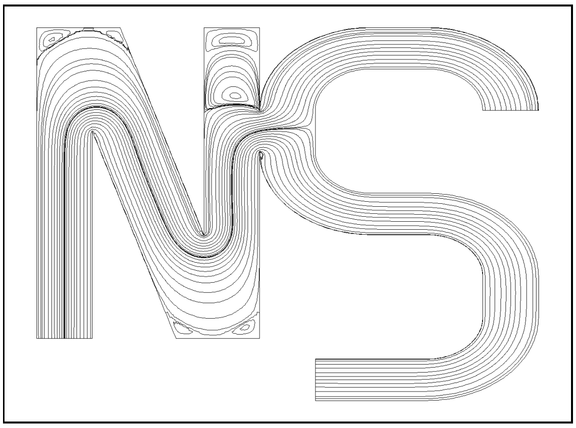
Parallele Lösung der stationären inkompressiblen Navier-Stokes Gleichungen

Friedhelm Schieweck

Fakultät für Mathematik
Otto-von-Guericke-Universität Magdeburg



Inhaltsverzeichnis

Kapitel 1

Einleitung

Gegenstand der vorliegenden Arbeit ist die parallele numerische Lösung der stationären inkompressiblen Navier-Stokes Gleichungen

$$\begin{aligned} -\nu\triangle u + (u\cdot\nabla)u + \nabla p &= f \qquad \text{in } \Omega\,, \\ \nabla\cdot u &= 0 \qquad \text{in } \Omega\,, \\ u &= g \qquad \text{auf } \partial\Omega \end{aligned} \tag{1.1}$$

für die Geschwindigkeit u und den Druck p in einem beschränkten Gebiet $\Omega \subset \mathbf{R}^d$ $(d = 2, 3)$ bei vorgegebener äußerer Kraft f und Dirichlet'schen Randwerten g für die Geschwindigkeit. Der Parameter ν bezeichnet die kinematische Viskosität und ist indirekt proportional zur Reynolds-Zahl $Re = LU/\nu$. Der Einfachheit halber seien hier die charakteristische Länge L und die charakteristische Geschwindigkeit U auf Eins normiert und $Re \equiv 1/\nu$.
Selbst bei diesem recht vereinfachten Modell für die numerische Strömungssimulation entstehen insbesondere im dreidimensionalen Fall bereits beträchtliche Anforderungen an die Computer-Hardware. Dies betrifft sowohl die Rechenleistung als auch die Menge an verfügbarem Speicherplatz. Die ständig wachsenden Anforderungen resultieren einerseits daraus, daß auf Grund höherer Genauigkeitsansprüche die Anzahl der Unbekannten steigt, und andererseits daraus, daß man an kürzeren Antwortzeiten interessiert ist. Beim Übergang zu komplexeren strömungsmechanischen Modellen kommt es zu einem weiteren Anwachsen der Computeranforderungen. Auf lange Sicht können nur Parallelrechner mit einer hinreichend großen Anzahl von Prozessoren diesen Erfordernissen gerecht werden, da die herkömmlichen seriellen Rechner physikalisch bedingte Leistungsgrenzen haben. Es ist daher eine wichtige Aufgabe der Forschung, solche parallelen Algorithmen zu entwickeln, die auch bei einer großen Anzahl von Prozessoren eine akzeptable Effizienz besitzen. In der vorliegenden Arbeit soll hierzu ein Beitrag geleistet werden. Es soll untersucht werden, welche Effekte bei der Parallelisierung eines Mehrgitterverfahrens auftreten, wenn mit wachsender Problemgröße die Anzahl der verwendeten Prozessoren erhöht wird.

Während man durch Verwendung eines hinreichend großen Netzwerkes von Prozessoren mit eigenem Speicher die Gesamtmenge an Speicherplatz prinzipiell beliebig skalieren

kann, ist es eine nichttriviale Aufgabe, bei einer immer größeren Anzahl von Prozessoren tatsächlich auch eine immer höhere Rechenleistung zu erreichen. Da alle Prozessoren gemeinsam an der Lösung eines globalen Problems arbeiten, ist es erforderlich, daß sie sich untereinander "abstimmen". Dadurch kommt es zu Verlustzeiten, in denen die Prozessoren nicht rechnen können. Solche Zeiten entstehen, wenn Daten auszutauschen sind (Kommunikation) oder wenn Prozessoren auf andere Prozessoren warten müssen (Synchronisation). Im allgemeinen wachsen die Verlustzeiten mit der Anzahl der verwendeten Prozessoren. Man ist daran interessiert, daß ihr Anteil an der Gesamtzeit möglichst gering ist, d.h. daß der parallele Algorithmus eine möglichst hohe *parallele Effizienz* besitzt.
Neben der parallelen spielt auch die *numerische Effizienz* des verwendeten Verfahrens eine wichtige Rolle. Bei der Bewertung der Gesamteffizienz eines parallelen Algorithmus ist zum Vergleich stets einer der besten sequentiellen Algorithmen heranzuziehen, die für das jeweilige Problem zur Verfügung stehen. Es wird sich im allgemeinen nicht lohnen, nur auf Grund besserer Parallelisierungseigenschaften einen langsamen Algorithmus zu verwenden, da die Rechenzeitverluste gegenüber einem schnellen Algorithmus meist schon so groß sind, daß sie selbst durch eine sehr gute parallele Effizienz nicht mehr kompensiert werden können. Die numerische Effizienz gibt das Verhältnis der Rechenzeit eines schnellen sequentiellen Vergleichsalgorithmus zur Rechenzeit des parallelen Algorithmus auf einem Prozessor an. Für eine gute numerische Effizienz wäre es ideal, einen solchen schnellen sequentiellen Algorithmus auch als parallelen Algorithmus zu verwenden. Dieser enthält jedoch meist rekursive Datenabhängigkeiten, die bei der parallelen Implementierung zu einer sehr schlechten parallelen Effizienz führen würden. Um zu einer akzeptablen parallelen Effizienz zu kommen, muß also der parallele Algorithmus gegenüber dem sequentiellen modifiziert werden. Dies führt im allgemeinen zu einer Verschlechterung in der Konvergenzgeschwindigkeit. Bei der Auswahl des parallelen Algorithmus ist es deshalb wichtig, einen geeigneten Kompromiß zwischen numerischer und paralleler Effizienz zu finden.

Die in der Literatur bekannten parallelen Algorithmen für partielle Differentialgleichungen basieren fast alle auf einer Zerlegung des Gebietes in Teilgebiete, von denen jedes einem anderen Prozessor zugeordnet wird. Jeder Prozessor bearbeitet und speichert dabei nur die Daten, die zu dem betreffenden Teilgebiet gehören, und eventuell noch jene in einem gewissen Überlappungsbereich mit den benachbarten Teilgebieten. Von der numerischen Seite her unterscheiden wir zwei Richtungen.

Bei der ersten Klasse von Verfahren gibt es eine äußere Iteration, wie z.B. die additive oder multiplikative Schwarz-Iteration oder eine CG- bzw. GMRES-Iteration mit geeigneter Block-Vorkonditionierung (siehe [10], [11], [9], [26]). In jedem Iterationschritt werden lokale Probleme auf den Teilgebieten (mit eventuellem Überlappungsbereich) gelöst und anschließend die lokalen Lösungen zu einer neuen Iterierten kombiniert. Als "innere" Löser für die lokalen Teilprobleme kann man effiziente Mehrgitter-Verfahren einsetzen [25]. Ein Vorteil dieses Zugangs ist, daß die Teilprobleme auf den einzelnen Prozessoren völlig unabhängig voneinander gelöst werden können. Kommunikation ist nur erforderlich, um die lokalen Lösungen miteinander zu kombinieren. Damit sind die

Effizienzverluste durch Kommunikation relativ gering. Ein Nachteil ist, daß im Vergleich zum besten sequentiellen Algorithmus (z.B. einem globalen Mehrgitter-Verfahren) Verluste an numerischer Effizienz durch die äußere Iteration und durch eventuelle Mehrfachberechnungen auf den Überlappungsbereichen entstehen.

Die Verfahren der zweiten Klasse werden häufig mit dem Begriff "Datenpartitionierung" oder "Gitterpartitionierung" charakterisiert. Bei der Konstruktion des parallelen Algorithmus versucht man, soweit wie möglich einen effizienten sequentiellen Algorithmus zu übernehmen (z.B. ein globales Mehrgitter-Verfahren), und verteilt die anfallenden Operationen entsprechend der Teilgebiete auf die Prozessoren. Die Berechtigung für diesen Zugang resultiert daraus, daß die meisten Operationen des sequentiellen Algorithmus (z.B. Generierung der Steifigkeitsmatrix, Matrix-Vektor-Multiplikation oder Prolongation und Restriktion in einem Mehrgitter-Verfahren) unabhängig voneinander für die verschiedenen Teilgebiete ausgeführt werden können. Lediglich die Operationen mit Daten am Rand der Teilgebiete oder solche mit globalen Daten (z.B. die Berechnung einer globalen Norm oder die Lösung eines globalen Grobgitterproblems) beinhalten eine Kopplung zu den Daten anderer Teilgebiete und machen eine entsprechende Kommunikation erforderlich. Diejenigen Komponenten des sequentiellen Algorithmus, die rekursive Datenabhängigkeiten zwischen den Teilgebieten beinhalten, sind im parallelen Algorithmus zu modifizieren. Beispiele solcher Komponenten im Mehrgitter-Verfahren sind Gauß-Seidel-artige oder ILU-Glätter. Mit Hilfe einer Block-Jacobi-Strategie, bei der die Blöcke den Teilgebieten entsprechen, werden im parallelen Algorithmus rekursive Datenabhängigkeiten auf die Daten der Teilgebiete beschränkt. Damit können die Prozessoren jeweils unabhängig voneinander ihre "eigenen" Unbekannten neu berechnen. Diese Block-Jacobi-Modifikation führt im Vergleich zum sequentiellen Algorithmus zu einer Reduzierung in der Konvergenzgeschwindigkeit. Jedoch sind bei einem Mehrgitter-Verfahren, wo sich die Modifikation nur auf das Glätten hochfrequenter Fehleranteile bezieht, die entstehenden Verluste an numerischer Effizienz relativ gering. Die Verfahren der Datenpartitionierung haben im Vergleich zu denen der ersten Klasse in der Regel eine bessere numerische Effizienz. Dieser Vorteil wird allerdings erkauft durch einen wesentlich höheren Aufwand an Kommunikation. Hier ist in fast jeder Phase des Algorithmus ein lokaler Austausch von Daten erforderlich.

Beide Klassen von Algorithmen haben ihre Vor- und Nachteile. Welcher Zugang der bessere ist, hängt vom konkreten Problem und in entscheidendem Maße von den Hardware-Eigenschaften des verwendeten Parallelrechners ab. Bei einem Rechner mit einer im Vergleich zur Rechenleistung schnellen Kommunikation werden die Verfahren der zweiten Klasse Vorteile haben. Ist jedoch die Kommunikationsgeschwindigkeit sehr gering (wie z.B. bei einem Workstation-Cluster), so sind die Verfahren der ersten Klasse bevorteilt.

Für strömungsmechanische Probleme ist die Datenpartitionierung basierend auf einer Gebietszerlegung die zur Zeit vorherrschende Technik zur Konstruktion paralleler Algorithmen ([35],[31],[72]). In dieser Arbeit werden wir ebenfalls diesen Zugang wählen. Zur Diskretisierung verwenden wir Finite Elemente und zur Lösung der entstehenden Gleichungssysteme ein Mehrgitter-Verfahren. Unsere Gebietszerlegung definieren wir mit Hilfe eines groben Gitters bestehend aus viereckigen bzw. hexaederförmigen Elementen, die wir Makroelemente nennen. Das eigentliche Finite-Elemente-Gitter entsteht durch uniforme Verfeinerung jedes dieser Makroelemente. Im Fall krummliniger Randteile wird der Verfeinerungsprozeß so modifiziert, daß für Elemente am Rand die neu entstehenden Ecken der Sohnelemente jeweils auf den Rand verschoben werden. Insgesamt erhalten wir ein blockstrukturiertes Gitter, welches auf jedem Makroelement eine sehr regelmäßige Struktur besitzt, aber dennoch völlig unstrukturiert hinsichtlich der Makroelementzerlegung sein kann. Durch die Wahl eines geeigneten Grobgitters kann damit eine statische, adaptive Gitteranpassung auf der Basis von a priori Wissen realisiert werden. Blockstrukturierte Gitter sind bei der Berechnung inkompressibler Strömungen recht verbreitet ([41],[42],[18],[46],[60],[53],[68],[35],[72]). Während sie meist in Verbindung mit Finite Volumen oder Finite Differenzen Verfahren eingesetzt werden, wollen wir hier ihre Eignung im Falle Finiter Elemente insbesondere unter dem Aspekt der Parallelisierung untersuchen.

Der Gebietszerlegung in Makroelemente $\{M_i\}$ ordnen wir ein entsprechendes Tasknetzwerk $\{T_i\}$ zu, bei dem die Task T_i jeweils diejenigen Daten bearbeitet und speichert, die zu sämtlichen Sohnelementen des Makroelementes M_i gehören. Die Tasks werden wiederum einer Menge von Prozessoren zugeordnet. Wenn genügend viele Prozessoren zur Verfügung stehen, verwenden wir für jede Task einen eigenen Prozessor. Ansonsten ist ein Lastverteilungsproblem zu lösen, um eine geeignete Abbildung der Tasks auf die Prozessoren zu finden. Heuristische Strategien zur Lastverteilung, die besonders bei adaptiven Methoden von großer Bedeutung sind, findet man in [1],[2],[14],[44]. In dieser Arbeit vernachlässigen wir das Problem der Lastverteilung, indem wir uns auf den Fall beschränken, daß für jede Makroelement-Task ein eigener Prozessor zur Verfügung steht. Da alle Makroelemente dieselbe Anzahl von Elementen besitzen, ergibt sich dann eine optimale Lastverteilung.

Die Effizienz des parallelen Algorithmus wird wesentlich beeinflußt von dem Ausmaß an erforderlicher Kommunikation. Bei der Finite Elemente Methode ist es sinnvoll, die Elemente und nicht etwa die Knoten auf die Prozessoren zu verteilen. Dies führt dazu, daß die Datenpartitionierung nicht disjunkt ist, denn die Unbekannten zu einem Knoten werden auf allen Prozessoren gehalten, denen ein Element mit diesem Knoten zugeordnet wurde. Damit haben zwei Prozessoren Daten zu kommunizieren, sobald die ihnen zugeordneten Teilgebiete mindestens einen gemeinsamen Knoten besitzen. Die Wahl der Finiten Elemente hat dabei Einfluß auf das Ausmaß der Kommunikationsanforderungen. Verwenden wir konforme Viereckelemente, welche Knoten in den Ecken haben, so gibt es zu einem inneren Makroelement M_i in der Regel 8 benachbarte Makroelemente, die mit M_i gemeinsame Knoten besitzen, d.h. bei einem lokalen Datenaustausch hat die Task T_i mit 8 anderen Tasks zu kommunizieren. Im Fall konformer Hexaederelemente sind es bereits 26 Kommunikationsanforderungen, die eine Task im allgemeinen bei einem lokalen Datenaustausch zu erfüllen hat. Finite Elemente Ansätze mit Knoten in den Ecken der Elemente führen also insbesondere im 3D-Fall zu hohen Kosten bei der lokalen Kommunikation. Es entsteht die Frage, ob man nicht Ansätze verwenden kann, bei denen die Knoten den Seiten der Elemente (Kanten im 2D-Fall, Flächen im 3D-Fall) zugeordnet sind. Dadurch würde sich die Anzahl der benachbarten Makroelemente, die gemeinsame Knoten mit einem gegebenen Makroelement haben, deutlich verringern. Solche Ansätze wurden im Fall von Viereck- bzw. Hexaederelementen von Rannacher und Turek [48] vorgeschlagen. Sie stellen ein Analogon zu den bekannten nichtkonformen Dreieck- bzw. Tetraederelementen von Crouzeix/Raviart (siehe [13]) dar. Die Geschwindigkeitsknoten liegen hier in den Mitten der Elementseiten und die Druckknoten in den Schwerpunkten der Elemente. Damit gibt es zu einem inneren Makroelement M_i im 2D-Fall 4 und im 3D-Fall nur 6 benachbarte Makroelemente, die gemeinsame Knoten mit M_i haben. Im Vergleich zu den konformen Viereck- bzw. Hexaederelementen erfordern diese Elemente also einen deutlich geringeren Aufwand beim lokalen Datenaustausch. Ein weiterer Vorteil ist, daß sie im Gegensatz zu den konformen Q_1/Q_0- bzw. Q_1/Q_1-Elementen stabil im Sinne von Babuška-Brezzi sind. Ferner hat Turek ([69],[70]) gezeigt, daß im instationären Fall die nichtkonformen Elemente sehr gut geeignet sind für die Realisierung diskreter Projektionsverfahren. Aus den genannten Gründen werden wir für die parallele numerische Lösung der Navier-Stokes-Gleichungen die von Rannacher und Turek vorgeschlagenen, nichtkonformen Viereck- bzw. Hexaederelemente einsetzen.

Von diesen Elementen wurden in [48] vier verschiedene Varianten angegeben. Sie entstehen daraus, daß man zwei Varianten für die Wahl der Knotenfunktionale mit zwei Varianten für die Parametrisierung der Ansatzfunktionen kombinieren kann. Bei den Knotenfunktionalen kann man den Funktionswert im Knoten (punktwert-orientierte Elemente) oder den Integralmittelwert auf der dem Knoten zugeordneten Elementseite (mittelwert-orientierte Elemente) verwenden. Die Ansatzfunktionen können mit Hilfe einer Transformation auf ein Referenzelement (parametrische Elemente) oder unter Verwendung eines, dem Element zugeordneten, lokalen Koordinatensystems (nichtparametrische Elemente) definiert werden. In [48] wurde nachgewiesen, daß bei den parametrischen Elementen der Interpolationsfehler auf einem Element K von einer Winkelgröße σ_K abhängt, die die Abweichung des Elementes von der Form eines Parallelogramms bzw. eines Parallelepipeds beschreibt. Für die nichtparametrischen Elemente wurden optimale Abschätzungen des Interpolationsfehlers angegeben, die keine Winkelabhängigkeit enthalten. Ein Beweis, daß die entsprechenden Fehlerkonstanten unabhängig vom jeweiligen Element sind, fehlte jedoch. In Kapitel 2 wird ein theoretischer Zugang für die Abschätzung des Interpolationsfehlers vorgestellt, mit dem sowohl die parametrischen als auch die nichtparametrischen Elemente einheitlich behandelt werden können. Die Idee besteht darin, die nichtparametrischen Elemente als "P_1-parametrische" Elemente zu interpretieren. Dabei wird das Originalelement mit Hilfe einer affinen (P_1-parametrischen) Abbildung auf ein Referenzelement transformiert, auf dem dann in üblicher Weise die Basisfunktionen definiert werden. Da sich nun die Referenzelemente mit dem Originalelement ändern können, wurde die übliche Interpolationstheorie für Finite Elemente (Ciarlet [12]) auf den Fall variabler Referenzelemente erweitert. Ebenso wurden für diesen Fall lokale inverse Ungleichungen bewiesen, die in der Konvergenzanalysis benötigt werden.

Um auch im Falle hoher Reynolds-Zahlen stabile numerische Lösungen zu erhalten, verwenden wir zur Diskretisierung des Konvektionstermes eine upwind-Technik. Sie wurde in [57] für die Crouzeix/Raviart-Elemente vorgeschlagen und läßt sich problemlos auf die nichtkonformen Viereck- bzw. Hexaederelemente übertragen ([67],[54]). Eine alternative Variante zur Stabilisierung ist die Stromlinien-Diffusions-Methode ([36],[37],[66]). Gegenüber dem upwind-Verfahren hat sie zwar eine höhere Genauigkeit, erfordert jedoch in jedem Schritt der nichtlinearen Iteration einen deutlich höheren Aufwand beim Aufbau der Steifigkeitsmatrix. Außerdem hängt die Qualität der numerischen Lösung von der geeigneten Wahl gewisser Parameter ab. Bei der upwind-Diskretisierung kann der sich ändernde, dem Konvektionsterm entsprechende Teil der Steifigkeitsmatrix sehr schnell generiert werden. Das entstehende Gleichungssystem hat günstige Eigenschaften, die sich positiv auf den Mehrgitterlöser auswirken. In praktischen Rechnungen erwies sich das upwind-Verfahren als sehr stabil und produzierte auch bei hohen Reynolds-Zahlen auf relativ groben Gittern physikalisch sinnvolle Lösungen. Der Nachteil dieser Diskretisierung liegt darin, daß sie nur von erster Ordnung genau ist. Dieser Mangel könnte durch Defektkorrektur mit einem Verfahren höherer Ordnung behoben werden.

Ein wichtiges Ziel dieser Arbeit besteht darin, eine Konvergenzanalysis für die upwind-Diskretisierung der Navier-Stokes-Gleichungen im Falle der nichtkonformen Viereck- bzw. Hexaederelemente bereitzustellen. Eine theoretische Analyse dieser upwind-Diskretisierung gibt es bisher nur für den Fall von Crouzeix/Raviart-Elementen ([57], [58]). Konvergenzaussagen für die nichtkonformen Viereck- bzw. Hexaederelemente wurden lediglich für das Stokes-Problem bewiesen (siehe [48]). Das Hauptproblem bei der Übertragung der für die Crouzeix/Raviart-Elemente existierenden Analysis bestand darin, daß die diskretisierte schwache Form des Konvektionstermes nicht mehr positiv ist. Durch die Einführung einer Modifikation in der upwind-Diskretisierung konnte die Positivitätseigenschaft gesichert werden. Da die Modifikation derart gewählt wurde, daß sie keine Verschlechterung in der Ordnung des Konsistenzfehlers bewirkt, ließen sich hierfür analoge Konvergenzaussagen zu den Crouzeix/Raviart-Elementen zeigen. Für die ursprüngliche upwind-Diskretisierung ergaben sich schwächere Resultate, die die Voraussetzung einer hinreichend kleinen Gitterweite erfordern.
Die Konvergenzanalysis soll so angelegt werden, daß sie den Fall adaptiv verfeinerter Gitter mit einschließt und in der Lage ist, eine theoretische Rechtfertigung für lokale Gitteranpassung zu geben. Hierzu werden adaptiv gewichtete, diskrete Sobolev-Normen eingeführt. Mit ihrer Hilfe erhalten wir eine Abschätzung des Diskretisierungsfehlers, die bei geeigneter adaptiver Gitterverfeinerung auch dann noch eine vernünftige Fehlerschranke liefert, wenn die entsprechenden globalen Sobolev-Normen der kontinuierlichen Lösung groß sind. Betrachten wir die Fehlerabschätzung für die vier verschiedenen Varianten der nichtkonformen Viereck- bzw. Hexaederelemente, so sind die mittelwert-orientierten, P_1-parametrischen Elemente die einzigen, bei denen die Fehlerschranke unabhängig von den oben erwähnten Winkelgrößen σ_K der Elemente ist. Für adaptiv verfeinerte Gitter, die sehr kleine Elemente mit einer Winkelgröße σ_K in der Größenordnung von Eins besitzen, sind deshalb die mittelwert-orientierten, P_1-parametrischen Elemente die einzigen, die eine sinnvolle Fehlerabschätzung garantieren. Bei blockstrukturierten Gittern, die durch uniforme Verfeinerung einer groben Makroelementzerlegung entstehen, können wir beweisen, daß sich die Winkelgröße σ_K in der Form $\sigma_K \leq Ch_K$ durch den Elementdurchmesser h_K abschätzen läßt. Daraus erhalten wir, daß alle vier Varianten der nichtkonformen Elemente auf blockstrukturierten Gittern dieselbe Feh-

lerordnung liefern, so daß man sich die vom Aufwand her günstigste Variante aussuchen kann. Diese Aussage gilt auch für adaptiv verfeinerte Gitter, die durch lokale Verfeinerung eines blockstrukturierten Gitters mit Hilfe von hängenden Knoten entstehen.

Zur Herleitung der Fehlerabschätzungen benötigen wir die Voraussetzung, daß der Ausdruck $\nu^{-2}\|f\|_{L^2(\Omega)}$ hinreichend klein ist. Damit werden praktisch interessante Fälle mit $O(1)$-Daten und hoher Reynolds-Zahl nicht erfaßt. Um dennoch Aussagen über die Größe des Diskretisierungsfehlers im Falle hoher Reynolds-Zahlen Re machen zu können, studieren wir in Kapitel 5 an einem akademischen Testbeispiel mit bekannter kontinuierlicher Lösung das Konvergenzverhalten in Abhängigkeit von Re. Für dieses Beispiel zeigt sich, daß der Geschwindigkeitsfehler in einem gewissen Re-Bereich wie $O(Re)$ wächst. Ferner stellt sich heraus, daß der Interpolationsfehler des Druckes mit dem Faktor Re in den Diskretisierungsfehler der Geschwindigkeit eingeht. Mit der Methode der Druckabspaltung geben wir eine Möglichkeit an, wie wir in diesem Falle die numerische Lösung erheblich verbessern können.

Neben der Analyse der upwind-Diskretisierung ist die parallele Lösung des entstehenden Gleichungssystems ein weiteres wichtiges Anliegen dieser Arbeit. Dabei wollen wir das Konzept der Datenpartitionierung basierend auf einer Makroelementzerlegung des Gebietes mit möglichst hoher Effizienz bei der Parallelisierung eines sequentiellen Mehrgitterverfahrens umsetzen. Unser Ziel besteht darin, die Technologie der Parallelisierung - insbesondere die Realisierung der lokalen und globalen Kommunikation - so zu entwickeln, daß sie auf beliebige Makroelementzerlegungen im 2D- und 3D-Fall anwendbar ist. Bei der konkreten Implementierung beschränken wir uns jedoch auf den 2D-Fall. Weiterhin wollen wir untersuchen, inwieweit unser Konzept im massiv-parallelen Fall (bis zu 1024 Prozessoren) tragfähig ist und welche Modifikationen hier gegebenenfalls erforderlich sind.

Um eine akzeptable parallele Effizienz zu erhalten, müssen bei der Parallelisierung die im sequentiellen Mehrgitter-Algorithmus enthaltenen, rekursiven Datenabhängigkeiten zwischen den Unbekannten zu verschiedenen Makroelementen vermieden werden. Derartige Abhängigkeiten kommen im Glätter sowie im Grobgitterlöser vor. Der Glätter, welcher auf einem von Vanka [71] vorgeschlagenen, blockweisen Gauß-Seidel Verfahren beruht, wird im parallelen Algorithmus folgendermaßen modifiziert. Jede Makroelement-Task T_i berechnet zunächst unabhängig von den anderen Tasks ihre "eigenen", zum Makroelement M_i gehörenden Unbekannten neu. Für Unbekannte, die zur gemeinsamen Seite zweier benachbarter Makroelemente M_i und M_j gehören, produzieren die entsprechenden Tasks T_i und T_j dabei unterschiedliche Werte. Diese werden nach einem lokalen Datenaustausch durch Mittelwertbildung abgeglichen. Anhand numerischer Testrechnungen zeigen wir, daß diese Modifikation des sequentiellen Glätters selbst bei einer hohen Anzahl von Makroelementen lediglich zu einer geringfügigen Verschlechterung in der Konvergenzrate des Mehrgitterverfahrens führt. Um beim Grobgitterlöser rekursive Datenabhängigkeiten zwischen den Makroelement-Tasks zu vermeiden, führen wir eine zusätzliche Solver-Task ein, so daß sämtliche voneinander abhängigen Daten in einer Task konzentriert sind. Bei der Lösung eines Grobgitterproblems wird der Solver-Task durch globale Kommunikation der Makroelement-Tasks T_i ein rechte-Seite-Vektor zugeschickt. Daraufhin berechnet die Solver-Task den zugehörigen Lösungsvektor und schickt ihn an das Tasknetzwerk $\{T_i\}$ zurück, wo er wiederum durch globale Kommunikation an alle Makroelement-Tasks verteilt wird.

Der entstehende parallele Mehrgitter-Algorithmus zeichnet sich einerseits durch eine gute numerische Effizienz, aber andererseits auch durch einen hohen Aufwand an Kommunikation aus. Es ist daher wichtig, daß die erforderlichen Kommunikationsoperationen möglichst effektiv umgesetzt werden. Für die lokale Kommunikation geben wir einen graphentheoretischen Zugang an, mit dem man in vielen Anwendungsfällen bereits die optimale Anzahl von Kommunikationstakten erreicht. Die globale Kommunikation definieren wir mit Hilfe einer rekursiven Zerlegung der Menge der Makroelemente, die einer rekursiven Gebietszerlegung entspricht. Wir können zeigen, daß auf diese Weise eine "send-up"-Operation (Informationen von allen Tasks sind in einer Task bereitzustellen) oder eine "send-down"-Operation (die Information einer Task ist an alle Tasks zu verschicken) in einer minimalen Anzahl von Kommunikationsschritten realisiert wird.

In Kapitel 8 untersuchen wir, wie sich die parallele Effizienz verhält, wenn wir das feinste Gitterlevel und die Anzahl der Makroelement-Tasks verändern. Effizienzverluste werden bei unserem Parallelisierungsansatz durch lokale und globale Kommunikation sowie durch den nichtparallelisierten Grobgitterlöser verursacht. Ihre Höhe hängt wesentlich von den Hardware-Eigenschaften des verwendeten Parallelrechners, insbesondere von dem Verhältnis zwischen Kommunikations- und Rechenleistung ab. Dies wird durch einen Vergleich der beiden Parsytec-Rechner GCel und GCPP verdeutlicht. Die Effizienzverluste sind um so geringer, je höher das feinste Gitterlevel, d.h. je größer die einem Prozessor zugeordnete Anzahl von Unbekannten ist.

Mit wachsender Anzahl von Makroelementen werden die durch den Grobgitterlöser verursachten Effizienzverluste dominant und führen zu einem drastischen Absinken der Gesamteffizienz. Um diesem Effekt entgegenzuwirken, ist es erforderlich, den Anteil des Grobgitterlösers an der Gesamtzeit zu reduzieren. Zu diesem Zweck untersuchen wir zunächst die Möglichkeit, den bisher verwendeten augmented-Lagrange-Löser durch ein anderes Iterationsverfahren zu ersetzen. Dabei betrachten wir als alternative Lösungsverfahren den Vanka-Glätter, das SIMPLE-Verfahren und den trivialen Löser, bei dem die Makroelement-Tasks die Grobgitterkorrektur für das gröbste Gitter einfach weglassen. Es zeigt sich, daß ab einer gewissen Anzahl von Makroelementen alle nicht-trivialen Grobgitterlöser so teuer werden, daß der triviale Löser trotz seiner schlechten Konvergenzrate die kürzeste Gesamtzeit liefert.
Eine weitere Möglichkeit, den Zeitanteil für den Grobgitterlöser zu reduzieren, besteht darin, anstelle einer Solver-Task mehrere, parallel arbeitende Solver-Tasks einzusetzen. Aus einem anderen Blickwinkel heraus wurde in [32] ein ähnlicher Vorschlag gemacht. Hierbei ging es darum, bei der Lösung des Grobgitterproblems die Menge aller Tasks durch "Agglomeration" auf eine kleinere Menge zu reduzieren. Wir stellen eine Variante zur Parallelisierung des Grobgitterlösers vor, die sich in natürlicher Weise aus der rekursiven Gebietszerlegung ergibt, welche bereits bei der Definition der globalen Kommunikation verwendet wurde. Jedem Teilgebiet Ω_k der Gebietszerlegung auf einer gewissen Stufe ordnen wir dabei eine Solver-Task S_k zu, die für sämtliche Operationen des

Grobgitterlösers mit den zu Ω_k gehörenden Daten zuständig ist. Bei dieser Wahl der Gebietszerlegung ergeben sich Vorteile für die Realisierung der Kommunikation zwischen den Makroelement-Tasks und den Solver-Tasks. Als Varianten des Grobgitterlösers parallelisieren wir auf diese Weise den Vanka-Glätter und das SIMPLE-Verfahren. Im Vergleich zu den entsprechenden Versionen mit einer Solver-Task führt die Parallelisierung des Grobgitterlösers zu einer wesentlichen Verbesserung in der Gesamtzeit. Während bei einer Solver-Task das SIMPLE-Verfahren dem Vanka-Glätter überlegen ist, lieferte bei den parallelisierten Varianten im massiv-parallelen Fall der Vanka-Glätter die besseren Resultate. Weiterhin stellte sich heraus, daß für den parallelisierten Vanka-Glätter die beste Rechenzeit jeweils mit der größten Anzahl verwendeter Solver-Tasks erreicht wurde. Dies führte auf die Idee, den Extremfall auszutesten, bei dem jedes Teilgebiet Ω_k bei der Parallelisierung des Vanka-Glätters nur aus einem Makroelement besteht. In diesem Fall ist es nicht nötig, Solver-Tasks einzurichten, da ihre Aufgaben von den Makroelement-Tasks mit übernommen werden können.

Anhand numerischer Testrechnungen vergleichen wir die verschiedenen vorgeschlagenen Varianten unseres parallelen Navier-Stokes Lösers, wobei wir die Anzahl der verwendeten Prozessoren im Bereich von 4 bis 1024 variieren. Dabei beschränken wir uns auf das driven-cavity-Problem, da wir hier die Anzahl der Teilgebiete systematisch verändern und so das Effizienzverhalten in Abhängigkeit von der Prozessoranzahl gut studieren können. Unsere Methoden sind jedoch so konzipiert, daß sie auf beliebig komplexe Geometrien anwendbar sind. Abbildung 1.1 zeigt beispielsweise den Stromlinienverlauf bei der Umströmung zweier fast geschlossener Halbschalen und eines viereckigen Hindernisses in einem Kanal ($Re = 500$). Die zugehörige Gebietszerlegung in 152 Makroelemente ist in Abbildung 1.2 dargestellt. Bei einer Gesamtanzahl von 775 296 Unbekannten (5 uniforme Verfeinerungen der Makroelementzerlegung) und unter Verwendung von 93 Prozessoren MPC601 des Parsytec-Rechners GCPP (12.8 MFlops pro Prozessor) dauerte die gesamte Rechnung 183 Sekunden. Eine Workstation HP 9000/735 (13.2 MFlops) benötigt für den entsprechenden sequentiellen Algorithmus 4485 Sekunden. Obwohl hier die Gesamteffizienz auf Grund der nichtoptimalen Lastverteilung und der relativ geringen Auslastung der recht leistungsstarken Prozessoren (nur etwa 5200 Unbekannte pro Makroelement-Task) noch nicht sehr hoch ist, erreichen wir bereits einen beachtlichen Rechenzeitgewinn.

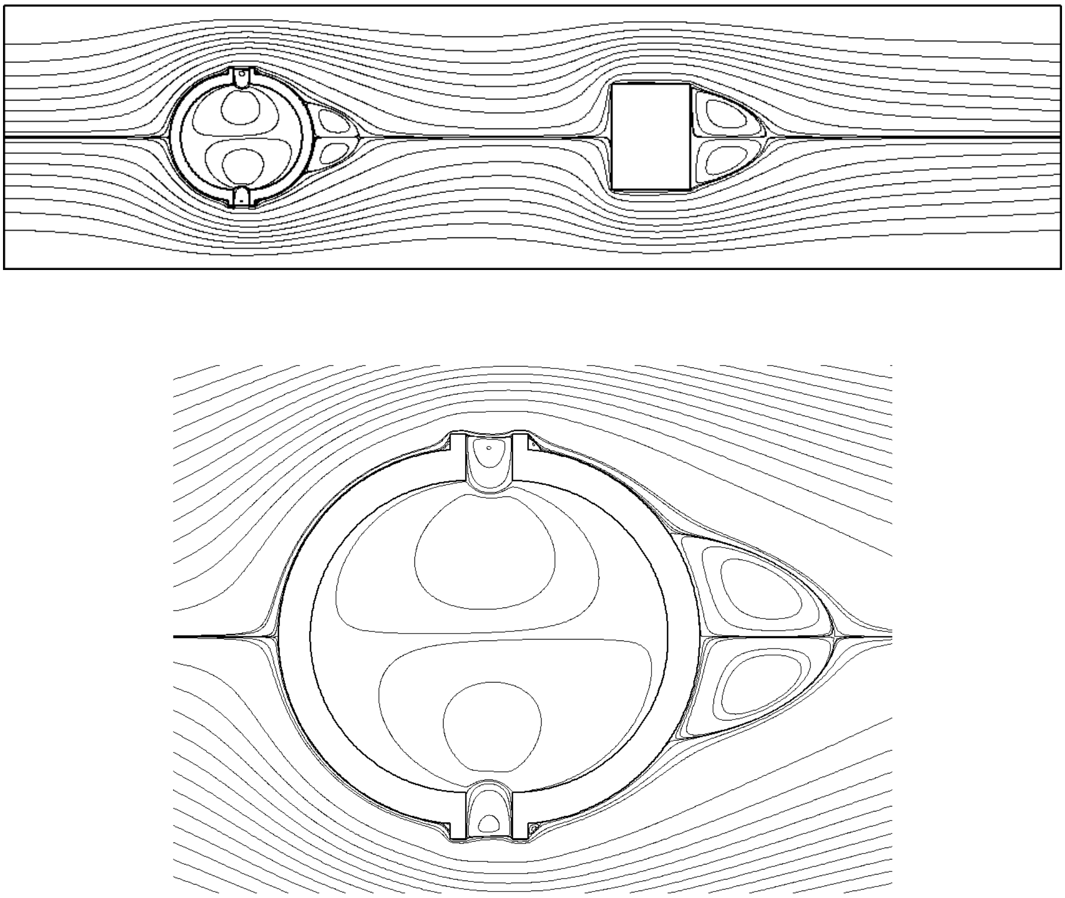


Abbildung 1.1: *Stromlinien bei der Umströmung von Hindernissen in einem Kanal (oben) und vergrößerte Darstellung eines Ausschnittes (unten)*

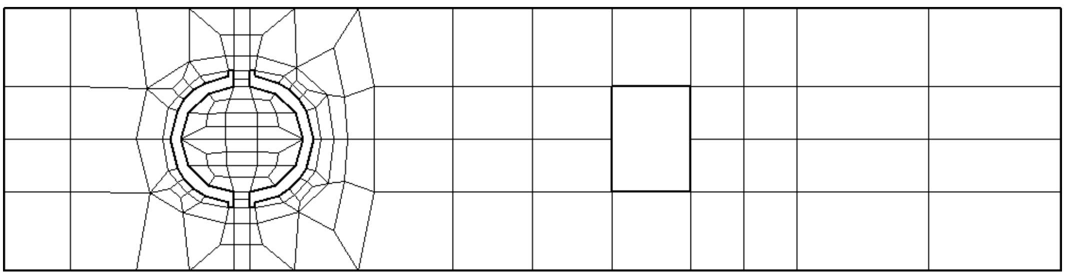


Abbildung 1.2: *Gebietszerlegung in 152 Makroelemente für das Strömungsproblem in Abbildung 1.1*

Kapitel 2

Nichtkonforme Viereck- und Hexaederelemente

Suchen wir für Geschwindigkeit und Druck ein Elementepaar, das eine möglichst geringe Zahl an Freiheitsgraden hat und dennoch stabil im Sinne von Babuška-Brezzi ist, so kommen wir auf das nichtkonforme Dreieckelement (Tetraederelement in 3D) von Crouzeix/Raviart [13] oder die von Rannacher und Turek [48] vorgeschlagenen nichtkonformen Viereckelemente (Hexaederelemente in 3D). Wir verwenden die nichtkonformen Viereck- bzw. Hexaederelemente, da hier das Verhältnis zwischen der Anzahl der Unbekannten und der erreichten Genauigkeit günstiger ist. Im 2D-Fall werden diese Elemente als *rotiert bilinear* bezeichnet, denn die Ansatzfunktionen für die Geschwindigkeit sind in gedrehten lokalen Koordinaten bilinear. Für den 3D-Fall ist diese Charakterisierung jedoch nicht richtig. Treffender wäre vielleicht der Name "nichtkonformes Seitenelement (face element)", da die Freiheitsgrade stets eineindeutig den Seiten der Elemente (Kanten im 2D-Fall, Flächen im 3D-Fall) zugeordnet sind. Damit gehört ein Geschwindigkeitsknoten, der nicht auf dem Gebietsrand liegt, zu genau zwei Elementen. Diese Tatsache erweist sich als vorteilhaft bei der Parallelisierung, da sich dadurch die lokale Kommunikation (zwischen Prozessoren zu benachbarten Teilgebieten) im Vergleich zu konformen Viereck- bzw. Hexaederelementen stark vereinfacht.

Auf der theoretischen Seite handeln wir uns jedoch mit den nichtkonformen Viereck- bzw. Hexaederelementen einige Schwierigkeiten ein. Rannacher und Turek [48] haben bereits gezeigt, daß im Falle der *parametrischen* nichtkonformen Elemente (die mit Hilfe einer bi- bzw. trilinearen Transformation auf dem Referenzelement definiert werden) die Abschätzung des Interpolationsfehlers auf einem Element K nicht nur Potenzen des Elementdurchmessers h_K enthält, sondern auch noch eine Winkelgröße σ_K. Diese Größe σ_K charakterisiert die Winkelabweichung des Elementes K von einem Parallelogramm bzw. Parallelepiped. Optimale Interpolations- und Konvergenzeigenschaften würden wir nur auf einem Gitter erhalten, bei dem für alle Elemente K eine gleichmäßige Abschätzung der Gestalt $\sigma_K \leq Ch_K$ gelten würde. Die Frage, auf welchen Gittern eine solche Abschätzung erfüllt ist, wurde bisher nicht untersucht. In Abschnitt 4.8 beweisen wir, daß sie für blockstrukturierte Gitter gilt, die durch uniforme Verfeinerung aus einer groben Makroelementzerlegung hervorgehen. Gerade diese Klasse von Gittern ist es auch, die wir für unsere Parallelisierungsstrategie benutzen.

Auf allgemeinen Gittern sind die parametrischen Elemente jedoch ungeeignet. Dies ist insbesondere für adaptiv verfeinerte Gitter der Fall, bei denen es Elemente K gibt, für die h_K sehr klein, aber $\sigma_K = O(1)$ ist. Deshalb haben Rannacher und Turek sogenannte *nichtparametrische Elemente* vorgeschlagen, die ohne Transformation auf das Referenzelement definiert werden. Hierzu wird auf jedem Element ein lokales schiefwinkliges Koordinatensystem eingeführt und anschließend jede Basisfunktion aus gewissen Polynomen dieser lokalen Koordinaten bestimmt. Für die so definierten Elemente wurden in [48] optimale Abschätzungen des Interpolationsfehlers angegeben, die keinerlei Abhängigkeit von der Winkelgröße σ_K beinhalten. Zum Beweis wurde jedoch nur vermerkt, daß man das Bramble-Hilbert Lemma auf jedem Element anwenden kann. Ein strenger Nachweis, daß alle in den Abschätzungen vorkommenden Konstanten vom konkreten Element K unabhängig sind, fehlte. Offene Fragen sind dabei zum Beispiel die Unisolvenz und Abschätzungen der Basisfunktionen auf einem Element K mit einer K-unabhängigen Konstanten.

In diesem Kapitel wollen wir unter anderem auch diese Fragen klären. Wir werden unsere Theorie so entwickeln, daß sich damit sowohl die beiden parametrischen (punktwert- und mittelwert-orientierte) als auch die beiden nichtparametrischen Elemente einheitlich behandeln lassen. Die wesentliche Idee dabei ist, die nichtparametrischen Elemente nach einer kleinen Modifikation als sogenannte *P_1-parametrische Elemente* mit Hilfe einer affinen (P_1-parametrischen) Referenztransformation zu interpretieren. Zu unterschiedlichen Originalelementen ergeben sich dann allerdings auch unterschiedliche Referenzelemente. Deshalb mußte die übliche Interpolationstheorie Finiter Elemente [12] auf den Fall variabler Referenzelemente verallgemeinert werden. In Abschnitt 2.3 geben wir allgemeine Bedingungen an, unter denen die entsprechenden Konstanten in den Abschätzungen des lokalen Interpolationsfehlers unabhängig vom konkreten Element sind. Bei der Beschreibung der vier verschiedenen nichtkonformen Elemente in Abschnitt 2.2 werden jeweils entsprechende Aussagen bereitgestellt, die zur Verifikation dieser Bedingungen dienen. In Abschnitt 2.4 zeigen wir lokale inverse Ungleichungen, die wir später für unsere Konvergenzanalysis benötigen. Der Beweis ist so angelegt, daß er den allgemeinen Fall variabler Referenzelemente erfaßt und somit die vier nichtkonformen Elemente einheitlich behandelt werden können. Zunächst wollen wir einige Bezeichnungen einführen und gewisse Vereinbarungen treffen.

2.1 Vereinbarungen und Bezeichnungen

Sei $\Omega \subset \mathbf{R}^d$, $d \leq 3$, ein Gebiet, von dem wir vereinfachend annehmen, daß es polyedrisch berandet sei und damit einen Lipschitz-stetigen Rand besitzt. $\mathbf{T}_h$ bezeichne eine reguläre Zerlegung von Ω in d-dimensionale Elemente K, wobei h das Maximum der Elementdurchmesser $h_K = diam(K)$ aller Elemente $K \in \mathbf{T}_h$ ist. Als Elemente K verwenden wir im 2D-Fall konvexe Vierecke und im 3D-Fall Hexaeder.

Den Begriff "Hexaeder" benutzen wir hier für einen räumlichen Körper mit 8 Ecken und 6 Seitenflächen, welcher sich als Bild des Einheitswürfels $[-1,+1]^3$ bezüglich einer trilinearen Abbildung ergibt. Die Seitenflächen wären also im allgemeinen keine ebenen Vierecke mehr und die Hexaederelemente nicht mehr unbedingt konvex. Für die Interpolationstheorie ist dies auch nicht nötig. Es genügt, wenn die Elemente *sternförmig* bezüglich einer einbeschriebenen Kugel sind [7]. Für die Konvergenzanalysis ist es jedoch vorteilhaft, wenn der Normalenvektor auf den Seitenflächen der Elemente jeweils konstant ist. Daher nehmen wir im folgenden vereinfachend an, daß alle Hexaederelemente durch ebene Vierecke begrenzt sind.

DEFINITION 2.1 *Ein Gebiet G heißt* **sternförmig bezüglich einer Menge B**, *wenn für einen beliebigen Punkt $x \in G$ die abgeschlossene konvexe Hülle der Menge $\{x\} \cup B$ in G liegt.*

Offenbar sind konvexe Elemente K sternförmig bezüglich einer beliebigen einbeschriebenen Kugel $B \subset K$. Zu einem gegebenen Viereck- bzw. Hexaederelement K mit den Ecken a_m, $m = 1, \dots, 2^d$, sei $C_K := 2^{-d} \sum_m a_m$ der Schwerpunkt von K. Weiterhin bezeichnen wir im folgenden mit

$$B(x,\rho) := \{y \in \mathbf{R}^d : |y - x| < \rho\} \tag{2.1}$$

die Kugel mit dem Mittelpunkt x und dem Radius ρ. Dann fordern wir, daß

$$\forall\, K \in \mathbf{T}_h \;\; \exists\, \rho_K > 0 : \quad K \text{ ist sternförmig bezüglich } B(C_K, \rho_K)\,. \tag{2.2}$$

Diese Forderung bedeutet keine starke Einschränkung hinsichtlich der Gitterkonstruktion. Während sie im 2D-Fall auf Grund der Konvexität der Elemente automatisch erfüllt ist, ist sie bei Hexaederelementen erfüllt, wenn deren Form nicht zu sehr entartet.

An die Familie der Zerlegungen $\mathbf{T}_h$ stellen wir eine Regularitätsforderung, die beinhaltet, daß die Form der Elemente nicht degeneriert. Bei Dreieck- oder Tetraederelementen formuliert man diese Forderung üblicherweise mit Hilfe der Bedingung

$$\frac{h_K}{\rho_K} \le \gamma_1 \qquad \forall\, K \in \mathbf{T}_h\,, \tag{2.3}$$

wobei ρ_K der größte Radius aller Kugeln ist, die in K enthalten sind. Wir verwenden hier eine modifizierte Definition von ρ_K,

$$\rho_K := sup\{\rho : K \text{ ist sternförmig bezüglich } B(C_K, \rho)\}\,, \tag{2.4}$$

und verlangen, daß das Gitter *shape-regulär* in folgendem Sinne ist (siehe [12] für den 2D-Fall).

DEFINITION 2.2 *Für ein Element $K \in \mathbf{T}_h$ sei $h_K = diam(K)$ der Durchmesser von K, $\underline{h}_K$ die kürzeste Kantenlänge in K und ρ_K definiert durch (2.4). Sei weiterhin $\underline{\gamma}_K$*

der kleinste Winkel, der zwischen benachbarten Kanten von K und im 3D-Fall auch zwischen benachbarten Seitenflächen von K auftritt, und $\overline{\gamma}_K$ der entsprechend größte Winkel. Dann heißt eine Familie $\{\mathbf{T}_h\}$ von Zerlegungen **shape-regulär**, *wenn es h-unabhängige Konstanten $\gamma_0 > 0$, $\gamma_1 > 0$ und $\delta_0 > 0$ gibt mit den Eigenschaften (2.3) sowie*

$$\frac{h_K}{\underline{h}_K} \leq \gamma_0 \qquad \textit{und} \qquad \delta_0 \leq \underline{\gamma}_K \leq \overline{\gamma}_K \leq \pi - \delta_0 \qquad \forall\, K \in \mathbf{T}_h\,. \tag{2.5}$$

Durch diese Regularitätsvoraussetzung werden stark anisotrope Gitter ausgeschlossen, jedoch sind adaptive Verfeinerungen, bei denen die Form der Elemente nicht entartet, erlaubt. Im Fall von Viereck- bzw. Hexaederelementen ist die Bedingung (2.3) allein nicht ausreichend für die Regularität des Gitters. Ein Beispiel hierfür ist ein Viereckelement K mit den Ecken $a_1 = (0,0)$, $a_2 = (h,0)$, $a_3 = (\varepsilon, h)$ und $a_4 = (0,h)$. Hier ist zwar die Bedingung (2.3) für beliebiges $\varepsilon \in (0,h)$ erfüllt, die Bedingung (2.5) jedoch für hinreichend kleine ε verletzt. Die Form von K degeneriert und die Jacobi-Matrix der bilinearen Referenzabbildung auf das Einheitselement wird für $\varepsilon \to 0$ nahezu singulär.

Jedem Element $K \in \mathbf{T}_h$ ordnen wir eine geometrische Größe σ_K zu, die wir später in Abschätzungen benötigen werden und die ein Maß für die Winkelabweichung des Elementes K von einem Parallelogramm bzw. einem Parallelepiped darstellt.

DEFINITION 2.3 *Im 2D-Fall definieren wir zu einem Viereckelement K die* **Größe $\sigma_{\mathbf{K}}$** *als den größten Winkel zwischen zwei gegenüberliegenden Kanten des Elementes K. Gemäß dieser Definition können wir im 3D-Fall jeder viereckigen Randseite Γ eines Hexaederelementes K eine Größe $\sigma_\Gamma^{(2D)}$ zuordnen und erklären σ_K als*

$$\sigma_K \;:=\; \max_{\Gamma \subset \partial K} \sigma_\Gamma^{(2D)}\,.$$

Weiterhin definieren wir zu einer gegebenen Zerlegung $\mathbf{T}_h$ die Größe σ_h durch $\sigma_h = \max_{K \in \mathbf{T}_h} \sigma_K$.

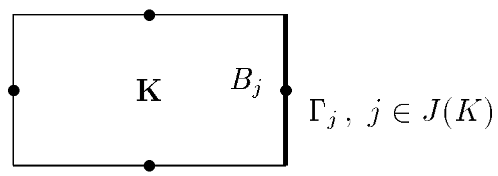


Abbildung 2.1: *Geschwindigkeitsknoten zum Element K*

Wir bezeichnen mit Γ_j die (d–1)–dimensionalen Seiten der Elemente $K \in \mathbf{T}_h$, wobei wir für $1 \leq j \leq N_I$ innere Seiten annehmen wollen und Randseiten $\Gamma_j \subset \partial\Omega$ für $N_I < j \leq N_I + N_B$. B_j sei der Schwerpunkt der Seite Γ_j, $J(K)$ die Indexmenge der Knoten B_j

zum Element K (siehe Abbildung 2.1), $J(\Omega) = \{j \,:\, 1 \leq j \leq N_I + N_B\}$ die Indexmenge aller Knoten sowie $J(\partial\Omega) = \{j \,:\, N_I < j \leq N_I + N_B\}$ die aller Randknoten.

Für die Definition der Polynomräume $P_k(G)$ und $Q_k(G)$ zu einem Gebiet $G \subset \mathbf{R}^d$ vereinbaren wir für einen Multiindex $\alpha = (\alpha_1, \ldots, \alpha_d)$ mit nichtnegativen ganzzahligen Komponenten α_i die Bezeichnungen $|\alpha| = \sum_{i=1}^d \alpha_i$, $|\alpha|_\infty = \max\{\alpha_i \,:\, 1 \leq i \leq d\}$ und $x^\alpha = \prod_{i=1}^d x_i^{\alpha_i}$ für $x \in \mathbf{R}^d$. Dann sei für nichtnegatives ganzzahliges k

$$P_k(G) := \{v \,:\, v(x) = \sum_{|\alpha| \leq k} c_\alpha x^\alpha\,,\ \forall\, x \in G\,,\ \ c_\alpha \in \mathbf{R}\} \tag{2.6}$$

und

$$Q_k(G) := \{v \,:\, v(x) = \sum_{|\alpha|_\infty \leq k} c_\alpha x^\alpha\,,\ \forall\, x \in G\,,\ \ c_\alpha \in \mathbf{R}\}\,. \tag{2.7}$$

Mit $C(G)$ bezeichnen wir den Raum der stetigen Funktionen über dem Gebiet G und mit $\|\cdot\|_{C(G)}$ die darauf erklärte übliche Maximumnorm. Zu einem Gebiet G sei $W^{m,p}(G)$ der Sobolev-Raum reellwertiger Funktionen, deren verallgemeinerte Ableitungen bis zur Ordnung m im Lebesgue-Raum $L^p(G)$ liegen, und $H^m(G)$ und $H_0^1(G)$ seien die üblichen Hilbert-Räume, die sich im Fall $p = 2$ ergeben. Mit $\|v\|_{m,p,G}$ bzw. $|v|_{m,p,G}$ bezeichnen wir die übliche Norm bzw. Halbnorm einer reellwertigen Funktion $v \in W^{m,p}(G)$ oder einer vektorwertigen Funktion $v \in (W^{m,p}(G))^d$. Die konkrete Bedeutung ergibt sich dabei stets aus dem Kontext. Sei $L_0^2(\Omega) := \{\, q \in L^2(\Omega) \,:\, \int_\Omega q\, dx = 0\}$ und $(\cdot,\cdot)_G$ das Skalarprodukt in $L^2(G)$ bzw. $(L^2(G))^d$, bei dem wir im Falle $G = \Omega$ den Index G weglassen. Zu einem Funktionenraum X bezeichne X^* den zugehörigen Dualraum. Für skalare Funktionen ϕ_j, $j \in J$, aus einem Funktionenraum X definieren wir die vektorwertige lineare Hülle $span^{(d)}\{\,\phi_j\,,\, j \in J\} \subset X^d$ als

$$span^{(d)}\{\,\phi_j\,,\, j \in J\} \;:=\; \{\, v \,:\, v = \sum_{j\in J} c_j\phi_j\,,\, c_j \in \mathbf{R}^d\,\}\,. \tag{2.8}$$

Die Bezeichnung $|\cdot|$ kann in Abhängigkeit vom jeweiligen Kontext unterschiedliche Bedeutung haben. Zu einem d-dimensionalen Gebiet $G \subset \mathbf{R}^d$ bezeichnet $|G|$ das d-dimensionale Inhaltsmaß von G, während für ein $(d-1)$-dimensionales Randstück $\Gamma \subset \partial G$ mit $|\Gamma|$ das $(d-1)$-dimensionale Inhaltsmaß von Γ gemeint ist. Außerdem verwenden wir die Bezeichnung $|v|$ für die euklidische Norm eines Vektors v.

Schließlich vereinbaren wir noch das Symbol $\#M$ für die Anzahl der Elemente einer endlichen Menge M.

2.2 Vier verschiedene nichtkonforme Elemente

Für die Approximation des Druckes verwenden wir stets den Raum der stückweise konstanten Funktionen

$$\mathrm{Q}_h := \{q_h \in \mathrm{L}_0^2(\Omega) \,:\, q_h(x) = q_K\,,\, \forall\, x \in K\,,\ \ \forall\, K \in \mathbf{T}_h\}\,. \tag{2.9}$$

Den Raum V_h zur Approximation der Geschwindigkeit definieren wir, indem wir die skalaren Basisfunktionen angeben, die ihn erzeugen. Zunächst stellen wir einige allgemeine Notationen bereit, mit deren Hilfe wir anschließend unsere vier verschiedenen nichtkonformen Elemente einheitlich beschreiben können.

Jeder Elementseite Γ_j, bzw. ihrem Schwerpunkt, den wir als Knoten B_j bezeichnen, wird eine Basisfunktion $\phi_j \in L^2(\Omega)$ zugeordnet. Mit Hilfe dieser Basisfunktionen definieren wir den Finite-Elemente-Raum V_h als

$$V_h := \left\{ \begin{array}{l} v_h \in (L^2(\Omega))^d \ : \ v_h(x) = \sum\limits_{j \in J(\Omega)} v_j \, \phi_j(x) \quad \text{mit} \quad v_j \in \mathbf{R}^d \\ \text{und} \quad v_j = 0 \quad \forall\, j \in J(\partial\Omega) \end{array} \right\} . \tag{2.10}$$

Im weiteren wollen wir die Konstruktion der ϕ_j erklären. Sie erfolgt separat auf jedem Element $K \in \mathbf{T}_h$. Für Knoten B_j, die nicht zu K gehören, verschwinden die entsprechenden Basisfunktionen, d.h.

$$\phi_j|_K \ = \ 0 \qquad \forall\, j \notin J(K) \, . \tag{2.11}$$

Wir beschreiben nun die Basisfunktionen ϕ_j zu den Knoten des Elementes K, d.h. für $j \in J(K)$. Zu ihrer Definition konstruieren wir Basisfunktionen auf einem *Referenzelement* $\widehat{K}$. Sei $F_K : \widehat{K} \to K$ eine geeignete bijektive *Referenzabbildung*, die über

$$\widehat{B}_{\widehat{j}} \ = \ F_K^{-1}(B_j)$$

jedem Knoten B_j von K einen *Referenzknoten* $\widehat{B}_{\widehat{j}}$ von $\widehat{K}$ mit einer *relativen Knotennummer* $\widehat{j} \in \widehat{J} := \{1, 2, \ldots, 2d\}$ zuordnet. Zum Zwecke einer einfacheren Notation nehmen wir im folgenden an, daß für unser festgehaltenes Element K gilt $\widehat{j} = j$ für $j \in J(K)$ (dies kann durch Umnumerieren aller Knoten B_j, $j \in J(\Omega)$ erreicht werden). Nachdem wir die Basisfunktionen $\widehat{\phi}_j$, $j \in \widehat{J}$, konstruiert haben, definieren wir ϕ_j als

$$\phi_j(x) \ := \ \widehat{\phi}_j(F_K^{-1}(x)) \qquad \forall\, x \in K \, . \tag{2.12}$$

Die Funktionen $\widehat{\phi}_j$ wählen wir nach dem Vorschlag von Rannacher/Turek [48] aus dem Polynomraum

$$\begin{aligned} \widehat{P} \ &= \ span\,\{1, \quad \widehat{x}_i, \, i = 1, \ldots, d, \quad \widehat{x}_i^2 - \widehat{x}_{i+1}^2 \, , i = 1, \ldots, d-1\} \\ &= \ span\,\{\, \widehat{p}_k(\widehat{x}) \, , \quad k = 1, \ldots, r \,\} \, , \end{aligned} \tag{2.13}$$

bei dem die Basispolynome zur Abkürzung mit $\widehat{p}_k$ bezeichnet seien. Für die Dimension von $\widehat{P}$ gilt $\dim(\widehat{P}) = r = 2d = \#\widehat{J} = \#J(K)$. Zur Festlegung von $\widehat{\phi}_j \in \widehat{P}$ verwenden wir r lineare stetige Knotenfunktionale $\widehat{N}_i \in \left(C(\overline{\widehat{K}})\right)^*$ und fordern

$$\widehat{N}_i(\widehat{\phi}_j) \ = \ \delta_{ij} \qquad \forall\, i \in \widehat{J} \, . \tag{2.14}$$

Dies bedeutet, daß die Ansatzkoeffizienten $c_k^{(j)}$, $k = 1, \ldots, r$, der Funktion $\widehat{\phi}_j$ mit

$$\widehat{\phi}_j(\widehat{x}) \ = \ \sum_{k=1}^{r} c_k^{(j)} \, \widehat{p}_k(\widehat{x}) \tag{2.15}$$

durch das lineare $r \times r$-Gleichungssystem

$$\sum_{k=1}^{r} \widehat{N}_i(\widehat{p}_k)\, c_k^{(j)} \;=\; \delta_{ij}\,, \qquad i = 1, \ldots, r\,, \tag{2.16}$$

festgelegt sind. Wir verlangen, daß die entsprechende Koeffizientenmatrix

$$\widehat{N} = (\widehat{N}_{ik}) \in \mathbf{R}^{r\times r} \qquad \text{mit} \qquad \widehat{N}_{ik} \;:=\; \widehat{N}_i(\widehat{p}_k) \tag{2.17}$$

regulär ist. Hieraus ergibt sich auch die $\widehat{P}$-*Unisolvenz*, daß ein beliebiges Polynom $\widehat{p} \in \widehat{P}$ durch die r Werte $\widehat{N}_i(\widehat{p})$, $i = 1, \ldots, r$, eindeutig bestimmt ist.

Die Knotenfunktionale $\widehat{N}_i$ werden aus vorgegebenen Knotenfunktionalen N_i^K zum Originalelement K abgeleitet. Der Grund dafür ist, daß wir primär an bestimmten Eigenschaften der Basisfunktionen ϕ_j auf dem Element K interessiert sind, welche wir mit Hilfe von linearen Knotenfunktionalen $N_i^K \in (C(\overline{K}))^*$ durch die Forderungen

$$N_i^K(\phi_j) \;=\; \delta_{ij} \qquad \forall\, i, j \in J(K) \tag{2.18}$$

formulieren. Um diese Forderungen zu erfüllen, definieren wir zu jedem vorgegebenen Knotenfunktional $N_i^K(\cdot)$ für eine beliebige Funktion $\widehat{\phi} \in C(\overline{\widehat{K}})$ den Wert des Knotenfunktionals $\widehat{N}_i$ als

$$\widehat{N}_i(\widehat{\phi}) \;:=\; N_i^K(\phi) \;=\; N_i^K(\widehat{\phi}\circ F_K^{-1})\,, \tag{2.19}$$

wobei ϕ die Funktion auf K bezeichnet, die der Funktion $\widehat{\phi}$ über die Vorschrift $\phi(x) = \widehat{\phi}(\widehat{x}) = \widehat{\phi}(F_K^{-1}(x))$ zugeordnet ist.

Mit der Festlegung der Knotenfunktionale $\widehat{N}_i$ sind durch (2.15) und (2.16) die Funktionen $\widehat{\phi}_j$ fixiert und damit über (2.12) auch die Basisfunktionen ϕ_j. Aus ihrer Konstruktion ergeben sich folgende Eigenschaften für eine beliebige Funktion $v_h \in V_h$ mit der Darstellung $v_h = \sum_{j\in J(\Omega)} v_j\phi_j$, wobei $v_j \in \mathbf{R}^d$. Auf einem Element $K{\in}\mathbf{T}_h$ kann v_h lokal charakterisiert werden durch

$$v_h|_K \;=\; \sum_{j\in J(K)} v_j\, \phi_j \tag{2.20}$$

bzw. durch

$$v_h|_K \in (P(K))^d \qquad \text{mit} \qquad P(K) \;:=\; \{\, p = \widehat{p}\circ F_K^{-1} \;:\; \widehat{p} \in \widehat{P}\,\}\,. \tag{2.21}$$

Im folgenden benötigen wir die Anwendung eines Knotenfunktionals auf eine Vektorfunktion. Diese sei wie folgt erklärt.

DEFINITION 2.4 *Sei $N \in X^*$ ein lineares stetiges Funktional, welches jeder Funktion ϕ aus einem gewissen Funktionenraum X skalarer Funktionen einen reellen Wert $N(\phi)$ zuordnet. Dann definieren wir für eine Vektorfunktion $v = (v_1, \ldots, v_d) \in X^d$ den Wert $N(v) \in \mathbf{R}^d$ als denjenigen d-dimensionalen Vektor, für dessen μ-te Komponente $(N(v))_\mu$ gilt*

$$(N(v))_\mu \;:=\; N(v_\mu)\,, \qquad \mu = 1, \ldots, d\,.$$

Aus (2.18) erhalten wir, daß zu einem festen Knotenindex $j \in J(\Omega)$ für alle Elemente K, die den Knoten B_j enthalten, die lokalen Knotenfunktionale $N_j^K(\cdot)$ denselben Wert liefern, d.h.

$$N_j^K(v_h) \;=\; v_j \;=:\; N_j(v_h) \qquad \forall\, K \in \mathbf{T}_h \quad \text{mit} \quad j \in J(K)\,, \quad \forall\, v_h \in V_h\,, \tag{2.22}$$

wodurch ein globales Knotenfunktional $N_j(\cdot)$ auf dem Raum V_h definiert wird.

Eine Funktion $v_h \in V_h$ ist also nicht völlig unstetig beim Übergang von einem Element K zu einem benachbarten Element K', sondern es bleibt wenigstens der Wert eines Knotenfunktionals erhalten, d.h.

$$N_j^K(v_h) \;=\; N_j^{K'}(v_h) \qquad \text{für} \quad j \in J(K) \cap J(K')\,.$$

Auf Grund der konkreten Gestalt der Knotenfunktionale (die wir unten erläutern werden) ergibt sich daraus, daß es zu jeder Komponente $v_{h,\mu}$ einer beliebigen Funktion $v_h = (v_{h,1}, \ldots, v_{h,d}) \in V_h$ auf jeder gemeinsamen Seite Γ_j zweier benachbarter Elemente K und K' mindestens einen Punkt $y_j^{(\mu)} \in \Gamma_j$ gibt, in dem die Funktion $v_{h,\mu}$ stetig ist. Diese Eigenschaft ist wichtig für den Nachweis einer diskreten Sobolev-Ungleichung im Raum V_h (siehe Abschnitt 4.2).

Wir kommen nun zur Definition der in dieser Arbeit verwendeten Knotenfunktionale.

DEFINITION 2.5 *Sei $K \in \mathbf{T}_h$ ein beliebiges Element, B_j mit $j \in J(K)$ ein Knoten auf der Elementseite Γ_j von K und ϕ eine hinreichend glatte Funktion, deren Einschränkung $\phi|_K$ stetig bis zum Rand ∂K des Elementes fortgesetzt sei. Dann heißt das lokale Knotenfunktional $N_j^K(\cdot)$ mit*

$$N_j^K(\phi) \;:=\; \phi|_K(B_j)$$

punktwert-orientiert *und das Funktional*

$$N_j^K(\phi) \;:=\; |\Gamma_j|^{-1} \int_{\Gamma_j} \phi|_K \, ds$$

mittelwert-orientiert. *Die dazugehörigen Finiten Elemente nennen wir punktwert-orientierte bzw. mittelwert-orientierte Elemente.*

Für eine Funktion v, die auf dem gesamten Gebiet Ω stetig ist, liefern die lokalen Knotenfunktionale $N_j^K(v)$ für alle Elemente K, die zum Knoten B_j gehören, denselben Wert, so daß auf Grund der Einbettung $H^2(\Omega) \subset C(\overline{\Omega})$ mit Hilfe von (2.22) ein globales Knotenfunktional $N_j(\cdot)$ auf dem Raum $(H^2(\Omega))^d + V_h$ definiert wird.

Um die Definition der Finiten Elemente abzuschließen, müssen wir nun noch die Referenzabbildung $F_K : \widehat{K} \to K$ konkretisieren. Wir betrachten die zwei Varianten $F_K \in (Q_1(\widehat{K}))^d$ und $F_K \in (P_1(\widehat{K}))^d$, bei denen wir die zugehörigen Finiten Elemente *Q_1-parametrisch* bzw. *P_1-parametrisch* nennen. Die vier verschiedenen Finiten Elemente ergeben sich daraus, daß wir jede Variante der Knotenfunktionale mit jeder Variante der Referenzabbildung kombinieren. Im folgenden werden wir zusätzlich zur Beschreibung der einzelnen Elemente jeweils gewisse Eigenschaften der Basisfunktionen nachweisen, die wir bei der Abschätzung des Interpolationsfehlers im nächsten Abschnitt benötigen.

$\mathbf{Q_1}$-parametrische Elemente

Das Referenzelement ist hier der d-dimensionale Einheitswürfel $\widehat{K} = [-1,+1]^d$. Seien $\hat{a}_m$, $m = 1, \ldots, 2^d$, die Ecken von $\widehat{K}$ und a_m die von K (siehe Abbildung 2.2). Dann ist

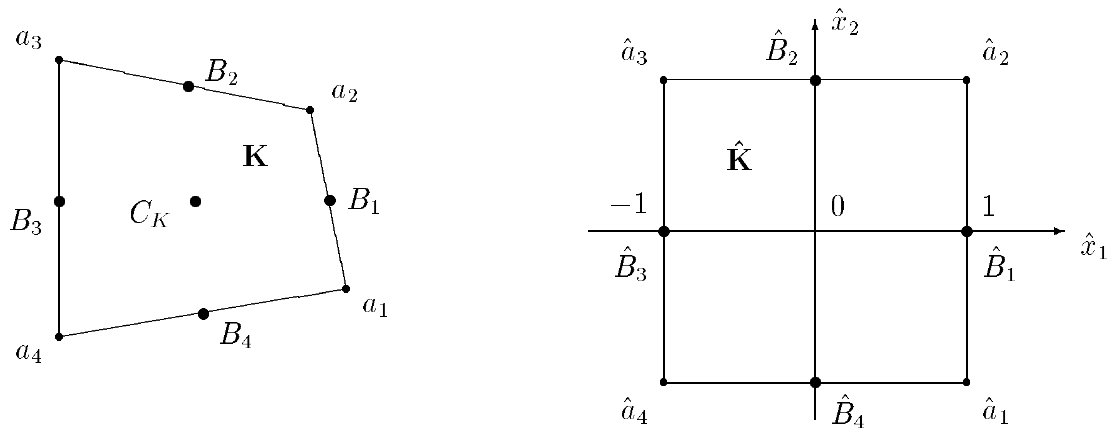


Abbildung 2.2: *Element K und zugehöriges Referenzelement $\hat{K}$*

die Referenzabbildung $F_K : \widehat{K} \to K$ eindeutig festgelegt durch die Bedingungen

$$F_K \in (Q_1(\widehat{K}))^d \qquad \text{und} \qquad a_m \;=\; F_K(\hat{a}_m)\;, \quad m = 1, \ldots, 2^d\;.$$

Die punktwert-orientierten $\mathbf{Q_1}$-parametrischen Elemente

Aus der Definition 2.5 und (2.19) erhalten wir hierfür folgende Knotenfunktionale auf dem Referenzelement

$$\widehat{N}_j(\hat{\phi}) \;=\; \hat{\phi}(\hat{B}_j)\;, \quad j \in \hat{J}\;. \tag{2.23}$$

Demzufolge ist die daraus abgeleitete Koeffizientenmatrix $\widehat{N}$ aus (2.17) zur Bestimmung der Ansatzkoeffizienten für die Basisfunktionen $\hat{\phi}_j$ unabhängig vom jeweiligen Element K. Diese Matrix ist regulär, woraus sich die $\widehat{P}$-Unisolvenz, die Existenz der Basisfunktionen $\hat{\phi}_j$ gemäß (2.14) sowie die Abschätzung

$$\|\hat{\phi}_j\|_{m,\infty,\widehat{K}} \;\leq\; C \qquad \forall\, j \in J(K) \quad \forall\, K {\in} \mathbf{T}_h \tag{2.24}$$

für beliebiges ganzzahliges $m \geq 0$ mit einer K-unabhängigen Konstanten ergeben.

Die mittelwert-orientierten $\mathbf{Q_1}$-parametrischen Elemente

Hier erhalten wir im 2D-Fall

$$\widehat{N}_j(\hat{\phi}) \;=\; |\widehat{\Gamma}_j|^{-1} \int_{\widehat{\Gamma}_j} \hat{\phi}\, d\hat{s}\;, \qquad j \in \hat{J}\;. \tag{2.25}$$

Analog zur Argumentation bei den punktwert-orientierten Elementen ergeben sich die $\widehat{P}$-Unisolvenz, die Existenz der Basisfunktionen $\widehat{\phi}_j$, ihre Unabhängigkeit vom jeweiligen Element K sowie die Abschätzung (2.24).

Im 3D-Fall gilt jedoch

$$\widehat{N}_j(\widehat{\phi}) \;=\; |\widehat{\Gamma}_j|^{-1} \int_{\widehat{\Gamma}_j} \widehat{\phi}\, \mu_j^K \, d\hat{s} \,, \qquad j \in \hat{J} \,, \tag{2.26}$$

mit einer Gewichtsfunktion μ_j^K, die sich aus der Transformation des Γ_j-Integrals von $N_j^K(\phi)$ auf ein Integral über $\widehat{\Gamma}_j$ ergibt. Dadurch hängt die über (2.17) zugeordnete Matrix $\widehat{N}$ vom jeweiligen Element K ab und wir haben sicherzustellen, daß sie regulär ist und ihre Inverse $\widehat{N}^{-1}$ für alle $K \in \mathbf{T}_h$ gleichmäßig beschränkt bleibt. Für die Gewichtsfunktion kann man die Abschätzung

$$|\,1 - \mu_j^K(\hat{s})\,| \;\le\; C\,\sigma_K \qquad \forall\, \hat{s} \in \widehat{\Gamma}_j \quad \forall\, K \in \mathbf{T}_h \tag{2.27}$$

nachweisen. Sei $\widehat{N}^{(0)}$ die Matrix, die gemäß (2.17) den K-unabhängigen Knotenfunktionalen $\widehat{N}_j(\cdot)$ aus (2.25) zugeordnet ist. Dann ergibt sich mit Hilfe von (2.27) die Abschätzung

$$\|\widehat{N} - \widehat{N}^{(0)}\| \;\le\; C\,\sigma_K \;.$$

Über ein Störungsargument bei Matrixinversen (siehe z.B. [23]) folgt, daß es ein hinreichend kleines $\sigma_0 > 0$ gibt, so daß für alle Gitter mit $\sigma_h = \max_{K \in \mathbf{T}_h} \sigma_K \le \sigma_0$ die Matrix $\widehat{N}$ stets regulär ist und ihre Inverse der Abschätzung

$$\|\widehat{N}^{-1}\| \;\le\; 2\,\|(\widehat{N}^{(0)})^{-1}\| \;\le\; C \qquad \forall\, K \in \mathbf{T}_h \tag{2.28}$$

genügt. Für die Ansatzkoeffizienten $c_k^{(j)}$ der Basisfunktionen $\widehat{\phi}_j(\widehat{x}) \;=\; \sum_{k=1}^r c_k^{(j)}\, \widehat{p}_k(\widehat{x})$ erhalten wir dann aus (2.16)

$$|c_k^{(j)}| \;\le\; \|\widehat{N}^{-1}\| \,,$$

woraus sich für beliebiges ganzzahliges $m \ge 0$

$$\|\widehat{\phi}_j\|_{m,\infty,\widehat{K}} \;\le\; \|\widehat{N}^{-1}\| \sum_{k=1}^r \|\widehat{p}_k\|_{m,\infty,\widehat{K}} \;\le\; C$$

ergibt. Damit haben wir unter der zusätzlichen Regularitätsforderung an das Gitter $\sigma_h \;\le\; \sigma_0$ (mit einem festen, hinreichend kleinen σ_0) sichergestellt, daß die Finite-Elemente-Basisfunktionen ϕ_j mit der Eigenschaft (2.18) existieren und die Normen der Referenzbasisfunktionen $\|\widehat{\phi}_j\|_{m,\infty,\widehat{K}}$ gleichmäßig für alle $K \in \mathbf{T}_h$ beschränkt sind.

BEMERKUNG 2.6 *Im 3D-Fall sind also bei den Q_1-parametrischen mittelwert-orientierten Elementen die Basisfunktionen $\widehat{\phi}_j$ für jedes Element K neu zu berechnen, was zu höheren Rechenzeitkosten im Vergleich zu den punktwert-orientierten Elementen führt. Diesen Nachteil könnten wir vermeiden, indem wir die Referenzbasisfunktionen $\widehat{\phi}_j$ mit Hilfe der festen K-unabhängigen Knotenfunktionale $\widehat{N}_j$ aus (2.25) bestimmen würden.*

Damit würden sich jedoch die Knotenfunktionale $N_j^K(\cdot)$ auf dem Originalelement K modifizieren zu

$$N_j^K(\phi) \;=\; |\widehat{\Gamma}_j|^{-1} \int_{\widehat{\Gamma}_j} \phi(F_K(\hat{s}))\, d\hat{s}$$

und es würde der eigentliche Vorteil der mittelwert-orientierten Elemente gegenüber den punktwert-orientierten verlorengehen, daß zu einer beliebigen Funktion $v_h \in V_h$ das Integral $\int_\Gamma [v_h]\, ds$ über die Sprünge von v_h auf jeder gemeinsamen Seite $\Gamma = \partial K \cap \partial K'$ zweier benachbarter Elemente K und K' verschwindet. Die Folge wäre, daß dann der Konsistenzfehler bei der Diskretisierung des Laplace-Operators (siehe Abschnitt 4.5) im Vergleich zu den punktwert-orientierten Elementen keine bessere Größenordnung mehr hätte. Im 3D-Fall sollte man also die einfacheren punktwert-orientierten Q_1-parametrischen Elemente den beiden denkbaren Varianten der mittelwert-orientierten Elemente vorziehen.

$\mathbf{P_1}$-parametrische Elemente

Rannacher und Turek [48] haben gezeigt, daß die Q_1-parametrischen Elemente (die bei ihnen parametrische Elemente heißen) nur dann optimale Interpolationseigenschaften besitzen, wenn das Gitter der zusätzlichen Regularitätsforderung $\sigma_h \le Ch$ genügt. Um diese Forderung zu vermeiden, haben sie sogenannte *nichtparametrische Elemente* vorgeschlagen. Dabei wird ein lokales, schiefwinkliges Koordinatensystem $\{\hat{x}_i\}_{i=1}^d$ definiert, indem die Schwerpunkte B_i und $B_{i'}$ gegenüberliegender Seiten von K verbunden werden. Ausgedrückt in diesen neuen Koordinaten $\hat{x}_i$ werden die Finite-Elemente-Basisfunktionen dann wieder aus dem Raum $\widehat{P}$ in (2.13) konstruiert. Wir modifizieren diesen Vorschlag etwas, indem wir ihn auf das Konzept der Referenztransformation übertragen, und kommen so zu unseren *P_1-parametrischen Elementen.*

Hierzu suchen wir eine affine Transformation $F_K \in (P_1(\widehat{K}))^d$, die dem Originalelement K ein Referenzelement $\widehat{K} = F_K^{-1}(K)$ in dem orthogonalen Koordinatensystem $\{\hat{x}_i\}_{i=1}^d$ zuordnet, wobei der Schwerpunkt von $\widehat{K}$ im Koordinatenursprung liegt und $diam(\widehat{K}) = O(1)$ gilt. Da eine affine Transformation bereits durch $(d+1)$ Bedingungen festgelegt ist, ist es im allgemeinen nicht möglich, F_K so zu konstruieren, daß alle Referenzecken $\hat{a}_m = F_K^{-1}(a_m)$, $m = 1, \ldots, 2^d$, für beliebiges $K \in \mathbf{T}_h$ gleich den festen Ecken $\hat{a}_{0m}$ des d-dimensionalen Einheitswürfels $\widehat{K}_0 = [-1,+1]^d$ sind, woraus $\widehat{K} = \widehat{K}_0$ folgen würde. Mit dem Originalelement K ändert sich also im allgemeinen auch das Referenzelement $\widehat{K}$. Es ist jedoch möglich, F_K so zu konstruieren, daß die Referenzknoten $\widehat{B}_j = F_K^{-1}(B_j)$ für beliebiges $K \in \mathbf{T}_h$ stets gleich den festen Schwerpunkten $\widehat{B}_{oj}$ der Seiten von $\widehat{K}_0 = [-1,+1]^d$ sind. Sei C_K der Schwerpunkt von K, $B_K = (b_1, \ldots, b_d)$ die $d \times d$-Matrix mit dem i-ten Spaltenvektor $b_i := B_i - C_K$ und seien die Knoten B_j von K mit $j \in J(K) = \widehat{J} = \{1, \ldots, 2d\}$ so numeriert, daß für $i = 1, \ldots, d$ jeweils die Knoten B_i und B_{i+d} auf gegenüberliegenden Seiten von K liegen (siehe Abbildung 2.3). Dann hat die Abbildung

$$x \;=\; F_K(\hat{x}) \;=\; B_K\,\hat{x} \,+\, C_K \tag{2.29}$$

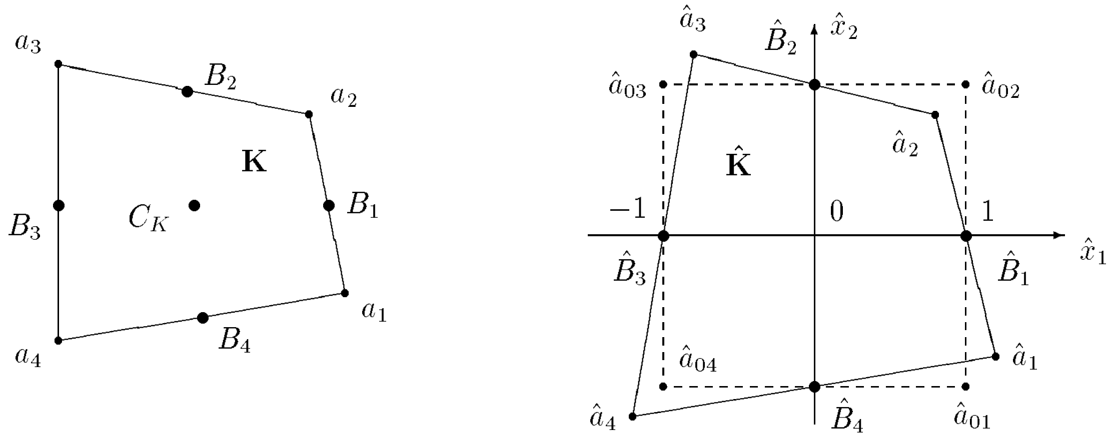


Abbildung 2.3: *P_1-parametrische Referenzabbildung* $F_K : \widehat{K} \to K$

die Eigenschaft, daß für $j \in J(K) = \widehat{J}$ gilt $\widehat{B}_j = F_K^{-1}(B_j) = \widehat{B}_{oj}$ mit $\widehat{B}_{oi} = \hat{e}_i$ und $\widehat{B}_{oi+d} = -\hat{e}_i$, $i = 1, \ldots, d$, wobei $\hat{e}_i$ den i-ten Einheitsvektor bezeichnet. Auf Grund der shape-Regularität des Gitters kann man für die Matrix B_K die Abschätzungen

$$\|B_K\| \leq C\, h_K \,, \qquad \|B_K^{-1}\| \leq C\, h_K^{-1} \tag{2.30}$$

sowie

$$\underline{C}\, h_K^d \leq det(B_K) \leq \overline{C}\, h_K^d \tag{2.31}$$

zeigen und es gilt für beliebiges $\hat{x} \in \widehat{K}$

$$|\hat{x}| = |B_K^{-1}(x - C_K)| \leq \|B_K^{-1}\|\, h_K \leq C \,.$$

Demzufolge sind alle Referenzelemente $\widehat{K}$ in einer hinreichend großen Kugel $\widehat{K}_R$ enthalten, d.h.

$$\exists\, R > 0 : \quad \widehat{K} \subset \widehat{K}_R := B(0, R) \qquad \forall\, K \in \mathbf{T}_h \,. \tag{2.32}$$

Mit Hilfe der P_1-parametrischen Referenzabbildung F_K aus (2.29) konstruieren wir nun unsere P_1-parametrischen Finite-Element-Basisfunktionen ϕ_j über (2.12) aus den Referenzbasisfunktionen $\widehat{\phi}_j$, welche wir wieder aus dem Polynomraum $\widehat{P}$ in (2.13) wählen.

Die punktwert-orientierten $\mathbf{P_1}$-parametrischen Elemente

Aus der Definition 2.5 und (2.19) erhalten wir dieselben Knotenfunktionale $\widehat{N}_j(\cdot)$ wie im Fall von Q_1-parametrischen punktwert-orientierten Elementen

$$\widehat{N}_j(\widehat{\phi}) = \widehat{\phi}(\widehat{B}_j) \,, \quad j \in \widehat{J} \,.$$

Da die Referenzknoten $\widehat{B}_j$ in beiden Fällen identisch sind, stimmen auch die Matrizen $\widehat{N}$ aus (2.17) und folglich auch die Ansatzkoeffizienten $c_k^{(j)}$ für die Basisfunktionen $\widehat{\phi}_j$ (siehe (2.15)) überein und sind unabhängig vom jeweiligen Element K. Damit sind die $\widehat{P}$-Unisolvenz und die Existenz der Basisfunktionen $\widehat{\phi}_j$ mit der Eigenschaft (2.14) für beliebiges $K \in \mathbf{T}_h$ gesichert. Für die Implementation dieser Elemente ergibt sich der Rechenzeitvorteil, daß die Ansatzkoeffizienten der Basisfunktionen $\widehat{\phi}_j$ nur einmal berechnet werden brauchen. Die gleichmäßige Abschätzung (2.24) der Normen $\|\widehat{\phi}_j\|_{m,\infty,\widehat{K}}$ für ganzzahliges $m \geq 0$ erhalten wir, indem wir die Basispolynome $\widehat{p}_k \in \widehat{P}$ mit Hilfe von (2.32) durch

$$\|\widehat{p}_k\|_{m,\infty,\widehat{K}} \;\leq\; \|\widehat{p}_k\|_{m,\infty,\widehat{K}_R} \;\leq\; C(R) \qquad \forall\, k = 1, \ldots, r$$

abschätzen.

Die mittelwert-orientierten $\mathbf{P}_1$-parametrischen Elemente

Hier liefern die Definition 2.5 und (2.19) sowohl im zweidimensionalen als auch im dreidimensionalen Fall die Referenzknotenfunktionale

$$\widehat{N}_j(\widehat{\phi}) \;=\; |\widehat{\Gamma}_j|^{-1} \int_{\widehat{\Gamma}_j} \widehat{\phi}\, d\hat{s}\;, \qquad j \in \widehat{J}\;.$$

Da sich mit den Elementen $K \in \mathbf{T}_h$ im allgemeinen auch das Referenzelement $\widehat{K}$ und damit auch dessen Seiten $\widehat{\Gamma}_j$ ändern, hängt die Koeffizientenmatrix $\widehat{N}$ zur Bestimmung der Ansatzkoeffizienten $c_k^{(j)}$ der Basisfunktionen $\widehat{\phi}_j$ (siehe (2.16)) vom jeweiligen Element K ab. Wir haben also zu klären, unter welchen Bedingungen die $\widehat{P}$-Unisolvenz (aus der sich auch die Existenz der Basisfunktionen $\widehat{\phi}_j$ mit der Eigenschaft (2.14) ergibt) sowie die gleichmäßige Abschätzung (2.24) für beliebiges $K \in \mathbf{T}_h$ gesichert sind. Dabei behandeln wir den zwei- und dreidimensionalen Fall getrennt.

<u>Der 2D-Fall</u>

Seien wieder $\widehat{a}_j$, $j = 1, \ldots, 4$, die Ecken von $\widehat{K}$, die so numeriert sein mögen, daß $\widehat{a}_j$ und $\widehat{a}_{j+1}$ † die Endpunkte der Kante $\widehat{\Gamma}_j$ mit dem Mittelpunkt $\widehat{B}_j$ sind (siehe Abbildung 2.3), und seien $\widehat{a}_{0j}$ die entsprechend numerierten Ecken von $\widehat{K}_0 = [-1,+1]^2$. Für den Vektor $\hat{\varepsilon}$ mit

$$\hat{\varepsilon} \;:=\; \widehat{a}_{01} \,-\, \widehat{a}_1 \tag{2.33}$$

gilt dann

$$\widehat{a}_j \;=\; \widehat{a}_{0j} \,+\, (-1)^j \hat{\varepsilon}\;, \qquad j = 1, \ldots, 4\;.$$

Damit wird also jedes Referenzelement $\widehat{K}$ eindeutig durch den Vektor $\hat{\varepsilon}$ charakterisiert und die Knotenfunktionale $\widehat{N}_j$, die zugehörige Matrix $\widehat{N}$ aus (2.17) und die Ansatzkoeffizienten $c_k^{(j)}$ können als Funktionen von $\hat{\varepsilon}$ aufgefaßt werden. Wenden wir zur Berechnung

† mit der Konvention $\widehat{a}_5 := \widehat{a}_1$

des $\widehat{\Gamma}_j$-Integrals in $\widehat{N}_j(\cdot)$ die Simpson-Formel an, die für $\widehat{\phi} = \widehat{p}_k \in \widehat{P} \subset P_2(\widehat{K})$ exakt ist, so erhalten wir

$$\widehat{N}_j(\widehat{p}_k) \;=\; \frac{2}{3}\,\widehat{p}_k(\widehat{B}_j) \;+\; \frac{1}{6}\sum_{m=j}^{j+1} \widehat{p}_k(\widehat{a}_{0m} + (-1)^m \hat{\varepsilon})\;.$$

Dies bedeutet, daß die Matrix $\widehat{N} = (\widehat{N}_j(\widehat{p_k}))_{jk}$ stetig von $\hat{\varepsilon}$ abhängt. Berechnet man die Determinante von $\widehat{N}$, so stellt sich heraus, daß

$$det(\widehat{N}) \;=\; det(\widehat{N}_0) \;>\; 0 \qquad \forall\, \hat{\varepsilon}\,,$$

wobei $\widehat{N}_0$ die zum Referenzelement $\widehat{K} = \widehat{K}_0$ gehörende Matrix ist. Demzufolge ist die Matrix $\widehat{N}$ stets regulär und die Inverse $\widehat{N}^{-1}$ hängt stetig von $\hat{\varepsilon}$ ab. Auf Grund von (2.32) liegt für jedes $K\in\mathbf{T}_h$ der zugehörige Vektor $\hat{\varepsilon}$ in der Menge $G = \{\hat{\varepsilon} :\ |\hat{\varepsilon}| \le R+\sqrt{2}\}$. Die Norm $\|\widehat{N}^{-1}\|$ ist als stetige Funktion von $\hat{\varepsilon}$ auf der kompakten Menge G beschränkt, woraus sich über (2.16) für die Ansatzkoeffizienten $c_k^{(j)}$ die gleichmäßige Abschätzung

$$|\, c_k^{(j)}\,| \;\le\; \|\widehat{N}^{-1}\| \;\le\; C \qquad \forall\, K \in \mathbf{T}_h$$

ergibt. Für die Basisfunktionen $\widehat{\phi}_j$ erhalten wir daraus mit Hilfe von (2.32)

$$\|\widehat{\phi}_j\|_{m,\infty,\widehat{K}} \;\le\; C\sum_{k=1}^{r}\|\widehat{p}_k\|_{m,\infty,\widehat{K}} \;\le\; C\sum_{k=1}^{r}\|\widehat{p}_k\|_{m,\infty,\widehat{K}_R} \;\le\; C\;.$$

Damit sind also die $\widehat{P}$-Unisolvenz sowie die gleichmäßige Abschätzung (2.24) für beliebiges $K\in\mathbf{T}_h$ gesichert.

Der 3D-Fall

Gegenüber dem 2D-Fall ist die Analyse der $\widehat{P}$-Unisolvenz und der gleichmäßigen Abschätzung (2.24) technisch wesentlich aufwendiger und weniger überschaubar. Deshalb begnügen wir uns hier mit einer eher qualitativen Aussage.

Zu jeder Ecke $\widehat{a}_m$, $m = 1, \dots, 8$, des Referenzelementes $\widehat{K}$ führen wir den Vektor $\hat{\varepsilon}_m \in \mathbf{R}^3$ mit

$$\hat{\varepsilon}_m \;:=\; \widehat{a}_m \;-\; \widehat{a}_{0m}$$

ein, der die Störung gegenüber der entsprechenden Ecke $\widehat{a}_{0m}$ des Standard-Referenzelementes $\widehat{K}_0 = [-1, +1]^3$ beschreibt. Dann wird jedes $\widehat{K}$ eindeutig durch den Vektor

$$\hat{\varepsilon} \;=\; (\hat{\varepsilon}_1, \dots, \hat{\varepsilon}_8) \;\in\; \mathbf{R}^{24}$$

charakterisiert und die Knotenfunktionale $\widehat{N}_j$, die zugehörige Matrix $\widehat{N}$ aus (2.17) und die Ansatzkoeffizienten $c_k^{(j)}$ können wieder als Funktionen von $\hat{\varepsilon}$ aufgefaßt werden. Sei $\widehat{N}_0$ die Matrix, die dem Vektor $\hat{\varepsilon}$=0, d.h. dem Referenzelement $\widehat{K} = \widehat{K}_0 = [-1, +1]^3$ entspricht. Diese Matrix ist regulär. Man kann zeigen, daß die Matrix $\widehat{N}$ als Funktion von $\hat{\varepsilon}$ an der Stelle $\hat{\varepsilon}$=0 stetig ist. Demzufolge gibt es ein hinreichend kleines $\varepsilon_0 > 0$, so daß

$$\|\widehat{N} - \widehat{N}_0\| \;\leq\; (2\,\|\widehat{N}_0^{-1}\|)^{-1} \qquad \forall\; |\hat{\varepsilon}| \leq \varepsilon_0 \,.$$

Über ein Störungsargument bei Matrixinversen (siehe z.B. [23]) erhalten wir daraus, daß die Matrix $\widehat{N}$ regulär ist und für ihre Inverse die Abschätzung

$$\|\widehat{N}^{-1}\| \;\leq\; 2\,\|\widehat{N}_0^{-1}\| \qquad \forall\; |\hat{\varepsilon}| \leq \varepsilon_0$$

gilt. Für welche Elemente K erfüllt nun der zu $\widehat{K} = F_K^{-1}(K)$ gehörende Störungsvektor $\hat{\varepsilon}$ die Bedingung $|\hat{\varepsilon}| \leq \varepsilon_0$? Wir verwenden folgendes Resultat.

LEMMA 2.7 *Es gibt eine K-unabhängige Konstante C, so daß gilt*

$$|\hat{\varepsilon}| \;\leq\; C\,\sigma_K \qquad \forall\, K \in \mathbf{T}_h \,, \tag{2.34}$$

wobei σ_K die in Definition 2.3 erklärte Winkelgröße des Elementes K ist.

Beweis.
Wir verwenden die Q_1-parametrische Referenzabbildung $F_Q : \widehat{K}_0 = [-1,+1]^3 \to K$, von der wir

$$|F_Q|_{2,\infty,\widehat{K}_0} \;\leq\; C\,h_K\sigma_K$$

zeigen können (siehe Lemma 2.10). Diese Abbildung hat außerdem die Eigenschaften $a_m = F_Q(\hat{a}_{0m})$ für $m = 1,\ldots,8$, $C_K = F_Q(0)$ und $B_i = F_Q(\hat{e}_i)$ für $i = 1,\ldots,3$, wobei $\hat{e}_i$ den i-ten Einheitsvektor bezeichnet. Eine beliebige Ecke $\hat{a}_{0m}$ von $\widehat{K}_0$ hat eine Darstellung in der Form $\hat{a}_{0m} = \sum_{i=1}^{3} \lambda_i\,\hat{e}_i$ mit geeigneten Zahlen $\lambda_i \in \{-1,+1\}$. Für $a_{0m} := B_K\hat{a}_{0m} + C_K$ erhalten wir dann

$$\begin{aligned} a_{0m} - C_K \;&=\; \sum_{i=1}^{3} \lambda_i\,B_K\hat{e}_i \;=\; \sum_{i=1}^{3} \lambda_i(B_i - C_K) \\ &=\; \sum_{i=1}^{3} \lambda_i\Big(F_Q(\hat{e}_i) - F_Q(0)\Big) \;=\; \sum_{i=1}^{3} \lambda_i\,DF_Q(\hat{\xi}_i)\,\hat{e}_i \,, \end{aligned}$$

wobei $DF_Q(\hat{\xi}_i)$ die Jacobi-Matrix von F_Q an einem Zwischenpunkt $\hat{\xi}_i \in \widehat{K}_0$ bezeichnet. Weiterhin gilt

$$a_m - C_K \;=\; F_Q(\hat{a}_{0m}) - F_Q(0) \;=\; DF_Q(\hat{\xi}_0)\,\hat{a}_{0m} \;=\; \sum_{i=1}^{3} \lambda_i\,DF_Q(\hat{\xi}_0)\,\hat{e}_i$$

mit einem geeigneten Zwischenpunkt $\hat{\xi}_0 \in \widehat{K}_0$. Daraus erhalten wir

$$|\,a_m - a_{0m}\,| \;=\; |\sum_{i=1}^{3} \lambda_i\Big(DF_Q(\hat{\xi}_0) - DF_Q(\hat{\xi}_i)\Big)\hat{e}_i\,| \;\leq\; C\,|F_Q|_{2,\infty,\widehat{K}_0} \;\leq\; C\,h_K\sigma_K \,.$$

Schließlich folgt mit (2.30)

$$|\hat{\varepsilon}_m| \;=\; |\,B_K^{-1}(a_m - a_{0m})\,| \;\leq\; \|B_K^{-1}\|\;|a_m - a_{0m}| \;\leq\; C\,\sigma_K$$

für beliebiges $m = 1, \ldots, 8$, woraus sich die Abschätzung (2.34) ergibt. □

Die Bedingung $|\hat{\varepsilon}| \leq \varepsilon_0$ kann also durch eine Forderung $\sigma_K \leq \sigma_0$ mit einem geeigneten σ_0 garantiert werden. Dies bedeutet, daß es ein hinreichend kleines $\sigma_0 > 0$ gibt, so daß für alle Gitter $\mathbf{T}_h$ mit $\sigma_h = \max_{K \in \mathbf{T}_h} \sigma_K \leq \sigma_0$ die Matrix $\widehat{N}$ stets regulär ist und ihre Inverse der Abschätzung

$$\|\widehat{N}^{-1}\| \;\leq\; 2\,\|\widehat{N}_0^{-1}\| \;\leq\; C \qquad \forall\;\; K \in \mathbf{T}_h$$

genügt. Analog zum 2D-Fall gewinnen wir daraus die gewünschte Abschätzung für die Basisfunktionen $\widehat{\phi}_j$.

Zur mathematisch strengen Absicherung der Unisolvenz sowie der gleichmäßigen Abschätzung (2.24) würden wir also im 3D-Fall die Gitterbedingung benötigen, daß die Winkelgröße σ_h hinreichend klein ist. Aus praktischer Sicht wäre dies eine recht einschneidende Bedingung, die den Vorteil dieses Elementes gegenüber den drei anderen Elementen (siehe Abschnitt 4.8) zum Teil wieder zunichte macht. Bisher ist es jedoch nicht gelungen, unter schwächeren Voraussetzungen die Unisolvenz und die Abschätzung (2.24) zu zeigen. Der entscheidende Punkt hierbei ist der Nachweis der Abschätzung

$$|det(\widehat{N})| \;\geq\; d_0 \;>\; 0 \qquad \forall\, |\hat{\varepsilon}| \leq \widehat{R}\,. \tag{2.35}$$

Dabei soll $\widehat{R}$ eine natürliche Schranke für die Norm des Störungsvektors $\hat{\varepsilon}$ bedeuten, die sich aus der shape-Regularität des Gitters ergibt. Durchgeführte Computersimulationen mit zufällig gewählten Vektoren $\hat{\varepsilon}$ geben Anlaß zu der Vermutung, daß die Abschätzung (2.35) für shape-reguläre Gitter erfüllt ist. Deshalb werden wir im folgenden bei den mittelwert-orientierten P_1-parametrischen Elementen im 3D-Fall die Unisolvenz und die Abschätzung (2.24) als erfüllt annehmen, ohne es jedesmal explizit zu erwähnen.

BEMERKUNG 2.8 *Die von Rannacher und Turek vorgeschlagenen "nichtparametrischen" Elemente unterscheiden sich von den hier eingeführten P_1-parametrischen Elementen. Mit unserem Zugang würden wir diese Elemente erhalten, wenn wir anstelle der Abbildung F_K aus (2.29) die folgende modifizierte Referenzabbildung $\widetilde{F}_K$ verwenden würden. Zu den Vektoren $b_i = B_i - C_K$, $i = 1, \ldots, d$, definieren wir modifizierte Vektoren $\widetilde{b}_i = \lambda_i b_i$ mit $\lambda_i = \|b_i\|/\|b_1\|$ sowie die $d{\times}d$-Matrix $\widetilde{B}_K = (\widetilde{b}_1, \ldots, \widetilde{b}_d)$, deren i-ter Spaltenvektor $\widetilde{b}_i$ ist. Dann erklären wir die affine Abbildung $\widetilde{F}_K$ als*

$$x \;=\; \widetilde{F}_K(\widehat{x}) \;=\; \widetilde{B}_K \widehat{x} \;+\; C_K$$

und die Basisfunktionen $\widehat{\phi}_j$ auf dem Referenzelement $\widehat{K} = \widetilde{F}_K^{-1}(K)$ in analoger Weise zu den P_1-parametrischen Elementen. In diesem Fall wären von den Referenzknoten $\widehat{B}_j$ nur noch $\widehat{B}_1 = \widehat{e}_1$ und $\widehat{B}_{1+d} = -\widehat{e}_1$ für alle Elemente $K \in \mathbf{T}_h$ fixiert. Hinsichtlich der $\widehat{P}$-Unisolvenz sowie der gleichmäßigen Abschätzung (2.24) erhalten wir qualitativ dieselben Aussagen. Ihr Nachweis ist jedoch technisch aufwendiger. Die "nichtparametrischen" Elemente haben den Vorteil (siehe [3]), daß sie auch im Fall anisotroper Gitter stabile Approximationen liefern, während man bei den P_1-parametrischen Elementen modifizierte quadratische Basisfunktionen verwenden müßte. Andererseits haben die P_1-parametrischen Elemente gewisse Vorteile bei der Implementierung.

2.3 Abschätzung des lokalen Interpolationsfehlers

Zu jedem der vier im vorhergehenden Abschnitt eingeführten Finite-Elemente-Räume $V_h \subset span^{(d)}\{\,\phi_j\,,\, j \in J(\Omega)\}$ definieren wir einen *globalen Interpolationsoperator* $I_h : (H^2(\Omega) \cap H_0^1(\Omega))^d \to V_h$ durch die Vorschrift

$$I_h v(x) \;=\; \sum_{j \in J(\Omega)} N_j(v)\, \phi_j(x) \tag{2.36}$$

mit

$$N_j(v) \;:=\; \begin{cases} v(B_j) & \text{für punktwert-orientierte Elemente,} \\ |\Gamma_j|^{-1} \int_{\Gamma_j} v\, ds & \text{für mittelwert-orientierte Elemente.} \end{cases} \tag{2.37}$$

Für $v \in (H_0^1(\Omega))^d$ verschwinden die Knotenwerte $N_j(v)$ zu Randknoten $j \in J(\partial\Omega)$ und es gilt $I_h v \in V_h$. Da nach Konstruktion alle Basisfunktionen ϕ_j mit $j \notin J(K)$ auf dem Element K verschwinden, hat die globale Interpolierte $I_h v$ die lokale Darstellung

$$I_h v(x) \;=\; \sum_{j \in J(K)} N_j(v)\, \phi_j(x) \qquad \forall\, x \in K\,. \tag{2.38}$$

Im weiteren wollen wir für eine beliebige Funktion $v \in (H^2(\Omega) \cap H_0^1(\Omega))^d$ Abschätzungen des Interpolationsfehlers $v - I_h v$ auf einem Element K herleiten. Derartige Abschätzungen wurden bisher nur für die Q_1-parametrischen Elemente von Rannacher und Turek [48] nachgewiesen, der Beweis erfaßt jedoch nicht den Fall von P_1-parametrischen Elementen. Wir werden einen Beweis angeben, der für alle vier Finite-Elemente-Räume aus Abschnitt 2.2 gleichermaßen zutrifft.

Der klassische Beweis von Interpolationsfehlerabschätzungen (Ciarlet [12]) würde aus folgenden drei Schritten bestehen.

(1) Der Interpolationsfehler $e = v - I_h v$ über einem Element K wird auf das Referenzelement $\widehat{K} = F_K^{-1}(K)$ transformiert und mit Hilfe von Normen der Referenzabbildung F_K und des Fehlers $\widehat{e} := e \circ F_K$ abgeschätzt. Es gilt $\widehat{e} = \widehat{v} - \widehat{I}\widehat{v}$, wobei $\widehat{I} : (H^2(\widehat{K}))^d \to (\widehat{P})^d$ ein Interpolationsoperator auf $\widehat{K}$ ist und $\widehat{v} = v \circ F_K$.

(2) Unter Verwendung des Bramble-Hilbert-Lemmas wird der Fehler auf $\widehat{K}$ abgeschätzt durch

$$|\,\widehat{v} - \widehat{I}\widehat{v}\,|_{m,2,\widehat{K}} \;\leq\; C\,|\widehat{v}|_{2,2,\widehat{K}}\,, \qquad m = 0, 1\,. \tag{2.39}$$

(3) Die Halbnorm $|\widehat{v}|_{2,2,\widehat{K}}$ wird wieder auf das Originalelement K transformiert und durch Terme der Referenzabbildung und die Norm $\|v\|_{2,2,K}$ abgeschätzt.

Beim Beweis unserer Interpolationsfehlerabschätzungen werden wir im wesentlichen auch in diesen drei Schritten vorgehen. Dabei lassen sich die Schritte (1) und (3) für alle vier Finite-Elemente-Räume aus Abschnitt 2.2 ohne größere Schwierigkeiten ausführen.

Ein Problem ergibt sich jedoch im Schritt (2). Hier wurde bisher stets angenommen, daß das Referenzelement $\widehat{K}$ und der Interpolationsoperator $\widehat{I}$ unabhängig vom jeweiligen Originalelement K sind. Dadurch ist dann auch die Konstante C in der Abschätzung (2.39) von K unabhängig. Für die P_1-parametrischen Elemente ändert sich jedoch mit dem Originalelement K im allgemeinen auch das Referenzelement $\widehat{K} = F_K^{-1}(K)$, so daß diese Konstante vom jeweiligen Element K abhängt. Das Problem besteht also darin, zu beweisen, daß es eine feste Konstante C gibt, so daß die Ungleichung (2.39) für alle $\widehat{K} = F_K^{-1}(K)$ mit $K \in \mathbf{T}_h$ gilt.

Im folgenden wollen wir zunächst für die Schritte (1) und (3) einige Hilfsmittel bereitstellen. Zur Referenzabbildung $F_K : \widehat{K} \to K$ definieren wir

$$J_{F_K}(\hat{x}) \;=\; det(\partial_j\, F_{K,i}(\hat{x})) \qquad \text{und} \qquad J_{F_K^{-1}}(x) \;=\; det(\partial_j\, F_{K,i}^{-1}(x)) \tag{2.40}$$

als die Determinanten der Jacobi-Matrizen von F_K und F_K^{-1}. Dann sind folgende Aussagen bekannt.

LEMMA 2.9 *Sei $F_K : \widehat{K} \to K$ eine hinreichend glatte, bijektive Abbildung, für die auch die inverse Abbildung F_K^{-1} hinreichend glatt ist. Dann liegt für eine beliebige Funktion $w \in W^{m,p}(K)$ mit $p \in [1,\infty]$ und ganzzahligem $m \geq 0$ die zugeordnete Funktion $\widehat{w} = w \circ F_K$ im Raum $W^{m,p}(\widehat{K})$ und es gelten mit jeweils K-unabhängigen Konstanten C die Abschätzungen*

$$|w|_{m,p,K} \;\leq\; C\; |J_{F_K}|^{1/p}_{0,\infty,\widehat{K}}\; |F_K^{-1}|^m_{1,\infty,K}\; |\widehat{w}|_{m,p,\widehat{K}}\,, \tag{2.41}$$

$$|\widehat{w}|_{m,p,\widehat{K}} \;\leq\; C\; |J_{F_K^{-1}}|^{1/p}_{0,\infty,K}\; |F_K|^m_{1,\infty,\widehat{K}}\; |w|_{m,p,K} \tag{2.42}$$

für $m \in \{0,1\}$ sowie

$$|\widehat{w}|_{2,p,\widehat{K}} \;\leq\; C\; |J_{F_K^{-1}}|^{1/p}_{0,\infty,K} \left(|F_K|^2_{1,\infty,\widehat{K}}\; |w|_{2,p,K} \;+\; |F_K|_{2,\infty,\widehat{K}}\; |w|_{1,p,K} \right). \tag{2.43}$$

Beweis.
Siehe Theorem 4.3.2 in [12]. □

Für ein shape-reguläres Gitter kann man mit Hilfe geometrischer Überlegungen sowohl für die Q_1-parametrische als auch für die P_1-parametrische Referenzabbildung F_K die Abschätzungen

$$|J_{F_K}|_{0,\infty,\widehat{K}} \;\leq\; C\, h_K^d\,, \qquad |J_{F_K}^{-1}|_{0,\infty,K} \;\leq\; C\, h_K^{-d}\,, \tag{2.44}$$

und

$$|F_K|_{1,\infty,\widehat{K}} \;\leq\; C\, h_K\,, \qquad |F_K^{-1}|_{1,\infty,K} \;\leq\; C\, h_K^{-1} \tag{2.45}$$

für ein beliebiges Element $K \in \mathbf{T}_h$ zeigen. Unterschiede zwischen beiden Abbildungen ergeben sich bei der Abschätzung der zweiten Ableitungen:

$$|F_K|_{2,\infty,\widehat{K}} \;\leq\; \begin{cases} 0 & \text{für } P_1\text{-parametrische Elemente,} \\ C\; h_K \sigma_K & \text{für } Q_1\text{-parametrische Elemente.} \end{cases} \tag{2.46}$$

Diese Aussage wollen wir für die Q_1-parametrischen Elemente beweisen.

LEMMA 2.10 *Sei $F_K : \widehat{K} = [-1,+1]^d \to K$ mit $F_K \in (Q_1(\widehat{K}))^d$ die Q_1-parametrische, bijektive Referenzabbildung eines Elementes $K \in \mathbf{T}_h$. Dann gilt die Abschätzung*

$$|F_K|_{2,\infty,\widehat{K}} \leq C\, h_K \sigma_K \qquad \forall\, K \in \mathbf{T}_h\,. \tag{2.47}$$

Beweis.
Wir behandeln den zwei- und dreidimensionalen Fall getrennt.

Der 2D-Fall

Seien a_m, $m = 1, \ldots, 4$, die Ecken eines beliebigen Elementes $K \in \mathbf{T}_h$ und $\widehat{a}_{0m}$ die entsprechenden Ecken des Referenzelementes $\widehat{K} = \widehat{K}_0 = [-1,+1]^2$ (siehe Abbildung 2.4). Dann hat die Q_1-parametrische Referenzabbildung F_K die Gestalt

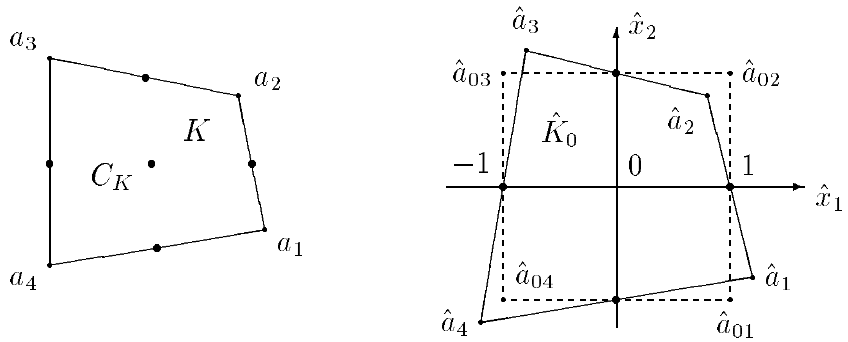


Abbildung 2.4: *Element K, Referenzelement $\widehat{K}_0 = [-1,+1]^2$ und P_1-parametrische Referenzecken $\widehat{a}_m$*

$$F_K(\widehat{x}_1, \widehat{x}_2) \;=\; a_4 \;+\; (a_1 - a_4)\widetilde{x}_1 \;+\; (a_3 - a_4)\widetilde{x}_2 \;+\; (-a_1 + a_2 - a_3 + a_4)\widetilde{x}_1\widetilde{x}_2$$

mit $\widetilde{x}_i = (1 + \widehat{x}_i)/2\,,\ \ i = 1, 2$, und wir erhalten

$$|F_K|_{2,\infty,\widehat{K}} \leq C\,|-a_1 + a_2 - a_3 + a_4|\,. \tag{2.48}$$

Um den Term $|-a_1 + a_2 - a_3 + a_4|$ abzuschätzen, verwenden wir die P_1-parametrische Referenzabbildung (2.29) aus Abschnitt 2.2, die wir hier mit F_P bezeichnen und die die Gestalt

$$F_P(\widehat{x}) \;=\; B_K\,\widehat{x} \;+\; C_K$$

hat. Seien $\widehat{a}_m := F_P^{-1}(a_m)\,,\ m = 1, \ldots, 4$, die entsprechenden Referenzecken bezüglich dieser affinen Abbildung und sei $\hat{\varepsilon}$ der Vektor $\hat{\varepsilon} := \widehat{a}_{01} - \widehat{a}_1$. Dann gilt $\widehat{a}_m = \widehat{a}_{0m} + (-1)^m \hat{\varepsilon}$ für $m = 1, \ldots, 4$ und es folgt mit (2.30)

$$|-a_1 + a_2 - a_3 + a_4| \;=\; |B_K(-\widehat{a}_1 + \widehat{a}_2 - \widehat{a}_3 + \widehat{a}_4)| \;=\; |4B_K\hat{\varepsilon}| \;\leq\; C\,h_K|\hat{\varepsilon}|\,. \tag{2.49}$$

Um die Abschätzung $|\hat{\varepsilon}| \leq C\sigma_K$ zu zeigen, betrachten wir das Paar t_1 und t_2 gegenüberliegender Kantenvektoren von K mit

$$t_1 \ := \ a_4 - a_3 \qquad \text{und} \qquad t_2 \ := \ a_1 - a_2 \ .$$

Für den Winkel α zwischen t_1 und t_2 gilt dann

$$\sigma_K \ \geq \ \alpha \ \geq \ |\sin\alpha\,| \ = \ \frac{|\,t_1 \times t_2\,|}{|t_1|\ |t_2|} \ .$$

Dabei definieren wir das Kreuzprodukt der 2D-Vektoren $t_i = (t_{i1}, t_{i2})$ als das übliche Kreuzprodukt der 3D-Vektoren $\bar{t}_i = (t_{i1}, t_{i2}, 0)$. Für die zugeordneten Referenzvektoren $\widehat{t}_1 := \widehat{a}_4 - \widehat{a}_3$ und $\widehat{t}_2 := \widehat{a}_1 - \widehat{a}_2$ gilt $t_i = B_K\widehat{t}_i$ und wir erhalten mit (2.31) und (2.30)

$$\frac{|\,t_1 \times t_2\,|}{|t_1|\ |t_2|} \ = \ \frac{|\det(B_K)|\ |\widehat{t}_1 \times \widehat{t}_2|}{|B_K\widehat{t}_1|\ |B_K\widehat{t}_2|} \ \geq \ \frac{Ch_K^2}{Ch_K^2}\ \frac{|\,\widehat{t}_1 \times \widehat{t}_2\,|}{|\widehat{t}_1|\ |\widehat{t}_2|} \ .$$

Aus (2.32) folgt $|\widehat{t}_i| \leq 2R$, so daß sich insgesamt mit einer K-unabhängigen Konstanten C die Abschätzung

$$\sigma_K \ \geq \ C\,|\,\widehat{t}_1 \times \widehat{t}_2\,|$$

ergibt. Unter Verwendung des Vektors $\hat{\varepsilon} = (\hat{\varepsilon}_1, \hat{\varepsilon}_2)$ erhalten wir $\widehat{t}_1 = -2\widehat{e}_2 + 2\hat{\varepsilon}$ und $\widehat{t}_2 = -2\widehat{e}_2 - 2\hat{\varepsilon}$ mit dem Einheitsvektor $\widehat{e}_2 = (0,1)$, woraus

$$|\,\widehat{t}_1 \times \widehat{t}_2\,| \ = \ 8\,|\hat{\varepsilon}_1| \ \leq \ C\,\sigma_K$$

folgt. Verwenden wir das andere Paar gegenüberliegender Kantenvektoren, so ergibt sich die analoge Abschätzung für die Komponente $\hat{\varepsilon}_2$. Demzufolge gilt

$$|\,\hat{\varepsilon}\,| \ \leq \ C\,\sigma_K$$

und die Behauptung (2.47) folgt aus (2.48) und (2.49).

<u>Der 3D-Fall</u>

Seien a_m, $m = 1, \ldots, 8$, die Ecken des Elementes $K \in \mathbf{T}_h$ (siehe Abbildung 2.5). Unter Verwendung der Abkürzung $\widetilde{x}_i = (1+\widehat{x}_i)/2,\ \ i = 1, \ldots, 3$, hat dann die Q_1-parametrische, bijektive Referenzabbildung $F_K : \widehat{K} = [-1, +1]^3 \to K$ die Darstellung

$$\begin{aligned} F_K(\widehat{x}_1, \widehat{x}_2, \widehat{x}_3) \ \ = \ \ & a_1 \ + \ (a_2 - a_1)\widetilde{x}_1 \ + \ (a_4 - a_1)\widetilde{x}_2 \ + \ (a_5 - a_1)\widetilde{x}_3 \\ & + (a_1 - a_2 + a_3 - a_4)\widetilde{x}_1\widetilde{x}_2 \ + \ (a_1 - a_2 + a_6 - a_5)\widetilde{x}_1\widetilde{x}_3 \\ & + (a_1 - a_4 + a_8 - a_5)\widetilde{x}_2\widetilde{x}_3 \\ & + (-a_1 + a_2 - a_3 + a_4 + a_5 - a_6 + a_7 - a_8)\widetilde{x}_1\widetilde{x}_2\widetilde{x}_3 \ . \end{aligned}$$

Daraus erhalten wir die Abschätzung

$$\begin{aligned} |F_K|_{2,\infty,\widehat{K}} \ \ \leq \ \ & C\Big(|\,a_1 - a_2 + a_3 - a_4\,| \ + \ |\,a_1 - a_2 + a_6 - a_5\,| \\ & \qquad + |\,a_1 - a_4 + a_8 - a_5\,| \ + \ |\,a_5 - a_6 + a_7 - a_8\,|\Big) \ . \end{aligned}$$

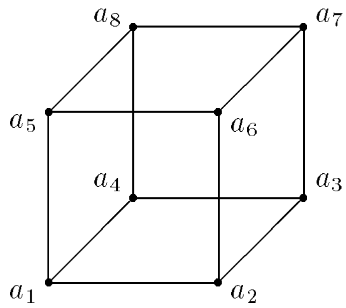


Abbildung 2.5: *Element K im 3D-Fall*

Sei Γ die Seitenfläche von K mit den Ecken $a_1, a_2, a_3, a_4,$ und $\sigma_\Gamma^{(2D)}$ der größte Winkel gegenüberliegender Kanten von Γ. Analog zum obigen Beweis für den 2D-Fall können wir zeigen

$$|\, a_1 - a_2 + a_3 - a_4 \,| \;\leq\; C\, h_K\, \sigma_\Gamma^{(2D)} \;\leq\; C\, h_K\, \sigma_K\ .$$

Die drei anderen Terme in der Abschätzung von $|F_K|_{2,\infty,\widehat{K}}$ können wir gleichermaßen abschätzen, indem wir eine entsprechende Seitenfläche von K betrachten. Damit ist die Behauptung (2.47) bewiesen. □

Für unsere beiden Referenzabbildungen aus Abschnitt 2.2 erhalten wir damit folgende Aussagen.

LEMMA 2.11 *Zu einem Element $K \in \mathbf{T}_h$ seien $F_K : \widehat{K} \to K$ die Q_1-parametrische oder P_1-parametrische Referenzabbildung aus Abschnitt 2.2 und h'_K die folgende modifizierte Gitterweite*

$$h'_K \;:=\; \begin{cases} h_K & \textit{für die } P_1\textit{-parametrische Referenzabbildung,} \\ h_K + \sigma_K & \textit{für die } Q_1\textit{-parametrische Referenzabbildung.} \end{cases}$$

Dann liegt für eine beliebige Funktion $v \in (W^{m,p}(K))^d$ mit $p \in [1,\infty]$ und ganzzahligem $m \geq 0$ die zugeordnete Funktion $\widehat{v} = v \circ F_K$ im Raum $(W^{m,p}(\widehat{K}))^d$ und es gelten für $m \in \{0,1\}$ die Abschätzungen

$$|v|_{m,p,K} \;\leq\; C\, h_K^{d/p-m}\, |\widehat{v}|_{m,p,\widehat{K}} \qquad \forall\, v \in (W^{m,p}(K))^d\,, \tag{2.50}$$

$$|\widehat{v}|_{m,p,\widehat{K}} \;\leq\; C\, h_K^{-d/p+m}\, |v|_{m,p,K} \qquad \forall\, v \in (W^{m,p}(K))^d \tag{2.51}$$

sowie

$$|\widehat{v}|_{2,p,\widehat{K}} \;\leq\; C\, h_K^{-d/p+1}\, h'_K\, \|v\|_{2,p,K} \qquad \forall\, v \in (W^{2,p}(K))^d \tag{2.52}$$

mit jeweils einer von K unabhängigen Konstanten C.

Beweis.
Auf jede Komponente $w = v_\mu$ der Funktion $v = (v_1, \dots, v_d)$ wenden wir Lemma 2.9 sowie die Abschätzungen (2.44), (2.45) und (2.46) an. $\square$

Wir kommen nun zur Abschätzung des Interpolationsfehlers $\widehat{v} - \widehat{I}\,\widehat{v}$ auf dem Referenzelement $\widehat{K}$. Sei $\widehat{I} : (H^2(\widehat{K}))^d \to (\widehat{P})^d$ der folgende Interpolationsoperator

$$\widehat{I}\,\widehat{v}\,(\widehat{x}) \;=\; \sum_{j\in\widehat{J}} \widehat{N}_j(\widehat{v})\,\widehat{\phi}_j(\widehat{x}) \qquad \forall\, \widehat{x} \in \widehat{K}\,, \tag{2.53}$$

wobei $\widehat{N}_j(\cdot)$ und $\widehat{\phi}_j$ zu jedem Finite-Elemente-Raum die entsprechenden Referenzknotenfunktionale und die dazugehörigen Basisfunktionen aus Abschnitt 2.2 sind.

Ein wesentliches Hilfsmittel, um zu einer K-gleichmäßigen Konstanten in der Abschätzung (2.39) zu kommen, ist das folgende Resultat über Polynomapproximation in Sobolev-Räumen [17],[7], bei dem die Konstanten nicht vom Gebiet abhängen.

LEMMA 2.12 *Sei $G \subset \mathbf{R}^d$ ein Gebiet mit dem Durchmesser h_G, welches sternförmig bezüglich einer Kugel $B \subset G$ vom Radius ρ_G ist, und sei γ eine Konstante mit*

$$\frac{h_G}{\rho_G} \;\le\; \gamma\,. \tag{2.54}$$

Dann gibt es zu jeder Funktion $w \in H^2(G)$ ein verallgemeinertes Taylor-Polynom $Qw \in P_1(G)$, welches den Abschätzungen

$$|w - Qw|_{m,2,G} \;\le\; C_1(d,\gamma)\,h_G^{2-m}\,|w|_{2,2,G}\,, \tag{2.55}$$

für $m \in \{0,1,2\}$ sowie

$$\|w - Qw\|_{0,\infty,G} \;\le\; C_2(d,\gamma)\,h_G^{2-d/2}\,|w|_{2,2,G} \tag{2.56}$$

für $d < 4$ genügt. Dabei sind die Konstanten $C_i(d,\gamma)$, i=1,2, zwar von d und γ, nicht aber vom Gebiet G abhängig.

Beweis.
Die Abschätzung (2.55) folgt unmittelbar aus Lemma 4.3.8 in [7] und (2.56) aus der Behauptung 4.3.2 in [7]. $\square$

Wir wollen diese Approximationsaussage auf die Gebiete $G = \widehat{K} = F_K^{-1}(K)$ anwenden. Um dabei auf K-gleichmäßige Abschätzungen zu kommen, benötigen wir, daß es K-unabhängige Konstanten $\widehat{h}_0$ und $\widehat{\gamma}_0$ gibt mit

$$h_{\widehat{K}} \;:=\; diam(\widehat{K}) \;\le\; \widehat{h}_0 \qquad \forall\, \widehat{K} = F_K^{-1}(K)\,,\; K \in \mathbf{T}_h\,, \tag{2.57}$$

und

$$\frac{h_{\widehat{K}}}{\rho_{\widehat{K}}} \leq \widehat{\gamma}_0 \qquad \forall\, \widehat{K} = F_K^{-1}(K)\,, \quad K \in \mathbf{T}_h\,. \tag{2.58}$$

Dabei bezeichnet $\rho_{\widehat{K}}$ analog zu (2.4) den größten Radius $\widehat{\rho}$ aller Kugeln $B(0,\widehat{\rho})$ mit dem Mittelpunkt $\widehat{x}=0$, so daß $\widehat{K}$ sternförmig bezüglich $B(0,\widehat{\rho})$ ist (man beachte, daß $C_{\widehat{K}}=0$ der Schwerpunkt des Referenzelementes $\widehat{K}$ ist).

Im Fall Q_1-parametrischer Elemente sind diese beiden Eigenschaften trivialerweise erfüllt, da es nur das eine Referenzelement $\widehat{K}=[-1,+1]^d$ gibt. Für die P_1-parametrischen Elemente wurde in Abschnitt 2.2 gezeigt, daß alle Referenzelemente $\widehat{K}$ in einer Kugel $\widehat{K}_R$ mit dem Radius R liegen (siehe (2.32)). Daraus ergibt sich die Eigenschaft (2.57) mit $\widehat{h}_0 = 2R$.

Wir zeigen nun (2.58) für die P_1-parametrischen Elemente. Sei $K\in\mathbf{T}_h$ ein beliebiges Element. Nach Voraussetzung (2.3) ist K sternförmig bezüglich einer Kugel $B(C_K,\rho_K)$ mit dem Schwerpunkt C_K von K als Mittelpunkt und einem Radius $\rho_K \geq h_K/\gamma_1$. Auf Grund der affinen Referenzabbildung F_K ist dann $\widehat{K}=F_K^{-1}(K)$ sternförmig bezüglich der Punktmenge $\widehat{B} := F_K^{-1}(B(C_K,\rho_K))$. Sei

$$\widehat{\rho}_{min} := 1/(\gamma_1 C_F)\,, \tag{2.59}$$

wobei C_F die Konstante in der Abschätzung $|F_K|_{1,\infty,\widehat{K}} \leq C h_K$ bezeichnet (siehe (2.45)). Für einen beliebigen Punkt $\widehat{y}$ der Kugel $B(0,\widehat{\rho}_{min})$ gilt dann

$$|F_K(\widehat{y}) - F_K(0)| \;\leq\; C_F h_K\, \widehat{\rho}_{min} \;\leq\; \rho_K\,,$$

d.h. $F_K(\widehat{y}) \in B(C_K,\rho_K)$ wegen $C_K = F_K(0)$. Demzufolge ist $B(0,\widehat{\rho}_{min}) \subset \widehat{B}$ und somit $\widehat{\rho}_{min} \leq \rho_{\widehat{K}}$. Daraus folgt die Abschätzung

$$\frac{h_{\widehat{K}}}{\rho_{\widehat{K}}} \leq \frac{2R}{\widehat{\rho}_{min}}$$

und damit die Eigenschaft (2.58) mit $\widehat{\gamma}_0 = 2R/\widehat{\rho}_{min}$.

Die Anwendung des Lemmas 2.12 liefert uns nun unter gewissen Voraussetzungen an den Interpolationsoperator $\widehat{I}$ die Abschätzung des Interpolationsfehlers auf dem Referenzelement mit einer vom Element und von h unabhängigen Konstanten.

LEMMA 2.13 *Sei $d \leq 3$ und seien die Eigenschaften (2.57) und (2.58) mit h-unabhängigen Konstanten $\widehat{h}_0$ und $\widehat{\gamma}_0$ erfüllt. Sei weiterhin zu jedem $K\in\mathbf{T}_h$ auf dem zugehörigen Referenzelement $\widehat{K}$ durch (2.53) ein Interpolationsoperator $\widehat{I} : (H^2(\widehat{K}))^d \to (\widehat{P})^d$ definiert, bei dem mit jeweils von h und K unabhängigen Konstanten C_N und C_ϕ die Knotenfunktionale $\widehat{N}_j(\cdot)$ der Abschätzung*

$$\max_{j\in\widehat{J}} |\,\widehat{N}_j(\widehat{v})\,| \;\leq\; C_N\, \|\widehat{v}\|_{0,\infty,\widehat{K}} \qquad \forall\, \widehat{v} \in (H^2(\widehat{K}))^d \quad \forall\, K \in \mathbf{T}_h \tag{2.60}$$

und die Basisfunktionen $\widehat{\phi}_j$ für $m \in \{0,1\}$ der Abschätzung

$$\sum_{j \in \widehat{J}} |\widehat{\phi}_j|_{m,2,\widehat{K}} \;\leq\; C_\phi \qquad \forall\, K \in \mathbf{T}_h \tag{2.61}$$

genügen und bei dem gilt

$$\widehat{I}\,\widehat{v} \;=\; \widehat{v} \qquad \forall\,\widehat{v} \in (P_1(\widehat{K}))^d\,. \tag{2.62}$$

Dann gibt es eine von h und K unabhängige Konstante C_I, so daß für $m \in \{0,1\}$ die Abschätzung

$$|\,\widehat{v} - \widehat{I}\,\widehat{v}\,|_{m,2,\widehat{K}} \;\leq\; C_I\,|\widehat{v}|_{2,2,\widehat{K}} \qquad \forall\,\widehat{v} \in (H^2(\widehat{K}))^d \quad \forall\, K \in \mathbf{T}_h \tag{2.63}$$

gilt.

Beweis.
Zu jeder Komponente $\widehat{v}_\mu$, $\mu = 1, \dots, d$, einer gegebenen vektorwertigen Funktion $\widehat{v} = (\widehat{v}_1, \dots, \widehat{v}_d) \in (H^2(\widehat{K}))^d$ sei $\widehat{q}_\mu = Q\,\widehat{v}_\mu \in P_1(\widehat{K})$ das zugeordnete verallgemeinerte Taylor-Polynom aus Lemma 2.12 und $\widehat{q} = (\widehat{q}_1, \dots, \widehat{q}_d)$ die entsprechende vektorwertige Funktion. Aus der Abschätzung (2.55) in Lemma 2.12 und der Voraussetzung (2.58) folgt dann für $m \in \{0,1\}$

$$|\,\widehat{v} - \widehat{q}\,|_{m,2,\widehat{K}} \;\leq\; C_1(d,\widehat{\gamma}_0)\,\widehat{h}_0^{2-m}\,|\widehat{v}|_{2,2,\widehat{K}}\,.$$

Mit Hilfe von (2.60), (2.61) erhalten wir für eine beliebige Funktion $\widehat{w} \in (H^2(\widehat{K}))^d$

$$|\,\widehat{I}\,\widehat{w}\,|_{m,2,\widehat{K}} \;\leq\; \sum_{j \in \widehat{J}} |\widehat{N}_j(\widehat{w})|\;|\widehat{\phi}_j|_{m,2,\widehat{K}} \;\leq\; C_N\,C_\phi\,\|\widehat{w}\|_{0,\infty,\widehat{K}}\,.$$

Daraus ergibt sich für $\widehat{w} = \widehat{v} - \widehat{q} \in (H^2(\widehat{K}))^d$ unter Verwendung der Abschätzung (2.56) aus Lemma 2.12

$$|\,\widehat{I}\,(\widehat{v} - \widehat{q})\,|_{m,2,\widehat{K}} \;\leq\; C_N\,C_\phi\,C_2(d,\widehat{\gamma}_0)\,\widehat{h}_0^{2-d/2}\,|\widehat{v}|_{2,2,\widehat{K}}\,.$$

Über die Dreiecksungleichung erhalten wir wegen $\widehat{I}\,\widehat{q} = \widehat{q}$

$$|\,\widehat{v} - \widehat{I}\,\widehat{v}\,|_{m,2,\widehat{K}} \;\leq\; |\,\widehat{v} - \widehat{q}\,|_{m,2,\widehat{K}} \;+\; |\,\widehat{I}\,(\widehat{v} - \widehat{q})\,|_{m,2,\widehat{K}} \;\leq\; C_I\,|\widehat{v}|_{2,2,\widehat{K}}$$

mit der K-unabhängigen Konstanten $C_I = C_1(d,\widehat{\gamma}_0)\,\widehat{h}_0^{2-m} + C_N\,C_\phi\,C_2(d,\widehat{\gamma}_0)\,\widehat{h}_0^{2-d/2}$. $\square$

Wir werden nun zeigen, daß die Voraussetzungen (2.60) – (2.62) für alle vier Finite-Elemente-Räume aus Abschnitt 2.2 erfüllt sind. Die Knotenfunktionale $\widehat{N}_j(\cdot)$ können wir für jeden Finite-Elemente-Raum auf Grund der Einbettung $H^2(\widehat{K}) \subset C(\,\overline{\widehat{K}}\,)$ abschätzen durch

$$|\,\widehat{N}_j(\widehat{\phi})\,| \;\leq\; C_N\,\|\widehat{\phi}\|_{0,\infty,\widehat{K}} \qquad \forall\,\widehat{\phi} \in H^2(\widehat{K})\,.$$

Dabei ist $C_N = 1$ außer im 3D-Fall bei Q_1-parametrischen mittelwert-orientierten Elementen. Dort gilt wegen (2.26) und (2.27)

$$C_N \;\le\; 1 + C \max_{K\in\mathbf{T}_h} \sigma_K \;\le\; 1 + C\,\pi\,.$$

Dies bedeutet, daß in jedem Fall die Voraussetzung (2.60) mit einer K-unabhängigen Konstanten C_N erfüllt ist. Im Abschnitt 2.2 haben wir für jeden Finite-Elemente-Raum die K-gleichmäßige Abschätzung (2.24) der Basisfunktionen $\widehat{\phi}_j$ gezeigt, wobei wir im 3D-Fall bei mittelwert-orientierten Elementen die zusätzliche Gitterbedingung $\sigma_h \le \sigma_0$ (siehe Definition 2.3) mit einem hinreichend kleinen σ_0 angenommen hatten. Daraus folgt mit (2.57)

$$\sum_{j\in\widehat{J}} |\widehat{\phi}_j|_{m,2,\widehat{K}} \;\le\; \sum_{j\in\widehat{J}} |\widehat{K}|^{1/2}\, |\widehat{\phi}_j|_{m,\infty,\widehat{K}} \;\le\; C\,\widehat{h}_0^{d/2}\,,$$

d.h. es gilt (2.61) mit einer K-unabhängigen Konstanten C_ϕ. Die Eigenschaft (2.62) ergibt sich für jeden Finite-Elemente-Raum aus der $\widehat{P}$-Unisolvenz und der Tatsache, daß $P_1(\widehat{K}) \subset \widehat{P}$.

Nach diesen Vorbereitungen sind wir in der Lage, eine lokale Fehlerabschätzung für den Interpolationsoperator I_h auf einem beliebigen Element $K\in\mathbf{T}_h$ zu beweisen.

LEMMA 2.14 *Sei $I_h : (H^2(\Omega)\cap H_0^1(\Omega))^d \to V_h$ der in (2.36) definierte globale Interpolationsoperator zu einem der vier Finite-Elemente-Räume V_h aus Abschnitt 2.2 und $\mathbf{T}_h$ eine shape-reguläre Zerlegung des Gebietes $\Omega \subset \mathbf{R}^d$ mit $d \le 3$, die bei mittelwert-orientierten Elementen im 3D-Fall der Bedingung $\sigma_h \le \sigma_0$ mit einem hinreichend kleinen σ_0 genügt. Mit der modifizierten Gitterweite*

$$h'_K \;:=\; \begin{cases} h_K & \text{für } P_1\text{-parametrische Elemente,} \\ h_K + \sigma_K & \text{für } Q_1\text{-parametrische Elemente,} \end{cases}$$

gilt dann für eine beliebige Funktion $v \in (H^2(\Omega)\cap H_0^1(\Omega))^d$ und ein beliebiges Element $K\in\mathbf{T}_h$ die Fehlerabschätzung

$$|\, v - I_h v\,|_{m,2,K} \;\le\; C\, h_K^{1-m}\, h'_K\, \|v\|_{2,2,K}\,, \qquad m \in \{0,1\}\,. \tag{2.64}$$

Beweis.
Sei $K\in\mathbf{T}_h$ ein beliebiges Element, $v \in (H^2(\Omega)\cap H_0^1(\Omega))^d$ eine gegebene Funktion und $\widehat{v} = v \circ F_K$ die ihr zugeordnete Funktion auf dem Referenzelement $\widehat{K} = F_K^{-1}(K)$. Auf Grund des Sobolev'schen Einbettungssatzes ist v stetig und es gilt wegen (2.37), der Definition 2.5 und (2.19)

$$N_j(v) \;=\; N_j^K(v) \;=\; \widehat{N}_j(\widehat{v}) \qquad \forall\, j \in J(K) = \widehat{J}\,.$$

Dabei haben wir zur Vereinfachung die globalen Knotennummern $j \in J(K)$ mit den relativen Knotennummern $j \in \widehat{J}$ auf dem Referenzelement $\widehat{K}$ identifiziert. Auf Grund von (2.38) und der Definition (2.53) des Interpolationsoperators $\widehat{I}$ gilt dann

$$\widehat{I_h v} \;=\; \sum_{j\in J(K)} N_j(v)\,\widehat{\phi}_j \;=\; \sum_{j\in\widehat{J}} \widehat{N}_j(\widehat{v})\,\widehat{\phi}_j \;=\; \widehat{I}\,\widehat{v}\,.$$

Daraus folgt mit Hilfe der Abschätzung (2.50) in Lemma 2.11 für $m \in \{0,1\}$

$$| v - I_h v |_{m,2,K} \leq C\, h_K^{d/2-m}\, | \widehat{v} - \widehat{I}\, \widehat{v} |_{m,2,\widehat{K}} .$$

Jeder Raum V_h erfüllt die Voraussetzungen von Lemma 2.13 und wir erhalten aus (2.63) und der Abschätzung (2.52) in Lemma 2.11

$$| v - I_h v |_{m,2,K} \leq C\, h_K^{d/2-m}\, C_I\, C\, h_K^{-d/2+1}\, h'_K\, \|v\|_{2,2,K} \leq C\, h_K^{1-m}\, h'_K\, \|v\|_{2,2,K} .$$

□

2.4 Inverse Ungleichungen

Für unsere Konvergenzanalysis benötigen wir lokale inverse Ungleichungen, um bei einer diskreten Funktion $v_h \in V_h$ eine Norm bzw. Halbnorm auf einem Element $K \in \mathbf{T}_h$ durch eine andere abschätzen zu können. Beim Beweis dieser Ungleichungen transformieren wir wieder vom Originalelement K auf das Referenzelement $\widehat{K}$, schätzen dort auf die gewünschte neue Norm bzw. Halbnorm ab und transformieren zurück. Dabei können wir für die Transformationen bereits die Abschätzungen aus Lemma 2.11 verwenden. Das Problem besteht darin, bei der Abschätzung der Normen auf $\widehat{K}$ zu zeigen, daß die entsprechenden Abschätzungskonstanten unabhängig vom jeweiligen Element $K \in \mathbf{T}_h$ sind. Ein Hilfsmittel hierfür ist das folgende Lemma.

LEMMA 2.15 *Seien $\widehat{G}_1 \subset \widehat{G}_2 \subset \mathbf{R}^d$ zwei feste endliche Gebiete, $\widehat{K}$ ein beliebiges Referenzelement mit*

$$\widehat{G}_1 \subset \widehat{K} \subset \widehat{G}_2$$

und $\widehat{P} = span\{ \widehat{p}_i\, ,\ i = 1, \dots, r\}$ der Polynomraum in (2.13). Dann gilt für $p, q \in [1, \infty]$ und beliebiges ganzzahliges $m \geq 0$ die Abschätzung

$$|\widehat{w}|_{m,p,\widehat{K}} \leq C\, \|\widehat{w}\|_{0,q,\widehat{K}} \qquad \forall\, \widehat{w} \in \widehat{P} \tag{2.65}$$

mit einer von $\widehat{K}$ unabhängigen Konstanten $C = C(p, q, m, \widehat{G}_1, \widehat{G}_2)$.

Beweis.
Einer beliebigen Funktion $\widehat{w} \in \widehat{P}$ können wir über die Basisdarstellung $\widehat{w} = \sum_{i=1}^r \widetilde{w}_i \widehat{p}_i$ eindeutig einen Vektor $\widetilde{w} = (\widetilde{w}_1, \dots, \widetilde{w}_r) \in \mathbf{R}^r$ zuordnen. Dann gilt

$$|\widehat{w}|_{m,p,\widehat{K}} \leq |\widehat{w}|_{m,p,\widehat{G}_2} \leq C(p, m, \widehat{G}_2)\, |\widetilde{w}|$$

mit $C(p, m, \widehat{G}_2) = \sum_{i=1}^r |\widehat{p}_i|_{m,p,\widehat{G}_2}$. Da die Basisfunktionen $\widehat{p}_i$ linear unabhängig sind, wird durch

$$\|\widetilde{w}\| := \|\sum_{i=1}^r \widetilde{w}_i\, \widehat{p}_i\|_{0,q,\widehat{G}_1} \qquad \forall\ \widetilde{w} = (\widetilde{w}_1, \dots, \widetilde{w}_r) \in \mathbf{R}^r$$

eine Norm auf $\mathbf{R}^r$ definiert. Auf dem endlich dimensionalen Raum $\mathbf{R}^r$ sind bekanntlich alle Normen äquivalent, also insbesondere auch die Norm $\|\cdot\|$ und die euklidische Norm $|\cdot|$. Demnach gibt es eine Konstante $C(q,\widehat{G}_1)$, so daß gilt

$$|\widetilde{w}| \leq C(q,\widehat{G}_1)\,\|\widetilde{w}\| \qquad \forall\, \widetilde{w} \in \mathbf{R}^r\,.$$

Da die Norm $\|\cdot\|$ von $\widehat{K}$ unabhängig ist, hängt auch diese Konstante nicht von $\widehat{K}$ ab. Insgesamt ergibt sich daraus

$$|\widehat{w}|_{m,p,\widehat{K}} \leq C(m,p,\widehat{G}_2)\,C(q,\widehat{G}_1)\,\|\widehat{w}\|_{0,q,\widehat{G}_1} \leq C\,\|\widehat{w}\|_{0,q,\widehat{K}}\,,$$

womit die Behauptung (2.65) bewiesen ist. □

Wir zeigen nun, daß es Gebiete $\widehat{G}_1$ und $\widehat{G}_2$ gibt, so daß die Bedingung

$$\widehat{G}_1 \subset \widehat{K} \subset \widehat{G}_2 \qquad \forall\, \widehat{K} = F_K^{-1}(K)\,,\quad K \in \mathbf{T}_h\,, \tag{2.66}$$

erfüllt ist. Im Fall von Q_1-parametrischen Elementen ist dies trivial. Betrachten wir nun die P_1-parametrischen Elemente. In Abschnitt 2.2 wurde nachgewiesen (siehe (2.32)), daß alle Referenzelemente $\widehat{K}$ in einer hinreichend großen Kugel $\widehat{K}_R$ liegen. In Abschnitt 2.3 haben wir gezeigt, daß jedes Referenzelement $\widehat{K}$ die Kugel $B(0,\widehat{\rho}_{min})$ enthält mit dem Mittelpunkt $\widehat{x}=0$ und dem festen, von h und K unabhängigen Radius $\widehat{\rho}_{min}$ aus (2.59). Für $\widehat{G}_1 := B(0,\widehat{\rho}_{min})$ und $\widehat{G}_2 := \widehat{K}_R$ ist also die Bedingung (2.66) erfüllt.

Mit Hilfe von Lemma 2.15 können wir die folgenden inversen Ungleichungen beweisen.

LEMMA 2.16 *Sei V_h einer der vier Finite-Elemente-Räume aus Abschnitt 2.2 und $K\in\mathbf{T}_h$ ein beliebiges Element. Dann gelten für $p,q \in [1,\infty]$ und $m \in \{0,1\}$ die Ungleichungen*

$$|v_h|_{m,p,K} \leq C\,h_K^{d(\frac{1}{p}-\frac{1}{q})}\,|v_h|_{m,q,K} \qquad \forall\, v_h \in V_h \tag{2.67}$$

und

$$|v_h|_{1,p,K} \leq C\,h_K^{d(\frac{1}{p}-\frac{1}{q})-1}\,\|v_h\|_{0,q,K} \qquad \forall\, v_h \in V_h\,, \tag{2.68}$$

wobei die Konstante C jeweils von K unabhängig ist.

Beweis.
Mit Hilfe der Referenzabbildung $F_K : \widehat{K} \to K$ ordnen wir der Funktion $v_h \in V_h$ auf dem Element K die Funktion $\widehat{v} := v_h \circ F_K$ auf dem Referenzelement $\widehat{K}$ zu. Nach Konstruktion von V_h ist dann $\widehat{v} \in (\widehat{P})^d$. Für jeden Finite-Elemente-Raum gibt es zwei feste endliche Gebiete $\widehat{G}_1$ und $\widehat{G}_2$, so daß die Bedingung (2.66) erfüllt ist. Wenden wir Lemma 2.15 mit $m=0$ auf jede Komponente $\widehat{w} := \widehat{v}_\mu \in \widehat{P}$ der vektorwertigen Funktion $\widehat{v} = (\widehat{v}_1,\ldots,\widehat{v}_d)$ an, so erhalten wir die Abschätzung

$$\|\widehat{v}\|_{0,p,\widehat{K}} \leq C\,\|\widehat{v}\|_{0,q,\widehat{K}} \qquad \forall\, \widehat{v} \in (\widehat{P})^d$$

mit einer K-unabhängigen Konstanten C. Sei $\widehat{w} := \widehat{\partial}_i \widehat{v}_\mu,\; i = 1, \ldots, d$, die i-te partielle Ableitung der μ-ten Komponente von $\widehat{v}$. Auf Grund der Struktur des Polynomraumes $\widehat{P}$ gilt $\widehat{w} \in \widehat{P}$. Mit Hilfe der Abschätzung (2.65) in Lemma 2.15 und aus der Definition der Halbnormen $|\widehat{v}|_{1,s,\widehat{K}}$ für $s = p$ und $s = q$ erhalten wir die Ungleichung

$$|\widehat{v}|_{1,p,\widehat{K}} \;\leq\; C\, |\widehat{v}|_{1,q,\widehat{K}} \qquad \forall\, \widehat{v} \in (\widehat{P})^d$$

mit einer Konstanten C, die unabhängig von K ist. Unter Verwendung von Lemma 2.11 können wir nun für $m \in \{0, 1\}$ die Funktion $v_h \in V_h$ abschätzen durch

$$|v_h|_{m,p,K} \;\leq\; C\, h_K^{d/p-m}\, |\widehat{v}|_{m,p,\widehat{K}} \;\leq\; C\, h_K^{d/p-m}\, |\widehat{v}|_{m,q,\widehat{K}} \;\leq\; C\, h_K^{d/p-m}\, h_K^{-d/q+m}\, |v_h|_{m,q,K}\;,$$

womit (2.67) bewiesen ist. Die Abschätzung (2.65) für $m = 1$ können wir auch auf den Fall einer vektorwertigen Funktion $\widehat{v} \in (\widehat{P})^d$ übertragen und die Behauptung (2.68) ergibt sich mit Hilfe von Lemma 2.11 aus

$$|v_h|_{1,p,K} \;\leq\; C\, h_K^{d/p-1}\, |\widehat{v}|_{1,p,\widehat{K}} \;\leq\; C\, h_K^{d/p-1}\, \|\widehat{v}\|_{0,q,\widehat{K}} \;\leq\; C\, h_K^{d/p-1}\, h_K^{-d/q}\, \|v_h\|_{0,q,K}\;.$$

$\square$

Kapitel 3

Finite Elemente Diskretisierung

Numerische Experimente zeigen, daß die Standard-Galerkin-Diskretisierung auf einem festen Gitter mit wachsender Reynolds-Zahl instabil wird. Deshalb verwenden wir zur Stabilisierung eine upwind-Technik für die Diskretisierung des Konvektionstermes. Ohmori/Ushijima [45] haben diese Technik für das nichtkonforme lineare Dreieckelement im Fall einer skalaren Diffusions-Konvektions-Gleichung vorgeschlagen. In [57], [63], [64], [58] wurde sie für das nichtkonforme P_1/P_0 Element von Crouzeix/Raviart auf die stationären inkompressiblen Navier-Stokes-Gleichungen übertragen. Dieselben Ideen konnten auch bei unseren nichtkonformen Viereck- und Hexaederelementen angewendet werden und haben sich bereits in vielen praktischen Rechnungen bewährt [67], [68], [61], [54].

Im Gegensatz zu den Crouzeix/Raviart-Elementen ist hier allerdings die upwind-diskretisierte schwache Form des Konvektionstermes nicht mehr positiv. Dies führt zu schwächeren Aussagen hinsichtlich der Existenz und Konvergenz diskreter Lösungen. Deshalb schlagen wir in Abschnitt 3.3 eine neue modifizierte upwind-Diskretisierung vor, welche einerseits die Positivität gewährleistet aber andererseits keine Verschlechterung in der Ordnung des Konsistenzfehlers zur Folge hat, so daß wir analoge theoretische Aussagen zu den Crouzeix/Raviart-Elementen [58] erhalten. Obwohl die modifizierte Diskretisierung bessere theoretische Eigenschaften hat, war in numerischen Experimenten bisher kein signifikanter Vorteil gegenüber der urprünglichen upwind-Version zu erkennen. Während wir bei der Konvergenzanalysis die modifizierte upwind-Diskretisierung betrachten, verwenden wir in numerischen Testrechnungen vorwiegend die ursprüngliche Variante.

3.1 Schwache Formulierung

Der Einfachheit halber beschränken wir uns im folgenden auf homogene Dirichlet'sche Randbedingungen. Für den Fall allgemeinerer Randbedingungen verweisen wir auf [34],

[22] und [16]. Seien V und Q die Räume $V{:=}(H_0^1(\Omega))^d$ und $Q{:=}L_0^2(\Omega)$. Dann lautet die schwache Formulierung des kontinuierlichen Navier-Stokes-Problems (1.1)

$$\begin{aligned}&\text{Finde } (u,p)\in V\times Q, \text{ so daß}\\ &\nu\, a(u,v)+n(u,u,v)+b(p,v) \;=\; (f,v) \qquad \forall\, v\in V\,,\\ &\qquad\qquad\qquad\qquad\quad b(q,u) \;=\; 0 \qquad\quad\;\; \forall\, q\in Q\,,\end{aligned} \tag{3.1}$$

wobei

$$a(u,v):=(\nabla u,\nabla v)\,,\quad b(p,v):=-(p,\nabla\cdot v)\,,\quad n(z,u,v):=(\,(z\cdot\nabla)u,v)\,. \tag{3.2}$$

Zur Diskretisierung des Problems (3.1) verwenden wir die Räume $V_h{\approx}V$ und $Q_h{\approx}Q$ aus Abschnitt 2.2. Da eine Funktion $v\in V_h$ im allgemeinen unstetig auf den Seiten der Elemente ist, benötigen wir elementweise definierte Bilinearformen. Sei

$$a_h(u,v):=\sum_{K\in\mathbf{T}_h}(\nabla u,\nabla v)_K \qquad \forall\, u,v\in V+V_h \tag{3.3}$$

und

$$\bar b_h(q,v)\;:=\;-\sum_{K\in\mathbf{T}_h}(q,\nabla\cdot v)_K \qquad \forall\, q\in Q,\;\forall\, v\in V+V_h\,. \tag{3.4}$$

Im Fall von punktwert-orientierten Elementen ist es vorteilhaft, eine numerisch integrierte Version von $\bar b_h(\cdot,\cdot)$ zu verwenden. Die Funktionen $v\in V_h$ sind dann in den Knoten B_j stetig, d.h. die Werte $v(B_j)$ wohl definiert. Wir integrieren jeden Summanden in (3.4) numerisch und setzen

$$\tilde b_h(q,v)\;:=\;-\sum_{K\in\mathbf{T}_h}|K|^{-1}\int_K q\,dx\sum_{j\in J(K)}|\Gamma_j|\,v(B_j)\cdot n_j \qquad \forall\, q\in Q,\;\forall\, v\in V_h\,, \tag{3.5}$$

wobei n_j den bezüglich K äußeren Normaleneinheitsvektor auf der Randseite Γ_j von K bezeichnet. Diese numerisch integrierte Version wurde bereits in [40] vorgeschlagen. Sie hat für die punktwert-orientierten Elemente gegenüber der Bilinearform $\bar b_h(\cdot,\cdot)$ den Vorteil, daß die Berechnung der entsprechenden Anteile in der Steifigkeitsmatrix stark vereinfacht wird und daß die diskrete Babuška-Brezzi-Bedingung auch ohne die Gitterforderung $\sigma_h\le\sigma_0$ mit einem hinreichend kleinen σ_0 gilt (siehe [48]). Als diskrete Bilinearform zu $b(\cdot,\cdot)$ wählen wir somit

$$b_h(q,v)\;=\;\begin{cases}\bar b_h(q,v)\,, & \text{wenn } V_h \text{ ein mittelwert-orientierter Raum ist,}\\ \tilde b_h(q,v)\,, & \text{wenn } V_h \text{ ein punktwert-orientierter Raum ist.}\end{cases} \tag{3.6}$$

Für die Diskretisierung der Trilinearform $n(\cdot,\cdot,\cdot)$ verwenden wir nicht die elementweise definierte Form

$$n_h(z,u,v)\;:=\;\sum_{K\in\mathbf{T}_h}(\,(z\cdot\nabla)u,v)_K\;, \tag{3.7}$$

sondern eine upwind-Modifikation $\bar n_h(\cdot,\cdot,\cdot)$, die wir im Abschnitt 3.3 erläutern werden.

Unser diskretes Navier-Stokes-Problem lautet somit:

$$\begin{aligned} &\text{Finde } (u_h, p_h) \in V_h \times Q_h, \text{ so daß} \\ &\nu\, a_h(u_h, v_h) + \bar{n}_h(u_h, u_h, v_h) + b_h(p_h, v_h) \;=\; (f, v_h) \qquad \forall\, v_h \in V_h\,, \\ &\qquad\qquad\qquad\qquad\qquad\qquad b_h(q_h, u_h) \;=\; 0 \qquad\qquad \forall\, q_h \in Q_h\,. \end{aligned} \tag{3.8}$$

Sei $W_h \subset V_h$ mit

$$W_h \;:=\; \{v_h \in V_h \;:\; b_h(q_h, v_h) = 0 \quad \forall\, q_h \in Q_h\} \tag{3.9}$$

der Unterraum der diskret-divergenzfreien Funktionen in V_h. Dann ist die Geschwindigkeitslösung u_h des Problems (3.8) gleichzeitig Lösung des Problems

$$\begin{aligned} &\text{Finde } u_h \in W_h, \text{ so daß} \\ &\nu\, a_h(u_h, v_h) + \bar{n}_h(u_h, u_h, v_h) \;=\; (f, v_h) \qquad \forall\, v_h \in W_h\,. \end{aligned} \tag{3.10}$$

3.2 Upwind-Diskretisierung des Konvektionstermes

In diesem Abschnitt erklären wir zunächst die ursprüngliche upwind-Variante $\tilde{n}_h$ für die Diskretisierung des Konvektionstermes, die bereits in vielen praktischen Rechnungen verwendet wurde [67], [68], [61], [54]. Sie basiert auf einer Zerlegung des Gebietes Ω in duale Gebiete. Dabei wird jedem Knoten B_l bzw. jeder Seite Γ_l eindeutig ein duales Gebiet R_l zugeordnet, welches folgendermaßen definiert ist.

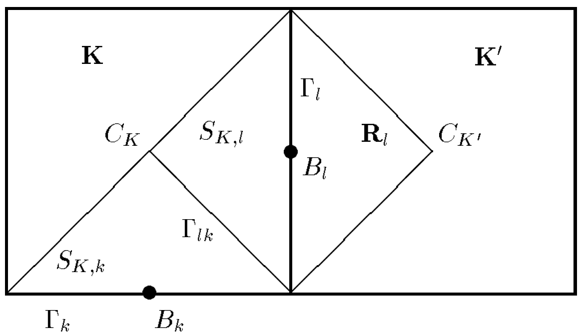


Abbildung 3.1: *Konstruktion des dualen Gebietes R_l*

Sei C_K der Schwerpunkt des Elementes K und $S_{K,l}$ die d-dimensionale Pyramide, welche Γ_l als Grundseite und C_K als Spitze hat (siehe Abbildung 3.1). Mit Λ_l bezeichnen wir die Menge aller Indizes $k \neq l$, für die sich Γ_k und Γ_l berühren und einem gemeinsamen Element K angehören. Für $k \in \Lambda_l$ sei $\Gamma_{lk} := \partial S_{K,l} \cap \partial S_{K,k}$ die gemeinsame (d-1)–dimensionale Seite von $S_{K,l}$ und $S_{K,k}$. Zu einer inneren Seite Γ_l gibt es zwei benachbarte Elemente K und K' mit $\Gamma_l = \partial K \cap \partial K'$ und wir definieren das duale Gebiet R_l als $R_l := S_{K,l} \cup S_{K',l}$. Im Fall einer Randseite $\Gamma_l \subset \partial K \cap \partial\Omega$ setzen wir $R_l := S_{K,l}$.

Wir benötigen noch einen sogenannten *lumping-Operator* L_h, der einer gegebenen Funktion v eine stückweise konstante Funktion $L_h v$ nach der Vorschrift

$$(L_h v)(x) := v^{(l)} \quad \forall\, x \in R_l \tag{3.11}$$

zuordnet. Dabei bezeichnen wir hier und im folgenden für eine Funktion v und $l \in J(\Omega)$ mit

$$v^{(l)} := N_l(v) \tag{3.12}$$

den Wert des Knotenfunktionals zum Knoten B_l.

Im weiteren wollen wir die upwind-Diskretisierung $\tilde{n}_h$ des Konvektionstermes mit

$$\tilde{n}_h(z,u,v) \;\approx\; n(z,u,w) \;=\; ((z\cdot\nabla)u,v)_\Omega$$

herleiten. Hierzu rechnen wir auf jedem dualen Gebiet R_l den Konvektionsterm mit Hilfe des lumping-Operators L_h und der Green'schen Formel um in Flußterme über die Randteile Γ_{lk} von R_l:

$$\begin{aligned}
n(z,u,v) &= \sum_{l\in J} (\,(z\cdot\nabla)u,v)_{R_l} \\
&= \sum_{l\in J} \{\,(\nabla\cdot(zu),v)_{R_l} - (\nabla\cdot z, u\cdot v)_{R_l}\} \\
&\approx \sum_{l\in J} \{\,(\nabla\cdot(zu),L_h v)_{R_l} - (\nabla\cdot z, L_h u\cdot L_h v)_{R_l}\} \\
&= \sum_{l\in J}\sum_{k\in\Lambda_l} \int_{\Gamma_{lk}} (z\cdot n_{lk})\,(u-u^{(l)})\cdot v^{(l)}\,d\gamma\,.
\end{aligned}$$

Dabei bezeichnet $J := J(\Omega)$ die Indexmenge aller Knoten, n_{lk} den äußeren Normaleneinheitsvektor auf dem Randteil Γ_{lk} von R_l und $\nabla\cdot(zu)$ einen d-dimensionalen Vektor, dessen i-te Komponente definiert ist durch

$$(\nabla\cdot(zu))_i \;:=\; \nabla\cdot(zu_i) \;=\; \sum_{j=1}^{d} \partial_j(z_j u_i)\,.$$

Die upwind-Idee besteht nun darin, daß man $u(x)$ auf Γ_{lk} durch einen festen Wert u^{upw} ersetzt, der einen benachbarten u-Wert in Richtung stromaufwärts (entgegen der Strömungsrichtung, englisch: "upstream" oder "upwind") in Bezug auf das Konvektionsfeld z repräsentiert. u^{upw} wird als Linearkombination der u-Werte in den Knoten B_l und B_k angesetzt

$$u^{\mathrm{upw}} := \lambda_{lk}(z)u^{(l)} \,+\, (1-\lambda_{lk}(z))\,u^{(k)}\,. \tag{3.13}$$

Die Funktion λ_{lk} ist so gewählt, daß $u^{\mathrm{upw}}\approx u^{(k)}$, falls die Konvektion z im Mittel über den Randteil Γ_{lk} in das duale Gebiet R_l hineinführt, und $u^{\mathrm{upw}}\approx u^{(l)}$, falls sie aus R_l herausführt. Allgemein wird $\lambda_{lk}(\cdot)$ angesetzt als

$$\lambda_{lk}(z) = \Phi(t) \quad \text{mit} \quad t := \frac{1}{\nu}\int_{\Gamma_{lk}} z\cdot n_{lk}\,d\gamma\,, \tag{3.14}$$

wobei die Gewichtsfunktion $\Phi(\cdot)$ folgenden Voraussetzungen genügen soll ([63]):

$$\begin{array}{ll} (A1) & \Phi(t) = 1 - \Phi(-t) \quad \forall\, t > 0 \quad \text{und} \quad 0 \le \Phi(t) \le 1 \quad \forall\, t \in \mathbf{R} \\ (A2) & \Phi(t) \ge \frac{1}{2} \quad \forall\, t \ge 0 \\ (A3) & g(t) := t\Phi(t) \quad \text{ist Lipschitz-stetig auf } \mathbf{R}. \end{array}$$

Mögliche Varianten für die Wahl von $\Phi(\cdot)$, die bereits in praktischen Rechnungen angewendet wurden ([52],[24],[67]), sind

$$\Phi_1(t) \;=\; \begin{cases} 1\,, & \text{falls} \quad t \ge 0 \\ 0\,, & \text{falls} \quad t < 0 \end{cases} \qquad \text{oder} \qquad \Phi_2(t) \;=\; \begin{cases} \frac{\frac{1}{2}+\theta t}{1+\theta t}\,, & \text{falls} \quad t \ge 0 \\ \frac{1}{2-2\theta t}\,, & \text{falls} \quad t < 0 \end{cases} \tag{3.15}$$

$\Phi_1(\cdot)$ wird als *einfaches* oder *scharfes upwind* bezeichnet und $\Phi_2(\cdot)$ als *Samarskij-upwind* ($\theta \in (0,1)$ ist ein zu wählender Parameter). Beide Funktionen erfüllen die obigen Voraussetzungen $(A1) - (A3)$. Praktische Rechnungen zeigen, daß das Samarskij-upwind in vielen Fällen genauere Lösungen liefert, daß aber andererseits die Steifigkeitsmatrix beim scharfen upwind bessere algebraische Eigenschaften (M-Matrix) hat, was zu einer besseren Konvergenzrate des Mehrgitterlösers führt.

Insgesamt erhalten wir folgende upwind-Diskretisierung des Konvektionstermes $n(\cdot,\cdot,\cdot)$

$$\tilde{n}_h(z,u,v) := \sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}} (z\cdot n_{lk})\,d\gamma\ (1-\lambda_{lk}(z))\,\left\{u^{(k)} - u^{(l)}\right\}\cdot v^{(l)}\,. \tag{3.16}$$

3.3 Modifikation zur Gewährleistung der Positivität

Wichtig für den Nachweis von Existenz- und Konvergenzaussagen der diskreten Lösungen ist die Positivität des diskretisierten Konvektionstermes, d.h.

$$\tilde{n}_h(w_h,v_h,v_h) \;\ge\; 0 \qquad \forall\, w_h \in W_h \quad \forall\, v_h \in V_h\,. \tag{3.17}$$

Diese Eigenschaft wurde in [57] für den Fall des Crouzeix/Raviart-Elementes bewiesen. Die entscheidende Idee dabei ist, daß man unter Verwendung der Trilinearform

$$\tilde{n}_h^{(2)}(z,u,v) \;:=\; -\sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}} (z\cdot n_{lk})\,d\gamma\ u^{(l)}\cdot v^{(l)} \tag{3.18}$$

sowie der Voraussetzungen (A1) und (A2) die Beziehung

$$\begin{aligned} &\tilde{n}_h(w_h,v_h,v_h) \;+\; \frac{1}{2}\tilde{n}_h^{(2)}(w_h,v_h,v_h) \;= \\ &\frac{1}{2}\sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}} (w_h\cdot n_{lk})\,d\gamma\ \left|v_h^{(l)} - v_h^{(k)}\right|^2 (\lambda_{lk}(w_h) - \frac{1}{2}) \;\ge\; 0 \end{aligned} \tag{3.19}$$

zeigen kann. Im Fall von Crouzeix/Raviart-Elementen gilt

$$\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}} (w_h\cdot n_{lk})\,d\gamma \;=\; 0 \qquad \forall\, w_h \in W_h \quad \forall\, l \in J\,, \tag{3.20}$$

da hier die Divergenz von $w_h \in W_h$ elementweise verschwindet. Daraus ergibt sich $\widetilde{n}_h^{(2)}(w_h, v_h, v_h) = 0$ und damit die Positivität (3.17) von $\widetilde{n}_h$. Für unsere Viereckelemente gilt zwar die Beziehung (3.19), aber nicht mehr (3.20), so daß wir auch nicht mehr $\widetilde{n}_h^{(2)}(w_h, v_h, v_h) = 0$ garantieren können. Um dennoch die Positivität für den diskretisierten Konvektionsterm zu gewährleisten, verwenden wir anstelle von $\widetilde{n}_h$ die modifizierte Form

$$\bar{n}_h(z,u,v) \;:=\; \widetilde{n}_h(z,u,v) \,+\, \frac{1}{2}\widetilde{n}_h^{(2)}(z,u,v)\,. \tag{3.21}$$

Dann gilt die folgende Aussage.

LEMMA 3.1 *Für die modifizierte upwind-Diskretisierung $\bar{n}_h$ in (3.21) mit $\widetilde{n}_h$ aus (3.16) und $\widetilde{n}_h^{(2)}$ aus (3.18) gilt unter den Voraussetzungen (A1) und (A2) die Abschätzung*

$$\bar{n}_h(w_h, v_h, v_h) \;\geq\; 0 \qquad \forall\, w_h \in W_h \quad \forall\, v_h \in V_h\,. \tag{3.22}$$

Beweis.
Die Behauptung ergibt sich aus (3.19), der Definition (3.14) von λ_{lk} und den Eigenschaften (A1) und (A2) der Gewichtsfunktion Φ. □

Es wird sich zeigen, daß diese Modifikation keine Verschlechterung in der Ordnung des Konsistenzfehlers von $\bar{n}_h$ bewirkt.

BEMERKUNG 3.2 *Für die ursprüngliche upwind-Form $\widetilde{n}_h$ in (3.16) kann man folgende abgeschwächte Positivitätseigenschaft zeigen*

$$\widetilde{n}_h(w_h, v_h, v_h) \;\geq\; -\,C\,h^{1/2}\,\|w_h\|_h\,\|v_h\|_h^2 \qquad \forall\, w_h \in W_h \quad \forall\, v_h \in V_h\,, \tag{3.23}$$

wobei $\|\cdot\|_h$ die in (4.1) definierte diskrete H^1-Norm ist. Der Nachweis ist jedoch technisch recht aufwendig. Wenn $w_h = i_h u \in W_h$ (i_h aus (4.5)) die divergenz-erhaltende Interpolierte einer beliebigen Funktion $u \in (H^2(\Omega) \cap H_0^1(\Omega))^d$ mit $\nabla\cdot u = 0$ ist, so läßt sich die verbesserte Positivitätseigenschaft

$$\widetilde{n}_h(i_h u, v_h, v_h) \;\geq\; -\,C\,h^{1/2}\widetilde{h}\,\|u\|_{2,2,\widetilde{h}}\,\|v_h\|_h^2 \qquad \forall\, v_h \in V_h\,, \tag{3.24}$$

zeigen. Dabei bezeichnen $\widetilde{h} = \max_{K\in\mathbf{T}_h} \widetilde{h}_K$ mit der modifizierten Schrittweite $\widetilde{h}_K$ in (4.30) und $\|\cdot\|_{2,2,\widetilde{h}}$ die entsprechende adaptiv gewichtete Sobolev-Norm (siehe Definition 4.2).

Kapitel 4

Konvergenzanalysis

Die hier verwendeten nichtkonformen Viereck- bzw. Hexaederelemente wurden bisher nur für das Stokes-Problem analysiert [48]. In diesem Kapitel stellen wir die Analysis für eine upwind-Diskretisierung der Navier-Stokes Gleichungen vor. Diese Diskretisierung wurde bereits in [57],[58] im Falle der nichtkonformen Crouzeix/Raviart-Elemente analysiert. Die Hauptschwierigkeit bei der Übertragung dieser Konvergenzanalysis auf die nichtkonformen Viereck- bzw. Hexaederelemente bestand darin, daß die schwache Form des diskretisierten Konvektionstermes hier nicht mehr positiv ist. Man kann nur noch eine abgeschwächte Form der Positivität zeigen. Dies hat zur Folge, daß die Existenz diskreter Lösungen nur für hinreichend kleine Gitterweiten $h \leq h_0(\nu)$ garantiert werden kann, wobei $h_0(\nu)$ mit wachsender Reynolds-Zahl $Re{=}\nu^{-1}$ sehr klein wird. Allerdings wird die Konvergenzordnung davon nicht negativ beeinflußt. Um das Problem mit der Positivität zu umgehen, verwenden wir die in Abschnitt 3.3 eingeführte modifizierte upwind-Diskretisierung. Hierfür können wir hinsichtlich Existenz und Konvergenz analoge Aussagen zu den Crouzeix/Raviart-Elementen [58] zeigen (siehe Abschnitt 4.7). Die entsprechenden Resultate für die ursprüngliche upwind-Diskretisierung geben wir jeweils nur als Bemerkung an. Ihr Beweis erfordert einen recht hohen technischen Aufwand und wird daher aus Platzgründen weggelassen.

Ein wichtiges Ziel bestand darin, die Konvergenzanalysis so anzulegen, daß sie in der Lage ist, die Verwendung adaptiv verfeinerter Gitter theoretisch abzusichern. Dies bedeutet insbesondere, daß das Gitter nicht mehr als quasi-uniform, sondern nur noch als shape-regulär vorausgesetzt werden darf. Eine Folge davon war, daß eine diskrete Sobolev-Ungleichung, die für die Abschätzung nichtlinearer Terme benötigt wird und die von Heywood und Rannacher [33] unter der Voraussetzung eines quasi-uniformen Gitters gezeigt wurde, noch einmal neu unter der schwächeren Voraussetzung eines shape-regulären Gitters bewiesen werden mußte. Hierzu war es nötig, eine neue Beweistechnik zu entwickeln (siehe Abschnitt 4.2), da sich die Ideen in [33] nicht übertragen ließen.

Ein anderer wichtiger Punkt bei der theoretischen Rechtfertigung adaptiver Gitteranpassung besteht darin, zu vermeiden, daß die Konstanten in den Abschätzungen des Diskretisierungsfehlers von globalen Sobolev-Normen der kontinuierlichen Lösung abhängen, da diese durch lokale Effekte sehr groß werden können. Deshalb führen wir in Abschnitt 4.1 adaptiv gewichtete diskrete Sobolev-Normen ein, bei denen ein großer Beitrag der lokalen Sobolev-Norm auf einem Element durch einen kleinen adaptiven Faktor kompensiert wird, sofern das Gitter an dieser Stelle entsprechend lokal verfeinert wurde. Die Verwendung dieser gitterabhängigen Sobolev-Normen liefert am Ende eine Abschätzung des Diskretisierungsfehlers, die bei geeignet angepaßtem Gitter auch dann noch eine sinnvolle a-priori-Fehlerschranke ergibt, wenn die globalen Sobolev-Normen der kontinuierlichen Lösung groß sind.

Um die Konvergenzanalysis überschaubarer zu gestalten, zergliedern wir ihre Hauptbestandteile in verschiedene Abschnitte. Nachdem wir in Abschnitt 4.2 eine diskrete Sobolev-Ungleichung für unsere Geschwindigkeitsräume V_h bereitgestellt haben, leiten wir in Abschnitt 4.3 lokale Approximationsaussagen für einen divergenz-erhaltenden Interpolationsoperator i_h her. Diesen Operator benötigen wir, um später in der Konvergenzanalysis zur divergenzfreien kontinuierlichen Geschwindigkeitslösung u eine diskret divergenzfreie Interpolierte $i_h u \in W_h \subset V_h$ mit gewissen Approximationseigenschaften definieren zu können. In Abschnitt 4.4 zeigen wir, daß alle vier definierten Raumpaare (V_h, Q_h) die h-gleichmäßige diskrete Babuška-Brezzi-Bedingung erfüllen. Die Konsistenzfehlerfunktionale, die durch die nichtkonforme Approximation der Terme "$-\triangle u$" und "∇p" entstehen, werden in Abschnitt 4.5 abgeschätzt. In Abschnitt 4.6 leiten wir zwei wichtige Aussagen über die upwind-Diskretisierung des Konvektionstermes her – die Stetigkeit der entsprechenden schwachen Form sowie eine Abschätzung des Konsistenzfehlers, der dadurch entsteht, daß wir anstelle der Galerkin-Diskretisierung unsere upwind-Diskretisierung verwenden. Nach diesen Vorbereitungen können wir in Abschnitt 4.7 Existenz- und Konvergenzaussagen für die diskreten Lösungen schlußfolgern.

In Abschnitt 4.8 stellen wir die Vor- und Nachteile der vier verschiedenen Räume V_h gegenüber. Die P_1-parametrischen mittelwert-orientierten Elemente sind die einzigen, bei denen der Diskretisierungsfehler unabhängig von den Winkelgrößen σ_K der Elemente K ist. Sie verursachen jedoch einen höheren Rechenaufwand. Für die anderen drei Räume V_h erhalten wir optimale Interpolations- und Konvergenzeigenschaften nur, wenn für das Gitter $\mathbf{T}_h$ eine gleichmäßige Abschätzung der Gestalt $\sigma_K \leq Ch_K$ für alle Elemente $K \in \mathbf{T}_h$ gilt. Wir beweisen, daß diese Abschätzung für blockstrukturierte Gitter, die durch uniforme Verfeinerung aus einer groben Makroelementzerlegung hervorgehen, erfüllt ist. Sie ist ebenfalls erfüllt, wenn wir adaptive Gitterverfeinerung mit Hilfe von hängenden Knoten durch lokale uniforme Verfeinerung von Makroelementen realisieren. In diesen Fällen können die vom Rechenaufwand her günstigeren Q_1-parametrischen Elemente verwendet werden.

4.1 Gitterabhängige Normen

Für unsere nichtkonformen Finite-Elemente-Räume V_h aus Abschnitt 2.2 sind die Funktionen $v_h \in V_h$ im allgemeinen unstetig auf den Randseiten der Elemente und liegen somit nicht in $V = (H_0^1(\Omega))^d$. Deshalb verwenden wir auf dem Raum $V + V_h$ eine elementweise definierte, diskrete H^1-Norm.

DEFINITION 4.1 *Für eine beliebige Funktion $v \in V + V_h$ heißt*

$$\|v\|_h \;:=\; \Big(\sum_{K\in\mathbf{T}_h} |v|_{1,2,K}^2 \Big)^{1/2} \tag{4.1}$$

die **diskrete $\mathbf{H^1}$-Norm** *von v.*

Man kann zeigen, daß $\|\cdot\|_h$ tatsächlich eine Norm auf $V + V_h$ ist.

Damit wir in der Lage sind, die Verwendung adaptiv-verfeinerter Gitter durch unsere Fehleranalysis rechtfertigen zu können, führen wir gitterabhängige, diskret gewichtete Sobolev-Normen ein.

DEFINITION 4.2 *Zu gegebener Zerlegung $\mathbf{T}_h$ eines Gebietes Ω heißt*

$$\alpha_K \;:=\; \frac{h_K}{h} \qquad \textit{mit} \qquad h \;=\; \max_{K\in\mathbf{T}_h} h_K \tag{4.2}$$

der **adaptive Faktor des Elementes** $K\in\mathbf{T}_h$ *und für eine Funktion $v \in W^{m,p}(\Omega)$ bzw. $v \in (W^{m,p}(\Omega))^d$ mit $1 \le p < \infty$ nennen wir*

$$\|v\|_{m,p,h} \;:=\; \Big(\sum_{K\in\mathbf{T}_h} \alpha_K^2\, \|v\|_{m,p,K}^p \Big)^{1/p} \tag{4.3}$$

die **adaptiv-gewichtete $\mathbf{W^{m,p}}$-Norm** *von v sowie*

$$|v|_{m,p,h} \;:=\; \Big(\sum_{K\in\mathbf{T}_h} \alpha_K^2\, |v|_{m,p,K}^p \Big)^{1/p} \tag{4.4}$$

die **adaptiv-gewichtete $\mathbf{W^{m,p}}$-Halbnorm** *von v. Für eine modifizierte Gitterweite h'_K † definieren wir $\|v\|_{m,p,h'}$ und $|v|_{m,p,h'}$ in analoger Weise, indem wir in (4.2) jeweils h_K durch h'_K und h durch h' ersetzen.*

Man kann zeigen, daß es sich bei $\|\cdot\|_{m,p,h}$ bzw. $|\cdot|_{m,p,h}$ tatsächlich um eine Norm bzw. Halbnorm handelt. Nach Definition ist $\alpha_K \le 1$ für $K\in\mathbf{T}_h$, woraus sich

$$\|u\|_{m,p,h} \;\le\; \|u\|_{m,p,\Omega} \qquad \text{und} \qquad |u|_{m,p,h} \;\le\; |u|_{m,p,\Omega}$$

für jede Zerlegung $\mathbf{T}_h$ des Gebietes Ω ergibt. Fehlerabschätzungen, die eine adaptiv-gewichtete Sobolev-Norm der exakten Lösung u verwenden, können in vielen Fällen wesentlich bessere Fehlerschranken liefern als solche, die die übliche Sobolev-Norm von u benutzen. Dies soll an dem folgenden Beispiel demonstriert werden.

† h'_K kann z.B. als Funktion von h_K und σ_K definiert sein

Für einen Fehlerterm $T(v_h)$ sei bereits die Abschätzung

$$|T(v_h)| \;\le\; \sum_{K\in\mathbf{T}_h} C\, h_K\, |u|_{2,2,K}\, |v_h|_{1,2,K}$$

nachgewiesen. Unter Verwendung der üblichen $W^{2,2}$-Halbnorm der exakten Lösung u würden wir $T(v_h)$ weiter abschätzen durch

$$|T(v_h)| \;\le\; C\, h \sum_{K\in\mathbf{T}_h} |u|_{2,2,K}\, |v_h|_{1,2,K} \;\le\; C\, h\, |u|_{2,2,\Omega}\, \|v_h\|_h\,, \tag{4.5}$$

während wir mit Hilfe der adaptiv-gewichteten $W^{2,2}$-Halbnorm

$$|T(v_h)| \;\le\; C\, h \sum_{K\in\mathbf{T}_h} \frac{h_K}{h}\, |u|_{2,2,K}\, |v_h|_{1,2,K} \;\le\; C\, h\, |u|_{2,2,h}\, \|v_h\|_h \tag{4.6}$$

erhalten würden. Gibt es nun Teilgebiete von Elementen K, in denen die exakte Lösung u nicht glatt ist in dem Sinne, daß $|u|_{2,2,K} \gg 1$, so gilt auch $|u|_{2,2,\Omega} \gg 1$ und (4.5) liefert eine schlechte obere Fehlerschranke. Ist hingegen das Gitter in diesen Teilgebieten geeignet adaptiv verfeinert, so daß jeder große Term $|u|_{2,2,K} \gg 1$ durch einen kleinen adaptiven Faktor $\alpha_K \ll 1$ kompensiert wird, so können wir erwarten, daß

$$|u|_{2,2,h} \;\ll\; |u|_{2,2,\Omega}$$

gilt und daß (4.6) eine sinnvolle, wesentlich bessere obere Fehlerschranke als (4.5) liefert.

Mit Hilfe adaptiv-gewichteter Sobolev-Normen kann also erklärt werden, warum man das Gitter nur in solchen Teilgebieten von Elementen K adaptiv verfeinern muß, wo die entsprechende lokale Sobolev-Norm $\|\cdot\|_{m,p,K}$ der exakten Lösung groß ist. Das Problem in der Praxis besteht allerdings darin, diese Teilbereiche herauszufinden, ohne die exakte Lösung zu kennen.

4.2 Eine diskrete Sobolev-Ungleichung

Für die Behandlung des nichtlinearen Konvektionstermes benötigen wir die Abschätzung der L^p-Norm einer beliebigen Funktion $w_h \in V_h$ durch die diskrete H^1-Norm $\|w_h\|_h$. Eine solche Abschätzung wurde bereits von Heywood/Rannacher [33] bewiesen, jedoch macht der Beweis von der Annahme eines quasiuniformen Gitters (d.h. $h/h_K \le C$ $\forall\, K\in\mathbf{T}_h$) Gebrauch. Mit Hilfe einer anderen Technik werden wir diese Abschätzung für den Fall shape-regulärer Gitter nachweisen, so daß auch adaptive Verfeinerungen zugelassen sind.

Die diskrete Sobolev-Ungleichung wird zunächst für skalare Funktionen bewiesen und dann auf den vektorwertigen Fall erweitert. Hierzu definieren wir für jeden Raum V_h aus Abschnitt 2.2 den Raum

$$S_h \;=\; span\,\{\, \phi_j\,,\;\; j \in J(\Omega) \setminus J(\partial\Omega) \,\} \tag{4.7}$$

als den zugehörigen Raum skalarer Finite-Elemente-Funktionen, für den $V_h = (S_h)^d$ gilt. Die Grundidee des Beweises besteht darin, zu vorgegebenem $z_h \in S_h$ eine Interpolierte $v_h \in \mathrm{H}_0^1(\Omega)$ zu konstuieren, auf die der übliche Sobolev'sche Einbettungssatz angewendet werden kann. Diese Interpolierte wird aus dem Raum $S_h^{konf} \subset \mathrm{H}_0^1(\Omega)$ der konformen isoparametrischen Q_1-Elemente gewählt. Im folgenden wollen wir ihre Konstruktion beschreiben.

Ein beliebiges $v_h \in S_h^{konf}$ hat die Darstellung

$$v_h(x) \;=\; \sum_{i\in E(\Omega)} v_i\,\psi_i(x)\,, \tag{4.8}$$

wobei $E(\Omega)$ die Indexmenge aller Ecken x_i der Elemente $K\in\mathbf{T}_h$ bezeichnet, ψ_i die Q_1-Basisfunktion zur Ecke x_i und $v_i = v_h(x_i)$. Wir definieren $v_h \in S_h^{konf}$, indem wir zu gegebenem $z_h \in S_h$ für jede Ecke x_i den Wert v_i festlegen.

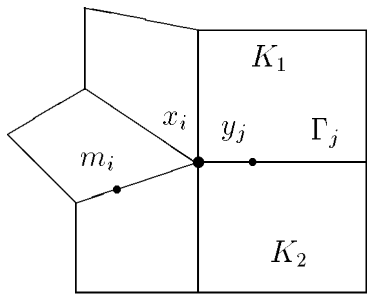


Abbildung 4.1: *Umgebung einer Ecke x_i*

An den Raum S_h stellen wir folgende Stetigkeitsanforderung. Für eine beliebige Funktion $z_h \in S_h$ möge es auf jeder gemeinsamen Seite Γ_j zweier benachbarter Elemente K_1 und K_2 mindestens einen *Stetigkeitspunkt* y_j der Funktion z_h geben (siehe Abbildung 4.1), d.h.

$$\begin{gathered}\forall\, z_h \in S_h \;\; \forall\, j \in J(\Omega)\setminus J(\partial\Omega) \quad \exists\, y_j = y_j(z_h) \in \Gamma_j = \partial K_1 \cap \partial K_2 \quad \text{mit} \\ z_{h|K_1}(y_j) \;=\; z_{h|K_2}(y_j)\,.\end{gathered} \tag{4.9}$$

Weiterhin fordern wir, daß es für jede Randseite $\Gamma_j \subset \partial\Omega$ mindestens einen Punkt y_j gibt, an dem die homogene Dirichlet'sche Randbedingung punktweise erfüllt wird, d.h.

$$\forall\, z_h \in S_h \;\; \forall\, j \in J(\partial\Omega) \quad \exists\, y_j = y_j(z_h) \in \Gamma_j \quad \text{mit} \quad z_h(y_j) \;=\; 0\,. \tag{4.10}$$

Für alle vier in Abschnitt 2.2 eingeführten Räume V_h nichtkonformer Viereckelemente sind die beiden Forderungen (4.9) und (4.10) an den jeweils zugeordneten Raum S_h erfüllt. Sei

$$M(i) \;=\; \{\; j \in J(\Omega) \;:\; x_i \in \overline{\Gamma}_j \;\} \tag{4.11}$$

die Indexmenge aller Seiten Γ_j, die die Ecke x_i enthalten, und sei

$$m_i \in \{\; y_j \;:\; j \in M(i)\;\} \quad \text{mit} \quad m_i \in \partial\Omega, \text{ falls } x_i \in \partial\Omega, \tag{4.12}$$

ein fest gewählter Punkt aus der Menge der Stetigkeitspunkte y_j. Dann setzen wir

$$v_i \ := \ z_h(m_i) \tag{4.13}$$

und die Definition von $v_h \in S_h^{konf}$ ist beendet. Im folgenden werden wir zwei Abschätzungen für die Interpolierte v_h beweisen. Hierzu bezeichnen wir mit

$$T(K) \ = \ \{\widetilde{K} \in \mathbf{T}_h \ : \ \partial\widetilde{K} \cap \partial K \neq \emptyset\} \tag{4.14}$$

die Menge aller zum Element K benachbarten Elemente (einschließlich K selbst).

LEMMA 4.3 *Sei $v_h \in S_h^{konf}$ die durch (4.8) und (4.13) definierte Interpolierte einer beliebigen Funktion $z_h \in S_h$. Weiterhin sei das Gitter als shape-regulär vorausgesetzt. Dann gelten die Abschätzungen*

$$\|z_h - v_h\|_{0,\infty,K} \ \leq \ C\, h_K \sum_{\widetilde{K}\in T(K)} |z_h|_{1,\infty,\widetilde{K}} \tag{4.15}$$

$$|v_h|_{1,2,K} \ \leq \ C \sum_{\widetilde{K}\in T(K)} |z_h|_{1,2,\widetilde{K}}\,. \tag{4.16}$$

Beweis.
Sei $K\in\mathbf{T}_h$ ein beliebiges Element, x_K dessen Schwerpunkt, $z_K := z_h(x_K)$ sowie x_i eine beliebige Ecke von K. Dann gibt es eine Kette $K = K_1 \to \ldots \to K_s$ von Elementen $K_\mu \in T(K)$ und eine zum Punkt m_i (siehe (4.12)) führende Kette von Stetigkeitspunkten $y_{j_0} = y_{j_1} \to \ldots \to y_{j_s} = m_i$ mit

$$y_{j_{\mu-1}}, y_{j_\mu} \ \in \ \overline{K}_\mu\,, \quad \mu = 1, \ldots, s\,.$$

Wegen (4.13) erhalten wir

$$|z_h(y_{j_0}) - v_i| \leq \sum_{\mu=1}^{s} |z_h(y_{j_{\mu-1}}) - z_h(y_{j_\mu})| \leq \sum_{\mu=1}^{s} h_{K_\mu}\, |z_h|_{1,\infty,K_\mu} \leq \sum_{\widetilde{K}\in T(K)} h_{\widetilde{K}} |z_h|_{1,\infty,\widetilde{K}}\,.$$

Für ein shape-reguläres Gitter existieren h-unabhängige Konstanten C_1, C_2 mit

$$C_1\, h_K \ \leq \ h_{\widetilde{K}} \ \leq \ C_2\, h_K \qquad \forall\, \widetilde{K} \in T(K)\,. \tag{4.17}$$

Damit folgt

$$|z_K - v_i| \ \leq \ |z_K - z_h(y_{j_0})| \ + \ |z_h(y_{j_0}) - v_i| \ \leq \ C\, h_K \sum_{\widetilde{K}\in T(K)} |z_h|_{1,\infty,\widetilde{K}}\,. \tag{4.18}$$

Sei $E(K) = \{\, i \ : \ x_i \in \overline{K}\}$ die Indexmenge aller Ecken x_i des Elementes K. Da die Basisfunktionen ψ_i mit $i \notin E(K)$ auf K verschwinden und eine konstante Funktion durch Q_1-Basisfunktionen exakt interpoliert wird, haben wir die Identität

$$v_h(x) \ - \ z_K \ = \ \sum_{i\in E(K)} (v_i - z_K)\, \psi_i(x) \qquad \forall\, x \in K\,.$$

Die Abschätzung (4.18) gilt für jedes $i \in E(K)$ und die Normen $\|\psi_i\|_{0,\infty,K}$ sind gleichmäßig beschränkt, woraus wir

$$\|v_h - z_K\|_{0,\infty,K} \;\le\; C\, h_K \sum_{\widetilde{K}\in T(K)} |z_h|_{1,\infty,\widetilde{K}} \tag{4.19}$$

erhalten. Mit Hilfe von

$$\|z_K - z_h\|_{0,\infty,K} \;\le\; h_K\, |z_h|_{1,\infty,K}$$

ergibt sich die Ungleichung (4.15).

Wir kommen nun zum Beweis der Ungleichung (4.16). Die konstante Funktion z_K liegt in dem von den Basisfunktionen ψ_i, $i \in E(K)$, aufgespannten Unterraum. Deshalb können wir eine inverse Ungleichung auf $z_K - v_h$ anwenden und erhalten mit (4.19)

$$|v_h|_{1,2,K} \;=\; |v_h - z_K|_{1,2,K} \;\le\; C\, h_K^{d/2-1} \|v_h - z_K\|_{0,\infty,K} \;\le\; C\, h_K^{d/2} \sum_{\widetilde{K}\in T(K)} |z_h|_{1,\infty,\widetilde{K}}\,.$$

Eine inverse Ungleichung auf $\widetilde{K}$ und die Abschätzung (4.17) liefern

$$|v_h|_{1,2,K} \;\le\; C\, h_K^{d/2} \sum_{\widetilde{K}\in T(K)} h_{\widetilde{K}}^{-d/2} |z_h|_{1,2,\widetilde{K}} \;\le\; C \sum_{\widetilde{K}\in T(K)} |z_h|_{1,2,\widetilde{K}}\,,$$

womit (4.16) bewiesen ist. □

Nach diesen Vorbereitungen kommen wir nun zum Beweis unserer diskreten Sobolev-Ungleichung.

LEMMA 4.4 *Sei $\mathbf{T}_h$ eine shape-reguläre Zerlegung des Gebietes $\Omega \subset \mathbf{R}^d$, $d \le 3$. Dann gelten für $1 \le p \le 6$ mit h-unabhängigen Konstanten $C = C(\Omega, p)$ die folgenden diskreten Sobolev-Ungleichungen*

$$\|z_h\|_{0,p,\Omega} \;\le\; C\, \|z_h\|_h \qquad \forall\, z_h \in S_h \tag{4.20}$$

und

$$\|w_h\|_{0,p,\Omega} \;\le\; C\, \|w_h\|_h \qquad \forall\, w_h \in V_h\,. \tag{4.21}$$

Beweis.
Zunächst weisen wir die Abschätzung (4.20) für beliebiges $z_h \in S_h$ nach. Es genügt, diese Ungleichung für den Fall p=6 zu beweisen. Für $1 \le p < 6$ folgt sie dann aus der stetigen Einbettung der L^p-Räume, d.h. aus

$$\|z_h\|_{0,p,\Omega} \;\le\; C(p,\Omega)\, \|z_h\|_{0,6,\Omega}\,.$$

Sei $v_h \in S_h^{konf}$ die durch (4.8), (4.13) definierte Interpolierte zu $z_h \in S_h$. Dann ist nach Konstruktion $v_h \in \mathrm{H}_0^1(\Omega)$ und der Sobolev'sche Einbettungssatz liefert für $d \le 3$

$$\|v_h\|_{0,6,\Omega} \;\le\; C\, \|v_h\|_{1,2,\Omega} \;\le\; C\, |v_h|_{1,2,\Omega}\,. \tag{4.22}$$

Für shape-reguläre Gitter ist die Anzahl $\#T(K)$ der Elemente von $T(K)$ gleichmäßig durch eine h-unabhängige Konstante C_T beschränkt, d.h.

$$\#T(K) \le C_T \qquad \forall\, K \in \mathbf{T}_h\,. \tag{4.23}$$

Unter Verwendung von (4.16) erhalten wir damit

$$|v_h|^2_{1,2,\Omega} \le \sum_{K\in\mathbf{T}_h} C C_T \sum_{\widetilde{K}\in T(K)} |z_h|^2_{1,2,\widetilde{K}} \le C C_T^2 \sum_{K\in\mathbf{T}_h} |z_h|^2_{1,2,K}\,,$$

woraus sich mit (4.22) insgesamt

$$\|v_h\|_{0,6,\Omega} \le C\,\|z_h\|_h \tag{4.24}$$

ergibt.

Für die Abschätzung des Fehlers $z_h - v_h$ benutzen wir (4.15), (4.23), (4.17) und erhalten

$$\begin{aligned}
\|z_h - v_h\|^6_{0,6,\Omega} &\le \sum_{K\in\mathbf{T}_h} C\, h_K^d\, \|z_h - v_h\|^6_{0,\infty,K} \\
&\le \sum_{K\in\mathbf{T}_h} C\, h_K^{d+6}\, C_T^5 \sum_{\widetilde{K}\in T(K)} |z_h|^6_{1,\infty,\widetilde{K}} \\
&\le \sum_{K\in\mathbf{T}_h} C\, C_T^5\, C_1^{-d-6} \sum_{\widetilde{K}\in T(K)} h_{\widetilde{K}}^{d+6}\, |z_h|^6_{1,\infty,\widetilde{K}} \\
&\le C\, C_T^6\, C_1^{-d-6} \sum_{K\in\mathbf{T}_h} h_K^{d+6}\, |z_h|^6_{1,\infty,K}\,.
\end{aligned}$$

In der letzten Summe können wir jeden Summanden mit Hilfe einer inversen Ungleichung und unter Verwendung von $d{\le}3$ abschätzen durch

$$h_K^{d+6}\, |z_h|^6_{1,\infty,K} \le C\, h_K^{6-2d}\, |z_h|^6_{1,2,K} \le C\, |z_h|^6_{1,2,K}\,,$$

woraus sich insgesamt

$$\|z_h - v_h\|^6_{0,6,\Omega} \le C \sum_{K\in\mathbf{T}_h} |z_h|^6_{1,2,K} \le C \left(\sum_{K\in\mathbf{T}_h} |z_h|^2_{1,2,K} \right)^3 = C\,\|z_h\|_h^6$$

ergibt. Die Ungleichung (4.20) für $p{=}6$ folgt mit (4.24) über die Dreiecksungleichung. Die Abschätzung (4.21) für die vektorwertige Funktion $w_h \in V_h$ ergibt sich, indem wir auf jede Komponente $(w_h)_\mu \in S_h$, $\mu = 1, \ldots, d$, die Ungleichung (4.20) anwenden. □

4.3 Divergenz-erhaltende Interpolation

Für die Abschätzung des Diskretisierungsfehlers und für den Nachweis der diskreten Babuška-Brezzi-Bedingung benötigen wir einen Interpolationsoperator $i_h : V \to V_h$, bei dem $i_h v$ die Divergenz von v in einem gewissen Sinne (siehe (4.26) und Bemerkung 4.7) erhält. Insbesondere soll die Interpolierte der kontinuierlichen Lösung $u \in V$ mit $\nabla \cdot u = 0$ diskret-divergenzfrei sein, d.h. $i_h u \in W_h$.

DEFINITION 4.5 *Sei $V_h \subset span^{(d)}\{\,\phi_j\,,\,j \in J(\Omega)\}$ einer der vier Finite-Elemente-Räume aus Abschnitt 2.2. Dann nennen wir zu einer gegebenen Funktion $v \in V = (H_0^1(\Omega))^d$ die Funktion $i_h v \in V_h$ mit*

$$i_h v(x) \;=\; \sum_{j \in J(\Omega)} v_j \phi_j(x) \qquad und \qquad v_j \;:=\; \frac{1}{|\Gamma_j|}\int_{\Gamma_j} v\,ds\,. \tag{4.25}$$

die **divergenz-erhaltende Interpolierte** *zu v.*

Nach dem Spursatz ist $i_h v$ für jede Funktion $v \in V{=}(H_0^1(\Omega))^d$ erklärt. Da die Randwerte von $v \in V$ verschwinden, gilt $v_j = 0$ für Indizes $j \in J(\partial\Omega)$, die zu Randseiten $\Gamma_j \subset \partial\Omega$ gehören. Daraus ergibt sich $i_h v \in V_h$. Im Falle eines mittelwert-orientierten Finite-Elemente-Raumes V_h sind $i_h v$ und die Finite-Elemente-Interpolierte $I_h v$ identisch.

Der Interpolationsoperator i_h hat folgende Eigenschaften.

LEMMA 4.6 *Sei V_h einer der vier Räume aus Abschnitt 2.2, Q_h aus (2.9) der Raum der stückweise konstanten Funktionen und $b_h(.,.)$ die in (3.6) definierte Bilinearform. Dann gilt für den durch (4.25) definierten Interpolationsoperator $i_h : V \to V_h$ die Beziehung*

$$b_h(q_h, i_h v) \;=\; b(q_h, v) \;=\; -(q_h, \nabla\cdot v) \qquad \forall\, q_h \in Q_h \quad \forall\, v \in V \tag{4.26}$$

sowie

$$i_h v \in W_h \qquad \forall\, v \in V \quad mit \quad \nabla\cdot v = 0\,. \tag{4.27}$$

Beweis.
Die Beziehung (4.26) folgt durch Einsetzen von (4.25) in die Definition von $b_h(\cdot,\cdot)$ unter Ausnutzung der Eigenschaften $1/|\Gamma_j| \int_{\Gamma_j} \phi_i\, ds = \delta_{ij}$ für mittelwert-orientierte bzw. $\phi_i(B_j) = \delta_{ij}$ für punktwert-orientierte Elemente. (4.27) ergibt sich aus (4.26) und der Definition von W_h. □

BEMERKUNG 4.7 *Der Operator i_h ist in folgendem Sinne divergenz-erhaltend. Für eine Funktion $v \in V$ ist die Divergenz $\nabla\cdot v \in L^2(\Omega)$ durch*

$$(q, \nabla\cdot v) \;=\; -\,b(q, v) \qquad \forall\, q \in L^2(\Omega)$$

an die Bilinearform $b(\cdot, v)$ gekoppelt. Da die diskrete Bilinearform $b_h(\cdot,\cdot)$ eine Approximation von $b(\cdot,\cdot)$ ist, können wir für eine nichtkonforme Funktion $v_h \in V_h$ eine diskrete Divergenz $\nabla_h \cdot v_h \in L^2(\Omega)$ über

$$(q, \nabla_h\cdot v_h) \;=\; -\,b_h(q, v_h) \qquad \forall\, q \in L^2(\Omega)$$

definieren. Dann bedeutet die divergenz-erhaltende Eigenschaft (4.26), daß für jedes $v \in V$ gilt

$$(q, \nabla_h\cdot(i_h v)) \;=\; (q, \nabla\cdot v) \qquad \forall\, q \in Q_h\,,$$

d.h., daß der Fehler "$\nabla\cdot v - \nabla_h\cdot(i_h v)$" orthogonal zum Raum Q_h ist.

Im folgenden wollen wir die L^2-Norm einer Funktion $v \in V$ auf der Randseite eines Elementes sowie auf der Randseite eines dualen Gebietes abschätzen.

LEMMA 4.8 *Sei* $\Gamma \in \{\Gamma_l \,:\, l \in J(K)\} \cup \{\Gamma_{lk} \,:\, l,k \in J(K),\ k \in \Lambda_l\}$ *die Randseite eines Elementes* $K \in \mathbf{T}_h$ *oder eine in* K *liegende Randseite eines dualen Gebietes (zur Definition von* Γ_{lk} *und* Λ_l *siehe Abschnitt 3.2). Dann gilt für eine beliebige Funktion* $v \in (H^1(K))^d$ *die Abschätzung*

$$||v||_{0,2,\Gamma} \;\leq\; C\, h_K^{-1/2}\,(\,||v||_{0,2,K} \,+\, h_K |v|_{1,2,K}\,) \tag{4.28}$$

mit einer von K *und* h *unabhängigen Konstanten* C.

Beweis.
Für $l \in J(K)$ und $k \in \Lambda_l \cap J(K)$ sei G_{lk} dasjenige d-dimensionale Simplex, welches die

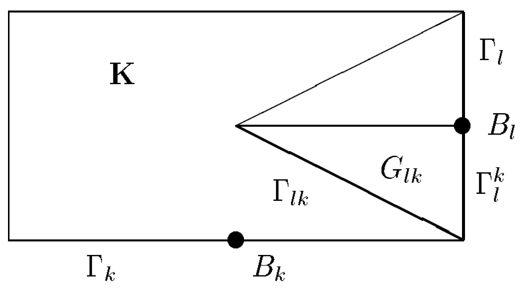


Abbildung 4.2: *Konstruktion von* G_{lk}

Randseite Γ_{lk} des dualen Gebietes R_l als $(d-1)$-dimensionale Seite und den Knoten B_l als zusätzliche Ecke hat (siehe Abbildung 4.2). Wir bezeichnen mit $\Gamma_l^k := \partial G_{lk} \cap \Gamma_l$ die Randseite von G_{lk}, die zum Randteil Γ_l des Elementes K gehört. Sei weiterhin $F_{lk} : \widehat{G} \to G_{lk}$ die affine Transformation von dem Referenz-Simplex $\widehat{G}$ (definiert durch die Ecken 0, $\hat{e}_1, \ldots, \hat{e}_d$, wobei $\hat{e}_i$ der i-te Einheitsvektor ist) auf das Simplex G_{lk}, die eine Darstellung der Form

$$x \;=\; F_{lk}(\hat{x}) \;=\; B_{lk}\hat{x} \,+\, b_{lk}$$

mit einer $d\times d$–Matrix B_{lk} und einem Vektor $b_{lk} \in \mathbf{R}^d$ haben möge, und sei $\hat{v}(\hat{x}) := v(F_{lk}(\hat{x}))$ die zu v gehörende Funktion auf $\widehat{G}$. Für eine beliebige $(d-1)$-dimensionale Randseite Γ von G_{lk} gilt dann die Beziehung

$$||v||_{0,2,\Gamma} \;=\; \left(|\Gamma|/|\hat{\Gamma}|\right)^{1/2} ||\hat{v}||_{0,2,\hat{\Gamma}}\,,$$

wobei $\hat{\Gamma} = F_{lk}^{-1}(\Gamma)$ die entsprechende Randseite von $\widehat{G}$ ist. Wenden wir den Spursatz auf $\widehat{G}$ an, so folgt

$$\|\hat{v}\|_{0,2,\hat{\Gamma}} \;\leq\; C\,(\,\|\hat{v}\|_{0,2,\widehat{G}} \,+\, |\hat{v}|_{1,2,\widehat{G}}\,)\,.$$

Da das Gitter $\mathbf{T}_h$ als shape-regulär vorausgesetzt wurde, ist auch das Simplex G_{lk} shape-regulär und es gelten die Abschätzungen (siehe [12])

$$\begin{aligned} |\hat{v}|_{m,2,\widehat{G}} &\leq C\,\|B_{lk}\|^m\,|G_{lk}|^{-1/2}\,|v|_{m,2,G_{lk}} \qquad m=0,1,\\ \|B_{lk}\| &\leq C\,h_{lk}\,, \end{aligned}$$

wobei $h_{lk} := \mathrm{diam}(G_{lk})$ den Durchmesser von G_{lk} bezeichnet. Insgesamt erhalten wir daraus

$$\begin{aligned} \|v\|_{0,2,\Gamma} &\leq C\,\left(|\Gamma|/|G_{lk}|\right)^{1/2}\left(\|v\|_{0,2,G_{lk}} + h_{lk}\,|v|_{1,2,G_{lk}}\right)\\ &\leq C\,h_{lk}^{-1/2}\left(\|v\|_{0,2,K} + h_K|v|_{1,2,K}\right). \end{aligned}$$

Da das Gitter shape-regulär ist, haben wir $h_{lk} \geq C\,h_K$ und es folgt die Abschätzung (4.28) für jede Randseite Γ von G_{lk}, also insbesondere auch für $\Gamma = \Gamma_{lk}$. Die Behauptung für $\Gamma = \Gamma_l$ ergibt sich aus der Zerlegung

$$\|v\|_{0,2,\Gamma_l}^2 = \sum_{k\in\Lambda_l\cap J(K)} \|v\|_{0,2,\Gamma_l^k}^2$$

und der bereits bewiesenen Abschätzung (4.28) für die Randseite $\Gamma = \Gamma_l^k$ von G_{lk}. □

Mit Hilfe der Abschätzung (4.28) können wir nun eine lokale Stabilitätseigenschaft des divergenz-erhaltenden Interpolationsoperators i_h nachweisen.

LEMMA 4.9 *Sei $V_h \subset span^{(d)}\{\,\phi_j\,,\,j\in J(\Omega)\}$ einer der vier Finite-Elemente-Räume aus Abschnitt 2.2 und $i_h : V \to V_h$ der in (4.25) definierte Interpolationsoperator. Dann gilt für eine beliebige Funktion $v \in V$, ein beliebiges Element $K\in\mathbf{T}_h$ und beliebiges ganzzahliges $m \geq 0$ die Stabilitätsabschätzung*

$$|i_h v|_{m,2,K} \leq C \sum_{j\in J(K)} |\phi_j|_{m,\infty,K}\,\left(\,\|v\|_{0,2,K} + h_K\,|v|_{1,2,K}\,\right). \tag{4.29}$$

Beweis.
Aus (4.28) folgt für v_j aus (4.25) mit $j \in J(K)$

$$|v_j| \leq C\,(|\Gamma_j|\,h_K)^{-1/2}\left(\,\|v\|_{0,2,K} + h_K\,|v|_{1,2,K}\,\right).$$

Da die Basisfunktionen ϕ_j mit $j \notin J(K)$ auf dem Element K verschwinden, gilt

$$\begin{aligned} |i_h v|_{m,2,K} &\leq \sum_{j\in J(K)} |v_j|\,|K|^{1/2}\,|\phi_j|_{m,\infty,K}\\ &\leq C\sum_{j\in J(K)}\left(\frac{|K|}{|\Gamma_j|\,h_K}\right)^{1/2}\left(\,\|v\|_{0,2,K} + h_K\,|v|_{1,2,K}\,\right)|\phi_j|_{m,\infty,K}\,. \end{aligned}$$

Die Behauptung (4.29) ergibt sich daraus, daß für ein shape-reguläres Gitter der Term $|K|/(|\Gamma_j|\,h_K)$ gleichmäßig beschränkt ist. □

Eine Aussage über den lokalen Interpolationsfehler von i_h macht das folgende Lemma.

LEMMA 4.10 *Sei $i_h : V \to V_h$ der divergenz-erhaltende Interpolationsoperator in (4.25) und $\widetilde{h}_K$ die folgende modifizierte Gitterweite eines Elementes $K \in \mathbf{T}_h$*

$$\widetilde{h}_K \;:=\; \begin{cases} h_K & \text{, für } P_1\text{-parametrische mittelwert-orientierte Elemente,} \\ h_K + \sigma_K & \text{, sonst.} \end{cases} \tag{4.30}$$

Dann gelten für eine beliebige Funktion $v \in (H^2(\Omega))^d \cap V$ und ein beliebiges Element $K \in \mathbf{T}_h$ die Abschätzungen

$$|v - i_h v|_{m,2,K} \;\leq\; C \widetilde{h}_K \, h_K^{1-m} \, \|v\|_{2,2,K} \,, \qquad m = 0, 1 \,, \tag{4.31}$$

und

$$\|v - i_h v\|_{0,2,\Gamma_{lk}} \;\leq\; C \widetilde{h}_K \, h_K^{1/2} \, \|v\|_{2,2,K} \qquad \forall \, l, k \in J(K) \,, \; k \in \Lambda_l \,. \tag{4.32}$$

Beweis.
Wir benutzen den zum Finite-Elemente-Raum V_h gehörenden natürlichen Interpolationsoperator $I_h : (H^2(\Omega))^d \cap V \to V_h$ (siehe Abschnitt 2.3) mit

$$I_h v(x) \;=\; \sum_{j \in J(\Omega)} N_j(v) \, \phi_j(x) \,.$$

Im Falle von mittelwert-orientierten Elementen sind i_h und I_h identisch und (4.31) folgt aus der Abschätzung des Interpolationsfehlers von I_h in Lemma 2.14.

Sei nun V_h ein Raum punktwert-orientierter Elemente, d.h. $N_j(v) = v(B_j)$ mit B_j als Schwerpunkt der Randseite Γ_j. Sei weiterhin $K \in \mathbf{T}_h$ ein beliebiges Element und $I_1 v \in (P_1(K))^d$ eine lineare Interpolierte von $v \in V$ auf K, die den Abschätzungen

$$|v - I_1 v|_{m,2,K} \;\leq\; C \, h_K^{2-m} |v|_{2,2,K} \,, \qquad m = 0, 1, 2 \,, \tag{4.33}$$

genügt (z.B. ein gemitteltes Taylor-Polynom [7]). Über die Dreiecksungleichung erhalten wir

$$|v - i_h v|_{m,2,K} \;\leq\; |v - I_1 v|_{m,2,K} \;+\; |I_1 v - i_h(I_1 v)|_{m,2,K} \;+\; |i_h(I_1 v - v)|_{m,2,K} \,. \tag{4.34}$$

Verwenden wir die lokale Stabilität (4.29) von i_h, (4.33) sowie die Abschätzung $|\phi_j|_{m,\infty,K} \leq C h_K^{-m}$ für $m = 0, 1$ (ergibt sich aus (2.50) und (2.24)), so folgt

$$|i_h(I_1 v - v)|_{m,2,K} \;\leq\; C \, h_K^{-m} \, h_K^2 \, |v|_{2,2,K} \;\leq\; C \, \widetilde{h}_K \, h_K^{1-m} \, \|v\|_{2,2,K} \,. \tag{4.35}$$

Aus der Abschätzung (2.64) des Interpolationsfehlers von I_h und (4.33) können wir schlußfolgern

$$|I_1 v - I_h(I_1 v)|_{m,2,K} \;\leq\; C \, \widetilde{h}_K \, h_K^{1-m} \, \|I_1 v\|_{2,2,K} \;\leq\; C \, \widetilde{h}_K \, h_K^{1-m} \, \|v\|_{2,2,K} \,. \tag{4.36}$$

Für die Funktion $w := I_1 v \in (P_1(K))^d$ gilt für $x \in K$

$$I_h w(x) - i_h w(x) \;=\; \sum_{j \in J(K)} (w(B_j) - w_j) \, \phi_j(x) \qquad \text{mit} \qquad w_j \;:=\; \frac{1}{|\Gamma_j|} \int_{\Gamma_j} w \, ds \,.$$

Im 2D-Fall haben wir $w(B_j) - w_j = 0$. Im 3D-Fall gilt dies im allgemeinen nicht mehr, man kann aber für beliebiges $w \in (P_1(K))^d$ und $j \in J(K)$ die Abschätzung

$$|w(B_j) - w_j| \;\leq\; C\,\sigma_K h_K\,|w|_{1,\infty,K} \;\leq\; C\,\sigma_K h_K\,|K|^{-1/2}\,|w|_{1,2,K} \tag{4.37}$$

nachweisen. Damit ergibt sich

$$|I_h w - i_h w|_{m,2,K} \;\leq\; \sum_{j\in J(K)} C\,\sigma_K h_K\,|w|_{1,2,K}|\phi_j|_{m,\infty,K}\,.$$

Verwenden wir $|\phi_j|_{m,\infty,K} \leq C h_K^{-m}$ und

$$|w|_{1,2,K} \;\leq\; |v|_{1,2,K} + |v - I_1 v|_{1,2,K} \;\leq\; C\,\|v\|_{2,2,K}\,,$$

so folgt wegen $\sigma_K \leq \widetilde{h}_K$

$$|I_h(I_1 v) - i_h(I_1 v)|_{m,2,K} \;\leq\; C\,\widetilde{h}_K\,h_K^{1-m}\,\|v\|_{2,2,K}\,.$$

Mit (4.36), (4.35), der Dreiecksungleichung (4.34) und (4.33) erhalten wir die Behauptung (4.31).

Die Abschätzung (4.32) ist eine einfache Schlußfolgerung aus (4.28) und (4.31). □

4.4 Babuška-Brezzi-Stabilität

Um eine stabile Druckapproximation zu erhalten, benötigen wir, daß das Raumpaar (V_h, Q_h) die sogenannte *diskrete Babuška-Brezzi-Bedingung* [8],[22] (oder auch *diskrete inf-sup-Bedingung*)

$$\inf_{0\neq q_h\in Q_h}\;\sup_{0\neq v_h\in V_h}\frac{b_h(q_h, v_h)}{\|q_h\|_{0,2,\Omega}\,\|v_h\|_h} \;\geq\; \beta \;>\; 0 \tag{4.38}$$

erfüllt. Die Gültigkeit dieser Stabilitätsbedingung ist natürlich auch an die konkrete Wahl der Bilinearform $b_h(.,.)$ gekoppelt. Wir wollen daran erinnern, daß wir im Fall von mittelwert-orientierten Elementen die Bilinearform $\bar{b}_h$ aus (3.4) und im Fall von punktwert-orientierten Elementen $\widetilde{b}_h$ aus (3.5) verwenden. Hätten wir für die punktwert-orientierten Elemente auch die Bilinearform $\bar{b}_h$ gewählt, so würden wir zur Gewährleistung der diskreten Babuška-Brezzi-Bedingung die zusätzliche Gitterforderung $\sigma_h \leq \sigma_0$ mit einem hinreichend kleinen σ_0 (siehe [48]) benötigen. Verwenden wir hingegen die numerisch integrierte Form $\widetilde{b}_h$, so entfällt diese Forderung.

Zum Nachweis der diskreten inf-sup-Bedingung benutzen wir zum einen die bekannte Stabilitätsbedingung (siehe [22]) für die kontinuierlichen Räume $V{:=}(H_0^1(\Omega))^d$ und $Q{:=}\mathrm{L}_0^2(\Omega)$

$$\inf_{0\neq q\in Q}\;\sup_{0\neq v\in V}\frac{b(q, v)}{\|q\|_{0,2,\Omega}\,|v|_{1,2,\Omega}} \;\geq\; \beta_0 \;>\; 0 \tag{4.39}$$

und zum anderen den divergenz-erhaltenden Interpolationsoperator i_h aus Abschnitt 4.3. Zunächst wollen wir die H^1-Stabilität von i_h beweisen.

LEMMA 4.11 *Für den in (4.25) definierten divergenz-erhaltenden Interpolationsoperator $i_h : V \to V_h$ gilt die Stabilitätsabschätzung*

$$\|i_h v\|_h \leq C\,|v|_{1,2,\Omega} \qquad \forall\, v \in V\,. \tag{4.40}$$

Beweis.
Sei $K \in \mathbf{T}_h$ ein beliebiges Element und $I_0 v \in (P_0(K))^d$ eine konstante Interpolierte von $v \in V$ (z.B. $I_0 v = 1/|K| \int_K v\, ds$), die den Abschätzungen

$$|v - I_0 v|_{m,2,K} \leq C\, h_K^{1-m} |v|_{1,2,K}\,, \qquad m = 0, 1\,,$$

genügt. Sowohl die Definition 4.5 des Operators i_h als auch die Abschätzung in Lemma 4.9 lassen sich sinngemäß auf Funktionen $w \in (H^1(\Omega))^d$ erweitern. Mit Hilfe der Ungleichung (4.29) für m=1 angewendet auf die Funktion $w = v - I_0 v$ und der Abschätzung

$$|\phi_j|_{1,\infty,K} \leq C\, h_K^{-1} \qquad \forall\, j \in J(K)$$

(welche aus (2.50) und (2.24) folgt) erhalten wir

$$|i_h(v - I_0 v)|_{1,2,K} \leq C\,|v|_{1,2,K}\,.$$

Verwenden wir, daß die konstante Funktion $I_0 v$ von i_h exakt interpoliert wird, so ergibt sich

$$|i_h v|_{1,2,K} \leq |i_h(v - I_0 v)|_{1,2,K} + |I_0 v|_{1,2,K} \leq C\,|v|_{1,2,K}\,.$$

Durch Quadrieren und Summation über alle $K \in \mathbf{T}_h$ erhalten wir die Behauptung (4.40).
□

Aus (4.39), (4.26) und (4.40) ergibt sich unmittelbar die diskrete Babuška-Brezzi-Bedingung für unsere Raumpaare (V_h, Q_h).

LEMMA 4.12 *Sei V_h einer der vier Räume aus Abschnitt 2.2, Q_h aus (2.9) der Raum der stückweise konstanten Funktionen und $b_h(.,.)$ gemäß (3.6) gewählt. Dann gilt die diskrete inf-sup-Bedingung (4.38) mit einer h-unabhängigen Konstanten $\beta \geq \beta_0/C$, wobei C die Konstante in der Stabilitätsabschätzung (4.40) ist.*

4.5 Konsistenzfehler der nichtkonformen Approximation

In den Navier–Stokes–Gleichungen werden die Terme "$-\triangle u$" und "∇p" mit Hilfe der Bilinearformen $a_h(\cdot,\cdot)$ und $b_h(\cdot,\cdot)$ diskretisiert. Auf Grund der nichtkonformen Finite-Elemente-Approximation (d.h. $V_h \not\subset V$) und auf Grund der numerischen Integration in der Bilinearform $b_h = \tilde{b}_h$ im Falle punktwert-orientierter Elemente entsteht dabei jeweils ein Konsistenzfehler. Konkret treten folgende Fehlerfunktionale auf.

DEFINITION 4.13 *Für festes* $u \in (H^2(\Omega))^d$ *und* $p \in H^1(\Omega)$ *bezeichnen* $E_u^{(1)}(\cdot)$ *und* $E_p^{(2)}(\cdot)$ *die folgenden, für beliebiges* $v_h \in V_h$ *definierten* **Konsistenzfehlerfunktionale**

$$E_u^{(1)}(v_h) \;:=\; a_h(u, v_h) - (-\triangle u, v_h)\,, \tag{4.41}$$

$$E_p^{(2)}(v_h) \;:=\; b_h(p, v_h) - (\nabla p, v_h)\,. \tag{4.42}$$

Im folgenden geben wir Abschätzungen dieser Fehlerfunktionale an, die wir später für unsere Konvergenzanalysis benötigen.

LEMMA 4.14 *Sei* $\overline{h}_K$ *die folgende modifizierte Gitterweite*

$$\overline{h}_K \;:=\; \begin{cases} h_K & \text{, wenn } V_h \text{ ein mittelwert-orientierter Raum ist,} \\ h_K + \sigma_K & \text{, wenn } V_h \text{ ein punktwert-orientierter Raum ist,} \end{cases}$$

und $\overline{h} := \max_{K \in \mathbf{T}_h} \overline{h}_K$. *Dann gilt für* $u \in (H^2(\Omega))^d$ *die Abschätzung*

$$|E_u^{(1)}(v_h)| \;\leq\; C\,\overline{h}\,\|u\|_{2,2,\overline{h}}\,\|v_h\|_h \qquad \forall\, v_h \in V_h\,. \tag{4.43}$$

Der Beweis dieser Abschätzung ist im wesentlichen identisch mit dem von Rannacher und Turek in [48]. Man muß lediglich anstelle der globalen Sobolev-Norm von u die adaptiv gewichtete verwenden.

Da wir für die punktwert-orientierten Elemente die numerisch integrierte Bilinearform $b_h = \tilde{b}_h$ verwenden, erhalten wir bei der Abschätzung des Fehlerfunktionals $E_p^{(2)}$ im Vergleich zu [48] ein verbessertes Resultat. Aus diesem Grunde werden wir den Beweis des folgenden Lemmas angeben.

LEMMA 4.15 *Für* $p \in H^1(\Omega)$ *gilt die Abschätzung*

$$|E_p^{(2)}(v_h)| \;\leq\; C\,h\,|p|_{1,2,h}\,\|v_h\|_h \qquad \forall\, v_h \in V_h\,. \tag{4.44}$$

Beweis.
Zu gegebenem $v_h \in V_h$ seien $v_j := N_j(v_h)$, $j \in J(\Omega)$, die entsprechenden Knotenwerte, die für Randknoten $j \in J(\partial\Omega)$ verschwinden (vgl. Abschnitt 2.2). Wir betrachten zunächst den Fall mittelwert-orientierter Elemente mit $v_j = |\Gamma_j|^{-1} \int_{\Gamma_j} v_h\, ds$ und $b_h = \overline{b}_h$ (siehe (3.4)), d.h.

$$b_h(p, v_h) \;=\; -\sum_{K \in \mathbf{T}_h} (p, \nabla \cdot v_h)_K\,.$$

Dann gilt unter der Voraussetzung $p \in H^1(\Omega)$

$$E_p^{(2)}(v_h) \;=\; -\sum_{K \in \mathbf{T}_h} \sum_{j \in J(K)} \int_{\Gamma_j} p\,(v_h - v_j) \cdot n_j\, ds\,,$$

wobei n_j den bezüglich K äußeren Normaleneinheitsvektor auf Γ_j bezeichnet und für v_h der vom Element K stammende Wert zu nehmen ist (aus Gründen der Übersichtlichkeit unterdrücken wir diese K-Abhängigkeit in unserer Bezeichnung). Sei $p_K \in P_0(K)$ die folgende konstante Interpolierte zu p auf dem Element K

$$p_K := \frac{1}{|K|} \int_K p \, dx \, .$$

Dann haben wir

$$E_p^{(2)}(v_h) \; = \; - \sum_{K \in \mathbf{T}_h} (\, A_K \, + \, B_K \,)$$

mit

$$A_K := \sum_{j \in J(K)} \int_{\Gamma_j} (p - p_K) \, (v_h - v_j) \cdot n_j \, ds \qquad \text{und} \qquad B_K := p_K \sum_{j \in J(K)} \int_{\Gamma_j} (v_h - v_j) \cdot n_j \, ds \, .$$

Verwenden wir die Ungleichung (4.28) und Standard-Interpolationsabschätzungen für die konstanten Interpolierten $p_K \approx p$ und $v_j \approx v_h$, so ergibt sich

$$|A_K| \; \leq \; \sum_{j \in J(K)} \|p - p_K\|_{0,2,\Gamma_j} \, \|v_h - v_j\|_{0,2,\Gamma_j} \; \leq \; C \, h_K \, |p|_{1,2,K} \, |v_h|_{1,2,K} \, .$$

Auf Grund der Definition von v_j gilt $\int_{\Gamma_j} (v_h - v_j) \, ds = 0$ und damit $B_K = 0$. Daraus ergibt sich insgesamt

$$|E_p^{(2)}(v_h)| \; \leq \; C \, h \sum_{K \in \mathbf{T}_h} (\frac{h_K}{h}) \, |p|_{1,2,K} \, |v_h|_{1,2,K} \; \leq \; C \, h \, |p|_{1,2,h} \, \|v_h\|_h \, .$$

Im Falle punktwert-orientierter Elemente haben wir $v_j = v_h(B_j)$ und $b_h = \widetilde{b}_h$ (siehe (3.5)), d.h.

$$b_h(p, v_h) \; = \; - \sum_{K \in \mathbf{T}_h} \sum_{j \in J(K)} p_K \, |\Gamma_j| \, v_h(B_j) \cdot n_j \; = \; - \sum_{K \in \mathbf{T}_h} \sum_{j \in J(K)} p_K \int_{\Gamma_j} v_j \cdot n_j \, ds \, .$$

Für $p \in H^1(\Omega)$ gilt dann

$$E_p^{(2)}(v_h) \; = \; \sum_{K \in \mathbf{T}_h} (p - p_K, \nabla \cdot v_h)_K \; - \; \sum_{K \in \mathbf{T}_h} \sum_{j \in J(K)} \int_{\Gamma_j} (p - p_K) \, (v_h - v_j) \cdot n_j \, ds$$

und die Behauptung (4.44) ergibt sich aus

$$| \, (p - p_K, \nabla \cdot v_h)_K \, | \; \leq \; C \, h_K \, |p|_{1,2,K} \, |v_h|_{1,2,K}$$

und der obigen Abschätzung für $|A_K|$. □

4.6 Analysis des Konvektionstermes

Zunächst zeigen wir die Stetigkeit von $\bar{n}_h(\cdot,\cdot,\cdot)$ auf V_h^3, die wir für die Abschätzung des Konsistenzfehlers benötigen.

LEMMA 4.16 *Unter den Voraussetzungen (A1) und (A3) aus Abschnitt 3.1 gilt*

$$|\,\bar{n}_h(z_h^1,u_h,v_h)\;-\;\bar{n}_h(z_h^2,u_h,v_h)\,|\;\le\;C\,\|z_h^1-z_h^2\|_h\,\|u_h\|_h\,\|v_h\|_h \tag{4.45}$$

für beliebige Funktionen $z_h^1, z_h^2, u_h, v_h \in V_h$ mit einer von h und ν unabhängigen Konstanten C .

Beweis.
Wir setzen $t_{lk}^i := \frac{1}{\nu}\int_{\Gamma_{lk}} z_h^i\cdot n_{lk}\,d\gamma$ für $i=1,2$ und erhalten mit Hilfe von (A1)

$$\begin{aligned} D &:= \tilde{n}_h(z_h^1,u_h,v_h)\;-\;\tilde{n}_h(z_h^2,u_h,v_h)\\ &= \sum_{l\in J}\sum_{k\in\Lambda_l}\nu\left\{t_{lk}^1\Phi(-t_{lk}^1)-t_{lk}^2\Phi(-t_{lk}^2)\right\}\left(u_h^{(k)}-u_h^{(l)}\right)\cdot v_h^{(l)}\,. \end{aligned}$$

Im folgenden bezeichnen wir mit K das Element, welches Γ_{lk} enthält. Verwenden wir (A3) sowie inverse Ungleichungen (siehe Lemma 2.16), so ergibt sich

$$\begin{aligned} |D| &\le C\sum_{l\in J}\sum_{k\in\Lambda_l}\left|\int_{\Gamma_{lk}}(z_h^1-z_h^2)\cdot n_{lk}\,d\gamma\right|\,|u_h^{(k)}-u_h^{(l)}|\,|v_h^{(l)}|\\ &\le C\sum_{l\in J}\sum_{k\in\Lambda_l}|\Gamma_{lk}|\;\|z_h^1-z_h^2\|_{0,\infty,K}\,h_K\,|u_h|_{1,\infty,K}\,\|v_h\|_{0,\infty,K}\\ &\le C\sum_{l\in J}\sum_{k\in\Lambda_l}h_K^{1-d}|\Gamma_{lk}|\;\|z_h^1-z_h^2\|_{0,4,K}\;|u_h|_{1,2,K}\,\|v_h\|_{0,4,K}\,. \end{aligned}$$

Mit Hilfe von $|\Gamma_{lk}|\le C h_K^{d-1}$ und der diskreten Sobolev-Ungleichung (4.21) erhalten wir

$$|D|\;\le\;C\,\|z_h^1-z_h^2\|_{0,4,\Omega}\,\|u_h\|_h\,\|v_h\|_{0,4,\Omega}\;\le\;C\,\|z_h^1-z_h^2\|_h\,\|u_h\|_h\,\|v_h\|_h\,.$$

Sei $w_h\in V_h$ die Funktion $w_h := z_h^2 - z_h^1$ und $w^{(l)} := N_l(w_h)$ der Wert des entsprechenden Knotenfunktionals zum Knoten $l\in J$ (siehe (2.37)). Dann gilt für die Trilinearform $\tilde{n}_h^{(2)}$ aus (3.18)

$$\begin{aligned} E &:= \tilde{n}_h^{(2)}(z_h^1,u_h,v_h)\;-\;\tilde{n}_h^{(2)}(z_h^2,u_h,v_h)\;=\;\sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}}w_h\cdot n_{lk}\,d\gamma\;\;u_h^{(l)}\cdot v_h^{(l)}\\ &= \sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}}(w_h-w^{(l)})\cdot n_{lk}\,d\gamma\;\;u_h^{(l)}\cdot v_h^{(l)}\,. \end{aligned}$$

Mit Hilfe der Abschätzung

$$\|w_h-w^{(l)}\|_{0,\infty,K}\;\le\;C\,h_K\,|w_h|_{1,\infty,K}$$

und unter Verwendung inverser Ungleichungen folgt

$$\begin{aligned}
|E| &\leq \sum_{l\in J}\sum_{k\in\Lambda_l} C\,|\Gamma_{lk}|\,h_K\,|w_h|_{1,\infty,K}\,\|u_h\|_{0,\infty,K}\,\|v_h\|_{0,\infty,K} \\
&\leq \sum_{l\in J}\sum_{k\in\Lambda_l} C\,|\Gamma_{lk}|\,h_K^{1-d}\,|w_h|_{1,2,K}\,\|u_h\|_{0,4,K}\,\|v_h\|_{0,4,K} \\
&\leq C\,\|w_h\|_h\,\|u_h\|_{0,4,\Omega}\,\|v_h\|_{0,4,\Omega} \;\leq\; C\,\|w_h\|_h\,\|u_h\|_h\,\|v_h\|_h\,.
\end{aligned}$$

Aus der Definition von $\bar{n}_h$ in (3.21) ergibt sich schließlich

$$|\,\bar{n}_h(z_h^1,u_h,v_h) \;-\; \bar{n}_h(z_h^2,u_h,v_h)\,| \;\leq\; |D| + \frac{1}{2}|E| \;\leq\; C\,\|z_h^1 - z_h^2\|_h\,\|u_h\|_h\,\|v_h\|_h\,.$$

□

Das folgende Lemma macht eine Aussage über den Konsistenzfehler im Konvektionsterm, der dadurch entsteht, daß wir anstelle der Galerkin-Diskretisierung n_h in (3.7) die upwind-Diskretisierung $\bar{n}_h$ verwenden.

LEMMA 4.17 *Sei $u \in (H^2(\Omega)\cap H_0^1(\Omega))^d$ eine gegebene Funktion mit $\nabla\cdot u = 0$ und $i_h : (H_0^1(\Omega))^d \to V_h$ der in (4.25) definierte divergenz-erhaltende Interpolationsoperator. Dann gilt unter der Voraussetzung (A1) die Abschätzung*

$$|\,n_h(u,u,w_h) - \bar{n}_h(i_hu,i_hu,w_h)\,| \;\leq\; C\,\widetilde{h}\,\|w_h\|_h \qquad \forall\, w_h\in V_h\,. \tag{4.46}$$

Dabei ist $\widetilde{h} = \max_{K\in\mathbf{T}_h}\widetilde{h}_K$ mit der in (4.30) definierten modifizierten Schrittweite $\widetilde{h}_K$ und C eine Konstante der Gestalt

$$C \;=\; C\,\|u\|_{0,4,h}\,|u|_{1,4,h} \;+\; C\,\|u\|_{C(\Omega)}\,\|u\|_{2,2,\widetilde{h}}\,, \tag{4.47}$$

die bezüglich h und $\widetilde{h}$ gleichmäßig beschränkt ist.

Beweis.
Wir zerlegen n_h and $\bar{n}_h$ in der Form

$$\begin{aligned}
n_h(u,v,w) &= n_h^{(1)}(u,v,w) + n_h^{(2)}(u,v,w) \\
\bar{n}_h(u,v,w) &= \widetilde{n}_h^{(1)}(u,v,w) + \frac{3}{2}\,\widetilde{n}_h^{(2)}(u,v,w)
\end{aligned}$$

mit

$$\begin{aligned}
n_h^{(1)}(u,v,w) &= \sum_{K\in\mathbf{T}_h}(\nabla\cdot(uv),w)_K \qquad\qquad n_h^{(2)}(u,v,w) = -\sum_{K\in\mathbf{T}_h}(\nabla\cdot u, v\cdot w)_K \\
\widetilde{n}_h^{(1)}(u,v,w) &= \sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}} u\cdot n_{lk}\,d\gamma\,\Big\{\lambda_{lk}(u)v^{(l)} + (1-\lambda_{lk}(u))\,v^{(k)}\Big\}\cdot w^{(l)} \\
\widetilde{n}_h^{(2)}(u,v,w) &= -\sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}} u\cdot n_{lk}\,d\gamma\;v^{(l)}\cdot w^{(l)}\,.
\end{aligned}$$

Aus $\nabla\cdot u = 0$ folgt $n_h^{(2)}(u,u,w_h) = 0$ für beliebiges $w_h \in V_h$. Im folgenden verwenden wir die Abkürzungen $v_h := i_h u$ und $\lambda_{lk} := \lambda_{lk}(v_h)$. Es gilt

$$n_h(u,u,w_h) - \bar{n}_h(v_h,v_h,w_h) \; = \; Y + n_h^{(1)}(u,u,w_h - L_h w_h) - \frac{3}{2}\,\tilde{n}_h^{(2)}(v_h,v_h,w_h)$$

mit

$$\begin{aligned} Y &= n_h^{(1)}(u,u,L_h w_h) - \tilde{n}_h^{(1)}(v_h,v_h,w_h) \\ &= \sum_{l\in J}\sum_{k\in\Lambda_l} \int\limits_{\Gamma_{lk}} \Big\{ u\cdot n_{lk} u - v_h\cdot n_{lk}\left[\lambda_{lk} v_h^{(l)} + (1-\lambda_{lk}) v_h^{(k)}\right]\Big\}\cdot w_h^{(l)}\, d\gamma\,. \end{aligned}$$

Mit Hilfe der Interpolationseigenschaft des lumping-Operators L_h aus (3.11) können wir für beliebiges $w_h \in V_h$ abschätzen

$$\begin{aligned} |n_h^{(1)}(u,u,w_h - L_h w_h)| &\le \sum_{K\in\mathbf{T}_h} C\,\|u\|_{0,4,K}\,|u|_{1,4,K}\,h_K\,|w_h|_{1,2,K} \\ &\le C\,h\Big(\sum_{K\in\mathbf{T}_h}\frac{h_K^2}{h^2}\|u\|_{0,4,K}^4\Big)^{1/4}\Big(\sum_{K\in\mathbf{T}_h}\frac{h_K^2}{h^2}|u|_{1,4,K}^4\Big)^{1/4}\|w_h\|_h \\ &= C\,h\,\|u\|_{0,4,h}\,|u|_{1,4,h}\,\|w_h\|_h\,. \end{aligned}$$

Wegen $\nabla\cdot u = 0$ und $u \in (H_0^1(\Omega))^d$ gilt

$$\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}} (u\cdot n_{lk})\,d\gamma \; = \; 0 \qquad \forall\, l\in J$$

und wir erhalten

$$\begin{aligned} |\tilde{n}_h^{(2)}(v_h,v_h,w_h)| &= \Big|\sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}}(u-v_h)\cdot n_{lk}\,d\gamma\; v_h^{(l)}\cdot w_h^{(l)}\Big| \\ &\le \sum_{l\in J}\sum_{k\in\Lambda_l}\|u-v_h\|_{0,2,\Gamma_{lk}}\,|\Gamma_{lk}|^{1/2}\,\|v_h\|_{0,\infty,\Gamma_{lk}}\,\|w_h\|_{0,\infty,\Gamma_{lk}}\,. \end{aligned}$$

Aus der Definition (4.25) des Interpolationsoperators i_h und der gleichmäßigen Beschränktheit der Basisfunktionen ϕ_j ergibt sich für $v_h = i_h u$ die Abschätzung

$$\|v_h\|_{0,\infty,\Gamma_{lk}} \le C\,\|u\|_{C(\Omega)} \qquad \forall\, l\in J \quad \forall\, k\in\Lambda_l\,.$$

Daraus folgt

$$|\tilde{n}_h^{(2)}(v_h,v_h,w_h)| \le C\,\|u\|_{C(\Omega)}\,T$$

mit

$$T := \sum_{l\in J}\sum_{k\in\Lambda_l}\|u-v_h\|_{0,2,\Gamma_{lk}}\,|\Gamma_{lk}|^{1/2}\,\|w_h\|_{0,\infty,\Gamma_{lk}}\,.$$

Im folgenden bezeichnen wir innerhalb einer Summe $\sum_{k\in\Lambda_l}$ mit K stets dasjenige Element, welches Γ_{lk} enthält. Verwenden wir die Abschätzung (4.32) des Interpolationsfehlers von i_h, eine inverse Ungleichung sowie die diskrete Sobolev-Ungleichung

(4.21), so können wir den Term T abschätzen durch

$$\begin{aligned}
|T| &\le \sum_{l\in J}\sum_{k\in\Lambda_l} C\,\widetilde{h}_K\, h_K^{1/2}\,\|u\|_{2,2,K}\,|\Gamma_{lk}|^{1/2}\,\|w_h\|_{0,\infty,K}\\
&\le \sum_{l\in J}\sum_{k\in\Lambda_l} C\,\widetilde{h}_K\left(h_K^{1-d}\,|\Gamma_{lk}|\right)^{1/2}\|u\|_{2,2,K}\,\|w_h\|_{0,2,K}\\
&\le C\sum_{K\in\mathbf{T}_h}\widetilde{h}_K\,\|u\|_{2,2,K}\,\|w_h\|_{0,2,K} \;\le\; C\,\widetilde{h}\,\|u\|_{2,2,\widetilde{h}}\,\|w_h\|_{0,2,\Omega}\\
&\le C\,\widetilde{h}\,\|u\|_{2,2,\widetilde{h}}\,\|w_h\|_h\,.
\end{aligned}$$

Den Term Y zerlegen wir in die drei Terme $Y = Y_1 + Y_2 + Y_3$ mit

$$\begin{aligned}
Y_1 &= \sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}}(u-v_h)\cdot n_{lk}\; u\cdot w_h^{(l)}\,d\gamma\,,\\
Y_2 &= \sum_{l\in J}\sum_{k\in\Lambda_l}\int_{\Gamma_{lk}} v_h\cdot n_{lk}\,(u-v_h)\cdot w_h^{(l)}\,d\gamma\,,\\
Y_3 &= \sum_{l\in J}\sum_{k\in\Lambda_l}\int\limits_{\Gamma_{lk}} v_h\cdot n_{lk}\left\{\lambda_{lk}\left[v_h - v_h^{(l)}\right] + (1-\lambda_{lk})\left[v_h - v_h^{(k)}\right]\right\}\cdot w_h^{(l)}\,d\gamma\,.
\end{aligned}$$

Für Y_1 und Y_2 erhalten wir unter Verwendung der Abschätzung von T

$$\begin{aligned}
|Y_1| &\le \sum_{l\in J}\sum_{k\in\Lambda_l}\|u-v_h\|_{0,2,\Gamma_{lk}}\,|\Gamma_{lk}|^{1/2}\,\|u\|_{0,\infty,\Gamma_{lk}}\,\|w_h\|_{0,\infty,\Gamma_{lk}} \;\le\; \|u\|_{C(\Omega)}\,T\\
&\le C\,\|u\|_{C(\Omega)}\,\|u\|_{2,2,\widetilde{h}}\;\widetilde{h}\,\|w_h\|_h
\end{aligned}$$

und

$$\begin{aligned}
|Y_2| &\le \sum_{l\in J}\sum_{k\in\Lambda_l}|\Gamma_{lk}|^{1/2}\,\|v_h\|_{0,\infty,\Gamma_{lk}}\,\|u-v_h\|_{0,2,\Gamma_{lk}}\,\|w_h\|_{0,\infty,\Gamma_{lk}} \;\le\; C\,\|u\|_{C(\Omega)}\,T\\
&\le C\,\|u\|_{C(\Omega)}\,\|u\|_{2,2,\widetilde{h}}\;\widetilde{h}\,\|w_h\|_h\,.
\end{aligned}$$

Mit Hilfe der Beziehungen $\Gamma_{lk} = \Gamma_{kl}$, $\lambda_{lk} = 1-\lambda_{kl}$ und $n_{lk} = -n_{kl}$ können wir für Y_3 die Darstellung

$$Y_3 = \frac{1}{2}\sum_{l\in J}\sum_{k\in\Lambda_l}\int\limits_{\Gamma_{lk}} v_h\cdot n_{lk}\left[\lambda_{lk}\left\{v_h - v_h^{(l)}\right\} + (1-\lambda_{lk})\left\{v_h - v_h^{(k)}\right\}\right]\cdot\left[w_h^{(l)} - w_h^{(k)}\right]d\gamma\,.$$

zeigen. Daraus ergibt sich

$$\begin{aligned}
|Y_3| &\le \sum_{l\in J}\sum_{k\in\Lambda_l} C\,|\Gamma_{lk}|\,\|v_h\|_{0,\infty,K}\,h_K\,|v_h|_{1,\infty,K}\,h_K\,|w_h|_{1,\infty,K}\\
&\le C\,\|u\|_{C(\Omega)}\sum_{l\in J}\sum_{k\in\Lambda_l}\left(|\Gamma_{lk}|\,h_K^{1-d}\right)h_K\,|v_h|_{1,2,K}\,|w_h|_{1,2,K}\,.
\end{aligned}$$

Auf Grund der Abschätzung (4.31) des Interpolationsfehlers von $v_h = i_h u$ gilt

$$|v_h|_{1,2,K} \le |u|_{1,2,K} + |u-v_h|_{1,2,K} \le (1 + C\widetilde{h}_K)\|u\|_{2,2,K} \le C\,\|u\|_{2,2,K}$$

und wir erhalten mit $h_K \le \widetilde{h}_K$

$$|Y_3| \;\le\; C\,\|u\|_{C(\Omega)} \sum_{K\in\mathbf{T}_h} \widetilde{h}_K\,\|u\|_{2,2,K}\,|w_h|_{1,2,K} \;\le\; C\,\|u\|_{C(\Omega)}\,\|u\|_{2,2,\widetilde{h}}\;\widetilde{h}\;\|w_h\|_h\,.$$

Fassen wir alle Ungleichungen zusammen, so folgt die Behauptung (4.46) mit einer Konstanten C der Gestalt (4.47). Da die adaptiv gewichteten Sobolev-Normen $\|u\|_{m,p,h}$ für beliebige Gitter $\mathbf{T}_h$ stets durch die übliche Sobolev-Norm $\|u\|_{m,p,\Omega}$ beschränkt sind, ist diese Konstante für festes u gleichmäßig bezüglich h und $\widetilde{h}$ beschränkt. □

BEMERKUNG 4.18 *Auf ähnliche Weise wie in den oben angeführten Beweisen kann man zeigen, daß die Aussagen von Lemma 4.16 und Lemma 4.17 ebenso für die ursprüngliche upwind-Form* $\widetilde{n}_h$ *gelten.*

4.7 Existenz und Konvergenz diskreter Lösungen

Nach den Vorbereitungen in den vorhergehenden Abschnitten kommen wir nun zu Aussagen über die Existenz und Konvergenz von Lösungen (u_h, p_h) unserer diskreten Navier-Stokes-Probleme. Der folgende Satz garantiert auf jedem shape-regulären Gitter für eine beliebige Reynolds-Zahl $Re = \nu^{-1}$ die Lösbarkeit des nichtlinearen diskreten Problems (3.8). Sein Beweis ist analog zu dem von Theorem 1 in [57].

SATZ 4.19 *Seien die Voraussetzungen (A1),(A2) und (A3) aus Abschnitt 3.1 erfüllt und* $f \in (L^2(\Omega))^d$*. Dann existiert wenigstens eine Lösung* $(u_h, p_h) \in V_h \times Q_h$ *des diskreten Navier-Stokes-Problems (3.8).*

Ein qualitatives Konvergenzresultat für beliebige Reynolds-Zahlen, bei dem jedoch nichts über die Konvergenzgeschwindigkeit ausgesagt wird, liefert der Satz 4.20. Da die ersten Ableitungen der Lösungen u_h nur elementweise existieren, führen wir zur Beschreibung ihrer Konvergenz den Einbettungsoperator $E_h : V_h \to (L^2(\Omega))^{d+d^2}$ ein, bei dem wir die Funktion $E_h v_h(x)$ für alle inneren Punkte x der Elemente $K \in \mathbf{T}_h$ durch

$$E_h v_h(x) \;:=\; (\,v_h(x), \nabla v_h(x)\,)$$

definieren. Weiterhin benötigen wir wieder die modifizierte Gitterweite

$$\widetilde{h} \;:=\; \max_{K\in\mathbf{T}_h} \widetilde{h}_K \qquad \text{mit } \widetilde{h}_K \text{ aus (4.30).} \tag{4.48}$$

Dann können wir anlog zu Theorem 2 in [57] den folgenden Satz beweisen.

SATZ 4.20 *Seien die Voraussetzungen (A1),(A2),(A3) erfüllt und* $f \in (L^2(\Omega))^d$. *Sei weiterhin* $\{\mathbf{T}_h\}$ *eine Gitterfolge, bei der sowohl die übliche Gitterweite* h *als auch die modifizierte Gitterweite* $\widetilde{h}$ *aus (4.48) gegen Null streben, und sei* $\{(u_h, p_h)\}$ *eine zugehörige Folge von Lösungen der diskreten Probleme (3.8). Dann gibt es eine Teilfolge* $\{(u_{h'}, p_{h'})\}$ *und eine Lösung* $(u,p) \in V \times Q$ *des stetigen Problems (3.1), so daß die Folge* $\{E_{h'} u_{h'}\}$ *in* $(L^2(\Omega))^{d+d^2}$ *stark gegen* $(u, \nabla u)$ *und die Folge* $\{p_{h'}\}$ *in* $L^2(\Omega)$ *schwach gegen* p *konvergiert. Unter der Regularitätsannahme* $(u,p) \in (H^2(\Omega))^d \times H^1(\Omega)$ *konvergiert die Teilfolge* $\{p_{h'}\}$ *auch stark in* $L^2(\Omega)$ *gegen* p.

Die Forderung $\widetilde{h} \to 0$ stellt außer im Fall von P_1-parametrischen mittelwert-orientierten Elementen eine Winkelbedingung an die Gitterfolge $\{\mathbf{T}_h\}$ dar. Für blockstrukturierte Gitter, bei der die Gitterfolge durch uniforme Verfeinerung aus einer gegebenen Anfangszerlegung definiert wird, ist diese Bedingung erfüllt (siehe Abschnitt 4.8).

Unter der Voraussetzung, daß der Ausdruck $\nu^{-2}\|f\|_{0,2,\Omega}$ hinreichend klein ist, können wir die Eindeutigkeit der stetigen und diskreten Lösung garantieren und eine Abschätzung für den Diskretisierungsfehler beweisen. Durch diese Voraussetzung sind wir allerdings auf Fälle mit kleiner Reynolds-Zahl oder mit kleinen Daten beschränkt.

SATZ 4.21 *Seien die Voraussetzungen (A1),(A2) und (A3) aus Abschnitt 3.1 erfüllt,* $f \in (L^2(\Omega))^d$ *und* $\nu^{-2}\|f\|_{0,2,\Omega} \le c_0$ *mit einem hinreichend kleinen* c_0. *Dann haben das stetige und diskrete Navier-Stokes-Problem (3.1) und (3.8) jeweils eine eindeutig bestimmte Lösung* (u,p) *und* (u_h, p_h). *Unter der Regularitätsvoraussetzung* $(u,p) \in (H^2(\Omega))^d \times H^1(\Omega)$ *gilt für Raumdimensionen* $d \le 3$ *die Fehlerabschätzung*

$$\|u - u_h\|_h \; + \; \|p - p_h\|_{0,2,\Omega} \; \le \; C\,\widetilde{h}\,, \tag{4.49}$$

wobei $\widetilde{h}$ *die modifizierte Gitterweite aus (4.48) und*

$$C = C(\nu, \|f\|_{0,2,\Omega}, \|u\|_{C(\Omega)}, \|u\|_{0,4,h}, |u|_{1,4,h}, \|u\|_{2,2,\widetilde{h}}, |p|_{1,2,h}) \tag{4.50}$$

eine bezüglich h *und* $\widetilde{h}$ *gleichmäßig beschränkte Konstante ist.*

Beweis.
Setzen wir in dem diskreten Problem (3.8) $v_h = u_h \in W_h$, so erhalten wir mit Hilfe der Positivität (3.22) von $\bar{n}_h$ und der diskreten Sobolev-Ungleichung (4.21) die a-priori Abschätzung

$$\|u_h\|_h \; \le \; \frac{C}{\nu}\,\|f\|_{0,2,\Omega} \tag{4.51}$$

für jede Geschwindigkeitslösung u_h. Nehmen wir an, wir hätten zwei diskrete Lösungen (u_h^1, p_h^1) und (u_h^2, p_h^2). Setzen wir in (3.8) als Testfunktion $v_h := u_h^1 - u_h^2 \in W_h$ ein, so ergibt sich unter Verwendung von Lemma 3.1 und Lemma 4.16

$$\nu\|v_h\|_h^2 = \bar{n}_h(u_h^2, u_h^2, v_h) - \bar{n}_h(u_h^1, u_h^1, v_h) \le \bar{n}_h(u_h^2, u_h^2, v_h) - \bar{n}_h(u_h^1, u_h^2, v_h) \le C\,\|u_h^2\|_h\,\|v_h\|_h^2\,.$$

Mit der a-priori Abschätzung für u_h^2 folgt

$$(\nu - \frac{C}{\nu}\,\|f\|_{0,2,\Omega})\,\|v_h\|_h^2 \;\leq\; 0$$

und daraus im Fall $\nu^{-2}\|f\|_{0,2,\Omega} \leq c_0$ mit einem hinreichend kleinen c_0 die Abschätzung $\|v_h\|_h \leq 0$, d.h. $u_h^1 = u_h^2$. Hieraus erhalten wir

$$b_h(p_h^1 - p_h^2, v_h) = 0 \qquad \forall v_h \in V_h$$

und $p_h^1 = p_h^2$ ergibt sich unmittelbar aus der diskreten Babuška-Brezzi-Bedingung (4.38). Die Eindeutigkeit der stetigen Lösung ist ein bekanntes Resultat [22].

Unter der Regularitätsvoraussetzung $(u,p) \in (H^2(\Omega))^d \times H^1(\Omega)$ gilt

$$\nu\, a_h(u, v_h) + n_h(u,u,v_h) + b_h(p, v_h) \;=\; (f, v_h) + \nu\, E_u^{(1)}(v_h) + E_p^{(2)}(v_h) \quad \forall\, v_h \in V_h\,,$$

wobei $E_u^{(1)}(\cdot)$ und $E_p^{(2)}(\cdot)$ die Konsistenzfehlerfunktionale aus (4.41) und (4.42) sind. Mit den Abkürzungen $w_h{:=}u_h - v_h$ und $E(w_h) := \nu\, E_u^{(1)}(w_h) + E_p^{(2)}(w_h)$ erhalten wir daraus für eine beliebige Funktion $v_h{\in}W_h$ die Identität

$$\begin{aligned}
\nu\,\|w_h\|_h^2 \;&=\; \nu\, a_h(u - v_h, w_h) + n_h(u,u,w_h) - \bar{n}_h(u_h,u_h,w_h) + b_h(p - p_h, w_h) - E(w_h) \\
&=\; \nu\, a_h(u - v_h, w_h) + \{n_h(u,u,w_h) - \bar{n}_h(v_h,v_h,w_h)\} \;-\; \bar{n}_h(v_h,w_h,w_h) \;+ \\
&\qquad \{\bar{n}_h(v_h,u_h,w_h) - \bar{n}_h(u_h,u_h,w_h)\} \;+ b_h(p - p_h, w_h) - E(w_h)\,. \qquad (4.52)
\end{aligned}$$

Die Terme auf der rechten Seite können wir unter Verwendung von Lemma 3.1, Lemma 4.16 und der a-priori Abschätzung (4.51) folgendermaßen abschätzen

$$\begin{aligned}
\nu\, a_h(u - v_h, w_h) \;&\leq\; \nu\,\|u - v_h\|_h\,\|w_h\|_h \\
-\bar{n}_h(v_h,w_h,w_h) \;&\leq\; 0 \\
|\bar{n}_h(v_h,u_h,w_h) - \bar{n}_h(u_h,u_h,w_h)| \;&\leq\; \frac{C}{\nu}\,\|f\|_{0,2,\Omega}\,\|w_h\|_h^2 \;\;\leq\;\; C c_0 \nu\,\|w_h\|_h^2\,.
\end{aligned}$$

Zur Abschätzung des b_h-Termes verwenden wir die Ungleichung

$$b_h(q, v_h) \;\leq\; C\,\|q\|_{0,2,\Omega}\,\|v_h\|_h \qquad \forall\, q \in L^2(\Omega) \quad \forall\, v_h \in V_h\,,$$

die im Fall $b_h = \bar{b}_h$ (siehe (3.4)) offensichtlich ist, die man aber auch für $b_h = \widetilde{b}_h$ (siehe (3.5)) zeigen kann. Sei $J_h p \in Q_h$ die stückweise konstante Interpolierte von $p \in H^1(\Omega) \cap L_0^2(\Omega)$, die für jeden inneren Punkt x_0 eines Elementes $K{\in}\mathbf{T}_h$ durch $J_h p(x_0) := |K|^{-1} \int_K p\,dx$ definiert ist. Dann gilt wegen $w_h \in W_h$

$$b_h(p - p_h, w_h) \;=\; b_h(p - J_h p, w_h) \;\leq\; C\,h\,|p|_{1,2,h}\,\|w_h\|_h\,.$$

Für den Störungsterm E erhalten wir aus Lemma 4.14, der Beziehung $\overline{h}_K \leq \widetilde{h}_K$ und Lemma 4.15 die Abschätzung

$$|\,E(w_h)\,| \;\leq\; C\,\widetilde{h}\,(\nu\|u\|_{2,2,\widetilde{h}} + |p|_{1,2,h})\,\|w_h\|_h \qquad \forall\, w_h \in V_h\,. \qquad (4.53)$$

Wählen wir $v_h = i_h u \in W_h$ mit dem divergenz-erhaltenden Interpolationsoperator i_h aus (4.25), verwenden Lemma 4.10 und die Abschätzung (4.46), so folgt aus (4.52) für ein hinreichend kleines c_0

$$\|w_h\|_h \le C\,\widetilde{h}$$

mit einer Konstanten C der Gestalt

$$C = C\,\|u\|_{2,2,\widetilde{h}} + \frac{C}{\nu}\Big(\|u\|_{0,4,h}\,|u|_{1,4,h} + \|u\|_{C(\Omega)}\,\|u\|_{2,2,\widetilde{h}}\Big) + \frac{C}{\nu}|p|_{1,2,h}\,.$$

Daraus ergibt sich die Abschätzung für den Geschwindigkeitsfehler

$$\|u-u_h\|_h \le \|u-v_h\|_h + \|w_h\|_h \le C\,\widetilde{h}\,.$$

Um den Fehler im Druck abzuschätzen, setzen wir $q_h := p_h - J_h p$ und nutzen für eine beliebige Funktion $w_h \in V_h$ die Identität

$$b_h(q_h, w_h) = b_h(p - J_h p, w_h) + \nu\, a_h(u-u_h, w_h) + n_h(u,u,w_h) - \bar{n}_h(u_h,u_h,w_h) - E(w_h)\,.$$

Sei wieder $v_h := i_h u$. Dann können wir die einzelnen Terme folgendermaßen abschätzen

$$\begin{aligned}
b_h(p - J_h p, w_h) &\le C\,h\,|p|_{1,2,h}\,\|w_h\|_h\\
\nu\, a_h(u-u_h, w_h) &\le \nu\,\|u-u_h\|_h\,\|w_h\|_h\\
n_h(u,u,w_h) - \bar{n}_h(u_h,u_h,w_h) &= \{n_h(u,u,w_h) - \bar{n}_h(v_h,v_h,w_h)\} + \{\bar{n}_h(v_h,v_h,w_h) -\\
&\quad\ \bar{n}_h(u_h,v_h,w_h)\} + \bar{n}_h(u_h, v_h - u_h, w_h)\\
&\le C\,\widetilde{h}\,\|w_h\|_h + C\,(\,\|v_h\|_h + \|u_h\|_h\,)\,\|v_h - u_h\|_h\,\|w_h\|_h\\
&\le C\,(\widetilde{h} + \|u-v_h\|_h + \|u-u_h\|_h)\,\|w_h\|_h\,,
\end{aligned}$$

wobei wir in der letzten Abschätzung die a-priori Abschätzung (4.51) benutzt haben. Verwenden wir (4.31), (4.53) und die bereits bewiesene Abschätzung für $\|u-u_h\|_h$, so folgt aus der diskreten Babuška-Brezzi-Bedingung (4.38)

$$\|q_h\|_{0,2,\Omega} \le \frac{1}{\beta}\,C(\nu,f,u,p)\,\widetilde{h}\,.$$

Daraus ergibt sich schließlich

$$\|p-p_h\|_{0,2,\Omega} \le \|p - J_h p\|_{0,2,\Omega} + \|q_h\|_{0,2,\Omega} \le C\,\widetilde{h}\,.$$

mit einer Konstanten C, die wie in (4.50) angegeben von ν,f,u und p abhängt. □

Im Fall von P_1-parametrischen mittelwert-orientierten Elementen gilt $\widetilde{h}_K = h_K$ und die Fehlerabschätzung (4.49) ist optimal. Für die anderen Finiten Elemente ist $\widetilde{h}_K = h_K + \sigma_K$ und wir erhalten eine optimale Fehlerabschätzung nur auf Gittern, die der Forderung $\sigma_K \le C h_K$ für alle $K \in \mathbf{T}_h$ genügen. Im folgenden Abschnitt werden wir zeigen, daß diese Forderung für blockstrukturierte Gitter erfüllt ist.

Dadurch, daß die Fehlerkonstante C in (4.49) von den adaptiv gewichteten Normen der exakten Lösung (u,p) abhängt, kann sie auch dann noch eine vernünftige Größenordnung haben, wenn die entsprechenden üblichen Sobolev-Normen von u und p groß sind. Voraussetzung ist natürlich, daß das Gitter an den Stellen geeignet adaptiv verfeinert ist, wo die lokalen Sobolev-Normen der exakten Lösung groß sind.

Aus der Fehlerabschätzung (4.49) erhalten wir auch eine L^2-Abschätzung des Geschwindigkeitsfehlers. Mit Hilfe der diskreten Sobolev-Ungleichung (4.21) und den Eigenschaften des Interpolationsoperators i_h (siehe Lemma 4.10) ergibt sich

$$\|i_h u - u_h\|_{0,2,\Omega} \le C\,\|i_h u - u_h\|_h \le C\left(\|i_h u - u\|_h + \|u - u_h\|_h\right) \le C\,\widetilde{h}$$

und daraus über die Dreiecksungleichung

$$\|u - u_h\|_{0,2,\Omega} \le \|u - i_h u\|_{0,2,\Omega} + \|i_h u - u_h\|_{0,2,\Omega} \le Ch\widetilde{h}\,\|u\|_{2,2,\widetilde{h}} + C\widetilde{h} \le C\,\widetilde{h}\,.$$

Diese Abschätzung ist natürlich nicht optimal. In [65] wurde jedoch durch ein eindimensionales Beispiel gezeigt, daß für die simple-upwind-Diskretisierung der Geschwindigkeitsfehler in der L^2-Norm im allgemeinen nur von erster Ordnung ist.

BEMERKUNG 4.22 *Verwenden wir anstelle von $\bar{n}_h$ die ursprüngliche upwind-Diskretisierung $\widetilde{n}_h$ aus (3.16), so lassen sich mit Hilfe der Aussagen in Bemerkung 3.2 und Bemerkung 4.18 folgende Resultate gewinnen. Für jede Reynolds-Zahl $Re = \nu^{-1}$ gibt es ein $h_0 = h_0(\nu, \Omega, f)$, so daß für alle $h \le h_0$ das diskrete Navier-Stokes-Problem (3.8) eine Lösung (u_h, p_h) hat, die der Stabilitätsabschätzung*

$$\|u_h\|_h \le \frac{C^{(s)}}{\nu}\,\|f\|_{0,2,\Omega} \tag{4.54}$$

genügt. Unter der Voraussetzung $\nu^{-2}\|f\|_{0,2,\Omega} \le c_0$ mit einem hinreichend kleinen c_0 gilt für jede diskrete Lösung (u_h, p_h), die der Stabilitätsabschätzung (4.54) genügt, die Fehlerabschätzung (4.49). Beide upwind-Diskretisierungen haben also dieselbe Fehlerordnung.

4.8 Vergleich der verschiedenen Finite Elemente Räume

In den vorhergehenden Abschnitten haben wir vier verschiedene Räume V_h zur Approximation der Geschwindigkeit eingeführt und ihre Eigenschaften bei der numerischen Lösung der Navier-Stokes-Gleichungen analysiert. Es entsteht die Frage nach einer Bewertung dieser unterschiedlichen Finiten Elemente. Im folgenden wollen wir ihre Vor- und Nachteile gegeneinander abwägen.

Vergleichen wir zunächst die Q_1-parametrischen und die P_1-parametrischen Elemente. Die Q_1-parametrischen Elemente haben den Vorteil, daß für alle Elemente $K \in \mathbf{T}_h$ das Referenzelement $\widehat{K} = [-1,+1]^d$ fest ist. Damit brauchen (mit Ausnahme der mittelwert-orientierten Elemente im 3D-Fall) die Basisfunktionen $\widehat{\phi}_j$ nur einmal berechnet werden und wir haben für $\widehat{K}$ feste Stützstellen (Gauß-Punkte) und Gewichte bei der numerischen Integration der Elementsteifigkeitsmatrizen. Die Jacobi-Matrix der Q_1-parametrischen Referenzabbildung muß allerdings für jede Stützstelle neu berechnet werden, da sie nicht konstant ist. Der entscheidende Nachteil der Q_1-parametrischen Elemente ist, daß der Fehler der Finite-Elemente-Interpolation von den Winkelgrößen σ_K der Elemente $K\in\mathbf{T}_h$ abhängt (siehe Lemma 2.14). Dies ist bei den P_1-parametrischen Elementen nicht der Fall. Der Preis für diesen Vorteil ist jedoch, daß sich das Referenzelement $\widehat{K}$ mit jedem Element K ändert. Damit müssen die Stützstellen und Gewichte für die numerische Integration jedesmal neu ermittelt werden. Außerdem müssen bei den mittelwert-orientierten P_1-parametrischen Elementen die Basisfunktionen $\widehat{\phi}_j$ jeweils neu durch das Lösen eines $(2d \times 2d)$-Gleichungssystems (siehe (2.16)) bestimmt werden. Im 2D-Fall kann dies alles noch recht effektiv implementiert werden, da alle Größen auf $\widehat{K}$ nur von einem Störungsvektor $\hat{\varepsilon} \in \mathbf{R}^2$ (siehe Abschnitt 2.2) abhängen. Im 3D-Fall ergibt sich jedoch ein recht hoher Rechenaufwand. Ein kleiner Vorteil der P_1-parametrischen Referenzabbildung besteht darin, daß ihre Jacobi-Matrix konstant ist und daher für alle Stützstellen auf $\widehat{K}$ nur einmal berechnet werden braucht. Generell muß aber gesagt werden, daß die P_1-parametrischen Elemente bei der Generierung der Steifigkeitsmatrix und des rechte-Seite-Vektors höhere Rechenzeitkosten verursachen als die Q_1-parametrischen Elemente.

Vergleichen wir die punktwert-orientierten mit den mittelwert-orientierten Elementen, so haben die punktwert-orientierten Elemente den Vorteil, daß die Basisfunktionen $\widehat{\phi}_j$ auf dem Referenzelement für alle Elemente $K\in\mathbf{T}_h$ feste Funktionen sind, während sie bei den mittelwert-orientierten Elementen (mit Ausnahme der Q_1-parametrischen Elemente im 2D-Fall) für jedes Element K jeweils neu durch das Lösen eines Gleichungssystems bestimmt werden müssen. Andererseits haben punktwert-orientierte Elemente den Nachteil, daß der Konsistenzfehler bei der Diskretisierung des Laplace-Operators von den Winkelgrößen σ_K der Elemente $K\in\mathbf{T}_h$ abhängt (siehe Lemma 4.14).

Aus Satz 4.21 entnehmen wir, daß die P_1-parametrischen mittelwert-orientierten Elemente die einzigen sind, bei denen der Diskretisierungsfehler unabhängig von den Winkelgrößen σ_K ist. Bei den anderen Finiten Elementen hängt er von den modifizierten Gitterweiten $\widetilde{h}_K = h_K + \sigma_K$ ab. In einem beliebigen Gitter gibt es im allgemeinen stets Elemente K, bei denen Winkel zwischen gegenüberliegenden Kanten in der Größenordnung $O(1)$ liegen, so daß $\sigma_K = O(1)$ gilt. Damit ist auch $\widetilde{h} = \max_{K\in\mathbf{T}_h} \widetilde{h}_K = O(1)$, obwohl die übliche Gitterweite $h = \max_{K\in\mathbf{T}_h} h_K$ sehr klein sein kann. Wir können also in diesem Fall keine sinnvolle Fehlerabschätzung mehr garantieren. Diese Situation tritt insbesondere bei adaptiven Verfeinerungsstrategien ein, wo im Prozeß der Gitterverfeinerung durch bestimmte Abschlußregeln neue Elemente mit $O(1)$-Winkeln zwischen gegenüberliegenden Kanten entstehen. Die Schlußfolgerung ist, daß für adaptive Gitterverfeinerung nur die P_1-parametrischen mittelwert-orientierten Elemente sinnvoll sind. Wenn das numerische Verfahren auch für die anderen Finiten Elemente effektiv sein soll, so benötigt man Gitter, bei denen die Winkelgrößen σ_K in der Größenordnung der Elementdurchmesser h_K liegen. Wir werden zeigen, daß diese Eigenschaft für bestimmte blockstrukturierte Gitter erfüllt ist.

Bei der Definition eines blockstrukturierten Gitters beginnen wir mit einer Zerlegung $T^{(0)}$ des Gebietes Ω in sogenannte *Makroelemente* $K_0 \in T^{(0)}$. Durch ℓ uniforme Verfeinerungsschritte ergeben sich hieraus rekursiv die Zerlegungen $T^{(1)}, \dots, T^{(\ell)}$, wobei die Zerlegung $T^{(k)}$ dadurch entsteht, daß wir jedes Element $K \in T^{(k-1)}$ durch Kantenhalbierung in 2^d Sohnelemente zerlegen. Unser blockstrukturiertes Gitter ist dann die Zerlegung $\mathbf{T}_h = T^{(\ell)}$.

In dem folgenden Lemma charakterisieren wir das Verhalten der Größen σ_K und h_K für alle Elemente $K \in T^{(\ell)}$, die zu einem Makroelement $K_0 \in T^{(0)}$ gehören.

LEMMA 4.23 *Sei $K_0 \in T^{(0)}$ ein beliebiges Makroelement einer shape-regulären Zerlegung $T^{(0)}$ und $K \subset K_0$ mit $K \in T^{(\ell)}$ ein beliebiges Sohnelement von K_0 auf dem Gitterlevel ℓ. Sei weiterhin $\sigma_0 := \sigma_{K_0}$ und $h_0 := h_{K_0}$. Dann gelten die Abschätzungen*

$$C_1\, 2^{-\ell} h_0 \;\le\; h_K \;\le\; C_2\; 2^{-\ell} h_0\,, \tag{4.55}$$

$$C_3\, 2^{-\ell} \sigma_0 \;\le\; \sigma_K \;\le\; C_4\; 2^{-\ell} \sigma_0\,, \tag{4.56}$$

mit Konstanten C_i, die jeweils unabhängig von K_0, K und ℓ sind.

Beweis.
Wir beginnen mit dem Beweis von (4.55). Sei $F : \widehat{K}_0 \to K_0$ mit $F \in (Q_1(\widehat{K}_0))^d$ die Q_1-parametrische Referenzabbildung zwischen dem Makroelement K_0 und dem Referenzelement $\widehat{K}_0 = [-1,+1]^d$. Die Gitterweite $h_K = diam(K)$ wird als Abstand zweier Ecken x und y des Elementes K angenommen. Dann sind die zugehörigen Referenzpunkte $\hat{x} = F^{-1}(x)$ und $\hat{y} = F^{-1}(y)$ auch Ecken im Referenzelement $\widehat{K} = F^{-1}(K)$. Auf Grund der Verfeinerungsstrategie ist $\widehat{K}$ ein d-dimensionaler Würfel mit der Kantenlänge $2^{-\ell+1}$ und es folgt

$$\frac{1}{2}\, 2^{-\ell} \;\le\; |\hat{x} - \hat{y}| \;\le\; \frac{1}{2}\sqrt{d}\, 2^{-\ell}\,.$$

Da die Makroelementzerlegung $T^{(0)}$ als shape-regulär vorausgesetzt wurde, gelten die Abschätzungen

$$|F|_{1,\infty,\widehat{K}_0} \;\le\; C\, h_0 \qquad \text{und} \qquad |F^{-1}|_{1,\infty,K_0} \;\le\; C\, h_0^{-1} \tag{4.57}$$

mit Konstanten C, die unabhängig von K_0 sind. Damit können wir h_K abschätzen durch

$$h_K \;=\; |\, F(\hat{x}) - F(\hat{y})\,| \;\le\; C\, |F|_{1,\infty,\widehat{K}_0}\, |\hat{x} - \hat{y}| \;\le\; C\, 2^{-\ell}\, h_0\,.$$

Andererseits gilt

$$|\hat{x} - \hat{y}| \;=\; |\, F^{-1}(x) - F^{-1}(y)\,| \;\le\; C\, |F^{-1}|_{1,\infty,K_0}\, h_K\,,$$

woraus sich

$$h_K \;\ge\; C^{-1} |F^{-1}|^{-1}_{1,\infty,K_0}\, |\hat{x} - \hat{y}| \;\ge\; C\, 2^{-\ell}\, h_0$$

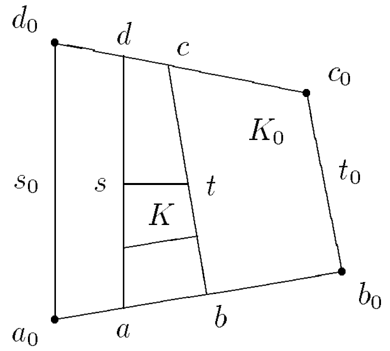


Abbildung 4.3: *Makroelement* K_0 *und Sohnelement* K *auf dem Gitterlevel* ℓ

ergibt.

Die Aussage (4.56) beweisen wir hier nur für den 2D-Fall. Den Beweis im 3D-Fall kann man auf die zweidimensionale Aussage zurückführen. Seien a_0, b_0, c_0, d_0 die Ecken des Makroelementes K_0 und $\overline{ad}$ und $\overline{bc}$ zwei benachbarte Gitterlinien der Zerlegung $T^{(\ell)}$, die gegenüberliegende Kanten des Elementes $K \in T^{(\ell)}$ enthalten (siehe Abbildung 4.3). Da die Gitterlinien von $T^{(\ell)}$ die Kanten von K_0 gleichmäßig teilen, gilt die Darstellung

$$\begin{aligned} a &= a_0 + i\,2^{-\ell}(b_0 - a_0)\,, & b &= a_0 + (i+1)2^{-\ell}(b_0 - a_0)\,, \\ d &= d_0 + i\,2^{-\ell}(c_0 - d_0)\,, & c &= d_0 + (i+1)2^{-\ell}(c_0 - d_0)\,, \end{aligned}$$

mit $i \in \{0, 1, \ldots, 2^\ell - 1\}$. Um die Winkel zwischen gegenüberliegenden Kanten von K_0 und K analysieren zu können, führen wir die Vektoren

$$s_0 := a_0 - d_0\,, \quad t_0 := b_0 - c_0\,, \qquad s := a - d\,, \quad t := b - c\,,$$

ein und definieren die Winkel $\alpha_0 := \angle(s_0, t_0)$ sowie $\alpha := \angle(s, t)$. Dann gilt die Beziehung

$$s \times t \;=\; 2^{-\ell}\,(s_0 \times t_0)\,,$$

wobei wir das Kreuzprodukt $v_1 \times v_2$ zweier 2D-Vektoren $v_k = (v_{k1}, v_{k2})$ als das übliche Kreuzprodukt der 3D-Vektoren $\bar{v}_k = (v_{k1}, v_{k2}, 0)$ definieren. Ohne Beschränkung der Allgemeinheit können wir $\alpha_0, \alpha \in [0, \pi]$ annehmen und erhalten

$$\sin\alpha \;=\; \frac{|s \times t|}{|s|\;|t|} \;=\; 2^{-\ell}\,\sin\alpha_0\,\frac{|s_0|}{|s|}\,\frac{|t_0|}{|t|}\,. \tag{4.58}$$

Man kann zeigen, daß mit $T^{(0)}$ auch die Zerlegung $T^{(\ell)}$ shape-regulär im Sinne von Definition 2.2 ist. Daraus erhalten wir $\alpha_0, \alpha \in [0, \pi - 2\delta_0]$, wobei $\delta_0 > 0$ die Konstante aus der Bedingung (2.5) für die shape-Regularität ist. Verwenden wir die Ungleichung

$$\sin z \;\le\; z \;\le\; C(\delta_0)\,\sin z \qquad \forall\, z \in [0, \pi - 2\delta_0]\,,$$

so ergibt sich aus (4.58)

$$\underline{C}\, 2^{-\ell}\, \alpha_0\, \frac{|s_0|}{|s|}\, \frac{|t_0|}{|t|} \;\leq\; \alpha \;\leq\; \overline{C}\, 2^{-\ell}\, \alpha_0\, \frac{|s_0|}{|s|}\, \frac{|t_0|}{|t|}\,. \tag{4.59}$$

Um den Term $|s_0|/|s|$ abzuschätzen, verwenden wir wieder die Q_1-parametrische Referenzabbildung $F : \widehat{K}_0 \to K_0$. Für die Referenzpunkte $\widehat{a} := F^{-1}(a)$ und $\widehat{b} := F^{-1}(b)$ gilt $\widehat{a} - \widehat{b} = -2\widehat{e}_2$ mit dem Einheitsvektor $\widehat{e}_2 = (0,1)$. Daraus erhalten wir mit (4.57)

$$2 \;=\; |\widehat{a} - \widehat{b}| \;=\; |\, F^{-1}(a) - F^{-1}(b)\,| \;\leq\; C\, |F^{-1}|_{1,\infty,K_0}\, |s| \;\leq\; C\, h_0^{-1} |s|\,.$$

Folglich gibt es eine von K_0 und K unabhängige Konstante C_0, so daß gilt

$$C_0\, h_0 \;\leq\; |s| \;\leq\; h_0\,.$$

Analog können wir diese Ungleichung auch für die Vektoren s_0, t und t_0 zeigen und wir erhalten die Abschätzung

$$C_0^2 \;\leq\; \frac{|s_0|}{|s|}\, \frac{|t_0|}{|t|} \;\leq\; C_0^{-2}\,.$$

Aus (4.59) folgt damit, daß es zwei von K_0, K und ℓ unabhängige Konstanten C_3 und C_4 gibt, so daß gilt

$$C_3\, 2^{-\ell}\, \alpha_0 \;\leq\; \alpha \;\leq\; C_4\, 2^{-\ell}\, \alpha_0\,.$$

Diese Ungleichung können wir analog auch für das andere Paar von Winkeln zwischen gegenüberliegenden Kanten von K_0 und K zeigen und es ergibt sich die Behauptung (4.56). □

Aus dem vorhergehenden Lemma erhalten wir unmittelbar die Abschätzung

$$C_5\, \frac{\sigma_{K_0}}{h_{K_0}} \;\leq\; \frac{\sigma_K}{h_K} \;\leq\; C_6\, \frac{\sigma_{K_0}}{h_{K_0}} \qquad \forall\, K \subset K_0 \in T^{(0)} \tag{4.60}$$

mit $C_5 = C_3/C_2$ und $C_6 = C_4/C_1$. Dies bedeutet, daß auf jedem Gitterlevel ℓ das Verhältnis σ_K/h_K eines Elementes $K \in T^{(\ell)}$ stets in der Größenordnung des Verhältnisses σ_{K_0}/h_{K_0} des zugeordneten Makroelementes K_0 liegt. Zur Makroelementzerlegung $T^{(0)}$ definieren wir die Größe

$$\mu_0 \;:=\; \max_{K_0 \in T^{(0)}} \frac{\sigma_{K_0}}{h_{K_0}}\,. \tag{4.61}$$

Dann erhalten wir für die modifizierte Gitterweite $\widetilde{h}_K = h_K + \sigma_K$ auf allen aus $T^{(0)}$ erzeugten blockstrukturierten Gittern $\mathbf{T}_h = T^{(\ell)}$ die Abschätzung

$$\widetilde{h}_K \;\leq\; (1 + C_6\mu_0)\, h_K \qquad\qquad \forall\, K \in \mathbf{T}_h\,. \tag{4.62}$$

Für die adaptiv gewichtete Sobolev-Norm $\|\cdot\|_{m,q,\widetilde{h}}$ mit $q \in [1,\infty)$ ergibt sich daraus wegen $\widetilde{h} = \max_{K\in\mathbf{T}_h} \widetilde{h}_K \geq h$

$$\|u\|_{m,q,\widetilde{h}} \;=\; \Big(\sum_{K\in\mathbf{T}_h} \Big(\frac{\widetilde{h}_K}{\widetilde{h}}\Big)^2 \|u\|_{m,q,K}^q \Big)^{1/q} \;\leq\; (1 + C_6\mu_0)^{2/q}\, \|u\|_{m,q,h}\,. \tag{4.63}$$

Die analoge Abschätzung gilt auch für die Halbnorm $|\cdot|_{m,q,\widetilde{h}}$. Aus Satz 4.21 können wir somit folgende Schlußfolgerung ziehen.

SATZ 4.24 *Sei $\mathbf{T}_h = T^{(\ell)}$ ein blockstrukturiertes Gitter, welches durch ℓ uniforme Verfeinerungsschritte aus einer shape-regulären Makroelementzerlegung $T^{(0)}$ entstanden ist, und sei μ_0 die in (4.61) definierte Gitterkonstante zu $T^{(0)}$. Dann gilt für jeden der vier Finite-Elemente-Räume V_h aus Abschnitt 2.2 unter den Voraussetzungen von Satz 4.21 die optimale Fehlerabschätzung*

$$\|u - u_h\|_h \ + \ \|p - p_h\|_{0,2,\Omega} \ \leq \ C\,h$$

mit einer Konstanten

$$C = C(\mu_0, \nu, \|f\|_{0,2,\Omega}, \|u\|_{C(\Omega)}, \|u\|_{0,4,h}, |u|_{1,4,h}, \|u\|_{2,2,h}, |p|_{1,2,h})\,,$$

die gleichmäßig bezüglich h beschränkt ist.

Weiterhin erkennen wir aus Satz 4.21 und Lemma 4.23, daß außer im Fall von P_1-parametrischen mittelwert-orientierten Elementen die Verwendung sehr kleiner Makroelemente K_0 zum Zwecke der Gitteradaption nur dann Sinn macht, wenn gleichzeitig auch die Winkelgröße σ_{K_0} sehr klein ist. Dies bedeutet, daß die Makroelemente in den fein vernetzten Teilgebieten nahezu Parallelogramme bzw. Parallelepipede sein müßten. Aus praktischer Sicht ist das jedoch eine zu einschneidende Forderung an die Gittergenerierung.

Zusammenfassend wollen wir folgendes festhalten.

- Verwendet man blockstrukturierte Gitter $\mathbf{T}_h = T^{(\ell)}$, die aus einer Makroelementzerlegung $T^{(0)}$ hervorgegangen sind, deren Gitterkonstante μ_0 (siehe (4.61)) nicht allzu groß ist, d.h. bei der sehr kleine Makroelemente $K_0 \in T^{(0)}$ nahezu die Form eines d-dimensionalen Parallelepipeds haben, so liefern alle vier verschiedenen Finite-Elemente-Approximationen dieselbe Fehlerordnung. Man kann in diesem Fall Finite Elemente wählen, die hinsichtlich des Rechenaufwandes günstig sind, also etwa die Q_1-parametrischen punktwert-orientierten Elemente.

- Realisiert man adaptive Verfeinerung mit Hilfe von hängenden Knoten, so ist jedes Element K Sohnelement eines gewissen Makroelementes K_0 auf einem bestimmten Level. Damit gilt die Abschätzung $\sigma_K \leq Ch_K$ für alle $K \in \mathbf{T}_h$, so daß wieder alle vier verschiedenen Elemente dieselbe Fehlerordnung liefern und man sich die vom Aufwand her günstigste Variante auswählen kann.

- Im allgemeinen Fall adaptiv verfeinerter Gitter sind die P_1-parametrischen mittelwert-orientierten Elemente die einzigen, die eine sinnvolle Fehlerabschätzung garantieren.

Kapitel 5

Abhängigkeit des Diskretisierungsfehlers von der Reynolds-Zahl

Aus praktischer Sicht hat unsere optimale Fehlerabschätzung in Satz 4.21 bzw. Satz 4.24 leider einen Nachteil. Es wird die Voraussetzung $\nu^{-2}\|f\|_{0,2,\Omega} \leq c_0$ mit einem hinreichend kleinen c_0 benötigt (c_0 kann aus der Fehleranalysis heraus quantifiziert werden). Dies bedeutet, daß die für die Praxis interessanten Fälle mit $O(1)$-Daten und hoher Reynolds-Zahl $Re{=}1/\nu$ nicht abgedeckt werden. Unsere Fehlerabschätzung ist bei hoher Reynolds-Zahl nur im Fall kleiner Daten anwendbar. Für die Fälle, in denen die Voraussetzung $\nu^{-2}\|f\|_{0,2,\Omega} \leq c_0$ nicht erfüllt ist (Nichteindeutigkeit der Lösung), könnte man versuchen, mit Hilfe der Theorie nichtsingulärer Lösungszweige (siehe [21],[22], [66]) zu Fehlerabschätzungen zu kommen. Selbst wenn dies gelingen würde, so würde die entsprechende Abschätzung erst für hinreichend kleine Gitterweiten $h \leq h_0(\nu)$ gelten mit $h_0(\nu) \to 0$ für $\nu \to 0$.

Es entsteht also die Frage, wie sich für praktisch relevante Gitterweiten h (für die im allgemeinen $h > h_0(\nu)$ gilt) der Diskretisierungsfehler im Falle hoher Reynolds-Zahlen verhält. In der Literatur gibt es hierzu kaum theoretische Aussagen. Für die spezielle Situation einer nahezu parallelen Rohrströmung haben Johnson und Rannacher [39] a posteriori und a priori Fehlerabschätzungen gezeigt, bei denen die Schranke für den Fehler in der L^2-Norm proportional zu Re^2 ist.

Andererseits hat unsere upwind-Diskretisierung in zahlreichen praktischen Testbeispielen auch für hohe Reynolds-Zahlen physikalisch sinnvolle Lösungen produziert, so daß es unwahrscheinlich erscheint, daß hier der L^2-Fehler von der Größenordnung CRe^2h^2 ist. Fast alle praktischen Testbeispiele haben allerdings den Nachteil, daß die exakte Lösung unbekannt ist und somit die Fehlernormen nicht bestimmt werden können. Es gibt zwar in der Literatur ausgewiesene Benchmark-Probleme, jedoch sind für die entsprechenden "Referenzlösungen" lediglich gewisse abgeleitete Kenngrößen dokumentiert.

Um überhaupt erste Aussagen über die Abhängigkeit des Diskretisierungsfehlers von der Reynolds-Zahl zu gewinnen, betrachten wir im folgenden Abschnitt ein akademisches Testbeispiel, bei dem wir eine kontinuierliche Lösung (u, p) vorgeben und dann die zugehörige rechte Seite f bestimmen. Nachdem die diskrete Lösung auf dem Computer berechnet wurde, können die Fehlernormen exakt ermittelt werden. Da das Testbeispiel die Voraussetzungen für unsere Theorie erfüllt, müßte sich für hinreichend kleine Reynolds-Zahlen die optimale Fehlerordnung in der H^1-Norm der Geschwindigkeit sowie in der L^2-Norm des Druckes zeigen.

Wir werden sehen, daß sich für unser Beispiel der Diskretisierungsfehler der Geschwindigkeit in einem bestimmten Re-Bereich wie $O(Re)$ verhält. Es ist überraschend, daß dies passieren kann, obwohl die kontinuierliche Lösung beliebig glatt ist, keine Grenzschichten besitzt und unabhängig von Re ist. Dieses einfache Beispiel zeigt, daß es für hohe Reynolds-Zahlen nicht nur eine Lücke in der theoretischen Begründung des Verfahrens gibt, sondern daß in diesem Fall der Diskretisierungsfehler tatsächlich sehr groß werden kann. Dies bedeutet, daß wir in praktischen Rechnungen mit $Re \gg 1$ die Genauigkeit der berechneten Lösung stets kritisch zu prüfen haben. In jüngster Zeit gibt es verstärkte Forschungsanstrengungen, um mit Hilfe von adaptiven Methoden und a posteriori Fehlerschätzern zu diskreten Lösungen zu kommen, bei denen der Diskretisierungsfehler innerhalb einer vorgegebenen Toleranz liegt.

Die Situation des $O(Re)$-Fehlerwachstums tritt nicht nur in unserem akademischen Testbeispiel auf, sondern z.B. auch für sogenannte "no-flow "-Probleme [15], wo in einer "beruhigten" Strömung ($u \approx 0$) lediglich durch das Vorhandensein einer Volumenkraft (z.B. Gravitation) ein Druckgradient entsteht. Ähnliche Effekte zeigen sich auch bei der Boussinesq-Approximation der nichtisothermen Navier-Stokes Gleichungen [16].

In Abschnitt 5.2 wollen wir versuchen, Ursachen für das beobachtete $O(Re)$-Fehlerwachstum herauszufinden. Dabei betrachten wir zunächst den einfacheren Fall des Stokes-Modells. Es zeigt sich, daß der Interpolationsfehler des Druckes mit dem Verstärkungsfaktor Re in den Diskretisierungsfehler eingeht. Mit der Idee der Druckabspaltung geben wir eine Möglichkeit an, wie man in diesem Fall eine wesentliche Verbesserung erreichen kann. Diese Technik läßt sich mit einer leichten Modifikation auch auf den Navier-Stokes-Fall übertragen. Am Beispiel einer Kanalströmung zeigen wir aber auch, daß es neben dem Einfluß des Druckes noch andere Ursachen für das Anwachsen des Diskretisierungsfehlers für $\nu \to 0$ gibt, die in weiteren Forschungsarbeiten zu klären sind.

5.1 Ein Testproblem

Wir betrachten das folgende 2D Navier–Stokes Problem auf dem Einheitsquadrat $\Omega = (0,1)^2$ mit der vorgegebenen kontinuierlichen Lösung

$$u = \begin{pmatrix} \psi_y \\ -\psi_x \end{pmatrix}, \qquad \psi(x,y) = x^2(x-1)^2y^2(y-1)^2, \qquad p(x,y) = x^3 + y^3 - 0.5,$$

und der zugehörigen rechten Seite $f := -\nu\triangle u + (u \cdot \nabla)u + \nabla p$. Bei diesem Beispiel sind alle Voraussetzungen für unsere Theorie in Bezug auf das Gebiet und die Glätte der Daten erfüllt. Die kontinuierliche Lösung ist beliebig glatt und hängt nicht von $Re = 1/\nu$ ab. Um auch im Fall höherer Reynolds-Zahlen die Berechnung der Diskretisierungsfehler $u - u_h$ und $p - p_h$ abzusichern, nehmen wir als Hypothese an, daß das kontinuierliche und diskrete Problem jeweils nur eine Lösung besitzen.

Das Gebiet Ω wird äquidistant in quadratische Elemente mit der Kantenlänge h zerlegt. Damit sind in diesem Fall die Q_1- und P_1-parametrischen Finiten Elemente identisch, außerdem gilt $\sigma_h = 0$, so daß für alle vier Finite Elemente Approximationen aus Abschnitt 2.2 die Fehlerabschätzung in Satz 4.21 von optimaler Ordnung ist. In Tabelle 5.1 geben wir zur ursprünglichen "einfachen" upwind-Version $\tilde{n}_h$ (Φ=Φ_1 aus (3.15)) für die punktwert- und mitelwert-orientierten Elemente die Normen der Diskretisierungsfehler in Abhängigkeit von h und Re an. Sowohl die modifizierte upwind-Diskretisierung $\bar{n}_h$ als auch das Samarskij-upwind liefern für dieses Beispiel nahezu dieselben Werte, so daß sie hier weggelassen werden. Abbildung 5.1 zeigt für die punktwert-orientierten Elemente

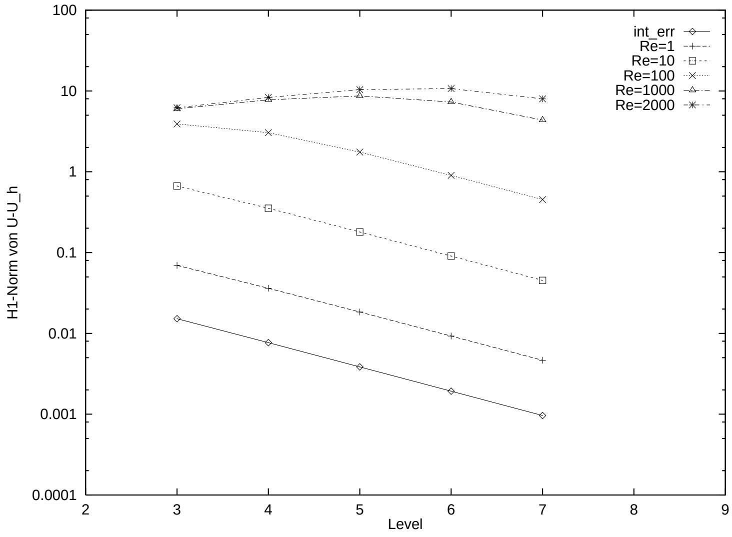
Tabelle 5.1: *Normen des Diskretisierungsfehlers für die punktwert-orientierten (*"pw-or"*) und mittelwert-orientierten (*"mw-or"*) Finiten Elemente*

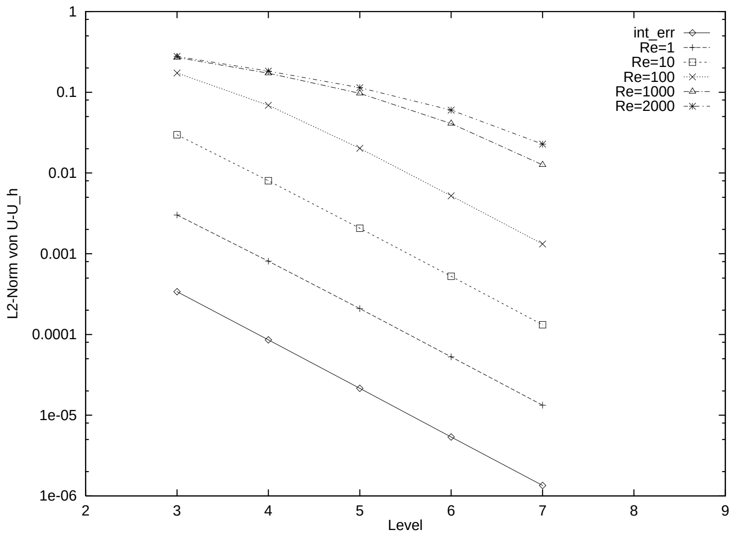
Re	h	$\|\|u-u_h\|\|_h$		$\|\|u-u_h\|\|_{0,2,\Omega}$		$\|\|p-p_h\|\|_{0,2,\Omega}$	
		pw-or	mw-or	pw-or	mw-or	pw-or	mw-or
0.2	1/16	.11 E−01	.90 E−02	.18 E−03	.12 E−03	.36 E−01	.35 E−01
	1/32	.54 E−02	.45 E−02	.47 E−04	.30 E−04	.18 E−01	.17 E−01
	1/64	.27 E−02	.23 E−02	.12 E−04	.76 E−05	.86 E−02	.86 E−02
	1/128	.14 E−02	.11 E−02	.29 E−05	.19 E−05	.43 E−02	.43 E−02
2	1/16	.71 E−01	.48 E−01	.16 E−02	.79 E−03	.36 E−01	.35 E−01
	1/32	.36 E−01	.24 E−01	.41 E−03	.20 E−03	.17 E−01	.17 E−01
	1/64	.18 E−01	.12 E−01	.10 E−03	.51 E−04	.86 E−02	.86 E−02
	1/128	.91 E−02	.61 E−02	.26 E−04	.13 E−04	.43 E−02	.43 E−02
20	1/16	.70 E+00	.48 E+00	.16 E−01	.79 E−02	.36 E−01	.35 E−01
	1/32	.36 E+00	.24 E+00	.41 E−02	.20 E−02	.17 E−01	.17 E−01
	1/64	.18 E+00	.12 E+00	.10 E−02	.50 E−03	.86 E−02	.86 E−02
	1/128	.91 E−01	.60 E−01	.26 E−03	.13 E−03	.43 E−02	.43 E−02
200	1/16	.48 E+01	.42 E+01	.11 E+00	.69 E−01	.36 E−01	.35 E−01
	1/32	.33 E+01	.24 E+01	.38 E−01	.20 E−01	.17 E−01	.17 E−01
	1/64	.18 E+01	.12 E+01	.10 E−01	.50 E−02	.86 E−02	.86 E−02
	1/128	.90 E+00	.60 E+00	.26 E−02	.13 E−02	.43 E−02	.43 E−02
2000	1/16	.83 E+01	.11 E+02	.18 E+00	.18 E+00	.36 E−01	.36 E−01
	1/32	.10 E+02	.13 E+02	.11 E+00	.99 E−01	.18 E−01	.18 E−01
	1/64	.11 E+02	.10 E+02	.60 E−01	.41 E−01	.88 E−02	.86 E−02
	1/128	.80 E+01	.59 E+01	.23 E−01	.12 E−01	.43 E−02	.43 E−02

zu verschiedenen Reynolds-Zahlen $Re \in \{1, 10, 100, 1000, 2000\}$ die Reduktion der Fehlernormen, wenn wir das Gitterlevel ℓ erhöhen, d.h. die Gitterweite $h = 2^{-\ell}$ schrittweise halbieren. Dabei stellt die durch "int-err" markierte Linie die Norm des entsprechenden Interpolationsfehlers dar (z.B. für die Geschwindigkeit $\|u - I_h u\|$, wobei $I_h u \in V_h$ die Finite Elemente Interpolierte zu u ist).

Wir sehen, daß für kleine und mittlere Reynolds-Zahlen der Geschwindigkeitsfehler in der diskreten H^1-Norm von der Ordnung $O(h)$ und in der L^2-Norm von der Ordnung $O(h^2)$ ist und daß ferner der Druckfehler in der L^2-Norm die Ordnung $O(h)$ hat. Damit werden unsere theoretischen Aussagen hinsichtlich der optimalen Fehlerordnung für hinreichend kleine Reynolds-Zahlen bestätigt. Allerdings verhält sich für hohe Reynolds-Zahlen ($Re \geq 1000$) der Geschwindigkeitsfehler in dem von uns betrachteten h-Bereich wie $O(1)$ in der diskreten H^1-Norm und wie $O(h)$ in der L^2-Norm. Die optimalen Ordnungen scheinen sich hier erst für sehr kleine h einzustellen. Gegenüber den punktwert-orientierten Elementen liefern die mittelwert-orientierten eine höhere Genauigkeit in der Geschwindigkeitsapproximation.

Betrachten wir die Diskretisierungsfehler bei festgehaltener Gitterweite h und wachsender Reynolds-Zahl Re, so erkennen wir zunächst, daß der Druckfehler unabhängig von Re ist. Für die Geschwindigkeit ist das Fehlerverhalten für die diskrete H^1-Norm in Abbildung 5.2 und für die L^2-Norm in Abbildung 5.3 dargestellt. Beide Fehlernormen verhalten sich in einem bestimmten Re-Bereich wie $O(Re)$. In Abbildung 5.4 ist für die feste Gitterweite $h = 1/64$ dargestellt, wie stark der Einfluß der Geschwindigkeitsfehler für wachsende Reynolds-Zahlen auf die Stromlinien der berechneten diskreten Lösung ist.





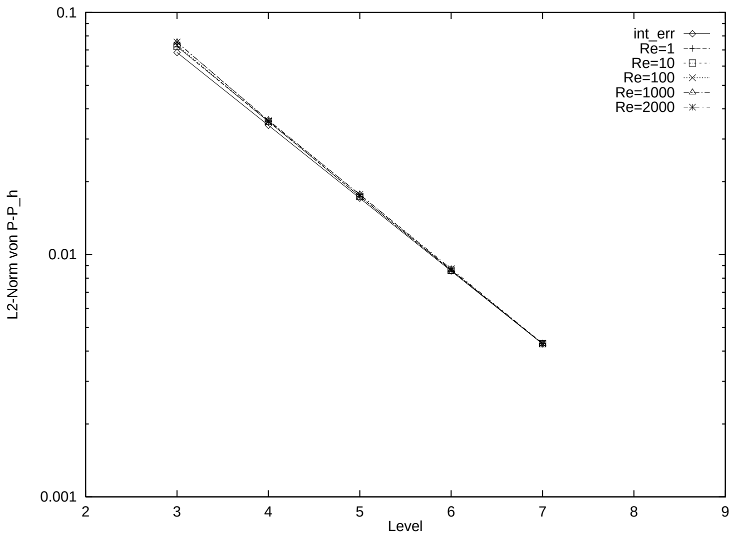


Abbildung 5.1: *Fehlerreduktion für* $h = 2^{-Level} \to 0$

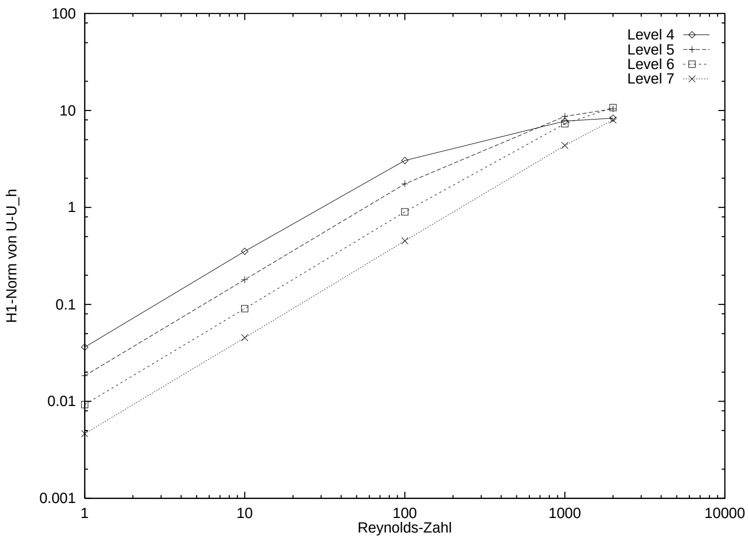


Abbildung 5.2: *Geschwindigkeitsfehler in der diskreten H^1-Norm für wachsende Re*

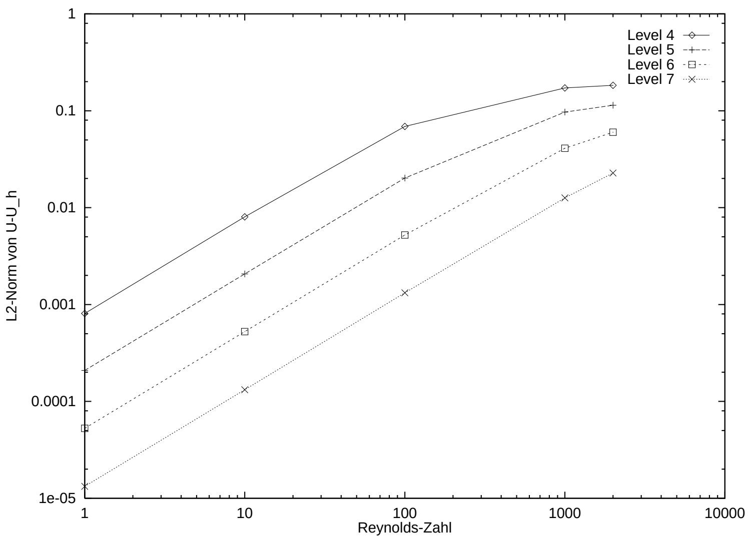
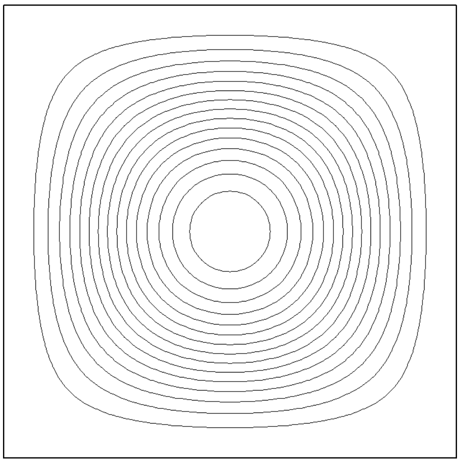
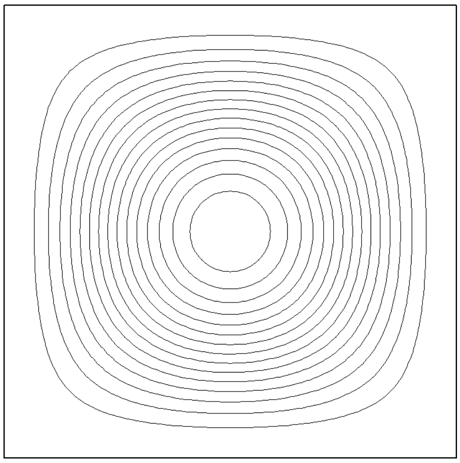


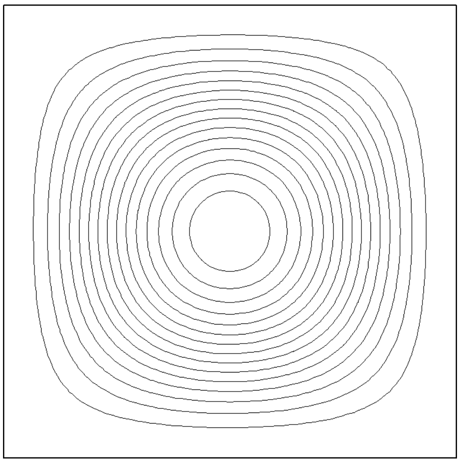
Abbildung 5.3: *Geschwindigkeitsfehler in der L^2-Norm für wachsende Re*



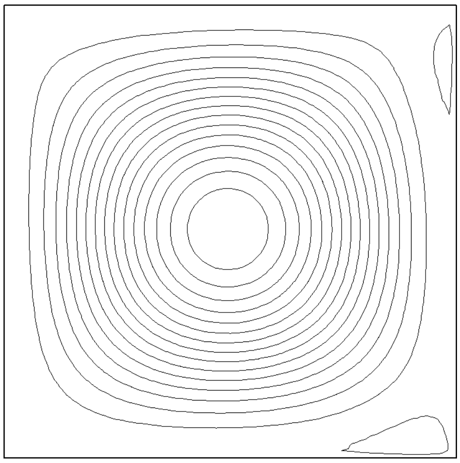
exakte Lösung



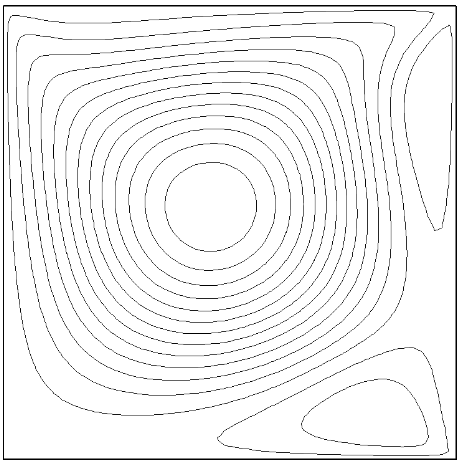
$Re = 0.2$



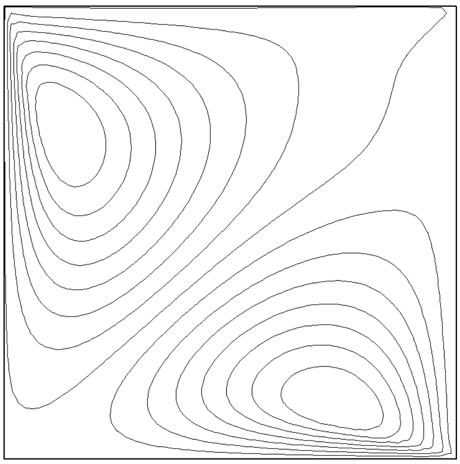
$Re = 2$



$Re = 20$



$Re = 200$



$Re = 2000$

Abbildung 5.4: *Stromlinien für* $h = 1/64$

5.2 Die Idee der Druckabspaltung

5.2.1 Das Stokes Problem

Um zu verstehen, wie es im obigen Testbeispiel zu einem derartigen Fehlerwachstum kommen kann, vereinfachen wir zunächst das Problem, indem wir den nichtlinearen Konvektionsterm weglassen. Für das entstehende ν-abhängige Stokes Problem untersuchen wir nun, ob sich ein ähnliches Fehlerverhalten in Abhängigkeit von der Reynolds-Zahl $Re = 1/\nu$ zeigt. Unser Problem für (u, p) lautet jetzt:

$$\begin{aligned} -\nu\triangle u \;+\; \nabla p \;&=\; f\,, \\ \nabla\cdot u \;&=\; 0\,, \\ u_{|\partial\Omega} \;&=\; 0\,. \end{aligned} \tag{5.1}$$

Als Testbeispiel betrachten wir wieder den Fall $\Omega = (0,1)^2$ und $f := -\nu\triangle u + \nabla p$ mit der vorgegebenen kontinuierlichen Lösung (u, p) aus dem vorhergehenden Abschnitt. Zur Berechnung der diskreten Lösung (u_h, p_h) verwenden wir äquidistante Gitter aus quadratischen Elementen mit der Kantenlänge h und die punktwert-orientierten Finiten Elemente.

Tabelle 5.2: *Fehlernormen beim Stokes Problem (5.1), h=1/128*

Re	$\|u - u_h\|_h$	$\|u - u_h\|_{0,2,\Omega}$	$\|p - p_h\|_{0,2,\Omega}$
2	0.912 E−02	0.264 E−04	0.429 E−02
20	0.907 E−01	0.264 E−03	0.429 E−02
200	0.907 E+00	0.264 E−02	0.429 E−02
2000	0.907 E+01	0.264 E−01	0.429 E−02

In Tabelle 5.2 zeigen wir zur festgehaltenen Gitterweite $h = 1/128$ die Fehlernormen der Diskretisierungsfehler für wachsende Reynolds-Zahlen und in Abbildung 5.5 die entsprechenden Stromlinienbilder der berechneten Geschwindigkeitslösung u_h. Wir sehen, daß die Fehler in der Geschwindigkeit wieder wie $O(Re)$ wachsen während die Güte der Druckapproximation unabhängig von Re ist. Es zeigt sich ein analoges Verhalten zu unserem Navier-Stokes Testbeipiel. Dies spricht dafür, daß die Ursache des $O(Re)$-Fehlerwachstums nicht primär im Vorhandensein des nichtlinearen Konvektionsterms zu suchen ist.

Für das Stokes Problem (5.1) läßt sich das beobachtete Fehlerverhalten sehr einfach theoretisch erklären. Aus Gründen einer kürzeren Darstellung verzichten wir hier auf die Verwendung adaptiv gewichteter Sobolev-Normen und berschränken uns auf den Fall blockstrukturierter Gitter, wo eine Abschätzung der Gestalt $\sigma_h \leq Ch$ gilt (siehe Abschnitt 4.8). Sei $(u, p) \in V \times Q$ die schwache Lösung des Problems (5.1) und $(u_h, p_h) \in V_h \times Q_h$ die zugehörige diskrete Lösung. Sei weiterhin $(\bar{u}, \bar{p}) \in V \times Q$ die schwache Lösung

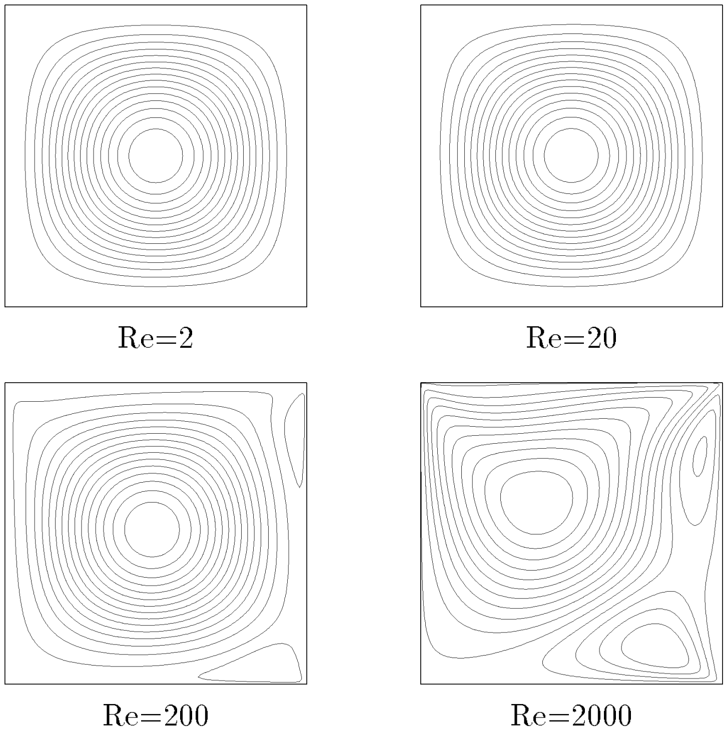


Abbildung 5.5: *Stromlinien beim Stokes Problem für* $h = 1/128$

des Problems (5.1) mit $\nu = 1$ und derselben rechten Seite f sowie $(\bar{u}_h, \bar{p}_h) \in V_h \times Q_h$ die entsprechende diskrete Lösung. Auf Grund der Linearität des Problems gilt dann

$$(u, p) = (\frac{1}{\nu}\bar{u}, \bar{p}) \qquad \text{sowie} \qquad (u_h, p_h) = (\frac{1}{\nu}\bar{u}_h, \bar{p}_h)\,.$$

Wenden wir die Analysis von Rannacher/Turek [48] auf das Stokes Problem für $(\bar{u}, \bar{p})$ mit ν=1 an, so folgen nach Rückskalierung auf (u, p) bzw. (u_h, p_h) die Fehlerabschätzungen

$$h\|u - u_h\|_h \,+\, \|u - u_h\|_{0,2,\Omega} \,\leq\, Ch^2\,\|u\|_{2,2,\Omega} \,+\, C\frac{h^2}{\nu}\,\|p\|_{1,2,\Omega} \tag{5.2}$$

und

$$\|p - p_h\|_{0,2,\Omega} \,\leq\, Ch\nu\,\|u\|_{2,2,\Omega} \,+\, Ch\,\|p\|_{1,2,\Omega}\,. \tag{5.3}$$

Durch (5.3) wird klar, weshalb sich die Druckapproximation mit wachsender Reynolds-Zahl $Re = 1/\nu$ nicht verschlechtert. Für die Geschwindigkeit erhalten wir aus (5.2) lediglich eine Fehlerschranke der Gestalt $CReh^m$. Aus der Tatsache $\|\bar{u} - \bar{u}_h\| = O(h^m)$ folgt nach Skalierung $\|u - u_h\| = O(Reh^m)$, woraus ersichtlich ist, daß sich der Faktor Re in dieser Fehlerschranke nicht vermeiden läßt.

In der Abschätzung (5.2) liegt aber auch der Schlüssel dafür, wie man das Fehlerwachstum in einem bestimmten Re-Bereich recht wirkungsvoll eindämmen kann. Wäre nämlich $\|p\|_{1,2,\Omega}$ klein, so würde der entsprechende Fehleranteil erst für relativ große Re dominant. Daraus entsteht die Idee der Druckabspaltung. Wir versuchen, ein äquivalentes Stokes Problem mit einer modifizierten rechten Seite $\widetilde{f}$ zu lösen, bei dem die H^1-Norm der kontinuierlichen Drucklösung $\widetilde{p}$ klein ist. Hierzu spalten wir in der Stokes-Gleichung (5.1) den Gradienten eines geeigneten Druckes p_s ab und betrachten das modifizierte Stokes Problem für $(\widetilde{u}, \widetilde{p})$

$$\begin{aligned} -\nu \triangle \widetilde{u} \,+\, \nabla \widetilde{p} \;&=\; \widetilde{f} \;:=\; f - \nabla p_s\,, \\ \nabla \cdot \widetilde{u} \;&=\; 0\,, \\ \widetilde{u}_{|\partial\Omega} \;&=\; 0\,. \end{aligned} \tag{5.4}$$

Wir nennen p_s den *separierten Druck* und fordern $p_s \in H^1(\Omega)$. Das Problem (5.4) hat offenbar die Lösung $(\widetilde{u}, \widetilde{p}) = (u, p - p_s)$, wenn (u,p) die Lösung von (5.1) ist. Sei $(\widetilde{u}_h, \widetilde{p}_h) \in V_h \times Q_h$ die diskrete Finite Elemente Lösung des modifizierten Stokes Problems (5.4). Wenden wir auf dieses Problem die Fehlerabschätzung (5.2) an, so ergibt sich

$$h\|u - \widetilde{u}_h\|_h \,+\, \|u - \widetilde{u}_h\|_{0,2,\Omega} \;\le\; Ch^2\,\|u\|_{2,2,\Omega} \,+\, C\frac{h^2}{\nu}\,\|p - p_s\|_{1,2,\Omega}\,. \tag{5.5}$$

Wenn also p_s eine gute Approximation von p mit $\|p - p_s\|_{1,2,\Omega} \ll \|p\|_{1,2,\Omega}$ ist, so erhalten wir im Fall hoher Reynolds-Zahlen für den Geschwindigkeitsfehler $u - \widetilde{u}_h$ eine wesentlich bessere Fehlerschranke als für $u - u_h$. Es bietet sich daher für hohe Reynolds-Zahlen an, als Näherungslösung unseres ursprünglichen Stokes Problems (5.1) nicht (u_h, p_h), sondern $(\widetilde{u}_h, \widetilde{p}_h + J_h p_s)$ zu verwenden, wobei $J_h : H^1(\Omega) \to Q_h$ der Finite Elemente Interpolationsoperator für den Druckraum ist. Für den modifizierten Druck $p_h^* := \widetilde{p}_h + J_h p_s$ erhalten wir die Fehlerabschätzung

$$\|p - p_h^*\|_{0,2,\Omega} \;\le\; Ch\nu\,\|u\|_{2,2,\Omega} \,+\, Ch\,(\,\|p\|_{1,2,\Omega} + \|p_s\|_{1,2,\Omega}\,)\,, \tag{5.6}$$

d.h. die Ordnung der Druckapproximation wird dadurch nicht beeinträchtigt.

Es gilt nun, einen separierten Druck $p_s \in H^1(\Omega)$ zu finden, der einerseits eine gute Näherung des exakten Druckes p ist und andererseits mit vertretbarem Aufwand berechenbar ist. Aus dem Stokes Problem ergibt sich für p die Gleichung

$$\nabla p \;=\; f \,+\, \nu \triangle u\,,$$

d.h. für hohe Reynolds-Zahlen ist $\nabla p \approx f$. Dies führt auf die Idee, den separierten Druck p_s aus einer konformen Finite-Elemente-Approximation der Gleichung $\nabla p = f$ zu bestimmen. Dazu sei $S_h^0 \subset H^1(\Omega) \cap L_0^2(\Omega)$ der Teilraum der konformen isoparametrischen Q_1-Elemente mit $\int_\Omega \theta_h\,dx = 0 \;\; \forall\,\theta_h \in S_h^0$. Dann definieren wir p_s als die Lösung des folgenden Neumann Problems:

$$\begin{aligned} &\textit{Finde } p_s \in S_h^0, \textit{ so daß} \\ &(\nabla p_s, \nabla\theta_h) \;=\; (f, \nabla\theta_h) \qquad \forall\,\theta_h \in S_h^0\,. \end{aligned} \tag{5.7}$$

Für dieses Problem sind die Existenz und Eindeutigkeit der Lösung gesichert. Unter der Regularitätsvoraussetzung $(u,p) \in (H^2(\Omega))^d \times H^2(\Omega)$ erhalten wir mit Hilfe von Standardtechniken die Abschätzung

$$\|p - p_s\|_{1,2,\Omega} \leq Ch\,\|p\|_{2,2,\Omega} + C\nu\,\|\triangle u\|_{0,2,\Omega}\,. \tag{5.8}$$

Setzen wir dieses Resultat in (5.5) ein, so ergibt sich die folgende verbesserte Fehlerabschätzung für die modifizierte Geschwindigkeitsapproximation $\tilde{u}_h$

$$h\|u - \tilde{u}_h\|_h + \|u - \tilde{u}_h\|_{0,2,\Omega} \leq Ch^2\,(\,\|u\|_{2,2,\Omega} + \|\triangle u\|_{0,2,\Omega}) + C\frac{h^3}{\nu}\,\|p\|_{2,2,\Omega}\,. \tag{5.9}$$

Tabelle 5.3 zeigt für den Fall $Re = 2000$, welche enorme Verbesserung die vorgeschlagene Technik der Druckabspaltung für den Geschwindigkeitsfehler in unserem Testbeipiel bringt.

Tabelle 5.3: *Vergleich der Fehlernormen mit Druckabspaltung ($u - \tilde{u}_h$) und ohne ($u - u_h$) für Re=2000*

h	$\|u - u_h\|_h$	$\|u - \tilde{u}_h\|_h$	$\|u - u_h\|_{0,2,\Omega}$	$\|u - \tilde{u}_h\|_{0,2,\Omega}$
1/16	0.707 E+02	0.748 E−01	0.161 E+01	0.185 E−02
1/32	0.360 E+02	0.105 E−01	0.415 E+00	0.125 E−03
1/64	0.181 E+02	0.248 E−02	0.105 E+00	0.120 E−04
1/128	0.907 E+01	0.109 E−02	0.264 E−01	0.238 E−05

Tabelle 5.4: *Fehlerverhalten bei der Methode der Druckabspaltung für wachsende Reynolds-Zahlen beim Stokes Problem*

h	Re	$\|u - \tilde{u}_h\|_h$	$\|u - \tilde{u}_h\|_{0,2,\Omega}$
1/16	2	0.852 E−02	0.139 E−03
	20	0.855 E−02	0.140 E−03
	200	0.113 E−01	0.231 E−03
	2000	0.748 E−01	0.185 E−02
1/32	2	0.430 E−02	0.367 E−04
	20	0.430 E−02	0.367 E−04
	200	0.441 E−02	0.386 E−04
	2000	0.105 E−01	0.125 E−03
1/64	2	0.216 E−02	0.930 E−05
	20	0.216 E−02	0.931 E−05
	200	0.216 E−02	0.934 E−05
	2000	0.248 E−02	0.120 E−04

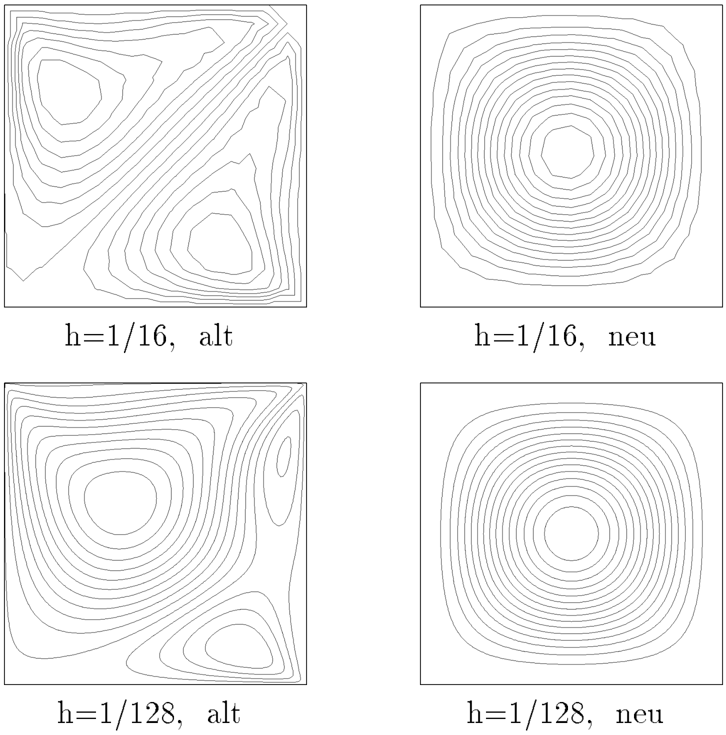


Abbildung 5.6: *Stromlinien beim Stokes Problem für Re=2000 mit Druckabspaltung (*"neu"*) und ohne (*"alt"*)*

In Abbildung 5.6 wird dies anhand der Stromlinienbilder verdeutlicht. Die Dimension des Neumann Problems für p_s ist gering im Vergleich zu der des eigentlichen Stokes Problems. Wenden wir außerdem zu dessen Lösung ein effektives Mehrgitterverfahren an, so ist der entstehende Mehraufwand für die Druckabspaltung vernachlässigbar klein.

Tabelle 5.4 zeigt, wie bei der Methode der Druckabspaltung für feste Gitterweiten h die Geschwindigkeitsfehler $u - \tilde{u}_h$ von der Reynolds-Zahl abhängen. In einem bestimmten Re-Bereich sind die Fehler von Re unabhängig und beginnen dann zu wachsen. Dieser Bereich wird aber für kleinere h größer, was in Übereinstimmung mit der Fehlerabschätzung (5.9) steht. Dies bedeutet, daß wir mit unserer Technik der Druckabspaltung das Fehlerwachstum bezüglich Re zwar nicht verhindern, aber dennoch für hinreichend feine Gitter recht wirkungsvoll eindämmen können.

5.2.2 Das Navier-Stokes Problem

Da uns im Falle hoher Reynolds-Zahlen für das Navier-Stokes Problem keine Fehleranalysis zur Verfügung steht, betrachten wir zunächst die linearisierten *Oseen Gleichungen*

$$\begin{aligned} -\nu\triangle u \;+\; (\beta\cdot\nabla)u \;+\; \nabla p \;&=\; f\,, \\ \nabla\cdot u \;&=\; 0\,, \\ u_{|\partial\Omega} \;&=\; 0\,, \end{aligned} \tag{5.10}$$

wobei $\beta \in \mathbf{R}^d$ ein gegebenes, hinreichend glattes Geschwindigkeitsfeld mit $\nabla\cdot\beta = 0$ sei. Wenden wir auf dieses Problem unsere upwind-Diskretisierung an, so können wir zu beliebigem ν in analoger Weise zur Analysis in Kapitel 4 die folgende Fehlerabschätzung zeigen

$$\|u - u_h\|_h \;\le\; Ch\,\|u\|_{2,2,\Omega} \;+\; C(\beta)\frac{h}{\nu}\,\|u\|_{2,2,\Omega} \;+\; C\frac{h}{\nu}\,\|p\|_{1,2,\Omega}\,. \tag{5.11}$$

Hierbei sind der zweite und dritte Summand auf der rechten Seite potentielle Quellen eines Fehlerwachstums für $\nu \to 0$. In Fällen mit $|\beta| \ll 1$ (wenn z.B. Ω ein Teilbereich mit langsamer Strömug innerhalb eines größeren Strömungsgebietes wäre) würde wie beim Stokes Problem der druckbehaftete Term für $Re \gg 1$ dominant werden. Es bietet sich dann wieder die Technik der Druckabspaltung an, bei der wir die Oseen Gleichungen mit der modifizierten rechten Seite $\tilde{f} := f - \nabla p_s$ diskretisieren und deren Geschwindigkeitslösung $\tilde{u}_h$ verwenden.

Die Frage ist nun, wie wir in diesem Fall einen geeigneten separierten Druck $p_s \in H^1(\Omega)$ mit $\|p - p_s\|_{1,2,\Omega} \ll \|p\|_{1,2,\Omega}$ bestimmen. Vernachlässigen wir wieder für $\nu \ll 1$ den Term "$\nu\triangle u$", so erhalten wir für p die Beziehung

$$\nabla p \;\approx\; f \;-\; (\beta\cdot\nabla)u\,. \tag{5.12}$$

Das Problem ist jetzt, daß wir den Konvektionsterm $(\beta\cdot\nabla)u$ nicht kennen. Betrachten wir jedoch das Oseen Problem als ein lineares Teilproblem innerhalb einer Fixpunktiteration zur Lösung des nichtlinearen Navier-Stokes Problems, so ist $\beta = u_h^{m-1}$ die alte Iterierte aus dem vorhergehenden Schritt. Dies führt zu der Idee, den Konvektionsterm in (5.12) durch

$$(\beta\cdot\nabla)u \;\approx\; (u_h^{m-1}\cdot\nabla)u_h^{m-1}$$

zu approximieren und den separierten Druck p_s dann aus einer konformen Finite Elemente Diskretisierung der (5.12) entsprechenden Gleichung zu bestimmen. Daraus ergibt sich die folgende modifizierte Fixpunktiteration zur Lösung des Navier-Stokes Problems. Sei (u_h^{m-1}, p_h^{m-1}) die alte Näherungslösung, ω ein geeigneter Relaxationsparameter und $J_h : H^1(\Omega) \to Q_h$ der Finite Elemente Interpolationsoperator für den Druckraum. Dann besteht die Berechnung der neuen Iterierten (u_h^m, p_h^m) aus den folgenden drei Schritten.

NICHTLINEARE ITERATION MIT DRUCKABSPALTUNG: $m-1 \to m$

(1) Bestimme $p_s^m \in S_h^0$, so daß

$$(\nabla p_s^m, \nabla\theta_h) \;=\; \sum_{K\in\mathbf{T}_h} (f - (u_h^{m-1}\cdot\nabla)u_h^{m-1}, \nabla\theta_h)_K \quad \forall\, \theta_h \in S_h^0 .$$

(2) Bestimme $(\tilde{u}_h^m, \tilde{p}_h^m) \in V_h \times Q_h$, so daß

$$\begin{aligned} \nu\, a_h(\tilde{u}_h^m, v_h) + \widetilde{n}_h(u_h^{m-1}, \tilde{u}_h^m, v_h) + b_h(\tilde{p}_h^m, v_h) &= (f - \nabla p_s^m, v_h) \quad && \forall\, v_h \in V_h, \\ b_h(q_h, \tilde{u}_h^m) &= 0 && \forall\, q_h \in Q_h. \end{aligned}$$

(3) Relaxation

$$\begin{aligned} u_h^m &= u_h^{m-1} + \omega\,(\tilde{u}_h^m - u_h^{m-1}) \\ p_h^m &= p_h^{m-1} + \omega\,(\tilde{p}_h^m + J_h p_s^m - p_h^{m-1}) \end{aligned}$$

Wie beim Stokes Problem ist auch hier der durch die Druckabspaltung entstehende Mehraufwand unwesentlich.

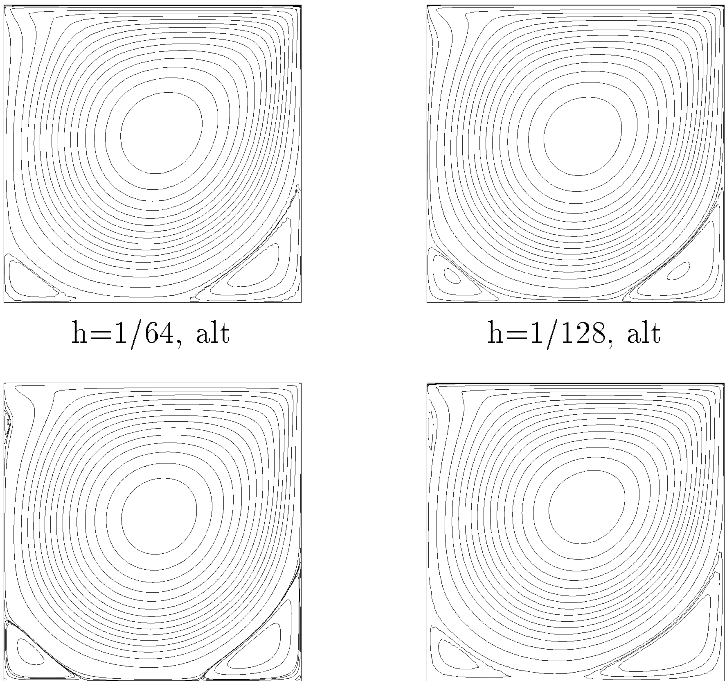
Wir betrachten wieder unser Testbeipiel aus Abschnitt 5.1. Zur Diskretisierung verwenden wir die punktwert-orientierten Q_1-parametrischen Elemente und die ursprüngliche einfache upwind-Version $\widetilde{n}_h$. Tabelle 5.5 zeigt einen Vergleich zwischen dem Geschwin-

Tabelle 5.5: *Vergleich der Fehlernormen mit Druckabspaltung* $(u - \widetilde{u}_h)$ *und ohne* $(u - u_h)$

h	Re	$\|u-u_h\|_h$	$\|u-\widetilde{u}_h\|_h$	$\|u-u_h\|_{0,2,\Omega}$	$\|u-\widetilde{u}_h\|_{0,2,\Omega}$
1/16	2	0.71 E−01	0.85 E−02	0.16 E−02	0.14 E−03
	20	0.70 E+00	0.85 E−02	0.16 E−01	0.14 E−03
	200	0.48 E+01	0.11 E−01	0.11 E+00	0.25 E−03
	2000	0.83 E+01	0.62 E−01	0.18 E+00	0.20 E−02
1/32	2	0.36 E−01	0.43 E−02	0.41 E−03	0.36 E−04
	20	0.36 E+00	0.43 E−02	0.41 E−02	0.35 E−04
	200	0.33 E+01	0.45 E−02	0.38 E−01	0.76 E−04
	2000	0.10 E+02	0.12 E−01	0.11 E+00	0.59 E−03
1/64	2	0.18 E−01	0.22 E−02	0.10 E−03	0.92 E−05
	20	0.18 E+00	0.22 E−02	0.10 E−02	0.91 E−05
	200	0.18 E+01	0.22 E−02	0.10 E−01	0.37 E−04
	2000	0.11 E+02	0.49 E−02	0.60 E−01	0.33 E−03

digkeitsfehler $u - u_h$ aus der bisherigen Diskretisierung und dem Fehler $u - \widetilde{u}_h$ aus der Methode der Druckabspaltung. Die Fehler im Druck sind bei beiden Methoden nahezu identisch. Wir sehen, daß die Druckabspaltung insbesondere für hohe Reynolds-Zahlen zu einer wesentlichen Verbesserung in der Geschwindigkeitsapproximation führt.

Um zu zeigen, daß die vorgeschlagene Technik nicht nur bei unserem akademischen Testbeispiel einen positiven Effekt hat, betrachten wir das driven-cavitiy-Problem mit $\Omega = (0,1)^2$, $f = 0$ und den Dirichlet'schen Randbedingungen $u(x,y)_{|\partial\Omega} = (1,0)^{\mathsf{T}}$ für $y = 1$ und $u(x,y)_{|\partial\Omega} = 0$ für $y < 1$. Das Gitter und die Diskretisierung seien wie im vorhergehenden Beispiel gewählt. Abbildung 5.7 zeigt zur Reynolds-Zahl Re=2000 den



h=1/64, alt   h=1/128, alt

h=1/256, alt   h=1/64, neu

Abbildung 5.7: *Stromlinien beim driven-cavity-Problem für* $Re = 2000$ *mit Druckabspaltung (*"neu"*) und ohne (*"alt"*)*

Stromlinienverlauf der diskreten Lösung u_h ohne Druckabspaltung für die Gitterweiten $h \in \{1/64, 1/128, 1/256\}$ sowie den der Lösung $\tilde{u}_h$ mit Druckabspaltung für $h = 1/64$. Dabei wurde lediglich ein Iterationsschritt mit Druckabspaltung ausgeführt und danach zur üblichen Fixpunktiteration mit der modifizierten festen rechten Seite $\widetilde{f} := f - \nabla p_s^1$ übergegangen. Während bei der ursprünglichen Diskretisierung der dritte Wirbel links oben erst für $h = 1/256$ erscheint, ist er bei der Methode der Druckabspaltung bereits für $h = 1/64$ sichtbar.

Es gibt jedoch auch Fälle, in denen man mit Hilfe der Druckabspaltung keine Verbesserung erreichen kann. Ein Beispiel ist die Poiseuille-Strömung, d.h. eine Strömung in einem Kanal $\Omega = (0,L) \times (0,H)$ mit $f = 0$ und parabolischem Ein- und Ausströmprofil, bei der die kontinuierliche Lösung (u,p) bekannt ist und die Gestalt

$$u(x,y) = (\alpha y(H-y)\,,\,0)^{\mathsf{T}}, \qquad p(x,y) = \nu\alpha(L-2x)\,,$$

hat. Hier gilt $\|p\|_{1,2,\Omega} = O(\nu)$, so daß der druckbehaftete Term in der Fehlerabschätzung (5.11) für alle ν beschränkt bleibt. Die Abspaltung eines Druckes p_s ist demzufolge nicht mehr erforderlich. Außerdem würde in diesem Fall für $\nu \ll 1$ sicherlich $\|p-p_s\|_{1,2,\Omega} > \|p\|_{1,2,\Omega}$ gelten, so daß eine Verbesserung der Genauigkeit nicht zu erwarten wäre. Dies wurde auch durch praktische Rechnungen bestätigt.

Wir verwenden also die Näherungslösung aus der ursprünglichen Diskretisierung ohne Druckabspaltung und analysieren das Fehlerverhalten für festes h und $\nu \to 0$. Als Parameter wählen wir L=2, H=0.5 und $\alpha = 4H^{-2}$. Das Gebiet Ω sei in quadratische Elemente der Kantenlänge h zerlegt. Zur Diskretisierung verwenden wir basierend auf den punktwert-orientierten Q_1-parametrischen Elementen sowohl die urprüngliche upwind-Version $\tilde{n}_h$ mit der Gewichtsfunktion Φ=Φ_1 aus (3.15) als auch die entsprechende modifizierte Variante $\bar{n}_h$(siehe Kapitel 3). In Abbildung 5.8 ist der Stromlinienverlauf

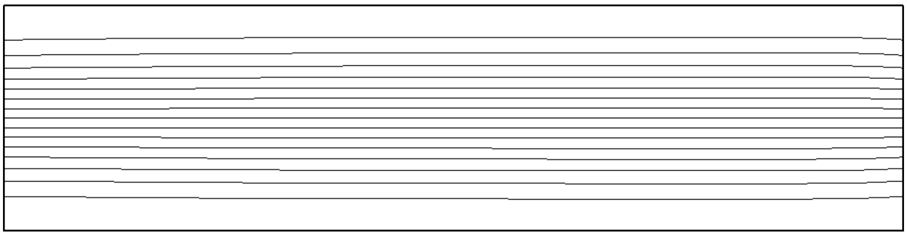


Abbildung 5.8: *Stromlinien zur upwind-Lösung beim Poiseuille-Problem für Re=2000 und* $h = 1/128$

der mit $\bar{n}_h$ berechneten Näherungslösung für h=1/128 und Re=2000 dargestellt. Tabelle 5.6 zeigt zu verschiedenen Gitterweiten das Verhalten der Diskretisierungsfehler bei wachsender Reynolds-Zahl. In diesem Beispiel gibt es kaum Unterschiede zwischen den beiden upwind-Varianten. Obwohl der Druck hier keinen negativen Einfluß mehr auf den Diskretisierungsfehler hat, kommt es dennoch zu einem Anwachsen des Geschwindigkeitsfehlers für $\nu \to 0$. Wie dieser Effekt zu erklären ist und welche Methoden es gibt, ihm entgegenzuwirken, ist Gegenstand weiterer Forschungsarbeiten.

Tabelle 5.6: *Normen des Diskretisierungsfehlers beim Poiseuille-Problem für die ursprüngliche upwind-Version "$\widetilde{n}_h$" und die modifizierte Variante "$\bar{n}_h$"*

h	Re	$\|u-u_h\|_h$		$\|u-u_h\|_{0,2,\Omega}$		$\|p-p_h\|_{0,2,\Omega}$	
		$\widetilde{n}_h$	$\bar{n}_h$	$\widetilde{n}_h$	$\bar{n}_h$	$\widetilde{n}_h$	$\bar{n}_h$
$\frac{1}{16}$	2	.265 E+00	.265 E+00	.827 E−02	.827 E−02	.163 E+00	.163 E+00
	20	.260 E+00	.260 E+00	.879 E−02	.879 E−02	.102 E−01	.102 E−01
	200	.320 E+00	.319 E+00	.149 E−01	.149 E−01	.768 E−02	.768 E−02
	2000	.570 E+00	.568 E+00	.271 E−01	.270 E−01	.805 E−02	.803 E−02
$\frac{1}{32}$	2	.136 E+00	.136 E+00	.218 E−02	.218 E−02	.552 E−01	.552 E−01
	20	.135 E+00	.135 E+00	.256 E−02	.256 E−02	.456 E−02	.456 E−02
	200	.180 E+00	.180 E+00	.736 E−02	.736 E−02	.446 E−02	.446 E−02
	2000	.525 E+00	.524 E+00	.213 E−01	.213 E−01	.399 E−02	.398 E−02
$\frac{1}{64}$	2	.683 E−01	.683 E−01	.561 E−03	.561 E−03	.230 E−01	.230 E−01
	20	.682 E−01	.682 E−01	.803 E−03	.803 E−03	.269 E−02	.269 E−02
	200	.956 E−01	.956 E−01	.379 E−02	.379 E−02	.237 E−02	.237 E−02
	2000	.367 E+00	.366 E+00	.149 E−01	.149 E−01	.186 E−02	.186 E−02
$\frac{1}{128}$	2	.342 E−01	.342 E−01	.146 E−03	.146 E−03	.109 E−01	.109 E−01
	20	.343 E−01	.343 E−01	.291 E−03	.291 E−03	.150 E−02	.150 E−02
	200	.497 E−01	.497 E−01	.195 E−02	.195 E−02	.122 E−02	.122 E−02
	2000	.222 E+00	.222 E+00	.925 E−02	.925 E−02	.866 E−03	.867 E−03

Kapitel 6

Mehrgitterverfahren

Das diskrete Navier-Stokes Problem (3.8) entspricht einem nichtlinearen algebraischen Gleichungssystem für die Knotenwerte der Geschwindigkeit und des Druckes. In diesem Kapitel beschreiben wir den Algorithmus, den wir zur sequentiellen Lösung dieses Gleichungssystems auf einem herkömmlichen Ein-Prozessor-Rechner verwenden. Dieser Algorithmus bildet gleichzeitig die Grundlage für unseren parallelen Löser, den wir im folgenden Kapitel behandeln. Zur Auflösung feiner Strukturen oder aber zur zuverlässigen Berechnung gewisser Kenngrößen der Strömung benötigt man hinreichend feine Gitter mit einer hohen Anzahl N_e von Elementen. Dies bedeutet, daß wir ein nichtlineares Gleichungssystem von sehr hoher Dimension zu lösen haben (etwa $5N_e$ im 2D-Fall und $10N_e$ im 3D-Fall). Es bietet sich daher der Einsatz eines Mehrgitterverfahrens an. Wir verwenden eine äußere nichtlineare Iteration und lösen jedes lineare Teilproblem mit Hilfe eines linearen Mehrgitterverfahrens. In früheren Untersuchungen [24] haben wir die Erfahrung gemacht, daß diese Variante insbesondere im Fall hoher Reynolds-Zahlen Vorteile gegenüber einer nichtlinearen Mehrgittermethode hat.

Zunächst behandeln wir die Konstruktion der Gitter und beschreiben dann in den beiden folgenden Abschnitten die nichtlineare Iteration sowie die Komponenten des linearen Mehrgitterlösers. Numerische Konvergenzraten für verschiedene Reynolds-Zahlen und Gitterweiten werden wir im nächsten Kapitel (Abschnitt 8.1) angeben, wo wir das Konvergenzverhalten des parallelen und sequentiellen Mehrgitter-Algorithmus vergleichen.

6.1 Blockstrukturierte Gitter

Für unser Mehrgitter-Verfahren verwenden wir blockstrukturierte Gitter, bei denen die Blöcke den Viereck- bzw. Hexaederelementen eines groben Gitters entsprechen. Diese Gitter bilden die Grundlage für unsere Parallelisierungsstrategie. Sie sind folgendermaßen definiert. Wir beginnen mit einer shape-regulären Zerlegung des Gebietes Ω in sogenannte *Makroelemente* M_i, $i = 1, \ldots, N_m$ (siehe Abbildung 6.1). Diese Zerlegung

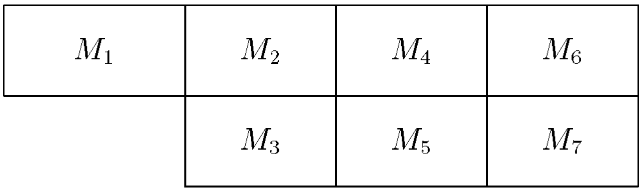


Abbildung 6.1: *Zerlegung von Ω in Makroelemente M_i*

entspricht dem gröbsten Gitter zum Level k=1. Durch uniforme Verfeinerung können nun unabhängig auf jedem Makroelement (also auch parallel) alle feineren Gitter konstruiert werden. Dazu wird jedes Element K des vorhergehenden Gitterlevels in 2^d Sohnelemente zerlegt (siehe Abbildung 6.2 für d=2), wobei als Ecken der Sohnelemente die Ecken, der Schwerpunkt, die Kantenmittelpunkte und im 3D-Fall außerdem noch die Schwerpunkte der Seitenflächen von K verwendet werden.

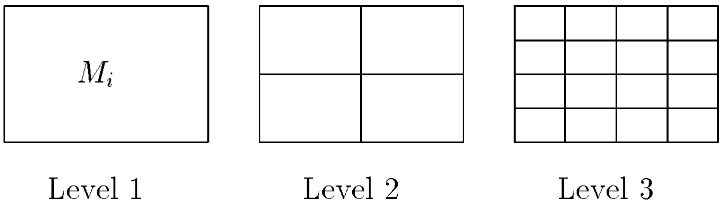


Abbildung 6.2: *Konstruktion feinerer Gitterlevel auf einem Makroelement M_i*

Im Fall eines krummlinigen Gebietsrandes, wird der Verfeinerungsprozeß so modifiziert, daß für Elemente, die an einem krummlinigen Randteil liegen, die entsprechenden Ecken der Sohnelemente jeweils auf den Rand des Gebietes verschoben werden (siehe Abbildung 6.3).

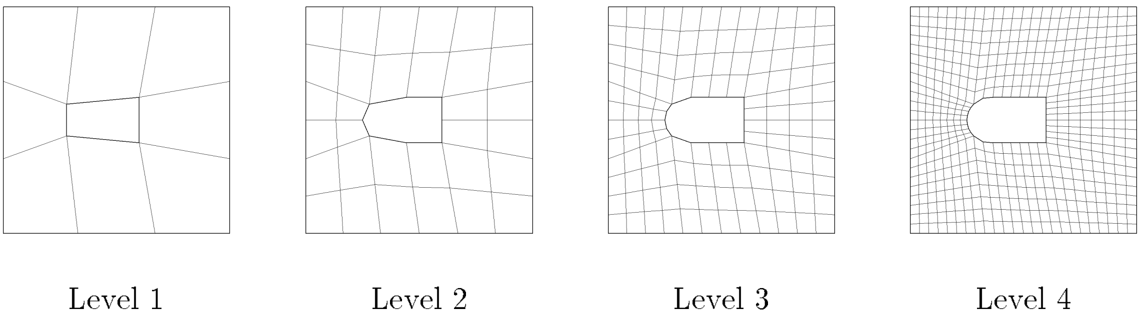


Abbildung 6.3: *Konstruktion feinerer Gitterlevel bei krummlinigen Randteilen*

Blockstrukturierte Gitter stellen einen Kompromiß zwischen zwei Dingen dar. Auf der einen Seite bieten sie noch eine relativ hohe Flexibilität, um komplexe Geometrien zu erfassen. Auf der anderen Seite hat man lokal auf den Makroelementen ein sehr regelmäßiges Gitter, das sich hinsichtlich der Datenstruktur einfach verwalten läßt und die Möglichkeit für eine Vektorisierung bietet. Außerdem gilt hier für die Winkelgröße σ_h die Abschätzung $\sigma_h \leq Ch$, so daß alle vier in Abschnitt 2.2 vorgeschlagenen Elementepaare dieselbe Konvergenzordnung besitzen (siehe Abschnitt 4.8) und man sich die vom Rechenaufwand her günstigste Variante (etwa die punktwert-orientierten Q_1-parametrischen Elemente) auswählen kann. Ein weiterer Vorteil blockstrukturierter Gitter ist, daß sie eine natürliche, nahezu optimale Lastverteilung für eine Parallelisierung liefern, da die Anzahl der Elemente auf jedem Makroelement dieselbe ist. Es bietet sich daher an, jedem Makroelement einen Prozessor zuzuordnen, der für alle Operationen mit den Daten dieses Makroelementes zuständig ist.

6.2 Nichtlineare Iteration

Im folgenden bezeichnen wir mit $\mathbf{u} \in \mathbf{R}^{N_u}$ und $\mathbf{p} \in \mathbf{R}^{N_p}$ jeweils den Vektor der Knotenwerte zur Lösung (u_h, p_h) des diskreten Navier-Stokes Problems (3.8), d.h. es gelte

$$u_h \;=\; \sum_{j=1}^{N_u} \mathbf{u}_j \varphi_j\,, \qquad p_h \;=\; \sum_{j=1}^{N_p} \mathbf{p}_j \chi_j\,,$$

wobei $\{\varphi_j\}$ die Basisfunktionen von V_h und $\{\chi_j\}$ die von Q_h seien. Das diskrete Problem entspricht dann einem nichtlinearen algebraischen Gleichungssystem der Struktur

$$L(\mathbf{y})\,\mathbf{y} = \mathbf{F} \tag{6.1}$$

mit

$$L(\mathbf{y}) := \begin{pmatrix} A(\mathbf{u}) & B \\ B^{\mathrm{T}} & 0 \end{pmatrix}, \quad \mathbf{y} := \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix}, \quad \mathbf{F} := \begin{pmatrix} \mathbf{f} \\ 0 \end{pmatrix}. \tag{6.2}$$

Dabei haben B und $\mathbf{f}$ die Komponenten $B_{ij} = b_h(\chi_j, \varphi_i)$ und $\mathbf{f}_i = (f, \varphi_i)$. Zu einem gegebenen Vektor $\mathbf{v} \in \mathbf{R}^{N_u}$ sind die Einträge der Matrix $A(\mathbf{v})$ definiert durch

$$(A(\mathbf{v}))_{ij} \;=\; \nu a_h(\varphi_j, \varphi_i) \;+\; n_h^*(v_h, \varphi_j, \varphi_i)\,,$$

wobei v_h die zu $\mathbf{v}$ gehörende Finite Elemente Funktion $v_h = \sum_j \mathbf{v}_j \varphi_j$ und n_h^* die gewählte upwind-Diskretisierung des Konvektionstermes ($\tilde{n}_h$ oder $\bar{n}_h$) bezeichnet.

Wie im vorhergehenden Abschnitt beschrieben, konstruieren wir für die Level $k = 1, \ldots, \ell$ durch sukzessive Verfeinerung eine Folge von Gittern $\mathbf{T}_{h_k}$ mit $h_{k+1} \approx h_k/2$. Unser Ziel besteht darin, das System (6.1) auf dem feinsten Gitterlevel ℓ zu lösen. Das Newton-Verfahren ist in diesem Zusammenhang nicht zu empfehlen, da es insbesondere im Fall hoher Reynolds-Zahlen auf lineare Probleme mit ungünstigen Matrixeigenschaften führt, so daß sich ein erhöhter Lösungsaufwand erforderlich macht. Deshalb benutzen wir als äußere Iteration eine einfache Fixpunktiteration, bei der eine Linearisierung der Form

$$L(\mathbf{y}^{n+1})\,\mathbf{y}^{n+1} \;\approx\; L(\mathbf{y}^{n})\,\mathbf{y}^{n+1}$$

verwendet wird. Die linearen Teilprobleme lösen wir mit Hilfe eines Mehrgitterverfahrens. Im folgenden beschreiben wir einen Schritt der Fixpunktiteration $\mathbf{y}^n \to \mathbf{y}^{n+1}$. Dabei bedeutet der Index k in den Bezeichnungen L_k und $\mathbf{y}_k$ das jeweilige Gitterlevel, auf dem diese Größen definiert sind. Für das feinste Gitterlevel $k = \ell$ wird dieser Index weggelassen.

<u>NICHTLINEARE ITERATION: $\mathbf{y}^{n+1} = \mathrm{NLI}\,(m_0, q_0, \omega, \mathbf{y}^n)$</u>

(a) Generiere zur alten Iterierten $\mathbf{y}^n$ die Steifigkeitsmatrizen

$$L := L(\mathbf{y}^n) \tag{6.3}$$

auf dem feinsten Gitter sowie

$$L_k := L_k(\mathbf{y}_k^n)\,, \qquad k = \ell-1,\ldots,1\,,$$

auf den gröberen Gittern, wobei $\mathbf{y}_k^n$ durch Restriktion von $\mathbf{y}_{k+1}^n$ berechnet wird ausgehend von $\mathbf{y}_\ell^n{=}\mathbf{y}^n$.

(b) Löse näherungsweise das lineare Problem

$$L\,\hat{\mathbf{y}}^{n+1} = \mathbf{F} \tag{6.4}$$

mit Hilfe von m Mehrgitter-Zyklen auf dem Level ℓ

$$\mathbf{z}^i = \mathrm{MGI}\,(\mathbf{z}^{i-1}, L, \mathbf{F}, \ell)\,, \quad i = 1,\ldots,m\,,$$

beginnend mit $\mathbf{z}^0{:=}\mathbf{y}^n$ unter Verwendung der Grobgittermatrizen L_k, $k = 1, \ldots, \ell-1$, aus Schritt (a). Die Anzahl m der Mehrgitter-Zyklen wird dynamisch gesteuert durch eine obere Schranke m_0 und eine vorgegebene Dämpfung q_0 des Anfangsdefektes, d.h. wir beenden die Mehrgitter-Iteration nach m Schritten, falls in einer geeigneten Norm $\|\cdot\|$ gilt

$$\frac{\|L\mathbf{z}^m - \mathbf{F}\|}{\|L\mathbf{z}^0 - \mathbf{F}\|} \;\le\; q_0 \;<\; 1 \tag{6.5}$$

oder falls $m{=}m_0$. Wenn $\|L\mathbf{z}^0 - \mathbf{F}\|$ hinreichend klein ist, wird die nichtlineare Iteration mit der Lösung $\mathbf{y}^n$ abgebrochen.

(c) Berechne die neue Iterierte $\mathbf{y}^{n+1}$ aus der gedämpften Korrektur

$$\mathbf{y}^{n+1} \;=\; \mathbf{y}^n \;+\; \omega(\tilde{\mathbf{y}}^{n+1} - \mathbf{y}^n)\,, \tag{6.6}$$

bei der $\tilde{\mathbf{y}}^{n+1}{:=}\mathbf{z}^m$ das Ergebnis aus Schritt (b) ist.

In unseren numerischen Testrechnungen haben wir die Parameterkonstellation $m_0 = 5$, $q_0 = 0.1$, $\omega = 0.9$ verwendet und die gewichtete Euklidische Norm $\|\mathbf{d}\| := (|\mathbf{d}_u|^2 + h^2|\mathbf{d}_p|^2)^{1/2}$ für Vektoren $\mathbf{d} = (\mathbf{d}_u, \mathbf{d}_p)^\mathsf{T} \in \mathbf{R}^{N_u+N_p}$ benutzt.

Sei $M(\mathbf{y}^n)$ die Matrix, so daß $M(\mathbf{y}^n)\mathbf{d}$ das Ergebnis von m Mehrgitter-Iterationen mit dem Startvektor $\mathbf{w}^0 = 0$ angewendet auf das System $L(\mathbf{y}^n)\mathbf{w} = \mathbf{d}$ ist. Dann läßt sich die oben beschriebene Fixpunktiteration in der Form

$$\mathbf{y}^{n+1} \;=\; \mathbf{y}^n \;-\; \omega M(\mathbf{y}^n)(L(\mathbf{y}^n)\mathbf{y}^n - \mathbf{F}) \tag{6.7}$$

schreiben. In vielen Fällen kann man das Konvergenzverhalten noch verbessern, indem man den Dämpfungsparameter ω in (6.7) in jedem Schritt neu durch Minimierung des Defektes der Näherung $\mathbf{y}^{n+1}(\omega) = \mathbf{y}^n + \omega(\tilde{\mathbf{y}}^{n+1} - \mathbf{y}^n)$ bestimmt [24], [67].

6.3 Komponenten des Mehrgitterlösers

In diesem Abschnitt beschreiben wir den Mehrgitter-Algorithmus MGI zur Lösung der linearen Teilprobleme innerhalb der äußeren nichtlinearen Iteration. Dabei beschränken wir uns auf die Angabe seiner Komponenten (für eine Beschreibung der Mehrgittermethode verweisen wir auf [27], [74]).

Als Mehrgitter-Zyklus wählen wir den F-Zyklus. Er stellt einen guten Kompromiß zwischen dem V- und W-Zyklus dar, da er einerseits vom Aufwand her nicht viel teurer als der V-Zyklus ist und andererseits fast die Konvergenzrate des W-Zyklus erreicht. Bei der Grobgitterkorrektur dämpfen wir den berechneten Korrekturvektor mit einem Faktor $\alpha \in (0,1)$. In unseren numerischen Testrechnungen haben wir den festen Wert $\alpha = 0.8$ verwendet. Man kann diesen Faktor jedoch auch jedesmal neu durch Minimierung eines Funktionals bestimmen und erhält in gewissen Fällen trotz des entstehenden Mehraufwandes ein schnelleres Verfahren [5], [24], [51].

6.3.1 Glätter

Da in der Systemmatrix L in (6.2) der untere Hauptdiagonalblock aus lauter Nullen besteht, können die sonst üblichen Glätter wie das Jacobi-, Gauß-Seidel-, SOR- oder ILU-Verfahren nicht eingesetzt werden. In der Literatur wurden daher spezielle Glätter vorgeschlagen wie etwa das DGS-Verfahren [6], das SIMPLE-Verfahren [47, 60], oder transformierende ILU-Glätter [75, 76, 50]. Wir verwenden hier einen blockweisen Gauß-Seidel-Glätter, der von Vanka [71] für ein Finite-Differenzen-Verfahren vorgeschlagen und in [52] auf den Fall der nichtkonformen Crouzeix-Raviart-Elemente übertragen wurde. Ein Vorteil dieses Glättungsverfahrens ist, daß es sich ohne größere Verluste an numerischer Effizienz bis hin zum massiv-parallelen Fall gut parallelisieren läßt.

Unser Glätter ist eine iterative Methode zur Lösung des Gleichungssystems

$$L\mathbf{z} \;=\; \begin{pmatrix} A & B \\ B^\mathsf{T} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ \mathbf{q} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix} \;=\; \mathbf{F} \tag{6.8}$$

für den Vektor $\mathbf{z}=(\mathbf{v},\mathbf{q})^\mathsf{T}$ auf dem jeweils aktuellen Gitterlevel k. Dabei lassen wir in unserer Notation den Index "k" zur Kennzeichnung des Gitterlevels weg. Die Idee des Vanka-Glätters besteht darin, durch alle Elemente des aktuellen Gitters zu laufen und für jedes Element jeweils die dazugehörigen Unbekannten durch Lösung eines zugeordneten lokalen Gleichungssystems neu zu berechnen.

Für eine genauere Beschreibung unseres Glätters benötigen wir einige Bezeichnungen. Zu einem gegebenen globalen Vektor $\mathbf{z}=(\mathbf{v},\mathbf{q})^\mathsf{T}$ und einem Element $K\in\mathbf{T}_h$ sei $\mathbf{z}_K=(\mathbf{v}_K,\mathbf{q}_K)^\mathsf{T}$ der Teilvektor aus denjenigen Komponenten von $\mathbf{z}$, die zu Unbekannten auf dem Element K gehören. Im zweidimensionalen Fall besteht $\mathbf{v}_K = (v_{1j_1}, v_{1j_2}, v_{1j_3}, v_{1j_4}, v_{2j_1}, v_{2j_2}, v_{2j_3}, v_{2j_4})^\mathsf{T}$ aus den 8 Geschwindigkeitswerten zu den vier Seitenmittenknoten von K und $\mathbf{q}_K = q_m$ aus dem einen konstanten Druckwert auf K (siehe Abbildung 6.4).

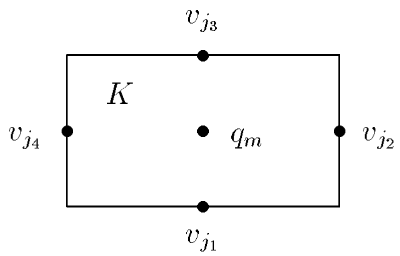


Abbildung 6.4: *Unbekannte zum Element K*

Zur Matrix A in (6.8) definieren wir für ein gegebenes Element K die "lokale Matrix" A_K als diejenige Teilmatrix von A, die entsteht, wenn wir aus A nur solche Zeilen und Spalten auswählen, die den Unbekannten von K entsprechen. Analog sei B_K die entsprechende Teilmatrix von B. Da zu einem Element nur ein einziger Druckwert gehört, ist B_K lediglich ein Spaltenvektor. Der Matrix L ordnen wir für $K\in\mathbf{T}_h$ die folgende "modifizierte lokale Systemmatrix" $\widetilde{L}_K$ zu

$$\widetilde{L}_K \;:=\; \begin{pmatrix} D_K & B_K \\ B_K^\mathsf{T} & 0 \end{pmatrix} \qquad \text{mit} \qquad D_K \;:=\; \mathrm{diag}(A_K)\,. \tag{6.9}$$

Weiterhin definieren wir zu einem gegebenen globalen Vektor $\mathbf{z}$ einen "modifizierten lokalen rechte-Seite-Vektor" $\widetilde{\mathbf{F}}_K(\mathbf{z})$ als

$$\widetilde{\mathbf{F}}_K(\mathbf{z}) \;:=\; \mathbf{F}_K \,-\, (L\mathbf{z})_K \,+\, \widetilde{L}_K\mathbf{z}_K\,, \tag{6.10}$$

wobei $\mathbf{F}_K$ und $(L\mathbf{z})_K$ analog zur obigen Definition von $\mathbf{z}_K$ die entsprechenden Teilvektoren der globalen Vektoren $\mathbf{F}$ und $L\mathbf{z}$ bedeuten. Nach diesen Vorbereitungen können wir einen Schritt des Vanka-Glätters $\mathbf{z}^{\mathrm{old}} \to \mathbf{z}^{\mathrm{new}}$ folgendermaßen beschreiben.

GLÄTTUNGSITERATION: $\mathbf{z}^{\text{old}} \to \mathbf{z}^{\text{new}}$

(a) initialisiere $\mathbf{z} := \mathbf{z}^{\text{old}}$;
for $K \in \mathbf{T}_h$ **do**
(1) löse das lokale System:

$$\widetilde{L}_K \widetilde{\mathbf{z}}_K \;=\; \widetilde{\mathbf{F}}_K(\mathbf{z})\;;$$

(2) update $\widetilde{\mathbf{z}}_K \to \mathbf{z}$:
d.h. aktualisiere den Teilvektor $\mathbf{z}_K$ von $\mathbf{z}$ gemäß
$\mathbf{z}_K := \widetilde{\mathbf{z}}_K$;
enddo

(b) $\mathbf{z}^{\text{new}} \;:=\; \mathbf{z}^{\text{old}} + \omega_S(\mathbf{z} - \mathbf{z}^{\text{old}})$;

Die Matrix $\widetilde{L}_K$ ist stets regulär und hat eine sehr einfache Struktur, so daß das lokale System im Schritt (a1) mit wenigen arithmetischen Operationen exakt gelöst werden kann. In unseren numerischen Testrechnungen haben wir für den Relaxationsschritt (b) den Wert $\omega_S = 0.9$ verwendet. Innerhalb des Mehrgitter-Verfahrens führen wir jeweils zwei Vorglättungsschritte und zwei Nachglättungsschritte aus.

6.3.2 Gittertransfer

Zur Erklärung des Gittertransfers verwenden wir im folgenden den Index "h" zur Bezeichnung von Größen, die zum feinen Gitter auf dem Level $k+1$ gehören, und den Index "H" für Größen auf dem groben Gitterlevel k.

Prolongation

Für die Definition des Prolongationsoperators $P : V_H \times Q_H \to V_h \times Q_h$ haben wir zu einer vorgegebenen Grobgitterfunktion $v_H \in V_H$ bzw. $q_H \in Q_H$ jeweils eine geeignete Feingitterfunktion $v_h \in V_h$ bzw. $q_h \in Q_h$ zu erklären.

Wir beginnen mit der Druckfunktion q_h. Sei $\kappa \in \mathbf{T}_h$ ein beliebiges Element des feinen Gitters und $K \in T_H$ mit $\kappa \subset K$ das zugehörige Vaterelement auf dem groben Gitter. Dann definieren wir den Wert der stückweise konstanten Funktion q_h auf κ als

$$q_h|_\kappa \;:=\; q_H|_K\,. \tag{6.11}$$

Die Geschwindigkeitsfunktion v_h wird festgelegt, indem wir die Knotenwerte $v_j^h \in \mathbf{R}^d$ in der Darstellung

$$v_h \;=\; \sum_{j\in J_h(\Omega)} v_j^h \phi_j^h$$

definieren. Dabei bezeichnen $J_h(\Omega)$ die Indexmenge der Feingitterknoten und ϕ_j^h die skalaren Basisfunktionen von V_h (siehe Abschnitt 2.2). Für den Fall eines Randknotens $j \in J_h(\partial\Omega)$ setzen wir $v_j^h := 0$. Sei B_j, $j \in J_h(\Omega) \setminus J_h(\partial\Omega)$, ein innerer Knoten und seien $\kappa, \kappa' \in \mathbf{T}_h$ die zwei eindeutig bestimmten Elemente, zu denen der Knoten B_j gehört. Haben die Elemente κ und κ' zwei unterschiedliche Vaterelemente $K, K' \in \mathbf{T}_H$ (siehe Abbildung 6.5), so hat die Grobgitterfunktion v_H im allgemeinen überall Sprünge entlang der gemeinsamen Seite Γ_j der Elemente κ und κ'. Damit sind die Werte der

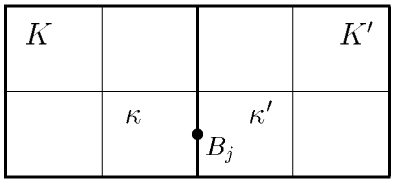


Abbildung 6.5: *Feingitterelemente κ und κ' zum Knoten B_j und zugehörige Vaterelemente K und K'*

lokalen Knotenfunktionale $N_j^\kappa(v_H)$ und $N_j^{\kappa'}(v_H)$ (siehe Definition 2.5) im allgemeinen unterschiedlich und wir definieren den Knotenwert v_j^h als arithmetisches Mittel

$$v_j^h \;:=\; \frac{1}{2}\left(N_j^\kappa(v_H) \;+\; N_j^{\kappa'}(v_H)\right) . \tag{6.12}$$

Falls die Elemente κ und κ' ein gemeinsames Vaterelement $K \in \mathbf{T}_H$ besitzen, braucht man zur Berechnung von v_j^h nur ein Knotenfunktional auszuwerten, da dann die beiden Werte $N_j^\kappa(v_H)$ und $N_j^{\kappa'}(v_H)$ identisch sind.

Restriktion

Wir benötigen zwei Restriktionsoperatoren

$$R^* : V_h^* \times Q_h^* \to V_H^* \times Q_H^* \qquad \text{und} \qquad R : V_h \times Q_h \to V_H \times Q_H \,,$$

wobei R^* Defektfunktionale vom feinen auf das grobe Gitter transformiert und R bei der Generierung der Grobgittermatrizen im Schritt (a) der nichtlinearen Iteration erforderlich ist.

Zur Definition von R^* verwenden wir den oben eingeführten Prolongationsoperator $P : V_H \times Q_H \to V_h \times Q_h$. Für ein gegebenes Defektfunktional $d_h \in V_h^* \times Q_h^*$ auf dem feinen Gitter erklären wir das zugeordnete Grobgitterfunktional $d_H = R^* d_h$ durch

$$\langle d_H, \theta_H \rangle := \langle d_h, P\theta_H \rangle \qquad \forall\, \theta_H \in V_H \times Q_H \,. \tag{6.13}$$

Wir kommen nun zur Definition von R. Zu einer vorgegebenen Geschwindigkeitsfunktion $v_h \in V_h$ erklären wir die zugeordnete Grobgitterfunktion $v_H \in V_H$ durch die Angabe der Knotenwerte $v_j^H \in \mathbf{R}^d$ in der Darstellung

$$v_H \;=\; \sum_{j \in J_H(\Omega)} v_j^H \phi_j^H \,,$$

bei der $J_H(\Omega)$ die Indexmenge der Grobgitterknoten und ϕ_j^H die skalaren Basisfunktionen von V_H bezeichnen. Für einen Randknoten $j \in J_H(\partial\Omega)$ setzen wir $v_j^H := 0$. Sei B_j, $j \in J_H(\Omega) \setminus J_H(\partial\Omega)$, ein innerer Grobgitterknoten und seien $\kappa_i \in \mathbf{T}_h$, $i = 1, \dots, 2^d$, diejenigen Feingitterelemente, die den Knoten B_j als Ecke besitzen (siehe Abbildung 6.6 für den zweidimensionalen Fall d=2). Dann definieren wir den Knotenwert v_j^H durch

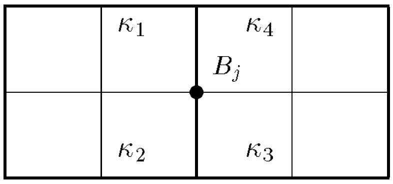


Abbildung 6.6: *Feingitterelemente $\kappa_1, \dots, \kappa_4$ zum Grobgitterknoten B_j*

die folgende Mittelung

$$v_j^H \;:=\; 2^{-d} \sum_{i=1}^{2^d} v_h|_{\kappa_i}(B_j)\,. \tag{6.14}$$

Wir wollen nun einer vorgegebenen Druckfunktion $q_h \in Q_h$ eine Grobgitterfunktion $q_H \in Q_H$ zuordnen. Dazu betrachten wir ein beliebiges Grobgitterelement $K \in \mathbf{T}_H$, welches auf dem feinen Gitter in die Sohnelemente $\kappa_i \in \mathbf{T}_h$, $i = 1, \dots, 2^d$, zerlegt sei. Den Wert der stückweise konstanten Funktion q_H auf K definieren wir dann als

$$q_H|_K \;:=\; 2^{-d} \sum_{i=1}^{2^d} q_h|_{\kappa_i}\,. \tag{6.15}$$

6.3.3 Grobgitterlöser

In einem F-Zyklus mit dem feinsten Gitterlevel ℓ gelangt man $(\ell-1)$ mal auf das gröbste Level $k = 1$, wo jeweils ein lineares Problem der Struktur

$$\begin{pmatrix} A & B \\ B^{\mathsf{T}} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ \mathbf{q} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}, \tag{6.16}$$

zu lösen ist. In unserer Bezeichnung lassen wir den Index "1" zur Kennzeichnung des Gitterlevels $k = 1$ weg. Die Matrix A=$A(\mathbf{u}^{old})$ wird zu Beginn eines jeden nichtlinearen Schrittes mit Hilfe der alten Geschwindigkeitsnäherung $\mathbf{u}^{old}$ auf Level $k = 1$ generiert und bleibt dann für alle m F-Zyklen, die innerhalb des nichtlinearen Schrittes ausgeführt werden, konstant. Wenden wir also zur Lösung von (6.16) eine Methode an, die eine vollständige oder unvollständige Faktorisierung der Systemmatrix oder ihrer Teilblöcke beinhaltet, so braucht diese Faktorisierung nur einmal zu Beginn des nichtlinearen Schrittes berechnet zu werden und kann dann mehrfach bei der Lösung der Grobgitterprobleme benutzt werden.

Um eine Faktorisierung der gesamten Steifigkeitsmatrix zu umgehen, verwenden wir zur näherungsweisen Lösung des Systems (6.16) den von Fortin/Glowinski [20] vorgeschlagenen *augmented-Lagrange-Algorithmus*. Hierfür benötigen wir lediglich die *LU*-Faktorisierung einer Matrix C von der Größe der Blockmatrix A. Der augmented-Lagrange-Algorithmus ist ein iteratives Verfahren. Da wir mit (6.16) ein System von Defektgleichungen zu lösen haben, beginnen wir mit der Startnäherung $\mathbf{v}^0=0$ und $\mathbf{q}^0=0$ und berechnen die jeweils nächste Iterierte nach folgender Vorschrift:

<u>AUGMENTED LAGRANGE ITERATION: $(\mathbf{v}^i, \mathbf{q}^i) \to (\mathbf{v}^{i+1}, \mathbf{q}^{i+1})$</u>

(a) Berechne $\mathbf{v}^{i+1}$ aus

$$(A + rBB^{\mathsf{T}})\mathbf{v}^{i+1} = (\mathbf{f} + rB\mathbf{g}) - B\mathbf{q}^i\,. \tag{6.17}$$

(b) Berechne $\mathbf{q}^{i+1}$ aus

$$\mathbf{q}^{i+1} = \mathbf{q}^i + \rho(B^{\mathsf{T}}\mathbf{v}^{i+1} - \mathbf{g})\,. \tag{6.18}$$

Analog zur Mehrgitteriteration innerhalb eines nichtlinearen Schrittes beenden wir diese Iteration, falls der Anfangsdefekt um einen vorgegebenen Faktor $q_1 \in (0,1)$ gedämpft wurde (siehe (6.5)) oder falls eine maximale Schrittzahl m_1 erreicht ist. In unseren Rechnungen haben wir die Werte $m_1=30$ und $q_1=0.1$ verwendet. Die Konvergenzeigenschaften der augmented Lagrange Iteration werden durch die Parameter r und ρ bestimmt. In unseren numerischen Testrechnungen stellten sich die Werte $r=200$ und $\rho=150$ als eine geeignete Parameterwahl heraus.

Im Schritt (a) der augmented Lagrange Iteration haben wir während eines gesamten nichtlinearen Schrittes stets ein Problem mit derselben Matrix $C=A + rBB^{\mathsf{T}}$ zu lösen. Deshalb wird zu Beginn jeder nichtlinearen Iteration (im Schritt (a) der Prozedur NLI, siehe Abschnitt 6.2) eine *LU*-Faktorisierung der Matrix C berechnet. Wenn wir im Durchschnitt m_a augmented Lagrange Iterationen zur Lösung eines Grobgitterproblems (6.16) benötigen und m F-Zyklen innerhalb eines nichtlinearen Schrittes ausführen, so können wir diese Faktorisierung $m_a(\ell - 1)m$ mal benutzen.

Kapitel 7

Parallelisierung

Auf Grund der hohen Dimension des entstehenden nichtlinearen Gleichungssystems stößt man bei der numerischen Lösung der Navier-Stokes Gleichungen sehr schnell an die Grenzen der Rechenleistung und Speicherplatzkapazität herkömmlicher Rechner. Eine Alternative bieten Parallelrechner, bei denen man hinreichend viele Prozessoren zur Verfügung hat, die zur gleichen Zeit rechnen oder Daten austauschen können und eigenen Speicherplatz besitzen. Das Problem besteht darin, einen parallelen Algorithmus zu finden, der einerseits eine gute numerische Effizienz hat und bei dem andererseits die Kommunikations- und Wartezeiten nicht zu hoch sind. In der CFD-Literatur ist die Datenpartitionierung auf der Basis einer Gebietszerlegung die vorherrschende Technik zur Konstruktion paralleler Algorithmen ([43],[35],[31], [72]).

Die Grundlage unserer Parallelisierungsstrategie ist ebenfalls eine Datenpartitionierung. Dabei sind die Teilgebiete, die den Prozessoren zugeordnet werden, die Makroelemente des gröbsten Gitters. Außerdem verwenden wir noch einen zusätzlichen Prozessor, der für das Lösen der Grobgitterprobleme zuständig ist. Von der numerischen Seite her unterscheidet sich unser paralleler Algorithmus von dem sequentiellen Mehrgitter-Algorithmus in Kapitel 6 nur durch eine Modifikation im Glättungsverfahren. In Kapitel 8 werden wir anhand numerischer Testrechnungen zeigen, daß dieser algorithmische Unterschied selbst bei einer hohen Anzahl von Makroelementen lediglich eine geringfügige Verschlechterung in der Konvergenzrate des Mehrgitterverfahrens bewirkt. Unser paralleler Navier-Stokes-Löser hat also bis hin zum massiv-parallelen Fall eine gute numerische Effizienz. Um eine gute parallele Effizienz zu erreichen, ist es wichtig, daß die erforderlichen Kommunikationsoperationen effizient umgesetzt werden. In den Abschnitten 7.3 und 7.4 behandeln wir, wie man die lokale und globale Kommunikation möglichst optimal realisieren kann. Dabei profitieren wir bei der lokalen Kommunikation von der Eigenschaft unserer nichtkonformen Elemente, daß jede Unbekannte zu höchstens zwei Elementen gehört.

7.1 Parallelisierungsstrategie

Unser Parallelisierungsansatz beruht auf einer Datenpartitionierung entsprechend der Zerlegung des Gebietes in Makroelemente. Dies bedeutet, daß wir sämtliche Daten und Operationen eines globalen Algorithmus zur Lösung des globalen Gleichungssystems $L(\mathbf{y})\mathbf{y} = \mathbf{F}$ so auf verschiedene Tasks verteilen, daß jeweils eine Task für ein Makroelement zuständig ist. Die Tasks werden dann einer gegebenen Menge zur Verfügung stehender Prozessoren zugeordnet. Sind genügend viele Prozessoren vorhanden, so verwenden wir für jede Task einen anderen Prozessor. Ansonsten haben wir ein Lastverteilungsproblem zu lösen, um eine geeignete Abbildung der Tasks auf die Prozessoren zu finden.

Bei der Wahl des Algorithmus sind wir bestrebt, soweit wie möglich unseren sequentiellen Mehrgitter-Algorithmus aus Kapitel 6 zu verwenden, um eine möglichst gute numerische Effizienz zu erreichen. Modifikationen sind nur erforderlich, um rekursive Abhängigkeiten zwischen den Daten der einzelnen Makroelemente zu vermeiden, da diese zu großen Wartezeiten zwischen den Tasks führen würden. Solche rekursiven Datenabhängigkeiten kommen im Grobgitterlöser und in unserem blockweisen Gauß-Seidel Glätter vor. Für die Lösung der Grobgitterprobleme verwenden wir eine zusätzliche Solver-Task, so daß alle voneinander abhängigen Daten in einer Task konzentriert sind und keine zusätzliche Kommunikation erforderlich ist. Den Glätter modifizieren wir, indem wir die sequentielle Schleife über alle Elemente des Gebietes ersetzen durch mehrere parallele Schleifen über die Elemente in den Makroelementen mit einem nachfolgenden Datenabgleich an den gemeinsamen Rändern der Makroelemente (siehe Abschnitt 7.2). Von der numerischen Seite her (d.h. wenn wir von der zusätzlichen Kommunikation im parallelen Löser absehen) sind der sequentielle und der parallele Mehrgitter-Algorithmus bis auf den Glätter in allen Komponenten identisch.

Zur Beschreibung der Datenpartitionierung führen wir zunächst einige Bezeichnungen ein, welche sinngemäß auf jedem Gitterlevel $k = 1, \dots, \ell$, verwendet werden, ohne den Index "k" explizit anzugeben. Wir vereinbaren, daß Randpunkte $x \in \partial M_i$ mit zum Makroelement M_i gehören. Sei $I(\Omega) := \{1, \dots, N\}$ die Indexmenge des globalen Lösungsvektors $\mathbf{y} = (\mathbf{u}, \mathbf{p})^\mathsf{T} \in \mathbf{R}^N$ und $I(M_i)$ für ein Makroelement M_i, $i = 1, \dots, N_m$, die Teilmenge der Indizes $\mu \in I(\Omega)$, bei denen die Komponente $\mathbf{y}_\mu$ zu M_i gehört (d.h. entweder einer Geschwindigkeitskomponente in einem Knoten $B_j \in M_i$ oder einem Druckwert in einem Element $K \subset M_i$ entspricht). Die Datenpartitionierung $I(\Omega) = \bigcup_i I(M_i)$ ist nicht disjunkt, denn die Geschwindigkeitswerte zu einem Knoten auf der gemeinsamen Seite Γ^{ij} zweier benachbarter Makroelemente M_i und M_j sind jeweils durch einen Index $\mu \in I(M_i) \cap I(M_j)$ erfaßt.
Sei $\tau_i : \{1, \dots, N_i\} \to I(M_i)$ mit $N_i := \#I(M_i)$ eine bijektive Abbildung, die für eine "lokale" Numerierung der Unbekannten auf M_i sorgt. Dann definieren wir zu einem beliebigen globalen Vektor $\mathbf{w} \in \mathbf{R}^N$ für ein gegebenes Makroelement M_i den Vektor $\mathbf{w}^{(i)} \in \mathbf{R}^{N_i}$ als den Teilvektor von $\mathbf{w}$, der aus den zu M_i gehörenden Komponenten besteht, d.h.

$$\mathbf{w}_l^{(i)} \;:=\; \mathbf{w}_{\tau_i(l)} \qquad \forall\, l \in \{1, \dots, N_i\}\,. \tag{7.1}$$

Ebenso können wir der globalen Matrix $L := L(\mathbf{y}^{\mathrm{old}}) \in \mathbf{R}^{N \times N}$ (siehe (6.2)) die zu M_i gehörende Teilmatrix $L^{(i)} \in \mathbf{R}^{N_i \times N_i}$ mit

$$L^{(i)}_{lm} \;:=\; L_{\tau_i(l)\tau_i(m)} \qquad \forall\, l, m \in \{1, \dots, N_i\}\,. \tag{7.2}$$

zuordnen.

Um den Algorithmus zur Lösung des globalen Gleichungssystems zu parallelisieren, konstruieren wir zur Zerlegung des Gebietes Ω in Makroelemente $\{M_i\}$ ein topologisch äquivalentes Tasknetzwerk $\{T_i\}$ (siehe Abbildung 7.1). Dies bedeutet, daß bei der Abbildung $M_i \to T_i$ zwei benachbarten Makroelementen M_i und M_j, die eine gemeinsame Randseite Γ^{ij} besitzen, auch zwei "benachbarte" Tasks T_i und T_j zugeordnet werden, welche durch einen virtuellen Kanal c_{ij} verbunden sind. Im folgenden bezeichnen wir mit $\mathcal{N}(i)$ die Menge aller Indizes $j \neq i$, für die die Makroelemente M_i und M_j eine gemeinsame Seite $\Gamma^{ij} = \partial M_i \cap \partial M_j$ haben. Dem Tasknetzwerk $\{T_i\}$ fügen wir noch eine Solver-Task S hinzu, die für die Lösung der globalen Grobgitterprobleme auf dem Level $k = 1$ zuständig ist. Auf jedem Gitterlevel werden nun die Größen $L := L(\mathbf{y}^{\mathrm{old}})$, $\mathbf{y}$ und $\mathbf{F}$ des globalen Gleichungssystems auf die Tasks $\{T_i\}$ aufgeteilt, d.h. im lokalen Speicher von T_i werden lediglich die zu M_i gehörenden "lokalen" Größen $L^{(i)}$, $\mathbf{y}^{(i)}$ und $\mathbf{F}^{(i)}$ gehalten. Die Task S speichert dagegen die zum gesamten Gebiet Ω gehörenden globalen Größen für das Gitterlevel $k = 1$.

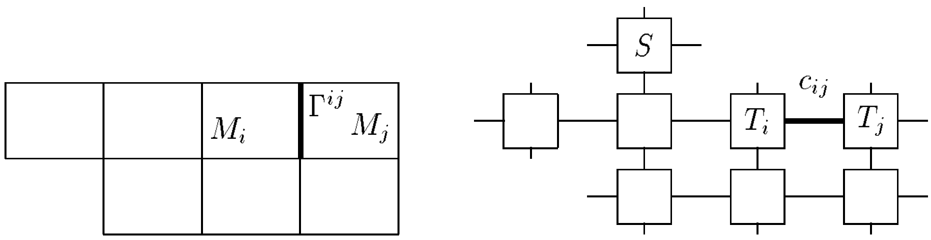


Abbildung 7.1: *Makroelement-Zerlegung $\{M_i\}$ und zugeordnetes Tasknetzwerk $\{T_i\}$ mit Solver-Task S*

Neben der Aufteilung der Daten erfolgt auch eine Aufteilung der arithmetischen Operationen des globalen Algorithmus auf die Tasks. Das Ziel der Task T_i besteht darin, den zum Makroelement M_i gehörenden Teilvektor $\mathbf{y}^{(i)}$ des globalen Lösungsvektors $\mathbf{y}$ zu bestimmen. Bei der parallelen Lösung des globalen Systems werden die Tasks $\{T_i\}$ gleichzeitig abgearbeitet, wobei zusätzlich die Solver-Task S zum Lösen der Grobgitterprobleme ausgeführt wird. Die Wirkungsweise einer Task T_i kann folgendermaßen beschrieben werden.

CHARAKTERISIERUNG DER TASK T_i:

(1) T_i ist zuständig für diejenigen arithmetischen Operationen des globalen Algorithmus, welche die Daten von M_i betreffen.

(2) Um Operationen mit Daten ausführen zu können, die zum Rand von M_i gehören, tauscht T_i Daten mit den lokalen Nachbartasks T_j, $j \in \mathcal{N}(i)$, über die virtuellen Kanäle c_{ij} aus. Wir nennen diese Art des Datenaustausches *lokale Kommunikation.*

(3) Um Operationen mit globalen Daten ausführen zu können (wie z.B. die Berechnung einer globalen Defektnorm oder das Lösen eines Grobgitterproblems), beteiligt sich T_i an einem globalen Informationsaustausch über das gesamte Tasknetzwerk $\{T_j\}$, bei dem am Ende jede Task die globale Information erhält. Wir nennen diese Art des Datenaustausches *globale Kommunikation.*

Eine genauere Erläuterung der lokalen und globalen Kommunikation erfolgt in den Abschnitten 7.3 und 7.4. Im folgenden wollen wir etwas näher auf die Lösung der Grobgitterprobleme eingehen.

Gehen wir davon aus, daß ein Grobgitterproblem eine kleine Größenordnung hat, so wäre es uneffektiv, die Lösung dieses Problems auf alle Tasks T_i zu verteilen, da sich dabei ein hoher Anteil an Kommunikationszeiten ergeben würde. Außerdem ist es auf Grund der globalen Datenabhängigkeit innerhalb des Grobgitterlösers angebracht, die Lösung der Grobgitterprobleme in einer Task ausführen zu lassen. Dies könnte in einer zusätzlichen Solver-Task S erfolgen, in einer ausgewählten Task T_{i^*} oder auch redundant in jeder der Tasks T_i (denn während der Lösung des Grobgitterproblems müßten die nichtbeteiligten Tasks ohnehin warten). Da sich zwischen der Solver-Task S und dem Tasknetzwerk $\{T_i\}$ eine zusätzliche Parallelität ergibt, haben wir uns für die Variante mit der Solver-Task entschieden. Es muß jedoch erwähnt werden, daß die Nichtparallelisierung des Grobgitterlösers für eine wachsende Zahl von Prozessoren zu einem Absinken der parallelen Effizienz führen wird. Mit diesem Problem werden wir uns näher in Kapitel 8 auseinandersetzen. Im folgenden wollen wir die Arbeit der Solver-Task S beschreiben.

CHARAKTERISIERUNG DER SOLVER-TASK S:

(1) In einem Initialisierungsschritt werden alle erforderlichen globalen Daten zum Gitterlevel 1 generiert, insbesondere die Matrix B und der Stokes-Anteil der Matrix A.

(2) Während des Schrittes (a) der nichtlinearen Iteration NLI (siehe Abschnitt 6.2) wartet die Task S auf den Geschwindigkeitsvektor $\mathbf{u}^{\text{old}} := \mathbf{u}^n$ zum Level 1, der über globale Kommunikation vom Tasknetzwerk $\{T_i\}$ an S geschickt wird. Sobald dieser Vektor angekommen ist, beginnt S mit der Generierung der Matrizen $A = A(\mathbf{u}^{\text{old}})$ und $C = A + rBB^{\mathsf{T}}$ und berechnet danach die LU-Zerlegung von C, welche später innerhalb der augmented Lagrange Iteration verwendet wird.

(3) Innerhalb der Mehrgitter-Zyklen wartet die Task S darauf, vom Tasknetzwerk $\{T_i\}$ den rechte-Seite-Vektor eines Grobgitterproblems zu er-

halten. Nachdem dieser Vektor angekommen ist, löst S das Grobgitterproblem bis zu einer bestimmten Genauigkeit durch eine gewisse Anzahl von augmented Lagrange Iterationen und schickt die berechnete Lösung an das Tasknetzwerk $\{T_i\}$ zurück. Danach wartet S auf den nächsten Vektor vom Tasknetzwerk. Dabei teilt ein vorangestelltes Steuersignal der Task S mit, ob es sich bei diesem Vektor um den rechte-Seite-Vektor eines weiteren Grobgitterproblems oder um den Geschwindigkeitsvektor $\mathbf{u}^{\text{old}}$ des nächsten nichtlinearen Schrittes oder gar um das Ende des Algorithmus handelt.

Zwischen der Solver-Task S und den Makroelement-Tasks T_i besteht eine Parallelität in folgendem Sinn. Sobald der Geschwindigkeitsvektor $\mathbf{u}^{\text{old}}$ auf Level 1 vom Tasknetzwerk $\{T_i\}$ an die Solver-Task S geschickt wurde, können auf der einen Seite die Tasks T_i ihre "Arbeit" bis zum ersten Auftreten eines Grobgitterproblems fortsetzen und auf der anderen Seite kann zeitgleich die Task S die Matrizen A und C generieren sowie die LU-Zerlegung von C berechnen. In vielen Fällen (wenn die Anzahl der Makroelemente nicht zu groß und das feinste Gitterlevel ℓ nicht zu klein ist) brauchen die Tasks T_i somit nicht mehr auf die Fertigstellung der LU-Zerlegung durch die Task S warten.

7.2 Paralleler Glätter

Wir wollen einen Glättungsschritt für das globale System $L\mathbf{z} = \mathbf{F}$ definieren, der parallel in den Tasks T_i ausgeführt werden kann. Hierzu modifizieren wir den blockweisen Gauß-Seidel Glätter unseres sequentiellen Mehrgitter-Algorithmus (siehe Abschnitt 6.3.1) dahingehend, daß wir die sequentielle Schleife über alle Elemente des Gebietes ersetzen durch mehrere parallele Schleifen über die Elemente in den Makroelementen M_i. In einem Glättungsschritt $\mathbf{z}^{\text{old}} \to \mathbf{z}^{\text{new}}$ aktualisiert also jede Task T_i ihre "eigenen", zu M_i gehörenden Komponenten $\mathbf{z}_\mu^{\text{old}}$, $\mu \in I(M_i)$. Dabei produzieren Tasks T_i und T_j, die zu benachbarten Makroelementen M_i und M_j gehören, für die gemeinsamen Komponenten $\mu \in I(M_i) \cap I(M_j)$ zunächst unterschiedliche Werte $\mathbf{z}_\mu^{T_i}$ und $\mathbf{z}_\mu^{T_j}$. In einem lokalen Kommunikationsschritt tauschen T_i und T_j diese Werte aus und verwenden dann als gemeinsamen neuen Wert den Mittelwert

$$\mathbf{z}_\mu^{\text{new}} \;:=\; \frac{1}{2}(\mathbf{z}_\mu^{T_i} + \mathbf{z}_\mu^{T_j}) \qquad \forall\, \mu \in I(M_i) \cap I(M_j)\,.$$

Hierbei kommt ein Vorteil der nichtkonformen Finiten Elemente besonders zur Geltung, daß nämlich eine beliebige Unbekannte $\mathbf{z}_\mu$ zu höchstens zwei Elementen und damit auch nur zu höchstens zwei Makroelementen gehört. Im Vergleich zu konformen Finiten Elementen wird dadurch die lokale Kommunikation wesentlich vereinfacht (dies gilt insbesondere im 3D-Fall).

Für die Beschreibung des parallelen Glätters vereinbaren wir einige Bezeichnungen. Wir verwenden die in Abschnitt 7.1 eingeführte Notation, wonach zu einem globalen

Vektor $\mathbf{w} \in \mathbf{R}^N$ der Vektor $\mathbf{w}^{(i)} \in \mathbf{R}^{N_i}$ den Teilvektor aus den Komponenten von $\mathbf{w}$ bezeichnet, die zum Makroelement M_i gehören. Entsprechend sei $L^{(i)}$ die zu M_i gehörende Teilmatrix der globalen Matrix L. Während wir mit $\mathbf{w}^{(i)} \in \mathbf{R}^{N_i}$ die zu den Makroelementen M_i gehörenden "lokalen" Vektoren eines globalen Vektors $\mathbf{w} \in \mathbf{R}^N$ bezeichnen, benutzen wir die Notation $\mathbf{w}^i \in \mathbf{R}^{N_i}$ zur Kennzeichnung von unabhängig definierten "Makroelement-Vektoren", für die im allgemeinen kein zugehöriger globaler Vektor $\mathbf{w} \in \mathbf{R}^N$ existiert, so daß für alle $i = 1, \ldots, N_m$ gilt

$$\mathbf{w}_l^i \;=\; \mathbf{w}_{\tau_i(l)} \qquad \forall\, l \in \{1, \ldots, N_i\}$$

(siehe Abschnitt 7.1 zur Definition von τ_i). Analog zu Abschnitt 6.3.1 führen wir Teilvektoren und Teilmatrizen auf Elementebene ein. Sei $K \subset M_i$ ein gegebenes Element. Zu einem Makroelement-Vektor $\mathbf{w}^i \in \mathbf{R}^{N_i}$ bezeichnen wir dann mit $\mathbf{w}_K^i$ den Teilvektor aus denjenigen Komponenten von $\mathbf{w}^i$, die zu Unbekannten auf dem Element K gehören. Zur globalen Steifigkeitsmatrix L sei $\widetilde{L}_K$ die in Abschnitt 6.3.1 definierte "modifizierte lokale Systemmatrix" zum Element K.

Für einen gegebenen Makroelement-Vektor $\mathbf{z}^i = (\mathbf{v}^i, \mathbf{q}^i)^{\mathsf{T}} \in \mathbf{R}^{N_i}$ mit dem Geschwin-

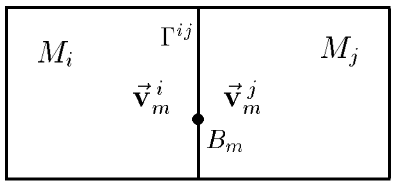


Abbildung 7.2: *Geschwindigkeitsvektoren $\vec{\mathbf{v}}_m^{\,i}$ und $\vec{\mathbf{v}}_m^{\,j}$ in den Tasks T_i und T_j zu einem Knoten B_m auf der gemeinsamen Seite Γ^{ij} der Makroelemente M_i und M_j*

digkeitsanteil $\mathbf{v}^i$ und dem Druckanteil $\mathbf{q}^i$ bezeichnen wir zu einem gegebenen Geschwindigkeitsknoten $B_m \in M_i$ mit $\vec{\mathbf{v}}_m^{\,i} \in \mathbf{R}^d$ den Teilvektor von $\mathbf{v}^i$, der den Geschwindigkeitskomponenten zum Knoten B_m entspricht. Zu einem Knoten B_m auf der gemeinsamen Seite Γ^{ij} zweier benachbarter Makroelemente M_i und M_j, $j \in \mathcal{N}(i)$, (siehe Abbildung 7.2) gibt es also einerseits den Geschwindigkeitsvektor $\vec{\mathbf{v}}_m^{\,i}$ in der Task T_i und andererseits den Vektor $\vec{\mathbf{v}}_m^{\,j}$ in der Task T_j.

In einem Schritt des parallelen Vanka-Glätters bestimmt die Task T_i aus einem gegebenen "lokalen" Vektor $(\mathbf{z}^{\text{old}})^{(i)}$ durch eigene Rechnung und durch Kommunikation mit den Nachbartasks T_j, $j \in \mathcal{N}(i)$, in folgender Weise einen neuen Vektor $(\mathbf{z}^{\text{new}})^{(i)}$.

Parallele Glättungsiteration in Task T_i : $\quad (\mathbf{z}^{\text{old}})^{(i)} \rightarrow (\mathbf{z}^{\text{new}})^{(i)}$

(a) bestimme mit Hilfe lokaler Kommunikation den Vektor

$$\mathbf{d}^i \;:=\; (L\mathbf{z}^{\text{old}})^{(i)} \,-\, L^{(i)}(\mathbf{z}^{\text{old}})^{(i)}\,; \tag{7.3}$$

(b) initialisiere $\mathbf{z}^i = (\mathbf{v}^i, \mathbf{q}^i)^\mathsf{T}$ als: $\quad \mathbf{z}^i := (\mathbf{z}^{\mathrm{old}})^{(i)}$;
for $K \subset M_i\,, K \in \mathbf{T}_h$ **do**
(1) löse das lokale System:

$$\widetilde{L}_K \widetilde{\mathbf{z}}_K \;=\; \widetilde{\mathbf{F}}_K \;;$$

mit

$$\widetilde{\mathbf{F}}_K \;:=\; \mathbf{F}_K^{(i)} \;-\; \mathbf{d}_K^i \;-\; (L^{(i)}\mathbf{z}^i)_K \;+\; \widetilde{L}_K \mathbf{z}_K^i$$

(2) update $\widetilde{\mathbf{z}}_K \to \mathbf{z}^i$:
d.h. aktualisiere den Teilvektor $\mathbf{z}_K^i$ von $\mathbf{z}^i$ gemäß
$\mathbf{z}_K^i := \widetilde{\mathbf{z}}_K$;
enddo

(c) **for** $j \in \mathcal{N}(i)$ **do**
(1) tausche über lokale Kommunikation mit der Task T_j zu allen Knoten B_m auf der Makroseite $\Gamma^{ij} := \partial M_i \cap \partial M_j$ die Geschwindigkeitsvektoren $\vec{\mathbf{v}}_m^{\,i}$ und $\vec{\mathbf{v}}_m^{\,j}$ aus ;
(2) berechne für alle Knoten $B_m \in \Gamma^{ij}$

$$\vec{\mathbf{v}}_m^{\,i} \;:=\; \frac{1}{2}(\,\vec{\mathbf{v}}_m^{\,i} \;+\; \vec{\mathbf{v}}_m^{\,j}\,)\,;$$

enddo

(d) $(\mathbf{z}^{\mathrm{new}})^{(i)} \;:=\; (\mathbf{z}^{\mathrm{old}})^{(i)} \;+\; \omega_S\Big(\mathbf{z}^i - (\mathbf{z}^{\mathrm{old}})^{(i)}\Big)$;

Vom Vektor $\mathbf{d}^i$ in (7.3) brauchen nur die Komponenten $\mathbf{d}_l^i$ bestimmt zu werden, bei denen der Index "l" einem Geschwindigkeitswert zu einem Knoten B_m auf einer Randseite $\Gamma^{ij} = \partial M_i \cap \partial M_j$ des Makroelementes M_i entspricht, d.h. für $l \in \tau_i^{-1}(I(M_i) \cap I(M_j))$ mit $j \in \mathcal{N}(i)$. Alle anderen Komponenten sind gleich Null. Jede Task T_i berechnet zunächst für alle Indizes $l \in \tau_i^{-1}(I(M_i) \cap I(M_j))$, $j \in \mathcal{N}(i)$, die Größe

$$\tilde{d}^{\,i}(l) \;:=\; \sum_{m \neq l} L_{lm}^{(i)}\,(\mathbf{z}^{\mathrm{old}})_m^{(i)}$$

und tauscht danach mit jeder Nachbartask T_j, $j \in \mathcal{N}(i)$, für alle $\mu \in I(M_i) \cap I(M_j)$ die Werte $\tilde{d}^{\,i}(\tau_i^{-1}(\mu))$ und $\tilde{d}^{\,j}(\tau_j^{-1}(\mu))$ aus. Die Komponente $\mathbf{d}_l^i$ in der Task T_i ergibt sich dann als

$$\mathbf{d}_l^i \;=\; \tilde{d}^{\,j}(\,\tau_j^{-1}(\tau_i(l))\,) \qquad \forall\, l \in \tau_i^{-1}(I(M_i) \cap I(M_j))\,,\; j \in \mathcal{N}(i)\,.$$

Lokale Kommunikation ist jeweils in den Teilschritten (a) und (c) des parallelen Glätters erforderlich. Wir benötigen also pro Glättungsiteration zwei lokale Kommunikationsschritte, in denen sämtliche Tasks $\{T_i\}$ entsprechende Daten mit ihren Nachbartasks austauschen.

7.3 Lokale Kommunikation

Bei einer lokalen Kommunikation hat jede Task T_i Daten mit ihren Nachbartasks T_j, $j \in \mathcal{N}(i)$, auszutauschen. Innerhalb unseres parallelen Mehrgitteralgorithmus kommt diese Art der Kommunikation sehr häufig vor, nämlich bei jeder Glättung, Restriktion, Prolongation, Berechnung des Defektvektors und Generierung der Blockmatrix $A = A(u^{\text{old}})$. Es ist daher wichtig, die lokale Kommunikation möglichst effizient zu realisieren.

Eine Möglichkeit besteht in der asynchronen Kommunikation. Dabei schicken alle Tasks T_i gleichzeitig Datenpakete an die entsprechenden "Adressaten" T_j, $j \in \mathcal{N}(i)$, ab, führen danach eventuell einige arithmetische Operationen mit inneren Makroelement-Daten aus und warten dann, bis alle für sie bestimmten Datenpakete von den "Absendern" T_j, $j \in \mathcal{N}(i)$, eingetroffen sind. Der Vorteil dieser Variante besteht darin, daß während der Kommunikation arithmetische Operationen ausgeführt werden können. Allerdings hat man keinen Einfluß mehr auf die effiziente Realisierung der Datenzustellung. Man kann nur hoffen, daß das Betriebssystem diese Aufgabe relativ gut löst.

Wir betrachten hier die synchrone Kommunikation, bei der jeweils gewisse Paare von Tasks (T_i, T_j) über virtuelle Kanäle c_{ij} synchronisiert Daten austauschen. Zu Beginn einer solchen Kommunikation schreibt jede Tasks T_i für jede ihrer Nachbartasks T_j, $j \in \mathcal{N}(i)$, sämtliche Daten, die an T_j zu schicken sind, in einen eigenen Datenpuffer $\mathcal{B}_{ij}$. Aus der Gitterstruktur ergibt sich, daß alle Datenpuffer die gleiche Länge haben. Im Tasknetzwerk $\{T_i\}$ wird dann in einer bestimmten Reihenfolge für jedes Paar $\{i, j\}$ mit $j \in \mathcal{N}(i)$ der Datenpuffer $\mathcal{B}_{ij}$ von T_i nach T_j und der Datenpuffer $\mathcal{B}_{ji}$ von T_j nach T_i geschickt. Dies soll mit einem möglichst hohen Maß an Parallelität erfolgen, d.h. in einer möglichst geringen Anzahl von synchronen Kommunikationstakten. Dabei gehen wir vereinfachend davon aus, daß bei einer beliebigen Anzahl disjunkter Paare aus benachbarten Tasks T_i und T_j, $j \in \mathcal{N}(i)$, die Kommunikation des Datenpuffers $\mathcal{B}_{ij}$ bzw. $\mathcal{B}_{ji}$ eine feste Zeitdauer beansprucht, die wir Kommunikationstakt nennen (d.h. wir vernachlässigen routing-Probleme bei der Realisierung virtueller Kanäle).

Die Aufgabe besteht nun darin, für das gesamte Tasksnetzwerk eine Folge paralleler Kommunikationsoperationen zu ermitteln, so daß alle Transferanforderungen der lokalen Kommunikation in einer minimalen Anzahl von Kommunikationstakten erfüllt werden. Dieses Problem wird vor dem Start des parallelen Programms mit Hilfe eines graphentheoretischen Zugangs durch ein Server-Programm gelöst. In der Initialisierungsphase des parallelen Programms wird dann jeder Task mitgeteilt, welche Transferoperation sie im k-ten Kommunikationstakt auszuführen hat.

Wir kommen nun zur graphentheoretischen Beschreibung unseres Kommunikationsproblems. Einer gegebenen Makroelementzerlegung $\{M_i\}$ ordnen wir einen "Kommunikationsgraphen" $G(X, E)$ zu, der folgendermaßen definiert ist. Die Knotenmenge X besteht aus allen inneren Geschwindigkeitsknoten B_m, $m = 1, \ldots, N_I$, auf dem gröbsten Gitterlevel 1, d.h. aus allen Schwerpunkten innerer Makroelementseiten (siehe Abbildung 7.3). Zwei Knoten $B_m, B_l \in X$ sind durch eine Kante $e_{ml} = [B_m, B_l] \in E$ verbunden, wenn sie einem gemeinsamen Makroelement angehören. Jeder Knoten B_m ent-

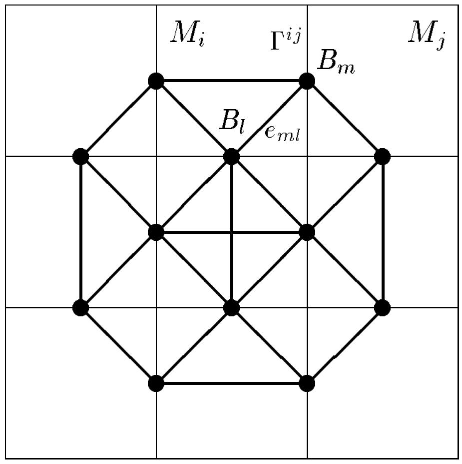


Abbildung 7.3: *Kommunikationsgraph $G(X, E)$ mit Knoten $B_m \in X$ auf den Makroelementseiten $\Gamma^{ij} = \partial M_i \cap \partial M_j$ (dünne Linien) und Kanten $e_{ml} \in E$ (dicke Linien)*

spricht in eindeutiger Weise einer gemeinsamen Seite Γ^{ij} zweier Makroelemente M_i und M_j und damit einem Paar benachbarter Tasks T_i und T_j mit $j \in \mathcal{N}(i)$. Unter einer *Färbung* des Graphen $G(X, E)$ verstehen wir eine Abbildung $\mathcal{F} : X \to \{1, \dots, N(\mathcal{F})\}$, die jedem Knoten $B_m \in X$ eine Farbe mit der Nummer $\mathcal{F}(B_m)$ zuordnet. $N(\mathcal{F})$ ist dabei die Gesamtanzahl der verwendeten Farben. Wir nennen $\mathcal{F}$ eine *zulässige Färbung* des Graphen $G(X, E)$, wenn für alle Kanten $e_{ml} = [B_m, B_l] \in E$ die zugehörigen Knoten B_m und B_l eine unterschiedliche Farbe haben (d.h. $\mathcal{F}(B_m) \neq \mathcal{F}(B_l)$).

Jeder zulässigen Färbung $\mathcal{F}$ entspricht eine Folge aus $N(\mathcal{F})$ parallelen Kommunikationsoperationen, in deren Ergebnis alle Paare benachbarter Tasks T_i und T_j, $j \in \mathcal{N}(i)$, jeweils einen der Datenpuffer $\mathcal{B}_{ij}$ oder $\mathcal{B}_{ji}$ übertragen haben. Dabei versenden im k-ten Kommunikationstakt alle diejenigen Task-Paare (T_i, T_j) einen Datenpuffer, für die der zugehörige Knoten B_m auf der Makroseite $\Gamma^{ij} = \partial M_i \cap \partial M_j$ die Farbe k hat ($\mathcal{F}(B_m) = k$). In einer weiteren Folge von $N(\mathcal{F})$ Kommunikationstakten wird dann der jeweils verbleibende Datenpuffer übertragen. Für die Realisierung einer lokalen Kommunikation benötigen wir also $2N(\mathcal{F})$ Kommunikationstakte. Zur Festlegung, welcher der beiden Datenpuffer zuerst kommuniziert werden soll, kann man die Prozessornummern $\texttt{pid}(i)$ der Tasks T_i verwenden und definieren, daß im Fall $\texttt{pid}(i) \leq \texttt{pid}(j)$ zuerst $\mathcal{B}_{ij}$ übertragen wird und andernfalls $\mathcal{B}_{ji}$.

Um eine minimale Anzahl von Kommunikationstakten bei der lokalen Kommunikation zu erreichen, benötigen wir also eine zulässige Färbung unseres Graphen mit einer minimalen Anzahl von Farben. Dies ist ein bekanntes Problem aus der Graphentheorie. Die minimale Anzahl von Farben zur zulässigen Färbung eines Graphen $G = G(X, E)$ wird *chromatische Zahl* $\chi(G)$ genannt. Ihre exakte Bestimmung gehört zur Klasse der NP-harten Probleme und würde einen zu hohen Aufwand erfordern [73]. Wir begnügen uns deshalb mit einer "fast optimalen" zulässigen Färbung, die wir durch Konstruktion einer Minimalgradfolge bestimmen. Der zugehörige Algorithmus (für dessen Beschreibung wir auf [73] verweisen) hat einen Aufwand von $O(N_m^2)$, wobei N_m die Anzahl der Makroelemente bezeichnet. Bisher lieferte er in allen unseren Anwendungsfällen bereits eine optimale Färbung. Die Aussage, daß eine gefundene zulässige Färbung optimal ist, können wir aus der folgenden unteren Abschätzung der chromatischen Zahl schlußfolgern. Nehmen wir an, daß es ein Makroelement M_i gibt, welches ganz im Innern des Gebietes $\Omega \subset \mathbf{R}^d$ liegt. Dann sind die $2d$ Seitenmittenknoten B_m des Makroelementes M_i Knoten des Graphen $G(X, E)$, welche alle durch eine Kante miteinander verbunden sind. Zur zulässigen Färbung dieser Knoten benötigt man $2d$ verschiedene Farben, woraus sich die Abschätzung

$$2d \;\leq\; \chi(G) \tag{7.4}$$

ergibt. Haben wir also eine zulässige Färbung $\mathcal{F}$ mit der Farbenanzahl $N(\mathcal{F}) = 2d$ konstruiert, so ist diese Färbung optimal.

Abschließend wollen wir noch eine obere Abschätzung für die chromatische Zahl angeben. Sei $\delta(G)$ die maximale Valenz des Graphen $G = G(X, E)$, d.h. die maximale Anzahl von Kanten, die von einem Knoten ausgehen. Dann gilt die Abschätzung $\chi(G) \leq \delta(G) + 1$ (siehe [73]). Da ein beliebiger Knoten $B_m \in X$ in höchstens zwei Makroelementen liegt, gehen von ihm höchstens $2(2d-1)$ Kanten aus. Daraus ergibt sich für die chromatische Zahl unseres Kommunikationsgraphen $G(X, E)$ die Abschätzung

$$\chi(G) \;\leq\; 4d - 1\,. \tag{7.5}$$

7.4 Globale Kommunikation

In unserem parallelen Mehrgitteralgorithmus benötigen wir globale Kommunikation im Zusammenhang mit dem Grobgitterlöser und für die Berechnung globaler Normen oder Skalarprodukte. Zur Vereinfachung des Programms der Solver-Task S wird die Kommunikation zwischen dem Tasknetzwerk $\{T_i\}$ und S stets über eine feste Task T_{i^*} abgewickelt, die wir im folgenden *Root-Task* nennen. Wenn also S einen globalen Vektor vom Tasknetzwerk $\{T_i\}$ erhalten soll, so ist dieser zunächst in T_{i^*} bereitzustellen und dann von T_{i^*} nach S zu verschicken. Soll andererseits der in S berechnete Lösungsvektor eines Grobgitterproblems jeder Task des Tasknetzwerkes $\{T_i\}$ zur Verfügung gestellt werden, so schickt S diesen Vektor lediglich an T_{i^*} und danach übernimmt das Tasknetzwerk seine weitere Verteilung. Im folgenden behandeln wir deshalb nur noch die globale Kommunikation im Tasknetzwerk $\{T_i\}$.

Zunächst wollen wir die verschiedenen Arten globaler Kommunikation charakterisieren. Hinsichtlich einer zu ermittelnden, globalen Information möge jede Task T_i bereits einen eigenen Anteil bestimmt und in einen lokalen Datenpuffer $\mathcal{B}_i$ geschrieben haben. Mit $|\mathcal{B}|$ sei im folgenden die Länge (Anzahl der Bytes) eines Datenpuffers $\mathcal{B}$ bezeichnet. Die "Summe" der Informationen zweier Tasks T_i und T_j charakterisieren wir durch den Datenpuffer $\mathcal{B}_i \oplus \mathcal{B}_j$. Bei der Verknüpfung "$\oplus$" unterscheiden wir zwei Varianten. Für die erste Variante, die wir "Verdichtung" nennen, gilt

$$|\mathcal{B}_i \oplus \mathcal{B}_j| \;=\; |\mathcal{B}_i| \;=\; |\mathcal{B}_j|\,,$$

d.h. die neue Information in $\mathcal{B}_i \oplus \mathcal{B}_j$ wird durch logische Verknüpfung (z.B. Summe oder Maximum) der entsprechenden Informationen in $\mathcal{B}_i$ und $\mathcal{B}_j$ gebildet und benötigt genausoviel Speicherplatz wie die Informationen in $\mathcal{B}_i$ bzw. $\mathcal{B}_j$. Die zweite Variante nennen wir "Akkumulation". Sie kommt bei der Erstellung eines globalen Vektors aus lokalen Komponenten vor. Zur Bestimmung von $\mathcal{B}_i \oplus \mathcal{B}_j$ werden die Informationen von $\mathcal{B}_i$ und $\mathcal{B}_j$ lediglich zusammengefügt und es gilt

$$|\mathcal{B}_i \oplus \mathcal{B}_j| \;=\; |\mathcal{B}_i| \,+\, |\mathcal{B}_j|\,.$$

Für unseren parallelen Mehrgitteralgorithmus benötigen wir die folgenden drei Arten globaler Kommunikation, die wir als "send-up"-, "send-down"- und "gossip"-Operation bezeichnen.

- **send-up** :
  In der Root-Task T_{i^*} wird in einem Datenpuffer $\mathcal{B}$ mit

  $$\mathcal{B} \;=\; \mathcal{B}_1 \oplus \mathcal{B}_2 \oplus \;\ldots\; \oplus \mathcal{B}_{N_m} \tag{7.6}$$

  die Summe der Informationen aller Tasks T_i bereitgestellt. Diese Operation wird benötigt, wenn der globale rechte-Seite-Vektor eines Grobgitterproblems an die Solver-Task S geschickt werden soll.

- **send-down** :
  Ein in der Root-Task T_{i^*} vorliegender Datenpuffer $\mathcal{B}$ wird jeder Task T_i, $i = 1, \ldots, N_m$, zur Verfügung gestellt. Diese Operation wird benötigt, wenn der von der Solver-Task S berechnete Lösungsvektor des Grobgitterproblems an alle Tasks T_i geschickt werden soll.

- **gossip** :
  Jeder Task T_i wird ein Datenpuffer $\mathcal{B}$ zur Verfügung gestellt, der gemäß (7.6) die Summe der Informationen aller Tasks T_j, $i = 1, \ldots, N_m$, enthält. Diese Operation wird zur Bestimmung globaler Normen oder Skalarprodukte benötigt.

Unser Ziel besteht darin, für diese Arten globaler Kommunikation eine möglichst effiziente Realisierung mit einem hohen Maß an Parallelität zu finden. Wir verwenden wieder die synchrone Kommunikation (die asynchrone Kommunikation würde hier eher zu einem "Chaos" im Datentransfer führen). Da bei der globalen Kommunikation vorwiegend kleine Datenpakete zu verschicken sind, wird die benötigte Gesamtzeit hauptsächlich durch die sogenannten "start-up"-Zeiten (Zeiten für den Verbindungsaufbau zu Beginn jeder Kommunikation) bestimmt. Entscheidend ist also, daß die Anzahl der Kommunikationsschritte, die man nacheinander zur Realisierung der globalen Kommunikation ausführen muß, möglichst klein ist. Ein Kommunikationsschritt besteht dabei aus mehreren, parallel ausgeführten Datentransferoperationen. Um zu einer effizienten globalen Kommunikation zu kommen, verwenden wir das Prinzip, daß eine Task nicht nur ihre eigene Information, sondern stets auch die gesamten Informationen, die sie von anderen Tasks erhalten hat, weiterschickt. Dadurch werden im Falle der Akkumulation die zu versendenden Datenpuffer ständig größer.

Im folgenden beweisen wir eine untere Abschätzung für die minimale Anzahl von Kommunikationsschritten, die man für eine send-up- oder send-down-Operation benötigt. Dabei gehen wir von der Voraussetzung aus, daß eine beliebige Task in einem Kommunikationsschritt nur mit höchstens einer anderen Task Daten austauschen kann.

LEMMA 7.1 *Zu einer gegebenen Menge von Tasks T_i, $i = 1, \ldots, N_m$, sei die natürliche Zahl s definiert durch*

$$2^{s-1} \; < \; N_m \; \leq \; 2^s \, . \tag{7.7}$$

Dann sind für eine send-up- oder send-down-Operation mindestens s Kommunikationsschritte erforderlich.

Beweis.
Wir beweisen die Aussage für den Fall einer send-up-Operation. Der Beweis für die send-down-Operation ist ähnlich. Nach einem Kommunikationsschritt kann eine beliebige Task höchstens über die Informationen von maximal 2 Tasks verfügen. Nach zwei Kommunikationsschritten kann jede Task höchstens die Informationen von maximal 2^2 Tasks besitzen. Durch vollständige Induktion ergibt sich daraus, daß nach $(s-1)$ Kommunikationsschritten eine beliebige Task höchstens die Informationen von maximal 2^{s-1} Tasks haben kann. Wegen (7.7) bedeutet dies, daß nach $(s-1)$ Kommunikationsschritten keine Task über die gesamte Information verfügt, woraus sich die Behauptung ergibt. □

Wir werden nun eine Möglichkeit für die Realisierung der send-up- und send-down-Operation angeben, bei der man die untere Schranke s für die Anzahl der Kommunikationsschritte auch tatsächlich erreicht. Zunächst behandeln wir die send-up-Operation. Die Idee besteht darin, in jedem Kommunikationsschritt die gesamte Information auf eine immer kleiner werdende Menge von "Mastertasks" zu konzentrieren, wobei jede Mastertask alle Informationen zu einem Teilgebiet einer rekursiven Gebietszerlegung enthält.

Wir beginnen mit der Definition einer rekursiven Gebietszerlegung, welche in Stufen $l = 1, \ldots, s$ erfolgt. Jedes Teilgebiet $\Omega_k^{(l)}$ einer Zerlegung $\Omega = \bigcup_k \Omega_k^{(l)}$ auf der Stufe "l" besteht aus gewissen Makroelementen, d.h.

$$\Omega_k^{(l)} \; := \; \bigcup\nolimits_{i \in I_k^{(l)}} M_i \tag{7.8}$$

mit $I_k^{(l)} \subset I := \{1, \dots, N_m\}$. Eine Gebietszerlegung entspricht also eindeutig einer Zerlegung der Indexmenge aller Makroelemente $I = \bigcup_k I_k^{(l)}$. Wir definieren daher unsere rekursive Gebietszerlegung über die folgende rekursive Zerlegung der Indexmenge $I_1^{(0)} := I$.

<u>Rekursive Gebietszerlegung</u>

for $l = 1, \dots, s$ **do**
  for $k = 1, \dots, 2^{l-1}$ **do**

- <u>falls $\#I_k^{(l-1)} > 1$:</u>

  zerteile $I_k^{(l-1)}$ in geeignete disjunkte Indexmengen

  $$I_k^{(l-1)} = I_{2k-1}^{(l)} \cup I_{2k}^{(l)}, \tag{7.9}$$

  so daß für $\mu \in \{2k-1, 2k\}$ gilt

  $$\#I_\mu^{(l)} \le 2^{s-l} \tag{7.10}$$

- <u>falls $\#I_k^{(l-1)} = 1$:</u>

  setze $\quad I_{2k-1}^{(l)} := I_{2k}^{(l)} := I_k^{(l-1)}$

  enddo
enddo

Ein Beispiel für eine derartige Gebietszerlegung zeigt Abbildung 7.4. Die Zerlegung (7.9) der Indexmenge $I_k^{(l-1)}$ sollte so erfolgen, daß einerseits die neuen Indexmengen möglichst gleichmächtig sind und andererseits die zugehörigen Gebiete $\Omega_{2k-1}^{(l)}$ und $\Omega_{2k}^{(l)}$ möglichst vernünftige zusammenhängende Gebiete ergeben (als Konstruktionsprinzip könnte man Koordinatenbisektion verwenden). Aus $N_m \le 2^s$ erhalten wir induktiv, daß die Forderung (7.10) stets erfüllbar ist.

Ausgehend von der rekursiven Zerlegung $I = \bigcup_k I_k^{(l)}$ definieren wir nun zu jedem Index $i \in I$ ein *Hierarchielevel* $\mathrm{HL}(i)$ und einen *Mastertask-Index* $\mathrm{MT}(i)$. Zu ihrer Bestimmung führen wir für jede Teilmenge $I_k^{(l)}$ einen *Rootindex* $r_k^{(l)} \in I_k^{(l)}$ ein, der die Nummer derjenigen Task angibt, auf die später bei der send-up-Operation die gesamte Information zum Gebiet $\Omega_k^{(l)}$ (d.h. von allen Tasks T_i mit $i \in I_k^{(l)}$) konzentriert wird. In Abbildung 7.4 wurde das Symbol $\Omega_k^{(l)}$ zur Bezeichnung eines Teilgebietes jeweils in das Makroelement M_i mit dem Rootindex $i = r_k^{(l)}$ geschrieben. Das Hierarchielevel $\mathrm{HL}(i)$ gibt nun an, auf wievielen Stufen $l = s, \dots, s - \mathrm{HL}(i) + 1$ der Index i als Rootindex vorkommt, und $\mathrm{MT}(i)$ ist der Rootindex der zugehörigen Teilmenge $I_k^{(l-1)} \ni i$ auf der nächst kleineren Stufe. Die Rootindizes auf der Stufe $l = s$ sind bereits dadurch eindeutig festgelegt, daß die Mengen $I_k^{(s)}$ wegen (7.10) nur noch aus einem Element $I_k^{(s)} = \{r_k^{(s)}\}$ bestehen. Die Hierarchielevel und Mastertask-Indizes werden nun folgendermaßen bestimmt.

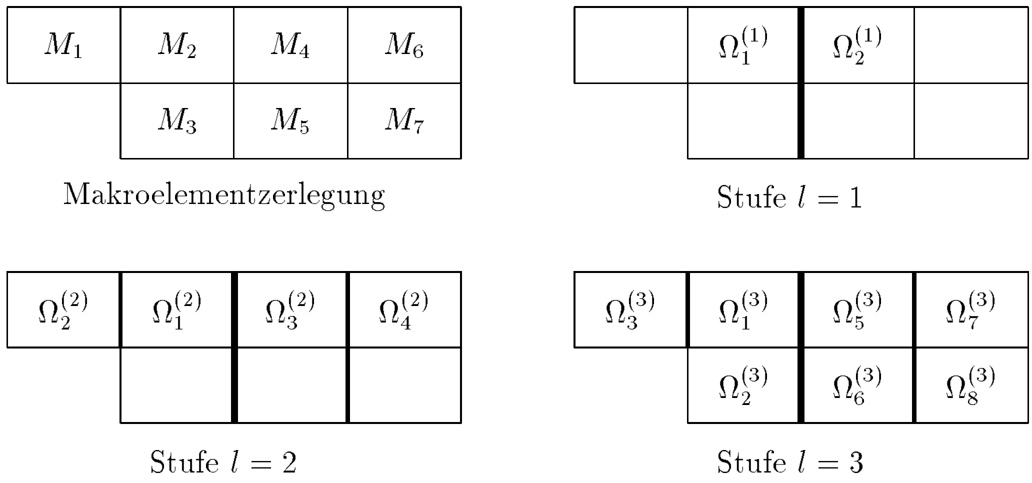


Abbildung 7.4: *Rekursive Gebietszerlegung zu einer vorgegebenen Makroelementzerlegung* $\{M_i\}$

Bestimmung von Hierarchielevel und Mastertask-Index

initialisiere $\forall\, i \in I$: $\mathrm{HL}(i) := 1\,;\ \mathrm{MT}(i) := i\,;$
for $l = s, s-1, \dots, 1$ **do**
  for $k = 1, \dots, 2^{l-1}$ **do**
    (1) $r_k^{(l-1)} := r_{2k-1}^{(l)}\,;$
    (2) $\mathrm{MT}(r_{2k}^{(l)}) := r_k^{(l-1)}\,;$
    (3) $\mathrm{HL}(r_{2k-1}^{(l)}) := s + 2 - l\,;$
  enddo
enddo

Im letzten Schritt erhalten wir so den Index $i^* = r_1^{(0)}$ der Root-Task T_{i^*} des gesamten Tasknetzwerkes $\{T_i\}$. Zur Realisierung der globalen Kommunikation wird nun eine baumartige Topologie eingerichtet, bei der jede Task T_i mit ihrer Mastertask T_j, $j = \mathrm{MT}(i)$, durch einen virtuellen Kanal c_{ij} verbunden ist. In Abbildung 7.5 ist dies für unser obiges Beispiel dargestellt. Hier gilt s=3 und

$$\begin{array}{lll}
\mathrm{HL}(i) = 1 \quad \text{für } i \in \{3,5,7\}, & \mathrm{MT}(3) = 2,\ \mathrm{MT}(5) = 4,\ \mathrm{MT}(7) = 6, \\
\mathrm{HL}(i) = 2 \quad \text{für } i \in \{1,6\}, & \mathrm{MT}(1) = 2,\ \mathrm{MT}(6) = 4, \\
\mathrm{HL}(4) = 3,\ \mathrm{HL}(2) = 4, & \mathrm{MT}(4) = 2,\ \mathrm{MT}(2) = 2\,.
\end{array}$$

Eine send-up-Operation kann dann folgendermaßen durch s Kommunikationsschritte realisiert werden.

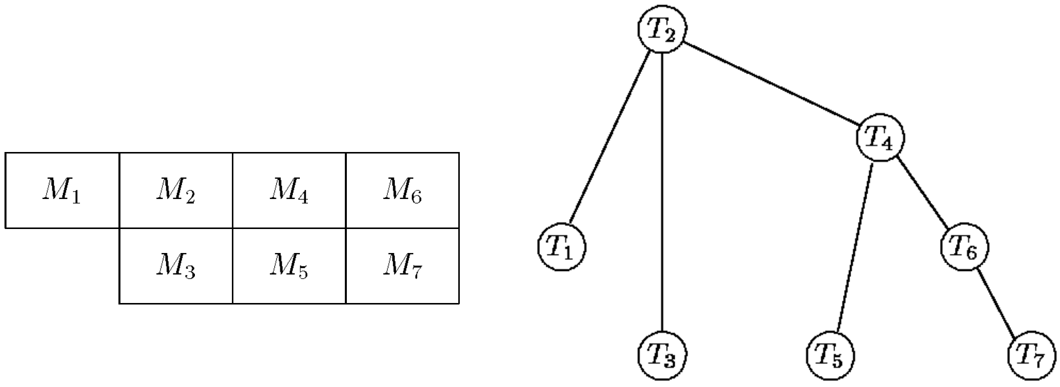


Abbildung 7.5: *Baumartige Topologie unter den Tasks $\{T_i\}$ für die globale Kommunikation*

Realisierung einer send-up-Operation

for $m = 1, \ldots, s$ **do**

(1) alle Tasks T_i mit $\mathrm{HL}(i) = m$ schicken parallel ihre Datenpuffer $\mathcal{B}_i$ an ihre Mastertask T_j mit $j = \mathrm{MT}(i)$;

(2) alle Mastertasks T_j mit $j = \mathrm{MT}(i)$ und $\mathrm{HL}(i) = m$ bilden parallel die neue Information

$$\mathcal{B}_j \ := \ \mathcal{B}_j \oplus \mathcal{B}_i \, ;$$

enddo

Nach dem letzten Kommunikationsschritt hat die Root-Task T_{i^*} die gesamte globale Information $\mathcal{B}_{i^*} = \mathcal{B}_1 \oplus \ldots \oplus \mathcal{B}_{N_m}$. Aus der Sicht einer beliebigen Task T_j bedeutet die send-up-Operation, daß T_j zunächst von allen ihr "untergeordneten" Tasks T_i mit $j = \mathrm{MT}(i)$ die Informationen einsammelt und diese dann zusammen mit der eigenen an die zugeordnete Mastertask weiterschickt.

Ein Vorteil dieser Variante der send-up-Operation besteht darin, daß alle Kommunikationsschritte bis zum Schritt m parallel in den Taskblöcken zu den disjunkten Teilgebieten $\Omega_k^{(s-m)}$ ablaufen. Ordnet man also die Tasks so den Prozessoren zu, daß die Taskblöcke $\{T_i : i \in I_k^{(l)}\}$ zu Teilgebieten $\Omega_k^{(l)}$ jeweils auf zusammenhängende Prozessoren abgebildet werden, so werden dadurch routing-Konflikte gering gehalten, die bei der parallelen Realisierung der virtuellen Kanäle zwischen Task und Mastertask entstehen können. Eine derartige Abbildung der Tasks auf die Prozessoren kann man relativ leicht erreichen, wenn man vorher die Makroelemente so umnumeriert, daß die entsprechenden neuen Indexmengen $\widetilde{I}_k^{(l)}$ sämtlich aus aufeinanderfolgenden natürlichen Zahlen bestehen. Ein weiterer Vorteil dieser send-up-Variante ist, daß in den letzten Kommunikationsschritten lediglich einige wenige Tasks Daten austauschen müssen, so daß unbeschäftigte Prozessoren zur Verfügung stehen, die zu einer schnellen Realisierung der virtuellen Kanäle zwischen den Tasks auf weiter entfernteren Prozessoren beitragen.

Indem wir die Schritte der send-up-Operation umdrehen, erhalten wir die folgende Realisierung der send-down-Operation, bei der ein in der Root-Task T_{i^*} vorhandener Datenpuffer $\mathcal{B}$ an alle Tasks verteilt wird.

Realisierung einer send-down-Operation

for $m = 1, \ldots, s$ **do**
  alle Tasks T_i mit $\mathrm{HL}(i) = s + 1 - m$ bekommen parallel von ihrer Mastertask T_j, $j = \mathrm{MT}(i)$, den Datenpuffer $\mathcal{B}$ geschickt ;
enddo

Für eine beliebige Task T_i bedeutet die send-down-Operation, daß T_i zunächst den Datenpuffer $\mathcal{B}$ von ihrer Mastertask T_j, $j = \mathrm{MT}(i)$, erhält und ihn dann an ihre "untergeordneten" Tasks in der Reihenfolge fallender Hierarchielevel weiterschickt.

Damit haben wir für die send-up- und send-down-Operation jeweils eine Realisierung gefunden, die nach Lemma 7.1 optimal hinsichtlich der Anzahl der erforderlichen Kommunikationsschritte ist. Die gossip-Operation realisieren wir, indem wir zunächst mit Hilfe einer send-up-Operation die globale Information in der Root-Task T_{i^*} bereitstellen und diese dann durch eine send-down-Operation an alle Tasks verteilen. Damit benötigen wir für diese Operation $2s$ Kommunikationsschritte, was für praktische Belange als ausreichend erscheint, da $s \approx \log_2 N_m$ lediglich logarithmisch mit der Taskanzahl N_m wächst. Aus theoretischer Sicht ist die Schrittanzahl "$2s$" nicht optimal [19]. Allerdings benötigen die in der Literatur vorgeschlagenen Varianten mit kleinerer Schrittanzahl wesentlich mehr virtuelle Kanäle, so daß nicht klar ist, ob dies bei der praktischen Realisierung auch tatsächlich zu einer kürzeren Gesamtzeit führt.

Kapitel 8

Effizienzanalyse

Nachdem wir in den beiden vorhergehenden Kapiteln den sequentiellen und parallelen Algorithmus zur Lösung des diskreten Navier-Stokes Problems beschrieben haben, wollen wir nun die Effizienz beider Verfahren untersuchen. Hierzu verwenden wir als Testbeispiel das driven-cavity-Problem. In Abschnitt 8.1 demonstrieren wir anhand dieses Beispiels, daß beide Algorithmen ein robustes Konvergenzverhalten hinsichtlich der Reynolds-Zahl, der Gitterweite und der Anzahl der Makroelemente haben. Bedingt durch die Modifikationen, die wir am sequentiellen Algorithmus vorgenommen haben, um zu einem geeigneten parallelen Algorithmus zu kommen, hat der parallele Algorithmus im allgemeinen eine schlechtere Konvergenzrate als der sequentielle. Wie schnell das parallele Verfahren im Vergleich zum sequentiellen ist, wird durch die *numerische Effizienz* E^{num} ausgedrückt. Sie ist definiert als

$$E^{num} \;:=\; \frac{t_1^{seq}}{t_1^{par}}\,, \tag{8.1}$$

wobei t_1^{seq} und t_1^{par} jeweils die Rechenzeit des sequentiellen und parallelen Algorithmus auf einem Prozessor bezeichnen. Da lediglich ein Prozessor verwendet wird, enthält t_1^{par} keine Kommunikations- und Wartezeiten. Eigentlich müßten wir für t_1^{seq} die Zeit des besten sequentiellen Algorithmus verwenden, der für unser Problem zur Verfügung steht. Es ist jedoch schwierig, diesen besten Algorithmus eindeutig auszumachen. Deshalb begnügen wir uns damit, als Vertreter für einen effizienten sequentiellen Algorithmus das in Kapitel 6 beschriebene Mehrgitter-Verfahren zu verwenden. In Abschnitt 8.1 zeigen wir, daß unser paralleler Algorithmus im Vergleich zu diesem Verfahren bis hin zum massiv-parallelen Fall eine gute numerische Effizienz besitzt.

Für eine gute Gesamteffizienz ist es andererseits wichtig, daß unser Algorithmus eine möglichst hohe *parallele Effizienz* E^{par} besitzt, d.h. daß der Ausdruck

$$E^{par} \;:=\; \frac{t_1^{par}}{n\,t_n^{par}} \tag{8.2}$$

möglichst groß ist. Dabei bezeichnen n die Anzahl der verwendeten Prozessoren und t_n^{par} die Zeit für den parallelen Algorithmus auf n Prozessoren. Gegenüber der Ausführungszeit auf einem Prozessor könnte man bei Verwendung von n Prozessoren maximal einen

Beschleunigungsfaktor ("speed-up") von n erreichen. Die parallele Effizienz $E^{par} \in (0,1)$ gibt an, welcher Anteil von diesem maximalen speed-up bedingt durch Kommunikations- und Wartezeiten tatsächlich erzielt wird. Die *Gesamteffizienz* E wird definiert als

$$E \ := \ \frac{t_1^{seq}}{n\, t_n^{par}} \ = \ E^{num} E^{par}\,, \tag{8.3}$$

d.h. nE ist der speed-up, den man erreicht, wenn man anstelle des sequentiellen Algorithmus auf einem Prozessor den parallelen Algorithmus auf n Prozessoren verwendet.

Für die Konstellation, daß sowohl für die Solver-Task S als auch für die Makroelement-Tasks T_i jeweils ein eigener Prozessor verwendet wird (d.h. $n = N_m + 1$), untersuchen wir in Abschnitt 8.2 das Verhalten der parallelen Effizienz in Abhängigkeit vom feinsten Gitterlevel ℓ und der Anzahl N_m der Makroelemente. Es zeigt sich, daß mit wachsender Anzahl von Makroelementen der Zeitanteil für das Lösen der Grobgitterprobleme dominant wird, was zu einer sinkenden parallelen Effizienz führt. Um diesem Effekt entgegenzuwirken, versuchen wir, diesen Zeitanteil zu reduzieren. Eine Möglichkeit besteht darin, den bislang verwendeten augmented Lagrange Löser durch ein anderes iteratives Verfahren zu ersetzen. In Abschnitt 8.3 untersuchen wir hierzu den Vanka-Glätter, das SIMPLE-Verfahren und die triviale Variante, bei der wir den Grobgitterlöser einfach weglassen. Eine andere Idee ist, anstelle der einen Solver-Task S mehrere, parallel arbeitende Solver-Tasks einzusetzen. Die entsprechende Parallelisierung des Grobgitterlösers behandeln wir in Abschnitt 8.4. Sie basiert auf derselben rekursiven Gebietszerlegung, die wir auch für unsere globale Kommunikation verwendet haben. Von unseren Varianten des Grobgitterlösers parallelisieren wir auf diese Weise den Vanka-Glätter und das SIMPLE-Verfahren. In Abschnitt 8.5 untersuchen wir die Frage nach der günstigsten Anzahl einzusetzender Solver-Tasks und die nach der besten Variante des parallelen Lösers für den massiv-parallelen Fall.

8.1 Numerische Effizienz

In diesem Abschnitt wollen wir das Konvergenzverhalten des sequentiellen und parallelen Algorithmus für verschiedene Reynolds-Zahlen, Gitterweiten und Makroelementzerlegungen untersuchen. Von der numerischen Seite her unterscheiden sich beide Algorithmen nur im Glättungsverfahren. Während sich in einem Schritt des sequentiellen Glätters auf Grund des Gauß-Seidel-artigen update-Mechanismus die neu berechnete Information über das gesamte Gebiet ausbreiten kann, ist dies beim parallelen Glätter jeweils nur innerhalb der Makroelemente möglich. Es ist also zu erwarten, daß der parallele Algorithmus eine schlechtere Konvergenzrate als der sequentielle hat. Andererseits sollte sie besser sein als bei der elementweisen Jacobi-Variante des Vanka-Glätters (siehe [56]). Da diese Jacobi-Variante auch ein sinnvolles Glättungsverfahren ist (wenn auch nicht das beste), können wir erwarten, daß die Konvergenzrate unseres parallelen Algorithmus selbst bei wachsender Anzahl von Makroelementen in einer vernünftigen Größenordnung bleibt. Wie groß die Unterschiede zum sequentiellen Algorithmus sind, wollen wir im folgenden anhand von Testrechnungen untersuchen.

Als Testbeispiel betrachten wir das driven-cavity-Problem mit $\Omega = (0,1)^2$, $f = 0$ und den Dirichlet'schen Randbedingungen $u(x,y)_{|\partial\Omega} = (1,0)^\mathsf{T}$ für $y = 1$ und $u(x,y)_{|\partial\Omega} = 0$ für $y < 1$. Die Makroelementzerlegung wählen wir als achsenparallele $M \times M$-Zerlegung des Gebietes Ω (siehe Abbildung 8.1), d.h. wir haben $N_m = M^2$ Makroelemente.

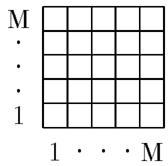


Abbildung 8.1: *Makroelementzerlegung beim driven-cavity-Problem*

Wir verwenden punktwert-orientierte Q_1-parametrische Elemente und die ursprüngliche upwind-Version $\tilde{n}_h$. Die nichtlineare Iteration zur Lösung des entstehenden Gleichungssystems wird beendet, wenn sowohl die Norm des Defektes als auch die maximale Änderung gegenüber der vorhergehenden Iterierten kleiner als 10^{-5} sind. Die numerischen Konvergenzraten, die wir für den sequentiellen und parallelen Algorithmus vergleichen wollen, sind folgendermaßen definiert. Seien $\mathbf{y}^0$ und $\mathbf{y}^k$ die Approximationen des Lösungsvektors $\mathbf{y}$ zu Beginn und am Ende der nichtlinearen Iteration (siehe Abschnitt 6.2). Die Konvergenzrate der nichtlinearen Iteration ρ^{nl} definieren wir dann als

$$\rho^{nl} := \left(\frac{||L(\mathbf{y}^k)\,\mathbf{y}^k - \mathbf{F}||}{||L(\mathbf{y}^0)\,\mathbf{y}^0 - \mathbf{F}||} \right)^{1/k} . \tag{8.4}$$

Seien $\mathbf{y}^{i,0}$ und $\mathbf{y}^{i,m_i}$ die Näherungslösungen zu Beginn und am Ende der Mehrgitter-Iteration im i-ten nichtlinearen Schritt. Dann bestimmen wir die Konvergenzrate der Mehrgitter-Iteration ρ^{mg} nach der Formel

$$\rho^{mg} := \frac{1}{k}\sum_{i=1}^{k}\rho_i \qquad \text{mit} \qquad \rho_i := \left(\frac{||L(\mathbf{y}^{i,0})\,\mathbf{y}^{i,m_i} - \mathbf{F}||}{||L(\mathbf{y}^{i,0})\,\mathbf{y}^{i,0} - \mathbf{F}||} \right)^{1/m_i} . \tag{8.5}$$

Mit ρ^{nl}_{seq}, ρ^{mg}_{seq} und ρ^{nl}_{par}, ρ^{mg}_{par} bezeichnen wir die entsprechenden Konvergenzraten für den sequentiellen und parallelen Algorithmus. Tabelle 8.1 zeigt einen Vergleich zwischen den Konvergenzraten beider Algorithmen. Dabei variieren wir das feinste Gitterlevel ℓ, die Anzahl N_m der Makroelemente sowie die Reynolds-Zahl Re. Beide Algorithmen zeigen ein robustes Konvergenzverhalten. Weiterhin erkennen wir, daß der Übergang vom sequentiellen zum parallelen Algorithmus bis hin zum massiv-parallelen Fall ($N_m \gg 1$) lediglich zu einer geringfügigen Verschlechterung in der Konvergenzrate führt.

Wenn wir für die obigen Testrechnungen die numerische Effizienz E^{num} des parallelen Algorithmus nach der Formel (8.1) bestimmen wollen, so gibt es ein praktisches Problem. Eigentlich müßten die CPU-Zeiten t_1^{seq} und t_1^{par} auf einem Prozessor unseres Parallelrechners gemessen werden. Ein solcher Prozessor verfügt aber nicht über ausreichend Speicherplatz, um die meist sehr großen globalen Probleme zu lösen. Deshalb benutzen wir einen leistungsfähigen seriellen Rechner mit hinreichend großer Speicherplatzkapazität, messen dort entsprechende Zeiten $\tilde{t}_1^{seq}$ und $\tilde{t}_1^{par}$ und verwenden den Wert

Tabelle 8.1: *Konvergenzraten des sequentiellen und parallelen Algorithmus beim driven-cavity-Problem für verschiedene feinste Gitterlevel ℓ, Anzahlen von Makroelementen N_m und Reynolds-Zahlen Re*

ℓ	N_m	$Re = 20$				$Re = 200$				$Re = 2000$			
		ρ^{nl}_{seq}	ρ^{nl}_{par}	ρ^{mg}_{seq}	ρ^{mg}_{par}	ρ^{nl}_{seq}	ρ^{nl}_{par}	ρ^{mg}_{seq}	ρ^{mg}_{par}	ρ^{nl}_{seq}	ρ^{nl}_{par}	ρ^{mg}_{seq}	ρ^{mg}_{par}
4	4	0.13	0.13	0.16	0.17	0.33	0.33	0.19	0.20	0.47	0.45	0.24	0.24
	16	0.13	0.13	0.18	0.19	0.35	0.36	0.27	0.29	0.54	0.56	0.35	0.39
	25	0.13	0.13	0.18	0.19	0.36	0.35	0.29	0.30	0.55	0.55	0.40	0.43
	36	0.13	0.14	0.17	0.18	0.33	0.33	0.28	0.29	0.52	0.52	0.35	0.37
	64	0.13	0.13	0.18	0.19	0.33	0.31	0.29	0.30	0.46	0.47	0.39	0.37
	100	0.14	0.14	0.19	0.19	0.32	0.30	0.29	0.30	0.52	0.57	0.39	0.44
	144	0.13	0.14	0.18	0.19	0.29	0.29	0.28	0.29	0.57	0.57	0.40	0.42
	256	0.14	0.14	0.20	0.20	0.31	0.29	0.28	0.29	0.58	0.59	0.43	0.44
	400	0.14	0.14	0.20	0.20	0.31	0.30	0.28	0.29	0.59	0.59	0.42	0.43
	576	0.13	0.13	0.19	0.19	0.31	0.32	0.27	0.28	0.57	0.57	0.40	0.41
5	4	0.13	0.13	0.18	0.18	0.35	0.35	0.27	0.28	0.54	0.55	0.35	0.38
	16	0.14	0.13	0.19	0.19	0.33	0.30	0.30	0.30	0.55	0.55	0.49	0.51
	25	0.14	0.14	0.19	0.20	0.29	0.30	0.30	0.31	0.50	0.50	0.40	0.42
	36	0.14	0.14	0.19	0.19	0.26	0.28	0.28	0.30	0.56	0.56	0.45	0.47
	64	0.14	0.14	0.20	0.21	0.31	0.32	0.30	0.32	0.56	0.58	0.47	0.49
	100	0.14	0.14	0.20	0.21	0.30	0.30	0.30	0.31	0.58	0.58	0.48	0.49
	144	0.13	0.13	0.20	0.20	0.32	0.32	0.29	0.30	0.57	0.57	0.46	0.47
	256	0.14	0.14	0.21	0.22	0.29	0.27	0.29	0.30	0.59	0.59	0.48	0.49
6	4	0.13	0.13	0.18	0.18	0.33	0.32	0.30	0.30	0.54	0.53	0.49	0.50
	16	0.14	0.14	0.20	0.20	0.31	0.33	0.31	0.33	0.51	0.53	0.55	0.58
	25	0.14	0.14	0.21	0.21	0.32	0.33	0.32	0.33	0.52	0.53	0.55	0.58
	36	0.13	0.13	0.20	0.20	0.34	0.34	0.31	0.32	0.55	0.55	0.48	0.50
	64	0.14	0.15	0.22	0.22	0.32	0.31	0.32	0.33	0.56	0.56	0.51	0.53
	100	0.14	0.14	0.22	0.23	0.30	0.30	0.32	0.33	0.57	0.58	0.52	0.53
	144	0.14	0.14	0.22	0.22	0.32	0.32	0.31	0.32	0.56	0.56	0.50	0.51

$\widetilde{E}^{num} := \tilde{t}_1^{seq}/\tilde{t}_1^{par}$ als Approximation von E^{num}. In Tabelle 8.2 geben wir zu den Testrechnungen aus Tabelle 8.1 die entsprechenden Werte $\widetilde{E}^{num}$ an, welche auf einer Workstation HP 9000/735 bestimmt wurden. Wir sehen, daß unser paralleler Algorithmus für verschiedene Reynolds-Zahlen und feinste Gitterlevel ℓ bis hin zum massiv-parallelen Fall eine gute numerische Effizienz besitzt.

BEMERKUNG 8.1 *Während für shape-reguläre Gitter der Vanka-Glätter ein geeignetes Glättungsverfahren ist, sind im Fall anisotroper Gitter gewisse Modifikationen erforderlich, um gute Konvergenzeigenschaften zu sichern. In diesem Fall sollte in der lokalen Systemmatrix $\widetilde{L}_K$ (siehe Abschnitt 6.3.1) die Diagonalmatrix D_K durch die volle Blockmatrix A_K ersetzt werden. Weiterhin ist es sinnvoll, anstelle einer elementweisen eine linienweise Glättung zu verwenden (siehe* [42], [46], [3]*).*

Tabelle 8.2: *Numerische Effizienz des parallelen Algorithmus für verschiedene feinste Gitterlevel ℓ, Anzahlen von Makroelementen N_m und Reynolds-Zahlen Re*

ℓ	N_m	$Re = 20$	$Re = 200$	$Re = 2000$
4	4	0.98	0.94	0.97
	16	0.92	0.86	0.79
	25	0.92	0.93	0.83
	36	0.83	0.87	0.88
	64	0.94	0.97	0.88
	100	0.91	0.97	0.75
	144	0.95	0.86	0.90
	256	0.97	1.02	0.96
	400	0.99	0.98	0.94
	576	0.99	0.99	0.99
5	4	0.94	0.94	0.84
	16	1.01	1.03	0.87
	25	0.94	0.86	0.90
	36	0.94	0.80	0.91
	64	0.94	0.82	0.87
	100	0.94	0.89	0.88
	144	0.94	0.91	0.92
	256	0.95	0.86	0.92
6	4	0.94	0.91	0.94
	16	0.94	0.86	0.86
	25	0.94	0.92	0.90
	36	0.94	0.90	0.90
	64	0.94	0.90	0.90
	100	0.94	0.92	0.90
	144	0.94	0.92	0.88

8.2 Parallele Effizienz

Anhand des Testbeispiels aus dem vorhergehenden Abschnitt wollen wir nun untersuchen, wie sich die parallele Effizienz E^{par} verhält, wenn wir die Anzahl N_m der Makroelemente und das feinste Gitterlevel ℓ verändern. Für die Solver-Task und die Makroelement-Tasks verwenden wir jeweils einen eigenen Prozessor, d.h. wir benutzen $n = N_m + 1$ Prozessoren. Mit der Anzahl N_m der Makroelemente erhöhen wir also gleichzeitig die Anzahl der Prozessoren und die Gesamtanzahl der Unbekannten. Tabelle 8.3 vermittelt einen Eindruck über die Größenordnung, in der sich die Gesamtanzahl der Unbekannten und die Anzahl der Unbekannten pro Makroelement-Task in unseren Testrechnungen bewegen. Bei der Angabe von Ergebnissen beschränken wir uns im folgenden stets auf den Fall Re=2000. Die Ergebnisse für die anderen Reynolds-Zahlen sind nahezu dieselben.

In unserem parallelen Algorithmus werden Verluste an paralleler Effizienz hauptsächlich durch lokale und globale Kommunikation verursacht sowie dadurch, daß die Grobgitterprobleme nur mit Hilfe eines einziges Prozessors gelöst werden. Verluste auf Grund

Tabelle 8.3: *Anzahl der Unbekannten pro Makroelement-Task und Gesamtanzahl der Unbekannten für verschiedene Gitterlevel ℓ und Anzahlen N_m von Makroelementen*

ℓ	Unbekannte pro Task	Unbekannte insgesamt		
		$N_m = 4$	$N_m = 25$	$N_m = 64$
4	352	1 216	7 840	20 224
5	1 344	4 992	31 680	81 408
6	5 248	20 224	127 360	326 656
7	20 736	81 408	510 720	1 308 672
8	82 432	326 656	2 045 440	5 238 784

ungleicher Lastverteilung können wir vernachlässigen, da jede Makroelement-Task die gleiche Anzahl von Elementen hat und somit nur durch den Gebietsrand geringfügige Unterschiede entstehen können. Ebenso sind in unserem Algorithmus die Verluste durch Doppelberechnungen vernachlässigbar klein. Wie stark sich die Kommunikation und der nichtparallelisierte Grobgitterlöser auf die parallele Effizienz auswirken, hängt wesentlich von der Hardware-Situation des verwendeten Parallelrechners ab. Hierbei spielt das Verhältnis zwischen Kommunikations- und Rechenleistung eine entscheidende Rolle. Wir wollen dies am Beispiel der beiden Parsytec-Rechner GCel und GCPP demonstrieren.

Tabelle 8.4 zeigt für unser Testbeispiel aus Abschnitt 8.1 zu wachsender Anzahl N_m von Makroelementen die parallele Effizienz E^{par} und die Rechenzeit t_n^{par} des parallelen Algorithmus auf dem GCel (Betriebssystem PARIX 1.2) und dem GCPP (Betriebssystem PARIX 1.3). Um einen Eindruck zu gewinnen, in welcher Relation diese Parallelrechner zu einer Workstation stehen, geben wir zusätzlich die CPU-Zeit $\tilde{t}_1^{seq}$ des sequentiellen Algorithmus auf einer Workstation HP 9000/735 (13.2 MFlops) an. Bei der Bestim-

Tabelle 8.4: *Parallele Effizienz E^{par} und Rechenzeiten in Sekunden für verschiedene Rechner beim driven-cavity-Testproblem mit dem Gitterlevel $\ell = 6$, Re=2000 und wachsender Anzahl N_m von Makroelementen*

	HP	GCel						GCPP					
N_m	$\tilde{t}_1^{seq}$	E^{par}	t_n^{par}	t_{mg}	t_l	t_g	t_s	E^{par}	t_n^{par}	t_{mg}	t_l	t_g	t_s
4	78	0.82	893	10.7	.50	.003	.09	0.51	41	.50	.17	.001	.01
9	165	0.85	953	11.2	.88	.008	.25	0.44	54	.64	.30	.003	.03
16	395	0.88	1144	10.5	.96	.012	.47	0.40	73	.68	.37	.004	.04
25	670	0.82	1274	11.6	1.08	.017	1.04	0.34	88	.81	.47	.007	.07
36	893	0.70	1397	13.3	1.18	.019	2.12	0.30	95	.91	.50	.008	.12
64	1934	0.62	1936	18.6	1.61	.031	6.97	0.32	109	1.03	.53	.009	.21

mung der parallelen Effizienz nach der Formel $E^{par} = t_1^{par}/(n\, t_n^{par})$ wurde zunächst die

Ausführungszeit $\tilde{t}_1^{par}$ des parallelen Algorithmus auf einer Workstation HP 9000/735 gemessen und dann t_1^{par} gemäß $t_1^{par} = \eta\,\tilde{t}_1^{par}$ berechnet. Dabei wurde der Skalierungsfaktor η aus dem Verhältnis zwischen der Rechenleistung der Workstation und der eines einzelnen Prozessors ermittelt ($\eta = 35.3$ beim GCel, $\eta = 1.03$ beim GCPP). In der Tabelle bezeichnen weiterhin t_{mg} die Gesamtzeit für einen Mehrgitterzyklus und t_l, t_g sowie t_s die Zeit für lokale und globale Kommunikation sowie für das Lösen der Grobgitterprobleme innerhalb eines Mehrgitterzyklus. Dabei werden durch t_g lediglich die Kommunikationszeiten für die Bestimmung globaler Normen erfaßt, während die Zeiten für die globale Kommunikation zwischen dem Tasknetzwerk und der Solver-Task in t_s enthalten sind. Ein Prozessor MPC601 des GCPP hat eine um den Faktor 34 höhere Rechenleistung als ein Prozessor T805 des GCel. Die Kommunikationsgeschwindigkeit des GCPP ist jedoch für unsere Anwendung gegenüber dem GCel lediglich um einen Faktor zwischen 2 und 3 gestiegen. Das Verhältnis zwischen Kommunikationsleistung und Rechenleistung hat sich also beim Übergang vom GCel zum GCPP verschlechtert. Dadurch ist der Anteil der Kommunikation an der Gesamtzeit beim GCPP wesentlich größer als beim GCel. Dies bedeutet, daß die Kommunikation auf dem GCPP zu deutlich höheren Effizienzverlusten führt als auf dem GCel.

Betrachten wir nun den Grobgitterlöser. Bei wachsender Anzahl N_m von Makroelementen erhöht sich die Dimension der Grobgitterprobleme. Damit benötigt einerseits die Solver-Task immer mehr Zeit zur Berechnung der Lösung und andererseits wächst der Zeitaufwand für die globale Kommunikation des rechte-Seite- und Lösungsvektors zwischen dem Tasknetzwerk und der Solver-Task. Für größere N_m überwiegt dabei die Lösungszeit gegenüber der Kommunikationszeit. Auf Grund der stärkeren Rechenleistung ist deshalb beim GCPP für eine gößere Anzahl von Makroelementen der Anteil des Grobgitterlösers an der Gesamtzeit geringer als beim GCel. Dies bedeutet, daß der nichtparallelisierte Grobgitterlöser bei einer größeren Anzahl von Makroelementen auf dem GCPP geringere Effizienzverluste verursacht als auf dem GCel.

Im folgenden wollen wir etwas näher darauf eingehen, warum man die Zeitanteile für die Kommunikation und den Grobgitterlöser als Effizienzverluste interpretieren kann. Sei $\beta_l \in (0,1)$ der Anteil der lokalen Kommunikation an der Gesamtzeit t_n^{par} des parallelen Algorithmus, d.h. $\beta_l t_n^{par}$ die Zeit für die lokale Kommunikation. Analog sei β_g der Anteil der Kommunikation zur Bestimmung globaler Normen und β_s der Anteil des Grobgitterlösers, der sich aus dem Anteil für das Lösen der Grobgitterprobleme und dem Anteil für die globale Kommunikation zwischen der Solver-Task und dem Tasknetzwerk zusammensetzt. Vereinfachend wollen wir annehmen, daß diese Anteile für alle $n-1$ Makroelement-Prozessoren gleich sind und daß es weder Wartezeiten auf Grund ungleicher Lastverteilung noch Doppelberechnungen gibt. Dann ist $(1-\beta)t_n^{par}$ mit $\beta := \beta_l + \beta_g + \beta_s$ die Zeit für sämtliche arithmetischen Operationen innerhalb einer Makroelement-Task. Schätzen wir die Zeit für das Lösen der Grobgitterprobleme durch t_n^{par} ab (die Solver-Task hat ebenfalls die Gesamtzeit t_n^{par}) und berücksichtigen wir, daß in der Zeit t_1^{par} keine Kommunikationszeiten enthalten sind, so erhalten wir die folgende Abschätzung

$$(n-1)(1-\beta)t_n^{par} \;\leq\; t_1^{par} \;\leq\; t_n^{par} + (n-1)(1-\beta)t_n^{par}\,.$$

Aus der Formel (8.2) ergibt sich damit für die parallele Effizienz E^{par}

$$(1-\beta) - \frac{1-\beta}{n} \leq E^{par} \leq (1-\beta) + \frac{\beta}{n}. \tag{8.6}$$

Für eine nicht allzu kleine Anzahl n von Prozessoren ist also die Zahl $\tilde{e} := 1 - \beta$ eine brauchbare Näherung für die parallele Effizienz E^{par}. Dies bedeutet, daß die Anteile β_l, β_g und β_s jeweils als Effizienzverluste angesehen werden können.

Anhand unseres Testbeispiels wollen wir uns nun einen Eindruck über die Größenordnung dieser Anteile verschaffen. In den folgenden Angaben beziehen sich $\beta_c := \beta_l + \beta_g$ und β_s jeweils auf die Gesamtzeit $\tilde{t}_n^{par}$ der nichtlinearen Iteration, d.h. wir vernachlässigen den Einfluß der Initialisierungsphase und betrachten nur die parallele Effizienz der nichtlinearen Iteration. Da der Anteil β_g in allen Testrechnungen sehr klein war (unterhalb von 0.02), haben wir aus Platzgründen β_l und β_g zu einem Kommunikationsanteil β_c zusammengefaßt. Tabelle 8.5 zeigt für verschiedene feinste Gitterlevel ℓ und verschiedene Anzahlen N_m von Makroelementen die Größen β_c, β_s und $\tilde{e} = 1 - \beta_c - \beta_s$ für den Rechner GCel. Entsprechende Resultate für den GCPP zeigt Tabelle 8.6. Dabei wurden

Tabelle 8.5: *Effizienzverluste β_c (Kommunikation) und β_s (Grobgitterlöser) auf dem Rechner GCel für verschiedene feinste Gitterlevel ℓ und wachsende Anzahl N_m von Makroelementen; $\tilde{e} = 1 - \beta_c - \beta_s$ ist eine Näherung für die parallele Effizienz*

	$\ell = 4$			$\ell = 5$			$\ell = 6$		
N_m	β_c	β_s	$\tilde{e}$	β_c	β_s	$\tilde{e}$	β_c	β_s	$\tilde{e}$
4	0.16	0.06	0.78	0.09	0.02	0.89	0.05	0.01	0.94
9	0.23	0.14	0.63	0.14	0.06	0.80	0.08	0.02	0.90
16	0.24	0.24	0.52	0.16	0.11	0.73	0.09	0.04	0.87
25	0.18	0.45	0.37	0.15	0.20	0.65	0.09	0.09	0.82
36	0.12	0.65	0.23	0.12	0.37	0.51	0.09	0.15	0.76
64	0.06	0.85	0.09	0.09	0.66	0.25	0.09	0.37	0.54
100	0.02	0.95	0.03	0.03	0.87	0.10	0.05	0.66	0.29

lediglich die Fälle $\ell \in \{6, 7, 8\}$ angegeben, da kleinere Werte von ℓ bei diesem Rechner keine praktisch sinnvolle Effizienz mehr liefern.

Aus den Tabellen können wir folgende Schlußfolgerungen ziehen. Erhöhen wir das Gitterlevel ℓ, so daß jedem Makroelement-Prozessor mehr Unbekannte zugewiesen werden, so verringert sich die Summe der Effizienzverluste $\beta = \beta_l + \beta_g + \beta_s$. Dies wird durch folgende Überlegung klar. Beim Übergang vom Level ℓ zum Level $\ell+1$ erhöht sich in einer Makroelement-Task pro Mehrgitterzyklus der Aufwand an arithmetischen Operationen etwa um den Faktor 4 (im 2D-Fall), während sich die Menge der lokal zu kommunizierenden Daten lediglich um den Faktor 2 erhöht. Die Zeit für die globale Kommunikation zur Bestimmung globaler Normen bleibt beim Übergang zu einem höheren Level unverändert. Der Aufwand zum Lösen der Grobgitterprobleme erhöht sich nur leicht und zwar dadurch, daß pro F-Zyklus nun ℓ statt vorher $\ell - 1$ Grobgitterprobleme zu lösen

Tabelle 8.6: *Effizienzverluste β_c, β_s und Näherung $\tilde{e}$ für die parallele Effizienz auf dem Rechner GCPP*

	$\ell = 6$			$\ell = 7$			$\ell = 8$		
N_m	β_c	β_s	$\tilde{e}$	β_c	β_s	$\tilde{e}$	β_c	β_s	$\tilde{e}$
4	0.33	0.025	0.65	0.15	0.009	0.84	0.07	0.003	0.93
9	0.46	0.045	0.49	0.25	0.018	0.73	0.11	0.006	0.88
16	0.54	0.057	0.41	0.30	0.024	0.68	0.15	0.008	0.84
25	0.57	0.084	0.34	0.34	0.039	0.62	0.16	0.014	0.82
36	0.54	0.127	0.33	0.34	0.061	0.60	0.17	0.022	0.80
64	0.51	0.198	0.29	0.34	0.105	0.56	0.18	0.039	0.78

sind. Das Verhältnis zwischen der Zeit $(1-\beta)t_n^{par}$ für die arithmetischen Operationen und der Zeit βt_n^{par} für die Kommunikation und den Grobgitterlöser verändert sich also beim Übergang zu Level $\ell+1$ zugunsten der arithmetischen Operationen. Dies bedeutet, daß die Summe β der Effizienzverluste sinkt.

Betrachten wir nun die Situation, daß das Gitterlevel ℓ fest ist und die Anzahl N_m der Makroelemente wächst. Während der Anteil β_c für die Kommunikation nach einer anfänglichen leichten Erhöhung nicht mehr wächst, erhöht sich der Anteil β_s für den Grobgitterlöser ständig und wird für hinreichend großes N_m zum dominierenden Anteil unter den Effizienzverlusten (gegenüber dem GCel tritt dieser Effekt auf dem GCPP erst bei größeren Werten von N_m ein). Dies kann folgendermaßen begründet werden. Auf der einen Seite bleibt in jeder Makroelement-Task der Aufwand für die arithmetischen Operationen und für die lokale Kommunikation mit den Nachbartasks für verschiedene N_m im wesentlichen gleich. Der Zeitaufwand für die Kommunikation zur Bestimmung globaler Normen wächst wie $O(\log N_m)$ (siehe Abschnitt 7.4); er ist jedoch sehr gering im Vergleich zu den anderen Zeitbestandteilen. Auf der anderen Seite steigt mit wachsendem N_m der Aufwand für den Grobgitterlöser relativ stark an, denn einerseits wächst der Aufwand für das Lösen der Grobgitterprobleme wie $O(N_m^\alpha)$ (α entsprechend der Komplexität des Lösungsverfahrens) und andererseits steigt der Aufwand für die globale Kommunikation zwischen der Solver-Task und dem Tasknetzwerk wie $O(N_m \log N_m)$.

Zusammenfassend wollen wir festhalten:

- Um eine akzeptable parallele Effizienz zu erreichen, sollte man den Prozessoren eine hinreichend große Anzahl von Unbekannten zuordnen. Falls das feinste Gitterlevel recht klein ist, ist es empfehlenswert, einem Prozessor mehrere Makroelement-Tasks zuzuordnen. Was "recht klein" konkret bedeutet, hängt von der jeweiligen Hardware-Situation ab.

- Im massiv-parallelen Fall ($N_m \gg 1$) verursacht der Grobgitterlöser große Verluste an paralleler Effizienz, während die anderen Effizienzverluste in einem praktisch vertretbaren Rahmen bleiben.

8.3 Verschiedene Grobgitterlöser

Im vorhergehenden Abschnitt haben wir gesehen, daß bei einer hohen Anzahl von Makroelementen der Grobgitterlöser den größten Verlust an paralleler Effizienz bewirkt. Für eine Verbesserung der Effizienz ist es also entscheidend, seinen Anteil an der Gesamtzeit zu reduzieren. Eine Möglichkeit, die wir in diesem Abschnitt untersuchen wollen, besteht darin, anstelle der bisher verwendeten augmented Lagrange Iteration ein anderes Lösungsverfahren einzusetzen. Eine weitere naheliegende Idee ist die Parallelisierung des Grobgitterlösers, mit der wir uns in den beiden folgenden Abschnitten beschäftigen wollen. Als alternative Verfahren zur Lösung der Grobgitterprobleme wählen wir daher nur solche aus, die sich auch recht gut parallelisieren lassen. Der augmented Lagrange Löser ist in dieser Hinsicht nicht sehr geeignet, da er direkte Methoden mit rekursiver Datenabhängigkeit beinhaltet.

Bei der Auswahl des Lösungsverfahrens kommt es auch auf einen geeigneten Kompromiß zwischen paralleler und numerischer Effizienz an. Der Extremfall hierbei ist der *triviale Löser*, bei dem die Makroelement-Tasks die Grobgitterkorrektur für das gröbste Gitter einfach weglassen. Dieser Löser hat zwar die beste parallele Effizienz ($\beta_s = 0$), aber gleichzeitig auch die schlechteste numerische Effizienz. In diesem Abschnitt vergleichen wir den augmented Lagrange Löser mit drei anderen Lösungsverfahren. Diese sind der Vanka-Glätter (siehe Abschnitt 6.3.1), das SIMPLE-Verfahren (siehe [47], [60]) und der oben erwähnte triviale Löser. Zum Vergleich verwenden wir unser driven-cavity-Beispiel aus den vorhergehenden Abschnitten mit Re=2000. Die Testrechnungen wurden auf dem Rechner GCel-1024 des "Paderborn Center for Parallel Computing" durchgeführt, bei dem wir maximal 1024 Prozessoren vom Typ T805 einsetzen können. Bei dem neueren Rechner GCPP haben wir einerseits nicht die Möglichkeit, den massiv-parallelen Fall auszutesten (maximal 192 Prozessoren), andererseits treten hier auf Grund der anderen Hardware-Konstellation größere Effizienzverluste durch den Grobgitterlöser erst bei einer noch größeren Anzahl von Prozessoren auf.

Tabelle 8.7 zeigt zu wachsender Anzahl N_m von Makroelementen die numerischen Konvergenzraten ρ^{mg} der Mehrgitter-Iteration (siehe (8.5)) und die Rechenzeiten t_n^{par} des parallelen Algorithmus für die vier verschiedenen Grobgitterlöser. Wir sehen, daß von den "nicht-trivialen" Grobgitterlösern für eine größere Anzahl von Makroelementen das SIMPLE-Verfahren das schnellste Verfahren ist. Der Vanka-Glätter schneidet zwar hier (bei einer Solver-Task) am schlechtesten ab, liefert aber - wie wir später sehen werden - die besten Resultate bei der Parallelisierung des Grobgitterlösers. Ab einer gewissen Anzahl von Makroelementen werden alle nicht-trivialen Grobgitterlöser auf Grund der wachsenden Größe der Grobgitterprobleme so teuer, daß der triviale Löser trotz seiner schlechten Konvergenzrate bessere Resultate liefert. Bei einem kleineren feinsten Gitterlevel ℓ gewinnt der triviale Löser sogar bei fast allen Werten von N_m, da hier der Vorteil der guten parallelen Effizienz gegenüber dem Nachteil der schlechteren Konvergenzrate überwiegt.

In den Testrechnungen für die Tabelle 8.7 wurde die Iteration des Grobgitterlösers abgebrochen, sobald der Anfangsdefekt mindestens um einen Faktor q_1=0.1 reduziert wurde

Tabelle 8.7: *Numerische Konvergenzraten ρ^{mg} der Mehrgitter-Iteration und Gesamtzeiten t_n^{par} des parallelen Algorithmus für die Grobgitterlöser* "V" *(Vanka-Glätter),* "S" *(SIMPLE-Verfahren),* "a.L." *(augmented Lagrange Methode) und* "no" *(trivialer Löser) bei unterschiedlichem feinsten Gitterlevel ℓ, Re=2000 und wachsender Anzahl N_m von Makroelementen (*"$*$" *bedeutet, daß dieser Fall aus Speicherplatzgründen nicht mehr gerechnet werden konnte)*

ℓ	N_m	ρ^{mg}				t_n^{par}			
		V	S	a.L.	no	V	S	a.L.	no
4	4	0.25	0.25	0.24	0.24	39	39	35	**33**
	9	0.34	0.34	0.35	0.37	70	65	64	**51**
	16	0.38	0.38	0.39	0.43	97	81	82	**48**
	25	0.43	0.44	0.43	0.59	181	112	127	**97**
	36	0.43	0.37	0.43	0.66	228	**84**	190	131
	64	0.44	0.44	0.39	0.74	709	250	376	**146**
	100	0.45	0.46	0.44	0.84	1503	483	1901	**222**
5	4	0.38	0.38	0.38	0.39	199	199	196	**192**
	9	0.44	0.44	0.45	0.51	256	**235**	263	284
	16	0.50	0.51	0.51	0.62	323	**304**	307	358
	25	0.42	0.47	0.42	0.69	338	274	**261**	405
	36	0.46	0.47	0.45	0.74	556	**372**	383	497
	64	0.50	0.51	0.49	0.82	1352	**656**	943	716
	100	0.50	0.52	0.50	0.86	2693	977	2623	**953**
6	4	0.50	0.50	0.50	0.51	898	**888**	893	960
	9	0.49	0.47	0.52	0.60	882	**832**	954	1238
	16	0.56	0.55	0.58	0.71	1165	1160	**1144**	1684
	25	0.58	0.60	0.58	0.76	1429	1311	**1274**	2102
	36	0.56	0.61	0.56	0.80	1648	1478	**1397**	2533
	64	0.53	0.54	0.53	0.85	2606	**1698**	1936	3357
	100	0.54	0.55	0.53	0.88	4507	**2214**	3998	4138
	400	0.51	0.53	$*$	0.93	42990	11910	$*$	**7360**

oder nach einer maximalen Anzahl von m_1=300 Iterationen. Durch die hohe maximale Iterationszahl lag das Schwergewicht also auf der Reduzierung des Anfangsdefektes. Die Effizienz des Vanka-Glätters und des SIMPLE-Verfahrens kann man im massivparallelen Fall verbessern, indem man die maximale Anzahl der Iterationen stärker beschränkt. Dadurch verschlechtert sich zwar die Konvergenzrate des Mehrgitterverfahrens, aber auf Grund des geringeren Verlustes an paralleler Effizienz steigt die Gesamteffizienz. Trotz der so erzielten Verbesserung bleibt jedoch für eine große Anzahl von Makroelementen der Anteil des Grobgitterlösers an der Gesamtzeit immer noch zu hoch. Eine weitere wesentliche Verbesserung können wir erreichen, indem wir anstelle von bisher einer Solver-Task mehrere einsetzen und den Grobgitterlöser parallelisieren.

8.4 Parallelisierung des Grobgitterlösers

In diesem Abschnitt beschreiben wir das prinzipielle Vorgehen, wie wir die Arbeit der Solver-Task S auf mehrere, parallel arbeitende Solver-Tasks $\{S_k\}$, $k = 1, \ldots, N_s$, verteilen. Als Parallelisierungstechnik verwenden wir wieder eine Datenpartitionierung entsprechend einer Zerlegung des Gebietes in Teilgebiete. Dabei nutzen wir dieselbe rekursive Gebietszerlegung wie für die Definition der globalen Kommunikation (siehe Abschnitt 7.4) und verwenden auf einer gewissen Stufe l die entsprechende Zerlegung in $N_s = 2^l$ Teilgebiete

$$\Omega \;=\; \bigcup\nolimits_{k=1}^{N_s} \Omega_k^{(l)} \qquad \text{mit} \qquad \Omega_k^{(l)} \;:=\; \bigcup\nolimits_{i \in I_k^{(l)}} M_i \,,$$

bei der jedes Teilgebiet aus einer gewissen Menge von Makroelementen besteht. Die Indexmengen $I_k^{(l)}$ seien disjunkt und möglichst gleichmächtig. Im folgenden verzichten wir auf die Angabe der Stufe l und benutzen die Bezeichnungen $\Omega_k := \Omega_k^{(l)}$ und $I_k := I_k^{(l)}$. Zur Lösung der Grobgitterprobleme ordnen wir jedem Teilgebiet Ω_k eine Solver-Task S_k zu. Sei T_{r_k} die Root-Task in der Baumstruktur der Makroelement-Tasks $\{T_i : i \in I_k\}$ zum Teilgebiet Ω_k (d.h. r_k ist der Rootindex $r_k^{(l)} \in I_k^{(l)}$ aus Abschnitt 7.4). Für die Kommunikation zwischen dem Task-Netzwerk $\{T_i\}$ und den Solver-Tasks verbinden wir jede Solver-Task S_k durch einen virtuellen Kanal mit der Root-Task T_{r_k}. Abbildung 8.2 zeigt am Beispiel des driven-cavity-Problems für eine 8x8-Makroelementzerlegung die entsprechende Gebietszerlegung in 4 Teilgebiete (Stufe l=2). In Abbildung 8.3 sind für

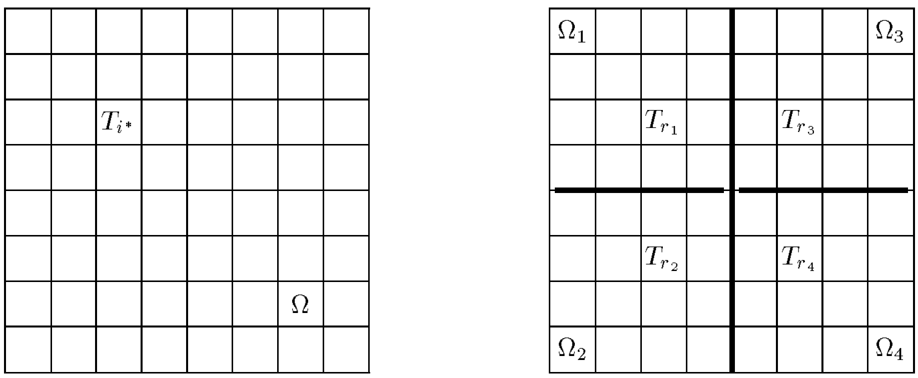


Abbildung 8.2: *Makroelementzerlegung des Gebietes Ω mit globaler Root-Task T_{i*} (links) und Gebietszerlegung in $N_s = 4$ Teilgebiete Ω_k mit entsprechenden Root-Tasks T_{r_k} (rechts)*

dieses Beispiel schematisch die 4 Solver-Tasks S_k mit den dazugehörigen Teilbäumen der Makroelement-Tasks $\{T_i : i \in I_k\}$ dargestellt. Wir wollen nun die Arbeit einer Solver-Task S_k näher beschreiben.

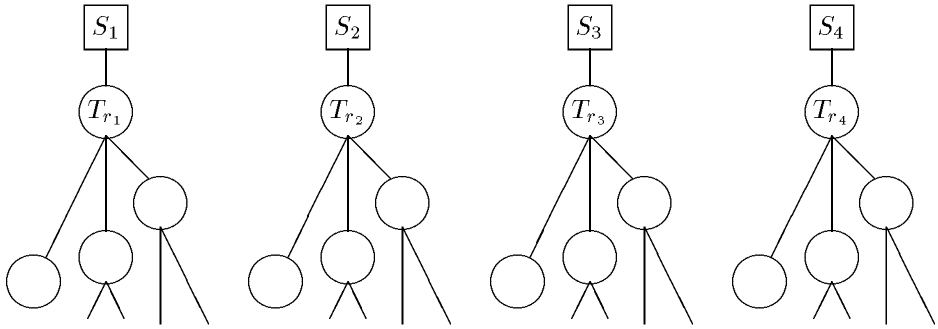


Abbildung 8.3: *Verbindung der Solver-Tasks S_k mit den Root-Tasks T_{r_k} der Teilbäume aus den Makroelement-Tasks zu den Teilgebieten Ω_k*

CHARAKTERISIERUNG DER SOLVER-TASK S_k:

(1) S_k ist zuständig für diejenigen Operationen des Grobgitterlösers, welche die Daten des Teilgebietes Ω_k betreffen.

(2) Bei der Kommunikation von rechte-Seite- und Lösungsvektoren zwischen Makroelement-Tasks und Solver-Tasks tauscht S_k lediglich Daten mit den Tasks $\{T_i : i \in I_k\}$ zum Teilgebiet Ω_k aus.

(3) Um bei der Lösung der Grobgitterprobleme Operationen mit Daten ausführen zu können, die zum Rand von Ω_k gehören, tauscht S_k Daten mit den Solver-Tasks S_j zu benachbarten Teilgebieten Ω_j aus (lokale Kommunikation).

(4) Für Operationen des Grobgitterlösers mit globalen Daten (z.B. Berechnung einer globalen Defektnorm) beteiligt sich S_k an einem globalen Informationsaustausch unter allen Solver-Tasks $\{S_j\}$ (globale Kommunikation).

Die lokale Kommunikation unter den Solver-Tasks können wir in analoger Weise zu den Makroelement-Tasks realisieren. Mit Hilfe des graphentheoretischen Zugangs aus Abschnitt 7.3 wird vor dem Start des parallelen Programms eine quasi-optimale Reihenfolge von Kommunikationstakten bestimmt und diese dann in der Initialisierungsphase allen Solver-Tasks zugeschickt.

Die globale Kommunikation der Solver-Tasks $\{S_k\}$ ergibt sich in natürlicher Weise aus der entsprechenden globalen Kommunikation unter den Root-Tasks $\{T_{r_k}\}$. Wenn zum Beispiel bei einer send-up-Operation im m-ten Kommunikationsschritt ($m > s-l$ mit s aus (7.7)) die Root-Task T_{r_k} ihre Daten an die Root-Task T_{r_j} schickt, so schickt bei der entsprechenden send-up-Operation der Solver-Tasks die Task S_k im $(m-s+l)$-ten Kommunikationsschritt ihre Daten an S_j. Eine send-up- bzw. send-down-Operation wird auf diese Weise in einer optimalen Anzahl von l Kommunikationsschritten realisiert.

Der Datentransfer eines Grobgittervektors (z.B. die rechte Seite des Grobgitterproblems) von den Makroelement-Tasks $\{T_i\}$ zu den Solver-Tasks $\{S_k\}$ wird gegenüber dem Fall einer Solver-Task in folgender Weise parallelisiert. Zunächst wird parallel in den einzelnen Taskblöcken $\{T_i : i \in I_k\}$ mit Hilfe einer send-up-Operation ($s - l$ Kommunikationsschritte) die entsprechende Information zum Teilgebiet Ω_k in der jeweiligen Root-Task T_{r_k} bereitgestellt (siehe Abbildung 8.3). In einem letzten Kommunikationsschritt schickt danach jede Root-Task T_{r_k} ihre Information an die zugeordnete Solver-Task S_k. Der Datentransfer einer berechneten Grobgitterlösung von den Solver-Tasks zu den Makroelement-Tasks erfolgt analog in umgekehrter Reihenfolge. Gegenüber dem Fall von einer Solver-Task ergeben sich zwei Vorteile. Einerseits verkleinert sich die Anzahl der Kommunikationsschritte von $s + 1$ auf $s + 1 - l$, was zu einer Verringerung in der Summe der start-up-Zeiten (Zeiten für den Verbindungsaufbau zu Beginn einer Kommunikation) führt. Andererseits wird die maximale Länge der zu verschickenden Datenpuffer bei N_s Solver-Tasks im Durchschnitt um den Faktor N_s reduziert, was eine Verringerung der Übertragungszeiten bewirkt.

Als Algorithmus zur parallelen Lösung der Grobgitterprobleme verwenden wir im wesentlichen den bei einer Solver-Task eingesetzten, sequentiellen Algorithmus (Vanka-Glätter oder SIMPLE-Verfahren). Um rekursive Datenabhängigkeiten zwischen den Solver-Tasks zu vermeiden, sind jedoch bei den Operationen mit Daten, die zum Rand der Teilgebiete Ω_k gehören, gewisse Modifikationen erforderlich. Beim Vanka-Glätter verwenden wir hierzu die in Abschnitt 7.2 beschriebene Technik, bei der zunächst jede Task S_k unabhängig neue Werte auf dem Rand von Ω_k berechnet und diese dann nach einer Kommunikation mit den Nachbartasks durch Mittelwertbildung abgleicht.
Bei der Geschwindigkeitsiteration des SIMPLE-Verfahrens berechnet jede Task S_k zunächst für alle Knoten auf dem Rand von Ω_k neue Werte nach dem Jacobi-Verfahren. Danach werden für alle inneren Knoten von Ω_k neue Werte nach dem Gauß-Seidel-Verfahren bestimmt. Bei der Druckkorrektur wird das globale SOR-Verfahren des sequentiellen Algorithmus folgendermaßen modifiziert. Jede Solver-Task S_k führt unabhängig eine SOR-Iteration für ihre "eigenen" Druckwerte auf Ω_k durch. Für die dabei benötigten Druckwerte aus den benachbarten Teilgebieten werden stets die alten Werte von der vorhergehenden SOR-Iteration verwendet, welche jeweils zu Beginn eines Iterationsschrittes durch lokale Kommunikation bereitgestellt werden.

Wie sich die Parallelisierung des Grobgitterlösers auf die Gesamteffizienz des Verfahrens auswirkt, wollen wir an einem Beispiel im Falle des Vanka-Glätters demonstrieren. Wir betrachten das driven-cavity-Problem mit einer äquidistanten 8x8-Zerlegung in Makroelemente und drei verschiedenen Gebietszerlegungen $\{\Omega_k\}$ in 2, 4 und 8 Teilgebiete (siehe Abbildung 8.4). Für die 64 Makroelement-Tasks verwenden wir 64 Prozessoren T805 des GCel. Eine Solver-Task S_k wird jeweils demjenigen Prozessor zugeordnet, der auch die entsprechende Root-Task T_{r_k} bearbeitet, d.h. die Gesamtanzahl der verwendeten Prozessoren beträgt 64. Sei t_{nl} die durchschnittliche Zeit für eine nichtlineare Iteration und t_{slv} die Gesamtzeit für das Lösen der Grobgitterprobleme innerhalb eines nichtlinearen Schrittes. Tabelle 8.8 zeigt für verschiedene feinste Gitterlevel ℓ das Verhalten von t_{nl} und t_{slv} bei wachsender Anzahl N_s von Solver-Tasks (als Reynolds-Zahl

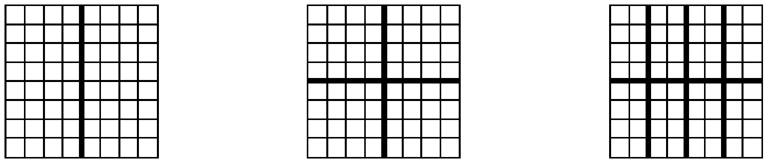


Abbildung 8.4: *Gebietszerlegungen beim driven-cavity-Beispiel für verschiedene Anzahlen von Solver-Tasks*

wurde der Wert Re=2000 verwendet). Beim Übergang von einer zu acht Solver-Tasks

Tabelle 8.8: *Rechenzeiten t_{nl} und t_{slv} in Sekunden bei der Parallelisierung des Grobgitterlösers (Vanka-Glätter) in Abhängigkeit von der Anzahl N_s der Solver-Tasks und dem feinsten Gitterlevel ℓ*

ℓ	Zeit	$N_s = 1$	$N_s = 2$	$N_s = 4$	$N_s = 8$
4	t_{nl}	20.2	10.8	7.5	6.4
	t_{slv}	17.1	8.2	5.0	3.9
5	t_{nl}	37.6	22.0	16.8	14.7
	t_{slv}	26.1	13.1	7.9	5.8
6	t_{nl}	96.4	60.6	52.0	48.3
	t_{slv}	44.2	22.1	13.5	9.8

wurde die Zeit t_{slv} unabhängig vom Gitterlevel etwa um den Faktor 4.5 reduziert. Die entsprechende Reduzierung der Gesamtzeit t_{nl} für einen nichtlinearen Schritt hängt jedoch vom Gitterlevel ab. Während wir bei $\ell = 4$ etwa den Faktor 3 erreichen, ergibt sich bei $\ell = 6$ lediglich ein Reduktionsfaktor von 2. Wenn also im Falle einer Solver-Task der Anteil des Grobgitterlösers an der Gesamtzeit recht hoch ist (wie z.B. bei $\ell = 4$), so führt die Parallelisierung des Grobgitterlösers zu einer wesentlichen Reduzierung der Gesamtzeit und somit (da sich die Anzahl der verwendeten Prozessoren nicht erhöht hat) zu einer wesentlichen Verbesserung der parallelen Effizienz.

8.5 Vergleich der verschiedenen Löser im massivparallelen Fall

Anhand numerischer Testrechnungen wollen wir in diesem Abschnitt untersuchen, welche der verschiedenen Varianten unseres parallelen Algorithmus die beste Gesamtzeit und damit die beste Gesamteffizienz liefert. Dabei sind wir insbesondere am massivparallelen Fall interessiert. Als Testbeispiel verwenden wir wieder das driven-cavity-Problem mit Re=2000 und einer $M \times M$-Zerlegung in $N_m = M^2$ Makroelemente. Die

Testrechnungen wurden auf dem Parsytec-Rechner GCel-1024 des "Paderborn Center for Parallel Computing" durchgeführt. Wir verwenden für jede Makroelement-Task T_i, $i = 1, \dots, N_m$, einen eigenen Prozessor und ordnen die Solver-Task S_k, $k = 1, \dots, N_s$, jeweils dem Prozessor zu, der die entsprechende Root-Task T_{r_k} bearbeitet, d.h. wir verwenden insgesamt $n = N_m$ Prozessoren.

Zunächst wollen wir am Beispiel einer 24x24-Zerlegung in N_m=576 Makroelemente untersuchen, wie sich die Gesamtzeit t_n^{par} des parallelen Algorithmus in Abhängigkeit von der Anzahl der Solver-Tasks verhält. Tabelle 8.9 zeigt für den Vanka-Glätter und das SIMPLE-Verfahren als Grobgitterlöser die entsprechenden Rechenzeiten auf dem GCel-1024. Für den Vanka-Glätter erhalten wir das beste Resultat jeweils bei der höchsten

Tabelle 8.9: *Gesamtzeit (in Sekunden) in Abhängigkeit von der Anzahl N_s der Solver-Tasks für verschiedene feinste Gitterlevel ℓ beim driven-cavity-Beispiel mit 576 Makroelement-Tasks; als Grobgitterlöser wurden der Vanka-Glätter oder das SIMPLE-Verfahren verwendet*

ℓ	Grobgitter-löser	$N_s = 8$	$N_s = 32$	$N_s = 64$	$N_s = 128$	$N_s = 256$
4	Vanka	$*$	1334	894	745	**620**
	SIMPLE	1872	1319	**1141**	1411	1502
5	Vanka	$*$	2447	1728	1408	**1107**
	SIMPLE	3463	2574	**2050**	2346	2615
6	Vanka	$*$	4501	3497	2900	**2274**
	SIMPLE	5665	4696	**3747**	4201	4760

Anzahl von 256 Solver-Tasks. Beim SIMPLE-Verfahren lohnt es sich dagegen nicht, sehr viele Solver-Tasks einzusetzen. Hier liefern 64 Solver-Tasks die kürzeste Gesamtzeit. Die Ursache liegt darin, daß sich mit zunehmender Anzahl von Solver-Tasks durch die kleiner werdenden Teilgebiete Ω_k die numerische Effizienz des parallelisierten SIMPLE-Lösers immer mehr verschlechtert.

Die Resultate beim Vanka-Glätter führen zu der Idee, den Extremfall auszutesten, bei dem jedes Teilgebiet Ω_k nur aus einem Makroelement besteht. In diesem Fall ist es überflüssig, Solver-Tasks einzurichten, da ihre Arbeit von den Makroelement-Tasks übernommen werden kann. In einem Schritt dieses Grobgitterlösers berechnet zunächst jede Task T_i die Unbekannten neu, die zu dem einen Grobgitterelement $K = M_i$ gehören, und gleicht dann nach lokaler Kommunikation mit den Nachbartasks T_j, $j \in \mathcal{N}(i)$, die gemeinsamen Geschwindigkeitswerte durch Mittelwertbildung ab (siehe Abschnitt 7.2). Es wird also unser paralleles Glättungsverfahren auf dem gröbsten Gitterlevel angewendet. Wir bezeichnen diesen Grobgitterlöser im folgenden mit "Vanka-all", da stets alle Makroelement-Tasks an der Berechnung der Grobgitterlösung beteiligt sind. Beim Abbruchkriterium wäre es hier zu teuer, nach jedem Schritt die Defektnorm der Näherungslösung durch globale Kommunikation zu bestimmen. Wir führen deshalb den Abbruchtest nur nach jeder 5. Iteration durch.

Im folgenden wollen wir anhand des driven-cavity-Beispiels die verschiedenen Varianten unseres parallelen Algorithmus miteinander vergleichen. Tabelle 8.10 zeigt zu wachsender Anzahl N_m von Makroelement-Tasks die Gesamtzeit t_n^{par} des parallelen Algorithmus auf $n = N_m$ Prozessoren für folgende Varianten des Grobgitterlösers: den Vanka-Glätter mit einer Solver-Task ("V-1"), den parallelisierten Vanka-Glätter ("V-par") mit der günstigsten Anzahl N_s von Solver-Tasks, das SIMPLE-Verfahren mit einer Solver-Task ("S-1"), das parallelisierte SIMPLE-Verfahren ("S-par") mit der günstigsten Anzahl N_s von Solver-Tasks, den in Abschnitt 8.3 beschriebenen trivialen Grobgitterlöser ("no") sowie das oben beschriebene Verfahren "Vanka-all" ("V-all"). Die Iteration bei den

Tabelle 8.10: *Gesamtzeiten t_n^{par} (in Sekunden) auf dem Rechner GCel für die verschiedenen parallelen Algorithmen bei wachsender Anzahl N_m von Makroelement-Tasks für das feinste Gitterlevel $\ell = 4$ und $\ell = 6$; es wurden $n = N_m$ Prozessoren verwendet; bei der Variante "V-all" ist $\widetilde{e} = 1 - \beta_l - \beta_g$ eine Näherung für die parallele Effizienz; ρ^{mg} bezeichnet die numerische Konvergenzrate der Mehrgitter-Iteration*

ℓ	N_m	V-1	V-par		S-1	S-par		no	V-all		
		t_n^{par}	N_s	t_n^{par}	t_n^{par}	N_s	t_n^{par}	t_n^{par}	t_n^{par}	$\widetilde{e}$	ρ^{mg}
4	4	39	4	39	39	2	39	**32**	60	0.61	0.25
	9	70	4	65	65	2	63	**51**	108	0.48	0.34
	16	97	8	79	81	2	75	**48**	124	0.45	0.38
	25	181	8	122	112	2	100	**97**	199	0.35	0.42
	36	228	16	116	**84**	2	100	131	143	0.42	0.43
	64	709	32	143	250	16	**130**	146	153	0.37	0.38
	100	1503	32	341	483	4	294	**222**	262	0.33	0.44
	400	17650	128	667	4283	16	983	1201	**465**	0.26	0.52
	576	*	256	620	7618	64	1141	1175	**413**	0.26	0.52
	1024	*	256	600	*	256	1273	*	**430**	0.25	0.52
6	4	897	4	**885**	888	2	897	960	969	0.90	0.50
	9	882	4	848	**832**	2	841	1238	980	0.84	0.49
	16	1165	8	**1095**	1160	4	1134	1684	1250	0.82	0.56
	25	1429	8	1246	1311	2	**1228**	2102	1380	0.77	0.55
	36	1648	16	**1302**	1478	4	1369	2533	1324	0.80	0.54
	64	2606	32	1452	1698	4	**1433**	3357	1488	0.75	0.53
	100	4507	32	1933	2214	4	1774	4138	**1766**	0.72	0.54
	400	42990	128	2873	11910	16	3578	7360	**2347**	0.63	0.58
	576	*	256	2274	19150	64	3747	7542	**1979**	0.65	0.55
	1024	*	256	2661	*	512	3694	*	**2374**	0.63	0.64

Grobgitterlösern wurde abgebrochen, sobald der Anfangsdefekt mindestens um den Faktor $q_1 = 0.1$ reduziert wurde oder nach einer maximalen Anzahl von m_1 Iterationen. Dabei haben wir für die einzelnen Varianten folgende Werte von m_1 verwendet: m_1=300 für "V-1" und "S-1", m_1=70 für "V-par" und "S-par" und m_1=75 für "V-all".

Wir sehen, daß ab einer gewissen Anzahl von Makroelementen die Parallelisierung des Grobgitterlösers sowie die kleinere maximale Schrittzahl m_1 zu einer wesentlichen Verbesserung in der Gesamteffizienz führen. Der triviale Grobgitterlöser "no" liefert zwar im Fall eines kleinen feinsten Gitterlevels ℓ bei einer moderaten Anzahl von Makroelementen die beste Gesamtzeit, ist jedoch für größere ℓ und im massiv-parallelen Fall nicht konkurrenzfähig zu den Varianten "V-par", "S-par" und "V-all".

Während bei einer Solver-Task das SIMPLE-Verfahren "S-1" dem Vanka-Glätter "V-1" überlegen ist, erweist sich bei den parallelisierten Versionen im massiv-parallelen Fall der Vanka-Glätter "V-par" als das bessere Verfahren. Ein wesentlicher Grund hierfür ist, daß bei der Parallelisierung des SIMPLE-Verfahrens im Vergleich zum Vanka-Glätter ein deutlich höherer Anteil an lokaler Kommunikation entsteht.

Im massiv-parallelen Fall liefert von allen betrachteten parallelen Lösern die Variante "Vanka-all" die beste Gesamtzeit. Um einen Eindruck über die Größenordnung ihrer parallelen Effizienz zu gewinnen, haben wir in Tabelle 8.10 die Werte $\tilde{e}{:=}1-\beta_l-\beta_g$ dargestellt. Dabei bedeuten die Größen β_l und β_g wie in Abschnitt 8.2 jeweils die Anteile der lokalen und globalen Kommunikation an der Gesamtzeit t_n^{par} des parallelen Algorithmus. Vernachlässigen wir den Anteil der Wartezeiten auf Grund ungleicher Lastverteilung sowie den Anteil an Doppelberechnungen, so sind β_l und β_g die einzigen Effizienzverluste und wir können $\tilde{e}$ als Näherung für die parallele Effizienz ansehen. Die numerische Effizienz der Variante "Vanka-all" können wir bei dieser Problemgröße (mehr als $5 \cdot 10^6$ Unbekannte bei $\ell = 6$ und $N_m = 1024$) nicht mehr praktisch (wie in Abschnitt 8.1) bestimmen. Deshalb geben wir in Tabelle 8.10 nur die numerische Konvergenzrate ρ^{mg} der Mehrgitter-Iteration (siehe (8.5)) an. Für $\ell = 6$ ist im Ansatz die Tendenz erkennbar (Übergang von $N_m{=}400$ zu $N_m{=}1024$), daß sich ρ^{mg} bei sehr großer Anzahl von Makroelementen verschlechtert. Dies wird dadurch verursacht, daß die größer werdenden Grobgitterprobleme auf Grund der Beschränkung der Iterationszahl ($m_1 = 75$) immer ungenauer gelöst werden. Damit verschlechtert sich bei sehr großer Anzahl von Makroelementen die numerische Effizienz. In dem von uns betrachteten Bereich bis 1024 Prozessoren bleibt sie jedoch noch in einem praktisch vertretbaren Rahmen. Durch eine Erhöhung der maximalen Iterationszahl m_1 würde sich zwar die numerische Effizienz verbessern, gleichzeitig aber die parallele Effizienz verschlechtern, da auf dem gröbsten Gitter das Verhältnis zwischen Kommunikations- und Rechenzeit extrem ungünstig ist. In Testrechnungen mit verschiedenen Werten von m_1 ergaben sich im Bereich $N_m \in [4, 1024]$ für $m_1 = 75$ die besten Resultate.

Insgesamt können wir einschätzen, daß unser paralleler Algorithmus "Vanka-all" im massiv-parallelen Fall bis 1024 Prozessoren eine zufriedenstellende Gesamteffizienz liefert. Bei einer noch größeren Anzahl von Makroelementen bzw. Prozessoren wäre es jedoch erforderlich, einen anderen Grobgitterlöser mit einer besseren numerischen Effizienz einzusetzen. Empfehlenswert wäre es, hierfür ein Mehrgitter-Verfahren zu verwenden, bei dem das Grobgitter mit der hohen Anzahl von Makroelementen stufenweise weiter vergröbert wird. Dabei tritt das Problem auf, zu einem beliebig gegebenen Grobgitter geeignete gröbere Gitter zu definieren. Einen interessanten Lösungsvorschlag hierzu findet man in [29]. Bei der Realisierung eines derartigen Mehrgitter-Verfahrens wird es dann wieder sinnvoll sein, eine gewisse Anzahl der in Abschnitt 8.4 beschriebenen Solver-Tasks $\{S_k\}$ einzuführen.

Literaturverzeichnis

[1] Bastian, P.: *Parallele adaptive Mehrgitterverfahren*, Dissertation, Universität Heidelberg, 1994.

[2] Bastian, P.: *Dynamic Load Balancing for Parallel Adaptive Multigrid Methods on Unstructured Meshes*, In: [72], 1-12.

[3] Becker, R., Rannacher, R.: *Finite element solution of the incompressible Navier-Stokes equations on anisotropically refined meshes.* In: [31], p. 52-62.

[4] Blum, H., Harig, J., Müller S.: **FEAT** . *Finite element analysis tools. Release 1.1. User Manual*, Technischer Report Nr. 554, SFB 123, Universität Heidelberg, January 1990.

[5] Braess, D., Verfürth, R.: *Multigrid methods for nonconforming finite element methods*, SIAM J. Numer. Anal. **27**, 979-986 (1990).

[6] Brandt, A., Dinar, D.: *Multigrid solutions to elliptic flow problems.* In "Numerical Methods for Partial Differential Equations", S. Parter (ed.), 53-147 (1979).

[7] Brenner, S.C., Scott, L.R.: *The Mathematical Theory of Finite Element Methods*, Texts in Applied Mathematics 15, Springer-Verlag, New York, 1994.

[8] Brezzi, F., Fortin, M.: *Mixed and Hybrid Finite Element Methods*, Springer-Verlag, New York, 1991.

[9] Cai, X.-C., Gropp, W.D., Keyes, D.E.: *Convergence rate estimate for a domain decomposition method*, Numer. Math. **61**, 153-169 (1992).

[10] Chan, T.F., Glowinski, R., Périaux, J., Widlund, O.B. (Eds.): *Second International Symposium on Domain Decomposition Methods*, SIAM, Philadelphia, PA, 1989.

[11] Chan, T.F., Glowinski, R., Périaux, J., Widlund, O.B. (Eds.): *Third International Symposium on Domain Decomposition Methods*, SIAM, Philadelphia, PA, 1990.

[12] Ciarlet, Ph.G.: *The Finite Element Method for Elliptic Problems*, North–Holland–Publ. Comp., Amsterdam/New York, 1978.

[13] Crouzeix, M., Raviart, P.A.: *Conforming and Nonconforming Finite Element Methods for Solving the Stationary Stokes Equations*, RAIRO Numer. Anal. **3**, 33-76 (1973).

[14] deCougny, H.L., Devine, K.D., Flaherty, J.E., Loy, R.M., Özturan, C., Shephard, M.S.: *Load balancing for the parallel adaptive solution of partial differential equations*, Appl. Numer. Math. **16**, 157-182 (1994).

[15] Dorok, O., Grambow, W., Tobiska, L.: *Aspects of Finite Element Discretizations for Solving the Boussinesq Approximation of the Navier-Stokes Equations.* In [31], p. 50-61.

[16] Dorok, O.: *Eine stabilisierte Finite Elemente Methode zur Lösung der Boussinesq-Approximation der Navier-Stokes-Gleichungen*, Dissertation, Universität Magdeburg, 1995.

[17] Dupont, T., Scott, R.: *Polynomial approximation of functions in Sobolev spaces*, Math. Comp. **34,** 441-463 (1979).

[18] Durst, F., Perić, M., Schäfer, M., Schreck, E.: *Parallelization of Efficient Numerical Methods for Flows in Complex Geometries.* In [35], p. 79-92.

[19] Even, S., Monien, B.: *On the number of rounds necessary to disseminate information.* In: Proc. 1st ACM Symp. on Parallel Algorithms and Architectures, 318-327, 1989.

[20] Fortin, M., Glowinski, R.: *Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary Value Problems*, North Holland, Amsterdam, New York, Oxford, 1983.

[21] Girault, V., Raviart, P.A.: *An analysis of upwind schemes for the Navier-Stokes equations,* SIAM J. Numer. Anal. **19**, 312-333 (1982).

[22] Girault, V., Raviart, P.A.: *Finite Element Methods for Navier–Stokes equations*, Springer, Berlin–Heidelberg 1986.

[23] Golub, G.H., Van Loan, C.F.: *Matrix Computations*, The Johns Hopkins University Press, 1989.

[24] Grambow, W., Risch, U., Schieweck, F.: *Experiences with the Multigrid Method Applied to High Reynolds Number, Steady, Incompressible Flow*, Preprint Math 6/90, TU Magdeburg, February 1990.

[25] Haase, G., Langer, U.: *On the use of multigrid preconditioners in the domain decomposition method.* In [28], 101-110.

[26] Haase, G., Langer, U., Meyer, A.: *Domain decomposition preconditioners with inexact subdomain solvers*, J. Numer. Lin. Alg. Appl. **1,** no.1, 27-41 (1992).

[27] Hackbusch, W.: *Multi–Grid Methods and Applications*, Springer, Berlin–Heidelberg, 1985.

[28] Hackbusch, W. (Ed.): *Parallel Algorithms for Partial Differential Equations*, Notes on Numerical Fluid Mechanics, Vol. 31, Vieweg-Verlag 1991.

[29] Hackbusch, W., Sauter, S.A.: *A New Finite Element Approach for Problems Containing Small Geometric Details*, Technical Report 95-06, Math. Sem., Universität Kiel, 1995.

[30] Hackbusch, W., Wittum, G. (Eds.): *Fast Solvers for Flow Problems*, Notes on Numerical Fluid Mechanics, Vol. 49, Vieweg-Verlag 1995.

[31] Hebeker, F.K., Rannacher, R., Wittum, G. (Eds.): *Numerical Methods for the Navier–Stokes Equations*, Notes on Numerical Fluid Mechanics, Vol. 47, Vieweg-Verlag 1994.

[32] Hempel, R., Schüller, A.: *Experiments with parallel multigrid algorithms using the SUPRENUM communications library.* GMD Arbeitspapiere 141, GMD St. Augustin, Germany, 1988.

[33] Heywood, J.G., Rannacher, R.: *Finite element approximation of the nonstationary Navier-Stokes problem, Part I: Regularity of solutions and second-order error estimates for spatial discretization*, SIAM J.Numer. Anal. **19**, no. 2, 275-311 (1982).

[34] Heywood, J., Rannacher, R., Turek, S.: *Artificial boundaries and flux and pressure conditions for the incompressible Navier-Stokes equations*, Int. J. Num. Meth. in Fluids, **22,** 325-352 (1996).

[35] Hirschel, E.H. (Ed.): *Flow Simulation with High-Performance Computers I.* Notes on Numerical Fluid Mechanics, Vol. 38, Vieweg-Verlag 1993.

[36] Hughes, T.J.R., Brooks, A.: *Streamline upwind/ Petrov Galerkin formulations for the convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations*, Comp. Meth. Appl. Mech. Eng. **54,** 199-259 (1982).

[37] Hughes, T.J.R., Franca, L.P., Balestra, M.: *A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accomodating equal order interpolation*, Comp. Meth. Appl. Mech. Eng. **59,** 85-99 (1986).

[38] Johnson, C.: *On computability and error control in CFD*, Int. J. Num. Meth. in Fluids, **20,** 777-788 (1995).

[39] Johnson, C., Rannacher, R.: *On error control in CFD.* In [31], p. 133-144.

[40] Langer, A.: *Zur Konstruktion diskret divergenzfreier Basen für nicht-konforme Finite-Elemente-Diskretisierungen des Stokes-Problems im* $\mathbf{R}^3$, Diplomarbeit, Universität Dortmund, 1993.

[41] Linden, J., Steckel, B., Stüben, K.: *Parallel multigrid solution of the Navier-Stokes equations on general 2D-domains*, Parallel Computing **7**, 461-475 (1988).

[42] Lonsdale, G., Schüller, A.: *Multigrid efficiency for complex flow simulations on distributed memory machines,* Parallel Computing **19**, 23-32 (1993).

[43] McBryan, O.A., Frederickson, P.O., Linden, J., Schüller, A., Solchenbach, K., Stüben, K., Thole, C.A., Trottenberg, U.: *Multigrid methods on parallel computers - a survey of recent developments*, Impact Comput. Sci. and Engrg. **3,** 1-75 (1991).

[44] Özturan, C., deCougny, H.L., Shephard, M.S., Flaherty, J.E.: *Parallel adaptive mesh refinement and redistribution on distributed memory computers*, Comput. Meth. Appl. Mech. Engrg. **119**, 123-137 (1994).

[45] Ohmori, K., Ushijima, T.: *A Technique of Upstream Type Applied to a Linear Nonconforming Finite Element Approximation of Convective Diffusion Equations*, RAIRO Numer. Anal. 18, 309–332 (1984).

[46] Oosterlee, C.W., Wesseling, P.: *A robust multigrid method for a discretization of the incompressible Navier-Stokes equations in general coordinates,* Impact Comput. Sci. and Engrg. **5,** 128-151 (1993).

[47] Patankar, S.V., Spalding, D.B.: *A calculation procedure for heat and mass transfer in threedimensional parabolic flows*, Int. J. Heat Mass Transfer **15**, 1787-1806 (1972).

[48] Rannacher, R., Turek, S.: *Simple Nonconforming Quadrilateral Stokes Element*, Numer. Methods Partial Differential Equations **8,** 97-111 (1992).

[49] Rannacher, R.: *On the numerical solution of the incompressible Navier-Stokes equations*, Z. Angew. Math. Mech. **73**, no. 9, 203-216 (1993).

[50] Reichert, H., Wittum, G.: *On the construction of robust smoothers for incompressible flow problems.* In [31], p. 207-216.

[51] Reusken, A.: *Steplength optimization and linear multigrid methods*, Numer. Math. **58,** no 8, 819-835 (1991).

[52] Risch, U., Schieweck, F.: *A Multigrid Method for Solving the Stationary Navier–Stokes Equations by FEM*, Preprint Math 5/88, Technical University Magdeburg, August 1988.

[53] Schäfer, M., Schreck, E., Wechsler, K.: *An Efficient Parallel Solution Technique for the Incompressible Navier-Stokes Equations.* In [31], p. 228-238.

[54] Schieweck, F.: *A parallel multigrid algorithm for solving the Navier–Stokes equations*, Impact Comput. Sci. and Engrg. **5,** 345-378 (1993).

[55] Schieweck, F.: *On the Order of Two Nonconforming Finite Element Approximations of Upwind Type for the Navier–Stokes Equations.* In [31], p. 249-258.

[56] Schieweck, F.: *Multigrid Convergence Rates of a Sequential and a Parallel Navier–Stokes Solver.* In: [30], p. 251-262.

[57] Schieweck, F., Tobiska, L.: *A nonconforming finite element method of upstream type applied to the stationary Navier–Stokes equation*, MMAN, 23, 627–647 (1989).

[58] Schieweck, F., Tobiska, L.: *An Error Estimate of Optimal Order for a Nonconforming Upwind Finite Element Discretization of the Navier–Stokes Equations.* Preprint Nr. 13, Fakultät für Mathematik, Universität Magdeburg, 1994. to appear in: *Numerical Methods for Partial Differential Equations.*

[59] Schieweck, F., Tobiska, L.: *A Parallelization Strategy for a Navier–Stokes Multigrid Solver Based on Macroelements.* In: [72], 176-187.

[60] Schreck, E., Perić, M.: *Computation of fluid flow with a parallel multigrid solver*, Int. J. Num. Meth. in Fluids **16,** 303-327 (1993).

[61] Schreiber, P., Turek, S.: *An Efficient Finite Element Solver for the Nonstationary Incompressible Navier-Stokes Equations in Two and Three Dimensions.* In [31], p. 268-277.

[62] Temam, R.: *Navier–Stokes Equations – Theory and Numerical Analysis*, North Holland Publ., 1979.

[63] Thiele, A., Tobiska, L.: *A weighted upwind finite element method for solving the stationary Navier–Stokes equations*, Wiss. Z. Techn. Univ. Magdeburg **33,** 13-20 (1989).

[64] Tobiska, L.: *A three-dimensional nonconforming finite element method of upstream type and its application to the Navier Stokes equations.* In: Proceedings of the workshop International Seminar on Applied Mathematics 1991, H.-G. Roos, A. Felgenhauer and L. Angermann, Eds., TU Dresden, 1991.

[65] Tobiska, L.: *Full and weighted upwind finite element methods.* In: Splines in Numerical Analysis, J.W. Schmidt and H. Spät, Eds., Mathematical Research, Vol. **52**, Akademie–Verlag Berlin, 1989.

[66] Tobiska, L., Verfürth, R.: *Analysis of a streamline diffusion finite element method for the Stokkes and Navier-Stokes equations*, to appear in SIAM J. Numer. Anal. 1996.

[67] Turek, S.: *Ein robustes und effizientes Mehrgitterverfahren zur Lösung der instationären, inkompressiblen, 2D Navier–Stokes–Gleichungen mit diskret divergenzfreien finiten Elementen*, Thesis, Universität Heidelberg, 1991.

[68] Turek, S.: *Tools for simulating non-stationary incompressible flow via discretely divergence-free finite element models*, Int. J. Num. Meth. in Fluids, **18,** 71-107 (1994).

[69] Turek, S.: *On discrete projection methods for the incompressible Navier-Stokes equations: An algorithmical approach*, Preprint (SFB 359), Universität Heidelberg, 1994.

[70] Turek, S.: *A comparative study of time stepping techniques for the incompressible Navier-Stokes equations: From fully implicit nonlinear schemes to semi-implicit projection methods*, Preprint (SFB 359), Universität Heidelberg, 1995. to appear in: Int. J. Num. Meth. in Fluids 1996.

[71] Vanka, S.: *Block-implicit multigrid calculation of two-dimensional recirculating flows*, Comp. Meth. Appl. Mech. Engrg., Vol. **59**, no.1, 29-48 (1986).

[72] Wagner, S. (Ed.): *Computational Fluid Dynamics on Parallel Systems*, Notes on Numerical Fluid Mechanics, Vol. 50, Vieweg-Verlag, 1995.

[73] Walther, H., Nägler, G.: *Graphen, Algorithmen, Programme*, Fachbuchverlag Leipzig, 1987.

[74] Wesseling, P.: *An introduction to multigrid methods*, John Wiley, Chichester, 1992.

[75] Wittum, G.: *On the Convergence of Multi-Grid Methods with Transforming Smoothers. Theory with Applications to the Navier-Stokes Equations*, Numer. Math. **57**, 15-38 (1990).

[76] Wittum, G.: *Distributive Iterationen für indefinite Systeme*, Dissertation, Univ. Kiel 1986.