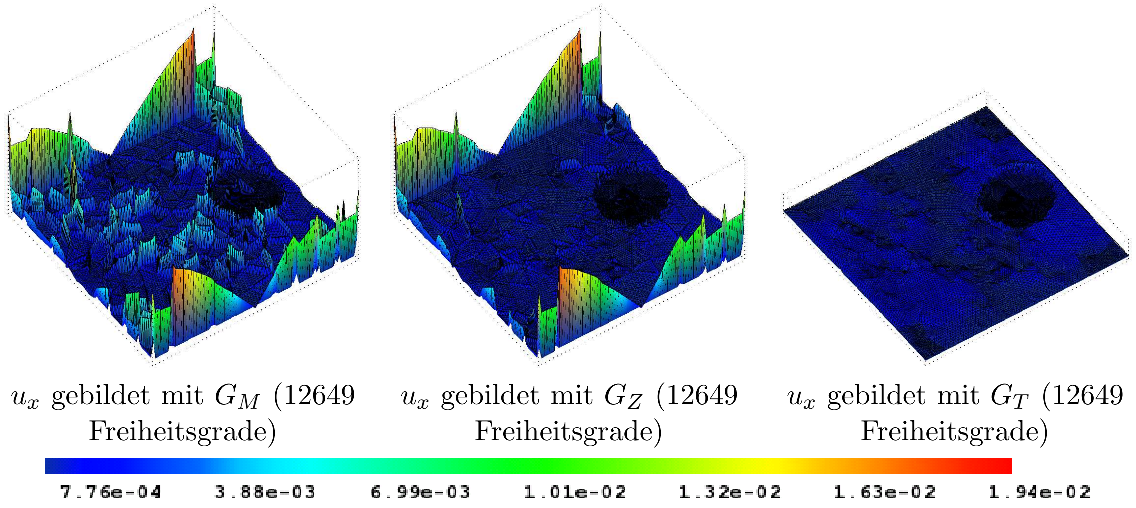

u_x gebildet mit G_M (12649 Freiheitsgrade)
u_x gebildet mit G_Z (12649 Freiheitsgrade)
u_x gebildet mit G_T (12649 Freiheitsgrade)

Abbildung 5.5: Gradientenrekonstruktion im Vergleich

	Einfache Rekonstruktion über Mittelung				Z^2				taylorbasierend			
Freiheits-		Fehler				Fehler				Fehler		
grade	Zeit	Max	L^2	Quot.	Zeit	Max	L^2	Quot.	Zeit	Max	L^2	Quot.
243	0.00s	1.49e-01	6.21e-02	-	0.01s	1.49e-01	5.99e-02	-	0.02s	7.35e-02	6.21e-02	-
858	0.00s	7.68e-02	2.48e-02	2.50	0.04s	7.68e-02	1.95e-02	3.07	0.04s	2.33e-02	1.82e-02	3.41
3233	0.00s	3.87e-02	9.20e-03	2.62	0.13s	3.87e-02	6.57e-03	2.96	0.16s	6.85e-03	5.01e-03	3.63
12649	0.01s	1.94e-02	3.30e-03	2.78	0.51s	1.94e-02	2.25e-03	2.92	0.63s	2.64e-03	1.34e-03	3.74
50053	0.11s	9.70e-03	1.16e-03	2.84	2.07s	9.70e-03	7.83e-04	2.87	2.50s	1.35e-03	3.55e-04	3.77

Tabelle 5.3: Gradientenrekonstruktion im Vergleich

Unter den Rekonstruktionsverfahren erzielt die taylorbasierende Methode die besten Resultate und liefert Reduktionsraten von näherungsweise zweiter Ordnung.
Das Z^2-Verfahren ist zwar in der L_2-Norm dem einfachen Rekonstruktionsverfahren überlegen, weist jedoch in etwa den gleichen Fehler in der Maximumsnorm auf, da dieser, wie Abbildung 5.5 verdeutlicht, am Rand auftritt. Dort liefern beide Verfahren wegen der geringen Anzahl an Elementen pro Knoten die gleichen Resultate.
Die CPU-Kosten des taylorbasierenden und des Z^2-Verfahrens sind, wie der Tabelle 5.3 entnommen werden kann, etwa gleich hoch, wobei das taylorbasierende Verfahren geringfügig aufwendiger ist. Das [27], Kapitel 4.2 entnommene Verfahren ist hingegen weitaus weniger rechenintensiv als die beiden anderen Verfahren, da im Laufe der Rekonstruktion hier keinerlei Gleichungssysteme auftreten. Auf Grund dieser Resultate wurde für das Splitting dieses und das taylorbasierende Verfahren verwendet, wobei in der Regel letzteres zum Einsatz kommt. Bei der expliziten Behandlung des nichtlinearen Terms wird die günstigste Form der Gradientenrekonstruktion eingesetzt, zum einen, da sonst die Kosten für diese Behandlung stark ansteigen und zum anderen, weil sich keine großen Vorteile bei der

Verwendung des taylorbasierenden Verfahrens einstellen.
In der folgenden Bemerkung 5.10 werden die Eigenschaften des taylorbasierten Gradientenrekonstruktionsverfahrens kurz zusammengefasst.

Bemerkung 5.10

- *Der Gradient von $f \in C^2(\bar{\Omega})$ wird auf einem regelmäßigen Gitter im Inneren von Ω von zweiter Ordnung approximiert. Für l aus dem Satz 5.4 gilt also $l = 2$.*
- *Numerische Tests haben ergeben, dass sich durch die beschriebene Gewichtung vergleichbare Reduktionsraten auf adaptiv verfeinerten Gittern ergeben (s. z.B. Tabelle 5.3).*
- *Der Gradient von $f \in C^2(\bar{\Omega})$ an den Knoten auf dem Rand von Ω wird nach numerischen Tests mindestens von erster Ordnung approximiert, wie man dem Maximumsfehler in der Tabelle 5.3 entnehmen kann.*

Im Kapitel 5.5.4 wird sich herausstellen, dass es wünschenswert ist, nicht nur die Ableitungen erster sondern auch zweiter Ordnung rekonstruieren zu können. Inwieweit dies möglich ist und mit welchen Fehlern dabei zu rechnen ist, wird an der Ableitung u_{xx} untersucht. Die in der Tabelle 5.4 erzeugten Ableitungen wurden alle durch das erneute Anwenden des Verfahrens zur Gradientenrekonstruktion auf die partielle Ableitung nach x berechnet. Bei der taylorbasierenden Rekonstruktion wäre es möglich, auf diesen Schritt zu verzichten, da das Verfahren direkt die partiellen Ableitungen bis zur zweiten Ordnung berechnet. Die so gewonnenen Werte sind jedoch geringfügig schlechter als die oben angegebenen, so dass wir einheitlich die obige Methode verwendet haben.

	Einfache Rekonstruktion über Mittelung			Z^2			taylorbasierend		
Freiheits-	Fehler			Fehler			Fehler		
grade	Max	L^2	Quot.	Max	L^2	Quot.	Max	L^2	Quot.
243	4.25e-01	2.47e-01	-	3.75e-01	2.21e-01	-	5.35e-01	2.53e-01	-
858	5.25e-01	1.98e-01	1.25	5.50e-01	1.46e-01	1.51	2.35e-01	9.27e-02	2.72
3233	5.20e-01	1.48e-01	1.33	5.51e-01	8.47e-02	1.75	1.56e-01	4.07e-02	2.27
12649	5.51e-01	1.05e-01	1.41	5.07e-01	5.29e-02	1.60	1.03e-01	1.71e-02	2.38
50053	5.08e-01	7.40e-02	1.42	5.51e-01	3.50e-02	1.51	1.03e-01	7.92e-03	2.16

Tabelle 5.4: Bildung von u_{xx} im Vergleich

Wie man sieht, ist das taylorbasierende Gradientenrekonstruktionsverfahren auch bei der Berechnung der partiellen Ableitungen zweiter Ordnung in der Lage, gute Reduktionsraten in der L_2-Norm zu gewährleisten. Allerdings stagniert die Fehlerreduktion des Maximumsfehlers. Wenn dies am Rand geschieht, ist die Konstruktion von Neumann-Randwerten, wie sie im Kapitel 5.5.4 versucht wird, problematisch.
Das Problem tritt am stärksten für die partiellen Ableitungen u_{xx} und u_{yy} auf, die den Laplace-Operator bilden. Die gemischte partielle Ableitung u_{xy} ist weniger stark betroffen, wie man Tabelle 5.5 entnehmen kann. Die Abbildungen 5.6 und 5.7 machen deutlich, dass beim zweimaligen Differenzieren nach einer Variablen die Fehler am Rand sehr groß werden können.

	Einfache Rekonstruktion über Mittelung			Z^2			taylorbasierend		
Freiheits-	Fehler			Fehler			Fehler		
grade	Max	L^2	Quot.	Max	L^2	Quot.	Max	L^2	Quot.
243	1.14e-00	4.86e-01	-	1.14e-00	4.77e-01	-	2.22e-01	1.94e-01	-
858	7.88e-01	3.01e-01	1.61	1.26e-00	3.14e-01	1.51	8.01e-02	6.86e-02	2.82
3233	8.19e-01	2.09e-01	1.44	1.06e-00	2.05e-01	1.53	5.50e-02	2.36e-02	2.91
12649	8.22e-01	1.47e-01	1.42	1.06e-00	1.38e-01	1.48	4.14e-02	9.11e-03	2.59
50053	8.22e-01	1.04e-01	1.41	1.05e-00	9.58e-02	1.44	4.08e-02	4.19e-03	2.17

Tabelle 5.5: Bildung von u_{xy} im Vergleich

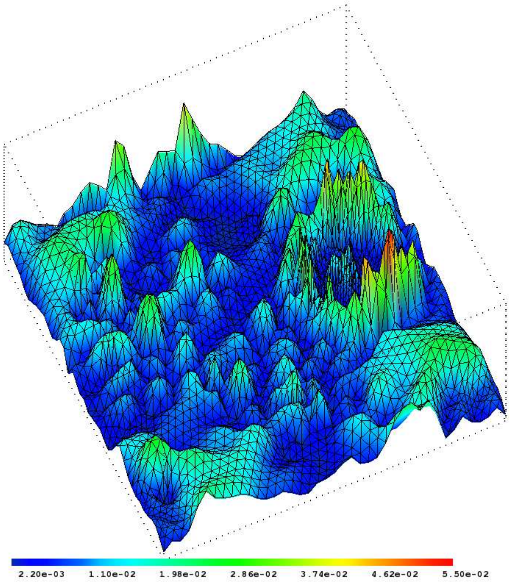


Abbildung 5.6: u_{xy} gebildet mit G_T (3233 Freiheitsgrade)

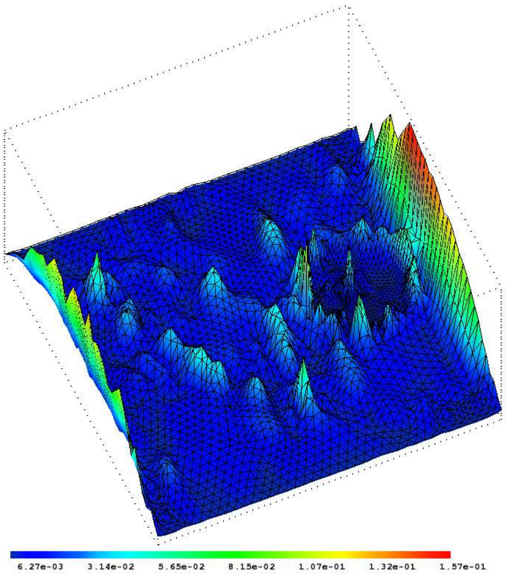


Abbildung 5.7: u_{xx} gebildet mit G_T (3233 Freiheitsgrade)

Daher ist es sinnvoll, den im Kapitel 5.5.4 benötigten Laplace-Operator mit Hilfe der Divergenzfreiheit umzuformulieren.

$$\nabla \cdot v = 0 \Leftrightarrow v_{1_x} + v_{2_y} = 0 \Leftrightarrow v_{1_x} = -v_{2_y} \tag{5.5}$$

$$\nabla^2 v_1 = v_{1_{xx}} + v_{1_{yy}} = v_{1_{yy}} - v_{2_{yx}} \tag{5.6}$$

$$\nabla^2 v_2 = v_{2_{xx}} + v_{2_{yy}} = v_{2_{xx}} - v_{1_{xy}} \tag{5.7}$$

Für $\nu \to 0$ ist die Formulierung des Laplace-Operators und die damit verbundene Fehlerreduktion, wie wir noch sehen werden, von geringer Bedeutung.

Definition 5.11
Wir bezeichen mit G^2 in Anlehnung an ∇^2 die Rekonstruktion des Laplace-Operators durch ggf. mehrfaches Anwenden einer Gradientenrekonstruktionstechnik mit Hilfe des in (5.5) gezeigten Schemas

$$\begin{aligned} G^2 v_1 &= v_{1_{yy}} - v_{2_{yx}} \ , \\ G^2 v_2 &= v_{2_{xx}} - v_{1_{xy}} \ . \end{aligned}$$

5.2 Der Finite-Element-Splitting-Algorithmus

Zunächst notieren wir einen Zeitschritt im Splitting für Spektrale Verfahren als Algorithmus:

Ein Zeitschritt im Splittingverfahren (Version I):

1. Berechne den Extrapolanten für den Druck $\bar{p}$
2. Löse die Gleichung für die Zwischengeschwindigkeit $\tilde{v}$ (explizit oder implizit)
3. Löse für die Druckaktualisierung die Gleichung
   $-\nabla^2 p_{update} = -\frac{\beta_0}{\triangle t} \nabla \cdot \tilde{v}^{n+1}$
4. Aktualisiere den Druck durch
   $p = p_e + p_{update}$
5. Sorge für die Divergenzfreiheit von v mittels
   $v^{n+1} = \tilde{v}^{n+1} + \frac{\triangle t}{\beta_0} \nabla p_{update}$

Die Übergänge zwischen den Gittern werden mit Hilfe der in Kapitel 4.5.1 vorgestellten Prolongations- und Restriktionsoperatoren vorgenommen. Das bedeutet, dass die benötigten Integrale analog zum Vorgehen bei Multigrid-Verfahren auf den feineren Gittern berechnet und dann durch unter Umständen mehrfaches Anwenden von P^t auf das gröbere übertragen werden. Die Werte für den Gradienten des Drucks werden in Schritt 1 und 3 mit Hilfe von P auf das feinere Gitter interpoliert.

Diese Version I des Splittings wurde von Heinrichs und Haschke mit Spektralen Verfahren erfolgreich eingesetzt. Für die in dieser Arbeit verwendeten finiten Elemente traten jedoch bei einigen Testfällen in Zusammenhang mit kleinen Zeitschrittweiten insbesondere bei der Navier-Stokes-Gleichung Probleme auf.

Die Ursache findet sich in den Schritten 3 bis 5. Die Methode der finiten Elemente ist weniger genau als die der Spektralen Verfahren und beim Differenzieren geht hier zusätzlich mehr Genauigkeit verloren.
Kritisch ist jedoch vor allem die Berechnung von $\nabla \cdot \tilde{v}^{n+1}$. Im Allgemeinen wird selbst für

die Darstellung der exakten Lösung v mit Hilfe des I_h-Operators nicht

$$\nabla \cdot (I_h v(t^{n+1})) = 0$$

gelten, da der Finite-Element-Raum V_h keine absolut divergenzfreie Darstellung erlaubt. Bei der Lösung der Gleichung in Schritt 2 und der Gradientenrekonstruktion entstehen darüber hinaus weitere Fehler.

Im Anschluss wird in Schritt 3 eine Aktualisierung auf der Basis dieser Daten berechnet. Der aus den oben angegebenen Gründen entstehende Fehler ist zwar in jedem einzelnen Zeitschritt gering, wird aber schrittweise zum Druck hinzuaddiert.

Ändert die Strömung über einen längeren Zeitraum lediglich ihre Stärke, aber nicht ihre Struktur, so addiert man die immer wieder gleichen Darstellungsfehler zum Druck hinzu. Durch den Faktor $\frac{\beta_0}{\triangle t}$ wird dieser Effekt für kleine Zeitschritte deutlich verstärkt. Im Laufe der Arbeit traten dadurch Instabilitäten abhängig von der Re-Zahl und der Differenz $\|\nabla \cdot v - G_T \cdot I_h v\|$ ab einer Schrittweite von ca. $\triangle t = \frac{1}{64}$ und 200 Schritten auf.

Der Druck ist hierbei stärker betroffen als die Geschwindigkeit, da hier der Fehler dank dem Faktor $\frac{\triangle t}{\beta_0}$ in Schritt 5 nicht durch die Wahl von $\triangle t$ verstärkt wird. Dadurch divergiert zunächst der Druck und durch die Kopplung der beiden Größen einige Zeitschritte später auch die Geschwindigkeit.

Es gibt nun drei Möglichkeiten, diesem Problem zu begegnen:

Zuerst kann man die Darstellung der Lösung im Finite-Element-Raum verbessern und damit auch die Genauigkeit der berechneten Approximation steigern, indem man diese im Ort verfeinert. Die Ortsverfeinerung führt schnell auf zu große CPU-Kosten und kann das Problem in der Praxis nicht beheben.

Zweitens kann man versuchen, das Verfahren zur Gradientenrekonstruktion zu verbessern. Diesen Ansatz haben wir mit der Konstruktion des G_T-Operators als Ersatz für den einfacheren G_M-Operator verfolgt. Damit war es zwar möglich, das Problem abzumildern, es ließ sich jedoch nicht völlig beseitigen.

Drittens kann man sich bemühen, einen Splittingschritt zu entwickeln, der den additiven Charakter der Druckaktualisierung unterbricht.

5.3 Konstruktion einer Näherung für den Druck

Zentral hierbei ist es, den Druckextrapolanten auf eine andere Weise zu bestimmen.
Hierzu betrachten wir erneut die Gleichung (2.2). Diese Gleichung stellen wir nun nach ∇p um und benutzen anschließend den $\nabla \cdot$ Operator. Dabei setzen wir voraus, dass v divergenzfrei ist und erhalten

$$-\nabla^2 p \;=\; -\nabla \cdot f + \nabla \cdot ((v \cdot \nabla) v)\ . \tag{5.8}$$

Den nichtlinearen Term kann man unter Ausnutzung der Divergenzfreiheit von v weiter vereinfachen zu

$$\begin{aligned} v_{1x}v_{1x} + v_1v_{1xx} + v_{2x}v_{1y} + v_2v_{1xy} + v_{1y}v_{2x} + v_1v_{2xy} + v_{2y}v_{2y} + v_2v_{2yy} \\ = v_{1x}v_{1x} + 2v_{2x}v_{1y} + v_{2y}v_{2y} + v_1\frac{d}{dx}(v_{1x} + v_{2y}) + v_2\frac{d}{dy}(v_{1x} + v_{2y}) \\ = v_{1x}v_{1x} + 2v_{2x}v_{1y} + v_{2y}v_{2y} \ . \end{aligned}$$

Damit sind wir in der Lage, eine Gleichung zur Berechnung einer Näherung für p^{n+1} aufzustellen, die im Stokes-Fall den Druck zum Zeitpunkt t^{n+1} direkt berechnen könnte.

$$-\nabla^2\bar{p}^{n+1} = -(f_{1x}^{n+1} + f_{2y}^{n+1}) \ . \tag{5.9}$$

Da wir am Stokes-Fall aber nur im Sinne eines Spezialfalles der Navier-Stokes-Gleichung interessiert sind, arbeiten wir im Folgenden mit diesem Ansatz:

$$-\nabla^2\bar{p}^{n+1} = -(f_{1x}^{n} + f_{2y}^{n}) + v_{1x}^n v_{1x}^n + 2v_{2x}^n v_{1y}^n + v_{2y}^n v_{2y}^n \ . \tag{5.10}$$

Mischformen im Sinne von

$$-\nabla^2\bar{p}^{n+1} = -(f_{1x}^{n+1} + f_{2y}^{n+1}) + v_{1x}^n v_{1x}^n + 2v_{2x}^n v_{1y}^n + v_{2y}^n v_{2y}^n \tag{5.11}$$

zum Erlangen einer höheren Ordnung haben sich leider als weniger stabil herausgestellt als der Ansatz 5.10.
Wird nun p mit Hilfe von 5.10 berechnet, so kann ausgeschlossen werden, dass sich die Fehler in der oben aufgezeigten Weise addieren.

Ein Zeitschritt im Splittingverfahren (Version II):

1. Berechne eine Näherung $\bar{p}^{n+1} = p^n$ an den Druck durch (5.10)
2. Löse die Gleichung für die Zwischengeschwindigkeit $\tilde{v}$ (explizit oder implizit)
3. Löse für die Druckaktualisierung die Gleichung $-\nabla^2 p_{update} = -\frac{\beta_0}{\triangle t}\nabla \cdot \tilde{v}^{n+1}$
4. Aktualisiere den Druck durch $p^{n+1} = \tilde{p}^n + p_{update}$.
5. Sorge für die Divergenzfreiheit von v mittels $v^{n+1} = \tilde{v}^{n+1} + \frac{\triangle t}{\beta_0}\nabla p_{update}$

Wenn man die Versionen I und II des Splittings vergleicht, erkennt man, dass bei Version II in jedem Zeitschritt ein Gleichungssystem mehr zu lösen ist, womit nun im Laufe eines Splittingschritts 2 · Picarditerationen + **2** Gleichungssysteme zu lösen sind. Das zusätzliche

Gleichungssystem ist symmetrisch und kleiner als diejenigen, die bei der Lösung der Geschwindigkeitsgleichungen auftreten. Der neue Schritt 1 stellt daher also nur eine geringe Steigerung der CPU-Kosten dar.

Um die Version II des Splittings kostenneutral zu realisieren, könnte man versuchen, z.B. im Stokes-Fall auf die Schritte 3 und 4 zu verzichten, da p hier durch Schritt 1 ohne den Einsatz einer Näherung an v berechnet werden kann. Für kleine Zeitschrittweiten könnte die Näherung des konvektiven Terms $(v \cdot \nabla)v$ in 5.10 so gut sein, dass man ebenfalls auf die Schritte 3 und 4 verzichten kann.

Im Laufe einiger Testrechnungen stellte sich jedoch heraus, dass das Splitting ohne die Schritte 3 und 4, welche die Divergenzfreiheit garantieren, qualitativ schlechter wird und in seltenen Fällen sogar instabil wird.

In diesem Algorithmus wird der Druck nun doppelt berechnet. Als Ergebnis des Druckaktualisierungsschritts ergibt sich p^{m+1} und anschließend wird $p(t^{m+1})$ erneut mit Hilfe der Gleichung (5.10) als $\bar{p}$ approximiert. Es stellt sich nun die Frage, wie sich diese beiden Näherungen zueinander verhalten. Dies hängt wesentlich von dem vorliegenden Problem ab. Sind die Neumann-Randbedingungen von großer Bedeutung und sowohl $\nabla \cdot v$ als auch $\nabla \cdot f$ gut mit einem Verfahren zur Gradientenrekonstruktion approximierbar, so kann (5.10) die bessere Näherung liefern. Im Regelfall erzeugt jedoch der Druckaktualisierungsschritt die genauere Näherung. Abbildung 5.8 zeigt die beiden Näherungen auf dem Testproblem III mit $\triangle t = 1/64, \nu = 1/1000$ und BDF(3). Das in Abschnitt 5.5.2 definierte Testproblem setzt sich aus mehreren trigonometrischen Funktionen zusammen und stellt daher vergleichsweise hohe Ansprüche an die Gradientenrekonstruktion. Wichtig für weitere Überlegungen in Kapitel 7 ist die Tatsache, dass der durch (5.10) berechnete Druck zwar in der Regel eine schlechtere Approximation an p ist, dass diese jedoch nicht das sonst übliche schlechte Einschwingverhalten des Splittings im Druck aufweist.

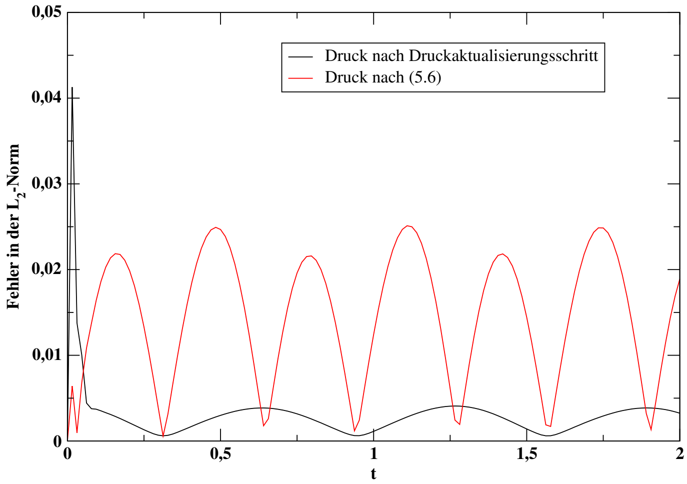


Abbildung 5.8: Druckvergleich

Bei vielen Problemen ergeben sich die besten Ergebnisse im Druck, wenn man nur alle $i \in \mathbb{N}$ Schritte eine Approximation auf Grund von (5.10) berechnet und bei den übrigen Schritten entsprechend dem Splitting Version I verfährt. Dabei sind Werte von $i = 1...10$ abhängig von dem Problem und der Zeitschrittweite $\triangle t$ sinnvoll gewesen. Da die optimale Anzahl an Schritten nicht adaptiv bestimmt werden konnte und im Extremfall schon $i = 2$ zum Divergieren des Verfahrens führte, ist es die einzige robuste Methode, in jedem Schritt wie in Version II des Splittings vorzugehen. Alle folgenden Angaben beziehen sich also auf Version II des Splittings und die nach dem Druckaktualisierungsschritt berechnete Approximation des Drucks.

5.4 Analyse des Splittingalgorithmus

Um die Stabilitäts- und Konvergenzeigenschaften des Splittingalgorithmus zu untersuchen, betrachten wir zunächst den Druckaktualisierungsschritt. In den Abschnitten 5.4.1 und 5.4.2 beschränken wir unsere Untersuchung auf die Stokes-Gleichung.

5.4.1 Splittingfehler

Beim Druckaktualisierungsschritt wird die Gültigkeit der Gleichung (3.18) postuliert, um mit ihrer Hilfe die Druckaktualisierung und darauf aufbauend die Geschwindigkeit v^{m+1} mit $\tilde{v}^{m+1}$ berechnen zu können. Die Gleichung (3.18) ist jedoch nicht exakt.

In einem Splittingschritt wird zunächst eine Zwischengeschwindigkeit $\tilde{v}^{m+1}$ zum Zeitpunkt t^{m+1} auf der Basis des Druckextrapolanten $\bar{p}^{m+1}$ berechnet. Die Differenz zwischen $\tilde{v}^{m+1}$ und der Lösung der Gleichung (3.13) v^{m+1} ist eine Folge der Verwendung des Druckextrapolanten statt des Drucks p^{m+1}.

Bilden wir die Differenz der Gleichungen (3.16), deren Lösung die Zwischengeschwindigkeit $\tilde{v}^{m+1}$ ist, und des Originalproblems (3.13), dessen Lösung v^{m+1} ist, so erhalten wir:

$$\frac{\beta_0}{\triangle t}(v^{m+1} - \tilde{v}^{m+1}) + \nabla(p^{m+1} - \bar{p}^{m+1}) = \nu\nabla^2(v^{m+1} - \tilde{v}^{m+1}) \text{ in } \Omega \; . \tag{5.12}$$

Definition 5.12 (Splittingdefektterm)
Den im Druckaktualisierungsschritt (3.18) für $\nu \neq 0$ oder $\nabla^2(v^{m+1} - \tilde{v}^{m+1}) \neq 0$ vernachlässigten Term aus (5.12)

$$\nu\nabla^2(v^{m+1} - \tilde{v}^{m+1})$$

bezeichnen wir im Folgenden als Splittingdefektterm.

Zur Abschätzung dieses Splittingdefektterms benötigen wir folgendes Lemma.

Lemma 5.13
Seien v^{m+1} und $\tilde{v}^{m+1} \in H^2(\Omega)$ und Ω erfülle die Voraussetzungen aus Satz 4.3, dann gilt für die Zwischengeschwindigkeit $\tilde{v}^{m+1}$ und die Lösung des diskretisierten Problems v^{m+1} (3.13) die Abschätzung

$$\|v_1^{m+1} - \tilde{v}_1^{m+1}\|_2 \leq C\|p_x^{m+1} - \bar{p}_x^{m+1}\|_0 \; , \tag{5.13}$$

$$\|v_2^{m+1} - \tilde{v}_2^{m+1}\|_2 \leq C\|p_y^{m+1} - \bar{p}_y^{m+1}\|_0 \; . \tag{5.14}$$

Beweis :
Die Abschätzung wird mit Hilfe der H^2-Regularität bewiesen. Für die Bilinearform $a(.,.)$ des Helmholtzproblems (3.22) sind beim Stokes-Fall alle Voraussetzungen für H^2-Regularität erfüllt und für Ω wurden die entsprechenden Eigenschaften vorausgesetzt. Subtrahiert man nun die beiden Variationsformulierungen

$$\begin{aligned} a(v_1^{m+1}, u) &= (f_1, u)_{L^2} - (p_x^{m+1}, u)_{L^2} \quad \forall u \in V_h \qquad (5.15) \\ a(\tilde{v}_1^{m+1}, u) &= (f_1, u)_{L^2} - (\bar{p}_x^{m+1}, u)_{L^2} \quad \forall u \in V_h \;, \qquad (5.16) \end{aligned}$$

so erhält man ein Variationsproblem mit homogenen Null-Dirichlet-Randbedingungen, dessen Lösung $v_1^{m+1} - \tilde{v}_1^{m+1}$ ist:

$$a(v_1^{m+1} - \tilde{v}_1^{m+1}, u) = -(p_x^{m+1} - \bar{p}_x^{m+1}, u)_{L_2} \quad \forall u \in V \;. \tag{5.17}$$

Da wir H^2-Regularität vorausgesetzt haben, erhalten wir nach der Definition 4.2 nun für die Variationsformulierung (5.17) die Abschätzung

$$\|v_1^{m+1} - \tilde{v}_1^{m+1}\|_2 \leq C \|p_x^{m+1} - \bar{p_x}^{m+1}\|_0 \;. \tag{5.18}$$

Der Beweis für v_2 und p_y folgt analog.

□

Mit diesem Lemma sind wir nun in der Lage, den Splittingdefektterm abzuschätzen.

Satz 5.14
Sei $[0, T_{ende}]$ durch eine äquidistante Zerlegung mit der Schrittweite $\triangle t$ diskretisiert und $p \in C^{s+1}(0, T_{ende})$. Des Weiteren sei $\bar{p}^{m+1}$ ein Extrapolant der Ordnung s in t an p^{m+1}, so gilt unter den Voraussetzungen von Lemma 5.13 für den Splittingdefektterm die folgende Abschätzung:

$$\nu \|\nabla^2 (v_1^{m+1} - \tilde{v}_1^{m+1})\|_0 \leq (\triangle t)^s \nu C \|\frac{\partial^s}{\partial t^s} p_x^{m+1-s}\|_0 + O(\triangle t) \;, \tag{5.19}$$

$$\nu \|\nabla^2 (v_2^{m+1} - \tilde{v}_2^{m+1})\|_0 \leq (\triangle t)^s \nu C \|\frac{\partial^s}{\partial t^s} p_y^{m+1-s}\|_0 + O(\triangle t) \;. \tag{5.20}$$

Beweis :
Aus Lemma 5.13 folgt mit $\|\nabla^2 u\|_0 \leq \|u\|_2 \forall u \in H^2$

$$\nu \|\nabla^2 (v_1^{m+1} - \tilde{v}_1^{m+1})\|_0 \leq \nu \|v_1^{m+1} - \tilde{v}_1^{m+1}\|_2 \leq \nu C \|\frac{\partial}{\partial x}(p^{m+1} - \bar{p}^{m+1})\|_0 \;. \tag{5.21}$$

Da $\bar{p}^{m+1}$ ein Extrapolant s-ter Ordnung an p^{m+1} ist, kommen wir mit Hilfe einer Taylorentwicklung zu

$$\nu C \|\frac{\partial}{\partial x}(p^{m+1} - \bar{p}^{m+1})\|_0 = (\triangle t)^s \nu C \|\frac{\partial^s}{\partial t^s} p_x^{m+1-s}\|_0 + O(\triangle t) \;. \tag{5.22}$$

Der Beweis für v_2 und p_y folgt analog.

□

Der durch den Splittingdefektterm erzeugte Fehler ist im Vergleich zu den anderen Fehlerquellen oft sehr klein. Wir werden daher in Kapitel 5.5.3 versuchen, die anderen Fehlerquellen auszuschalten, um diesen Fehler nachweisen zu können.
Es ist wichtig festzuhalten, dass die Abschätzung aus 5.14 für kleine ν, welche die besonders interessanten Fälle in der Strömungsmechanik darstellen, einen entsprechend kleinen Fehler durch den Splittingdefektterm erwarten lässt. Diese Analyse deckt sich mit den Aussagen von Turek in [42], S.21f, dass Splittingverfahren ein schlechteres Konvergenzverhalten für große ν aufweisen. Für $\nu \to 0$ geht der Splittingfehler gegen Null.

5.4.2 Zeitdiskretisierungsfehler

Dieser entsteht u.a. durch die Approximation des Ableitungsoperators $\frac{\partial}{\partial t}$ durch ein BDF-Verfahren und wird im Folgenden als *Zeitfehler* bezeichnet. Hier verwenden wir BDF-Verfahren bis zur dritten Ordnung. Dieser Fehleranteil wird also maximal mit $\triangle t^3$ verringert. Ist der Splittingfehler vernachlässigbar klein, so können wir den idealisierten Druckaktualisierungsschritt auf seine Fehlerreduktionsordnung hin betrachten, um den Zeitdiskretisierungsfehler für den Druck und die Geschwindigkeit abschätzen zu können. Dabei orientieren wir uns an [24] und betrachten die Gleichung (3.18), in welche wir die exakte Lösung einbringen

$$\frac{\beta_0}{\triangle t}(v^{m+1} - I_h v(t^{m+1}) + I_h v(t^{m+1}) - \tilde{v}^{m+1}) + \tag{5.23}$$

$$\nabla(p^{m+1} - I_h p(t^{m+1}) + I_h p(t^{m+1}) - \bar{p}_l^{m+1}) = 0 \text{ in } \Omega \ , \tag{5.24}$$

wobei hier erneut der Splittingdefektterm vernachlässigt wurde. Diese Gleichung lässt sich umformen zu

$$v^{m+1} - I_h v(t^{m+1}) + \frac{\triangle t}{\beta_0}\nabla(p^{m+1} - I_h p(t^{m+1})) \tag{5.25}$$

$$= \triangle t \left(\frac{\tilde{v}^{m+1} - I_h v(t^{m+1})}{\triangle t} + \frac{1}{\beta_0}\nabla(\bar{p}^{m+1} - I_h p(t^{m+1})) \right) \ . \tag{5.26}$$

Da die Zwischengeschwindigkeit $\tilde{v}^{m+1}$ durch eine Zeitdiskretisierung von l-ter Ordnung und der Extrapolant $\bar{p}^{m+1}$ von s-ter Ordnung ist, erhalten wir den Fehler

$$v^{m+1} - I_h v(t^{m+1}) + \frac{\triangle t}{\beta_0}\nabla(p^{m+1} - I_h p(t^{m+1})) \ ,$$

der mit

$$O(\triangle t^l) + O(\triangle t^{s+1}) \tag{5.27}$$

gegen Null geht. Mit der Wahl $r = l = s+1$ erhält man für die Stokes-Gleichung ein Verfahren r-ter Ordnung in der Zeit.

5.4.3 Fehler durch die Finite-Element-Diskretisierung

Genau entgegengesetzt zum Splittingfehler verhält es sich mit dem Fehler, der durch die Lösung auf der Basis der finiten Elemente entsteht, im weiteren Verlauf kurz *Ortsfehler* genannt. Das Cèa-Lemma 4.7 sagt eine schlechtere Qualität bei der Finite-Element-Lösung für $\nu \to 0$ voraus. Der Einsatz der Stromliniendiffusionsmethode bei den durch das Splitting der Navier-Stokes-Gleichung erzeugten Konvektions-Diffusions-Reaktions-Gleichungen sorgt hier für die nötige Stabilität. Allerdings lässt sich in diesem Fall mittels (4.17) nur noch eine Fehlerreduktion in Abhängigkeit von $h\sqrt{h}$ zeigen, wobei in der Abschätzung $\nu \approx 0$ angenommen wird, da wir hier von Fällen mit kleinem Diffusionskoeffizienten ausgehen. Bei dem Galerkinverfahren ohne Stromliniendiffusion hingegen wird wie am Ende von Kapitel 4.1 eine Reduktion mit h^2 vorausgesetzt. Abhängig von der Anzahl der Elemente, auf denen die Stabilisierungstechnik angewendet wird, welche sich mit $\nu \to 0$ vergrößert, ist also ein höherer Fehler bzw. ein schlechteres Konvergenzverhalten zu erwarten. Kommen wir nun zum Zeitdiskretisierungsfehler.

5.4.4 Linearisierungsfehler

Bei diesem Fehler, der seinen Ursprung in der expliziten oder impliziten Behandlung des nichtlinearen Terms hat, darf bei der Wahl $v_e := v^n$ die erste Ordnung in der Zeit bei der Fehlerreduktion als sicher angesehen werden. Bei einer ausreichend guten Startnäherung v_e geht dieser Fehler mit steigender Anzahl an Picarditerationen gegen Null.

5.4.5 Ableitungsfehler

Ableitungsfehler entstehen durch die Approximation von ∇ durch ein Verfahren zur Gradientenrekonstruktion. Bei einer impliziten Behandlung des nichtlinearen Terms treten sie bei der Bildung von ∇p und $\nabla \cdot v$ im Algorithmus auf.

Bei einer expliziten Behandlung kommt die Bildung von ∇v_1 und ∇v_2 multipliziert mit den Picarditerationen pro Schritt (**PIPS**) hinzu.

Diese im Kapitel 5.1.4 diskutierten Fehler entstehen in dieser Form pro Zeitschritt und sind abhängig von der Art des Gitters und dessen Feinheit h.

5.5 Numerische Ergebnisse und Analyse

5.5.1 Zur Implementierung

Zur Implementierung des Splittings wurde eine FEM-Bibliothek in C gemäß dem C99-Standard geschrieben, die auf die bewährten Bibliotheken Lapack, ATLAS und UMFPACK aufbaut.
Als Compiler für alle Bibliotheken wurde der GNU C Compiler in der Version 3.3.1 verwendet.
Die Rechnungen wurden auf verschiedenen Rechnern der Ingenieurmathematik und der numerischen Mathematik der Universität Duisburg-Essen unter dem Betriebssystem Linux durchgeführt. Die angegebenen CPU-Zeiten beziehen sich entweder auf Rechner mit einem AMD Athlon-Prozessor, 1800 MHz, L2-Cache 256 KB (Rechnertyp I) oder auf Rechner mit einem AMD Opteron-Prozessor, 2000 MHz, L2-Cache 1024 KB (Rechnertyp II).

Im Kapitel 4.1 wurde die Nodale Basis eingeführt. Es existieren mehrere Alternativen zu dieser Wahl der Basis des Finite-Element-Raumes V_h. Eine dieser Alternativen ist die hierarchische Basis. Die Idee der hierarchischen Basis ist es, die Basis des Raumes V_h

aus einer Hierarchie von Basisfunktionen aufzubauen, die nach der Größe ihres Trägers gestaffelt werden. Der grobe Verlauf der Funktion wird durch die Basisfunktionen mit großem Träger bestimmt, während die Basisfunktionen mit kleinem Träger - diese werden auch hierarchischer Überschuss oder hierarchisches Komplement genannt - Korrekturen in diesem Verlauf darstellen. Die quadratischen finiten Elemente wurden mit Hilfe eines solchen hierarchischen Ansatzes wie in [6], S.91 umgesetzt.

5.5.2 Die Testprobleme

Die Testprobleme mit einer analytischen Lösung werden dadurch konstruiert, dass man eine auf Ω divergenzfreie Funktion v und dazu eine Funktion p wählt. v und p werden anschließend in die Navier-Stokes- bzw. Stokes-Gleichung eingesetzt, um anschließend die rechte Seite f so zu bestimmen, dass die Gleichung erfüllt ist. Durch dieses Vorgehen erhält man Problemstellungen, mit denen man das Splitting testen kann. Die Kenntnis der analytischen Lösung ermöglicht es dann, die Fehler in verschiedenen Normen zu bestimmen.

Solche *analytischen Testbeispiele* sind nicht in jeder Hinsicht typisch für anwendungsorientierte Aufgabenstellungen in der Strömungsmechanik. Bei den *analytischen Testbeispielen* ist es z.B. möglich, Testbeispiele für sehr große Reynoldszahlen mit vergleichsweise wenigen Freiheitsgraden zu lösen. Dies liegt darin begründet, dass die Lösung des analytischen Testbeispiels für alle Reynoldszahlen gleich glatt bleibt. Die Natur der Strömung ändert sich also nicht von laminar zu turbulent. Bei einer anwendungsbezogenen Fragestellung setzt ab einer gewissen Reynoldszahl die Wirbelbildung ein und die Strömung wird turbulent. Eine turbulente Strömung stellt jedoch ganz andere Anforderungen an den Algorithmus, erfordert wesentlich mehr Freiheitsgrade und ist nur in Ansätzen auf Großrechnern bzw. Clustern zu berechnen.

Die angegebenen Fehler verstehen sich wie folgt:

$$ER(u) = \max_{0.2<t^m} \|v_1(t^m) - u(t^m)\| \; ; u \in V_h \; ;$$

auf Abweichungen wird ggf. hingewiesen. Die Schwelle $0.2 < t^m$ dient dem Herausfiltern des in der Regel ein oder maximal zwei Schritte dauernden Einschwingprozesses, welcher diesem Splittingverfahren eigen ist. Da die größte von uns verwendete Schrittweite 0.2 ist, wird folglich mindestens der erste Schritt nicht gewertet. Dieser ungewöhnliche Fehler im ersten Schritt betrifft primär den Druck. Zur Vereinheitlichung der Auswertung wurde die Schwelle für die Geschwindigkeit beibehalten.

Stokes Testproblem : nichtpolynomialer Kreisfluss in $[0,1] \times [0,1]$

Dieses Testproblem, wir werden es im Folgenden mit der Ziffer **I** bezeichnen, ist dem Artikel [21] entliehen. Als Lösung wird dabei

$$\begin{aligned} v_1(x,y,t) &= z(t)r(x)s(y) \, , \\ v_2(x,y,t) &= -z(t)r(y)s(x) \end{aligned}$$

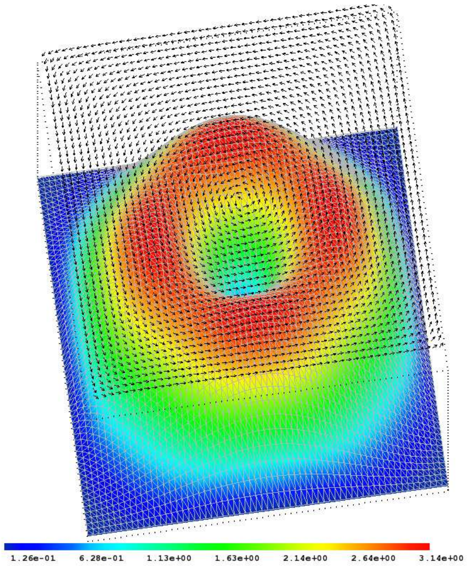


Abbildung 5.9: Geschwindigkeit $\|v\|_2$

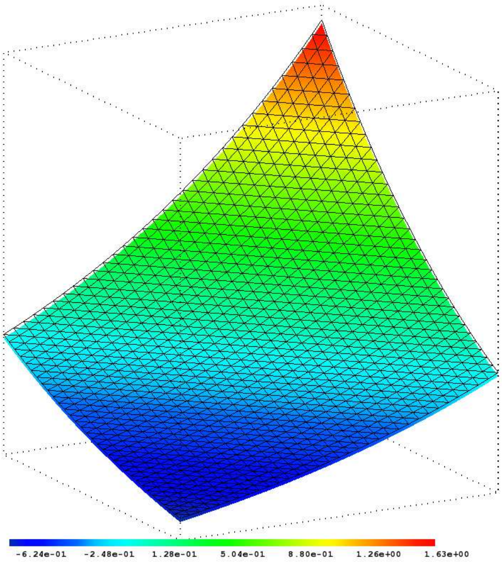


Abbildung 5.10: Druck p

mit

$$\begin{aligned} r(x) &= \sin^2(\pi x)\ , \\ s(x) &= r'(x) \end{aligned}$$

und

$$z(t) = (2\pi - 1 + \sin(2 * \pi t))\frac{1}{2\pi}\ t \in [0,8]$$

gewählt und die rechte Seite f entsprechend angepasst. Die Druckfunktion lautet:

$$p(x,y,t) = \cos(2\pi t)(\exp(x+y+1) - C)\ ,$$

wobei $C \in \mathbb{R}$ entsprechend der Drucknormierung gewählt wird.
In diesem Fall entspricht der Druck nicht völlig den Null-Neumann-Randbedingungen, welche häufig bei Splittingverfahren angewendet werden. Allerdings sind die Abweichungen gering und der Druck ist sehr glatt. Wird er auf der Basis von Null-Neumann-Randbedingungen berechnet, ist an der Ecke $(1,1)$ ein entsprechend großer Fehler im Druck zu erwarten.

Stokes Testproblem : polynomialer Kreisfluss in $[0,1] \times [0,1]$

Dieses Testproblem (**II**) stammt aus dem Übersichtswerk [32], S.22f. Als Geschwindigkeit wird dabei

$$\begin{aligned} v_1(x,y,t) &= x^2(1-x)^2(2y - 6y^2 + 4y^3)t^\mu\ , t \in [0,2] \\ v_2(x,y,t) &= -y^2(1-y)^2(2x - 6x^2 + 4x^3)t^\mu\ , t \in [0,2] \end{aligned}$$

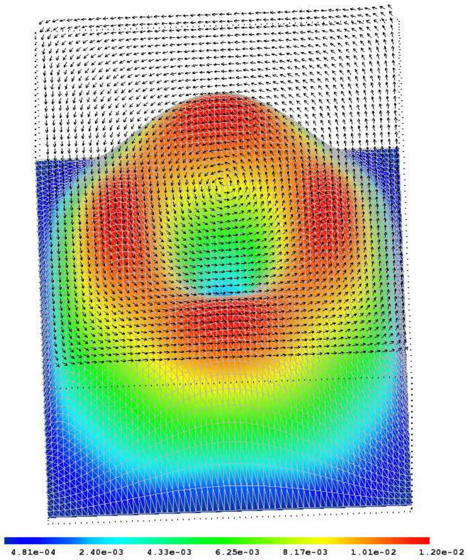


Abbildung 5.11: Geschwindigkeit $\|v\|_2$

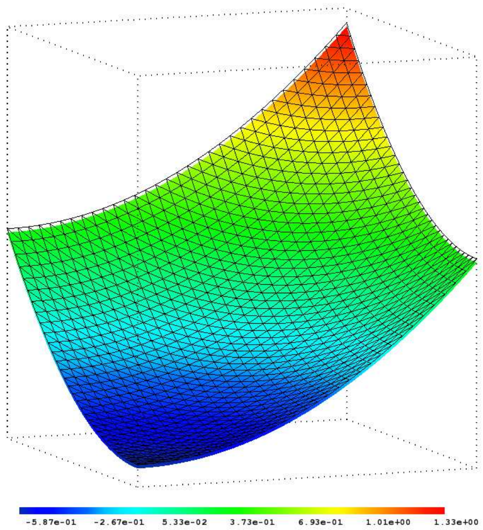


Abbildung 5.12: Druck p

gewählt. Die zugehörige Druckfunktion lautet:

$$p(x,y,t) \quad = \quad (x^2+y^2-\frac{2}{3})t^\gamma \ ; \ \gamma \in [0,1] \ t \in [0,2].$$

Das Testproblem wird also definiert durch die Parameter (ν,γ,μ). In [32] wird ausschließlich der Fall $\mu = 0$ verwendet. Unsere Testkonfiguration ist hingegen durch $(\nu, \frac{1}{2}, \mu \in \{0,1\})$ gegeben. Die Approximation im Ort wird dadurch erleichtert, dass die Lösung für den Druck ein Polynom niedriger Ordnung ist. Wie schon beim Testproblem I ist je nach Wahl der Neumann-Randbedingungen besonders im Bereich der Ecke $(1,1)$ ein entsprechend großer Fehler im Druck zu erwarten.
Der Testzeitraum ist der kürzeste aller Testbeispiele. Bei allen anderen Testproblemen wurde das Zeitintervall auf $[0,8]$ angesetzt, um die Langzeitstabilität des Verfahrens beurteilen zu können. Dies hat sich als notwendig erwiesen, da z.B. die in Kapitel 5.2 für den Navier-Stokes-Fall beschriebenen Schwierigkeiten erst bei längeren Zeitintervallen auftreten.

Navier-Stokes Testproblem : nichtpolynomialer Fluss durch das Einheitsquadrat

Das erste Testproblem für die Navier-Stokes-Gleichung (**III**) entstammt Artikel [21]. Als Lösung für die Geschwindigkeit wird dabei

$$\begin{aligned} v_1(x,y,t) &= \cos(5t)(\sin(\pi x/2)\cos(\pi y/2)) \ , \\ v_2(x,y,t) &= -\cos(5t)(\sin(\pi y/2)\cos(\pi x/2)) \end{aligned}$$

gewählt und auch hier wird $v|_{\partial\Omega}$ zur Festlegung der Dirichlet-Randwerte für die Geschwindigkeit verwendet. Die Druckfunktion lautet:

$$p(x,y,t) = \frac{\cos^2(5t)}{4}(\cos(\pi x)+\cos(\pi y)) + 10\cos(5t)(x+y-1) \ .$$

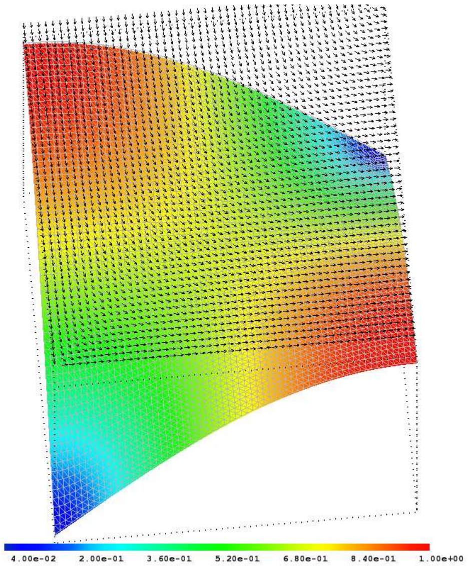


Abbildung 5.13: Geschwindigkeit $\|v\|_2$

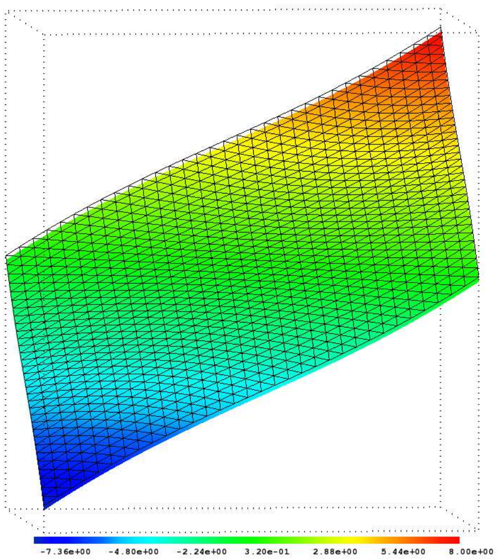


Abbildung 5.14: Druck p

Wie man sieht, ist die zeitliche Entwicklung des Drucks in seinen kurz- und langwelligeren Anteilen unterschiedlich und erneut ist nicht mit einem optimalen Konvergenzverhalten zu rechnen, wenn die Neumann-Randwerte nicht angepasst werden.

Navier-Stokes Testproblem : polynomialer Kreisfluss durch eine Spule

Dieses Testproblem **IV** wird auf einer Spule berechnet. Sei $r = \sqrt{x^2 + y^2}$, dann ist $\Omega = \{(x, y) \in \mathbb{R}^2 | 1 \leq r \leq 2\}$. Die Geometrie muss also mit finiten Elementen angenähert werden und kann nicht exakt abgebildet werden. Für die Geschwindigkeit wird dabei

$$\begin{aligned} v_1(x, y, t) &= -y(0.25 - (r - 1.5)^2) \sin(2\pi t) , \\ v_2(x, y, t) &= x(0.25 - (r - 1.5)^2) \sin(2\pi t) \end{aligned}$$

gewählt und die rechte Seite f entsprechend angepasst. Die Druckfunktion lautet

$$p(x, y, t) = y \sin(x) \sin(2\pi t),$$

womit der Druck in diesem Beispiel die größten Ansprüche an die richtige Wahl der Randwerte stellt. Die Lösungen sind hingegen recht glatt und daher leichter zu approximieren als die in den Testproblemen I und III.

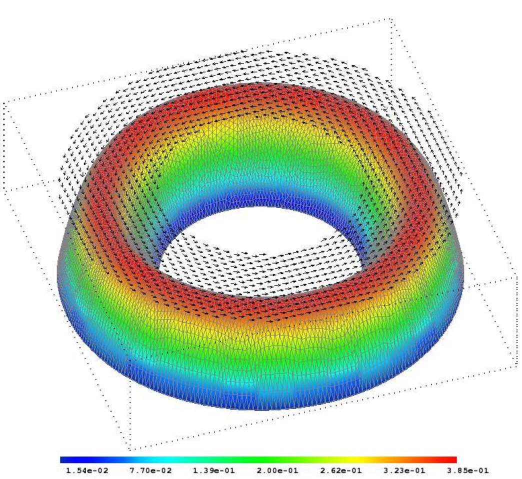


Abbildung 5.15: Geschwindigkeit $\|v\|_2$

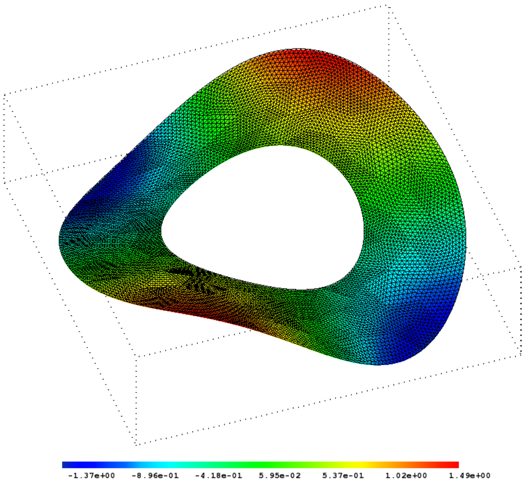


Abbildung 5.16: Druck p

5.5.3 Der Splittingfehler

Δt	Freiheitsgrade	$\nu = 1$		$\nu = 1/10$		$\nu = 1/100$	
		$\lVert f - f_h \rVert_0$	Quotient	$\lVert f - f_h \rVert_0$	Quotient	$\lVert f - f_h \rVert_0$	Quotient
$v_1 = v_2$							
0.2	66049	4.464e-04	-	1.477e-03	-	1.751e-03	-
0.1	66049	1.766e-04	2.528	4.062e-04	3.636	4.413e-04	3.968
0.05	66049	6.390e-05	2.764	1.098e-04	3.700	1.115e-04	3.957
0.025	66049	2.090e-05	3.057	2.908e-05	3.774	2.814e-05	3.964
p							
0.2	16641	1.251e-02	-	4.156e-03	-	6.504e-04	-
0.1	16641	5.071e-03	2.468	1.221e-03	3.405	1.912e-04	3.401
0.05	16641	1.850e-03	2.741	3.436e-04	3.553	4.992e-05	3.830
0.025	16641	6.060e-04	3.053	9.086e-05	3.782	2.740e-05	1.822

Tabelle 5.6: Splittingfehler

Um die Wirkung des Splittingfehlers analysieren zu können, wird das Testbeispiel II modifiziert. Wir verwenden hier als Druck die Funktion

$$p(x,y,t) = t\left(\frac{(x-0.5)^3}{3} + \frac{(y-0.5)^3}{3} - \frac{x}{4} - \frac{y}{4} + \frac{1}{4}\right),$$

womit $\nabla p = 0$ auf $\partial\Omega$ gilt und damit die homogenen Neumann-Randbedingungen ohne Genauigkeitsverlust eingesetzt werden können. Mit diesem p und der Wahl $\mu = 1$ sind sowohl der Druck als auch die Geschwindigkeit linear in der Zeit. Wir setzen also ein BDF(1) zur Zeitdiskretisierung von $\frac{\partial}{\partial t}$ ein, was im linearen Fall exakt ist. Den Fehler, welcher sich durch die Finite-Element-Diskretisierung ergibt, versuchen wir durch eine feine Diskretisierung zu

minimieren. Da wir den Stokes-Fall betrachten, treten keine Fehler durch die Behandlung des nichtlinearen Terms auf.
Es verbleiben folglich noch zwei Fehlerquellen, zum einen der Fehler, welcher bei der Gradientenrekonstruktion entsteht, und zum anderen der Splittingfehler. Die Gradientenrekonstruktion ist unabhängig von $\triangle t$ und ν, so dass bei der Variation dieser Größen lediglich der Splittingfehler beeinflusst wird. Der Extrapolant ist von erster Ordnung, also $\bar{p}^{m+1} = p^m$.

Die Veränderungen für die Fehler in Druck und Geschwindigkeit zeigen sich in Tabelle 5.6. Es fällt auf, dass sich im Druck für kleiner werdende ν die durch die Abschätzung 5.18 erwarteten Verbesserungen ergeben. Bei der Geschwindigkeit ist die Analyse zunächst komplizierter, da der L_2-Fehler größer statt kleiner wird. Der Druckaktualisierungsschritt (3.18) beeinflusst jedoch sowohl den Druck als auch die Geschwindigkeit. Der Druck ist unmittelbarer betroffen, da seine ganze Entwicklung von Zeitpunkt t^m auf t^{m+1} auf der Lösung von (3.18) beruht.
Es stellt sich nun folgende Frage: Hat sich der Gesamtfehler aus Druck und Geschwindigkeit verringert und sich dabei ggf. anders auf die beiden Größen Druck und Geschwindigkeit verteilt? Wie man der Tabelle 5.7 entnehmen kann, ist dies der Fall. Der dort angegebene Quotient des Gesamtfehlers für kleiner werdende ν liegt maximal bei 4.347. Bei einem um den Faktor $\frac{1}{10}$ verkleinerten ν scheint dies auf den ersten Blick zu gering. Bedenkt man jedoch die negativen Effekte auf die Genauigkeit der finiten Elemente, welche durch die Gradientenbildung verstärkt werden, so ist eine Reduktion von erster Ordnung in ν nicht zu erwarten.

Kommen wir zurück zur Tabelle 5.6. Für $\nu = 1$ zeigt sich in etwa eine Reduktion des Splittingfehlers von der Ordnung 1 bis ca. 1.5 nach $\triangle t$. Diese Ordnung wird von der Abschätzung (5.18) für einen Extrapolanten erster Ordnung vorhergesagt. Mit kleiner werdendem ν erhöhen sich die Reduktionsraten weiter, bis sie näherungsweise die zweite Ordnung erreichen. Der Anstieg der Ordnung ist durch sich positiv verstärkende Effekte zu erklären. In diesem Beispiel ist laut 5.27 p von zweiter Ordnung in der Zeit und v von erster. Da auf Grund des Splittingdefektterms die Druckaktualisierungsgleichung nur näherungsweise gilt, ist nicht mit der vollen Ordnung für p zu rechnen. Je kleiner der Splittingdefektterm wird, desto besser wird die Approximation von p^m, wodurch wiederum $\tilde{v}^{m+1}$ verbessert wird. Dies reduziert den Ausdruck $\|\nabla^2(v^{m+1} - \tilde{v}^{m+1})\|_0$, womit erneut der Splittingdefektterm verringert wird. Das führt zu einer positiven Rückkopplung, die den Splittingfehler wesentlich besser reduziert als die Abschätzung 5.18 erwarten lässt.

$\|p\|_2 + \|v_1\|_2 + \|v_2\|_2$					
Δt	$\nu = 1$	$\nu = 1/10$	Quotient	$\nu = 1/100$	Quotient
0.2	1.340e-2	7.110e-3	1.885	4.152e-3	1.712
0.1	5.424e-3	2.033e-3	2.668	1.073e-3	1.893
0.05	1.977e-3	5.632e-4	3.511	2.729e-4	2.063
0.025	6.478e-4	1.490e-4	4.347	8.368e-5	1.781

Tabelle 5.7: System-Splittingfehler nach ν

5.5.4 Bestimmung von Neumann-Randbedingungen für den Druck

Wie schon in Kapitel 2.4 ausgeführt wurde, ergeben sich für den Druck in der Regel inhomogene Neumann-Randbedingungen, die mittels

$$\nabla p = f - (\underbrace{\frac{\partial v}{\partial t} + (v \cdot \nabla) v}_{\diamondsuit} - \nu \nabla^2 v) \qquad \text{auf } \partial\Omega,\ t \in [0, t_{end}]b \tag{5.28}$$

bestimmt werden können.
Hierbei kann f exakt angegeben werden. Der mit $\diamondsuit$ bezeichnete Teil entfällt für die Probleme mit homogenen Dirichlet-Randbedingungen in v, ansonsten kann je nach Problem $\frac{\partial v}{\partial t}$ analytisch angegeben oder mit einer geeigneten BDF-Formel approximiert werden. Die Bildung des nichtlinearen Terms am Rand ist durch die in Kapitel 5.1.3 beschriebene Methode mit hoher Genauigkeit möglich.
Das Hauptproblem liegt darin, den Wert $\nu \nabla^2 v$ zu bestimmen. Hierbei sind mehrere Wege denkbar. Zum einen kann man ∇p mit Hilfe einer Massenmatrix

$$M(\nabla p)_i = (\frac{\partial v_i}{\partial t}, \phi)_0 + (v \cdot \nabla v_i, \phi)_0 - \nu(\nabla^2 v_i, \phi)_0 \text{ in } \Omega\ ; (i = \{1, 2\})$$

errechnen und so wie beim Galerkinverfahren die direkte Bildung von $\nabla^2 v$ vermeiden, indem man mit Hilfe der partiellen Integration ∇ auf ϕ umwälzt. Dazu müssen aber zwei Gleichungssysteme entsprechend der Größe des Finite-Element-Raumes von p zusätzlich gelöst werden.

Um die zusätzlichen CPU-Kosten für das Lösen der Gleichungssysteme zu vermeiden, werden wir ausschließlich auf eine Approximation auf der Basis der Gradientenrekonstruktion setzen. Dieser Weg ist schneller, doch leider für ∇^2 am Rand des Gebiets recht ungenau. Um eine möglichst gute Approximation zu gewährleisten, nutzen wir den Laplace-Operator in der Form (5.5), die durch die Divergenzfreiheit von v möglich wurde. Wie man in den Tabellen 5.4 und 5.5 erkennt, kann der dabei auftretende Fehler trotzdem noch groß werden. Für Fluide mit einer geringen spezifischen Viskosität ist diese Genauigkeit jedoch in der Praxis völlig ausreichend, da ν die Bedeutung des Diffusionsterms stark reduziert. Wie im Abschnitt 5.5.4 noch weiter ausgeführt wird, konnte diese Approximation für $\nu < \frac{1}{10}$ in allen Tests erfolgreich verwendet werden. $\nu \approx \frac{1}{10}$ stellt einen Grenzbereich dar, bei dem die erfolgreiche Anwendung sowohl von der Gitter-Diskretisierung als auch von der Natur der Strömung abhängt. Für $\nu > \frac{1}{10}$ werden homogene Neumann-Randbedingungen genutzt. Alternativ könnte in solchen Fällen auf die oben geschilderte Methode zurückgegriffen werden.

Damit können wir die Gleichung (5.28) im Schritt 1 des Splittingverfahrens dazu verwenden, realistische Randbedingungen für den Druck zu gewinnen. Im Schritt 3 werden weiterhin homogene Neumann-Randbedingungen verwendet, da $\nabla p - \nabla \bar{p} \approx 0$ erwartet werden darf.

Tabelle 5.8 zeigt die Fehlerreduktion auf dem Stokes-Testproblem II mit $\mu = 0$, also konstantem v. Die Ergebnisse bestätigen die von Rannacher in (3.27) postulierten Auswirkungen der homogenen Neumann-Randbedingungen auf den Druck. Für $\triangle t \to 0$ geht der Einfluss in der Fläche zurück, was sich in der L_2-Norm des Fehlers widerspiegelt. Direkt am

	Freiheits-	Null-Neumann-Randbedingungen			Angepasste Neumann-Randbedingungen		
Δt	grade	$\|f - f_h\|_0$	Quotient	$\|f - f_h\|_{max}$	$\|f - f_h\|_0$	Quotient	$\|f - f_h\|_{max}$
			v_1				
0.1	16641	2.075e-02	-	9.698e-02	1.673e-03	-	1.030e-02
0.05	16641	1.449e-02	1.432	4.358e-02	5.833e-04	2.867	1.922e-03
0.025	16641	1.044e-02	1.388	3.238e-02	2.028e-04	2.877	6.917e-04
0.0125	16641	7.464e-03	1.399	2.578e-02	7.246e-05	2.798	2.882e-04
0.00625	16641	5.461e-03	1.367	2.009e-02	2.643e-05	2.742	2.152e-04
0.003125	16641	4.284e-03	1.275	1.645e-02	1.046e-05	2.525	1.652e-04
			v_2				
0.1	16641	2.075e-02	-	9.698e-02	1.673e-03	-	1.029e-02
0.5	16641	1.449e-02	1.432	4.358e-02	5.834e-04	2.867	1.921e-03
0.025	16641	1.044e-02	1.388	3.238e-02	2.028e-04	2.877	6.913e-04
0.0125	16641	7.464e-03	1.399	2.578e-02	7.242e-05	2.800	2.877e-04
0.00625	16641	5.461e-03	1.367	2.009e-02	2.631e-05	2.753	2.147e-04
0.003125	16641	4.284e-03	1.275	1.645e-02	1.025e-05	2.567	1.647e-04
			p				
0.1	4225	5.112e-03	-	6.822e-02	4.361e-03	-	8.851e-03
0.05	4225	3.862e-03	1.324	4.130e-02	1.688e-03	2.584	5.100e-03
0.025	4225	3.016e-03	1.280	2.222e-02	1.822e-04	9.266	1.064e-03
0.0125	4225	2.233e-03	1.351	1.746e-02	6.416e-05	2.839	3.259e-04
0.00625	4225	1.569e-03	1.423	2.178e-02	5.762e-05	1.114	2.348e-04
0.003125	4225	1.078e-03	1.456	2.249e-02	5.759e-05	1.000	2.068e-04

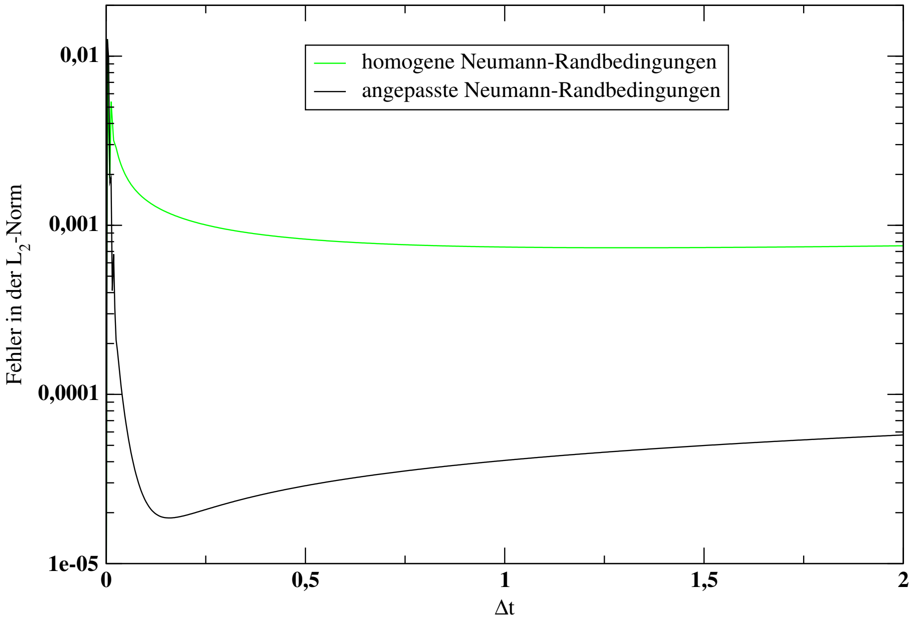


Abbildung 5.17: Vergleich der Neumann-Randbedingungen mit $\triangle t = 0.003125$, $\nu = 1/1000$

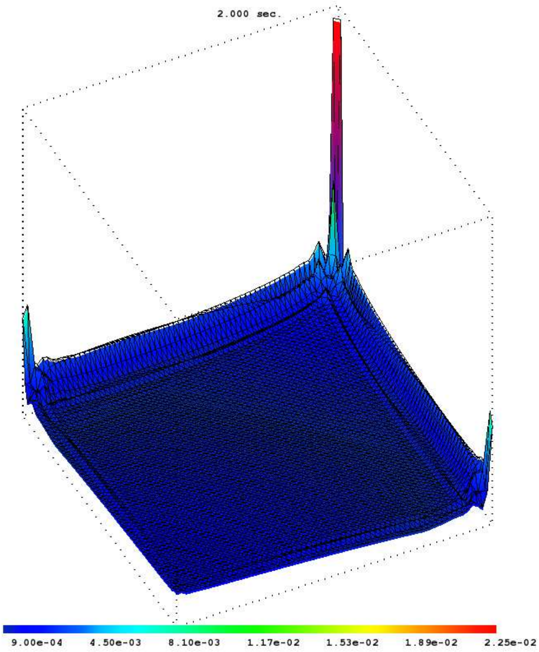


Abbildung 5.18: Fehler für homogene Neumann-Randbedingungen bei $t = 2$

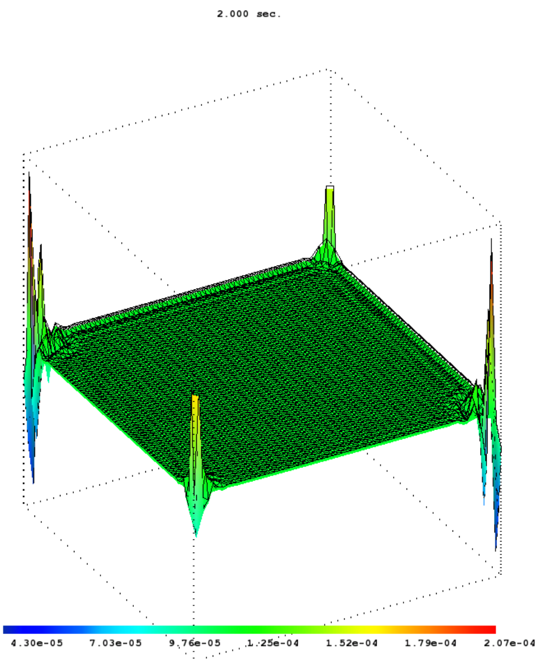


Abbildung 5.19: Fehler für angepasste Neumann-Randbedingungen bei $t = 2$

Rand bleibt der Fehler jedoch fast unverändert, wie man an der Maximumsnorm des Drucks ablesen kann. Dadurch wird jedoch die Konvergenz des Splittings insgesamt verschlechtert, so dass durch die homogenen Neumann-Randbedingungen die Reduktionsordnung verschlechtert wird.

Mit den angepassten Neumann-Randbedingungen hingegen wird, wie nach Abschnitt 5.4.2 erwartet, ein besseres Reduktionsverhalten erreicht. Dass nicht die zweite Ordnung erreicht wird, liegt am Einfluss des ortsabhängigen Fehlers, der hier auf Grund des geringen Zeitfehlers schnell an Bedeutung gewinnt. Da v konstant ist und somit ohne Genauigkeitsverlust differenziert werden kann, ist nur die Ordnung in p für die Ordnung des Splittings entscheidend.

Die Fehlerentwicklung des Splittings in der Zeit wird in Abbildung 5.17 dargestellt. Die Plots 5.18 und 5.19 zeigen den Fehler zu diesem Zeitpunkt und machen deutlich, dass der Fehler im Druck in beiden Fällen am Rand auftritt. Im Fall der angepassten Neumann-Randbedingungen hat dieser Fehler seinen Ursprung in der Rekonstruktion von $\nabla^2 v$, die besonders in den Ecken problematisch ist. Entsprechend tritt dort der größte Fehler auf. Die trotzdem im Vergleich zu den homogenen Neumann-Randbedingungen erreichte Verbesserung bewegt sich im Bereich von zwei Zehnerpotenzen.

5.5.5 Die Stokes-Gleichung

Die Abbildungen 5.20 und 5.21 zeigen den L_2-Fehler der ersten Geschwindigkeitskomponente und des Drucks im Beispiel I. Zur Diskretisierung der Zeitableitung wurde ein BDF-Verfahren dritter Ordnung verwendet. Für den Druck, der mit Hilfe von 5.10 berechnet wird, gilt immer $\bar{p}^{m+1} = p^m$. Bei weitergehenden Extrapolationen war das Verfahren nicht mehr in allen Testfällen robust. Um den Ortsfehler nicht dominant werden zu lassen, wurde nach je zwei Verfeinerungen in der Zeit eine Verfeinerung im Ort vorgenommen. Das Startgitter besteht aus der kanonischen Zerlegung des Einheitsquadrats mit 4225 Freiheitsgraden für die Geschwindigkeit und 1089 für den Druck. Das feinste durch reguläre Zerlegung der vorangegangenen erzeugte Gitter hat 66049 bzw. 16641 Freiheitsgrade.

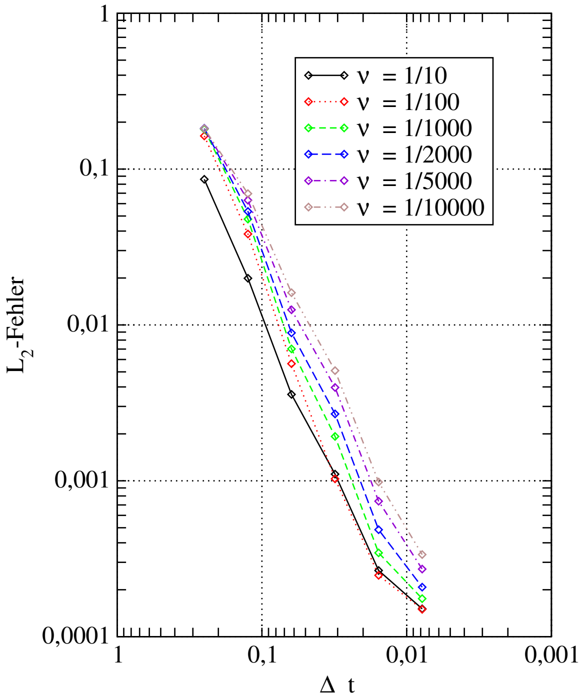


Abbildung 5.20: Ordnung des Splittings in v_1

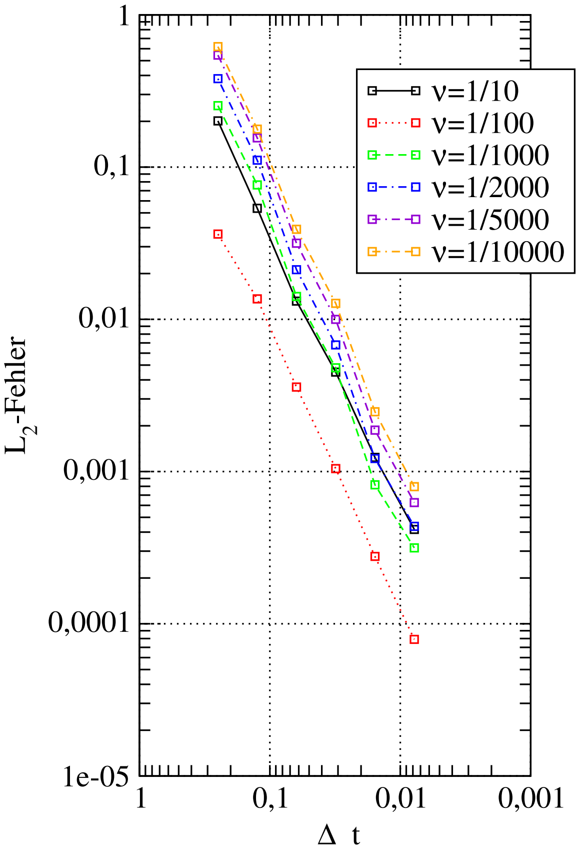


Abbildung 5.21: Ordnung des Splittings in p

Um die Ordnung des Verfahrens in der Zeit zu bestimmen, betrachten wir den Quotienten $\frac{ER_{\Delta t}(v_i)}{ER_{\Delta t/2}(v_i)}$ für die Geschwindigkeit und $\frac{ER_{\Delta t}(p)}{ER_{\Delta t/2}(p)}$ für den Druck. Der $\log_2$ des Quotienten gibt die Ordnung des Verfahrens an. Da $\Delta t = 2^{-\tau} \Leftrightarrow \log \Delta t = -\tau \log(2)$ gilt und der Fehler $ER(\Delta t) = C \cdot 2^{-\frac{\sigma}{\Delta t}}$ sich in der logarithmischen Darstellung als $\log(ER(\Delta t)) = \log(C) - \frac{\sigma}{\Delta t}) \log(2) = \log(C) + \frac{\sigma}{\tau}$ schreiben lässt, erscheint die Fehlerentwicklung in den logarithmisch skalierten Achsen als Gerade. Die Steigung dieser Geraden entspricht der Ordnung σ. Auf diese Weise lässt sich die Ordnung des Splittings visualisieren.

Die Ordnung des Verfahrens ist, wie man den Abbildungen 5.20 und 5.21 entnehmen kann, im Durchschnitt zwei. Teilweise kann eine Reduktion bis zum Faktor 6 auftreten, was etwa der Ordnung 2.5 entspricht. Dieses Verhalten entspricht den theoretischen Überlegungen aus Abschnitt 5.4.2. Da die Geschwindigkeit von dritter Ordnung ist, der Druck jedoch nur von zweiter, verringert dies die Ordnung des Splittings auf die des Drucks. Trotzdem lohnt sich der Einsatz eines BDF-Verfahrens dritter Ordnung, da der Fehler hier in der Regel niedriger ist als beim Einsatz des Verfahrens zweiter Ordnung.

	Freiheits-	Fehlerentwicklung beim FEM-Splitting			Fehlerentwicklung bei [21]	
Δt	grade	$\|f - f_h\|_0$	Quotient	$\|f - f_h\|_{max}$	$\|f - f_h\|_0$	Quotient
			v_1			
1/4	4225	2.146e-02	-	4.617e-02	k.A.	-
1/8	4225	1.134e-02	1.892	3.604e-02	k.A.	-
1/16	16641	8.019e-03	1.415	2.998e-02	k.A.	-
1/32	16641	5.319e-03	1.507	2.015e-02	9.818e-04	-
1/64	66049	3.222e-03	1.651	1.267e-02	2.513e-04	3.907
1/128	66049	1.813e-03	1.777	7.268e-03	6.333e-05	3.968
			v_2			
1/4	4225	2.022e-02	-	4.530e-02	k.A.	-
1/8	4225	1.082e-02	1.869	3.827e-02	k.A.	-
1/16	16641	7.964e-03	1.358	3.012e-02	k.A.	-
1/32	16641	5.317e-03	1.498	2.016e-02	1.053e-03	-
1/64	66049	3.221e-03	1.650	1.267e-02	2.696e-04	3.906
1/128	66049	1.813e-03	1.777	7.267e-03	6.796e-05	3.998
			p			
1/4	1089	3.544e-01	-	1.431e-00	k.A.	-
1/8	1089	3.286e-01	1.079	1.386e-00	k.A.	-
1/16	4225	2.562e-01	1.282	1.213e-00	k.A.	-
1/32	4225	1.776e-01	1.442	9.964e-01	5.831e-02	-
1/64	16641	1.144e-01	1.553	7.863e-01	2.666e-02	2.564
1/128	16641	6.991e-02	1.636	6.025e-01	9.617e-03	2.004

Tabelle 5.9: Testproblem I Ansatz BDF(3) mit $\nu = 1$ und Null-Neumann-Randbedingungen

Aus den Abbildungen ist ersichtlich, dass die Konvergenzordnung unabhängig von ν ist. Für kleine ν verschlechtert sich jedoch die Genauigkeit der Finite-Element-Approximation, so dass der Fehler für $\Delta t = \frac{1}{128}$ für kleine ν größer wird. Der dominanter werdende Ortsfehler erklärt auch die schlechtere Reduktionsordnung von $\Delta t = \frac{1}{64}$ auf $\Delta t = \frac{1}{128}$.

Der Einfluss der approximierten Neumann-Randbedingungen fällt bei der Betrachtung der Graphen für $\nu = 1/10$ und $\nu = 1/100$ auf. Während für $\nu = 1/10$ die Fehler bei der Approximation des Laplace-Operators mit Hilfe der Verfahren zur Gradientenrekonstruktion ein gutes Konvergenzverhalten verhindern, ist das Konvergenzverhalten bei $\nu = 1/100$ optimal. Die Nachteile durch den kleineren Viskositätskoeffizienten sind noch nicht nachweisbar und der Laplace-Term bei der Druckapproximation ist bereits hinreichend klein.

Der Fall $\nu = 1$ ist in der Tabelle 5.9 gesondert dargestellt, um einen Vergleich mit dem Splitting aus [21] zu ermöglichen. In diesem Fall musste aus den in Kapitel 5.5.4 dargelegten Gründen auf homogene Neumann-Randwerte zurückgegriffen werden.

5.5.6 Die Navier-Stokes-Gleichung

Konvergenzverhalten

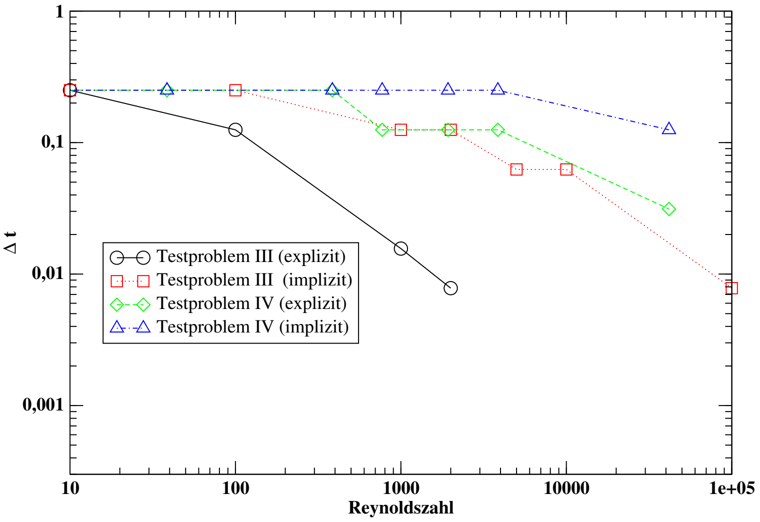


Abbildung 5.22: Maximales Δt mit BDF(3)

Die Abbildung 5.22 zeigt das maximale $\triangle t = \frac{1}{2^m+1}; m = 1, ..., 6$, bei dem das Verfahren für eine bestimmte Reynoldszahl konvergiert. Die Werte für das Testproblem III und IV sind dabei etwas versetzt, da sich hier für gleiche ν unterschiedliche Reynoldszahlen ergeben. ν wurde über einem Wertebereich von $1 \cdot 10^{-01}$ bis $1 \cdot 10^{-05}$ variiert. Dabei ergibt sich wie erwartet, dass das implizite Vorgehen größere Zeitschrittweiten erlaubt und auch robuster ist als das explizite. Beim expliziten Vorgehen ist die Analyse für große Reynoldszahlen jedoch komplexer.

CFL-Bedingung für das explizite Verfahren Bei der expliziten Behandlung des nichtlinearen Terms nimmt die Gleichung (3.29) für große Reynoldszahlen einen hyperbolischen Charakter an. Dies liegt darin begründet, dass der Laplace-Operator stark an Bedeutung verliert. Da nun der dominante Operator in (3.29) explizit behandelt wird, müssen für diese Fälle ähnliche Beschränkungen für die Schrittweite in Kauf genommen werden, wie sie durch die in [33], S.468ff ausführlich beschriebene Courant-Friedrichs-Levy-Bedingung (kurz CFL-Bedingung) für die explizite Behandlung von hyperbolischen Differentialgleichungen gelten. In diesen Fällen muss also für die Zeitschrittweite folgende Ungleichung erfüllt sein:

$$\frac{\triangle t}{h} \leq C(Re) \; . \tag{5.29}$$

Diese Beschränkung birgt für das explizite Vorgehen große Probleme. Wie wir bereits im Abschnitt 5.4.5 diskutiert haben, entstehen durch den Ableitungsfehler bei der expliziten Behandlung des nichtlinearen Terms mehr Fehler pro Zeitschritt. Dies erfordert eine bessere Diskretisierung im Ort, was aber auf Grund der Bedingung (5.29) wieder mehr Zeitschritte erfordert. Dies kann dazu führen, dass das explizite Vorgehen für einige Probleme bei großen Reynoldszahlen nicht möglich ist.

CPU-Kosten

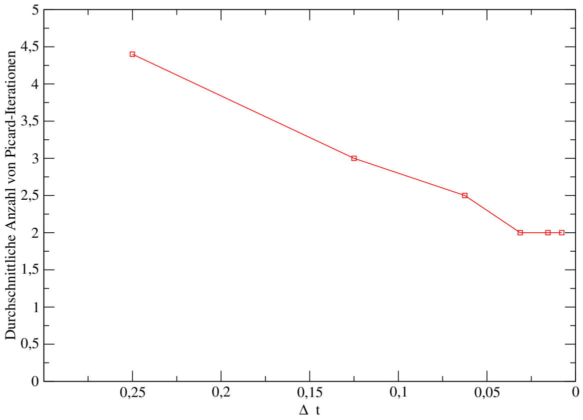


Abbildung 5.23: Testproblem IV mit $\nu = 100$ bzw. $Re \approx 39$ und BDF(3)

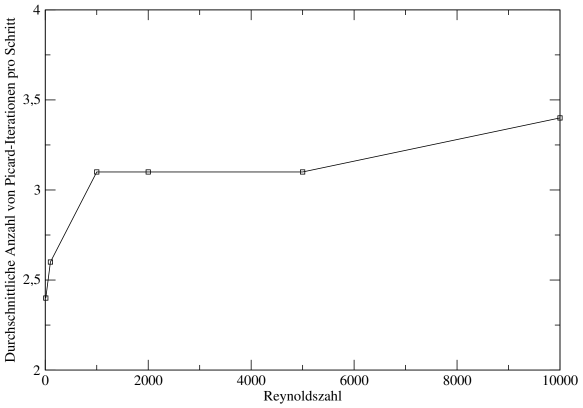


Abbildung 5.24: Testproblem III mit $\Delta t = 1/32$ und BDF(3)

Die CPU-Kosten eines Splittingschritts hängen maßgeblich von der Anzahl der Picard-Iterationen ab. Die Abbildung 5.23 zeigt die Abhängigkeit der Anzahl der Picard-Iterationen von der Zeitschrittweite. Wie zu erwarten war, werden für kleine Zeitschrittweiten weniger Picard-Iterationen benötigt. Ebenso verhält es sich, wie Abbildung 5.24 zeigt, für kleine Reynoldszahlen.

Eine wichtige Schlussfolgerung ist, dass eine Verdoppelung der Zeitschritte nicht notwendigerweise zu einer Verdoppelung der CPU-Kosten führt.

Der größte Teil der CPU-Kosten entsteht beim Lösen der Gleichungssysteme, der zweitgrößte bei der Gradientenrekonstruktion. Die Lösung der Gleichungssysteme ist unterschiedlich aufwendig. Die symmetrischen Gleichungssysteme, die bei der Druckberechnung auftreten, sind die billigsten, da sie die kleinsten sind. Für die linearen Gleichungssysteme, die bei der expliziten Behandlung des nichtlinearen Terms auftreten, existieren ebenfalls gute Löser, wie schon in Kapitel 4.5 besprochen wurde.

Weiterhin ist es lohnenswert, für diese Gleichungen viel in einen guten Vorkonditionierer zu investieren, da bei einer äquidistanten Schrittweite die Galerkinmatrix für jeden Zeitschritt gleich ist und somit - abhängig von der Natur des Fehlerschätzers - ohne Neuaufwand immer die gleiche Vorkonditionierungsmatrix verwendet werden kann. Dies ist bei einem impliziten Vorgehen nicht möglich, da sich k aus Gleichung 4.3 in jedem Schritt ändert und somit der Konvektionsanteil der Galerkinmatrix in jedem Schritt neu assembliert werden muss.

Soll das Splitting auf Parallelrechnern implementiert werden, müssen sowohl die Gleichungssystemlöser als auch die Gradientenrekonstruktion parallelisiert werden. Eine effiziente Parallelisierung der Gradientenrekonstruktion ist möglich, da die Gradienten an jedem Knoten unabhängig von den anderen Knoten rekonstruiert werden können. Die Arbeit kann also direkt auf die vorhandenen Prozessoren aufgeteilt werden.

Für die symmetrischen Gleichungssysteme stehen eine Reihe von Verfahren z.B. aus dem Bereich der Gebietszerlegung bzw. Domain Decomposition zur parallelen Lösung zur Verfügung. Parallele Algorithmen für die nichtsymmetrischen Gleichungssysteme werden ebenfalls entwickelt, z.B. von Dahmen, Müller und Schlinkmann [8]. Diese sind jedoch aufwendiger als die Verfahren für symmetrische Matrizen.

Ordnung des Verfahrens

Die Abbildungen 5.25 bis 5.32 zeigen die Reduktionsordnung des Splittings bei den Testproblemen III und IV. Wie schon bei der Stokes-Gleichung im Abschnitt 5.5.5 zeigt sich, dass das Verfahren von zweiter Ordnung in der Zeit ist. Für große ν Werte verliert das Verfahren für schwierig zu approximierende Neumann-Randwerte an Effizienz.

Mehr noch als bei der Stokes-Gleichung treten bei Problemen mit einer geringen spezifischen Viskosität und damit steigender Reynoldszahl die Fehler durch die Finite-Element-Diskretisierung in den Vordergrund.

Nimmt man an, dass die Fehler in Zeit (E_Z) und Ort (E_O), wie es die Abschätzung aus Satz 4.14 schon für einfache parabolische Probleme verdeutlicht hat, additiv im Sinne von

$$E_G = E_Z + E_O$$

verknüpft sind, so kann man bei einer Reduktion des zeitabhängigen Fehlers von der Ordnung l nur für den Fall $E_O \ll E_Z$ eine ähnliche Reduktion des gesamten Fehlers (E_G) erwarten. $E_O \ll E_Z$ ist jedoch für kleine Zeitschrittweiten und große Reynoldszahlen nicht mehr zu erwarten, was die geringeren Reduktionsraten erklärt, die in den Abbildungen 5.25 bis 5.32 zu sehen sind.

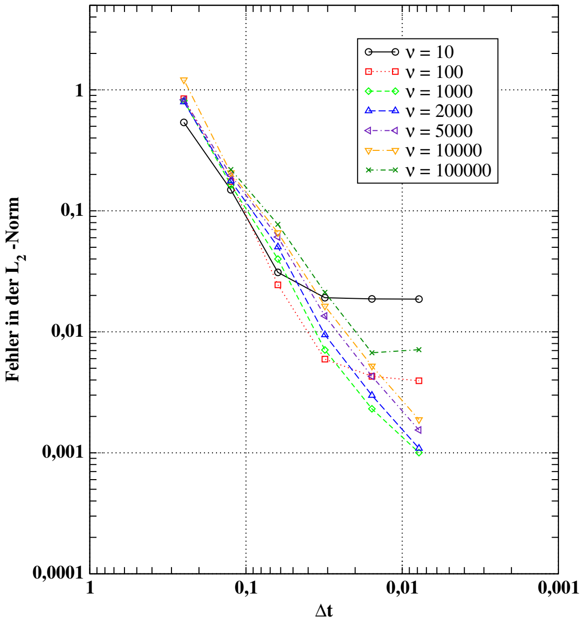


Abbildung 5.25: Ordnung von v_1 im Testproblem IV (implizit)

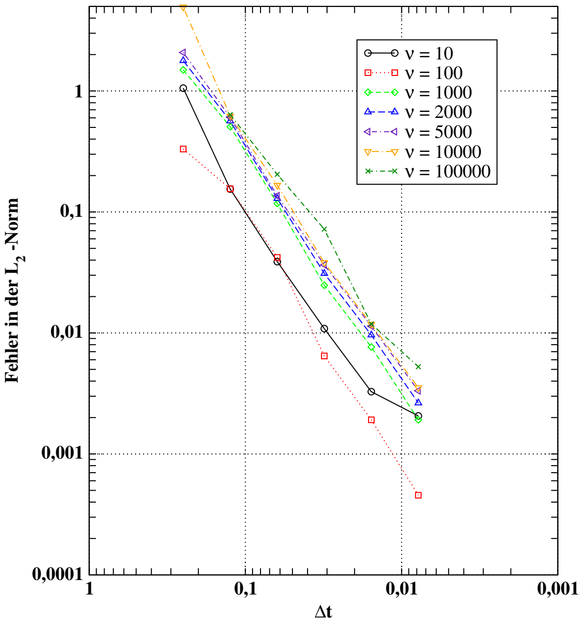


Abbildung 5.26: Ordnung des Drucks im Testproblem IV (implizit)

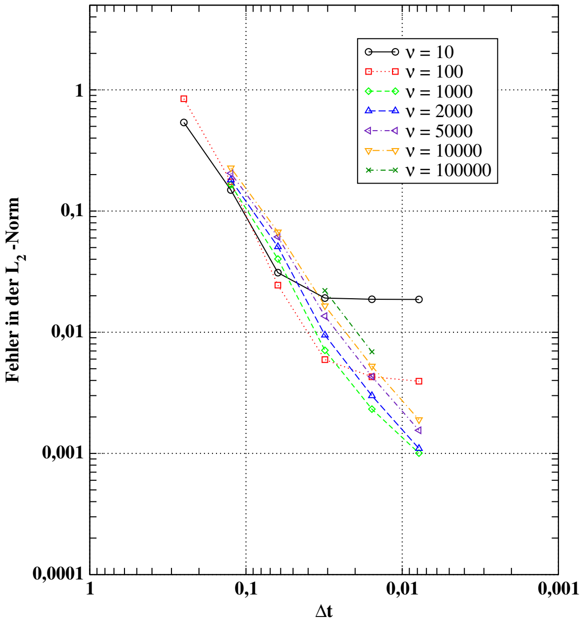


Abbildung 5.27: Ordnung von v_1 im Testproblem IV (explizit)

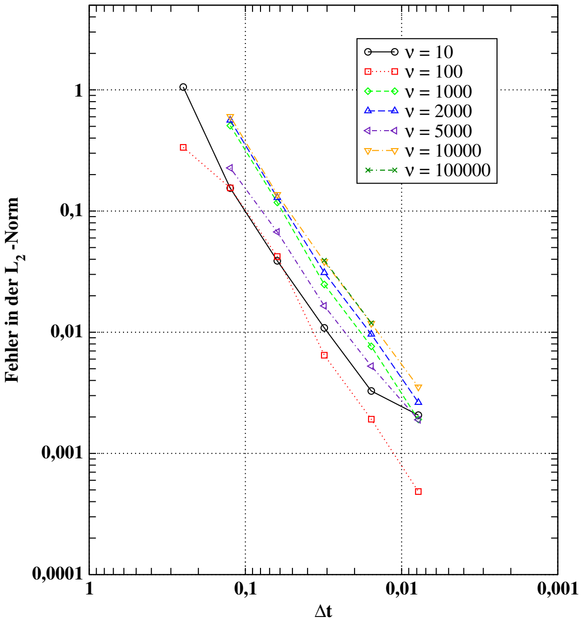


Abbildung 5.28: Ordnung des Drucks im Testproblem IV (explizit)

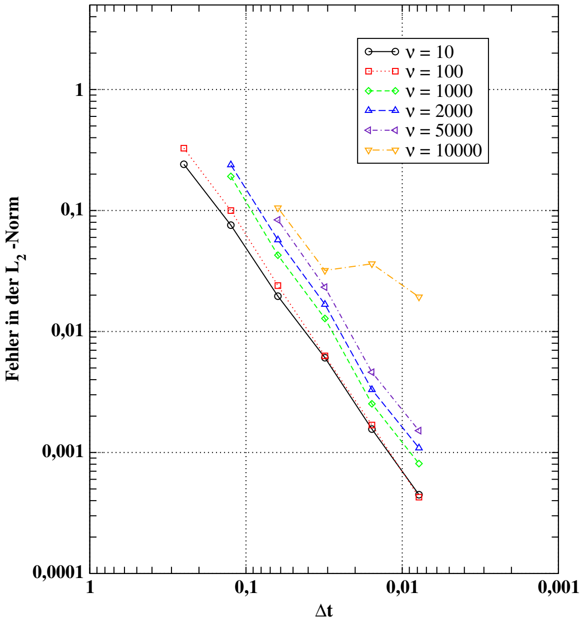


Abbildung 5.29: Ordnung von v_1 im Testproblem III (implizit)

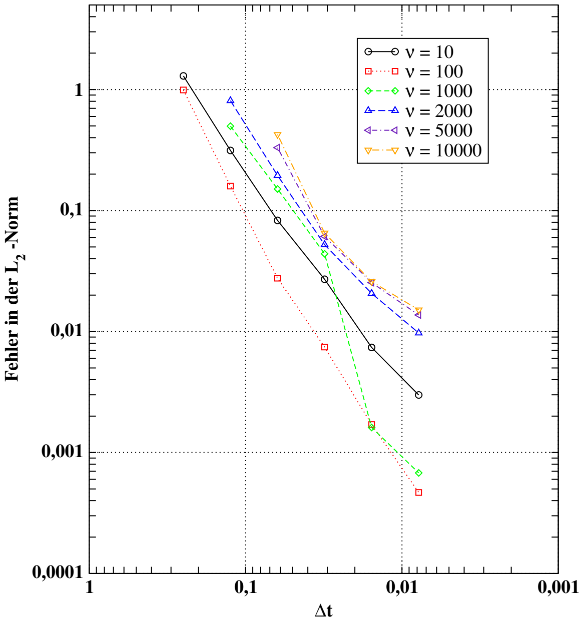


Abbildung 5.30: Ordnung des Drucks im Testproblem III (implizit)

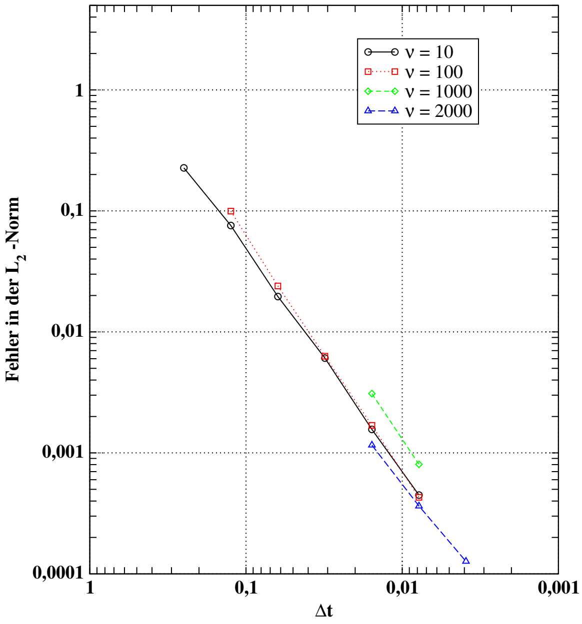


Abbildung 5.31: Ordnung von v_1 im Testproblem III (explizit)

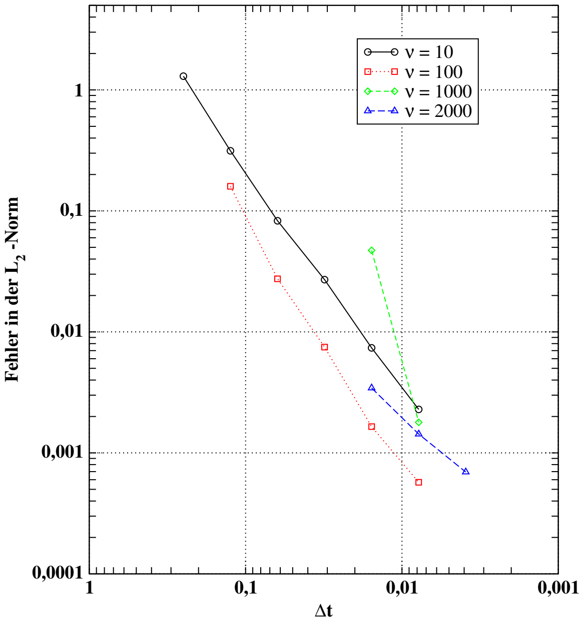


Abbildung 5.32: Ordnung des Drucks im Testproblem III (explizit)

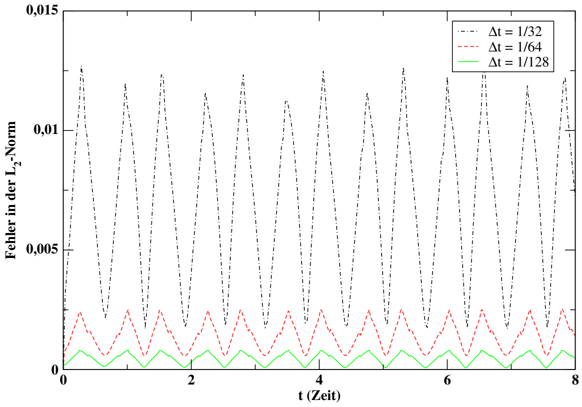


Abbildung 5.33: Fehlerverlauf bei Testproblem III mit $\nu = 1/1000$

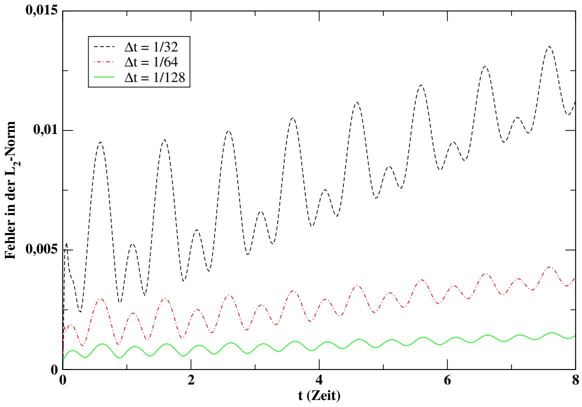


Abbildung 5.34: Fehlerverlauf bei Testproblem IV mit $\nu = 1/5000$

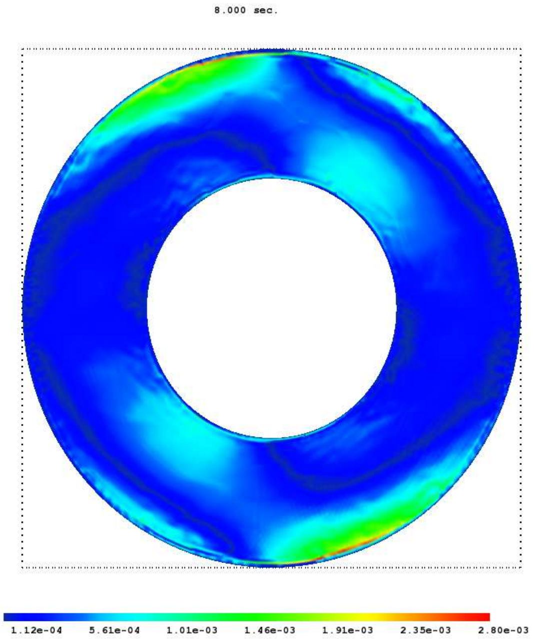


Abbildung 5.35: Fehler in v_1 BDF(3); implizit beim Testproblem IV zum Zeitpunkt $t = 4$ mit $\nu = 1/5000$

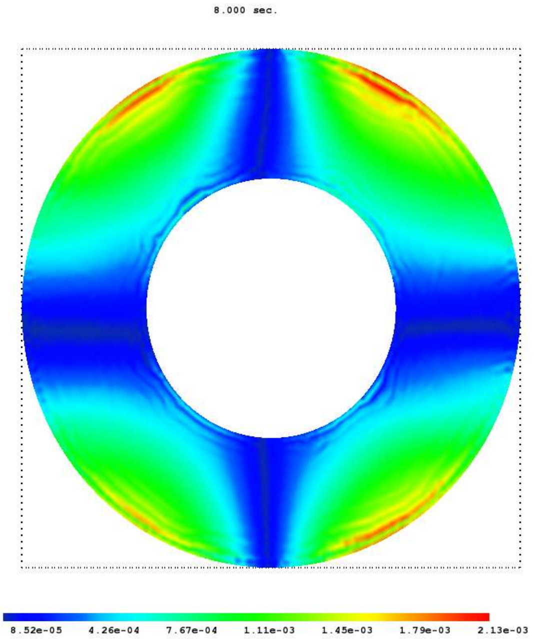


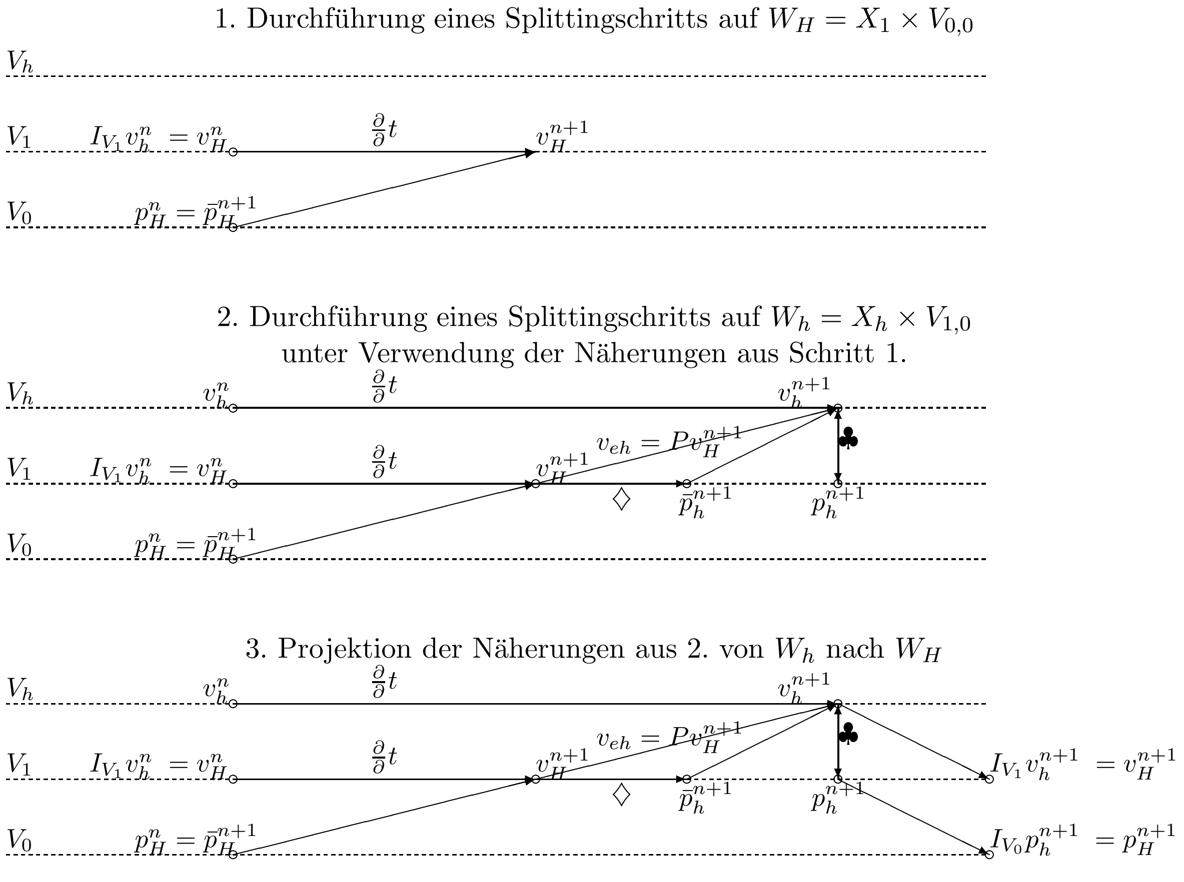
Abbildung 5.36: Fehler in p BDF(3); implizit beim Testproblem IV zum Zeitpunkt $t = 4$ mit $\nu = 1/5000$

6 Multigrid-Postprocessing

Im folgenden Kapitel werden Techniken untersucht, um mit ihnen im Anschluss an einen Splittingschritt die berechnete Lösung $(v_H, p_H) \in W_H$ zu verbessern.

6.1 Grundalgorithmus

Abbildung 6.1 zeigt die Grundstruktur der verwendeten Vorgehensweise, wobei die Druckkonstruktion über die Gleichung (5.10) mit $\diamondsuit$ und über den Druckaktualisierungsschritt mit $\clubsuit$ bezeichnet wird. Des Weiteren gilt $X_1 = V_1 \times V_1$ und $X_h = V_h \times V_h$.


1. Durchführung eines Splittingschritts auf $W_H = X_1 \times V_{0,0}$

2. Durchführung eines Splittingschritts auf $W_h = X_h \times V_{1,0}$ unter Verwendung der Näherungen aus Schritt 1.

3. Projektion der Näherungen aus 2. von W_h nach W_H

Abbildung 6.1: Schematische Darstellung des Postprocessings

Mit V_0 wird der Finite-Element-Raum bezeichnet, in dem der Druck approximiert wird; derjenige, in dem die Approximation der Geschwindigkeit berechnet wird, hingegen mit V_1.

Sei $V_h \supset V_1$ ein Finite-Element-Raum, so dass mit $W_h = V_h \times V_{1,0}$ die inf-sub-Bedingung aus Kapitel 4.4 erfüllt ist. Dabei ist der Raum $V_{1,0}$ bzw. $V_{h,0}$ der Unterraum von V_1 bzw. V_h, für dessen Elemente u

$$\int_\Omega u\, dx = 0$$

gilt.

Zunächst wird die Lösung in dem Raum $W_H = X_1 \times V_{0,0}$ berechnet. Hierbei kann auf eine Konstruktion des Drucks mit der Gleichung (5.10) verzichtet werden, da die notwendige Stabilitätsverbesserung durch die Neukonstruktion des Drucks im Postprocessingschritt erfolgt. Hierbei wird die Geschwindigkeit v_H^{n+1} benutzt, um mit Hilfe der Gleichung (5.10) eine Näherung an den aktuellen Druck $\bar{p}_h^{n+1}$ in $V_{1,0}$ zu berechnen. Dabei ist es nun möglich, f^{n+1} sowie v_H^{n+1} statt f^n und v_H^n in (5.10) zu verwenden. Besonders für große Schrittweiten ergibt sich hierdurch eine bessere Startnäherung für den Druck mit aktuelleren Neumann-Randbedingungen.

Mit diesem Druck wird nun ein weiterer Splittingschritt als Postprocessing angeschlossen. Durch die genauere Drucknäherung sowie die Startnäherung $v_{h_e} = Pv_H^{n+1}$ erreicht man einen Zuwachs an Genauigkeit.

Im Anschluss an das Postprocessing wird die Lösung in W_h mittels I_{V_1} bzw. I_{V_0} nach W_H projiziert. Dadurch wird die Entwicklung der Lösung in W_H ebenfalls durch das Postprocessing verbessert. Die Lösung in W_h wird gespeichert, um im nächsten Schritt für die Approximation von $\frac{\partial}{\partial t}$ in V_h durch das BDF-Verfahren genutzt zu werden.
Da bei der Konstruktion des Drucks mittels der Gleichung (5.10), wie Abbildung 5.8 zeigt, kurzfristig Genauigkeit im Vergleich zu der Näherung verloren geht, die aus dem Druckaktualisierungsschritt hervorgeht, bietet der Splittingschritt in W_H den Vorteil, dass darauf verzichtet werden kann.

Durch die Konstruktion von $\bar{p}_h^{n+1}$ auf der Basis von f^{n+1} und v_H^{n+1} wird die Drucknäherung verbessert. Mit einem BDF-Verfahren dritter Ordnung ist für die Stokes-Gleichung nach Kapitel 5.4.2 klar, dass mit einer Drucknäherung von zweiter Ordnung in der Zeit ein Verfahren dritter Ordnung in der Zeit möglich ist. In wieweit durch das Multigrid-Postprocessing hierzu die Voraussetzungen geschaffen werden, soll im Folgenden untersucht werden.

Bemerkung 6.1 (Generelle Voraussetzung)
Es sei für die folgenden Betrachtungen vorausgesetzt, dass $v_i \in C^2(\bar{\Omega})$ $i = 1, 2$ ist.

Bemerkung 6.2
Sei die Navier-Stokes-Gleichung mit homogenen Null-Dirichlet-Randbedingungen am Rand gegeben. Die Drucknäherung wird dann durch die Lösung der folgenden partiellen Differentialgleichung berechnet, wobei die partiellen Ableitungen in (6.1) mit Hilfe eines Gradientenrekonstruktionsverfahrens bestimmt werden.

$$-\nabla^2 \bar{p}^{n+1} = -(f_{1x}^n + f_{2y}^n) + v_{H1x}^n v_{H1x}^n + 2 v_{H2x}^n v_{H1y}^n + v_{H2y}^n v_{H2y}^n \quad \text{in } \Omega, \tag{6.1}$$

$$\nabla \bar{p}^{n+1} = f^{n+1} + \nu G^2 v_H^{n+1} \quad \text{auf } \partial\Omega\,, \tag{6.2}$$

$$\int_\Omega \bar{p}\, dx = 0 \tag{6.3}$$

Die Ordnung der Drucknäherung hängt von der Ordnung ab, mit der die Differenz aus der exakten Druckkonstruktion

$$\begin{aligned} -\nabla^2 p^{n+1} &= -(f^n_{1x}+f^n_{2y}) + v^n_{1x}v^n_{1x} + 2v^n_{2x}v^n_{1y} + v^n_{2y}v^n_{2y} \quad \textit{in } \Omega, && (6.4)\\ \nabla p^{n+1} &= f^{n+1} + \nu\nabla^2 v^{n+1} \quad \textit{auf } \partial\Omega\ , && (6.5)\\ \int_\Omega p\ dx &= 0 && (6.6) \end{aligned}$$

und der Konstruktion von $\bar{p}$, (6.1) - (6.3), gegen

$$\begin{aligned} -\nabla^2(p^{n+1}-\bar{p}^{n+1}) &= 0 \quad \textit{in } \Omega, && (6.7)\\ \nabla(p^{n+1}-\bar{p}^{n+1}) &= 0 \quad \textit{auf } \partial\Omega\ , && (6.8)\\ \int_\Omega (p^{n+1}-\bar{p}^{n+1})\ dx &= 0\ , && (6.9) \end{aligned}$$

was $p^{n+1}-\bar{p}^{n+1}=0$ impliziert, geht.

Definition 6.3
Sei $f \in V_h$ und M die Menge aller Knoten einer Triangulierung, die auf dem Rand von Ω liegen, das ist

$$\|f\|_{\max,h,\partial\Omega} := \max_{m\in M}\{|f(m)|\}\ .$$

Satz 6.4
Sei die Navier-Stokes-Gleichung mit homogenen Null-Dirichlet-Randbedingungen gegeben. Des weiteren setzen wir voraus, dass das in Kapitel 5 vorgestellte Splitting ein Verfahren zweiter Ordnung in der Zeit und im Ort ist, und zwar in dem Sinne, dass $\|Iv - v_h\|_0 \leq O(h^2) + O(\triangle t^2)$ gilt, wobei $I : V \to V_h$ hier wieder den Interpolationsoperator bezeichnet. Dann gilt :

$$\begin{aligned} \|\nabla^2(p^{n+1}-p_H^{n+1})\|_0 &\leq O(h^2) + C_2C_V + O(\triangle t^2)\cdot O(h^{-2}) && (6.10)\\ \|\nabla(p^{n+1}-\bar{p}^{n+1})\|_{\max,h,\partial\Omega} &= \nu\|\nabla^2 v - G^2 v_H\|_{\max,h,\partial\Omega} && (6.11) \end{aligned}$$

Beweis :
Sei p^{n+1} der exakte Druck zum Zeitpunkt t^{n+1}. Durch Subtraktion der in Bemerkung 6.2 aufgeführten Randbedingungen erhält man :

$$\|\nabla(p^{n+1}-\bar{p}^{n+1})\|_{\max,h,\partial\Omega} = \nu\|\nabla^2 v - G^2 v_H\|_{\max,h,\partial\Omega}\ .$$

Nun betrachten wir die beiden folgenden Gleichungen:

$$\begin{aligned} -\nabla^2 p^{n+1} &= -(f^{n+1}_{1x}+f^{n+1}_{2y}) + v^{n+1}_{1x}v^{n+1}_{1x} + 2v^{n+1}_{2x}v^{n+1}_{1y} + v^{n+1}_{2y}v^{n+1}_{2y}\ , && (6.12)\\ -\nabla^2 \bar{p}_h^{n+1} &= -(f^{n+1}_{1x}+f^{n+1}_{2y}) + v^{n+1}_{H1x}v^{n+1}_{H1x} + 2v^{n+1}_{H2x}v^{n+1}_{H1y} + v^{n+1}_{H2y}v^{n+1}_{H2y}\ . && (6.13) \end{aligned}$$

Subtrahiert man nun (6.12) und (6.13), so erhält man:

$$\begin{aligned} \|\nabla^2(p^{n+1}-p_H^{n+1})\|_0 &= \|(v^{n+1}_{1x}v^{n+1}_{1x} - v^{n+1}_{H1x}v^{n+1}_{H1x}) + (v^{n+1}_{2y}v^{n+1}_{2y} - v^{n+1}_{H2y}v^{n+1}_{H2y})\\ &+ 2(v^{n+1}_{2x}v^{n+1}_{1y} - v^{n+1}_{H2x}v^{n+1}_{H1y})\|_0\\ &\leq \underbrace{\|v^{n+1}_{1x}v^{n+1}_{1x} - v^{n+1}_{H1x}v^{n+1}_{H1x}\|_0}_{I.} + \underbrace{\|v^{n+1}_{2y}v^{n+1}_{2y} - v^{n+1}_{H2y}v^{n+1}_{H2y}\|_0}_{II.}\\ &+ \underbrace{2\|v^{n+1}_{2x}v^{n+1}_{1y} - v^{n+1}_{H2x}v^{n+1}_{H1y}\|_0}_{III.} \end{aligned}$$

Mit Hilfe der umgekehrten Dreiecksungleichung und Satz 5.4 erhalten wir:

$$\begin{align} & \|v_{H1x}^{n+1}\|_0 - \|v_{1x}^{n+1}\|_0 \le \|v_{1x}^{n+1} - v_{H1x}^{n+1}\|_0 \le O(h^2) + C_2 + O(\triangle t^k) \cdot O(h^{-2}) \tag{6.14} \\ \Leftrightarrow \quad & \|v_{H1x}^{n+1}\|_0 \le \|v_{1x}^{n+1}\|_0 + O(h^2) + C_2 + O(\triangle t^k) \cdot O(h^{-2}) \tag{6.15} \end{align}$$

Nun beginnen wir mit der Abschätzung für $I.$, die Abschätzung für den Term $II.$ läuft analog.

$$\begin{aligned} & \|v_{1x}^{n+1} v_{1x}^{n+1} - v_{H1x}^{n+1} v_{H1x}^{n+1}\|_0 = \|(v_{1x}^{n+1} - v_{H1x}^{n+1})(v_{1x}^{n+1} + v_{H1x}^{n+1})\|_0 \\ \le \quad & \|v_{1x}^{n+1} - v_{H1x}^{n+1}\|_0 \|v_{1x}^{n+1} + v_{H1x}^{n+1}\|_0 \\ \underbrace{\le}_{Satz 5.4} \quad & (O(h^2) + C_2 + O(\triangle t^k) \cdot O(h^{-2})) \|v_{1x}^{n+1} + v_{H1x}^{n+1}\|_0 \\ \underbrace{\le}_{(6.14)} \quad & (O(h^2) + C_2 + O(\triangle t^k) \cdot O(h^{-2}))(\underbrace{\|2v_{1x}^{n+1}\|_0}_{=C_{1V}} + O(h^2) + C_2 + O(\triangle t^k) \cdot O(h^{-2})) \\ \le \quad & O(h^2) + C_2 C_{1V} + O(\triangle t^k) \cdot O(h^{-2}) \end{aligned}$$

Mit $C_{2V} = \max\{\|v_{1y}^{n+1}\|_0, \|v_{H2x}^{n+1}\|_0\}$ folgt nun:

$$\begin{aligned} \|v_{2x}^{n+1} v_{1y}^{n+1} - v_{H2x}^{n+1} v_{H1y}^{n+1}\|_0 &= \|v_{2x}^{n+1} v_{1y}^{n+1} - v_{H2x}^{n+1} v_{1y}^{n+1} + v_{H2x}^{n+1} v_{1y}^{n+1} - v_{H2x}^{n+1} v_{H1y}^{n+1}\|_0 \\ &\le \|v_{2x}^{n+1} v_{1y}^{n+1} - v_{H2x}^{n+1} v_{1y}^{n+1}\|_0 + \|v_{H2x}^{n+1} v_{1y}^{n+1} - v_{H2x}^{n+1} v_{H1y}^{n+1}\|_0 \\ &\le \|v_{1y}^{n+1}\|_0 \|v_{2x}^{n+1} - v_{H2x}^{n+1}\|_0 + \|v_{H2x}^{n+1}\|_0 \|v_{1y}^{n+1} - v_{H1y}^{n+1}\|_0 \\ &\le O(h^2) + C_2 C_{2V} + O(\triangle t^k) \cdot O(h^{-2}) \end{aligned}$$

Mit $C_V = \frac{\max\{C_{2V}, C_{1V}\}}{3}$ folgt somit die Behauptung.

□

Bemerkung 6.5
Wie schon in Kapitel 5.1.4 klar wurde, kann im Allgemeinen nicht davon ausgegangen werden, dass

$$\|\nabla^2 v - G^2 v_h\|_{\max,h,\partial\Omega} \to 0 \quad (h \to 0)$$

gilt. In den numerischen Tests hat sich jedoch herausgestellt, dass

$$\|\nabla^2 v - G^2 v_h\|_{\max,h,\partial\Omega} \to C_1 \quad (h \to 0)$$

gilt. Damit erhält man:

$$\nu \|\nabla^2 v - G^2 v_h\|_{\max,h,\partial\Omega} \le \nu C_1 \to 0 \qquad (\nu \to 0) \, .$$

Bemerkung 6.6 (Praktische Größenordnungen)
Für kleiner werdende ν verliert der Term $\nu \|\nabla^2 v - G^2 v_H\|_{\max,h,\partial\Omega}$ zunehmend an Bedeutung und führt nur noch zu geringen Störungen.

Bemerkung 6.7 (Praktische Auswirkungen des Satzes 6.4)
Der Satz 6.4 besagt, dass sich für eine feste Triangulierung, also für konstantes h und hinreichend kleines ν, der zeitabhängige Fehleranteil bei der Druckkonstruktion von zweiter Ordnung verringert. Dies schafft laut Kapitel 5.4.2 die Voraussetzungen für ein Verfahren dritter Ordnung.

Werden h und $\triangle t$ verringert werden gelten die in Bemerkung 5.9 gemachten Aussagen.

Bei diesen Überlegungen wurde die Projektion der Resultate aus dem Raum W_h in den Raum W_H noch nicht bedacht, so dass die Möglichkeit besteht, dass sich die Ordnung des Basissplittings erhöht, was schließlich wieder dem Gesamdtalgorithmus zu gute kommt, so das in Einzelfällen auch Reduktionsraten oberhalb der dritten Ordnung denkbar sind.

Bemerkung 6.8 (Allgemeine Randbedingungen)
Liegen wie bei Testproblem III keine homogenen Nullrandbedingungen vor, so treten am Rand durch die Approximation von $(v \cdot \nabla)v$ durch $(v_H \cdot G_T)v_H$ hinzu, so das diese Probleme im Allgemeinen schwerer zu behandeln sind.

Der Splittingschritt im Raum W_H kann alternativ auch als Vorkonditionierer für den Splittingschritt in W_h betrachtet werden. Die Durchführung dieses Splittingschritts kann die Anzahl der Picard-Iterationen für den Splittingschritt in W_h senken, was zur Reduktion der CPU-Kosten beiträgt.

Im Folgenden betrachten wir zwei Varianten. In der ersten wählen wir V_h als den Finite-Element-Raum mit dem Gitter von V_1, jedoch mit quadratischen Ansatzfunktionen, wodurch sich ein Taylor-Hood-Element vom Typ P2/P1 ergibt. In der zweiten Variante geht V_h durch eine globale, reguläre Verfeinerung aus V_1 hervor, wodurch wir die lineare Variante des Taylor-Hood-Elements erhalten.

6.2 Explizite Strategie mit P2/P1 Taylor-Hood-Elementen

Wir betrachten zunächst die Postprocessingvariante mit $P2/P1$ Taylor-Hood-Elementen und einer expliziten Behandlung des nichtlinearen Terms.

Vorbereitung für Splittingverfahren mit Postprocessing:

0. Berechne eine Startnäherung $p_h^0 = \bar{p}_h^1$ für den Druck durch (5.10)

Ein Zeitschritt im Splittingverfahren mit Postprocessing:

1. Löse die implizit Gleichung für die Zwischengeschwindigkeit $\tilde{v}_{l_{h/2}}^{n+1}$ mit linearen finiten Elementen
2. Löse für die Druckaktualisierung die Gleichung $-\nabla^2 p_{h_{update}} = -\frac{\beta_0}{\triangle t}\nabla \cdot \tilde{v}_{l_{h/2}}^{n+1}$
3. Sorge für die Divergenzfreiheit in v mittels $\hat{v}_{l_{h/2}}^{n+1} = \tilde{v}_{l_{h/2}}^{n+1} + \frac{\triangle t}{\beta_0}\nabla p_{h_{update}}$
4. Berechne durch (5.10) eine Näherung $\bar{p}_{h/2}^{n+1}$ mit Hilfe von $\hat{v}_{l_{h/2}}^{n+1}$ und f^{n+1}
5. Löse analog zum Vorgehen bei der expliziten Behandlung des nichtlinearen Terms die Gleichung für die Zwischengeschwindigkeit $\tilde{v}_q^{n+1}$ mit quadratischen Ansatzfunktionen und dem Startwert $v_e = P_q \hat{v}_{l_{h/2}}^{n+1}$ sowie $\bar{p}_{h/2}^{n+1}$
6. Löse für die Druckaktualisierung die Gleichung $-\nabla^2 p_{h/2_{update}} = -\frac{\beta_0}{\triangle t}\nabla \cdot \tilde{v}_q^{n+1}$ und berechne $p_{h/2}^{n+1} = \bar{p}_{h/2}^{n+1} + p_{h/2_{update}}$
7. Sorge für die Divergenzfreiheit in v mittels $v_q^{n+1} = \tilde{v}_q^{n+1} + \frac{\triangle t}{\beta_0}\nabla p_{h/2_{update}}$
8. Berechne die Restriktionen $v_{l_{h/2}}^{n+1} = I_l v_q^{n+1}$ und $\bar{p}_h^{n+2} = p_h^{n+1} = I p_{h/2}^{n+1}$

Laut [6], S.73, Satz 6.4 und S.85, Bemerkung 7.4 kann für die finiten Elemente mit quadratischen Ansatzfunktionen bei hinreichend glatter Lösung mit deutlich verbesserten Approximationseigenschaften gerechnet werden. Im Falle einer H^3-Regularität des Problems wären für die quadratischen Dreieckselemente eine Fehlereduktion von dritter Ordnung möglich. Dies ist jedoch nicht auf dem gesamten Gebiet zu erwarten, da hierzu $\partial\Omega$ in der Regel glatt sein müsste, was mit einem Polygonzug nicht erreicht werden kann.

Das Ziel, eine genauere Approximation zu gewinnen, soll bei dieser Strategie zusätzlich zu den im Abschnitt 6.1 besprochenen Effekten durch die quadratischen Ansätze erreicht werden. Des Weiteren besteht die Möglichkeit, dass die aktuellere Startnäherung $v_e = P_q \hat{v}^{n+1}_{l_{h/2}}$ als Vorkonditionierer auf den Splittingschritt wirkt und somit das explizite Vorgehen für größere Zeitschrittweiten verwendet werden kann. Wie bereits in Kapitel 5.5.1 erwähnt wurde, sind die quadratischen finiten Elemente mit Hilfe einer hierarchischen Basis implementiert worden. P_q ist dabei ein Prolongationsoperator, der analog zu dem entsprechenden Operator aus Kapitel 4.5.1 definiert wird. Das bedeutet, dass die Werte an den Freiheitsgraden des quadratischen Anteils der hierarchischen Basis linear interpoliert werden. Inwieweit mit der so berechneten Näherung größere Zeitschrittweiten möglich sind, hängt u.a. davon ab, wie gut die Qualität der linear interpolierten Werte an den zusätzlichen Freiheitsgraden ist. Falls das Verhalten der Funktion zwischen den beiden Knoten aus dem Vorkonditionierungsschritt stark von linearen Ansätzen abweicht, ist dies nicht zu erwarten. Anderenfalls könnten erhebliche Verbesserungen im Konvergenzverhalten des expliziten Vorgehens erreicht werden.

Der explizite Ansatz ist jedoch trotz der Vorkonditionierung durch das implizite Verfahren auf W_H nicht auf alle Testproblemen für höhere ($\geq$ 2000) Reynoldszahlen robust. Eine wesentliche Reduktion der Schrittweitenbegrenzung konnte numerisch nicht nachgewiesen werden.

Tabelle 6.1: Bedeutung der Triangulierung; Testproblem IV mit BDF(3)

	Fehler bei $\nu = 1/5000$ bzw. $Re \approx 1925$			
Δt	Freiheitsgrade	$\|f - f_h\|_{L^2}$	Freiheitsgrade	$\|f - f_h\|_{L^2}$
	v_1			
1/32	7472/29408	4.071e-03	29408/116672	div.
1/64	7472/29408	9.901e-04	29408/116672	5.148e-04
1/128	7472/29408	7.322e-04	29408/116672	2.226e-04
1/256	7472/29408	div.	29408/464768	1.608e-04

Tabelle 6.1 zeigt die mit diesem Vorgehen erzielten Ergebnisse bei einer Reynoldszahl von 2000 und damit an den Grenzen des Einsatzbereichs der expliziten Behandlung des nichtlinearen Terms. Hier divergiert das Verfahren für $\Delta t = 1/32$ nach einer Gitterverfeinerung. Als Begründung kommen zum einen die CFL Bedingung und zum anderen die in Satz 6.4 bzw. Bemerkung 6.7 geschilderten Effekte in Frage. Die Tatsache, dass dieses Verhalten jedoch ausschließlich bei der expliziten Splittingvariante auftritt und auch hier erst für höhere Reynoldszahlen läßt auch die CFL Bedingung schließen.

Für kleine Zeitschrittweiten sieht man hingegen, dass das Verfahren auf zu groben Gittern nicht stabil ist, da sich hierbei, zusätzlich zu dem in Satz 4.14 für zeitabhängigen Probleme erwähnte Effekt, die pro Zeitschritt entstehenden Fehler bei der Gradientenrekonstruktion addieren. Diese kommen bei der expliziten Variante öfter zum Einsatz als bei der im folgenden Abschnitt 6.3 beschrieben impliziten.

Auch bei niedrigen Reynoldszahlen führt jedoch eine Zeitschrittweitenreduktion durch die in Kapitel 5.5.6 besprochenen Effekte ohne feinere Gitter nicht zu hohen Fehlerreduktionsraten, wie Tabelle 6.2 an Hand des Testbeispiels IV zeigt.

Definition 6.9
In den folgenden Tabellen wird der Fehler zusätzlich in einer „rekonstruierten H_1-Norm“

$$\|f - f_h\|_{G_T/H_1} = \|f - f_h\|_0 + \|\nabla f - G_T f_h\|_0$$

gemessen, da diese Differenz für den Algorithmus von großer Bedeutung ist.

Die in den Tabellen angegebenen Picard-Iterationen pro Schritt (PIPS) verstehen sich dabei als Summe aus den PIPS auf dem feinen und groben Gitter.
Um einen Vergleich zu ermöglichen, sind die Gitter bei den Rechnungen aus Tabelle 6.2 so gewählt worden, dass die Anzahl der Freiheitsgrade auf dem Raum V_h der aus Kapitel 5.5.6 entspricht. Die Anzahl der Freiheitsgrade wird dabei in der Form W_H/W_h angegeben.

Die in der Tabelle 6.2 sichtbaren Sprünge in den Reduktionsraten nach einem Gitterwechsel sind auf die Bemerkung 6.7 Effekte zurückzuführen. Im Fall einer expliziten Behandlung des nichtlinearen Terms kommt es zu mehr hochfrequenten Fehleranteilen als bei einer impliziten, wo ähnliche Effekte bei keinem numerischen Test zu Tage traten.
Im Fall eines konstanten Gitters zeigt sich an Hand es Testbeispieles IV wie der ortsabhängige Fehleranteil, wie in Kapitel 5.5.6 besprochen, für $\triangle t \to 0$ dominanter wird und damit die Reduktionsraten sinken.

Klar wird in diesem Zusammenhang auch, dass - wie erwartet - der in der Abschätzung (6.10) auftretende Summand $C_2 C_V$ wenn nicht Null so doch vernachläßigbar klein ist.

Für $Re \ll 2000$ ist das Verfahren stabil und wie man Tabelle 6.2 entnehmen kann, kann eine Reduktion des Fehlers um den Faktor 8, also dritter Ordnung, erreicht werden.

Für höhere Reynoldszahlen wird im nächsten Abschnitt eine implizite Vorgehensweise vorgestellt, die sich darüber hinaus als weitaus robuster in Bezug auf die durch eine Gitterverfeinerung entstehenden unerwünschten Effekte herausgestellt hat.

Testproblem III mit $Re = 10.0$ bzw. $\nu = 1/10$

Δt	Freiheitsgrade	PIPS	$\|f - f_h\|_{L^2}$	Quotient	$\frac{\|f-f_h\|_{L^2}}{\|f_{max}\|_{L^2}}$	$\|f - f_h\|_{G_T/H_1}$	$\|f - f_h\|_{\max}$
				v_1			
1/4	1089/4225	7.8	1.389e-01	-	2.778e-01	2.159	5.210e-01
1/8	1089/4225	5.8	1.815e-02	7.655	3.630e-02	3.029e-01	7.262e-02
1/16	4225/16641	4.9	6.287e-03	2.886	1.257e-02	1.675e-01	2.486e-02
1/32	4225/16641	4.4	8.757e-04	7.180	1.751e-03	2.347e-02	3.743e-03
1/64	16641/66049	4.0	3.019e-04	2.901	6.037e-04	1.265e-02	1.544e-03
1/128	16641/66049	4.0	5.240e-05	5.761	1.048e-04	1.704e-03	2.334e-04
				v_2			
1/4	1089/4225	7.8	1.360e-01	-	2.720e-01	2.335	6.167e-01
1/8	1089/4225	5.8	2.094e-02	6.495	4.188e-02	3.134e-01	6.530e-02
1/16	4225/16641	4.9	6.378e-03	3.283	1.276e-02	1.684e-01	2.465e-02
1/32	4225/16641	4.4	8.840e-04	7.215	1.768e-03	2.375e-02	3.705e-03
1/64	16641/66049	4.0	3.036e-04	2.911	6.073e-04	1.269e-02	1.540e-03
1/128	16641/66049	4.0	5.219e-05	5.817	1.044e-04	1.701e-03	2.323e-04
				p			
1/4	289/1089	7.8	5.741e-01	-	1.475e-01	2.849	1.788
1/8	289/1089	5.8	7.465e-02	7.691	1.918e-02	6.044e-01	2.392e-01
1/16	1089/4225	4.9	2.771e-02	2.694	7.119e-03	3.450e-01	1.394e-01
1/32	1089/4225	4.4	4.109e-03	6.743	1.056e-03	8.541e-02	2.776e-02
1/64	4225/16641	4.0	1.875e-03	2.192	4.817e-04	4.606e-02	1.757e-02
1/128	4225/16641	4.0	2.716e-04	6.904	6.978e-05	1.755e-02	7.304e-03

Testproblem IV mit Re = 770 bzw $\nu = 1/2000$

				v_1			
1/8	7472/29408	5.7	2.057e-01	-	3.381e-01	7.727e-01	1.308e-01
1/16	7472/29408	4.2	2.960e-02	6.949	4.865e-02	1.016e-01	1.893e-02
1/32	7472/29408	4.0	3.955e-03	7.486	6.499e-03	2.161e-02	2.692e-03
1/64	7472/29408	4.0	7.723e-04	5.121	1.269e-03	1.900e-02	1.090e-03
1/128	7472/29408	4.0	5.163e-04	1.496	8.485e-04	1.880e-02	1.171e-03
1/256	7472/29408	3.9	5.035e-04	1.026	8.274e-04	1.894e-02	1.640e-03
				v_2			
1/8	7472/29408	5.7	2.057e-01	-	3.381e-01	7.200e-01	1.314e-01
1/16	7472/29408	4.2	2.961e-02	6.948	4.866e-02	1.015e-01	1.892e-02
1/32	7472/29408	4.0	3.953e-03	7.490	6.497e-03	2.176e-02	2.712e-03
1/64	7472/29408	4.0	7.360e-04	5.371	1.210e-03	1.911e-02	1.089e-03
1/128	7472/29408	4.0	4.799e-04	1.533	7.887e-04	1.889e-02	1.111e-03
1/256	7472/29408	3.9	4.718e-04	1.017	7.753e-04	1.907e-02	1.624e-03
				p			
1/8	1928/7472	5.7	3.110e-02	-	1.388e-02	1.050e-01	1.548e-02
1/16	1928/7472	4.2	4.923e-03	6.317	2.198e-03	1.614e-02	3.010e-03
1/32	1928/7472	4.0	1.469e-03	3.353	6.555e-04	2.794e-03	1.605e-03
1/64	1928/7472	4.0	1.403e-03	1.047	6.260e-04	2.294e-03	1.454e-03
1/128	1928/7472	4.0	1.408e-03	0.996	6.284e-04	2.517e-03	1.441e-03
1/256	1928/7472	3.9	1.416e-03	0.994	6.322e-04	3.449e-03	1.465e-03

Tabelle 6.2: Explizites Postprocessing mit quadratischen Ansatzfunktionen und BDF(3)

6.3 Implizite Strategie mit linearen Taylor-Hood-Elementen

Basierend auf dem Grundalgorithmus aus Abschnitt 6.1 wird nun ein robustes Postprocessing insbesondere für moderate und hohe Reynoldszahlen vorgestellt.
Hierzu wird der nichtlineare Term implizit behandelt. Im Postprocessingschritt werden lineare Ansatzfunktionen verwendet.

0. Berechne eine Startnäherung $p_h^0 = \bar{p}_h^1$ für Druck durch (5.10)

Ein Zeitschritt im Splittingverfahren mit Postprocessing:

1. Löse die implizite Gleichung für die Zwischengeschwindigkeit $\tilde{v}_{h/2}^{n+1}$ mit linearen finiten Elementen
2. Löse die Gleichung zur Druckaktualisierung
   $-\nabla^2 p_{h_{update}} = -\frac{\beta_0}{\triangle t} \nabla \cdot \tilde{v}_{h/2}^{n+1}$
3. Sorge für die Divergenzfreiheit in v mittels
   $\hat{v}_{h/2}^{n+1} = \tilde{v}_{h/2}^{n+1} + \frac{\triangle t}{\beta_0} \nabla p_{h_{update}}$
4. Berechne eine Näherung $\bar{p}_{h/2}^{n+1}$ auf der Basis von $\hat{v}_{h/2}^{n+1}$ für den Druck durch (5.10)
5. Löse implizit die Gleichung für die Zwischengeschwindigkeit $\tilde{v}_{h/4}^{n+1}$ mit dem Startwert $v_e = P\hat{v}_{h/2}^{n+1}$ und $\bar{p}_{h/2}^{n+1}$ aus Schritt 4
6. Löse die Gleichung zur Druckaktualisierung
   $-\nabla^2 p_{h/2_{update}} = -\frac{\beta_0}{\triangle t} \nabla \cdot \tilde{v}_{h/4}^{n+1}$ und berechne $p_{h/2}^{n+1} = \bar{p}_{h/2}^{n+1} + p_{h/2_{update}}$
7. Sorge für die Divergenzfreiheit in v mittels $v_{h/4}^{n+1} = \tilde{v}_{h/4}^{n+1} + \frac{\triangle t}{\beta_0} \nabla p_{h/2_{update}}$
8. Berechne die Restriktionen $v_{h/2}^{n+1} = I v_{h/4}^{n+1}$ und $p_h^{n+1} = I p_{h/2}^{n+1}$

Die Tabellen 6.4 und 6.5 zeigen hierbei die Wirkung dieser Strategie im Sinne einer Vorkonditionierung. Das Verfahren wird beschleunigt, während sich gleichzeitig die Genauigkeit verbessert.

Die in den Tabellen 6.4 und 6.5 angegebenen Picard-Iterationen pro Schritt (PIPS) verstehen sich dabei erneut als Summe aus den PIPS auf dem feinen und groben Gitter.

Δt	Freiheits-grade	PIPS	Fehler bei Re = 2000					
			$\|f - f_h\|_0$	Quotient	$\|f - f_h\|_{G_T/H_1}$	Quotient	$\|f - f_h\|_{max}$	Quotient
			v_1					
1/8	16641	12.7	6.284e-02	-	1.824e-00	-	2.070e-01	-
1/8	66049	14.3	9.192e-02		5.481e-00		3.675e-01	
1/16	66049	6.6	5.516e-03	10.139	3.325e-01	5.485	2.046e-02	10.117
1/32	66049	4.9	6.066e-04	9.093	3.122e-02	10.650	4.011e-03	5.101
1/64	66049	4.2	1.347e-04		1.364e-02		3.271e-03	
1/64	263169	4.3	1.107e-04	5.480	1.163e-02	2.684	1.477e-03	2.715
1/128	66049	4.0	8.978e-05		1.105e-02		2.854e-03	
1/128	263169	4.0	2.413e-05	4.588	4.158e-03	2.797	1.314e-03	1.124
			v_2					
1/8	16641	12.7	7.320e-02	-	2.535e-00	-	1.814e-01	-
1/8	66049	14.3	9.831e-02		6.042e-00		4.017e-01	
1/16	66049	6.6	6.930e-03	10.056	3.196e-01	7.931	2.002e-02	9.061
1/32	66049	4.9	6.554e-04	10.057	3.193e-02	10.009	4.214e-03	4.751
1/64	66049	4.2	1.313e-04		1.354e-02		3.286e-03	
1/64	263169	4.3	1.215e-04	5.392	1.161e-02	2.750	1.477e-03	2.853
1/128	66049	4.0	8.997e-05		1.100e-02		2.845e-03	
1/128	263169	4.0	2.415e-05	5.033	4.152e-03	2.796	1.314e-03	1.124
			p					
1/8	4225	12.7	2.206e-01	-	9.569e-01	-	6.768e-01	-
1/8	16641	14.3	3.754e-01		1.599e-00		1.134e-00	
1/16	16641	6.6	2.362e-02	9.341	9.342e-02	10.242	6.239e-02	10.847
1/32	16641	4.9	6.951e-03	3.398	2.583e-02	3.616	1.753e-02	3.559
1/64	16641	4.2	1.581e-03		1.257e-02		7.324e-03	
1/64	66049	4.3	1.835e-03	3.787	7.412e-03	3.484	6.906e-03	2.538
1/128	16641	4.0	3.266e-04		1.056e-02		6.639e-03	
1/128	66049	4.0	4.774e-04	3.845	6.527e-03	1.135	3.431e-03	2.0128

Tabelle 6.3: Testproblem III; Verhalten der implitziten Postprocessing-Strategie bei unterschiedlichen $\triangle t$-h-Kombinationen

Die in der Tabelle 6.3 Quotienten beziehen sich jedweil aus die beste Kombination aus Zeitschrittweite $\triangle t$ und Gitterweite h.

Die numerisch bestimmten Reduktionsraten liegen oberhalb der zweiten Ordnung, können jedoch für $\triangle t \rightarrow 0$ erneut nur bei gleichzeitiger Gitterverfeinerung erreicht werden. Dies gilt, siehe Kapitel 5.4, insbesondere für große Reynolds- bzw. Péclet-Zahlen. Die größten Verbesserungen ergeben sich bei der Verwendung sowohl der expliziten als auch der impliziten Postprocessing-Strategie für den Druck. Hier wird oft eine um mehr als eine Zehnerpotenz exaktere Näherung in der L^2-Norm erreicht.

Wie man an der Tabelle 6.3 sieht ist die impliziten Behandlung des nichtlinearen Terms weniger Anfällig für die in Abschnitt 6.1 diskutieren Gitterabhängigkeiten. Für die verhältnismäßig große Zeitschrittweite $\triangle t = 1/8$ treten durch die Gitterverfeinerung noch Verschlechterungen des Resultates auf, für kleiner Zeitschrittweiten nicht mehr.

An Hand der Maxiumsnorm erkennt man bei kleiner werden Fehlern die wachsende Bedeutung der konstruierten Randbedingungen. In der Größenordnung von ν beginnt die Laplace-Rekonstruktion an Bedeutung zu gewinnen und verlangsamt die Fehlerreduktions in der Maximusnorm. Die Fehler in der Maximusnorm treten dabei wie schon in Kapitel 5.1.4 klar wurde, am Rand auf.

Tabelle 6.4: Splitting mit und ohne lineares Postprocessing im Vergleich auf Testproblem IV mit $\nu = 1/5000$ bzw $Re \approx 1925$

	Freiheitsgrade	Freiheitsgrade	CPU	lineares Postprocessing			CPU	ohne Postprocessing			Beschleu-
Δt	1. Schritt	2. Schritt	Zeit	PIPS	$\Vert f - f_h \Vert_0$	Quotient	Zeit	PIPS	$\Vert f - f_h \Vert_0$	Quotient	nigung
v_1											
1/8	7472	29408	0 h 29 m	6.2	1.216e-01	-	0 h 39 m	4.3	1.222e-01	-	1.34
1/16	7472	29408	0 h 51 m	4.5	1.768e-02	6.880	0 h 57 m	2.4	4.035e-02	3.029	1.12
1/32	29408	116672	7 h 11 m	3.9	2.254e-03	7.843	7 h 42 m	2.0	6.779e-03	5.952	1.07
1/64	29408	116672	11 h 24 m	3.0	3.026e-04	7.448	15 h 21 m	2.0	2.247e-03	3.018	1.35
1/128	29408	116672	22 h 44 m	3.0	1.629e-04	1.857	30 h 18 m	2.0	7.920e-04	2.836	1.33
v_2											
1/8	7472	29408	0 h 29 m	6.2	1.216e-01	-	0 h 39 m	4.3	2.015e-01	-	1.34
1/16	7472	29408	0 h 51 m	4.5	1.769e-02	6.875	0 h 57 m	2.4	4.961e-02	4.061	1.12
1/32	29408	116672	7 h 11 m	3.9	2.254e-03	7.849	7 h 42 m	2.0	8.557e-03	5.798	1.07
1/64	29408	116672	11 h 24 m	3.0	2.970e-04	7.591	15 h 21 m	2.0	2.443e-03	3.503	1.35
1/128	29408	116672	22 h 44 m	3.0	1.587e-04	1.871	30 h 18 m	2.0	7.691e-04	3.176	1.33
p											
1/8	1928	7472	0 h 29 m	6.2	1.907e-02	-	0 h 39 m	4.3	5.087e-01	-	1.34
1/16	1928	7472	0 h 51 m	4.5	2.827e-03	6.746	0 h 57 m	2.4	1.145e-01	4.443	1.12
1/32	7472	29408	7 h 11 m	3.9	4.260e-04	6.636	7 h 42 m	2.0	2.735e-02	4.187	1.07
1/64	7472	29408	11 h 24 m	3.0	3.348e-04	1.273	15 h 21 m	2.0	8.960e-03	3.052	1.35
1/128	7472	29408	22 h 44 m	3.0	3.299e-04	1.015	30 h 18 m	2.0	2.843e-03	3.152	1.33

Tabelle 6.5: Splitting mit und ohne lineares Postprocessing im Vergleich auf Testproblem III mit $\nu = 1/5000$ bzw $Re = 5000$

	Freiheitsgrade	Freiheitsgrade	CPU	lineares Postprocessing			CPU	ohne Postprocessing			Beschleu-
Δt	1. Schritt	2. Schritt	Zeit	PIPS	$\|f - f_h\|_0$	Quotient	Zeit	PIPS	$\|f - f_h\|_0$	Quotient	nigung
v_1											
1/8	1089	4225	0 h 6 m	11.7	3.680e-02	-	0 h 7 m	6.7	3.088e-01	-	1.16
1/16	4225	16641	0 h 40 m	6.6	4.321e-03	8.515	0 h 54 m	5.0	6.422e-02	4.808	1.40
1/32	4225	16641	1 h 6 m	5.0	1.026e-03	4.213	1 h 17 m	2.9	2.293e-02	2.801	1.16
1/64	16641	66049	9 h 19 m	4.1	2.322e-04	4.418	9 h 45 m	2.2	4.520e-03	5.072	1.04
1/128	16641	66049	18 h 24 m	4.0	1.504e-04	1.544	18 h 27 m	2.0	1.503e-03	3.008	1.00
v_2											
1/8	1089	4225	0 h 6 m	11.7	5.868e-02	-	0 h 7 m	6.7	3.224e-01	-	1.16
1/16	4225	16641	0 h 40 m	6.6	5.861e-03	10.01	0 h 54 m	5.0	6.962e-02	4.630	1.40
1/32	4225	16641	1 h 6 m	5.0	1.114e-03	5.260	1 h 17 m	2.9	2.290e-02	3.040	1.16
1/64	16641	66049	9 h 19 m	4.1	2.627e-04	4.242	9 h 45 m	2.2	4.535e-03	5.051	1.04
1/128	16641	66049	18 h 24 m	4.0	1.498e-04	1.753	18 h 27 m	2.0	1.517e-03	2.989	1.00
p											
1/8	289	1089	0 h 6 m	11.7	1.201e-01	-	0 h 7 m	6.7	2.669	-	1.16
1/16	1089	4225	0 h 40 m	6.6	2.869e-02	4.187	0 h 54 m	5.0	2.203e-01	12.11	1.40
1/32	1089	4225	1 h 6 m	5.0	5.687e-03	5.045	1 h 17 m	2.9	8.028e-02	2.744	1.16
1/64	4225	16641	9 h 19 m	4.1	1.846e-03	3.081	9 h 45 m	2.2	2.994e-02	2.681	1.04
1/128	4225	16641	18 h 24 m	4.0	3.859e-04	4.783	18 h 27 m	2.0	1.398e-02	2.142	1.00

7 Adaptivität in Zeit und Ort

Einen Algorithmus nennt man adaptiv, wenn er sich den Gegebenheiten des Problems selbstständig anpassen kann.
Für den Splittingalgorithmus werden dazu zwei verschiedene Strategien benötigt, zum einen eine adaptive Wahl der Zeitschrittweite und zum anderen eine adaptive Anpassung des Gitters. Betrachten wir zunächst die Zeitschrittweitensteuerung.

7.1 Adaptivität in der Zeit

Die Adaptivität in der Zeit ist von großer Bedeutung für das Splitting, da hierdurch nicht nur die Qualität der Lösung verbessert, sondern auch die Konvergenz sichergestellt werden kann. Durch eine geeignete Zeitschrittweitensteuerung wird das Splitting somit zu einem robusten Algorithmus. Um Adaptivität in der Zeit realisieren zu können, benötigen wir ein BDF-Verfahren auf variablen Diskretisierungen $0 = t_0 < t_1 < t_2 \cdots < t_{n-1} < t_n = t_{ende}$ des Intervalls $[0, t_{ende}]$. Dabei bezeichnen wir mit $\triangle t_{m+1}$ die Differenz $t_{m+1} - t_m$.

Das implizite Euler-Verfahren benötigt hierfür keine Veränderungen, da es als Einschrittverfahren sehr flexibel mit variablen Gittern umgehen kann. Zur Entwicklung von Verfahren höherer Ordnung konstruiert man, wie schon bei den BDF-Verfahren mit konstanter Schrittweite, ein Interpolationspolynom $q(t)$ durch (t_{m+1}, y_{m+1}) und die zurückliegenden l Werte $(t_{m-l+1}, y_{m-l+1}) \ldots (t_m, y_m)$. Nun fordert man, dass $q(t_{m+1})$ die Differentialgleichung

$$q'(t_{m+1}) = f(t_{m+1}, t_{m+1})$$

erfüllt. Mit Hilfe der Lagrange-Polynome

$$L_{m,j} = \prod_{i=0;i\neq j}^{l} \frac{t - t_{m-i+1}}{t_{m-j+1} - t_{m-i+1}}$$

erhält man folgende Darstellung

$$q(t) = \sum_{j=1}^{l+1} y_{m-l+j} L_{m,j} \Rightarrow q'(t) = \sum_{j=1}^{l+1} y_{m-l+j} L'_{m,j} \; .$$

Um die Koeffizienten β_i zu berechnen, muss also

$$\frac{\partial}{\partial t} L_{m,j}(t) \bigg|_{t=t_{m+1}}$$

ausgewertet werden. Für das BDF(2) ergibt sich laut [40], S.337f.

$$\frac{1}{\triangle t_{m+1}}\left(\underbrace{\frac{1+2w_{m+1}}{1+w_{m+1}}}_{=\beta_0} v_{m+1}-\underbrace{(1+w_{m+1})}_{=\beta_1} v_m+\underbrace{\frac{w_{m+1}^2}{1+w_{m+1}}}_{=\beta_2} v_{m-1}\right)=f_{m+1}\ ,$$

mit $w_{m+1}=\triangle t_{m+1}/\triangle t_m$.
Für das Verfahren dritter Ordnung führt das analoge Vorgehen auf die folgenden Koeffizienten.

$$\begin{aligned}
\beta_3 &= \left(\triangle t_{m+1}\cdot\frac{\partial}{\partial t}\frac{(t-t_{m-1})(t-t_m)(t-t_{m+1})}{(t_{m-2}-t_{m-1})(t_{m-2}-t_m)(t_{m-2}-t_{m+1})}\right)|_{t=t_{m+1}}\\
&= \frac{(t_{m+1}-t_m)(t_{m+1}-t_{m-1})(t_{m+1}-t_m)}{(t_{m-2}-t_{m-1})(t_{m-2}-t_m)(t_{m-2}-t_{m+1})}\\
\beta_2 &= \left(\triangle t_{m+1}\cdot\frac{\partial}{\partial t}\frac{(t-t_{m-1})(t-t_m)(t-t_{m+1})}{(t_{m-2}-t_{m-1})(t_{m-2}-t_m)(t_{m-2}-t_{m+1})}\right)|_{t=t_{m+1}}\\
&= \frac{(t_{m+1}-t_m)(t_{m+1}-t_{m-2})(t_{m+1}-t_m)}{(t_{m-1}-t_{m-2})(t_{m-1}-t_m)(t_{m-1}-t_{m+1})}\\
\beta_1 &= \left(\triangle t_{m+1}\cdot\frac{\partial}{\partial t}\frac{-(t_{m+1}-t_{m-2})(t_{m+1}-t_{m-1})}{(t_m-t_{m-1})(t_m-t_{m-2})}\right)|_{t=t_{m+1}}\\
&= \frac{(t_{m+1}-t_m)(t_{m+1}-t_{m-1})(t_{m+1}-t_m)}{(t_{m-2}-t_{m-1})(t_{m-2}-t_m)(t_{m-2}-t_{m+1})}\\
\beta_0 &= \left(\triangle t_{m+1}\cdot\frac{\partial}{\partial t}\frac{(t-t_{m-1})(t-t_m)(t-t_{m+1})}{(t_{m-2}-t_{m-1})(t_{m-2}-t_m)(t_{m-2}-t_{m+1})}\right)|_{t=t_{m+1}}\\
&= 1+\frac{t_{m+1}-t_m}{t_{m+1}-t_{m-1}}+\frac{t_{m+1}-t_m}{t_{m+1}-t_{m-2}}
\end{aligned}$$

Die ersten l Schritte können zum Beispiel mit einem Verfahren niedriger Ordnung erzeugt werden. Hierbei wird der erste Schritt mit einem impliziten Euler-Verfahren und der zweite mit dem BDF-Verfahren zweiter Ordnung berechnet.

7.1.1 Stabilitätsbeschränkungen und Schrittweitensteuerung

Die BDF-Verfahren zweiter und dritter Ordnung sind nicht für jede Diskretisierung von $[0, t_{ende}]$ stabil. Wie in dem Artikel [18] von Grigorieff nachzulesen ist, konnte er in Bezug auf die Stabilität von BDF-Verfahren beweisen, dass das BDF-Verfahren zweiter Ordnung für

$$0\le\frac{\triangle t_{m+1}}{\triangle t_m}\le 2.414$$

und das BDF-Verfahren dritter Ordnung für

$$0.836\le\frac{\triangle t_{m+1}}{\triangle t_m}\le 1.127$$

stabil ist. Diese Schranken aus [18] werden im Allgemeinen als zu pessimistisch angesehen (vgl. z.B. [40], S.338), da sie alle möglichen Schrittweitenänderungen berücksichtigen. Daher werden vom Splittingalgorithmus für das BDF-Verfahren dritter Ordnung die Schranken

$$0.75\le\frac{\triangle t_{m+1}}{\triangle t_m}\le 1.25$$

verwendet.
Zur Schrittweitensteuerung wird ein heuristisches Kriterium verwendet. Dies geschieht mit Hilfe der Differenz zwischen der mit einem Zeitschrittverfahren mit der Schrittweite $\triangle t_m/2$ berechneten Näherung $v_{\triangle t_m/2}(t_m)$ und der mit der Schrittweite $\triangle t$ $v_{\triangle t_m}(t_m)$. Wir gehen also davon aus, dass der durch das Zeitschrittverfahren erzeugte relative Fehler

$$\hat{e}_t(t_m) \approx \frac{\|v_{\triangle t_m/2}(t_m) - v_{\triangle t_m}(t_m)\|}{(1-2^{-l})\|v_{\triangle t_m/2}(t_m)\|} \tag{7.1}$$

ist, wobei mit l die Ordnung des Verfahrens bezeichnet wird. Wird der Fehler durch die Halbierung der Zeitschrittweite um den Faktor l reduziert, hält dieser Ansatz. Da das BDF(3)-Verfahren in der Regel nur eine Reduktion um den Faktor 4 bewirkt, sollte hier $l = 2$ statt $l = 3$ verwendet werden.
Nun wird dem Algorithmus eine Fehlertoleranz ε_{Ttol} vorgegeben. Falls $e_t(t_m) \approx \varepsilon_{Ttol}$ ist, kann der Algorithmus mit der bisherigen Schrittweite verfahren. Anderenfalls wird die neue Schrittweite $\triangle t_{m+1}$ mittels

$$\triangle t_{m+1} = \left(\frac{\varepsilon_{Ttol}}{e_t(t_m)}\right)^{1/l-1} \triangle t_m$$

berechnet. Liegt $\triangle t_{m+1}/\triangle t_m$ außerhalb der oben angegebenen Schranken, so wird die jeweilige Schranke verwendet. Daneben ist es sinnvoll, weitere absolute Sicherheitsschranken für $\triangle t_{m+1}$ zu setzen, z.B. $0.001 \leq \triangle t_{m+1} \leq 0.25$. Die obere Schranke dient der Absicherung der Stabilität. Die untere hat zwei Funktionen: Zum einen treten innerhalb des Algorithmus Fehler pro Schritt auf, die so begrenzt bleiben, und zum anderen dient sie als obere Grenze für die CPU-Kosten, die man bereit ist für die geforderte Genauigkeit zu investieren.

Die in (7.1) verwendete Norm muss abhängig vom Problem gewählt werden. Bei der Verwendung einer globalen Norm wie der L_2-Norm ist nicht sichergestellt, dass diese auf Veränderungen, die sich z.B. an einer Einspritzstelle ereignen, stark genug reagiert. Das andere Extrem ist die Maximumsnorm, welche jede lokale Veränderung misst, dabei jedoch u.U. zu einer zu pessimistischen Abschätzung kommt und auch ein sehr sprunghaftes Verhalten bei der Wahl von $\triangle t_m$ verursachen kann. Eine Alternative stellt zum Beispiel die Verwendung einer Patchnorm dar, welche den Fehler über einzelnen Teilen des Gebietes misst. Wir werden für die folgenden numerischen Rechnungen die L_1-Norm verwenden.

Bei der Verwendung von BDF-Verfahren auf einer variablen Diskretisierung des Zeitintervalls ergeben sich bei den Verfahren höherer Ordnung Konsequenzen für den Druckaktualisierungsschritt. Dieser ist für näherungsweise konstante Schrittweiten, wie sie durch die Schranken des BDF(3) sichergestellt sind, optimal. Bei stark schwankenden Zeitschrittweiten, wie sie z.B. beim BDF(2) auftreten können, kommt es trotz einer Reduktion der Zeitschrittweite in dem jeweiligen Zeitschritt nicht immer zu der erwarteten Verbesserung im Druck. Der Grund ist im für den Druckaktualisierungsschritt entscheidenden Faktor β_0 zu suchen. Betrachtet man diesen für kleine w, so sieht man, dass auch β_0 klein wird. Das bedeutet, dass im Fall einer Schrittweitenreduktion der Druckaktualisierungsschritt an Bedeutung verliert, was zwar aus dem Splittingschema, nicht jedoch aus der Zeitschrittweite $\triangle t_{m+1}$ heraus gerechtfertigt ist. Für das BDF(2) gilt $1 < \beta_0 \leq 3$, wobei 3 im Fall einer äquidistanten Schrittweite angenommen wird.

7.1.2 Numerische Ergebnisse

Zur Untersuchung der zeitlichen Adaptivität verwenden wir eine Abwandlung des Testproblems III mit

$$\begin{aligned} v_1(x,y,t) &= z_{1|2}(t)(\sin(\pi x/2)\cos(\pi y/2)) \;, \\ v_2(x,y,t) &= -z_{1|2}(t)(\sin(\pi y/2)\cos(\pi x/2)) \end{aligned}$$

und

$$p(x,y,t) = \sin(t)(\cos(\pi x) + \cos(\pi y)) \;.$$

Für die erste Variante gilt

$$z_1(t) = \frac{\arctan(t-3)}{4\arctan(3)} + 0.75 \;,$$

für die zweite

$$z_2(t) = 0.5\exp(-5|t-3|) + 0.5$$

mit $t \in [0,6]$. Da $z_2 \notin C^1[0,6]$, stellt dies an das Zeitschrittverfahren in der Umgebung von $t = 3$ eine besondere Herausforderung dar, weshalb wir hier auf ein BDF(2) zurückgreifen, welches auf Grund der Schrittweitensteuerung schneller in der Lage ist, auf solche Veränderungen zu reagieren. Dem Algorithmus wird als Fehlergrenze $5 \cdot 10^{-4}$ übergeben und es wird eine Zerlegung mit 16641 Freiheitsgraden für die Geschwindigkeit und 4225 für den Druck verwendet. Der nichtlineare Term wird implizit behandelt. Alle Berechnungen im Abschnitt 7.1.2 werden ohne Postprocessing berechnet.

Wie man an den Graphen 7.2 und 7.3 erkennen kann, wird durch den Einfluss des Drucks auf die Genauigkeit von v auch die zeitliche Entwicklung des Drucks bei der Wahl der Zeitschrittweite berücksichtigt. Durch die adaptive Steuerung in der Zeit ist der Algorithmus in der Lage, die schwierige Stelle $t = 3$ zu erkennen und das Splitting dort mit kleinen Zeitschrittweiten stabil und den Fehler in akzeptabler Weise klein zu halten.

Dass die Schranke von $5 \cdot 10^{-4}$ nicht eingehalten wird, liegt darin begründet, dass durch die adaptive Anpassung der Zeitschrittweite nur eine Fehlerquelle kontrolliert werden kann. Jedoch handelt es sich bei dem Splittingverfahren um die wichtigste, da das Verfahren nun mit dieser adaptiven Steuerung und einer impliziten Behandlung des nichtlinearen Terms wesentlich robuster ist.

Abschließend wird das zeitadaptive Splitting auf ein Problem aus der Arbeit [1] angewendet. Die exakte Lösung lautet in diesem Fall:

$$\begin{aligned} v_1(x,y,t) &= 1 - 2\cos(\pi x - t)\sin(\pi y - t) \;, \\ v_2(x,y,t) &= 1 + 2\cos(\pi y - t)\sin(\pi x - t) \end{aligned}$$

und

$$p(x,y,t) = -\cos(4\pi(x-t)) - \cos(4\pi(y-t)) \;,$$

wobei $t \in [0,\pi]$ und $\nu = \frac{1}{250}$ ist, was einer Reynoldszahl von ca. 800 entspricht. Bei Ω handelt es sich erneut um das Einheitsquadrat mit dem aus Kapitel 5 bekannten kanonischen Gitter mit 16641 Freiheitsgraden in v und 4225 in p.

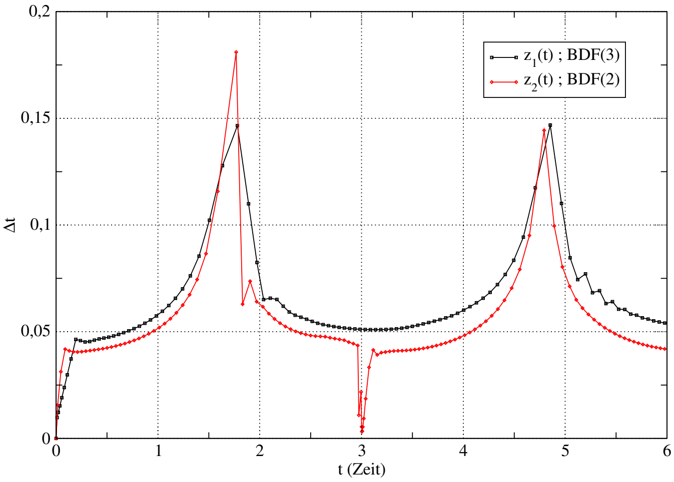


Abbildung 7.1: Gewählte Schrittweite

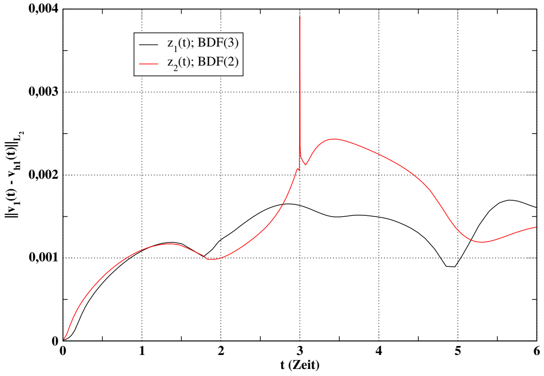


Abbildung 7.2: Fehler in der ersten Geschwindigkeitskomponente (L_2-Norm)

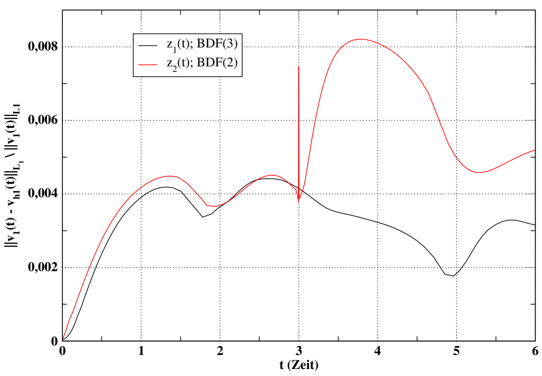


Abbildung 7.3: Relativer Fehler in der ersten Geschwindigkeitskomponente (L_1-Norm)

In diesem Fall verändert sich die Struktur der Lösung weit stärker mit der Zeit als in den vorangegangenen Beispielen, da die Zeitabhängigkeit nicht einfach mittels einer Funktion $z(t)$ abgespalten werden kann.

Der Splittingalgorithmus wird mit den gleichen Einstellungen wie beim modifizierten Testproblem III durchgeführt.

Wie man an den Abbildungen 7.4 bis 7.6 erkennen kann, wird die Schranke von $5 \cdot 10^{-4}$ näherungsweise gehalten, wobei die Überschreitungen erneut auf die anderen Fehlerquellen zurückzuführen sind.
Der Fehler im Druck ist ebenfalls bis auf das charakteristische Einschwingverhalten konstant. Der erhöhte Fehler beim Start des Splittings ist hier besonders ausgeprägt, da der Startwert v^0 nicht aus dem Splittingprozess hervorgegangen ist und keine einfache Struktur besitzt, wie in den Fällen, in denen es sich um die Nullfunktion handelt. Dies und die sehr kleine

Zeitschrittweite am Anfang verstärken diesen ansonsten geringeren Effekt. Auch unter diesen schwierigen Umständen bleibt der Startfehler vertretbar klein und im Anschluss bis auf einen kurzen Ausreißer unter $1 \cdot 10^{-2}$.

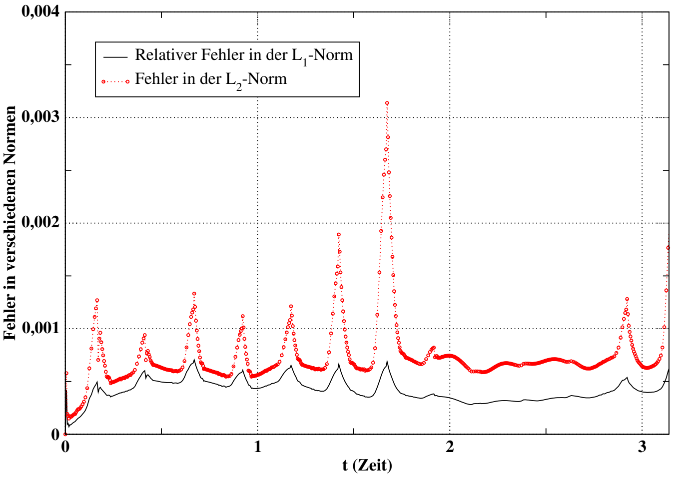


Abbildung 7.4: Fehler in der ersten Geschwindigkeitskomponente

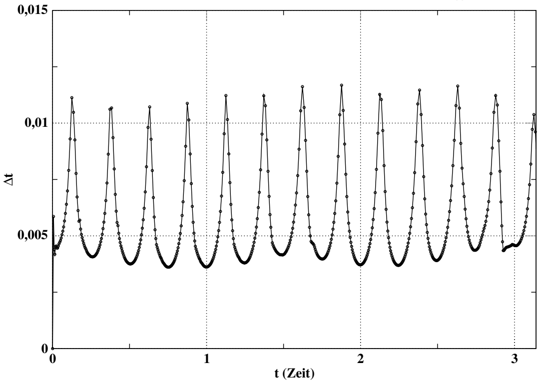


Abbildung 7.5: Gewählte Schrittweite

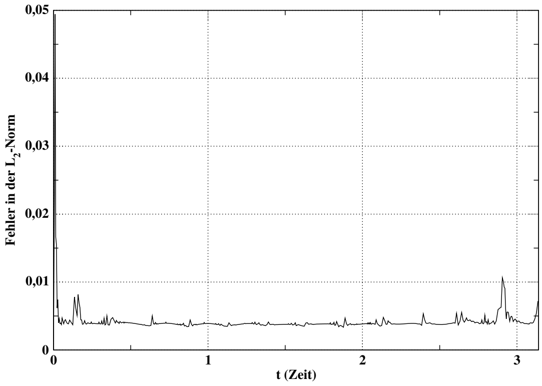


Abbildung 7.6: Fehler im Druck

7.2 Adaptivität im Ort

Die erste Triangulierung, die von einem Gebiet Ω gemacht wird, ist gelegentlich verhältnismäßig grob und orientiert sich hauptsächlich an der Geometrie von Ω. Diese Geometrie wird in der ersten Triangulierung angenähert und das Gitter, je nachdem wie genau die Geometrie erfasst werden soll, entsprechend fein oder grob gewählt.
Auf einer solchen Triangulierung berechnet man nun die Galerkin-Approximation v_h und entscheidet auf Grund einer a-posteriori-Fehlerabschätzung adaptiv, ob ein oder mehrere Dreiecke verfeinert werden müssen.

7.2.1 Der verwendete Verfeinerungsalgorithmus

Es gibt verschiedene Strategien zur Verfeinerung der Dreiecke. Eine solche Verfeinerung kann entweder lokal, also nur auf den Elementen mit großem Anteil am Fehler, oder global, also für die ganze Triangulierung, vorgenommen werden.

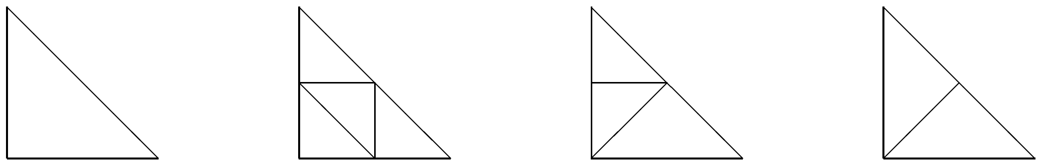


Abbildung 7.7: Finites Element: unverfeinert - rot verfeinert - blau verfeinert - grün verfeinert

Die häufigste Verfeinerungsstrategie ist die reguläre Verfeinerung eines Dreiecks in vier Teildreiecke, auch rote Verfeinerung genannt. Dies geschieht durch das Verbinden der Kantenmittelpunkte. Wendet man diese Methode nur auf einer ausgewählten Menge von Dreiecken und nicht auf der ganzen Triangulierung an, so treten Punkte auf, welche Eckpunkte der verfeinerten, nicht jedoch Eckpunkte der nicht verfeinerten angrenzenden Dreiecke sind. Solche Knoten werden als „hängende Knoten“ bezeichnet. Um eine zulässige Triangulierung zu erhalten, werden die entsprechenden Dreiecke ebenfalls verfeinert. Dabei kommen die oben abgebildeten grünen und blauen Verfeinerungsstrategien zum Tragen.
Problematisch ist bei wiederholtem Verfeinern der grüne Ansatz. Wenn hier mehrfach nicht die längste Seite des Dreiecks geteilt wird, so entstehen extrem spitze Dreiecke, welche zu Instabilitäten führen. Daher wird oft, zum Beispiel in [43], empfohlen, die grünen Kanten vor einer erneuten Verfeinerung zu entfernen. Wir gehen ähnlich wie die PDE-Toolbox von MatLab einen anderen Weg (die Referenz zum Befehl refinemesh findet man in [30], S.5-91) und erlauben die grüne Verfeinerung nur für den Fall, dass die längste Kante halbiert wird. Ist dies nicht der Fall, wird mit der längsten Kante blau verfeinert. Ähnlich gehen wir vor, wenn die blaue Verfeinerung ohne die längste Kante durchgeführt werden würde, dann wird stattdessen rot verfeinert. Dieses Vorgehen führt zu einer geringfügig höheren Anzahl an verfeinerten Dreiecken. Der Algorithmus lässt sich nun wie folgt formulieren:

Gitterverfeinerungsalgorithmus:

1. Verfeinere alle ausgewählten Dreiecke regulär.
2. Füge einen Knoten auf der längsten Kante jedes Dreiecks hinzu, das bereits einen hängenden Knoten hat.
3. Wiederhole Schritt 2, bis keine neuen Knoten mehr hinzugefügt werden.
4. Erzeuge die neuen Dreiecke. Wenn auf jeder Kante ein neuer Knoten entstanden ist durch die rote, wenn zwei neue entstanden sind durch die blaue und wenn nur ein neuer entstanden ist durch die grüne Verfeinerung.

7.2.2 Fehlerindikator auf der Basis der Gradientenrekonstruktion

Die Auswahl der Dreiecke, welche verfeinert werden sollen, geschieht mit Hilfe eines Fehlerschätzers $\hat{\eta}$. Es gibt verschiedene Arten von Fehlerschätzern, z.B. residuenbasierende oder hierarchische. Da für das Splitting in jedem Fall eine Gradientenrekonstruktion implementiert wurde, werden wir diese auch als Fehlerschätzer verwenden.
Die Grundidee eines Fehlerschätzers mittels Gradientenrekonstruktion besteht darin, die Differenz des Gradienten der diskreten Lösung über einem Element mit dem durch ein Verfahren wie z.B. das Z^2-Verfahren rekonstruierten Gradienten zu vergleichen. Man betrachte also

$$\hat{\eta} = \sum_{T \in \mathcal{T}_h} \int_T \|Gu_h - \nabla u_h d\|_{L^2} \Omega \tag{7.2}$$

mit $G = \{G_M, G_Z, G_T\}$ für den gesamten Fehler und das Integral über dem einzelnen Element für den lokalen Fehler.
Ist für G die Superkonvergenzeigenschaft bewiesen, so ist $\hat{\eta}$ als Fehlerschätzer geeignet. Die Superkonvergenzeigenschaft lässt sich laut [27], S.185 nur unter sehr einschränkenden Voraussetzungen an das Gitter und die Glätte der Lösung zeigen. Somit ist ein strenger mathematischer Beweis für den Fehlerschätzer $\hat{\eta}$ in der Regel nicht möglich. $\hat{\eta}$ ist jedoch ein guter Fehlerindikator, mit dessen Hilfe kritische Elemente ermittelt werden können. Dabei hat sich in den numerischen Tests herausgestellt, dass man für die auftretenden Probleme bessere Gitter erhält, wenn der Fehler nicht wie in (7.2) mit der H^1-Seminorm auf jedem Dreieck gemessen wird, sondern knotenweise. Dazu definieren wir

$$\hat{\eta}_a = \frac{1}{\#\triangle(a)} \sum_{K \subset \triangle(a)} |\ Gu_h(x_a, y_a) - (\nabla u_h)|_K\ | \quad ,$$

dabei sei wie in Kapitel 5 wieder für den Knoten a die Menge $\triangle(a) := \bigcup_{K\,:\,o \in \partial K} K$ die Dreiecksumgebung von a und $\#\triangle(a)$ die Mächtigkeit der Menge. Ist $\hat{\eta}_a$ größer als ein gegebener Toleranzwert, wird der Knoten in die Menge der Verfeinerungskandidaten $\hat{K}$ aufgenommen. Der maximale Fehler in dieser Menge dient als Richtwert für die Menge der Elemente, die schließlich verfeinert werden. Gilt für einen Knoten $\hat{\eta}_a > \frac{\hat{\eta}_{\max}}{\gamma}$, wobei $\hat{\eta}_{\max} = \max_{i \in \mathcal{I}_h}\{\hat{\eta}_i\}$ ist, so werden alle Dreiecke, die den Knoten a beinhalten, verfeinert. In Abhängigkeit von γ werden unterschiedlich viele Elemente zur Verfeinerung ausgewählt, wobei im Folgenden alle Rechnungen mit $\gamma = 2$ durchgeführt wurden.

Dieser Fehlerindikator wird im Laufe des Splittings auf Konvektions-Diffusions-Reaktions-Gleichungen angewendet. Um die Effizienz des Fehlerschätzers in Zusammenhang mit solchen Gleichungen zu demonstrieren, benutzen wir zwei zeitunabhängige Probleme vom Typ der Gleichung (4.1), das erste mit der Lösung

$$v(x, y) = \exp\left(-\frac{x^6 + y^6}{\nu}\right)$$

(„Eckenlayer“) und den Variablen

$$\nu = 10^{-4}\ , \quad k = (\ \frac{1}{4} - x\ ,\ \frac{1}{4} - y\ )\ , \quad r = 1\ ,$$

wobei die Bezeichnungen der Variablen denen aus der Gleichung (4.1) entsprechen. Die exakte Lösung des zweiten Problems („Hutlayer“) lautet

$$v_\nu(x,y) = \tanh\left(\frac{\frac{1}{16} - (x-\frac{1}{2})^2 - (y-\frac{1}{2})^2}{\nu}\right)$$

und die Variablen sind

$$\nu = 10^{-3}\ , \quad k = (\ (4x-2)\ v_\nu\ ,\ (4y-2)\ v_\nu\ )\ . \quad r = 4\ .$$

Die Ergebnisse aus Tabelle 7.1 und 7.2 sowie die Abbildungen 7.9 und 7.11 zeigen die Effizienz des Verfahrens.

Freiheitsgrade	$\|f - f_h\|_{\max}$	$\|f - f_h\|_{L^2}$
289	6.161607e-02	8.791782e-03
350	1.699135e-02	2.719085e-03
519	8.615437e-03	8.428856e-04
1137	3.694315e-03	4.001344e-04
3407	2.080191e-03	1.430412e-04
11926	1.641810e-03	6.969482e-05
39380	5.979644e-04	3.462880e-05
131338	2.480636e-04	1.519130e-05

Tabelle 7.1: „Eckenlayer“: Ergebnisse bei adaptiver Gitterverfeinerung

Freiheitsgrade	$\|f - f_h\|_{\max}$	$\|f - f_h\|_{L^2}$
289	1.085150e+00	2.326590e-01
504	6.134885e-01	1.504399e-01
918	3.921532e-01	8.718015e-02
1670	3.256176e-01	4.742402e-02
3291	2.587719e-01	2.256633e-02
7944	1.217857e-01	8.681484e-03
19816	8.944284e-02	2.755447e-03
38240	3.058579e-02	9.529964e-04
52754	1.145180e-02	6.467369e-04
99574	7.762168e-03	4.222369e-04

Tabelle 7.2: „Hutlayer“: Ergebnisse bei adaptiver Gitterverfeinerung

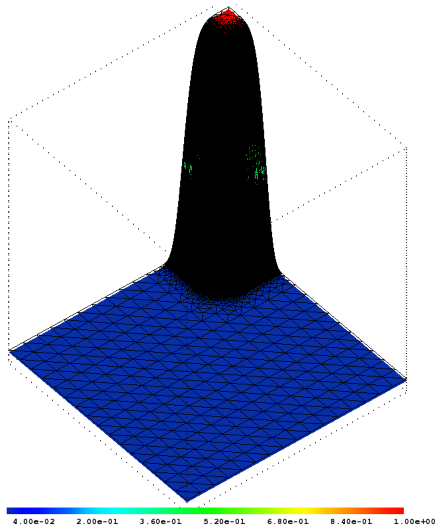


Abbildung 7.8: „Eckenlayer“ mit adaptiv verfeinertem Gitter (131338 Freiheitsgrade)

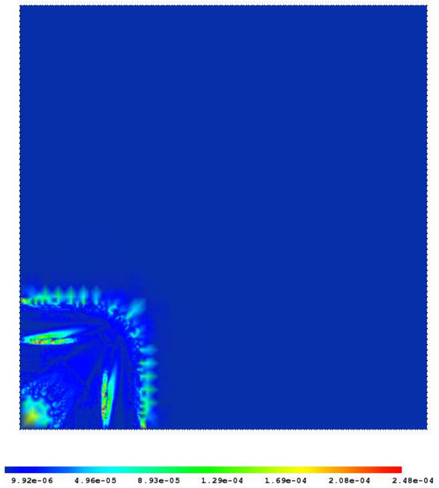


Abbildung 7.9: Fehler beim „Eckenlayer“ (131338 Freiheitsgrade)

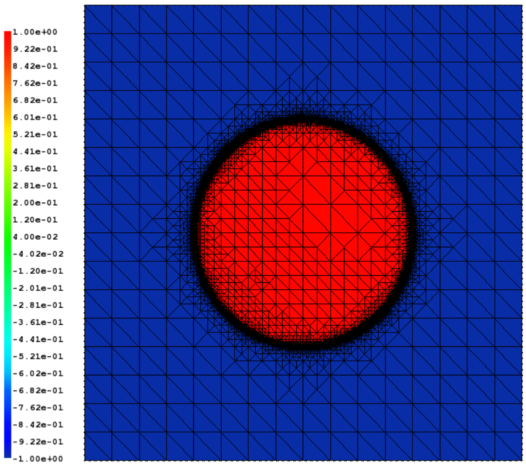


Abbildung 7.10: „Hutlayer“ mit adaptiv verfeinertem Gitter (99574 Freiheitsgrade)

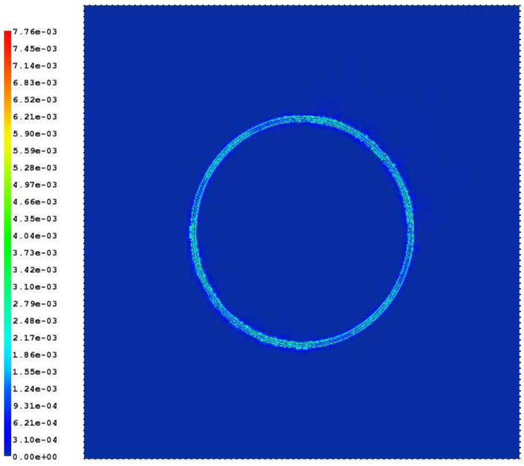


Abbildung 7.11: Fehler beim „Hutlayer“ (99574 Freiheitsgrade)

7.2.3 Anwendung auf das Splitting

Wir wenden also den Fehlerindikator $\hat{\eta}$ aus (7.2) auf jede der Geschwindigkeitskomponenten v_i an. Mit $D_i = \{T \in \mathcal{I} | \hat{\eta}_{v_i}(T) > \varepsilon_{Otol}\}$ wird die Menge aller Elemente bezeichnet, auf denen der Fehlerindikator über einer Schwelle ε_{Otol} liegt.
O.b.d.A. gelte für die $T_j \in D_i : \hat{\eta}_{v_i}(T_j) \geq \hat{\eta}_{v_i}(T_{j-1})$. Sei nun

$$d_i = \max_{T_j \in D_i} \{\hat{\eta}_{v_i}(T_j)\}$$

und

$$\bar{D}_i = \{T \in D_i \mid \hat{\eta}_{v_i}(T) \geq \frac{d_i}{n}\} \; ; \; n \in \mathbb{N} \; ,$$

wobei $\hat{\eta}_{v_i}(T)$ für den Fehler in der Geschwindigkeitskomponente v_i auf Element T steht.
Es werden alle Elemente aus $\bar{D}_1 \cup \bar{D}_2$ zur Verfeinerung an den oben beschriebenen Algorithmus übergeben, wodurch der neue Ansatzraum V_{k+1} entsteht.
Um weitere Interpolationen zu vermeiden, werden wir alle v_i weiterhin auf dasselbe Gitter verwenden. Denkbar wäre alternativ ein Aufsplitten der Gitterhierarchien für jede Geschwindigkeitskomponente, die dann jedoch einige Anstrengungen und dabei womöglich auch einen Genauigkeits- oder Stabilitätsverlust durch Abbildungen zwischen den Gittern erzeugen könnte.
Damit entsteht die folgende Hierarchie von Gittern bzw. Finite-Element-Räumen:

$$V_p \subset V_0 \subset V_1 \subset \cdots \subset V_k$$

Der Druck wird nach wie vor auf dem Gitter V_p berechnet. Mit V_0 bezeichnen wir das Startgitter der Geschwindigkeitskomponenten, welches durch eine globale rote Verfeinerung aus V_p hervorgeht. V_k mit $0 < k \in \mathbb{N}$ entsteht wie oben beschrieben auf der Basis von V_{k-1}.
Die $\inf - \sup$-Bedingung ist damit für den Raum $W_k = V_k \times V_p$ wie in Kapitel 4.4 beschrieben weiterhin erfüllt, da sie ja bereits für $W_0 = V_0 \times V_p$ erfüllt war.

Direkt zu Beginn muss v_p hinreichend fein gewählt werden, weil das Druckgitter bei diesem Vorgehen unverändert bleibt. Da der Druck jedoch häufig weitaus glatter ist als die Geschwindigkeitskomponenten und da die Laplace-Gleichungen, die zur Berechnung des Drucks verwendet werden, gut konditioniert sind, ist dies in der Regel kein Nachteil. Wird jedoch eine Anpassung auch im Druck gewünscht, so kann diese auf natürliche Weise mit Hilfe des Postprocessings aus Kapitel 6 erreicht werden. Hierbei wird der Druck im Postprocessingschritt auf dem adaptiv verfeinerten Gitter berechnet. Die $\inf - \sup$-Bedingung bleibt dabei gewahrt, da die Geschwindigkeit auf demselben Gitter jedoch mit quadratischen Funktionen bzw. auf einem global regulär verfeinerten Gitter berechnet wird.

Anzumerken bleibt, dass es bei physikalischen Problemen kein Nachteil ist, dass die Gitteränderung sich ausschließlich an der Geschwindigkeit orientiert, da Geschwindigkeit und Druck stärker voneinander abhängen als bei den *analytischen* Beispielen. Das bedeutet, dass auch dort ein komplexeres Verhalten im Druck zu erwarten ist, wo dies in der Geschwindigkeit vorliegt und umgekehrt.

7.2.4 Auswirkung der Gitteränderung auf das Splitting

Weit schwieriger als die Frage nach den zu verfeinernden Elementen ist die nach der Interpolation der Funktion an den neu entstandenen Knoten zu beantworten. Abhängig von der

Ordnung l des eingesetzten BDF-Verfahrens müssen für die l zurückliegenden Näherungen an die Geschwindigkeit an den neuen Knoten Funktionswerte interpoliert werden. Eine einfache lineare Interpolation zwischen den beiden auf der Kante liegenden alten Werten ist sehr effektiv mit Hilfe des Prolongationsoperators P zu berechnen. Diese Näherung berücksichtigt jedoch nicht die geforderte Divergenzfreiheit von v, so dass die im Splittingalgorithmus häufig verwendete Annahme, dass die zurückliegende Näherung $v_{m-j} \in V_{k-1}$ mit $0 < j < l$ divergenzfrei ist, an den neuen Knoten von V_k in der Regel verletzt wird.

Die Auswirkung findet sich primär im Druckaktualisierungsschritt, wodurch nach einer Gitteranpassung der Druck hier nicht in der gewohnten Genauigkeit aktualisiert werden kann. Die Größe des Genauigkeitsverlustes hängt von der Anzahl der neuen Knoten und von dem verwendeten BDF-Verfahren ab. Je höher die Ordnung des BDF-Verfahrens, umso größer ist die Störung durch die neuen Knoten und umso länger hält sie an. Die Auswirkungen bleiben so lange in immer schwächerer Form bestehen wie interpolierte Knoten in die Approximation von $\frac{\partial}{\partial t}$ eingehen.
Mehr als ein Verfeinerungsschritt im Ort pro Zeitschritt sollte vermieden werden, da die interpolierten Knoten dann u.U. wieder auf der Basis von zuvor interpolierten Knoten berechnet werden.

Eine Möglichkeit diesem Problem zu begegnen, wäre eine andere Interpolationstechnik. Versuche, die bei der Gradientenrekonstruktion verwendete Technik zu diesem Zweck zu adoptieren und um eine entsprechende Bedingung für die Divergenzfreiheit zu ergänzen, waren nicht für alle Testprobleme erfolgreich. Es kam in einigen Fällen zu Instabilitäten. Eine Alternative besteht darin, den gesuchten Wert zwischen den Werten der beiden benachbarten Knoten zu vermuten und Werte aus diesem Intervall zu testen, um eine divergenzfreiere Interpolation zu erreichen. Der Interpolationsoperator für das adaptive Splitting im Ort bleibt also weiterhin für Untersuchungen interessant.

Diese Probleme betreffen jedoch primär den Druck. Der Druckextrapolant $\bar{p}$ wird in jedem Splittingschritt mit Hilfe von (5.10) neu rekonstruiert. Da die Gleichung (5.10) lediglich die Divergenzfreiheit von v_m benutzt, um durch Termumformungen Ableitungen zweiten Grades zu vermeiden, ist $\bar{p}$ nicht in dem Maße betroffen wie der Druckaktualisierungsschritt. Da die Zwischengeschwindigkeit auf der Basis von $\bar{p}$ berechnet wird, ist die Geschwindigkeit in der Regel nicht oder kaum betroffen.

Wird der Druck also als Hilfsgröße gesehen, um die Geschwindigkeit zu berechnen, ist das vorgestellte Vorgehen problemlos. Alternativ kann man das Gitter sehr langsam ändern, um nicht zu viele neue Knoten pro Schritt zuzulassen.
Wenn das Verfahren in Zusammenhang mit einem Postprocessing-Schritt verwendet wird, reduzieren sich die Auswirkungen der neuen Knoten im Allgemeinen.

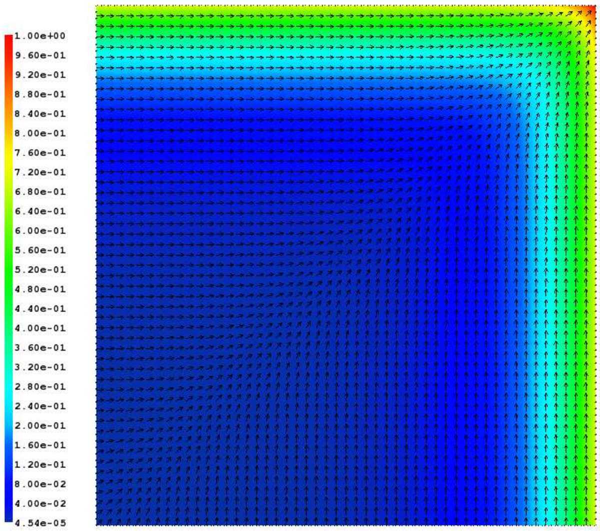


Abbildung 7.12: Geschwindigkeit v und $\|v\|_2$

7.2.5 Numerische Ergebnisse

Als Testproblem verwenden wir mit $\Omega = [0,1]^2$ und $t \in [0,4]$ die Funktion

$$\begin{aligned} v_1(x,y,t) &= \frac{\sin(\pi t)}{\sqrt{2}} e^{10(y-1)} , \\ v_2(x,y,t) &= \frac{\sin(\pi t)}{\sqrt{2}} e^{10(x-1)} . \end{aligned}$$

Der Druck wird so gewählt, dass an der Stelle, an die das Fluid strömt, ein Unterdruck herrscht, an dem Ort, den das strömende Fuid verlässt, hingegen ein Überdruck.

$$p(x,y,t) = -\sin(\pi t)(x^2 + y^2 - \frac{2}{3}) .$$

Erneut wird die rechte Seite f entsprechend angepasst. Damit gibt es nun in der Geschwindigkeit einen Randlayer, den es mit Hilfe der Ortsadaptivität aufzulösen gilt. Der Test wird mit einer $Re = 1000 = \frac{1}{\nu}$ und einer konstanten Zeitschrittweite von $\triangle t = 1/32$ durchgeführt. Ω wird im ersten Zeitschritt durch die kanonische Zerlegung des Einheitsquadrats mit 1089 Freiheitsgraden im Druck und 4225 in der Geschwindigkeit diskretisiert. Die Anzahl der Freiheitsgrade bezieht sich auf die Finite-Element-Räume vor dem Postprocessing-Schritt. Da ein expliziter Postprocessing-Schritt mit P2/P1 Taylor-Hood-Elementen verwendet wird, reicht diese geringe Zahl an Freiheitsgraden aus und der Druck wird, wie oben beschrieben, ebenfalls von der Gitterverfeinerung profitieren. Als Zeitschrittverfahren kommt ein BDF dritter Ordnung mit einer äquidistanten Zeitschrittweite von $\frac{1}{32}$ zum Einsatz.
Ob eine Verfeinerung im Ort nötig ist, wird alle fünf Schritte überprüft, und dann wird ggf. nach dem oben angegebenen Schema verfahren.

In der Abbildung 7.13 erkennt man die negativen Auswirkungen der Gitteränderung auf den Druck. Nach fünf Schritten wird zum ersten Mal das Gitter verändert, wodurch der

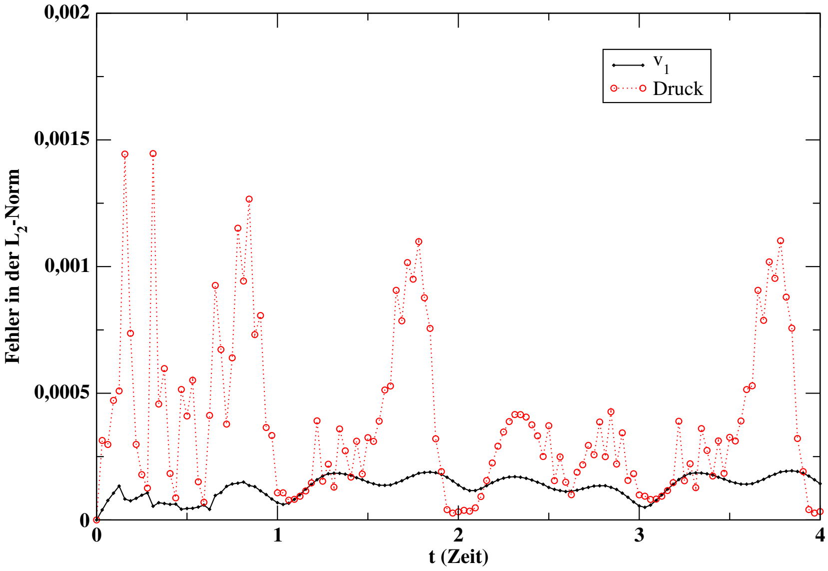


Abbildung 7.13: Fehlerverlauf in $v_1 \approx v_2$ und p beim Splitting mit adaptiver Gitteranpassung

Fehler im Druck ansteigt. Insgesamt bleibt der Fehler in der L_2-Norm des Drucks jedoch unter $1.5 \cdot 10^3$ klein.

Die Anpassung des Gitters wird durch die Abbildung 7.14 illustriert. Man erkennt, dass der Randlayer gefunden und aufgelöst wird.

In dieser Arbeit wurde auf eine Gittervergröberung verzichtet. Sie ist natürlich mit dem entsprechenden Aufwand realisierbar. Wegen den Störungen im Druck sollte bei Gitterveränderungen zurückhaltend agiert werden. Ein Knoten, der nur für wenige Schritte entfernt und anschließend wieder hinzugenommen wird, ist im Allgemeinen für den Splittingalgorithmus nicht sinnvoll.

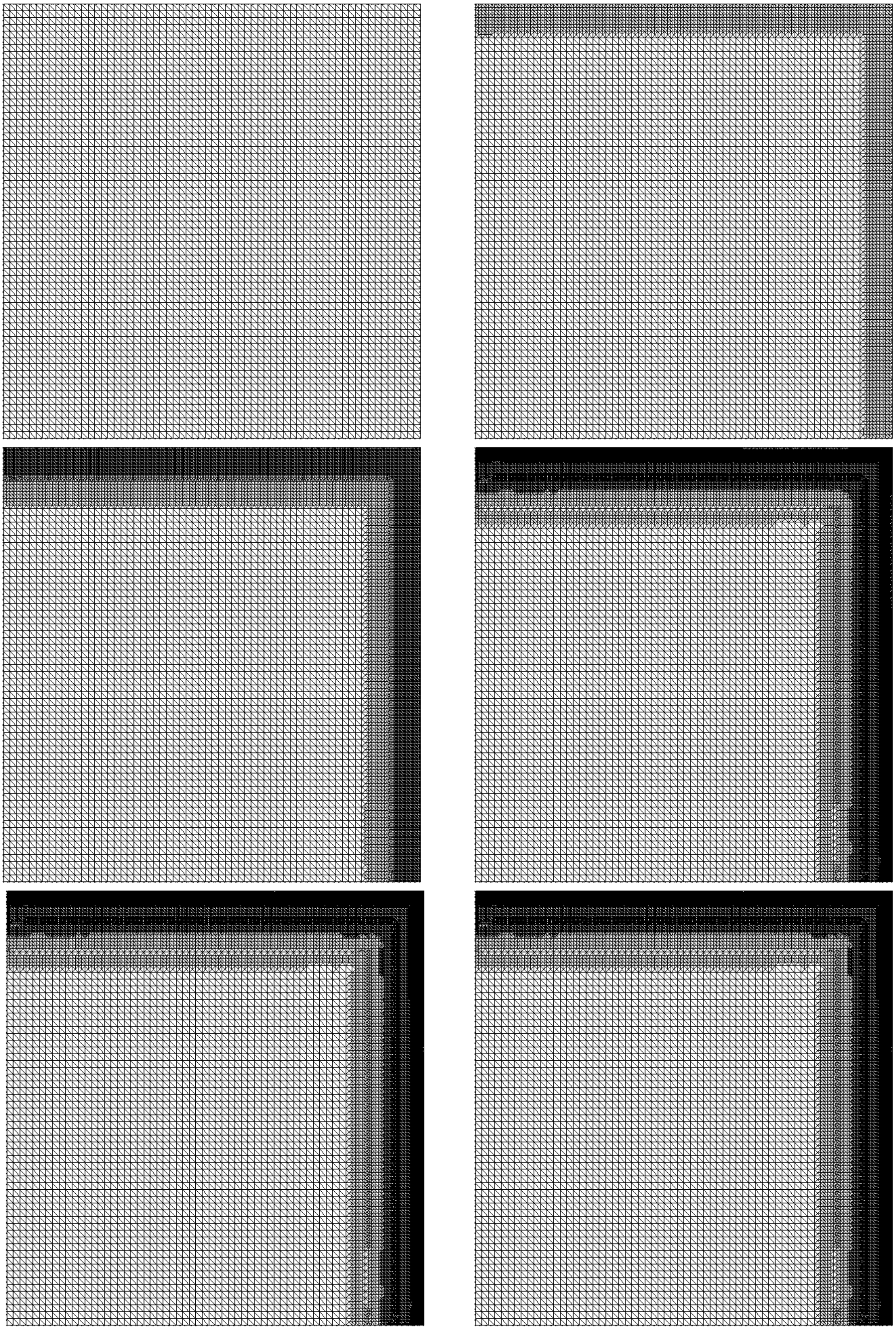


Abbildung 7.14: Gitter zu den Zeitpunkten $t = 0.0000$ (4225 Freiheitsgrade), $t = 0.1562$ (5963 Freiheitsgrade), $t = 0.3125$ (14102 Freiheitsgrade), $t = 0.4688$ (33622 Freiheitsgrade), $t = 2.5000$ und $t = 4.0000$ (34621 Freiheitsgrade)

7.3 Kombinierte Adaptivität in Ort und Zeit

Unter einem volladaptiven Splittingalgorithmus verstehen wir einen, bei dem beide Kontrollmechanismen verwendet werden.
Dabei ist es wesentlich, die höhere Genauigkeit von der adaptiven Steuerung der Schrittweite zu verlangen. Zum einen ist diese frei von negativen Begleiterscheinungen, zum anderen kann der Algorithmus zur Gitterverfeinerung nicht die einzelnen Fehlerquellen unterscheiden.
So würde der Fehlerindikator aus dem letzten Absatz natürlich auch Gebiete verfeinern, bei denen es zu Störungen auf Grund einer zu großen Zeitschrittweite kommt, obwohl die Ortsverfeinerung dort zu keinen Verbesserungen führen kann.
Daher geht man beim volladaptiven Splitting wie folgt vor:

1. Entsprechend Abschnitt 7.1 $\triangle t_{k+1}$ bestimmen
2. Alle m Zeitschritte die Lösung v durch den Fehlerindikator aus Abschnitt 7.2 abschätzen
    a) Ggf. ein neues Gitter generieren und die fehlenden Knoten der zurückliegenden l Zeitschritte mit Hilfe des Operators P auf das neue Gitter anpassen
    b) Auf dem neuen Gitter den Splittingschritt mit $\triangle t_{k+1}$ erneut durchführen

Wird ein Postprocessing-Schritt verwendet, so finden alle diese Berechnungen in dem Raum W_H statt. Der Raum W_h paßt sich entweder automatisch durch die quadratischen Ansatzfunktionen oder bei linearen Ansatzfunktionen durch eine globale, reguläre Verfeinerung an den Raum W_H an.

Als Testfall verwenden wir das schon in Abschnitt 7.2.5 vorgestellte Problem und benutzen erneut eine Splittingvariante mit explizitem Postprocessing-Schritt und P2/P1 Taylor-Hood-Elementen. Mit Hilfe der adaptiven Zeitschrittweitensteuerung bestimmt der

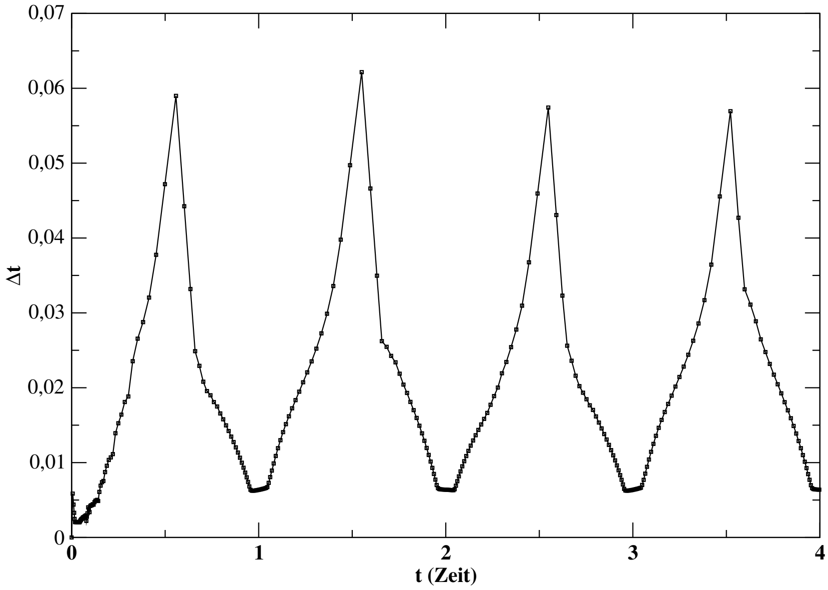


Abbildung 7.15: Gewählte Zeitschrittweite beim volladaptiven Splittingalgorithmus

Algorithmus die Schrittweite, wie in Abbildung 7.15 dargestellt wird, mit $\varepsilon_{Ttol} = 5 \cdot 10^{-4}$ und $\varepsilon_{Otol} = 5 \cdot 10^{-2}$. Durch die adaptive Wahl von $\triangle t_m$ wird das Intervall $[0, 4]$ nicht wie bei einer Zeitschrittweite von $\triangle = 1/32$ in 128 Schritten durchlaufen, sondern in 285.

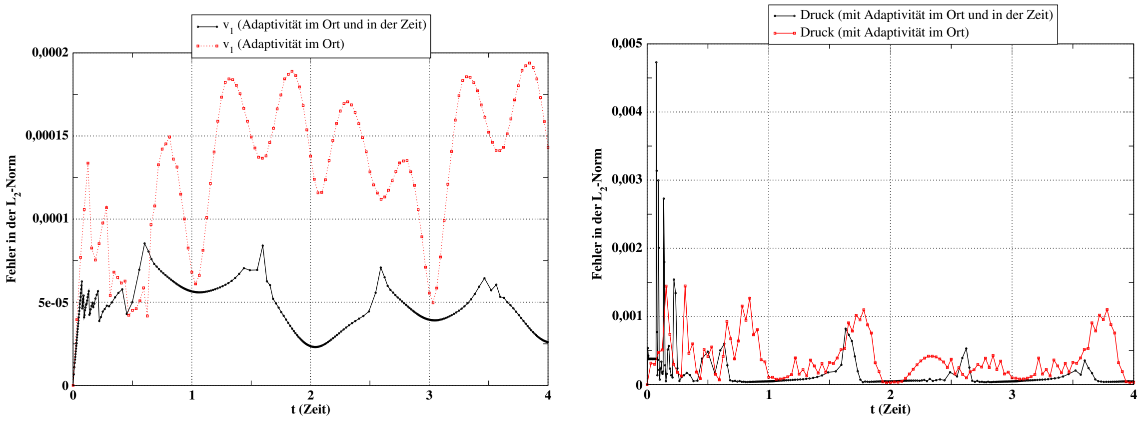


Abbildung 7.16: Fehlerentwicklung beim volladaptiven Splitting im Vergleich zu den Ergebnissen aus Abschnitt 7.2.5

Bei der Betrachtung des in der Abbildung 7.16 dargestellten Fehlerverlaufs fällt auf, dass die Störung im Druck hier größer ist als die in Abschnitt 7.2.5. Dies erklärt sich durch die kleinere Zeitschrittweite zum Zeitpunkt der Gitteränderung. Der Faktor $\beta_0/\triangle t$ verstärkt hier den beschriebenen Effekt.

Langfristig profitiert der Druck jedoch ebenfalls von der adaptiven Zeit- und Gitteranpassung, sodass, nachdem die größten Anpassungen im Gitter durchgeführt wurden, für $t > 0.5$ sowohl in der Geschwindigkeit als auch im Druck eine qualitativ bessere Lösung erreicht werden konnte. Wie Abbildung 7.16 zeigt, konnte die Geschwindigkeit durch das volladaptive Splitting in hohem Maße verbessert werden.

Abbildung 7.17 veranschaulicht die Anpassung des Gitters. Mit 30678 Freiheitsgraden ist das letzte Gitter ungefähr 10% kleiner als das Gitter ohne Zeitadaptivität mit 33662. Es ergab sich eine geringere Zahl von Freiheitsgraden, obwohl etwa doppelt so oft die Möglichkeit zu einer Gitteranpassung bestand, da die Zeitschrittweite kleiner und dadurch die Anzahl der Zeitschritte größer war. Man kann daraus schließen, dass bei einer Zeitschrittweite von $\triangle t = 1/32$, wie oben bereits beschrieben, noch einige Elemente verfeinert wurden, bei denen der Fehler seine Ursache nicht in einer unzureichenden Diskretisierung im Ort hatte.

Dies zeigt eine der Stärken des vorgestellten Vorgehens. Bei Problemen wie dem im nächsten Kapitel vorgestellten Driven Cavity Problem, wo das Hauptinteresse auf der Geschwindigkeit liegt und dies zu einem festen Zeitpunkt, kann der volladaptive Algorithmus effektiv eingesetzt werden.

Problematischer sind Fälle, in denen der Druck zu allen Zeitpunkten von großem Interesse ist und bei denen kleine Zeitschrittweiten mit großen Anpassungen am Gitter zusammenfallen. Hier bietet es sich u.U. an, $\bar{p}$ auszuwerten, was zwar, wie in Abbildung 5.8 deutlich wurde, in der Regel zu einem Verlust an Genauigkeit führt, jedoch robuster gegenüber dem oben beschriebenen Effekt ist.

Ein solcher Fall ist das Problem „Fluss um einen Zylinder“ aus Kapitel 8.2, bei dem völlig auf eine adaptive Anpassung des Gitters verzichtet wurde.

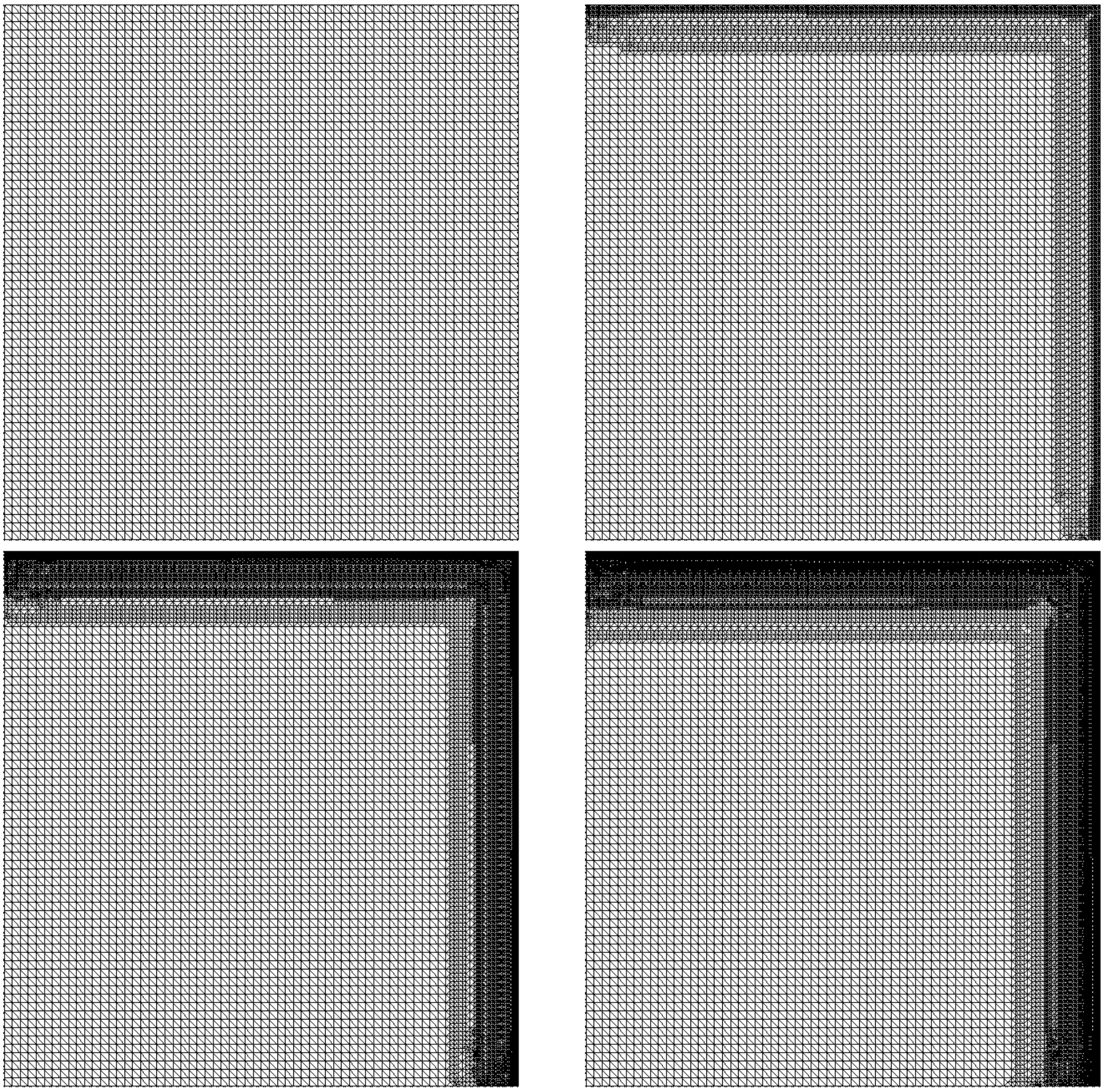


Abbildung 7.17: Gitter zu den Zeitpunkten $t = 0.0000$ (4225 Freiheitsgrade), $t = 0.1550$ (8109 Freiheitsgrade), $t = 0.3029$ (21834 Freiheitsgrade), $t = 0.4516$ (30678 Freiheitsgrade)

8 Numerische Resultate auf Standardproblemen der Strömungsmechanik

8.1 Driven Cavity

Ein klassisches Testproblem der numerischen Strömungsmechanik ist das so genannte Driven Cavity Problem. Als Modell dient dabei ein mit einem Fluid gefüllter rechteckiger Behälter. Drei Seiten des Behälters sind fest, die vierte wird über ein Antriebsband mit einer vorgegebenen Geschwindigkeit bewegt, wodurch sich das Fluid in dem Gefäß in Bewegung setzt. Abbildung 8.1 zeigt eine schematische Darstellung.

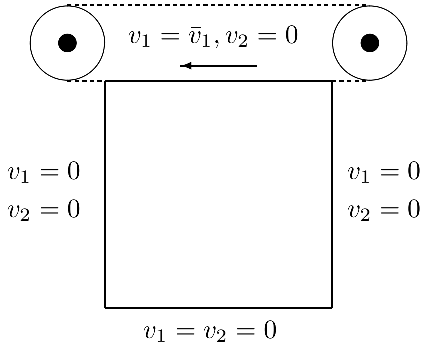


Abbildung 8.1: Schematische Darstellung der Randbedingungen des Driven Cavity Problems

Die dadurch induzierte Hauptströmung bewegt das Fluid entgegen dem Uhrzeigersinn. In den beiden unteren Ecken $(0,0)$ und $(0,1)$ kommt es zur Bildung von entgegengesetzten Strömungen. Im Laufe der Zeit bewegt sich das ruhige Zentrum der Hauptströmung von der oberen linken Ecke ins Zentrum des Gefäßes, worin der Name des Problems begründet liegt. Abhängig von der Höhe der Reynoldszahl konvergiert das Problem gegen einen stationären Zustand. Bis zu einer Reynoldszahl von 7500 gilt dies als gesichert. Dabei wird die an der beweglichen Wand auftretende Geschwindigkeit als charakteristische Geschwindigkeit des Problems betrachtet. Als charakteristische Länge wird 1 verwendet, also ein Gefäß mit der ¡¡¡¡entsprechenden Kantenlänge, folglich sei $\Omega = [0,1]^2$.

Wir fassen das Driven Cavity Problem hierbei als einen zeitabhängigen physikalischen Prozess auf, bei dem die Strömung des Fluids sowie der Druck über den gesamten Zeitraum vom Einsetzen der Strömung bis zum Erreichen des stationären Zustands von Interesse ist.
Ist man lediglich an dem stationären Zustand interessiert, empfehlen sich andere Verfahren, welche den stationären Zustand mit Hilfe der zeitunabhängigen Navier-Stokes-Gleichung

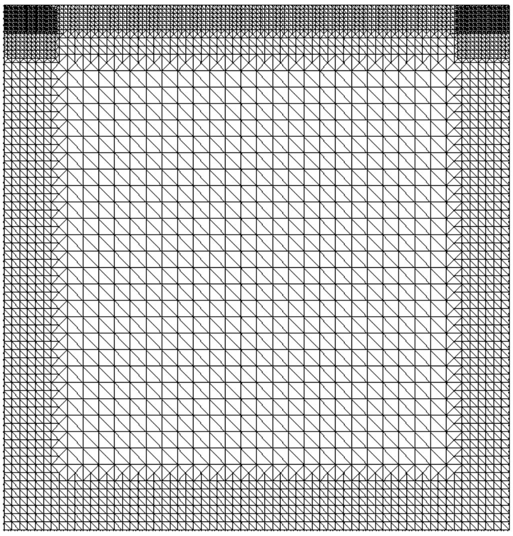


Abbildung 8.2: Gitter für das Driven Cavity Problem

berechnen, vgl. hierzu z.B. [42].

Grundlage des auf Ω erzeugten Gitters ist eine grobe, kanonische Zerlegung des Einheitsquadrats. Alle Elemente mit einem Abstand von 0.1 vom Rand werden wie in Kapitel 7.2 beschrieben verfeinert. Anschließend wird der Verfeinerungsprozess für alle Elemente in den beiden oberen Ecken $[0, 0.1] \times [0.9, 1]$ und $[0.9, 1] \times [0.9, 1]$ erneut durchgeführt. Abschließend wird diese Prozedur noch einmal für alle Elemente aus $[0, 1] \times [0.95, 1]$ wiederholt. Dadurch ergibt sich das Gitter aus Abbildung 8.2. Es besitzt 3775 Knoten, eine minimale Gittergröße von $h_{\min} \approx 5.524 \cdot 10^{-03}$ sowie eine maximale von $h_{\max} \approx 4.419 \cdot 10^{-02}$.

8.1.1 Regularisiertes Driven Cavity

Der durch das Band ausgelöste Antrieb wird beim regularisierten Driven Cavity Problem durch $\bar{v}_1(x) = -16x^2(1-x)^2$ definiert.

Neben der eigentlichen Strömung v sind auch die durch die Strömung berechenbare Stromfunktion ψ und die Wirbelfunktion ω von Interesse, wobei gilt:

$$v_1 = \frac{\partial \psi}{\partial y}, \quad v_2 = -\frac{\partial \psi}{\partial x}, \tag{8.1}$$

$$\omega = \frac{\partial u_2}{\partial x} - \frac{\partial u_1}{\partial y}. \tag{8.2}$$

In Veröffentlichungen, die Spektrale Verfahren zur Ortsdiskretisierung verwenden, wird in der Regel der stationäre Zustand durch

$$\frac{\|\omega^{n+1} - \omega^n\|_{\max}}{\triangle t^{n+1} \cdot \|\omega^{n+1}\|_{\max}} \le \varepsilon_w \tag{8.3}$$

definiert. Wir werden hingegen die eigentliche Strömungsfunktion als Abbruchkriterium verwenden, um den Aufwand für die Bildung von ω^{n+1} in jedem Zeitschritt zu sparen. Auch ist ω auf Grund der Gradientenrekonstruktion für gitterspezifische Einflüsse anfälliger als v. Wir charakterisieren also einen stationären Zustand im Folgenden durch

$$\frac{\|v_i^{n+1} - v_i^n\|_{\max}}{\triangle t \cdot \|v_i^{n+1}\|_{\max}} < \varepsilon_v \; (i = 1, 2) \; , \tag{8.4}$$

wobei wir für das regularisierte Driven Cavity $\varepsilon_v = 1 \cdot 10^{-3}$ wählen.
Die Stromfunktion wird durch Lösen der partiellen Differentialgleichung

$$\begin{aligned} \nabla^2 \psi &= -\omega \quad \text{in } \Omega \\ \psi &= 0 \quad \text{auf } \partial\Omega \end{aligned}$$

berechnet.

Tabelle 8.1: Stromfunktion beim regularisierten Driven Cavity mit Re=100

max $\lvert\psi\rvert$ (Stromfunktion) 14725 Freiheitsgrade in v und 3775 Freiheitsgrade in p $\triangle t = \frac{1}{32}$ BDF(3)		
[4]	[22]	N
x-Koordinate		
0.37-0.42	0.37 - 0.42	0.390625
y-Koordinate		
0.73-0.78	0.73 - 0.78	0.75
max $\lvert\psi\rvert$		
$8.2694 - 8.5371 \cdot 10^{-2}$	$8.2695 - 8.3402 \cdot 10^{-2}$	$8.3692 \cdot 10^{-2}$

Tabelle 8.2: Driven Cavity mit Re=400 und Postprocessing

max $\lvert\psi\rvert$ (Stromfunktion) 58153 Freiheitsgrade in v und 14725 Freiheitsgrade in p $\Delta t = \frac{1}{32}$ BDF(3)		
[4]	[22]	Pq
x-Koordinate		
0.40-0.43	0.40 - 0.43	0.421875
y-Koordinate		
0.58-0.63	0.58 - 0.65	0.6171875
max $\lvert\psi\rvert$		
$8.5192 - 8.5777 \cdot 10^{-2}$	$8.4007 - 8.5715 \cdot 10^{-2}$	$8.5870 \cdot 10^{-2}$

Tabelle 8.1 enthält den maximalen Wert von $|\psi|$ auf den inneren Freiheitsgraden. Dabei werden die Resultate aus [4] und [22] mit den Ergebnissen vergleichen, die durch das Standardsplitting (N) mit einer impliziten Behandlung des Konvektionsterms aus Kapitel 5.2

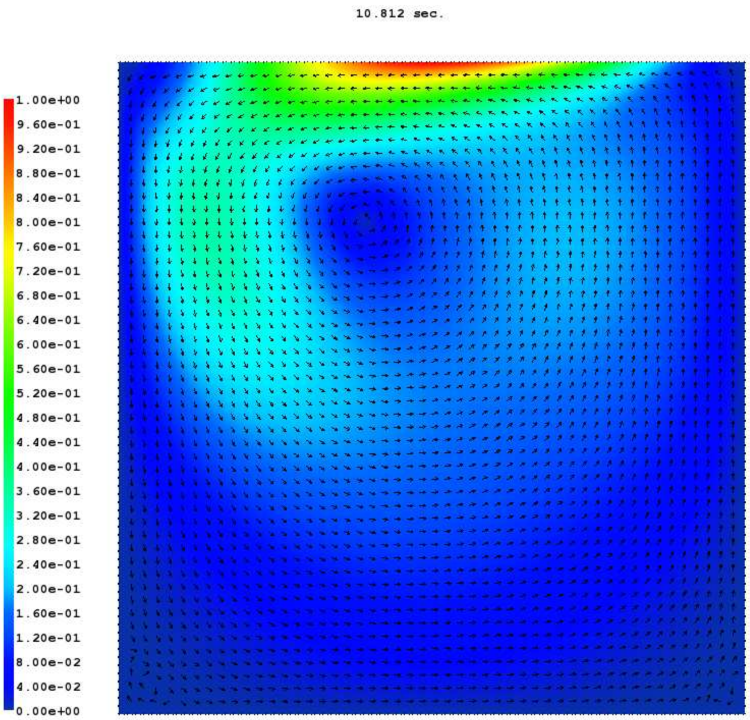


Abbildung 8.3: v und $\|v\|_2$ beim regularisierten Driven Cavity mit Re=100

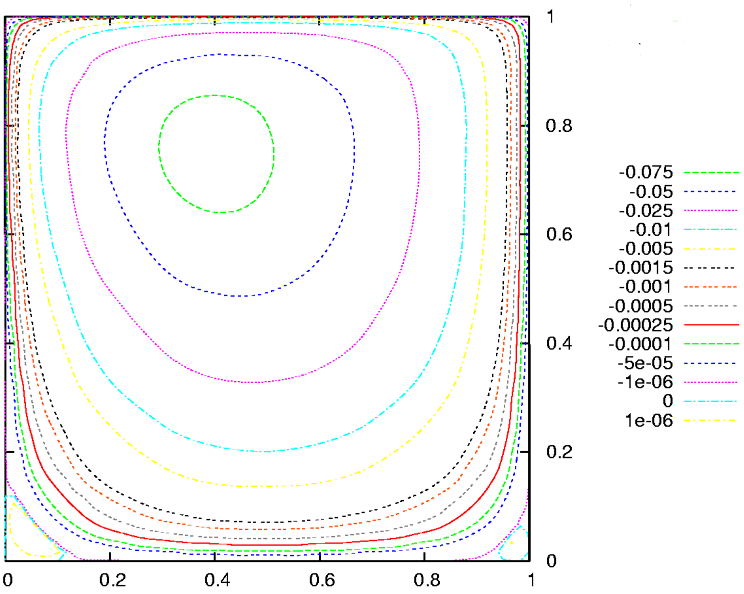


Abbildung 8.4: Stromfunktion beim regularisierten Driven Cavity mit Re=100

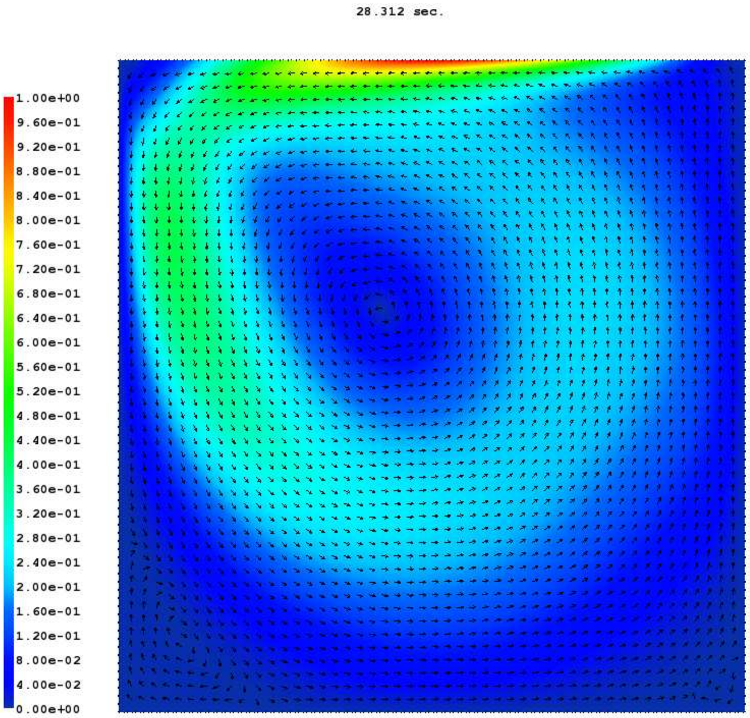


Abbildung 8.5: v und $\|v\|_2$ beim regularisierten Driven Cavity mit Re=400 und Postprocessing

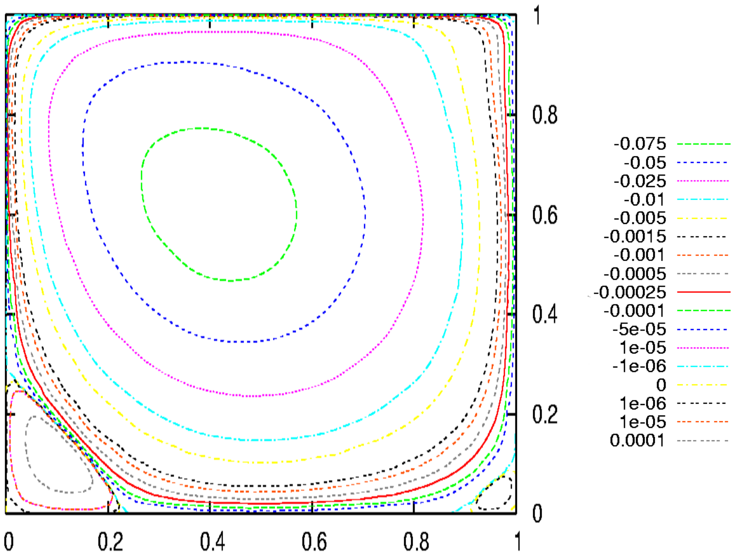


Abbildung 8.6: Stromfunktion beim regularisierten Driven Cavity mit Re=400 und Postprocessing

erzielt worden sind. Die in Tabelle 8.2 aufgeführten Ergebnisse wurden hingegen mit Hilfe eines Postprocessings mit quadratischen finiten Elementen (Pq) aus Kapitel 6.2 erzielt.

Der stationäre Zustand stellte sich im Fall $Re = 100$ nach dem oben angegebenen Kriterium zum Zeitpunkt $t = 10.8125$ ein, ε_w aus (8.3) betrug zu diesem Zeitpunkt $2.8379 \cdot 10^{-04}$. Im Fall $Re = 400$ stellte sich zum Zeitpunkt $t = 28.3125$ ein, ε_w aus (8.3) betrug zu diesem Zeitpunkt $2.829697 \cdot 10^{-04}$.

Wie man den Tabellen 8.1 und 8.2 entnehmen kann, war es mit Hilfe des Splittingalgorithmus möglich, eine hinreichend genaue Lösung mit einer vergleichsweise geringen Anzahl an Freiheitsgraden zu berechnen.

8.1.2 Driven Cavity

Die häufigste Variante des Driven Cavity Problems stellt die Wahl von $\bar{v}_1(x) = -1$ dar. Durch den so ausgelösten Antrieb kommt es an den beiden oberen Ecken zur Bildung von starken Singularitäten, welche hohe Anforderungen an das Verfahren zur Lösung der partiellen Differentialgleichungen im Ort stellen. Die Abbildungen 8.8 und 8.10 zeigen die stationären Lösungen für $Re = 1000$ und veranschaulichen die Singularitäten in den Ecken. Werden globale Methoden wie die Spektralen Verfahren eingesetzt, ist es in der Regel nötig, so wie bei [22], die entstehenden Singularitäten zuvor herauszurechnen. Dadurch sind die auf diese Weise berechneten Lösungen sehr genau, jedoch kann man diese Vorgehensweise und damit das verwendete Verfahren nicht ohne Weiteres auf Problemstellungen mit komplexeren Geometrien übertragen. Mit Hilfe der Finite-Element-Methode ist es möglich, das Problem mit vorhandenen Singularitäten direkt zu lösen. Die Singularitäten verschlechtern dabei die Genauigkeit der berechneten Approximation im Vergleich zu reguläreren Problemen, wie z.B. der Driven Cavity Variante aus Abschnitt 8.1.1. Insgesamt ist es daher nicht zu erwarten, die Benchmarkwerte der Veröffentlichungen mit Spektralen Verfahren wie [15] und [5] zu erreichen.

Um die verschiedenen Varianten des Splittingalgorithmus zu untersuchen, vergleichen wir zunächst die Ergebnisse für den Fall $\nu = 1/1000$ bzw. $Re = 1000$. Als Vergleichswerte verwenden wir das Profil der Geschwindigkeitskomponenten entlang der Geraden $x = 0.5$ sowie $\max|\psi|$.

Dabei bezeichnen wir wie schon im letzten Abschnitt die verschiedenen Varianten des Splittings wie folgt: N steht für das Standardsplitting mit einer impliziten Behandlung des Konvektionsterms aus Kapitel 5.2, Pq für das Splitting mit einem Postprocessingschritt mit quadratischen finiten Elementen aus Kapitel 6.2 und Pl für das mit linearen finiten Elementen aus Kapitel 6.3. Dazu wird der Wert für das Abbruchskriterium ε_v aus der Gleichung (8.4) angegeben.

Die in den Tabellen aufgeführte Anzahl an Freiheitsgraden bezieht sich bei den Splittingvarianten Pq und Pl auf den Postprocessingschritt. Wie man an den Tabellen 8.3 und 8.4 erkennen kann, zeigt sich erwartungsgemäß, dass für diese Reynoldszahl die Variante Pq, gefolgt von Pl, die besten Ergebnisse liefert. Durch ein schärferes Abbruchskriterium gelingt es dem Standardsplitting, die Ergebnisse weiter zu verbessern, da hierdurch mehr

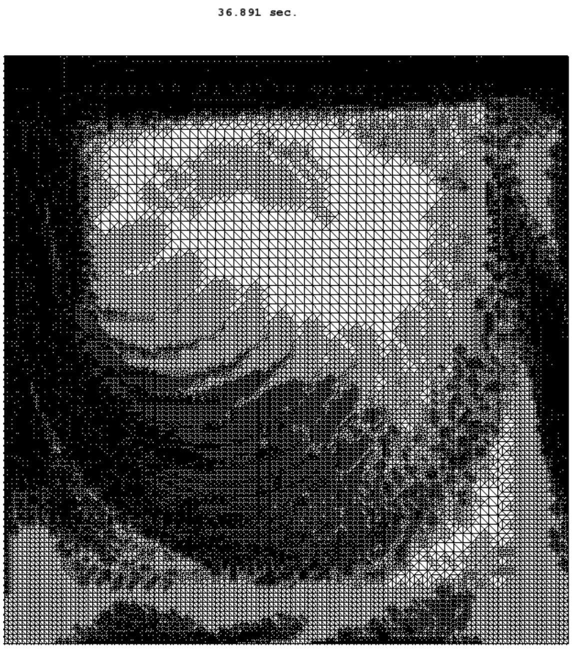


Abbildung 8.7: Adaptiv verfeinertes Gitter mit 81187 Knoten, Driven Cavity mit Re=1000

Picard-Iterationen durchgeführt werden.

Die erfolgreichste Variante Pq wird zusätzlich auf einem adaptiv verfeinerten Gitter getestet. Das Startgitter entspricht erneut der Abbildung 8.2 mit 58153 Freiheitsgraden für v und 14725 für p im Postprocessingschritt. Der Algorithmus darf bis zu 100000 Knoten im Gitter des Vorkonditionierungsschritts verwenden und prüft alle 50 Zeitschritte, ob und wo dazu die Notwendigkeit besteht. Die abgebildeten Gitter entsprechen den Gittern von v im Vorkonditionierungsschritt. Die Abbildung 8.7 zeigt den Abschluss des Verfeinerungsprozesses. Dabei wurden 323248 Freiheitsgrade für jede Geschwindigkeitskomponente und 81187 Freiheitsgrade für den Druck verwendet.

Tabelle 8.3: Stromfunktion beim Driven Cavity mit Re=1000

max $\lvert\psi\rvert$ (Stromfunktion) , $\triangle t = 1/64$						
mit 231121 Freiheitsgraden in v und 58153 in p						adaptives Gitter
[15]	[5]	$N_{1\cdot 10^{-3}}$	$Pq_{1\cdot 10^{-3}}$	$Pl_{1\cdot 10^{-3}}$	$N_{5\cdot 10^{-4}}$	$Pq_{5\cdot 10^{-4}}$
x-Koordinate						
0.4687	0.4692	0.4726562	0.46875	0.46875	0.46875	0.46875
y-Koordinate						
0.5655	0.5652	0.5625	0.5625	0.5625	0.5625	0.5625
max $\lvert\psi\rvert$						
0.117929	0.1189366	0.135	0.128	0.130	0.130	0.122

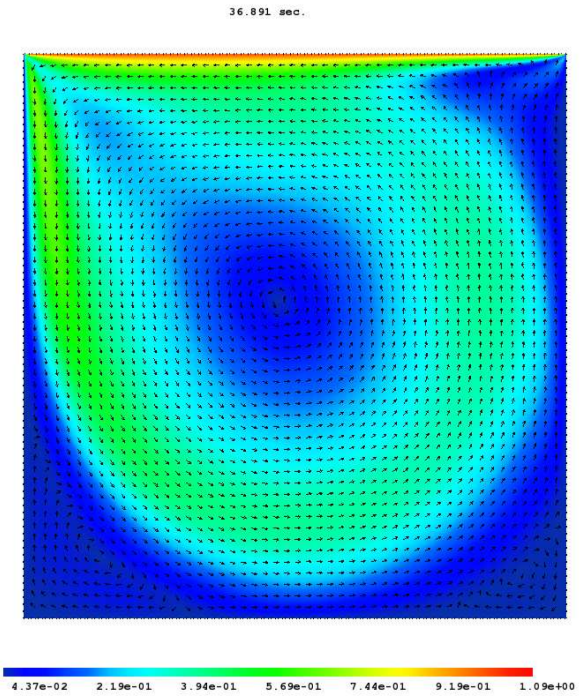


Abbildung 8.8: v und $\|v\|_2$ beim Driven Cavity mit Re=1000

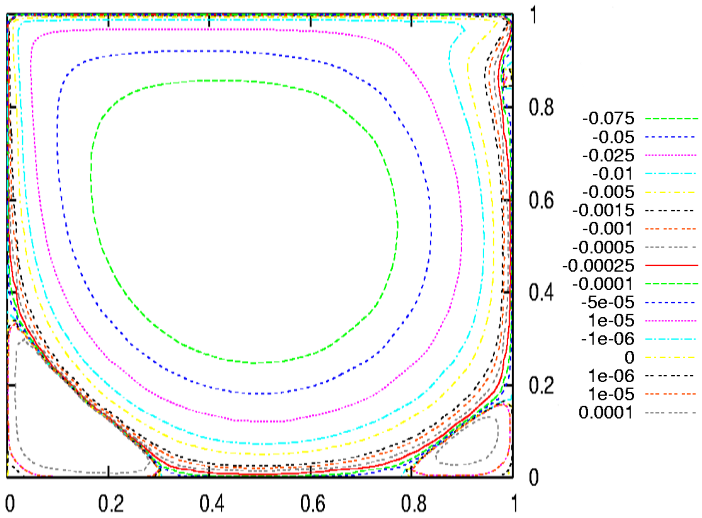


Abbildung 8.9: Stromlinien beim Driven Cavity mit Re=1000

	v_1 mit 231121 Freiheitsgraden in v und 58153 in p						adaptives Gitter
y	[15]	[5]	$N_{1\cdot 10^{-3}}$	$Pq_{1\cdot 10^{-3}}$	$Pl_{1\cdot 10^{-3}}$	$N_{5\cdot 10^{-4}}$	$Pq_{5\cdot 10^{-4}}$
1.0000	-1.00000	-1.0000000	-1.000	-1.000	-1.000	-1.000	-1.000
0.9766	-0.65928	-0.6644227	-0.710	-0.685	-0.689	-0.690	-0.667
0.9688	-0.57492	-0.5808359	-0.635	-0.597	-0.610	-0.611	-0.584
0.9609	-0.51117	-0.5169277	-0.579	-0.547	-0.558	-0.560	-0.523
0.9531	-0.46604	-0.4723329	-0.545	-0.511	-0.517	-0.518	-0.485
0.8516	-0.33304	-0.3372212	-0.393	-0.369	-0.375	-0.376	-0.350
0.7344	-0.18719	-0.1886747	-0.217	-0.205	-0.209	-0.210	-0.197
0.6172	-0.05702	-0.0570178	-0.068	-0.064	-0.066	-0.066	-0.061
0.5000	0.06080	0.0620561	0.067	0.064	0.065	0.064	0.061
0.4531	0.10648	0.1081999	0.119	0.114	0.115	0.115	0.109
0.2813	0.27805	0.2803696	0.312	0.297	0.300	0.301	0.284
0.1719	0.38289	0.3885691	0.450	0.422	0.428	0.381	0.398
0.1016	0.29730	0.3004561	0.360	0.332	0.355	0.344	0.311
0.0625	0.20196	0.2023300	0.240	0.222	0.231	0.232	0.210
0.0547	0.18109	0.1812881	0.214	0.198	0.206	0.210	0.187
0.0000	0.00000	0.0000000	0.000	0.000	0.000	0.000	0.000

Tabelle 8.4: Funktionswerte von v_1 entlang der Geraden $x = 0.5$ beim Driven Cavity Problem mit Re=1000 und $\triangle t = 1/64$

Re	x-Koordinate	y-Koordinate	$\max\lvert\psi\rvert$
3200	0.484375	0.539062	0.1373552
5000	0.484375	0.312500	0.1483011

Tabelle 8.5: Stromfunktion ($\max\lvert\psi\rvert$) beim Driven Cavity mit 58153 Freiheitsgraden in v und in p 14725; $\triangle t = 1/64$

Für höhere Reynoldszahlen kann die explizite Variante Pq nicht mehr eingesetzt werden, daher entstammen die folgenden Ergebnisse alle der Variante Pl.

Dass der Algorithmus die stationäre Lösung für hohe Reynoldszahlen wie $Re = 5000$ mit ausgeprägten Singularitäten berechnen kann, zeigt die extreme Robustheit dieser Splittingvariante.

In der Regel liegen an den bei [15] angegebenen y-Koordinaten keine Knoten. Die Vergleichswerte des Splittingverfahrens aus den Tabellen 8.4 und 8.6 entstanden durch lineare Interpolation.

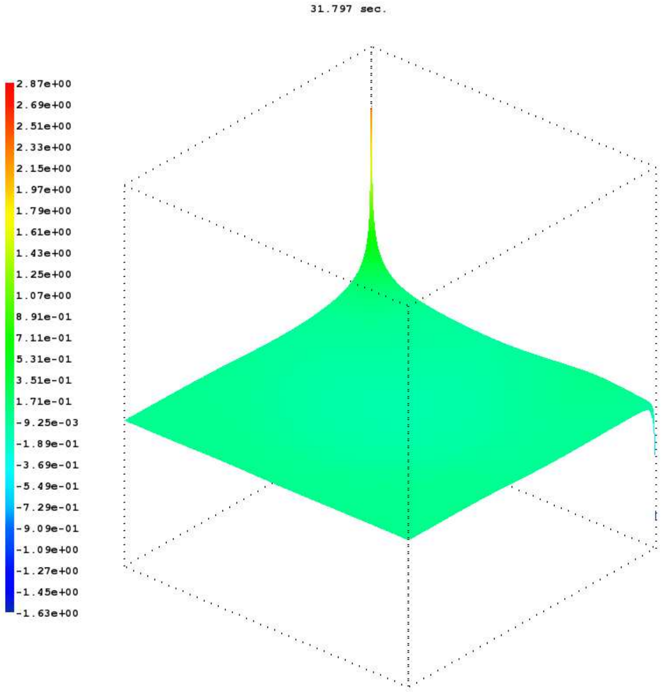


Abbildung 8.10: Druck beim Driven Cavity mit Re=1000

	Re=3200		Re=5000	
y	[15]	P_l	[15]	P_l
1.0000	-1.00000	-1.000	-1.00000	-1.000
0.9766	-0.53236	-0.564	-0.48223	-0.556
0.9688	-0.48296	-0.528	-0.46120	-0.551
0.9609	-0.46547	-0.519	-0.45992	-0.556
0.9531	-0.46101	-0.511	-0.46036	-0.559
0.8516	-0.34682	-0.397	-0.33556	-0.433
0.7344	-0.19791	-0.236	-0.20087	-0.320
0.6172	-0.07156	-0.092	-0.08183	-0.110
0.5000	0.04272	0.040	0.03038	0.040
0.4531	0.08663	0.092	0.07404	0.098
0.2813	0.24427	0.277	0.22855	0.030
0.1719	0.34323	0.387	0.33050	0.410
0.1016	0.41933	0.474	0.40435	0.476
0.0625	0.35344	0.434	0.42901	0.499
0.0547	0.32407	0.403	0.41165	0.479
0.0000	0.00000	0.000	0.00000	0.000

Tabelle 8.6: Funktionswerte von v_1 entlang der Geraden $x = 0.5$ beim Driven Cavity Problem mit verschiedenen Reynoldszahlen mit $\triangle t = 1/64$

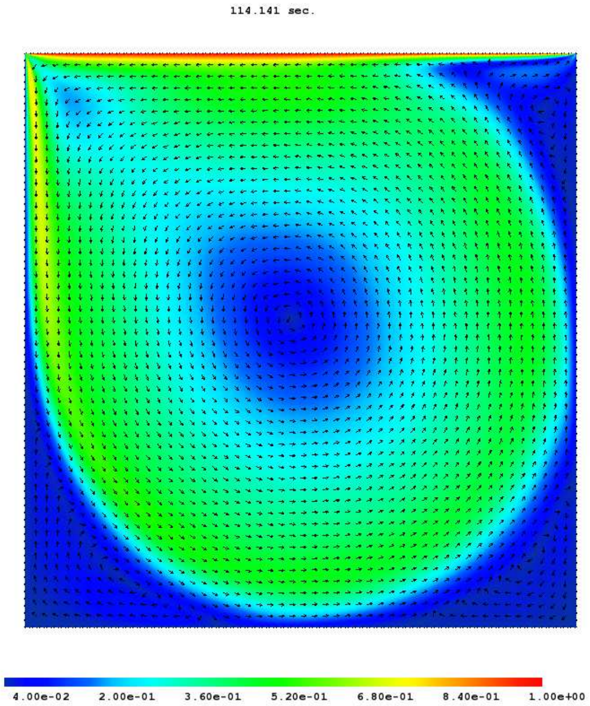


Abbildung 8.11: v und $\|v\|_2$ beim Driven Cavity mit Re=3200

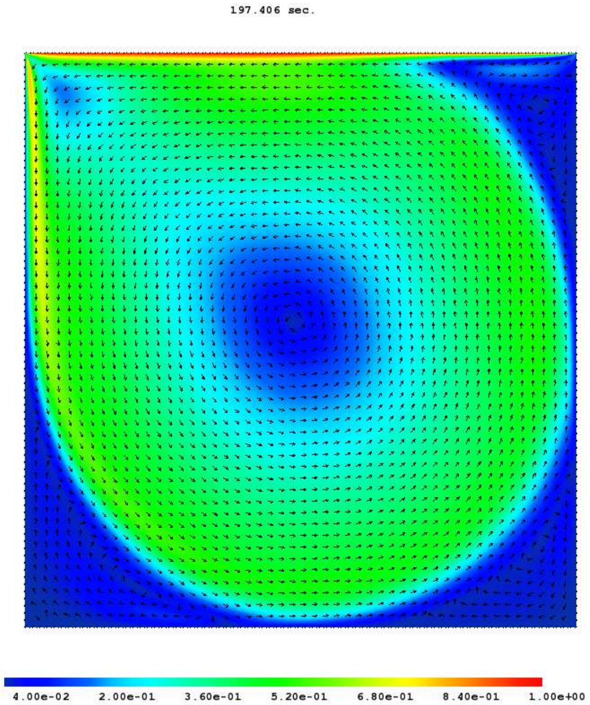


Abbildung 8.12: v und $\|v\|_2$ beim Driven Cavity mit Re=5000

8.2 Das DFG-Problem „Fluss um einen Zylinder“

Die folgende Fragestellung stammt aus dem DFG-Forschungsprogramm “Flow Simulation on High Performance Computers”. Das Problem ist von besonderem praktischen Interesse, da es Fragen aus den Ingenieurwissenschaften aufgreift. Hintergrund dieses Problems ist es, Versuche im Windkanal durch numerische Simulationen zu ergänzen und ggf. eines Tages zu ersetzen. Das 2D Testproblem, beschrieben in [36], besteht aus dem Fluss um einen Zylinder mit kreisförmigem Querschnitt mit dem Durchmesser $d = 0.1\ m$. Die Geometrie und die Randbedingungen werden in Abbildung 8.13 dargestellt. Ω ist dabei der Strömungskanal mit dem Zylinder. Die spezifische Viskosität wird mit $\nu = 10^{-3}\ \frac{\mathrm{m}^2}{\mathrm{s}}$ angegeben und die Dichte mit $\varrho = 1\ \frac{\mathrm{kg}}{\mathrm{m}^3}$, so dass in diesem Beispiel $p = \hat{p}$ gilt.

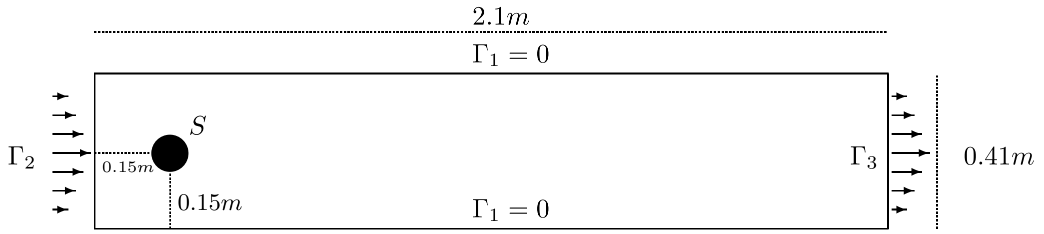


Abbildung 8.13: Schematische Darstellung der Randbedingungen des DFG-Problems *Fluss um einen Zylinder*

Die Randbedingungen für Γ_2 sind abhängig von der Variante des Testproblems. Wir werden im Folgenden zwei Testprobleme aus [36] betrachten: Zum einen das Problem 2D-2 (unsteady,periodic), für welches die Stärke des einströmenden Fluids durch

$$v(0,y,t) = 0.41^{-2}(6y(0.41-y),0)\ ,\quad 0 \le y \le 0.41\ . \tag{8.5}$$

gegeben wird. Diese Randbedingungen erzeugen unabhängig von der Startkonfiguration nach einiger Zeit ein periodisches Verhalten der Lösung.

Zum anderen wird das Problem 2D-3 (unsteady) behandelt, für welches die Stärke des einströmenden Fluids bei Γ_2 wie folgt definiert ist:

$$v(0,y,t) = 0.41^{-2}\sin(\pi t/8)(6y(0.41-y),0)\ ,\quad 0 \le y \le 0.41\ . \tag{8.6}$$

Die Strömung soll dabei über einen Zeitraum von 8 Sekunden untersucht werden, womit $t \in [0,8]$ ist.

Die Bedingungen für den Rand Γ_3 sind nicht in der Problemspezifikation enthalten und müssen geeignet gewählt werden. Wir orientieren uns dabei an der Wahl von John [25] und verwenden $v(0,y,t) = v(2.2,y,t)$. Hierzu gibt es zum Beispiel die in [42], Kapitel 3.5 beschriebene Alternative, Randbedingungen für Γ_3 aus der Strömung zu generieren.
Auf Grund der Randbedingungen (8.5) und (8.6) handelt es sich um eine laminare Strömung. Für die Reynoldszahl gilt $Re(t) = 100$ bzw. $0 \le Re(t) \le 100$. Es wurde eine Reihe von Benchmark-Rechnungen für dieses Problem durchgeführt, so in [36] und auch in [25], um

Referenzwerte für numerische Verfahren zu erzeugen.

Dabei sind drei Größen von Interesse. Die Erste ist der drag coefficient oder Widerstandsbeiwert, welcher mit der Variablen c_D bezeichnet wird. Er ist eine dimensionslose Größe, die den Widerstand angibt, dem ein Körper in einer Strömung ausgesetzt ist, oder anders formuliert: Er ist ein Maß dafür, wie schwer es ist, den Körper in der Strömung auf der gleichen x-Koordinaten zu halten.

Die laminare Grenzschicht Um den drag coefficient zu verstehen, muss man sich das Verhalten des Fluids im Bereich der so genannten „laminaren Grenzschicht“ [1] verdeutlichen.
Da die Geschwindigkeit an festen Wänden gleich Null ist, muss die Abnahme der Geschwindigkeit insbesondere für große Reynoldszahlen fast völlig in einer dünnen Grenzschicht an den Wänden erfolgen. Dadurch treten in diesem Bereich große Geschwindigkeitsgradienten auf, wobei der Übergang zwischen der laminaren Strömung in der Grenzschicht und der sie umgebenden Grundströmung stetig verläuft.
In dieser dünnen Grenzschicht verläuft die Strömung im Wesentlichen parallel zu der umströmten Oberfläche. Das bedeutet, dass

$$\begin{aligned} & v_1\hat{n}_1 + v_2\hat{n}_2 = \hat{n}\cdot v \approx 0 && (8.7)\\ \Leftrightarrow\quad & v_2 \approx -\frac{\hat{n}_1}{\hat{n}_2}v_1 && (8.8)\end{aligned}$$

in der laminaren Grenzschicht gilt, wenn wir den Normalenvektor auf S, welcher in Ω hinein orientiert ist, mit $\hat{n}$ bezeichnen.
Der c_D-Wert errechnet sich laut [36] mit Hilfe der folgenden Formel:

$$c_D = \frac{2F_D}{d\rho V_{\max}^2}\ .$$

Dabei steht F_D für die drag force, welche mittels des folgenden Wegintegrals berechnet wird:

$$F_D = \int_S \rho\nu\frac{\partial v_{ts}}{\partial\hat{n}}\hat{n}_2 - p\hat{n}_1\, dS\ .$$

Der lift coefficient oder Auftriebsbeiwert ist analog zum drag coefficient ein Maß dafür, wie schwer es ist, den Gegenstand fixiert in der y-Richtung zu halten. Da der Zylinder nicht mittig in den Kanal eingefügt wurde, werden im Laufe der Rechnung Kräfte auf ihn wirken, die Druck in Richtung der y-Achse ausüben. Der c_L-Wert wird wie folgt berechnet:

$$c_L = \frac{2F_L}{d\rho V_{\max}^2}$$

mit

$$F_L = -\int_S \rho\nu\frac{\partial v_{ts}}{\partial\hat{n}}\hat{n}_1 + p\hat{n}_2\, dS\ .$$

Diese beiden Werte sind von großem praktischem Interesse. Der drag coefficient dient z.B. zur Optimierung der Form eines Fahrzeugs in Bezug auf einen möglichst geringen Luftwiderstand,

---

[1] Zum Begriff laminare Grenzschicht siehe z.B. [28], S.165.

was zur Reduktion des Kraftstoffverbrauchs führt.
Die letzte Benchmarkgröße ist die Differenz des Drucks zwischen zwei gegenüberliegenden Punkten auf dem Rand des Zylinders:

$$P_{\text{diff}} = p(x, y, 8) - p(x, y, 8)$$

Da sich alle Benchmarkgrößen auf Werte am Rand S beziehen, ist dieses Problem als besonders schwierig für das in dieser Dissertation entwickelte Splittingverfahren anzusehen. Aus den vorangegangen Betrachtungen wissen wir, dass das Splitting hier oft mit größeren Fehlern zu kämpfen hat als im Rest des Gebietes Ω. Besonders die Bedeutung des Drucks, dessen Randwerte bei einem Splitting am Rand rekonstruiert werden müssen, ist hier eine größere Herausforderung.

Darüber hinaus ist nicht klar, wie groß die Auswirkungen der Approximation des Kreises S durch einen Polygonzug sind. Zur Berechnung des drag und lift coefficient über die oben angegebenen Wegintegrale existieren jedoch einige Alternativen, die zu einer besseren Genauigkeit und größeren Unabhängigkeit von der approximierten Geometrie von S führen. Hierzu werden wir die Wegintegrale in Flächenintegrale umwandeln. Der hier vorgestellte Ansatz wurde erstmals in dem Artikel [26] veröffentlicht, wo er für die stationäre Navier-Stokes-Gleichung eingesetzt wurde. Für die Anwendung auf die instationäre Navier-Stokes-Gleichung wird dazu die Gleichung (2.2)

$$\frac{\partial}{\partial t} v - \nu \nabla^2 v + (v \cdot \nabla) v + \nabla p = 0$$

mit $f = 0$ verwendet, mit einer Funktion $u \in (H_1(\Omega))^2$ multipliziert und anschließend über Ω integriert.

$$\begin{aligned}
0 &= \int_\Omega \frac{\partial}{\partial t} v \cdot u - \nu \nabla^2 v \cdot u + (v \cdot \nabla) v \cdot u + \nabla p \cdot u \, d\Omega \\
&= \int_\Omega -\nu \begin{pmatrix} v_{1_{xx}} + v_{1_{yy}} \\ v_{2_{xx}} + v_{2_{yy}} \end{pmatrix} \cdot u + \nabla p \cdot u \, d\Omega + \int_\Omega (v \cdot \nabla) v \cdot u + \frac{\partial}{\partial t} v \cdot u \, d\Omega \\
&= \int_\Omega -\nu [(v_{1_{xx}} + v_{1_{yy}}) u_1 + (v_{2_{xx}} + v_{2_{yy}}) u_2] + p_x u_1 + p_y u_2 \, d\Omega + \int_\Omega (v \cdot \nabla) v \cdot u + \frac{\partial}{\partial t} v \cdot u \, d\Omega \\
&= \int_{\partial\Omega} -\nu [n \cdot (\nabla v_1) u_1 + n \cdot (\nabla v_2) u_2] + p u \cdot n \, ds \\
&+ \int_\Omega \nu \underbrace{\nabla v : \nabla u}_{=\text{trace}(\nabla v \cdot \nabla u^t)} + (v \cdot \nabla) v \cdot u - p (\nabla \cdot u) + \frac{\partial}{\partial t} v \cdot u \, d\Omega
\end{aligned}$$

Nun bringt man das Randintegral auf die andere Seite.

$$\Leftrightarrow \int_{\partial\Omega} \nu [n \cdot (\nabla v_1) u_1 + n \cdot (\nabla v_2) u_2] - p (u \cdot n) \, ds \tag{8.9}$$

$$= \int_\Omega \nu \nabla v : \nabla u + (v \cdot \nabla) v \cdot u - p (\nabla \cdot u) \, d\Omega \tag{8.10}$$

Seien nun $u_d, u_l \in (H_1(\Omega))^2$ mit $u_d|_S = (1,0)^t$ und $u_l|_S = (0,1)^t$, wobei der Rand des Zylinders mit S bezeichnet wird. Auf allen anderen Rändern verschwinden die Funktionen. Setzt man nun u_d bzw. u_l in (8.9) ein und multipliziert das mit der gewünschten Konstante $-\frac{2}{L\,U_{\max}^2}$, erhält man:

$$\begin{aligned} & -\frac{2}{L\,U_{\max}^2}\int_S \nu[n\cdot\nabla v_1] - pn_1\,ds && (8.11)\\ = & -\frac{2}{L\,U_{\max}^2}\int_\Omega \frac{\partial}{\partial t}v\cdot u_d + \nu\nabla v : \nabla u_d + (v\cdot\nabla)v\cdot u_d - p(\nabla\cdot u_d)\,d\Omega && (8.12)\end{aligned}$$

$$\begin{aligned} & -\frac{2}{L\,U_{\max}^2}\int_S \nu[n\cdot\nabla v_2] - pn_2\,ds && (8.13)\\ = & -\frac{2}{L\,U_{\max}^2}\int_\Omega \frac{\partial}{\partial t}v\cdot u_l + \nu\nabla v : \nabla u_l + (v\cdot\nabla)v\cdot u_l - p(\nabla\cdot u_l)\,d\Omega\,. && (8.14)\end{aligned}$$

Nun verwenden wir (8.7), was uns folgende Umformung erlaubt:

$$\begin{aligned} F_D &= \int_S \nu\frac{\partial v_{ts}}{\partial\hat n}\hat n_2 - p\hat n_1\,dS = \int_S \nu\frac{\partial(v_1\hat n_2 - v_2\hat n_1)}{\partial\hat n}\hat n_2 - p\hat n_1\,dS\\ &= \int_S \nu\frac{\partial(v_1\hat n_2 + \frac{\hat n_1^2}{\hat n_2}v_1)}{\partial\hat n}\hat n_2 - p\hat n_1\,dS = \int_S \nu\frac{\partial v_1}{\partial\hat n} - p\hat n_1\,dS\\ &= \int_S \nu\hat n\cdot\nabla v_1 - p\hat n_1\,dS = -\int_S \nu n\cdot\nabla v_1 - pn_1\,dS\,,\end{aligned}$$

wobei $n = -\hat n$ und $\rho = 1$ gilt. Analog geht man mit $v_1 = -\frac{\hat n_2}{\hat n_1}v_2$ für F_L vor. Abschließend lassen sich nach dem Einsetzen der Werte die folgenden Formeln für den drag und lift coefficient angeben:

$$\begin{aligned} c_d &= -20\int_\Omega \frac{\partial}{\partial t}v\cdot u_d + \nu\nabla v : \nabla u_d + (v\cdot\nabla)v\cdot u_d - p(\nabla\cdot u_d)\,d\Omega && (8.15)\\ c_l &= -20\int_\Omega \frac{\partial}{\partial t}v\cdot u_l + \nu\nabla v : \nabla u_l + (v\cdot\nabla)v\cdot u_l - p(\nabla\cdot u_l)\,d\Omega\,. && (8.16)\end{aligned}$$

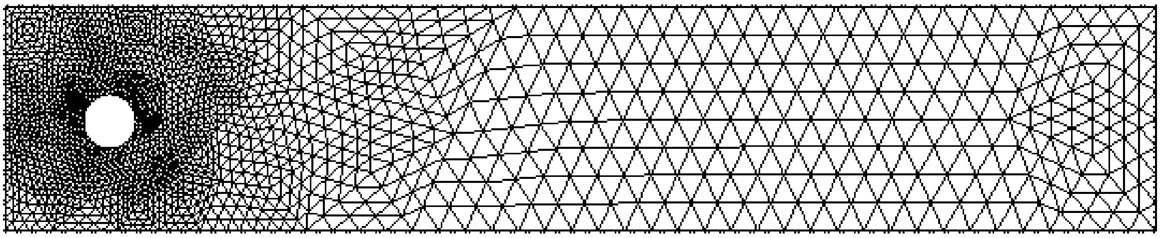


Abbildung 8.14: Startgitter mit 1730 Knoten

8.2.1 2D-3 (unsteady)

Die Benchmarkgrößen für den Fall 2D-3 (unsteady) aus [36] beziehen sich auf $c_{D,\max} = \max_{t\in[0,8]}\{c_D(t)\}$ bzw. $c_{L,\max} = \max_{t\in[0,8]}\{c_L(t)\}$ und den Zeitpunkt, zu dem das Maximum angenommen wird. Dabei gilt $c_{D,\max}^{\text{ref}} = [2.93, 2.97]$ und $c_{L,\max}^{\text{ref}} = [0.47, 0.49]$.

Zusätzlich zu diesen punktuellen Vergleichswerten stehen zum Vergleich der Benchmarkgrößen die Referenzwerte aus [25], Tabelle IV zur Verfügung, welche mit einer Zeitschrittweite von $\triangle t = 0.00125$ auf einem Taylor-Hood-Element des Typs P2/P1 mit 399616 Freiheitsgraden in v und 50240 Freiheitsgraden in p berechnet wurden. Da hier kein Splittingverfahren verwendet wurde, führt das auf ein Gleichungssystem mit 449856 Unbekannten. Als Zeitschrittverfahren wird in [25] das zum Beispiel bei [42], Kapitel 3.2.1 beschriebene Fractional-step-θ-Schema verwendet. Dabei wird in Makroschritten vorgegangen, die jeweils drei Unterschritte umfassen. Wie bei [42], S.164ff nachgelesen werden kann, entspricht der Aufwand für einen Makroschritt drei Schritten klassischer Verfahren, wie z.B. der hier verwendeten BDF-Verfahren. Um die Vergleichbarkeit zu erleichtern, sind daher in Tabelle 8.7 zwei Zeitschrittweiten angegeben, zum einen die des Makroschritts und zum anderen die der Unterschritte.

Zur Berechnung der in Tabelle 8.7 angegebenen Werte wurde ein Splittingverfahren mit einer impliziten Behandlung des nichtlinearen Terms eingesetzt. Hierzu wurde eine Diskretisierung mit 139344 Freiheitsgraden in v und 35048 Freiheitsgraden in p verwendet. Diese entsteht, indem alle Elemente aus der in Abbildung 8.14 gezeigten Starttriangulierung verfeinert werden, deren Abstand kleiner als 0.075 von $(0.2, 0.2)$ ist. Anschließend wird das daraus entstandene Gitter als Ganzes regulär verfeinert. Dieses Gitter bildet die Grundlage für V_p.

Splitting Version	$\triangle t$	$t(c_{d,\max})$	$c_{d.\max}$	$t(c_{l,\max})$	$c_{l,\max}$	$p_{\text{diff}}(8s)$
N	1/128	3.9296875	2.9685242	5.7109375	0.59604227	-0.11100459
N	1/400	3.9325	2.9505269	5.6925	0.50337405	-0.11077072
P_l	1/400	3.93	2.9509076	5.695	0.49461359	-0.11086049
N	1/500	3.936	2.9485705	5.69	0.49883271	-0.11107477
P_l	1/1000	3.934	2.9478232	5.688	0.49117886	-0.11053843
P_l	1/1200	3.93	2.9465880	5.686667	0.49084030	-0.11048193
P_l	1/1600	3.93	2.9450870	5.685625	0.49031280	-0.11041277
John:04	$\frac{1/800(M.)}{1/2400(U.)}$	3.93625	2.9509216	5.6925	0.47811979	-0.11158097

Tabelle 8.7: Die Ergebnisse in den Benchmarkwerten des „Fluss um einen Zylinder“-Problems 2D-3 mit 139344 Unbekannten in v und 35048 in p

Die Tabelle 8.7 zeigt, dass bereits für relativ große Zeitschrittweiten das Benchmarkintervall für den drag coefficient und die SDruckdifferenz erreicht wird.
Darüber hinaus zeigen die Abbildungen 8.15 und 8.16, dass mit ca. einem Drittel der Unbekannten vergleichbare Ergebenisse zu den Resultaten in [25] erzielt werden konnten. Dazu kommt, dass die hier beim Splitting auftretenden Matrizen weniger dicht besetzt sind,

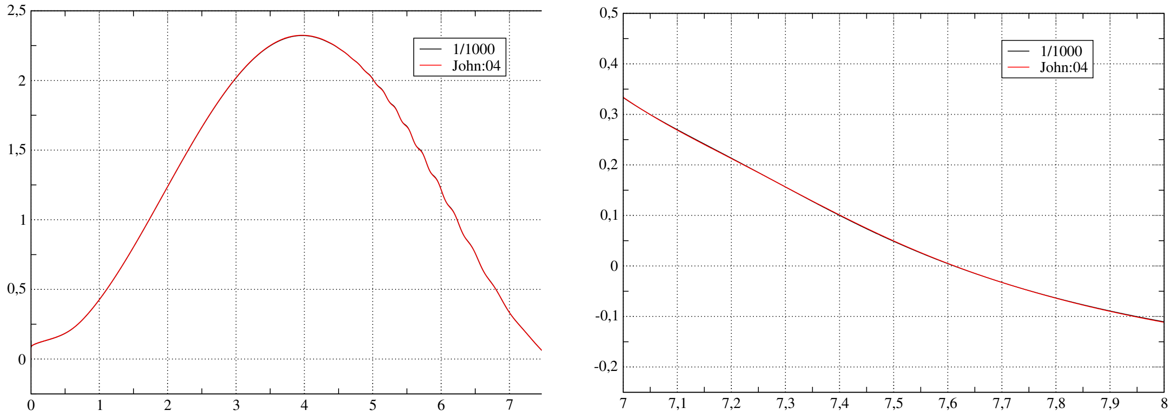


Abbildung 8.15: p_{diff} (2D-3) Splitting $\triangle t = \frac{1}{1000}$; Benchmarkergebnisse aus [25] John:04

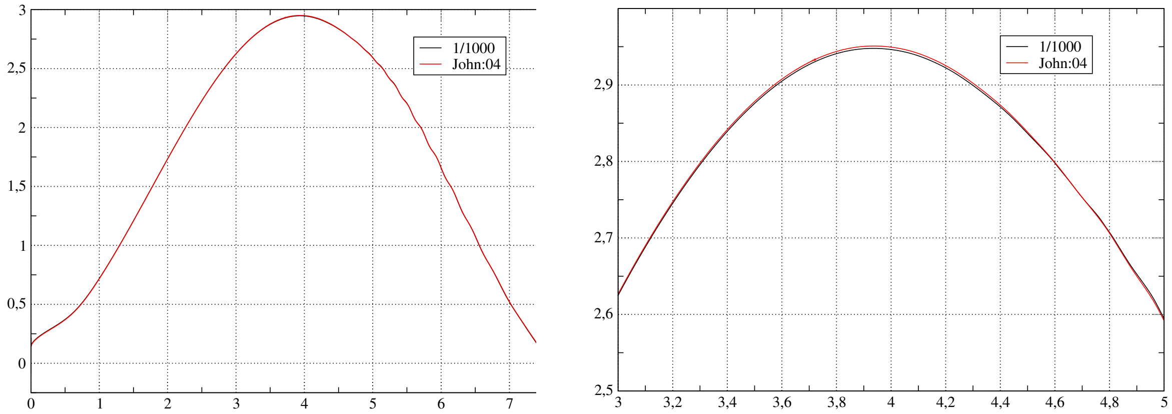


Abbildung 8.16: drag coefficient Splitting $\triangle t = \frac{1}{1000}$; Benchmarkergebnisse aus [25] John:04

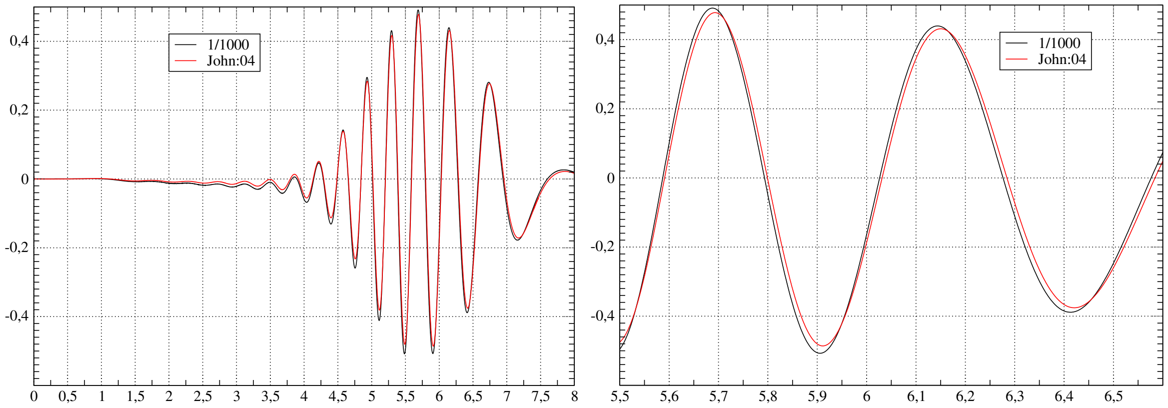


Abbildung 8.17: lift coefficient Splitting $\triangle t = \frac{1}{1000}$; Benchmarkergebnisse aus [25] John:04

da beim impliziten Postp lineare Ansatzfunktionen verwendet wurden, während in [25] auf quadratische Basisfunktionen zurückgegriffen wird. Dazu kommt, dass die Lösung zweier kleinerer Gleichungssysteme billiger ist, als die es nicht entkoppelten Problems.

Der lift coeffizient ist wie auch zum Beispiel in [36] und [25] nachgelesen werden kann der problematischte Benchmarkwert. Hier werden auch vom Splittingverfahren kleiner Zeitschrittweiten benötigt um den oberen Rand des Benchmarkintervalls mit 0.49 zu erreichen.

Darüber hinaus dienen die Ergebnisse in der Tabelle 8.7 erneut zur Gegenüberstellung der Resultate mit implizitem (P_I) und ohne Postprocessing (N). Wie man bei einer Zeitschrittweite von $\triangle t = 1/400$ und dem problematischen Benchmarkwert c_l sieht ist auch bei diesem Problem das Splitting mit Postprozessing bei der gleichen Anzahl an Unbekannten genauer, als das ohne.

Die folgenden Abildungen 8.18 und 8.19 zeigen die Strömung und den Druck in charakteristischen Phasen der Simulation. Bei der Wahl der Zeitpunkte wurde die vergleichbare Abbildung in [25] als Vorbild gewählt.

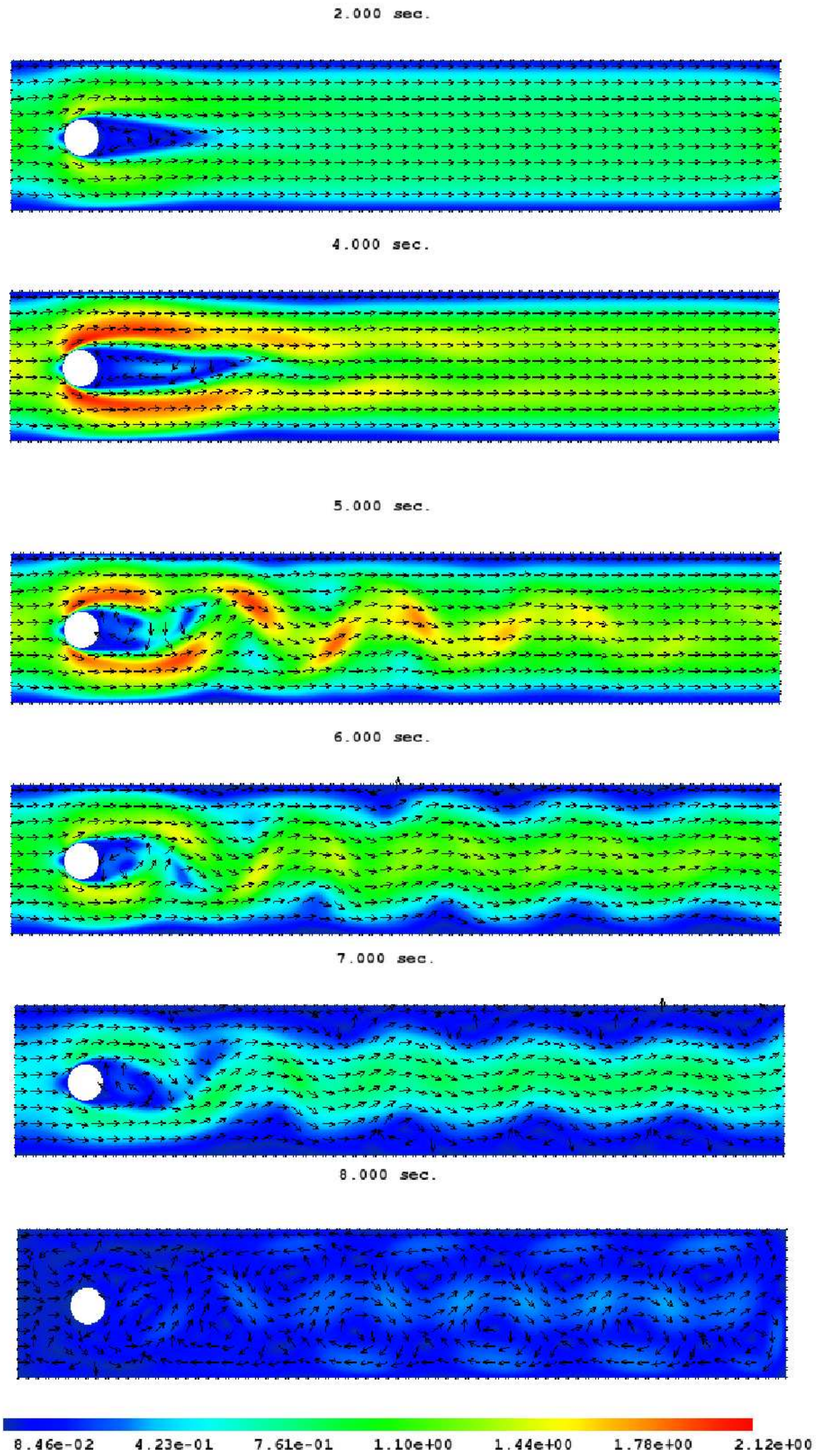


Abbildung 8.18: Die Strömung bei t=2s,4s,5s,6s,7s und 8s

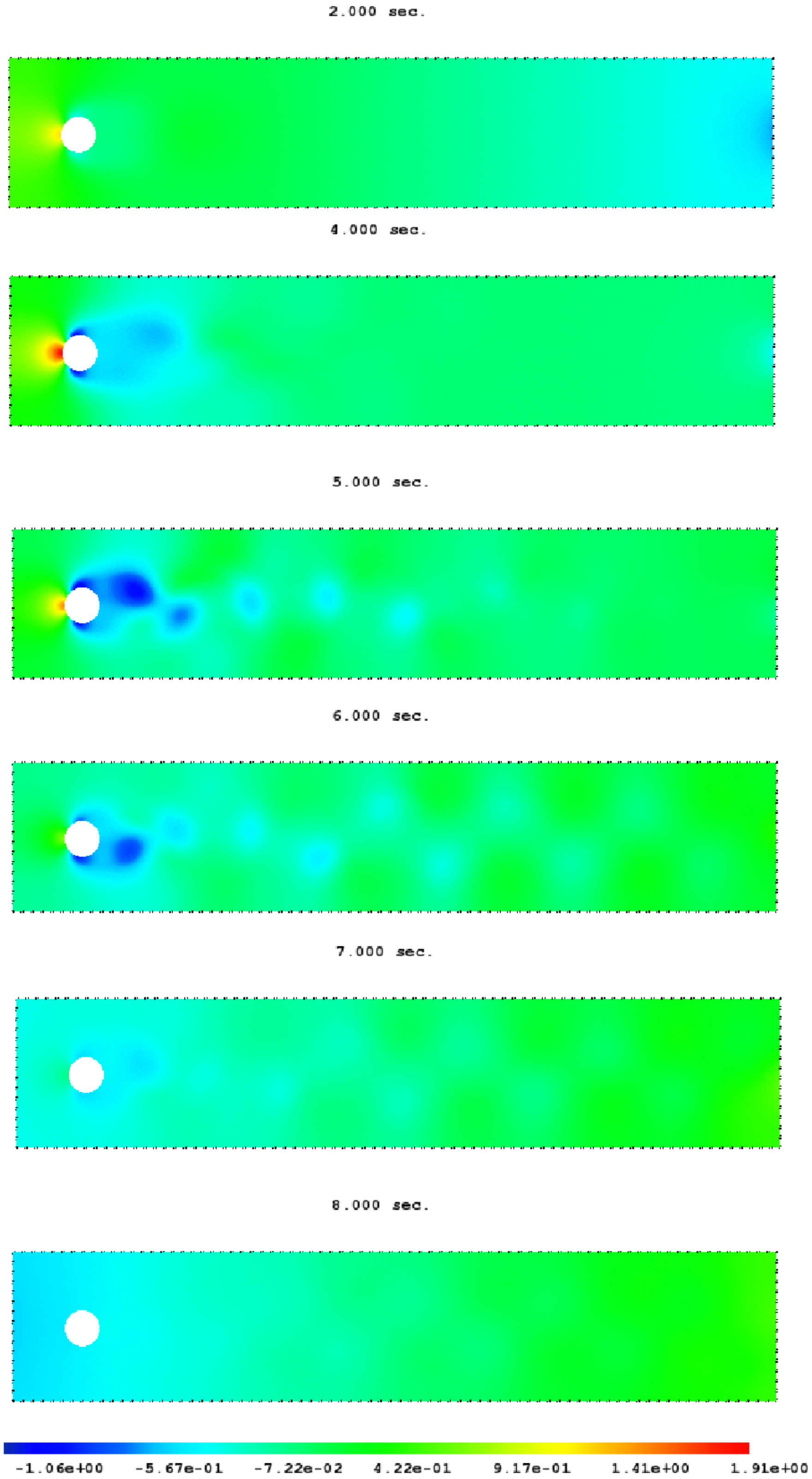


Abbildung 8.19: Der Druck bei t=2s,4s,5s,6s,7s und 8s

8.2.2 2D-2 (unsteady,periodic)

Um einen unstetigen Start zu vermeiden wird der Ein- und Ausströmende Fuild durch

$$v(0,y,t) = v(2.2,y,t) = \begin{cases} 0.41^{-2}\sin(5\pi t)(6y(0.41-y),0) & \text{, für } t \in [0,0.1[\\ 0.41^{-2}(6y(0.41-y),0) & \text{, für } t \in [0.1,8] \end{cases}$$

simuliert. Wie am an der Abbildung 8.20 und 8.21 erkennen kann setzt das periodische Verhalten der Funktion zwischen 2s und 4s ein und ist ab dann stabil.
Die Abbildung 8.20 zeigt der Verlauf des drag coefficient erst nach einer Sekunde, da vor dem erreichen des periodischen Verhaltens vergleichsweise große Werte auftreten, die sonst die Darstellung erschweren würden. Die Tabelle 8.8 zeigt die mit einem impliziten Postprocessing berechneten Werte für das Testproblem 2D-2.

$\triangle t$	$c_{d.\max}$	$c_{l,\max}$	Strouhal Zahl
1/250	3.2565	1.0114	0.2906
1/500	3.2404	1.0005	0.2941
1/1000	3.2298	0.9944	0.2959
Untere Grenze	3.2200	0.9900	0.2950
Obere Grenze	3.2400	1.0100	0.3050

Tabelle 8.8: Die Ergebnisse in den Benchmarkwerten des „Fluss um einen Zylinder“-Problems 2D-2 mit 139344 Unbekannte in v und 35048 in p

Die Grenzwerte für die Benchmarkgrößen entstammen [36], wobei für Berechnungen des drag und des lift coefficient die gleichen Ansätze wie schon im Abschnitt 8.2.1 verwendet werden.
Die Strouhal Zahl ist definiert durch

$$St = \frac{fd}{\bar{V}}$$

Wobei f ist die Frequenz der Änderung des Strömungsabrisses ist.

$$\bar{V} := \frac{2v(0,0.41/2,t)}{3}$$

Was im Fall von 2D-2 mit Hilfe sich mit Hilfe der Periode $\mathcal{P}$ sich wie folgt ausdrücken lässt

$$St = 0.1 \cdot f = \frac{1}{10\mathcal{P}}\ .$$

$\mathcal{P}$ lässt sich numerisch einfach berechnen durch den Abstand zwischen zwei Vorzeichenwechsel von $c_l(t)$ mal zwei ausdrücken. Für die in der Tabelle 8.8 angegebene Strouhal Zahl wurde dabei die letzte Periode in dem Zeitintervall $[0,8]$ verwendet.

Wie man sieht erreicht das Splittingverfahren mit einem impliziten Postprocessing hier erneut gute Werte. In Bezug auf den schwierigen Benchmarkwert c_l schneidet der Algorithmus hier besser ab als bei der letzten Variante 2D-3.

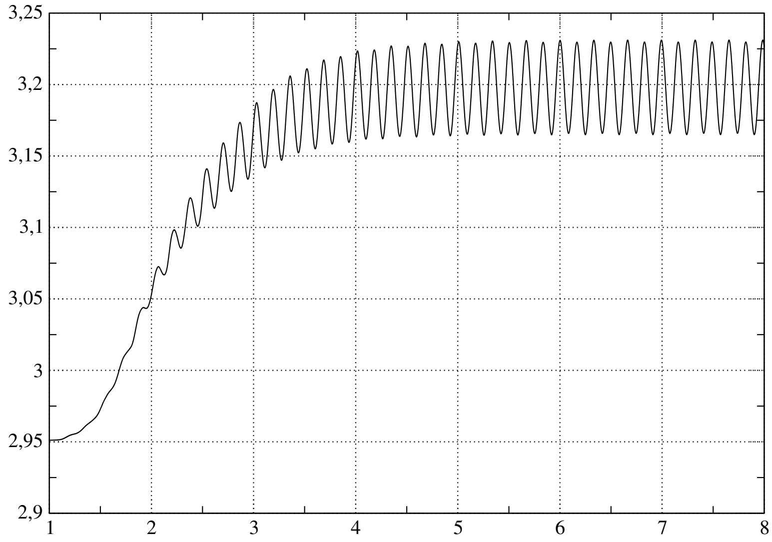


Abbildung 8.20: drag coefficient (2D-2) $\triangle t = 1/1000$

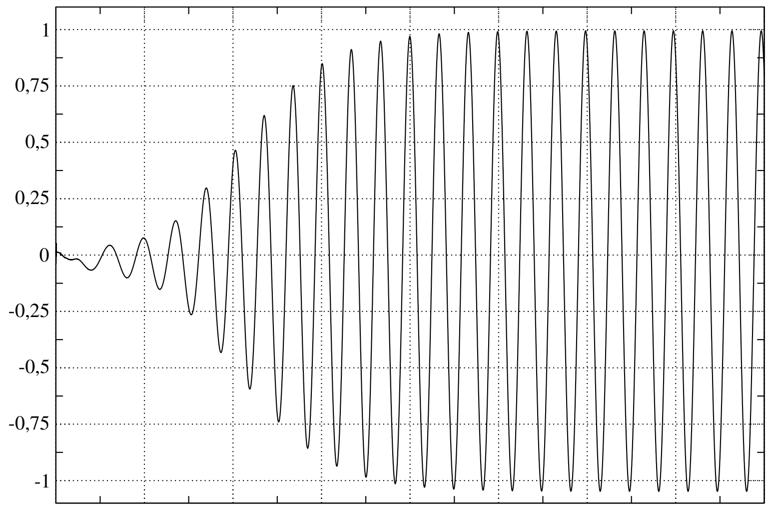


Abbildung 8.21: lift coefficient (2D-2) $\triangle t = 1/1000$

8.2.3 Adaptivität in der Zeit

An beiden Testproblemen 2D-2 und 2D-3 wird nun das in Kapitel 7.1 beschriebene adaptive Verfahren in der Zeit getestet. Dabei wird erneut die robuste Splittingvariante mit implizitem Postprocessing verwendet. Die Abschätzung des Zeitdiskretisierungsfehlers geschieht daher, wie in Abschnitt 7.3 dargelegt wurde, im Raum W_H. Da das im letzten Abschnitt verwendete Gitter hierfür zu grob ist, wird dies nun global regulär verfeinert. Damit hat der Raum W_H nun 139344 Freiheitsgrade in v und 35048 in p, der Raum W_h entsprechend 139344 in p und 555680 für die Geschwindigkeit.

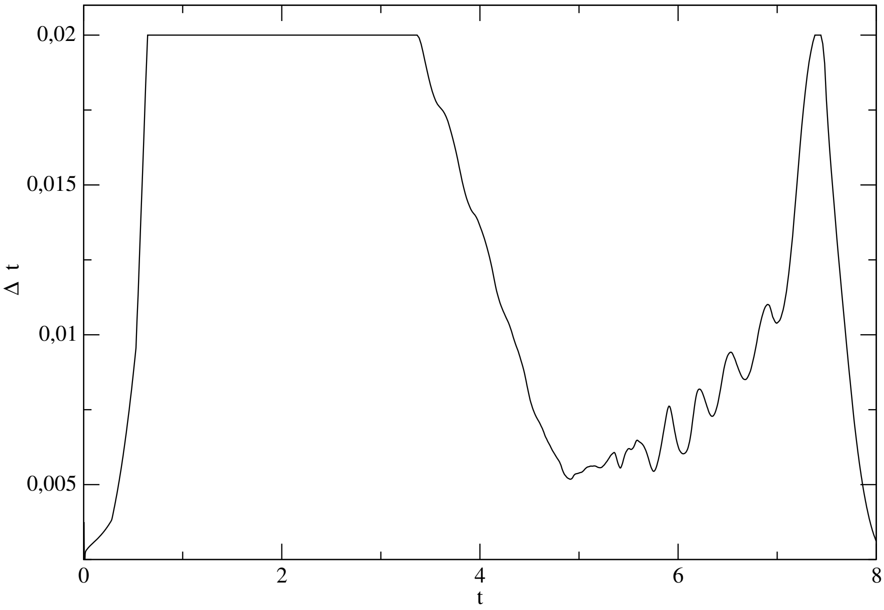


Abbildung 8.22: 2D-3 : $\triangle t$

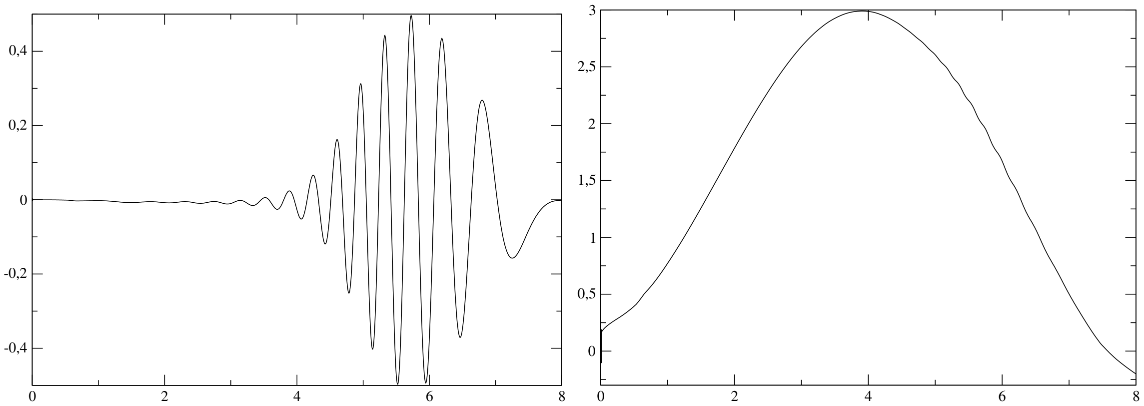


Abbildung 8.23: 2D-3 : lift coefficient (links) und drag coefficient (rechts)

Wie an Hand der Abbildung 8.18 deutlich wird, ist bei dem Testproblem 2D-3 die

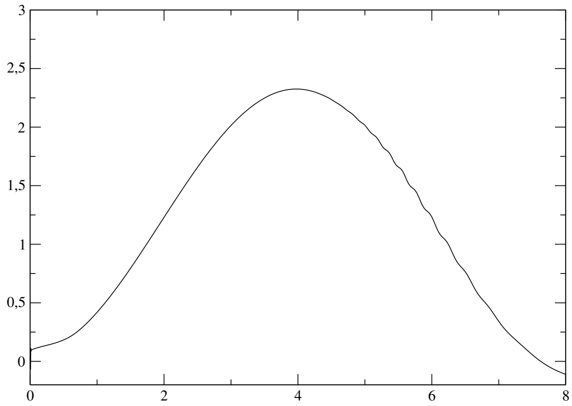


Abbildung 8.24: 2D-3 : p_{diff}

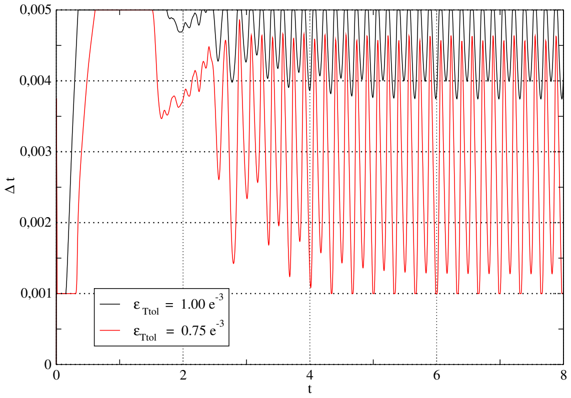


Abbildung 8.25: 2D-2 : $\triangle t$

Strömung sehr lange von sehr einfacher Struktur, bis etwa zum Strömungsabriss. Der Algorithmus zur adaptiven Zeitschrittweitensteuerung mit $\varepsilon_{Ttol} = 1 \cdot 10^{-3}$ erkennt diese Charakteristika der Strömung und hält, vgl. Abb. 8.23, bis kurz vor dem Strömungsabriss an der in Kapitel 7.1.1 erwähnten maximalen Sicherheitsschranke, hier $\triangle t \leq 1/50$, für $\triangle t$ als Zeitschrittweite fest. In der kritischen Phase der wechselnden Umströmung des Zylinders wird die Zeitschrittweite stark reduziert. Die Abbildungen 8.23 und 8.24 zeigen die mit Hilfe dieses Verfahrens ermittelten Benchmarkwerte. Wie man sieht, entsprechen die Plots denen aus Abschnitt 8.2.1 mit fester Zeitschrittweite.

ε_{Ttol}	durchschnittliche Zeitschrittweite	$c_{d.\max}$	$c_{l,\max}$	Strouhal Zahl
$1.0 \cdot 10^{-3}$	$0.004246 \approx 1/235$	3.2439	1.0104	0.29811
$7.5 \cdot 10^{-4}$	$0.002479 \approx 1/400$	3.2285	1.0031	0.30022
Untere Grenze	-	3.2200	0.9900	0.2950
Obere Grenze	-	3.2400	1.0100	0.3050

Tabelle 8.9: Die Ergebnisse in den Benchmarkwerten des „Fluss um einen Zylinder“-Problems 2D-2 mit 555680 Unbekannten in v und 139344 in p

Der Parameter ε_{Ttol} wurde für das Testproblem 2D-2 variiert und zum einen mit $\varepsilon_{Ttol} = 1 \cdot 10^{-3}$, zum anderen mit $\varepsilon_{Ttol} = 0.75 \cdot 10^{-3}$ gestartet. Die Sicherheitsschranken für die adaptive Zeitschrittweitenbestimmung waren, wie der Abbildung 8.25 entnommen werden kann, $1/1000 \leq \triangle t \leq 1/200$. Die Änderung der Zeitschrittweite wird in der Abbildung 8.25 dargestellt. Wie man der Tabelle 8.9 entnehmen kann, konnten auf diese Weise mit einer recht großen durchschnittlichen Zeitschrittweite Ergebnisse innerhalb der vorgesehenen Intervalle für die Benchmarkwerte erreicht werden.

9 Fazit

Mit dem in dieser Arbeit entwickelten Verfahren steht mit Hilfe der adaptiven Schrittweitensteuerung eine sehr effiziente Methode zur Lösung der zeitabhängigen Navier-Stokes-Gleichung zur Verfügung. Insbesondere die implizite Variante des Splittings konnte als äußerst robuste Methode für große Schrittweiten und Reynoldszahlen eingesetzt werden.

Die explizite Behandlung des nichtlinearen Terms bietet für Probleme mit niedrigeren Reynoldszahlen die Möglichkeit, ein Verfahren zu verwenden, bei dem ausschließlich symmetrische Matrizen entstehen. Das erlaubt den Einsatz verschiedenster effektiver Löser für die auftretenden Gleichungssysteme. Wird das Verfahren mit einer konstanten Schrittweite verwendet, ist in diesem Fall auch die Galerkinmatrix in jedem Zeitschritt gleich, wodurch sich besonders effektive Möglichkeiten zur Vorkonditionierung bieten.

Zu den guten Ergebnissen trägt sowohl der Einsatz einer Gradientenrekonstruktion mit hoher Genauigkeit als auch die verbesserte Druckapproximation bei. Letztere wird durch an die Problemstellung angepasste Neumann-Randbedingungen erreicht. Insbesondere die Verfahren zur Gradientenrekonstruktion ermöglichen es, den Genauigkeitsverlust bei der Verwendung der Gradienten von v und p zu minimieren.

Der Fehlerindikator auf der Basis der Gradientenrekonstruktion ist in der Lage, eine adaptive Gitterverfeinerung vorzunehmen. Zusätzliche Fehlerquellen entstehen bei der Anpassung der Gitter in Zusammenhang mit dem Druck durch die mangelhafte Divergenzfreiheit der interpolierten Werte.

Einen besonderen Fortschritt gegenüber anderen Splitting- bzw. Projektionsverfahren stellt die durch die vorgestellten Strategien zum Postprocessing bzw. Preconditioning erreichte höhere Ordnung dar. Das Verfahren weist Reduktionsraten oberhalb der zweiten Ordnung in der Zeit auf. Dies geschieht in allen numerischen Tests ohne zusätzliche CPU-Kosten und macht das vorgestellte Splitting zu einem sehr attraktiven Verfahren. Wesentlich hierfür ist, dass auch schon auf dem gröbsten verwendeten Gitter das Splitting konvergiert.

Mit Hilfe der Analysen aus den Kapiteln 5.1 und 6.1 kann das Verhalten des Splittings mit einer der Postprocessingvarianten erklärt und in einigen Fällen vorhergesagt werden. Die optimalen Voraussetzungen werden von Problemen mit homogenen Null-Dirichlet-Randbedingungen für die Geschwindigkeit erfüllt, welche, wie nach den Ausführungen in Kapitel 2.4 klar wurde, zugleich auch die häufigsten Randbedingungen bei praktischen Fragestellungen sind.

Die Ergebnisse aus Kapitel 8 zeigen, dass das Splittingverfahren nicht auf konstruierte, analytische Beispiele beschränkt ist, sondern auch sehr gute Resultate auf den Standardproblemen der Strömungsmechanik liefert. Es ist in der Lage, Singularitäten, wie sie beim Driven Cavity Problem auftreten, zu verkraften und kann sich mittels der adaptiven Gitterverfeinerung an solche Probleme anpassen.

Das „Fluss um einen Zylinder"-Problem zeigt, dass auch komplexe, nicht konvexe Geometrien kein Problem sowohl für das Splittingverfahren als auch das darin verwendete Gradientenrekonstruktionsverfahren darstellen. Im Gegenteil, gerade bei diesen anwendungsorientierten Fragestellungen zeigt das Splitting seine Stärken und liefert für das Testproblem 2D-3 und besonders für das Testproblem 2D-2 Ergebnisse, die sich schon bei vergleichsweise großen Zeitschrittweiten innerhalb der Benchmarkintervalle befinden. An Hand dieser Testprobleme konnten auch die Verfahren zur adaptiven Zeitschrittweitensteuerung erfolgreich getestet werden und lieferten Ergebnisse innerhalb der vorgegebenen Intervalle.

Interessant für die weitere Forschung sind die Fragestellungen, ob und wie eine divergenzfreie Interpolation die Resultate des Splittings auf adaptiv verfeinerten Gittern verbessern kann und besonders inwieweit sich der positive Effekt der Zwei-Gitter-Nachbereitung durch weitere Gitter steigern lässt.

Abbildungsverzeichnis

Tabellenverzeichnis

Literaturverzeichnis

[1] Almgen, Ann S. ; Bell, John B. ; Colella, Phillip ; Howell, Louis H. ; Welcome, Michael L.: A Conservative Adaptive Projection Method for the Variable Density Incompressible Navier-Stokes Equations. In: *J. Comp. Phys.* 142 (1998), S. 1–46

[2] Alt, Hans W.: *Lineare Funktionalanalysis*. 2. Berlin : Springer, 1992

[3] Bey, Jürgen: *Finite-Volumen- und Mehrgitter-Verfahren für elliptische Randwertprobleme*. Stuttgart : Teubner, 1998

[4] Botella, O.: On the solution of the Navier-Stokes equations using Chebyshev projection schemes with third-order accuracy in time. In: *Computers & Fluids* 26 (1997), S. 107–116

[5] Botella, O. ; Peyret, R.: Benchmark spectral results on the lid-driven cavity flow. In: *Computers & Fluids* 27 (1998), S. 421–433

[6] Braess, Dietrich: *Finite Elemente*. Berlin : Springer, 1997

[7] Brenner, Susanne C. ; Scott, L. R.: *The Mathematical Theory of Finite Element Methods*. New York : Springer, 1994

[8] Dahmen, Wolfgang ; Müller, Siegfried ; Schlinkmann, Thomas: On an Adaptive Multigrid Solver for Convection-Dominated Problems. In: *SIAM J. Sci. Comput.* 23 (2001), S. 781–804

[9] Davis, T. A.: *UMFPACK - Version 4.4 User Guide*. http://www.cise.ufl.edu/research/sparse/umfpack/: Dept. of Computer and Information Science and Engineering - Univ. of Florida, 2005

[10] Davis, T. A.: UMFPACK - an unsymmetric-pattern multifrontal method with a column pre-ordering strategy. In: *ACM Trans. Math. Software* 30 (204), Nr. 2, S. 196–199

[11] Deuflhard, Peter ; Bornemann, Folkmar: *Numerische Mathematik II*. Berlin : De Gruyter, 1994

[12] Deuflhard, Peter ; Hohmann, Andreas: *Numerische Mathematik I*. Berlin : De Gruyter, 1993

[13] Eriksson, K. ; Estep, D. ; Hansbo, P. ; Johnson, C.: *Computational Differential Equations*. Cambridge : Cambridge University Press, 1996

[14] Feynman, R. P. ; Leighton, R. B. ; Sands, M.: *Vorlesungen über Physik Bd.II*. München : Oldenbourg Verlag GmbH, 1991

[15] Ghia, U. ; Ghia, K. ; Shin, C.T.: High-Re solutions of Navier-Stokes equations and a multigrid method. In: *Journal of Comp. Phys.* 48 (1982), S. 387–411

[16] Greenbaum, Anne: *Iterative Methods for Solving Linear Systems.* Philadelphia : SIAM, 1997

[17] Greiner, W. ; H.Stock: *Hydrodynamik.* Frankfurt am Main : Verlag Harri Deutsch, 1991

[18] Grigorieff, R.D.: Stability of Multistep-Methods on Variable Grids. In: *Numerische Mathematik* 42 (1983), S. 359–377

[19] Hackbusch, Wolfgang: *Theorie und Numerik elliptischer Differentialgleichungen.* Stuttgart : Teubner, 1986

[20] Hackbusch, Wolfgang: *Iterative Lösung großer schwachbesetzter Gleichungssysteme.* Stuttgart : Teubner, 1991

[21] Haschke, H. ; Heinrichs, W.: Splitting techniques with staggered grids for the Navier-Stokes equations in the 2d case . In: *Journal of Comp. Phys.* 168 (2001), S. 131–154

[22] Haschke, Heike: *Splitting-Techniken zur spektralen Approximation der Navier-Stokes- und Boussinesq-Gleichungen*, Universität-GH Essen, Diss., November 2001

[23] Heinrichs, Wilhelm: Splitting techniques for the pseudospectral approximation of the unsteady Stokes equations. In: *SIAM J. Numer. Anal.* 30 (1993), S. 19–39

[24] Heinrichs, Wilhelm: Splitting techniques for the unsteady Stokes equations. In: *SIAM J. Numer. Anal.* 35 (1998), S. 1646–1662

[25] John, Volker: Reference values for drag and lift of a two-dimensional time-dependent flow around a cylinder. In: *Internation Journal for Numerical Methods in Fluids* 44 (2004), S. 777–788

[26] John, Volker ; Matthies, Gunar: Higher order finite element discretizations in a benchmark problem for the 3D Navier Stokes equations. In: *Internation Journal for Numerical Methods in Fluid Mechanics* 40 (2002), S. 775–798

[27] Knabner, Peter ; Angermann, Lutz: *Numerik partieller Differentialgleichungen.* Berlin : Springer, 2000

[28] Landau, L. D. ; Lifschitz, E. M.: *Hydrodynamik.* 3nd. Berlin : Akademie-Verlag, 1974

[29] Maday, Y. ; Patera, A.T. ; Rønquist, E.: An operator-integration-factor splitting method for time-dependent problems: application to incompressible fluid flow. In: *J. Sci. Comput.* 5 (1990), S. 263–292

[30] MathWorks: *MATLAB Partial Differential Equation Toolbox User's Guide.* 1995

[31] Morton, K. W.: *Numerical Solution of Convection-Diffusion Problems.* London : Chapman and Hall, 1996

[32] Prohl, Andreas: *Projection und Quasi-Compressibility Methods for Solving the Incompressible Navier-Stokes Equations.* Stuttgart : B.G. Teubner, 1997

[33] Quarteroni, Alfio ; Valli, Alberto: *Numerical Approximation of Partial Differential Equations.* Berlin : Springer, 1994

[34] Rannacher, R.: On Chorin's projection for the incompessible Navier-Stokes equations. In: Heywood, J.G. (Hrsg.) ; Masuda, K. (Hrsg.) ; R.Rautmann (Hrsg.) ; Solonnikov, S.A. (Hrsg.): *The Navier-Stokes Equations II - Theory and Numerical Methods*, Springer-Verlag, 1991 (Lecture Notes in Math.)

[35] Roos, Hans-Görg ; Stynes, Martin ; Tobiska, Lutz: *Numerical Methods for Singularly Perturbed Differential Equations.* Berlin : Springer, 1996

[36] Schäfer, M. ; Turek, S.: The benchmark problem 'flow around a cylinder'. In: (ed.), Hirchel E. (Hrsg.): *Notes on Numerical Fluid Mechanics* Bd. 52, Vieweg Verlag Braunschweig, 1996, S. 547–566

[37] Serrin, J.: The initial value problem for the Navier-Stokes equations. In: *Nonlinear Probl.* 83 (1963), S. 69–98

[38] Sohr, H.: Zur Regularitätstheorie der instationären Gleichungen von Navier-Stokes. In: *Math.Z.* 184 (1983), S. 359–375

[39] Starke, Gerhard: Multilevel Computation of Streamline Diffusion Finite Element Discretizations for Convection-Diffusion Problems. In: Griebel, Michael (Hrsg.) ; Iliev, Oleg P. (Hrsg.) ; Margenov, Svetozar D. (Hrsg.) ; Vassilevski, Panayot S. (Hrsg.): *Large-Scale Scientific Computations of Engineering and Environmental Problems* Bd. 62, Vieweg, 1998, S. 281–289

[40] Strehmel, Karl ; Weiner, Rüdiger: *Numerik gewöhnlicher Differentialgleichungen.* Stuttgart : Teubner, 1995

[41] Thomée, Vidar: *Galerkin Finite Element Methods for Parabolic Problems.* Berlin : Springer, 1997

[42] Turek, Stefan: *Efficient Solvers for Incompressible Flow Problems.* Berlin : Springer, 1999

[43] Verfürth, Rüdiger: *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques.* Chichester-Stuttgart : Wiley-Teubner, 1996

[44] Vogel, H.: *Gerthsen Physik.* Berlin : Springer, 1997

[45] Zienkiewicz, O. C. ; Zhu, J. Z.: The superconvergent patch recovery and a posteriori error estimates. Part I,II. In: *Int. J. Num. Meth. Engrg.* 33 (1992), S. 1331–1364,1365–1382