Parallele Lösung der inkompressiblen Navier–Stokes Gleichungen auf adaptiv verfeinerten Gittern

Zur Theorie und Anwendung der nichtkonformen P_1/P_0–Finite Elemente Diskretisierung

DISSERTATION

zur Erlangung des akademischen Grades
doctor rerum naturalium
(Dr. rer. nat.),

genehmigt durch
die Fakultät für Mathematik
der Otto–von–Guericke–Universität Magdeburg

von Diplommathematiker Volker JOHN

geb. am 18. Oktober 1966
in Bischofswerda

Gutachter: Prof. Dr. Lutz Tobiska
Prof. Dr. Gabriel Wittum
Prof. Dr. Hans–Görg Ross

Eingereicht am: 21.03.1997

Verteidigung am: 25.08.1997

Inhaltsverzeichnis

Kapitel 0

Einleitung

Die Untersuchung inkompressibler Strömungen ist ein wichtiger Zweig der Strömungsmechanik. Das zugrunde liegende Modell sind die inkompressiblen Navier–Stokes Gleichungen. In dieser Arbeit werden Strömungen bei konstanter Temperatur betrachtet, so daß die zu berechnenden physikalischen Größen in den inkompressiblen Navier–Stokes Gleichungen die Geschwindigkeit der Strömung und die Druckverteilung sind. Es werden zwei grundlegende physikalische Erhaltungsgesetze beschrieben: die Erhaltung des linearen Impulses und die Erhaltung der Masse. Die inkompressiblen Navier–Stokes Gleichungen sind heutzutage nicht nur als eigenständiges Problem, sondern in zunehmendem Maße als Basis anderer Modelle von Interesse. Das sind beispielsweise numerische Simulationen von Kristallwachstumsprozessen oder Probleme, bei denen Geschwindigkeit und Druck noch mit anderen Feldern wie Temperatur, Konzentration oder magnetischen Feldern gekoppelt sind. Die (mehrfache) Lösung inkompressibler Navier–Stokes Gleichungen ist ein Kernstück vieler numerischer Verfahren für die genannten Probleme, siehe zum Beispiel Knobloch [Kno96] oder Dorok [Dor95].

Im Rahmen der **numerischen Strömungssimulation (Computational Fluid Dynamics)** werden seit den 60–er Jahren numerische Methoden zum Lösen inkompressibler Navier–Stokes Gleichungen untersucht. In einem ersten Schritt werden die Gleichungen diskretisiert. Das Herzstück einer jeden Diskretisierung stellt die Wahl der diskreten (endlichdimensionalen) Ansatzräume für die Geschwindigkeit und den Druck dar. Die Lösung des diskreten, nichtlinearen algebraischen Gleichungssystem wird durch eine Folge von Lösungen linearer Gleichungssysteme approximiert. Die Lösung großer, schwach besetzter, linearer Probleme ist das Kernstück vieler Algorithmen innerhalb der numerischen Strömungssimulation. Zur Beurteilung eines Verfahrens gibt es drei grundlegende Kriterien. Das ist zum ersten die Genauigkeit der berechneten Lösungen, die wesentlich von der Wahl der diskreten Ansatzräume abhängt. Das zweite Kriterium ist der numerische Aufwand zur Berechnung der Lösungen und das dritte ist die Robustheit der Verfahren, das heißt die Unabhängigkeit ihres Verhaltens vom konkreten Problem. Diese beiden Punkte sind wesentlich an den verwendeten Löser geknüpft. Es ist bekannt, daß es Wechselwirkungen zwischen dem Diskretisierungskonzept und dem verwendeten Löser gibt, daß insbesondere die Wahl eines guten Lösers nicht unabhängig von der zugrunde liegenden Diskretisierung ist.

Im Laufe der letzten Jahrzehnte haben sich die Diskretisierungstechniken stürmisch entwickelt. Der Blick in die Literatur zeigt, daß fast ausschließlich **Finite Elemente Methoden (FEM)** und Finite Volumen Methoden (FVM) zur Diskretisierung der Navier–Stokes Gleichungen genutzt werden. Häufig verwendete Finite Elemente Diskretisierungen sind das konforme P_2/P_1–Element (Taylor–Hood Element), das nichtkonforme P_1/P_0–Element, das nichtkonforme rotierte Q_1/P_0–Element sowie das konforme P_1/P_1–Element mit geeigneter Stabilisierung. Betrachtet man nur die Genauigkeit der berechneten Lösungen, so liefern Finite Elemente Diskretisierungen mit hoher lokaler Ordnung die besten Ergebnisse. Doch bereits Fortin und Fortin [FF85] haben bei einem Vergleich verschiedener zweidimensionaler Finite Elemente Diskretisierungen festgestellt, daß die Implementation dieser Diskretisierungen, die Assemblierung der Matrizen und das Lösen der Gleichungssysteme sehr aufwendig ist. Dies führt dazu, daß vom Kosten–Effizienz Standpunkt einige Finite Elemente niedriger Ordnung konkurrenzfähig sind. Eine Rangordnung der oben genannten Finite Elemente Diskretisierungen bezüglich des Verhältnisses von Kosten und Effizienz ist nicht geklärt. Wir haben uns aus zweierlei Gründen für das **nichtkonforme P_1/P_0–Finite Element** von Crouzeix und Raviart [CR73] entschieden. Zum einen erfüllt dieses Finite Elemente Paar die Babuška–Brezzi Stabilitätsbedingung (2.12) mit einer positiven Konstanten, die unabhängig von der Gitterfeinheit und der Regularitätskonstante des Gitters ist. Deswegen benötigt man keine zusätzliche Stabilisierung. Der zweite Grund ist eng mit Parallelrechnern verknüpft. Bei der Implementierung eines Programmes mit Finite Elemente Diskretisierungen auf Parallelrechnern kommt man nicht umhin, eine Reihe von Unbekannten auf mehreren Prozessoren zu speichern. Während der numerischen Lösung der Gleichungen ist mehrfach ein Datenaustausch zwischen den verschiedenen Abspeicherungen dieser Unbekannten notwendig. Der Datenaustausch ist mit Kommunikationen verbunden, die zu Effizienzverlusten bei den numerischen Verfahren führen. Ein Vorteil des nichtkonformen P_1/P_0–Finite Elementes besteht darin, daß eine Unbekannte auf höchstens zwei Prozessoren abgespeichert werden muß. Damit reicht **eine einzige Kommunikation** zu einem **vollständigen Datenaustausch zwischen allen Abspeicherungen einer Unbekannten** aus. aus. Diesen Vorzug besitzen konforme Finite Elemente Diskretisierungen nicht. Die Wahl der nichtkonformen P_1/P_0–Finite Elemente Diskretisierung birgt jedoch auch Schwierigkeiten in sich. Wie auch in dieser Arbeit, Kapitel 6 und 7, deutlich wird, verursacht der durch die Nichtkonformität entstehende Konsistenzfehler Zusatzterme in Beweisen, deren Abschätzung Probleme bereiten kann. Ein anderes Problem liegt in der Approximation der Geschwindigkeit durch eine stückweise lineare Funktion und die des Druckes durch eine stückweise konstante Funktion. Auf Grund der niedrigen lokalen Polynomgrade ist zu erwarten, daß man relativ viele Unbekannte im Vergleich zu Ansatzfunktionen höheren Grades benötigt, um numerische Lösungen mit vergleichbarer Genauigkeit zu erhalten.

Neben der Gestalt der Funktionen ist die zugrunde liegende Vernetzung der wesentliche Bestandteil der Definition eines Finite Elemente Raumes und der Schlüssel zur Berechnung guter Lösungen mit einer stark reduzierten Anzahl der Unbekannten. Es ist wohlbekannt, daß mit einer dem Problem angepaßten Vernetzung die Genauigkeit der berechneten Lösung wesentlich erhöht werden kann. Methoden, die diese Anpassung selbständig auf Grundlage der vorhandenen numerischen Daten ausführen, nennt man **adaptive Methoden**. Ausgehend von der Lösung des diskreten Problems auf einem vorgegebenen Gitter wird mit Hilfe **lokaler a posteriori Fehlerschätzer** dieses Gitter lokal verfeinert. Informationen über die Genauigkeit der berechneten Lösung versucht man mit **globalen a posteriori Fehlerschätzern** zu gewinnen. Die globalen Fehlerabschätzungen enthalten jedoch eine Konstante, die problemabhängig sehr groß sein kann. Für eine zuverlässige Fehlerschätzung sind Informationen über die Größenordnung dieser Konstante wichtig. In dieser Arbeit werden lokale a posteriori Fehlerschätzer für verschiedene Normen und ein globaler a posteriori Fehlerschätzer für den Fehler in der L^2–Norm der Geschwindigkeit hergeleitet. Bei der Untersuchung bereitet vor allem die Abschätzung von Termen Schwierigkeiten, deren Ursprung die Nichtkonformität der Diskretisierung ist. Außerdem wird ein Weg zur numerischen Approximation der problemabhängigen Konstanten in der globalen a posteriori Fehlerabschätzung präsentiert.

Ein ähnliches Bild wie bei den Diskretisierungen bietet sich bei den Lösern für die auftretenden linearen Systeme. Auf der einen Seite wurden in den letzten Jahrzehnten ungeheure Fortschritte erzielt, die es heutzutage erlauben, Systeme mit mehreren 10 Millionen Unbekannten zu lösen. Auf der anderen Seite ist die Menge der eingesetzten Verfahren mit all ihren Variationen nahezu unüberschaubar. In umfangreichen Rechnungen zu Benchmarkproblemen [ST96a] wurde die Effizienz beziehungsweise Ineffizienz mancher Methoden erkannt. Insbesondere hat sich die Erkenntnis durchgesetzt, daß bei Verwendung von Diskretisierungen niedriger lokaler Ordnung **Mehrgitterverfahren** an den entscheidenden Stellen der Methoden nötig sind, um schnelle Löser zu erhalten. In dieser Arbeit wird das Verhalten zweier Methoden studiert, die auf Mehrgitterverfahren basieren. Zum einen handelt es sich um gekoppelte Mehrgitterverfahren. Bei diesen wird die Mehrgittertechnik auf ein Gleichungssystem angewandt, das sowohl die Geschwindigkeits– als auch die Druckunbekannten enthält. Zum anderen betrachten wir Druckkorrektur Verfahren. Bei diesen Verfahren ist die Berechnung des Druckes und der Geschwindigkeit entkoppelt. Wesentlicher Bestandteil dieser Verfahren ist eine Iteration zur Berechnung (Korrektur) des Druckes. Innerhalb dieser Iteration werden abwechselnd Gleichungssysteme für die Druck– beziehungsweise Geschwindigkeitsunbekannten gelöst, was jeweils durch den Einsatz von Mehrgittermethoden erfolgt. Gegenstand numerischer Untersuchungen sind ein Vergleich des Verhaltens von gekoppelten Mehrgitterverfahren und Druckkorrektur Verfahren bei der Lösung stationärer und instationärer Navier–Stokes Gleichungen, die Effizienz dieser Verfahren auf Parallelrechnern, der Einfluß von Verfahrensparametern sowie die Abhängigkeit einer guten Wahl der Verfahrensparameter von der Anzahl der verwendeten Prozessoren.

Zur Untersuchung von Effekten, deren Ursache dominante Konvektion ist, werden neben den Navier–Stokes Gleichungen auch skalare Konvektions–Diffusions Gleichungen betrachtet. Konvektions–Diffusions Gleichungen sind im allgemeinen einfacher analytisch zu untersuchen und numerisch zu lösen. Die dabei gewonnenen Erkenntnisse und benutzten Techniken kann man vielfach erfolgreich auf die Navier–Stokes Gleichungen übertragen.

Ein zentrales Anliegen dieser Arbeit besteht darin, die Vor– und Nachteile der nichtkonformen P_1/P_0–Finite Elemente Diskretisierung an Hand numerischer Experimente gegeneinander abzuwägen, um eine Basis für fundierte Vergleiche mit anderen Diskretisierungen zu schaffen.

Gliederung der Arbeit

Die vorliegende Arbeit gliedert sich in vier Teile. In den Kapiteln 1 und 2 werden die untersuchten Probleme und ihre Diskretisierungen eingeführt sowie wesentliche Konvergenzaussagen zitiert.

Kapitel 3 bis 5 sind der Lösung der linearen Sattelpunktprobleme gewidmet. Im Kapitel 3 werden nach einigen Aussagen zur Lösbarkeit der Systeme die verwendeten Mehrgitterverfahren beschrieben. Den Abschluß dieses Kapitels bildet ein Überblick über bisher bekannte Resultate zur Konvergenztheorie der verwendeten Verfahren. Kapitel 4 ist der Implementierung der Verfahren auf einem Parallelrechner gewidmet. Insbesondere werden die Stellen in den Algorithmen herausgearbeitet, an denen Kommunikationen notwendig sind. Aus diesen Überlegungen leiten sich einige Variationen der verwendeten Verfahren ab, die numerisch untersucht werden. Die Darstellungen in den Kapiteln 3 und 4 sind bewußt ausführlich gehalten, da unserer Meinung nach diese Aspekte in der Literatur oft zu kurz kommen. In Kapitel 5 werden die numerischen Kosten der verwendeten Löser untersucht. Dies dient der Vorbereitung der numerischen Vergleiche.

Der dritter Teil der Arbeit, Kapitel 6 und 7, enthält die wesentlichen theoretischen Ergebnisse. In Kapitel 6 werden lokale und globale a posteriori Fehlerschätzer für die nichtkonforme P_1–Diskretisierung von Konvektions–Diffusions Gleichungen hergeleitet. Die lokalen a posteriori Fehlerschätzer in einer diskreten H^1–Norm sind Erweiterungen von Resultaten für Laplace Gleichungen. Die lokalen sowie die globalen Fehlerschätzer in der L^2–Norm sind neu. Die Beweisideen aus Kapitel 6 werden in Kapitel 7 auf die nichtkonforme P_1/P_0–Finite Elemente Diskretisierung der Navier–Stokes Gleichungen übertragen. Es werden lokale Fehlerschätzer in einer diskreten Energienorm hergeleitet, die Erweiterungen bekannter Resultate für die Stokes Gleichungen sind.

Der lokale Fehlerschätzer für die L^2–Norm der Geschwindigkeit und die H^{-1}–Norm des Druckes sowie der globale Fehlerschätzer für die L^2–Norm der Geschwindigkeit sind neu.

Den umfangreichsten Teil der Arbeit stellen die numerischen Untersuchungen in den Kapiteln 8 bis 10 dar. Kapitel 8 ist den Konvektions–Diffusions Gleichungen gewidmet. Es werden unterschiedliche Stabilisierungen sowie die Arbeitsweise verschiedener lokaler a posteriori Fehlerschätzer studiert. Darüber hinaus wird ein Weg zur numerischen Approximation der bereits oben erwähnten problemabhängigen Konstanten innerhalb der globalen Fehlerschätzung in der L^2–Norm präsentiert. Kapitel 9 beinhaltet numerische Tests für die stationären Navier–Stokes Gleichungen. Die Güte der verwendeten nichtkonformen P_1/P_0–Finite Elemente Diskretisierung wird an Hand eines Benchmarkproblems untersucht. An diesem Problem findet auch die numerische Untersuchung der verschiedenen Löser statt. Außerdem werden Beispiele zur Arbeitsweise von lokalen a posteriori Fehlerschätzern präsentiert. Numerische Untersuchungen zu den instationären Navier–Stokes Gleichungen finden sich in Kapitel 10. Ein zentraler Punkt ist wiederum die Güte der nichtkonformen P_1/P_0–Finite Elemente Diskretisierung in Verbindung mit unterschiedlichen Zeitdiskretisierungen. Dies wird an zwei Benchmarkproblemen studiert. Den Abschluß dieses Kapitels bildet die Untersuchung verschiedener Löser für die auftretenden Sattelpunktprobleme.

Die verwendeten Bezeichnungen sind in Anhang A aufgelistet. Neben Standardbezeichnungen für Funktionenräume und Normen haben wir uns vor allem an den von Ciarlet [Cia91], S. 335 ff., vorgeschlagenen Bezeichnungen orientiert.

An dieser Stelle möchte ich all denjenigen Dank sagen, die dazu beigetragen haben, daß diese Arbeit in der vorliegenden Form entstehen konnte. Dieser Dank gilt zuerst meinem Betreuer Prof. Dr. Lutz Tobiska für die wissenschaftliche Leitung und Förderung in den letzten Jahren.

Meinen Freunden und Kollegen am Institut und in anderen Forschungseinrichtungen, insbesondere Dr. Ottfried Dorok, Prof. Dr. William Layton, Dr. Joseph Maubach und Dr. Friedhelm Schieweck, möchte ich für viele hilfreiche Diskussionen und gute Hinweise danken.

Desweiteren gilt mein Dank dem „ug–Team“ in Stuttgart und Heidelberg, insbesondere Dr. Peter Bastian, für das zur Verfügung stellen von Modulen des Programmpaketes „ugp1.0“. Ohne diese wäre die ohnehin schon sehr aufwendige Programmierarbeit noch wesentlich angewachsen. Ein Teil der präsentierten Abbildungen wurde mit dem Graphikprogramm „grape“ erstellt. Hiermit möchte ich mich bei dem „grape–Team“ in Bonn und Freiburg für das zur Verfügung stellen dieses Programmes bedanken.

Bei Dr. Walfred Grambow möchte ich mich für die viele Arbeit bedanken, die er investieren mußte, damit der von mir genutzte Parallelrechner relativ stabil arbeitete.

Prof. em. Dr. Herbert Goering und Dr. Regina Sarazin möchte ich für das Korrekturlesen der Arbeit ganz herzlich danken.

Mein besonderer Dank gilt meiner Ehefrau Anja für ihr innerhalb der letzten Monate aufgebrachtes Verständnis für diese Arbeit sowie das Korrekturlesen.

Kapitel 1

Skalare Konvektions–Diffusions Gleichungen

Skalare Konvektions–Diffusions Gleichungen beschreiben den Transport einer skalaren Größe. Dabei unterscheidet man zwei Arten des Transportes, den diffusiven und den konvektiven. Der diffusive Transport wird mit Hilfe zweiter Ableitungen (Laplace Operator) und der konvektive mit Hilfe erster Ableitungen beschrieben. Dominiert der konvektive Transport, so entstehen im allgemeinen Grenzschichten. Standardformulierungen von diskreten Problemen erweisen sich als instabil. Dieses sind Effekte, die auch bei Navier–Stokes Gleichungen auftreten. Die analytische Untersuchung und die numerische Lösung von Konvektions–Diffusions Gleichungen ist oft wesentlich einfacher, als die der Navier–Stokes Gleichungen. Es ist jedoch möglich, eine Reihe von analytischen als auch numerischen Methoden für Konvektions–Diffusions Gleichungen auf die Navier–Stokes Gleichungen zu übertragen.

In diesem Kapitel werden zunächst die Konvektions–Diffusions Gleichungen eingeführt und Resultate betreffs ihrer Lösbarkeit aufgelistet. Dann wird der Raum der nichtkonformen P_1–Finite Elemente definiert. Im Anschluß werden zwei Stabilisierungsmethoden für diskrete Konvektions–Diffusions Gleichungen erläutert.

1.1 Schwache Formulierungen und Lösbarkeit

In diesem Abschnitt betrachten wir Konvektions–Diffusions Gleichungen

$$\begin{aligned} -\nabla \cdot \nu \nabla u + b \cdot \nabla u + cu &= f && \text{in } \Omega \\ u &= g && \text{auf } \partial\Omega_D \\ \nu \partial u / \partial n &= g_N && \text{auf } \partial\Omega_N. \end{aligned} \tag{1.1}$$

Hierbei ist $\Omega \subset \mathbb{R}^d, d = 2, 3$, ein beschränktes Gebiet mit Lipschitz–Rand $\partial\Omega = \partial\Omega_D \cup \partial\Omega_N$, $\partial\Omega_D \cap \partial\Omega_N = \emptyset$ und $\text{meas}(\partial\Omega_D) > 0$. Die Funktion $\nu \in L^\infty(\Omega)$ ist beschränkt und positiv

$$\nu(x) \geq \nu_0 > 0 \quad \forall\, x \in \Omega.$$

Für die anderen Koeffizienten von (1.1) treffen wir folgende Voraussetzungen:

$$b \in \left(C^1(\overline{\Omega})\right)^d, c \in L^\infty(\Omega), f \in H^{-1}(\Omega), g \in H^{1/2}(\partial\Omega_D), g_N \in H^{-1/2}(\partial\Omega_N). \tag{1.2}$$

Der erste Summand der ersten Gleichung in (1.1) wird Diffusionsterm genannt, der zweite Konvektionsterm und der dritte Reaktionsterm. Ist $\|\nu\|_\infty \ll \|b(x)\|_\infty$, so spricht man von singulär gestörten Problemen. Ist $b \equiv \mathbf{0}$ und $c \equiv 0$, so hat man als Spezialfall die Poisson–Gleichung.

Sei $V_0 = \{v \in H^1(\Omega), v = 0$ auf $\partial\Omega_D\}$. Dann kann man (1.1) in eine schwache Formulierung überführen:

Finde $u \in H^1(\Omega)$ mit

$$\begin{array}{rcll} a(u,v) + b(u,v) + c(u,v) &=& (f,v) + \langle g_N, v\rangle_{\partial\Omega_N} & \forall v \in V_0 \\ u &=& g & \text{auf } \partial\Omega_D\,. \end{array} \tag{1.3}$$

Hierbei bezeichnen $(\cdot,\cdot)$ das Skalarprodukt in $L^2(\Omega)$

$$(u,v) = \int_\Omega uv dx$$

und $\langle\cdot,\cdot\rangle_{\partial\Omega_N}$ das Skalarprodukt in $L^2(\partial\Omega_N)$

$$\langle u,v\rangle_{\partial\Omega_N} = \int_{\partial\Omega_N} uv ds.$$

Die Bilinearformen sind als

$$a(u,v) = (\nu\nabla u, \nabla v), \quad b(u,v) = (b\cdot\nabla u, v), \quad c(u,v) = (cu,v)$$

definiert. Problem (1.3) wird als schwache Formulierung mit konvektiver Form des Konvektionsterms bezeichnet. Die sogenannte schwache Formulierung mit schiefsymmetrischer Form des Konvektionsterms erhält man mit Hilfe der Beziehung $b\cdot\nabla u = \nabla\cdot(ub) - u(\nabla\cdot b)$:

Finde $u \in H^1(\Omega)$ mit

$$\begin{array}{rcll} a(u,v) + b_s(u,v) + c_s(u,v) &=& (f,v) + \langle g_N, v\rangle_{\partial\Omega_N} & \forall v \in V_0 \\ u &=& g & \text{auf } \partial\Omega_D \end{array} \tag{1.4}$$

mit

$$b_s(u,v) = \frac{1}{2}(b\cdot\nabla u + \nabla\cdot(ub), v), \quad c_s(u,v) = ((c - 1/2\nabla\cdot b)u, v)\,.$$

Die Benennung rührt daher, daß sich für $v \in H_0^1(\Omega)$ oder $u \in H_0^1(\Omega)$ mittels partieller Integration $b_s(u,v) = -b_s(v,u)$ ergibt.

Lösungen von (1.1) nennt man starke Lösungen und Lösungen von (1.3) beziehungsweise (1.4) schwache Lösungen der Konvektions–Diffusions Gleichung. Eine starke Lösung kann man nur erwarten, falls alle Funktionen hinreichend glatt sind, das Gebiet einen hinreichend glatten Rand besitzt und die Randbedingungen gewisse Kompatibilitätsbedingungen erfüllen. Präzise Aussagen dazu findet man zum Beispiel bei Gilbarg und Trudinger [GT83]. Bei vielen praktischen Anwendungen sind jedoch diese Voraussetzungen nicht erfüllt, so daß für diese nur schwache Lösungen betrachtet werden können.

Im folgenden wollen wir Aussagen über die Lösbarkeit der schwachen Probleme (1.3) und (1.4) und über Eigenschaften der Lösungen aus der Literatur zusammenfassen.

Satz 1.1.1 *(Lösbarkeit des Dirichletproblems) Sei $\partial\Omega_N = \emptyset$. Es gelten die Voraussetzungen (1.2) an die Glattheit der Koeffizientenfunktionen. Zusätzlich seien $c \geq 0$ und $f \in L^2(\Omega)$. Dann gibt es eine eindeutige Lösung $u \in H^1(\Omega)$ der Probleme (1.3) und (1.4). Die Lösung hängt stetig von den Eingangsdaten ab, und es gilt die a priori Abschätzung*

$$\|u\|_1 \leq C(\|f\|_0 + \|g\|_{1/2,\partial\Omega}) \tag{1.5}$$

mit einer nicht von u abhängigen Konstanten C.

Beweis: Dies sind die Aussagen von Theorem 8.3 und Corollary 8.7 aus Gilbarg und Trudinger [GT83]. Die Voraussetzung $c \geq 0$ entspricht der dortigen Voraussetzung (8.8) für $b \in \left(C^1(\overline{\Omega})\right)^d$. Wegen der Forderungen bezüglich der Beschränktheit der Koeffizientenfunktionen ist Voraussetzung (8.6) in [GT83] erfüllt. □

Man kann auch eine andere Voraussetzung an die Koeffizientenfunktionen stellen:

$$c - 1/2(\nabla \cdot b) \geq c_0 > 0 \quad \text{in } \Omega. \tag{1.6}$$

Mit Hilfe dieser Beziehung kann die positive Definitheit von (1.3) und (1.4) bewiesen werden und damit die Lösbarkeit der diskreten Probleme.

Satz 1.1.2 *(Lösbarkeit des gemischten Randwertproblems) Seien $\nu = const, \nu > 0, g = 0, g_N = 0$. Gelte (1.6) und überdies $b \cdot n \geq 0$ auf $\partial\Omega_N$. Dann sind (1.3) und (1.4) eindeutig lösbar.*

Beweis: Diese Aussage findet man bei Roos, Stynes und Tobiska [RST96], Seite 175 f. . □

Eine Aussage über die Abhängigkeit der Stabilitätskonstanten C in (1.5) von ν findet sich in [RST96], Chapter III, Lemma 1.18.

Satz 1.1.3 *Die Voraussetzungen von Satz 1.1.2 seien erfüllt. Zusätzlich sei $u \in H^2(\Omega)$. Dann gilt die a priori Abschätzung*

$$\nu^{3/2}\|u\|_2 + \nu^{1/2}\|u\|_1 + \|u\|_0 \leq C\|f\|_0. \tag{1.7}$$

Besitzt u exponentielle Randgrenzschichten, so ist (1.7) scharf.

1.2 Der nichtkonforme P_1–Finite Elemente Raum

In diesem Abschnitt wird der Raum der nichtkonformen stückweise linearen Finite Elemente eingeführt, der sogenannten nichtkonformen P_1–Finite Elemente.

Sei $\Omega \subset \mathbb{R}^2$ und $\mathcal{T}_h$ eine reguläre Triangulierung von Ω mit Dreiecken im Sinne von Ciarlet [Cia91], Section 17, (H1). Mit $\{T_i\}_{i=1}^{N_p}$ bezeichnen wir die Menge der Elemente von $\mathcal{T}_h$, mit $\{E_i\}_{i=1}^{N_u}$ die Menge der Kanten der Dreiecke, die nicht auf $\partial\Omega_D$ liegen sowie mit $\{E_i\}_{i=N_u+1}^{N_u+N_D}$ die Menge der Kanten, die auf dem Dirichletrand liegen. Desweiteren seien B_i , $i = 1, \ldots, N_u + N_D$, die Mittelpunkte der Kanten. Der Durchmesser des Dreiecks T (längste Kante) wird mit h_T bezeichnet, die Länge der Kante E mit h_E, und wir definieren

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

Der Raum der nichtkonformen, stückweise linearen Finite Elemente V_h ist definiert als

$$V_h := \left\{ \begin{array}{c} v_h \in L^2(\Omega) \mid v_h|_T \in P_1(T)\ \forall T \in \mathcal{T}_h,\ v_h \text{ ist stetig in } B_i, i = 1, \ldots, N_u, \\ \text{und } v_h(B_i) = g(B_i),\ i = N_u + 1, \ldots, N_u + N_D \end{array} \right\}. \tag{1.8}$$

Mit V_{h0} bezeichnen wir den Raum der nichtkonformen, stückweise linearen Finite Elemente mit homogenen Randdaten in den Kantenmittelpunkten, die auf dem Dirichletrand liegen:

$$V_{h0} := \left\{ \begin{array}{c} v_h \in L^2(\Omega) \mid v_h|_T \in P_1(T)\ \forall T \in \mathcal{T}_h,\ v_h \text{ ist stetig in } B_i, i = 1, \ldots, N_u, \\ \text{und } v_h(B_i) = 0,\ i = N_u + 1, \ldots, N_u + N_D \end{array} \right\}.$$

Die Funktionen von V_h sind stückweise linear, können unstetig auf den Kanten der Dreiecke und am Rand sein, sind jedoch stetig in den Kantenmittelpunkten. Wir versehen V_h mit der Standardbasis $\{\phi_i\}_{i=1}^{N_u+N_D}$ mit

$$\phi_i \in V_h, \quad \phi_i(B_j) = \delta_{ij}.$$

Eine derartige Basisfunktion ist in Abbildung 1.1 zu sehen. Eine Erweiterung der nichtkonfor-

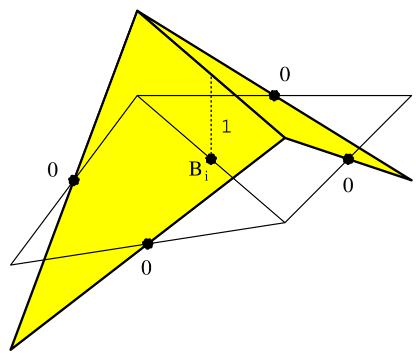


Abbildung 1.1: Standardbasisfunktion des nichtkonformen P_1–Finite Elemente Raumes.

men P_1–Finite Elemente Diskretisierung auf drei Dimensionen ist möglich. Viele theoretische Ergebnisse lassen sich in kanonischer Weise übertragen. Da sich jedoch die numerischen Untersuchungen auf zwei Dimensionen beschränken, betrachten wir auch hier nur diesen Fall.

Lemma 1.2.1 *Die Basisfunktionen $\{\phi_i\}_{i=1}^{N_u+N_D}$ von V_h sind L^2–orthogonal.*

Beweis: Der Durchschnitt der Träger zweier verschiedener Basisfunktionen besteht aus höchstens einem Dreieck. Sei T_{ij} der gemeinsame Träger von ϕ_i und ϕ_j, $i \neq j$. Die Funktion $\phi_i\phi_j$ ist auf T_{ij} eine quadratische Funktion. Deswegen kann das Skalarprodukt $(\phi_i, \phi_j) = (\phi_i, \phi_j)_{T_{ij}}$ mit der Seitenmittelpunktregel exakt ausgewertet werden, und man sieht sofort $(\phi_i, \phi_j) = 0$. □

Wir definieren eine gitterabhängige Seminorm in V_h

$$|v_h|_h := \left(\sum_{T \in \mathcal{T}_h} |v_h|_{1,T}^2 \right)^{1/2} \tag{1.9}$$

und eine gitterabhängige H^1–Norm in V_h

$$\|v_h\|_h := \left(|v_h|_h^2 + \|v_h\|_0^2 \right)^{1/2}.$$

Lemma 1.2.2 $|\cdot|_h$ *ist in V_{h0} eine Norm.*

Beweis: Man hat noch $|v_h|_h = 0 \Rightarrow v_h = 0$ zu zeigen. Aus $|v_h|_h = 0$ folgt, daß v_h elementweise konstant ist. Aus der Stetigkeit in den Seitenmittelpunkten folgt, daß v_h auf Ω konstant ist. Wegen der vorgegebenen homogenen Randwerte folgt schließlich $v_h = 0$. □

Lemma 1.2.3 *Die elementweise definierte Bilinearform $a_h(\cdot,\cdot) : V_h \times V_h \to \mathbb{R}$*

$$a_h(u_h, v_h) = \sum_{T \in \mathcal{T}_h} (\nu \nabla u_h, \nabla v_h)_T$$

ist symmetrisch, und es gilt

$$a_h(v_h, v_h) \geq \nu_0 |v_h|_h^2 \quad \forall v_h \in V_h. \tag{1.10}$$

Beweis: Die Symmetrie ist offensichtlich. Eigenschaft (1.10) folgt aus der Positivität von ν. □

Analog zu $a_h(\cdot,\cdot)$ werden diskrete Versionen der im vorherigen Abschnitt eingeführten Bilinearformen als Summe über die Elemente definiert, beispielsweise

$$b_h(u_h, v_h) := \sum_{T\in\mathcal{T}_h} (b\cdot\nabla u_h, v_h)_T \quad \text{und} \quad c_h(u_h, v_h) := \sum_{T\in\mathcal{T}_h} (cu_h, v_h)_T.$$

1.3 Upwind Diskretisierungen

Die Standard Galerkin Diskretisierung der schwachen Formulierungen (1.3) und (1.4) besteht darin, ein diskretes Problem zu betrachten, indem man die stetigen Räume durch diskrete Räume und die stetigen Bilinearformen durch die diskreten Bilinearformen ersetzt. Für singulär gestörte Probleme stellt sich jedoch heraus, daß diese Vorgehensweise zu instabilen diskreten Problemen führt. Die berechneten Lösungen haben beispielsweise große Oszillationen innerhalb von Grenzschichten. Ein Grund hierfür ist, daß bei dominanter Konvektion eine vorherrschende Richtung existiert, in der Informationen über die Lösung transportiert werden. Dem muß man bei der Diskretisierung des konvektiven Terms Rechnung tragen. In diesem und im folgenden Kapitel werden zwei stabile Diskretisierungsmethoden, Upwind und Stromlinien–Diffusions Methoden, vorgestellt.

Wir beginnen mit der Beschreibung der Upwind Diskretisierung für zweidimensionale, nichtkonforme Finite Elemente. Dazu wird Ω in sogenannte duale Gebiete ω_i zerlegt, die jeweils mit einem Knoten B_i verbunden sind. Sei E_i die Kante in Ω, auf der B_i liegt, und seien T, T' die Dreiecke, die E_i als gemeinsame Kante besitzen. Weiterhin seien S_T und $S_{T'}$ die Schwerpunkte der Dreiecke. Wir bezeichnen mit Λ_i die Menge aller Indizes $k \neq i$, für die die Knoten B_k und B_i zu einem gemeinsamen Element gehören. Das sind vier Indizes. Das duale Gebiet ω_i wird nun als Vereinigung der beiden Dreiecke gebildet, die E_i als eine Kante und S_T beziehungsweise $S_{T'}$ als E_i gegenüberliegenden Eckpunkt besitzen, siehe Abbildung 1.2. Die Kante zwischen E_i und E_k wird als Γ_{ik} und die äußere Einheitsnormale bezüglich ω_i mit n_{ik} bezeichnet. Für Kanten auf dem Rand besteht das duale Gebiet nur aus einem Dreieck.

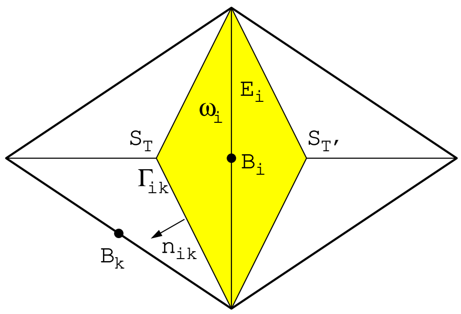


Abbildung 1.2: Duales Gebiet zur Upwind Diskretisierung.

Als nächstes wird ein Operator L_h eingeführt, der eine stetige Funktion v auf eine in den dualen Gebieten stückweise konstante Funktion $L_h v$ abbildet

$$L_h v(x) = v(B_i) \quad \forall x \in \omega_i.$$

Um Informationen über die Konvektionsrichtung einfach nutzen zu können, wird nun der konvektive Term mit Hilfe des Operators L_h durch ein Integral über die Ränder der dualen Gebiete approximiert

$$\begin{aligned}(b\cdot\nabla u, v) &= \sum_{i=1}^{N_u+N_D}(b\cdot\nabla u, v)_{\omega_i} = \sum_{i=1}^{N_u+N_D}\Big[(\nabla\cdot(ub), v)_{\omega_i} - (\nabla\cdot b, uv)_{\omega_i}\Big]\\ &\approx \sum_{i=1}^{N_u+N_D}\Big[(\nabla\cdot(ub), L_h v)_{\omega_i} - (\nabla\cdot b, L_h(uv))_{\omega_i}\Big]\\ &= \sum_{i=1}^{N_u+N_D} v(B_i)\left((\nabla\cdot(ub),1)_{\omega_i} - u(B_i)(\nabla\cdot b,1)_{\omega_i}\right)\\ &= \sum_{i=1}^{N_u+N_D}\sum_{k\in\Lambda_i}\left(\langle ub\cdot n_{ik},1\rangle_{\Gamma_{ik}} - u(B_i)\langle b\cdot n_{ik},1\rangle_{\Gamma_{ik}}\right)v(B_i)\\ &= \sum_{i=1}^{N_u+N_D}\sum_{k\in\Lambda_i}\langle b\cdot n_{ik}, u-u(B_i)\rangle_{\Gamma_{ik}} v(B_i). \qquad (1.11)\end{aligned}$$

In der Upwind Diskretisierung wird nun u durch eine Linearkombination

$$u \approx \lambda_{ik}(b)u(B_i) + (1-\lambda_{ik}(b))u(B_k) \qquad (1.12)$$

ersetzt. Das Kurvenintegral $\langle b\cdot n_{ik},1\rangle_{\Gamma_{ik}}$ beschreibt den konvektiven Fluß durch die Kurve Γ_{ik}. Die Richtung des Flusses ist durch das Vorzeichen von $\langle b\cdot n_{ik},1\rangle_{\Gamma_{ik}}$ bestimmt. Ist $\langle b\cdot n_{ik},1\rangle_{\Gamma_{ik}} < 0$, so ist der konvektive Fluß in ω_i hineingerichtet. Der konvektive Transport, der bei singulär gestörten Problemen dominiert, erfolgt somit vom Knoten B_k zum Knoten B_i. Deshalb wählt man $\lambda_{ik}\in[0,1/2)$ in (1.12), so daß bei der Diskretisierung des konvektiven Terms im Knoten B_i der Wert $u(B_k)$ einen stärkeren Einfluß als $u(B_i)$ besitzt. Ist $\langle b\cdot n_{ik},1\rangle_{\Gamma_{ik}} > 0$, so wird infolge analoger Überlegungen $\lambda_{ik}\in(1/2,1]$ gewählt. Man erhält durch Einsetzen von (1.12) in (1.11) folgende Diskretisierung von $(b\cdot\nabla u, v)$

$$\tilde b_h(u_h,v_h) = \sum_{i=1}^{N_u+N_D}\sum_{k\in\Lambda_i}\langle b\cdot n_{ik}, (1-\lambda_{ik}(b))(u_h(B_k)-u_h(B_i)\rangle_{\Gamma_{ik}} v_h(B_i).$$

In theoretischen Untersuchungen der Upwind Methoden werden an die Koeffizienten λ_{ik} noch einige Bedingungen gestellt. Dazu definiert man

$$\lambda_{ik}(b) = \Phi(t) \quad \text{mit} \quad t = \frac{1}{2\nu}\langle b\cdot n_{ik},1\rangle_{\Gamma_{ik}}. \qquad (1.13)$$

Die Funktion $\Phi(t)$ muß

$$\left.\begin{array}{llcl} (A1) & \Phi(t) & = & 1-\Phi(-t)\ \ \forall t>0, \quad 0\le\Phi(t)\le 1\ \forall t\in\mathbb{R},\\ (A2) & \Phi(t) & \ge & 1/2\ \ \forall t>0,\\ (A3) & g(t) & = & t\Phi(t) \text{ ist Lipschitz–stetig auf } \mathbb{R}, \end{array}\right\} \qquad (1.14)$$

erfüllen, siehe Thiele und Tobiska [TT89]. Das Randintegral in (1.13) wird man im allgemeinen nur numerisch approximieren können. Wir nennen diese Approximation b_{ik} und fordern

$$|\langle b\cdot n_{ik},1\rangle_{\Gamma_{ik}} - \mathsf{b}_{ik}| \le ch^3 \quad \text{sowie} \quad \mathsf{b}_{ik} = \mathsf{b}_{ki} \text{ für } \Gamma_{ik}\cap\partial\Omega = \emptyset. \qquad (1.15)$$

Mit dem Ansatz

$$\Phi(t) = \frac{1}{2} + \operatorname{sgn}(t)\Psi(|t|)$$

ist die Bedingung

$$1 - \Phi(-t) = \frac{1}{2} - \operatorname{sgn}(-t)\Psi(|-t|) = \frac{1}{2} + \operatorname{sgn}(t)\Psi(|t|) = \Phi(t)$$

erfüllt. Einige oft verwendete Upwind Verfahren lassen sich mit Hilfe der Funktionenschar $\Psi_n(t) = \frac{1}{2}\left(t/(t+1)\right)^n$ beschreiben:

$$\begin{aligned} n = 0 &\ :\ \Psi_0(t) = 1/2, \text{ einfaches oder scharfes Upwind,} && (1.16)\\ n = 1 &\ :\ \Psi_1(t) = t/(2(t+1)), \text{ Samarskij Upwind,} && (1.17)\\ n \to \infty &\ :\ \text{Übergang zu zentraler Diskretisierung.} \end{aligned}$$

Die restlichen der Bedingungen $(A1)$ bis $(A3)$ überprüft man leicht.

Nutzt man eine Upwind Diskretisierung, so hat das zu lösende diskrete Gleichungssystem die Gestalt

Finde $u_h \in V_h$ mit

$$\begin{aligned} a_h(u_h, v_h) + \tilde{b}_h(u_h, v_h) + c_h(u_h, v_h) &= (f, v_h) + \langle g_N, v_h\rangle_{\partial\Omega_N} && \forall v_h \in V_{h0}\\ u_h &= g(B_i) && i = N_u{+}1, \ldots, N_u{+}N_D. \end{aligned} \tag{1.18}$$

Bei Roos, Stynes und Tobiska [RST96], Chapter III, Theorem 3.22, Remark 3.23, findet man die folgende Aussage über die Lösbarkeit von (1.18).

Satz 1.3.1 *Seien $\nu \in \mathbb{R}$, b, c und f hinreichend glatt, $c > 0$ sowie (1.15) erfüllt. Desweiteren seien die Innenwinkel aller Dreiecke der gegebenen Triangulierung kleiner oder gleich $\pi/2$. Verwendet man das scharfe Upwind Verfahren (1.16), dann ist (1.18) eindeutig lösbar. Für hinreichend kleines h gilt die Fehlerabschätzung*

$$\left(\nu|u - u_h|_h^2 + \|u - u_h\|_0^2\right)^{1/2} \leq C\frac{h}{\sqrt{\nu}}\left(\|u\|_2 + \|f\|_{1,p}\right) \tag{1.19}$$

mit $p > 2$ und C unabhängig von ν. Nutzt man ein dreidirektionales Gitter, so kann diese Abschätzung zu

$$\left(\nu|u - u_h|_h^2 + \|u - u_h\|_0^2\right)^{1/2} \leq Ch\left(\|u\|_2 + \|f\|_{1,p}\right) \tag{1.20}$$

verbessert werden.

In einem dreidirektionalen Gitter besitzen die Kanten der Dreiecke nur drei ausgezeichnete Richtungen, siehe zum Beispiel Gitter 1 in Abbildung 8.1.

1.4 Nichtkonforme Stromlinien–Diffusions FEM

Die Stromlinien–Diffusions FEM (SDFEM) wurde für konforme Finite Elemente von Hughes und Brooks [HB79] entwickelt. Die ursprüngliche Motivation lag in der Addition künstlicher Diffusion entlang der Stromlinien. Inzwischen gibt es auch andere Interpretationen dieser Methode, so als Petrov–Galerkin Verfahren. Die SDFEM wurde für konforme Diskretisierungen intensiv untersucht, und ihre Eigenschaften sind gut bekannt. Einen Überblick gibt die Monographie von Roos, Stynes und Tobiska [RST96], Kapitel 3.2.1. Auf nichtkonforme Finite Elemente wurde diese Methode in [JMT96] übertragen und analysiert. Wir bezeichnen die nichtkonforme SDFEM im folgenden mit NSDFEM.

Wir betrachten homogene Dirichletprobleme, also $\partial\Omega_D = \partial\Omega$. Desweiteren sei $\nu \in \mathbb{R}$ eine positive Konstante. Mit

$$[|v|]_E = v|_{T_1} - v|_{T_2}$$

wird der Sprung einer Funktion vom Element T_2 zum Element T_1 über die gemeinsame Kante E bezeichnet. Wir betrachten zwei NSDFEM

Finde $u_h \in V_h$ mit

$$\begin{aligned} a_h^i(u_h, v_h) &= l_h(v_h) \quad \forall v_h \in V_{h0},\ i = 1, 2 \\ u_h &= 0 \qquad\quad \text{auf } \partial\Omega\,, \end{aligned} \tag{1.21}$$

wobei die Bilinearformen durch
NSD1 (konvektive Form)

$$\begin{aligned} a_h^1(u_h, v_h) &= a_h(u_h, v_h) + b_h(u_h, v_h) + c_h(u_h, v_h) \\ &+ \sum_{T\in\mathcal{T}_h} (-\nabla\cdot\nu\nabla u_h + b\cdot\nabla u_h + cu_h, \delta_T b\cdot\nabla v_h)_T + \sum_E \sigma_E \langle [|u_h|]_E\,, [|v_h|]_E\rangle_E, \end{aligned} \tag{1.22}$$

NSD2 (schiefsymmetrische Form)

$$\begin{aligned} a_h^2(u_h, v_h) &= a_h(u_h, v_h) + b_{s,h}(u_h, v_h) + c_{s,h}(u_h, v_h) \\ &+ \sum_{T\in\mathcal{T}_h} (-\nabla\cdot\nu\nabla u_h + b\cdot\nabla u_h + cu_h, \delta_T b\cdot\nabla v_h)_T + \sum_E \sigma_E \langle [|u_h|]_E\,, [|v_h|]_E\rangle_E \end{aligned} \tag{1.23}$$

sowie die Linearform l_h durch

$$l_h(v_h) = (f_h, v_h) + \sum_{T\in\mathcal{T}_h} (f_h, \delta_T b\cdot\nabla v_h)_T.$$

gegeben sind. Die Summe über die Elemente stellt die Stromlinien–Diffusions Stabilisierung dar. Der Faktor δ_T ist der Stromlinien–Diffusions Parameter. Der Unterschied zu konformen Methoden besteht im Term

$$\sum_E \sigma_E \langle [|u_h|]_E\,, [|v_h|]_E\rangle_E,$$

der dort wegen der Stetigkeit von u_h und v_h verschwindet. Darüber hinaus lassen sich bei konformen Methoden die konvektive und schiefsymmetrische Form ineinander umformen, was bei NSD1 und NSD2 nicht der Fall ist. Eine geeignete Wahl der Kontrollparameter folgt aus theoretischen Resultaten, siehe Bemerkung 1.4.4.

Das Gebiet Ω sei mittels eines regulären, quasiuniformen Gitters (nach Ciarlet [Cia91], Section 17, S. 135) trianguliert. Wir führen die gitterabhängige nichtkonforme Stromlinien–Diffusions Norm

$$|||v||| := \left(\sum_{T\in\mathcal{T}_h} \left\{ \nu|v|_{1,T}^2 + c_0\|v\|_{0,T}^2 + \delta_T\|b\cdot\nabla v\|_{0,T}^2 \right\} + \sum_E \sigma_E \|\,[|v|]_E\,\|_{0,E}^2 \right)^{1/2}$$

ein. Die Konstante c_0 ist in (1.6) definiert. Auf Grund der regulären Triangulierung gelten die inversen Ungleichungen

$$\begin{aligned} \|v_h\|_{0,T} &\le d_1 h_T^{-1}|v_h|_{1,T} \quad \forall v_h \in V_h,\ \forall T\in\mathcal{T}_h, \\ \|v_h|_T\|_{0,E} &\le d_0 h_E^{-1/2}\|v_h\|_{0,T} \quad \forall v_h \in V_h,\ \forall E\subset\partial T. \end{aligned}$$

Die Kontrollparameter in den Bilinearformen a_h^1 und a_h^2 werden wie folgt gewählt

$$a_h^1 \quad : \quad 0 < \delta_T \le \min\left(\frac{c_0}{4\|c\|^2_{L^\infty(\Omega)}}, \frac{h_T^2}{2\nu d_1^2}\right), \ \sigma_E \ge \frac{3d_0^2\,\|b\|^2_{0,\infty,E}}{c_0\,h_E}, \tag{1.24}$$

$$a_h^2 \quad : \quad 0 < \delta_T \le \min\left(\frac{c_0}{2\|c\|^2_{L^\infty(\Omega)}}, \frac{h_T^2}{2\nu d_1^2}\right), \ \sigma_E \ge 0. \tag{1.25}$$

Mit dieser Wahl läßt sich die folgende Aussage über die Lösbarkeit der diskreten Probleme treffen, siehe [JMT96]:

Satz 1.4.1 *Erfüllen die Kontrollparameter δ_T und σ_E die Beziehungen (1.24) beziehungsweise (1.25), dann sind die Bilinearformen a_h^1 beziehungsweise a_h^2 in V_h positiv definit, das heißt*

$$a_h^i(v_h, v_h) \ge \frac{1}{2}|||v_h|||^2 \quad \forall v_v \in V_h, i = 1, 2.$$

Jetzt kommen wir zu den Konvergenzeigenschaften der Methode. Mit ω_E bezeichnen wir die beiden Elemente, die E als gemeinsame Kante besitzen, wobei E nicht auf dem Rand von Ω liegt. In [JMT96], Theorem 1, ist folgende Aussage bewiesen:

Satz 1.4.2 *Seien die Voraussetzungen aus Satz 1.4.1 erfüllt. Darüber hinaus gelte $u \in H^2(\Omega)$. Dann gilt für die NSD1–Methode die Fehlerabschätzung*

$$|||u - u_h||| \le Ch\left(\sum_{T\in\mathcal{T}_h} \lambda_T |u|^2_{2,T}\right)^{1/2} + Ch^{3/2}\left(\sum_E \lambda_K |u|^2_{2,\omega_E}\right)^{1/2} \tag{1.26}$$

und für die NSD2–Methode

$$|||u - u_h||| \le Ch\left(\sum_{T\in\mathcal{T}_h} \lambda_T |u|^2_{2,T}\right)^{1/2} + Ch^{1/2}\left(\sum_E \lambda_K |u|^2_{2,\omega_E}\right)^{1/2}. \tag{1.27}$$

In beiden Fällen sind

$$\lambda_T = \nu + h_T^2 + \delta_T + \delta_T^{-1}h_T^2, \quad \lambda_K = \min\left(\frac{1}{\sigma_E}, \frac{h_E}{\nu}, \frac{h_E}{h_T^2}\right).$$

Bemerkung 1.4.3 Verallgemeinerung der Ergebnisse. Die obigen Ergebnisse lassen sich auf drei Dimensionen und nichtkonforme Finite Elemente höherer Ordnung ausdehnen, für Details siehe [JMT96].

Bemerkung 1.4.4 Die optimale Wahl der Parameter. Die Wahl der Kontrollparameter muß so erfolgen, daß die Konvergenzordnungen in (1.26) und (1.27) optimiert werden. Für konvektionsdominante Probleme ist im allgemeinen $\nu < h$. Für $\nu > h$ ist keine Stabilisierung notwendig, und man setzt $\delta_T = \sigma_E = 0$. Die Wahl von δ_T beeinflußt nur λ_T. Optimal ist $\delta_T \sim h_T$, woraus $\lambda_T \sim h_T$ folgt. Der erste Term in den Fehlerabschätzungen hat damit den Faktor $h^{3/2}$.

Der Parameter σ_E beeinflußt nur λ_K. Für die NSD1–Methode hat man bereits den Faktor $h^{3/2}$ vor dem zweiten Term. Demzufolge reicht die Wahl von $\sigma_E \sim 1$, um vor beiden Termen die gleichen Potenzen von h zu erhalten. Man erhält die Abschätzungen

$$|||u - u_h||| \le Ch^{3/2}|u|_2 \quad \text{oder} \quad \|u - u_h\|_0 \le Ch^{3/2}|u|_2. \tag{1.28}$$

Dies stimmt mit Resultaten für die konforme SDFEM überein. Für diese Wahl von σ_E ist die Lösbarkeit der diskreten Aufgabe in Satz 1.4.1 allerdings nicht gesichert. Man müßte dazu $\sigma_E \sim h_E^{-1}$ wählen, womit man ebenfalls Abschätzung (1.28) erhält. Bei Testrechnungen, Abschnitt 8.1, stellt sich jedoch heraus, daß $\sigma_E \sim 1$ die bessere Wahl ist.

Für die NSD2–Methode stellt $\sigma_E \sim h_E^{-2}$ die optimale Wahl bezüglich der Konvergenzgeschwindigkeit dar. In diesem Fall erhält man ebenfalls (1.28). Bei numerischen Beispielen hat sich die Wahl von $\sigma_E \sim h_E^{-1}$ zum Erreichen der optimalen Konvergenzordnung als ausreichend herausgestellt.

Bemerkung 1.4.5 Auswirkungen des Sprungterms. Sowohl analytische als auch numerische Ergebnisse zeigen, daß man den Term

$$\sum_E \sigma_E \langle [\![u_h]\!]_E\,, [\![v_h]\!]_E \rangle_E$$

nicht vernachlässigen darf. Dieser Term vergrößert jedoch das Besetztheitsmuster der zu a_h^i, $i = 1, 2$, gehörenden Matrizen im Vergleich zu den Matrizen der Upwind Methoden. Dies wird mit Abbildung 1.3 veranschaulicht. Während bei Upwind Methoden eine Basisfunktion (Kreisring) nur mit den Basisfunktionen verbunden ist, die mit ihr einen gemeinsamen Träger vom Maß größer als Null besitzen (Kreise), ist bei der NSDFEM eine Basisfunktion mit allen Basisfunktionen verbunden, deren Träger mit ihrem Träger mindestens eine Kante gemeinsam hat. So hat man bei Upwind Methoden im allgemeinen 5 Nichtnullelemente pro Matrixzeile, bei NSDFEM dagegen 13. Dies erhöht die Kosten von Multiplikationen mit dieser Matrix. Schwerer wiegt jedoch, daß mit der Koppelung entfernter Basisfunktionen Vorteile der Nutzung nichtkonformer Funktionen auf Parallelrechnern zunichte gemacht werden, siehe Bemerkung 4.1.3.

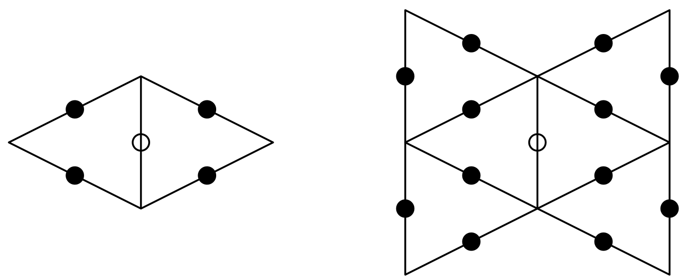


Abbildung 1.3: Die Koppelungen der Basisfunktionen bei Upwind Methoden (links) und NSDFEM (rechts).

Die Diskretisierungen (1.18) und (1.21) führen zu großen, schwach besetzten linearen Gleichungssystemen. Wir lösen diese mit einem Mehrgitterverfahren. Dieses Mehrgitterverfahren ist Bestandteil der Druckkorrektur Verfahren zur Lösung der diskreten Stokes und Navier–Stokes Gleichungen. Es wird im Zusammenhang mit der Lösung des dort auftretenden Sattelpunktproblems ausführlich beschrieben und analysiert. Deswegen sei an dieser Stelle auf die Abschnitte 3.3, 4.3 und 5.2 verwiesen.

Kapitel 2

Die inkompressiblen Navier–Stokes Gleichungen

In diesem Kapitel werden zunächst bekannte Resultate bezüglich der Lösbarkeit der Stokes Gleichungen sowie der stationären und instationären Navier–Stokes Gleichungen aufgelistet und Linearisierungen der Navier–Stokes Gleichungen beschrieben. Anschließend wird der nichtkonforme P_1/P_0–Finite Elemente Raum eingeführt, mit dessen Hilfe die Ortsdiskretisierung der Navier–Stokes Gleichungen erfolgt. Im folgenden Abschnitt werden Verfahren zur Diskretisierung der Ableitung der Geschwindigkeit nach der Zeit vorgestellt. Den Abschluß dieses Kapitels bildet die Beschreibung einer modifizierten Aufgabenstellung, der sogenannten Druckseparation.

Zunächst betrachten wir kurz den physikalischen Hintergrund der inkompressiblen Navier–Stokes Gleichungen und führen einige charakteristische Größen ein. Die inkompressiblen Navier–Stokes Gleichungen sind die mathematische Formulierung zweier mechanischer Erhaltungsgesetze, die Erhaltung der Masse m und die Erhaltung des linearen Impulses $m\mathbf{v}$. Diese Zustandsgrößen können sich auf dreierlei Arten ändern:

- durch Konvektion mit einer Geschwindigkeit $\mathbf{v}$,
- infolge einer inneren Quelle $\mathbf{F}$,
- durch Diffusion (Molekularbewegung). Dabei handelt es sich um einen Ausgleichsprozeß. Der einfachste Ansatz zur Beschreibung der diffusiven Änderung des Impulses ist $-\delta\nabla(m\mathbf{v})$, wobei δ ein Diffusionskoeffizient ist.

Die Herleitung der Navier–Stokes Gleichungen ist in vielen Werken ausführlich beschrieben, zum Beispiel bei Landau und Lifschitz [LL66] oder Feistauer [Fei93]. Für inkompressible Medien, daß heißt Medien mit konstanter Dichte ρ, und Probleme ohne Massenquellen erhält man aus der Massenbilanz die Inkompressibilitätsbedingung

$$\nabla \cdot \mathbf{v} = 0 \tag{2.1}$$

sowie die lineare Impulsbilanz

$$\left(\frac{\partial \mathbf{v}}{\partial t'} + (\mathbf{v}\cdot\nabla)\mathbf{v}\right) - \nu\Delta\mathbf{v} + \frac{1}{\rho}\nabla P = \frac{1}{\rho}\mathbf{F}. \tag{2.2}$$

Hierbei sind ν die kinematische Viskosität des Mediums, P der Druck und $\partial\mathbf{v}/\partial t'$ die Ableitung der Geschwindigkeit nach der Zeit. Das Gleichungssystem (2.2), (2.1) wird inkompressible Navier–Stokes Gleichungen genannt.

Die Funktionen im Gleichungssystem (2.1), (2.2) sind dimensionsbehaftet. Um dies zu ändern, führt man die Größen

- L – eine für das Problem charakteristische Länge,

- U – eine für das Problem charakteristische Geschwindigkeit,
- T – eine für das Problem charakteristische Zeit

ein. Anschließend wird eine Variablentransformation

$$x = \frac{x'}{L}, \quad \mathbf{u} = \frac{\mathbf{v}}{U}, \quad t = \frac{t'}{T}$$

durchgeführt. Setzt man dies in die dimensionsbehafteten Gleichungen ein, so ergibt sich (wobei alle Ableitungen jetzt nach den neuen Variablen zu verstehen sind)

$$\frac{L}{UT}\frac{\partial \mathbf{u}}{\partial t} - \frac{\nu}{UL}\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla\frac{P}{\rho U^2} = \frac{L}{\rho U^2}\mathbf{F}, \qquad \nabla\cdot\mathbf{u} = 0.$$

Man setzt

$$p = \frac{P}{\rho U^2}, \quad Re = \frac{UL}{\nu}, \quad St = \frac{L}{UT}, \quad \mathbf{f} = \frac{L}{\rho U^2}\mathbf{F}$$

und erhält die Navier–Stokes Gleichungen in der dimensionslosen Form

$$St\frac{\partial \mathbf{u}}{\partial t} - \frac{1}{Re}\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p = \mathbf{f}, \quad \nabla\cdot\mathbf{u} = 0.$$

Die Konstante Re wird *Reynoldszahl* und die Konstante St *Strouhalzahl* genannt. Zwei stationäre Strömungen sind ähnlich, wenn ihre Reynoldszahlen übereinstimmen, zwei instationäre Strömungen sind ähnlich, wenn sie in Reynolds– und Strouhalzahl übereinstimmen.

Zur numerischen Lösung der Navier–Stokes Gleichungen verwenden wir eine Transformation mit $L = 1m$, $U = 1m/s$ und $T = 1s$, so daß dimensionslose Gleichungen der Gestalt

$$\frac{\partial \mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p = \mathbf{f}, \quad \nabla\cdot\mathbf{u} = 0 \tag{2.3}$$

mit $p = P/\rho, \mathbf{f} = \mathbf{F}/\rho$ gelöst werden.

2.1 Die Stokes Gleichungen

Die Stokes Gleichungen beschreiben inkompressible Strömungen ohne oder mit vernachlässigbar kleinem konvektiven Transport, sogenannte kriechende Strömungen. Der Einfachheit halber betrachten wir hier nur die stationären Gleichungen. Dann reduzieren sich die Navier–Stokes Gleichungen (2.3) zu

$$\begin{array}{rcll} -\nu\Delta\mathbf{u} + \nabla p &=& \mathbf{f} & \text{in } \Omega \\ \nabla\cdot\mathbf{u} &=& 0 & \text{in } \Omega \\ \mathbf{u} &=& \mathbf{g} & \text{auf } \partial\Omega. \end{array} \tag{2.4}$$

Der fundamentale Unterschied zu den Navier–Stokes Gleichungen besteht in der Linearität von (2.4). Zudem ist $-\nu\Delta$ ein selbstadjungierter und positiv definiter Operator in geeigneten Räumen. Infolge dieser beiden Eigenschaften ist eine analytische Untersuchung der Stokes Gleichungen wesentlich einfacher als die der Navier–Stokes Gleichungen.

Ausgangspunkt der Untersuchungen ist eine Variationsformulierung von (2.4). Seien $V = (H^1(\Omega))^d, V_0 = (H_0^1(\Omega))^d, d = 2,3,$ und $Q = L_0^2(\Omega)$. Diese Räume erfüllen die wichtige inf–sup Bedingung oder Babuška–Brezzi Bedingung

$$\exists\beta > 0 \ : \quad \inf_{q\in Q}\sup_{\mathbf{v}\in V_0}\frac{(\nabla\cdot\mathbf{v}, q)}{|\mathbf{v}|_1\|q\|_0} \geq \beta.$$

Eine Variationsformulierung von (2.4) ist :

Finde $(\mathbf{u}, p) \in V \times Q$ mit

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) - (p, \nabla \cdot \mathbf{v}) &= (\mathbf{f}, \mathbf{v}) \quad \forall \mathbf{v} \in V_0 \\ (q, \nabla \cdot \mathbf{u}) &= \mathbf{0} \qquad \forall q \in Q \\ \mathbf{u} &= \mathbf{g} \qquad \text{auf } \partial\Omega \end{aligned} \tag{2.5}$$

mit

$$a(\mathbf{u}, \mathbf{v}) = \sum_{i,j=1}^{d} \int_{\Omega} \nu \frac{\partial \mathbf{u}_i}{\partial x_j} \frac{\partial \mathbf{v}_i}{\partial x_j} dx.$$

Über die Lösbarkeit und Regularität der Lösung findet man bei Girault/Raviart [GR86], Theorem 5.1, Remark 5.3, folgende Aussage:

Satz 2.1.1 *Ist $\Omega \subset R^d, d = 2, 3$, ein beschränktes Gebiet mit Lipschitz–stetigem Rand $\partial\Omega$. Für alle $\mathbf{f} \in (H^{-1}(\Omega))^d$ und $\mathbf{g} \in (H^{1/2}(\partial\Omega))^d$ mit $\langle \mathbf{g} \cdot \mathbf{n}, 1 \rangle_{\partial\Omega} = 0$ existiert eine eindeutig bestimmte Lösung $(\mathbf{u}, p) \in V \times Q$ von (2.5). Es gilt die a priori Abschätzung*

$$\|\mathbf{u}\|_1 + \|p\|_0 \leq C \left(\|\mathbf{f}\|_{-1} + \|\mathbf{g}\|_{1/2, \partial\Omega} \right)$$

mit C unabhängig von $\mathbf{f}$ und $\mathbf{g}$.

2.2 Die stationären Navier–Stokes Gleichungen

Dieser Abschnitt ist der schwachen Formulierung, der Lösbarkeit und der Linearisierung der stationären Navier–Stokes Gleichungen gewidmet.

Die stationären Navier–Stokes Gleichungen sind

$$\begin{aligned} -\nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p &= \mathbf{f} \qquad \text{in } \Omega \\ \nabla \cdot \mathbf{u} &= 0 \qquad \text{in } \Omega \\ \mathbf{u} &= \mathbf{g} \qquad \text{auf } \partial\Omega. \end{aligned} \tag{2.6}$$

Seien V und Q wie im vorangegangenen Abschnitt definiert. Dann geht man zu einer schwachen Formulierung von (2.6) über :

Finde $(\mathbf{u}, p) \in V \times Q$ mit

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) + b(\mathbf{u}, \mathbf{u}, \mathbf{v}) - (p, \nabla \cdot \mathbf{v}) &= (\mathbf{f}, \mathbf{v}) \quad \forall \mathbf{v} \in V_0 \\ (q, \nabla \cdot \mathbf{u}) &= \mathbf{0} \qquad \forall q \in Q \\ \mathbf{u} &= \mathbf{g} \qquad \text{auf } \partial\Omega \end{aligned} \tag{2.7}$$

mit dem trilinearen Term

$$b(\mathbf{w}, \mathbf{u}, \mathbf{v}) = ((\mathbf{w} \cdot \nabla)\mathbf{u}, \mathbf{v}).$$

Die schwache Formulierung (2.7) nennt man *konvektive Form.* Durch partielle Integration erhält man verschiedene äquivalente Formulierungen, die in dieser Arbeit jedoch nicht betrachtet werden.

Die Lösbarkeit

Sei $W = \{\mathbf{w} \in H_0^1(\Omega)^d, \nabla \cdot \mathbf{w} = 0\}$. Die Norm der Trilinearform $b(\cdot, \cdot, \cdot)$ in W^3 sei definiert durch

$$\|b(\cdot, \cdot, \cdot)\|_{W^3} := \sup_{\mathbf{u}, \mathbf{v}, \mathbf{w} \in W} \frac{b(\mathbf{w}, \mathbf{u}, \mathbf{v})}{|\mathbf{u}|_1 |\mathbf{v}|_1 |\mathbf{w}|_1}.$$

Ferner definieren wir die Norm von $\mathbf{f}$ in W' (dualer Raum von W) durch

$$\|\mathbf{f}\|_{W'} := \sup_{\mathbf{v}\in W} \frac{(\mathbf{f},\mathbf{v})}{|\mathbf{v}|_1}$$

und setzen

$$\rho(\mathbf{u}) := \sup_{\mathbf{v}\in W} \frac{b(\mathbf{v},\mathbf{u},\mathbf{v})}{|\mathbf{v}|_1^2}, \qquad \|r(\mathbf{u})\|_{W'} := \sup_{\mathbf{v}\in W} \frac{(\mathbf{f},\mathbf{v}) - a(\mathbf{u},\mathbf{v}) - b(\mathbf{u}_0,\mathbf{u},\mathbf{v})}{|\mathbf{v}|_1}.$$

Desweiteren sei

$$\nu_0 := \inf\{\rho(\mathbf{u}) + (\|b(\cdot,\cdot,\cdot)\|_{W^3}\|r(\mathbf{u})\|_{W'})^{1/2}; \mathbf{u}\in V \text{ mit } \nabla\cdot\mathbf{u} = 0,\ \mathbf{u}|_{\partial\Omega} = \mathbf{g}\}.$$

Girault und Raviart zeigen in [GR86], Chapter IV,Theorem 2.3 und 2.4, folgenden Existenz– und Eindeutigkeitssatz:

Satz 2.2.1 *Sei $\Omega \in \mathbb{R}^d, d = 2,3$ ein beschränktes Gebiet mit einem Lipschitz–stetigen Rand $\partial\Omega$. Seien $\mathbf{f}\in H^{-1}(\Omega), \mathbf{g} \in \left(H^{1/2}(\Omega)\right)^d$ mit $\langle \mathbf{g}\cdot\mathbf{n}, 1\rangle_{\partial\Omega} = 0$ gegeben. Dann existiert wenigstens eine Lösung $(\mathbf{u},p)\in V\times Q$ von (2.7). Ist zudem $\nu > \nu_0$, dann ist die Lösung eindeutig.*

Bemerkung 2.2.2 Eindeutigkeit der Lösung für homogene Dirichletbedingungen. Für $\mathbf{g}\equiv\mathbf{0}$ vereinfacht sich die Forderung für die Einzigkeit der Lösung zu

$$\|b(\cdot,\cdot,\cdot)\|_{W^3}\|\mathbf{f}\|_{W'}/\nu^2 < 1.$$

Die Zahl $\|b(\cdot,\cdot,\cdot)\|_{W^3}$ ist eine Konstante. Es ist daher erforderlich, daß die rechte Seite $\mathbf{f}$ „hinreichend klein“ oder ν „hinreichend groß“ ist.

Die Linearisierung

Die numerischen Lösung der Navier–Stokes Gleichungen erfolgt iterativ. Wir verwenden eine *äußere nichtlineare Iteration*, das heißt die Navier–Stokes Gleichungen werden zuerst linearisiert und dann die linearen Probleme gelöst.

Eine erste Möglichkeit ist das Newton Verfahren. Sei $(\mathbf{u}^n, p^n)$ eine gegebene Approximation der Lösung von (2.6). Die linearisierten Navier–Stokes Gleichungen für die Unbekannten $(\mathbf{u}^{n+1}, p^{n+1})$ lauten (vergleiche bei Girault/Raviart [GR86], S. 365)

$$\begin{array}{rcll} -\nu\Delta\mathbf{u}^{n+1} + (\mathbf{u}^n\cdot\nabla)\mathbf{u}^{n+1} + (\mathbf{u}^{n+1}\cdot\nabla)\mathbf{u}^n + \nabla p^{n+1} &=& \mathbf{f} + (\mathbf{u}^n\cdot\nabla)\mathbf{u}^n & \text{in } \Omega\\ \nabla\cdot\mathbf{u}^{n+1} &=& 0 & \text{in } \Omega\\ \mathbf{u}^{n+1} &=& \mathbf{g} & \text{auf } \partial\Omega. \end{array}$$

Der Term $(\mathbf{u}^{n+1}\cdot\nabla)\mathbf{u}^n$ kann zu Schwierigkeiten bei der numerischen Lösung eines zugehörigen diskreten Problems führen, da die Diskretisierung von $-\nu\Delta\mathbf{u}^{n+1} + (\mathbf{u}^n\cdot\nabla)\mathbf{u}^{n+1} + (\mathbf{u}^{n+1}\cdot\nabla)\mathbf{u}^n$ indefinit werden kann. Deshalb wird oft

$$(\mathbf{u}^{n+1}\cdot\nabla)\mathbf{u}^n \approx (\mathbf{u}^n\cdot\nabla)\mathbf{u}^n$$

approximiert. Man erhält eine Fixpunktiteration der Gestalt

$$\begin{array}{rcll} -\nu\Delta\mathbf{u}^{n+1} + (\mathbf{u}^n\cdot\nabla)\mathbf{u}^{n+1} + \nabla p^{n+1} &=& \mathbf{f} & \text{in } \Omega\\ \nabla\cdot\mathbf{u}^{n+1} &=& 0 & \text{in } \Omega\\ \mathbf{u}^{n+1} &=& \mathbf{g} & \text{auf } \partial\Omega. \end{array}$$

Diese linearen Gleichungen mit Konvektionsterm werden als Oseen Gleichungen bezeichnet. Da wir in numerischen Tests nur die Fixpunktiteration verwendet haben, beschränken wir uns bei den folgenden Darstellungen auf diese.

Für die praktische Durchführung von Iterationsverfahren ist es günstiger, statt der neuen Iterierten $(\mathbf{u}^{n+1}, p^{n+1})$ die Korrektur $(\delta\mathbf{u}^{n+1}, \delta p^{n+1}) = (\mathbf{u}^{n+1}, p^{n+1}) - (\mathbf{u}^n, p^n)$ zu berechnen. In der schwachen Formulierung hat man in jedem Schritt der Fixpunktiteration ein lineares Problem der Gestalt

Finde $(\delta\mathbf{u}^{n+1}, \delta p^{n+1}) \in V_0 \times Q$ mit

$$\begin{aligned}
(\nu\nabla\delta\mathbf{u}^{n+1}, \nabla\mathbf{v}) + ((\mathbf{u}^n\cdot\nabla)\delta\mathbf{u}^{n+1}, \mathbf{v}) - (\delta p^{n+1}, \nabla\cdot\mathbf{v})& \\
= (\mathbf{f}, \mathbf{v}) - (\nu\nabla\mathbf{u}^n, \nabla\mathbf{v}) - ((\mathbf{u}^n\cdot\nabla)\mathbf{u}^n, \mathbf{v}) + (p^n, \nabla\cdot\mathbf{v}) &\quad \forall\mathbf{v}\in V_0 \\
(q, \nabla\cdot\delta\mathbf{u}^{n+1}) = -(q, \nabla\cdot\mathbf{u}^n) &\quad \forall q\in Q
\end{aligned}$$

zu lösen. In diesem Problem ist eine Funktion $\delta\mathbf{u}^{n+1}$ mit homogenen Dirichlet–Bedingungen gesucht. Die rechte Seite ist der Defekt für die alte Näherung $(\mathbf{u}^n, p^n)$. Die Diskretisierung des linearen Problems ist im Abschnitt 2.4 beschrieben.

Der trilineare Term

Es sind einige Eigenschaften und Abschätzungen des trilinearen Terms $b(\mathbf{u}, \mathbf{v}, \mathbf{w}) = ((\mathbf{u}\cdot\nabla)\mathbf{v}, \mathbf{w})$ der Navier–Stokes Gleichungen zusammengestellt, die später benötigt werden.

Lemma 2.2.3 *Seien* $\mathbf{u}, \mathbf{v}, \mathbf{w} \in (H^1(\Omega))^d$. *Darüber hinaus sei eine der drei Forderungen* $\mathbf{v} \in (H_0^1(\Omega))^d$, $\mathbf{w} \in (H_0^1(\Omega))^d$, $\mathbf{u}\cdot n = 0$ *auf* $\partial\Omega$ *erfüllt. Dann gilt*

$$b(\mathbf{u}, \mathbf{v}, \mathbf{w}) + b(\mathbf{u}, \mathbf{w}, \mathbf{v}) = -(\nabla\cdot\mathbf{u}, \mathbf{v}\cdot\mathbf{w}).$$

Beweis: Mit partieller Integration erhält man

$$(\nabla\cdot\mathbf{u}, \mathbf{v}\cdot\mathbf{w}) = \langle\mathbf{u}\cdot n, \mathbf{v}\cdot\mathbf{w}\rangle_{\partial\Omega} - ((\mathbf{u}\cdot\nabla)\mathbf{v}, \mathbf{w}\ ) - ((\mathbf{u}\cdot\nabla)\mathbf{w}, \mathbf{v}).$$

□

Folgerung 2.2.4 *Die Bezeichnungen und Voraussetzungen seien wie im vorangegangenen Lemma. Dann sind*

$$b(\mathbf{u}, \mathbf{v}, \mathbf{v}) = ((\mathbf{u}\cdot\nabla)\mathbf{v}, \mathbf{v}) = -1/2(\nabla\cdot\mathbf{u}, \mathbf{v}\cdot\mathbf{v}), \quad b(\mathbf{u}, \mathbf{v}, \mathbf{u}) + b(\mathbf{u}, \mathbf{u}, \mathbf{v}) = -(\nabla\cdot\mathbf{u}, \mathbf{v}\cdot\mathbf{u}).$$

Ist $\mathbf{u}$ *divergenzfrei, das heißt* $\nabla\cdot\mathbf{u} = 0$, *so gelten*

$$b(\mathbf{u}, \mathbf{v}, \mathbf{w}) + b(\mathbf{u}, \mathbf{w}, \mathbf{v}) = 0, \quad b(\mathbf{u}, \mathbf{v}, \mathbf{w}) = 1/2(b(\mathbf{u}, \mathbf{v}, \mathbf{w}) - b(\mathbf{u}, \mathbf{w}, \mathbf{v})).$$

Einige Abschätzungen des Betrages des trilinearen Terms findet man in Temam [Tem83, Lemma 2.1], Constantin und Foias [CF88, Proposition 6.1] oder Layton und Tobiska [LT96]:

$$|b(\mathbf{u}, \mathbf{v}, \mathbf{w})| \leq c\|\mathbf{u}\|_0|\mathbf{v}|_1\|\mathbf{w}\|_\infty, \tag{2.8}$$

$$|b(\mathbf{u}, \mathbf{v}, \mathbf{w})| \leq c\|\mathbf{u}\|_0|\mathbf{v}|_{1,\infty}\|\mathbf{w}\|_0. \tag{2.9}$$

In jeder dieser Abschätzungen müssen die Funktionen natürlich die erforderliche Regularität besitzen, damit die Normen in den rechten Seiten existieren.

2.3 Die instationären Navier–Stokes Gleichungen

Die instationären Navier–Stokes Gleichungen haben die Form

$$\begin{array}{rcll} \mathbf{u}_t(t) - \nu\Delta\mathbf{u}(t) + (\mathbf{u}(t)\cdot\nabla)\mathbf{u}(t) + \nabla p(t) &=& \mathbf{f}(t) & \text{in } \Omega\times(0,T] \\ \nabla\cdot\mathbf{u}(t) &=& 0 & \text{in } \Omega\times(0,T] \\ \mathbf{u}(t) &=& \mathbf{g}(t) & \text{auf } \partial\Omega\times(0,T] \\ \mathbf{u} &=& \mathbf{u}_0 & \text{für } t=0, \end{array} \tag{2.10}$$

wobei $\mathbf{u}_t(t)$ die Ableitung von $\mathbf{u}(t)$ nach der Zeit bezeichnet. Im folgenden wird die Abhängigkeit von t in der Bezeichnung nicht mehr betont. Von der starken Formulierung (2.10) geht man auf eine schwache Formulierung über. Dazu werden die Funktionenräume V und W sowie

$$H = \overline{\left\{\mathbf{u}\in(C_0^\infty(\Omega))^d\ ,\ \nabla\cdot\mathbf{u}=0\right\}}^{L^2(\Omega)}$$

benötigt. Eine schwache Formulierung lautet

Finde $(\mathbf{u}(t), p(t)) \in L^2(0,T;V)\times L^2(0,T;Q)$ mit

$$\begin{array}{rcll} (\mathbf{u}_t,\mathbf{v}) + a(\mathbf{u},\mathbf{v}) + b(\mathbf{u},\mathbf{u},\mathbf{v}) - (p,\nabla\cdot\mathbf{v}) &=& (\mathbf{f},\mathbf{v}) & \forall\mathbf{v}\in V_0,\ \text{für fast alle } t\in(0,T] \\ (q,\nabla\cdot\mathbf{u}) &=& \mathbf{0} & \forall q\in Q,\ \text{für fast alle } t\in(0,T] \\ \mathbf{u} &=& \mathbf{g} & \text{auf } \partial\Omega,\ \text{für fast alle } t\in(0,T] \\ \mathbf{u} &=& \mathbf{u}_0 & \text{für } t=0. \end{array} \tag{2.11}$$

Eine Existenz– und Eindeutigkeitsaussage findet man bei Feistauer [Fei93], Theorem 8.7.7 und 8.7.65:

Satz 2.3.1 *Seien $\mathbf{g}(x,t)\equiv\mathbf{0}$ für $x\in\partial\Omega, t\in[0,T]$, $\mathbf{f}\in L^2(0,T;W')$ und $\mathbf{u}_0\in H$. Dann existiert wenigstens eine Lösung von (2.11). Es gilt $u\in L^\infty(0,T;H)$. Ist $d=2$, so ist dies die einzige Lösung.*

2.4 Die Diskretisierung im Ort mit dem nichtkonformen P_1/P_0–Finite Element

Die Wahl diskreter Räume zur numerischen Lösung der Stokes und Navier–Stokes Gleichungen bietet eine Schwierigkeit: der Raum zur Approximation der Geschwindigkeit V_{h0} muß hinreichend viele Freiheitsgrade im Vergleich zum Raum zur Approximation der Druckunbekannten Q_{h0} besitzen. Die Forderung an die beiden Räume läßt sich in einer hinreichenden und notwendigen Bedingung ausdrücken, nämlich in einem diskreten Analogon zur Babuška–Brezzi Bedingung (BB–Bedingung)

$$\exists\beta>0\ :\quad \inf_{q\in Q_{h0}}\sup_{\mathbf{v}\in V_{h0}}\frac{(\nabla\cdot\mathbf{v},q)}{|\mathbf{v}|_h\|q\|_0}\geq\beta, \tag{2.12}$$

wobei β unabhängig von h ist. Die Bedeutung der BB–Bedingung ist ausführlich im Buch von Brezzi und Fortin [BF91] diskutiert. Die einfachste Wahl, eine stückweise lineare konforme Geschwindigkeit und ein stückweise konstanter Druck, erfüllt die diskrete BB–Bedingung nicht, siehe zum Beispiel bei Girault und Raviart [GR86], Kapitel II,2.1. Aus diesem Grunde ist die Entwicklung und Untersuchung stabiler Paare von diskreten Räumen notwendig. Ein Weg führt zu nichtkonformer Approximation der Geschwindigkeit.

Sei eine reguläre Triangulierung $\mathcal{T}_h$ von Ω mit Dreiecken gegeben. Die Anzahl der Kanten, die nicht auf dem Dirichletrand liegen, sei $N_u/2$, und die Anzahl der Kanten auf dem Dirichletrand sei $N_D/2$. Wir wählen als V_h beziehungsweise V_{h0} die zweidimensionalen Analoga zu den nichtkonformen P_1–Finite Elemente Räumen aus Abschnitt 1.2

$$V_h := \left\{ \begin{array}{r} \mathbf{v}_h \in (L^2(\Omega))^2 \,|\, \mathbf{v}_h|_T \in (P_1(T))^2 \ \forall T \in \mathcal{T}_h, \ \mathbf{v}_h \text{ ist stetig in } B_i, i=1,\ldots,N_u/2, \\ \text{und } \mathbf{v}_h(B_i) = \mathbf{g}(B_i), \ i = N_u/2+1,\ldots,N_u/2+N_D/2 \end{array} \right\}$$

und

$$V_{h0} := \left\{ \begin{array}{r} \mathbf{v}_h \in (L^2(\Omega))^2 \,|\, \mathbf{v}_h|_T \in (P_1(T))^2 \ \forall T \in \mathcal{T}_h, \ \mathbf{v}_h \text{ ist stetig in } B_i, i=1,\ldots,N_u/2, \\ \text{und } \mathbf{v}_h(B_i) = \mathbf{0}, \ i = N_u/2+1,\ldots,N_u/2+N_D/2 \end{array} \right\}.$$

Zur Approximation des Druckes wählen wir den Raum der stückweise konstanten Funktionen

$$Q_{h0} = \left\{ q_h \in L_0^2(\Omega) \mid q_h|_T \in P_0(T), \ \forall T \in \mathcal{T}_h \right\}.$$

Dieser nichtkonforme P_1/P_0–Finite Elemente Raum wird oft nach Crouzeix und Raviart benannt, die diesen in [CR73] eingeführt und analysiert haben. Er hat die folgenden Eigenschaften:

Satz 2.4.1 *Die Räume V_{h0} und Q_{h0} erfüllen die diskrete BB–Bedingung (2.12). Die Konstante β ist unabhängig von h und den Innenwinkeln der Dreiecke der Triangulierung $\mathcal{T}_h$.*

Außerdem wird der Raum

$$Q_h = \left\{ q_h \in L^2(\Omega) \mid q_h|_T \in P_0(T), \ \forall T \in \mathcal{T}_h \right\}$$

benötigt, dessen Dimension um eins größer ist als die von Q_{h0}. Die Räume V_h und Q_h werden mit den Standardbasen ausgestattet:

$$V_h = \operatorname{span}\{\phi_i\}_{i=1}^{N_u+N_D}$$

mit

$$\phi_i(B_j) = \begin{pmatrix} \delta_{ij} \\ 0 \end{pmatrix}, \quad \phi_{i+N_u/2+N_D/2}(B_j) = \begin{pmatrix} 0 \\ \delta_{ij} \end{pmatrix} \quad i,j = 1,\ldots,N_u/2+N_D/2, \tag{2.13}$$

sowie

$$Q_h = \operatorname{span}\{\psi_j\}_{j=1}^{N_p} \quad \text{ mit } \psi_j = \begin{cases} 1 & x \in T_j \\ 0 & \text{sonst} \end{cases}. \tag{2.14}$$

Analog zu den Konvektions–Diffusions Gleichungen werden elementweise erklärte Bi– und Trilinearformen definiert

$$\begin{aligned} a_h(\mathbf{u}_h, \mathbf{v}_h) &= \sum_{T \in \mathcal{T}_h} (\nu \nabla \mathbf{u}_h, \nabla \mathbf{v}_h)_T, \\ b_h(\mathbf{u}_h, \mathbf{v}_h, \mathbf{w}_h) &= \sum_{T \in \mathcal{T}_h} ((\mathbf{u}_h \cdot \nabla)\mathbf{v}_h, \mathbf{w}_h)_T, \\ \tilde{b}_h(\mathbf{u}_h, \mathbf{v}_h, \mathbf{w}_h) &= \sum_{i=1}^{N_u} \sum_{k \in \Lambda_i} \langle \mathbf{u}_h \cdot n_{ik}, (1-\lambda_{ik}(\mathbf{u}_h))(\mathbf{v}_h(B_k) - \mathbf{v}_h(B_i) \rangle_{\Gamma_{ik}} \cdot \mathbf{w}_h(B_i). \end{aligned}$$

Die Trilinearform $\tilde{b}_h(\mathbf{u}_h, \mathbf{v}_h, \mathbf{w}_h)$ ist das Analogon zur Upwind Diskretisierung aus Abschnitt 1.3, wobei hier die Rolle des Konvektionsvektors b von $\mathbf{u}_h$ übernommen wird.

Die Stokes Gleichungen

Die diskrete Approximation der Stokes Gleichungen (2.5) lautet

Finde $(\mathbf{u}_h, p_h) \in V_h \times Q_{h0}$ mit

$$\begin{array}{rcll} a_h(\mathbf{u}_h, \mathbf{v}_h) - (p_h, \nabla \cdot \mathbf{v}_h) & = & (\mathbf{f}_h, \mathbf{v}_h) & \forall \mathbf{v}_h \in V_{h0} \\ (q_h, \nabla \cdot \mathbf{u}_h) & = & \mathbf{0} & \forall q_h \in Q_{h0} \\ \mathbf{u}_h & = & \mathbf{g}_h . & \end{array} \tag{2.15}$$

Bemerkung 2.4.2 Approximation der Randdaten. Die Wahl einer geeigneten diskreten Randbedingung $\mathbf{g}_h$ kann unter Umständen Probleme bereiten. Für das stetige Problem gilt auf Grund der Massenbilanz $\langle \mathbf{g} \cdot n, 1\rangle_{\partial\Omega} = 0$. Ist das diskrete Analogon

$$\sum_{E \subset \partial\Omega} \int_E \mathbf{g}_h \cdot n \, ds = 0 \tag{2.16}$$

nicht erfüllt, so kann man die diskrete Massenbilanz $(q_h, \nabla \cdot \mathbf{u}_h) = \mathbf{0} \;\; \forall q_h \in Q_h$ nicht erfüllen. Eine geeignete Wahl des Wertes im Dirichletknoten B_i bei inhomogenen Randdaten ergibt sich durch numerische Approximation von

$$\mathbf{g}_h(B_i) = \frac{1}{h_{E_i}} \int_{E_i} \mathbf{g} ds.$$

Oft reicht die Standardwahl $\mathbf{g}_h(B_i) = \mathbf{g}(B_i)$ aus, damit der Fehler in der diskreten Massenbilanz im Rundungsfehlerbereich liegt. Ist dies nicht der Fall, kann man eine Integrationsformel höherer Ordnung verwenden. Nutzt man etwa die Simpsonregel, so erhält man

$$\mathbf{g}_h(B_i) = (\mathbf{g}(B_{i0}) + 4\mathbf{g}(B_i) + \mathbf{g}(B_{i1}))/6.0,$$

wobei B_{i0} und B_{i1} Anfangs- und Endpunkt der Kante E sind.

Ein anderes Problem tritt bei einem krummlinigen Rand auf. Dieser wird durch einen Polygonzug so approximiert, daß die Punkte B_{i0} und B_{i1} der Kante E auf dem Rand liegen. Im allgemeinen wird sich aber der Punkt B_i nicht auf dem Rand befinden. Ist die Randkurve durch eine Parameterdarstellung gegeben und sind t_{i0} und t_{i1} die Parameter zu den Punkten B_{i0} und B_{i1}, dann wird dem Punkt B_i der Funktionswert des Randpunktes mit dem Parameter $(t_{i0} + t_{i1})/2$ zugeordnet.

Folgenden Existenz– und Eindeutigkeitssatz findet man bei Crouzeix und Raviart [CR73], Theorem 2 und Abschnitt 7:

Satz 2.4.3 *Seien die Voraussetzungen von Satzes 2.1.1 erfüllt, und sei Ω ein polygonal berandetes Gebiet. Dann hat das diskrete Stokes Problem (2.15) genau eine Lösung $(\mathbf{u}_h, p_h)$.*

Für homogene Randwerte wird in Crouzeix und Raviart [CR73], Theorem 3, 4 und 6, folgende a priori Abschätzung bewiesen:

Satz 2.4.4 *Sei Ω ein polygonal berandetes Gebiet, $\mathbf{u} \in (H^2(\Omega))^2 \cap (H_0^1(\Omega))^2, p \in H^1(\Omega) \cap L_0^2(\Omega)$ die Lösung der stetigen Stokes Gleichungen mit homogenen Dirichletrandbedingungen und $(\mathbf{u}_h, p_h)$ die Lösung der diskreten Stokes Gleichungen (2.15). Dann gilt für hinreichend kleines h die a priori Fehlerabschätzungen*

$$h|\mathbf{u} - \mathbf{u}_h|_h + \|\mathbf{u} - \mathbf{u}_h\|_0 + h\|p - p_h\|_0 \le c(\nu) h^2 (|\mathbf{u}|_2 + |p|_1).$$

Bemerkung 2.4.5 Verwendung des Raumes Q_h anstelle von Q_{h0}. Auf Grund der einfachen Basisfunktionen von Q_h wird in der numerischen Praxis ein diskretes System gelöst, bei welchem Q_{h0} in (2.15) durch Q_h ersetzt wird. Man erhält im Vergleich zu (2.15) eine zusätzliche Gleichung und das entsprechende lineare System hat den Rangabfall 1, vergleiche Lemma 3.1.2. Für dieses rangdefiziente System berechnet man eine Lösung, indem man beispielsweise einen Wert des Druckes festlegt. Im Anschluß wird die so berechnete Lösung in Q_{h0} projiziert, womit man die Lösung von (2.15) erhält.

Mit Hilfe der Basisfunktionen (2.13) und (2.14) erhält man Matrixrepräsentationen der diskreten Bi– und Trilinearformen

$$\begin{array}{lll} D & := & a_h(\phi_i, \phi_j)_{i,j=1,\ldots,N_u} \\ B & := & (\psi_j, \phi_i)_{i=1,\ldots,N_u; j=1,\ldots,N_p} \end{array}$$

: diskreter, gewichteter Laplace Operator,
: diskreter Gradient.

Die Matrix D hat die Gestalt

$$D = \begin{pmatrix} A_1 & 0 \\ 0 & A_1 \end{pmatrix}.$$

Die diskreten Funktionen besitzen eine eindeutige Darstellung

$$\mathbf{u}_h = \sum_{i=1}^{N_u} u_i \phi_i, \quad p_h = \sum_{j=1}^{N_p} p_j \psi_j, \quad \mathbf{f}_h = \sum_{i=1}^{N_u} (\mathbf{f}, \phi_i)\phi_i = \sum_{i=1}^{N_u} f_i \phi_i.$$

Wir bezeichnen die Koeffizientenvektoren mit $u = (u_1, \ldots, u_{N_u})^T$, $p = (p_1, \ldots, p_{N_p})^T$ und $f = (f_1, \ldots, f_{N_u})^T$. Damit erhält das diskrete Stokes Problem (2.15) die Gestalt

$$\begin{pmatrix} D & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}. \tag{2.17}$$

Die stationären Navier–Stokes Gleichungen

Die mit einer Upwind Methode stabilisierten diskreten Navier–Stokes Gleichungen lauten

Finde $(\mathbf{u}_h, p_h) \in V_h \times Q_{h0}$ mit

$$\begin{array}{rcll} a_h(\mathbf{u}_h, \mathbf{v}_h) + \tilde{b}_h(\mathbf{u}_h, \mathbf{u}_h, \mathbf{v}_h) - (p_h, \nabla \cdot \mathbf{v}_h) & = & (\mathbf{f}_h, \mathbf{v}_h) & \forall \mathbf{v}_h \in V_{h0} \\ (q_h, \nabla \cdot \mathbf{u}_h) & = & \mathbf{0} & \forall q_h \in Q_{h0} \\ \mathbf{u}_h & = & \mathbf{g}_h & \text{auf } \partial\Omega, \end{array} \tag{2.18}$$

wobei die Wahl von $\mathbf{g}_h$ wie in Bemerkung 2.4.2 erfolgen kann.

Aussagen über die Lösbarkeit des diskreten Problems (2.18) mit homogenen Randwerten findet man zum Beispiel bei Roos, Stynes und Tobiska [RST96], Chapter IV, Theorem 2.3, 2.7 und Remark 2.10.

Satz 2.4.6 *Sei $\mathbf{f} \in (L^2(\Omega))^2$. Erfüllt die Upwind Funktion $\Phi(t)$ die Bedingungen (A1), (A2) und (A3) aus (1.14), dann hat (2.18) wenigstens eine Lösung.*

Sei darüber hinaus $\nu > \nu_0(\Omega, \mathbf{f})$ hinreichend groß. Dann haben sowohl die stetigen Navier–Stokes Gleichungen (2.7) als auch die diskreten Navier–Stokes Gleichungen (2.18) eindeutige Lösungen $(\mathbf{u}, p)$ beziehungsweise $(\mathbf{u}_h, p_h)$. Unter den Voraussetzungen $\mathbf{u} \in (H^2(\Omega))^2 \cap (H_0^1(\Omega))^2$ und $p \in H^1(\Omega) \cap L_0^2(\Omega)$ gelten die a priori Abschätzungen

$$|\mathbf{u} - \mathbf{u}_h|_h \le \frac{c}{\nu_0} Mh, \quad \|\mathbf{u} - \mathbf{u}_h\|_0 \le cMh, \quad \|p - p_h\|_0 \le \frac{c}{\nu_0^2} Mh,$$

wobei M ein Polynom zweiten Grades in $\|\mathbf{u}\|_2$ und $\|p\|_1$ mit von h und ν_0 unabhängigen Koeffizienten ist.

Die diskreten Navier–Stokes Gleichungen (2.18) werden mittels Fixpunktiteration gelöst. Sei $(\mathbf{u}_h^n, p_h^n)$ die gegenwärtige Iterierte der nichtlinearen Iteration, mit $(\delta\mathbf{u}_h^{n+1}, \delta p_h^{n+1})$ wird die gesuchte Korrektur bezeichnet. Dann lautet das mit einer Upwind Methode stabilisierte lineare Problem in der Fixpunktiteration

Finde $(\delta\mathbf{u}_h^{n+1}, \delta p_h^{n+1}) \in V_{h0} \times Q_{h0}$ mit

$$\begin{aligned} a_h(\delta\mathbf{u}_h^{n+1}, \mathbf{v}_h) + \tilde{b}_h(\mathbf{u}_h^n, \delta\mathbf{u}_h^{n+1}, \mathbf{v}_h) - (\delta p_h^{n+1}, \nabla\cdot\mathbf{v}_h) & \\ = (\mathbf{f}_h, \mathbf{v}_h) - a_h(\mathbf{u}_h^n, \mathbf{v}_h) - \tilde{b}_h(\mathbf{u}_h^n, \mathbf{u}_h^n, \mathbf{v}_h) + (p_h^n, \nabla\cdot\mathbf{v}_h) & \quad \forall \mathbf{v}_h \in V_{h0} \\ (q_h, \nabla\cdot\delta\mathbf{u}_h^{n+1}) = -(q_h, \nabla\cdot\mathbf{u}_h^n) & \quad \forall q_h \in Q_{h0}. \end{aligned} \tag{2.19}$$

Auch hier wird wiederum anstelle von Q_{h0} der Raum Q_h bei der numerischen Lösung der Gleichungen verwendet, vergleiche Bemerkung 2.4.5. Mit Hilfe der Basisfunktionen (2.13) und (2.14) kann man das diskrete Problem (2.19) in Matrixform schreiben. Dazu wird

$$C(\mathbf{u}_h^n) \ := \ \tilde{b}_h(\mathbf{u}_h^n, \phi_i, \phi_j)_{i,j=1,\dots,N_u} \quad : \ \text{konvektive Matrix}$$

definiert. Diese Matrix hat die Gestalt

$$C(\mathbf{u}_h^n) = \begin{pmatrix} C_1(\mathbf{u}_h^n) & 0 \\ 0 & C_1(\mathbf{u}_h^n) \end{pmatrix}.$$

Es existieren eindeutige Darstellungen

$$\mathbf{u}_h^n = \sum_{i=1}^{N_u} u_i^n \phi_i, \quad \delta\mathbf{u}_h^{n+1} = \sum_{i=1}^{N_u} \delta u_i^{n+1}\phi_i, \quad p_h^n = \sum_{j=1}^{N_p} p_j^n \psi_j, \quad \delta p_h^{n+1} = \sum_{j=1}^{N_p} p_j^{n+1}\psi_j,$$

$$\mathbf{f}_h = \sum_{i=1}^{N_u} (\mathbf{f}, \phi_i)\phi_i = \sum_{i=1}^{N_u} f_i\phi_i.$$

Die Koeffizientenvektoren werden mit $u^n, \delta u^{n+1}, p^n, \delta p^{n+1}$ und f bezeichnet. Dann hat das lineare Problem in jedem Schritt der Fixpunktiteration die Gestalt

$$\begin{pmatrix} D + C(\mathbf{u}_h^n) & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} \delta u^{n+1} \\ \delta p^{n+1} \end{pmatrix} = \begin{pmatrix} f - (D + C(\mathbf{u}_h^n))u^n - Bp^n \\ -B^T u^n \end{pmatrix}. \tag{2.20}$$

2.5 Die Diskretisierung in der Zeit

Die prinzipielle Vorgehensweise zur numerischen Lösung der instationären Navier–Stokes Gleichungen besteht darin, die Lösung in einer Folge diskreter Zeitpunkte zu berechnen. Dazu muß die Zeitableitung $\mathbf{u}_t$ mittels eines Verfahrens zur Lösung gewöhnlicher Differentialgleichungen geeignet diskretisiert werden. In jedem Zeitschritt hat man ein diskretes System zu lösen, welches eine zu den stationären Navier–Stokes Gleichungen ähnliche Gestalt hat.

Modifizierte Ein–Schritt–θ–Methoden

Ein–Schritt–θ–Methoden zur Lösung gewöhnlicher Differentialgleichungen $y'(t) = f(t, y(t))$ haben die Gestalt

$$\frac{y(t_{k+1}) - y(t_k)}{t_{k+1} - t_k} = \theta f(t_{k+1}, y(t_{k+1})) + (1-\theta) f(t_k, y(t_k)) \tag{2.21}$$

mit $0 \leq \theta \leq 1$. Für $\theta = 1$ erhält man das implizite Euler Verfahren, welches von erster Ordnung konvergiert. Die Wahl $\theta = 1/2$ führt auf die Trapezregel oder das Crank–Nicolson Verfahren. Dieses Verfahren konvergiert von zweiter Ordnung. Ein–Schritt–θ–Methoden können als implizite Adams Verfahren angesehen werden, siehe bei Hairer, Nørsett und Wanner [HNW87].

Zur Lösung der instationären Navier–Stokes Gleichungen betrachten wir modifizierte Ein–Schritt-θ-Methoden, wie sie von Turek [Tur96] vorgeschlagen wurden. Die Modifikation besteht darin, daß die Diskretisierung (2.21) nur auf die Geschwindigkeit $\mathbf{u}$ und die rechte Seite $\mathbf{f}$, jedoch nicht auf den Druck p angewandt wird. Der Druck wird für beliebiges θ wie beim impliziten Euler Verfahren behandelt. Diese Modifikation hat sich in numerischen Tests als vorteilhaft erwiesen. Man erhält ein System der Gestalt

$$\begin{aligned} \frac{\mathbf{u}-\mathbf{u}(t_k)}{\tau} + \theta(-\nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u}) + \nabla p &= \theta\mathbf{f}(t_{k+1}) + (1-\theta)\mathbf{f}_{alt}(t_k) && \text{in } \Omega \\ \nabla\cdot\mathbf{u} &= 0 && \text{in } \Omega \\ \mathbf{u} &= \mathbf{g}(t_{k+1}) && \text{auf } \partial\Omega, \end{aligned}$$

wobei $\mathbf{u} = \mathbf{u}(t_{k+1}), \tau = t_{k+1} - t_k$ $p = p(t_{k+1})$ und $\mathbf{f}_{alt}(t_k) = \nu\Delta\mathbf{u}(t_k) - (\mathbf{u}(t_k)\cdot\nabla)\mathbf{u}(t_k) + \mathbf{f}(t_k)$ sind. Der nichtlineare Term kann wie im Abschnitt 2.2 beschrieben linearisiert werden. Geht man im Zeitpunkt t_{k+1} zu einer schwachen Formulierung über und diskretisiert diese, so führt die Zeitableitung auf einen Term der Gestalt $\frac{1}{\tau}(\mathbf{u}_h, \mathbf{v}_h)$. Die Diskretisierung im Ort erfolgt analog zu den stationären Navier–Stokes Gleichungen. Wir definieren die Matrix

$$M \quad := \quad (\phi_i, \phi_j)_{i,j=1,\dots,N_u} \quad : \quad \text{Massenmatrix.}$$

Die Massenmatrix ist wegen der L^2–Orthogonalität der Basisfunktionen (2.13) eine Diagonalmatrix.

Wie in den vorherigen Abschnitten gehen wir zu den Basisdarstellungen der diskreten Funktionen mit den entsprechenden Koeffizientenvektoren über. Wir betrachten einen Schritt der nichtlinearen Iterationen. Sei (u^n, p^n) die n–te Iterierte der nichtlinearen Iteration zum Zeitpunkt t_{k+1} und (u^K, p^K) die letzte Iterierte zum Zeitpunkt t_k. Gesucht ist eine Korrektur $(\delta u^{n+1}, \delta p^{n+1})$. Diese wird bei der Fixpunktiteration durch die Lösung von

$$\begin{pmatrix} \frac{1}{\tau}M + \theta D + \theta C(\mathbf{u}_h^n) & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} \delta u^{n+1} \\ \delta p^{n+1} \end{pmatrix} = \begin{pmatrix} f_{def}(t_{k+1}) + f_{alt}(t_k) \\ -B^T\mathbf{u}_h^n \end{pmatrix} \tag{2.22}$$

berechnet. Hierbei sind

$$\begin{aligned} f_{def}(t_{k+1}) &= \theta\left[f(t_{k+1}) - (D + C(\mathbf{u}_h^n))u^n - Bp^n\right] - \frac{1}{\tau}Mu^n, \\ f_{alt}(t_k) &= (1-\theta)\left[-(D + C(\mathbf{u}_h^K))u^K + f(t_k)\right] + \frac{1}{\tau}u^K. \end{aligned}$$

BDF(2)

Das BDF(2) Verfahren zur Lösung von $y'(t) = f(t, y(t))$ approximiert $y'(t_{k+1})$ durch

$$y'(t_{k+1}) \approx \frac{\tau_k(2\tau_{k+1} + \tau_k)y(t_{k+1}) - (\tau_{k+1} + \tau_k)^2 y(t_k) + \tau_{k+1}^2 y(t_{k-1})}{\tau_{k+1}\tau_k(\tau_{k+1} + \tau_k)}, \tag{2.23}$$

wobei $\tau_{k+1} = t_{k+1} - t_k$ und $\tau_k = t_k - t_{k-1}$ sind, siehe zum Beispiel bei Hairer, Nørsett und Wanner [HNW87], Seite 401ff.. Das Verfahren konvergiert von zweiter Ordnung, falls eine Konstante c existiert, so daß $\tau_{k+1}/\tau_k \leq c$ ist für $k = 0, 1, \dots$.

Wendet man (2.23) auf die Navier–Stokes Gleichungen an, so erhält man mit $\mathbf{u} = \mathbf{u}(t_{k+1})$ und $p = p(t_{k+1})$

$$\begin{aligned} \frac{2\tau_{k+1}+\tau_k}{\tau_{k+1}(\tau_{k+1}+\tau_k)}\mathbf{u} - \nu\Delta\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p &= \mathbf{f}(t_{k+1}) + \frac{\tau_{k+1}+\tau_k}{\tau_k\tau_{k+1}}\mathbf{u}(t_k) \\ &\quad - \frac{\tau_{k+1}}{\tau_k(\tau_{k+1}+\tau_k)}\mathbf{u}(t_{k-1}) \qquad \text{in } \Omega \\ \nabla\cdot\mathbf{u} &= 0 \qquad \text{in } \Omega \\ \mathbf{u} &= \mathbf{g} \qquad \text{auf } \partial\Omega. \end{aligned}$$

Für äquidistante Zeitschritte, das heißt $\tau_{k+1} = \tau_k = \tau$, lautet diese Formel

$$\begin{aligned} \frac{3}{2\tau}\mathbf{u} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &= \mathbf{f}(t_{k+1}) + \frac{2}{\tau}\mathbf{u}(t_k) - \frac{1}{2\tau}\mathbf{u}(t_{k-1}) \qquad \text{in } \Omega \\ \nabla\cdot\mathbf{u} &= 0 \qquad \text{in } \Omega \\ \mathbf{u} &= \mathbf{g} \qquad \text{auf } \partial\Omega. \end{aligned}$$

Die weitere Vorgehensweise und die Bezeichnungen sind wie bei den modifizierten Ein–Schritt–θ–Methoden. Mit u^{K-1} bezeichnen wir den Koeffizientenvektor der letzten Iterierten im Zeitschritt t_{k-1}. Der einfacheren Notation halber beschränken wir uns auf äquidistante Zeitschritte. Letztlich hat man in jedem Schritt der Fixpunktiteration ein System der Gestalt

$$\begin{pmatrix} \frac{3}{2\tau}M + D + C(\mathbf{u}_h^n) & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} \delta u^{n+1} \\ \delta p^{n+1} \end{pmatrix} = \begin{pmatrix} f_{def}(t_{k+1}) + f_{alt}(t_k, t_{k-1}) \\ -B^T\mathbf{u}_h^n \end{pmatrix} \tag{2.24}$$

zu lösen. Dabei sind

$$f_{def}(t_{k+1}) = f(t_{k+1}) - (D + C(\mathbf{u}_h^n))u^n - Bp^n - \frac{3}{2\tau}Mu^n, \quad f_{alt}(t_k, t_{k-1}) = \frac{2}{\tau}u^K - \frac{1}{2\tau}u^{K-1}.$$

2.6 Die Druckseparation

In diesem Abschnitt wird eine Modifikation der bisherigen Aufgabenstellungen vorgestellt, die bei geringen numerischen Kosten teilweise beträchtliche Verbesserungen der numerischen Resultate nach sich ziehen kann, die sogenannte *Druckseparation.* Diese Verfahrensweise ist ausführlich von Schieweck [Sch96] sowie Dorok [Dor95] beschrieben und analysiert. Für die nichtkonforme P_1/P_0–Finite Elemente Diskretisierung der Stokes–Gleichungen wird von Dorok die a priori Abschätzung

$$|\mathbf{u} - \mathbf{u}_h|_h \le c\left(h\|\mathbf{u}\|_2 + h\nu^{-1}\|p\|_1\right)$$

gezeigt, wobei c unabhängig von ν ist. Es wird deutlich, daß man für kleine ν und Lösungen p mit großer H^1–Norm große Fehler $|\mathbf{u} - \mathbf{u}_h|_h$ zu erwarten hat. Die Idee besteht nun darin, ein zu den Stokes (2.4), Oseen oder Navier–Stokes Gleichungen (2.6) äquivalentes Problem zu lösen, bei dem der gesuchte Druck $\tilde{p}$ eine kleine H^1–Norm besitzt und p mit Hilfe von $\tilde{p}$ berechnet werden kann. Dabei soll das modifizierte stetige System die gleiche Lösung $\mathbf{u}$ wie das Originalsystem besitzen. Ziel der Modifizierung ist ein kleinerer Fehler $|\mathbf{u} - \mathbf{u}_h|_h$. Wir wollen die Vorgehensweise an Hand der Navier–Stokes Gleichungen erläutern.

Sei $p_S \in H^1(\Omega)$ ein vorgegebener Druck mit $p_S \approx p$. Anstelle von (2.6) wird das System

$$\begin{aligned} -\nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla\tilde{p} &= \mathbf{f} - \nabla p_S \qquad \text{in } \Omega \\ \nabla\cdot\mathbf{u} &= 0 \qquad \text{in } \Omega \\ \mathbf{u} &= \mathbf{g} \qquad \text{auf } \partial\Omega \end{aligned}$$

betrachtet, wobei $\tilde{p} = p - p_S$ ist. Mit einer geeigneten Wahl von p_S ist die H^1–Norm von $\tilde{p}$ viel kleiner als die von p. Die Freiheit des Verfahrens liegt nun in der Wahl von p_S. Man kann, wie Schieweck [Sch96], die Impulsbilanz $\nabla p = \mathbf{f} + \nu\Delta\mathbf{u} - (\mathbf{u}\cdot\nabla)\mathbf{u}$ dazu nutzen. Eine Alternative ist die Verwendung von Zwischenergebnisse der bisherigen numerischen Rechnungen. Sei

$$I_{C^0}^{Q_h} : Q_h \mapsto C^0(\Omega), \quad I_{C^0}^{Q_h} p(S) = \sum_{S\in T} \frac{p(T)}{|T|},$$

der Interpolationsoperator, der eine stückweise konstante Funktion p mittels lokaler Mittelung in allen Eckpunkten S der Dreiecke $T \in \mathcal{T}_h$, in eine stückweise lineare, stetige Funktion überführt. Wir nutzen die folgenden Möglichkeiten der Wahl von p_S :

1. Berechne p_S aus $\nabla p_S = \mathbf{f}$.
2. Berechne p_S aus $\nabla p_S = \mathbf{f} - (\mathbf{u}_h^{n_0}\cdot\nabla)\mathbf{u}_h^{n_0}$ im (n_0+1)–ten Schritt der nichtlinearen Iteration.
3. Berechne p_S aus $\nabla p_S = \mathbf{f} - (\mathbf{u}_h^{n}\cdot\nabla)\mathbf{u}_h^{n}$ im $(n+1)$–ten Schritt der nichtlinearen Iteration, $n = 0, 1, 2, \ldots$.
4. Berechne p_S aus $p_S = I_{C^0}^{Q_h} p_h^n$ im $(n+1)$–ten Schritt der nichtlinearen Iteration, $n = 0, 1, 2, \ldots$.
5. Für zeitabhängige Probleme ist $p_h^K(t_k)$ die letzte Iterierte des Druckes im Zeitpunkt t_k. Berechne im Zeitpunkt t_{k+1} den separierten Druck $p_S = I_{C^0}^{Q_h} p_h^K(t_k)$.

Bei der Wahl von 1. oder 2. ist innerhalb der nichtlinearen Iteration einmalig und bei der Nutzung von 3. in jedem Iterationsschritt ein Poisson–Problem mit Neumannrandwerten für den Druck zu lösen. Bei 4. ist in jedem Iterationsschritt und bei 5. vor der nichtlinearen Iteration die oben beschriebene lokale Mittelung notwendig. Die Diskretisierung der Poisson–Probleme erfolgt mit konformen, stückweise linearen Ansatzfunktionen. Am wenigsten aufwendig sind die Varianten 4 und 5, am aufwendigsten die Variante 3.

Kapitel 3

Zur Lösung der linearen Sattelpunktprobleme

Die im vorangegangenen Kapitel vorgestellten Linearisierungen und Diskretisierungen der stationären und instationären Navier–Stokes Gleichungen führen alle auf ein schwach besetztes, lineares Gleichungssystem der Gestalt

$$\mathcal{A}\begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix}\begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} r \\ s \end{pmatrix}. \tag{3.1}$$

In diesem Kapitel wollen wir zuerst einige Aussagen über die Lösbarkeit dieses Gleichungssystems machen und im Anschluß verschiedene Lösungsvarianten diskutiert. Den Abschluß bildet ein Überblick über bekannte theoretische Resultate zur Konvergenz der vorgestellten Verfahren.

3.1 Aussagen zur Lösbarkeit

Dieser Abschnitt befaßt sich mit Konsequenzen für die Lösbarkeit der linearen Systeme (3.1), die aus der Nutzung des Raumes Q_h anstelle von Q_{h0} entstehen, siehe Bemerkung 2.4.5. Zunächst wollen wir einige Eigenschaften der Matrix A zusammenfassen.

Satz 3.1.1 *i) Für die diskreten Stokes Gleichungen (2.17) ist*

$$A = \begin{pmatrix} A_1 & 0 \\ 0 & A_1 \end{pmatrix}$$

symmetrisch und positiv definit.

ii) Für die mittels Fixpunktiteration linearisierten stationären Navier–Stokes Gleichungen (2.20) ist bei Stabilisierung mit scharfem Upwind (1.16) oder Samarskij–Upwind (1.17) die Matrix

$$A = \begin{pmatrix} A_1(\mathbf{u}_h^n) & 0 \\ 0 & A_1(\mathbf{u}_h^n) \end{pmatrix}$$

für alle $\mathbf{u}_h^n$ mit $(\nabla \cdot \mathbf{u}_h^n, q_h) = 0\ \forall q_h \in Q_h$ regulär. Falls die Triangulierung kein stumpfwinkliges Dreieck enthält, so ist bei Nutzung von scharfem Upwind A sogar eine M–Matrix.

iii) Für hinreichend kleine Zeitschritte ist die Matrix A der instationären Diskretisierungen (2.22) und (2.24) regulär.

Beweis: i) Das folgt aus der Definition $A = a_h(\phi_i, \phi_j)$, da diese Bilineraform symmetrisch und positiv definit ist.

ii) Den ersten Teil findet man in Schieweck und Tobiska [ST96b], Theorem 3. Der zweite Teil folgt aus Chapter III, Theorem 3.22 der Monographie von Roos, Stynes und Tobiska [RST96].

iii) Für hinreichend kleine Zeitschritte ist A streng diagonal dominant mit positiven Hauptdiagonalelementen. Die Behauptung folgt durch Anwendung des Kreissatzes von Geršgorin. □

Lemma 3.1.2 *Sei $N_u > N_p$. Für Dirichletprobleme hat die Matrix B den Rangabfall 1 und es ist* $\ker(B) = span\{1_{N_p}\}$.

Beweis: Seien $b_1, \ldots, b_{N_p}$ die Spalten von B. Wir suchen Vektoren $c = (c_1, \ldots, c_{N_p})$, so daß

$$\sum_{i=1}^{N_p} c_i b_i = 0_{N_u} \tag{3.2}$$

gilt.

Sei $\phi_j \in V_h, 1 \leq j \leq N_u$, eine beliebige Basisfunktion. Der Träger sei die Vereinigung der Elemente T_{j1} und T_{j2}, Abbildung 3.1, wobei $c_{j1} \neq 0$ sei. Die j-te Zeile von B besitzt genau zwei

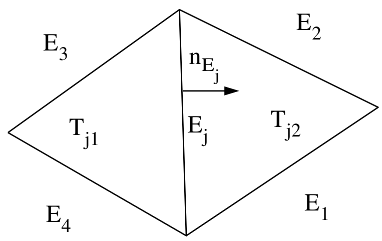


Abbildung 3.1: Träger der Basisfunktion ϕ_j.

Nichtnulleinträge

$$\begin{aligned} b_{j,j1} = &\quad (\psi_{j1}, \nabla \cdot \phi_j)_{T_{j1}} \qquad \text{in der } j1\text{–ten Spalte,} \\ b_{j,j2} = &\quad (\psi_{j2}, \nabla \cdot \phi_j)_{T_{j2}} \qquad \text{in der } j2\text{–ten Spalte,} \end{aligned}$$

wobei ψ_{j1}, ψ_{j2} die charakteristischen Funktionen von T_{j1} und T_{j2} sind. Mittels partieller Integration erhält man

$$\begin{aligned} (\psi_{j1}, \nabla \cdot \phi_j)_{T_{j1}} &= \langle \phi_j \cdot n, 1 \rangle_{\partial T_{j1}} = \langle \phi_j \cdot n_{E_j}, 1 \rangle_{E_j}, \\ (\psi_{j2}, \nabla \cdot \phi_j)_{T_{j2}} &= \langle \phi_j \cdot n, 1 \rangle_{\partial T_{j2}} = -\langle \phi_j \cdot n_{E_j}, 1 \rangle_{E_j}. \end{aligned} \tag{3.3}$$

Die Integrale über die Kanten $E_1, \ldots, E_4$ verschwinden, da $\phi_j \cdot n$ dort in einer Komponente eine lineare Funktion ist, die im Mittelpunkt der jeweiligen Kante den Wert Null annimmt und in der anderen Komponente ohnehin verschwindet. Aus (3.3) folgt notwendig $c_{j1} = c_{j2}$. Da dies für zwei beliebige benachbarte Elemente gilt, folgt mittels Induktion als notwendige Bedingung für (3.2) $c_i = c_{j1} = c\ ,\ i = 1, \ldots, N_p$. □

Lemma 3.1.3 *Seien die Voraussetzungen von Lemma 3.1.2 erfüllt, sei A invertierbar und gelte*

$$x^T A^{-1} x = 0 \quad \Longleftrightarrow \quad x = 0. \tag{3.4}$$

Dann hat das lineare Gleichungssystem (3.1) den Rang $N_u + N_p - 1$, und es ist

$$\ker \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} = \operatorname{span} \begin{pmatrix} 0_{N_u} \\ 1_{N_p} \end{pmatrix}.$$

Beweis: Sei

$$\begin{pmatrix} u \\ p \end{pmatrix} \in \ker \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix},$$

also

$$\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} 0_{N_u} \\ 0_{N_p} \end{pmatrix}. \tag{3.5}$$

Da A invertierbar ist, kann man die erste Gleichung nach u auflösen und in die zweite Gleichung einsetzen. Man erhält

$$B^T A^{-1} B p = 0_{N_p}$$

und daraus

$$p^T B^T A^{-1} B p = (Bp)^T A^{-1} (Bp) = 0.$$

Wegen Voraussetzung (3.4) folgt $Bp = 0_{N_u}$ und mit Lemma 3.1.2 $p = c 1_{N_p}, c \in \mathbb{R}$. Setzt man dies in (3.5) ein, erhält man aus der ersten Gleichung $Au = 0$ und, da A regulär ist, $u = 0$. Damit folgt die Behauptung des Lemmas. □

Lemma 3.1.4 *Ist der symmetrische Anteil $(A^{-1} + A^{-T})/2$ von A^{-1} positiv definit, so ist Bedingung (3.4) erfüllt.*

Beweis: Es ist

$$x^T A^{-1} x = (x^T A^{-1} x)^T = x^T A^{-T} x. \tag{3.6}$$

Da die Matrix $(A^{-1} + A^{-T})/2$ positiv definit ist, gilt $x^T \left((A^{-1} + A^{-T})/2\right) x \neq 0 \quad \forall x \neq 0_{N_u}$. Daraus folgt $x^T A^{-1} x + x^T A^{-T} x \neq 0 \quad \forall x \neq 0_{N_u}$. Wegen (3.6) gilt dies genau dann, falls $x^T A^{-1} x \neq 0 \quad \forall x \neq 0_{N_u}$. □

Satz 3.1.5 *Es seien die Voraussetzungen der Lemmata 3.1.2 und 3.1.3 erfüllt. Das Residuum der kleinsten Quadratelösung von (3.1) ist $c(0_{N_u}, 1_{N_p})^T$ mit $c = \left(1_{N_p}^T s\right)/N_p$.*

Das lineare Gleichungssystem (3.1) ist genau dann lösbar, falls $1_{N_p}^T s = 0$.

Beweis: Wir betrachten die Normalgleichungen zu (3.1)

$$\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix}^T \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix}^T \begin{pmatrix} r \\ s \end{pmatrix}.$$

Dieses System ist lösbar, jede Lösung ist eine Quadratmittellösung. Sei $(u,p)^T$ eine Lösung. Aus der Theorie der Quadratmittelprobleme ist bekannt, siehe zum Beispiel Kiełbasinskiński und Schwetlick [KS88, S. 247],

$$\begin{pmatrix} r - Au - Bp \\ s - B^T u \end{pmatrix} \perp \operatorname{im} \left(\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \right) = \left(\ker \left(\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \right) \right)^\perp = \left(\operatorname{span} \left\{ \begin{pmatrix} 0_{N_u} \\ 1_{N_p} \end{pmatrix} \right\} \right)^\perp,$$

das heißt

$$\begin{pmatrix} r - Au - Bp \\ s - B^T u \end{pmatrix} \in \operatorname{span}\left\{ \begin{pmatrix} 0_{N_u} \\ 1_{N_p} \end{pmatrix} \right\}.$$

Das bedeutet, die Quadratmittellösung erfüllt die ersten Gleichungen. Es gilt $s - B^T u = c 1_{N_p}$, $c \in \mathbb{R}$ und mit Lemma 3.1.2 folgt

$$1_{N_p}^T s = 1_{N_p}^T s - 1_{N_p}^T B^T u = c 1_{N_p}^T 1_{N_p} = c N_p.$$

Ist $1_{N_p}^T s = 0$, dann ist $c = 0$, und $(u,p)^T$ löst (3.1).

Sei $(u,p)^T$ eine Lösung von (3.1). Dann gilt $s = B^T u$, und daraus folgt unter Nutzung von Lemma 3.1.2

$$1_{N_p}^T s = 1_{N_p}^T B^T u = u^T B 1_{N_p} = 0.$$

□

Folgerung 3.1.6 *Es seien die Voraussetzungen des Satzes 3.1.5 erfüllt. Dann hat die Matrix $\tilde{\mathcal{A}}$ Vollrang, die durch das Streichen einer oder mehrerer Spalten von B und der entsprechenden Zeilen von B^T in $\mathcal{A}$ entsteht.*

Beweis: Wie man im Beweis von Lemma 3.1.2 erkennt, hat das Streichen einer Spalte in B zur Folge, daß die restlichen Spalten linear unabhängig sind. Das ändert sich beim Streichen weiterer Spalten nicht. Wir bezeichnen die durch das Streichen hervorgegangene Matrix mit $\tilde{B}$. Dann ist

$$\tilde{\mathcal{A}} = \begin{pmatrix} A & \tilde{B} \\ \tilde{B}^T & 0 \end{pmatrix}.$$

Die Vollrangeigenschaft von $\tilde{\mathcal{A}}$ wird nun analog zum Beweis von Lemma 3.1.3 gezeigt. □

3.2 Druckkorrektur Verfahren

Die hier beschriebenen Methoden werden in der Literatur auch als diskrete Projektionsverfahren bezeichnet. Der wohl bekannteste Vertreter ist das SIMPLE–Verfahren von Patankar und Spalding [PS72].

Die Richardson Iteration

Wir gehen vom linearen Gleichungssystem (3.1) aus. Löst man die erste Gleichung nach u auf und setzt dies in die zweite Gleichung ein, so erhält man das Schur–Komplement System bezüglich p:

$$B^T A^{-1} B p = B^T A^{-1} r - s.$$

Diese Gleichung kann man in eine Fixpunktgleichung der Gestalt

$$p = p + \alpha (B^T C^{-1} B)^{-1} (-B^T A^{-1} B p + B^T A^{-1} r - s) \tag{3.7}$$

umformen, wobei C eine reguläre $(N_u \times N_u)$–Matrix ist und $\alpha > 0$ ein Dämpfungsparameter. Die Fixpunktgleichung (3.7) kann mittels einer Fixpunktiteration (Richardson Iteration)

$$p^0 = 0, \quad p^{m+1} = p^m + \alpha (B^T C^{-1} B)^{-1} (-B^T A^{-1} B p^m + B^T A^{-1} r - s) \quad m = 0, 1, \ldots, \tag{3.8}$$

bei geeigneter Wahl von C und α gelöst werden. Dabei muß man in jedem Iterationsschritt die Lösung eines linearen Gleichungssystems der Gestalt

$$B^T C^{-1} Bx = -B^T A^{-1} Bp^m + B^T A^{-1} r - s$$

berechnen. Wird die Iteration nach Schritt $\mathsf{M}-1$ beendet, kann man eine von p^{M} abhängige Approximation u^{M} von u mittels

$$u^{\mathsf{M}} = A^{-1}(r - Bp^{\mathsf{M}}) \tag{3.9}$$

berechnen.

Zur Durchführung der Iteration (3.8) benötigt man die Inverse von A. Diese approximieren wir durch eine Matrix $A_{\gamma,j}^{-1} \approx A^{-1}$, wobei $A_{\gamma,j}^{-1}$ für j Mehrgitterzyklen angewandt auf ein System $Ax = b$ steht. Eine ausführliche Beschreibung des Mehrgitterverfahrens befindet sich in Abschnitt 3.3. Anstelle von (3.8) führen wir in der Praxis die Richardson Iteration

$$p^0 = 0, \quad p^{m+1} = p^m + \alpha(B^T C^{-1} B)^{-1}(-B^T A_{\gamma,j}^{-1} Bp^m + B^T A_{\gamma,j}^{-1} r - s) \quad m = 0, 1, \ldots, \tag{3.10}$$

durch, wobei in jedem Iterationsschritt ein lineares System der Gestalt

$$B^T C^{-1} Bx = -B^T A_{\gamma,j}^{-1} Bp^m + B^T A_{\gamma,j}^{-1} r - s \tag{3.11}$$

und der Dimension N_p zu lösen ist.

Zur Lösung von (3.9) muß noch einmal ein Gleichungssystem mit der Matrix A gelöst werden. Diesen Schritt kann man sich sparen, falls man die Approximation an u durch

$$\tilde{u}^{\mathsf{M}} = A_{\gamma,j}^{-1}(r - Bp^{\mathsf{M}-1}) + \frac{1}{\alpha} C^{-1} B(p^{\mathsf{M}} - p^{\mathsf{M}-1}) \tag{3.12}$$

berechnet. Der Term $A_{\gamma,j}^{-1} r$ wird bereits vor der Iteration (3.10) und der Term $A_{\gamma,j}^{-1} Bp^{\mathsf{M}-1}$ im letzten Iterationsschritt von (3.10) berechnet. Dieses Verfahren wurde von Turek [Tur96] vorgeschlagen und legt besonderes Gewicht auf die Erfüllung der diskreten Massenbilanz, während (3.9) Gewicht auf die Erfüllung der diskreten Impulsbilanz legt.

Lemma 3.2.1 *Falls das Gleichungssystem (3.11) exakt gelöst wird, so gilt* $B^T \tilde{u}^{\mathsf{M}} = s$.

Beweis: Unter Ausnutzung von (3.12) und (3.10) ergibt sich

$$\begin{aligned} B^T \tilde{u}^{\mathsf{M}} &= B^T A_{\gamma,j}^{-1}(r - Bp^{\mathsf{M}-1}) + \frac{1}{\alpha} B^T C^{-1} B(p^{\mathsf{M}} - p^{\mathsf{M}-1}) \\ &= B^T A_{\gamma,j}^{-1} r - B^T A_{\gamma,j}^{-1} Bp^{\mathsf{M}-1} \\ &\quad + \frac{1}{\alpha} B^T C^{-1} B \left(-\alpha (B^T C^{-1} B)^{-1} (-B^T A_{\gamma,j}^{-1} Bp^{\mathsf{M}-1} + B^T A_{\gamma,j}^{-1} r - s) \right) \\ &= s. \end{aligned}$$

□

Bemerkung 3.2.2 Diskrete Divergenzfreiheit bei Nutzung von (3.12). Für die Stokes Gleichungen ist $s = 0$. Damit ist $\tilde{u}^{\mathsf{M}}$ diskret divergenzfrei. Für die stationären und instationären Navier–Stokes Gleichungen ist $s = -B^T u^n$, wobei u^n die gegenwärtige Iterierte der nichtlinearen Iteration und $\tilde{u}^{\mathsf{M}} = \delta u^{n+1}$ die Korrektur sind. Dann ist die nächste Iterierte $u^{n+1} = u^n + \tilde{u}^{\mathsf{M}}$ ebenfalls diskret divergenzfrei.

Die Freiheit im Verfahren (3.10) liegt in der Wahl von C im Vorkonditionierer. In den nächsten Abschnitten und bei den numerischen Testrechnungen (Kapitel 9 und 10) werden wir uns mit folgenden Fragen beschäftigen :

- Welche numerischen Verfahren sind zur Lösung von (3.11) geeignet?
- Wie kann C gewählt werden?
- Wie beeinflußt die Wahl von C die Wahl des Dämpfungsparameters α?
- Wie beeinflußt die Singularität von $B^TC^{-1}B$ die numerischen Verfahren?
- Was kosten die numerischen Verfahren?

Symmetrische Vorkonditionierer

Bei den instationären Problemen (2.22) und (2.24) mit kleinen Zeitschritten dominiert in A die Diskretisierung von $\mathbf{u}_t$. Dies ist eine Diagonalmatrix. Deshalb liegt es in diesem Falle nahe, als Approximation für A^{-1} eine Diagonalmatrix C^{-1} zu wählen. Einige Möglichkeiten sind

- die Einheitsmatrix $C = I_{N_u}$,
- die Massenmatrix $C = M$,
- die Diagonale von A, also $C = \operatorname{diag}(A)$.

Eine andere Möglichkeit besteht darin, in der Richardson Iteration (3.10) keinen Vorkonditionierer zu verwenden und nur zu dämpfen, das heißt $B^TC^{-1}B = I_{N_p}$ zu wählen. Führt man nur eine Iteration aus, so resultiert dies in den bekannten Uzawa–Algorithmus, siehe [BF91].

In all diesen Fällen ist $B^TC^{-1}B$ eine symmetrische, positiv semidefinite Matrix. Sie repräsentiert ein diskretes (gewichtetes) Laplace Problem für den Druck mit Neumannrandbedingungen. Es gilt $\ker(B^TC^{-1}B) = \operatorname{span}\{1_{N_p}\}$, vergleiche Lemma 3.1.2. Das Festlegen des Wertes einer einzelnen Komponente des Vektors p führt auf ein positiv definites Problem für die übrigen Komponenten. Lösungsverfahren dafür sind :

- ein Mehrgitterverfahren für die Druckunbekannten,
- das Verfahren der konjugierten Gradienten (CG) von Stiefel und Hestenes [HS52].

Das Mehrgitterverfahren für die Druckunbekannten wird ausführlich im Abschnitt 3.4 beschrieben.

Unsymmetrische Vorkonditionierer

Die Matrix C^{-1} stellt eine Approximation an A^{-1} dar. Insbesondere beim stationären Problem (2.20) sowie auch bei relativ großen Zeitschritten in den instationären Problemen hat man im allgemeinen keine dominierende Diagonale in A. Deswegen ist es notwendig, nun eine bessere Approximation an A^{-1}, als die im vorangegangenen Abschnitt vorgeschlagenen, zu nutzen.

Eine gute Approximation $A^{-1}_{\gamma,j}$ an A^{-1} benötigen wir bereits in (3.10) beziehungsweise (3.11). Aus diesem Grund bietet sich die Wahl von $C^{-1} = A^{-1}_{\gamma,j_1}$ an. Da A im allgemeinen nicht symmetrisch ist, sind dies A^{-1}_{γ,j_1} und $B^TC^{-1}B$ ebenfalls nicht. Aus diesem Grunde benötigen wir zur Lösung von (3.11) ein zuverlässiges Verfahren zur Lösung von linearen Systemen mit unsymmetrischen Matrizen. Wir haben GMRES(restart) von Saad und Schultz [SS86] gewählt. Wählt man $C^{-1} = A^{-1}_{\gamma,j}$, also $j = j1$, so entspricht das dem Lösen von

$$B^TA^{-1}_{\gamma,j}Bp = B^TA^{-1}_{\gamma,j}r - s$$

mittels GMRES(restart).

3.3 Das Mehrgitterverfahren für die Geschwindigkeitsunbekannten

Ein lineares Mehrgitterverfahren für die Geschwindigkeitsunbekannten benutzen wir bei der Berechnung von $A_{\gamma,j}^{-1}$ in (3.10) und im Falle der Wahl $C^{-1} = A_{\gamma,j_1}^{-1}$. Der erste Index γ gibt die Art der Mehrgitterzyklen an:
- $\gamma = 1$ steht für den V–Zyklus,
- $\gamma = 2$ steht für den W–Zyklus.

Der zweite Index gibt die Anzahl der Zyklen an.

In diesem Abschnitt werden die einzelnen Bestandteile des Mehrgitterverfahrens erläutert. Eine schematische Darstellung des Verfahrens findet man im Abschnitt 4.3.

Ein Mehrgitterverfahren beinhaltet im wesentlichen fünf Komponenten:
- den Glätter,
- die Restriktion,
- die Prolongation,
- den Grobgitterlöser,
- die Schrittweitenkontrolle.

Die Glätter

Glätter sind Iterationsverfahren, die auf jedem Gitterlevel die jeweils hochfrequenten Fehleranteile glätten sollen. Da wir bei einer parallelen Programmabarbeitung Datenaustausch innerhalb einer Glättungsiteration vermeiden wollen, bieten sich Block–Jacobi–Glätter an. Die Blöcke werden durch die Verteilung der Unbekannten auf die Prozessoren bestimmt. Die durch die Blöcke gegebenen Systeme werden durch ein weiteres Iterationsverfahren gelöst, welches im allgemeinen bessere Konvergenzeigenschaften als das Jacobi–Verfahren hat. Wir verwenden
- Block–Jacobi Iteration mit innerer Gauß–Seidel Iteration,
- Block–Jacobi Iteration mit innerer ILU_β–Iteration.

Ist die Programmabarbeitung seriell, so existiert nur ein Block, nämlich die gesamte Matrix. Die Glätter sind dann Gauß–Seidel Iteration beziehungsweise ILU_β–Iteration. Die implementierte ILU_β–Iteration erlaubt keine zusätzliche Auffüllung der bestehenden schwach besetzten Matrixstruktur. Die Beträge, der sich während der Zerlegung außerhalb der Matrixstruktur ergebenden Elemente, können, multipliziert mit β, zur Hauptdiagonale von U addiert werden. Hier ist U die obere Dreiecksmatrix der unvollständigen Zerlegung $A \approx LU$. Glätter dieser Gestalt wurden intensiv von Wittum [Wit89a] studiert.

Die Restriktion

Seien $\mathcal{T}_l$ und $\mathcal{T}_{l+1}$ zwei aufeinanderfolgende Gitter in der Hierarchie des Mehrgitters. Die damit verbundenen diskreten Räume der Geschwindigkeit seien $V_{h,l}$ und $V_{h,l+1}$. Wir haben als Restriktion $R[\mathbf{u}]_l^{l+1}$ die L^2–Projektion gewählt:

$$\begin{array}{rccl} R[\mathbf{u}]_l^{l+1}: & V_{h,l+1} & \mapsto & V_{h,l} \\ & \mathbf{u}_h & \to & R[\mathbf{u}]_l^{l+1}\mathbf{u}_h \text{ mit } \left(R[\mathbf{u}]_l^{l+1}\mathbf{u}_h - \mathbf{u}_h, \mathbf{v}_h\right) = 0 \quad \forall \mathbf{v}_h \in V_{h,l}. \end{array} \tag{3.13}$$

Sei $\{\phi_{i,l}\}$ eine Basis von $V_{h,l}$ und $\{\phi_{i,l+1}\}$ von $V_{h,l+1}$. Es reicht aus, wenn (3.13) für alle Elemente der Menge $\{\phi_{i,l}\}$ erfüllt ist. Seien

$$\mathbf{u}_h = \sum_i \alpha_i \phi_{i,l+1}, \qquad R[\mathbf{u}]_l^{l+1}\mathbf{u}_h = \sum_j \beta_j \phi_{j,l}.$$

Dann gilt

$$0 = \left(\sum_j \beta_j \phi_{j,l} - \sum_i \alpha_i \phi_{i,l+1}, \phi_{k,l} \right) \quad \forall \phi_{k,l} \in \{\phi_{i,l}\}$$

und somit wegen der L_2-Orthogonalität der Basisfunktionen

$$\beta_k = \frac{(\sum_i \alpha_i \phi_{i,l+1}, \phi_{k,l})}{(\phi_{k,l}, \phi_{k,l})} = \frac{\sum_i \alpha_i \sum_{T \in \mathcal{T}_{l+1}} (\phi_{i,l+1}, \phi_{k,l})_T}{(\phi_{k,l}, \phi_{k,l})}. \tag{3.14}$$

Der Zähler verschwindet bei vorgegebenem $\phi_{k,l}$ für diejenigen $\phi_{i,l+1}$, für die meas$(\phi_{i,l+1} \cap \phi_{k,l}) = 0$ ist. Damit sieht man, daß es sich bei der L^2–Projektion um eine lokale Operation handelt. Wir nennen ein Element auf Level l *Vaterelement* und die durch Verfeinerung aus ihm hervorgegangenen Elemente auf Level $l+1$ *Sohnelemente*. Auf Grund der verschiedenen Abschlußtechniken (Abbildung 3.2) bei adaptiver Verfeinerung sind gewisse Fallunterscheidungen bezüglich der Lage von Sohnelementen im Vaterelement erforderlich, siehe Abbildung 3.3. Aus den Verfeinerungstechniken ist ersichtlich, daß ein Sohnelement entweder ein Viertel (Typ 1,2,4), die Hälfte (Typ 3) oder das gleiche Maß wie sein Vater haben kann.

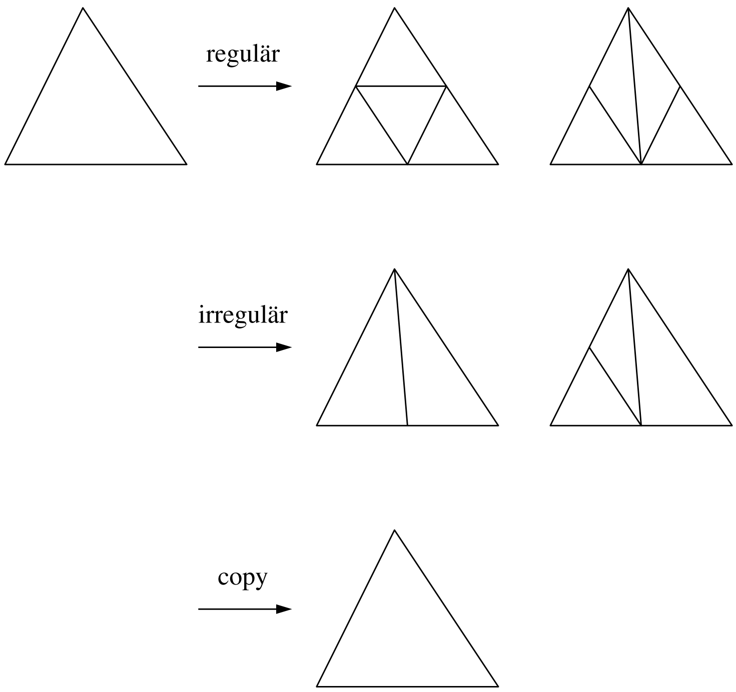


Abbildung 3.2: Die Verfeinerungsregeln.

Wir betrachten die Restriktion aus der Sicht der Sohnelemente. Aus Darstellung (3.14) sieht man, daß man die Werte β_k mittels einer Schleife über die Sohnelemente akkumulieren kann. Jedes Sohnelement T kennt sein Maß $|T|$ und die Funktionswerte in den Knoten auf seinen Kanten. Die Skalarprodukte $(\phi_{i,l+1}, \phi_{k,l})_T$ lassen sich mit Hilfe der Seitenmittelpunktregel einfach berechnen:

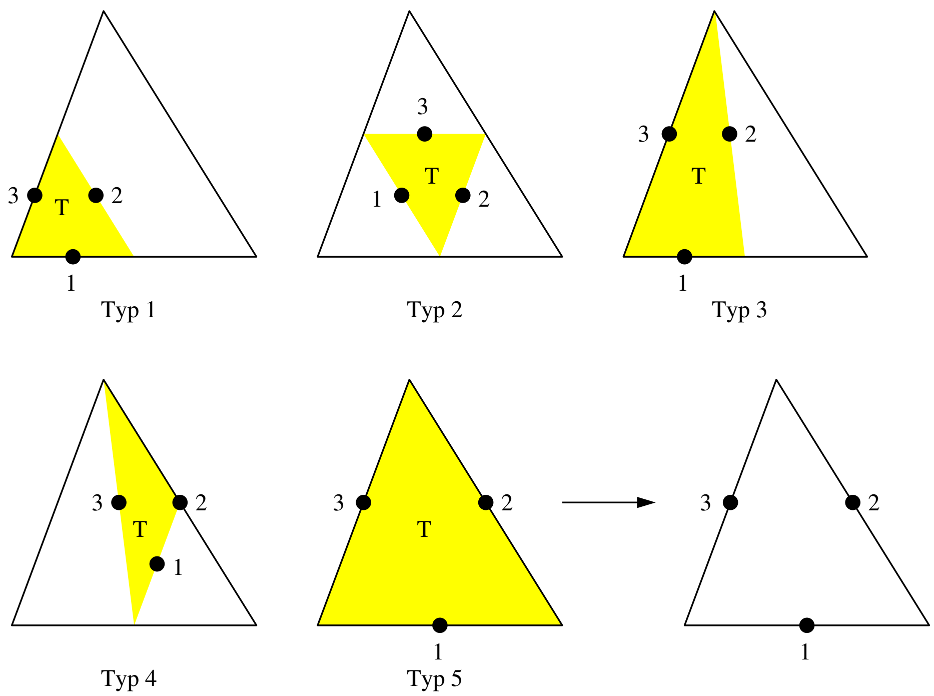


Abbildung 3.3: Die möglichen Lagen von Sohnelementen in ihrem Vaterelement.

$$\begin{array}{llllllllll}
\text{Typ 1:} & (\phi_{1,l+1},\phi_{1,l})_T & = & |T|/3, & (\phi_{1,l+1},\phi_{2,l})_T & = & -|T|/6, & (\phi_{1,l+1},\phi_{3,l})_T & = & |T|/6, \\
& (\phi_{2,l+1},\phi_{1,l})_T & = & |T|/6, & (\phi_{2,l+1},\phi_{2,l})_T & = & 0, & (\phi_{2,l+1},\phi_{3,l})_T & = & |T|/6, \\
& (\phi_{3,l+1},\phi_{1,l})_T & = & |T|/6, & (\phi_{3,l+1},\phi_{2,l})_T & = & -|T|/6, & (\phi_{3,l+1},\phi_{3,l})_T & = & |T|/3;
\end{array}$$

$$\begin{array}{llllllllll}
\text{Typ 2:} & (\phi_{1,l+1},\phi_{1,l})_T & = & |T|/6, & (\phi_{1,l+1},\phi_{2,l})_T & = & 0, & (\phi_{1,l+1},\phi_{3,l})_T & = & |T|/6, \\
& (\phi_{2,l+1},\phi_{1,l})_T & = & |T|/6, & (\phi_{2,l+1},\phi_{2,l})_T & = & |T|/6, & (\phi_{2,l+1},\phi_{3,l})_T & = & 0, \\
& (\phi_{3,l+1},\phi_{1,l})_T & = & 0, & (\phi_{3,l+1},\phi_{2,l})_T & = & |T|/6, & (\phi_{3,l+1},\phi_{3,l})_T & = & |T|/6;
\end{array}$$

$$\begin{array}{llllllllll}
\text{Typ 3:} & (\phi_{1,l+1},\phi_{1,l})_T & = & |T|/3, & (\phi_{1,l+1},\phi_{2,l})_T & = & -|T|/6, & (\phi_{1,l+1},\phi_{3,l})_T & = & |T|/6, \\
& (\phi_{2,l+1},\phi_{1,l})_T & = & 0, & (\phi_{2,l+1},\phi_{2,l})_T & = & |T|/6, & (\phi_{2,l+1},\phi_{3,l})_T & = & |T|/6, \\
& (\phi_{3,l+1},\phi_{1,l})_T & = & 0, & (\phi_{3,l+1},\phi_{2,l})_T & = & 0, & (\phi_{3,l+1},\phi_{3,l})_T & = & |T|/3;
\end{array}$$

$$\begin{array}{llllllllll}
\text{Typ 4:} & (\phi_{1,l+1},\phi_{1,l})_T & = & |T|/6, & (\phi_{1,l+1},\phi_{2,l})_T & = & -|T|/6, & (\phi_{1,l+1},\phi_{3,l})_T & = & |T|/6, \\
& (\phi_{2,l+1},\phi_{1,l})_T & = & -|T|/6, & (\phi_{2,l+1},\phi_{2,l})_T & = & |T|/3, & (\phi_{2,l+1},\phi_{3,l})_T & = & |T|/6, \\
& (\phi_{3,l+1},\phi_{1,l})_T & = & 0, & (\phi_{3,l+1},\phi_{2,l})_T & = & |T|/6, & (\phi_{3,l+1},\phi_{3,l})_T & = & |T|/6;
\end{array}$$

$$\begin{array}{llllllllll}
\text{Typ 5:} & (\phi_{1,l+1},\phi_{1,l})_T & = & |T|/3, & (\phi_{1,l+1},\phi_{2,l})_T & = & 0, & (\phi_{1,l+1},\phi_{3,l})_T & = & 0, \\
& (\phi_{2,l+1},\phi_{1,l})_T & = & 0, & (\phi_{2,l+1},\phi_{2,l})_T & = & |T|/3, & (\phi_{2,l+1},\phi_{3,l})_T & = & 0, \\
& (\phi_{3,l+1},\phi_{1,l})_T & = & 0, & (\phi_{3,l+1},\phi_{2,l})_T & = & 0, & (\phi_{3,l+1},\phi_{3,l})_T & = & |T|/3.
\end{array}$$

Mit Hilfe dieser Beziehungen läßt sich der Zähler in (3.14) für alle k berechnen. Eine weitere Schleife über die Knoten auf Level l ist notwendig, um β_k zu berechnen. Diese Vorgehensweise vereinfacht sich für gleichmäßige Verfeinerungen ein wenig. In Abbildung 3.4 ist zu sehen, daß sich der restringierte Wert eines Knotens auf Level l als Linearkombination von 14 Werten auf Level $l+1$ berechnet. Bei Elementen mit unterschiedlich großem Inhalt setzt sich das Gewicht der beiden Konten auf der Sohnkante, in Abbildung 3.4 die Knoten mit dem Gewicht 1/4, aus zwei unterschiedlich großen Anteilen zusammen, die getrennt berechnet werden.

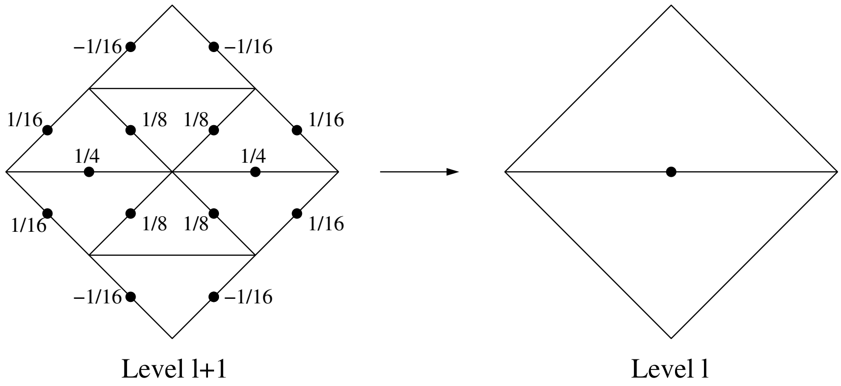


Abbildung 3.4: Die Restriktion als lokale Mittelung und die Gewichte bei gleich großen Elementen.

Die Prolongation

Für die Prolongation $P[\mathbf{u}]_{l+1}^{l}$ benutzen wir ebenfalls die L_2–Projektion:

$$\begin{array}{rcll} P[\mathbf{u}]_{l+1}^{l}: & V_{h,l} & \mapsto & V_{h,l+1} \\ & \mathbf{u}_h & \to & P[\mathbf{u}]_{l+1}^{l}\mathbf{u}_h \text{ mit } \left(P[\mathbf{u}]_{l+1}^{l}\mathbf{u}_h - \mathbf{u}_h, \mathbf{v}_h\right) = 0 \quad \forall \mathbf{v}_h \in V_{h,l+1}. \end{array} \tag{3.15}$$

Seien wie oben $\{\phi_{i,l}\}$ eine Basis von $V_{h,l}$ und $\{\phi_{i,l+1}\}$ von $V_{h,l+1}$. Es reicht aus, wenn (3.15) für alle Elemente der Menge $\{\phi_{i,l+1}\}$ erfüllt ist. Wir nutzen wieder die Basisdarstellungen

$$\mathbf{u}_h = \sum_j \beta_j \phi_{j,l}, \qquad P[\mathbf{u}]_{l+1}^{l}\mathbf{u}_h = \sum_i \alpha_i \phi_{i,l+1}.$$

Aus der Definition der L_2–Projektion und wegen der L_2–Orthogonalität der Basisfunktionen erhält man

$$\alpha_k = \frac{(\sum_j \beta_j \phi_{j,l}, \phi_{k,l+1})}{(\phi_{k,l+1}, \phi_{k,l+1})} = \frac{\sum_j \beta_j \sum_{T \in \mathcal{T}_{l+1}} (\phi_{j,l}, \phi_{k,l+1})_T}{(\phi_{k,l+1}, \phi_{k,l+1})} \quad \forall \phi_{k,l+1} \in \{\phi_{i,l+1}\}. \tag{3.16}$$

Die lokalen Integrale in (3.16) lassen sich nach den oben angegebenen Formeln auswerten. Mit einer Schleife über die Elemente des feinen Gitters $\mathcal{T}_{l+1}$ läßt sich der Zähler für jedes α_k berechnen und anschließend mit einer Schleife über die Knoten von $\mathcal{T}_{l+1}$ jedes α_k selber. Den Abschluß der Prolongation bildet die Addition der prolongierten Werte (eventuell gedämpft) zu den Ergebnissen des Vorglättens.

Mit Hilfe der Definitionen der L_2–Projektionen (3.13), (3.15) läßt sich die Adjungiertheit der Gittertransferoperationen einfach zeigen:

$$(R[\mathbf{u}]_{l}^{l+1}\mathbf{u}_h, \mathbf{v}_h) = (\mathbf{u}_h, \mathbf{v}_h) = (\mathbf{u}_h, P[\mathbf{u}]_{l+1}^{l}\mathbf{v}_h) \quad \forall \mathbf{u}_h \in V_{h,l+1}, \mathbf{v}_h \in V_{h,l}.$$

Der Grobgitterlöser

Der Grobgitterlöser hängt von der Größe des Grobgittersystems ab. Für kleine Systeme, bei denen das Grobgitter auf einem Prozessor vorhanden ist, nehmen wir einen direkten Löser (LU–Zerlegung mit Spaltenpivotsuche). Ist das Grobgitter über mehrere Prozessoren verteilt, dann benutzen wir ein iteratives Verfahren: entweder Block–Jacobi–Iterationen mit inneren Gauß–Seidel Iterationen oder ILU_β–Iterationen. Dabei wird entweder das Residuum um einen vorher festgelegten Faktor reduziert oder eine vorgegebene Anzahl von Iterationsschritten ausgeführt, siehe Abschnitt 4.3.

Die Schrittweitenkontrolle

Die Schrittweitenkontrolle ist kein notwendiger Bestandteil eines Mehrgitterverfahrens. Ihre Anwendung kann jedoch die Effizienz des Mehrgitterverfahrens wesentlich erhöhen. Es ist möglich, sie auf allen Gittern oder auch nur auf dem feinsten Gitter einzusetzen. Wir verwenden die zweite Variante.

Sei ein System der Form $Au = b$ zu lösen, die gegenwärtige Iterierte sei u^i, und δu^{i+1} sei die vom Mehrgitterverfahren vorgeschlagene Korrektur. Diese wird nur als Korrekturrichtung akzeptiert und

$$u^{i+1} = u^i + \kappa \delta u^{i+1}$$

gesetzt, wobei $\kappa \in \mathbb{R}$ so gewählt wird, daß die euklidische Norm des Residuums

$$\Big((b - Au^{i+1})^T (b - Au^{i+1})\Big)^{1/2}$$

minimiert wird. Diese Minimierungsaufgabe für κ hat die Lösung

$$\kappa = \frac{(b - Au^i)^T (A\delta u^{i+1})}{(A\delta u^{i+1})^T (A\delta u^{i+1})}. \tag{3.17}$$

Eine Schrittweitenkontrolle wurde von Peisker und Braess [PB87] für das nichtkonforme Morley Element und von Braess und Verfürth [BV90] für das nichtkonforme P_1–Element vorgeschlagen. In beiden Arbeiten werden symmetrische, positiv definite Probleme betrachtet, und die Schrittweitenkontrolle erfolgt in bezug auf die Minimierung eines Energiefunktionals. Analytische Untersuchungen für konforme Diskretisierungen findet man bei Reusken [Reu91].

Adaptivität

Die Möglichkeit lokaler Netzverfeinerungen wirft bei der Verwendung von Mehrgitterverfahren neue Fragen auf, da einige Level nun nicht mehr das gesamte Gebiet Ω überdecken. Insbesondere ist festzulegen, auf welche Knoten auf den einzelnen Leveln der Glätter anzuwenden ist. Diese Festlegung bestimmt, in welchen Knoten sich das Residuum ändert und demzufolge nach der Glättung neu berechnet werden muß.

Wir verwenden eine Glättung, die auf eine Umgebung der lokal verfeinerten Gebiete angewandt wird. Diese Vorgehensweise wurde ursprünglich von Rivara [Riv84] vorgeschlagen, von Bastian [Bas96] weiter untersucht und in einem größeren Programmpaket (*ugp*) implementiert. Der Ausgangspunkt unserer Implementation ist ein Derivat dieses Paketes, nämlich *ugp*1.0, siehe vor allem Kapitel 4. Die Verfeinerungsroutinen dieses Paketes liefern folgerichtig ein lokales Gitter, welches eine Umgebung der verfeinerten Elemente enthält. Dieses ist in Abbildung 3.5 schattiert dargestellt.

Die Vorgehensweise bei der Glättung wird an Hand des Modelles in Abbildung 3.5 erklärt. Dort wurde ein Element regulär verfeinert und die lokale Verfeinerung mit einer irregulären Verfeinerung abgeschlossen. Es seien in Abbildung 3.5 keine Knoten auf dem Dirichletrand vorhanden. Als nach einer Verfeinerung neu entstandene Knoten werden all jene angesehen, die mit einem regulär verfeinerten Element in Verbindung stehen. Diese sind mit einem gefüllten Quadrat gekennzeichnet. Die direkten Nachbarknoten, auf denen ebenfalls geglättet wird, sind mit einem gefüllten Kreis versehen. In allen Knoten des Levels mit lokal verfeinertem Gitter, die mit einer Quadratlinie gekennzeichnet sind, wird lediglich das Residuum berechnet. In den direkten Nachbarknoten der mit dem gefüllten Kreis gekennzeichneten Knoten ändert es sich im Laufe der Glättungsiteration. Bei der Restriktion wird somit ein Wert an das nächsttiefere Level übergeben, der der Glättung auf dem lokal verfeinerten Level Rechnung trägt. Die Knoten auf dem Rand des lokal verfeinerten Gebietes, die mit einer Kreislinie gekennzeichnet sind,

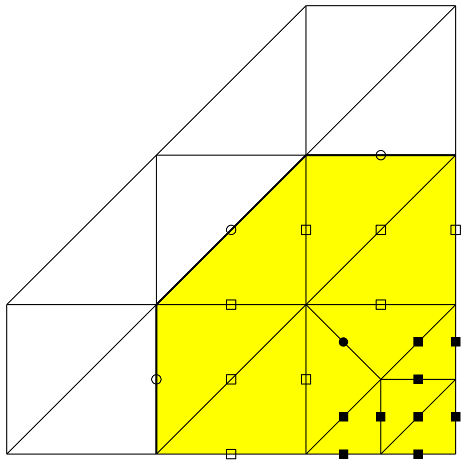


Abbildung 3.5: Lokale Verfeinerung und Klassifizierung der Knoten.

werden zur Berechnung des Residuums benötigt. Auf ihnen selbst wird kein neues Residuum berechnet. Der Gittertransfer läßt sich wie oben beschrieben auch auf lokal verfeinerten Gittern durchführen.

3.4 Das Mehrgitterverfahren für die Druckunbekannten

Ein Mehrgitterverfahren für die Unbekannten des Druckes verwenden wir bei der Lösung von (3.11), falls $B^TC^{-1}B$ eine symmetrische Matrix ist. Im folgenden sei C eine Diagonalmatrix $C = \mathrm{diag}(c_i), i = 1, \ldots, N_u$. Wir beschränken uns abermals auf die Beschreibung der Mehrgitterkomponenten. Eine schematische Darstellung des Mehrgitterverfahrens findet sich im Abschnitt 4.4.

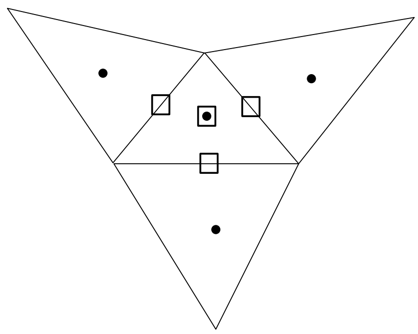
Die explizite Speicherung der Iterationsmatrix

Die Matrix $B^TC^{-1}B$ ist zunächst nur implizit vorhanden. Im Mehrgitterverfahren wird zur Berechnung des Residuums das Produkt dieser Matrix mit einem Vektor ständig benötigt. Es ist ineffizient, dieses Produkt jedes Mal als Hintereinanderausführung von drei Matrix–Vektor Multiplikationen zu berechen. Außerdem würde die Wahl eines guten Glätters Probleme bereiten. Man hat also $B^TC^{-1}B$ explizit bereitzustellen.

Dieses Produkt beschreibt einen diskreten, gewichteten Laplace Operator. Deshalb wird oft an Stelle von $B^TC^{-1}B$ ein vor der Mehrgitteriteration assemblierter gewichteter Laplace Operator für die Richardson–Iteration verwendet. Dieser Weg scheitert hier jedoch daran, daß wir nur stückweise konstante Ansatzfunktionen für den Druck verwenden. Es ist allerdings möglich, die Elemente von $B^TC^{-1}B$ auszurechnen und geschickt zu speichern. Es gilt

$$(B^TC^{-1}B)_{ij} = \left(\sum_{k=1}^{N_u} b_{ki} b_{kj} c_k^{-1} \right), i, j = 1, \ldots, N_p. \tag{3.18}$$

Eine Zeile (Spalte) von $B^TC^{-1}B$ hat im allgemeinen vier Nichtnulleinträge. Jedes Element ist durch $B^TC^{-1}B$ nur mit seinen unmittelbaren Nachbarn gekoppelt. Da $B^TC^{-1}B$ symmetrisch ist, braucht man die Nichthauptdiagonalelemente nur einmal zu berechnen und speichert sie an den Verbindungen der Nachbarelemente, den Kanten, ab, siehe Abbildung 3.6.



□ **Matrixeinträge**

Abbildung 3.6: Die explizite Speicherung von $B^T C^{-1} B$.

Der Glätter

Da die Matrixelemente explizit vorliegen, bereitet es keine Schwierigkeiten, einen gängigen Glätter zu nutzen. Wir verwenden einen Block–Jacobi Glätter mit inneren Gauß-Seidel Iterationen.

Der Gittertransfer

Für den Gittertransfer wählen wir wiederum L^2–Projektionen. Seien $\mathcal{T}_{l+1}$ und $\mathcal{T}_l$ aufeinanderfolgende Gitter der Gitterhierarchie und $Q_{h,l+1}, Q_{h,l}$ die zugehörigen diskreten Räume des Druckes mit den Basen $\{\psi_{i,l+1}\}$ und $\{\psi_{i,l}\}$. Die Restriktion ist definiert als

$$\begin{array}{rcll} R[p]_l^{l+1}: & Q_{h,l+1} & \mapsto & Q_{h,l} \\ & p_h & \rightarrow & R[p]_l^{l+1} p_h \text{ mit } (p_h - R[p]_l^{l+1} p_h, q_h) = 0 \quad \forall q_h \in Q_{h,l}. \end{array} \tag{3.19}$$

Seien

$$p_h = \sum_i \alpha_i \psi_{i,l+1}, \qquad R[p]_l^{l+1} p_h = \sum_j \beta_j \psi_{j,l}.$$

Es ist wieder ausreichend, wenn (3.19) für alle Basisfunktionen gilt. Mit Hilfe der Basisdarstellungen von p_h und $R[p]_l^{l+1} p_h$ sowie $(\psi_{j,l}, \psi_{i,l}) = c\delta_{ij}$ erhält man

$$\beta_k = \frac{(\sum_i \alpha_i \psi_{i,l+1}, \psi_{k,l})}{(\psi_{k,l}, \psi_{k,l})} = \frac{\sum_i \alpha_i \sum_{T \in \mathcal{T}_{l+1}} (\psi_{i,l+1}, \psi_{k,l})_T}{(\psi_{k,l}, \psi_{k,l})}. \tag{3.20}$$

Da $(\psi_{i,l+1}, \psi_{k,l})$ nur für Basisfunktionen $\psi_{i,l+1}$ von Söhnen des Elementes nicht verschwindet, auf dem $\psi_{k,l}$ Basisfunktion ist, hat die Summe im Zähler höchstens vier Summanden die ungleich Null sind. Je nach Verfeinerungsregel ist $(\psi_{i,l+1}, \psi_{k,l}) = c\,(\psi_{k,l}, \psi_{k,l})$ mit $c \in \{1/4, 1/2, 1\}$ und die L^2–Projektion ist eine lokale Mittelung. Für gleichmäßige Verfeinerungen ist $\beta_k = \sum_{i=1}^4 \alpha_i$ für $\text{meas}(\text{supp}\psi_{i,l+1} \cap \text{supp}\psi_{k,l}) > 0$.

Für die Prolongation $P[p]_{l+1}^l$ wählen wir ebenfalls die L^2–Projektion. Mit den obigen Bezeichnungen ist das

$$\begin{array}{rcll} P[p]_l^{l+1}: & Q_{h,l} & \mapsto & Q_{h,l+1} \\ & p_h & \rightarrow & P[p]_{l+1}^l p_h \text{ mit } (p_h - P[p]_{l+1}^l p_h, q_h) = 0 \quad \forall q_h \in Q_{h,l+1}, \end{array} \tag{3.21}$$

womit man

$$\alpha_k = \frac{\sum_j \alpha_j \sum_{T \in \mathcal{T}_{l+1}} (\psi_{k,l+1}, \psi_{j,l})_T}{(\psi_{k,l+1}, \psi_{k,l+1})} = \beta_k$$

erhält. Das heißt, daß die Prolongation die Werte der Vaterelemente auf die Sohnelemente überträgt.

Aus der Definition der Gittertransferoperationen läßt sich ihre Adjungiertheit zeigen. Mit (3.19) und (3.21) ergibt sich

$$\left(R[p]_l^{l+1} p_h, q_h\right) = (p_h, q_h) = \left(p_h, P[p]_{l+1}^l q_h\right) \quad \forall p_h \in Q_{h,l+1}, \forall q_h \in Q_{h,l}.$$

Der Grobgitterlöser

Das Grobgitterproblem lösen wir iterativ durch ein gedämpftes Block–Jacobi Verfahren mit inneren Gauß–Seidel Iterationen. Die Abbruchkriterien sind die gleichen wie im Mehrgitterverfahren für die Unbekannten der Geschwindigkeit.

Die Schrittweitenkontrolle

Analog zum Mehrgitterverfahren für die Geschwindigkeitsunbekannten erhöht die Anwendung einer Schrittweitenkontrolle die Effizienz auch in diesem Mehrgitterverfahren manchmal beträchtlich. Der Schrittweitenfaktor κ berechnet sich hier als

$$\kappa = \frac{(b - B^T C^{-1} B p^i)^T (B^T C^{-1} B \delta p^{i+1})}{(B^T C^{-1} B \delta p^{i+1})^T (B^T C^{-1} B \delta p^{i+1})}, \tag{3.22}$$

wobei b die rechte Seite des zu lösenden Gleichungssystems ist, p^i die gegenwärtige Iterierte und δp^{i+1} die vom Mehrgitterverfahren vorgeschlagene Korrektur. Als neue Iterierte wird $p^{i+1} = p^i + \kappa \delta p^{i+1}$ genommen.

Adaptivität

Es wird nach den gleichen Prinzipien wie beim Mehrgitterverfahren für die Geschwindigkeitsunbekannten vorgegangen, siehe Abbildung 3.7. Die mit einem gefüllten Quadrat gekennzeichneten Knoten werden als neu entstandene Knoten angesehen. Auf sie sowie auf ihre Nachbarn (gefüllte Kreise) wird die Glättung angewandt. In den mit der Quadratlinie gekennzeichneten Knoten wird nach der Glättung das Residuum neu berechnet. Die mit einer Kreislinie gekennzeichneten Knoten dienen lediglich der Berechnung dieses Residuums.

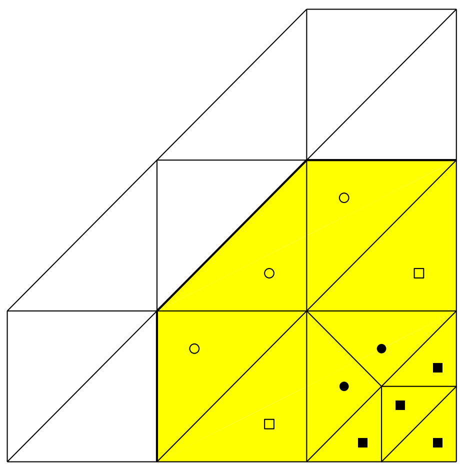


Abbildung 3.7: Lokale Verfeinerung und Klassifizierung der Knoten im Mehrgitter für die Druckunbekannten.

3.5 Das gekoppelte Mehrgitterverfahren

In diesem Abschnitt betrachten wir ein Verfahren, in denen die Unbekannten u und p des Gleichungssystems (3.1) gleichzeitig (gekoppelt) berechnet werden. Die Lösung erfolgt natürlich iterativ. Von den gegenwärtig bekannten Verfahren erscheinen allein Mehrgittermethoden zur gekoppelten Lösung von (3.1) geeignet.

Ein gekoppeltes Mehrgitterverfahren für eine nichtkonforme Finite Elemente Diskretisierung (rotiertes Q_1/P_0–Element) ist ausführlich von Schieweck in [Sch93] und [Sch96] beschrieben. Wir halten uns eng an die dortige Vorgehensweise und beschränken uns abermals auf die Beschreibung der Komponenten des Mehrgitters.

Die Residuumsberechnung

Sei $N_{el,l}$ die Anzahl der Elemente auf Level l. Bedecken die Elemente dieses Levels das gesamte Gebiet Ω, so müssen nach der Berechnung des Residuums die Komponenten bezüglich des Druckes in den Raum $\text{span}\{1_{N_{el,l}}\}^\perp$ projeziert werden, damit das Gleichungssystem mit dem neuberechneten Residuum als rechter Seite lösbar ist, vergleiche Satz 3.1.5. Sind d_i die Komponenten des Residuums bezüglich des Druckes, dann ist die Konstante

$$c = \frac{\sum_{i=1}^{N_{el,l}} d_i}{N_{el,l}} \tag{3.23}$$

zu berechnen und von $d_i, i = 1, \ldots, N_{el,l}$, zu subtrahieren.

Der Glätter

Die besondere Gestalt der Systemmatrix (reflinsys) mit dem unteren Hauptdiagonalblock, der aus lauter Nullen besteht, verbietet den Einsatz üblicher Glätter, wie punktweise Gauß–Seidel, SOR oder ILU, ohne vorherige Modifizierung.

Wir verwenden den sogenannten *Vanka–Glätter*. Er wurde von Vanka in [Van86] für Finite Differenzen Verfahren entwickelt und von Risch und Schieweck in [RS88] auf das nichtkonforme P_1/P_0–Element übertragen. Beim Vanka–Glätter handelt es sich um einen Block–Gauß–Seidel Glätter, wobei ein Block aus all den Unbekannten besteht, die mit einem Element $T \in \mathcal{T}_l, l = 1, \ldots, L$ und dessen Kanten verbunden sind, siehe Abbildung 3.8. Von den Randelementen abgesehen, sind das sieben Unbekannte pro Block. Sei $z_T = (u_{11}, \ldots, u_{32}, p_1)^T$ und

$$\mathcal{A}_T = \begin{pmatrix} A_T & B_T \\ B_T^T & 0 \end{pmatrix} \in \mathbb{R}^{7\times 7}$$

der Block der Matrix $\mathcal{A}_l$, der die zu z_T gehörenden Zeilen und Spalten enthält. Hierbei ist B_T ein Vektor. Weiter sei

$$D_T = \begin{pmatrix} \text{diag}(A_T) & B_T \\ B_T^T & 0 \end{pmatrix} \in \mathbb{R}^{7\times 7}.$$

Als nächstes wird die rechte Seite für ein lokales Gleichungssystem für den Vektor z_T definiert. Mit $\tilde{\mathcal{A}}_l$ bezeichnen wir diejenige Matrix, die aus der zum Gitter $\mathcal{T}_l$ gehörenden Matrix $\mathcal{A}_l$ entsteht, wenn man in $\mathcal{A}_l$ alle Koeffizienten durch Null ersetzt, die zu D_T gehören. Die Vektoren z_T werden als Bestandteile eines Vektors z betrachtet, der die Dimension und die gleiche Anordnung der Variablen wie der Vektor (u_l, p_l) besitzt. Der Index l zeigt an, daß die Unbekannten zum Level l gehören. Die Komponenten des Vektors z werden durch die Lösung der Gleichungssysteme auf den Elementen berechnet. Bereits berechnete Komponenten werden für die Generierung der rechten Seite der lokalen Gleichungssysteme im Sinne eines Gauß–Seidel Verfahrens genutzt. Die rechte Seite des lokalen Gleichungssystems auf dem Element T ist durch

$$\begin{pmatrix} r_T \\ s_T \end{pmatrix} = \begin{pmatrix} r_l \\ s_l \end{pmatrix}_T - \left(\tilde{\mathcal{A}}_l z\right)_T$$

definiert. Das Symbol $(\cdot)_T$ zeigt an, daß nur die zu T gehörenden Komponenten genommen werden. Eine Glättungsiteration für (u_l, p_l) hat die Gestalt (siehe ebenfalls Schieweck [Sch96])

`initialisiere` $z = 0$
`for` $(T \in \mathcal{T}_l)$
$\{$
    `löse`

$$D_T z_T = (r_T, s_T)^T$$

    $z = z + z_T$ `für die` T `betreffenden Komponenten`
$\}$
$(u_l, p_l)^T := (u_l, p_l)^T + \omega_s z.$

Hierbei ist ω_s ein Dämpfungsfaktor.

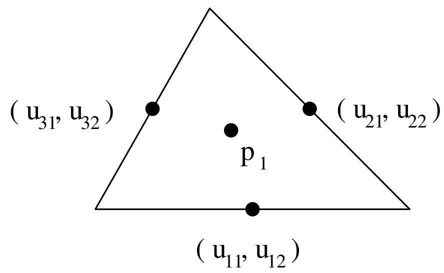


Abbildung 3.8: Die Unbekannten des Vanka–Glätters.

Unter gewissen Bedingungen ist es aus Gründen der Stabilität besser, statt der Matriz D_T die Matrix $\mathcal{A}_T$ beim Glätten zu verwenden. Dies kann beispielsweise bei anisotropen Gittern von Vorteil sein, siehe bei Schieweck in [Sch96], Bemerkung 8.1 . Diesen Glätter nennen wir im weiteren *stabilisierten Vanka-Glätter.*

Der Gittertransfer und der Grobgitterlöser

Als Gittertransferoperationen für die Geschwindigkeits- und Druckunbekannten verwenden wir die in den Abschnitten 3.3 und 3.4 vorgestellten L^2–Projektionen. Das Grobgitterproblem wird näherungsweise mit einem der obigen Glätter gelöst.

Die Schrittweitenkontrolle

Genauso wie bei den bisher vorgestellten Mehrgitterverfahren ist die Verwendung einer Schrittweitenkontrolle auch beim gekoppelten Verfahren oft von Vorteil. Die euklidische Norm des Residuums

$$\begin{pmatrix} r \\ s \end{pmatrix} - \mathcal{A} \begin{pmatrix} u \\ p \end{pmatrix}^i$$

wird in der vom Verfahren vorgeschlagene Korrekturrichtung durch den Schrittweitenfaktor

$$\kappa = \frac{((r^T, s^T) - \mathcal{A}(u^T, p^T)^i)^T (\mathcal{A}(\delta u^T, \delta p^T)^{i+1})}{(\mathcal{A}(\delta u^T, \delta p^T)^{i+1})^T (\mathcal{A}(\delta u^T, \delta p^T)^{i+1})} \tag{3.24}$$

minimiert. Hierbei sind $(u^T, p^T)^i$ die bisherige Iterierte und $(\delta u^T, \delta p^T)^{i+1}$ die vom Mehrgitterverfahren vorgeschlagene Korrektur.

Adaptivität

Die Anwendung des gekoppelten Mehrgitterverfahrens auf lokal verfeinerten Netzen erfolgt nach den gleichen Prinzipien wie bei den anderen Mehrgitterverfahren. Insbesondere erfolgt die Glättung für alle Unbekannte, die sich in einer Umgebung der neuen Unbekannten des lokal verfeinerten Gitters befinden. In Abbildung 3.9 ist das vom Programm erzeugte lokal verfeinerte Gitter schattiert dargestellt, die neuen Unbekannten mit einem gefüllten Quadrat und ihre direkten Nachbarn, auf denen ebenfalls geglättet wird, mit einem gefüllten Kreis. In den Unbekannten, die mit einer Quadratlinie gekennzeichnet sind, wird nur das Residuum neu berechnet, und in allen anderen Unbekannten (Kreislinie) wird auf dem lokal verfeinerten Gitter nichts verändert. Insbesondere gibt es nun Funktionswerte des Druckes, die sich nicht ändern, sozusagen Randwerte. Wegen Folgerung 3.1.6 ist mit der Festlegung dieser Druckwerte das Gleichungssystem auf dem lokal verfeinerten Gitter für jede rechte Seite eindeutig lösbar. Deswegen muß das Residuum bezüglich des Druckes nicht projeziert werden und die Konstante c aus (3.23) muß nicht berechnet werden.

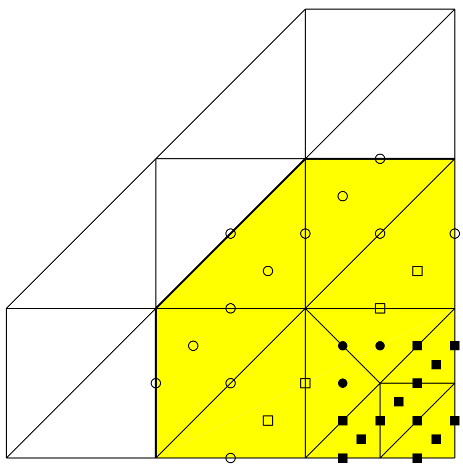


Abbildung 3.9: Lokale Verfeinerung und Klassifizierung der Knoten im gekoppelten Mehrgitter.

3.6 Das Mehrgitterverfahren für konforme Finite Elemente

Ein Mehrgitterverfahren für konforme, lineare Finite Elemente wird zur Lösung des Neumannproblems für den Druck bei den Varianten 1, 2 und 3 der Wahl des separierten Druckes p_S angewandt, siehe Abschnitt 2.6. Außerdem werden damit die im Abschnitt 8.1 vorgestellten numerischen Resultate für konforme Upwind Finite Elemente Methoden berechnet.

Das von uns verwendete Mehrgitterverfahren ist ausführlich von Bastian [Bas96] beschrieben und wurde direkt dem Programmpaket *ugp*1.0 entnommen. Wir wollen hier nur die wichtigsten der von uns verwendeten Komponenten erwähnen. Die Gittertransferoperationen sind die kanonische Restriktion und Prolongation. Beim Mehrgitterverfahren zur Berechnung des separierten Druckes verwenden wir als Glätter Block–Jacobi–Iterationen mit inneren Gauß–Seidel–Iterationen. Für die konforme Upwind FEM verwenden wir Block–Jacobi–Iterationen mit inneren ILU_β–Iterationen. Das Grobgittersystem wird mit Hilfe des Glätters iterativ gelöst. Auf eine Schrittweitenkontrolle verzichten wir. Bei den numerischen Tests wurden immer V–Zyklen verwendet.

3.7 Zur Konvergenztheorie der Verfahren

Definition 3.7.1 *Man nennt ein Iterationsverfahren robust für eine Problemklasse, wenn es konvergiert und die Konvergenzrate durch eine Konstante $\rho < 1$ beschränkt ist, wobei ρ unabhängig vom gestellten Problem (insbesondere von den Koeffizientenfunktionen) und vom gewählten Gitter ist.*

Bei der analytischen Untersuchung der oben vorgestellten Verfahren treten drei Schwierigkeiten auf:

- Den zu lösenden Problemen fehlen Eigenschaften, die eine einfache analytische Untersuchung ermöglichen, insbesondere die Selbstadjungiertheit.
- Die Diskretisierungen sind nichtkonform.
- Die Verfeinerungen sind adaptiv.

Die Literatur zur Konvergenztheorie von Mehrgittermethoden ist zu jedem einzelnen der Punkte übersichtlich.

Eine Mehrgitterbeweistechnik

Die übliche Herangehensweise zur Untersuchung der Konvergenz von Mehrgitterverfahren für ein System $Au = b$ beginnt mit der Untersuchung von 2–Gittermethoden zwischen den Leveln l und $l-1$. Seien eine Restriktion R, eine Prolongation P, ein Glätter S_l sowie Systemmatrizen A_l und A_{l-1} gegeben. Dann hat eine 2–Gittermethode die Iterationsmatrix

$$T_2(\nu_1, \nu_2) := S_l^{\nu_2}(I_l - PA_{l-1}^{-1}RA_l)S_l^{\nu_1}.$$

Oft wird der Fall $\nu_2 = 0$ betrachtet. Man beginnt mit der Zerlegung

$$\|T_2(\nu_1, 0)\| \le \|A_l^{-1} - PA_{l-1}^{-1}R\| \|A_l S_l^{\nu_1}\|$$

in geeigneten Normen. Die beiden Faktoren werden getrennt abgeschätzt, wobei an alle auftretenden Operatoren gewisse Voraussetzungen gestellt werden. Für partielle Differentialgleichungen von 2.Ordnung müssen die Approximationseigenschaft

$$\|A_l^{-1} - PA_{l-1}^{-1}R\| \le ch_l^2$$

und die Glättungseigenschaft

$$\|A_l S_l^{\nu_1}\| \le ch_l^{-2}\eta(\nu_1)$$

mit $\eta(\nu_1) \to 0$ für $\nu_1 \to \infty$ bewiesen werden. Für hinreichend viele Glättungsschritte erhält man die Abschätzung

$$\|T_2(\nu_1, 0)\| \le c < 1 \quad \forall\, l \ge 1.$$

Daraus kann die Konvergenz des Mehrgitterverfahrens mit W–Zyklus hergeleitet werden, siehe bei Hackbusch [Hac85], Seite 160 ff. Der Konvergenzbeweis von V–Zyklen erfordert andere Techniken und stärkere Voraussetzungen. Die Approximationseigenschaft hängt von der zugrunde liegenden Differentialgleichung, genauer von ihrer Diskretisierung, sowie den Gittertransferoperationen ab.

Singulär gestörte Probleme

Konvergenzanalysen von Mehrgitterverfahren für Konvektions–Diffusions Probleme findet man in einigen Arbeiten von Reusken [Reu93], [Reu94], [Reu95] und [Reu96]. Es werden zunächst eindimensionale und dann zweidimensionale Aufgaben mit speziell gewählten Koeffizientenfunktionen und Randbedingungen betrachtet. Die untersuchten Mehrgitterverfahren haben jedoch speziell gewählte Gittertransferoperatoren und Grobgittersysteme. Es wird gezeigt, daß dann die Verwendung einfacher Glätter, zum Beispiel des gedämpften Jacobi Verfahrens, zu robusten 2–Gittermethoden führt. In den beiden zuletzt zitierten Arbeiten werden 2–Gittermethoden entwickelt und untersucht, die auf der Blockstruktur einer Matrix

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

beruhen. Die unteren Blöcke sind mit den Unbekannten des Grobgitters und die oberen Blöcke mit den Unbekannten, die sich nur auf dem feinen Gitter befinden, verbunden. Die Matrix A wird mit einer unvollständigen LU–Zerlegung faktorisiert. Schwerpunkt der numerischen Berechnungen ist die Lösung eines Schur–Komplement Systems auf dem Grobgitter. Für einige spezielle Typen von singulär gestörten Problemen wird die Robustheit dieses Verfahrens bewiesen.

In der Praxis wird im allgemeinen versucht, den Schwierigkeiten der Problemklasse mit angepaßten Glättern zu begegnen. Wittum untersucht in [Wit89b] den ILU_β–Glätter für anisotrope Probleme der Gestalt (1.1) mit $b = (0,0)^T, c = 0$ und

$$\nu(x) = \begin{pmatrix} \bar{\nu}(x) & 0 \\ 0 & 1 \end{pmatrix}$$

mit $\bar{\nu}(x) << 1$. Für diese Probleme wird die Robustheit der 2–Gittermethode mit ILU_β als Glätter gezeigt.

In [Wit90] betrachtet Wittum Glätter für die Stokes und Navier–Stokes Gleichungen. Bei diesen Glättern wird die Matrix

$$\begin{pmatrix} A_l & B_l \\ B_l^T & 0 \end{pmatrix}$$

zuerst im Sinne einer Druckkorrektur Methode mit einer Matrix von rechts transformiert

$$\begin{pmatrix} A_l & B_l \\ B_l^T & 0 \end{pmatrix} \begin{pmatrix} I_l & A_l^{-1}B_l \\ 0 & -I_l \end{pmatrix} = \begin{pmatrix} A_l & 0 \\ B_l^T & B_l^T A_l^{-1} B_l \end{pmatrix} \tag{3.25}$$

und das transformierte System mit Hilfe bekannter Iterationsverfahren geglättet, zum Beispiel mit ILU_β. Sei M die ILU_β– Zerlegung von (3.25). Dann hat die Glättungsiteration die Gestalt

$$\begin{pmatrix} u_l \\ p_l \end{pmatrix}^{i+1} = \begin{pmatrix} u_l \\ p_l \end{pmatrix}^{i} - \begin{pmatrix} I_l & A_l^{-1}B_l \\ 0 & -I_l \end{pmatrix} M^{-1} \left[\begin{pmatrix} A_l & B_l \\ B_l^T & 0 \end{pmatrix} \begin{pmatrix} u_l \\ p_l \end{pmatrix}^{i} - \begin{pmatrix} r_l \\ s_l \end{pmatrix}^{i} \right]. \tag{3.26}$$

Für Paare konformer Finiter Elemente, zum Beispiel das Taylor–Hood Element, wird die Konvergenz von 2–Gittermethoden für die Stokes Gleichungen unter der Verwendung der Glättung (3.26) bewiesen, falls hinreichend viele (Vor–) Glättungsschritte ausgeführt werden und $\beta > 1$ gilt. Für die linearisierten Navier–Stokes Gleichungen (2.20) wird für $\beta > 1$ und hinreichend feine Gitter mit $h_l \leq h_0(\nu)$, $l > 1$ die Glättungseigenschaft der Iteration (3.26) in gewissen Normen gezeigt.

Nichtkonforme Finite Elemente Methoden

Die Übertragung von Mehrgittermethoden auf nichtkonforme Finite Elemente ist von McCormick als Problem 3.4 in [McC87], S.171, genannt.

Bei der Untersuchung von Mehrgitterverfahren für nichtkonforme Finite Elemente Methoden spielt die Wahl geeigneter Gittertransferoperationen eine entscheidende Rolle. Braess und Verfürth [BV90] und [Bra96] betrachten die L^2–Projektion und Brenner [Bre89], [Bre90] eine Mittelung von Werten als Interpolationsoperator. Diese Mittelung ist für Triangulierungen mit gleichgroßen Dreiecken mit der L^2–Projektion identisch und spart im allgemeinen Fall etwas numerische Arbeit. Die Restriktion ist jeweils als der adjungierte Operator definiert.

Braess und Verfürth betrachten in [BV90] die Poisson Gleichung (1.1) ($b = (0, 0)^T, c = 0$) mit homogenen Dirichletbedingungen, welche mit dem P_1–nichtkonformen Finiten Element (1.8) diskretisiert wird. Es wird die Mehrgitterkonvergenz in der L^2– und H^1–Norm für den W–Zyklus bewiesen. Als Glätter wird das gedämpfte Jacobi Verfahren verwendet. Es sind wiederum hinreichend viele Glättungsiterationen notwendig. Für die Konvergenz in der H^1–Norm ist zudem eine Schrittweitenkontrolle erforderlich. Ähnliche Ergebnisse für Problem (1.1) mit $b = (0, 0)^T, c \geq c_0 \geq 0$ und homogenen Dirichletbedingungen sowie gemischte Randwertprobleme mit homogenen Randbedingungen erhält Brenner in [Bre89].

Dieses Resultat wird in [Bre90] auf die Stokes Gleichungen bei Verwendung nichtkonformer, stückweise linearer und divergenzfreier Finiter Elemente ausgedehnt.

In allen angegebenen Arbeiten über Mehrgitterverfahren für nichtkonforme Finite Elemente ist die Symmetrie und die positive Definitheit der zugrunde liegenden Operatoren grundlegend zur Beweisführung.

Adaptive Mehrgitterverfahren

Theoretische Untersuchungen der Konvergenz von Iterationsverfahren auf adaptiv verfeinerten Gittern sind uns nicht bekannt.

Es bleibt festzuhalten, daß Konvergenzbeweise für die von uns verwendeten Verfahren und für die uns interessierenden Problemklassen offen sind.

Kapitel 4

Die Implementierung der Verfahren auf Parallelrechnern

Dieses Kapitel ist der Implementation der im Kapitel 3 vorgestellten Verfahren zur Lösung des Sattelpunktproblems (3.1) auf einem Massiv–Parallelrechner gewidmet.

Bei der Implementation wurde auf Module des Programmpaketes *ugp*1.0 von Bastian zurückgegriffen. Die Datenstrukturen, Kommunikationsroutinen und verschiedene Verfahren zum Lastausgleich von *ugp*1.0 hat Bastian in seiner Dissertation [Bas96] ausführlich beschrieben. Wir wollen hier nur insoweit darauf eingehen, als es für das Verständnis der parallelen Implementation unbedingt notwendig ist.

Es ist die Struktur der Matrix A zugrunde gelegt, die sich bei Upwind Diskretisierungen ergibt, siehe Abbildung 1.3.

4.1 Das Datenkonzept

Das Programmpaket *ugp*1.0 („unstructured grids parallel") ist zur numerischen Lösung partieller Differentialgleichungen auf unstrukturierten Gittern konzipiert. Es ist in ANSI–C geschrieben. Zur Behandlung unstrukturierter Datenmengen bietet sich eine Programmierung mit Hilfe von C–Strukturen an, bei welcher die Strukturen mittels Listen verbunden sind. Die meisten Operationen im Programm sind mit dem Durchlaufen derartiger Listen verbunden. Das ist ein wesentlicher Unterschied zu Programmen, die mit strukturierten Datenmengen arbeiten. Bei diesen wendet man im allgemeinen Indexarithmetik an, zum Beispiel Dorok [Dor95]. Es hat sich bei numerischen Experimenten erwiesen, siehe Rüde [Rüd96], daß Programme mit unstrukturierten Datenmengen auf Grund sehr schlechter Cash–Ausnutzung oft ein Vielfaches der Rechenzeit im Vergleich zu Programmen mit strukturierten Datenmengen benötigen. Der Vorteil unstrukturierter Datenmengen liegt jedoch in der einfachen (oder überhaupt erst möglichen) Verwendung adaptiver Techniken. Zudem bieten die flexiblen Datenstrukturen die Möglichkeit, neue Verfahren relativ schnell zu implementieren, zu testen und zu variieren.

Die Bezeichnungen vieler Strukturen in *ugp*1.0 sind selbsterklärend: PROBLEM, DOMAIN, MULTIGRID, GRID, ELEMENT, NODE, VERTEX, EDGE und LINK. Die NODE–Struktur ist mit einer GRID–Struktur verbunden. Im Mehrgitterkontext gibt es mehrere NODE-Strukturen mit denselben geometrischen Daten. Diese werden in der VERTEX–Struktur gespeichert. Die LINK–Struktur ist ein Pointer, auf dem Datenspeicherung möglich ist. Die obigen Strukturen haben wir übernommen und insbesondere die EDGE–Struktur zur Abspeicherung der Koeffizienten der nichtkonformen P_1–Funktionen wesentlich ausgebaut. In *ugp*1.0 war lediglich vorgesehen, daß Variablen in den Eckpunkten der Drei– und Vierecke, daß heißt in den NODE–

Strukturen gespeichert werden können. Analog zum Mehrgitterkontext werden wir hier auch von „Vater-“, „Sohn-“ und „Nachbar“–Strukturen sprechen, vergleiche auch Abbildung 4.1. Wir nennen eine Struktur *Top–Struktur*, falls sie in der Hierarchie keine Sohn–Struktur mehr besitzt.

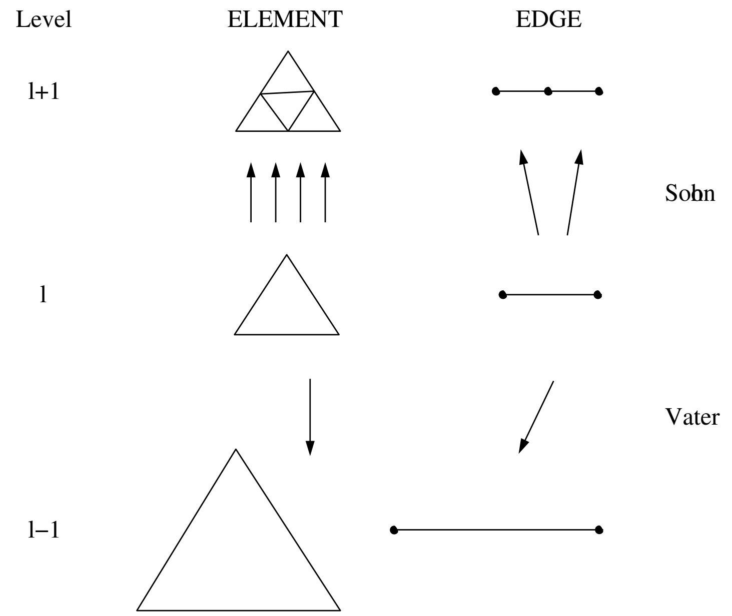


Abbildung 4.1: Sohn– und Vater–Strukturen.

Wir wollen jetzt die wichtigsten Verbindungen der Strukturen auf einem Level beschreiben. Die mit • versehenen Punkte stellen Erweiterungen von unserer Seite dar.

- ELEMENT–Strukturen haben Pointer zu
  - allen zugehörigen NODE–Strukturen.
  - • allen zugehörigen EDGE–Strukturen.
  - allen Nachbar–ELEMENT–Strukturen.

  NODE–Strukturen haben Pointer zu
  - allen Nachbar–NODE–Strukturen, das sind NODE–Strukturen, die zu einer gemeinsamen ELEMENT–Struktur gehören. Diese Verbindungen sind LINK–Strukturen.
  - allen von ihr ausgehenden EDGE–Strukturen.
  - der, die Geometrie beschreibende, VERTEX–Struktur.

  VERTEX–Strukturen haben Pointer zu
  - allen NODE-Strukturen, die die gleichen geometrischen Daten besitzen.

  EDGE–Strukturen haben Pointer zu
  - den beiden NODE–Strukturen, die die Kante des Elementes beschreiben.
  - • den beiden anliegenden ELEMENT–Strukturen.
  - • allen Nachbar–EDGE–Strukturen. Das sind die anderen EDGE–Strukturen der anliegenden ELEMENT–Strukturen. Diese Pointer sind LINK–Strukturen.

Bei Verwendung der NSDFEM sind weitere LINK–Strukturen einzuführen, die die Diskretisierung der Sprungterme speichern, siehe Abbildung 1.3.

Desweiteren gibt es Verkettungen gleichartiger Strukturen verschiedener Level.

- Eine ELEMENT–Struktur kennt seine Vater–ELEMENT–Struktur und seine (bis zu vier) Sohn–ELEMENT–Strukturen.

- Eine NODE–Struktur kennt seine Vater–NODE–Struktur und seine Sohn–NODE–Struktur.
- Eine EDGE–Struktur kennt seine Vater–EDGE–Struktur und seine (bis zu zwei) Sohn–EDGE–Strukturen.

Die beschriebenen Verkettungen existieren natürlich nur, falls die entsprechenden Strukturen existieren. Anderenfalls zeigen die Pointer auf NULL.

Für die Implementierung der Algorithmen auf einem Parallelrechner muß das bisherige Datenkonzept erweitert werden. Es kann nämlich jetzt vorkommen, daß sich Strukturen, zwischen denen normalerweise eine Verbindung besteht, auf verschiedenen Prozessoren befinden. Insbesondere kann dies für die ELEMENT–Strukturen benachbarter Elemente zutreffen. Informationen über die Nachbar–ELEMENT–Struktur auf einem anderen Prozessor werden in einer sogenannten INTERFACE–Struktur gespeichert. Der Rand eines Teilgebietes, welches sich auf einem Prozessor befindet, wird *Interface* genannt, sofern er nicht zu $\partial\Omega$ gehört. Natürlich kennt jede ELEMENT–Struktur auch in diesem Falle seine NODE– und EDGE–Strukturen. Deswegen können sich Strukturen für ein und dasselbe geometrische Objekt (Knoten oder Kante) auf mehreren Prozessoren befinden. Eine dieser Strukturen wird dann als *MASTER–Struktur* bezeichnet, die anderen als *COPY–Strukturen.* COPY–Strukturen von ELEMENT– und LINK–Strukturen sind nicht nötig.

Das Programmpaket *ugp*1.0 ist elementweise orientiert. Es gilt folgender wichtiger Grundsatz:

> Jede ELEMENT–Struktur kennt entweder seine Vater–ELEMENT–Struktur, oder sie befindet sich auf Level 0.

Nach der Verfeinerung mit dem anschließenden Lastausgleich kann es passieren, daß sich nicht mehr alle Sohn–ELEMENT–Strukturen einer ELEMENT–Struktur auf demselben Prozessor wie die ELEMENT–Struktur befinden. Wegen des obigen Grundsatzes ist es notwendig, mit der Verschiebung einer Sohn–ELEMENT–Struktur auf einen anderen Prozessor eine Kopie der Vater–ELEMENT–Struktur mitzuschicken. Die ursprüngliche ELEMENT–Struktur wird dann als MASTER–ELEMENT–Struktur und die Kopie als *SLAVE–ELEMENT–Struktur* bezeichnet. Eine MASTER–ELEMENT–Struktur kennt all seine SLAVE–ELEMENT–Strukturen. Es gilt ein weiterer Grundsatz:

> Die Vater–ELEMENT–Struktur der ELEMENT–Struktur eines aus irregulärer Verfeinerung hervorgegangenen Elementes ist eine MASTER–ELEMENT–Struktur.

Elemente, die durch irreguläre Verfeinerungen entstehen oder die vom Typ COPY sind, siehe Abbildung 3.2, werden vor der nächsten Verfeinerung wieder gelöscht. Damit verhindert der zweite Grundsatz, daß SLAVE–ELEMENT–Strukturen ohne Sohn–ELEMENT–Strukturen auf den Prozessoren existieren. Eine SLAVE–ELEMENT–Struktur kennt all seine NODE– und EDGE–Strukturen, welche Kopien der NODE– und EDGE–Strukturen der MASTER–ELEMENT–Struktur sind. Sind diese Strukturen auf dem Prozessor, auf dem sich die SLAVE–ELEMENT–Struktur befindet, noch nicht als MASTER– oder COPY–Struktur vorhanden, dann werden sie ebenfalls als SLAVE–Struktur bezeichnet.

Abbildung 4.2 stellt eine typische Situation dar. Dabei wurde eine Sohn–ELEMENT–Struktur der MASTER–ELEMENT–Struktur MEL0 auf Prozessor 1 verschoben. Es wird die Kopie SLAVE–ELEMENT–Struktur SEL0 der MASTER–ELEMENT–Struktur MEL0 ebenfalls auf Prozessor 1 gesandt. Auf Prozessor 1 sind wegen der MASTER–ELEMENT–Struktur MEL1 bereits COPY–Strukturen der MASTER–NODE–Strukturen MN1 und MN2 sowie der MASTER–EDGE–Struktur MED1 vorhanden. An diesen wird nichts geändert. Es werden lediglich noch Kopien der MASTER–NODE–Struktur MN0 und der MASTER–EDGE–Strukturen MED0 und MED2 angelegt und als SLAVE–Strukturen bezeichnet.

Somit hat man zwei verschiedene Arten von Verbindungen zwischen den Prozessoren, zum

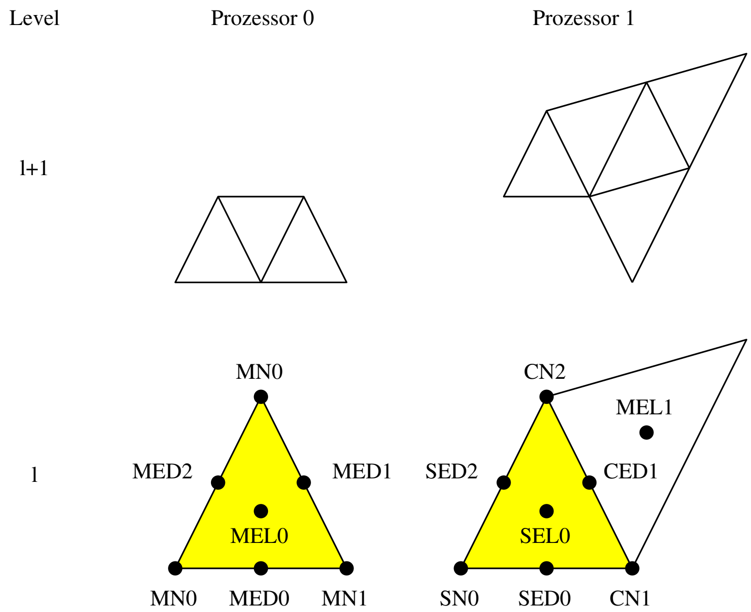


Abbildung 4.2: MASTER–, SLAVE– und COPY–Strukturen.

einen zwischen MASTER– und COPY–Strukturen und zum anderen zwischen MASTER– und SLAVE–Strukturen. Während die SLAVE–Strukturen nur als eine Art Zwischenspeicher anzusehen sind, sind COPY–Strukturen, genau wie ihre MASTER–Strukturen, aktiv in die Berechnung der numerischen Ergebnisse involviert. Eine Unterscheidung dieser Strukturen ist nur bei wenigen Operationen erforderlich.

Definition 4.1.1 *Ein Wert x auf einem Prozessor P_i wird mit $(P_i \to x)$ bezeichnet. Dieser Wert heißt konsistent gespeichert, falls*

$$x = (P_i \to x) \quad \forall i$$

gilt. Er heißt inkonsistent gespeichert, falls

$$x = \sum_i (P_i \to x)$$

gilt.

Definition 4.1.2 *Ein Wert x, der innerhalb einer Struktur gespeichert ist, wird mit*

$$Struktur \to x$$

bezeichnet. Er heißt innerhalb einer Struktur konsistent gespeichert, wenn für alle COPY–Strukturen gilt

$$x = MASTER\text{–}Struktur \to x = COPY\text{–}Struktur \to x \ .$$

Er heißt inkonsistent gespeichert, wenn gilt

$$x = MASTER\text{–}Struktur \to x + \sum COPY\text{–}Struktur \to x \ ,$$

wobei über alle COPY–Strukturen summiert wird.

Im deutschsprachigen Raum werden konsistent gespeicherte Werte manchmal auch als „vom überlappenden Typ“ und inkonsistent gespeicherte Werte als „vom additiven Typ“ bezeichnet, siehe zum Beispiel Langer [Lan92].

Von entscheidender Bedeutung für die Effizienz der Algorithmen auf Parallelrechnern ist eine gleichmäßige Verteilung der Last. Das bedeutet, daß alle Prozessoren möglichst gleich viel zwischen je zwei Synchronisationen zu tun haben. Synchronisationen sind *Kommunikationen* (Datenaustausch mit einem Senden und einem Empfangen von Informationen), bei denen die Prozessoren aufeinander warten müssen, um weiterarbeiten zu können. Insbesondere ist zu vermeiden, daß wenige Prozessoren wesentlich mehr als die anderen zu tun haben.

Bei Verwendung adaptiver Algorithmen mit lokaler Verfeinerung ist es unvermeidlich, daß sich nach einem Verfeinerungsschritt ein Lastungleichgewicht einstellt. Dieses muß mit einem dynamischem Lastausgleich behoben werden. Wir konnten auf die Lastverteilungsroutinen des Programmes *ugp*1.0 zurückgreifen. Bastian hat sie im Rahmen seiner Dissertation [Bas96] ausführlich beschrieben. Es werden hier nur ganz knapp die wichtigsten Aspekte erwähnt.

Die Lastverteilung für multiplikative Mehrgitterverfahren hat zum Ziel, die Elemente jedes Levels möglichst gleichmäßig auf alle Prozessoren zu verteilen. Dadurch erreicht man geringe Wartezeiten in den Synchronisationen, vergleiche Kapitel 5.4. in [Bas96]. Das Lastverteilungsproblem läßt sich auf ein Graphenabbildungsproblem zurückführen. Dieses ist NP–vollständig und wird in *ugp*1.0 näherungsweise durch ein Koordinatenbisektionsverfahren mit alternierenden Richtungen gelöst. Weiterhin ist wichtig zu erwähnen, daß diese Prozedur nicht auf einzelne Elemente, sondern auf Cluster von Elementen angewandt wird. Damit werden die Dimension des Lastverteilungsproblems und die Speicheranforderungen des Lastverteilungsalgorithmus wesentlich verringert.

Bei vielen Problemen verbraucht die Berechnung und Ausführung der Lastverteilung nur einen unwesentlichen Teil der gesamten Rechenzeit.

Bemerkung 4.1.3 Vorteile der nichtkonformen Finiten Elemente Diskretisierung auf Parallelarchitekturen. Bei elementorientierten Programmen bietet die Verwendung nichtkonformer Finiter Elemente bedeutende Vorteile bei der Implementierung auf Parallelarchitekturen im Vergleich zu konformen Finiten Elementen. Bei linearen Elementen besteht der Träger jeder nichtkonformen Basisfunktion aus höchstens zwei Elementen. Sind deren ELEMENT–Strukturen auf unterschiedlichen Prozessoren gespeichert, müssen Werte auf der EDGE–Struktur, die zu beiden ELEMENT–Strukturen gehört, zweimal gespeichert werden. Ein Informationsaustausch ist mit einer einzigen Kommunikation möglich. Bei linearen konformen Finite Elementen werden Werte in den NODE–Strukturen gespeichert. Eine NODE–Struktur gehört bei strukturierten Gittern etwa zu 4 bis 8 ELEMENT–Strukturen, bei adaptiven Gittern ist eine obere Grenze a priori nicht bekannt. Diese ELEMENT–Strukturen können auf mehreren Prozessoren verteilt sein. Die entsprechende NODE–Struktur wird immer mit abgespeichert. Neben dem erhöhten Speicheraufwand ist auch eine größere Anzahl von Kommunikationen zum Informationsaustausch nötig. Man spricht hier von *Crosspoints*. Diese Crosspoints können bei den nichtkonformen P_1– und rotierten Q_1–Finite Elementen nicht auftreten.

4.2 Die Assemblierung der Matrizen und der rechten Seite

Für die Implementation von Algorithmen auf parallelen Architekturen ist es notwendig, daß alle Einzelroutinen bezüglich ihrer Ein– und Ausgabe eine wohldefinierte Art der Speicherung der Variablen besitzen.

Die Speicherung der Matrixeinträge erfolgt innerhalb von Strukturen. Dies ist in Abbildung 4.3 illustriert. Die kleinen Kreise bezeichnen jeweils einen gespeicherten Matrixkoeffizienten. Der

Diagonaleintrag a_{ii} ist innerhalb der EDGE–Struktur i gespeichert. Von dieser EDGE–Struktur geht eine Liste von LINK–Strukturen (die Pfeile) aus, innerhalb derer die Nichtdiagonalkoeffizienten der i–ten Zeile von A gespeichert sind. Diese Art der Speicherung setzt für A die Gestalt

$$A = \begin{pmatrix} A_1 & 0 \\ 0 & A_1 \end{pmatrix}$$

voraus. Die Matrix B wird ähnlich gespeichert, nun auf LINK–Strukturen zwischen ELEMENT– und EDGE–Strukturen. Hierbei müssen die Basisfunktionen $(\phi_i, 0)^T$ und $(0, \phi_i)^T$ getrennt betrachtet werden. Wird die erste mit Index i und die zweite mit Index j versehen, so kennzeichnen die Kreise im rechten Teil von Abbildung 4.3 die Speicherung der zugehörigen Koeffizienten von B. Durch Verkettung der LINK–Strukturen untereinander ist ein schneller Zugriff auf alle Koeffizienten sowohl von der EDGE– als auch von der ELEMENT–Struktur möglich.

Die Assemblierung der Matrizen und der rechten Seite erfolgt mittels Durchlaufens der Listen der ELEMENT–Strukturen in jeder GRID–Struktur. Dies geschieht auf jedem Prozessor separat, Kommunikation wird nicht benötigt. Da die ELEMENT–Strukturen des Trägers einer nichtkonformen P_1–Funktion auf unterschiedlichen Prozessoren gespeichert sein können, sind Daten, die die Unbekannten der Geschwindigkeit betreffen, inkonsistent gespeichert. Das sind die Matrix A und die rechte Seite f. Tabelle 4.1 gibt einen Überblick über die Art der Speicherung der assemblierten Matrizen und Vektoren. Die Vektoren der Unbekannten u und p werden immer konsistent gespeichert.

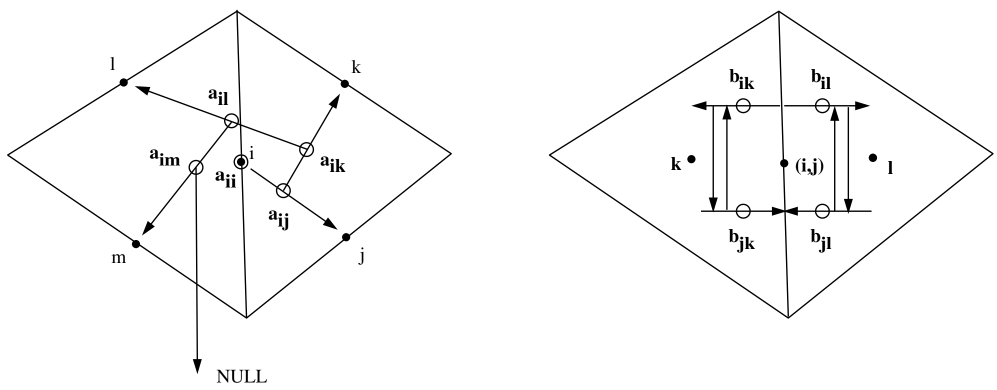


Abbildung 4.3: Die LINK–Strukturen zwischen den ELEMENT– und EDGE–Strukturen sowie die Speicherung der Matrixelemente.

Die Art der Speicherung der Matrizen und Vektoren bestimmt natürlich die Art der Speicherung von Matrix–Vektor Produkten. Die wichtigsten sind ebenfalls in Tabelle 4.1 aufgeführt. Dabei ist zu beachten, daß die Koeffizienten der Matrix B zwar konsistent gespeichert sind, bei der Multiplikation Bp aber für einen Ergebniswert die Werte p von jeweils zwei benachbarten Elementen notwendig sind. Die zugehörigen ELEMENT–Strukturen können sich auf verschiedenen Prozessoren befinden, und somit ist das Produkt Bp inkonsistent gespeichert.

Vor der Mehrgitteriteration werden die beim Gittertransfer (3.14) und (3.16) häufig benötigten Diagonalelemente der Massenmatrix $M_{kl} = (\phi_{k,l}, \phi_{k,l})$ berechnet und gespeichert. Dafür ist eine einmalige Kommunikation notwendig.

Durch die mit der Assemblierung vorgegebenen Speicherungsarten sind die Art der Speicherung für die Ein– und Ausgabe vieler Routinen auf kanonische Art und Weise vorgegeben. Man hat jetzt eine minimale Anzahl von Stellen in den Algorithmen zu finden, in denen Kommunikationen notwendig sind, um inkonsistent gespeicherte Vektoren oder einzelne Werte in konsistente Werte zu konvertieren.

Vektor/Matrix	Art der Speicherung
A	inkonsistent
B	konsistent
f	inkonsistent
u	konsistent
p	konsistent
Au	inkonsistent
$B^T u = -s$	konsistent
Bp	inkonsistent
$r = f - Au - Bp$	inkonsistent

Tabelle 4.1: Speicherung der Vektoren, Matrizen und Matrix–Vektor Produkte.

4.3 Das Mehrgitterverfahren für die Geschwindigkeitsunbekannten

Die Vorbereitung

Vor der Mehrgitteriteration ist der Glätter vorzubereiten. Nach der Assemblierung ist die Matrix A, genauer die Diagonale von A, inkonsistent gespeichert. Wir verwenden Block–Jacobi Glätter, wobei die Blöcke konsistente Matrizen sind. Deswegen wird die konsistente Diagonale mittels einer Kommunikation erzeugt und auf einem zusätzlichen Speicherplatz abgelegt.

Die Effizienz der ILU_β– und der Gauß–Seidel Iteration hängt von der Anordnung der Unbekannten ab. Es besteht die Möglichkeit verschiedener lexikographischer Anordnungen. Für Beispiele siehe Hackbusch [Hac85], Seite 50f.. Gemäß der gewählten Anordnung werden die Unbekannten, das heißt die ihnen entsprechenden EDGE–Strukturen auf jedem Prozessor umgeordnet. Kommunikation ist nicht nötig. Im Anschluß erfolgt eine Durchnumerierung der EDGE–Strukturen.

Nutzt man auf den Prozessoren die ILU_β–Iteration, so wird als letzte Phase der Vorbereitung die ILU_β–Zerlegung berechnet und gespeichert. Dabei wird der Speicherplatz des konsistenten Diagonalelementes von A mit dem Diagonalelement von U überschrieben.

Die Glättung und die Residuumsberechnung

Wir betrachten jetzt die Schritte des Mehrgitterverfahrens, die auf einem Level erfolgen, nämlich das Glätten und die Berechnung des Residuums. Aus Tabelle 4.1 kann man entnehmen, daß das Residuum r inkonsistent gespeichert wird.

Wir bezeichnen das Ergebnis einer Glättungsiteration mit δx und den Vektor, auf den die Ergebnisse der Glättungsiterationen aufdatiert werden, mit x. Dieser Vektor x wird am Ende jedes Mehrgitterzyklus zu u addiert. Deswegen ist es sinnvoll, x und damit δx konsistent zu speichern. Nun ist auf jedem Prozessor die Blockmatrix ebenfalls konsistent gespeichert, mit der die Gauß–Seidel Iteration durchgeführt wird beziehungsweise mit der die ILU_β–Zerlegung berechnet worden ist. Aus Kompatibilitätsgründen muß dann die rechte Seite des Gleichungssystems, auf welches die Glättung angewandt wird, auch konsistent gespeichert sein. Das heißt, vor dem Glätten muß das Residuum in einen konsistenten Vektor konvertiert werden.

Beim Glätten unterscheiden wir zwei Vorgehensweisen. Zum einen ist es möglich, nur die MASTER–EDGE–Struktur in den Glättungsprozess einzubeziehen. Die zugehörige COPY–EDGE–Struktur wird auf ihrem Prozessor nicht beachtet. Nach dem Glätten wird die für die

MASTER–EDGE–Struktur berechnete Korrektur δx zur COPY–EDGE–Struktur übergeben. Die andere Variante besteht darin, daß sowohl MASTER– als auch COPY–EDGE–Struktur auf ihren Prozessoren beim Glätten involviert sind. Das liefert auf diesen Strukturen im allgemeinen unterschiedliche Werte. Das Ergebnis des Glättens ist schließlich das arithmetische Mittel dieser Werte. Der parallele Aufwand besteht bei beiden Varianten in einer Kommunikation.

Wir bevorzugen die zweite Variante. Für konvektionsdominante Probleme ist die Reihenfolge der Unbekannten für die Effizienz des Glätters wichtig. In der ersten Variante kann sich eine „falsche Wahl" von MASTER– und COPY–EDGE–Struktur ebenfalls nachteilig auf die Effizienz auswirken. Gegebenenfalls werden Informationen von der COPY–EDGE–Struktur in Konvektionsrichtung nicht mehr transportiert, da diese Struktur nicht am Glätten teilnimmt. Dieser Effekt wird auf jeden Fall gemildert, wenn man die zweite Variante nutzt.

Nach dem Glätten sind alle Daten dergestalt vorhanden, daß die Berechnung des neuen, inkonsistent gespeicherten Residuums erfolgen kann.

Der Gittertransfer

Die Gittertransferoperationen sind ausführlich im Abschnitt 3.3 beschrieben. Zur Berechnung der Koeffizienten der restringierten Funktion nutzen wir Formel (3.14). Wir betrachten die Restriktion von Level $l+1$ auf Level l. Mit P_m bezeichnen wir Prozessor m. Die Koeffizienten α_i sind inkonsistent gespeichert. Betrachtet man den Zähler in (3.14), so ergibt sich wegen dieser Speicherung und der Verteilung der ELEMENT–Strukturen auf den Prozessoren

$$\sum_i \alpha_i \sum_{T\in\mathcal{T}_{l+1}} (\phi_{i,l+1}, \phi_{k,l})_T = \sum_i \sum_m (P_m \to \alpha_i) \sum_m \left(\sum_{T\in\mathcal{T}_{l+1}, T\in P_m} (\phi_{i,l+1}, \phi_{k,l})_T \right) .$$

Der Zähler enthält zwei Summen über die Prozessoren, die nicht in eine Summe über die Prozessoren zusammengefaßt werden können. Damit hat man zur Berechnung des Zählers zwei Kommunikationen auszuführen. Wir starten, indem wir die Koeffizienten α_i in konsistente Werte überführen. Dann haben wir

$$\sum_i \sum_m (P_m \to \alpha_i) \sum_m \left(\sum_{T\in\mathcal{T}_{l+1}, T\in P_m} (\phi_{i,l+1}, \phi_{k,l})_T \right) = \sum_m \sum_i \left(\alpha_i \sum_{T\in\mathcal{T}_{l+1}, T\in P_m} (\phi_{i,l+1}, \phi_{k,l})_T \right) \quad (4.1)$$

zu berechnen. Dies wird in einer Schleife über alle ELEMENT–Strukturen von Level l (MASTER– und SLAVE–Element–Strukturen) getan. Die akkumulierten Werte werden in den EDGE–Strukturen der jeweiligen ELEMENT–Strukturen gespeichert. Ein Teil der Werte befindet sich allerdings noch in den EDGE–Strukturen der SLAVE–Element–Strukturen von Level l. Diese werden mittels einer Kommunikation von den SLAVE– an ihre MASTER–ELEMENT–Strukturen übergeben und zu den Werten auf deren EDGE–Strukturen addiert. Der letzte Schritt der Restriktion besteht in einer Schleife über die ELEMENT–Strukturen von Level l, bei der der Quotient (3.14) ohne weitere Kommunikation berechnet wird. Das Ergebnis ist eine inkonsistente Speicherung der Koeffizienten β_k der restringierten Funktion.

Die Eingangsdaten der Prolongation sind konsistent gespeichert. Wir betrachten den Zähler der zu berechnenden Werte α_k aus (3.16). Es ist

$$\sum_j \beta_j \sum_m \sum_{T\in\mathcal{T}_{l+1}, T\in P_m} (\phi_{j,l}, \phi_{k,l+1})_T = \sum_m \sum_j \left(\beta_j \sum_{T\in\mathcal{T}_{l+1}, T\in P_m} (\phi_{j,l}, \phi_{k,l+1})_T \right) .$$

Diese Darstellung ähnelt (4.1). Mit einer Schleife über alle ELEMENT–Strukturen von Level l (auch den SLAVE–ELEMENT–Strukturen) kann man die Summen über den Index j auf den einzelnen Prozessoren berechnen. Dazu ist es jedoch nötig, daß vorher die Koeffizienten β_j zu den EDGE–Strukturen der SLAVE–ELEMENT–Strukturen übertragen werden. Da das Ergebnis der Prolongation wieder konsistent sein soll, muß der Zähler in (3.16) zum Schluß in einen konsistenten Vektor konvertiert werden. Nach dieser zweiten Kommunikation läßt sich α_k mittels einer Schleife über die EDGE–Strukturen auf Level $l+1$ und den vor der Mehrgitteriteration berechneten Werten $(\phi_{k,l+1}, \phi_{k,l+1})$ ohne weitere Kommunikation berechnen.

Der Grobgitterlöser

Auf dem gröbsten Gitter werden zwei Fälle unterschieden. Bei einfachen Gebieten und Problemen ist das Grobgitter im allgemeinen so klein, daß es auf einem Prozessor abgespeichert wird. In diesem Fall wird das Grobgittersystem direkt mit Gauß–Elimination und Spaltenpivotsuche gelöst.

Bei komplizierteren Geometrien und Problemen können mit dem Grobgitter bereits mehrere hundert (oder sogar tausend) Unbekannte verbunden sein. Das führt dazu, daß auch auf dem Grobgitter ein iterativer Löser eingesetzt werden muß. Zudem ist das Grobgitter im allgemeinen auf mehrere Prozessoren verteilt. In diesem Fall verwenden wir eine der im Abschnitt 3.3 vorgestellten Glättungsiterationen, um die Norm des Residuums auf dem Grobgitter zu verringern.

Hierbei unterscheiden wir wiederum zwei Herangehensweisen. Die erste versucht, die Norm des Residuums um einen vorgegebenen Faktor zu verringern. Dann muß man als zusätzliche Operation in jedem Iterationsschritt diese Norm zur Kontrolle der Iteration berechnen. Dies entspricht der normalen Vorgehensweise auf einem seriellen Computer. Auf einem Parallelrechner gibt es dabei jedoch Nachteile. Die Normberechnung ist jedes Mal mit einer Kommunikation verbunden. Verglichen mit der Gesamtmenge befinden sich auf dem Grobgitter wenig Daten. Es ist eine a priori unbekannte Anzahl von Iterationen durchzuführen, bei der die Prozessoren wenig zu rechnen, aber viel zu kommunizieren haben, falls das Grobgitter gut verteilt ist. Ist das Grobgitter nur auf wenige Prozessoren verteilt, so haben diese zu arbeiten und die anderen müssen warten. Beide Alternativen sind schlecht. Deshalb besteht unsere zweite Herangehensweise darin, auf dem Grobgitter möglichst wenig zu tun, das heißt nur eine geringe Anzahl von Iterationen durchzuführen und unabhängig von der erzielten Reduktion der Norm des Residuums das Grobgitter wieder zu verlassen. Bei dieser Herangehensweise ist es nicht mehr nötig, zur Kontrolle der Iteration die Norm des Residuums zu berechnen.

Das Verfahren

Das multiplikative Mehrgitterverfahren für die Unbekannten der Geschwindigkeiten hat damit folgende Gestalt :

```
Mehrgitter_fuer_Geschwindigkeit(level)                /* Kommunikationen */
{
   if (level==0)
      {
         berechne Norm des Residuums                                   (1)
         while (Norm zu gross)
            {
               mache Residuum konsistent                               (1)
               wende Glaetter an
               mittele Ergebnis auf den Interfaces                     (1)
```

```
                addiere gedaempftes Ergebnis zu den vorhandenen Werten
                berechne Residuum
                berechne Norm des Residuums                          (1)
             }
         }
      else
         {
            for (i=0; i<Vorglaettungsschritte; i++)
             {
                mache Residuum konsistent                            (1)
                wende Glaetter an
                mittele Ergebnis auf den Interfaces                  (1)
                addiere gedaempftes Ergebnis zu den vorhandenen Werten
                berechne Residuum
             }
            restringiere Residuum                                    (2)
            for (i=0;i<gamma;i++) Mehrgitter_fuer_Geschwindigkeit(level-1);
            interpoliere Korrektur                                   (2)
            for (i=0; i<Nachglaettungsschritte; i++)
             {
                berechne Residuum
                mache Residuum konsistent                            (1)
                wende Glaetter an
                mittele Ergebnis auf den Interfaces                  (1)
                addiere gedaempftes Ergebnis zu den vorhandenen Werten
             }
         }
}
```

In diesem Schema sind die zweite Variante des Glätters und die erste Variante der Grobgitterkorrektur berücksichtigt. Wir verwenden immer die euklidische Norm für das Residuum.

Bei der Verwendung des W–Zyklus hat man viel Arbeit auf den unteren Leveln zu investieren. Auf all diesen Leveln tritt das gleiche Problem wie beim Grobgitter auf, nämlich daß sowohl eine gute Verteilung der Unbekannten dieser Level auf den Prozessoren als auch die Konzentration dieser Unbekannten auf wenigen Prozessoren Nachteile auf Parallelarchitekturen besitzen. Aus diesem Grunde muß man beim W–Zyklus von vornherein mit Effizienzverlusten bei steigender Prozessoranzahl rechnen.

Bei lokaler Verfeinerung und der Verwendung des F– oder W–Zyklus gibt es Top–EDGE–Strukturen, die mehreren Vor– und Nachglättungsiterationen unterworfen sind. Dies sind die Top–EDGE–Strukturen auf unteren Leveln. Die Gesamtkorrektur ergibt sich als Summe der Korrekturen nach den Nachglättungsiterationen. Die dafür vorgesehen Speicherplätze werden bei der folgenden Vorglättungsiteration überschrieben. Aus diesem Grunde ist die Korrektur der Nachglättungsiterationen extra abzuspeichern. Dies ist bei gleichmäßiger Verfeinerung und beliebiger Zyklusart sowie bei lokaler Verfeinerung und V–Zyklus nicht nötig! Wir wollen diesen zusätzlichen Speicherplatz nicht bereitstellen und verwenden als Alternative vereinfachte F– und W–Zyklen. Diese führen F– beziehungsweise W–zyklusartige Iterationen nur auf solchen Leveln durch, die das gesamte Gebiet Ω überdecken. Ansonsten verhält er sich wie ein V–Zyklus. Mit $BASELEVEL$ bezeichnen wir das feinste Level, dessen Gitter Ω überdeckt. Dann wird vor dem rekursiven Aufruf des Mehrgitterverfahrens die Zeile

```
if (level > BASELEVEL+1) then gamma = 1
```

eingefügt.

Die Schrittweitenkontrolle

Die Schrittweitenkontrolle erfolgt nach dem Mehrgitterzyklus. Das Anfangsresiduum $b - Au^i$, siehe (3.17), ist immer noch gespeichert. So hat man nur eine Matrix–Vektor Multiplikation $A\delta u^{i+1}$ und die Berechnung der beiden Skalarprodukte durchzuführen. Sowohl das Anfangsresiduum als auch das Produkt der Matrix–Vektor Multiplikation sind inkonsistent gespeichert. Für eine genaue Berechnung des Faktors der Schrittweitenkontrolle werden beide Vektoren mit einer gemeinsamen Kommunikation konsistent gemacht und bei der Berechnung der Skalarprodukte nur die MASTER–EDGE–Struktur berücksichtigt. Da man bei der Berechnung von (3.17) Zähler und Nenner noch als Summe über die Prozessoren berechnen muß, was ebenfalls mit einer Kommunikation geschieht, hat man insgesamt 2 Kommunikationen auszuführen.

Überblick

Zum Abschluß dieses Abschnittes geben wir in Tabelle 4.2 einen Überblick über die Anzahl der Kommunikationen in den einzelnen Komponenten des Verfahrens. Diese Anzahl gibt einen

	Anzahl	Anzahl/Level
Glätter	2	$\nu_1 + \nu_2$
Residuum	0	$\nu_1 + \nu_2$
Restriktion	2	1
Prolongation	2	1
Schrittweitenkontrolle	2	1
Grobgitter (Var. 1)	3	nicht bekannt
Grobgitter (Var. 2)	2	gering, vorgegeben

Tabelle 4.2: Anzahl der Kommunikationen im Mehrgitter für die Geschwindigkeitsunbekannten.

gewissen Hinweis über die durch die Parallelität entstehenden Kosten des Algorithmus. Es ist jedoch zu beachten, daß die Länge der Kommunikationen, das heißt die zu transportierenden Datenmengen, auf den einzelnen Leveln unterschiedlich ist. Diese Längen hängen natürlich auch von der Verteilung der Elemente auf den Prozessoren und von der Länge der Interfaces ab. Aus diesem Grund ist es wenig aussagekräftig, die Anzahl der Kommunikationen, ähnlich wie die Anzahl der Operationen im Kapitel 5, zu addieren und die Summe als Maß für die parallelen Kosten zu betrachten. Auf diese Kosten an konkreten Beispielen wird bei den numerischen Tests in den Kapiteln 9 und 10 eingegangen.

4.4 Das Mehrgitterverfahren für die Druckunbekannten

Die Vorbereitung

Das Mehrgitterverfahren wird auf ein lineares System mit der Matrix $B^TC^{-1}B$ angewandt, die vor der Mehrgitteriteration explizit bereitgestellt werden muß. Die Unbekannten und die rechte Seite sind in den ELEMENT–Strukturen gespeichert, woraus folgt, daß die Speicherung konsistent ist.

Sei $C = \mathrm{diag}(A)$. Die Diagonale von A ist inkonsistent gespeichert, das heißt

$$(B^T C^{-1} B)_{ij} = \left(\sum_{k=1}^{N_u} b_{ki} b_{kj} \left(\sum_m (P_m \to a_{kk}) \right)^{-1} \right), i, j = 1, \ldots, N_p,$$

vergleiche (3.18). Die Elemente von $B^T C^{-1} B$ sollen konsistent gespeichert sein. Deswegen muß zuerst die Diagonale von A in einen konsistenten Vektor umgewandelt werden. Wählt man $C = I_{N_p}$ oder $C = M$, so entfällt diese Kommunikation hier. Falls die Konvertierung der Diagonalen von A in einen konsistenten Vektor an dieser Stelle geschieht, kann man sich diesen Schritt für die Vorbereitung des Glätters im Mehrgitterverfahren der Geschwindigkeitsunbekannten sparen. Die Matrixkomponente b_{ki} ist auf einer LINK–Struktur von der EDGE–Struktur k zur ELEMENT–Struktur i gespeichert. Nachdem die Diagonale von A konsistent ist, können alle Summanden in (3.18) berechnet werden, für die die EDGE–Struktur k und die anliegenden ELEMENT–Strukturen i und j sich auf demselben Prozessor befinden. Zur Berechnung der Summanden, die mit einer EDGE–Struktur auf einem Interface verbunden sind, ist eine zusätzliche Kommunikation erforderlich. Das Ergebnis ist eine konsistente Speicherung von $B^T C^{-1} B$. Die Diagonalelemente sind in den ELEMENT–Strukturen und die Nichtdiagonalelemente in EDGE–Strukturen gespeichert.

Zur Vorbereitung des Gauß–Seidel Glätters ist eine Umordnung und Durchnumerierung der ELEMENT–Strukturen nötig. Die Ordnungsstrategien sind wie im Mehrgitterverfahren der Geschwindigkeitsunbekannten. Um eine eindeutige Lösbarkeit des Gleichungssystems zu gewährleisten, wird ein zufällig bestimmter Druckwert fest gewählt.

Das Glätten und die Residuumsberechnung

Da alle Vektoren konsistent gespeichert sind, sind Ein– und Ausgabe des Blockglätters als konsistent definiert. Kommunikation ist für den Glättungsschritt nicht notwendig. Bei der Matrizenmultiplikation zur Residuumsberechnung werden zur Berechnung des Wertes auf einer ELEMENT–Struktur die Werte aller Nachbar–ELEMENT–Strukturen gebraucht. Deswegen ist zur Ausführung dieses Schrittes eine Kommunikation zwischen benachbarten ELEMENT–Strukturen auf verschiedenen Prozessoren erforderlich.

Die Gittertransferoperationen

Beim Gittertransfer hat man ebenfalls keine Probleme mit der Speicherungsart der Vektoren. Bei der Restriktion geben die MASTER–ELEMENT–Strukturen von Level $l+1$ ihre Werte, multipliziert mit ihrem Flächeninhalt, an ihre Vater–ELEMENT–Strukturen auf Level l, vergleiche mit dem Zähler in (3.20). Danach müssen die SLAVE–ELEMENT–Strukturen auf Level l die empfangenen Werte zu ihren MASTER–ELEMENT–Strukturen mittels einer Kommunikation übergeben. In einer Schleife über die MASTER–ELEMENT–Strukturen auf Level l wird schließlich der Quotient (3.20) berechnet.

Bei der Prolongation von Level l auf Level $l+1$ werden die zu prolongierenden Werte zuerst von den MASTER–ELEMENT–Strukturen auf Level l an ihre SLAVE–ELEMENT–Strukturen übergeben. Das erfordert eine Kommunikation. Im nächsten Schritt geben alle ELEMENT–Strukturen auf Level l die Werte zu ihren Sohn–ELEMENT–Strukturen. Diese addieren die erhaltenen Werte zu den Ergebnissen des Vorglättens.

Der Grobgitterlöser

Aus demselben Grund wie bei der Grobgitterkorrektur des Mehrgitterverfahrens der Geschwindigkeitsunbekannten unterscheiden wir auch hier zum einen die Herangehensweise, die die Norm des Residuums mittels hinreichend vieler Iterationen um einen vorgegebenen Faktor reduziert, und zum anderen die Herangehensweise, die nur wenige Iterationen auf dem Grobgitter investiert. Bei beiden Vorgehensweisen hat man in einem Iterationsschritt das Residuum zu berechnen (eine Kommunikation) und den Glätter anzuwenden. Bei der ersten Variante muß zusätzlich noch die Norm des Residuums zur Kontrolle der Iteration berechnet werden. Dafür ist eine Kommunikation notwendig.

Das Verfahren

Die Implementation des Mehrgitterverfahrens für die Druckunbekannten auf einem Parallelrechner hat damit die folgende Gestalt:

```
Mehrgitter_fuer_Druck(level)                          /* Kommunikationen */
{
   if (level==0)
      {
         berechne Norm des Residuums                                 (1)
         while (Norm zu gross)
            {
               wende Glaetter an
               addiere gedaempftes Ergebnis zu den vorhandenen Werten
               berechne Residuum                                     (1)
               berechne Norm des Residuums                           (1)
            }
      }
   else
      {
         for (i=0; i<Vorglaettungsschritte; i++)
            {
               wende Glaetter an
               addiere gedaempftes Ergebnis zu den vorhandenen Werten
               berechne Residuum                                     (1)
            }
         restringiere Residuum                                       (1)
         for (i=0;i<gamma;i++) Mehrgitter_fuer_Druck(level-1);
         interpoliere Korrektur                                      (1)
         for (i=0; i<Nachglaettungsschritte; i++)
            {
               berechne Residuum                                     (1)
               wende Glaetter an
               addiere gedaempftes Ergebnis zu den vorhandenen Werten
            }
      }
}
```

Hierbei ist die erste Variante des Grobgitterlösers berücksichtigt worden.

Die Schrittweitenkontrolle

Die Berechnung des Schrittweitenfaktors ist durch (3.22) gegeben. Das Anfangsresiduum ist noch gespeichert. Deshalb muß nur noch der Vektor $B^T C^{-1} B \delta p^{i+1}$ berechnet werden. Das erfordert eine Kommunikation. Zur Berechnung der beiden Skalarprodukte sind zwei Summen über die Prozessoren zu berechnen, was ebenfalls mit einer Kommunikation erfolgt.

Überblick

Zum Abschluß dieses Abschnittes geben wir in Tabelle 4.3 einen Überblick über die Anzahl der Kommunikationen in den einzelnen Komponenten des Verfahrens.

	Anzahl	Anzahl/Level
Glätter	0	$\mu_1 + \mu_2$
Residuum	1	$\mu_1 + \mu_2$
Restriktion	1	1
Prolongation	1	1
Schrittweitenkontrolle	2	1
Grobgitter (Var. 1)	2	nicht bekannt
Grobgitter (Var. 2)	1	gering, vorgegeben

Tabelle 4.3: Anzahl der Kommunikationen im Mehrgitter für die Druckunbekannten.

4.5 Das gekoppelte Mehrgitterverfahren

Die Vorbereitung

Die Effizienz des Vanka–Glätters hängt von der Reihenfolge der Unbekannten ab. Deswegen erfolgt zuerst eine Umordnung der ELEMENT–Strukturen gemäß einer gewählten lexikographischen Umordnungsstrategie. Desweiteren wird mittels einer Kommunikation die Diagonale von A konsistent gemacht und auf einem zusätzlichen Speicherplatz innerhalb der EDGE–Strukturen gespeichert. Darüber hinaus wird der Nenner in (3.23) berechnet, was mit einer Kommunikation pro Level verbunden ist.

Das Glätten und die Residuumsberechnung

Die Berechnung des Residuums erfordert eine Matrix–Vektor Multiplikation mit der Matrix $\mathcal{A}_l$. Das Resultat ist gemäß Tabelle 4.1 gespeichert. Überdeckt das gegenwärtige Level das gesamte Gebiet Ω, muß nach der Berechnung des Residuums die Konstante c in (3.23) berechnet werden, die die berechneten Komponenten des Residuums der Massenbilanz in den Raum $\operatorname{span}\{1_{N_p}\}^\perp$ projiziert. Das erfordert eine Kommunikation für den Zähler.

Ein– und Ausgabe des Glätters sind als konsistent gespeichert definiert. Dazu müssen zunächst die Geschwindigkeitskomponenten des Residuums konsistent gemacht werden. Beim Glätten selbst ist dank der vorhandenen konsistenten Diagonalen von A keine Kommunikation nötig. Es gibt wiederum die zwei in Abschnitt 4.3 beschriebenen Varianten, die EDGE–Strukturen auf den Interfaces zu behandeln. Wir wählen die Mittelung der Komponenten, wofür nach dem Glätten noch eine Kommunikation notwendig ist.

Die Gittertransferoperationen

Es müssen sowohl die die Geschwindigkeit betreffenden Werte, die in den EDGE–Strukturen, als auch die den Druck betreffenden Werte, die in den ELEMENT–Strukturen gespeichert sind, restringiert und prolongiert werden. Die Implementation der entsprechenden Routinen ist in den Abschnitten 4.3 und 4.4 beschrieben. Für das gekoppelte Mehrgitter werden diese Routinen, mit einem Unterschied, genutzt. Natürlich werden bei der Restriktion nicht zwei Kommunikationen von den SLAVE– zu den MASTER–ELEMENT–Strukturen zur Übertragung der jeweiligen Werte durchgeführt. Stattdessen werden die Werte gemeinsam mit einer Kommunikation übergeben, so daß man hier für die Restriktion insgesamt zwei Kommunikationen ausführen muß. Analog geht man bei der Prolongation vor, bei der auch zwei Kommunikationen auftreten.

Der Grobgitterlöser

Das Grobgittersystem wird iterativ mit Hilfe des Vanka–Glätters gelöst. Auch hier unterscheiden wir die im Abschnitt 4.3 beschriebenen Herangehensweisen. Die dortigen Ausführungen lassen sich wortwörtlich hierher übertragen.

Das Verfahren

Die Implementation des gekoppelten Mehrgitterverfahrens auf einem Parallelrechner hat damit die folgende Gestalt:

```
Gekoppeltes_Mehrgitter(level)                        /* Kommunikationen */
{
   if (level==0)
      {
         berechne Norm des Residuums                                (1)
         while (Norm zu gross)
            {
               mache Residuum konsistent                            (1)
               wende Glaetter an
               mittele Geschwindigkeitsunbekannte auf den Interfaces    (1)
               addiere gedaempftes Ergebnis zu den vorhandenen Werten
               berechne Residuum
               projiziere Druckkomponenten des Residuums            (1)
               berechne Norm des Residuums                          (1)
            }
      }
   else
      {
         for (i=0; i<Vorglaettungsschritte; i++)
            {
               mache Residuum konsistent                            (1)
               wende Glaetter an
               mittele Geschwindigkeitsunbekannte auf den Interfaces    (1)
               addiere gedaempftes Ergebnis zu den vorhandenen Werten
               berechne Residuum
               projiziere Druckkomponenten des Residuums            (1)
            }
         restringiere Residuum                                      (2)
         for (i=0;i<gamma;i++) Gekoppeltes_Mehrgitter(level-1);
         interpoliere Korrektur                                     (2)
         for (i=0; i<Nachglaettungsschritte; i++)
```

```
        {
           berechne Residuum
           projiziere Druckkomponenten des Residuums                 (1)
           mache Residuum konsistent                                 (1)
           wende Glaetter an
           mittele Geschwindigkeitsunbekannte auf den Interfaces     (1)
           addiere gedaempftes Ergebnis zu den vorhandenen Werten
        }
    }
}
```

Für die Verwendung des F– und W–Zyklus auf lokal verfeinerten Gittern treffen auch hier die im Abschnitt 4.3 gemachten Aussagen zu.

Die Schrittweitenkontrolle

Der Schrittweitenkontrollparameter wird nach (3.24) berechnet. Eine Matrix–Vektor Multiplikation mit anschließender Umwandlung des inkonsistenten Ergebnisses in ein konsistentes, die gleichzeitige Konvertierung des noch vorhandenen inkonsistenten Anfangsresiduums, sowie die abschließende Summation von Zähler und Nenner in (3.24) über die Prozessoren erfordern 2 Kommunikationen.

Überblick

Auch hier geben wir zum Abschluß in Tabelle 4.4 einen Überblick über die Anzahl der Kommunikationen in den einzelnen Komponenten des Verfahrens. Man stellt fest, daß die Anzahl der

	Anzahl	Anzahl/Level
Glätter	2	$\lambda_1 + \lambda_2$
Residuum	1	$\lambda_1 + \lambda_2$
Restriktion	2	1
Prolongation	2	1
Schrittweitenkontrolle	2	1
Grobgitter (Var. 1)	4	nicht bekannt
Grobgitter (Var. 2)	3	gering, vorgegeben

Tabelle 4.4: Anzahl der Kommunikationen für das gekoppelte Mehrgitter.

Kommunikationen pro Level nur unwesentlich höher ist, als beim Mehrgitterverfahren für die Geschwindigkeitsunbekannten. Die Länge mancher Kommunikationen, zum Beispiel der Austausch von MASTER– und SLAVE–ELEMENT–Strukturen beim Gittertransfer, ist hier allerdings doppelt so groß.

4.6 GMRES(restart) für die Druckunbekannten

Wählen wir in (3.10) $C^{-1} = A_{\gamma,j}^{-1}$, so verwenden wir GMRES(restart) zur Lösung dieses Gleichungsystems. Die Implementation folgt der Originalarbeit von Saad und Schultz [SS86]. Bei diesem Verfahren sind keine Maßnahmen gegen den Rangabfall von $B^T A_{\gamma,j}^{-1} B$ erforderlich.

Das Verfahren benötigt zusätzlichen Speicherplatz, dessen Größe von der Wahl des Parameters RESTART abhängt. Das ist zum einen eine Matrix der Dimension $N_p \times$ RESTART, die mit den Unbekannten verbunden ist. Diese wird deshalb genauso wie die Unbekannten auf die Prozessoren verteilt. Daneben werden eine Matrix und einige Vektoren gebraucht, deren Dimensionen allein von RESTART abhängen. Da RESTART im allgemeinen sehr viel kleiner als die Anzahl der Topelemente auf den einzelnen Prozessoren ist, werden diese Matrix und diese Vektoren vollständig auf allen Prozessoren gespeichert. Damit werden auch alle Operationen, die mit dieser Matrix und diesen Vektoren ausgeführt werden müssen, auf allen Prozessoren ausgeführt. Zusätzliche Vektoren der Länge N_p, die direkt mit den Unbekannten verbunden sind, werden innerhalb der ELEMENT–Strukturen gespeichert.

Da das Verfahren GMRES(restart) eine Eingittermethode ist, braucht man bezüglich des Druckes nur die Daten der Topelemente. Insbesondere kann man sich die Assemblierung des rechten oberen (beziehungsweise des linken unteren) Blockes von $\mathcal{A}_l$ auf allen Gitterebenen sparen, die kein Topelement besitzen.

Während einer Iteration hat man zwei Skalarprodukte zu berechnen, wofür jeweils kommuniziert werden muß. Hinzu kommen die Kommunikationen der Matrix–Vektor Multiplikationen, das sind die Kommunikationen der Anwendung von $A_{\gamma,j}^{-1}$. Diese sind ausführlich im Abschnitt 4.3 beschrieben.

Beim Restart des Verfahrens wird ebenfalls eine Matrix–Vektor Multiplikation benötigt, die mit den Kommunikationen der Anwendung von $A_{\gamma,j}^{-1}$ verbunden ist.

Insgesamt stellt man fest, daß dieses Verfahren sowohl bezüglich des Speicherbedarfes als auch in bezug auf die benötigten Kommunikationen ein aufwendiges Verfahren ist.

4.7 Das Verfahren der konjugierten Gradienten für die Druckunbekannten

Wir verwenden dieses Verfahren, neben dem Mehrgitterverfahren für die Druckunbekannten, falls C in (3.10) eine Diagonalmatrix ist. Das Verfahren ist wie im Buch von Axelsson und Barker [AB84] angegeben implementiert. Es muß gewährleistet sein, daß das Verfahren auf ein Gleichungssystem mit einer regulären Matrix angewandt wird. Dazu wählen wir einen zufällig herausgesuchten Druckwert fest und streichen die entsprechende Zeile und Spalte in $B^TC^{-1}B$.

Alle benötigten Vektoren sind direkt mit den Unbekannten verbunden und werden innerhalb der ELEMENT–Strukturen gespeichert. Da das CG–Verfahren, ebenso wie das Verfahren GMRES(restart), eine Eingittermethode ist, kann man bei der Assemblierung der Level ein wenig sparen, die kein Topelement besitzen.

Während einer Iteration müssen drei Kommunikationen ausgeführt werden. Zum einen sind zwei Skalarprodukte zu berechnen und zum anderen ist eine Matrix–Vektor Multiplikation mit der Matrix $B^TC^{-1}B$ durchzuführen. Daß dafür eine Kommunikation nötig ist, wurde bereits im Abschnitt 4.4 diskutiert.

Kapitel 5

Die numerischen Kosten der Lösungsverfahren

In diesem Abschnitt werden Abschätzungen für die Kosten der vorgestellten Lösungsverfahren angegeben. Dabei wird vorausgesetzt, daß die gegebene Triangulierung so fein ist, daß die Anzahl der Randkanten gegenüber der Anzahl der inneren Kanten vernachlässigt werden kann. Weiterhin soll die Anzahl der irregulären Verfeinerungen vernachlässigbar gegenüber der Anzahl der regulären Verfeinerungen sein. Diese beiden Voraussetzungen führen dazu, daß wir annehmen können, daß jedes Element drei Kanten mit Knoten sowie jedes Vaterelement vier Sohnelemente besitzt.

Die unten gegebenen Abschätzungen dienen als Anhaltspunkt für eine wechselseitige Einordnung der vorgestellten numerischen Verfahren bezüglich der Anzahl der numerischen Operationen. Es ist jedoch bekannt, daß die Laufzeit eines Programmes heutzutage nicht nur durch die Anzahl der numerischen Operationen, sondern auch entscheidend durch die Zeit bestimmt wird, die für Speicherzugriffe notwendig ist. Untersuchungen dazu wurden beispielsweise von Rüde [Rüd96] durchgeführt.

Es ist die Struktur der Matrix A zugrunde gelegt, die bei Upwind Diskretisierungen auftritt, siehe Abbildung 1.3.

5.1 Die Anzahl der Unbekannten

Wir nennen alle Elemente, die keine Söhne besitzen, *Topelemente* . Deren Anzahl sei N_{Topel}. Wir wollen die numerischen Kosten mit Hilfe von N_{Topel} angeben. Jedes Topelement besitzt drei Kanten, auf jeder Kante existiert ein zweidimensionaler Vektor der Geschwindigkeitsunbekannten, und jede Kante gehört zu zwei Elementen. Daraus folgt die Beziehung

$$N_u \approx 3N_p = 3N_{Topel}. \tag{5.1}$$

Für Mehrgitterverfahren ist auch die Gesamtanzahl der Unbekannten im Mehrgitter wichtig. Sei $N_{el,l}$ die Anzahl der Elemente auf Level l. Für gleichmäßige Verfeinerungen gilt $N_{el,l+1} = 4N_{el,l}$. Damit kann die Gesamtanzahl der Elemente durch

$$\sum_{l=0}^{L} N_{el,l} = \sum_{l=0}^{L} 4^{l-L} N_{el,L} < \frac{4}{3} N_{el,L} = \frac{4}{3} N_{Topel} \tag{5.2}$$

abgeschätzt werden. Bei adaptiver Verfeinerung sind die Topelemente über mehrere Level verteilt. Sei $N_{Topel,l}$ die Anzahl der Topelemente auf Level l. Dann gilt

$$\begin{aligned}
\sum_{l=0}^{L} N_{el,l} &= \sum_{l=0}^{L} N_{Topel,l} + \sum_{l=1}^{L} 4^{-1} N_{Topel,l} + \sum_{l=2}^{L} 4^{-2} N_{Topel,l} + \cdots + \sum_{l=L}^{L} 4^{-L} N_{Topel,l} \\
&\leq \sum_{l=0}^{L} N_{Topel,l} (1 + 4^{-1} + 4^{-2} + \ldots + 4^{-L}) < \frac{4}{3} \sum_{l=0}^{L} N_{Topel,l} = \frac{4}{3} N_{Topel}. \qquad (5.3)
\end{aligned}$$

Die numerischen Kosten werden in *opms* gemessen, das sind die Kosten für eine Multiplikation der eine Addition folgt, siehe Kiełbasinskiński und Schwetlick [KS88, S. 74]. Solch eine Operation bezeichnen wir mit *axpy*. In einigen Fällen werden der Einfachheit und Übersichtlichkeit halber, auch einzelne Multiplikationen als axpy gezählt, einzelne Additionen jedoch vernachlässigt.

5.2 Matrix–Vektor Produkte

Tabelle 5.1 gibt Auskunft über die Kosten verschiedener Matrix–Vektor Produkte. Bei der Abschätzung braucht man nur zu nutzen, daß eine Zeile von A im allgemeinen 5 Nichtnulleinträge und eine Zeile von B zwei Nichtnulleinträge besitzt sowie Beziehung (5.1) gilt.

Operation	Kosten in $N_{Topel} opms$
Au	15
Bp	6
$B^T u$	6
$\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix}$	27

Tabelle 5.1: Kosten für Matrix–Vektor Multiplikationen.

5.3 Das Mehrgitterverfahren für die Geschwindigkeitsunbekannten

Wir betrachten einen V–Zyklus und setzen voraus, daß für jede Unbekannte im Mehrgitter der gleiche Betrag an numerischer Arbeit geleistet werden muß. Werden auf jedem Gitter die gleiche Anzahl von Glättungsiterationen ausgeführt, so ist diese Annahme zumindest für die Unbekannten, die nicht zum Grobgitter gehören, realistisch. Mit den Beziehungen (5.1) und (5.2) beziehungsweise (5.3) kann die Gesamtanzahl der Geschwindigkeitsunbekannten mit $4N_{Topel}$ nach oben abgeschätzt werden.

Alle Arbeiten, die vor der Mehrgitteriteration zu tun sind (zum Beispiel die ILU–Zerlegung) werden vernachlässigt. Sei ν_1 die Anzahl der Vorglättungsschritte und ν_2 die Anzahl der Nachglättungsschritte. Wir nehmen an, daß diese Anzahl auf jedem Level gleich ist. Die anfallenden Kosten für eine Unbekannte sind in Tabelle 5.2 angegeben. Hinzu kommen die Kosten C_{gr} für die Lösung des Grobgitterproblems. Damit sind die Gesamtkosten einer Iteration im V–Zyklus

$$4\left[(\nu_1 + \nu_2)(C_{gl} + C_{rb}) + C_{re} + C_{pr} + C_{sk}\right] N_{Topel} + C_{gr} \quad opms. \qquad (5.4)$$

Operation	Kosten in *opms*	Anzahl
Glätten	C_{gl}	$\nu_1 + \nu_2$
Residuumberechnung	C_{rb}	$\nu_1 + \nu_2$
Restriktion	C_{re}	1
Prolongation	C_{pr}	1
Schrittweitenkontrolle	C_{sk}	1

Tabelle 5.2: Kosten für eine Unbekannte in einem V–Zyklus.

Manche Autoren, zum Beispiel Dorok [Dor95], verwenden zur zusätzlichen Kontrolle der Glättungsiteration die euklidische Norm des Residuums. Die Berechnung dieser Norm bedeutet jedoch zusätzliche numerische Kosten und vor allem Kommunikation auf Parellelrechnern. Wir verwenden die Kontrolle mit Hilfe dieser Norm lediglich auf dem Grobgitter. Auf allen anderen Gittern ist die vorgegebene maximale Anzahl der Glättungsiterationen im allgemeinen so klein, daß sich ein zusätzliches Abbruchkriterium nicht lohnt.

Wir nehmen nun an, daß C_{gr} vernachlässigt werden kann. Diese Annahme wird durch numerische Ergebnisse unterstützt. So beträgt der Anteil der Grobgitteriteration an der Gesamtzeit bei Rechnungen auf einem Prozessor nur 0.1% bei dem im Abschnitt 9.3 untersuchten Druckkorrektur Verfahren. Die Schrittweitenkontrolle wird nur auf dem feinsten Gitter durchgeführt. Deshalb ist (5.4) bezüglich C_{sk} eine Abschätzung nach oben.

Die Multiplikation mit A kostet für jede Unbekannte $5opms$. Zusammen mit der Aufdatierung kostet die Residuenberechnung $C_{rb} = 6opms$. Wir betrachten jetzt die Restriktion. Auf Grund der elementweisen Vorgehensweise bei der Berechnung des Zählers in (3.14), vergleiche Abschnitte 3.3 und 4.3, wird (bei gleichmäßiger Verfeinerung) der Wert im Knoten einer Kante auf Level l durch eine gewichtete Summe von 16 Werten berechnet. Hinzu kommt eine zusätzliche Division bei der Berechnung von β_k, so daß $C_{re} = 17opms$ gilt. Die Abarbeitungsweise von Restriktion und Prolongation ist ähnlich, es ist $C_{pr} = 17opms$.

Zur Abschätzung von C_{gl} müssen wir zwischen den verschiedenen Glättern unterscheiden. Nehmen wir ILU_β als Glätter, so haben wir eine Vorwärtselimination mit einer schwach besetzten Matrix (2 Nichtnulleinträge pro Zeile) sowie eine Rückwärtssubstitution mit einer schwach besetzten Matrix (3 Nichtnulleinträge pro Zeile) durchzuführen. Anschließend muß aufdatiert werden. In diesem Fall ist $C_{gl} = 6opms$.

Wir betrachten die Normalform (4.2.9b) des Gauß–Seidel Verfahrens aus Hackbusch [Hac91]. Dann hat man im ersten Iterationsschritt nur eine Rückwärtssubstitution mit einer schwach besetzten Matrix (3 Nichtnulleinträge pro Zeile) durchzuführen. Bei allen weiteren Iterationsschritten ist zudem der Defekt neu zu berechnen, das heißt eine Matrix–Vektor Multiplikation mit 5 *opms* pro Unbekannte. Mit der anschließenden Aufdatierung der Ergebnisse gilt für eine Gauß–Seidel Iteration $C_{gl} = 4opms$, für zwei $C_{gl} = 13opms$ usw. .

Die Schrittweitenkontrolle besteht aus einer Matrix–Vektor Multiplikation ($5opms$) und der Berechnung von zwei Skalarprodukten ($2opms$). Damit folgt $C_{sk} = 7opms$.

	C_{gl}	C_{rb}	C_{re}	C_{pr}	C_{sk}
opms	6	6	17	17	7

Tabelle 5.3: Kosten im Mehrgitterverfahren der Geschwindigkeitsunbekannten.

Eine von uns häufig verwendete Wahl ist $\nu_1 = \nu_2 = 3$ und der Glätter ILU_β. Damit ergeben sich für einen V–Zyklus die Kosten zu

$$4(6(6+6)+17+17+7)N_{Topel}opms = 452N_{Topel}opms.$$

Im W–Zyklus steigen die numerischen Kosten pro Unbekannte mit jedem Level in der Gitterhierarchie nach unten um den Faktor 2. Das ist hier sogar eine Abschätzung nach oben, da wir auf den unteren Leveln keine Schrittweitenkontrolle durchführen. Seien C_{na} die numerischen Kosten pro Element auf dem feinsten Gitter. Dann gilt für die Gesamtheit der numerischen Kosten die Abschätzung

$$\sum_{l=0}^{L} 2^{L-l} C_{na} N_{el,l} = C_{na} \sum_{l=0}^{L} \left(\frac{2}{4}\right)^{L-l} N_{el,L} < 2C_{na}N_{Topel} \tag{5.5}$$

für gleichmäßige Verfeinerungen. Für adaptive Verfeinerungen überlegt man sich analog zum V–Zyklus, daß Abschätzung (5.5) ebenfalls gilt. Die numerischen Kosten für die Kanten von Topelementen lassen sich auf gleiche Art und Weise wie beim V–Zyklus abschätzen. Die Kosten für einen W–Zyklus erhält man, indem man die Kosten eines V–Zyklus mit den gleichen Komponenten mit dem Faktor 1.5 multipliziert, vergleiche auch bei Hackbusch [Hac85].

Der F–Zyklus läßt sich nicht in einer rekursiven Form wie V– und W–Zyklus schreiben. Man startet mit dem Durchlaufen der Gitterhierarchie vom feinsten Gitter zum Grobgitter und fährt dann wie bei einem vollen Mehrgitter V–Zyklus fort. Für jedes Level ist die numerische Arbeit eines V–Zyklus zu leisten. Somit lassen sich die numerischen Kosten des F–Zyklus mit

$$\sum_{l=0}^{L} C_{na} N_{el,l} = C_{na} \sum_{l=0}^{L} \left(\frac{1}{4}\right)^{l} N_{el,L} < \frac{4}{3} C_{na} N_{Topel}$$

abschätzen. Die numerischen Kosten eines F–Zyklus erhält man, indem man die Kosten eines V–Zyklus mit den gleichen Komponenten mit dem Faktor 4/3 multipliziert.

5.4 Das Mehrgitterverfahren für die Druckunbekannten

Wir betrachten einen V–Zyklus. Dann gibt Tabelle 5.2 Aufschluß über die Kosten pro Unbekannte.

Die Matrix $B^T C^{-1} B$ ist explizit vorhanden. Sie ist symmetrisch und in einer Reihe (Spalte) sind im allgemeinen vier Nichtnullelemente. Die Berechnung des Residuums kostet demzufolge $C_{rb} = 5opms$. Desweiteren ist $C_{re} = 4opms$, da der Wert eines Vaterelementes ein gewichtetes Mittel der Werte von vier Sohnelementen ist. Bei der Prolongation muß der Wert vom Vaterelement an die Sohnelemente nur übergeben und zum Ergebnis des Vorglättens aufdatiert. Da Additionen vernachlässigt werden, ist $C_{pr} = 0opms$.

Ein Glättungsschritt mit einem Gauß–Seidel Verfahren besteht aus einer Rückwärtssubstitution, wobei in der Dreiecksmatrix höchstens drei Elemente pro Zeile von Null verschieden sind. Zusammen mit der anschließenden Aufdatierung des Ergebnisvektors ist $C_{gl} = 4opms$ eine Abschätzung nach oben. Die Schrittweitenkontrolle setzt sich abermals aus einer Matrix–Vektor Multiplikation und zwei Skalarprodukten zusammen. Es ist $C_{sk} = 6opms$. Eine Zusammenstellung der Kosten der einzelnen Komponenten ist in Tabelle 5.4 gegeben.

Seien μ_1, μ_2 die Anzahl der Vor- und Nachglättungsschritte. Wir wählen oft $\mu_1 = \mu_2 = 3$. Unter Vernachlässigung der Kosten für die Lösung des Grobgitterproblems sind die numerischen Kosten für einen V–Zyklus

$$4((3+3)(5+4)+4+6)\frac{N_{Topel}}{3}opms \approx 85N_{Topel}opms.$$

	C_{gl}	C_{rb}	C_{re}	C_{pr}	C_{sk}
$opms$	4	5	4	0	6

Tabelle 5.4: Kosten im Mehrgitterverfahren der Druckunbekannten.

Im Vergleich zum Mehrgitterverfahren für die Geschwindigkeitsunbekannten ist der numerische Aufwand pro Iteration für das Mehrgitterverfahren für die Druckunbekannten gering.

5.5 Das gekoppelte Mehrgitterverfahren

Bei der Abschätzung der Kosten für ein gekoppeltes Mehrgitterverfahren sind Differenzierungen zwischen den Unbekannten des Druckes und der Geschwindigkeit notwendig.

Wir beginnen mit den Druckunbekannten. Aus dem vorangegangenen Abschnitt wissen wir, daß $C_{re}(p) = 4opms$ und $C_{pr}(p) = 0opms$ sind. Diese Unbekannten werden mit B multipliziert ($6opms$). Hinzu kommt die Projektion der Werte des Residuums bezüglich des Druckes und die Aufdatierung. Bei der Projektion ist die Konstante c aus (3.23) zu berechnen. Da die Konstante $N_{el,l}$ vor der Iteration bereitgestellt wird und ansonsten nur Additionen auftreten, werden diese numerischen Kosten vernachlässigt. Es gilt $C_{rb}(p) = 7opms$. Der Vanka– und der stabilisierte Vanka–Glätter arbeiten elementweise. Deshalb ist es günstig, die Glättungskosten nur bezüglich der Druckunbekannten zu betrachten. Man hat sowohl Kosten bei der Assemblierung als auch bei der Lösung der lokalen Probleme. Beim Vanka–Glätter benötigt man $30opms$ zur Assemblierung und $31opms$ zur Lösung des linearen Systems pro Element. Für den stabilisierten Vanka–Glätter kostet die Assemblierung nur $18opms$ pro Element, die Lösung des lokalen Systems (mit Spaltenpivotisierung) dagegen $161opms$. Mit der Aufdatierung stehen $C_{gl}(p) = 62opms$ für den Vanka–Glätter und $C_{gl}(p) = 180opms$ pro Element für den stabilisierten Vanka–Glätter als Aufwand zu Buche.

Die Geschwindigkeitsunbekannten werden mit A ($5opms$) und B^T ($2opms$) multipliziert. Zusammen mit der Aufdatierung kostet das $C_{rb}(u) = 8opms$. Nach dem Glätten braucht man hier nur noch die Aufdatierung zu betrachten $C_{gl}(u) = 1opms$. Aus Abschnitt 5.2 sind $C_{re}(u) = 17opms$ sowie $C_{pr}(u) = 17opms$ bekannt. Beachtet man, daß die Anzahl der Geschwindigkeitsunbekannten das Dreifache der Anzahl der Druckunbekannten ist, so erhält man $C_{rb} = 31opms, C_{re} = 55opms, C_{pr} = 55opms$ und für den Vanka–Glätter $C_{gl} = 65opms$. Die Schrittweitenkontrolle besteht aus einer Matrix–Vektor Multiplikation ($27opms$) und zwei Skalarprodukten über alle Unbekannte ($8opms$), also $C_{sk} = 35opms$.

	C_{gl}	C_{rb}	C_{re}	C_{pr}	C_{sk}
$opms$	65	31	55	51	35

Tabelle 5.5: Kosten im gekoppelten Mehrgitterverfahren.

Nach (5.4) ergeben sich für einen V–Zyklus mit Vanka–Glätter, mit λ_1 Vorglättungsschritten und λ_2 Nachglättungsschritten (auf allen Leveln) die Kosten zu

$$\frac{4}{3}\left[(\lambda_1 + \lambda_2)(65 + 31) + 55 + 51 + 35\right] N_{topel} opms + C_{gr} opms. \tag{5.6}$$

5.6 GMRES(restart) für die Druckunbekannten

Das Verfahren GMRES(restart) verwenden wir zur Lösung des Gleichungssystems (3.11) bei der Wahl von $C^{-1} = A_{\gamma,j}^{-1}$.

Wir haben uns bei der Implementation an die Originalarbeit von Saad und Schultz [SS86] gehalten. Deshalb können wir die dort gemachten Abschätzungen für die numerischen Kosten übernehmen. Neben der Matrix–Vektor Multiplikation kostet ein Iterationsschritt durchschnittlich

$$\left(RESTART + 3 + \frac{1}{RESTART}\right) N_{Topel} opms.$$

Das Matrizenprodukt $B^T A_{\gamma,j}^{-1} B$ beinhaltet sowohl Mehrgitterschritte für die Geschwindigkeitsunbekannten, deren Kosten mit $C_{velo,V}$ bezeichnet werden, als auch Multiplikationen mit B. Somit ergeben sich die durchschnittlichen Kosten für einen Iterationsschritt zu

$$\left[RESTART + 3 + \frac{1}{RESTART} + 12 + C_{velo,V}\right] N_{Topel} opms.$$

Wir haben oft $C^{-1} = A_{1,1}^{-1}$ sowie $restart \in [10, 25]$ gewählt. Mit den Abschätzungen des Abschnittes 5.2 erhält man für diese Wahl $C_{velo,V} = 452 opms$. Die Kosten für eine Iteration mit GMRES(restart) belaufen sich damit auf $477 - 492 N_{Topel} opms$.

5.7 Das Verfahren der konjugierten Gradienten für die Druckunbekannten

Wählt man eine symmetrische Matrix C, so kann man (3.11) mit Hilfe des CG–Verfahrens lösen. Die Implementation des Verfahrens erfolgte nach einem Buch von Axelsson und Barker [AB84]. Im weiteren sei C eine Diagonalmatrix. Dann kann die Matrix $B^T C^{-1} B$ vor der Iteration, wie im Abschnitt 3.4 beschrieben, explizit bereitgestellt werden. Tabelle 5.6 gibt Auskunft über die während eines Iterationsschrittes auftretenden Operationen und ihre Kosten. Die Gesamtkosten für eine CG-Iteration belaufen sich auf $9 N_{Topel} opms$.

Operation	Kosten in $opms$	Anzahl	Kosten in $N_{Topel} opms$
$B^T C^{-1} B p$	$N_{Topel} + N_u$	1	4
Skalarprodukt	N_{Topel}	2	2
axpy	N_{Topel}	3	3

Tabelle 5.6: Kosten für eine cg–Iteration.

5.8 Die Kosten der Lösung des linearen Systems (3.1)

Wir beginnen mit den Druckkorrektur Verfahren. Bei diesen entstehen Kosten bei der Richardson Iteration (3.10) und bei der Berechnung der neuen Korrektur für die Geschwindigkeit (3.9) beziehungsweise (3.12).

Zunächst wird eine Richardson Iteration betrachtet. In dieser wird $A_{\gamma,j}^{-1}$ auf einen Vektor angewandt, mit B^T multipliziert, das Gleichungsystem (3.11) gelöst und das Ergebnis zur bisherigen Iterierten aufdatiert. Mit M bezeichnen wir die Anzahl der Richardson Iterationen und mit L die Anzahl der Iterationen, die zur Lösung von (3.11) genutzt werden.

Berechnet man die Korrektur für die Geschwindigkeit aus der Impulsbilanz (3.9), so sind noch eine Multiplikation mit B und eine Anwendung von $A_{\gamma,j}^{-1}$ auf einen Vektor durchzuführen. Nutzt man stattdessen die Massenbilanz (3.12), so muß man eine Multiplikation mit B, eine Multiplikation mit der Diagonalmatrix C^{-1} und eine axpy–Operation ausführen.

In Tabelle 5.7 sind für verschiedene Anzahlen von Iterationen die Anzahl der Operationen zusammengestellt. Dabei werden für die Mehrgitterverfahren die in den Abschnitten 5.2 und 5.4 gewählten Parameter zugrunde gelegt. Dann berechnen sich die Kosten einer Richardson Iteration zu $\mathsf{M}(453{+}\mathsf{L}C_{ite})N_{Topel}opms$, wobei C_{ite} die Kosten der iterativen Lösung von (3.11) sind. Die Lösung von (3.9) kostet $452N_{Topel}opms$ und die von (3.12) $10N_{Topel}opms$. Bei GMRES(restart) werden die Anzahl der Operationen für $RESTART = 10$ angegeben.

		Mehrgitter		CG		GMRES(10)	
M	L	(3.9)	(3.12)	(3.9)	(3.12)	(3.9)	(3.12)
5	3	3992	3550	2852	2410	2837	2395
5	5	4842	4400	2942	2500		
5	10	6967	6525	3167	2725		
10	3	7532	7090	5252	4810	5222	4780
10	5	9232	8790	5432	4990		
10	10	13482	13040	5882	5440		
20	3	14612	14170	10052	9610	9992	9550
20	5	18012	17570	10412	9970		
20	10	26512	26070	11312	10870		

Tabelle 5.7: Die Kosten in $N_{Topel}opms$ für die Lösung von (3.1) mit Druckkorrektur Verfahren und V–Zyklen in den Mehrgitterverfahren.

Zur Berechnung der Kosten bei der Verwendung des gekoppelten Mehrgitterverfahrens mit Vanka Glätter muß man in (5.6) die Anzahl der Glättungsschritte einsetzen und mit der Anzahl der Zyklen M multiplizieren. In Tabelle 5.8 sind die Kosten für verschiedene Parameter präsentiert.

M	$\lambda_1 = \lambda_2$	V–Zyklus	F–Zyklus	W–Zyklus
5	3	4780	6373	7170
5	5	7340	9787	11010
5	10	13740	18320	20610
10	3	9560	12747	14340
10	5	14680	19573	22020
10	10	27480	36640	41220

Tabelle 5.8: Die Kosten in $N_{Topel}opms$ für die Lösung von (3.1) mit gekoppelten Mehrgitterverfahren und Vanka Glätter.

Alle vorgestellten Verfahren berechnen nur eine Näherungslösung für das System (3.1). Deshalb sind nicht die numerischen Kosten, sondern das Verhältnis der Qualität der Näherungslösung zu den numerischen Kosten das entscheidende Kriterium für die Beurteilung eines Verfahrens. Numerische Untersuchungen zu dieser Fragestellung findet man in den Abschnitten 9.3 und 10.3.

Kapitel 6

A Posteriori Fehlerabschätzungen für Konvektions–Diffusions Gleichungen

Dieses Kapitel beginnt mit einer Einführung über die Bedeutung und Aufgaben von a posteriori Fehlerschätzern. Es werden bisher bekannte Ergebnisse über residuenbasierte a posteriori Fehlerschätzer für nichtkonforme Finite Elemente Diskretisierungen zitiert und die in diesem Kapitel präsentierten Ergebnisse eingeordnet. Abschnitt 6.2 enthält die für dieses Kapitel benötigten Voraussetzungen und Bezeichnungen. Abschnitt 6.3 befaßt sich mit Abschneidefunktionen, die ein wesentliches Hilfsmittel zur Untersuchung der lokalen a posteriori Fehlerschätzer in den Abschnitten 6.4 und 6.5 sind. Im Abschnitt 6.6 wird ein globaler a posteriori Fehlerschätzer in der L^2–Norm studiert. Den Abschluß dieses Kapitels bilden Ausführungen zur Umsetzung adaptiver Methoden auf Computern.

6.1 Die Aufgaben von a posteriori Fehlerschätzern

Der Einsatz adaptiver Methoden für die präzise und schnelle numerische Lösung partieller Differentialgleichungen, die zum Beispiel aus der Strömungsmechanik kommen, ist während der letzten beiden Jahrzehnte rapide gewachsen. Adaptive Methoden haben das Ziel, die numerische Lösung einer Gleichung innerhalb einer vorgeschriebenen Fehlertoleranz mit minimalem numerischen Aufwand zu berechnen. A posteriori Fehlerschätzer sind die wichtigsten Werkzeuge in adaptiven Methoden. Vom praktischen Standpunkt sind zweiseitige Abschätzungen wünschenswert, eine lokale Abschätzung nach unten und eine globale Abschätzung nach oben. Mit Hilfe der lokalen Abschätzung nach unten wird die adaptive Gittergenerierung gesteuert. Die globale Abschätzung dient als Maß für die Qualität der berechneten Lösung und somit als Abbruchkriterium für den Lösungsprozeß.

Seien eine partielle Differentialgleichung mit Lösung u, eine zugehörige diskrete Gleichung mit Lösung u_h, eine Norm $\|\cdot\|$ sowie ein elementweise berechneter a posteriori Fehlerschätzer η_T gegeben. Die gängige Praxis besteht darin, bei Elementen mit großem geschätzten Fehler η_T verstärkt numerische Arbeit zu investieren, um das bisher berechnete Ergebnis zu verbessern. Diese verstärkte numerische Arbeit kann sich äußern in

- adaptiver Gitterverfeinerung, h–Methode,
- Erhöhung des Grades der lokalen Approximation, p–Methode,
- Gittermanipulationen wie Ausrichtung oder Streckung,
- Kombinationen der obigen Methoden.

Wir verwenden die adaptive Gitterverfeinerung.

Sei η ein globaler a posteriori Fehlerschätzer.

Definition 6.1.1 *Der Effizienzindex eines Fehlerschätzers ist der Quotient zwischen geschätztem und wahrem Fehler:*

$$I_{eff} := \frac{\eta}{\|u - u_h\|_\Omega}.$$

Ein Fehlerschätzer heißt effizient, falls I_{eff} und I_{eff}^{-1} für alle Triangulierungen beschränkt sind. Er heißt asymptotisch exakt, falls $I_{eff} \to 1$ für $h = \max_{T \in \mathcal{T}_h}\{h_T\} \to 0$ gilt.

Ein effizienter Fehlerschätzer heißt robust für eine Problemklasse, falls I_{eff} und I_{eff}^{-1} unabhängig vom gestellten Problem beschränkt sind, insbesondere unabhängig von den Koeffizientenfunktionen.

Die Robustheit von Fehlerschätzern bezüglich des Diffusionsparameters ν in Konvektions–Diffusions und Navier–Stokes Gleichungen ist ein ernsthaftes Problem. Es stellt sich heraus, daß diese Frage eng mit der Stabilitätskonstanten des dualen Problems verknüpft ist. Einzig Angermann stellt in [Ang95] einen robusten Fehlerschätzer für Konvektions-Diffusions Gleichungen vor, jedoch für eine relativ komplizierte Operatornorm. Für die Navier–Stokes Gleichungen haben sich vor allem Johnson und Rannacher um erste Ergebnisse verdient gemacht, siehe [JR94], [JRB95] und [Joh95b].

Im Laufe der letzten beiden Jahrzehnte wurden verschiedene Typen von a posteriori Fehlerschätzern entwickelt. Einen guten Überblick bietet Verfürth in [Ver96b]. Wir befassen uns in diesem Kapitel mit sogenannten *residuenbasierten a posteriori Fehlerschätzern.* Diese beinhalten das Residuum der starken Formulierung der zu lösenden Differentialgleichung und Terme, die die Verbindung von starker und schwacher Formulierung darstellen. Das sind im allgemeinen Terme mit Sprüngen von Funktionen oder deren Ableitungen über Elementkanten. Zur analytischen Untersuchung dieser Art von Fehlerschätzern gibt es im wesentlichen zwei Beweistechniken. Eine geht auf Verfürth [Ver94b], [Ver94a], [Ver96b] und die andere auf Johnson [EEHJ95] zurück. Die von Verfürth entwickelte Beweistechnik startet mit einer allgemeinen Äquivalenz zwischen Fehler und Residuum in einer Umgebung der Lösung in geeignet gewählten Banachräumen. Die dabei auftretende Norm des Residuums ist allerdings numerisch nicht berechenbar. Nun wird ein geeignet gewählter endlichdimensionaler Raum gesucht, in dem man diese Norm so approximieren kann, daß der dabei auftretende Fehler von höherer Ordnung ist. In dem endlichdimensionalen Raum werden im wesentlichen partielle Integration, die Galerkin Orthogonalität des diskreten Problems und Interpolationsabschätzungen als technische Hilfsmittel benötigt. Dabei ist oft ein recht umfangreicher und komplizierter technischer Apparat notwendig. Mit der Beweistechnik von Verfürth werden sowohl lokale Abschätzungen nach unten als auch globale Abschätzungen nach oben bewiesen. Einige Züge dieser Technik verwenden wir bei den lokalen Fehlerabschätzungen in den Abschnitten 6.4 und 6.5. Mit der Technik von Johnson werden globale Abschätzungen nach oben gezeigt. Da wir diese Technik im Abschnitt 6.6 verwenden, wird sie dort näher erläutert. Mit Hilfe beider Techniken wurden a posteriori Fehlerschätzer für Konvektions–Diffusions und Navier–Stokes Gleichungen mit konformen Diskretisierungen entwickelt. Diese besitzen bei der jeweiligen Problemklasse die gleiche Gestalt. Die Nichtlinearität der Navier–Stokes Gleichungen spielt beweistechnisch eine untergeordnete Rolle, für detaillierte Beweise siehe auch [Joh95a] und Roos, Stynes, Tobiska [RST96].

Um die Ergebnisse auf nichtkonforme Diskretisierungen übertragen zu können, ist eine zusätzliche Kontrolle des Konsistenzfehlers erforderlich. Dari et.all [DDPV96], [DDP95] haben gezeigt, daß zusätzliche Terme in globalen Fehlerschätzern für diese Kontrolle notwendig sind. Sie leiteten a posteriori Fehlerschätzer in der Energienorm für die Laplace Gleichung [DDPV96] und für die Stokes Gleichungen [DDP95] her. Der zusätzliche Term ist eine Summe von Sprüngen der Tangentialableitung der nichtkonformen Funktion über die Kanten. Fehlerschätzer für die Energienorm und die nichtkonforme P_1–Diskretisierung der Laplace Gleichung werden von Wohlmuth in [Woh95] betrachtet. Eine wichtige Rolle in den dortigen Beweisen spielt eine Saturationsannahme bezüglich der Lösung u des stetigen Problems und der Lösung eines konformen diskreten Problems in einem Ansatzraum vom lokalen Polynomgrad zwei.

In diesem Kapitel betrachten wir die nichtkonforme P_1–Finite Elemente Diskretisierung von Konvektions–Diffusions Gleichungen. Es werden lokale a posteriori Fehlerschätzer in der diskreten H^1–Norm (1.9) und in der L^2–Norm sowie ein globaler Fehlerschätzer in der L^2–Norm hergeleitet. Die lokalen Fehlerschätzer in der diskreten H^1–Norm sind Erweiterungen der Ergebnisse von Wohlmuth [Woh95] und Dari et.all [DDPV96] von Laplace auf Konvektions–Diffusions Gleichungen. Die Fehlerschätzer in der L^2–Norm sind neu.

6.2 Voraussetzungen und Notationen

In diesem Kapitel werden der Einfachheit halber homogene Dirichletprobleme betrachtet. Für die Lösung wird

$$u \in H^2(\Omega) \cap H_0^1(\Omega) \tag{6.1}$$

vorausgesetzt. Wir definieren für $v \in H_0^1(\Omega)$

$$\mathrm{curl} v := \begin{pmatrix} -\frac{\partial v}{\partial y} \\ \frac{\partial v}{\partial x} \end{pmatrix}.$$

Die Dreiecke T_1 und T_2 seien benachbart mit der gemeinsamen Kante E. Zu E gehört die Normale $n_E = (n_1, n_2)$, die der Eindeutigkeit halber von T_1 nach T_2 gerichtet ist. Der Sprung einer Funktion $v_h \in V_{h0}$ über E wird mit

$$[|v_h|]_E = v_h|_{T_1} - v_h|_{T_2}$$

und die Tangente an E mit $t_E = (-n_2, n_1)$ bezeichnet.

Wir setzen eine reguläre Triangulierung $\mathcal{T}_h$ voraus, siehe Ciarlet [Cia91], Chapter 17, (H1). Das heißt, es existiert eine von h unabhängige Konstante C, so daß

$$\frac{h_T}{\rho_T} \le C \quad \text{oder} \quad \frac{h_T}{h_E} \le C \quad \forall T \in \mathcal{T}_h, \forall E \subset \partial T \tag{6.2}$$

gilt. Hierbei ist h_T der Durchmesser (längste Seite) von T, h_E die Länge der Kante E und ρ_T der Durchmesser des Inkreises von T. Durch Bedingung (6.2) ist der kleinste Innenwinkel der Dreiecke nach unten durch eine von der Triangulierung unabhängige Konstante beschränkt.

Mit ω_E bezeichnen wir die zwei Dreiecke T und T_1, die die gemeinsame Kante E besitzen. Mit ω_T wird die Vereinigung der Dreiecke bezeichnet, die eine gemeinsame Kante mit dem Dreieck T besitzen, vergleiche Abbildung 6.1 für typische Situationen im Inneren von Ω.

Desweiteren sei X_h der Raum der konformen, stückweise linearen Funktionen, die auf $\partial\Omega$ verschwinden. Es gilt $X_h \subset V_{h0}$. Mit $R_{X_h} : H^2(\Omega) \mapsto X_h$ bezeichnen wir den Standard–Knoten–Interpolationsoperator. Es gelten die Interpolationsabschätzungen

$$\|v - R_{X_h} v\|_{0,T} \le C_1 h_T^2 \|v\|_{2,T} \; \forall v \in H^2(T), \tag{6.3}$$

$$\|v - R_{X_h} v\|_{0,E} \le C_2 h_E^{3/2} \|v\|_{2,T} \; \forall v \in H^2(T), E \subset \partial T. \tag{6.4}$$

Die Konstanten C_1 und C_2 hängen nur von der Triangulierung, genauer vom kleinsten Innenwinkel der Dreiecke, ab. Wir setzen $C_I := \max\{C_1, C_2\}$.

Die Funktionen aus V_{h0} erfüllen

$$\int_E q\, [|v_h|]_E \, ds = 0 \quad \forall q \in P_0(E), v_h \in V_{h0}. \tag{6.5}$$

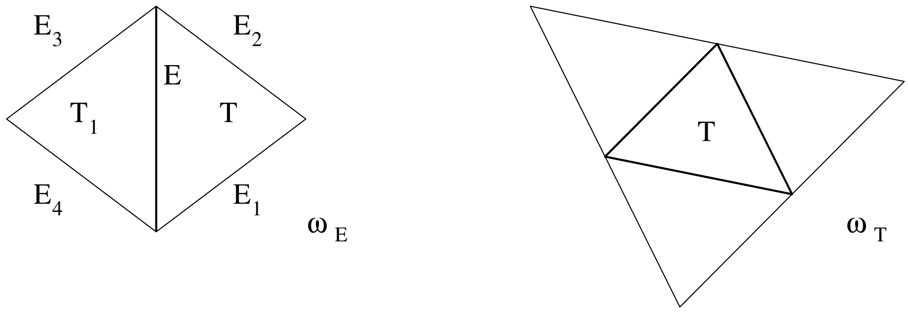


Abbildung 6.1: Die lokalen Gebiete ω_E und ω_T.

Mit $\mathcal{M}_E^k : H^1(T) \mapsto P_k(E)$ wird die L^2–Projektion der Restriktion auf E von $P_k(T)$ bezeichnet. Es gilt, siehe John, Maubach, Tobiska [JMT96], Lemma 2,

$$\left| \int_E u(v - \mathcal{M}_E^k v)\, ds \right| \leq C h_T^{m+1/2} \|u\|_{0,E} |v|_{m+1,T} \tag{6.6}$$

mit $0 \leq m \leq k$, für alle Elemente $T \subset \omega_E$, $u \in L^2(E)$ und $v \in H^{m+1}(T)$.

Im folgenden werden häufig inverse Abschätzungen von Normen und Seminormen für Polynome in Sobolevräumen benötigt.

Lemma 6.2.1 *Sei eine reguläre Triangulierung $\mathcal{T}_h$ gegeben. Sei $\omega = T$ oder $\omega = \omega_E$ oder $\omega = \omega_T$. Falls $\omega = \omega_E$ ist, so ist $\mathsf{h} = h_E$, anderenfalls $\mathsf{h} = h_T$.*

i) Für $m \geq 1, p \in [1, \infty]$ gilt

$$\sum_{T \subset \omega} |v|_{m,p,T} \leq \frac{C}{\mathsf{h}} \sum_{T \subset \omega} |v|_{m-1,p,T} \quad \forall v \text{ mit } v|_T \in P_k(T), \forall T \subset \omega. \tag{6.7}$$

ii) Für $m \geq 0, p \in [1, \infty], q \in [1, \infty], p \geq q$ gilt

$$\sum_{T \subset \omega} |v|_{m,p,T} \leq C \mathsf{h}^{2/p - 2/q} \sum_{T \subset \omega} |v|_{m,q,T} \quad \forall v \text{ mit } v|_T \in P_k(T), \forall T \subset \omega. \tag{6.8}$$

iii) Für alle Elemente $T \in \mathcal{T}_h$ und alle Kanten $E \subset \partial T$ gilt

$$\|v|_E\|_{0,E} \leq C h_E^{-1/2} \|v\|_{0,T} \quad \forall v \in P_k(T). \tag{6.9}$$

Beweis: Die Beweis von i) und ii) folgt Ciarlet [Cia91], Theorem 17.2. Jedes Element T wird durch eine affine Transformation auf ein Referenzelement abgebildet und dort wird die Äquivalenz von Normen in endlichdimensionalen Räumen genutzt. Die Wahl $\mathsf{h} = h_E$ für $\omega = \omega_E$ sowie $\mathsf{h} = h_T$ für $\omega = \omega_T$ ist wegen der Regularität der Triangulierung möglich. Zum Beweis von iii) benötigt man noch eine Standardabschätzung der Spur von v. $\square$

Nicht von h_T, h_E und ν abhängige positive Konstanten werden mit C bezeichnet.

6.3 Abschneidefunktionen

In diesem Abschnitt werden Funktionen eingeführt, deren Träger nur aus einem oder zwei Dreiecken einer gegebenen Triangulierung bestehen und die zudem gewisse Glattheitsbedingungen erfüllen. Diese Funktionen werden *Abschneidefunktionen* genannt. Das Produkt solch einer Funktion mit einer in ganz Ω definierten Funktion v schneidet sozusagen alle Teile von v ab, die nicht zum Träger der Abschneidefunktion gehören. Abschneidefunktionen sind ein Standardwerkzeug zum Beweis lokaler Fehlerabschätzungen, siehe zum Beispiel bei Verfürth [Ver96b] und [Ver96a].

Sei $T \in \mathcal{T}_h$ ein beliebiges Dreieck und $E \subset \partial T$. Dann bezeichnen wir mit λ_E diejenige lineare Funktion, die auf E identisch verschwindet und im Eckpunkt von T, der E gegenüber liegt, den Wert 1 annimmt. Mit den Bezeichnungen von Abbildung 6.1 definieren wir die sogenannten *Blasenfunktionen*

$$\mathrm{B}_T = \begin{cases} 27\lambda_E\lambda_{E_1}\lambda_{E_2} & \text{in } T \\ 0 & \text{sonst,} \end{cases} \qquad \mathrm{B}_E = \begin{cases} 4\lambda_{E_1}\lambda_{E_2} & \text{in } T \\ 4\lambda_{E_3}\lambda_{E_4} & \text{in } T_1 \\ 0 & \text{sonst.} \end{cases}$$

In Abbildung 6.2 sind diese Abschneidefunktionen dargestellt. Blasenfunktionen, die für die lokalen Fehlerabschätzungen in der L^2–Norm genutzt werden, müssen Element von $C^1(\bar{\Omega})$ sein. Zuerst nehmen wir

$$\mathrm{B}_T^* = \mathrm{B}_T^2.$$

Die Konstruktion der Kantenblasenfunktion B_E^* ist etwas komplizierter, vergleiche dazu Ab-

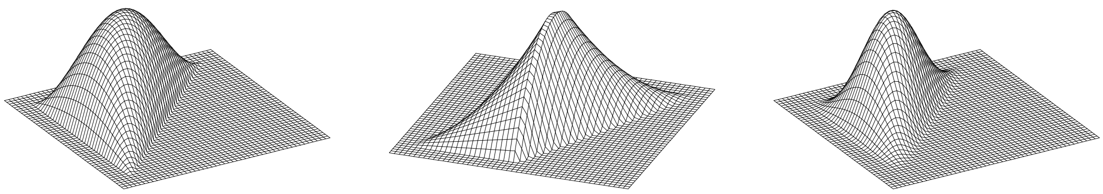


Abbildung 6.2: Die Abschneidefunktionen $\mathrm{B}_T, \mathrm{B}_E, \mathrm{B}_T^*$.

bildung 6.3. Sei E die gemeinsame Kante der Dreiecke T_1 und T_2. Die anderen Kanten von T_1 werden mit E_1 und E_2 bezeichnet. Das Dreieck T_1 wird an E gespiegelt, wobei E_1 und E_2 in E_1^* und E_2^* übergehen. Auf die gleiche Art und Weise erhält man E_3^* und E_4^* durch Spiegelung von T_2 an E. Wir definieren

$$\mathrm{B}_E^* = \begin{cases} 256\lambda_{E_1}^2\lambda_{E_2}^2\lambda_{E_1^*}^2\lambda_{E_2^*}^2 & \text{auf } T_1 \\ 256\lambda_{E_3}^2\lambda_{E_4}^2\lambda_{E_3^*}^2\lambda_{E_4^*}^2 & \text{auf } T_2 \\ 0 & \text{sonst.} \end{cases}$$

Lemma 6.3.1 *Die Abschneidefunktionen haben die folgenden Eigenschaften:*

i) $$\mathrm{B}_T, \mathrm{B}_E \in C^0(\bar{\Omega}),\ \mathrm{B}_T|_T \in P_3(T), \mathrm{B}_E|_T \in P_2(T)\ \forall T \subset \omega_E,$$

ii) $$\mathrm{B}_T^*, \mathrm{B}_E^* \in C^1(\bar{\Omega}),\ \mathrm{B}_T^*|_T \in P_6(T), \mathrm{B}_E^*|_T \in P_8(T)\ \forall T \subset \omega_E,$$

iii) $$0 \le \mathrm{B}_T, \mathrm{B}_E, \mathrm{B}_T^*, \mathrm{B}_E^* \le 1,$$

iv) $$\mathrm{B}_T = 0, \mathrm{B}_T^* = 0 \textit{ auf } \partial T, \quad \mathrm{B}_E = 0, \mathrm{B}_E^* = 0 \textit{ auf } \partial\omega_E, \tag{6.10}$$

v) $$\nabla \mathrm{B}_T^* = \mathbf{0} \textit{ auf } \partial T, \quad \nabla \mathrm{B}_E^* = \mathbf{0} \textit{ auf } \partial\omega_E, \tag{6.11}$$

vi) $$\nabla \mathrm{B}_E^* \cdot n_E = 0 \textit{ auf } E. \tag{6.12}$$

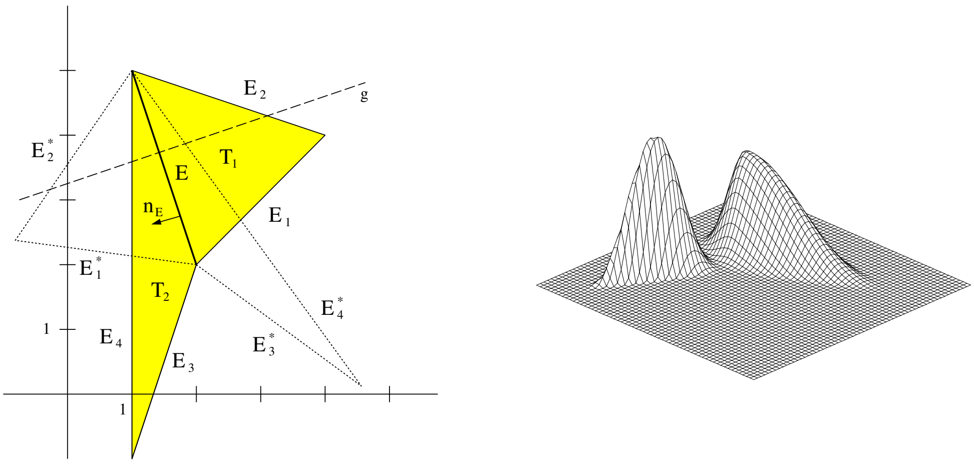


Abbildung 6.3: Die Abschneidefunktion B_E^*.

vii) B_E^* *ist stetig auf* E.
viii) $\mathrm{B}_E^*|_E$ *ist auf* E *eine gerade Funktion bezüglich des Kantenmittelpunktes.*

Beweis: Die Punkte i) - v) folgen direkt aus den Definitionen der Abschneidefunktionen.
vi) Wir betrachten eine Gerade g durch E in Normalenrichtung und das Dreieck T_1, siehe Abbildung 6.3. Die Definition von B_E^* auf T_1 wird auf das Spiegelbild von T_1 fortgesetzt. Dann ist $\mathrm{B}_E^*|_g$ auf dieser Geraden nach Definition stetig differenzierbar und spiegelsymmetrisch zu E. Deswegen hat $\mathrm{B}_E^*|_g$ ein lokales Extremum auf E, also $\nabla \mathrm{B}_E^* \cdot n_E = 0$ im Schnittpunkt der Geraden g und E. Dies gilt für alle Punkte auf E. Analog geht man mit T_2 vor, womit die Behauptung folgt.
vii) Wir betrachten zunächst T_1. Es gelten auf E $\lambda_{E_1} = \lambda_{E_1^*}$ und $\lambda_{E_2} = \lambda_{E_2^*}$. Deshalb folgt $\mathrm{B}_E^*|_E = 256\lambda_{E_1}^4\lambda_{E_2}^4$. Bezüglich T_2 gelten $\lambda_{E_3} = \lambda_{E_3^*} = \lambda_{E_1}$ und $\lambda_{E_4} = \lambda_{E_4^*} = \lambda_{E_2}$ woraus ebenfalls $\mathrm{B}_E^*|_E = 256\lambda_{E_1}^4\lambda_{E_2}^4$ folgt.
viii) Im Beweis zu Punkt vii) wurde $\mathrm{B}_E^* = 256\lambda_{E_1}^4\lambda_{E_2}^4$ auf E hergeleitet. Da $\lambda_{E_1}\lambda_{E_2}|_E$ spiegelsymmetrisch zum Kantenmittelpunkt ist, gilt dies auch für B_E^*. □

In den folgenden Abschnitten werden gleichmäßige Normäquivalenzen von gewichteten L^2–Normen in endlichdimensionalen Räumen benötigt.

Lemma 6.3.2 *Es gelten die folgenden Ungleichungen*

$$\|v\|_{0,T}^2 \leq C\int_T v^2 \mathrm{B}_T dx \quad \forall v \in P_k(T), \tag{6.13}$$

$$\|v\|_{0,T}^2 \leq C\int_T v^2 \mathrm{B}_T^* dx \quad \forall v \in P_k(T), \tag{6.14}$$

$$\|v\mathrm{B}_T\|_{0,T} \leq \|v\|_{0,T} \quad \forall v \in P_k(T), \tag{6.15}$$

$$\|v\mathrm{B}_T^*\|_{0,T} \leq \|v\|_{0,T} \quad \forall v \in P_k(T), \tag{6.16}$$

$$\|v\|_{0,E}^2 \leq C\int_E v^2 \mathrm{B}_E dx \quad \forall v \in P_k(E), \tag{6.17}$$

$$\|v\|_{0,E}^2 \leq C\int_E v^2 \mathrm{B}_E^* dx \quad \forall v \in P_k(E), \tag{6.18}$$

$$
\begin{aligned}
\|v\mathrm{B}_E\|_{0,\omega_E} &\leq \|v\|_{0,\omega_E} \quad \forall v \textit{ mit } v|_{T_i} \in P_k(T_i), T_i \subset \omega_E, &(6.19)\\
\|v\mathrm{B}^*_E\|_{0,\omega_E} &\leq \|v\|_{0,\omega_E} \quad \forall v \textit{ mit } v|_{T_i} \in P_k(T_i), T_i \subset \omega_E, &(6.20)
\end{aligned}
$$

wobei C eine Konstante unabhängig von T und h_T ist.

Beweis: (6.13) : Sei $\hat{T}$ das Referenzelement mit den Ecken $(0,0)$, $(1,0)$ und $(0,1)$. Man überprüft leicht, daß

$$
\hat{v} \to \left(\int_{\hat{T}} \hat{v}^2 \mathrm{B}_{\hat{T}} \right)^{1/2}
$$

eine Norm in $P_k(\hat{T})$ ist. Wegen der Äquivalenz von Normen in endlichdimensionalen Räumen existiert eine Konstante $C > 0$, so daß

$$
\|\hat{v}\|_{0,\hat{T}} \leq C \left(\int_{\hat{T}} \hat{v}^2 \mathrm{B}_{\hat{T}} \right)^{1/2}
$$

gilt. Die Ungleichung (6.13) folgt nun durch Transformation von T auf $\hat{T}$, Verwendung der Normäquivalenz und Rücktransformation. Mit derselben Argumentation beweist man auch (6.14), (6.17) und (6.18), wobei das mögliche Verschwinden von B^*_E auf einer Menge vom Maß Null im Inneren von ω_E, siehe Abbildung 6.3, hier keinen Einfluß hat.

Zum Beweis von (6.15), (6.16), (6.19) und (6.20) benötigt man nur die Beschränktheit der Blasenfunktionen. □

6.4 Lokale Abschätzung des Fehlers nach unten in der diskreten H^1–Norm

Sei $T \in \mathcal{T}_h$ ein beliebiges Element der regulären Triangulierung. Wir definieren

$$
\eta^2_{T,1} := h_T^2 \|f_h + \nu\Delta u_h - b_h \cdot \nabla u_h - c_h u_h\|^2_{0,T} + \sum_{E \subset \partial T} h_E \| [|\nu \nabla u_h \cdot n_E|]_E \|^2_{0,E}, \quad (6.21)
$$

$$
\begin{aligned}
\eta^2_{T,2} := {} & h_T^2 \|f_h + \nu\Delta u_h - b_h \cdot \nabla u_h - c_h u_h\|^2_{0,T} \\
& + \sum_{E \subset \partial T} \left(h_E \| [|\nu \nabla u_h \cdot n_E|]_E \|^2_{0,E} + h_E^{-1} \|\nu\, [|u_h|]_E \|^2_{0,E} \right). \quad (6.22)
\end{aligned}
$$

Hierbei sind f_h, b_h und c_h Approximationen an f, b und c, die so gewählt werden, daß die Einschränkung von $f_h + \nu\Delta u_h - b_h \cdot \nabla u_h - c_h u_h$ auf jedes Element T zu $P_k(T)$ für ein k gehört. Die Größen $\eta_{T,i}$ sind Linearkombinationen aus der Norm des Residuums der starken Formulierung (1.1) der Konvektions–Diffusions Gleichungen und den Sprüngen von Gradienten der Funktion u_h über die Kanten der Elemente. Der dritte Term in $\eta_{T,2}$ mißt die Nichtkonformität der Funktion u_h. Wir wollen zeigen, daß unter gewissen Bedingungen an die Approximationen der Koeffizientenfunktionen in (6.21), eine Ungleichung der Gestalt

$$
\eta_{T,i} \leq C(\nu + h_T)|u - u_h|_{h,\omega_T} + \mathcal{O}(h_T^2 |u - u_h|_{h,\omega_T}) \quad \forall T \in \mathcal{T}_h,\ i = 1,2
$$

gilt, das heißt, $\eta_{T,i}$ ist eine Abschätzung des Fehlers in $\nu|u - u_h|_{h,\omega_T}$ in einer Umgebung ω_T von T nach unten.

Im folgenden wird jeder Term von (6.21) und (6.22) einzeln abgeschätzt. Dazu werden Standardtechniken genutzt, siehe zum Beispiel Verfürth [Ver96b]. Diese basieren auf der Multiplikation der Terme von (6.21) und (6.22) mit geeigneten Blasenfunktionen und partieller Integration, wobei die auftretenden Randintegrale durch die Blasenfunktionen absorbiert werden.

Lemma 6.4.1 *Sei u die Lösung von (1.1) und $u_h \in V_{h0}$ eine beliebige Funktion. Dann gilt die Ungleichung*

$$\begin{aligned} & h_T\|f_h + \nu\Delta u_h - b_h\cdot\nabla u_h - c_h u_h\|_{0,T} \\ & \quad \le C\Big[\nu|u-u_h|_{1,T} + h_T\left(\|b\|_{\infty,T}|u-u_h|_{1,T} + \|c\|_{\infty,T}\|u-u_h\|_{0,T} + \|f-f_h\|_{0,T}\right) \\ & \qquad +(\|b-b_h\|_{\infty,T} + h_T\|c-c_h\|_{\infty,T})\|u_h\|_{0,T}\Big] \end{aligned} \tag{6.23}$$

für alle Elemente T einer regulären Triangulierung $\mathcal{T}_h$.

Beweis: Wir schreiben abkürzend $w_h = -\nu\Delta u_h + b_h\cdot\nabla u_h + c_h u_h$. Es ist

$$\begin{aligned} & \|f_h - w_h\|_{0,T}^2 \\ \stackrel{(6.13)}{\le} \;& C\int_T (f_h-w_h)^2 \mathrm{B}_T\,dx \\ = \;& C\left[\int_T (f-w_h)(f_h-w_h)\mathrm{B}_T\,dx + \int_T (f_h-f)(f_h-w_h)\mathrm{B}_T\,dx\right] \\ \stackrel{(1.1)}{=} \;& C\left[\int_T (-\nu\Delta(u-u_h) + b\cdot\nabla(u-u_h) + c(u-u_h))(f_h-w_h)\mathrm{B}_T\,dx\right. \\ & \left. + \int_T ((b-b_h)\cdot\nabla u_h + (c-c_h)u_h)(f_h-w_h)\mathrm{B}_T\,dx + \int_T (f_h-f)(f_h-w_h)\mathrm{B}_T\,dx\right] \\ \stackrel{p.I.,(6.10)}{=} \;& C\left[\int_T \nu\nabla(u-u_h)\cdot\nabla\left((f_h-w_h)\mathrm{B}_T\right)dx\right. \\ & + \int_T (b\cdot\nabla(u-u_h) + c(u-u_h))(f_h-w_h)\mathrm{B}_T\,dx \\ & \left. + \int_T ((b-b_h)\cdot\nabla u_h + (c-c_h)u_h)(f_h-w_h)\mathrm{B}_T\,dx + \int_T (f_h-f)(f_h-w_h)\mathrm{B}_T\,dx\right] \\ \le \;& C\,[\nu|u-u_h|_{1,T}|(f_h-w_h)\mathrm{B}_T|_{1,T} \\ & +(\|b\|_{\infty,T}|u-u_h|_{1,T} + \|c\|_{\infty,T}\|u-u_h\|_{0,T})\|(f_h-w_h)\mathrm{B}_T\|_{0,T} \\ & + \left(\|b-b_h\|_{\infty,T}|u_h|_{1,T} + \|c-c_h\|_{\infty,T}\|u_h\|_{0,T}\right)\|(f_h-w_h)\mathrm{B}_T\|_{0,T} \\ & +\|f_h-f\|_{0,T}\|(f_h-w_h)\mathrm{B}_T\|_{0,T}] \\ \stackrel{(6.7)}{\le} \;& C\left[\nu h_T^{-1}|u-u_h|_{1,T}\|(f_h-w_h)\mathrm{B}_T\|_{0,T}\right. \\ & +(\|b\|_{\infty,T}|u-u_h|_{1,T} + \|c\|_{\infty,T}\|u-u_h\|_{0,T})\|(f_h-w_h)\mathrm{B}_T\|_{0,T} \\ & + \left(h_T^{-1}\|b-b_h\|_{\infty,T} + \|c-c_h\|_{\infty,T}\right)\|u_h\|_{0,T}\|(f_h-w_h)\mathrm{B}_T\|_{0,T} \\ & +\|f_h-f\|_{0,T}\|(f_h-w_h)\mathrm{B}_T\|_{0,T}] \\ \stackrel{(6.15)}{\le} \;& C\left[\nu h_T^{-1}|u-u_h|_{1,T} + \|b\|_{\infty,T}|u-u_h|_{1,T} + \|c\|_{\infty,T}\|u-u_h\|_{0,T} + \|f_h-f\|_{0,T}\right. \\ & \left. + \left(h_T^{-1}\|b-b_h\|_{\infty,T} + \|c-c_h\|_{\infty,T}\right)\|u_h\|_{0,T}\right]\|f_h-w_h\|_{0,T}. \end{aligned}$$

Hieraus folgt die Behauptung. $\square$

Bevor die Abschätzung der Sprünge der Normalenableitung über die Kanten erfolgt, wollen wir darauf hinweisen, daß $[|\nu\nabla u_h \cdot n_E|]_E$ auf jeder Kante eine konstante Funktion ist. Mit $[|\nu\nabla u_h \cdot n_E|]_E^C$ wird die konstante Fortsetzung von $[|\nu\nabla u_h \cdot n_E|]_E$ auf ω_E bezeichnet. Für reguläre Triangulierungen gilt

$$\left\| [|\nu\nabla u_h \cdot n_E|]_E^C \right\|_{0,\omega_E} \leq C h_E^{1/2} \| [|\nu\nabla u_h \cdot n_E|]_E \|_{0,E}. \tag{6.24}$$

Lemma 6.4.2 *Unter den Voraussetzungen einer regulären Triangulierung $\mathcal{T}_h$ und der Regularität (6.1) für die exakte Lösung u von (1.1) gilt für jede beliebige Funktion $u_h \in V_{h0}$ und für alle Kanten E*

$$\begin{aligned} & h_E^{1/2} \| [|\nu\nabla u_h \cdot n_E|]_E \|_{0,E} \\ \leq\ & C\Big[\nu |u-u_h|_{h,\omega_E} + h_E \left(\|b\|_{\infty,\omega_E} |u-u_h|_{h,\omega_E} + \|c\|_{\infty,\omega_E} \|u-u_h\|_{0,\omega_E} + \|f-f_h\|_{0,\omega_E} \right) \\ & + \left(\|b-b_h\|_{\infty,\omega_E} + h_E \|c-c_h\|_{\infty,\omega_E} \right) \|u_h\|_{0,\omega_E} \Big]. \end{aligned} \tag{6.25}$$

Beweis: Da ∇u fast überall stetig ist, gilt

$$\begin{aligned} \| [|\nu\nabla u_h \cdot n_E|]_E \|_{0,E}^2 \ &\overset{(6.17)}{\leq}\ C \int_E \left([|\nu\nabla u_h \cdot n_E|]_E \right)^2 \mathrm{B}_E ds \\ &=\ C \int_E [|\nu\nabla(u_h - u) \cdot n_E|]_E \left([|\nu\nabla u_h \cdot n_E|]_E \, \mathrm{B}_E \right) ds \\ &\overset{p.I.,(6.10)}{=}\ C \sum_{T\subset\omega_E} \Bigg(\int_T \nu\nabla(u_h - u) \cdot \nabla\left([|\nu\nabla u_h \cdot n_E|]_E^C \, \mathrm{B}_E \right) dx \\ &\qquad + \int_T \nu\Delta(u_h - u) \, [|\nu\nabla u_h \cdot n_E|]_E^C \, \mathrm{B}_E dx \Bigg). \end{aligned}$$

Nun werden die beiden Summanden getrennt abgeschätzt. Zunächst gilt

$$\begin{aligned} & \sum_{T\subset\omega_E} \int_T \nu\nabla(u_h - u) \cdot \nabla\left([|\nu\nabla u_h \cdot n_E|]_E^C \, \mathrm{B}_E \right) dx \\ \leq\ & \nu |u - u_h|_{h,\omega_E} | [|\nu\nabla u_h \cdot n_E|]_E^C \, \mathrm{B}_E |_{1,\omega_E} \\ \overset{(6.7)}{\leq}\ & C h_E^{-1} \nu |u - u_h|_{h,\omega_E} \| [|\nu\nabla u_h \cdot n_E|]_E^C \, \mathrm{B}_E \|_{0,\omega_E} \\ \overset{(6.19)}{\leq}\ & C h_E^{-1} \nu |u - u_h|_{h,\omega_E} \| [|\nu\nabla u_h \cdot n_E|]_E^C \|_{0,\omega_E} \\ \overset{(6.24)}{\leq}\ & C h_E^{-1/2} \nu |u - u_h|_{h,\omega_E} \| [|\nu\nabla u_h \cdot n_E|]_E \|_{0,E}. \end{aligned}$$

Der zweite Summand wird in drei Anteile zerlegt:

$$\begin{aligned} & \sum_{T\subset\omega_E} \int_T \nu\Delta(u_h - u) \, [|\nu\nabla u_h \cdot n_E|]_E^C \, \mathrm{B}_E dx \\ \leq\ & \|\nu\Delta(u_h - u)\|_{0,\omega_E} \| [|\nu\nabla u_h \cdot n_E|]_E^C \, \mathrm{B}_E \|_{0,\omega_E} \\ \overset{(1.1)}{\leq}\ & \Big(\| -f_h + b\cdot\nabla u_h + c u_h + f - b\cdot\nabla u - c u \|_{0,\omega_E} + \| f_h + \nu\Delta u_h - b_h \cdot \nabla u_h - c_h u_h \|_{0,\omega_E} \\ & + \| (b - b_h)\cdot\nabla u_h - (c - c_h) u_h \|_{0,\omega_E} \Big) \| [|\nu\nabla u_h \cdot n_E|]_E^C \, \mathrm{B}_E \|_{0,\omega_E}. \end{aligned}$$

Den ersten Term kann man wie folgt abschätzen:

$$\begin{aligned}
&\|-f_h + b\cdot\nabla u_h + cu_h + f - b\cdot\nabla u - cu\|_{0,\omega_E} \| [\![\nu\nabla u_h\cdot n_E]\!]_E^C \mathrm{B}_E\|_{0,\omega_E}\\
&\quad\le \Big[\|f-f_h\|_{0,\omega_E} + |b|_{\infty,\omega_E}|u-u_h|_{h,\omega_E} + |c|_{\infty,\omega_E}\|u-u_h\|_{0,\omega_E}\Big] \| [\![\nu\nabla u_h\cdot n_E]\!]_E^C \mathrm{B}_E\|_{0,\omega_E}\\
&\quad\overset{(6.19)}{\le} \Big[\|f-f_h\|_{0,\omega_E} + |b|_{\infty,\omega_E}|u-u_h|_{h,\omega_E} + |c|_{\infty,\omega_E}\|u-u_h\|_{0,\omega_E}\Big] \| [\![\nu\nabla u_h\cdot n_E]\!]_E^C \|_{0,\omega_E}.
\end{aligned}$$

Zur Abschätzung des zweiten Terms benötigt man (6.23) für die Dreiecke, die ω_E bilden, sowie $h_E \le h_T \le Ch_E$:

$$\begin{aligned}
&\|f_h + \nu\Delta u_h - b_h\cdot\nabla u_h - c_h u_h\|_{0,\omega_E} \| [\![\nu\nabla u_h\cdot n_E]\!]_E^C \mathrm{B}_E\|_{0,\omega_E}\\
&\overset{(6.19)}{\le} Ch_E^{1/2}\Big[\nu h_E^{-1}|u-u_h|_{h,\omega_E} + (\|b\|_{\infty,\omega_E}|u-u_h|_{h,\omega_E} + \|c\|_{\infty,\omega_E}\|u-u_h\|_{0,\omega_E}\\
&\qquad + \|f_h - f\|_{0,\omega_E}) + (h_E^{-1}\|b-b_h\|_{\infty,\omega_E} + \|c-c_h\|_{\infty,\omega_E})\|u_h\|_{0,\omega_E}\Big] \| [\![\nu\nabla u_h\cdot n_E]\!]_E \|_{0,E}.
\end{aligned}$$

Der dritte Term wird wie in Lemma 6.4.1 abgeschätzt:

$$\begin{aligned}
&\|(b-b_h)\cdot\nabla u_h - (c-c_h)u_h\|_{0,\omega_E} \| [\![\nu\nabla u_h\cdot n_E]\!]_E^C \mathrm{B}_E\|_{0,\omega_E}\\
&\quad\le Ch_E^{1/2}\|u_h\|_{0,\omega_E}\left(h_E^{-1}\|b-b_h\|_{\infty,\omega_E} + \|c-c_h\|_{\infty,\omega_E}\right) \| [\![\nu\nabla u_h\cdot n_E]\!]_E \|_{0,E}.
\end{aligned}$$

Die Zusammenfassung der einzelnen Abschätzungen liefert die Behauptung des Lemmas. □

Der Sprung der Tangentialableitung $\left[\!\left[\frac{\partial u_h}{\partial t_E}\right]\!\right]_E$ von u_h ist auf jeder Kante eine konstante Funktion. Mit $\left[\!\left[\frac{\partial u_h}{\partial t_E}\right]\!\right]_E^C$ wird die Fortsetzung auf ω_E bezeichnet. Für reguläre Triangulierungen gilt

$$\left\| \left[\!\left[\frac{\partial u_h}{\partial t_E}\right]\!\right]_E^C \right\|_{0,\omega_E} \le Ch_E^{1/2} \left\| \left[\!\left[\frac{\partial u_h}{\partial t_E}\right]\!\right]_E \right\|_{0,E}. \tag{6.26}$$

Lemma 6.4.3 *Sei $\mathcal{T}_h$ eine reguläre Triangulierung. Unter den Voraussetzungen (6.1) an die Lösung gilt für alle $u_h \in V_{h0}$ und alle Kanten E*

$$h_E^{1/2} \left\| \left[\!\left[\nu\frac{\partial u_h}{\partial t_E}\right]\!\right]_E \right\|_{0,E} \le C\nu|u-u_h|_{h,\omega_E}. \tag{6.27}$$

Beweis: Sei $\omega_E = T_1 \cup T_2$. Mit den gleichen Schritten wie im Beweis von Lemma 6.4.2 gelangt man zu

$$\begin{aligned}
\left\| \left[\!\left[\nu\frac{\partial u_h}{\partial t_E}\right]\!\right]_E \right\|_{0,E}^2 &\le C\left\| \sum_{T\subset\omega_E}\int_T \nabla\left(\left[\!\left[\nu\frac{\partial u_h}{\partial t_E}\right]\!\right]_E^C \mathrm{B}_E\right)\cdot(\mathrm{curl}\,(\nu(u-u_h)))\,dx\right\|\\
&\le C\Big(\|\mathrm{curl}\,(\nu(u-u_h))\|_{0,T_1} + \|\mathrm{curl}\,(\nu(u-u_h))\|_{0,T_2}\Big) \left\| \left[\!\left[\nu\frac{\partial u_h}{\partial t_E}\right]\!\right]_E^C \mathrm{B}_E \right\|_{0,\omega_E}\\
&\le Ch_E^{-1}\nu|u-u_h|_{h,\omega_E} \left\| \left[\!\left[\nu\frac{\partial u_h}{\partial t_E}\right]\!\right]_E^C \right\|_{0,\omega_E}\\
&\le Ch_E^{-1/2}\nu|u-u_h|_{h,\omega_E} \left\| \left[\!\left[\nu\frac{\partial u_h}{\partial t_E}\right]\!\right]_E \right\|_{0,E}.
\end{aligned}$$

□

Bemerkung 6.4.4 Sprünge der nichtkonformen Funktion. Für das stückweise lineare nichtkonforme Finite Element gelten

$$\left\| \left[\left| \frac{\partial u_h}{\partial t_E} \right|\right]_E \right\|_{0,E}^2 = \frac{4}{h_E} \left(u_h(P_1) - u_h(P_2) + u_h(P_3) - u_h(P_4) \right)^2$$

und

$$\| [|u_h|]_E \|_{0,E}^2 = \frac{h_E}{3} \left(u_h(P_1) - u_h(P_2) + u_h(P_3) - u_h(P_4) \right)^2 ,$$

wobei die Bezeichnungen von Abbildung 6.4 verwendet wurden. Man kann also die Sprünge der Tangentialableitung über die Kanten durch die Sprünge der Funktion ersetzen. Mit den Voraussetzungen von Lemma 6.4.3 folgt

$$h_E^{-1/2} \| \nu [|u_h|]_E \|_{0,E} \le C\nu |u - u_h|_{h,\omega_E} \tag{6.28}$$

für jede Kante E.

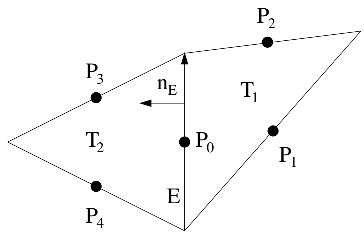


Abbildung 6.4: Abbildung zu Bemerkung 6.4.4.

Wir setzen voraus, daß $\|u_h\|_{0,T}$ für alle Elemente gleichmäßig beschränkt bezüglich h_T und ν ist. Desweiteren sind wir daran interessiert, Terme mit $\|f-f_h\|_{0,\omega_T}$, $\|b-b_h\|_{\infty,\omega_T}$ und $\|c-c_h\|_{\infty,\omega_T}$ als von höherer Ordnung zu betrachten und setzen die Approximationseigenschaften

$$\left.\begin{array}{ll} \|b-b_h\|_{\infty,\omega_T}\|u_h\|_{0,\omega_T} \le Ch_T^2|u-u_h|_{h,\omega_T}, & \|f-f_h\|_{0,\omega_T} \le Ch_T|u-u_h|_{h,\omega_T}, \\ \|c-c_h\|_{\infty,\omega_T}\|u_h\|_{0,\omega_T} \le Ch_T|u-u_h|_{h,\omega_T} & \end{array}\right\} \tag{6.29}$$

voraus.

Satz 6.4.5 Lokale a posteriori Fehlerabschätzung für den Fehler in der diskreten $\boldsymbol{H}^1$–Norm. *Unter der Regularitätsvoraussetzung (6.1) an die Lösung u, der gleichmäßigen Beschränktheit von $\|u_h\|_{0,T}$ bezüglich h_T und ν für alle Elemente, den Approximationseigenschaften (6.29) und einer regulären Triangulierung gilt die a posteriori Abschätzung*

$$\begin{aligned} \eta_{T,i} \;\le\; & C\Big[\nu|u-u_h|_{h,\omega_T} + h_T\left(\|b\|_{\infty,\omega_T}|u-u_h|_{h,\omega_T} + \|c\|_{\infty,\omega_T}\|u-u_h\|_{0,\omega_T}\right)\Big] \\ & + \mathcal{O}(h_T^2|u-u_h|_{h,\omega_T}), \; i = 1,2, \end{aligned} \tag{6.30}$$

für $u_h \in V_{h0}$ und jedes Element $T \in \mathcal{T}_h$.

Sind h_T hinreichend klein im Sinne von $h_T\|b\|_{\infty,\omega_T} < \nu$ und $\|u-u_h\|_{0,\omega_T} \le \mathcal{O}(h_T|u-u_h|_{h,\omega_T})$, dann gilt

$$\eta_{T,i} \le C\nu|u-u_h|_{h,\omega_T} + \mathcal{O}(h_T^2|u-u_h|_{h,\omega_T}), \; i = 1,2,$$

für $u_h \in V_{h0}$.

Beweis: Die Aussagen ergeben sich aus Lemma 6.4.1, Lemma 6.4.2, Lemma 6.4.3 und (6.28). □

Bemerkung 6.4.6 Die Rolle der Nichtkonformität der Diskretisierung. Die Abschätzung (6.30) gilt sowohl für konforme als auch für nichtkonforme P_1–Finite Elemente. Natürlich verschwinden im konformen Fall die Sprünge der diskreten Funktion über die Kanten.

Bemerkung 6.4.7 Die Voraussetzungen von Satz 6.4.5. Satz 6.4.5 gilt für jede Funktion $u_h \in V_{h0}$, für die $\|u_h\|_{0,T}$ gleichmäßig bezüglich h_T und ν beschränkt ist. Dies ist unabhängig davon, ob u_h ein diskretes Problem, welches (1.1) approximiert, löst oder nicht. Insbesondere kann die lokale Fehlerschätzung auf die Lösung u_h einer Methode angewandt werden, die Stabilisierung und numerische Integration beinhaltet. Die Approximationseigenschaften (6.29) sind lediglich bei der Berechnung der lokalen Fehlerschätzer (6.21) und (6.22) erforderlich.

Falls u_h mit einer Methode berechnet wird, die gleichmäßige L^2–Stabilität garantiert, das heißt $\|u_h\|_0 \le C\|f\|_0$ mit C unabhängig von ν und vom Gitter, dann ist die Voraussetzung der gleichmäßigen Beschränktheit von $\|u_h\|_T$ erfüllt. Die gleichmäßige L^2–Stabilität kann bewiesen werden, falls die Diskretisierung koerzitiv bezüglich einer diskreten Energienorm ist, zum Beispiel für die nichtkonforme Stromlinien–Diffusions Methoden (1.21) – (1.23), Satz 1.4.1, und für Upwind Methoden (1.18) mit Theorem 3 aus Ohmori und Ushijima [OU84].

Bemerkung 6.4.8 Die Verallgemeinerung der Ergebnisse. Die Ausdehnung der Ergebnisse auf Ansatzfunktionen höheren Grades erfordert eine mehr technische Fortsetzungen der Sprünge $[\![\nu\nabla u_h \cdot n_E]\!]_E$ und $\left[\!\!\left[\frac{\partial u_h}{\partial t_E}\right]\!\!\right]_E$ auf ω_E dergestalt, daß Ungleichungen der Form (6.24) und (6.26) gelten, siehe zum Beispiel Verfürth [Ver96b].

6.5 Lokale Abschätzung des Fehlers nach unten in der L^2–Norm

Die Voraussetzungen seien wie im Abschnitt 6.4. Für jedes Element T definieren wir die lokalen a posteriori Fehlerschätzer

$$\tilde{\eta}_{T,1}^2 := h_T^4\|f_h + \nu\Delta u_h - b_h\cdot\nabla u_h - c_h u_h\|_{0,T}^2 + \sum_{E\subset\partial T} h_E^3\|\,[\![\nu\nabla u_h\cdot n_E]\!]_E\,\|_{0,E}^2, \tag{6.31}$$

$$\begin{aligned}\tilde{\eta}_{T,2}^2 := {} & h_T^4\|f_h + \nu\Delta u_h - b_h\cdot\nabla u_h - c_h u_h\|_{0,T}^2 \\ & + \sum_{E\subset\partial T}\left(h_E^3\|\,[\![\nu\nabla u_h\cdot n_E]\!]_E\,\|_{0,E}^2 + h_E\|\nu\,[\![u_h]\!]_E\,\|_{0,E}^2\right).\end{aligned} \tag{6.32}$$

Die Funktionen b_h, c_h und f_h sind Approximationen an b, c und f, die sichern sollen, daß $f_h + \nu\Delta u_h - b_h\cdot\nabla u_h - c_h u_h \in P_k(T)$ für jedes Element T und ein k ist. Die Struktur von $\tilde{\eta}_{T,i}$ ist die gleiche wie von $\eta_{T,i}$, $i = 1,2$. Die Gewichte der Terme sind jedoch verschieden. Es wird gezeigt, daß unter geeigneten Approximationseigenschaften von b_h, c_h und f_h eine lokale a posteriori Fehlerabschätzung der Form

$$\tilde{\eta}_{T,2} \le C(\nu + h_T)\|u - u_h\|_{0,\omega_T} + \mathcal{O}(h_T^3\|u - u_h\|_{0,\omega_T}) \quad \forall T \in \mathcal{T}_h$$

gilt und für einen konstanten Konvektionsvektor b sogar

$$\tilde{\eta}_{T,1} \le C(\nu + h_T)\|u - u_h\|_{0,\omega_T} + \mathcal{O}(h_T^3\|u - u_h\|_{0,\omega_T}) \quad \forall T \in \mathcal{T}_h.$$

Die grundlegende Vorgehensweise bei den Beweisen in diesem Abschnitt ist die gleiche wie im Abschnitt 6.4. Wegen zusätzlicher partieller Integrationen entstehen neue Integrale auf den Rändern der lokalen Gebiete ω_T und ω_E sowie auf der Kante E in ω_E. Zur Behandlung dieser Randintegrale werden die speziellen Blasenfunktionen B_T^* und B_E^* benötigt.

Lemma 6.5.1 *Seien u die Lösung von (1.1) und $u_h \in V_{h0}$ eine beliebige Funktion. Dann gilt die Abschätzung*

$$\begin{aligned}
&h_T^2\|f_h + \nu\Delta u_h - b_h \cdot \nabla u_h - c_h u_h\|_{0,T} \\
&\quad\leq\; C\Big[\nu\|u-u_h\|_{0,T} + h_T\|b\|_{\infty,T}\|u-u_h\|_{0,T} + h_T^2\left(\|\nabla\cdot b\|_{\infty,T} + \|c\|_{\infty,T}\right)\|u-u_h\|_{0,T} \\
&\qquad + h_T^2\|f-f_h\|_{0,T} + h_T(\|b-b_h\|_{\infty,T} + h_T\|c-c_h\|_{\infty,T})\|u_h\|_{0,T}\Big] \qquad (6.33)
\end{aligned}$$

für alle Elemente T einer regulären Triangulierung $\mathcal{T}_h$.

Beweis: Der Beweis beginnt wie in Lemma 6.4.1 mit B_T ersetzt durch B_T^*. Sei wiederum $w_h = -\nu\Delta u_h + b_h\cdot\nabla u_h + c_h u_h$.

$$\begin{aligned}
&\|f_h - w_h\|_{0,T}^2 \\
&\overset{(6.14)}{\leq}\; C\int_T (f_h-w_h)^2\mathrm{B}_T^* dx \\
&\overset{p.I.,(6.10)}{=}\; C\Bigg[\int_T \nu\nabla(u-u_h)\cdot\nabla\left((f_h-w_h)\mathrm{B}_T^*\right)dx \\
&\qquad + \int_T (b\cdot\nabla(u-u_h) + c(u-u_h))(f_h-w_h)\mathrm{B}_T^* dx \\
&\qquad + \int_T ((b-b_h)\cdot\nabla u_h + (c-c_h)u_h)(f_h-w_h)\mathrm{B}_T^* dx + \int_T (f_h-f)(f_h-w_h)\mathrm{B}_T^* dx\Bigg] \\
&\overset{p.I.}{=}\; C\Bigg[-\int_T \nu(u-u_h)\Delta\left((f_h-w_h)\mathrm{B}_T^*\right)dx - \int_T ((u-u_h)\nabla\cdot\left((f_h-w_h)\mathrm{B}_T^* b\right)dx \\
&\qquad + \int_T c(u-u_h)(f_h-w_h)\mathrm{B}_T^* dx + \int_T ((b-b_h)\cdot\nabla u_h + (c-c_h)u_h)(f_h-w_h)\mathrm{B}_T^* dx \\
&\qquad + \int_T (f_h-f)(f_h-w_h)\mathrm{B}_T^* dx\Bigg] \\
&\leq\; C\Big[\nu\|u-u_h\|_{0,T}\|\Delta\left((f_h-w_h)\mathrm{B}_T^*\right)\|_{0,T} + \|u-u_h\|_{0,T}\Big(\|\nabla\cdot b\|_{\infty,T}\|(f_h-w_h)\mathrm{B}_T^*\|_{0,T} \\
&\qquad + \|b\|_{\infty,T}\|\nabla(f_h-w_h)\mathrm{B}_T^*\|_{0,T}\Big) + \|c\|_{\infty,T}\|u-u_h\|_{0,T}\|(f_h-w_h)\mathrm{B}_T^*\|_{0,T} \\
&\qquad + \left(\|b-b_h\|_{\infty,T}|u_h|_{1,T} + \|c-c_h\|_{\infty,T}\|u_h\|_{0,T}\right)\|(f_h-w_h)\mathrm{B}_T^*\|_{0,T} \\
&\qquad + \|f-f_h\|_{0,T}\|(f_h-w_h)\mathrm{B}_T^*\|_{0,T}\Big] \\
&\leq\; C\Big[\nu\|u-u_h\|_{0,T}|\left((f_h-w_h)\mathrm{B}_T^*\right)|_{2,T} + \|u-u_h\|_{0,T}\Big(\|\nabla\cdot b\|_{\infty,T}\|(f_h-w_h)\mathrm{B}_T^*\|_{0,T} \\
&\qquad + \|b\|_{\infty,T}|(f_h-w_h)\mathrm{B}_T^*|_{1,T}\Big) + \|c\|_{\infty,T}\|u-u_h\|_{0,T}\|(f_h-w_h)\mathrm{B}_T^*\|_{0,T} \\
&\qquad + \|f-f_h\|_{0,T}\|(f_h-w_h)\mathrm{B}_T^*\|_{0,T} \\
&\qquad + \left(h_T^{-1}\|b-b_h\|_{\infty,T} + \|c-c_h\|_{\infty,T}\right)\|u_h\|_{0,T}\|(f_h-w_h)\mathrm{B}_T^*\|_{0,T}\Big] \\
&\overset{(6.7),(6.16)}{\leq}\; C\left[\nu h_T^{-2}\|u-u_h\|_{0,T} + h_T^{-1}\|b\|_{\infty,T}\|u-u_h\|_{0,T} + \left(\|\nabla\cdot b\|_{\infty,T} + \|c\|_{\infty,T}\right)\|u-u_h\|_{0,T}\right. \\
&\qquad \left. + \|f-f_h\|_{0,T} + \left(h_T^{-1}\|b-b_h\|_{\infty,T} + \|c-c_h\|_{\infty,T}\right)\|u_h\|_{0,T}\right]\|f_h-w_h\|_{0,T}.
\end{aligned}$$

Die Randintegrale bei der zweiten partiellen Integration verschwinden wegen der Eigenschaften (6.10) und (6.11) von B_T^*. Damit folgt die Behauptung des Lemmas. □

Lemma 6.5.2 *Seien eine reguläre Triangulierung sowie eine exakte Lösung u mit der Regularität (6.1) gegeben. Dann gilt für jede Funktion $u_h \in V_{h0}$ und jede Kante E*

$$h_E^{1/2} \|\nu \, [\![u_h]\!]_E \, \|_{0,E} \le C \left(\nu \|u - u_h\|_{0,\omega_E} + \nu h_E^2 |u|_{2,\omega_E} \right). \tag{6.34}$$

Beweis: Seien $R_{X_h} u \in X_h$ die Interpolierte von u und $\omega_E = T_1 \cup T_2$. Mit (6.9) und (6.3) erhält man

$$\begin{aligned}
\|\nu \, [\![u_h]\!]_E \, \|_{0,E}^2 &= \int_E \left[\nu \, (u_h - R_{X_h} u) \, |_{T_1} - \nu \, (u_h - R_{X_h} u) \, |_{T_2} \right]^2 ds \\
&\le 2 \left\{ \int_E \left[\nu \, (u_h - R_{X_h} u) \, |_{T_1} \right]^2 ds + \int_E \left[\nu \, (u_h - R_{X_h} u) \, |_{T_2} \right]^2 ds \right\} \\
&\le C h_E^{-1} \|\nu (u_h - R_{X_h} u)\|_{0,\omega_E}^2 \\
&\le C \left(\nu^2 h_E^{-1} \|u - u_h\|_{0,\omega_E}^2 + \nu^2 h_E^3 |u|_{2,\omega_E}^2 \right).
\end{aligned}$$

□

Lemma 6.5.3 *Seien eine reguläre Triangulierung sowie eine exakte Lösung u mit der Regularität (6.1) gegeben. Dann gilt für jede Funktion $u_h \in V_{h0}$ und jede Kante E*

$$\begin{aligned}
& h_E^{3/2} \| \, [\![\nu \nabla u_h \cdot n_E]\!]_E \, \|_{0,E} \\
&\le C \Big[\nu \|u - u_h\|_{0,\omega_E} + h_E \|b\|_{\infty,\omega_E} \|u - u_h\|_{0,\omega_E} + h_E^4 |b|_{1,\infty,\omega_E} |u|_{2,\omega_E} \\
&\quad + h_E^2 \left(\|\nabla \cdot b\|_{\infty,\omega_E} + |b|_{1,\infty,\omega_E} + \|c\|_{\infty,\omega_E} \right) \|u - u_h\|_{0,\omega_E} + h_E^2 \|f - f_h\|_{0,\omega_E} \\
&\quad + h_E (\|b - b_h\|_{\infty,\omega_E} + h_E \|c - c_h\|_{\infty,\omega_E}) \|u_h\|_{0,\omega_E} \Big].
\end{aligned} \tag{6.35}$$

Beweis: Die Beweisführung ist wie in Lemma 6.4.2 mit B_E ersetzt durch B_E^*:

$$\begin{aligned}
& \| \, [\![\nu \nabla u_h \cdot n_E]\!]_E \, \|_{0,E}^2 \\
&\le \quad \cdots \\
&\overset{p.I.,(6.10)}{=} C \left\| \sum_{T \subset \omega_E} \left(\int_T \nu \nabla (u_h - u) \cdot \nabla ([\![\nu \nabla u_h \cdot n_E]\!]_E^C \, \mathrm{B}_E^*) dx \right. \right. \\
&\qquad \left. \left. + \int_T \nu \Delta (u_h - u) \, [\![\nu \nabla u_h \cdot n_E]\!]_E^C \, \mathrm{B}_E^* dx \right) \right\|.
\end{aligned}$$

Diese beiden Terme werden getrennt abgeschätzt:

$$\begin{aligned}
& \sum_{T \subset \omega_E} \int_T \nu \nabla (u_h - u) \cdot \nabla ([\![\nu \nabla u_h \cdot n_E]\!]_E^C \, \mathrm{B}_E^*) dx \\
&\overset{p.I.}{=} \sum_{T \subset \omega_E} - \int_T \nu (u_h - u) \, [\![\nu \nabla u_h \cdot n_E]\!]_E^C \, \Delta \mathrm{B}_E^* dx \\
&\qquad + \int_{\partial \omega_E} \nu (u_h - u) \, [\![\nu \nabla u_h \cdot n_E]\!]_E^C \, (\nabla \mathrm{B}_E^* \cdot n_{\omega_E}) ds \\
&\qquad + \int_E \nu \, [\![u_h]\!]_E \, [\![\nu \nabla u_h \cdot n_E]\!]_E \, (\nabla \mathrm{B}_E^* \cdot n_E) ds \\
&\le \nu \|u - u_h\|_{0,\omega_E} \| \, [\![\nu \nabla u_h \cdot n_E]\!]_E^C \, \Delta \mathrm{B}_E^* \|_{0,\omega_E} \\
&\overset{(6.7),(6.20)}{\le} C h_E^{-3/2} \nu \|u - u_h\|_{0,\omega_E} \| \, [\![\nu \nabla u_h \cdot n_E]\!]_E \, \|_{0,E}.
\end{aligned}$$

Bei dieser partiellen Integration verschwindet das Randintegral auf $\partial\omega_E$, da dort $\nabla \mathrm{B}_E^* = \mathbf{0}$ ist. Auf Grund der Nichtkonformität der Diskretisierung verschwindet das Randintegral auf der inneren Kante E nur, weil (6.12) gilt. Für den zweiten Summanden wird (1.1) genutzt und der konvektive Teil partiell integriert:

$$
\begin{aligned}
&\sum_{T\subset\omega_E}\int_T \nu\Delta(u_h-u)\,[|\nu\nabla u_h\cdot n_E|]_E^C\,\mathrm{B}_E^* dx\\
&= \sum_{T\subset\omega_E}\Bigg[\int_T ((f-f_h)-c(u-u_h))\,[|\nu\nabla u_h\cdot n_E|]_E^C\,\mathrm{B}_E^* dx\\
&\quad+\int_T (u-u_h)\,[|\nu\nabla u_h\cdot n_E|]_E^C\,\nabla\cdot(\mathrm{B}_E^* b)\,dx\\
&\quad+\int_T (f_h+\nu\Delta u_h-b_h\cdot\nabla u_h-c_h u_h)\,[|\nu\nabla u_h\cdot n_E|]_E^C\,\mathrm{B}_E^* dx\\
&\quad-\int_T ((b-b_h)\cdot\nabla u_h+(c-c_h)u_h)\,[|\nu\nabla u_h\cdot n_E|]_E^C\,\mathrm{B}_E^* dx\Bigg]\\
&\quad-\int_{\partial\omega_E}(u-u_h)\,[|\nu\nabla u_h\cdot n_E|]_E^C\,(\mathrm{B}_E^* b\cdot n_{\omega_E})ds+\int_E [|u_h|]_E\,[|\nu\nabla u_h\cdot n_E|]_E\,\mathrm{B}_E^*(b\cdot n_E)ds.
\end{aligned}
$$

Das Randintegral auf $\partial\omega_E$ verschwindet, da B_E^* auf $\partial\omega_E$ verschwindet. Nun betrachten wir das Integral auf der inneren Kante E. Sei B_E der Mittelpunkt von E und b_0 die konstante Funktion mit dem Wert $b_0 = b(B_E)$. Es folgt

$$
\begin{aligned}
&\int_E [|u_h|]_E\,[|\nu\nabla u_h\cdot n_E|]_E\,\mathrm{B}_E^*(b\cdot n_E)ds\\
&= \int_E [|u_h|]_E\,[|\nu\nabla u_h\cdot n_E|]_E\,\mathrm{B}_E^*(b_0\cdot n_E)ds+\int_E [|u_h|]_E\,[|\nu\nabla u_h\cdot n_E|]_E\,\mathrm{B}_E^*((b-b_0)\cdot n_E)ds.
\end{aligned}
$$

Das erste Integral verschwindet, da $[|u_h|]_E$ eine ungerade und B_E^* eine gerade Funktion bezüglich des Mittelpunktes von E ist:

$$
\int_E [|u_h|]_E\,[|\nu\nabla u_h\cdot n_E|]_E\,\mathrm{B}_E^*(b_0\cdot n_E)ds = [|\nu\nabla u_h\cdot n_E|]_E\,(b_0\cdot n_E)\int_E [|u_h|]_E\,\mathrm{B}_E^* ds = 0. \qquad (6.36)
$$

Auf Grund der gemachten Voraussetzungen gilt $b\in\left(W^{1,\infty}(E)\right)^2$. Mit Taylor–Entwicklung erhält man

$$
\|(b-b_0)\cdot n_E\|_{\infty,E}\le Ch_E|b\cdot n_E|_{1,\infty,E}.
$$

Nutzt man (6.34) mit $\nu=1$ folgt

$$
\begin{aligned}
&\int_E [|u_h|]_E\,[|\nu\nabla u_h\cdot n_E|]_E\,\mathrm{B}_E^*(b\cdot n_E)ds\\
&\le\; Ch_E|b\cdot n_E|_{1,\infty,E}\|\,[|u_h|]_E\,\|_{0,E}\|\,[|\nu\nabla u_h\cdot n_E|]_E\,\|_{0,E}\\
&\le\; Ch_E^{1/2}|b|_{1,\infty,\omega_E}\left(\|u-u_h\|_{0,\omega_E}+h_E^2|u|_{2,\omega_E}\right)\|\,[|\nu\nabla u_h\cdot n_E|]_E\,\|_{0,E}.
\end{aligned}
$$

Die übrigen Integrale kann man mit Hilfe der Cauchy–Schwarz Ungleichung, (6.24) und (6.33) abschätzen. Man erhält

$$
\sum_{T\subset\omega_E}\int_T ((f-f_h)-c(u-u_h))\,[|\nu\nabla u_h\cdot n_E|]_E^C\,\mathrm{B}_E^* dx
$$

$$\leq \; Ch_E^{1/2}\left(\|f-f_h\|_{0,\omega_E}+\|c\|_{\infty,\omega_E}\|u-u_h\|_{0,\omega_E}\right)\|\,[\![\nu\nabla u_h\cdot n_E]\!]_E\,\|_{0,E},$$

$$\sum_{T\subset\omega_E}\int_T (u-u_h)\,[\![\nu\nabla u_h\cdot n_E]\!]_E^C\,\nabla\cdot(\mathrm{B}_E^* b)\,dx$$
$$\leq \; C\|u-u_h\|_{0,\omega_E}\left(h_E^{-1/2}\|b\|_{\infty,\omega_E}+h_E^{1/2}\|\nabla\cdot b\|_{\infty,\omega_E}\right)\|\,[\![\nu\nabla u_h\cdot n_E]\!]_E\,\|_{0,E},$$

$$\sum_{T\subset\omega_E}\int_T \left((b-b_h)\cdot\nabla u_h+(c-c_h)u_h\right)[\![\nu\nabla u_h\cdot n_E]\!]_E^C\,\mathrm{B}_E^*\,dx$$
$$\leq \; Ch_E^{1/2}\|u_h\|_{0,\omega_E}\left(h_E^{-1}\|b-b_h\|_{\infty,\omega_E}+\|c-c_h\|_{\infty,\omega_E}\right)\|\,[\![\nu\nabla u_h\cdot n_E]\!]_E\,\|_{0,E},$$

$$\sum_{T\subset\omega_E}\int_T (f_h+\nu\Delta u_h-b_h\cdot\nabla u_h-c_h u_h)\,[\![\nu\nabla u_h\cdot n_E]\!]_E^C\,\nabla\cdot(\mathrm{B}_E^* b)\,dx$$
$$\leq \; Ch_E^{1/2}\|f_h+\nu\Delta u_h-b_h\cdot\nabla u_h-c_h u_h\|_{0,\omega_E}\|\,[\![\nu\nabla u_h\cdot n_E]\!]_E\,\|_{0,E}.$$

Aus den einzelnen Abschätzungen folgt die Behauptung des Lemmas. □

Wir setzen auch in diesem Abschnitt voraus, daß $\|u_h\|_{0,T}$ für alle Elemente gleichmäßig beschränkt bezüglich h_T und ν ist. Um die Terme mit $\|f-f_h\|_{0,\omega_T}$, $\|b-b_h\|_{\infty,\omega_T}$ und $\|c-c_h\|_{\infty,\omega_T}$ als Terme höherer Ordnung betrachten zu können, fordern wir die Approximationseigenschaften

$$\left.\begin{array}{l}\|b-b_h\|_{\infty,\omega_T}\|u_h\|_{0,\omega_T}\leq Ch_T^2\|u-u_h\|_{0,\omega_T}, \quad \|f-f_h\|_{0,\omega_T}\leq Ch_T\|u-u_h\|_{0,\omega_T},\\ \|c-c_h\|_{\infty,\omega_T}\|u_h\|_{0,\omega_T}\leq Ch_T\|u-u_h\|_{0,\omega_T}.\end{array}\right\} \tag{6.37}$$

Satz 6.5.4 Lokale a posteriori Fehlerabschätzung für den Fehler in der L^2–Norm. *Unter der Regularitätsvoraussetzung (6.1) an die Lösung u, der gleichmäßigen Beschränktheit von $\|u_h\|_{0,T}$ bezüglich h_T und ν für alle Elemente, den Approximationseigenschaften (6.37) und einer regulären Triangulierung gilt die a posteriori Abschätzung*

$$\begin{aligned}\tilde\eta_{T,2} \;\leq\; & C\left[\left(\nu+h_T\|b\|_{\infty,\omega_T}+h_T^2|b|_{1,\infty,\omega_T}+h_T^2\|c\|_{\infty,\omega_T}\right)\|u-u_h\|_{0,\omega_T}\right.\\ & \left.+h_T^2\left(\nu+h_T^2|b|_{1,\infty,\omega_T}\right)|u|_{2,\omega_T}\right]+\mathcal{O}(h_T^3\|u-u_h\|_{0,\omega_T})\end{aligned} \tag{6.38}$$

für $u_h\in V_{h0}$ und jedes Element $T\in\mathcal{T}_h$. Existiert eine Konstante C mit

$$h_T^2|u|_{2,\omega_T}\leq C\|u-u_h\|_{0,\omega_T},$$

wobei C nicht von h_T und ν abhängt, dann gilt

$$\tilde\eta_{T,2}\leq C\left(\nu+h_T\|b\|_{\infty,\omega_T}+h_T^2|b|_{1,\infty,\omega_T}+h_T^2\|c\|_{\infty,\omega_T}\right)\|u-u_h\|_{0,\omega_T}+\mathcal{O}(h_T^3\|u-u_h\|_{0,\omega_T})$$

für $u_h\in V_{h0}$. Ist zusätzlich h_T klein genug im Sinne von $h_T\|b\|_{\infty,\omega_T}<\nu$, $h_T^2|b|_{1,\infty,\omega_T}<\nu$ und $h_T^2\|c\|_{\infty,\omega_T}<\nu$, dann gilt die Abschätzung

$$\tilde\eta_{T,2}\leq C\nu\|u-u_h\|_{0,\omega_T}+\mathcal{O}(h_T^3\|u-u_h\|_{0,\omega_T})$$

für $u_h\in V_{h0}$.

Ist b eine konstante Funktion mit $b=b_h$, so gilt

$$\tilde\eta_{T,1}\leq C\left[\left(\nu+h_T\|b\|_{\infty,\omega_T}+h_T^2\|c\|_{\infty,\omega_T}\right)\|u-u_h\|_{0,\omega_T}\right]+\mathcal{O}(h_T^3\|u-u_h\|_{0,\omega_T})$$

für $u_h \in V_{h0}$ und jedes Element $T \in \mathcal{T}_h$. Ist h_T klein genug im Sinne von $h_T\|b\|_{\infty,\omega_T} < \nu$ und $h_T^2\|c\|_{\infty,\omega_T} < \nu$, dann gilt für $u_h \in V_{h0}$

$$\tilde{\eta}_{T,1} \leq C\nu\|u - u_h\|_{0,\omega_T} + \mathcal{O}(h_T^3\|u - u_h\|_{0,\omega_T}).$$

Beweis: Die Abschätzung (6.38) folgt aus direkt den Lemmata in diesem Abschnitt.

Sei b eine konstante Funktion mit $b = b_h$. In diesem Fall kann Lemma 6.5.3 ohne die Verwendung von (6.34) bewiesen werden, da (6.36) mit $b_0 = b$ gilt. Es tauchen keine Sprünge von u_h auf, die abgeschätzt werden müssen. Damit erhält man schon mit $\eta_{T,1}$ eine lokale a posteriori Fehlerabschätzung. □

Bemerkung 6.5.5 Die Rolle der Nichtkonformität der Diskretisierung. Im Beweis von Lemma 6.5.3 war die Nichtkonformität der Diskretisierung der kritische Punkt. Das wesentliche Instrument zur Überwindung der Schwierigkeiten war die spezielle Wahl der Kantenblasenfunktion B_E^*.

Bemerkung 6.5.6 Die lokale $\boldsymbol{H^2}$–Seminorm von $\boldsymbol{u}$. Eine Abschätzung der Gestalt $|u|_{2,\omega_T} \leq Ch_T$ gilt gleichmäßig in ν in Gebieten, in denen die Lösung glatt ist, daß heißt in denen $|u|_{2,\omega_T}$ nicht von ν abhängt. Die Situation in Randgrenzschichten ist jedoch anders. Wir betrachten ein Beispiel, das von Roos [Roo96] untersucht wurde. Es werden hinreichend glatte Funktionen in (1.1), $\Omega = (0,1)^2$ und Kompatibilitätsbedingungen für die rechte Seite an den Ecken von $\bar{\Omega}$ vorausgesetzt. Dann kann u in der Form $u = v + w$ dargestellt werden, wobei $|v|_2$ gleichmäßig beschränkt für ν ist. Roos zeigt punktweise Abschätzungen des Betrages der Ableitungen von w in den Randgrenzschichten. Nutzt man diese Abschätzungen und betrachtet ein gleichschenkliges, rechtwinkliges Dreieck mit Durchmesser h_T so findet man $|u|_{2,T} = \mathcal{O}(h_T^{1/2}\nu^{-3/2})$ für $h_T \geq \nu$. Damit ist der Term $\nu|u|_{2,\omega_T}$ in (6.38) erst gleichmäßig für ν beschränkt, wenn $h_T = \mathcal{O}(\nu)$ gilt.

Die Aussage der Bemerkungen 6.4.7 und 6.4.8 gelten auch für Satz 6.5.4. Die Sprünge der nichtkonformen Funktion über die Kanten kann man durch Sprünge der Tangentialableitung über die Kanten ersetzen, siehe Bemerkung 6.4.4.

6.6 Die Abschätzung des globalen Fehlers in der L^2–Norm

Die von uns verwendete Beweistechnik geht auf Johnson zurück [EEHJ95] und umfaßt vier Schritte:

1. Fehlerdarstellung mit einem linearisierten stetigen dualen Problem,
2. Nutzung der Galerkin Orthogonalität des diskreten Problems,
3. Interpolationsabschätzung für die Lösung des dualen Problems,
4. starke Stabilität des dualen Problems.

Wir betrachten Konvektions–Diffusions Gleichungen (1.1) mit homogenen Dirichletbedingungen. Das zugehörige duale Problem lautet

Finde $z_w \in H_0^1(\Omega)$ mit

$$(\nu\nabla z_w, \nabla v) - (b \cdot \nabla z_w, v) + ((c - \nabla \cdot b)z_w, v) \quad = \quad (w, v) \quad \forall v \in V_0. \tag{6.39}$$

Für die Lösung des dualen Problems werden H^2–Regularität

$$z_w \in H^2(\Omega) \cap H_0^1(\Omega) \tag{6.40}$$

und starke Stabilität vorausgesetzt, das heißt, es existiert eine Konstante $C_S = C_S(\nu)$, so daß

$$\|z_w\|_2 \leq C_S \|w\|_0 \tag{6.41}$$

gilt.

Ein technisches Werkzeug ist die Funktion $\hat{f}_h$. Sie wird als rechte Seite der Standard–Galerkin Formulierung von (1.3) im Raum X_h für eine gegebene Funktion $u_h \in V_{h0}$ definiert:

$$\sum_{T\in\mathcal{T}_h} (\nu\nabla u_h, \nabla v_h)_T + (b\cdot\nabla u_h, v_h)_T + (cu_h, v_h)_T = (\hat{f}_h, v_h) \quad \forall v_h \in X_h. \tag{6.42}$$

Kann man alle Integrale der linken Seite exakt berechnen, so ist die Funktion $\hat{f}_h$ berechenbar.

Satz 6.6.1 Globale a posteriori Fehlerabschätzung für den Fehler in der L^2–Norm. *Unter der Regularitätsvoraussetzung (6.1) an die exakte Lösung u, der Regularität (6.39) und Stabilität (6.41) der Lösung des dualen Problems sowie der Regularität der Triangulierung gilt die globale a posteriori Fehlerabschätzung*

$$\begin{aligned} \|u-u_h\|_0 &\leq CC_IC_S\eta + CC_IC_S\left(\sum_{T\in\mathcal{T}_h} h_T^4\|f-f_h\|_{0,T}^2\right)^{1/2} + CC_S\|f-\hat{f}_h\|_0 \qquad (6.43)\\ &= CC_IC_S\left(\sum_{T\in\mathcal{T}_h}\eta_T^2\right)^{1/2} + CC_IC_S\left(\sum_{T\in\mathcal{T}_h} h_T^4\|f-f_h\|_{0,T}^2\right)^{1/2} + CC_S\|f-\hat{f}_h\|_0 \end{aligned}$$

mit

$$\eta_T^2 = h_T^4\|f_h+\nu\Delta u_h - b\cdot\nabla u_h - cu_h\|_{0,T}^2 + \sum_{E\subset\partial T} h_E^3\| [\![\nu\nabla u_h\cdot n_E]\!]_E \|_{0,E}^2 + h_E\| [\![u_h]\!]_E \|_{0,E}^2 \tag{6.44}$$

für alle $u_h \in V_{h0}$.

Beweis: Wir gehen vom stetigen dualen Problem (6.39) aus, setzen $w = u - u_h \in L^2(\Omega)$ und bezeichnen die Lösung mit z. Da im allgemeinen $u - u_h \notin V_0$ ist, können wir $u - u_h$ nicht ohne weiteres als Testfunktion nutzen. Für nichtkonforme Testfunktionen ergibt sich beim Übergang von der starken zur schwachen Formulierung noch ein Zusatzterm. Testet man die starke Formulierung des dualen Problems mit $u - u_h$ und integriert partiell, so erhält man

$$\begin{aligned} \|u-u_h\|_0^2 &= \sum_{T\in\mathcal{T}_h} (\nu\nabla z, \nabla(u-u_h))_T - (b\cdot\nabla z, u-u_h)_T + ((c-\nabla\cdot b)z, u-u_h)_T \\ &\quad - \sum_{E\subset\partial T}\int_E \nu\nabla z\cdot n_E(u-u_h)ds. \qquad (6.45) \end{aligned}$$

Da $X_h = V_0 \cap V_{h0}$ ist, folgt aus (1.3)

$$\sum_{T\in\mathcal{T}_h} a(u,v_h)_T + (b\cdot\nabla u, v_h)_T + (cu, v_h)_T = (f, v_h) \quad \forall v_h \in X_h.$$

Aus dieser Beziehung und aus (6.42) erhält man die approximierte Galerkin Orthogonalität

$$0 = \sum_{T\in\mathcal{T}_h} (\nu\nabla(u-u_h), \nabla v_h)_T + (b\cdot\nabla(u-u_h), v_h)_T + (c(u-u_h), v_h)_T - (f-\hat{f}_h, v_h)_T \quad \forall v_h \in X_h. \tag{6.46}$$

Wir wählen $v_h = R_{X_h} z = z_h$ und subtrahieren dies von (6.45). Wegen der Stetigkeit von z und der Stetigkeit fast überall von ∇u folgt mit partieller Integration:

$$
\begin{aligned}
&\|u-u_h\|_0^2 \\
&= \sum_{T\in\mathcal{T}_h} \Big[(\nu\nabla(z-z_h), \nabla(u-u_h))_T - (b\cdot\nabla z, u-u_h)_T - (b\cdot\nabla(u-u_h), z_h)_T \\
&\qquad -((\nabla\cdot b)z, u-u_h)_T + (c(u-u_h), z-z_h)_T + (f-\hat{f}_h, z_h)_T + \sum_{E\subset\partial T}\int_E \nu\nabla z\cdot n_E u_h ds \Big] \\
&= \sum_{T\in\mathcal{T}_h} \Big[(-\nu\Delta(u-u_h) + b\cdot\nabla(u-u_h) + c(u-u_h), z-z_h)_T \\
&\qquad + \sum_{E\subset\partial T}\int_E \nu\nabla(u-u_h)\cdot n_E(z-z_h)ds + \int_E \nu\nabla z\cdot n_E u_h ds - \int_E (b\cdot n_E)(u-u_h)z ds \Big] \\
&\qquad + (f-\hat{f}_h, z_h).
\end{aligned}
$$

Die Abschätzung der beiden ersten Terme ist Standard:

$$
\begin{aligned}
&\sum_{T\in\mathcal{T}_h} (-\nu\Delta(u-u_h) + b\cdot\nabla(u-u_h) + c(u-u_h), z-z_h)_T \\
&= \sum_{T\in\mathcal{T}_h} (f_h + \nu\Delta u_h - b\cdot\nabla u_h - cu_h, z-z_h)_T + (f-f_h, z-z_h)_T \\
&\le \sum_{T\in\mathcal{T}_h} \|f_h + \nu\Delta u_h - b\cdot\nabla u_h - cu_h\|_{0,T}\|z-z_h\|_{0,T} + \|f-f_h\|_{0,T}\|z-z_h\|_{0,T} \\
&\overset{(6.3)}{\le} C_I \sum_{T\in\mathcal{T}_h} h_T^2\Big(\|f_h + \nu\Delta u_h - b\cdot\nabla u_h - cu_h\|_{0,T} + \|f-f_h\|_{0,T}\Big)|z|_{2,T} \\
&\overset{(6.41)}{\le} CC_IC_S \left(\sum_{T\in\mathcal{T}_h} h_T^4\|f_h + \nu\Delta u_h - b\cdot\nabla u_h - cu_h\|_{0,T}^2 + h_T^4\|f-f_h\|_{0,T}^2\right)^{1/2} \|u-u_h\|_0
\end{aligned}
$$

und

$$
\begin{aligned}
&\sum_{T\in\mathcal{T}_h}\sum_{E\subset\partial T}\int_E \nu\nabla(u-u_h)\cdot n_E(z-z_h)ds \\
&= \sum_E -\int_E [|\nu\nabla u_h\cdot n_E|]_E\,(z-z_h)ds \\
&\le \sum_E \|\,[|\nu\nabla u_h\cdot n_E|]_E\,\|_{0,E}\|z-z_h\|_{0,E} \\
&\overset{(6.4)}{\le} C_I\sum_E h_E^{3/2}\|\,[|\nu\nabla u_h\cdot n_E|]_E\,\|_{0,E}|z|_{2,\omega_E} \\
&\overset{(6.41)}{\le} CC_IC_S\left(\sum_E h_E^3\|\,[|\nu\nabla u_h\cdot n_E|]_E\,\|_{0,E}^2\right)^{1/2}\|u-u_h\|_0.
\end{aligned}
$$

Für den dritten Term erhält man

$$
\sum_{T\in\mathcal{T}_h}\sum_{E\subset\partial T}\int_E \nu\nabla z\cdot n_E u_h ds \quad = \quad \sum_E\int_E \nu\nabla z\cdot n_E\,[|u_h|]_E\,ds
$$

$$
\begin{aligned}
&\stackrel{(6.5)}{=} \sum_E \int_E \nu(\nabla z - \mathcal{M}_E^0(\nabla z)) \cdot n_E \, [|u_h|]_E \, ds \\
&\stackrel{(6.6)}{\leq} C \sum_E h_E^{1/2} \|\nu \, [|u_h|]_E \, \|_{0,E} |\nabla z \cdot n_E|_{1,\omega_E} \\
&\stackrel{(6.41)}{\leq} C \, C_S \left(\sum_E h_E \|\nu \, [|u_h|]_E \, \|_{0,E}^2 \right)^{1/2} \|u - u_h\|_0.
\end{aligned}
$$

Die Abschätzung des vierten Terms startet mit einer Aufsplittung dieses Terms

$$
\begin{aligned}
&\sum_{T \in \mathcal{T}_h} \sum_{E \subset \partial T} \int_E (b \cdot n_E) u_h z ds \\
&\quad = \sum_E \int_E (b(B_E) \cdot n_E) \, [|u_h|]_E \, z ds + \sum_E \int_E ((b - b(B_E)) \cdot n_E) \, [|u_h|]_E \, z ds,
\end{aligned}
$$

wobei B_E der Mittelpunkt von E ist. Der erste Summand wird wie folgt abgeschätzt

$$
\begin{aligned}
\sum_E \int_E (b(B_E) \cdot n_E) \, [|u_h|]_E \, z ds &\stackrel{(6.5)}{=} \sum_E (b(B_E) \cdot n_E) \int_E [|u_h|]_E \, (z - \mathcal{M}_E^0(z)) ds \\
&\stackrel{(6.6)}{\leq} C \sum_E h_E^{1/2} \|b(B_E) \cdot n_E\|_{\infty,E} \| \, [|u_h|]_E \, \|_{0,E} |z|_{1,\omega_E}.
\end{aligned}
$$

Für den zweiten Term gilt

$$
\sum_E \int_E ((b - b(B_E)) \cdot n_E) \, [|u_h|]_E \, z ds \leq \sum_E \|(b - b(B_E)) \cdot n_E\|_{\infty,E} \|z\|_{0,E} \| \, [|u_h|]_E \, \|_{0,E}.
$$

Bei Schieweck und Tobiska [ST96b] ist die Abschätzung

$$
\|z\|_{0,E} \leq C \left(h_E^{-1/2} \|z\|_{0,\omega_E} + h_E^{1/2} |z|_{1,\omega_E} \right) \qquad \forall z \in H^2(\omega_E)
$$

bewiesen. Da wegen den vorausgesetzten Glattheitsbedingungen $b \in (W^{1,\infty}(E))^2$ gilt, folgt mit Taylor–Entwicklung

$$
\|(b - b(B_E)) \cdot n_E\|_{\infty,E} \leq C h_E |b \cdot n_E|_{1,\infty,E}.
$$

Setzt man dies ein, so erhält man

$$
\begin{aligned}
\sum_E \int_E ((b - b(B_E)) \cdot n_E) \, [|u_h|]_E \, z ds &\leq C \sum_E |b \cdot n_E|_{1,\infty,E} \left(h_E^{1/2} \|z\|_{0,\omega_E} + h_E^{3/2} |z|_{1,\omega_E} \right) \\
&\leq C \sum_E |b \cdot n_E|_{1,\infty,E} h_E^{1/2} \|z\|_{1,\omega_E}.
\end{aligned}
$$

Hierbei wurden die höheren Potenzen von h_E in die Konstante C absorbiert. Da $\|b(B_E) \cdot n_E\|_{\infty,E} \leq \|b \cdot n_E\|_{\infty,E}$ ist, erhält man die Abschätzung

$$
\begin{aligned}
\sum_{T \in \mathcal{T}_h} \sum_{E \subset \partial T} \int_E (b \cdot n_E) u_h z ds &\leq C \left(\sum_E h_E \|b \cdot n_E\|_{1,\infty,E}^2 \| \, [|u_h|]_E \, \|_{0,E}^2 \right)^{1/2} \|z\|_1 \\
&\stackrel{(6.41)}{\leq} C \, C_S \left(\sum_E h_E \|b \cdot n_E\|_{1,\infty,E}^2 \| \, [|u_h|]_E \, \|_{0,E}^2 \right)^{1/2} \|u - u_h\|_0.
\end{aligned}
$$

Mit Hilfe von (6.41) erhält man schließlich noch

$$(f - \hat{f}_h, z_h) \le \|f - \hat{f}_h\|_0 \left(\|z - z_h\|_0 + \|z\|_0\right) \le CC_S\|f - \hat{f}_h\|_0\|u - u_h\|_0.$$

Die Behauptung ergibt sich aus den einzelnen Abschätzungen. □

Bemerkung 6.6.2 Der Term $\|f - \hat{f}_h\|_0$. Der Term $\|f - \hat{f}_h\|_0$ mißt alle Effekte, die sich daraus ergeben, daß u_h nicht Lösung der Standard–Galerkin Formulierung von (1.3) in X_h ist. Dies sind die Fehler, die sich durch die Stabilisierung, durch numerische Integration bei der Aufstellung der Matrix und der rechten Seite und durch den Abbruch des Iterationsverfahrens bei der Lösung des linearen Systems ergeben. Der Term $\|f - \hat{f}_h\|_0$ ist prinzipiell berechenbar, man wird ihn in der numerischen Praxis allerdings als von höherer Ordnung betrachten. Damit dies gerechtfertigt ist, braucht man wegen des Abbruchfehlers des Iterationsverfahrens, daß u_h eine Approximation der Lösung von (1.18) oder (1.21) ist.

Bemerkung 6.6.3 Die Rolle des konformen Unterraumes X_h von V_{h0}. Für das duale Problem wird keine Lösung eines diskreten Problems benötigt, sondern nur eine Interpolierte z_h. An dieser Stelle ist es vorteilhaft, daß V_{h0} den konformen Teilraum X_h enthält und man $z_h \in X_h$ wählen kann. Somit ergeben sich bei der nachfolgenden partiellen Integration keine Integrale auf Grund von Unstetigkeiten von z_h. Wegen $X_h \subset V_{h0}$ kann man das duale Problem sozusagen konform behandeln.

Bemerkung 6.6.4 Sprünge der Tangentialableitungen über die Kanten. Die Sprünge der nichtkonformen Funktion über die Kanten kann man durch Sprünge der Tangentialableitungen über die Kanten ersetzen, vergleiche Bemerkung 6.4.4.

Bemerkung 6.6.5 A posteriori Fehlerabschätzung für $|u - u_h|_h$. Dari et. all [DDPV96] haben eine a posteriori Fehlerabschätzung für $|u - u_h|_h$ und die Laplace Gleichung bewiesen. Die verwendete Technik der Helmholtz–Zerlegung des Fehlers konnten wir nicht auf Konvektions–Diffusions Gleichungen übertragen.

Für konforme Finite Elemente Diskretisierungen kann man mit der in diesem Abschnitt verwendeten Beweistechnik auch a posteriori Fehlerabschätzungen für $|u - u_h|_h$ beweisen. Man verwendet das duale Problem (6.39) mit dem Funktional

$$f : H^{-1}(\Omega) \mapsto \mathbb{R}, \quad w \to \sum_{T\in\mathcal{T}_h} (\nabla(u - u_h), \nabla w)_T$$

als rechte Seite. Für die Lösung z kann man nun nur $z \in H_0^1(\Omega)$ sowie H^1–Stabilität voraussetzen. Bei der Verwendung nichtkonformer Finite Elemente Diskretisierungen erhält man beim Übergang von der starken zur schwachen Formulierung des dualen Problems wie im Beweis von Satz 6.6.1 die Integrale

$$\sum_{T\in\mathcal{T}_h} \sum_{E\subset\partial T} \int_E \nu\nabla z \cdot n_E(u - u_h)ds.$$

Dieses Integral müßte durch $\|\nabla z\|_{U(E)}$ abgeschätzt werden, wobei $U(E)$ eine Umgebung von E ist, um die Stabilität des dualen Problems ausnutzen zu können. Eine derartige Abschätzung ist noch offen.

Bemerkung 6.6.6 Zur Robustheit des Fehlerschätzers. Der Fehlerschätzer η ist nicht robust im Sinne von Definition 6.1.1. Nach Satz 1.1.3 kann bei singulär gestörten Problemen im schlimmsten Fall $C_S(\nu) = \mathcal{O}(\nu^{-3/2})$ gelten. Für kleine ν ist C_S sehr groß, so daß der a posteriori Fehlerschätzer η allein, ohne Kenntnis der Größnordnung von C_S, keine Aussagekraft über den Fehler $\|u - u_h\|_0$ besitzt, siehe auch Abschnitt 8.3.

Bemerkung 6.6.7 Die Abschätzung des konvektiven Terms. Bei Gleichungen ohne Konvektion, $b \equiv (0,0)^T$, verändern sich die lokalen Summanden im Fehlerschätzer zu

$$\hat{\eta}_T^2 = h_T^4 \| f + \nu\Delta u_h - b \cdot \nabla u_h - c u_h \|_{0,T}^2 + \sum_{E \subset \partial T} h_E^3 \| [\![\nu \nabla u_h \cdot n_E]\!]_E \|_{0,E}^2 + h_E \| \nu [\![u_h]\!]_E \|_{0,E}^2, \quad (6.47)$$

das heißt, die Sprünge der nichtkonformen Funktion werden mit dem Faktor ν gewichtet. Den durch Summation von $\hat{\eta}_T^2$ definierten Fehlerschätzer nennen wir $\hat{\eta}$. Die Abschätzung des Terms

$$\sum_{T \in \mathcal{T}_h} \sum_{E \subset \partial T} \int_E (b \cdot n_E) u_h z ds$$

im Beweis von Satz 6.6.1 scheint in der Tat nicht optimal zu sein. Man nutzt dabei nicht die H^2–Regularität von z, sondern nur H^1–Regularität. Dadurch verschenkt man möglicherweise eine Potenz von h_E. Alternativ könnte man zusätzlich $\|z\|_1 \le C_S \nu \|u - u_h\|_0$ voraussetzen. Dies würde beispielsweise gelten, falls die a priori Abschätzung (1.7) scharf ist. In diesen Fällen erhält man im Konvektionsfall $b \not\equiv (0,0)^T$ ebenfalls Wichtungen der Sprünge der nichtkonformen Funktionen mit ν oder $h_E \|b\|_{\infty,E}$. Diese Wichtung würde zum lokalen Fehlerschätzer $\tilde{\eta}_{T,2}$ korrespondieren, siehe (6.32).

6.7 Verfeinerungs– und Vergröberungsstrategien

In den Abschnitten 6.4 und 6.5 wurden Schätzer η_T und $\tilde{\eta}_T$ des lokalen Fehlers untersucht, die zur adaptiven Gittergenerierung dienen sollen. Mit Hilfe dieser lokalen Fehlerschätzer wird jedem Element T ein Wert zugeordnet, an Hand dessen entschieden werden muß, ob das Element verfeinert oder vergröbert wird oder ob es unverändert bleiben soll. In diesem Abschnitt werden Strategien und Kriterien vorgestellt, mit deren Hilfe man diese Entscheidung treffen kann. Die gewählte Strategie mit den gewählten Parametern kann das Verhalten des numerischen Verfahrens und die berechneten Ergebnisse entscheidend beeinflussen.

Sei η_T ein beliebiger lokaler Fehlerschätzer. Die grundlegende Strategie der adaptiven Gittergenerierung besteht in der folgenden Vorgehensweise:

```
Markiere_Element(REFTOL, COARSETOL, η̄)
{
    markiere Element T zur Verfeinerung falls η_T ≥ REFTOL η̄
    markiere Element T zur Vergroeberung falls η_T ≤ COARSETOL η̄
}
```

In den Verfeinerungsprozeß werden keine irregulär verfeinerten Elemente einbezogen, da diese nicht verfeinert werden dürfen. Diese werden gelöscht, und an deren Stelle werden die entsprechenden Vaterelemente verwendet. Es gibt verschiedene Möglichkeiten, $\bar{\eta}$ zu wählen:

1. *Maximaler_lokaler_Fehler.*
$$\bar{\eta} = \eta_{max} = \max_T \eta_T.$$
Bei dieser Wahl kann ein einzelner großer Wert η_T entscheidenden Einfluß auf die adaptive Gittergenerierung besitzen.
2. *Mittlerer_lokaler_Fehler.*
$$\bar{\eta} = \eta_{aver} = \left(\frac{1}{N_{Topel}} \sum_T \eta_T^2 \right)^{1/2}.$$

3. *Anteil_des_globalen_Fehlers.* Eine von Dörfler [Dör96] vorgeschlagene Herangehensweise besteht darin, die kleinste Menge $\{T_i\}$ von Elementen zu finden, deren Summe lokaler geschätzter Fehler ein gewisser Teil des geschätzten globalen Fehlers ist,

$$\left(\sum_{T_i} \eta_T^2\right)^{1/2} \geq (1 - REFTOL) \left(\sum_{T} \eta_T^2\right)^{1/2}. \tag{6.48}$$

Desweiteren finde man die größte Menge $\{T_j\}$ mit

$$\left(\sum_{T_j} \eta_T^2\right)^{1/2} \leq COARSETOL \left(\sum_{T} \eta_T^2\right)^{1/2}. \tag{6.49}$$

Man markiert T zur Verfeinerung, falls $T \in \{T_i\}$ ist und zur Vergröberung, falls $T \in \{T_j\}$ ist. Bei dieser Vorgehensweise hat man ein Optimierungsproblem zu lösen, was approximativ geschieht. Mit Hilfe der Strategien 1) oder 2) und vorgegebenen Toleranzen REFTOL1 und COARSETOL1 werden Mengen $\{T_i\}$ und $\{T_j\}$ ausgewählt. Erfüllen diese Mengen die Forderungen (6.48) oder (6.49) nicht, so werden REFTOL1 solange verkleinert beziehungsweise COARSTOL1 vergrößert, bis man Mengen mit den geforderten Eigenschaften gefunden hat.

Darüber hinaus ist es sinnvoll, für stationäre Probleme eine minimale Anzahl von Elementen vorzugeben, die verfeinert oder vergröbert werden müssen. Im allgemeinen findet nach jeder Veränderung des Gitters eine neue numerische Lösung des Problems statt. Das lohnt sich nur, wenn hinreichend viele neue Unbekannte vorhanden sind. Aus diesem Grund kann man eine minimale Anzahl der zu verfeinernden Elemente MINREF in % angeben. Für instationäre Probleme macht diese Strategie jedoch keinen Sinn.

Ein typischer Algorithmus zur Markierung der Elemente zur Verfeinerung sieht wie folgt aus:

```
Markierung_fuer_Verfeinerung()
{
  1. Berechne alle lokalen Fehlerschaetzer η_T.
  2. Waehle eine der obigen Regeln zur Berechnung von η̄.
  3. Lese nutzerdefinierte Werte REFTOL, COARSETOL, MINREF ein.
  4. Markiere alle Elemente T mit η_T ≥ REFTOL η̄ zur Verfeinerung und
     alle Elemente T mit η_T ≤ COARSETOL η̄ zur Vergroeberung.
  5. Falls weniger als MINREF Elemente zur Verfeinerung und Vergroeberung
     markiert wurden, dann verkleinere REFTOL und vergrößere COARSETOL.
     Gehe zu 4.
}
```

Elemente, die zur Verfeinerung markiert sind, werden auch verfeinert. Aus Gründen des adaptiven Gitterabschlusses werden auch die Nachbarn dieser Elemente (irregulär) verfeinert. Vergröberungen finden nur statt, falls alle Söhne eines Elementes zur Vergröberung markiert sind und sich dies mit dem Abschluß der adaptiven Gitter vereinbaren läßt. In stationären Problemen haben Vergröberungen im allgemeinen keine Bedeutung.

Neben den obigen Toleranzen muß bei praktischen Rechnungen entschieden werden, ab welchem Level überhaupt adaptiv verfeinert werden soll (BASELEVEL). Für einen effizienten adaptiven Algorithmus sollten die wichtigsten Eigenschaften der Lösung, zum Beispiel die Lage von Grenzschichten, schon auf dem BASELEVEL erkennbar sein. Manchmal ist es auch sinnvoll, eine maximale Verfeinerungstiefe MAXLEVEL vorzugeben.

Kapitel 7

A Posteriori Fehlerabschätzungen für die Stokes und Navier–Stokes Gleichungen

In diesem Kapitel werden die a posteriori Fehlerabschätzungen für Konvektions–Diffusions Gleichungen aus den Abschnitten 6.4 bis 6.6 auf die Stokes und Navier–Stokes Gleichungen übertragen. Die grundlegenden Beweisideen sind die gleichen. Es werden jedoch andere Abschätzungstechniken für den konvektiven Term benötigt. Die lokale Fehlerabschätzung in der diskreten Energienorm stellt eine Erweiterung der Ergebnisse von Dari, Durán und Padra [DDP95] von Stokes Gleichungen auf Navier–Stokes Gleichungen dar. Die Abschätzungen in der L^2–Norm sind neu.

7.1 Voraussetzungen und Notationen

Wir betrachten in diesem Kapitel homogenen Dirichletprobleme und setzen voraus, daß diese eindeutig lösbar sind. Die Regularitätsvoraussetzung an die Lösung der Stokes Gleichungen (2.4) und der Navier–Stokes Gleichungen (2.6) ist

$$\mathbf{u} \in \left(H^2(\Omega)\right)^2 \cap \left(H_0^1(\Omega)\right)^2, \ p \in H^1(\Omega) \cap L_0^2(\Omega). \tag{7.1}$$

Für $\mathbf{v} = (v_1, v_2) \in (H_0^1(\Omega))^2$ definieren wir

$$\mathrm{curl}\mathbf{v} := \begin{pmatrix} -\frac{\partial v_1}{\partial y} & \frac{\partial v_1}{\partial x} \\ -\frac{\partial v_2}{\partial y} & \frac{\partial v_2}{\partial x} \end{pmatrix}.$$

Der Mittelwert einer Funktion auf der gemeinsamen Kante E der Dreiecke T_1 und T_2 wird mit

$$[|v|]_A = \frac{1}{2}(v|_{T_1} + v|_{T_2})$$

bezeichnet. Es gilt

$$[|uv|]_E = [|u|]_A \, [|v|]_E + [|u|]_E \, [|v|]_A \, .$$

Sei X_h der Raum der zweidimensionalen konformen stückweise linearen Funktionen. Die Eigenschaften (6.3) und (6.4) des Interpolationsoperators R_{X_h} übertragen sich auf vektorielle Funktionen. Darüber hinaus benötigen wir zusätzlich einen Interpolationsoperator $R_{Q_h} : H^1(\Omega) \mapsto Q_h$ mit der Eigenschaft

$$\|v - R_{Q_h} v\|_{0,T} \le C_3 h_T |v|_{1,\tilde{\omega}_T}, \tag{7.2}$$

wobei $\tilde{\omega}_T$ die Vereinigung aller Dreiecke ist, die einen gemeinsamen Punkt mit dem Dreieck T besitzen, siehe bei Verfürth [Ver96b]. Der Interpolationsoperator von Clément [Clé75] erfüllt diese Bedingung. Wir definieren $C_I := \max\{C_1, C_2, C_3\}$.

In diesem Kapitel wird eine reguläre Triangulierung $\mathcal{T}_h$ von Ω vorausgesetzt. Weitere verwendete Bezeichnungen sind im Abschnitt 6.2 eingeführt. Desweiteren werden die im Abschnitt 6.3 definierten Abschneidefunktionen benötigt.

Einschränkungen der Bilinearform $a_h(\cdot,\cdot)$ und der Trilinearform $b_h(\cdot,\cdot,\cdot)$ auf Teilgebiete $\omega \subset \Omega$ werden mit $a_\omega(\cdot,\cdot)$ beziehungsweise $b_\omega(\cdot,\cdot,\cdot)$ bezeichnet. Die folgende Aufsplittung des trilinearen Terms wird häufig benutzt

$$b_\omega(\mathbf{u},\mathbf{u},\mathbf{v}) - b_\omega(\mathbf{u}_h,\mathbf{u}_h,\mathbf{v}) = -b_\omega(\mathbf{u}-\mathbf{u}_h,\mathbf{u}-\mathbf{u}_h,\mathbf{v}) + b_\omega(\mathbf{u},\mathbf{u}-\mathbf{u}_h,\mathbf{v}) + b_\omega(\mathbf{u}-\mathbf{u}_h,\mathbf{u},\mathbf{v}). \quad (7.3)$$

Alle Beweise werden für die Navier–Stokes Gleichungen geführt. Die Ergebnisse für die Stokes Gleichungen erhält man durch Vernachlässigung der Abschätzungen für den nichtlinearen Term.

7.2 Lokale Abschätzung des Fehlers nach unten in der Energienorm

Wir definieren die lokalen Fehlerschätzer

$$\eta_{T,1}^2 := h_T^2\|\mathbf{f}_h - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h\|_{0,T}^2 + \sum_{E\subset\partial T} h_E\|\,[\![\nu\nabla\mathbf{u}_h\cdot n_E - p_h I n_E]\!]_E\,\|_{0,E}^2 + \|\nabla\cdot\mathbf{u}_h\|_{0,T}^2, \quad (7.4)$$

$$\begin{aligned}\eta_{T,2}^2 &:= h_T^2\|\mathbf{f}_h - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h\|_{0,T}^2 \\ &\quad + \sum_{E\subset\partial T}\left(h_E\|\,[\![\nu\nabla\mathbf{u}_h\cdot n_E - p_h I n_E]\!]_E\,\|_{0,E}^2 + h_E^{-1}\|\,[\![\mathbf{u}_h]\!]_E\,\|_{0,E}^2\right) + \|\nabla\cdot\mathbf{u}_h\|_{0,T}^2. \quad (7.5)\end{aligned}$$

Hierbei ist I die zweidimensionale Einheitsmatrix und $\mathbf{f}_h$ eine polynomiale Approximation an $\mathbf{f}$. Der Aufbau der Fehlerschätzer ist wie bei Konvektions–Diffusions Gleichungen. Der Fehlerschätzer $\eta_{T,2}$ besitzt im Gegensatz zu $\eta_{T,1}$ einen Term, der die Nichtkonformität der Funktion $\mathbf{u}_h$ mißt. Es wird gezeigt, daß in einer Umgebung der Lösung $(\mathbf{u},p)$ von (2.6) eine lokale Abschätzung der Gestalt

$$\eta_{T,i} \le C\left(\|\mathbf{u}-\mathbf{u}_h\|_{h,\omega_T}^2 + \|p-p_h\|_{0,\omega_T}^2\right)^{1/2} + \mathcal{O}\left(h_T^2\left(\|\mathbf{u}-\mathbf{u}_h\|_{h,\omega_T}^2 + \|p-p_h\|_{0,\omega_T}^2\right)^{1/2}\right),$$

$i = 1, 2$, gilt. Die Beweisidee ist die gleiche wie bei Konvektions–Diffusions Gleichungen, siehe Abschnitt 6.4. Der wesentlich Unterschied besteht in der Behandlung des konvektiven Terms, da der Konvektionsvektor bei den Navier–Stokes Gleichungen nun die Lösung selbst ist.

Lemma 7.2.1 *Sei $(\mathbf{u},p)$ die Lösung der Navier–Stokes Gleichungen und $(\mathbf{u}_h,p_h)\in V_{h0}\times Q_h$ ein beliebiges Paar von Funktionen. Dann gilt die Abschätzung*

$$\begin{aligned}h_T\|\mathbf{f}_h - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h\|_{0,T} \le\ & C\Big[\nu|\mathbf{u}-\mathbf{u}_h|_{1,T} + \|\mathbf{u}-\mathbf{u}_h\|_{0,T}|\mathbf{u}-\mathbf{u}_h|_{1,T} + \|\mathbf{u}\|_{0,T}|\mathbf{u}-\mathbf{u}_h|_{1,T} \\ & + \|\mathbf{u}-\mathbf{u}_h\|_{0,T}|\mathbf{u}|_{1,T} + \|p-p_h\|_{0,T} + h_T\|\mathbf{f}-\mathbf{f}_h\|_{0,T}\Big] \quad (7.6)\end{aligned}$$

für alle $T\in\mathcal{T}_h$.

Beweis: Der Beginn des Beweises ist analog zu Lemma 6.4.1. Sei $\mathbf{w}_h = (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h$. Dann ist

$$\|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}^2$$

$$
\begin{aligned}
&= \int_T (\mathbf{f}_h - \mathbf{w}_h)^2 dx \le C \int_T (\mathbf{f}_h - \mathbf{w}_h)^2 \mathrm{B}_T dx \\
&= C\Bigg[a_T(\mathbf{u} - \mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) + b_T(\mathbf{u}, \mathbf{u}, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) - b_T(\mathbf{u}_h, \mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) \\
&\quad + \int_T (p_h - p)\nabla \cdot ((\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) dx + \int_T (\mathbf{f}_h - \mathbf{f}) \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T dx \Bigg].
\end{aligned}
$$

Der erste und der letzte Summand werden wie bei den Konvektions–Diffusions Gleichungen abgeschätzt. Man erhält

$$
a_T(\mathbf{u} - \mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) \le C \frac{\nu}{h_T} |\mathbf{u} - \mathbf{u}_h|_{1,T} \|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}
$$

und

$$
\int_T (\mathbf{f}_h - \mathbf{f}) \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T dx \le \|\mathbf{f} - \mathbf{f}_h\|_{0,T} \|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}.
$$

Der Term, der den Druck enthält, wird wie folgt behandelt:

$$
\begin{aligned}
\int_T (p_h - p)\nabla \cdot ((\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) dx &\le \|p - p_h\|_{0,T} \|\nabla \cdot ((\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T\|_{0,T} \\
&\le \|p - p_h\|_{0,T} |(\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T|_{1,T} \\
&\overset{(6.7),(6.15)}{\le} C h_T^{-1} \|p - p_h\|_{0,T} \|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}.
\end{aligned}
$$

Es bleibt die Abschätzung der nichtlinearen Konvektionsterme mit Hilfe der Aufsplittung (7.3):

$$
\begin{aligned}
& b_T(\mathbf{u}, \mathbf{u}, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) - b_T(\mathbf{u}_h, \mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) \\
&\quad = -b_T(\mathbf{u} - \mathbf{u}_h, \mathbf{u} - \mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) + b_T(\mathbf{u}, \mathbf{u} - \mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) \\
&\qquad + b_T(\mathbf{u} - \mathbf{u}_h, \mathbf{u}, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T) \\
&\quad \overset{(2.8)}{\le} \Big[\|\mathbf{u} - \mathbf{u}_h\|_{0,T} |\mathbf{u} - \mathbf{u}_h|_{1,T} + \|\mathbf{u}\|_{0,T} |\mathbf{u} - \mathbf{u}_h|_{1,T} \\
&\qquad + \|\mathbf{u} - \mathbf{u}_h\|_{0,T} |\mathbf{u}|_{1,T} \Big] \|(\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T\|_{\infty,T} \\
&\quad \overset{(6.8),(6.15)}{\le} C h_T^{-1} \Big[\|\mathbf{u} - \mathbf{u}_h\|_{0,T} |\mathbf{u} - \mathbf{u}_h|_{1,T} + \|\mathbf{u}\|_{0,T} |\mathbf{u} - \mathbf{u}_h|_{1,T} \\
&\qquad + \|\mathbf{u} - \mathbf{u}_h\|_{0,T} |\mathbf{u}|_{1,T} \Big] \|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}.
\end{aligned}
$$

Faßt man alle Abschätzungen zusammen, so erhält man die Behauptung. □

Als nächstes werden die Terme mit den Sprüngen der Normalenableitung von $\mathbf{u}_h$ und den Sprüngen von p_h abgeschätzt. Der Sprung $[|\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E|]_E$ ist auf jeder Kante eine konstante vektorielle Funktion. Mit $[|\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C$ wird die Fortsetzung auf ω_E bezeichnet. Für reguläre Triangulierungen gilt

$$
\left\| [|\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \right\|_{0,\omega_E} \le C h_E^{1/2} \| [|\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E|]_E \|_{0,E}. \tag{7.7}
$$

Lemma 7.2.2 *Unter den Voraussetzungen einer regulären Triangulierung und der Regularität (7.1) für die exakte Lösung* $(\mathbf{u}, p)$ *gilt für jede Funktion* $(\mathbf{u}_h, p_h) \in V_{h0} \times Q_h$ *und für alle Kanten* E *die Abschätzung*

$$
\begin{aligned}
h_E^{1/2} \| [|\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E|]_E \|_{0,E} \le\ & C \big[\nu |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} + \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_E} |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} \\
& + \|\mathbf{u}\|_{0,\omega_E} |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} + \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_E} |\mathbf{u}|_{1,\omega_E} \\
& + \|p - p_h\|_{0,\omega_E} + h_E \|\mathbf{f} - \mathbf{f}_h\|_{0,\omega_E} \big].
\end{aligned}
$$

Beweis: Die Vorgehensweise ist wie in Lemma 6.4.2. Die Funktionen $\nabla \mathbf{u}$ und p sind fast überall stetig. Es ist

$$
\begin{aligned}
&\| [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E \|_{0,E}^2 \\
&\quad \le C \Big\| a_{\omega_E}(\mathbf{u}_h - \mathbf{u}, [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E) \\
&\qquad - \int_{\omega_E} (p_h - p) \nabla \cdot ([\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E) dx \\
&\qquad + \sum_{T \subset \omega_E} \int_T (\nu \Delta(\mathbf{u}_h - \mathbf{u}) - \nabla(p_h - p)) \cdot [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E dx \Big\| .
\end{aligned}
$$

Die ersten Summanden werden wie in Lemma 6.4.2 abgeschätzt. Man erhält mit (7.7)

$$
\begin{aligned}
&a_{\omega_E}(\mathbf{u}_h - \mathbf{u}, [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E) \\
&\qquad - \int_{\omega_E} (p - p_h) \nabla \cdot ([\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E) dx \\
&\quad \le C h_E^{-1/2} (\nu |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} + \|p - p_h\|_{0,\omega_E}) \| [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E \|_{0,E}.
\end{aligned}
$$

Der letzte Summand wird in zwei Anteile zerlegt. Die Behandlung des nichtlinearen Konvektionsterms erfolgt wie in Lemma 7.2.1 mit Hilfe der Aufsplittung (7.3):

$$
\begin{aligned}
&\sum_{T \subset \omega_E} \int_T (\nu \Delta(\mathbf{u}_h - \mathbf{u}) - \nabla(p_h - p)) \cdot [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E dx \\
&= \int_{\omega_E} (\mathbf{f} - \mathbf{f}_h) \cdot [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E dx \\
&\qquad - b_{\omega_E}(\mathbf{u}, \mathbf{u}, [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E) + b_{\omega_E}(\mathbf{u}_h, \mathbf{u}_h, [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E) \\
&\qquad + \sum_{T \subset \omega_E} \int_T (\mathbf{f}_h - (\mathbf{u}_h \cdot \nabla) \mathbf{u}_h) \cdot [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E dx \\
&\overset{(2.8)}{\le} C h_E^{1/2} \Big[\|\mathbf{f} - \mathbf{f}_h\|_{0,\omega_E} + h_E^{-1} \Big(\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_E} |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} + \|\mathbf{u}\|_{0,\omega_E} |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} \\
&\qquad + \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_E} |\mathbf{u}|_{1,\omega_E} \Big) + \|\mathbf{f}_h - (\mathbf{u}_h \cdot \nabla) \mathbf{u}_h\|_{0,\omega_E} \Big] \| [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E \|_{0,E}.
\end{aligned}
$$

An dieser Stelle wird nicht genutzt, daß $(\mathbf{u}_h, p_h)$ die diskrete Gleichung löst, sondern es ist lediglich eine Null addiert worden. Den letzten Summanden kann man mit (7.6) abschätzen. Man erhält

$$
\begin{aligned}
&\sum_{T \subset \omega_E} \int_T (\nu \Delta(\mathbf{u}_h - \mathbf{u}) - \nabla(p_h - p)) \cdot [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E^C \mathrm{B}_E dx \\
&\le C \Big[h_E^{-1/2} \Big(\nu |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} + \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_E} |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} + \|\mathbf{u}\|_{0,\omega_E} |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E} \\
&\qquad + \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_E} |\mathbf{u}|_{1,\omega_E} + \|p - p_h\|_{0,\omega_E} \Big) + h_E^{1/2} \|\mathbf{f} - \mathbf{f}_h\|_{0,\omega_E} \Big] \| [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E \|_{0,E}.
\end{aligned}
$$

Damit folgt die Behauptung. □

Lemma 7.2.3 *Sei $\mathcal{T}_h$ eine reguläre Triangulierung. Unter der Voraussetzung (7.1) an die Regularität der exakten Lösung gilt für alle $\mathbf{u}_h \in V_{h0}$ und alle Kanten E*

$$
h_E^{-1/2} \| [\![\mathbf{u}_h]\!]_E \|_{0,E} \le C |\mathbf{u} - \mathbf{u}_h|_{h,\omega_E}.
$$

Beweis: Die Beweisführung ist analog zu Lemma 6.4.3 und Bemerkung 6.4.4. Dort sind lediglich Eigenschaften von u und u_h genutzt worden, die unabhängig von der zugrunde liegenden Gleichung waren. Diese Eigenschaften besitzen $\mathbf{u}$ und $\mathbf{u}_h$ ebenfalls. □

Lemma 7.2.4 *Es gilt für alle $\mathbf{u}_h \in V_{h0}$ und für alle $T \in \mathcal{T}_h$*

$$\|\nabla \cdot \mathbf{u}_h\|_{0,T} \leq |\mathbf{u} - \mathbf{u}_h|_{1,T}.$$

Beweis: Bei dieser Abschätzung braucht man nur die Divergenzfreiheit von $\mathbf{u}$ auszunutzen:

$$\|\nabla \cdot \mathbf{u}_h\|_{0,T} = \|\nabla \cdot (\mathbf{u} - \mathbf{u}_h)\|_{0,T} \leq |\mathbf{u} - \mathbf{u}_h|_{1,T}.$$

□

Wir sind daran interessiert, den Term $\|\mathbf{f} - \mathbf{f}_h\|_{0,\omega_T}$ als von höherer Ordnung anzusehen und setzen deswegen

$$\|\mathbf{f} - \mathbf{f}_h\|_{0,\omega_T} = \mathcal{O}\left(h_T \left(\|\mathbf{u} - \mathbf{u}_h\|^2_{h,\omega_T} + \|p - p_h\|^2_{0,\omega_T}\right)^{1/2}\right) \tag{7.8}$$

voraus.

Satz 7.2.5 Lokale a posteriori Fehlerabschätzung für die Navier–Stokes Gleichungen und den Fehler in der Energienorm. *Unter der Regularitätsvoraussetzung (7.1) an die exakte Lösung $(\mathbf{u}, p)$ und der Approximationseigenschaft (7.8) gilt*

$$\begin{aligned}\eta_{T,i} \quad \leq \quad & C\,(|\mathbf{u} - \mathbf{u}_h|_{h,\omega_T} + \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}|\mathbf{u} - \mathbf{u}_h|_{h,\omega_T} + \|\mathbf{u}\|_{0,\omega_T}|\mathbf{u} - \mathbf{u}_h|_{h,\omega_T} \\ & +\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}|\mathbf{u}|_{1,\omega_T} + \|p - p_h\|_{0,\omega_T}) + \mathcal{O}\left(h_T^2 \left(\|\mathbf{u} - \mathbf{u}_h\|^2_{h,\omega_T} + \|p - p_h\|^2_{0,\omega_T}\right)^{1/2}\right),\end{aligned} \tag{7.9}$$

$i = 1, 2$, für alle Elemente T einer regulären Triangulierung $\mathcal{T}_h$.

Ist die berechnete Geschwindigkeit $\mathbf{u}_h$ bereits hinreichend nahe an der Lösung $\mathbf{u}$ im Sinne von

$$\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T} \leq C\|\mathbf{u}\|_{0,\omega_T} \quad \textit{oder} \quad |\mathbf{u} - \mathbf{u}_h|_{h,\omega_T} \leq C|\mathbf{u}|_{1,\omega_T},$$

so gilt für $(\mathbf{u}_h, p_h) \in V_{h0} \times Q_h$ und $T \in \mathcal{T}_h$

$$\eta_{T,i} \leq C\|\mathbf{u}\|_{1,\omega_T} \left(\|\mathbf{u} - \mathbf{u}_h\|^2_{h,\omega_T} + \|p - p_h\|^2_{0,\omega_T}\right)^{1/2} + \mathcal{O}\left(h_T^2 \left(\|\mathbf{u} - \mathbf{u}_h\|^2_{h,\omega_T} + \|p - p_h\|^2_{0,\omega_T}\right)^{1/2}\right). \tag{7.10}$$

Beweis: Die Aussage (7.9) ergibt sich aus den vorangegangenen Lemmata. Ist $\mathbf{u}_h$ hinreichend nahe an $\mathbf{u}$, so kann man $\|\mathbf{u}-\mathbf{u}_h\|_{0,\omega_T}|\mathbf{u}-\mathbf{u}_h|_{h,\omega_T}$ in einen der anderen beiden Terme absorbieren, die sich bei der Abschätzung des nichtlinearen Terms ergeben haben, und dann

$$\|\mathbf{u}\|_{0,\omega_T}|\mathbf{u} - \mathbf{u}_h|_{h,\omega_T} + \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}|\mathbf{u}|_{h,\omega_T} \leq \|\mathbf{u}\|_{1,\omega_T}\|\mathbf{u} - \mathbf{u}_h\|_{h,\omega_T}$$

abschätzen. □

Die Aussagen der Bemerkungen 6.4.6 und 6.4.8 lassen sich sinngemäß hierher übertragen. Der Faktor $\|\mathbf{u}\|_{1,\omega_T}$ in (7.10) hat zur Folge, daß diese Abschätzung im allgemeinen nicht gleichmäßig in ν ist.

Satz 7.2.6 Lokale a posteriori Fehlerabschätzung für die Stokes Gleichungen und den Fehler in der Energienorm. *Für die Stokes Gleichungen (2.4) werden die Fehlerschätzer*

$$\begin{aligned}
\eta_{T,1}^2 &:= h_T^2\|\mathbf{f}_h\|_{0,T}^2 + \sum_{E\subset\partial T} h_E \|\, [\![\nu\nabla\mathbf{u}_h\cdot n_E - p_h I n_E]\!]_E \,\|_{0,E}^2 + \|\nabla\cdot\mathbf{u}_h\|_{0,T}^2 \\
\eta_{T,2}^2 &:= h_T^2\|\mathbf{f}_h\|_{0,T}^2 + \sum_{E\subset\partial T} \left(h_E \|\, [\![\nu\nabla\mathbf{u}_h\cdot n_E - p_h I n_E]\!]_E \,\|_{0,E}^2 + h_E^{-1}\|\, [\![\mathbf{u}_h]\!]_E \,\|_{0,E}^2 \right) \\
&\quad + \|\nabla\cdot\mathbf{u}_h\|_{0,T}^2 \qquad (7.11)
\end{aligned}$$

definiert. Unter der Regularitätsvoraussetzung (7.1) an die exakte Lösung $(\mathbf{u},p)$ *und der Approximationseigenschaft (7.8) gilt für alle* $(\mathbf{u}_h,p_h)\in V_{h0}\times Q_h$

$$\eta_{T,i} \le C\left(|\mathbf{u}-\mathbf{u}_h|_{h,\omega_T}^2 + \|p-p_h\|_{0,\omega_T}^2\right)^{1/2} + \mathcal{O}\left(h_T^2\left(\|\mathbf{u}-\mathbf{u}_h\|_{h,\omega_T}^2 + \|p-p_h\|_{0,\omega_T}^2\right)^{1/2}\right),$$

$i=1,2$, *für alle Elemente* T *einer regulären Triangulierung* $\mathcal{T}_h$.

7.3 Lokale Abschätzung des Fehlers nach unten in der L^2–Norm für die Geschwindigkeit und in einer gitterabhängigen L^2–Norm für den Druck

In Fortsetzung der bisherigen Vorgehensweise wird ein Fehlerschätzer der Gestalt

$$\begin{aligned}
\tilde\eta_T^2 &:= h_T^4\|\mathbf{f}_h - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h\|_{0,T}^2 + \sum_{E\subset\partial T}\left(h_E^3 \|\, [\![\nu\nabla\mathbf{u}_h\cdot n_E - p_h I n_E]\!]_E \,\|_{0,E}^2 + h_E \|\, [\![\mathbf{u}_h]\!]_E \,\|_{0,E}^2 \right) \\
&\quad + h_T^2\|\nabla\cdot\mathbf{u}_h\|_{0,T}^2 \qquad (7.12)
\end{aligned}$$

definiert, wobei $\mathbf{f}_h$ eine polynomiale Approximation an $\mathbf{f}$ ist. Die grundlegende Vorgehensweise ist wie im Abschnitt 6.5. Die Abschätzung des nichtlinearen Konvektionstermes erfolgt wie im Abschnitt 7.2.

Lemma 7.3.1 *Sei eine reguläre Triangulierung* $\mathcal{T}_h$ *gegeben. Dann gilt für alle* $\mathbf{u}_h\in V_{h0}$ *und alle Elemente* $T\in\mathcal{T}_h$

$$h_T\|\nabla\cdot\mathbf{u}_h\|_{0,T} \le C\|\mathbf{u}-\mathbf{u}_h\|_{0,T}. \qquad (7.13)$$

Beweis: Nutzt man die Divergenzfreiheit von $\mathbf{u}$ so folgt

$$\begin{aligned}
\|\nabla\cdot\mathbf{u}_h\|_{0,T}^2 &\overset{(6.14)}{\le} C\int_T (\nabla\cdot\mathbf{u}_h)(\nabla\cdot\mathbf{u}_h)\mathrm{B}_T^* dx \\
&= C\int_T \nabla\cdot(\mathbf{u}_h-\mathbf{u})(\nabla\cdot\mathbf{u}_h)\mathrm{B}_T^* dx \\
&\overset{p.I.,(6.10)}{=} C\int_T (\mathbf{u}_h-\mathbf{u})\nabla((\nabla\cdot\mathbf{u}_h)\mathrm{B}_T^*) dx \\
&\le C\|\mathbf{u}-\mathbf{u}_h\|_{0,T}\|\nabla((\nabla\cdot\mathbf{u}_h)\mathrm{B}_T^*)\|_{0,T} \\
&\overset{(6.7),(6.16)}{\le} C h_T^{-1}\|\mathbf{u}-\mathbf{u}_h\|_{0,T}\|\nabla\cdot\mathbf{u}_h\|_{0,T}.
\end{aligned}$$

□

Lemma 7.3.2 *Seien eine reguläre Triangulierung $\mathcal{T}_h$ sowie eine exakte Lösung $(\mathbf{u}, p)$ mit der Regularität (7.1) gegeben. Dann gilt für alle $(\mathbf{u}_h, p_h) \in V_{h0} \times Q_h$ und für alle Elemente $T \in \mathcal{T}_h$*

$$\begin{aligned} h_T^2 \|\mathbf{f}_h - (\mathbf{u}_h \cdot \nabla)\mathbf{u}_h\|_{0,T} \quad \leq \quad & C\Big[(\nu + \|\mathbf{u}\|_{0,T} + \|\mathbf{u} - \mathbf{u}_h\|_{0,T})\|\mathbf{u} - \mathbf{u}_h\|_{0,T} + h_T\|p - p_h\|_{0,T} \\ & + h_T^2 \|\mathbf{f} - \mathbf{f}_h\|_{0,T}\Big]. \end{aligned} \tag{7.14}$$

Beweis: Sei $\mathbf{w}_h = (\mathbf{u}_h \cdot \nabla)\mathbf{u}_h$. Ähnlich wie im Beweis von Lemma 6.5.1 erhält man nach zweimaliger partieller Integration der Terme die $\mathbf{u}$ enthalten

$$\begin{aligned} & \|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}^2 \\ & \leq C \Bigg[\int_T -\nu(\mathbf{u} - \mathbf{u}_h) \cdot \Delta((\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*)dx + b_T(\mathbf{u}, \mathbf{u}, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*) - b_T(\mathbf{u}_h, \mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*) \\ & \qquad + \int_T (p_h - p)\nabla \cdot ((\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*)dx + \int_T (\mathbf{f} - \mathbf{f}_h) \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^* dx \Bigg]. \end{aligned}$$

Es wurden die Eigenschaften $\Delta \mathbf{u}_h = 0, \nabla p_h = \mathbf{0}$ in T sowie (6.10) und (6.11) genutzt. Der erste und der letzte Term werden in gewohnter Art und Weise abgeschätzt. Man erhält

$$\int_T -\nu(\mathbf{u} - \mathbf{u}_h) \cdot \Delta((\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*)dx \leq C h_T^{-2} \nu \|\mathbf{u} - \mathbf{u}_h\|_{0,T} \|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}$$

und

$$\int_T (\mathbf{f} - \mathbf{f}_h) \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^* dx \leq C\|\mathbf{f} - \mathbf{f}_h\|_{0,T}\|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}.$$

Die Abschätzung des Terms, der den Druck enthält, erfolgt wie im Beweis von Lemma 7.2.1. Zur Abschätzung des nichtlinearen Konvektionsterms wird noch eine partielle Integration angewandt. Mit Lemma 2.2.3 ergibt sich

$$\begin{aligned} & b_T(\mathbf{u}, \mathbf{u}, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*) - b_T(\mathbf{u}_h, \mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*) \\ & = \quad -b_T(\mathbf{u}, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*, \mathbf{u}) - \int_T (\nabla \cdot \mathbf{u})\mathbf{u} \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^* dx \\ & \qquad + b_T(\mathbf{u}_h, (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*, \mathbf{u}_h) + \int_T (\nabla \cdot \mathbf{u}_h)\mathbf{u}_h \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^* dx. \end{aligned}$$

Die Randintegrale verschwinden wegen (6.10). Der zweite Summand verschwindet wegen der Divergenzfreiheit von $\mathbf{u}$. Weiter gilt

$$\begin{aligned} & \int_T (\nabla \cdot \mathbf{u}_h)\mathbf{u}_h \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^* dx \\ & = \quad \int_T (\nabla \cdot \mathbf{u}_h)(\mathbf{u}_h - \mathbf{u}) \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^* dx + \int_T (\nabla \cdot \mathbf{u}_h)\mathbf{u} \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^* dx \\ & \leq \quad \Big(\|\mathbf{u} - \mathbf{u}_h\|_{0,T}\|\nabla \cdot \mathbf{u}_h\|_{0,T} + \|\mathbf{u}\|_{0,T}\|\nabla \cdot \mathbf{u}_h\|_{0,T}\Big)\|(\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^*\|_{\infty,T}. \end{aligned}$$

Unter Verwendung von (7.13), (6.8) und (6.16) erhält man schließlich

$$\int_T (\nabla \cdot \mathbf{u}_h)\mathbf{u}_h \cdot (\mathbf{f}_h - \mathbf{w}_h)\mathrm{B}_T^* dx \quad \leq \quad C h_T^{-2} \left(\|\mathbf{u} - \mathbf{u}_h\|_{0,T}^2 + \|\mathbf{u}\|_{0,T}\|\mathbf{u} - \mathbf{u}_h\|_{0,T}\right) \|\mathbf{f}_h - \mathbf{w}_h\|_{0,T}.$$

Desweiteren ist

$$\begin{aligned}
&-b_T(\mathbf{u},(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*,\mathbf{u})+b_T(\mathbf{u}_h,(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*,\mathbf{u}_h)\\
&\quad = \quad b_T(\mathbf{u}-\mathbf{u}_h,(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*,\mathbf{u}-\mathbf{u}_h)+b_T(\mathbf{u},(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*,\mathbf{u}_h-\mathbf{u})\\
&\qquad\quad +b_T(\mathbf{u}_h-\mathbf{u},(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*,\mathbf{u})\\
&\quad \overset{(2.9)}{\leq} \quad \Big[\|\mathbf{u}-\mathbf{u}_h\|_{0,T}\|\mathbf{u}-\mathbf{u}_h\|_{0,T}+\|\mathbf{u}\|_{0,T}\|\mathbf{u}-\mathbf{u}_h\|_{0,T}+\|\mathbf{u}-\mathbf{u}_h\|_{0,T}\|\mathbf{u}\|_{0,T}\Big]\\
&\qquad\quad \|\nabla((\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*)\|_{\infty,T}.
\end{aligned}$$

Unter Verwendung von (6.8) und (6.7) ergibt sich

$$\begin{aligned}
&-b_T(\mathbf{u},(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*,\mathbf{u})+b_T(\mathbf{u}_h,(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*,\mathbf{u}_h)\\
&\quad\leq\quad Ch_T^{-2}\left(\|\mathbf{u}-\mathbf{u}_h\|_{0,T}^2+\|\mathbf{u}\|_{0,T}\|\mathbf{u}-\mathbf{u}_h\|_{0,T}\right)\|\mathbf{f}_h-\mathbf{w}_h\|_{0,T}
\end{aligned}$$

und als Gesamtabschätzung für den Konvektionsterm

$$\begin{aligned}
&b_T(\mathbf{u},\mathbf{u},(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*)-b_T(\mathbf{u}_h,\mathbf{u}_h,(\mathbf{f}_h-\mathbf{w}_h)\mathrm{B}_T^*)\\
&\quad\leq\quad Ch_T^{-2}\left(\|\mathbf{u}-\mathbf{u}_h\|_{0,T}^2+\|\mathbf{u}\|_{0,T}\|\mathbf{u}-\mathbf{u}_h\|_{0,T}\right)\|\mathbf{f}_h-\mathbf{w}_h\|_{0,T}.
\end{aligned}$$

Damit ist die Behauptung bewiesen. □

Lemma 7.3.3 *Seien eine reguläre Triangulierung sowie eine exakte Lösung $(\mathbf{u},p)$ mit der Regularität (7.1) gegeben. Dann gilt für jede Funktion $\mathbf{u}_h\in V_{h0}$ und jede Kante E*

$$h_E^{1/2}\|\,[\![\mathbf{u}_h]\!]_E\,\|_{0,E}\leq C\left(\|\mathbf{u}-\mathbf{u}_h\|_{0,\omega_E}+h_E^2|\mathbf{u}|_{2,\omega_E}\right). \tag{7.15}$$

Beweis: Die Beweisführung ist analog zu Lemma 6.5.2. Dort wurden nur Eigenschaften von u und u_h verwendet, die unabhängig von der zugrunde liegenden Gleichung waren. Diese Eigenschaften besitzen $\mathbf{u}$ und $\mathbf{u}_h$ ebenfalls. □

Lemma 7.3.4 *Seien eine reguläre Triangulierung und eine exakte Lösung $(\mathbf{u},p)$ mit der Regularität (7.1) gegeben. Dann gilt für alle $(\mathbf{u}_h,p_h)\in V_{h0}\times Q_h$ und für alle Kanten E*

$$\begin{aligned}
&h_E^{3/2}\|\,[\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hIn_E]\!]_E\,\|_{0,E}\\
&\quad\leq\; C\Big[\Big((\nu+\|\mathbf{u}\|_{0,\omega_E}+\|\mathbf{u}-\mathbf{u}_h\|_{0,\omega_E}+h_E^{1/2}\|\,[\![\mathbf{u}_h]\!]_A\,\|_{0,E})\|\mathbf{u}-\mathbf{u}_h\|_{0,\omega_E}+h_E\|p-p_h\|_{0,\omega_E}\Big)\\
&\qquad\quad +h_E^{5/2}\|\,[\![\mathbf{u}_h]\!]_A\,\|_{0,E}|\mathbf{u}|_{2,\omega_E}+h_E^2\|\mathbf{f}-\mathbf{f}_h\|_{0,\omega_E}\Big].
\end{aligned}$$

Beweis: Die Funktionen $\nabla\mathbf{u}$ und p sind fast überall stetig. Es ist, vergleiche Beweis von Lemma 7.2.2,

$$\begin{aligned}
&\|\,[\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hIn_E]\!]_E\,\|_{0,E}^2\\
&\quad\leq\; C\Bigg[a_{\omega_E}(\mathbf{u}_h-\mathbf{u},[\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hIn_E]\!]_E^C\,\mathrm{B}_E^*)\\
&\qquad\quad -\int\limits_{\omega_E}(p_h-p)\nabla\cdot([\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hIn_E]\!]_E^C\,\mathrm{B}_E^*)dx\\
&\qquad\quad +\sum_{T\subset\omega_E}\int\limits_T(\nu\Delta(\mathbf{u}_h-\mathbf{u})-\nabla(p_h-p))\cdot[\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hIn_E]\!]_E^C\,\mathrm{B}_E^*dx\Bigg].
\end{aligned}$$

Der Druckterm wird wie in Lemma 7.2.2 abgeschätzt. Der erste Summand wird partiell integriert und wie in Lemma 6.5.3 abgeschätzt,

$$\begin{aligned} & a_{\omega_E}(\mathbf{u}_h - \mathbf{u}, [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^*) \\ &= -\int_{\omega_E} \nu(\mathbf{u}_h - \mathbf{u})\nabla\cdot\Big(\nabla([|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^*)\Big)dx \\ &\quad + \int_E [|\mathbf{u}_h|]_E \cdot [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E \, (\nabla \mathrm{B}_E^* \cdot n_E)ds. \end{aligned}$$

Das Integral auf der inneren Kante E verschwindet, da $\nabla \mathrm{B}_E^* \cdot n_E = 0$ gilt. Unter Nutzung von (6.7) und (6.20) erhält man

$$a_{\omega_E}(\mathbf{u}_h - \mathbf{u}, [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^*) \leq C h_E^{-3/2}\nu\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_E}\| \, [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E \, \|_{0,E}.$$

Es bleibt der letzte Summand. Mit (2.6) und einer partiellen Integration des nichtlinearen Konvektionsterms erhält man

$$\begin{aligned} & \sum_{T\subset\omega_E}\int_T (\nu\Delta(\mathbf{u}_h - \mathbf{u}) - \nabla(p_h - p))\cdot[|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^* dx \\ &= \int_{\omega_E} (\mathbf{f} - \mathbf{f}_h)\cdot[|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^* dx \\ &\quad + \sum_{T\subset\omega_E}\int_T (\mathbf{f}_h - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h)\cdot[|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^* dx \\ &\quad + b_{\omega_E}\Big(\mathbf{u}, [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^*, \mathbf{u}\Big) - \int_{\omega_E}(\nabla\cdot\mathbf{u})\mathbf{u}\cdot\Big([|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^*\Big)\,dx \\ &\quad - b_{\omega_E}\Big(\mathbf{u}_h, [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^*, \mathbf{u}_h\Big) \\ &\quad + \sum_{T\subset\omega_E}\int_T (\nabla\cdot\mathbf{u}_h)\mathbf{u}_h\cdot\Big([|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E^C \, \mathrm{B}_E^*\Big)\,dx \\ &\quad - \int_E [|(\mathbf{u}_h\cdot n_E)\mathbf{u}_h|]_E \cdot ([|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E \, \mathrm{B}_E^*)\,ds. \end{aligned}$$

Die Abschätzung des ersten Terms ist Standard. Den zweiten Term kann man mit (7.14) abschätzen und die restlichen Flächenintegrale wie im Beweis von Lemma 7.3.2. Für das Kantenintegral auf E erhält man mit Hilfe von (7.15)

$$\begin{aligned} & \int_E [|(\mathbf{u}_h\cdot n_E)\mathbf{u}_h|]_E \cdot ([|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E \, \mathrm{B}_E^*)\,ds \\ &= \int_E ([|\mathbf{u}_h\cdot n_E|]_E \, [|\mathbf{u}_h|]_A + [|\mathbf{u}_h\cdot n_E|]_A \, [|\mathbf{u}_h|]_E)\cdot([|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E \, \mathrm{B}_E^*)\,ds \\ &\leq \| \, [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E \, \mathrm{B}_E^*\|_{\infty,E}\Big(\| \, [|\mathbf{u}_h\cdot n_E|]_E \, \|_{0,E}\| \, [|\mathbf{u}_h|]_A \, \|_{0,E} \\ &\qquad + \| \, [|\mathbf{u}_h\cdot n_E|]_A \, \|_{0,E}\| \, [|\mathbf{u}_h|]_E \, \|_{0,E}\Big) \\ &\leq C h_E^{-1/2}\| \, [|\mathbf{u}_h|]_A \, \|_{0,E}\| \, [|\mathbf{u}_h|]_E \, \|_{0,E}\| \, [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E \, \|_{0,E} \\ &\leq C h_E^{-1}\| \, [|\mathbf{u}_h|]_A \, \|_{0,E}\Big(\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_E} + h_E^2|\mathbf{u}|_{2,\omega_E}\Big)\| \, [|\nu\nabla\mathbf{u}_h \cdot n_E - p_h I n_E|]_E \, \|_{0,E}. \end{aligned}$$

Kombiniert man die einzelnen Abschätzungen, so ergibt sich die Behauptung. □

Der Term $\|\mathbf{f} - \mathbf{f}_h\|_{0,\omega_T}$ soll wieder als von höherer Ordnung betrachtet werden. Dazu setzen wir die Approximationseigenschaft

$$\|\mathbf{f} - \mathbf{f}_h\|_{0,\omega_T} = \mathcal{O}\left(h_T \left(\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}^2 + h_T^2 \|p - p_h\|_{0,\omega_T}^2\right)^{1/2}\right) \tag{7.16}$$

voraus.

Satz 7.3.5 Lokale a posteriori Fehlerabschätzung für die Navier–Stokes Gleichungen und den Fehler in der L^2–Norm der Geschwindigkeit und einer gitterabhängigen L^2–Norm des Druckes. *Für eine reguläre Triangulierung, unter der Regularitätsvoraussetzung (7.1) für die exakte Lösung $(\mathbf{u}, p)$ sowie der Approximationseigenschaft (7.16) gilt für alle $(\mathbf{u}_h, p_h) \in V_{h0} \times Q_h$ und alle Elemente $T \in \mathcal{T}_h$*

$$\begin{aligned}
\tilde{\eta}_T \quad \le \quad & C \left\{ \left[\left(1 + \|\mathbf{u}\|_{0,\omega_T} + \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T} + \sum_{E \subset \partial T} h_E^{1/2} \| [\![\mathbf{u}_h]\!]_A \|_{0,E} \right)^2 \|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}^2 \right.\right. \\
& \left.\left. + h_T^2 \|p - p_h\|_{0,\omega_T}^2 \right]^{1/2} + \sum_{E \subset \partial T} \left(h_E^2 + h_E^{5/2} \| [\![\mathbf{u}_h]\!]_A \|_{0,E} \right) |\mathbf{u}|_{2,\omega_T} \right\} \\
& + \mathcal{O}\left(h_T^3 \left(\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}^2 + h_T^2 \|p - p_h\|_{0,\omega_T}^2 \right)^{1/2} \right).
\end{aligned}$$

Die Aussagen der Bemerkungen 6.4.8, 6.5.5 und 6.5.6 lassen sich sinngemäß hierher übertragen.

Satz 7.3.6 Lokale a posteriori Fehlerabschätzung für die Stokes Gleichungen und den Fehler in der L^2–Norm der Geschwindigkeit und einer gitterabhängigen L^2–Norm des Druckes. *Für die Stokes Gleichungen (2.4) werden die Fehlerschätzer*

$$\begin{aligned}
\tilde{\eta}_{T,1}^2 \quad &:= \quad h_T^4 \|\mathbf{f}_h\|_{0,T}^2 + \sum_{E \subset \partial T} h_E^3 \| [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E \|_{0,E}^2 + h_T^2 \|\nabla \cdot \mathbf{u}_h\|_{0,T}^2 \\
\tilde{\eta}_{T,2}^2 \quad &:= \quad h_T^4 \|\mathbf{f}_h\|_{0,T}^2 + \sum_{E \subset \partial T} \left(h_E^3 \| [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h I n_E]\!]_E \|_{0,E}^2 + h_E \| [\![\mathbf{u}_h]\!]_E \|_{0,E}^2 \right) \\
& \quad + h_T^2 \|\nabla \cdot \mathbf{u}_h\|_{0,T}^2
\end{aligned} \tag{7.17}$$

definiert. Für eine reguläre Triangulierung, unter der Regularitätsvoraussetzung (7.1) für die exakte Lösung $(\mathbf{u}, p)$ sowie der Approximationseigenschaft (7.16) gilt für alle $(\mathbf{u}_h, p_h) \in V_{h0} \times Q_h$ und alle Elemente $T \in \mathcal{T}_h$

$$\begin{aligned}
\tilde{\eta}_{T,1} \;&\le\; C \left(\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}^2 + h_T^2 \|p - p_h\|_{0,\omega_T}^2 \right)^{1/2} + \mathcal{O}\left(h_T^3 \left(\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}^2 + h_T^2 \|p - p_h\|_{0,\omega_T}^2 \right)^{1/2} \right), \\
\tilde{\eta}_{T,2} \;&\le\; C \left(\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}^2 + h_T^2 \|p - p_h\|_{0,\omega_T}^2 \right)^{1/2} + \mathcal{O}\left(h_T^3 \left(\|\mathbf{u} - \mathbf{u}_h\|_{0,\omega_T}^2 + h_T^2 \|p - p_h\|_{0,\omega_T}^2 \right)^{1/2} \right) \\
& \quad + C h_E^2 |\mathbf{u}|_{2,\omega_E}.
\end{aligned}$$

7.4 Die Abschätzung des globalen Fehlers für die Geschwindigkeit in der L^2–Norm

Die Beweisführung aus Abschnitt 6.6 läßt sich auch bei den Navier–Stokes Gleichungen mit homogenen Dirichletbedingungen

Finde $(\mathbf{u}, p) \in V_0 \times Q$ mit

$$a(\mathbf{u}, \mathbf{v}) + b(\mathbf{u}, \mathbf{u}, \mathbf{v}) - (p, \nabla \cdot \mathbf{v}) + (q, \nabla \cdot \mathbf{u}) \quad = \quad (\mathbf{f}, \mathbf{v}) \quad \forall (\mathbf{v}, q) \in V_0 \times Q \tag{7.18}$$

anwenden. Die starke Formulierung eines linearisierten, stetigen, dualen Problems mit dem Fehler $\mathbf{u} - \mathbf{u}_h$ als rechter Seite lautet

$$\begin{aligned} -\nu \Delta \mathbf{z} - (\mathbf{u} \cdot \nabla)\mathbf{z} + \nabla \mathbf{u}_h \cdot \mathbf{z} - \nabla s &= \mathbf{u} - \mathbf{u}_h && \text{in } \Omega \\ \nabla \cdot \mathbf{z} &= 0 && \text{in } \Omega \\ \mathbf{z} &= \mathbf{0} && \text{auf } \partial\Omega. \end{aligned} \tag{7.19}$$

Für die Lösung $(\mathbf{z}, s)$ setzen wir die Regularität

$$(\mathbf{z}, s) \in (H^2(\Omega) \cap H_0^1(\Omega))^2 \times (H^1(\Omega) \cap L_0^2(\Omega)) \tag{7.20}$$

und die Stabilität

$$\|\mathbf{z}\|_2 + \|s\|_1 \leq C_S(\nu, \mathbf{u}, \mathbf{u}_h)\|\mathbf{u} - \mathbf{u}_h\|_0 \tag{7.21}$$

voraus. Für die Geschwindigkeitskomponente $\mathbf{u}$ der Lösung von (7.18) verlangen wir zusätzlich

$$\mathbf{u} \in \left(W^{1,\infty}(\Omega)\right)^2. \tag{7.22}$$

Wie im Abschnitt 6.6 wird eine Funktion $\hat{\mathbf{f}}_h$ mit Hilfe der Standard–Galerkin Diskretisierung von (7.18) im Raum $X_h \times Q_h$ definiert:

$$\begin{aligned} &\sum_{T \in \mathcal{T}_h} a_T(\mathbf{u}_h, \mathbf{v}_h) + b_T(\mathbf{u}_h, \mathbf{u}_h, \mathbf{v}_h) - (p_h, \nabla \cdot \mathbf{v}_h)_T + (q_h, \nabla \cdot \mathbf{u}_h)_T \\ &\qquad = \sum_{T \in \mathcal{T}_h} (\hat{\mathbf{f}}_h, \mathbf{v}_h)_T \qquad \forall (\mathbf{v}_h, q_h) \in X_h \times Q_h. \end{aligned} \tag{7.23}$$

Satz 7.4.1 Globale a posteriori Fehlerabschätzung für die Navier–Stokes Gleichungen und den Fehler der Geschwindigkeit in der L^2–Norm. *Seien die Regularitätsvoraussetzung (7.1), (7.22) für die exakte Lösung $(\mathbf{u}, p)$ von (7.18), die Regularität (7.20) und die Stabilität (7.21) für die Lösung $(\mathbf{z}, s)$ von (7.19) und eine reguläre Triangulierung gegeben. Darüber hinaus sei $\|\mathbf{u} \cdot n_E\|_{1,\infty,E}$ für jede Kante E gleichmäßig beschränkt in ν. Dann gilt die globale a posteriori Fehlerabschätzung*

$$\begin{aligned} \|\mathbf{u} - \mathbf{u}_h\|_0 &\leq CC_IC_S\eta + CC_IC_S \left(\sum_{T \in \mathcal{T}_h} h_T^4 \|\mathbf{f} - \mathbf{f}_h\|_{0,T}^2 \right)^{1/2} + CC_S\|\mathbf{f} - \hat{\mathbf{f}}_h\|_{0,T} \\ &= CC_IC_S \left(\sum_{T \in \mathcal{T}_h} \eta_T^2 \right)^{1/2} + CC_IC_S \left(\sum_{T \in \mathcal{T}_h} h_T^4 \|\mathbf{f} - \mathbf{f}_h\|_{0,T}^2 \right)^{1/2} + CC_S\|\mathbf{f} - \hat{\mathbf{f}}_h\|_0 \end{aligned} \tag{7.24}$$

mit

$$\eta_T^2 = h_T^4 \|\mathbf{f}_h - (\mathbf{u}_h \cdot \nabla)\mathbf{u}_h\|_{0,T}^2 + h_T^2 \|\nabla \cdot \mathbf{u}_h\|_{0,T}^2 + \sum_{E \subset \partial T} h_E^3 \| [\![\nu \nabla \mathbf{u}_h \cdot n_E - p_h n_E]\!]_E \|_{0,E}^2 + h_E \| [\![\mathbf{u}_h]\!]_E \|_{0,E}^2$$

für $(\mathbf{u}_h, p_h) \in V_{h0} \times Q_h$.

Beweis: Die Beweisführung ist wie für Satz 6.6.1. Wird das duale Problem (7.19) mit der nichtkonformen Funktion $\mathbf{u}-\mathbf{u}_h$ getestet, so erhält man mit dem Fehler $p-p_h$ als Testfunktion aus Q

$$\begin{aligned}\|\mathbf{u}-\mathbf{u}_h\|_0^2 &= \sum_{T\in\mathcal{T}_h}\Big[(\nu\nabla\mathbf{z},\nabla(\mathbf{u}-\mathbf{u}_h))_T + b_T(\mathbf{u},\mathbf{u}-\mathbf{u}_h,\mathbf{z}) + b_T(\mathbf{u}-\mathbf{u}_h,\mathbf{u}_h,\mathbf{z})\\ &\quad +(s,\nabla\cdot(\mathbf{u}-\mathbf{u}_h))_T - (p-p_h,\nabla\cdot\mathbf{z})_T \\ &\quad + \sum_{E\subset\partial T}\int_E \nu(\nabla\mathbf{z}\cdot n_E)\cdot\mathbf{u}_h + s(\mathbf{u}_h\cdot n_E) + (\mathbf{z}\cdot\mathbf{u}_h)(\mathbf{u}\cdot n_E)\,ds\Big].\end{aligned} \tag{7.25}$$

Aus (7.18) und (7.23) erhält man die approximierte Galerkin Orthogonalität

$$\begin{aligned}0 &= \sum_{T\in\mathcal{T}_h}\Big[(\nu\nabla(\mathbf{u}-\mathbf{u}_h),\nabla\mathbf{v}_h)_T + b_T(\mathbf{u},\mathbf{u},\mathbf{v}_h) - b_T(\mathbf{u}_h,\mathbf{u}_h,\mathbf{v}_h) - (p-p_h,\nabla\cdot\mathbf{v}_h)_T\\ &\quad +(q_h,\nabla\cdot(\mathbf{u}-\mathbf{u}_h))_T - (\mathbf{f}-\hat{\mathbf{f}}_h,\mathbf{v}_h)_T\Big] \qquad \forall(\mathbf{v}_h,q_h)\in X_h\times Q_h.\end{aligned}$$

Setzt man $\mathbf{v}_h = R_{X_{h0}}\mathbf{z} = \mathbf{z}_h$ und $q_h = R_{Q_h}s = s_h$ und addiert dies zu (7.25), so ergibt sich

$$\begin{aligned}&\|\mathbf{u}-\mathbf{u}_h\|_0^2\\ &= \sum_{T\in\mathcal{T}_h}\Big[(\nu\nabla(\mathbf{z}-\mathbf{z}_h),\nabla(\mathbf{u}-\mathbf{u}_h))_T + (s-s_h,\nabla\cdot(\mathbf{u}-\mathbf{u}_h))_T - (p-p_h,\nabla\cdot(\mathbf{z}-\mathbf{z}_h))_T\\ &\quad -b_T(\mathbf{u},\mathbf{u},\mathbf{z}_h) + b_T(\mathbf{u}_h,\mathbf{u}_h,\mathbf{z}_h) + b_T(\mathbf{u},\mathbf{u}-\mathbf{u}_h,\mathbf{z}) + b_T(\mathbf{u}-\mathbf{u}_h,\mathbf{u}_h,\mathbf{z}) + (\mathbf{f}-\hat{\mathbf{f}}_h,\mathbf{z}_h)_T\\ &\quad + \sum_{E\subset\partial T}\int_E \nu(\nabla\mathbf{z}\cdot n_E)\cdot\mathbf{u}_h + s(\mathbf{u}_h\cdot n_E) + (\mathbf{z}\cdot\mathbf{u}_h)(\mathbf{u}\cdot n_E)\,ds\Big]\\ &= \sum_{T\in\mathcal{T}_h}\Big[(-\nu\Delta(\mathbf{u}-\mathbf{u}_h) + (\mathbf{u}\cdot\nabla)\mathbf{u} - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h + \nabla(p-p_h),\mathbf{z}-\mathbf{z}_h)_T\\ &\quad +(s-s_h,\nabla\cdot(\mathbf{u}-\mathbf{u}_h))_T + (\mathbf{f}-\hat{\mathbf{f}}_h,\mathbf{z}_h)_T\\ &\quad + \sum_{E\subset\partial T}\int_E (\nu\nabla(\mathbf{u}-\mathbf{u}_h)\cdot n_E - (p-p_h)n_E)\cdot(\mathbf{z}-\mathbf{z}_h) + \nu(\nabla\mathbf{z}\cdot n_E)\cdot\mathbf{u}_h + s(\mathbf{u}_h\cdot n_E)\\ &\quad +(\mathbf{z}\cdot\mathbf{u}_h)(\mathbf{u}\cdot n_E)\,ds\Big].\end{aligned}$$

Die Abschätzung der ersten Terme ist wie bei konformen Methoden, wobei man nutzt, daß $\nabla\mathbf{u}$ und p fast überall stetig sind:

$$\begin{aligned}&\sum_{T\in\mathcal{T}_h}(-\nu\Delta(\mathbf{u}-\mathbf{u}_h) + (\mathbf{u}\cdot\nabla)\mathbf{u} - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h + \nabla(p-p_h),\mathbf{z}-\mathbf{z}_h)_T\\ &= \sum_{T\in\mathcal{T}_h}(\mathbf{f}_h + \nu\Delta\mathbf{u}_h - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h - \nabla p_h,\mathbf{z}-\mathbf{z}_h)_T + (\mathbf{f}-\mathbf{f}_h,\mathbf{z}-\mathbf{z}_h)_T\\ &\overset{(6.3)}{\leq} C_I\sum_{T\in\mathcal{T}_h}h_T^2\,(\|\mathbf{f}_h + \nu\Delta\mathbf{u}_h - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h - \nabla p_h\|_{0,T} + \|\mathbf{f}-\mathbf{f}_h\|_{0,T})\,|\mathbf{z}|_{2,T}\\ &\overset{(7.19)}{\leq} CC_IC_S\left(\sum_{T\in\mathcal{T}_h}h_T^4\|\mathbf{f}_h + \nu\Delta\mathbf{u}_h - (\mathbf{u}_h\cdot\nabla)\mathbf{u}_h - \nabla p_h\|_{0,T}^2 + h_T^4\|\mathbf{f}-\mathbf{f}_h\|_{0,T}^2\right)^{1/2}\|\mathbf{u}-\mathbf{u}_h\|_0,\end{aligned}$$

$$
\begin{aligned}
&\sum_{T\in\mathcal{T}_h}(s-s_h,\nabla\cdot(\mathbf{u}-\mathbf{u}_h))_T\\
&\overset{(7.2)}{\leq}\quad C_I\sum_{T\in\mathcal{T}_h}h_T\|\nabla\cdot\mathbf{u}_h\|_{0,T}|s|_{1,\tilde\omega_T}\\
&\overset{(7.19)}{\leq}\quad CC_IC_S\left(\sum_{T\in\mathcal{T}_h}h_T^2\|\nabla\cdot\mathbf{u}_h\|_{0,T}^2\right)^{1/2}\|\mathbf{u}-\mathbf{u}_h\|_0,
\end{aligned}
$$

$$
\begin{aligned}
&\sum_{T\in\mathcal{T}_h}\sum_{E\subset\partial T}\int_E(\nu\nabla(\mathbf{u}-\mathbf{u}_h)\cdot n_E-(p-p_h)n_E)\cdot(\mathbf{z}-\mathbf{z}_h)ds\\
&=\quad\sum_E\int_E-[\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hn_E]\!]_E\cdot(\mathbf{z}-\mathbf{z}_h)ds\\
&\leq\quad\sum_E\|\,[\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hn_E]\!]_E\,\|_{0,E}\|\mathbf{z}-\mathbf{z}_h\|_{0,E}\\
&\overset{(6.4)}{\leq}\quad C_I\sum_E h_E^{3/2}\|\,[\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hn_E]\!]_E\,\|_{0,E}|\mathbf{z}|_{2,\omega_E}\\
&\overset{(7.21)}{\leq}\quad CC_IC_S\left(\sum_E h_E^3\|\,[\![\nu\nabla\mathbf{u}_h\cdot n_E-p_hn_E]\!]_E\,\|_{0,E}^2\right)^{1/2}\|\mathbf{u}-\mathbf{u}_h\|_0.
\end{aligned}
$$

Die infolge der Nichtkonformität von $\mathbf{u}_h$ sich ergebenden Terme werden wie folgt abgeschätzt:

$$
\begin{aligned}
\sum_{T\in\mathcal{T}_h}\sum_{E\subset\partial T}\int_E\nu(\nabla\mathbf{z}\cdot n_E)\cdot\mathbf{u}_hds\quad&=\quad\sum_E\int_E\nu(\nabla\mathbf{z}\cdot n_E)\cdot[\![\mathbf{u}_h]\!]_E\,ds\\
&\overset{(6.5)}{=}\quad\sum_E\int_E\nu((\nabla\mathbf{z}-\mathcal{M}_E^0(\nabla\mathbf{z}))\cdot n_E)\cdot[\![\mathbf{u}_h]\!]_E\,ds\\
&\overset{(6.6)}{\leq}\quad C\sum_E h_E^{1/2}\|\nu\,[\![\mathbf{u}_h]\!]_E\,\|_{0,E}|\nabla\mathbf{z}\cdot n_E|_{1,\omega_E}\\
&\overset{(7.21)}{\leq}\quad C\,C_S\left(\sum_E h_E\|\nu\,[\![\mathbf{u}_h]\!]_E\,\|_{0,E}^2\right)^{1/2}\|\mathbf{u}-\mathbf{u}_h\|_0,\\
\sum_{T\in\mathcal{T}_h}\sum_{E\subset\partial T}\int_E s(\mathbf{u}_h\cdot n_E)ds\quad&\overset{(6.5)}{=}\quad\sum_E\int_E(s-\mathcal{M}_E^0(s))\,[\![\mathbf{u}_h]\!]_E\cdot n_Eds\\
&\overset{(6.6)}{\leq}\quad C\sum_E h_E^{1/2}\|\,[\![\mathbf{u}_h]\!]_E\cdot n_E\|_{0,E}|s|_{1,\omega_E}\\
&\overset{(7.21)}{\leq}\quad C\,C_S\left(\sum_E h_E\|\,[\![\mathbf{u}_h]\!]_E\,\|_{0,E}^2\right)^{1/2}\|\mathbf{u}-\mathbf{u}_h\|_0,\\
\sum_{T\in\mathcal{T}_h}\sum_{E\subset\partial T}\int_E(\mathbf{z}\cdot\mathbf{u}_h)(\mathbf{u}\cdot n_E)ds\quad&\leq\quad CC_S\left(\sum_E h_E\|\mathbf{u}\cdot n_E\|_{1,\infty,E}^2\|\,[\![u_h]\!]_E\,\|_{0,E}^2\right)^{1/2}\|\mathbf{u}-\mathbf{u}_h\|_0.
\end{aligned}
$$

Die Abschätzung des letzten Terms erfolgt mit den gleichen Argumenten wie die Abschätzung des entsprechenden Terms in Satz 6.6.1. Mit (7.21) folgt

$$
(\mathbf{f}-\hat{\mathbf{f}}_h,\mathbf{z}_h)\leq\|\mathbf{f}-\hat{\mathbf{f}}_h\|_0\,(\|\mathbf{z}-\mathbf{z}_h\|_0+\|\mathbf{z}\|_0)\leq CC_S\|\mathbf{f}-\hat{\mathbf{f}}_h\|_0\|\mathbf{u}-\mathbf{u}_h\|_0.
$$

□

Bemerkung 7.4.2 Kommentare zur globalen Fehlerabschätzung. Die Aussagen der Bemerkungen 6.6.2 bis 6.6.7 lassen sich sinngemäß hierher übertragen. Insbesondere ist der hergeleitete Fehlerschätzer nicht robust. Untersuchungen über die Abhängigkeit der Stabilitätskonstanten C_S von ν findet man beispielsweise bei Johnson, Rannacher und Boman [JRB95]. Die Abschätzung des konvektiven Terms im Zusammenhang mit der Nichtkonformität der Diskretisierung bereitet auch hier die größten Probleme. Aus dieser Abschätzung resultiert die einschränkende Voraussetzung, daß $\|\mathbf{u}\cdot n_E\|_{1,\infty,E}$ für alle Kanten E gleichmäßig in ν beschränkt sein soll. Verzichtet man auf die Gleichmäßigkeit, so erhält man in (7.24) neben C_S noch einen im allgemeinen von ν abhängigen Faktor.

Für die Stokes Gleichungen hat das duale Problem, von den beiden konvektiven Termen auf der linken Seite abgesehen, die gleiche Gestalt wie (7.19). Für die Lösung des dualen Problems werden die gleichen Regularitäts– und Stabilitätseigenschaften wie bei den Navier–Stokes Gleichungen gefordert. Die Stabilitätskonstante hängt nicht mehr von $\mathbf{u}_h$ ab.

Folgerung 7.4.3 Globale a posteriori Fehlerabschätzung für die Stokes Gleichungen und den Fehler der Geschwindigkeit in der L^2–Norm. *Seien die Regularitätsvoraussetzung (7.1) für die exakte Lösung $(\mathbf{u},p)$ von (2.4), die Regularität (7.20) und die Stabilität (7.21) für die Lösung $(\mathbf{z},s)$ des zugehörigen dualen Problems und eine reguläre Triangulierung gegeben. Dann gilt die globale a posteriori Fehlerabschätzung*

$$\begin{aligned}\|\mathbf{u}-\mathbf{u}_h\|_0 &\le CC_IC_S\eta + CC_IC_S\left(\sum_{T\in\mathcal{T}_h} h_T^4\|\mathbf{f}-\mathbf{f}_h\|_{0,T}^2\right)^{1/2} + CC_S\|\mathbf{f}-\hat{\mathbf{f}}_h\|_0\\ &= CC_IC_S\left(\sum_{T\in\mathcal{T}_h}\eta_T^2\right)^{1/2} + CC_IC_S\left(\sum_{T\in\mathcal{T}_h} h_T^4\|\mathbf{f}-\mathbf{f}_h\|_{0,T}^2\right)^{1/2} + CC_S\|\mathbf{f}-\hat{\mathbf{f}}_h\|_0\end{aligned}$$

mit

$$\eta_T^2 = h_T^4\|\mathbf{f}_h\|_{0,T}^2 + h_T^2\|\nabla\cdot\mathbf{u}_h\|_{0,T}^2 + \sum_{E\subset\partial T} h_E^3\|\,[\![\nu\nabla\mathbf{u}_h\cdot n_E - p_hn_E]\!]_E\,\|_{0,E}^2 + h_E\|\,[\![\mathbf{u}_h]\!]_E\,\|_{0,E}^2$$

für alle $(\mathbf{u}_h,p_h)\in V_{h0}\times Q_h$.

Kapitel 8

Numerische Ergebnisse für Konvektions–Diffusions Gleichungen

In diesem Kapitel werden Ergebnisse numerischer Testrechnungen für Konvektions–Diffusions Gleichungen vorgestellt. Zunächst findet ein Vergleich von Upwind Diskretisierungen und nichtkonformen Stromlinien-Diffusions Finite Elemente Methoden statt. Danach wird die Arbeitsweise der im Kapitel 6 hergeleiteten lokalen a posteriori Fehlerschätzer studiert und zum Schluß der globale Fehlerschätzer in der L^2–Norm untersucht. Dabei wird ein Weg zur Approximation der in der Fehlerabschätzung auftretenden Konstanten C_S vorgestellt.

8.1 Vergleich von Upwind und Stromlinien–Diffusions FEM

In diesem Abschnitt werden Ergebnisse numerischer Testrechnungen der in den Kapiteln 1.3 und 1.4 vorgestellten Stabilisierungsmethoden für singulär gestörte Probleme präsentiert. Neben der Verifikation der Fehlerabschätzungen (1.19), (1.20) (1.26), (1.27) und (1.28) steht ein Vergleich der verschiedenen Methoden in bezug auf die Genauigkeit der berechneten Resultate und den Lösungsaufwand im Vordergrund. Der Vergleich findet mit Blick auf eine weitere Verwendung zur Lösung von Navier–Stokes Gleichungen mit unstrukturierten Gittern auf Parallelrechnern statt.

In den Tabellen der folgenden Abschnitte werden Fehler in verschiedenen Normen angegeben, die mittels numerischer Integration approximiert wurden. Zur Berechnung der Fehler in der L^2–Norm verwenden wir die Eckpunkt–Schwerpunkt–Seitenmittelpunkt–Regel, siehe Ciarlet [Cia91, (25.19)], und zur Berechnung des Fehlers in der diskreten H^1–Seminorm die Seitenmittelpunkt–Regel, [Cia91, (25.17)]. Die Eckpunkt–Schwerpunkt–Seitenmittelpunkt–Regel ist exakt für Polynome dritten Grades und die Seitenmittelpunkt–Regel für Polynome zweiten Grades. Bei Verwendung nichtkonformer Finiter Elemente kann die Seitenmittelpunkt–Regel zur Berechnung des Fehlers in der L^2–Norm zu entstellenden Ergebnissen führen, siehe Tabelle 8.16. Desweiteren wird eine Stromlinien–Diffusionsnorm eingeführt:

$$\|v\|_S := \left(\nu|v|_h^2 + c_0\|v\|_0^2 + \sum_{T\in\mathcal{T}_h} \delta_T\|b\cdot\nabla v\|_{0,T}^2\right)^{1/2} .$$

Dabei ist δ_T der Stromlinien–Diffusionsparameter, und c_0 ist in (1.6) definiert.

Es werden auch einige Ergebnisse von Testrechnungen mit konformen linearen Finiten Elementen angegeben, die zu einer besseren Einordnung der entsprechenden nichtkonformen Ergebnisse dienen.

Alle Beispiele in diesem Abschnitt sind auf dem Einheitsquadrat definiert. Die verwendeten Grobgitter (Level 0) sind in Abbildung 8.1 und die Anzahl der Unbekannten bei gleichmäßiger Verfeinerung in Tabelle 8.1 zu sehen.

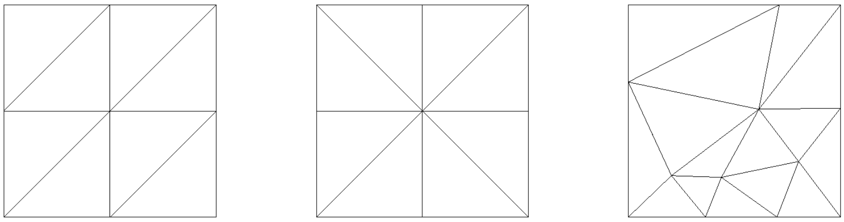


Abbildung 8.1: Die verwendeten Grobgitter, Gitter 1, 2, 3.

Level	Gitter 1, 2		Gitter 3	
	n.konf.	konf.	n.konf.	konf.
0	8	1	18	4
4	3008	961	5688	1849
5	12160	3969	22896	7537
6	48896	16129	91872	30433
7	196096	65025	368064	122305

Tabelle 8.1: Die Anzahl der Unbekannten der Gitter von Abbildung 8.1.

Wir studieren zunächst die Konvergenzordnungen für festes ν und eine glatte Lösung.

Beispiel 8.1.1 Eine glatte polynomiale Lösung. Die rechte Seite f wird so gewählt, daß

$$u(x,y) = 100(1-x)^2 x^2 y(1-2y)(1-y)$$

die exakte Lösung von

$$\begin{aligned} -\nu\Delta u + (3,2)\cdot\nabla u + 2u &= f \quad \text{in } (0,1)\times(0,1) \\ u &= 0 \quad \text{auf } \partial\Omega \end{aligned}$$

ist, siehe Abbildung 8.2. Wir haben $\nu = 10^{-6}$ gewählt und die Testrechnungen auf Gitter 1 durchgeführt. In Übereinstimmung mit Bemerkung 1.4.4 ist $\delta_T = h_T$.

Lev.	konf. Samar.	n.konf. scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	4.153-2	2.500-2	2.500-2	3.951-3	1.281-3	1.399-3	6.586-3	1.404-3
5	2.132-2	1.263-2	1.263-2	1.272-3	3.220-4	3.626-4	3.217-3	3.605-4
6	1.080-2	6.348-3	6.347-3	3.692-4	8.138-5	9.339-5	1.611-3	9.213-5
7	5.435-3	3.181-3	3.181-3	8.694-5	2.046-5	2.369-5	8.102-4	2.326-5
Ord.	0.991	0.997	0.997	2.086	1.992	1.979	0.992	1.986

Tabelle 8.2: Beispiel 8.1.1, Fehler und Konvergenzordnung für $\|u - u_h\|_0$.

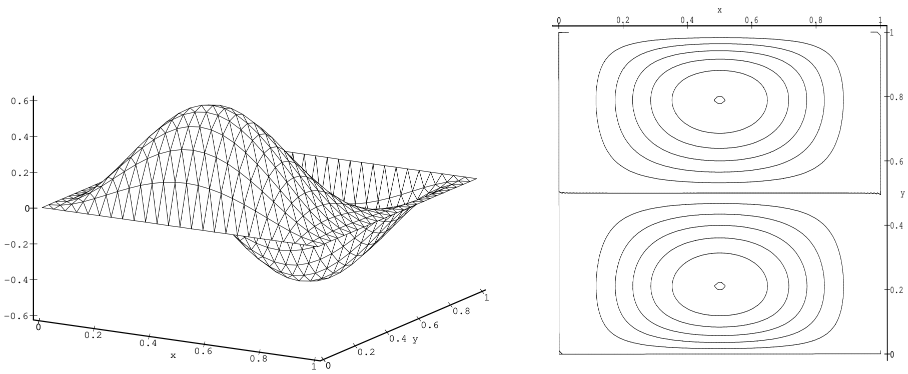


Abbildung 8.2: Beispiel 8.1.1, exakte Lösung und Isolinien.

Lev.	konf. Samar.	n.konf. scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	5.023-1	1.789+0	1.789+0	5.624-1	1.273-1	1.327-1	9.299-1	1.364-1
5	3.285-1	1.786+0	1.786+0	3.578-1	6.450-2	6.696-2	9.194-1	6.883-2
6	2.214-1	1.784+0	1.784+0	2.056-1	3.251-2	3.373-2	9.157-1	3.466-2
7	1.523-1	1.782+0	1.783+0	9.372-2	1.632-2	1.694-2	9.140-1	1.741-2
Ord.	0.540	0.001	0.001	1.133	0.994	0.993	0.003	0.993

Tabelle 8.3: Beispiel 8.1.1, Fehler und Konvergenzordnung für $|u - u_h|_h$.

Lev.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	3.569-2	3.529-2	4.001-2	3.918-2	4.027-2
5	1.258-2	1.246-2	1.499-2	1.621-2	1.511-2
6	4.430-3	4.402-3	5.528-3	7.996-3	5.582-3
7	1.560-3	1.555-3	2.007-3	4.758-3	2.028-3
Ord.	1.505	1.501	1.462	0.749	1.461

Tabelle 8.4: Beispiel 8.1.1, Fehler und Konvergenzordnung für $\|u - u_h\|_S$.

Die Konvergenzordnungen wurden unter Verwendung der Ergebnisse von Level 6 und 7 berechnet. Da das zugrunde liegende Gitter dreidirektional ist, gilt für die Upwind Methoden (konform und nichtkonform) die Fehlerabschätzung (1.20), das heißt insbesondere $\|u - u_h\|_0 \leq Ch$. In Tabelle 8.2 ist diese Konvergenzordnung für alle Upwind Methoden zu erkennen. Ein Nachteil nichtkonformer Upwind Diskretisierungen zeigt sich in der Stagnation des Fehlers $|u - u_h|_h$, Tabelle 8.3. Ein Unterschied zwischen scharfem und Samarskij Upwind ist nicht zu erkennen.

Die NSDFEM zeigen in allen Normen die erwartete Konvergenzordnung, es ist sogar eine Superkonvergenz zu beobachten. Dieses Phänomen ist bei konformen SDFEM gut bekannt und wurde intensiv von Zhou [Zho95] untersucht. Desweiteren ist zu sehen, daß optimale Konvergenzordnungen bereits mit einem geringeren Einfluß des Sprungterms erzielt werden, als es die bisher vorliegenden theoretischen Ergebnisse erwarten lassen. So ist für NSD1 die Wahl $\sigma_E = 1$ und für NSD2 die Wahl von $\sigma_E = h_E^{-1}$ hinreichend. Vernachlässigt man die Sprungterme, also $\sigma_E = 0$, so sind die Ergebnisse deutlich schlechter als mit ihnen. Die Konvergenzordnung ist zwar hoch, was aber an der abnehmenden Bedeutung der Stabilisierung für fortschreitende Verfeinerung liegt, bis sie für $h < \nu$ nicht mehr notwendig ist.

Lev.	konf. Samar.	n.konf. scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	1	1	1	351	11	101	8	104
5	1	1	1	469	10	244	10	248
6	1	1	1	457	13	632	13	639
7	1	1	1	372	19	1738	19	1752

Tabelle 8.5: Beispiel 8.1.1, Anzahl der Mehrgitteriterationen.

Die diskreten Systeme wurden alle mit Mehrgitterverfahren gelöst, siehe Abschnitt 3.3 und 3.6. Als Glätter wurde jeweils ILU_β verwendet, wobei die Struktur der unvollständigen Zerlegung mit dem Besetztheitsmuster der Matrix A übereinstimmt, vergleiche Abbildung 1.3. Deshalb ist ein Glättungsschritt für die NSDFEM kostenintensiver als für die Upwind Methoden. Die Parameter im Mehrgitterzyklus wurden für alle Testrechnungen in diesem Abschnitt gleich gewählt. Die wichtigsten sind

$$\gamma = 1 \text{ (V–Zyklus) },\ \nu_1 = \nu_2 = 3,\ \beta = 1.0.$$

Beim Glätten wurde nicht gedämpft. In Tabelle 8.5 ist die Anzahl der Mehrgitteriterationen zu sehen, die notwendig ist, um eine euklidische Norm des Residuums kleiner als 10^{-8} zu erzielen. Der Aufwand bei den Upwind Methoden ist sehr gering, während bei den NSDFEM lediglich die Wahl $\sigma_E = 1$ die Lösung des Gleichungssystems mit vertretbarem Aufwand gestattet. Bei der Wahl $\sigma_E = h_E^{-1}$ ist die Abhängigkeit der Iterationszahlen von h_E deutlich erkennbar.

Beispiel 8.1.2 Eine Lösung mit moderater innerer Grenzschicht. Die rechte Seite f wird so gewählt, daß

$$u(x,y) = 16x(1-x)y(1-y)\left[\frac{1}{2} + \frac{\arctan\left(200(r_0^2 - (x-x_0)^2 - (y-y_0)^2)\right)}{\pi}\right]$$

mit $r_0 = 0.25$ und $x_0 = y_0 = 0.5$ die exakte Lösung von

$$\begin{aligned} -\nu\Delta u + (2,3)\cdot\nabla u + 2u &= f \quad \text{in } (0,1)\times(0,1) \\ u &= 0 \quad \text{auf } \partial\Omega \end{aligned}$$

ist, siehe Abbildung 8.3. Die Testrechnungen wurden für $\nu = 10^{-4}$ auf den Gittern 2 (Tabellen 8.6 - 8.9) und 3 (Tabellen 8.10 - 8.13) durchgeführt. In Übereinstimmung mit Bemerkung 1.4.4 wurde $\delta_T = h_T$ gewählt.

Außer für Tabelle 8.10 wurden die Konvergenzordnungen mit Hilfe der Ergebnisse auf den Leveln 6 und 7 berechnet. Bei den Ergebnissen in Tabelle 8.10 schwankt die Konvergenzordnung zwischen den einzelnen Leveln beträchtlich. Deshalb wird dort eine gemittelte Konvergenzordnung mit den Werten von Level 5 und 7 angegeben. Bei den Upwind Methoden ist zu erkennen, daß die Konvergenzordnungen für $\|u - u_h\|_0$ mit jeder Verfeinerung steigen, Tabellen 8.6 und 8.10. Die höhere Anzahl der Unbekannten bei der nichtkonformen Diskretisierung liefert einen

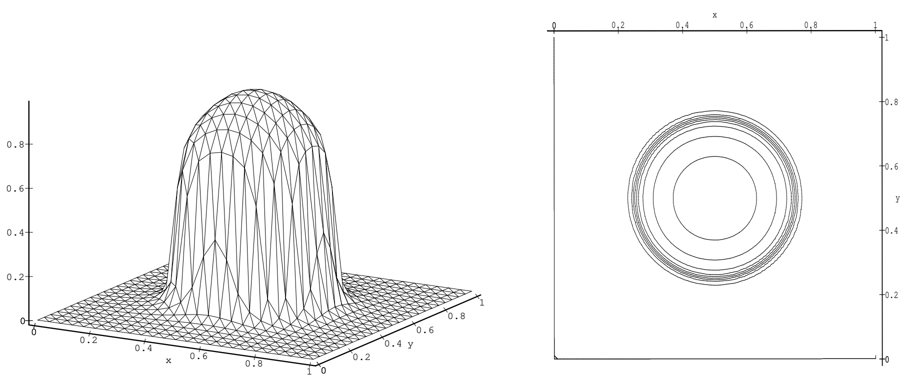


Abbildung 8.3: Beispiel 8.1.2, exakte Lösung und Isolinien.

Lev.	konf. Samar.	n.konf. scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	9.463-2	7.404-2	7.402-2	6.807-1	3.105-2	3.190-2	4.230-2	3.212-2
5	5.474-2	4.632-2	4.629-2	1.771-1	9.998-3	1.535-2	1.928-2	1.543-2
6	3.333-2	2.845-2	2.840-2	3.015-2	2.884-3	7.241-3	9.789-3	7.269-3
7	1.984-2	1.704-2	1.698-2	4.404-3	7.893-4	2.768-3	5.214-3	2.780-3
Ord.	0.750	0.739	0.743	2.775	1.869	1.387	0.909	1.387

Tabelle 8.6: Beispiel 8.1.2, Gitter 2, Fehler und Konvergenzordnung für $\|u - u_h\|_0$.

Lev.	konf. Samar.	n.konf. scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	4.183+0	3.931+0	3.940+0	6.399+1	2.493+0	1.799+0	3.410+0	1.809+0
5	2.712+0	3.599+0	3.618+0	2.525+1	1.237+0	1.307+0	2.412+0	1.310+0
6	1.789+0	3.389+0	3.428+0	6.314+0	5.769-1	8.397-1	1.974+0	8.405-1
7	1.194+0	3.174+0	3.248+0	1.174+0	2.589-1	4.594-1	1.792+0	4.600-1
Ord.	0.589	0.095	0.078	2.427	1.156	0.870	0.140	0.870

Tabelle 8.7: Beispiel 8.1.2, Gitter 2, Fehler und Konvergenzordnung für $|u - u_h|_h$.

etwas kleineren Fehler als die konforme Diskretisierung. In bezug auf den Fehler $|u - u_h|_h$, Tabellen 8.7 und 8.11, ist die konforme Diskretisierung wiederum deutlich besser. Die Unterschiede zwischen scharfem und Samarskij Upwind sind minimal.

Auch in diesem Beispiel liefern die NSDFEM wesentlich bessere Ergebnisse als die Upwind Methoden. Dies gilt sowohl für den Fehler in der L^2–Norm, Tabellen 8.6, 8.10, als auch der diskreten H^1–Seminorm, Tabellen 8.7, 8.11. Die theoretisch vorhergesagten Konvergenzordnungen sind in der Mehrzahl der Tests zu erkennen oder werden sogar übertroffen. Die schlechtesten Ergebnisse erhält man abermals bei Vernachlässigung der Sprungterme in der Diskretisierung, siehe vor allem Abbildung 8.4. Innerhalb der Grenzschichten sind sehr starke Oszillationen zu sehen, die bei Verwendung der Sprungterme deutlich gedämpft werden (man beachte die Skalierung).

Lev.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	1.267+0	4.666-1	5.153-1	4.728-1	5.155-1
5	4.113-1	2.020-1	2.551-1	2.089-1	2.551-1
6	1.073-1	7.544-2	1.779-1	8.339-2	1.179-1
7	3.019-2	2.713-2	5.125-2	3.680-2	5.125-2
Ord.	1.830	1.476	1.202	1.180	1.202

Tabelle 8.8: Beispiel 8.1.2, Gitter 2, Fehler und Konvergenzordnung für $\|u - u_h\|_S$.

Lev.	konf. Samar.	n.konf scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	1	1	1	647	37	138	19	140
5	1	1	1	443	44	342	21	346
6	1	2	2	209	46	832	22	836
7	1	3	3	130	45	2235	23	2243

Tabelle 8.9: Beispiel 8.1.2, Gitter 2, Anzahl der Mehrgitteriterationen.

Lev.	konf. Samar.	n.konf scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	7.133-2	5.798-2	5.793-2	4.182-1	2.714-2	2.275-2	3.266-2	2.287-2
5	3.946-2	2.032-2	2.027-2	-	3.541-3	4.849-3	6.243-3	4.861-3
6	2.219-2	1.852-2	1.841-2	-	1.487-3	2.720-3	4.036-3	2.732-3
7	1.241-2	6.459-3	6.374-3	-	2.601-4	5.005-4	1.161-3	5.030-4
Ord.	0.834	0.827	0.834	-	1.884	1.638	1.213	1.636

Tabelle 8.10: Beispiel 8.1.2, Gitter 3, Fehler und Konvergenzordnung für $\|u - u_h\|_0$.

Lev.	konf. Samar.	n.konf scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	3.300+0	3.277+0	3.286+0	4.472+1	2.574+0	1.615+0	3.291+0	1.626+0
5	2.162+0	2.800+0	2.817+0	-	9.277-1	9.148-1	1.949+0	9.154-1
6	1.357+0	2.557+0	2.590+0	-	4.039-1	5.101-1	1.558+0	5.105-1
7	8.451-1	2.356+0	2.418+0	-	1.813-1	2.541-1	1.430+0	2.543-1
Ord.	0.683	0.118	0.099	-	1.155	1.005	0.123	1.005

Tabelle 8.11: Beispiel 8.1.2, Gitter 3, Fehler und Konvergenzordnung für $|u - u_h|_h$.

Auf Gitter 3 divergiert das Mehrgitterverfahren sogar bei Vernachlässigung der Sprungterme.

Die Parameter zur Lösung der diskreten Systeme wurden wie im Beispiel 8.1.1 gewählt. Die dortigen Ausführungen über das Lösungsverhalten lassen sich hierher übertragen.

Lev.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	8.430-1	3.850-1	4.378-1	3.884-1	4.380-1
5	-	1.585-1	2.091-1	1.615-1	2.091-1
6	-	5.919-2	9.116-2	6.340-2	9.116-2
7	-	2.120-2	3.614-2	2.767-2	3.613-2
Ord.	-	1.481	1.335	1.196	1.335

Tabelle 8.12: Beispiel 8.1.2, Gitter 3, Fehler und Konvergenzordnung für $\|u - u_h\|_S$.

Lev.	konf. Samar.	n.konf. scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	3	3	3	837	66	308	49	311
5	3	5	5	div.	70	713	48	716
6	4	7	7		76	1526	48	1529
7	5	12	11		72	3657	61	3669

Tabelle 8.13: Beispiel 8.1.2, Gitter 3, Anzahl der Mehrgitteriterationen.

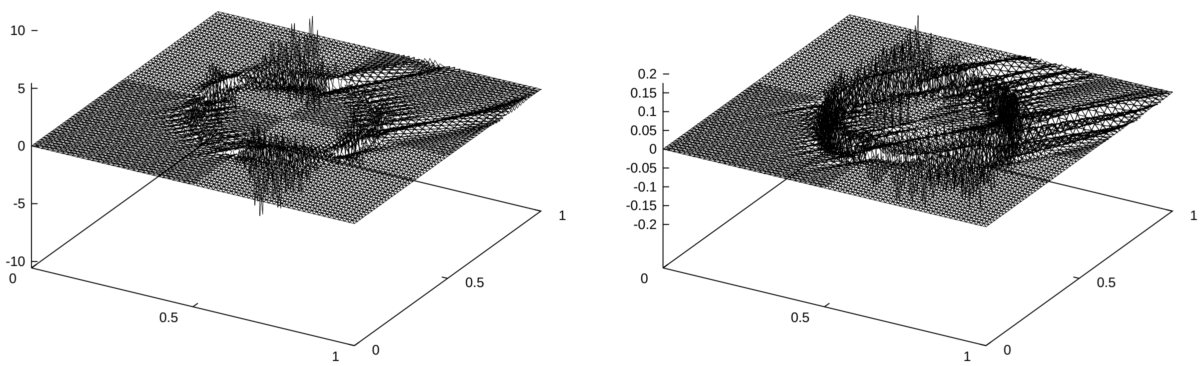


Abbildung 8.4: Beispiel 8.1.2, nichtkonformer Fehler ohne (links) und mit (rechts) Nutzung der Sprungterme, NSD1.

Dieses Beispiel zeigt, daß man mit NSDFEM auch auf unstrukturierten Gittern optimale Konvergenzordnungen erzielen kann. Wie im Beispiel 8.1.1 sind die NSDFEM bezüglich der Genauigkeit der Lösung den Upwind Methoden überlegen. Das mag zum einen an der relativ dicken Grenzschicht und zum anderen am relativ großen Parameter ν liegen. Der Lösungsaufwand ist bei den NSDFEM abermals deutlich höher.

Beispiel 8.1.3 Eine Lösung mit Grenzschichten am Ausflußrand. Die rechte Seite f und die Randbedingung g werden so gewählt, daß

$$u(x,y,\nu) = xy^2 - y^2 \exp\left(\frac{2(x-1)}{\nu}\right) - x\exp\left(\frac{3(y-1)}{\nu}\right) + \exp\left(\frac{2(x-1)+3(y-1)}{\nu}\right)$$

exakte Lösung von

$$\begin{aligned} -\nu\Delta u + (2,3)\cdot\nabla u + u &= f \quad \text{in } (0,1)\times(0,1) \\ u &= g \quad \text{auf } \partial\Omega \end{aligned}$$

ist, siehe Abbildung 8.5. Die Testrechnungen wurden für $\nu = 10^{-6}$ auf Gitter 1 ausgeführt. Bei diesem Beispiel wurde $\delta_T = 0.25h_T$ gewählt. Die Lösung verschwindet auf zwei Kanten von Ω und ist auf den anderen zwei Seiten betragsmäßig sehr klein.

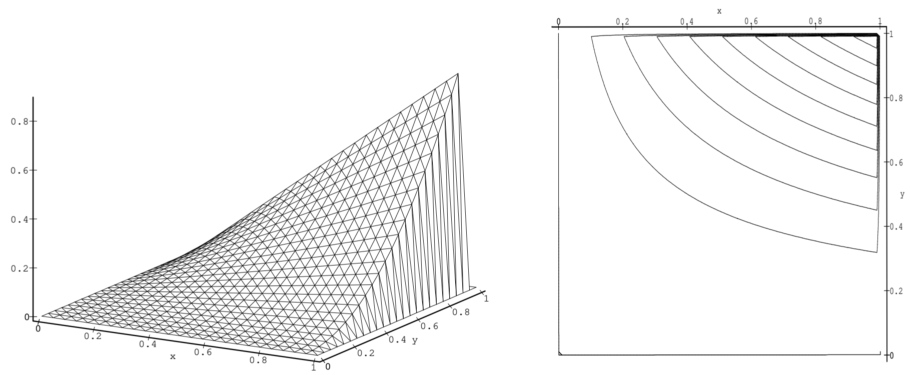


Abbildung 8.5: Beispiel 8.1.3, exakte Lösung und Isolinien.

Lev.	konf. Samar.	n.konf. scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	5.112-2	3.627-2	3.627-2	1.698+0	6.783-2	7.663-2	7.231-2	7.697-2
5	3.952-2	2.533-2	2.534-2	2.292+1	4.822-2	5.481-2	5.153-2	5.494-2
6	2.810-2	1.780-2	1.780-2	2.700+1	3.418-2	3.898-2	3.657-2	3.903-2
7	1.992-2	1.254-2	1.254-2	2.397+1	2.420-2	2.765-2	2.591-2	2.766-2
Ord.	0.498	0.510	0.510	0.170	0.499	0.499	0.499	0.494

Tabelle 8.14: Beispiel 8.1.3, Fehler und Konvergenzordnung für $\|u - u_h\|_0$ in Ω.

Die Konvergenzordnungen wurden mit den Fehlern auf den Leveln 6 und 7 berechnet. Da für alle Verfeinerungen die Breite der Grenzschichten kleiner als h ist, kann man auf Grund des Interpolationsfehlers höchstens eine Konvergenzordnung von 0.5 für $\|u - u_h\|_0$ erwarten. Diese wird von allen Upwind Methoden und NSDFEM mit Sprungterm erreicht, Tabelle 8.14. Den quantitativ kleinsten Fehler liefern hier die nichtkonformen Upwind Methoden. In Abbildung 8.6 ist zu sehen, daß die Upwind Methoden Lösungen mit scharfen Grenzschichten berechnen, währenddessen die NSDFEM die Grenzschichten verschmieren. Für die graphische Darstellung wurde die Lösung auf eine konforme Funktion projiziert. Untersucht man den Fehler außerhalb

Lev.	konf. Samar.	n.konf scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	8.562-3	5.558-3	5.558-3	6.315-2	2.368-3	2.257-3	2.762-3	2.267-3
5	4.250-3	2.865-3	2.865-3	4.672-3	9.300-5	7.995-5	7.211-4	8.277-5
6	2.172-3	1.436-3	1.436-3	1.210-5	6.735-6	7.378-6	3.589-4	9.886-6
7	1.084-3	7.188-4	7.188-4	1.645-6	1.693-6	1.862-6	1.788-4	2.499-6
Ord.	1.007	0.998	0.998	2.874	1.996	1.988	1.004	1.984

Tabelle 8.15: Beispiel 8.1.3, Fehler und Konvergenzordnung für $\|u - u_h\|_0$ in $0 \le x, y \le 0.9$.

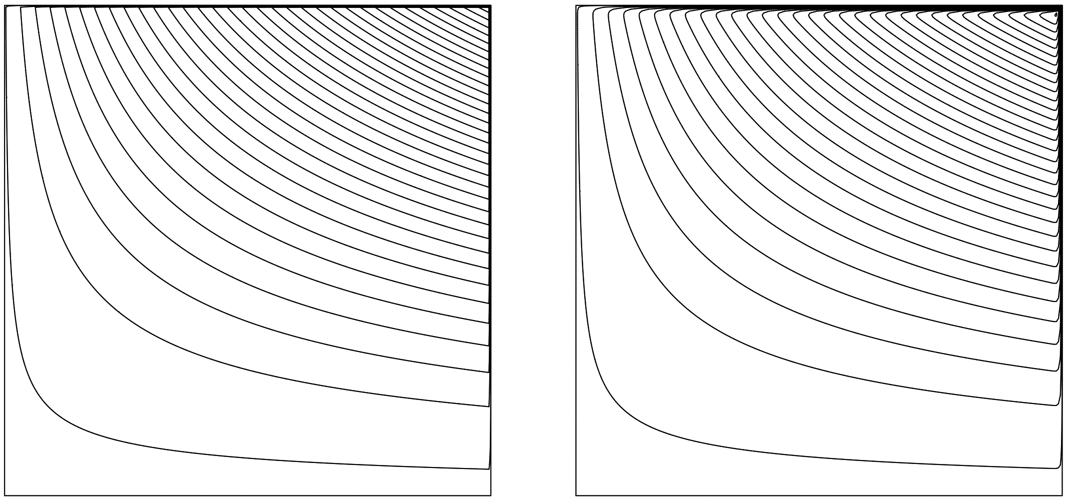


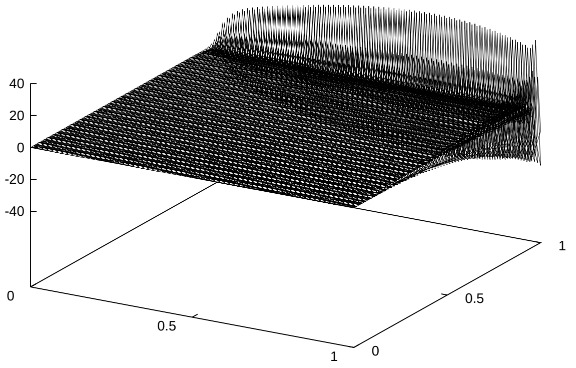
Abbildung 8.6: Beispiel 8.1.3, Lösung mit nichtkonformer Upwind Methode (links) und NSD1 (rechts).

der Grenzschichten, etwa in $0 \le x, y \le 0.9$, so erkennt man, daß hier die NSDFEM Lösungen mit hoher Genauigkeit erzielen und die optimalen Konvergenzordnungen in diesen Teilgebieten erreicht und sogar übertroffen werden, Tabelle 8.15. Auch bei diesem Beispiel führt die Vernachlässigung der Sprungterme in der NSDFEM zu sehr großen Oszillationen innerhalb der Grenzschichten, siehe Abbildung 8.7 (man beachte die Skalierung).

Level	SMP	ESSMP
4	7.815-3	3.627-2
5	3.925-3	2.534-2
6	1.966-3	1.780-2
7	9.837-4	1.254-2
Ord.	0.999	0.510

Tabelle 8.16: Beispiel 8.1.3, Berechnung des Fehlers $\|u - u_h\|_0$ mit der Seitenmittelpunkt–Regel und der Eckpunkt–Schwerpunkt–Seitenmittelpunkt–Regel.

In Tabelle 8.16 wird veranschaulicht, daß die Verwendung der Seitenmittelpunkt–Regel zur Berechnung des Fehlers $\|u - u_h\|_0$ bei nichtkonformen Methoden zu vollkommen falschen Er-



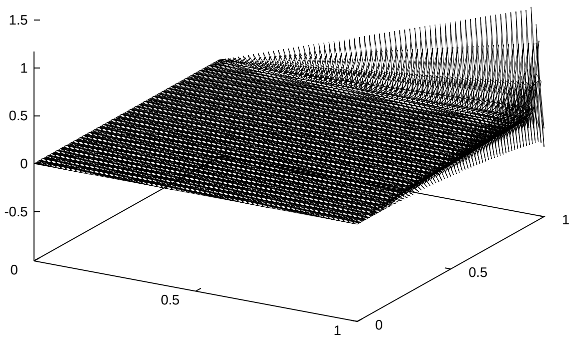


Abbildung 8.7: Beispiel 8.1.3, nichtkonformer Fehler ohne (links) und mit (rechts) Nutzung der Sprungterme, NSD1.

gebnissen führen kann. Selbst wenn die Funktionswerte der numerischen Lösung u_h auf den Seitenmittelpunkten sehr genau sind, können die Funktionswerte von u_h an den Eckpunkten der Dreiecke sehr weit von denen von u abweichen. Deshalb benötigt man eine Integrationsformel, die sowohl die Eckpunkte als auch die Seitenmittelpunkte berücksichtigt.

Lev.	konf. Samar.	n.konf. scharf	n.konf. Samar.	NSD1 $\sigma_E = 0$	NSD1 $\sigma_E = 1$	NSD1 $\sigma_E = h_E^{-1}$	NSD2 $\sigma_E = 1$	NSD2 $\sigma_E = h_E^{-1}$
4	1	1	1	283	6	75	5	76
5	1	1	1	473	7	169	7	171
6	1	1	1	646	9	445	10	449
7	1	1	1	582	9	992	15	1029

Tabelle 8.17: Beispiel 8.1.3, Anzahl der Mehrgitteriterationen.

Die Parameter zur Lösung des diskreten Systems wurden wie im Beispiel 8.1.1 gewählt. Bezüglich des Lösungsverhaltens können die Ausführungen von dort übernommen werden.

In diesem Beispiel sind die Upwind Methoden, sowohl im Hinblick auf die Genauigkeit als auch auf den numerischen Aufwand, den NSDFEM überlegen.

Bemerkung 8.1.4 Auswertung der numerischen Tests.

1. **konform – nichtkonform.** Die nichtkonformen FEM haben auf den gleichen Gittern etwa dreimal soviel Unbekannte wie die konformen. Betrachtet man die nichtkonformen Upwind Methoden, so ist der Fehler $\|u - u_h\|_0$ etwa um den Faktor 1.1 - 2 geringer als bei den konformen Methoden, währenddessen der Fehler $|u - u_h|_h$ größer ist und stagniert. Es zeigen sich bei Konvektions–Diffusions Gleichungen wenig Vorteile bei der Verwendung nichtkonformer FEM. Wegen des Fehlers $|u - u_h|_h$ sind konforme FEM sogar vorzuziehen.

2. **Upwind Methoden.** Zwischen den Ergebnissen mit scharfem und Samarskij Upwind sind bei keinem Beispiel signifikante Unterschiede festgestellt worden.

3. **NSDFEM.** Die NSDFEM weisen ähnliche Eigenschaften wie die konformen Stromlinien–Diffusions FEM auf. Sie kombinieren gute Stabilität und hohe Genauigkeit außerhalb von Grenzschichten. Grenzschichten werden verschmiert.

   Die Sprungterme sind notwendig um gute Lösungen ohne Oszillationen in Grenzschichten zu erhalten. Alle numerischen Tests legen nahe, daß man optimale Konvergenzordnungen auch mit einem geringeren Einfluß der Sprungterme erreichen kann, als es die bisherigen theoretischen Ergebnisse aussagen. Für NSD1 scheint $\sigma_E = 1$ und für NSD2 $\sigma_E = h_E^{-1}$ zu reichen.

4. **Qualität der Lösungen.** NSDFEM haben sich bei glatten Lösungen und Lösungen mit moderaten Grenzschichten als den Upwind Methoden überlegen herausgestellt. Scharfe Grenzschichten werden von den Upwind Methoden besser aufgelöst.

5. **Numerischer Aufwand.** Der numerische Aufwand zur Lösung der diskreten Systeme ist bei den NSDFEM ein Vielfaches im Vergleich zu den Upwind Methoden.

6. **Parallelisierung.** Eine Motivation zur Verwendung nichtkonformer FEM ist der geringe Kommunikationsaufwand infolge des Nichtauftretens von Crosspoints, Bemerkung 4.1.3. Die Koppelung von Basisfunktionen, deren gemeinsamer Träger nur eine Kante ist, macht bei den NSDFEM diesen Vorteil zunichte.

7. **Adaptivität.** Auch die Verwendung adaptiver Verfahren gestaltet sich bei den NSDFEM komplizierter als bei Upwind Methoden. Wählt man die in Abschnitt 3.3 beschriebene Herangehensweise, so reicht das bisher vom Programm bereitgestellte lokal verfeinerte Gitter nicht aus, um alle sich ändernden Residuenwerte nach dem Glätten zu berechnen.

Fazit: Legt man die Qualität der Lösung und den numerischen Aufwand als Bewertungsmaßstab zugrunde, dann ist NSD1 mit $\sigma_E = 1$ die einzig praktisch sinnvolle NSDFEM. Zur Übertragung auf Navier–Stokes Gleichungen halten wir Upwind Methoden, vor allem auf Grund der Punkte 5 und 6, für geeigneter.

8.2 Adaptive Methoden

Dieser Abschnitt beschäftigt sich mit numerischen Ergebnissen, die unter Verwendung lokal verfeinerter Netze erzielt worden sind. Dabei werden die Vorteile der lokal verfeinerten Gitter im Vergleich zu gleichmäßig verfeinerten Netzen in bezug auf die Genauigkeit der Lösung und die Anzahl der benötigten Unbekannten deutlich. Es werden Resultate miteinander verglichen, die mit Hilfe der im Abschnitt 6.4 hergeleiteten a posteriori Fehlerschätzer $\eta_{T,1}$ und $\eta_{T,2}$ für den lokalen Fehler in der diskreten H^1–Norm sowie der im Abschnitt 6.5 hergeleiteten a posteriori Fehlerschätzer $\tilde{\eta}_{T,1}$ und $\tilde{\eta}_{T,2}$ für den lokalen Fehler in der L^2–Norm erzielt wurden. Die Fehlerschätzer wurden auch auf ein Problem angewandt, welches die in diesen Abschnitten gemachte Regularitätsvoraussetzung (6.1) nicht erfüllt.

Unter einem *Schritt* verstehen wir die Befehlsreihenfolge

$$\text{Assemblieren} \to \text{Lösen} \to \text{Fehlerschätzen} \to \text{Verfeinern} \to \text{Interpolieren}.$$

Beispiel 8.2.1 Eine Lösung mit Grenzschichten am Ausflußrand. Wir verwenden das Problem aus Beispiel 8.1.3. Eine Verbesserung der mit uniformer Verfeinerung erzielten Lösung ist hier nur mit einer besseren Auflösung der Grenzschichten zu erzielen. Deswegen muß man von einem Fehlerschätzer erwarten, daß er Gitter erzeugt, bei denen vor allem im Bereich dieser Grenzschichten verfeinert wird.

Für die in Tabelle 8.18 dokumentierten Tests wurden folgende Parameter gewählt (vergleiche Abschnitt 6.7):

Markierungsmethode = Mittlerer_lokaler_Fehler
REFTOL = 2.0
COARSETOL = 0.0
MAXLEVEL = 99
BASELEVEL = 3
MINREF = 0.1 (= 10 %).

Die Ergebnisse bei Verwendung von $\eta_{T,2}$ und $\tilde{\eta}_{T,2}$ unterschieden sich kaum von denen mit $\eta_{T,1}$ beziehungsweise $\tilde{\eta}_{T,1}$.

	$\eta_{T,1}$			$\tilde{\eta}_{T,1}$		
Schritt	Unbekannte	$\|\|u-u_h\|\|_0$	$\|u-u_h\|_h$	Unbekannte	$\|\|u-u_h\|\|_0$	$\|u-u_h\|_h$
6	2798	5.036-2	4.975+4	2650	5.029-2	4.996+4
9	11521	1.725-2	1.853+4	9842	2.465-2	2.553+4
12	51544	5.008-3	8.390+3	33101	1.099-2	1.460+4
13	88310	3.272-3	6.398+3	48198	1.223-2	1.281+4
15				101781	8.582-3	9.116+3
unif. 7	196096	1.254-2	3.516+4			

Tabelle 8.18: Beispiel 8.2.1, Ergebnisse bei Verwendung lokal verfeinerter Gitter mit verschiedenen Fehlerschätzern.

Die mit den Fehlerschätzern erzielten Ergebnisse sind deutlich besser als mit uniformer Verfeinerung. So würde man bei gleichmäßiger Verfeinerung und einer Konvergenzordnung von 0.5 rund 50 Millionen Unbekannte brauchen, um ein ähnlich genaues Ergebnis bezüglich des Fehlers in der L^2–Norm zu erzielen, wie bei Verwendung von $\eta_{T,1}$ und 88 000 Unbekannten (Schritt 13). Zwischen der Wirkungsweise der Fehlerschätzer gibt es jedoch auch Unterschiede. Der Fehlerschätzer $\eta_{T,1}$ verfeinert nur innerhalb der Grenzschichten, siehe Abbildung 8.8. Nach jedem Verfeinerungsschritt verringert sich der Fehler sowohl in der L^2–Norm als auch in der diskreten H^1–Norm. Der Fehlerschätzer $\tilde{\eta}_{T,1}$ verfeinert zwar auch vorwiegend innerhalb der Grenzschichten, doch ist die Verfeinerungstiefe geringer als bei dem mit $\eta_{T,1}$ erzeugten Gitter. Die Fehler in beiden Normen sind auf dem von $\tilde{\eta}_{T,1}$ erzeugten Gitter größer. In den Verfeinerungsschritten, in denen $\tilde{\eta}_{T,1}$ die Verfeinerungstiefe innerhalb der Grenzschichten nicht vergrößert, verringert sich der Fehler in der L^2–Norm nicht, siehe Tabelle 8.18, von Schritt 12 zu 13.

Beispiel 8.2.2 Eine Lösung mit geringer Regularität. Wir betrachten das Problem

$$\begin{array}{rcll} -10^{-3}\Delta u + (1,0)\cdot\nabla u &=& 0 & \text{in } (0,1)\times(0,1) \\ u &=& 0 & \text{für } y=0, 0\le x\le 1 \\ u &=& 0 & \text{für } y=1, 0\le x\le 1 \\ u &=& 0 & \text{für } x=0, |y-0.5|>0.05 \\ u &=& 1 & \text{für } x=0, |y-0.5|<0.05 \\ \nu\partial u/\partial n &=& 0 & \text{für } x=1, 0\le y\le 1. \end{array} \tag{8.1}$$

Die Lösung von (8.1) läßt sich als Fourierreihe

$$u(x,y) = \sum_{k=1}^{\infty} \varphi_k(x)\chi_k(y) = \sum_{k=1}^{\infty} (\theta_k\varphi_{1k}(x) + \varrho_k\varphi_{2k}(x))\chi_k(y),$$

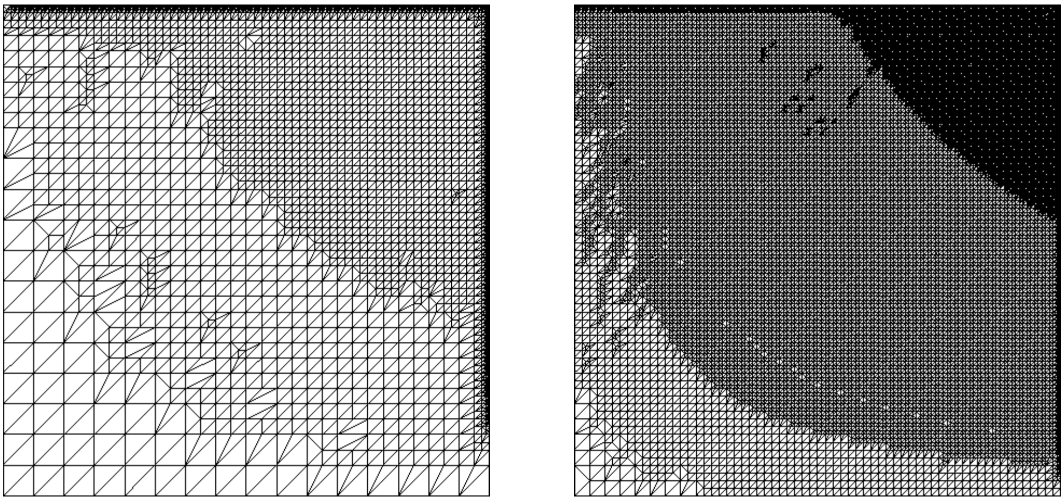


Abbildung 8.8: Beispiel 8.2.1, lokal verfeinerte Gitter, links mit $\eta_{T,1}$ (Schritt 13), rechts mit $\tilde{\eta}_{T,1}$ (Schritt 15) erzeugt.

mit

$$\varphi_{1k}(x) = e^{x/(2\nu)} \frac{\sinh(\zeta_k(1-x))}{\sinh(\zeta_k)}, \quad \varphi_{2k}(x) = e^{(x-1)/(2\nu)} \frac{\sinh(\zeta_k x)}{\sinh(\zeta_k)}$$

darstellen, wobei

$$\zeta_k = \sqrt{\frac{1}{(2\nu)^2} + (k\pi)^2}$$

ist. Die Konstanten sind

$$\theta_k = \frac{2}{k\pi}(\cos(0.45k\pi) - \cos(0.55k\pi)), \quad \varrho_k = \theta_k \left(\frac{\zeta_k e^{1/(2\nu)}}{\sinh(\zeta_k)} \right) \Big/ \left(\frac{1}{2\nu} + \zeta_k \frac{\cosh(\zeta_k)}{\sinh(\zeta_k)} \right)$$

und die Funktionen, die nur von y abhängen, sind

$$\chi_k(y) = \sin(k\pi y).$$

Die Lösung ist in Abbildung 8.9 dargestellt.

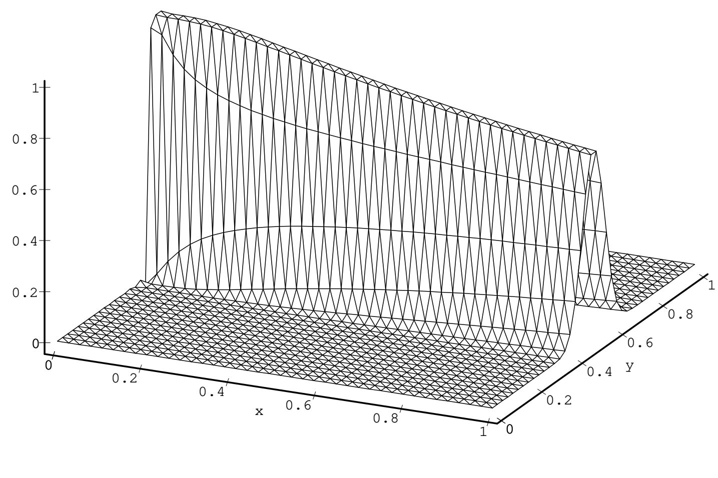


Abbildung 8.9: Lösung von Beispiel 8.2.2.

Wegen der Sprünge in der Randbedingung gilt für die Lösung $u \notin H^1(\Omega)$. Die in den Abschnitten 6.4 und 6.5 gemachten Voraussetzungen an u sind nicht erfüllt. Bei diesem Beispiel sind am Ausflußrand Neumannbedingungen vorgegeben. Die theoretischen Untersuchungen in den Abschnitten 6.4 und 6.5 haben sich jedoch auf Dirichletprobleme beschränkt. Ähnlich wie bei konformen FEM haben wir das Nichterfülltsein der Neumannbedingungen als zusätzlichen Term in die Fehlerschätzer aufgenommen, siehe Verfürth [Ver96b].

Die in Tabelle 8.19 präsentierten Ergebnisse wurden mit folgenden Parametern im Verfeinerungsprozeß berechnet, (vergleiche Abschnitt 6.7):

Markierungsmethode = Anteil_des_globalen_Fehlers
REFTOL = 0.5
COARSETOL = 0.0
MAXLEVEL = 99
BASELEVEL = 3
MINREF = 0.1 (= 10 %).

	$\eta_{T,1}$		$\eta_{T,2}$		$\tilde\eta_{T,1}$		$\tilde\eta_{T,2}$	
Schritt	Unb.	$\|\|u-u_h\|\|_0$	Unb.	$\|\|u-u_h\|\|_0$	Unb.	$\|\|u-u_h\|\|_0$	Unb.	$\|\|u-u_h\|\|_0$
6	2411	7.155-2	2411	7.155-2	2413	5.255-2	2413	5.255-2
9	7939	1.594-2	7990	1.137-2	7194	3.327-2	7194	3.327-2
12	27023	2.013-3	27130	1.444-3	22339	9.908-3	22396	9.911-3
15	93411	1.026-3	91179	5.504-4	73586	5.563-3	73434	5.519-3
16	137456	5.858-4	137389	3.145-4	109296	5.733-3	109106	2.954-3
17					156105	3.282-3	157077	3.327-3
unif. 7	196352	2.009-2						

Tabelle 8.19: Beispiel 8.2.2, Ergebnisse bei Verwendung lokal verfeinerter Gitter mit verschiedenen Fehlerschätzern.

Es ist wiederum zu beobachten, daß die mit Hilfe der adaptiven h–Methode erzielten Ergebnisse wesentlich besser als die mit gleichmäßiger Verfeinerung sind. Die Fehlerschätzer lassen sich trotz der fehlenden Regularität der Lösung mit Erfolg anwenden. Zwischen den Ergebnissen mit $\eta_{T,1}$ und $\eta_{T,2}$ sowie mit $\tilde\eta_{T,1}$ und $\tilde\eta_{T,2}$ gibt es nur kleine Unterschiede. Abermals ist zu beobachten, daß $\eta_{T,1}$ und $\eta_{T,2}$ im wesentlichen innerhalb der Grenzschichten verfeinern, währenddessen die mit $\tilde\eta_{T,1}$ und $\tilde\eta_{T,2}$ erzeugten lokalen Verfeinerungen ein größeres Gebiet überdecken, siehe Abbildung 8.10. Wie im Beispiel 8.2.1 sind die mit $\eta_{T,1}$ und $\eta_{T,2}$ erzielten Ergebnisse besser als die mit $\tilde\eta_{T,1}$ und $\tilde\eta_{T,2}$. Bei Verfeinerungsschritten, die die Verfeinerungstiefe innerhalb der Grenzschichten nicht erhöhen, ist wieder keine Verringerung des Fehlers festzustellen. Das leichte Anwachsen führen wir auf Effekte der numerischen Integration bei der Berechnung des Fehlers zurück.

Setzt man den Parameter MINREF = 0.0, so erhält man die in Tabelle 8.20 präsentierten Ergebnisse. Die erzeugten Gitter sehen ähnlich aus wie in Abbildung 8.10. Die Anzahl der ausgeführten Schritte hat sich drastisch erhöht. Dadurch hat sich die Rechenzeit im Vergleich zu den Rechnungen mit MINREF = 0.1 vervielfacht. Man sieht an diesem Beispiel die Bedeutung von MINREF für die Effizienz der Verfahren. Ein anderer bemerkenswerter Aspekt bei der Wahl von MINREF = 0.0 ist, daß man mit $\eta_{T,1}$ überhaupt ein gutes Gitter erzeugen kann. Für einen entsprechenden Fehlerschätzer in der Energienorm bei Verwendung konformer Finite Elemente ist das nicht der Fall, siehe [Joh94]. Hier wird nur an den Sprüngen der Randbedingung verfeinert. Es macht sich bemerkbar, daß die konforme diskrete Lösung versucht, eine Funktion zu approximieren, die nicht in $H^1(\Omega)$ liegt, und der Fehler in der starken H^1–Norm geschätzt

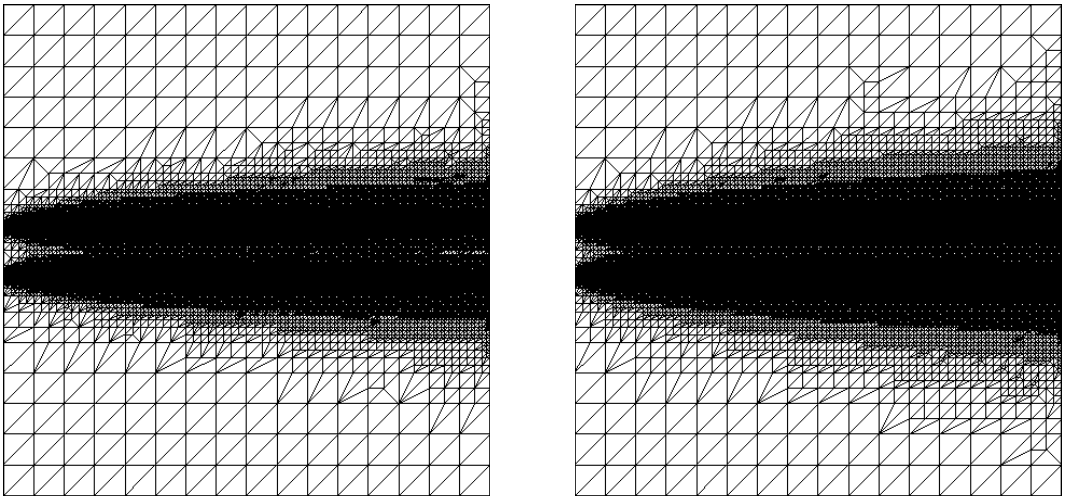


Abbildung 8.10: Beispiel 8.2.2, lokal verfeinerte Gitter mit $\eta_{T,1}$ (links) und $\tilde{\eta}_{T,1}$ (rechts).

wird. Die nichtkonforme Lösung u_h und der Fehlerschätzer $\eta_{T,1}$ in der schwächeren diskreten H^1–Norm haben mit den Sprüngen der Randbedingung offenbar keine Probleme.

	$\eta_{T,1}$		$\tilde{\eta}_{T,1}$	
Schritt	Unbekannte	$\|u-u_h\|_0$	Unbekannte	$\|u-u_h\|_0$
20	1405	6.286-2	1406	9.599-2
40	5945	1.065-2	3690	6.393-2
60	34934	1.971-3	10085	1.608-2
80			33891	8.376-3

Tabelle 8.20: Beispiel 8.2.2, Ergebnisse bei Verwendung lokal verfeinerter Gitter mit verschiedenen Fehlerschätzern und MINREF = 0.0.

Beispiel 8.2.3 Eine Lösung mit unterschiedlichen Singularitäten. Wir betrachten nun ein Problem in einem L–förmigen Gebiet

$$\begin{aligned} -10^{-6}\Delta u + (3,1)\cdot\nabla u + u &= 100r(r-0.5)(r-\sqrt{2}/2) && \text{in } (0,1)\times(0,1)\setminus[0.5,1]\times[0,0.5] \\ u &= 0 && \text{auf } \partial\Omega. \end{aligned}$$

Hierbei ist $r=\sqrt{(x-0.5)^2+(y-0.5)^2}$. Dieses Testproblem ist von Adam [Ada95] übernommen. Eine analytische Darstellung der Lösung ist nicht bekannt. Die Lösung besitzt eine Ecksingularität an der einspringenden Ecke $(0.5, 0.5)$, Randgrenzschichten und eine innere Grenzschicht von der einspringenden Ecke in Richtung der Konvektion. Ein wichtiges Kriterium für die Qualität einer berechneten Lösung ist die Auflösung der inneren Grenzschicht.

Die gewählten Parameter zur Generierung des adaptiven Gitters sind

Markierungsmethode = Maximaler_lokaler_Fehler
REFTOL = 0.5
COARSETOL = 0.005
MAXLEVEL = 99
BASELEVEL = 3
MINREF = 0.1 .

Auch bei diesem Beispiel sind kaum Unterschiede bei der Anwendung von $\eta_{T,2}$ anstatt $\eta_{T,1}$ beziehungsweise $\tilde{\eta}_{T,2}$ anstatt $\tilde{\eta}_{T,1}$ zu beobachten. In Abbildung 8.11 sind die erzeugten Gitter mit den berechneten Lösungen zu sehen. Man erkennt, daß der Fehlerschätzer $\eta_{T,1}$ im Gebiet der inneren Grenzschicht nicht so stark verfeinert, wie an den anderen Singularitäten. Das führt zu einer schlechteren Auflösung dieser Grenzschicht im Vergleich zur mit Hilfe des Fehlerschätzer $\tilde{\eta}_{T,1}$ erzielten Lösung. Die dargestellten Lösungen wurden jeweils auf Gittern mit etwa 125 000 Unbekannten berechnet. Wiederum erstreckt sich die von $\tilde{\eta}_{T,1}$ erzeugte lokale Verfeinerung über ein größeres Gebiet als die von $\eta_{T,1}$.

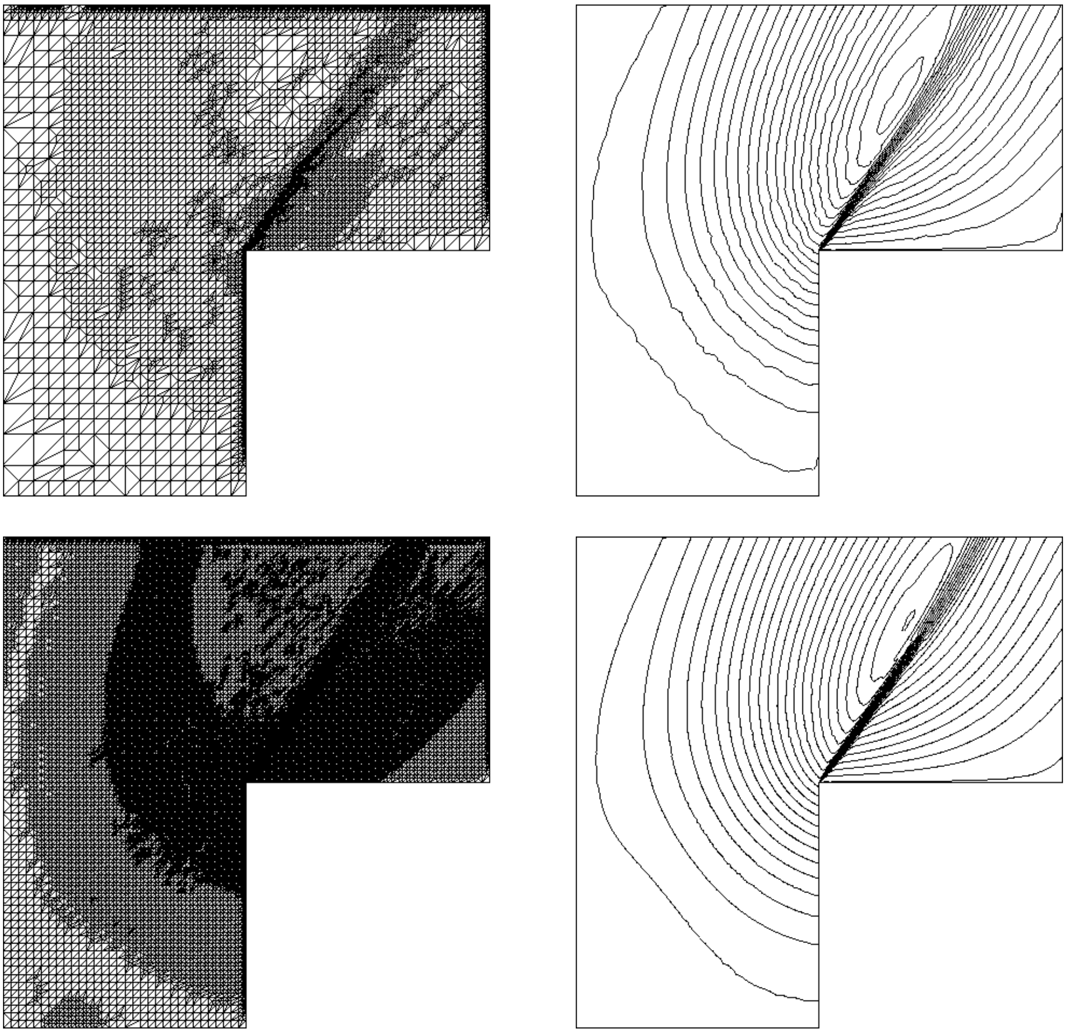


Abbildung 8.11: Beispiel 8.2.3, lokal verfeinerte Gitter und berechnete Lösungen, oben mit $\eta_{T,1}$, unten mit $\tilde{\eta}_{T,1}$.

Wählt man MINREF=0.0, so erzeugt $\eta_{T,1}$ ein ähnliches Gitter wie in Abbildung 8.11. Das ist insofern bemerkenswert, als daß ein Fehlerschätzer in der Energienorm für konforme Finite Elemente in diesem Fall im Bereich der inneren Grenzschicht nicht verfeinert, siehe [DJR+96].

Bemerkung 8.2.4 Auswertung der numerischen Tests.

1. **uniforme Verfeinerung – adaptive Verfeinerung.** Die Lösungen aller präsentierten Beispiele besitzen Singularitäten. Die Verwendung der Fehlerschätzer $\eta_{T,i}$ und $\tilde{\eta}_{T,i}$, $i = 1, 2$,

führte in jedem der Beispiele zu deutlich besseren Ergebnissen mit weniger Unbekannten als bei uniformer Verfeinerung.

2. **$\eta_{T,1}$ – $\eta_{T,2}$ und $\tilde{\eta}_{T,1}$ – $\tilde{\eta}_{T,2}$.** Es gab kaum Unterschiede bei der Verwendung von $\eta_{T,2}$ anstelle von $\eta_{T,1}$ beziehungsweise von $\tilde{\eta}_{T,2}$ anstelle von $\tilde{\eta}_{T,1}$.

3. **$\eta_{T,i}$ – $\tilde{\eta}_{T,i}, i = 1, 2$.** Für Probleme mit einer Singularität, Beispiel 8.2.1, oder mit mehreren gleichartigen Singularitäten, Beispiel 8.2.2, haben sich die mit $\eta_{T,i}$ erzeugten Gitter als besser herausgestellt. Im Beispiel 8.2.3 mit unterschiedlichen Singularitäten erzeugten $\tilde{\eta}_{T,i}$, $i = 1, 2$, besser angepaßte Gitter.

   In allen Testrechnungen erzeugten $\tilde{\eta}_{T,i}$, $i = 1, 2$, bei gleichen Parametern im Verfeinerungsprozeß und gleicher Anzahl von Schritten, Gitter mit weniger Unbekannten als $\eta_{T,i}$. Die von $\tilde{\eta}_{T,i}$ generierten lokal verfeinerten Gitter überdeckten ein größeres Gebiet und hatten eine geringere maximale Verfeinerungstiefe. Diese Erscheinungen wurde auch bei einer Vielzahl anderer Testrechnungen beobachtet.

4. **Verfeinerungsstrategien.** Alle zu wählenden Parameter beeinflussen das adaptiv erzeugte Gitter und damit die berechnete Lösung. Die Bedeutung des Parameters MINREF für effiziente adaptive Verfahren wurde im Beispiel 8.2.2 gezeigt. Eine geeignete Wahl liegt unserer Meinung nach in MINREF $\in [0.05, 0.15]$. Der Einfluß der anderen Parameter scheint weniger bedeutsam zu sein. Die in den Beispielen gewählten Werte entsprechen Erfahrungswerten für eine vernünftige Wahl.

5. **Theoretische Voraussetzungen der Herleitung der Fehlerschätzer sind nicht erfüllt.** Im Beispiel 8.2.2 wurde gezeigt, daß man die Fehlerschätzer $\eta_{T,i}$ und $\tilde{\eta}_{T,i}$, $i = 1, 2$, erfolgreich zur Gittergenerierung für Beispiele nutzen kann, in denen die Lösung nicht die in der Theorie erforderlichen Regularitätseigenschaften besitzt.

6. **Rechenzeiten und numerischer Aufwand.** Wir bezeichnen mit t_{uni} die Zeit zur Lösung eines Problems auf dem feinsten Gitter eines uniform verfeinerten Netzes. Bei den adaptiven Methoden war nur ein Bruchteil der Unbekannten sowie von t_{uni} nötig, um vergleichbare Ergebnisse wie bei uniformer Verfeinerung zu erzielen. Um die in den Tabellen als feinste adaptive Verfeinerung angegebenen Resultate zu berechnen, brauchte man jedoch ein Mehrfaches von t_{uni}. Es waren zwar weniger Unbekannte als bei gleichmäßiger Verfeinerung, jedoch waren viel mehr Schritte, insbesondere Lösungsprozesse, auszuführen. Es hat sich darüber hinaus gezeigt, daß man zur Lösung der diskreten Probleme auf lokal verfeinerten Gittern im allgemeinen mehr Iterationen brauchte als auf uniformen Gittern. Der numerische Aufwand zur Berechnung der Fehlerschätzungen und zur Markierung der Elemente war vernachlässigbar im Vergleich zum Gesamtaufwand, vergleiche auch [Joh94].

8.3 Die numerische Schätzung des globalen L^2–Fehlers

Im Kapitel 6.6 wurde der globale a posteriori Fehlerschätzer η für den Fehler in der L^2–Norm hergeleitet. An dieser Stelle wollen wir die Fehlerschätzung und den (numerisch approximierten) wahren Fehler miteinander vergleichen.

Wie bereits in Bemerkung 6.6.6 festgestellt worden ist, ist η nicht robust. Ist die Stabilitätskonstante C_S des stetigen dualen Problems (6.39) groß, so ist die alleinige Berechnung von η wertlos für eine Fehlerschätzung. Man muß sich irgendwie eine Vorstellung über die Größe von C_S verschaffen. Die Konstante C_S ist für gewöhnlich nicht berechenbar, da man die Lösung des stetigen dualen Problems (6.39) im allgemeinen nicht kennt und nicht numerisch berechnen kann. In unserer heuristischen Herangehensweise gehen wir deswegen von einem diskreten dualen Problem aus. Dieses wird genauso wie das ursprüngliche Problem stabilisiert, also entweder mit Upwind Methoden oder nichtkonformer Stromlinien–Diffusions Diskretisierung. Sei u_{2h} die berechnete Lösung auf dem Gitter, aus dem das Gitter hervorgegangen ist, auf dem u_h berechnet wurde. Dann wählen wir als rechte Seite des diskreten dualen Problems $u_h - u_{2h}$ und bezeichnen dessen Lösung mit $\hat{z}_h$. Desweiteren sei $\bar{z}_h$ die konforme, stückweise lineare Funktion, die man durch eine gewichtete Mittelung der nichtkonformen Funktionswerte in den Eckpunkten P_i der Dreiecke erhält:

$$\bar{z}_h(P_i) = \sum_{T\in\mathcal{T}_h, P_i\in T} \frac{\hat{z}_h(P_i)}{|T|}.$$

Seien $\hat{z}_h^1$ und $\bar{z}_h^1$ die stückweise konstanten Gradienten von $\hat{z}_h$ und $\bar{z}_h$. Mit einem weiteren Mittelungsprozeß,

$$\sum_{T\in\mathcal{T}_h, P_i\in T} \frac{\hat{z}_h^1(T)}{|T|}, \qquad \sum_{T\in\mathcal{T}_h, P_i\in T} \frac{\bar{z}_h^1(T)}{|T|},$$

werden diese in konforme, stückweise lineare Funktionen überführt. Von diesen kann man wieder numerisch die Ableitung bestimmen, die $\hat{z}_h^2$ und $\bar{z}_h^2$ genannt werden. Zur Approximation von C_S hat man nun die Varianten

$$\hat{C}_S = \frac{(\|\hat{z}_h^2\|_0^2 + \|\hat{z}_h^1\|_0^2 + \|\hat{z}_h\|_0)^{1/2}}{\|u_h - u_{2h}\|_0} \qquad \text{und} \qquad \bar{C}_S = \frac{(\|\bar{z}_h^2\|_0^2 + \|\bar{z}_h^1\|_0^2 + \|\bar{z}_h\|_0)^{1/2}}{\|u_h - u_{2h}\|_0}.$$

Bei numerischen Tests haben wir festgestellt, daß sich $\hat{C}_S$ und $\bar{C}_S$ wenig unterscheiden. Bei den konkreten Testbeispielen war $\hat{C}_S$ immer etwas größer als $\bar{C}_S$.

Auf Grund der rechten Seite $u_h - u_{2h}$ des diskreten dualen Problems wird mit der obigen Vorgehensweise eher die Stabilitätskonstante für die rechte Seite $u - u_{2h}$ approximiert, währendessen der Fehlerschätzer η für den Fehler $\|u - u_h\|_0$ berechnet wird.

Beispiel 8.3.1 Ein Beispiel ohne Konvektion. Wegen der Vermutung, daß im Beweis von Satz 6.6.1 der konvektive Term nicht optimal abgeschätzt worden ist, siehe Bemerkung 6.6.7, beginnen wir mit einem Beispiel, bei dem $b \equiv (0,0)^T$ ist. Die rechte Seite f wird so gewählt, daß

$$u(x,y,\nu) = 16x(1-x)y(1-y)\left[\frac{1}{2} + \frac{\arctan\left((2/\sqrt{\nu})(r_0^2 - (x-x_0)^2 - (y-y_0)^2)\right)}{\pi}\right]$$

mit $r_0 = 0.25$ und $x_0 = y_0 = 0.5$ die exakte Lösung von

$$\begin{aligned} -\nu\Delta u + 2u &= f \quad \text{in } (0,1)\times(0,1) \\ u &= 0 \quad \text{auf } \partial\Omega \end{aligned}$$

ist. Für $\nu = 10^{-4}$ ist die exakte Lösung in Abbildung 8.3 zu sehen. Die Lösungen u hängen von ν ab, insbesondere wächst $\|u\|_2$ für $\nu \to 0$. Das führt dazu, daß die asymptotische Konvergenzordnung von 2 für kleine ν auf den Leveln, deren Ergebnisse in Tabelle 8.21 präsentiert sind, noch nicht erreicht ist. Als Grobgitter wurde Gitter 1 aus Abbildung 8.1 verwendet, welches uniform verfeinert wurde.

Wir haben den Fehlerschätzer $\hat{\eta}$ verwendet, siehe Bemerkung 6.6.7. Die Resultate in Tabelle 8.21 zeigen, daß $\hat{\eta}$ tatsächlich nur für $\nu = 10^{-1}$ eine obere Schranke des globalen Fehlers liefert. In allen anderen Fällen ist $\hat{\eta}$ kleiner als der Fehler und insbesondere für $\nu \le 10^{-5}$ wird zudem eine viel zu hohe Konvergenzordnung vorgegaukelt. Die mit der oben beschriebenen Heuristik berechneten Fehlerabschätzungen liefern in jeder Testrechnung eine obere Schranke für den Fehler. Die Schranke liegt ungefähr 3– bis 10mal höher als der Fehler. Für kleine Werte von ν spiegelt sich die reduzierte Konvergenzordnung in der Konvergenzordnung von $\bar{C}_S\hat{\eta}$ wider.

ν	Level	$\|u-u_h\|_0$	Ord.	$\hat{\eta}$	Ord.	$\bar{C}_S$	$\bar{C}_S\hat{\eta}$	Ord.
10^{-1}	4	5.26-3	2.00	1.57-3	1.98	3.44	5.41-3	1.68
	5	1.32-4	2.00	3.94-4	1.99	3.67	1.45-3	1.90
	6	3.30-5	2.00	9.87-5	2.00	3.74	3.69-4	1.97
	7	8.35-6	1.98	2.47-5	2.00	3.75	9.26-5	2.00
10^{-3}	4	3.96-3	1.95	9.52-5	2.14	131.3	1.25-2	1.02
	5	1.02-3	1.95	2.79-5	1.77	174.8	4.89-3	1.36
	6	2.60-4	1.98	7.38-6	1.91	166.9	1.23-3	1.99
	7	6.53-5	1.99	1.87-6	1.98	194.4	3.64-4	1.76
10^{-5}	4	2.94-2	0.83	1.75-4	2.68	485.7	8.50-2	0.61
	5	1.38-2	1.09	2.21-5	2.98	1822.3	4.03-2	1.08
	6	5.39-3	1.36	2.51-6	3.14	5580.2	1.40-2	1.53
	7	1.71-3	1.65	4.15-7	2.60	12462.9	5.17-3	1.44
10^{-6}	4	4.14-2	0.63	2.27-4	2.55	494.8	1.12-1	0.43
	5	2.45-2	0.75	3.40-5	2.74	1977.0	6.72-2	0.74
	6	1.30-2	0.92	4.83-6	2.81	7351.2	3.55-2	0.92
	7	5.70-3	1.19	5.88-7	3.03	25924.0	1.52-2	1.22

Tabelle 8.21: Beispiel 8.3.1, berechneter und geschätzter Fehler mit $\hat{\eta}$.

Beispiel 8.3.2 Ein Beispiel mit Konvektion. Wir betrachten Beispiel 8.1.3 für unterschiedliche Werte von ν. Da die Lösung für kleine ν exponentielle Grenzschichten besitzt, ist in diesen Fällen die a priori Abschätzung (1.7) scharf. Bemerkung 6.6.7 folgend, verwenden wir den Fehlerschätzer $\hat{\eta}$, siehe (6.47). Die Rechnungen wurden mit der Samarskij Upwind Stabilisierung auf Gitter 1, siehe Abbildung 8.1, sowie mit uniformer Verfeinerung durchgeführt.

Die Ergebnisse der globalen Fehlerschätzung, Tabelle 8.22, zeigen die gleichen Charakteristiken wie im vorangegangenen Beispiel. Für kleine Werte von ν erhält man mit Hilfe der heuristischen Approximation der Stabilitätskonstanten eine robuste Schätzung des globalen Fehlers. Vor allem die tatsächlichen Konvergenzordnungen werden gut widergespiegelt.

Bemerkung 8.3.3 Auswertung der numerischen Tests.

1. **Globale Fehlerschätzung mit $\bar{C}_S$.** In den präsentierten Beispielen wurde der globale L^2–Fehler für kleine ν vom Fehlerschätzer $\hat{\eta}$ völlig unterschätzt. Mit Hilfe der heuristisch berechneten Konstante $\bar{C}_S$ war eine wesentlich zuverlässigere Fehlerschätzung durch $\bar{C}_S\hat{\eta}$ möglich. Die Konvergenzordnung des wahren Fehlers wurde durch $\bar{C}_S\hat{\eta}$ gut widergespiegelt.

2. **Offene Probleme.** Die Berechnung von $\bar{C}_S$ und $\hat{C}_S$ ist mit der Lösung eines zusätzlichen Problems verbunden, die zumindest in den gewählten Beispielen mindestens so aufwendig wie die Lösung des eigentlichen Problems war. Es wurden schlechtere Konvergenraten der Mehrgitteriteration für das duale Problem festgestellt. Bei Verwendung der NSDFEM divergierte das Mehrgitterverfahren sogar. Alle Ergebnisse wurden auf uniform verfeinerten Gittern berechnet. Der Grund dafür ist, daß lokal verfeinerte Gitter, die gut für das Originalproblem sind, im allgemeinen ungeeignet für das duale Problem sein werden. Man würde ein lokal verfeinertes „duales Gitter“ brauchen, was mit zusätzlichem Aufwand verbunden ist.

ν	Level	$\|u - u_h\|_0$	Ord.	$\hat{\eta}$	Ord.	$\bar{C}_S$	$\bar{C}_S\hat{\eta}$	Ord.
10^{-1}	4	5.07-3	1.60	1.43-2	1.87	2.24	3.19-2	1.46
	5	1.46-3	1.80	3.73-3	1.94	2.92	1.09-2	1.55
	6	3.90-4	1.90	9.50-4	1.97	3.52	3.35-3	1.70
	7	1.01-4	1.96	2.40-4	1.99	3.88	9.29-4	1.85
10^{-3}	4	3.57-2	0.54	3.02-2	1.50	6.6	2.00-1	0.48
	5	2.49-2	0.52	1.06-2	1.51	13.0	1.39-1	0.53
	6	1.74-2	0.51	3.70-3	1.52	25.1	9.31-2	0.58
	7	1.19-2	0.55	1.27-3	1.55	46.9	5.93-2	0.65
10^{-5}	4	3.63-2	0.53	3.04-2	1.49	6.8	2.07-1	0.46
	5	2.53-2	0.52	1.08-2	1.50	13.7	1.48-1	0.48
	6	1.78-2	0.51	3.81-3	1.50	27.6	1.05-1	0.49
	7	1.25-2	0.51	1.35-3	1.50	55.2	7.43-2	0.50
10^{-6}	4	3.63-2	0.53	3.04-2	1.49	6.8	2.07-1	0.46
	5	2.53-2	0.52	1.08-2	1.50	13.7	1.48-1	0.48
	6	1.78-2	0.51	3.81-3	1.50	27.6	1.05-1	0.49
	7	1.25-2	0.51	1.35-3	1.50	55.3	7.44-2	0.50

Tabelle 8.22: Beispiel 8.3.2, berechneter und geschätzter Fehler mit $\hat{\eta}$.

Kapitel 9

Numerische Ergebnisse für die stationären Navier–Stokes Gleichungen

In diesem Kapitel werden Ergebnisse von Testrechnungen für die stationären Navier–Stokes Gleichungen vorgestellt. Zuerst wird die Wirkungsweise der in den Abschnitten 7.2 und 7.3 hergeleiteten lokalen a posteriori Fehlerschätzer untersucht. Im Abschnitt 9.2 werden Eigenschaften der verwendeten nichtkonformen P_1/P_0–Finite Elemente Diskretisierung an Hand eines Benchmarkproblems aus dem DFG–Schwerpunktprogramm „Flow simulation with high–performance computers“ untersucht. An diesem Benchmarkproblem werden zum Abschluß dieses Kapitels verschiedene Löser für die auftretenden Sattelpunktprobleme miteinander verglichen und ihr Verhalten in bezug auf einige Verfahrensparameter und die verwendete Anzahl von Prozessoren eines Parallelrechners studiert.

9.1 Adaptive Methoden

In diesem Abschnitt wird die Wirkungsweise der in den Abschnitten 7.2 und 7.3 untersuchten lokalen a posteriori Fehlerschätzer für die stationären Stokes und Navier–Stokes Gleichungen an Hand numerischer Tests studiert. Ein Problem bereitet das Finden von Testbeispielen, die einerseits Singularitäten besitzen und deren exakte Lösung andererseits bekannt ist.

Beispiel 9.1.1 Stokes Gleichungen in einem Kreis mit einem Schlitz. Das Gebiet Ω sei der Kreis mit dem Mittelpunkt $(0,0)$ und dem Radius 1, welcher einen Schlitz zwischen den Punkten $(0,0)$ und $(1,0)$ besitzt, siehe Abbildung 9.1. Wir betrachten die Stokes Gleichungen

$$\begin{aligned} -\Delta \mathbf{u} + \nabla p &= \mathbf{0} && \text{in } \Omega \\ \nabla \cdot \mathbf{u} &= 0 && \text{in } \Omega \\ \mathbf{u} &= \mathbf{g} && \text{auf } \partial\Omega, \end{aligned} \tag{9.1}$$

wobei die Randbedingung $\mathbf{g}$ so gewählt ist, daß

$$\mathbf{u} = \frac{3}{2}\sqrt{r}\left(\cos\frac{\theta}{2} - \cos\frac{3\theta}{2}, 3\sin\frac{\theta}{2} - \sin\frac{3\theta}{2}\right), \quad p = -\frac{6}{\sqrt{r}}\cos\frac{\theta}{2}$$

mit $r^2 = x^2 + y^2$ und $0 \le \theta \le 2\pi$ die exakte Lösung von (9.1) ist. Die Lösung ist in Abbildung 9.2 zu sehen. Dieses Beispiel ist von Bank und Welfert [BW91] entlehnt. Man findet es auch bei

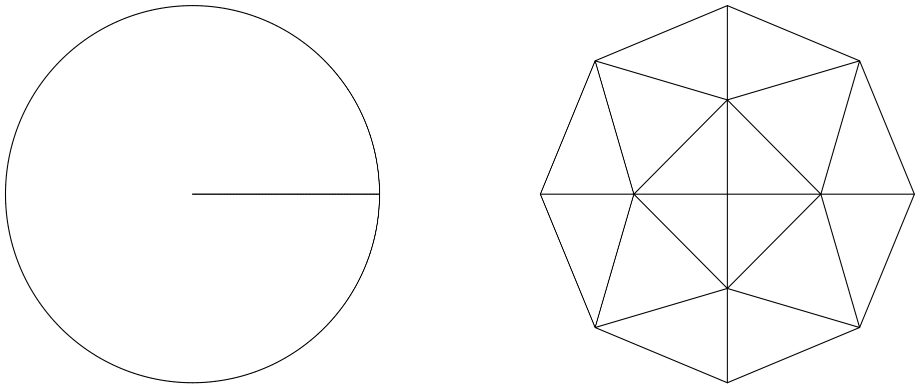


Abbildung 9.1: Gebiet und Grobgitter (Level 0) für die Beispiele 9.1.1 und 9.1.2.

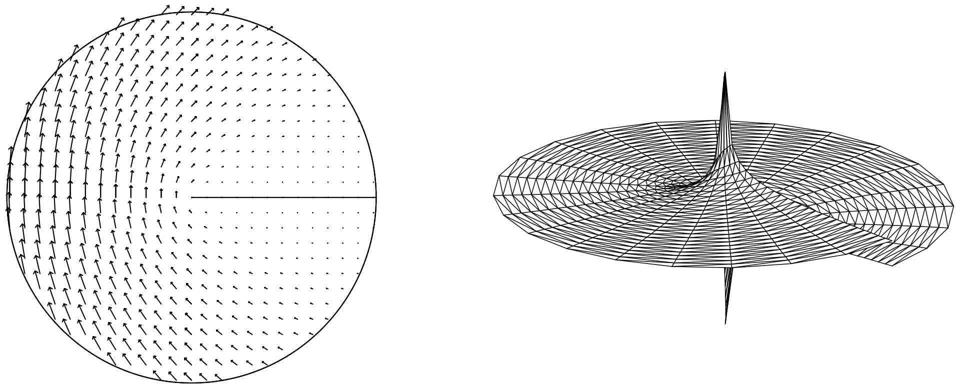


Abbildung 9.2: Lösung der Beispiele 9.1.1 und 9.1.2, links Geschwindigkeit, rechts Druck.

Verfürth [Ver96b]. In diesen Arbeiten wurden a posteriori Fehlerschätzer für die Energienorm und konforme Finite Elemente Methoden mit diesem Beispiel getestet.

Der Druck besitzt eine Singularität im Punkte $(0,0)$. Für die exakte Lösung gilt $\mathbf{u} \notin (H^2(\Omega))^2$ und $p \notin H^1(\Omega)$. Die in den Abschnitten 7.2 und 7.3 gemachten Voraussetzungen sind nicht erfüllt. Als Kriterien für die Qualität der berechneten Lösung $(\mathbf{u}_h, p_h)$ haben wir $\|\mathbf{u} - \mathbf{u}_h\|_0$ und die Auflösung der Singularität des Druckes gewählt. Diese Auflösung wird mit Hilfe des kleinsten Wertes $p_{h,min}$ und des größten Wertes $p_{h,max}$ von p_h gemessen. Die Rechnungen wurden mit Druckseparation, Variante 4, durchgeführt. Das verwendete Grobgitter ist in Abbildung 9.1 zu sehen.

Für die Stokes Gleichungen werden die lokalen Fehlerschätzer $\eta_{T,2}$, siehe (7.11), und $\tilde{\eta}_{T,2}$, siehe (7.17), verwendet. Da die rechte Seite $\mathbf{f}$ konstant Null ist, verschwindet der Term mit dem starken Residuum der Impulserhaltungsgleichung in $\eta_{T,2}$ und $\tilde{\eta}_{T,2}$. Beim Fehlerschätzen werden daher die Sprünge von $\nabla \mathbf{u}_h$ und p_h über die Elementkanten sowie das starke Residuum der Massenbilanz genutzt. Die Ergebnisse der durchgeführten Rechnungen sind in den Tabellen 9.1 bis 9.3 dargestellt. Für die adaptiven Testrechnungen wurden folgende Parameter gewählt (vergleiche Abschnitt 6.7):

Markierungsmethode = Maximaler_lokaler_Fehler
REFTOL = 0.5
COARSETOL = 0.0
MAXLEVEL = 99
BASELEVEL = 2
MINREF = 0.1 (= 10 %).

Schritt	Unb. Geschw.	Unb. Druck	$\|\mathbf{u}-\mathbf{u}_h\|_0$	$p_{h,min}$	$p_{h,max}$
4	12480	4096	4.624-2	-33.8	33.8
5	49536	16384	2.366-2	-48.8	48.8
6	197376	65536	1.197-2	-69.6	69.6
7	787968	262144	6.017-3	-98.9	98.9

Tabelle 9.1: Beispiel 9.1.1, Ergebnisse bei Verwendung eines uniform verfeinerten Gitters.

Schritt	Unb. Geschw.	Unb. Druck	$\|\mathbf{u}-\mathbf{u}_h\|_0$	$p_{h,min}$	$p_{h,max}$
6	3730	1266	1.876-2	-69.3	69.3
9	12420	4179	5.719-3	-198.0	198.2
12	40940	13712	1.777-3	-561.3	561.8
14	83656	27979	7.036-4	-1121.3	1121.6
16	172590	57667	4.486-4	-2245.8	2246.9
18	352654	117749	1.649-4	-4492.9	4457.2

Tabelle 9.2: Beispiel 9.1.1, Ergebnisse bei Verwendung des Fehlerschätzers $\eta_{T,2}$.

Schritt	Unb. Geschw.	Unb. Druck	$\|\mathbf{u}-\mathbf{u}_h\|_0$	$p_{h,min}$	$p_{h,max}$
6	3642	1241	2.934-2	-48.7	48.5
9	12790	4301	9.038-3	-98.0	98.1
12	35916	12047	2.864-3	-195.2	195.5
14	67542	22639	1.568-3	-276.8	278.2
16	127294	42582	8.817-4	-376.4	376.4
18	237298	79311	6.458-4	-396.2	396.0
20	421518	140792	3.445-4	-562.4	559.0

Tabelle 9.3: Beispiel 9.1.1, Ergebnisse bei Verwendung des Fehlerschätzers $\tilde{\eta}_{T,2}$.

Als erstes stellt man fest, daß sich bei diesem Beispiel der Einsatz adaptiver Methoden lohnt. So erhält man beispielsweise bei Nutzung von $\eta_{T,2}$ mit rund 15 000 Unbekannten (Schritt 9) etwa gleichwertige Ergebnisse wie mit gleichmäßiger Verfeinerung und rund 1 050 000 Unbekannten (Schritt 7), das heißt mit knapp 1.5% der Unbekannten ! Bei weiterer Verfeinerung mit Hilfe sowohl von $\eta_{T,2}$ als auch $\tilde{\eta}_{T,2}$ erhält man wesentlich bessere Ergebnisse als mit gleichmäßiger Verfeinerung. Der Grund hierfür ist die höhere Verfeinerungstiefe in der Umgebung des Punktes $(0,0)$, siehe Abbildungen 9.3 und 9.4. Dabei sieht man aber auch schon die unterschiedliche Arbeitsweise der beiden lokalen Fehlerschätzer. Analog wie bei den Konvektions–Diffusions Gleichungen verfeinert $\eta_{T,2}$ vor allem direkt an der Singularität, siehe Abbildung 9.3. Mit jedem Verfeinerungsschritt wird die Verfeinerungstiefe an der singulären Stelle erhöht. Genauso wie bei den Konvektions–Diffusions Gleichungen ist dies besser als die von $\tilde{\eta}_{T,2}$ erzeugte lokale Verfeinerung, die ein größeres Gebiet überdeckt, siehe Abbildung 9.4. Die Verwendung von $\eta_{T,1}$ anstatt $\eta_{T,2}$ beziehungsweise von $\tilde{\eta}_{T,1}$ anstelle von $\tilde{\eta}_{T,2}$ brachte keine wesentlichen Änderungen in den erzielten Ergebnissen.

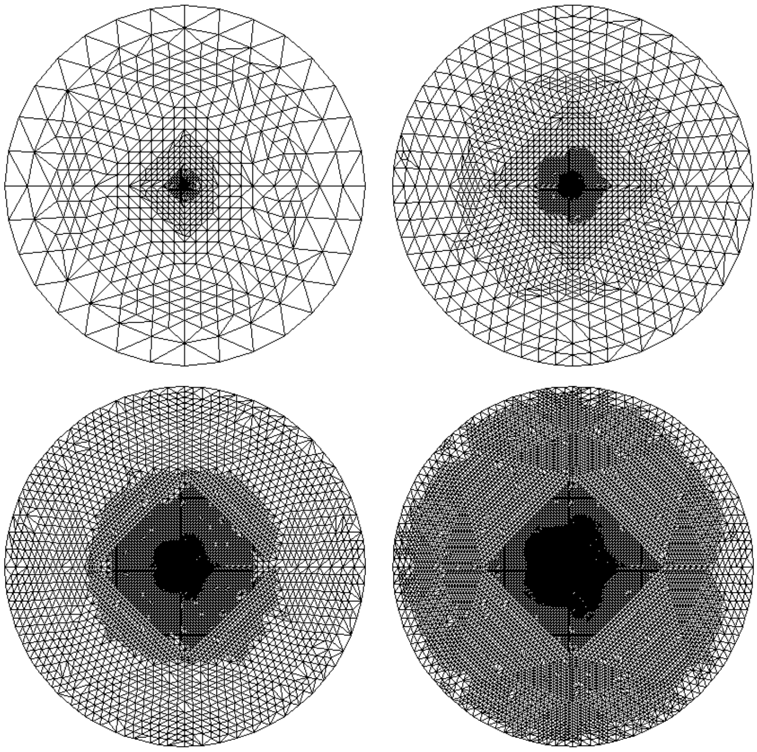


Abbildung 9.3: Beispiel 9.1.1, mit $\eta_{T,2}$ erzeugte Gitter, Schritte 6, 9, 12 und 14.

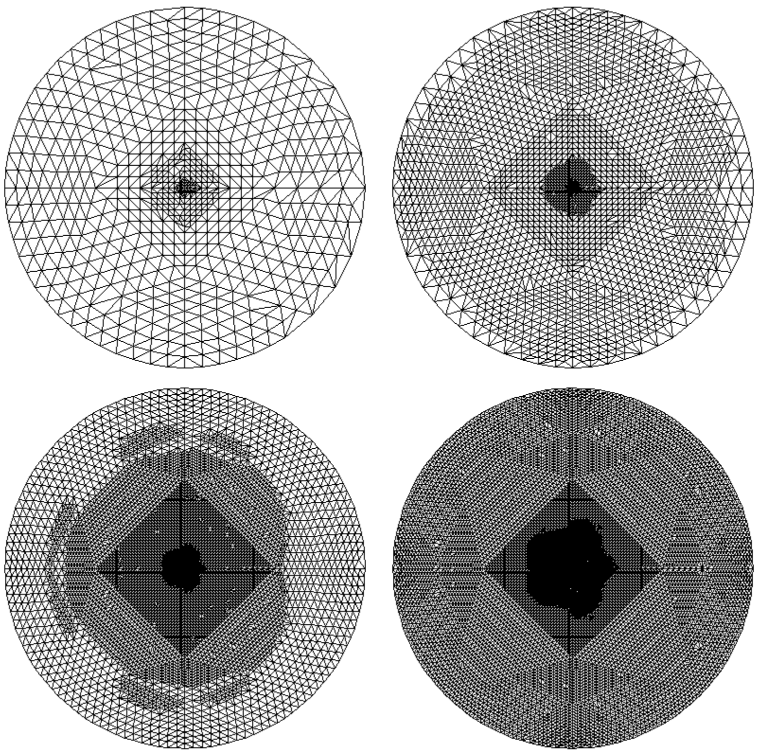


Abbildung 9.4: Beispiel 9.1.1, mit $\tilde{\eta}_{T,2}$ erzeugte Gitter, Schritte 6, 9, 12 und 14.

Beispiel 9.1.2 Navier–Stokes Gleichungen in einem Kreis mit einem Schlitz. Wir betrachten die Navier–Stokes Gleichungen

$$\begin{aligned} -\nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &= \mathbf{f} && \text{in } \Omega \\ \nabla\cdot\mathbf{u} &= 0 && \text{in } \Omega \\ \mathbf{u} &= \mathbf{g} && \text{auf } \partial\Omega \end{aligned}$$

in dem in Abbildung 9.1 skizzierten Gebiet Ω. Die rechte Seite $\mathbf{f} = (\mathbf{u}\cdot\nabla)\mathbf{u}$ und die Randbedingung $\mathbf{g}$ sind so gewählt, daß

$$\mathbf{u} = \frac{3}{2}\sqrt{r}\left(\cos\frac{\theta}{2} - \cos\frac{3\theta}{2}, 3\sin\frac{\theta}{2} - \sin\frac{3\theta}{2}\right), \quad p = -\frac{6\nu}{\sqrt{r}}\cos\frac{\theta}{2}$$

mit $r^2 = x^2 + y^2$ und $0 \le \theta \le 2\pi$ die exakte Lösung von Beispiel 9.1.2 ist, siehe Abbildung 9.2. Für die Testrechnungen haben wir $\nu = 1/20$ gewählt. Somit ist der Betrag des Druckes im Vergleich zu Beispiel 9.1.1 um den Faktor 20 reduziert.

Schritt	Unb. Geschw.	Unb. Druck	$\|\mathbf{u}-\mathbf{u}_h\|_0$	$p_{h,min}$	$p_{h,max}$
4	12480	4096	5.861-2	-1.71	1.63
5	49536	16384	2.555-2	-2.42	2.37
6	197376	65536	1.172-2	-3.45	3.43
7	787968	262144	5.624-3	-4.92	4.91

Tabelle 9.4: Beispiel 9.1.2, Ergebnisse bei Verwendung eines uniform verfeinerten Gitters.

Schritt	Unb. Geschw.	Unb. Druck	$\|\mathbf{u}-\mathbf{u}_h\|_0$	$p_{h,min}$	$p_{h,max}$
6	8026	2724	5.548-2	-2.87	1.86
9	25476	8558	1.936-2	-7.91	5.59
12	72424	24238	1.191-2	-22.43	16.12
14	133544	44328	9.711-3	-44.02	32.28
16	293842	98119	5.402-3	-88.18	64.43
17	400720	133764	4.276-3	-123.91	90.73

Tabelle 9.5: Beispiel 9.1.2, Ergebnisse bei Verwendung des Fehlerschätzers $\eta_{T,1}$.

Schritt	Unb. Geschw.	Unb. Druck	$\|\mathbf{u}-\mathbf{u}_h\|_0$	$p_{h,min}$	$p_{h,max}$
6	8168	2771	5.077-2	-2.93	2.61
9	26874	9030	1.659-2	-8.01	7.68
12	94000	31431	8.535-3	-22.59	22.04
14	220684	73722	7.465-3	-44.93	44.13
16	427520	142716	3.974-3	-88.61	87.62

Tabelle 9.6: Beispiel 9.1.2, Ergebnisse bei Verwendung des Fehlerschätzers $\eta_{T,2}$.

Für die adaptiven Testrechnungen haben wir die folgenden Parameter gewählt (vergleiche Abschnitt 6.7):

Schritt	Unb. Geschw.	Unb. Druck	$\|\mathbf{u}-\mathbf{u}_h\|_0$	$p_{h,min}$	$p_{h,max}$
6	8410	2851	6.024-2	-1.84	1.68
9	24914	8371	2.351-2	-3.58	3.38
12	66728	22348	1.562-2	-5.11	4.87
14	122164	40857	7.043-3	-7.08	6.94
16	219220	73264	6.626-3	-10.05	9.86
18	420406	140352	3.519-3	-14.20	13.98

Tabelle 9.7: Beispiel 9.1.2, Ergebnisse bei Verwendung des Fehlerschätzers $\tilde{\eta}_T$.

Markierungsmethode = Maximaler_lokaler_Fehler
REFTOL = 0.5
COARSETOL = 0.0
MAXLEVEL = 99
BASELEVEL = 3
MINREF = 0.1 (= 10 %).

In den Tabellen 9.5 bis 9.7 sind Resultate für die lokalen Fehlerschätzer $\eta_{T,1}$, $\eta_{T,2}$ und $\tilde{\eta}_T$, siehe (7.4), (7.5) und (7.12), präsentiert. Bei diesem Beispiel verschwindet das Residuum der starken Formulierung der linearen Impulsbilanz in den Fehlerschätzern nicht. Ganz im Gegenteil, wie man in den Abbildungen 9.5 und 9.6 sieht, ist dieser Anteil wesentlich. An den Stellen mit hoher Geschwindigkeit und hohem Geschwindigkeitsgradienten wird stark lokal verfeinert. Die Fehlerschätzer $\eta_{T,1}$ und $\eta_{T,2}$ erzeugten vergleichbare Gitter mit etwa gleich guten Lösungen. Alle lokalen Fehlerschätzer erkennen die Singularität des Druckes im Punkt $(0,0)$. Die Verfeinerungstiefe der mit $\eta_{T,1}$ und $\eta_{T,2}$ erzeugten Gitters ist in $(0,0)$ höher. Das führt zu besseren Extremwerten des berechneten Druckes im Vergleich zur Verwendung von $\tilde{\eta}_T$. Diese Extremwerte sind auf allen lokal verfeinerten Gittern deutlich besser als auf dem uniform verfeinerten Gitter. Die Fehler in der L^2–Norm der Geschwindigkeit sind letzten Endes auch kleiner als bei uniformer Verfeinerung. Dazu benötigen die lokal verfeinerten Gitter gut die Hälfte der Unbekannten des uniform verfeinerten Gitters. Das sind wesentlich mehr als beim Stokes Problem.

Die Vorteile der Nutzung adaptiver Gitterverfeinerung sind auch bei diesem Beispiel deutlich, obwohl sie nicht so groß wie beim Stokes Problem, Beispiel 9.1.1, sind. Vor allem die Singularität des Druckes wird bei Verwendung beider lokaler Fehlerschätzer wesentlich besser aufgelöst. Zur Verbesserung der Qualität der berechneten Geschwindigkeit braucht man etwa die Hälfte der Unbekannten der uniformen Verfeinerung. Die lokalen Fehlerschätzer $\eta_{T,1}$ und $\eta_{T,2}$ haben sich erwartungsgemäß $\tilde{\eta}_T$ überlegen gezeigt.

Bei der Verwendung lokal verfeinerter Gitter gab es Probleme bei der Lösung der Gleichungssysteme. Dies hängt mit der unterschiedlichen Güte der Randapproximation im Zusammenhang mit den inhomogenen Randdaten zusammen. Sowohl die Standardwahl der Randdaten, als auch Integrationsformeln höherer Ordnung ergaben eine recht große Abweichung von der diskreten Kompatibilitätsbedingung (2.16). Dies führte zu einer Stagnation der Norm des Residuums bei Werten um 10^{-6}.

Bemerkung 9.1.3 Auswertung der numerischen Tests. Die Aussagen von Bemerkung 8.2.4 lassen sich sinngemäß hierher übertragen.

1. $\boldsymbol{\eta_{T,1}}$ **und** $\boldsymbol{\eta_{T,2}}$**.** Die mit Hilfe der Fehlerschätzer $\eta_{T,1}$ und $\eta_{T,2}$ erzeugten Gitter unterschieden sich bei den präsentierten Beispielen nur wenig. Die berechneten Lösungen waren etwa gleich genau.

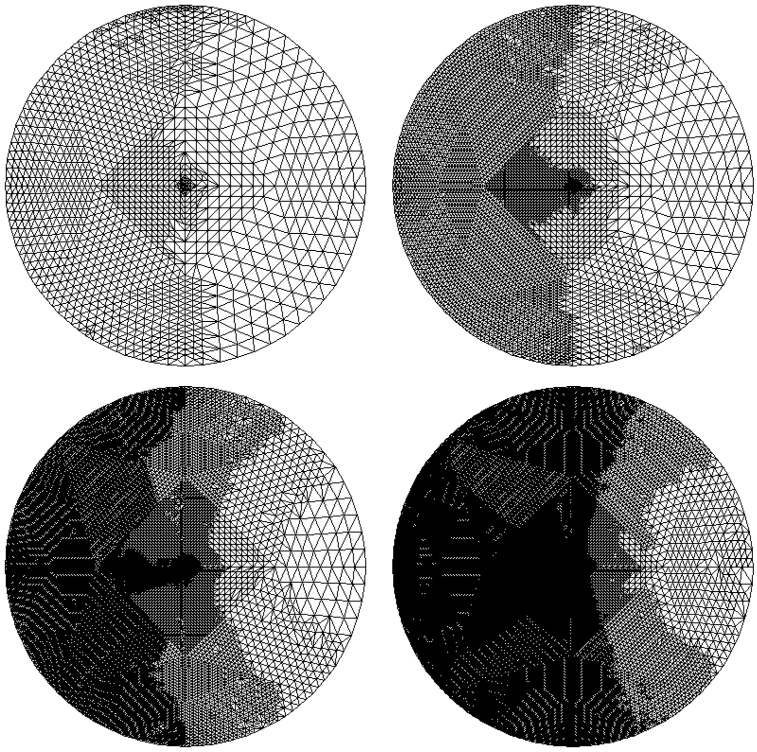


Abbildung 9.5: Beispiel 9.1.2, mit $\eta_{T,2}$ erzeugte Gitter, Schritte 6, 9, 12 und 14.

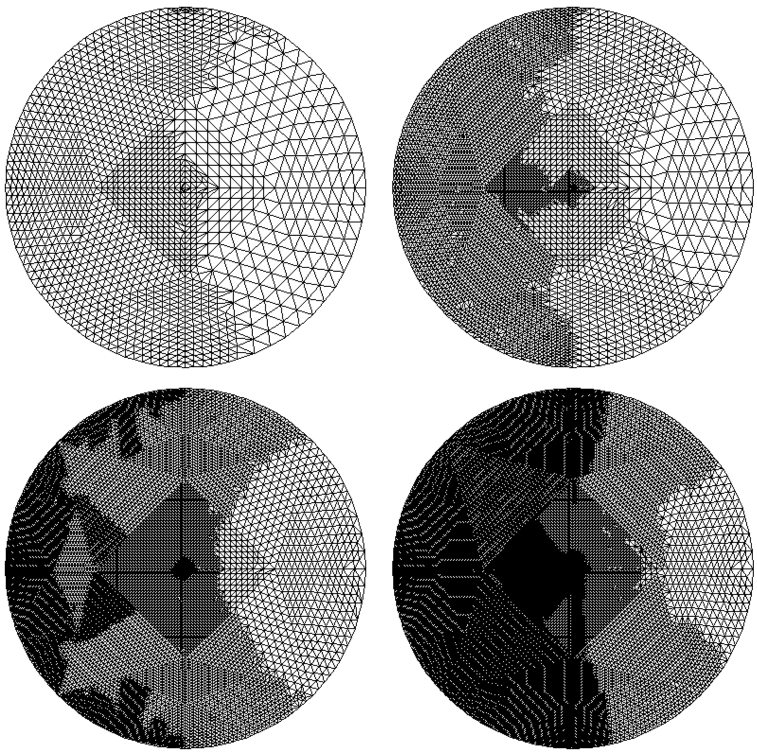


Abbildung 9.6: Beispiel 9.1.2, mit $\tilde{\eta}_T$ erzeugte Gitter, Schritte 6, 9, 12 und 14.

9.2 Strömung durch einen Kanal um einen Zylinder, ein Benchmarkproblem

In diesem Abschnitt werden numerische Ergebnisse zu einem Benchmarkproblem des DFG–Schwerpunktprogrammes „Flow simulation with high–performance computers“ vorgestellt, siehe Schäfer und Turek [ST96a].

Bei dem Benchmarkproblem handelt es sich um eine Strömung durch den in Abbildung 9.7 skizzierten Kanal. Im vorderen Teil des Kanals befindet sich ein Zylinder, der nicht symmetrisch

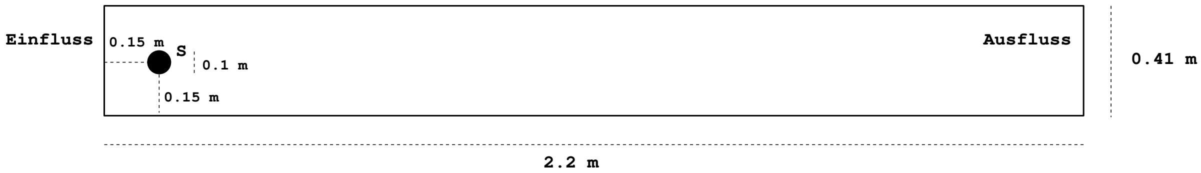


Abbildung 9.7: Gebiet für die Benchmarkprobleme.

zum Rand des Kanals ist. Zu berechnende Benchmarkkennwerte sind der Widerstandsbeiwert c_w und der Auftriebsbeiwert c_a, die als

$$c_w = \frac{20}{U^2} \int\limits_S \left(\nu \frac{\partial \mathbf{u}_{t_S}}{\partial n} n_y - p n_x \right) dS, \qquad c_a = -\frac{20}{U^2} \int\limits_S \left(\nu \frac{\partial \mathbf{u}_{t_S}}{\partial n} n_x + p n_y \right) dS, \tag{9.2}$$

definiert sind. Hierbei ist $n = (n_x, n_y)^T$ der Normalenvektor an S, der in das Gebiet Ω hineinzeigt, $t_S = (n_y, -n_x)^T$ der Tangentialvektor, u_{t_S} die Tangentialgeschwindigkeit und U ist die in der Einleitung zu Kapitel 2 eingeführte charakteristische Geschwindigkeit des Problems. Die Dichte ist mit $\rho = 1kg/m^3$ und die kinematische Zähigkeit mit $\nu = 10^{-3}m^2/s$ gegeben. Die charakteristische Länge des Problems ist als der Durchmesser des Zylinders $L = 0.1m$ definiert. Der Widerstandsbeiwert c_w ist ein Maß für die Kraft, die auf den Zylinder in Richtung der Strömung wirkt, und der Auftriebsbeiwert c_a ein Maß für die Kraft senkrecht zur Strömung. Es gibt einige Vorgehensweisen, die gesuchten Kennwerte mit den vorhandenen numerischen Daten zu berechnen, siehe Tabellen 9.15 und 9.16.

Wir führen die Testrechnungen mit drei unterschiedlichen Grobgittern durch, die in Abbildung 9.8 zu sehen sind. In Tabelle 9.8 sind die wichtigsten Daten dieser Gitter bei gleichmäßiger

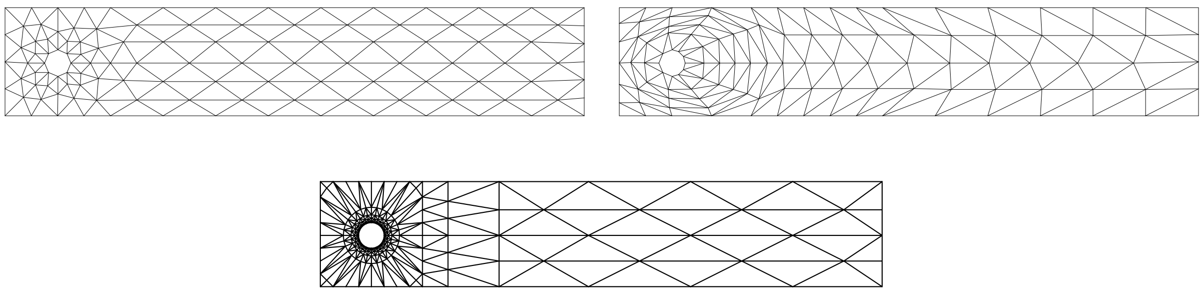


Abbildung 9.8: Die verwendeten Grobgitter, Gitter 1, 2 und 3.

Verfeinerung, nämlich die Anzahl der Unbekannten und die Güte der Approximation des Zylinders mit Hilfe eines Polygonzuges, enthalten. Man erkennt in Abbildung 9.8, daß die Region um den Zylinder bei allen Gittern mit etwas kleineren Elementen trianguliert ist, als beispielsweise

	Gitter 1			Gitter 2			Gitter 3		
Level	Geschw.	Druck	S	Geschw.	Druck	S	Geschw.	Druck	S
0	532	192	8	490	115	10	1092	388	32
3	36512	12288	64	34160	11520	80	73920	24832	256
4	146752	49152	128	137440	46080	160	296832	99328	512
5	588416	196608	256	551360	184320	320	1189632	397312	1024

Tabelle 9.8: Die Anzahl der Unbekannten bei gleichmäßiger Verfeinerung.

das Ende des Kanals. Das liegt an der Geometrie des Gebietes. Man kann jedoch bei Gitter 1 und Gitter 2 keinesfalls von einer speziellen a priori Verfeinerung um den Zylinder, mit dem Ziel möglichst guter Ergebnisse für die Benchmarkkennwerte, sprechen.

Beispiel 9.2.1 Ein Benchmarkproblem für die stationären Navier–Stokes Gleichungen (2.6). Wir betrachten

$$\begin{array}{rcll} -10^{-3}\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &=& \mathbf{0} & \text{in } \Omega \\ \nabla\cdot\mathbf{u} &=& 0 & \text{in } \Omega \\ \mathbf{u}(0,y) = \mathbf{u}(2.2,y) &=& 0.41^{-2}\,(1.2y(0.41-y),0) & 0 \le y \le 0.41 \\ \mathbf{u}(x,0) = \mathbf{u}(x,0.41) &=& (0,0) & 0 \le x \le 2.2 \\ \mathbf{u}(S) &=& (0,0). & \end{array}$$

Die charakteristische Geschwindigkeit in diesem Problem ist die Höchstgeschwindigkeit des parabelförmigen Ein– und Ausströmprofils $U = 0.2m/s$. Damit ergibt sich für diese Strömung die Reynoldszahl $Re = UL/\nu = 20$. Neben (9.2) ist die Differenz des Druckes zwischen Anfang und Ende des Zylinders

$$\Delta p := p(0.15, 0.2) - p(0.25, 0.2)$$

ein dritter Benchmarkkennwert in diesem Beispiel. Eine numerische Lösung dieses Problems ist in den Abbildungen 9.9 und 9.10 zu sehen. Hinter dem Zylinder entstehen zwei Wirbel. Der höchste Druck bildet sich am Anfang des Zylinders aus. Infolge der Rückströmung durch die Wirbel ist ober– und unterhalb des Zylinders, ein wenig hinter der Mitte, ein erhöhter Druck zu beobachten.

Ergebnisse mit den verschiedenen Druckseparationen

Wir wollen zunächst die Diskretisierung finden, die die besten Ergebnisse für Beispiel 9.2.1 liefert. Dazu beginnen wir mit den im Abschnitt 2.6 vorgestellten Varianten der Druckseparation. Da $\mathbf{f} = \mathbf{0}$ ist, entspricht die erste Variante einer Diskretisierung ohne Separation des Druckes. In Variante 2 wurde $n_0 = 1$ gewählt, das heißt, das Ergebnis nach der ersten nichtlinearen Iteration wurde verwendet. Die Rechnungen wurden auf Gitter 1 und mit Samarskij Upwind durchgeführt. Die Benchmarkkennwerte wurden mit Hilfe der Variante 5, siehe Tabellen 9.15, 9.16, berechnet.

Die Ergebnisse in den Tabellen 9.9, 9.10 und 9.11 zeigen, daß man mit allen Arten der Druckseparation bessere Resultate erzielt, als ohne diese.

Für eine konkrete Testreihe sind die Rechenzeiten in bezug auf die Lösung des Problems ohne Druckseparation in Tabelle 9.12 präsentiert. Die auftretenden linearen Systeme wurden mit dem gekoppelten Mehrgitterverfahren und Vanka Glätter gelöst. Die nichtlineare Iteration auf den einzelnen Leveln brach bei einer euklidischen Norm des Residuums von kleiner als 10^{-7} ab. Man erkennt, daß die Lösung eines Neumannproblems für den Druck in jedem nichtlinearen

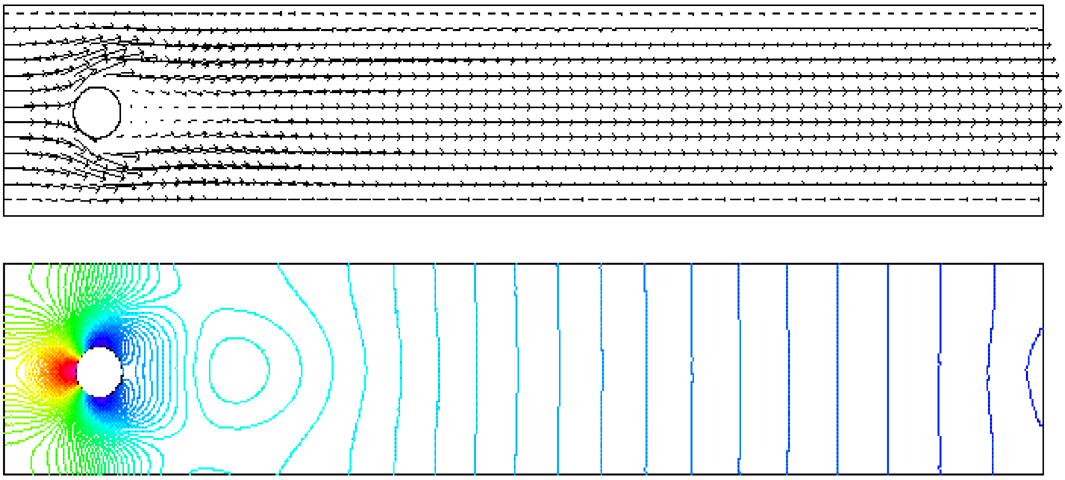


Abbildung 9.9: Die Lösung von Beispiel 9.2.1 in Ω, oben Geschwindigkeit, unten Druck.

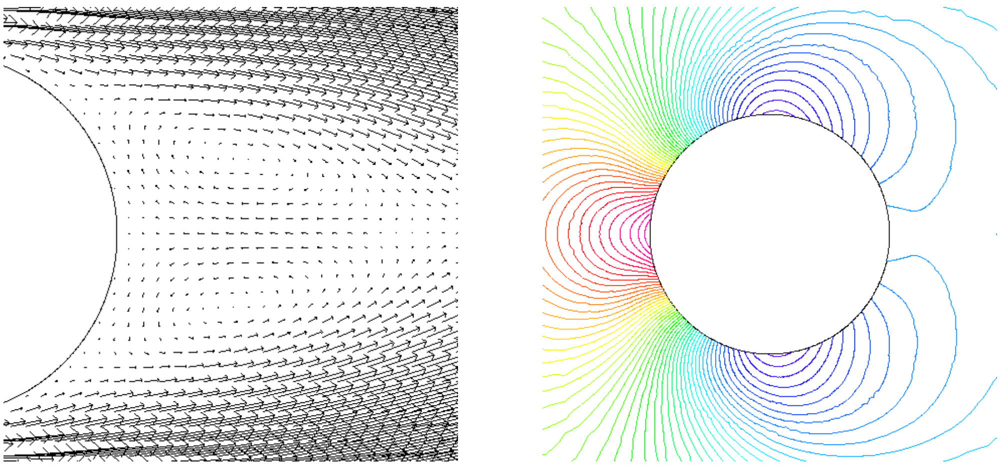


Abbildung 9.10: Die Lösung von Beispiel 9.2.1 um den Zylinder, links Geschwindigkeit, rechts Druck.

Druckseparation Level	ohne	Var. 2	Var. 3	Var. 4
3	5.6990	5.6650	5.6637	5.6797
4	5.6211	5.6101	5.6100	5.6132
5	5.5911	5.5883	5.5883	5.5889

Tabelle 9.9: Der c_w–Wert bei verschiedenen Arten der Druckseparation, Benchmarkintervall: $c_w \in [5.57, 5.59]$.

Iterationsschritt, Variante 3, recht viel Zeit in Anspruch nimmt, wobei es vielleicht auch gereicht hätte, das Neumannproblem weniger genau als bei dieser Testserie zu lösen. Bei der Anwendung von Druckseparation Variante 4 benötigt man zum Erreichen des Abbruchkriteriums einige Mehrgitterzyklen weniger als bei der Lösung des Problems ohne Druckseparation. Deswegen spart man etwas Rechenzeit.

Sowohl die berechneten Ergebnisse bezüglich der Benchmarkkennwerte als auch die benötig-

Druckseparation Level	ohne	Var. 2	Var. 3	Var. 4
3	2.638-3	3.715-3	4.002-3	5.040-3
4	9.016-3	9.365-3	9.380-3	9.745-3
5	1.030-2	1.027-2	1.038-2	1.049-2

Tabelle 9.10: Der c_a–Wert bei verschiedenen Arten der Druckseparation, Benchmarkintervall: $c_a \in [1.04-2, 1.10-2]$.

Druckseparation Level	ohne	Var. 2	Var. 3	Var. 4
3	0.1123	0.1126	0.1126	0.1135
4	0.1152	0.1152	0.1152	0.1154
5	0.1164	0.1164	0.1164	0.1166

Tabelle 9.11: Die Druckdifferenz Δp bei verschiedenen Arten der Druckseparation, Benchmarkintervall: $\Delta p \in [0.1172, 0.1176]$.

Druckseparation Level	ohne	Var. 2	Var. 3	Var. 4
3	100	104	128	117
4	100	103	132	95
5	100	103	128	78

Tabelle 9.12: Die Kosten der verschiedenen Arten der Druckseparation auf einem seriellen Rechner.

ten Rechenzeiten zeigen, daß Druckseparation Variante 4 eine gute Wahl für das Benchmarkproblem 9.2.1 ist. Alle weiteren Rechnungen wurden deshalb damit durchgeführt.

Vergleich von scharfem und Samarskij Upwind

Ein weiterer Bestandteil der Diskretisierung ist die Stabilisierungsmethode. In Tabelle 9.13 sind die Ergebnisse bei Verwendung von scharfem und von Samarskij Upwind gegenübergestellt. Die Tests wurden auf Gitter 1, mit Druckseparation Variante 4 und mit der Berechnung der Benchmarkkennwerte nach Variante 5, Tabellen 9.15, 9.16, durchgeführt.

Es ist in Tabelle 9.13 augenscheinlich, daß insbesondere die Beiwerte c_w und c_a bei scharfem Upwind deutlich schlechter sind als bei Samarskij Upwind. Für eine ähnliche Genauigkeit würde man beim scharfen Upwind ein bis zwei Verfeinerungsstufen mehr brauchen. Aus diesem Grunde sind alle Ergebnisse, die im weiteren präsentiert werden, unter Verwendung der Samarskij Upwind Diskretisierung berechnet worden.

	scharfes Upwind			Samarskij Upwind		
Level	c_w	c_a	Δp	c_w	c_a	Δp
3	5.8729	-8.184-3	0.1153	5.6797	5.040-3	0.1135
4	5.7528	2.777-3	0.1164	5.6132	9.745-3	0.1154
5	5.6728	7.286-3	0.1169	5.5889	1.049-2	0.1166
Benchmark-intervall	[5.5700, 5.5900]	[1.040-2, 1.100-2]	[0.1172, 0.1176]	[5.5700, 5.5900]	[1.040-2, 1.100-2]	[0.1172, 0.1176]

Tabelle 9.13: Das Benchmarkproblem 9.2.1 mit Verwendung von scharfem und Samarskij Upwind.

Rechnungen auf anderen Gittern

Der Vergleich der Ergebnisse auf Gitter 1 und 2, Tabelle 9.13 rechte Spalte und Tabelle 9.14, zeigt, daß die Benchmarkkennwerte auf Gitter 1 genauer als auf Gitter 2 berechnet wurden. So braucht man beim c_a–Wert eine Verfeinerungsstufe mehr auf Gitter 2, um eine ähnliche Genauigkeit wie auf Gitter 1 zu erzielen. Wir nehmen an, daß dies an der regelmäßigeren Triangulierung in der Umgebung des Zylinders liegt. Auf dem angepaßten Gitter 3 erhält man die besten Ergebnisse. Etwa 400 000 Unbekannte auf Level 4 reichen aus, um für den Widerstandsbeiwert c_w sowie die Druckdifferenz Δp genauere Werte als auf Gitter 1 mit knapp 800 000 Unbekannten zu erhalten. Ein c_a–Wert innerhalb des Benchmarkintervalls wird auf Gitter 3 jedoch erst mit 1.2 Millionen Unbekannten berechnet. Im folgenden sind die meisten Ergebnisse auf Gitter 1 berechnet wurden, für die anderen Gitter werden aber auch noch einige Resultate präsentiert.

	Gitter 2			Gitter 3		
Level	c_w	c_a	Δp	c_w	c_a	Δp
3	5.7011	-1.220-2	0.1120	5.6066	1.185-2	0.1163
4	5.6199	5.254-3	0.1148	5.5797	1.136-2	0.1170
5	5.5906	9.332-3	0.1162	5.5784	1.089-2	0.1173
Benchmark-intervall	[5.5700, 5.5900]	[1.040-2, 1.100-2]	[0.1172, 0.1176]	[5.5700, 5.5900]	[1.040-2, 1.100-2]	[0.1172, 0.1176]

Tabelle 9.14: Das Benchmarkproblem 9.2.1 mit Verwendung der Gitter 2 und 3.

Zur Berechnung der Benchmarkkennwerte

Der Widerstandsbeiwert c_w und der Auftriebsbeiwert c_a sind in (9.2) definiert. Bei ihrer numerischen Berechnung müssen zum einen die stetigen Funktionen $\mathbf{u}$ und p durch diskrete Funktionen ersetzt werden. Zum anderen sind noch gewisse Umformungen der Formel (9.2) möglich. Schon Dorok hat in [Dor95] festgestellt, daß unterschiedliche Berechnungswege zu zum Teil erheblich voneinander abweichenden Resultaten führen.

Sei $n = (n_x, n_y)^T$ die Einheitsnormale an S, die in Ω hineinzeigt, $t_S = (n_y, -n_x)$ und $\mathbf{u} = (u_1, u_2)$. Dann ist

$$\frac{\partial \mathbf{u}_{t_S}}{\partial n} = (u_{1x}n_y - u_{2x}n_x)n_x + (u_{1y}n_y - u_{2y}n_x)n_y, \tag{9.3}$$

wobei mit u_{ix}, u_{iy} die Ableitung der i–ten Komponente von $\mathbf{u}$ nach x beziehungsweise y gemeint ist. Wegen der Divergenzfreiheit von $\mathbf{u}$ gilt

$$0 = \nabla \cdot \mathbf{u} = u_{1x} + u_{2y}. \tag{9.4}$$

Da $\mathbf{u}$ in beiden Komponenten auf dem Rand S konstant ist, gelten

$$\frac{\partial u_1}{\partial t_S} = u_{1x}n_y - u_{1y}n_x = 0 \text{ auf } S, \tag{9.5}$$

$$\frac{\partial u_2}{\partial t_S} = u_{2x}n_y - u_{2y}n_x = 0 \text{ auf } S. \tag{9.6}$$

Setzt man (9.5) und (9.6) in (9.3) ein, so ergibt sich

$$\frac{\partial \mathbf{u}_{t_S}}{\partial n} = u_{1y}(n_x^2 + n_y^2) - u_{2x}(n_x^2 + n_y^2) = u_{1y} - u_{2x}. \tag{9.7}$$

Damit hat man die Berechnungsvorschriften

$$\left.\begin{aligned} c_w &= 20U^{-2}\int\limits_S \nu\left[(u_{1x}n_y - u_{2x}n_x)n_x + (u_{1y}n_y - u_{2y}n_x)n_y\right]n_y - pn_x dS \\ c_a &= -20U^{-2}\int\limits_S \nu\left[(u_{1x}n_y - u_{2x}n_x)n_x + (u_{1y}n_y - u_{2y}n_x)n_y\right]n_x + pn_y dS \end{aligned}\right\} \tag{9.8}$$

und

$$\left.\begin{aligned} c_w &= 20U^{-2}\int\limits_S \nu\left[u_{1y} - u_{2x}\right]n_y - pn_x dS \\ c_a &= -20U^{-2}\int\limits_S \nu\left[u_{1y} - u_{2x}\right]n_x + pn_y dS \end{aligned}\right\}. \tag{9.9}$$

Man kann desweiteren Berechnungsvorschriften für c_w und c_a herleiten, die nicht auf Randintegralen über S, sondern auf Flächenintegralen beruhen. Wir betrachten die Impulsbilanz in der starken Formulierung der Navier–Stokes Gleichungen (2.6). Es sei eine Funktion $\mathbf{v} \in \left(H^1(\Omega)\right)^2$ mit $\mathbf{v} = \mathbf{0}$ auf $\partial\Omega \setminus S$ gegeben. Durch partielle Integration erhält man

$$\mathbf{0} = \nu(\nabla\mathbf{u}, \nabla\mathbf{v}) + b(\mathbf{u}, \mathbf{u}, \mathbf{v}) - (p, \nabla\cdot\mathbf{v}) - (\mathbf{f}, \mathbf{v}) - \int\limits_S p\cdot(\mathbf{v}\cdot n)dS + \int\limits_S \nu(\nabla\mathbf{u}\cdot n)\cdot\mathbf{v}dS,$$

wobei n die oben definierte in Ω gerichtete Einheitsnormale ist. Wählt man speziell $\mathbf{v} = (v_1, 0)^T$ mit $v_1|_S = 1$, so ist

$$-\int\limits_S p\cdot(\mathbf{v}\cdot n)dS = -\int\limits_S pn_x dS.$$

Mit Hilfe von (9.4), (9.6) und (9.7) erhält man

$$\begin{aligned} \int\limits_S \nu(\nabla\mathbf{u}\cdot n)\cdot\mathbf{v}dS &= \int\limits_S \nu(u_{1x}n_x + u_{1y}n_y)dS = \int\limits_S \nu(-u_{2y}n_x + u_{1y}n_y)dS \\ &= \int\limits_S \nu(-u_{2x}n_y + u_{1y}n_y)dS = \int\limits_S \nu\frac{\partial\mathbf{u}_{t_S}}{\partial n}n_y dS. \end{aligned}$$

Wählt man $\mathbf{v} = (0, v_2)^T$ mit $v_2|_S = 1$, so folgen analog

$$-\int\limits_S p\cdot(\mathbf{v}\cdot n)dS = -\int\limits_S pn_y dS \quad \text{und} \quad \int\limits_S \nu(\nabla\mathbf{u}\cdot n)\cdot\mathbf{v}dS = -\int\limits_S \nu\frac{\partial\mathbf{u}_{t_S}}{\partial n}n_x dS.$$

Damit ist $\forall \mathbf{v} \in (H^1(\Omega))^2$

$$\left.\begin{aligned} c_w &= -20U^{-2}\Big(\nu(\nabla\mathbf{u},\nabla\mathbf{v}) + b(\mathbf{u},\mathbf{u},\mathbf{v}) - (p,\nabla\cdot\mathbf{v}) - (\mathbf{f},\mathbf{v})\Big) \text{ mit } \mathbf{v} = (v_1,0)^T, v_1|_S = 1 \\ c_a &= -20U^{-2}\Big(\nu(\nabla\mathbf{u},\nabla\mathbf{v}) + b(\mathbf{u},\mathbf{u},\mathbf{v}) - (p,\nabla\cdot\mathbf{v}) - (\mathbf{f},\mathbf{v})\Big) \text{ mit } \mathbf{v} = (0,v_2)^T, v_2|_S = 1 \end{aligned}\right\}. \quad (9.10)$$

In den Berechnungsvorschriften (9.8), (9.9) und (9.10) werden die stetigen Größen durch diskrete ersetzt. Anstelle von $\mathbf{u}$ wird $\mathbf{u}_h$ und statt p werden p_h oder die Projektion $I_{C^0}^{Q_h} p_h$ verwendet (vergleiche Abschnitt 2.6). Bei der Vorschrift (9.10) müssen noch geeignete Funktionen $\mathbf{v}$ festgelegt werden. Aus Kostengründen sollten diese einen kleinen Träger besitzen. Zum einen haben wir stückweise lineare und konforme Funktionen $\mathbf{v}_c$ genommen, die nur in den Eckpunkten der Dreiecke, die auf S liegen, in der entsprechenden Komponente den Wert 1 annehmen. Zum anderen haben wir (9.10) auch mit linearen und nichtkonformen Funktionen $\mathbf{v}_{nc}$ getestet, die nur in den Kantenmittelpunkten, die auf S liegen, in der entsprechenden Komponente den Wert 1 annehmen. Hier gilt jedoch $\mathbf{v}_{nc} \notin (H^1(\Omega))^2$. Die Ergebnisse mit den verschiedenen Berechnungsvorschriften und diskreten Funktionen sind für Gitter 1 bei Verwendung von Samarskij Upwind und Druckseparation Variante 4 in den Tabellen 9.15 und 9.16 zusammengestellt.

Var.	Formel	Druck	$\mathbf{v}$	Level 3	Level 4	Level 5
1	(9.8)	p_h	-	5.5577	5.5390	5.5498
2	(9.8)	$I_{C^0}^{Q_h} p_h$	-	5.5430	5.5354	5.5494
3	(9.9)	p_h	-	5.6592	5.5935	5.5781
4	(9.9)	$I_{C^0}^{Q_h} p_h$	-	5.6445	5.5899	5.5777
5	(9.10)	p_h	$\mathbf{v}_c$	5.6797	5.6132	5.5889
6	(9.10)	$I_{C^0}^{Q_h} p_h$	$\mathbf{v}_c$	5.6559	5.5954	5.5785
7	(9.10)	p_h	$\mathbf{v}_{nc}$	5.6290	5.5776	5.5700
8	(9.10)	$I_{C^0}^{Q_h} p_h$	$\mathbf{v}_{nc}$	5.5985	5.5626	5.5629

Tabelle 9.15: c_w–Werte mit verschiedenen Arten der Berechnung, Benchmarkintervall: $c_w \in [5.57, 5.59]$.

Var.	Formel	Druck	$\mathbf{v}$	Level 3	Level 4	Level 5
1	(9.8)	p_h	-	3.116-3	8.698-3	1.006-2
2	(9.8)	$I_{C^0}^{Q_h} p_h$	-	3.273-3	8.629-3	9.924-3
3	(9.9)	p_h	-	3.831-3	9.283-3	1.040-2
4	(9.9)	$I_{C^0}^{Q_h} p_h$	-	3.988-3	9.214-3	1.027-2
5	(9.10)	p_h	$\mathbf{v}_c$	5.040-3	9.745-3	1.049-2
6	(9.10)	$I_{C^0}^{Q_h} p_h$	$\mathbf{v}_c$	4.085-3	9.503-3	1.041-2
7	(9.10)	p_h	$\mathbf{v}_{nc}$	5.431-3	9.597-3	1.039-2
8	(9.10)	$I_{C^0}^{Q_h} p_h$	$\mathbf{v}_{nc}$	4.488-3	9.200-3	1.013-2

Tabelle 9.16: c_a–Werte mit verschiedenen Arten der Berechnung, Benchmarkintervall: $c_a \in [1.04-2, 1.10-2]$.

Die Verwendung lokal verfeinerter Gitter

Es stellt sich die Frage, ob man die bisher erzielten Ergebnisse mittels lokal verfeinerter Gitter noch verbessern kann. Das Problem besteht in den Kriterien für eine geeignete lokale Verfeinerung. Die in den Abschnitten 7.2 und 7.3 hergeleiteten a posteriori Fehlerschätzer schätzen den lokalen Fehler der Geschwindigkeit und des Druckes in der H^1– beziehungsweise L^2–Norm, jedoch nicht Fehler für die Kennwerte dieses Benchmarkproblems. Die Entwicklung von Fehlerschätzern für die Benchmarkkennwerte ist möglich, beispielsweise untersuchen Becker und Rannacher in [Bec95] und [BR96] einen a posteriori Fehlerschätzern für c_w. Dieser Fehlerschätzer wird mit Hilfe einer Verallgemeinerung der Beweistechnik von Johnson et. all [EEHJ95] hergeleitet, siehe Abschnitt 6.6. In der schwachen Formulierung des dualen Problems ist die rechte Seite anstelle einer Norm des Fehlers ein interessierendes Funktional, zum Beispiel der Widerstandsbeiwert. Becker und Rannacher ermitteln eine Näherungslösung dieses dualen Problems, berechnen lokale Normen dieser Lösung, setzen diese in die Fehlerabschätzung ein und nutzen die so erhaltenen Werte zur Gittergenerierung.

Wir haben die Fehlerschätzer $\eta_{T,1}, \eta_{T,2}$ und $\tilde{\eta}_T$ mit folgenden Parametern verwendet (siehe Abschnitt 6.7):

Markierungsmethode = Maximaler_lokaler_Fehler
REFTOL = 0.5
COARSETOL = 0.0
MAXLEVEL = 99
BASELEVEL = 3
MINREF = 0.05 .

Die Daten der mit Hilfe der lokalen Fehlerschätzer erzeugten Gitter sowie die berechneten Benchmarkkennwerte sind in den Tabellen 9.17 bis 9.19 zu sehen. Die Auswirkungen der lokalen Verfeinerungen auf die Güte der einzelnen Benchmarkkennwerte ist unterschiedlich und am besten in den Diagrammen in Abbildung 9.11 zu erkennen. Dort sind die berechneten Benchmarkkennwerte in Abhängigkeit von der Anzahl der Unbekannten dargestellt. Es zeigt sich, daß man auf allen lokal verfeinerten Gittern eine Verbesserung des berechneten Widerstandsbeiwertes erhält. Auf den mit $\eta_{T,1}$ und $\eta_{T,2}$ erzeugten Gittern benötigt man etwa 200 000 Unbekannte, um Werte im Benchmarkintervall zu erhalten, währenddessen man bei uniformer Verfeinerung knapp 800 000 Unbekannte braucht. Beim Auftriebsbeiwert ist die Situation diffiziler. Auf den mit $\eta_{T,1}$ und $\tilde{\eta}_T$ erzeugten Gittern schwankt der Auftriebsbeiwert erheblich. Man erhält mit weniger Unbekannten als bei uniformer Verfeinerung keine Ergebnisse, die ständig innerhalb des Benchmarkintervalls liegen. Einzig bei Verwendung von $\eta_{T,2}$ sind die Ergebnisse besser als mit uniformer Verfeinerung. Ab etwa 200 000 Unbekannten befinden sich alle berechneten Auftriebsbeiwerte innerhalb des Benchmarkintervalls. Die Druckdifferenz wird vor allem auf den von $\eta_{T,2}$, aber auch auf den von $\tilde{\eta}_T$, erzeugten Gittern, besser als mit uniformer Verfeinerung berechnet. Dafür scheinen die Verfeinerungen hinter dem Zylinder von Bedeutung zu sein, die in Abbildung 9.12 zu erkennen sind. Auf dem mit $\eta_{T,1}$ erzeugten Gittern ist hinter dem Zylinder nicht verfeinert und die Druckdifferenz verbessert sich im Vergleich zu uniformer Verfeinerung nicht.

Der Fehlerschätzer $\eta_{T,2}$ erzeugte lokal verfeinerte Gitter, auf denen sich alle berechneten Benchmarkkennwerte im Vergleich zu uniformer Verfeinerung verbessert haben. Nutzt man $\eta_{T,1}$ anstelle von $\eta_{T,2}$, so hat man lediglich eine Verbesserung des Widerstandsbeiwertes. Hier ist ein Beispiel, bei dem die Sprünge der nichtkonformen Funktion über die Kanten eine wesentliche Rolle bei der Erzeugung der lokalen Verfeinerung spielen. Auf den mit $\tilde{\eta}_T$ erzeugten lokal verfeinerten Gittern erhält man bessere Widerstandsbeiwerte und Druckdifferenzen als mit gleichmäßiger Verfeinerung.

Schritt	Geschw.	Druck	S	c_w	c_a	Δp
4	44014	14789	65	5.6175	3.030-3	0.1132
6	73704	24697	79	5.6085	1.064-2	0.1134
8	116112	38855	94	5.5943	8.907-3	0.1146
10	193310	64606	111	5.5890	1.002-2	0.1157
12	314730	105129	139	5.5861	9.848-3	0.1165
14	503282	168039	178	5.5832	1.099-2	0.1165
16	836944	279301	215	5.5823	1.067-2	0.1168
Benchmarkintervall				[5.5700, 5.5900]	[1.040-2, 1.100-2]	[0.1172, 0.1176]

Tabelle 9.17: Das Benchmarkproblem 9.2.1 mit Verwendung lokaler Verfeinerung und $\eta_{T,1}$.

Schritt	Geschw.	Druck	S	c_w	c_a	Δp
4	43992	14794	98	5.6152	6.418-3	0.1145
6	81038	27174	161	5.5934	1.129-2	0.1160
8	120084	40221	231	5.5921	1.000-2	0.1170
10	211620	70764	264	5.5873	1.093-2	0.1168
12	334698	111832	304	5.5842	1.070-2	0.1169
14	483840	161620	433	5.5831	1.055-2	0.1173
16	786500	262598	509	5.5817	1.066-2	0.1172
Benchmarkintervall				[5.5700, 5.5900]	[1.040-2, 1.100-2]	[0.1172, 0.1176]

Tabelle 9.18: Das Benchmarkproblem 9.2.1 mit Verwendung lokaler Verfeinerung und $\eta_{T,2}$.

Schritt	Geschw.	Druck	S	c_w	c_a	Δp
4	43016	14465	91	5.6231	6.855-3	0.1141
6	101744	34056	101	5.6134	1.530-2	0.1162
8	149890	50146	152	5.5967	1.043-2	0.1163
10	237494	79394	159	5.5925	9.644-3	0.1162
12	361258	120684	176	5.5900	1.046-2	0.1172
14	503246	168084	275	5.5862	1.097-2	0.1168
16	732354	244518	305	5.5836	1.043-2	0.1168
Benchmarkintervall				[5.5700, 5.5900]	[1.040-2, 1.100-2]	[0.1172, 0.1176]

Tabelle 9.19: Das Benchmarkproblem 9.2.1 mit Verwendung lokaler Verfeinerung und $\tilde{\eta}_T$.

Bei der numerischen Lösung der Probleme auf den lokal verfeinerten Gittern traten keine Schwierigkeiten auf.

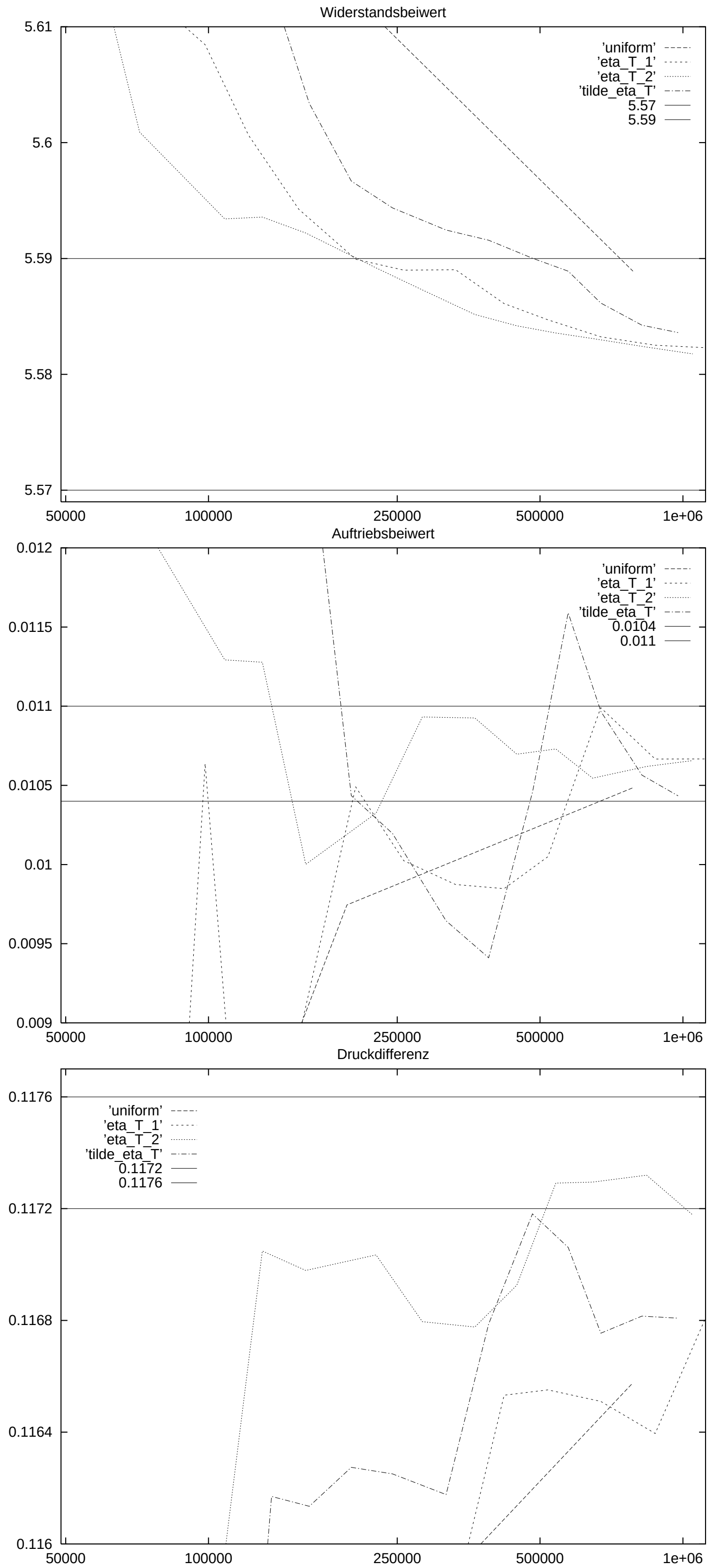


Abbildung 9.11: Die berechneten Benchmarkkennwerte in Abhängigkeit von der Anzahl der Unbekannten.

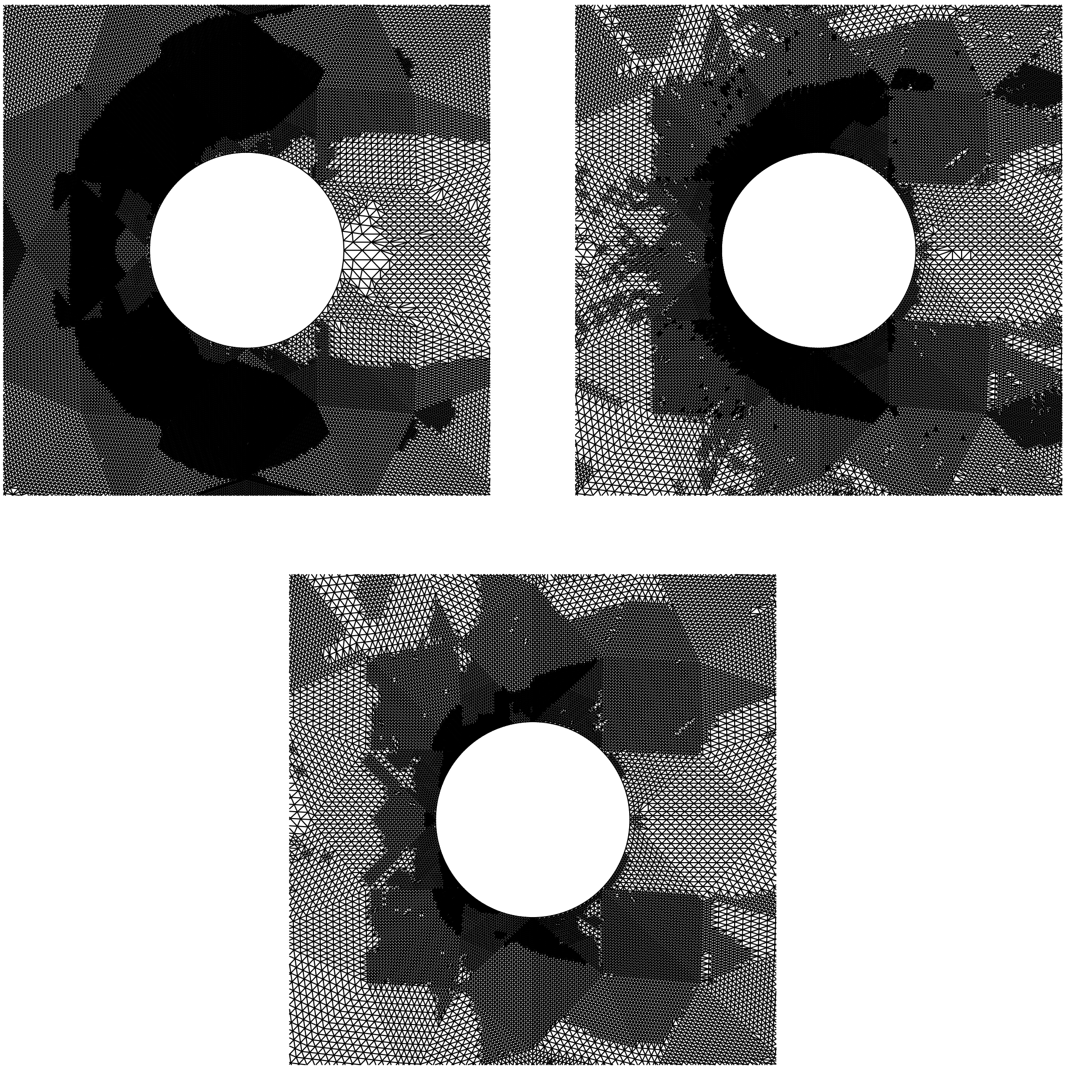


Abbildung 9.12: Die Verfeinerung um den Zylinder, links oben $\eta_{T,1}$, rechts oben $\eta_{T,2}$, unten $\tilde{\eta}_T$, jeweils Schritt 14.

Auswertung der numerischen Tests

Bemerkung 9.2.2 Auswertung der numerischen Tests.

1. **Genauigkeit der nichtkonformen P_1/P_0–Finite Elemente Diskretisierung.** Mit Hilfe der nichtkonformen P_1/P_0–Finite Element Diskretisierung konnten die geforderten Benchmarkkennwerte mit guter Genauigkeit berechnet werden. Die dazu benötigte Anzahl von Freiheitsgraden ist jedoch recht hoch. Bei den präsentierten Testrechnungen wurde die Abhängigkeit der berechneten Kennwerte vom verwendeten Grobgitter und von der verwendeten Berechnungsvorschrift deutlich.

2. **Druckseparation.** Die Verwendung aller Varianten der Druckseparation brachte mit einem vertretbaren numerischen Mehraufwand eine Verbesserung der Ergebnisse im Vergleich zur Diskretisierung ohne Druckseparation.

3. **Scharfes und Samarskij Upwind.** In Abwesenheit von scharfen Grenzschichten, wie bei diesem Beispiel, ist die Diskretisierung mit Samarskij Upwind der mit scharfem Upwind

deutlich überlegen.

4. **Lokal verfeinerte Gitter.** Die Verwendung lokal verfeinerter Gitter, die mit $\eta_{T,2}$ erzeugt wurden, brachte bei diesem Beispiel eine Verbesserung aller berechneten Benchmarkkennwerte bei gleichzeitiger Verringerung der Anzahl der Unbekannten. Die Verwendung von $\eta_{T,2}$ brachte deutlich bessere Ergebnisse als die von $\eta_{T,1}$.

9.3 Die Untersuchung verschiedener Löser

In diesem Abschnitt werden die in den Kapiteln 3 bis 5 vorgestellten Löser für das Sattelpunktproblem (3.1) numerisch untersucht. Die Untersuchungen wurden am Beispiel 9.2.1 durchgeführt.

Sofern nicht anders vermerkt, wurden die folgenden Parameter bei den Verfahren verwendet:

- Fixpunktiteration (2.19) zur Lösung des nichtlinearen Problems,
- Abbruch der nichtlinearen Iteration bei euklidischer Norm des Residuums kleiner als 10^{-7},
- Standardapproximation der Randdaten, siehe Bemerkung 2.4.2,
- Druckseparation Variante 4,
- Stabilisierung mit Samarskij Upwind,
- Schrittweitenkontrolle,
- Grobgitteriteration Variante 1, wobei die euklidische Norm des Residuums um den Faktor 0.1 verringert wird, siehe Abschnitt 4.3,
- keine Dämpfung in der Grobgitter– und Glättungsiteration,
- $\beta = 1.0$ bei der ILU_β–Zerlegung,
- Verwendung von $A_{1,1}^{-1}$ im Druckkorrektur Verfahren,
- Gitter 1,
- lexikographische Anordnung der Knoten auf einem Prozessor: ein Knoten mit den Koordinaten (x_0, y_0) ist ein Vorgänger eines Knotens mit den Koordinaten (x_1, y_1) genau dann wenn $(x_0 < x_1) \vee ((x_0 = x_1) \wedge (y_0 < y_1))$ ist.

Die Standardapproximation der Randdaten reicht hier aus, um das gestellte Abbruchkriterium zu erfüllen. Für schärfere Abbruchkriterien, etwa euklidische Norm des Residuums kleiner als 10^{-10}, ist eine bessere Approximation der Randdaten notwendig.

Das Verhalten der Iterationsverfahren kann von den Startwerten der nichtlinearen Iteration abhängen. Wir haben sowohl Testrechnungen durchgeführt, bei denen der Startwert $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$ war, als auch Testrechnungen, bei denen die interpolierte Lösung des nächsttieferen Levels als Startwert verwendet wurde.

Als nächstes werden die Abkürzungen in den Tabellen dieses Abschnitts erklärt:

Löser	GMRES(k)	Druckkorrektur Verfahren mit GMRES(k) zur Lösung des Schur–Komplement Systems (3.11) für den Druck,
	Vanka–V	gekoppeltes Mehrgitterverfahren mit Vanka–Glätter, V–Zyklus,
	Vanka–F	gekoppeltes Mehrgitterverfahren mit Vanka–Glätter, F–Zyklus,
	Vanka–W	gekoppeltes Mehrgitterverfahren mit Vanka–Glätter, W–Zyklus,
	stab. Vanka	gekoppeltes Mehrgitterverfahren mit stabilisiertem Vanka–Glätter,
M	Anzahl der Richardson Iterationen im Druckkorrektur Verfahren beziehungsweise Anzahl der Mehrgitterzyklen im gekoppelten Mehrgitterverfahren pro nichtlinearem Iterationsschritt,	
Gl.	Anzahl der Glättungsiterationen in den verwendeten Mehrgitterverfahren, wobei die Anzahl der Vor– und Nachglättungsiterationen immer gleich gewählt wurde,	

Kost.	numerische Kosten in $opmsN_{Topel}$ für die Lösung eines Sattelpunktproblems, vergleiche Kapitel 5,
Lev.	Level, auf dem die Rechnungen ausgeführt wurden,
Pr.	Anzahl der verwendeten Prozessoren, das sind beim Parsytec GC PowerPlus zwei CPU mit einer gemeinsamen Speichereinheit,
It.	Anzahl der benötigten nichtlinearen Iterationen und Anzahl der benötigten Mehrgitterzyklen, da oft auf Grund des Erfülltseins der Abbruchbedingung in der letzten oder in den letzten beiden nichtlinearen Iterationen nicht mehr die vorgegebene Anzahl von Mehrgitterzyklen ausgeführt wurde,
Zeit	CPU–Zeit in Sekunden,
Kom.	Anteil der Kommunikation an der Gesamtzeit in %,
Grg.	Anteil der Iterationen auf dem Grobgitter an der Gesamtzeit in %,
Zeit/Zyk.	benötigte Zeit für einen Mehrgitterzyklus in Sekunden.

In den unten angegebenen Tabellen befinden sich links vom Doppelstrich immer die gegebenen Werte und rechts davon die Werte, die sich in den Rechnungen ergeben haben.

Die Rechnungen auf einem Prozessor wurden, sofern nicht anders vermerkt, auf einer Workstation HP 9000 J 200 durchgeführt. Als Parallelrechner hatten wir einen Parsytec GC PowerPlus mit 48 Prozessoren, das sind je zwei CPU mit einem Speicher, zur Verfügung. Dieser Rechner ist dafür bekannt, daß die Kommunikationsleistung im Vergleich zur Rechenleistung zu gering ist. Damit machen sich Operationen, die viel Kommunikation erfordern besonders stark in einer Erhöhung der Rechenzeit bei der Verwendung vieler Prozessoren bemerkbar.

Vergleich von Druckkorrektur Verfahren und gekoppelten Mehrgitterverfahren

Um einen wenigstens einigermaßen fairen Vergleich durchzuführen, werden die im Kapitel 5 abgeschätzten numerischen Kosten der Verfahren als Ausgangspunkt genommen. In den Tabellen 9.20 und 9.21 werden das Druckkorrektur Verfahren mit GMRES(10) zur Lösung des Schur–Komplement Systems (3.11) für den Druck und das gekoppelte Mehrgitterverfahren mit dem Vanka Glätter unter Verwendung des V–, F– und W–Zyklus miteinander verglichen. Es wurden Testserien durchgeführt, bei denen zum einen relativ wenig und zum anderen viel numerische Arbeit in die Approximation der Lösung der linearen Probleme (3.1) investiert wurde.

Beim Druckkorrektur Verfahren mußte die approximierte Lösung des linearen Problems gedämpft werden (mit dem Faktor 0.3 beziehungsweise 0.5), um überhaupt Konvergenz zu erzielen. Der Glätter im Mehrgitterverfahren für die Geschwindigkeitsunbekannten wurde auf allen Leveln mit dem Faktor 0.5 gedämpft. Es traten auch Konvergenzprobleme auf, wenn man beim Mehrgitterverfahren zur Berechnung von $A_{\gamma,j}^{-1} = A_{1,1}^{-1}$ die Schrittweitenkontrolle (3.17) anwandte. Der kritische Punkt beim Druckkorrektur Verfahren ist die Lösung des Schur–Komplement Systems (3.11) für den Druck. Hierbei zeigte sich, daß das Verfahren GMRES(10) bei der Glättung der hochfrequenten Fehleranteile keine Probleme hat, bei der Glättung der niedrigfrequenten Anteile jedoch fast stagniert. Dies führte zu einer sehr langsamen Iterationsgeschwindigkeit der nichtlinearen Iteration, die man in den Tabellen 9.20 und 9.21 an Hand der hohen Iterationszahlen erkennen kann. Nach spätestens 1000 Schritten wurden die Iterationen abgebrochen. Die Anwendung des Mehrgitterverfahrens auf die Druckunbekannten zur Lösung des Schur–Komplement Systems (3.11) führte immer zur Divergenz des Verfahrens. Die dort verwendete Matrix C^{-1} ist eine zu schlechte Approximation an A^{-1}, als daß das Richardson Verfahren (3.10) konvergiert.

Die Tabellen 9.20 und 9.21 zeigen deutlich, daß das gekoppelte Mehrgitterverfahren zur Lösung der linearen Systeme, die bei der nichtlinearen Iteration der diskretisierten stationären Navier–Stokes Gleichungen auftreten, wesentlich besser geeignet ist. Aber auch hier gibt es zum

Löser	M	Gl.	Kost.	Lev.	Pr.	It.	Zeit	Kom.	Grg.
GMRES(10)	15	3	7607	4	1	1000	301000	0.0	0.1
Vanka–V	5	5	7340	4	1	131/652	36700	0.0	1.2
Vanka–F	5	4	8080	4	1	60/297	17900	0.0	2.7
Vanka–W	5	3	7170	4	1	34/170	9180	0.0	9.7
GMRES(10)	15	3	7607	4	8	1000	55100	9.1	1.8
Vanka–V	5	5	7340	4	8	142/706	6930	3.8	7.4
Vanka–F	5	4	8080	4	8	71/350	4760	11.1	28.9
Vanka–W	5	3	7170	4	8	53/264	3700	12.9	40.6
GMRES(10)	15	3	7607	5	16	1000	112000	8.4	0.4
Vanka–V	5	5	7340	5	16	355/1721	34100	3.1	4.2
Vanka–F	5	4	8080	5	16	151/754	18000	8.0	16.9
Vanka–W	5	3	7170	5	16	99/495	15300	15.5	41.8
GMRES(10)	50	3	24302	4	1	420	394000	0.0	0.1
Vanka–V	10	8	22360	4	1	46/450	37200	0.0	0.8
Vanka–F	10	6	22987	4	1	22/209	17400	0.0	1.9
Vanka–W	10	5	22020	4	1	14/131	10300	0.0	6.8
GMRES(10)	50	3	24302	4	8	452	75300	9.4	1.8
Vanka–V	10	8	22360	4	8	48/469	6620	3.6	5.1
Vanka–F	10	6	22987	4	8	26/251	4340	9.8	22.8
Vanka–W	10	5	22020	4	8	20/191	3490	11.4	31.6
GMRES(10)	50	3	24302	5	16	1000	338000	8.5	0.3
Vanka–V	10	8	22360	5	16	162/1606	44800	2.6	2.9
Vanka–F	10	6	22987	5	16	62/611	19400	6.8	12.8
Vanka–W	10	5	22020	5	16	36/343	13600	13.1	32.1

Tabelle 9.20: Die Lösung des Benchmarkproblems 9.2.1 mit verschiedenen Lösern und Startwert $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$.

Teil beträchtliche Unterschiede zwischen V–, F– und W–Zyklus. Bei der schlechten Startwert $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$ der nichtlinearen Iteration, Tabelle 9.20, ist der W–Zyklus mit einigem Abstand zum F–Zyklus und mit großem Abstand zum V–Zyklus die schnellste Variante des gekoppelten Mehrgitterverfahrens. Die Differenz in den Rechenzeiten der verschiedenen Zyklenarten verringert sich bei einem besseren Startwert, siehe Tabelle 9.21. Unverändert benötigt man mit dem V–Zyklus die längste Zeit unter den gekoppelten Mehrgitterverfahren. Weiterhin fällt auf, daß sich die Überlegenheit des W–Zyklus bei wachsender Anzahl von Prozessoren verringert. Bei den Rechnungen auf 8 und 16 Prozessoren, Tabelle 9.21, sind der F– und W–Zyklus ungefähr gleich schnell. Auch der V–Zyklus gewinnt gegenüber dem W–Zyklus. Nimmt man beispielsweise die Rechnungen von Tabelle 9.21, Level 4, oberer Teil, so braucht man mit dem V–Zyklus auf einem Prozessor etwa 3.23mal so lange wie mit dem W–Zyklus, auf 8 Prozessoren aber nur noch 1.395mal so lange. Diese vergleichsweise Verschlechterung des W–Zyklus liegt an der wesentlich besseren Skalierung der V– und F–Zyklen auf Parallelrechnern. Die Frage der Skalierung wird weiter unten in diesem Abschnitt näher untersucht, siehe Seite 157.

An dieser Stelle sei noch vermerkt, daß sich die angegebenen Kosten der Lösungsmethoden, trotz der Vernachlässigung der Kosten für die Lösung der Grobgittersysteme, in den Rechenzeiten auf einem Prozessor relativ gut widerspiegeln. In Tabelle 9.20 ist das Verhältnis der Kosten von GMRES(10), Vanka–V, Vanka–F und Vanka–W wie 1 : 0.96 : 1.06 : 0.94 in den Rechnungen

Löser	M	Gl.	Kost.	Lev.	Pr.	It.	Zeit	Kom.	Grg.
GMRES(10)	15	3	7607	4	1	415	131000	0.0	0.1
Vanka–V	5	5	7340	4	1	55/271	15400	0.0	1.0
Vanka–F	5	4	8080	4	1	24/118	7280	0.0	4.4
Vanka–W	5	3	7170	4	1	18/88	4770	0.0	8.6
GMRES(10)	15	3	7607	4	8	430	21300	6.7	1.7
Vanka–V	5	5	7340	4	8	57/279	2790	4.5	7.1
Vanka–F	5	4	8080	4	8	31/153	2000	10.3	26.5
Vanka–W	5	3	7170	4	8	29/141	2000	13.6	40.1
GMRES(10)	15	3	7607	5	16	307	34500	7.8	0.3
Vanka–V	5	5	7340	5	16	45/222	4240	3.0	2.8
Vanka–F	5	4	8080	5	16	31/151	3600	7.8	15.5
Vanka–W	5	3	7170	5	16	24/166	3520	15.4	40.0
GMRES(10)	50	3	24302	4	1	142	143000	0.0	0.1
Vanka–V	10	8	22360	4	1	19/181	15200	0.0	0.7
Vanka–F	10	6	22987	4	1	11/96	8140	0.0	3.1
Vanka–W	10	5	22020	4	1	9/73	5750	0.0	5.8
GMRES(10)	50	3	24302	4	8	151	25900	10.7	1.7
Vanka–V	10	8	22360	4	8	21/200	2840	3.6	4.7
Vanka–F	10	6	22987	4	8	12/105	1720	7.7	16.6
Vanka–W	10	5	22020	4	8	11/97	1790	11.9	31.2
GMRES(10)	50	3	24302	5	16	160	54400	8.0	0.2
Vanka–V	10	8	22360	5	16	22/208	5770	2.7	2.2
Vanka–F	10	6	22987	5	16	12/106	3330	6.5	10.8
Vanka–W	10	5	22020	5	16	9/82	3240	13.1	31.1

Tabelle 9.21: Die Lösung des Benchmarkproblems 9.2.1 mit verschiedenen Lösern, Startwert ist die interpolierte Lösung des nächsttieferen Levels.

mit wenig Aufwand zur Lösung des linearen Systems und 1 : 0.92 : 0.95 : 0.91 in den Rechnungen mit hohem Aufwand. Das Verhältnis der Zeiten für eine nichtlineare Iteration ist 1 : 0.94 : 1 : 0.90 beziehungsweise 1 : 0.88 : 0.89 : 0.84. Da bei den gekoppelten Mehrgitterverfahren in den letzten nichtlinearen Iterationen oft nicht die vorgegebene Anzahl der Mehrgitterzyklen ausgeführt wurde, haben wir zur Berechnung der angegebenen Verhältnisse die Anzahl der nichtlinearen Iterationen als Quotient der Gesamtzahl der Mehrgitterzyklen und M definiert.

Das gekoppelte Mehrgitterverfahren mit Vanka Glätter bei unterschiedlicher Anzahl der Glättungsiterationen

Wie bereits bei der Beschreibung des gekoppelten Mehrgitterverfahrens klar wird, gibt es eine Anzahl von Parametern, die man wählen kann oder muß. Die optimale Wahl der einzelnen Parameter kann vom konkreten Problem abhängen.

Als ein wichtiger Parameter erscheint uns die Anzahl der Glättungsiterationen. In den Tabellen 9.22 und 9.23 sind Ergebnisse zweier Testserien dokumentiert, wiederum mit unterschiedlichen Startwerten für die nichtlineare Iteration. Zunächst erkennt man, daß mit wachsender Anzahl von Glättungsiterationen die benötigte Anzahl der nichtlinearen Iterationen und damit der Mehrgitterzyklen zurückgeht. Bei den Rechnungen auf dem seriellen Rechner macht sich

Löser	M	Gl.	Kost.	Lev.	Pr.	It.	Zeit	Kom.	Grg.
Vanka–W	10	2	10500	4	4	28/275	5350	11.6	40.2
Vanka–W	10	3	14340	4	4	23/216	5090	10.0	33.0
Vanka–W	10	4	18180	4	4	22/213	5960	9.0	29.1
Vanka–W	10	5	22020	4	4	18/170	5440	8.3	25.1
Vanka–W	10	10	41220	4	4	11/101	5310	6.0	15.0
Vanka–W	10	2	10500	5	1	43/420	66500	0.0	4.6
Vanka–W	10	3	14340	5	1	35/340	72400	0.0	4.2
Vanka–W	10	4	18180	5	1	30/288	76300	0.0	2.8
Vanka–W	10	5	22020	5	1	24/233	74600	0.0	2.3
Vanka–W	10	10	41220	5	1	17/154	90000	0.0	1.3
Vanka–W	10	2	10500	5	16	64/627	16100	17.3	49.3
Vanka–W	10	3	14340	5	16	50/489	14900	15.5	41.9
Vanka–W	10	4	18180	5	16	45/441	15500	14.3	36.9
Vanka–W	10	5	22020	5	16	36/343	13600	13.1	32.1
Vanka–W	10	10	41220	5	16	20/190	12000	10.4	20.9

Tabelle 9.22: Benchmarkproblem 9.2.1, Lösungsverhalten des gekoppelten Mehrgitter Verfahrens in Abhängigkeit von der Anzahl der Glättungsiterationen, Startwert $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$.

Löser	M	Gl.	Kost.	Lev.	Pr.	It.	Zeit	Kom.	Grg.
Vanka–W	10	2	10500	4	4	15/140	2700	11.2	39.5
Vanka–W	10	3	14340	4	4	13/119	2790	9.7	32.5
Vanka–W	10	4	18180	4	4	11/104	2870	8.7	27.8
Vanka–W	10	5	22020	4	4	9/78	2460	7.7	23.3
Vanka–W	10	10	41220	4	4	7/54	2820	6.0	13.8
Vanka–W	10	2	10500	5	1	13/121	19100	0.0	5.6
Vanka–W	10	3	14340	5	1	10/88	18700	0.0	4.2
Vanka–W	10	4	18180	5	1	8/73	19300	0.0	3.3
Vanka–W	10	5	22020	5	1	7/61	19400	0.0	2.7
Vanka–W	10	10	41220	5	1	5/35	20400	0.0	1.4
Vanka–W	10	2	10500	5	16	17/166	4180	17.1	47.7
Vanka–W	10	3	14340	5	16	14/129	3860	15.3	40.3
Vanka–W	10	4	18180	5	16	11/98	3400	14.0	34.8
Vanka–W	10	5	22020	5	16	9/82	3240	13.1	31.1
Vanka–W	10	10	41220	5	16	6/49	3100	10.4	20.0

Tabelle 9.23: Benchmarkproblem 9.2.1, Lösungsverhalten des gekoppelten Mehrgitter Verfahrens in Abhängigkeit von der Anzahl der Glättungsiterationen, Startwert ist die interpolierte Lösung des nächsttieferen Levels.

dies jedoch nicht in einer Verringerung der Rechenzeiten bemerkbar, ganz im Gegenteil. Die Testergebnisse auf 4 Prozessoren lassen keine klare Tendenz erkennen. Auffällig sind die langen Rechenzeiten mit 4 Glättungsiterationen. Das beste Verhalten zeigt sich einmal mit 3 und das andere Mal mit 5 Glättungsiterationen. Auf 16 Prozessoren ist die Sachlage jedoch wieder eindeutig. Je mehr Glättungsiterationen man nutzt, desto weniger Zeit benötigt man. Das Verhalten des Algorithmus in bezug auf die Anzahl der Glättungsiterationen hat sich beim Übergang von einem zu 16 Prozessoren umgekehrt. Die optimale Anzahl der Glättungsiterationen hängt von der Anzahl der Prozessoren ab. Im W–Zyklus geht bei dieser Strategie vor allem der Anteil der Rechenzeit zurück, die auf den unteren Leveln, insbesondere auf dem Grobgitter, benötigt wird. Wie bereits im Abschnitt 4.3 diskutiert, treten auf den unteren Leveln starke Verluste durch Kommunikation auf, die sich besonders beim W–Zyklus bemerkbar machen. Das Verhalten des W–Zyklus mit einem reduzierten Aufwand auf dem Grobgitter wird weiter unten untersucht, siehe Seite 155.

Der stabilisierte Vanka Glätter im gekoppelten Mehrgitter

Der stabilisierte Vanka Glätter wird von Schieweck in [Sch96], Bemerkung 8.1, zur Nutzung auf anisotropen Gittern vorgeschlagen. Bei anisotropen Gittern gibt es Elemente, deren Seitenlängen sich um mehrere Zehnerpotenzen unterscheiden. Obwohl man Gitter 1 nicht als anisotrop bezeichnen kann, zeigen die Tabellen 9.24 und 9.25, daß sich die Verwendung des stabilisierten Vanka Glätters im W–Zyklus des gekoppelten Mehrgitterverfahrens für Beispiel 9.2.1 lohnt. Die Rechenzeiten sind zum Teil wesentlich geringer als mit dem Vanka Glätter, bei etwa gleichem numerischen Aufwand. Analog zum Vanka Glätter erzielt man beim stabilisierten Vanka Glätter mit wenig Glättungsschritten die schnellsten Rechenzeiten auf nur einem Prozessor. Je mehr Prozessoren man verwendet, desto günstiger ist es, die Anzahl der Glättungsschritte zu erhöhen.

Glätter	Gl.	Kost.	Lev.	Pr.	It.	Zeit	Kom.	Grg.
Vanka	3	14340	4	1	19/176	9050	0.0	6.1
stab. Vanka	1	11380	4	1	20/190	5960	0.0	13.8
stab. Vanka	2	19940	4	1	15/143	7500	0.0	11.2
stab. Vanka	4	37060	4	1	11/101	8830	0.0	6.2
Vanka	3	14340	4	8	26/248	3460	13.6	40.9
stab. Vanka	1	11380	4	8	33/322	2810	11.4	41.7
stab. Vanka	2	19940	4	8	21/208	2360	9.4	30.5
stab. Vanka	4	37060	4	8	12/110	2030	7.2	20.2
Vanka	3	14340	5	16	50/489	14900	15.5	41.9
stab. Vanka	1	11380	5	16	58/576	10700	13.1	43.3
stab. Vanka	2	19940	5	16	40/390	10200	10.9	33.7
stab. Vanka	4	37060	5	16	21/208	8410	8.6	22.8

Tabelle 9.24: Benchmarkproblem 9.2.1, Vergleich der Glätter im gekoppelten Mehrgitter, $\mathsf{M} = 10$, W–Zyklus, Startwert $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$.

Der Einfluß der Schrittweitenkontrolle beim gekoppelten Mehrgitter

Bei allen bisher vorgestellten Testrechnungen mit den gekoppelten Mehrgitterverfahren wurde die Schrittweitenkontrolle (3.24) angewandt. Es stellt sich natürlich die Frage, inwieweit sich dieser Mehraufwand lohnt.

In den Tabellen 9.26 und 9.27 sind die Anzahl von Iterationen und Rechenzeiten für das gekoppelte Mehrgitterverfahren mit und ohne Schrittweitenkontrolle gegenübergestellt. Der Startwert der nichtlinearen Iteration für die Ergebnisse von Tabelle 9.26 war $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$ und für die Ergebnisse von Tabelle 9.27 die interpolierte Lösung auf dem nächsttieferen Level.

Glätter	Gl.	Kost.	Lev.	Pr.	It.	Zeit	Kom.	Grg.
Vanka	3	14340	4	1	10/89	4800	0.0	9.8
stab. Vanka	1	11380	4	1	11/102	3290	0.0	15.7
stab. Vanka	2	19940	4	1	9/82	4000	0.0	9.9
stab. Vanka	4	37060	4	1	7/68	4810	0.0	6.1
Vanka	3	14340	4	8	14/127	1750	13.3	39.8
stab. Vanka	1	11380	4	8	17/166	1420	11.4	40.3
stab. Vanka	2	19940	4	8	12/111	1330	9.3	29.9
stab. Vanka	4	37060	4	8	8/65	1210	7.3	19.7
Vanka	3	14340	5	16	14/129	3860	15.3	40.3
stab. Vanka	1	11380	5	16	18/175	3250	12.6	41.7
stab. Vanka	2	19940	5	16	11/101	2630	10.4	31.6
stab. Vanka	4	37060	5	16	6/50	2000	8.2	21.0

Tabelle 9.25: Benchmarkproblem 9.2.1, Vergleich der Glätter im gekoppelten Mehrgitter, $\mathsf{M} = 10$, W–Zyklus, Startwert ist interpolierte Lösung des nächsttieferen Levels.

In beiden Tabellen sieht man sofort, daß sich bei der Verwendung des V–Zyklus die Schrittweitenkontrolle nicht auszahlt. Bei den Rechnungen auf Level 4 geht zwar die Gesamtanzahl der Mehrgitterzyklen etwas zurück, aber der durch die Schrittweitenkontrolle verursachte Mehraufwand läßt letzten Endes die Rechenzeiten sogar leicht ansteigen. Auf Level 5 braucht man mit Schrittweitenkontrolle sogar eine höhere Anzahl von Mehrgitterzyklen als ohne, was sich in beträchtlich ansteigenden Rechenzeiten niederschlägt. Beim F–Zyklus lohnt sich der Einsatz der Schrittweitenkontrolle in den meisten Testrechnungen. Auf Level 5 zeigt sich deutlich der Einfluß des Startwertes der nichtlinearen Iteration auf das Verhalten des Verfahrens. Beim Startwert $(\mathbf{0}, 0)$ führt die Verwendung der Schrittweitenkontrolle zu einer Erhöhung der Anzahl der Iterationen und der Rechenzeit. Nimmt man dagegen die interpolierte Lösung des nächsttieferen Levels als Startwert, so resultiert die Schrittweitenkontrolle in einer deutlichen Reduktion der Iterationsanzahl sowie der Rechenzeit. Man spart etwa ein Viertel an Rechenzeit. Beim W–Zyklus lohnt sich die Verwendung der Schrittweitenkontrolle in jedem Testbeispiel deutlich. Die Einsparungen an Rechenzeit liegen zwischen 9 und 36 %.

Auch bei diesen Testrechnungen zeigt sich, daß selbst die nun verbesserten Rechenzeiten bei der Verwendung des V–Zyklus nicht mit den Rechenzeiten der F– und W–Zyklen mit (und sogar ohne) Schrittweitenkontrolle konkurrieren können.

Die beiden Varianten der Grobgitteriteration

Im Abschnitt 4.3 wurde diskutiert, daß es sich auf Parallelrechnern lohnen könnte, von der standardmäßigen Iteration auf dem Grobgitter abzuweichen. Anstatt die euklidische Norm des Residuums um einen vorgegebenen Faktor zu verringern, wird nur eine festgelegte Anzahl von Iterationsschritten ausgeführt, ohne die euklidische Norm des Residuums zu berechnen.

Wegen des hohen Anteils der Grobgitteriteration an der Gesamtzeit betrachten wir in diesem Abschnitt nur das gekoppelte Mehrgitterverfahren mit dem W–Zyklus. Wir vergleichen die Standardvariante, bei der die euklidische Norm des Residuums um den Faktor 0.1 verringert wird, mit Grobgitteriterationen, die $10, 20$ beziehungsweise 30 Iterationen ohne Normberechnung durchführen. Die Resultate dieses Vergleiches sind in Tabelle 9.28 zu finden. Die Startwert der nichtlinearen Iteration in diesen Rechnungen war $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$.

Löser	Swk.	M	Gl.	Lev.	Pr.	It.	Zeit	Zeit/Zyk.
Vanka–V	nein	10	8	4	4	48/465	12300	26.5
Vanka–V	ja	10	8	4	4	47/463	12700	27.4
Vanka–F	nein	10	4	4	4	40/394	8590	21.8
Vanka–F	ja	10	4	4	4	32/309	7000	22.7
Vanka–W	nein	10	3	4	4	35/344	8060	23.4
Vanka–W	ja	10	3	4	4	23/216	5090	23.6
Vanka–V	nein	10	8	5	16	119/1180	31900	27.0
Vanka–V	ja	10	8	5	16	162/1606	44800	27.9
Vanka–F	nein	10	4	5	16	89/878	20200	23.0
Vanka–F	ja	10	4	5	16	97/960	22900	23.9
Vanka–W	nein	10	3	5	16	65/642	19000	29.6
Vanka–W	ja	10	3	5	16	50/489	14900	30.5

Tabelle 9.26: Benchmarkproblem 9.2.1, Lösungsverhalten mit und ohne Schrittweitenkontrolle und Startwert $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$.

Löser	Swk.	M	Gl.	Lev.	Pr.	It.	Zeit	Zeit/Zyk.
Vanka–V	nein	10	8	4	4	21/205	5440	26.5
Vanka–V	ja	10	8	4	4	21/198	5550	28.0
Vanka–F	nein	10	4	4	4	18/174	3770	21.7
Vanka–F	ja	10	4	4	4	17/158	3570	22.6
Vanka–W	nein	10	3	4	4	17/161	3740	23.2
Vanka–W	ja	10	3	4	4	13/119	2790	23.4
Vanka–V	nein	10	8	5	16	17/164	4370	26.6
Vanka–V	ja	10	8	5	16	22/208	5770	27.7
Vanka–F	nein	10	4	5	16	20/192	4350	22.7
Vanka–F	ja	10	4	5	16	15/139	3240	23.2
Vanka–W	nein	10	3	5	16	16/150	4200	28.0
Vanka–W	ja	10	3	5	16	14/129	3860	29.9

Tabelle 9.27: Benchmarkproblem 9.2.1, Lösungsverhalten mit und ohne Schrittweitenkontrolle, Startwert ist die interpolierte Lösung des nächsttieferen Levels.

Die Ergebnisse von Tabelle 9.28 zeigen, daß sich das Verfahren wie erwartet verhält. Auf einem Prozessor braucht es die geringste Rechenzeit, falls man die Standarditeration auf dem Grobgitter durchführt oder eine feste, aber hohe Anzahl von Iterationsschritten vorgibt. Auf 4 Prozessoren ist man mit 20 beziehungsweise 30 vorgegebenen Iterationsschritten schon beträchtlich schneller als mit der Standarditeration. Auf 8 und 16 Prozessoren ist das Verfahren mit Standardgrobgitteriteration deutlich die langsamste Variante. Erreicht man auf 8 Prozessoren mit 20 und 30 Iterationen auf dem Grobgitter die besten Zeiten, so ist dies bei 16 Prozessoren mit 10 und 20 Iterationen der Fall.

Die Tendenz ist eindeutig. Je mehr Prozessoren man nutzt, desto weniger sollte man auf dem Grobgitter investieren. Die dabei sich erhöhende Anzahl der Mehrgitterzyklen wird durch die Zeiteinsparungen auf dem Grobgitter mehr als wett gemacht.

Löser	Grobgitter	M	Gl.	Lev.	Pr.	It.	Zeit	Kom.	Grg.
Vanka–W	Var. 1, Red. 0.1	10	3	4	1	19/176	9050	0.0	6.1
Vanka–W	Var. 2, 10 Schr.	10	3	4	1	26/247	12000	0.0	0.7
Vanka–W	Var. 2, 20 Schr.	10	3	4	1	21/203	9900	0.0	1.3
Vanka–W	Var. 2, 30 Schr.	10	3	4	1	19/184	9050	0.0	1.9
Vanka–W	Var. 1, Red. 0.1	10	3	4	4	23/216	5090	10.0	33.0
Vanka–W	Var. 2, 10 Schr.	10	3	4	4	35/341	5650	3.9	4.5
Vanka–W	Var. 2, 20 Schr.	10	3	4	4	27/260	4500	4.1	8.5
Vanka–W	Var. 2, 30 Schr.	10	3	4	4	25/235	4240	4.4	12.0
Vanka–W	Var. 1, Red. 0.1	10	3	4	8	26/248	3460	13.6	40.9
Vanka–W	Var. 2, 10 Schr.	10	3	4	8	37/359	3230	6.1	8.9
Vanka–W	Var. 2, 20 Schr.	10	3	4	8	31/299	2930	6.2	15.9
Vanka–W	Var. 2, 30 Schr.	10	3	4	8	29/278	2940	6.4	22.0
Vanka–W	Var. 1, Red. 0.1	10	3	5	16	50/489	14900	15.5	41.9
Vanka–W	Var. 2, 10 Schr.	10	3	5	16	68/673	13000	6.8	9.0
Vanka–W	Var. 2, 20 Schr.	10	3	5	16	63/621	12900	7.3	16.1
Vanka–W	Var. 2, 30 Schr.	10	3	5	16	60/590	13300	6.9	22.3

Tabelle 9.28: Benchmarkproblem 9.2.1, Lösungsverhalten mit verschiedenen Arten der Grobgitteriteration, Startwert $(\mathbf{u}^0, p^0) = (\mathbf{0}, 0)$.

Die Skalierung des gekoppelten Mehrgitterverfahrens

Eine wichtige Fragestellung besteht darin, wie sich die Algorithmen bei unterschiedlicher Anzahl von Prozessoren verhalten. Eine in der Theorie oft verwendete Größe, zur Beschreibung dieses Verhaltens, ist die parallele Effizienz eines Algorithmus für ein Problem

$$E_p := \frac{\text{Rechenzeit des schnellsten sequentiellen Algorithmus auf einem Prozessor}}{p \cdot \text{Rechenzeit bei der Nutzung des parallelen Algorithmus auf p Prozessoren}},$$

siehe Ortega und Voigt [OV85]. Ein Problem bei dieser Definition ist, daß man den Zähler im allgemeinen nicht kennt, da kein „schnellster“ Algorithmus zur Verfügung steht. Oft wird statt des „schnellsten“ der eigene Algorithmus auf einem Prozessor zu Vergleichszwecken genommen. Ein weiteres Problem besteht in der Hardwareabhängigkeit von E_p. Bei der von uns verwendeten Parallelarchitektur ist die Kommunikation im Vergleich zur Rechenleistung der Prozessoren langsam. Deswegen machen sich Kommunikationen besonders stark in Effizienzeinbußen bemerkbar.

Als eine weitere Problematik der obigen Definition empfinden wir, daß die Aufgaben, die man sinnvollerweise auf einem und auf p Prozessoren rechnet, unterschiedlich sind. Zum einen werden für ein festes Problem die Teilprobleme auf den einzelnen Prozessoren mit Erhöhung der Anzahl der Prozessoren immer kleiner. Das führt natürlicherweise zu einer Erhöhung der Kommunikation und zu schlechten Effizienzen. Zum anderen, sind Probleme, zu deren Lösung man p Prozessoren verwendet, oft zu groß, um auf einem Prozessor gelöst zu werden. Zumindest im Bereich der Strömungsmechanik ist die Nutzung vieler Prozessoren, ohne diese mit Daten auszulasten, praxisfern. Die Parallelrechner dienen in der Strömungsmechanik nicht nur der Beschleunigung von Rechnungen, sondern auch der Verwaltung immer größerer Datenmengen.

Aus diesem Grunde haben wir zur Untersuchung der Skalierung des gekoppelten Mehrgitterverfahrens die Strategie gewählt, daß wir, von einem Prozessor angefangen, die Problemgröße und die Anzahl der Prozessoren, im Rahmen unserer Möglichkeiten, jeweils um den Faktor 4 erhöht haben. Damit waren Konfigurationen von einem, 4 und 16 Prozessoren auf dem von uns verwendeten Parsytec GC jeweils mit Daten ausgelastet. Wir definieren einen Effizienzindex $E(m,n)$ zwischen m und n Prozessoren als

$$E(m,n) := \frac{T_m}{T_n}\,, \quad m < n,$$

wobei T_m und T_n die jeweiligen Rechenzeiten sind. Die von uns erzielten Ergebnisse sind in Tabelle 9.29 dokumentiert. Es wurden jeweils 10 nichtlineare Iterationen mit je 10 Mehrgitterzyklen durchgeführt.

Zuerst fällt auf, daß beim V–Zyklus keine Verluste auftreten, daß im Gegenteil sogar die Effizienz ein wenig größer als 1 ist. Durch das höhere Level geht der Anteil der kommunikationsintensiven Grobgitteriteration an der Gesamtrechenzeit zurück. Der F–Zyklus zeigt ein ähnliches Verhalten. Die Erhöhung der Rechenzeit durch die Kommunikation ist gering und der Anteil der Zeit für die Grobgitteriteration an der Gesamtrechenzeit nimmt ab. Die nahezu konstante Zeit für einen V–Zyklus und das leichte Anwachsen der Zeit für einen F–Zyklus ist auch in den Tabellen 9.26 und 9.27 zu beobachten. Die sehr gute Effizienz ist aber auch eine Folge davon, daß das verwendete nichtkonforme P_1/P_0–Finite Element keine Crosspoints besitzt, vergleiche Bemerkung 4.1.3. Daß bei der Verwendung des W–Zyklus Effizienzverluste auftreten, wurde bereits im Abschnitt 4.3 diskutiert. Bei dem von uns gewählten konkreten Beispiel erhalten wir Effizienzen von $E(1,4) = 0.85, E(1,16) = 0.68, E(4,16) = 0.80$. Bei einer Vervierfachung der Anzahl der Prozessoren vervielfacht sich die Rechenzeit etwa um den Faktor 1.2, bei 16–mal mehr Prozessoren um den Faktor 1.5. Bei den Ergebnissen aus den Tabellen 9.26 und 9.27 vervielfacht sich die Rechenzeit pro W–Zyklus beim Übergang von 4 zu 16 Prozessoren etwa um den Faktor 1.2 bis 1.3.

Löser	M	Gl.	Lev.	Pr.	It.	Zeit	Kom.	Grg.
Vanka–V	10	3	3	1	10	1290	0.0	11.0
Vanka–V	10	3	4	4	10	1273	3.1	7.2
Vanka–V	10	3	5	16	10	1267	3.6	4.0
Vanka–F	10	3	3	1	10	1810	0.0	23.8
Vanka–F	10	3	4	4	10	1860	7.2	20.8
Vanka–F	10	3	5	16	10	1880	8.4	16.6
Vanka–W	10	3	3	1	10	1990	0.0	28.6
Vanka–W	10	3	4	4	10	2330	10.2	32.2
Vanka–W	10	3	5	16	10	2910	14.9	39.2

Tabelle 9.29: Benchmarkproblem 9.2.1, Skalierung der gekoppelten Mehrgitterverfahren bei voll ausgelasteten Prozessoren.

Auswertung der numerischen Tests

Bemerkung 9.3.1 Auswertung der numerischen Tests.

1. **Druckkorrektur Verfahren – gekoppeltes Mehrgitterverfahren.** Mit dem gekoppelten Mehrgitterverfahren bereitete die Lösung der auftretenden linearen Probleme keine Schwierigkeiten. Die Druckkorrektur Verfahren in der von uns gewählten Form haben sich als ungeeignet erwiesen.

2. **Gekoppeltes Mehrgitterverfahren.** Bei dem von uns gewählten Testbeispiel hat sich der W–Zyklus für schlechte Startwerte und auf wenigen Prozessoren dem F– und V–Zyklus

deutlich überlegen gezeigt. Für bessere Startwerte und einer erhöhten Anzahl von Prozessoren waren W– und F–Zyklus nahezu gleich gut, nur der V–Zyklus fiel ab. Die Unterlegenheit des V–Zyklus gegenüber dem W–Zyklus in bezug auf die Gesamtrechenzeit nimmt ebenfalls mit wachsender Anzahl von Prozessoren ab.

3. **Schrittweitenkontrolle.** Beim F– und W–Zyklus brachte die Verwendung der Schrittweitenkontrolle teilweise erhebliche Effizienzverbesserungen. Beim V–Zyklus, der im allgemeinen schlechtere Korrekturen als der W–Zyklus berechnet, konnten mit Schrittweitenkontrolle keine signifikanten Verbesserungen erzielt werden.

4. **Vanka und stabilisierter Vanka Glätter.** Der stabilisierte Vanka Glätter hat sich in bezug auf die Rechenzeit dem Vanka Glätter überlegen gezeigt.

5. **Gekoppeltes Mehrgitterverfahren mit W–Zyklus auf unterschiedlicher Anzahl von Prozessoren.** Die optimale Wahl verschiedener Parameter im gekoppelten Mehrgitterverfahren mit W–Zyklus ist von der Anzahl der Prozessoren abhängig. Während auf einem seriellen Computer wenig Glättungsiterationen und die Standardvariante der Grobgitteriteration die schnellsten Rechenzeiten nach sich ziehen, sind bei wachsender Anzahl von Prozessoren viele Glättungsiterationen und eine geringe Iterationsanzahl auf dem Grobgitter wesentlich günstiger.

6. **Skalierung des gekoppelten Mehrgitterverfahrens.** Bei einer Vervielfachung der Anzahl der Prozessoren um den Faktor n und gleichzeitiger Vervielfachung der Problemgröße um den Faktor n zeigte der V–Zyklus eine nahezu konstante Zeit pro Mehrgitterzyklus. Die Rechenzeit eines F–Zyklus nahm nur leicht zu, währenddessen sich die Zeit für einen W–Zyklus merklich erhöhte. Bei dem von uns gewählten Parametern war bei $n = 4$ eine Erhöhung um den Faktor 1.2 bis 1.3 zu beobachten.

Kapitel 10

Numerische Ergebnisse für die instationären Navier–Stokes Gleichungen

In diesem Kapitel werden numerische Ergebnisse für zwei Benchmarkprobleme des Schwerpunktprogrammes „Flow simulation with high performance computers“ der DFG präsentiert, die unter Verwendung der nichtkonformen P_1/P_0–Finite Elemente Diskretisierung und der im Abschnitt 2.5 vorgestellten Zeitdiskretisierungen erzielt wurden. Den Abschluß des Kapitels bildet ein Vergleich einiger Löser für die Sattelpunktprobleme, die bei der Linearisierung der diskretisierten instationären Navier–Stokes Gleichungen auftreten.

10.1 Strömung durch einen Kanal um einen Zylinder mit stationärer Einströmung, ein Benchmarkproblem

Beispiel 10.1.1 In diesem Beispiel wird eine Strömung durch den in Abbildung 9.7 skizzierten Kanal betrachtet die durch die instationären Navier–Stokes Gleichungen (2.10) beschrieben wird. Die Einströmung ist zeitunabhängig. Es bildet sich eine periodische Strömung, eine sogenannte Kármánsche Wirbelstraße, aus, siehe zum Beispiel [Wie65].

Das Problem lautet

$$\begin{array}{rcll} \mathbf{u}_t - 10^{-3}\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &=& \mathbf{0} & \text{in } \Omega\times[t_0,t_0+T] \\ \nabla\cdot\mathbf{u} &=& 0 & \text{in } \Omega\times[t_0,t_0+T] \\ \mathbf{u}(0,y)=\mathbf{u}(2.2,y) &=& 0.41^{-2}\,(6y(0.41-y),0) & y\in[0,0.41], t\in[t_0,t_0+T] \\ \mathbf{u}(x,0)=\mathbf{u}(x,0.41) &=& (0,0) & x\in[0,2.2], t\in[t_0,t_0+T] \\ \mathbf{u}(S) &=& (0,0) & t\in[t_0,t_0+T] \\ \mathbf{u}(t_0) &=& \mathbf{u}_0. & \end{array}$$

Die Anfangsbedingung $\mathbf{u}_0$ ist ein Stömungszustand, an dem der Auftriebsbeiwert c_a, siehe (9.2), ein Maximum besitzt. Wir bezeichnen den Zeitpunkt des Anfangszustandes mit t_0 und die Länge einer Periode mit T. Als Benchmarkkennwerte hat man die Beiwerte c_w und c_a als Funktionen der Zeit, ihre Maxima und die Druckdifferenz nach einer halben Periode

$$\Delta p := p(0.15, 0.2, t_0+T/2) - p(0.25, 0.2, t_0+T/2)$$

zu berechnen. Die charakteristische Geschwindigkeit bei diesem Beispiel ist $U = 1m/s$. Das ist das Maximum der parabelförmigen Einströmung. Damit ist die Reynoldszahl dieser Strömung

$Re = 100$. Die charakteristische Zeit ist die Länge einer Periode T. Ein weiterer Benchmarkkennwert ist die in der Einleitung zu Kapitel 2 eingeführte Strouhalzahl

$$St = \frac{L}{UT} = \frac{0.1}{T}.$$

Vergleich der Zeitdiskretisierungen

Die numerischen Ergebnisse des stationären Benchmarkproblems, Abschnitt 9.2, haben gezeigt, daß man mit Gitter 1, Samarskij Upwind und Druckseparation die besten Ergebnisse erzielt. Diese Bestandteile der Ortsdiskretisierung haben wir auch in diesem Abschnitt genutzt, wobei bei der Druckseparation abermals Variante 4 genommen wurde.

Zunächst wollen wir die Ergebnisse bei Verwendung verschiedener Zeitdiskretisierungen miteinander vergleichen. Die getesteten Zeitdiskretisierungen sind 1–Schritt–θ–Methoden mit $\theta = 1$ (implizites Euler Verfahren), $\theta = 1/2$ (modifiziertes Crank–Nicolson Verfahren) und BDF(2). Wir führten diese Tests mit einer Zeitschrittweite von $\tau = 0.005$ aus.

Verfahren	Level	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	Δp	St
impl. Euler	3	79	3.0860	0.3518	2.2804	0.2532
mod. Crank–Nic.	3	74	3.2968	0.8158	2.3887	0.2703
BDF(2)	3	77	3.2781	0.6060	2.3760	0.2597
impl. Euler	4	73	3.0989	0.4860	2.3259	0.2740
mod. Crank–Nic.	4	69	3.2624	0.9214	2.4282	0.2899
BDF(2)	4	71	3.2638	0.8493	2.4205	0.2817
impl. Euler	5	70	3.0953	0.5461	2.3668	0.2857
mod. Crank–Nic.	5	67	3.2377	0.9647	2.4615	0.2985
BDF(2)	5	68	3.2500	0.9550	2.4631	0.2941
Benchmarkintervall			[3.2200, 3.2400]	[0.9800, 1.020]	[2.4600, 2.5000]	[0.2950, 0.3050]

Tabelle 10.1: Beispiel 10.1.1, Ergebnisse mit verschiedenen Zeitdiskretisierungen und $\tau = 0.005$.

Die Ergebnisse der Testrechnungen sind in Tabelle 10.1 dokumentiert. Die dritte Spalte gibt die Anzahl der Zeitschritte pro Periode an. Mit allen Zeitdiskretisierungen konnte das wesentliche Merkmal der Strömung, nämlich ihre Periodizität, erfaßt werden. Beim Blick auf die zu berechnenden Kennwerte zeigen sich allerdings deutliche Unterschiede. Unabhängig von der Verfeinerung im Ort erhält man mit dem impliziten Euler Verfahren die mit Abstand schlechtesten Ergebnisse. Eine Zeitdiskretisierung erster Ordnung ist nicht ausreichend, um die Benchmarkkennwerte mit guter Genauigkeit zu berechnen. Die besten Resultate erhält man mit dem modifizierten Crank–Nicolson Verfahren. Für die Level 3 und 4 der Ortsdiskretisierung sind vor allem der maximale c_a–Wert und die Strouhalzahl St bei diesem Verfahren wesentlich besser als bei BDF(2). Das Schwingen der Strömung ist viel stärker ausgeprägt. Erst auf Level 5 werden die Unterschiede zwischen den beiden Verfahren geringer, wobei die mit dem modifizierten Crank–Nicolson Verfahren erzielten Ergebnisse immer noch besser sind.

Bei den Testrechnungen hat sich gezeigt, daß man für die höhere Genauigkeit des modifizierten Crank–Nicolson Verfahrens bei der Lösung der linearen Probleme zu bezahlen hat. Die linearen Systeme in den Testrechnungen von Tabelle 10.1 wurden mit einem gekoppelten Mehrgitterverfahren mit W–Zyklus, Vanka Glätter, $\mathsf{M} = 5$, $\lambda_1 = \lambda_2 = 3$ gelöst. In Tabelle 10.2 ist der durchschnittliche Aufwand pro Zeitschritt zu sehen. In der vierten Spalte befinden sich die

Anzahl der nichtlinearen Iterationen und die Gesamtzahl der Mehrgitterzyklen, die nötig waren, um eine kleinere euklidische Norm des Residuums als 10^{-5} zu erhalten. So ist der Mehraufwand beim modifizierten Crank–Nicolson Verfahren im Vergleich zu BDF(2) bezüglich der Rechenzeit ungefähr ein Drittel.

Verfahren	Level	Proz.	It./Schritt	t/Schritt
impl. Euler	4	8	6/24	242
mod. Crank–Nic.	4	8	11/52	491
BDF(2)	4	8	9/36	369
impl.Euler	5	16	7/26	559
mod. Crank–Nic.	5	16	11/46	969
BDF(2)	5	16	9/36	763

Tabelle 10.2: Beispiel 10.1.1, Durchschnittlicher Aufwand der verschiedenen Zeitdiskretisierungen pro Zeitschritt.

Die Benchmarkkennwerte als Funktionen der Zeit sind in Abbildung 10.1 präsentiert.

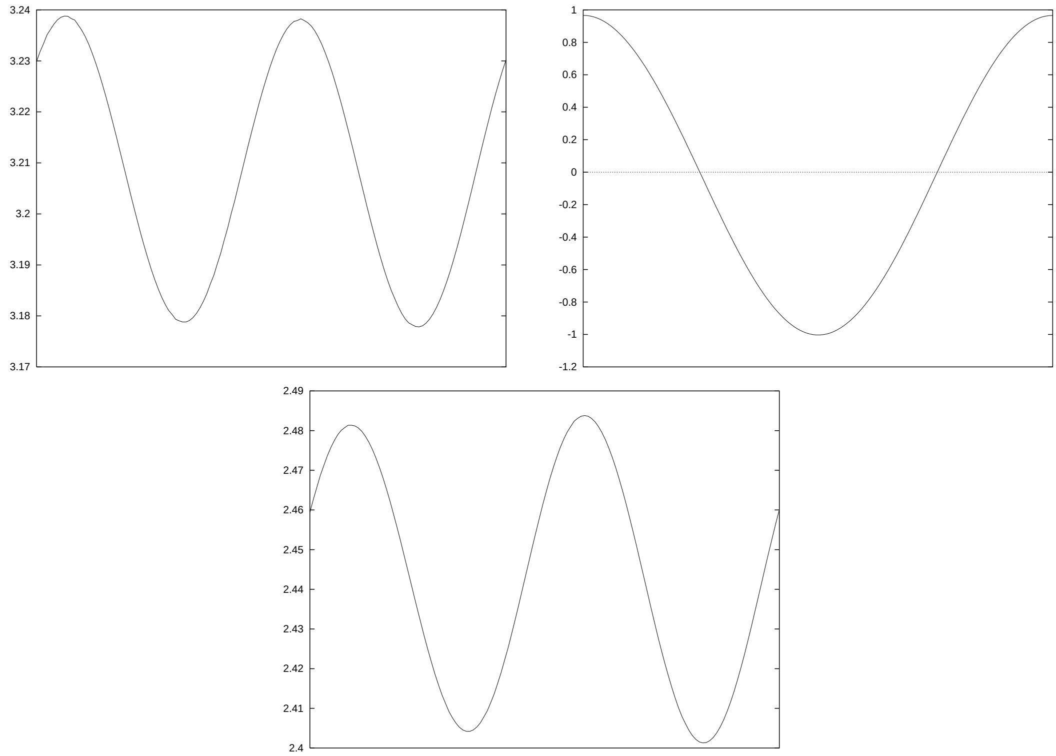


Abbildung 10.1: $c_w(t)$ (links oben), $c_a(t)$ (rechts oben) und $\Delta p(t)$ (unten), eine Periode, modifiziertes Crank–Nicolson Verfahren, Level 5, $\tau = 0.0025$.

Ergebnisse bei Verwendung verschiedener Zeitschrittweiten

In diesem Abschnitt werden Ergebnisse vorgestellt, die mit verschiedenen Zeitschrittweiten τ erzielt wurden. Aus Tabelle 10.1 kann man entnehmen, daß die berechneten Werte für die Benchmarkkennwerte stark von der Feinheit der Ortsdiskretisierung abhängen. Die Abhängigkeit der

Resultate von der Zeitschrittweite ist in Tabelle 10.3 dokumentiert. Einzig beim Maximum des Widerstandsbeiwertes $c_{w_{max}}$ sind mit einer Verkleinerung der Zeitschrittweite wesentliche Verbesserungen festzustellen. Beim BDF(2) Verfahren wird das Maximum des Auftriebsbeiwertes $c_{a_{max}}$ beim Übergang von $\tau = 0.01$ zu $\tau = 0.005$ gedämpft, was bezüglich des Benchmarkintervalls zu einer Verschlechterung der Ergebnisse führt. Alles in allem ist der Einfluß der Zeitschrittweite auf die Resultate wesentlich geringer als der Einfluß der Feinheit der Ortsdiskretisierung. Der Diskretisierungsfehler im Ort dominiert den in der Zeit.

Verfahren	Level	τ	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	Δp	St
mod. Crank–Nic.	3	0.01	38	3.3223	0.8108	2.3856	0.2632
mod. Crank–Nic.	3	0.005	74	3.2968	0.8158	2.3887	0.2703
mod. Crank–Nic.	3	0.0025	146	3.2526	0.8129	2.3920	0.2740
mod. Crank–Nic.	4	0.01	35	3.2714	0.9248	2.4273	0.2857
mod. Crank–Nic.	4	0.005	69	3.2624	0.9214	2.4282	0.2899
mod. Crank–Nic.	4	0.0025	138	3.2495	0.9188	2.4280	0.2899
mod. Crank–Nic.	5	0.01	34	3.2419	0.9689	2.4673	0.2941
mod. Crank–Nic.	5	0.005	67	3.2377	0.9647	2.4615	0.2985
mod. Crank–Nic.	5	0.0025	135	3.2370	0.9624	2.4610	0.2963
BDF(2)	3	0.01	39	3.3131	0.6417	2.3818	0.2564
BDF(2)	3	0.005	77	3.2781	0.6060	2.3760	0.2597
BDF(2)	3	0.0025	155	3.2421	0.6129	2.3825	0.2581
BDF(2)	4	0.01	35	3.3000	0.9060	2.4351	0.2857
BDF(2)	4	0.005	71	3.2638	0.8493	2.4205	0.2817
BDF(2)	4	0.0025	142	3.2442	0.8422	2.4195	0.2817
BDF(2)	5	0.01	33	3.2813	1.0134	2.4771	0.3030
BDF(2)	5	0.005	68	3.2500	0.9550	2.4631	0.2941
BDF(2)	5	0.0025	136	3.2409	0.9415	2.4539	0.2941
Benchmarkintervall				[3.2200, 3.2400]	[0.9800, 1.020]	[2.4600, 2.5000]	[0.2950, 0.3050]

Tabelle 10.3: Beispiel 10.1.1, Ergebnisse bei Verwendung verschiedener Zeitschrittweiten.

Vergleicht man das modifizierte Crank–Nicolson Verfahren und BDF(2) für verschiedene Zeitschrittweiten an Hand von Tabelle 10.3, so erhält man mit dem modifizierten Crank–Nicolson Verfahren immer die besseren Ergebnisse. Vor allem der maximale c_a–Wert und die Strouhalzahl werden genauer berechnet. Mit steigender Verfeinerung der Ortsgitters nähern sich die mit BDF(2) erzielten Resultate denen des modifizierten Crank–Nicolson Verfahrens an, ohne diese jedoch zu erreichen.

Der Einfluß der Druckseparation

Ebenso wie beim stationären Benchmarkproblem 9.2.1 erhält man mit Verwendung der Druckseparation genauere Ergebnisse als ohne, vergleiche Tabellen 10.3 und 10.4 für BDF(2). Für das modifizierte Crank–Nicolson Verfahren sieht die Situation ähnlich aus. Anders als beim stationären Problem haben sich die Kosten für die Lösung der linearen Gleichungssysteme bei allen Varianten der Druckseparation drastisch erhöht. Mit Druckseparation stieg die Anzahl der benötigten nichtlinearen Iterationen, um das gestellte Abbruchkriterium zu erfüllen. Die Lösungszeit für einen Zeitschritt vervielfacht sich um den Faktor 2 bis 4 im Vergleich zur Diskretisierung ohne Druckseparation.

Level	τ	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	Δp	St
3	0.01	41	3.3692	0.5137	2.3499	0.2439
3	0.005	81	3.3477	0.5239	2.3599	0.2469
3	0.0025	159	3.3139	0.5642	2.3730	0.2516
4	0.01	36	3.3294	0.8594	2.4219	0.2778
4	0.005	73	3.2963	0.8103	2.4134	0.2740
4	0.0025	145	3.2769	0.8094	2.4090	0.2759
5	0.01	34	3.2948	1.0058	2.4749	0.2941
5	0.005	69	3.2616	0.9465	2.4562	0.2899
5	0.0025	137	3.2552	0.9383	2.4508	0.2920
Benchmarkintervall			[3.2200, 3.2400]	[0.9800, 1.020]	[2.4600, 2.5000]	[0.2950, 0.3050]

Tabelle 10.4: Beispiel 10.1.1, Ergebnisse mit BDF(2), ohne Druckseparation.

Rechnungen auf anderen Gittern

Es wurden einige Rechnungen auf den Gittern 2 und 3 durchgeführt, siehe Abbildung 9.8. Die Ergebnisse mit Gitter 2 sind in Tabelle 10.5 dokumentiert. Vergleicht man diese Resultate mit den entsprechenden auf Gitter 1, Tabelle 10.4, so erkennt man, daß vor allem der maximale Widerstandsbeiwert $c_{w_{max}}$ auf Gitter 2 deutlich schlechter berechnet wurde. Bei allen anderen Kennwerten sind nur geringe Unterschiede zwischen den beiden Gittern festzustellen. Die Kennwerte $c_{a_{max}}$ und Δp wurden auf Gitter 2 sogar etwas genauer berechnet. Die Verwendung des a priori um den Zylinder stark verfeinerten Gitters 3, Level 4, bringt keine Verbesserung der Ergebnisse im Vergleich zu Gitter 1, siehe Tabelle 10.6. Vor allem der Auftriebsbeiwert c_a und die Strouhalzahl St sind noch recht ungenau. Diese Werte sind sogar schlechter als auf Gitter 1, Level 4, obwohl man bei Gitter 3 doppelt soviel Unbekannte im Ort hat.

Level	τ	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	Δp	St
4	0.01	36	3.4426	0.8759	2.4357	0.2778
4	0.005	74	3.3945	0.8208	2.4230	0.2703
4	0.0025	147	3.3570	0.8119	2.4230	0.2721
Benchmarkintervall			[3.2200, 3.2400]	[0.9800, 1.020]	[2.4600, 2.500]	[0.2950, 0.3050]

Tabelle 10.5: Beispiel 10.1.1, Ergebnisse mit BDF(2) auf Gitter 2, ohne Druckseparation.

Bemerkung 10.1.2 Auswertung der numerischen Tests.

1. **Ortsdiskretisierung.** Mit der verwendeten nichtkonformen P_1/P_0–Finite Elemente Diskretisierung ist man in der Lage, die Genauigkeitsanforderungen an die Benchmarkkennwerte weitestgehend zu erfüllen. In den Testrechnungen brauchte man einerseits recht viele Freiheitsgrade, andererseits wurde keine a priori Anpassung der Gitter verwendet.

Level	τ	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	Δp	St
4	0.01	37	3.1988	1.089	2.4680	0.2703
4	0.005	73	3.2022	1.096	2.4671	0.2740
4	0.0025	146	3.2113	1.115	2.4741	0.2740
Benchmarkintervall			[3.2200, 3.2400]	[0.9800, 1.020]	[2.4600, 2.500]	[0.2950, 0.3050]

Tabelle 10.6: Beispiel 10.1.1, Ergebnisse mit dem modifizierten Crank–Nicolson Verfahren auf Gitter 3, mit Druckseparation.

2. **Zeitdiskretisierung.** Obwohl die Ortsdiskretisierung nur von 1.Ordnung ist, reichte eine Zeitdiskretisierung von 1.Ordnung nicht aus, um befriedigende Ergebnisse zu liefern. Auch bei den verwendeten Zeitdiskretisierungen 2.Ordnung waren zum Teil wesentliche Unterschiede in den Lösungen zu erkennen. Bei diesem Beispiel hat sich das modifizierte Crank–Nicolson Verfahren als besser als die BDF(2) Methode erwiesen. Der numerische Aufwand zur Lösung der linearen Probleme war beim modifizierten Crank–Nicolson Verfahren spürbar höher.

   Eine Verfeinerung der Zeitschrittweite brachte keine wesentliche Verbesserung der Lösungsqualität. An dieser Stelle bemerkt man die Dominanz des Fehlers in der Ortsdiskretisierung über den Fehler in der Zeitdiskretisierung.

3. **Druckseparation.** Die Verwendung der Druckseparation führte zu einer Verbesserung der berechneten Ergebnisse. Sie war allerdings mit einer Erhöhung der Rechenzeiten um den Faktor 2 bis 4 verbunden.

4. **Offene Probleme.** Eine adaptive Zeitschrittweitensteuerung zur Reduzierung des Gesamtaufwandes der Rechnungen wäre wünschenswert.

10.2 Strömung durch einen Kanal um einen Zylinder mit instationärer Einströmung, ein Benchmarkproblem

Beispiel 10.2.1 Hierbei handelt es sich um ein drittes Benchmarkproblem für zweidimensionale Navier–Stokes Gleichungen aus dem DFG–Schwerpunktprogramm „Flow simulation with high performance computers“. Diesmal wird eine zeitabhängige Strömung durch den in Abbildung 9.7 skizzierten Kanal betrachtet, bei welcher die Einströmung ebenfalls zeitabhängig ist. Die Gleichungen lauten

$$\begin{aligned}
\mathbf{u}_t - 10^{-3}\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &= \mathbf{0} && \text{in } \Omega\times[0,8]\\
\nabla\cdot\mathbf{u} &= 0 && \text{in } \Omega\times[0,8]\\
\mathbf{u}(0,y) = \mathbf{u}(2.2,y) &= 0.41^{-2}\left(6y(0.41-y)\sin(\pi t/8),0\right) && y\in[0,0.41], t\in[0,8]\\
\mathbf{u}(x,0) = \mathbf{u}(x,0.41) &= (0,0) && x\in[0,2.2], t\in[0,8]\\
\mathbf{u}(S) &= (0,0) && t\in[0,8]\\
\mathbf{u}(t=0) &= \mathbf{0}.
\end{aligned}$$

Die charakteristische Geschwindigkeit bei diesem Problem ist $U = 1m/s$. Das ist wiederum die Höchstgeschwindigkeit der parabelförmigen Einströmung im Zeitintervall. Damit hat diese Strömung eine Reynoldszahl von $Re = 100$.

Wir bezeichnen die Differenz des Druckes zwischen Anfang und Ende des Zylinders mit

$$\Delta p(t) := p(0.15, 0.2, t) - p(0.25, 0.2, t).$$

Die zu berechnenden Benchmarkkennwerte sind c_w, c_a und $\Delta p(t)$ als Funktionen der Zeit, die Maxima von c_w und c_a sowie $\Delta p(8.0)$.

Die Einströmung verhält sich im Laufe der Zeit wie der nichtnegative Teil einer Sinuskurve. Sie startet also bei $(0, 0)$. Mit wachsender Einströmgeschwindigkeit fließt das Fluid zunächst ähnlich wie beim stationären Benchmarkproblem, Beispiel 9.2.1, am Zylinder vorbei. Die Kraft in Richtung der Strömung, die durch den Widerstandsbeiwert c_w beschrieben wird, erhöht sich stetig, siehe Abbildung 10.2. Nach etwa der Hälfte der Zeit erfolgt das Umschlagen in einen schwingenden Zustand. Dies ist sehr schön am Auftriebsbeiwert c_a zu sehen, der die Kraft orthogonal zur Strömung beschreibt. Bei abklingender Einströmung schwingt das System aus. Es kommt sogar zu einem Rückfluß, was sich in der negativen Druckdifferenz und dem negativen c_w–Wert bemerkbar macht.

Ergebnisse mit verschiedenen Zeitdiskretisierungen und Zeitschrittweiten

In Tabelle 10.7 sind Ergebnisse mit dem modifizierten Crank–Nicolson Verfahren und mit BDF(2), sowie den äquidistanten Zeitschrittweiten $\tau = 0.01$ und $\tau = 0.005$ präsentiert. Auch hier ist zu sehen, daß bei Level 3 und 4 der Ortsdiskretisierung das modifizierte Crank–Nicolson Verfahren dem BDF(2) Verfahren deutlich überlegen ist. Auf Level 3 sind die Schwingungen des Systems bei BDF(2) kaum ausgeprägt. Die Ergebnisse auf Level 5 gleichen sich an, wobei das modifizierte Crank–Nicolson Verfahren immer noch ein wenig besser ist, vor allem im maximalen Widerstandsbeiwert $c_{w_{max}}$.

Verfahren	Level	τ	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	$\Delta p(8.0)$
Crank–Nic.	3	0.01	800	3.1307	0.2499	-0.1050
Crank–Nic.	3	0.005	1600	3.1037	0.2691	-0.1024
Crank–Nic.	4	0.01	800	3.0228	0.4184	-0.1066
Crank–Nic.	4	0.005	1600	3.0228	0.4195	-0.1094
Crank–Nic.	4	0.0025	3200	3.0230	0.4162	-0.1102
Crank–Nic.	5	0.01	800	2.9742	0.4755	-0.0978
Crank–Nic.	5	0.005	1600	2.9738	0.4962	-0.1006
Crank–Nic.	5	0.0025	3200	2.9749	0.4808	-0.1011
BDF(2)	3	0.01	800	3.1821	0.0620	-0.1176
BDF(2)	3	0.005	1600	3.1590	0.0498	-0.1198
BDF(2)	4	0.01	800	3.0586	0.4055	-0.1075
BDF(2)	4	0.005	1600	3.0493	0.3521	-0.1088
BDF(2)	5	0.01	800	2.9898	0.5550	-0.1003
BDF(2)	5	0.005	1600	2.9887	0.4805	-0.1023
Benchmarkintervall				[2.9300, 2.9700]	[0.4700, 0.4900]	[-0.1150, -0.1050]

Tabelle 10.7: Beispiel 10.2.1, Ergebnisse mit verschiedenen Zeitdiskretisierungen.

Die berechneten maximalen Widerstandsbeiwerte sind ein wenig zu hoch, die maximalen Auftriebsbeiwerte für die genauesten Rechnungen liegen im Benchmarkintervall oder sind geringfügig zu hoch,und die berechneten Druckdifferenzen tendieren ebenfalls zu höheren Werten, als sie im Benchmarkintervall vorgegeben sind.

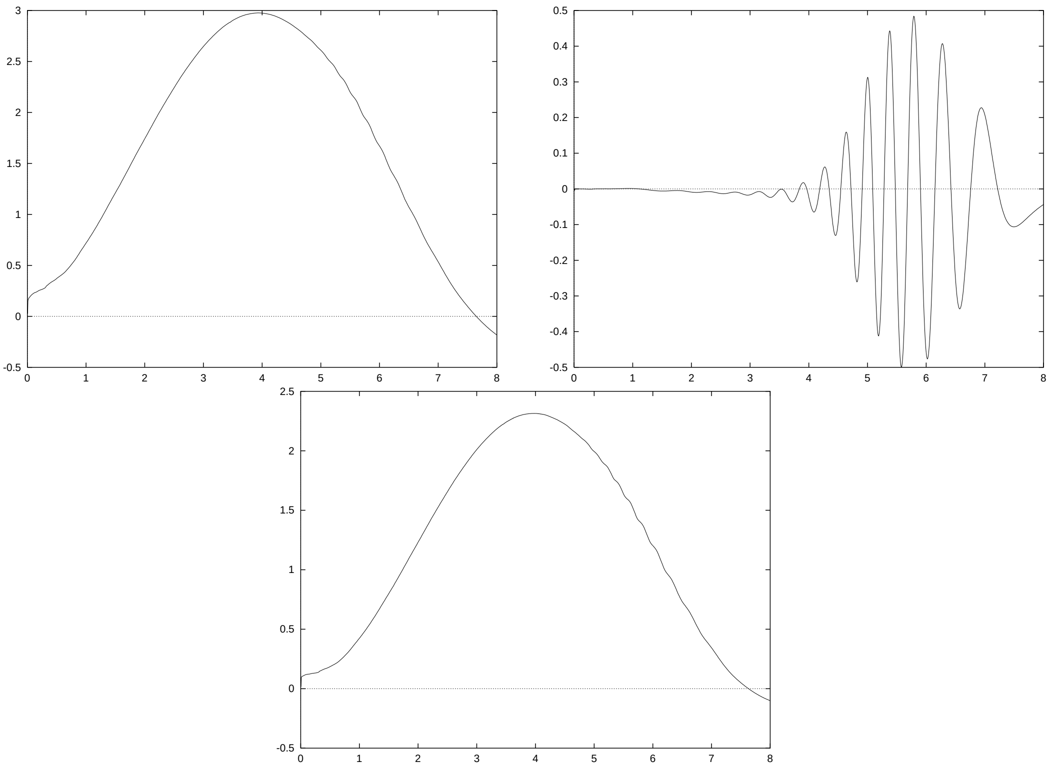


Abbildung 10.2: $c_w(t)$ (oben links), $c_a(t)$ (oben rechts) und $\Delta p(t)$ (unten), modifiziertes Crank–Nicolson Verfahren, Level 5, $\tau = 0.005$.

Der Einfluß der Druckseparation

Für das BDF(2) Verfahren sind Ergebnisse, die ohne Druckseparation berechnet wurden, in Tabelle 10.8 zu sehen. Hierbei erkennt man, daß einige Resultate besser sind als mit Druckseparation. Das betrifft insbesondere die maximalen Widerstandsbeiwerte. Bei der aufwendigsten Rechnung (Level 5, $\tau = 0.005$) liegt er innerhalb des Benchmarkintervalls. Für die Rechnungen auf Level 5 sind auch die Druckdifferenzen für $t = 8$ ein wenig näher am Benchmarkintervall als mit Druckseparation.

Level	τ	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	$\Delta p(8.0)$
3	0.01	800	3.0485	0.0194	-0.1171
3	0.005	1600	3.0748	0.0117	-0.1185
4	0.01	800	3.0317	0.4074	-0.1090
4	0.005	1600	2.9927	0.3321	-0.1090
5	0.01	800	3.0066	0.5556	-0.1059
5	0.005	1600	2.9503	0.4875	-0.1033
Benchmarkintervall			[2.9300, 2.9700]	[0.4700, 0.4900]	[-0.1150, -0.1050]

Tabelle 10.8: Beispiel 10.2.1, Ergebnisse mit BDF(2), ohne Druckseparation.

Rechnungen auf anderen Gittern

Dieses Benchmarkproblem wurde auch auf Gitter 2, Abbildung 9.8, mit BDF(2) und ohne Druckseparation gerechnet. Beim Vergleich der Resultate von Tabelle 10.9 mit denen von Tabelle 10.8 sieht man die starke Abhängigkeit der berechneten maximalen Beiwerte von der Wahl des Gitters. Hier hat man allein durch die Nutzung von Gitter 2 anstelle von Gitter 1 einen spürbaren Qualitätsverlust der Lösung.

Level	τ	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	$\Delta p(8.0)$
5	0.01	800	3.0599	0.6326	-0.1100
5	0.005	1600	3.0617	0.5557	-0.1078
Benchmarkintervall			[2.9300, 2.9700]	[0.4700, 0.4900]	[-0.1150, -0.1050]

Tabelle 10.9: Beispiel 10.2.1, Ergebnisse mit BDF(2) auf Gitter 2, ohne Druckseparation.

Bei der Verwendung von Gitter 3 bereitet es keine Schwierigkeiten, die Werte $c_{w_{max}}$ und $\Delta p(8.0)$ innerhalb ihrer Benchmarkintervalle zu berechnen, siehe Tabelle 10.10. Der berechnete maximale c_a–Wert ist allerdings, wie im Beispiel 10.1.1, noch relativ ungenau.

Level	τ	Schritte	$c_{w_{max}}$	$c_{a_{max}}$	$\Delta p(8.0)$
4	0.01	800	2.9391	0.5216	-0.1080
4	0.005	1600	2.9420	0.5414	-0.1053
Benchmarkintervall			[2.9300, 2.9700]	[0.4700, 0.4900]	[-0.1150, -0.1050]

Tabelle 10.10: Beispiel 10.2.1, Ergebnisse mit dem modifizierten Crank–Nicolson Verfahren auf Gitter 3, mit Druckseparation.

Bemerkung 10.2.2 Auswertung der numerischen Tests. Die Punkte 1, 2 und 4 von Bemerkung 10.1.2 lassen sich an diese Stelle übertragen. Die Nutzung der Druckseparation brachte bei Beispiel 10.2.1 bei vielen, aber nicht bei allen Testrechnungen eine Verbesserung der Lösungsqualität.

10.3 Die Untersuchung verschiedener Löser

In diesem Abschnitt werden die in den Kapiteln 3 bis 5 vorgestellten Löser für das Sattelpunktproblem (3.1) untersucht, das nach der Diskretisierung der linearisierten, instationären Navier–Stokes Gleichungen zu lösen ist. Die Untersuchungen wurden am Beispiel 10.1.1 durchgeführt.

Der Unterschied zum Sattelpunktproblem der linearisierten, stationären Navier–Stokes Gleichungen besteht in den Eigenschaften der Matrix A. Durch die Diskretisierung von $\mathbf{u}_t$ hat A nun den Summanden M/τ, wobei M eine Diagonalmatrix ist und τ die Zeitschrittweite. Insbesondere für kleine τ dominiert M/τ in A, und A ist eine diagonaldominante Matrix. Deswegen ist zu erwarten, daß das Verhalten der Druckkorrektur Verfahren, bei denen $C = \mathrm{diag}(A)$ im

Vorkonditionierer der Richardson Iteration (3.10) gewählt wird, nun wesentlich besser als bei den stationären Navier–Stokes Gleichungen ist.

Zunächst wollen wir die Abkürzungen in den Tabellen dieses Abschnittes erklären:

Löser	Dr-Ko MG (3.9)	Druckkorrektur Verfahren mit Mehrgitterverfahren für die Druckunbekannten zur Lösung des linearen Systems (3.11) innerhalb der Richardson Iteration und Geschwindigkeitsberechnung mit (3.9),
	Dr-Ko MG (3.12)	Druckkorrektur Verfahren mit Mehrgitterverfahren für die Druckunbekannten zur Lösung des linearen Systems (3.11) innerhalb der Richardson Iteration und Geschwindigkeitsberechnung mit (3.12),
	Dr-Ko CG (3.9)	Druckkorrektur Verfahren mit CG zur Lösung des linearen Systems (3.11) innerhalb der Richardson Iteration und Geschwindigkeitsberechnung mit (3.9),
	Vanka–V	gekoppeltes Mehrgitterverfahren mit Vanka–Glätter, V–Zyklus,
	Vanka–F	gekoppeltes Mehrgitterverfahren mit Vanka–Glätter, F–Zyklus,
	Vanka–W	gekoppeltes Mehrgitterverfahren mit Vanka–Glätter, W–Zyklus,

- M — Anzahl der Richardson Iterationen im Druckkorrektur Verfahren beziehungsweise Anzahl der Mehrgitterzyklen im gekoppelten Mehrgitterverfahren pro nichtlinearem Iterationsschritt,
- L — Anzahl der Iterationen zur Lösung von (3.11) bei den Druckkorrektur Verfahren,
- Gl. — Anzahl der Glättungsiterationen in den verwendeten Mehrgitterverfahren, wobei die Anzahl der Vor– und Nachglättungsiterationen immer gleich gewählt wurde,
- Kost. — numerische Kosten in $opmsN_{Topel}$ für die Lösung eines Sattelpunktproblems, vergleiche Kapitel 5,
- τ — Zeitschrittweite,
- It. — Anzahl der benötigten nichtlinearen Iterationen pro Zeitschritt,
- t/Per. — CPU–Zeit in Sekunden zur Berechnung einer Periode, inklusive der Zeitschritte in den Zeitpunkten t_0 und $t_0 + F$,
- t/Schr. — durchschnittliche CPU–Zeit in Sekunden pro Zeitschritt,
- Kom. — Anteil der Kommunikation an der Gesamtzeit in %,
- Grg. — Anteil der Iterationen auf dem Grobgitter an der Gesamtzeit in %.

In den Tabellen befinden sich wieder links vom Doppelstrich die vorgegebenen Größen und rechts davon die berechneten Werte. Bei schwankenden Werten erfolgt eine Angabe „von – bis“.

Für die Verfahren wurden die folgenden Parameter verwendet:

- modifiziertes Crank–Nicolson Verfahren,
- Fixpunktiteration (2.19) zur Lösung der nichtlinearen Probleme,
- Abbruch der nichtlinearen Iteration bei euklidischer Norm des Residuums kleiner als 10^{-5},
- Standardapproximation der Randdaten, siehe Bemerkung 2.4.2,
- keine Druckseparation,
- Stabilisierung mit Samarskij Upwind,
- Schrittweitenkontrolle für alle Mehrgitterverfahren,
- im gekoppelten Mehrgitter F– und W–Zyklus wurde die Grobgitteriteration Variante 2 mit 20 Iterationen verwendet; im gekoppelten Mehrgitter V–Zyklus, in den Mehrgitterverfahren für die Geschwindigkeit und den Druck wurde die Variante 1 verwendet, wobei die euklidische Norm des Residuums um den Faktor 0.1 verringert wurde, siehe Abschnitt 4.3,
- keine Dämpfung in der Grobgitter– und Glättungsiteration des gekoppelten Mehrgitterverfahrens und des Mehrgitterverfahrens für den Druck; Dämpfung mit Faktor 0.5 im Mehrgitterverfahren der Geschwindigkeitsunbekannten,

- $\beta = 1.0$ bei der ILU_β–Zerlegung,
- Verwendung von $A_{1,1}^{-1}$ und $C = \mathrm{diag}(A)$im Druckkorrektur Verfahren,
- Gitter 1,
- lexikographische Anordnung der Knoten auf einem Prozessor: ein Knoten mit den Koordinaten (x_0, y_0) ist ein Vorgänger eines Knotens mit den Koordinaten (x_1, y_1) genau dann wenn $(x_0 < x_1) \vee ((x_0 = x_1) \wedge (y_0 < y_1))$ gilt.

Wie bereits im Abschnitt 10.1 bemerkt worden ist, benötigt man mit Druckseparation bei diesem Beispiel längere Rechenzeiten als ohne. Man vergleiche dazu die Ergebnisse von Tabelle 10.2 mit denen der Tabellen 10.12 und 10.13.

Die Druckkorrektur Verfahren haben sich als sehr sensitiv gegenüber der Wahl des Dämpfungsparameters α in der Richardson Iteration (3.10) erwiesen. Eine gute Wahl dieses Parameters hing nicht nur von der Zeitschrittweite τ, sondern auch von der Feinheit der Ortsdiskretisierung ab. Schon kleine Veränderungen nach oben führten zur Divergenz des Druckkorrektur Verfahrens und Abweichungen nach unten bewirkten eine langsamere Konvergenz. Die gewählten Parameter in den Testrechnungen sind in Tabelle 10.11 zusammengestellt.

τ	0.01	0.005	0.0025
Tabelle 10.12, Level 4	0.15	0.25	0.4
Tabelle 10.13, Level 5	0.05	0.1	0.2

Tabelle 10.11: Benchmarkproblem 10.1.1, Wahl des Dämpfungsparameters α in der Richardson Iteration (3.10).

Das Verhalten der Verfahren ist in den Tabellen 10.12 und 10.13 dokumentiert. Wir betrachten zuerst die Druckkorrektur Verfahren. Bei der Nutzung von CG zur Lösung der in der Richardson Iteration auftretenden Gleichungssysteme (3.11) konvergiert das Druckkorrektur Verfahren für große Zeitschritte nicht und für kleine mit erheblichen Schwierigkeiten, siehe Tabelle 10.12. Die maximale Anzahl der Schritte innerhalb der nichtlinearen Iteration war mit 30 vorgeschrieben. Das CG Verfahren allein innerhalb der Druckkorrektur Verfahren ist nicht ausreichend. Man benötigt einen geeigneten Vorkonditionierer. Ganz anders sieht es mit der Nutzung des Mehrgitterverfahrens für die Druckunbekannten aus. Es gibt, bei geeignet gewähltem Dämpfungsparameter α, keine Probleme mit der Konvergenz. Die Verwendung von (3.12) anstelle von (3.9) zur Berechnung der Geschwindigkeit spart ein wenig Rechenzeit. Die berechneten Benchmarkkennwerte sind bei beiden Varianten nahezu gleich. Die Verwendung der Schrittweitenkontrolle in beiden Mehrgitterverfahren erweist sich für die Beschleunigung der Konvergenz als sehr wichtig, vergleiche die Rechenzeiten der Tabellen 10.12 und 10.13 mit denen von 10.14 und 10.15.

Obwohl bei unserer Implementierung der Verfahren die Druckkorrektur Verfahren mit Mehrgitter für den Druck immer noch schlechter als die gekoppelten Mehrgitterverfahren abschneiden, sind die Unterschiede, vor allem für kleine Zeitschrittweiten, wesentlich geringer als bei den stationären Navier–Stokes Gleichungen. An dieser Stelle soll die Abhängigkeit der präsentierten Zahlen von der verwendeten Hardware, aber auch der Implementierung der Verfahren, der Datenstruktur et cetera, betont werden. Unter anderen Voraussetzungen als den unsrigen ist es vorstellbar, daß die Druckkorrektur Verfahren bei Verwendung kleiner Zeitschrittweiten den gekoppelten Mehrgitterverfahren überlegen sind. Wir glauben jedoch nicht, daß diese Überlegenheit sehr groß sein wird. Falls das Problem nicht die Auflösung sehr kleiner Zeitskalen erfordert, will man ohnehin kleine Zeitschrittweiten vermeiden. Im Beispiel 10.1.1 hat sich gezeigt, daß die

Löser	M	Gl.	L	Kost.	τ	It.	t/Per.	t/Schr.	Kom.	Grg.
Vanka–V	5	5	-	7340	0.01	8	14100	382	4-6	3-6
Vanka–F	5	4	-	8080	0.01	6	10400	281	4.4	3.7
Vanka–W	5	3	-	7170	0.01	5-6	9010	244	6.3	8.2
Dr-Ko MG (3.9)	10	3	3	7532	0.01	8	21200	574	9-13	3-7
Dr-Ko MG (3.12)	10	3	3	7090	0.01	8	20200	546	8-12	3-7
Dr-Ko CG (3.9)	10	3	50	9482	0.01	30	43300	1171	7.2	0.3
Vanka–V	5	5	-	7340	0.005	4-5	14700	204	4-6	2-6
Vanka–F	5	4	-	8080	0.005	3-4	11600	161	4.6	3.7
Vanka–W	5	3	-	7170	0.005	3-4	10500	146	6.4	8.2
Dr-Ko MG (3.9)	10	3	3	7532	0.005	4	21100	289	8-12	3-8
Dr-Ko MG (3.12)	10	3	3	7090	0.005	4	19700	274	8-13	3-8
Dr-Ko CG (3.9)	10	3	50	9482	0.005	30	83800	1164	7.0	0.2
Vanka–V	5	5	-	7340	0.0025	3	16700	117	5.0	3-5
Vanka–F	5	4	-	8080	0.0025	3	16400	114	4.4	3.6
Vanka–W	5	3	-	7170	0.0025	3	14600	102	5.9	8.1
Dr-Ko MG (3.9)	10	3	3	7532	0.0025	3	25800	180	9-13	3-8
Dr-Ko MG (3.12)	10	3	3	7090	0.0025	3	24600	172	9-14	3-8
Dr-Ko CG (3.9)	10	3	50	9482	0.0025	25-30	158000	1109	6.6	0.2

Tabelle 10.12: Benchmarkproblem 10.1.1, Lösungsverhalten bei Verwendung verschiedener Löser, modifiziertes Crank–Nicolson Verfahren, Level 4, 8 Prozessoren.

Verkleinerung der Zeitschrittweite von $\tau = 0.005$ auf $\tau = 0.0025$ keine wesentliche Verbesserung der Ergebnisse gebracht hat.

Löser	M	Gl.	L	Kost.	τ	It.	t/Per.	t/Schr.	Kom.	Grg.
Vanka–V	5	5	-	7340	0.01	12-17	48900	1359	2.9	1.1
Vanka–F	5	4	-	8080	0.01	9-12	37600	1045	3.6	2.7
Vanka–W	5	3	-	7170	0.01	9-10	32300	896	6.7	9.2
Dr-Ko MG (3.12)	10	3	3	7090	0.01	19-22	106000	2960	8-11	2-5
Vanka–V	5	5	-	7340	0.005	8-10	56500	807	2.4	1.1
Vanka–F	5	4	-	8080	0.005	6-7	42300	605	3.6	2.7
Vanka–W	5	3	-	7170	0.005	6	38000	543	7.1	9.0
Dr-Ko MG (3.12)	10	3	3	7090	0.005	10	95900	1370	8-11	2-5
Vanka–V	5	5	-	7340	0.0025	4-5	56800	411	2.9	1.1
Vanka–F	5	4	-	8080	0.0025	3-4	43100	312	3.6	2.7
Vanka–W	5	3	-	7170	0.0025	3	39400	285	6.6	9.1
Dr-Ko MG (3.12)	10	3	3	7090	0.0025	5	86500	627	7-10	2-4

Tabelle 10.13: Benchmarkproblem 10.1.1, Lösungsverhalten bei Verwendung verschiedener Löser, modifiziertes Crank–Nicolson Verfahren, Level 5, 16 Prozessoren.

Innerhalb der gekoppelten Mehrgitterverfahren zeigt sich der W–Zyklus abermals überlegen. Diese Überlegenheit ist für $\tau = 0.01$ recht deutlich. Mit einer Verkleinerung der Zeitschrittweite nimmt sie ab. Die Verwendung der Schrittweitenkontrolle beschleunigt die Konvergenz der

Verfahren beträchtlich, vergleiche die Rechenzeiten der Tabellen 10.12 und 10.13 mit denen von 10.14 und 10.15.

Löser	M	Gl.	L	τ	It.	t/Per.	t/Schr.	Kom.	Grg.
Vanka–V	5	5	-	0.005	9	27900	387	3.0	1.8
Vanka–F	5	4	-	0.005	6-7	20100	280	4.1	3.9
Vanka–W	5	3	-	0.005	6	18200	252	6.1	8.7
Dr-Ko MG (3.12)	10	3	3	0.005	7-12	35500	493	9-14	3-9

Tabelle 10.14: Benchmarkproblem 10.1.1, Lösungsverhalten bei Verwendung verschiedener Löser ohne Schrittweitenkontrolle, modifiziertes Crank–Nicolson Verfahren, Level 4, 8 Prozessoren.

Löser	M	Gl.	L	τ	It.	t/Per.	t/Schr.	Kom.	Grg.
Vanka–V	5	5	-	0.005	14-17	94500	1349	2.4	1.2
Vanka–F	5	4	-	0.005	10-11	69700	996	3.7	2.8
Vanka–W	5	3	-	0.005	9-10	61200	874	7.0	9.5
Dr-Ko MG (3.12)	10	3	3	0.005	25-31	260800	3726	8-10	2-4

Tabelle 10.15: Benchmarkproblem 10.1.1, Lösungsverhalten bei Verwendung verschiedener Löser ohne Schrittweitenkontrolle, modifiziertes Crank–Nicolson Verfahren, Level 5, 16 Prozessoren.

Bemerkung 10.3.1 Auswertung der numerischen Tests.

1. **Druckkorrektur Verfahren – gekoppeltes Mehrgitterverfahren.** Die gekoppelten Mehrgitterverfahren haben sich den von uns verwendeten Druckkorrektur Verfahren überlegen gezeigt. Diese Überlegenheit ist allerdings längst nicht mehr so dramatisch wie bei den stationären Navier–Stokes Gleichungen, insbesondere für kleine Zeitschrittweiten. Als ein wesentlicher Nachteil der Druckkorrektur Verfahren erscheint uns die starke Sensitivität gegenüber dem Dämpfungsfaktor α der Richardson Iteration.

2. **Gekoppeltes Mehrgitterverfahren.** Bei diesem Testbeispiel hat sich der W–Zyklus als das schnellste gekoppelte Mehrgitterverfahren erwiesen. Der Abstand zum F–Zyklus ist im allgemeinen jedoch ziemlich gering. Nur mit dem V–Zyklus traten, insbesondere für große Zeitschrittweiten, spürbare Erhöhungen der Rechenzeit auf.

3. **Schrittweitenkontrolle.** Die Verwendung der Schrittweitenkontrolle hat die Konvergenzgeschwindigkeit aller Verfahren beträchtlich erhöht.

Kapitel 11

Zusammenfassung und Ausblick

Die vorliegende Arbeit ist ein Beitrag zur Theorie und Numerik der stationären und instationären inkompressiblen Navier–Stokes Gleichungen. Auf der Basis einer nichtkonformen P_1/P_0–Finite Elemente Diskretisierung wurde gezeigt, daß man zweidimensionale Probleme, die im Rahmen von Benchmarkrechnungen der DFG definiert wurden, mit akzeptablem Aufwand genügend genau lösen kann. Dabei erwiesen sich die Verwendung von Samarskij Upwind anstelle von scharfem Upwind sowie die Nutzung von Druckseparation als wichtig. Bei der Lösung dieser Probleme haben sich speziell gekoppelte Mehrgitterverfahren als effiziente Löser erwiesen. Die Effizienz dieser Verfahren wurde durch einige Verfahrensparameter beeinflußt, deren geeignete Wahl von der Anzahl der verwendeten Prozessoren abhängt. Für den V– und F–Zyklus wurden nur geringe parallele Effizienzverluste festgestellt. Trotz höherer Verluste an paralleler Effizienz hat sich der W–Zyklus bei moderater Anzahl verwendeter Prozessoren als bestes Verfahren herausgestellt. Die auf uniform verfeinerten Gittern berechneten Ergebnisse konnten vielfach mit Hilfe lokaler Gitterverfeinerung deutlich verbessert werden. Die vorliegende Arbeit enthält analytische Untersuchungen zu den verwendeten lokalen a posteriori Fehlerschätzern. Diese enthalten im Vergleich zu Fehlerschätzern für konforme Diskretisierungen zusätzlich die Sprünge der nichtkonformen Funktion über Elementkanten. Darüber hinaus wurde ein globaler a posteriori Fehlerschätzer für den Fehler in der L^2–Norm der Geschwindigkeit hergeleitet, welcher ebenfalls Sprünge der nichtkonformen Funktion über Elementkanten besitzt. Die globale Fehlerabschätzung hat als Faktor die Stabilitätskonstante eines stetigen, linearisierten dualen Problems. Am Beispiel von Konvektions–Diffusions Gleichungen wurde exemplarisch ein Weg zur Schätzung dieser Konstanten und damit zu einer robusten Fehlerschätzung aufgezeigt.

Zur Berechnung guter Ergebnisse in den Benchmarkproblemen waren, auch bei lokaler Verfeinerung, feine Gitter mit relativ vielen Unbekannten nötig. Bei einer ähnlich feinen Vernetzung dreidimensionaler Gebiete potenziert sich die Anzahl der Unbekannten vom Zweidimensionalen mit 1.5. Dies übersteigt das Speichervermögen der heutzutage genutzten Rechentechnik. Daher wird man im Dreidimensionalen auf wesentlich gröberen Gittern als im Zweidimensionalen arbeiten müssen. Zur Berechnung guter Ergebnisse mit der dreidimensionalen P_1/P_0–Finite Elemente Diskretisierung wird es nötig sein, die Triangulierung sehr sorgfältig zu wählen. Eine wesentliche Bedeutung wird dabei adaptiven Methoden zur Gittergenerierung zukommen. Neben lokaler Gitterverfeinerung wird auch die Nutzung anderer Methoden zur Gittermanipulation, wie etwa die Verschiebung von Knoten, wichtig sein.

Anhang A

Verwendete Bezeichnungen

$a(\cdot,\cdot)$	Bilinearform des diffusiven Terms,	S. 8
$a_h(\cdot,\cdot)$	diskrete Bilinearform des diffusiven Terms,	S. 10
$a_h^i(u_h, v_h)$	Bilinearformen der NSDFEM,	(1.22), (1.23)
$a_\omega(\cdot,\cdot)$	Einschränkung von $a_h(\cdot,\cdot)$ auf $\omega \subset \Omega$,	S. 98
$\mathcal{A}$	Systemmatrix für die Navier–Stokes Gleichungen,	(3.1)
$\mathcal{A}_l$	Systemmatrix für die Navier–Stokes Gleichungen auf Level l,	S. 44
$\mathcal{A}_T$	Teil von $\mathcal{A}_T$ der mit dem Element T verbunden ist,	S. 44
A	linker oberer Block von $\mathcal{A}$,	(3.1)
A_l	linker oberer Block von $\mathcal{A}_l$,	S. 47
$A_{\gamma,j}^{-1}$	Mehrgitterzyklus der Form γ mit j Zyklen für ein System $Ax = b$,	S. 34
b	Koeffizientenfunktion des konvektiven Terms,	(1.1)
$b(\cdot,\cdot)$	Bilinearform des konvektiven Terms,	S. 8
$b(\cdot,\cdot,\cdot)$	Trilinearform der Navier–Stokes Gleichungen,	(2.7)
$b_\omega(\cdot,\cdot,\cdot)$	Einschränkung von $b_h(\cdot,\cdot,\cdot)$ auf $\omega \subset \Omega$,	S. 98
$b_h(\cdot,\cdot)$	diskrete Bilinearform des konvektiven Terms,	S. 11
$b_s(\cdot,\cdot)$	Bilinearform des konvektiven Terms der schiefsymmetrischen Formulierung,	S. 8
$\tilde{b}(\cdot,\cdot)$	Upwind Diskretisierung des konvektiven Terms,	S. 12
B	rechter oberer Block von $\mathcal{A}$, diskreter Gradient,	(3.1)
B_l	rechter oberer Block von $\mathcal{A}_l$,	S. 47
B_i	Mittelpunkt der Seite E_i,	S. 9
B_E	Kantenblasenfunktion,	S. 78
B_E^*	Kantenblasenfunktion,	S. 78
B_T	Elementblasenfunktion,	S. 78
B_T^*	Elementblasenfunktion,	S. 78
c	Koeffizientenfunktion des reaktiven Terms,	(1.1)
$c(\cdot,\cdot)$	Bilinearform des reaktiven Terms,	S. 8
c_a	Auftriebsbeiwert,	(9.2)
$c_h(\cdot,\cdot)$	diskrete Bilinearform des reaktiven Terms,	S. 11
$c_s(\cdot,\cdot)$	Bilinearform des reaktiven Terms der schiefsymmetrischen Formulierung,	S. 8
c_w	Widerstandsbeiwert,	(9.2)
c_0	positive, gleichmäßige untere Schranke von $c - \nabla \cdot b/2$,	(1.6)

C	Matrix im Vorkonditionierer des Druckkorrektur Verfahrens,	S. 33
$C(\mathbf{u}_h^n)$	konvektive Matrix,	S. 26
C_{gl}	numerische Kosten des Glättens,	S. 69
C_{gr}	numerische Kosten des Grobgitterlösers,	S. 69
C_{pr}	numerische Kosten der Prolongation,	S. 69
C_{re}	numerische Kosten der Restriktion,	S. 69
C_{rb}	numerische Kosten der Residuumsberechnung,	S. 69
C_{sk}	numerische Kosten der Schrittweitenkontrolle,	S. 69
C_I	Konstante bei Interpolationsabschätzungen,	S. 76
C_S	Stabilitätskonstante des dualen Problems,	(6.41), (7.21)
$\hat{C}_S, \bar{C}_S$	numerische Approximationen an C_S,	S. 128
$\mathrm{curl}\, v$		S. 76
E_i	Seiten von T_i,	S. 9
$E(m,n)$	parallele Effizienz zwischen m und n Prozessoren,	S. 158
$f, \mathbf{f}$	rechte Seite,	(1.1), (2.6)
$f_h, \mathbf{f}_h$	diskrete rechte Seite, polynomiale Approximation der rechten Seite in den Fehlerschätzern,	(2.18), (6.21), (7.4)
$\hat{f}_h$	diskrete rechte Seite eines konformen Problems,	(6.42)
g, g	Dirichletrandbedingung,	(1.1), (2.6)
g_h, g_h	diskrete Dirichletrandbedingung,	(2.18)
g_N	Neumannrandbedingung,	(1.1)
h	$\max_{T\in\mathcal{T}_h}\{h_T\}$, Gitterparameter,	S. 9
h_E	Länge der Kante E,	S. 9
h_T	Durchmesser (längste Seite) eines Elementes,	S. 9
I_N	Einheitsmatrix der Dimension N	
L	charakteristische Länge,	S. 17
L	Anzahl der Iterationen zur Lösung des Druckproblems bei den Druckkorrektur Verfahren,	S. 72
M	Massenmatrix,	S. 27
M	Anzahl der Richardson Iterationen bei Druckkorrektur Verfahren oder der Mehrgitterzyklen bei gekoppelten Mehrgitterverfahren,	S. 72
$\mathcal{M}_E^k$	Projektion einer Funktion auf ein Polynom vom Grad k auf der Kante E,	S. 77
N_D	Anzahl der Knoten auf $\partial\Omega_D$,	S. 9
N_p	Anzahl der Unbekannten des Druckes; das entspricht der Anzahl der Elemente der feinsten Triangulierung $= \dim Q_h$,	S. 9
$N_{el,l}$	Anzahl der Elemente auf Level l,	S. 44
N_{Topel}	Anzahl der Elemente der feinsten Triangulierung; entspricht der Anzahl der Unbekannten des Druckes,	S. 67
N_u	Anzahl der Unbekannten der Geschwindigkeit; bei Navier–Stokes entspricht das der doppelten Anzahl der inneren Kanten und der Kanten auf dem Neumannrand der Triangulierung, $= \dim V_h$,	S. 9
p	Druck in den stetigen Navier–Stokes Gleichungen,	(2.6)
p_h	Druck in den diskreten Navier–Stokes Gleichungen,	(2.18)
p_S	separierter Druck,	S. 28

P_i	Prozessor i,	S. 53
$(P_i \to x)$	Wert der Variablen x auf Prozessor i,	S. 53
$P_k(T)$	Raum der Polynome vom Grad kleiner als k auf dem Element T	
$P[\mathbf{u}]_{l+1}^{l}$	Prolongation der Unbekannten der Geschwindigkeit von Level l auf $l+1$,	(3.15)
$P[p]_{l+1}^{l}$	Prolongation der Unbekannten des Druckes von Level l auf $l+1$,	(3.21)
$\Delta p, \Delta p(t)$	Druckdifferenz in den Benchmarkproblemen der Navier–Stokes Gleichungen,	S. 139, 160
Q	$L_0^2(\Omega)$	
Q_{h0}, Q_h	diskreter Raum für den Druck,	S. 23
$Q_{h,l}$	diskreter Raum für den Druck auf Level l,	S. 42
$R[\mathbf{u}]_l^{l+1}$	Restriktion der Unbekannten der Geschwindigkeit von Level $l+1$ auf l,	(3.13)
$R[p]_l^{l+1}$	Restriktion der Unbekannten des Druckes von Level $l+1$ auf l,	(3.19)
R_{Q_h}	Clément–Interpolationsoperator in den Raum Q_h,	S. 97
R_{X_h}	Standard–Knoten–Interpolationsoperator in den Raum X_h,	S. 76
Re	Reynoldszahl,	S. 18
s	Lösung für dem Druck im dualen Problem der Navier–Stokes Gleichungen,	(7.19)
s_h	Interpolierte von s,	S. 108
St	Strouhalzahl,	S. 18
T	charakteristische Zeit,	S. 17
$\mathcal{T}_h$	reguläre Triangulierung,	S. 9
$\mathcal{T}_l$	Level l von $\mathcal{T}_h$	
T_i	Elemente von $\mathcal{T}_h$,	S. 9
$u, \mathbf{u}$	Lösung eines stetigen Problems	
$u_h, \mathbf{u}_h$	Lösung eines diskreten Problems	
U	charakteristische Geschwindigkeit,	S. 17
V	$(H^1(\Omega))^d, d = 2, 3$	
V_h	diskreter Ansatzraum für die Geschwindigkeit (Navier–Stokes Gleichungen) oder die Lösung (Konvektions–Diffusions Gleichungen),	(1.8), S. 23
V_{h0}	diskreter Ansatzraum für die Geschwindigkeit mit homogenen Randbedingungen,	S. 9, 23
$V_{h,l}$	diskreter Ansatzraum für die Geschwindigkeit auf Level l	
V_0	Funktionen aus $(H_1(\Omega))^d$, $d = 2, 3$, mit homogenen Randdaten auf dem Dirichletrand	
W	divergenzfreie Funktionen aus V,	S. 19
X_h	Raum der konformen, stückweise linearen Funktionen in skalar oder vektoriell,	S. 76
z_w	Lösung eines stetigen dualen Problems,	(6.39)
$z, \mathbf{z}$	Lösung eines stetigen dualen Problems mit dem Fehler als rechter Seite,	S. 91, (7.19)
$z_h, \mathbf{z}_h$	Interpolierte von $z, \mathbf{z}$,	S. 91, 108

α	Dämpfungsparameter in der Richardson Iteration der Druckkorrekturverfahren,	(3.7)
β	Parameter von ILU$_\beta$,	S. 36
γ	Art eines Mehrgitterzyklus,	S. 36
δ_T	Stromliniendiffusionsparameter,	(1.22)
δ_{ij}	Kronecker Symbol	
$\eta_{T,i}$	lokaler a posteriori Fehlerschätzer in der Energienorm,	(6.21), (6.22), (7.4), (7.5),
$\tilde{\eta}_{T,i}, \tilde{\eta}_T$	lokaler a posteriori Fehlerschätzer in der L^2–Norm,	(6.31), (6.32), (7.12)
η, η_T	globaler a posteriori Fehlerschätzer in der L^2–Norm,	(6.43), (7.24)
$\hat{\eta}$	modifizierter globaler a posteriori Fehlerschätzer in der L^2–Norm,	(6.47)
θ	Parameter bei den Ein–Schritt-θ-Methoden,	(2.21)
κ	Faktor in Schrittweitenkontrollen	(3.17), (3.22), (3.24)
λ_E	lineare Funktion, die auf der Kante E verschwindet,	S. 78
λ_1	Anzahl der Vorglättungsschritte im gekoppelten Mehrgitter,	S. 71
λ_2	Anzahl der Nachglättungsschritte im gekoppelten Mehrgitter,	S. 71
μ_1	Anzahl der Vorglättungsschritte im Mehrgitter für den Druck,	S. 71
μ_2	Anzahl der Nachglättungsschritte im Mehrgitter für den Druck,	S. 71
ν	Konstante im diffusiven Term,	(1.1), (2.6)
ν_0	positive, gleichmäßige untere Schranke von $\nu(x)$,	S. 7
ν_1	Anzahl der Vorglättungsschritte im Mehrgitter für die Geschwindigkeit,	S. 69
ν_2	Anzahl der Nachglättungsschritte im Mehrgitter für die Geschwindigkeit,	S. 69
ρ_T	Durchmessers des Inkreises des Elementes T,	S. 76
σ_E	Kontrollparameter der NSDFEM,	(1.22)
τ	Zeitschrittweite,	S. 27
ϕ_i	Basisfunktion von V_h,	S. 10, (2.13)
$\phi_{i,l}$	Basisfunktion von $V_{h,l}$,	S. 36
$\Phi(t)$	Upwind Funktion,	(1.13)
ψ_j	Basisfunktion von Q_h,	(2.14)
$\psi_{j,l}$	Basisfunktion von $Q_{h,l}$,	S. 42
ω_E	Dreiecke, die E als gemeinsame Kante besitzen,	S. 77
ω_T	Dreiecke, die eine gemeinsame Kante mit T besitzen,	S. 77
Ω	Grundgebiet,	(1.1)
$\partial\Omega$	Rand von Ω,	S. 7
$\partial\Omega_D$	Rand von Ω mit Dirichletbedingungen,	(1.1)
$\partial\Omega_N$	Rand von Ω mit Neumannbedingungen,	(1.1)

$\vert \cdot \vert_h$	gitterabhängige H^1–Seminorm,	(1.9)
$\Vert \cdot \Vert_h$	gitterabhängige H^1–Norm,	S. 10
$\Vert \cdot \Vert_S$	Stromlinien–Diffusionsnorm,	S. 111
$\vert\vert\vert \cdot \vert\vert\vert$	nichtkonforme Stromlinien–Diffusionsnorm,	S. 14
$[\vert v \vert]_E$	Sprung der Funktion v über die Kante E,	S. 76
$[\vert v \vert]_E^C$	Fortsetzung des Sprunges der Funktion v über die Kante E,	S. 82, 83
$\langle \cdot, \cdot \rangle_E$	L^2–Skalarprodukt auf der Kante E, analog für $\partial\Omega_N$,	S. 8
$(\cdot, \cdot)$	Bilinearform auf auf Ω,	S. 8,
$(\cdot, \cdot)_T$	Einschränkung einer Bilinearform auf ein Element T, analog für $\omega_E, \omega_T, \ldots$	
0_N	Vektor mit lauter Nullen der Länge N,	S. 31
1_N	Vektor mit lauter Einsen der Länge N,	S. 31

Literaturverzeichnis

[AB84] O. Axelsson, V.A. Barker. *Finite Element Solutions of Boundary Value Problems.* Academic Press, 1984.

[Ada95] D. Adam. *Eine gleichmäßig konvergente Diskretisierungstechnik für konvektionsdominante Konvektions–Diffusions–Probleme.* Dissertation, TU Dresden, 1995.

[Ang95] L. Angermann. Balanced a posteriori error estimates for finite–volume type discretizations of convection–dominated elliptic problems. *Computing*, 55:305 – 323, 1995.

[Bas96] P. Bastian. *Parallele adaptive Mehrgitterverfahren.* Teubner Skripten zur Numerik. Teubner–Verlag, 1996.

[Bec95] R. Becker. *An Adaptive Finite Element Method for the Incompressible Navier–Stokes Equations on Time–dependent Domains.* Dissertation, Universität Heidelberg, 1995.

[BF91] F. Brezzi, M. Fortin. *Mixed and Hybrid Finite Element Methods*, Band 15 von *Springer Series in Computational Mathematics.* Springer-Verlag, 1991.

[BR96] R. Becker, R. Rannacher. Weighted a posteriori error control in fe methods. Preprint 96 - 1, SFB 359, Universität Heidelberg, 1996.

[Bra96] D. Braess. Multigrid methods for nonconforming finite elements. In *International Workshop on the Recent Advances in Applied Mathematics, Kuwait*, pages 87 – 96. Kuwait University, 1996.

[Bre89] S.C. Brenner. An optimal order multigrid method for p1 nonconforming finite elements. *Math. Comp.*, 52:1 – 15, 1989.

[Bre90] S.C. Brenner. A nonconforming multigrid method for the stationary Stokes equations. *Math. Comp.*, 55(192):411 – 437, 1990.

[BV90] D. Braess, R. Verfürth. Multigrid methods for nonconforming finite element methods. *SIAM J. Numer. Anal.*, 27:979 – 986, 1990.

[BW91] R.E. Bank, B.D. Welfert. A posteriori error estimates for the Stokes problem. *SIAM J. Numer. Anal.*, 28(3):591 – 623, 1991.

[CF88] P. Constantin, C. Foias. *Navier–Stokes equations.* University of Chicago Press. Chicago, London, 1988.

[Cia91] P.G. Ciarlet. Basic error estimates for elliptic problems. In P.G. Ciarlet, J.L. Lions, editors, *Handbook of Numerical Analysis II*, pages 19–351. North–Holland Amsterdam, New York, Oxford, Tokyo, 1991.

[Clé75] Ph. Clément. Approximation by finite element functions using local regularization. *RAIRO Anal. Numer.*, 2:77 – 84, 1975.

[CR73] M. Crouzeix, P.-A. Raviart. Conforming and nonconforming finite element methods for solving the stationary Stokes equations I. *R.A.I.R.O. Analyse Numérique*, 7:33–76, 1973.

[DDP95] E. Dari, R. Durán, C. Padra. Error estimators for nonconforming finite element approximations of the Stokes problem. *Math. Comp.*, 64(211):1017 – 1033, 1995.

[DDPV96] E. Dari, R. Duran, C. Padra, V. Vampa. A posteriori error estimators for nonconforming finite element methods. *M2AN*, 30(4):385 – 400, 1996.

[DJR+96] O. Dorok, V. John, U. Risch, F. Schieweck, L. Tobiska. Parallel finite element methods for the incompressible Navier–Stokes equations. In E.H. Hirschel, editor, *Flow Simulation with High–Performance Computers II*, Band 52 von *Notes on Numerical Fluid Mechanics*, pages 20 – 33. Vieweg, 1996.

[Dor95] O. Dorok. *Eine stabilsierte Finite–Element–Methode zur Lösung der Boussinesq–Approximation der Navier–Stokes–Gleichungen.* Dissertation, Otto-von-Guericke Universität Magdeburg, Fakultät für Mathematik, 1995.

[Dör96] W. Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM J. Numer. Anal.*, 33(3):1106 – 1124, 1996.

[EEHJ95] K. Eriksson, D. Estep, P. Hansbo, C. Johnson. Introduction to adaptive methods for differential equations. *Acta Numerica*, pages 105–158, 1995.

[Fei93] M. Feistauer. *Mathematical Methods in Fluid Dynamics*, Band 67 von *Pitman Monographs and Surveys in Pure and Applied Mathematics.* Longman Scientific & Technical, 1993.

[FF85] M. Fortin, A. Fortin. Experiments with several elements for viscous incompressible flows. *International Journal for numerical methods in Fluids*, 5:911–928, 1985.

[GR86] V. Girault, P.-A. Raviart. *Finite Element Methods for Navier–Stokes equations.* Springer-Verlag, Berlin-Heidelberg-New York, 1986.

[GT83] D. Gilbarg, N.S. Trudinger. *Elliptic Partial Differential Equations of Second Order*, Band 224 von *Grundlehren der mathematischen Wissenschaften.* Springer, 1983.

[Hac85] W. Hackbusch. *Multi-Grid Methods and Applications.* Springer-Verlag, Berlin-Heidelberg-New York, 1985.

[Hac91] W. Hackbusch. *Iterative Lösung großer schwach besetzter Gleichungssysteme.* Teubner, Stuttgart, 1991.

[HB79] T.J.R. Hughes, A.N. Brooks. A multidimensional upwind scheme with no crosswind diffusion. In T.J.R. Hughes, editor, *Finite Element Methods for Convection Dominated Flows, AMD vol.34.* ASME, New York, 1979.

[HNW87] E. Hairer, S.P. Nørsett, G. Wanner. *Solving ordinary differential equations I. Nonstiff problems.* Springer-Verlag, Berlin, 1987.

[HS52] M. R. Hestenes, E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Standards*, 49:409–436, 1952.

[JMT96] V. John, J.M. Maubach, L. Tobiska. Nonconforming streamline–diffusion–finite–element–methods for convection–diffusion problems. Preprint 10/96, Otto-von-Guericke Universität Magdeburg, Fakultät für Mathematik, 1996. to appear in Numer. Math.

[Joh94] V. John. A comparison of some error estimators for convection–diffusion problems on a parallel computer. Preprint 12/94, Otto-von-Guericke Universität Magdeburg, Fakultät für Mathematik, 1994.

[Joh95a] V. John. Analyzing two–level methods for the Navier–Stokes equations by a posteriori error estimation. Technical Report ICMA–95–194, University of Pittsburgh, Department of Mathematics and Statistics, 1995.

[Joh95b] C. Johnson. On computability and error control in CFD. *Int. Jour. Num. Meth. Fluids*, 20:777–788, 1995.

[JR94] C. Johnson, R. Rannacher. On error control in CFD. In Wittum Hebeker, Rannacher, editor, *Numerical Methods for the Navier–Stokes Equations*, Band 47 von *Notes on Numerical Fluid Mechanics*, pages 133 – 144. Vieweg, 1994.

[JRB95] C. Johnson, R. Rannacher, M. Boman. Numerics and hydrodynamics stability: Toward error control in computational fluid dynamics. *SIAM J. Numer. Anal.*, 32(4):1058 – 1079, 1995.

[Kno96] P. Knobloch. *Solvability and Finite Element Discretization of a Mathematical Model Related to Czochralski Crystal Growth.* Dissertation, Otto–von–Guericke Universität Magdeburg, 1996.

[KS88] A. Kiełbasiński, H. Schwetlick. *Numerische lineare Algebra.* Deutscher Verlag der Wissenschaften Berlin, 1988.

[Lan92] U. Langer. Substrukturtechnik und Schwarzsche Methoden. *GAMM - Mitteilungen*, 2:86 – 103, Oktober 1992.

[LL66] L.D. Landau, E.M. Lifschitz. *Lehrbuch der theoretischen Physik*, Band VI, Hydrodynamik. Akademie–Verlag Berlin, 1966.

[LT96] W. Layton, L. Tobiska. A two–level method with backtracking for the Navier-Stokes equations. Preprint 6/96, Otto-von-Guericke Universität Magdeburg, Fakultät für Mathematik, 1996.

[McC87] S. McCormick, editor. *Multigrid Methods.* SIAM Frontiers in Applied Mathematics 3. Society for Industrial and Applied Mathematics, Philadelphia, 1987.

[OU84] K. Ohmori, T. Ushijima. A technique of upstream type applied to a linear nonconforming finite element approximation of convective diffusion equations. *R.A.I.R.O. Anal. Numer.*, 18(3):309–332, 1984.

[OV85] J.M. Ortega, R.G. Voigt. *Solution of Partial Differential Equations on Vector and Parallel Computers.* SIAM, Philadelphia, 1985.

[PB87] P. Peisker, D. Braess. A conjugate gradient method and a multigrid algorithm for Morley's finite element approximation of the biharmonic equation. *Numer. Math.*, 50:567 – 586, 1987.

[PS72] S.V. Patankar, D.B. Spalding. A calculation procedure for heat and mass transfer in three–dimensional parabolic flows. *Int. J. Heat Mass Transfer*, 15:1787 – 1806, 1972.

[Reu91] A. Reusken. Steplength optimization and linear multigrid methods. *Numer. Math.*, 58:819 – 838, 1991.

[Reu93] A. Reusken. Multigrid with matrix–dependent transfer operators for a singular perturbation problem. *Computing*, 50:199 – 211, 1993.

[Reu94] A. Reusken. Multigrid with matrix–dependent transfer operators for convection–diffusion problems. In P. Hemker, P. Wesseling, editors, *Multigrid Methods IV*, pages 269 – 280. Birkhäuser, 1994.

[Reu95] A. Reusken. A multigrid method based on incomplete Gaussian elimination. RANA 95-13, Eindhoven University of Technology, 1995. to appear in NLAA.

[Reu96] A. Reusken. On a robust multigrid solver. *Computing*, 56:303–322, 1996.

[Riv84] M.C. Rivara. Design and data structure of a fully adaptive multi–grid finite element software. *ACM Trans. on Math. Software*, 10:242 – 264, 1984.

[Roo96] H.-G. Roos. A priori estimates, asymptotic expansions and Shishkin decompositions. Report MATH–NM–21–96, Technical University of Dresden, 1996.

[RS88] U. Risch, F. Schieweck. A multigrid method for solving the stationary Navier– Stokes equations by FEM. *Preprint Math 5/88 TU Magdeburg*, 1988.

[RST96] H.-G. Roos, M. Stynes, L. Tobiska. *Numerical Methods for Singularly Perturbed Differential Equations.* Springer, 1996.

[Rüd96] U. Rüde. Performance aspects of iterative methods on superscalar computers. In *Virtual Proceedings of the 9th International GAMM-Workshop on Parallel Multigrid Methods.* http://www.numa.uni-linz.ac.at/Workshops/proceedings.html, 1996.

[Sch93] F. Schieweck. A parallel multigrid algorithm for solving the Navier–Stokes equations. *Impact Comp. Sci. Engrg.*, 5:345 – 378, 1993.

[Sch96] F. Schieweck. *Parallele Lösung der stationären inkompressiblen Navier–Stokes Gleichungen.* Otto-von-Guericke Universität Magdeburg, Fakultät für Mathematik, 1996. Habilitation.

[SS86] Y. Saad, M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856 – 869, 1986.

[ST96a] M. Schäfer, S. Turek. The benchmark problem "Flow around a cylinder". In E.H. Hirschel, editor, *Flow Simulation with High–Performance Computers II*, Band 52 von *Notes on Numerical Fluid Mechanics.* Vieweg, 1996.

[ST96b] F. Schieweck, L. Tobiska. An optimal order error estimate for an upwind discretization of the Navier–Stokes equations. *Num. Meth. Part. Diff. Equ.*, 12:407 – 421, 1996.

[Tem83] R. Temam. *Navier–Stokes equations and Nonlinear Functional Analysis*, Band 41 von *CBMS–NSF Conf. Ser.* SIAM. Philadelphia, 1983.

[TT89] A. Thiele, L. Tobiska. A weighted upwind finite element method for solving the stationary Navier–Stokes equations. *WZ TU Magdeburg*, 33:13–20, 1989.

[Tur96] St. Turek. A comparative study of time–stepping techniques for the incompressible Navier–Stokes equations: From fully implicit non–linear schemes to semi–implicit projection methods. *Int. J. Num. Meth. Fluids*, 22:987 – 1011, 1996.

[Van86] S. Vanka. Block-implicit multigrid calculation of two-dimensional recirculating flows. *Comp. Meth. Appl. Mech. Eng.*, 59(1):29–48, 1986.

[Ver94a] R. Verfürth. A posteriori error estimates for nonlinear problems. Finite element discretization of elliptic equations. *Math. Comp.*, 62(206):445 – 475, 1994.

[Ver94b] R. Verfürth. A posteriori error estimation and adaptive mesh–refinement techniques. *J. Comp. Appl. Math.*, 50:67 – 83, 1994.

[Ver96a] R. Verfürth. A posteriori error estimates for nonlinear problems. L^r–estimates for finite element discretizations of elliptic equations. Bericht Nr. 199, Ruhr–Universität Bochum, Fakultät für Mathematik, 1996.

[Ver96b] R. Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh–Refinement Techniques.* Wiley and Teubner, 1996.

[Wie65] K. Wieghardt. *Theoretische Strömungslehre*, Band 4 von *Leitfäden der angewandten Mathematik und Mechanik.* Teubner Stuttgart, 1965.

[Wit89a] G. Wittum. Multi–grid methods for Stokes and Navier–Stokes equations. Transforming smoothers: Algorithms and numerical results. *Numer. Math.*, 54:543 – 563, 1989.

[Wit89b] G. Wittum. On the robustness of ILU–smoothing. *SIAM J. Sci. Stat. Comput.*, 10:699–717, 1989.

[Wit90] G. Wittum. On the convergence of multi–grid methods with transforming smoothers. Theory with applications to the Navier–Stokes equations. *Numer. Math.*, 57:15 – 38, 1990.

[Woh95] B. Wohlmuth. *Adaptive Multilevel–Finite–Elemente Methoden zur Lösung elliptischer Randwertprobleme.* Dissertation, Technische Universität München, 1995.

[Zho95] G. Zhou. How accurate is the streamline diffusion finite element method? Preprint 95 - 22 (SFB 359), University of Heidelberg, 1995.

Index