

Vorkonditionierungsstrategien für das stabilisierte Oseen-Problem

Theorie und Anwendung

Diplomarbeit

vorgelegt von
Timo Johannes Heister
aus
Hannover

angefertigt am
Institut für Numerische und Angewandte Mathematik
der
Georg-August-Universität zu Göttingen
2008

Inhaltsverzeichnis

Einleitung	1
1 Grundlagen	3
1.1 Funktionalanalysis	3
1.1.1 Funktionenräume	3
1.1.2 Lösbarkeit, Eindeutigkeit und Approximation von Sattelpunktproblemen	8
1.2 Matrixidentitäten und Eigenwertprobleme	10
1.2.1 Darstellungen von Inversen	10
1.2.2 Symmetrie von Matrizen	11
1.2.3 Eigenwerte	12
1.3 Numerische Lineare Algebra	13
1.3.1 Generalized Minimal Residual Method (GMRES)	13
1.3.2 Vorkonditionierung	17
1.3.3 Flexible Generalized Minimal Residual Method (FGMRES)	19
2 Problemstellung und Diskretisierung	21
2.1 Behandlung der instationären Navier-Stokes-Gleichungen	21
2.1.1 Zeitdiskretisierung	22
2.1.2 Linearisierung durch Fixpunktiteration	23
2.2 Oseen-Problem	23
2.2.1 Herleitung der Variationsformulierung	24
2.3 Finite-Elemente-Diskretisierung	26
2.3.1 Triangulierung	26
2.3.2 Finite-Elemente-Räume	27
2.3.3 Matrixdarstellung des Variationsproblems	30
2.3.4 Lösbarkeit des diskreten Problems	31
2.4 Stabilisierung	31
2.4.1 Grad-Div-Stabilisierung	32
2.4.2 Residuale Stabilisierung	33
2.4.3 Stabilisierung durch lokale Projektion	34
2.5 Betrachtung des linearen Systems	35
2.5.1 Drucknormierung des diskreten Systems	38
2.5.2 Eigenschaften der Massematrix	39
2.5.3 Definition des Schurkomplements	40
2.5.4 Spezialfall: Stokes-Problem	40
2.5.5 Eigenwertbetrachtungen	41

3	Vorkonditionierungsstrategien	47
3.1	Klassifizierung der Vorkonditionierer	47
3.1.1	Blockdiagonal-Vorkonditionierung	48
3.1.2	Blockdreiecks-Vorkonditionierung	50
3.2	Behandlung der Teilprobleme	51
3.2.1	Approximation des Geschwindigkeitsblocks	51
3.2.2	Approximation des Schurkomplements	51
3.3	Vorkonditionierer	52
3.3.1	Blockdiagonal-Vorkonditionierer von Silvester und Wathen	52
3.3.2	Pressure-convection-diffusion-Vorkonditionierer	53
3.3.3	BFBt-Vorkonditionierer	54
3.3.4	Olshanskii-Vorkonditionierer	55
3.3.5	Grad-Div-Vorkonditionierer	60
4	Numerische Ergebnisse	65
4.1	Implementierung und verwendete Bibliotheken	65
4.2	Testprobleme	66
4.2.1	Problem 1: glattes Oseen-Problem	66
4.2.2	Problem 2: Lid-Driven Cavity	66
4.2.3	Problem 3: Beltrami-Flow	67
4.3	Implementierungsdetails zu den Vorkonditionierern	67
4.4	Numerische Tests	68
4.4.1	Grad-Div-Vorkonditionierer	68
4.4.2	Vergleichsrechnungen	71
4.4.3	Anwendung	73
4.4.4	Stationäres Navier-Stokes-Problem	76
4.5	Eigenwertplots	77
5	Fazit	83
	Literaturverzeichnis	86

Einleitung

Die Navier-Stokes-Gleichungen beschreiben die Strömung in inkompressiblen newtonschen Flüssigkeiten oder Gasen und werden zur numerischen Strömungssimulation verwendet. Es handelt sich um ein nichtlineares System partieller Differentialgleichungen zweiter Ordnung, bei dem eine Geschwindigkeit u und ein zugehöriger Druck p gesucht werden, die folgendes System erfüllen:

$$\begin{aligned}\frac{\partial u}{\partial t} - \nu \Delta u + (u \cdot \nabla)u + \nabla p &= f, \\ \nabla \cdot u &= 0,\end{aligned}$$

wobei neben der Viskosität ν und den äußeren Kräften f noch Rand- und Anfangsbedingungen zu stellen sind. Die beiden Gleichungen stellen die Impuls- beziehungsweise Masseerhaltung des Fluids sicher. Diskretisierung der Zeit und Auflösung der Nichtlinearität im Term $(u \cdot \nabla)u$ durch eine Fixpunktiteration führen zu den sogenannten verallgemeinerten Oseen-Gleichungen, die nun linear und stationär sind:

$$\begin{aligned}-\nu \Delta u + cu + (w \cdot \nabla)u + \nabla p &= f, \\ \nabla \cdot u &= 0.\end{aligned}$$

Zur Durchführung der Fixpunktiteration muss dieses Problem zu jedem Zeitschritt mehrfach für unterschiedliche w und f gelöst werden.

Die schnelle Lösung des Oseen-Problems ist essentiell für die numerische Strömungssimulation. Aufgrund der Problemgröße ist eine direkte Lösung des linearen Gleichungssystems nicht sinnvoll, weswegen iterative Löser verwendet werden müssen. Die Lösungsgeschwindigkeit von iterativen Lösungsverfahren wie dem GMRES-Verfahren hängt von der Eigenwertverteilung der Systemmatrix ab. Da das Gleichungssystem schlecht konditioniert sein kann, ist eine Vorkonditionierung von großer Bedeutung, wodurch die Eigenwertverteilung des Systems für den iterativen Löser verbessert wird.

Diese Arbeit behandelt die Analyse verschiedener Vorkonditionierer, die die Struktur des Oseen-Problems nutzen. Mit einer stabilen Finite-Elemente-Diskretisierung und symmetrischer oder ohne Stabilisierung hat die Systemmatrix Sattelpunktgestalt und lässt sich als Blockmatrix notieren:

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

Es lässt sich zeigen, dass Vorkonditionierer in Blockdreiecks-Gestalt der Form

$$P^{-1} = \begin{pmatrix} \tilde{A} & B^T \\ 0 & \tilde{S} \end{pmatrix}^{-1}$$

effektiv zur Lösung sind. Die Schwierigkeit bei der Verwendung liegt darin, eine gute aber schnell zu berechnende Approximation an die Inverse des Schurkomplements $S := -BA^{-1}B^T$ zu bilden. Neben der Vorstellung einiger klassischer Vorkonditionierer wird in dieser Arbeit ein Vorkonditionierer aus [OB06] untersucht, der die Approximation von S durch eine algebraische Veränderung der Systemmatrix M vereinfacht. Auf dieser Idee aufbauend, wird der sogenannte Grad-Div-Vorkonditionierer entwickelt, analysiert und getestet.

Ein weiterer Aspekt bei der Lösung des Oseen-Problems ist die Verminderung von unphysikalischen Oszillationen durch Stabilisierung des Systems. Die Stabilisierung hat einen nicht zu vernachlässigenden Einfluß auf die algebraischen Eigenschaften des Systems und damit auf den Vorkonditionierer. Der Grad-Div-Vorkonditionierer erreicht die algebraische Veränderung der Systemmatrix, die zur einfachen Approximation von S nötig ist, durch Verwendung von Grad-Div-Stabilisierung.

In Kapitel 1 werden die notwendigen mathematischen Grundlagen eingeführt, die das Verständnis der Problemstellung ermöglichen. Neben der Funktionalanalysis zur Problembeschreibung werden einige Grundlagen der linearen Algebra zur späteren Analyse der Eigenwerte der vorkonditionierten Systeme benötigt. Zuletzt wird das GMRES-Verfahren als bekannter Vertreter der Krylowraum-Verfahren zur iterativen Lösung linearer Gleichungssysteme erläutert.

In Kapitel 2 wird die Problemstellung der Navier-Stokes-Gleichungen und des Oseen-Problems formuliert. Mit der Finite-Elemente-Diskretisierung wird das lineare Gleichungssystem zum Oseen-Problem definiert und analysiert. Ein weiterer wichtiger Aspekt ist die Stabilisierung und deren Einfluß auf das lineare System.

Kapitel 3 enthält die theoretische Behandlung verschiedener Vorkonditionierer für das diskretisierte Oseen-Problem. Nach Möglichkeit werden Abschätzungen für die Eigenwertverteilung der vorkonditionierten Systeme gegeben, die Aufschluss auf die Konvergenzgeschwindigkeit liefern. Zentral ist hierbei die Abhängigkeit von der Gitterfeinheit h und der Viskosität ν des Fluids.

Den Abschluß der Arbeit liefert Kapitel 4 mit der numerischen Analyse der in Kapitel 3 vorgestellten Vorkonditionierer.

Kapitel 1

Grundlagen

In diesem Kapitel werden die für das Verständnis der Arbeit nötigen mathematischen Grundlagen behandelt. So werden zunächst die zur korrekten Formulierung notwendigen Funktionenräume motiviert und eingeführt. Des Weiteren gibt es einen kurzen Überblick über Lösbarkeit, Eindeutigkeit und Approximation von Sattelpunktproblemen, welche zur Untersuchung der Lösbarkeit des Oseen-Problems später benötigt werden. Zuletzt wird noch ein iterativer Löser zur Lösung des entstehenden linearen Gleichungssystems mit zugehöriger Konvergenzanalyse behandelt.

1.1 Funktionalanalysis

1.1.1 Funktionenräume

In den folgenden Abschnitten sei $\Omega \subset \mathbb{R}^d$, mit $d \in \{2, 3\}$ ein **beschränktes Gebiet** (d.h. offen und zusammenhängend) mit Lipschitzrand und es bezeichnet $\bar{\Omega}$ den **Abschluß** von Ω und $\partial\Omega = \bar{\Omega} \setminus \Omega$ den **Rand** von $\bar{\Omega}$.

Definition 1.1: Sei $u : \Omega \rightarrow \mathbb{R}$, dann bezeichnet man mit

$$\text{supp } u := \overline{\{x \in \Omega : u(x) \neq 0\}}$$

den **Träger** der Funktion u (also den Abschluß in Ω von der Menge von Punkten in Ω an denen u nicht verschwindet).

Definition 1.2: Es sei $\alpha = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}_0^N$ ein N -Tupel nicht-negativer ganzer Zahlen, ein sogenannter **Multiindex**. Mit $|\alpha| := \sum_i \alpha_i$ sei die **Ordnung** von α bezeichnet.

Dann definiert

$$D^\alpha := \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \dots \partial x_N^{\alpha_d}}$$

den Differentiationsoperator zum Multiindex α . Die Komponente α_i gibt dabei an, wie oft in Richtung x_i partiell differenziert werden soll.

Des Weiteren sind der Nabla- und der Laplace-Operator durch

$$\nabla := \left(\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_d} \right)^T, \quad \Delta := \sum_{k=1}^d \frac{\partial^2}{\partial x_k^2}$$

gegeben.

Definition 1.3: Es bezeichne $C(\Omega)$ den **Raum der reellwertigen stetigen Funktionen** $u : \Omega \rightarrow \mathbb{R}$ auf Ω und für $k \in \mathbb{N} \cup \{\infty\}$ bezeichne $C^k(\Omega)$ den **Raum aller k -mal stetig partiell differenzierbaren Funktionen** auf Ω . Mit $C^k(\overline{\Omega})$ werden die bis auf den Rand stetig fortsetzbaren k -mal stetig partiell differenzierbaren Funktionen bezeichnet:

$$C^k(\overline{\Omega}) = \{u : \Omega \rightarrow \mathbb{R} \mid D^\alpha u \in C(\Omega) \text{ und stetig fortsetzbar auf } \overline{\Omega}, \forall |\alpha| \leq k\}.$$

Mit punktweiser Addition und Multiplikation und folgenden Normen werden diese Räume zu Banachräumen:

$$\|u\|_{C(\Omega)} := \sup_{x \in \Omega} |u(x)|,$$
$$\|u\|_{C^k(\overline{\Omega})} := \sup_{x \in \overline{\Omega}} \sum_{|\alpha| \leq k} |D^\alpha u(x)|.$$

Des Weiteren werden später Funktionen benötigt, die auf dem Rand von Ω verschwinden:

Definition 1.4: Es sei

$$C_0^k(\Omega) := \{u \in C^k(\Omega) : \text{supp } u \subset \Omega \text{ ist kompakt}\}$$

mit $k \in \mathbb{N} \cup \{\infty\}$ die Menge der k -mal differenzierbaren Funktionen mit kompaktem Träger.

Satz 1.5: Für $y, v \in C^1(\overline{\Omega})$ gilt die **Formel der partiellen Integration**

$$\int_{\Omega} v(x) \frac{\partial}{\partial x_i} y(x) \, dx = \int_{\partial\Omega} v(x) y(x) \cdot n_i(x) \, ds - \int_{\Omega} y(x) \frac{\partial}{\partial x_i} v(x) \, dx, \quad (1.1)$$

wobei $n_i(x)$ die i -te Komponente des äußeren Normalenvektors am Punkt $x \in \partial\Omega$ ist.

Für die Lösungstheorie von Differentialgleichungen ist der Begriff der klassischen Differentiation zu restriktiv. Dies lässt sich exemplarisch an der Poisson-Gleichung

$$-\Delta u = f \text{ in } \Omega, \quad u = 0 \text{ auf } \partial\Omega. \quad (1.2)$$

auf einem Einheitsquadrat verdeutlichen. Man findet für eine unstetige rechte Seite f keine Lösung $u \in C^2$ zu (1.2). Unstetige äußere Kräfte treten in physikalischen Modellen häufig auf, so dass die Betrachtung von Lösungen mit schwächeren Regularitätsanforderungen sinnvoll ist.

Deswegen wird einen "schwächerer" Lösungsbegriff benötigt, so dass z.B. beim Poisson-Problem die Lösung u nicht mehr zweimal differenzierbar sein muss. Dadurch wird die Differentialgleichung nicht mehr "stark" erfüllt, aber dafür in einem "schwachen" Sinn. Die Grundlagen dafür werden in den folgenden zwei Abschnitten gegeben.

Räume Lebesgue-integrierbarer Funktionen

Die Basis werden Räume integrierbarer Funktionen bilden. Die Grundlagen über Lebesgue-Integration werden vorausgesetzt, können aber in [Alt99] Kapitel A.1 nachgelesen werden. Es bezeichne μ das Lebesgue-Maß.

Definition 1.6: Sei $1 \leq p \leq \infty$ eine ganze Zahl. Für meßbare Funktionen $u : \Omega \rightarrow \mathbb{R}$ betrachten wir die Äquivalenzrelation, die gerade solche Funktionen in einer Äquivalenzklasse zusammenfasst, die sich nur auf einer Menge mit Maß 0 unterscheiden.

Dann bezeichne $L^p(\Omega)$ die Menge aller Äquivalenzklassen mit

$$\|u\|_{L^p(\Omega)} < \infty \quad (\text{für } p < \infty),$$

dabei sei

$$\|u\|_{L^p(\Omega)} := \left(\int_{\Omega} |u(x)|^p \, dx \right)^{1/p},$$

bzw. $L^\infty(\Omega)$ die Menge aller Äquivalenzklassen von fast überall gleichmäßig beschränkten Funktionen in Ω , d.h.

$$L^\infty(\Omega) := \left\{ u : \Omega \rightarrow \mathbb{R} \text{ meßbar} \mid \exists C > 0 : \|u\|_{L^\infty(\Omega)} \leq C \right\}$$

mit

$$\|u\|_{L^\infty(\Omega)} := \operatorname{ess\,sup}_{x \in \Omega} |u(x)|. \quad (1.3)$$

Bemerkung 1.7: Mit $\operatorname{ess\,sup}$ in (1.3) bezeichnen wir das **essentielle Supremum** einer Funktion $u : \Omega \rightarrow \mathbb{R}$:

$$\begin{aligned} \operatorname{ess\,sup} u &:= \inf \{ c \geq 0 \mid u(x) \leq c \text{ fast überall} \} \\ &= \inf \{ c \geq 0 \mid \mu(\{x \mid u(x) > c\}) = 0 \}. \end{aligned}$$

Das essentielle Supremum darf also von u auf Nullmengen des Lebesgue-Maßes μ überschritten werden.

Satz 1.8: Sei $1 \leq p \leq \infty$, dann ist $\|u\|_{L^p(\Omega)}$ eine Norm auf $L^p(\Omega)$ und $L^p(\Omega)$ wird mit geeigneter Definition von Addition und Skalarmultiplikation auf den Äquivalenzklassen und der Norm zu einem normierten Raum.

Mit dem Skalarprodukt

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

wird der $L^2(\Omega)$ zum Hilbertraum.

Beweis: Einen Beweis findet man in [Alt99], 1.10. \square

Satz 1.9: Für $1 \leq p \leq \infty$ ist $L^p(\Omega)$ mit $\|\cdot\|_{L^p(\Omega)}$ vollständig und damit sogar Banachraum.

Beweis: Dies ist für $p < \infty$ der Satz von Fischer/Riesz. Ein Beweis findet sich zum Beispiel in [Lub06] Theorem 6.15. Für $p = \infty$ sei auf Lemma 1.11 in [Alt99] verwiesen. \square

Satz 1.10: Die Menge $C_0^\infty(\Omega)$ liegt dicht in $L^p(\Omega)$ für $1 \leq p < \infty$.

Beweis: Siehe [Lub06], Satz 7.8. \square

Sobolev-Räume

Mit der Definition von schwacher Differentiation lassen sich schließlich die Sobolev-Räume einführen.

Definition 1.11: *Es bezeichne*

$$L^1_{\text{loc}}(\Omega) := \left\{ u : \Omega \longrightarrow \mathbb{R} \text{ meßbar} \mid \int_K |u(x)| \, dx < \infty \quad \forall K \subset \Omega \text{ kompakt} \right\}$$

die Menge der **lokal integrierbaren Funktionen** in Ω .

Definition 1.12: *Sei $u \in L^1_{\text{loc}}(\Omega)$ und α ein Multiindex. Falls eine Funktion $w \in L^1_{\text{loc}}(\Omega)$ mit*

$$\int_{\Omega} u(x) D^{\alpha} v(x) \, dx = (-1)^{|\alpha|} \int_{\Omega} w(x) v(x) \, dx \quad \forall v \in C_0^{\infty}(\Omega)$$

existiert, so heißt w **schwache Ableitung** von u der Ordnung α . Bezeichnet wird sie mit $D^{\alpha}u := w$ ebenso wie die klassische Ableitung.

Wenn die klassische Ableitung $D^{\alpha}u$ existiert, so stimmt die schwache Ableitung damit überein. Die schwache Ableitung ist dabei so konstruiert, dass sie gerade die Formel der partiellen Integration (1.1) erfüllt.

Definition 1.13: *Sei $1 \leq p \leq \infty$ und $k \in \mathbb{N}$, dann bezeichnet*

$$W^{k,p}(\Omega) := \{v \in L^p(\Omega) \mid \exists D^{\alpha}v \in L^p(\Omega) \quad \forall \alpha : |\alpha| \leq k\}$$

den Raum der zur p -ten Potenz integrierbaren Funktionen mit schwachen Ableitungen der Ordnung k , die wieder zur p -ten Potenz integrierbar sind. Der Raum $W^{k,p}(\Omega)$ wird als **Sobolev-Raum** bezeichnet.

Satz 1.14: *Der $W^{k,p}(\Omega)$ wird mit der Norm*

$$\|u\|_{W^{k,p}} = \begin{cases} \left(\sum_{|\alpha| \leq k} \|D^{\alpha}u\|_{L^p(\Omega)}^p \right)^{1/p} & \text{für } 1 \leq p < \infty, \\ \sum_{|\alpha| \leq k} \|D^{\alpha}u\|_{L^{\infty}(\Omega)} & \text{für } p = \infty \end{cases}$$

zum Banach-Raum.

Beweis: Siehe [Alt99], 1.15. □

Von besonderem Interesse ist der Sobolev-Raum mit $k = 1$ und $p = 2$, der durch

$$H^1(\Omega) := W^{1,2}(\Omega)$$

ausgezeichnet wird. Die H^1 -Norm lässt sich dabei auch durch

$$\begin{aligned} \|u\|_{H^1(\Omega)} = \|u\|_1 &= \left(\int_{\Omega} \left[u^2 + \left(\frac{\partial u}{\partial x_1} \right)^2 + \cdots + \left(\frac{\partial u}{\partial x_d} \right)^2 \right] dx \right)^{1/2} \\ &= \left(\|u\|_{L^2(\Omega)}^2 + \|\nabla u\|_{L^2(\Omega)}^2 \right)^{1/2} \end{aligned}$$

ausdrücken.

Definition 1.15: Wir definieren weiter die H^1 -Seminorm durch

$$\begin{aligned} |u|_1 &:= \left(\sum_{|\alpha|=1} \|D^\alpha u\|_{L^2(\Omega)}^2 \right)^{1/2} \\ &= \left(\int \left[\left(\frac{\partial u}{\partial x_1} \right)^2 + \cdots + \left(\frac{\partial u}{\partial x_d} \right)^2 \right] dx \right)^{1/2} \\ &= \|\nabla u\|_{L^2(\Omega)}. \end{aligned}$$

Betrachten wir eine Funktion $u \in H^1(\Omega)$, so können wir keine Aussage über die Funktionswerte von u auf dem Rand $\partial\Omega$ treffen, da wir u dort beliebig abändern können und trotzdem in der gleichen Äquivalenzklasse in $L^2(\Omega)$ bleiben.

Weil wir die Sobolev-Räume später zur Darstellung von Lösungen verwenden wollen und Randbedingungen Teil der Differentialgleichung sind, benötigen wir Kontrolle auf dem Rand.

Definition 1.16: Wir bezeichnen den Abschluß von $C_0^\infty(\Omega)$ unter der Norm $\|\cdot\|_{H^1(\Omega)}$ mit $H_0^1(\Omega)$.

Satz 1.17: Der $H_0^1(\Omega)$ ist mit dem H^1 -Skalarprodukt

$$(u, v)_{H^1(\Omega)} := \sum_{|\alpha| \leq 1} \int D^\alpha u D^\alpha v \, dx$$

ein Hilbertraum.

Beweis: Der $H_0^1(\Omega)$ ist abgeschlossener Unterraum des H^1 und damit mit $\|\cdot\|_{H^1(\Omega)}$ ein Banachraum. Die Norm lässt sich über das H^1 -Skalarprodukt durch $\|u\|_{H^1(\Omega)} = (u, u)_{H^1(\Omega)}$ definieren, wodurch der $H_0^1(\Omega)$ zum Hilbertraum mit dem H^1 -Skalarprodukt wird. \square

Satz 1.18 (Poincaré-Friedrichs-Ungleichung): Es existiert eine Konstante $C > 0$, die vom beschränkten Gebiet Ω abhängt, so dass gilt:

$$\|u\|_1 \leq C|u|_1 \quad \forall u \in H_0^1(\Omega).$$

Beweis: Siehe [Alt99] 4.10. \square

Satz 1.19: Die volle H_1 -Norm und die H_1 -Seminorm sind äquivalente Normen auf dem $H_0^1(\Omega)$.

Beweis: Nach Definition gilt für beliebige $u \in H_0^1(\Omega)$

$$|u|_1 \leq \|u\|_1$$

und die andere Richtung gilt nach Satz 1.18. \square

Satz 1.20: Für $u \in H_0^1(\Omega)$ gilt:

$$|u|_1 = \|\nabla u\| \geq \|\nabla \cdot u\|.$$

Beweis: Siehe [ESW05] Gleichung (5.167). \square

1.1.2 Lösbarkeit, Eindeutigkeit und Approximation von Sattelpunktproblemen

Um die Lösbarkeit des Oseen-Problems zu klären, kann man sich auf die allgemeine Lösbarkeitstheorie von Sattelpunktproblemen zurückziehen. Doch selbst bei diesem abstrakten Ansatz, ist es möglich neben hinreichenden Kriterien für Existenz und Eindeutigkeit von Lösungen Fehlerabschätzungen zu zeigen.

Seien V, Q reelle Hilberträume mit den zugehörigen Normen $\|\cdot\|_V$ bzw. $\|\cdot\|_Q$. Es bezeichne V^*, Q^* die Dualräume von V bzw. Q und mit $\langle \cdot, \cdot \rangle$ seien die Dualitätsprodukte zwischen den Räumen bezeichnet.

Seien weiter

$$\begin{aligned} a(\cdot, \cdot) &: V \times V \longrightarrow \mathbb{R}, \\ b(\cdot, \cdot) &: V \times Q \longrightarrow \mathbb{R} \end{aligned}$$

stetige Bilinearformen, $f \in V^*$ und $g \in Q^*$.

Gegenstand der weiteren Betrachtungen ist nun das gemischte Variationsproblem

$$\begin{cases} \text{finde } (u, p) \in V \times Q : \\ a(u, v) + b(v, p) = \langle f, v \rangle & \forall v \in V, \\ b(u, q) = \langle g, q \rangle & \forall q \in Q. \end{cases} \quad (1.4)$$

Wir halten nun die erste Variable in der Bilinearform $a(\cdot, \cdot)$ fest und erhalten eine Linearform $a(u, \cdot) \in V^*$. Nach dem Riesz'schen Darstellungssatz (siehe [Alt99], Satz 4.6) existiert zu $a(u, \cdot)$ ein eindeutiger Operator $A : V \longrightarrow V^*$ mit

$$\langle Au, v \rangle = a(u, v) \quad \forall u, v \in V.$$

Analog definieren wir $B : V \longrightarrow Q^*$ zur Bilinearform $b(\cdot, \cdot)$.

Des Weiteren definieren wir $B^* : Q^{**} \longrightarrow V^*$ als dualen Operator von B , wobei wir Q^{**} mit Q identifizieren. Dabei ist das Bild $B^*q := z$ von $q \in Q$ gegeben durch:

$$\langle Bu, q \rangle = \langle u, z \rangle \quad \forall u \in V.$$

Mit den Operatoren A, B und B^* erhalten wir ein zu (1.4) äquivalentes Operatorgleichungssystem:

$$\begin{cases} \text{finde } (u, p) \in V \times Q : \\ Au + B^*p = f, \\ Bu = g. \end{cases} \quad (1.5)$$

Wir definieren nun dazu den Operator $\Lambda : V \times Q \longrightarrow V^* \times Q^*$ mit

$$\Lambda(u, p) := (Au + B^*p, Bu). \quad (1.6)$$

Definition 1.21: Eine Bilinearform $a : V \times V \longrightarrow \mathbb{R}$ heißt W -elliptisch mit $W \subseteq V$, falls ein $\delta > 0$ existiert mit

$$a(v, v) \geq \delta \|v\|_V^2 \quad \forall v \in W.$$

Dabei wird δ als **Elliptizitätskonstante** bezeichnet.

Zur weiteren Betrachtung benötigen wir den Teilraum

$$Z := \{v \in V \mid b(v, q) = 0 \forall q \in Q\} \subseteq V.$$

Satz 1.22: Sei $a(\cdot, \cdot)$ Z -elliptisch mit Elliptizitätskonstante $\delta > 0$ und $b(v, q)$ erfülle die Babuška-Brezzi-Bedingung, d.h.

$$\exists \beta > 0 : \inf_{q \in Q \setminus \{0\}} \sup_{v \in V \setminus \{0\}} \frac{b(v, q)}{\|v\|_V \|q\|_Q} \geq \beta.$$

Dann ist der Operator Λ ein Isomorphismus und es existiert der Lösungsoperator Λ^{-1} . Damit hat das gemischte Variationsproblem (1.4) genau eine Lösung.

Des Weiteren gilt für die Lösung (u, p) die Abschätzung

$$\|(u, p)\|_{V \times Q} \leq M(\alpha, \delta, \beta) \cdot \|(f, g)\|_{V^* \times Q^*}$$

wobei

$$\begin{aligned} M(\alpha, \delta, \beta) &:= \max\{\delta^{-1} + \beta^{-1}(1 + \alpha\delta^{-1}), (\beta^{-1} + \alpha\beta^{-2})(1 + \alpha\delta^{-1})\}, \\ \alpha &:= \|A\|. \end{aligned}$$

Beweis: siehe [Bre74], Corollary 1.1 und Proposition 1.1. \square

Approximation des gemischten Problems

Wir sind nun daran interessiert, die Lösbarkeit des gemischten Problems (1.4) in Unterräumen zu untersuchen. Eine Anwendung wäre hier die Approximation der kontinuierlichen Räume durch endlichdimensionale Unterräume. Von Interesse ist hier neben der Lösbarkeit des approximierenden Problems auch die Güte der approximierten Lösung im Vergleich zur ursprünglichen Lösung.

Seien V_h und Q_h abgeschlossene Teilräume von V bzw. Q . Dann lautet das approximierende Problem

$$\begin{cases} \text{finde } (u, p) \in V_h \times Q_h : \\ a(u, v) + b(v, p) = \langle f, v \rangle & \forall v \in V_h, \\ b(u, q) = \langle g, q \rangle & \forall q \in Q_h. \end{cases} \quad (1.7)$$

Analog zu Z definieren wir nun $Z_h := \{v \in V_h \mid b(v, q) = 0 \forall q \in Q_h\} \subseteq V_h$. Im Allgemeinen gilt hier jedoch $Z_h \not\subseteq Z$.

Eine Antwort auf die Frage der Lösbarkeit von (1.7) bietet der folgende Satz:

Satz 1.23: Sei $a(\cdot, \cdot)$ Z_h -elliptisch mit $\delta^* > 0$ und $b(v, q)$ erfülle die diskrete Babuška-Brezzi-Bedingung, d.h. es existiert eine von der Wahl der Räume V_h und Q_h unabhängige Konstante $\beta^* > 0$, so dass gilt:

$$\inf_{q \in Q_h \setminus \{0\}} \sup_{v \in V_h \setminus \{0\}} \frac{b(v, q)}{\|v\|_V \|q\|_Q} \geq \beta^*. \quad (1.8)$$

Dann hat das approximierende Variationsproblem (1.7) genau eine Lösung $(u_h, p_h) \in V_h \times Q_h$ und es gilt für die Lösung $(u, p) \in V \times Q$ des gemischten Problems (1.4) zur rechten Seite (f, g) die Abschätzung

$$\|u - u_h\| + \|p - p_h\| \leq \sigma_h \left(\inf_{v_h \in V_h} \|u - v_h\| + \inf_{q_h \in Q_h} \|p - q_h\| \right) \quad (1.9)$$

mit

$$\sigma_h := 1 + M(\alpha, \delta^*, \beta^*)(\alpha + \beta), \quad \alpha := \|A\|, \quad \beta := \|B\|$$

und M aus Satz 1.22.

Beweis: siehe [Bre74], Theorem 2.1. \square

Die Abschätzung (1.9) liefert anschaulich eine Beschränkung des Fehlers der Lösung des approximierenden Problems durch die Bestapproximation in den Unterräumen V_h und Q_h der richtigen Lösung.

1.2 Matrixidentitäten und Eigenwertprobleme

Im Folgenden werden wir einige Sätze über Matrizen und Eigenwerte benötigen, die hier kurz eingeführt werden sollen.

1.2.1 Darstellungen von Inversen

Lemma 1.24: Seien X, Y, Z Matrizen passender Größe, dabei X und Z regulär. Dann gilt:

$$\begin{pmatrix} X & Y \\ 0 & Z \end{pmatrix}^{-1} = \begin{pmatrix} X^{-1} & -X^{-1}YZ^{-1} \\ 0 & Z^{-1} \end{pmatrix}. \quad (1.10)$$

Beweis: Nachrechnen des Matrixprodukts liefert den Beweis. \square

Folgender Satz beschreibt, wie eine Störung der Matrix A sich als Störung ihrer Inversen ausdrücken lässt. Für $D = I$ findet man diesen Satz in [GV96].

Satz 1.25 (Woodbury-Matrixidentität): Seien A, U, D und V Matrizen passender Größe und seien A, D und $(D^{-1} + VA^{-1}U)$ invertierbar. Dann gilt:

$$(A + UDV)^{-1} = A^{-1} - A^{-1}U(D^{-1} + VA^{-1}U)^{-1}VA^{-1}.$$

Beweis: Nachrechnen ergibt:

$$\begin{aligned} & [A + UDV] \left[A^{-1} - A^{-1}U(D^{-1} + VA^{-1}U)^{-1}VA^{-1} \right] \\ &= I + UDVA^{-1} - [A + UDV]A^{-1}U(D^{-1} + VA^{-1}U)^{-1}VA^{-1} \\ &= I + UDVA^{-1} - (U + UDVA^{-1}U)(D^{-1} + VA^{-1}U)^{-1}VA^{-1} \\ &= I + UDVA^{-1} - UD(D^{-1} + VA^{-1}U)(D^{-1} + VA^{-1}U)^{-1}VA^{-1} \\ &= I + UDVA^{-1} - UDVA^{-1} \\ &= I. \quad \square \end{aligned}$$

1.2.2 Symmetrie von Matrizen

Die in diesem Abschnitt betrachteten Matrizen sind, falls nicht anders erwähnt, über dem Körper der komplexen Zahlen \mathbb{C} definiert.

Vorweg benötigen wir einige Definitionen:

Definition 1.26: Sei $A = (a_{ij}) \in \mathbb{C}^{n \times n}$ eine beliebige quadratische Matrix. **Transponieren** kennzeichnen wir mit A^T , **transponieren und invertieren** mit A^{-T} . Wir bezeichnen mit

$$A^* = \overline{A^T} = (\overline{a_{ji}})$$

die **Komplexkonjugation von A**. Gilt

$$A^* = A,$$

so heißt A **hermitesch** beziehungsweise für eine rein reellwertige Matrix A **symmetrisch**. Gilt hingegen

$$A^* = -A,$$

so heißt A **schief-hermitesch** beziehungsweise **schief-symmetrisch**. Ist

$$A^{-1} = A^*,$$

so ist A **unitär** beziehungsweise **orthogonal** für eine Matrix A mit rein reellen Einträgen.

Definition 1.27: Der **Rayleigh-Quotient** zur quadratischen Matrix A und zum Vektor $x \in \mathbb{C}^n$ ist definiert durch

$$R_A(x) := \frac{x^* Ax}{x^* x}. \quad (1.11)$$

Satz 1.28: Die Menge der Rayleigh-Quotienten

$$G[A] := \{R_A(x) \mid x \neq 0\} \quad (1.12)$$

zur Matrix A enthält die Eigenwerte von A .

Beweis: Sei λ Eigenwert von A mit zugehörigem Eigenvektor v , d.h. $Av = \lambda v$. Dann gilt:

$$\frac{v^* Av}{v^* v} = \frac{v^* \lambda v}{v^* v} = \lambda \in G[A]. \quad \square$$

Satz 1.29: Sei H hermitesch. Dann gilt:

$$\lambda_{\min}(H) = \min_{x \neq 0} \frac{x^* H x}{x^* x}, \quad (1.13)$$

$$\lambda_{\max}(H) = \max_{x \neq 0} \frac{x^* H x}{x^* x}. \quad (1.14)$$

Beweis: Siehe [SB93] (6.4.3), S. 346. \square

Satz 1.30 (Spektralsatz): Sei A hermitesch. Dann existiert eine unitäre Matrix U und eine Diagonalmatrix $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$ aus den Eigenwerten von A mit

$$A = U^* \Lambda U = U^* \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix} U.$$

Beweis: Siehe [SB93] Theorem (6.4.2), S. 346. \square

Satz 1.31: Sei A schief-hermitesch, d.h. $A^* = -A$ und $N := -A^2$. Dann sind die Eigenwerte von A rein imaginär, N ist symmetrisch und die Eigenwerte von N sind reell und nicht-negativ.

Beweis: Sei λ Eigenwert der Matrix A mit Eigenvektor x . Ohne Einschränkung der Allgemeinheit sei $x^*x = 1$. Dann gilt:

$$\bar{\lambda} = x^* \bar{\lambda} x = (x^* \lambda x)^* = (x^* A x)^* = x^* A^* x = x^* (-A) x = -x^* A x = -x^* \lambda x = -\lambda.$$

Damit ist λ rein imaginär und es gilt:

$$N x = -A^2 x = -\lambda A x = -\lambda^2 x, \tag{1.15}$$

das heißt $-\lambda^2 \geq 0$ ist Eigenwert von N . \square

1.2.3 Eigenwerte

Lemma 1.32: Seien $X, Y \in \mathbb{R}^{n \times n}$ reguläre Matrizen, dann besitzen XY und YX die gleichen Eigenwerte.

Beweis: Sei λ Eigenwert von XY mit Eigenvektor x , d.h. $XYx = \lambda x$.

Setze $y := X^{-1}x$, dann ist λ Eigenwert von YX mit Eigenvektor y , denn:

$$YXy = YX X^{-1}x = Yx = X^{-1}XYx = X^{-1}\lambda x = \lambda X^{-1}Xy = \lambda y. \quad \square$$

Folgender Satz liefert Schranken für die Eigenwerte einer beliebigen regulären Matrix:

Satz 1.33 (Bendixsons Theorem): Sei $A \in \mathbb{C}^{n \times n}$ zerlegt in

$$A = H_1 + \mathbf{i}H_2,$$

wobei H_1 und H_2 hermitesch sind. Dann gilt für alle Eigenwerte λ von A :

$$\begin{aligned} \lambda_{\min}(H_1) &\leq \text{Re } \lambda \leq \lambda_{\max}(H_1), \\ \lambda_{\min}(H_2) &\leq \text{Im } \lambda \leq \lambda_{\max}(H_2). \end{aligned}$$

Beweis: Es gilt nach Satz 1.28 und wegen $\text{Re } z = \frac{1}{2}(z + \bar{z})$:

$$\text{Re } \lambda \leq \max_{x \neq 0} \text{Re} \frac{x^* A x}{x^* x} = \max_{x \neq 0} \frac{1}{2x^* x} (x^* A x + x^* A^* x).$$

Nun ist $A + A^* = 2 \cdot H_1$, da H_1 und H_2 hermitesch. Nach Satz 1.29 folgt:

$$\text{Re } \lambda \leq \dots = \max_{x \neq 0} \frac{1}{x^* x} (x^* H_1 x) = \lambda_{\max}(H_1).$$

Die anderen Abschätzungen funktionieren analog, siehe auch [SB93]. \square

1.3 Numerische Lineare Algebra

Die Oseen-Gleichungen werden wir später mittels einer Finite-Elemente-Methode als lineares Gleichungssystem diskretisieren. Um dieses lineare System in einem Rechner zu lösen, benötigen wir einen geeigneten Algorithmus.

Man unterscheidet bei Lösern für lineare Gleichungssysteme zwischen iterativen und direkten Verfahren. Iterative Verfahren bieten gegenüber direkten Verfahren zwei Vorteile. Erstens lassen sie sich in der Regel leichter parallelisieren und zweitens benötigen sie weniger Speicher, der bei großen dreidimensionalen Problemen zum Engpaß bei direkten Verfahren wird.

Wir beschränken uns hier auf die Vorstellung und Auswertung des GMRES-Verfahrens. Der relativ hohe Speicher- und Rechenaufwand vom GMRES-Verfahren im Vergleich zu anderen Krylov-Verfahren (wie zum Beispiel BiCG-Stab) fallen in unserem Fall weniger ins Gewicht, da wir viel Rechenzeit pro Iterationsschritt in einem Vorkonditionierer verwenden und dadurch die Iterationszahl gering bleibt.

Wir werden nicht das klassische GMRES-Verfahren verwenden, sondern eine Variation mit dem Namen FGMRES, mehr dazu in 1.3.3.

1.3.1 Generalized Minimal Residual Method (GMRES)

Es sei folgendes lineare Gleichungssystem

$$Ax = b$$

mit $A \in \mathbb{R}^{n \times n}$ regulär und $b \in \mathbb{R}^n$ gegeben, zu dem die Lösung $x \in \mathbb{R}^n$ gesucht ist.

In der Klasse der Krylov-Unterraum-Verfahren wird eine Näherung $x_m \approx x$ ermittelt. Dazu ist eine Startlösung x_0 gegeben, die auch identisch Null gewählt sein kann, wenn man keine bessere Näherung besitzt, was bei uns meist der Fall ist.

In der sogenannten Generalized Minimal Residual Method (GMRES-Verfahren) wird die Näherung x_m in einem affinen Unterraum $x_0 + \mathcal{K}_m$ des \mathbb{R}^n gesucht:

Definition 1.34: Der **Krylov-Raum** zur Matrix $A \in \mathbb{R}^{n \times n}$ und zum Vektor $v \in \mathbb{R}^n$ mit $m \leq n$ lautet

$$\mathcal{K}_m(A, v) = \text{span}\{v, Av, A^2v, \dots, A^{m-1}v\} \subseteq \mathbb{R}^n.$$

Der GMRES-Algorithmus benötigt dazu vom Raum $\mathcal{K}_m(A, v)$ eine Orthonormalbasis. Diese Basis v_1, \dots, v_m wird mit dem sogenannten Arnoldi-Prozess berechnet. Es handelt sich dabei um eine Implementierung einer Gram-Schmidt-Orthogonalisierung:

ALGORITHMUS 1: Arnoldi-Prozeß

```
in: Matrix A, Startvektor  $v_1$  mit  $\|v_1\| = 1$ 
out: Orthonormalbasis  $(v_1, \dots, v_m)$ 
```

```
1 for j=1, ..., m
2   berechne  $h_{ij} := (Av_j, v_i)$  für  $i = 1, \dots, j$ 
3   berechne  $w_j := Av_j - \sum_{i=1}^j h_{ij}v_i$ 
4   berechne  $h_{j+1,j} := \|w_j\|$ 
5   if  $h_{j+1,j} = 0$  then STOP
6   berechne  $v_{j+1} := w_j/h_{j+1,j}$ 
7 next
```

Wir nehmen im Folgenden an, dass der Algorithmus korrekt terminiert und nicht in Zeile 5 abbricht, was bei exakter Arithmetik nur geschieht, wenn $\dim \mathcal{K}_m(A, v_1) < m$ gilt.

Zusätzlich zur Orthonormalbasis generiert der Arnoldi-Prozeß eine obere Hessenberg-Matrix (eine rechte, obere Dreiecksmatrix mit einer zusätzlichen Nebendiagonale):

$$\overline{H}_m = (h_{ij}) \in \mathbb{R}^{(m+1) \times m}.$$

Es bezeichne $H_m \in \mathbb{R}^{m \times m}$ die Matrix, die aus \overline{H}_m durch Streichung der letzten Zeile entsteht, und sei $V_m := (v_1 \ \dots \ v_m)$ die Matrix mit den Spaltenvektoren v_1, \dots, v_m . Dann gilt folgendes Lemma:

Lemma 1.35: *Mit den oben definierten Matrizen gilt*

$$AV_m = V_{m+1}\overline{H}_m.$$

Beweis: Siehe [Saa96] Proposition 6.5. \square

Im GMRES-Verfahren wird zum gegebenen Startvektor x_0 der Vektor $x_m \in x_0 + \mathcal{K}_m(A, v_1)$ mit $v_1 := r_0/\beta$, $\beta := \|r_0\|$ und dem Startresiduum $r_0 := b - Ax_0$ berechnet.

Bezüglich der Orthonormalbasis v_1, \dots, v_m lässt sich x_m mit $y \in \mathbb{R}^m$ schreiben als

$$x_m = x_0 + V_m y.$$

Der Vektor y wird dabei so gewählt, dass die Norm vom Residuum $J := \|b - Ax_m\|$ minimal wird. Eine leichte Rechnung liefert bereits die Idee des Algorithmus:

$$\begin{aligned} J &= \|b - Ax_m\| \\ &= \|b - A(x_0 + V_m y)\| \\ &= \|r_0 - AV_m y\| \\ &= \|\beta v_1 - V_{m+1} \overline{H}_m y\| \\ &= \|V_{m+1}(\beta e_1 - \overline{H}_m y)\| \\ &= \|\beta e_1 - \overline{H}_m y\|. \end{aligned}$$

Zur Berechnung von y ist also das Ausgleichsproblem $\min_y \|\beta e_1 - \overline{H}_m y\|$ zu lösen. Mittels sogenannter Givens-Rotationen wird dazu die Matrix \overline{H}_m in $m - 1$ Schritten auf obere Dreiecksgestalt gebracht. Wenn man die $m - 1$ Givens-Rotationen zur orthogonalen Matrix Q

zusammenfasst und die neue obere Dreiecksmatrix mit R_m bezeichnet, so gilt $QR_m = \bar{H}_m$ und damit

$$\begin{aligned} J &= \|\beta e_1 - \bar{H}_m y\| \\ &= \|\beta e_1 - QR_m y\| \\ &= \|Q(Q^T \beta e_1 - R_m y)\| \\ &= \|Q^T \beta e_1 - R_m y\|. \end{aligned}$$

Durch Rückwärtseinsetzen lässt sich y dadurch explizit ausrechnen.

Im GMRES-Algorithmus selber wird über die Größe m iteriert. In jedem Schritt wird die Basis des Krylov-Raums um ein Element vergrößert, die Givens-Rotation berechnet und das Residuum ausgewertet. Sobald das Residuum eine vorgegebene Toleranz ρ unterschreitet, terminiert der Algorithmus.

ALGORITHMUS 2: GMRES (Pseudo-Code)

```

in: Matrix  $A$ , Startlösung  $x_0$ , Toleranz  $\rho$ 
out: Näherungslösung  $x$ 


---


1 berechne  $r_0 := b - Ax_0$ ,  $\beta := \|r_0\|$ ,  $v_1 := r_0/\beta$ 
2 for  $m = 1, \dots$ 
3   berechne die neuen Einträge für  $m$  von
    $V_m, \bar{H}_m$  mittels Arnoldi-Prozeß
4   wähle  $y_m$ , so dass  $\|Q^T \beta e_1 - R_m y_m\|$  minimal
5   if  $\|Q^T \beta e_1 - R_m y_m\| < \rho$  goto 7
6 next
7 setze  $x := x_0 + V_m y_m$ 

```

Variante mit Restart

Pro Iterationsschritt steigt der Rechen- und Speicheraufwand linear an, da alle Basisvektoren vom Krylov-Raum gespeichert werden müssen und der neue Vektor mit allen orthogonalisiert werden muss. Damit steigt der gesamte Rechen- und Speicheraufwand quadratisch mit der Zahl der durchgeführten Iterationen.

Aus diesem Grund wurde eine Variante GMRES(k) mit Restart ($k \geq 1$) entwickelt. Nach einer vorgegebenen Anzahl von k Iterationen wird der aufgebaute Krylov-Raum zurückgesetzt und die Zwischenlösung als Startlösung für die nächsten k Iterationen verwendet.

Vereinfacht gesagt wird der GMRES Algorithmus alle k Iterationen neu gestartet und damit ist die Iteration $k + 1$ wieder so günstig durchzuführen wie Iteration 1.

Die Wahl des Parameters k ist nicht trivial, denn je kleiner k ist, desto größer ist die Wahrscheinlichkeit, dass die Konvergenzgeschwindigkeit sinkt bzw. der Algorithmus scheitert. Anschaulich kann es dadurch passieren, dass die richtige Suchrichtung für einen Abstieg des Residuums schlechter bzw. nicht gefunden wird, wenn der Krylov-Raum eine zu geringe Dimension besitzt.

Andererseits darf k nicht zu groß werden, da sonst der Aufwand pro Iteration erheblich ansteigt

und der Algorithmus dadurch stark verlangsamt wird.

Konvergenzabschätzungen

Das volle GMRES-Verfahren konvergiert für alle regulären Matrizen in endlich vielen Schritten.

Satz 1.36: *Sei A regulär. Dann bricht GMRES genau dann im Arnoldiprozess ab, wenn bereits die exakte Lösung x_k berechnet wurde.*

Beweis: Siehe [Saa96] Proposition 6.10. \square

Für das GMRES(k)-Verfahren gilt dieses Resultat nicht mehr, dennoch lässt sich folgender Satz beweisen:

Satz 1.37: *Ist die Systemmatrix A positiv definit, konvergiert GMRES(k) für alle $k \geq 1$.*

Beweis: Siehe [Saa96] Theorem 6.7. \square

Für den nächsten Satz benötigen wir die sogenannte Kondition einer Matrix:

Definition 1.38: *Die Kondition einer regulären Matrix A ist definiert durch*

$$\text{cond}(A) = \|A\|_2 \|A^{-1}\|_2.$$

Die Konvergenzgeschwindigkeit des vollen GMRES-Verfahrens ohne Restart hängt nicht direkt von der Kondition der Matrix ab, wie der nächste Satz zeigt.

Satz 1.39: *Sei A diagonalisierbar, d.h. es existiert die Zerlegung*

$$A = X \Lambda X^{-1}$$

mit der Diagonalmatrix $\Lambda := \text{diag}(\lambda_1, \dots, \lambda_n)$ von Eigenwerten von A . Dann gilt für das Residuum im Schritt m des GMRES-Verfahrens

$$\|r_m\|_2 \leq \text{cond}(X) \epsilon^{(m)} \|r_0\|_2$$

mit

$$\epsilon^{(m)} := \min_{p \in P_m, p(0)=1} \max_i |p(\lambda_i)|,$$

d.h. $\epsilon^{(m)}$ ist der maximale Funktionswert an den Eigenwerten λ_i des Minimalpolynoms vom Grad m mit Ordinatenabschnitt 1.

Beweis: siehe [Saa96] Proposition 6.15. \square

Die Iterationszahl und damit die Konvergenzgeschwindigkeit hängt somit von der Eigenwertverteilung von A ab.

Aus der Definition von $\epsilon^{(m)}$ lassen sich drei Regeln für geringe Iterationszahlen formulieren:

1. Clusterbildung: Häufung der Eigenwerte in möglichst wenigen kleinen Clustern im Gegensatz zu einer weitestgehend äquidistanten Verteilung,
2. Distanz zur Null: Vermeidung von Clustern beim Ursprung beziehungsweise keine gegen Null verlaufenden Folgen,
3. kleines Eigenwertspektrum.

Das Resultat aus Satz 1.39 gilt nur für das volle GMRES-Verfahren und nur für diagonalisierbare Matrizen. Falls A nicht diagonalisierbar ist, gibt es keine Konvergenzresultate. Rechnungen zeigen aber, dass auch in diesem Falle die Clusterbildung für schnelle Konvergenz sorgt. Genauso verhält sich das GMRES(k)-Verfahren. Den Satz selber werden wir in der Arbeit nicht weiter benötigen, er dient hier nur zur Erläuterung des Einflusses der Eigenwertverteilung auf die Iterationszahl.

Zur Veranschaulichung finden sich in Abbildung 1.1 Bilder zu vier verschiedenen hypothetischen Eigenwertverteilungen. In allen Bildern ist das Minimalpolynom mit $m = 4$ für verschiedene Verteilungen von 20 Eigenwerten dargestellt. In den Titeln über den Plots finden sich die zugehörigen Werte für $\epsilon^{(4)}$. Von links oben nach rechts unten:

1. ein Cluster um 0.5 und ein Cluster um 1.5 ($\epsilon^{(4)} \approx 0.03$),
2. äquidistante Verteilung zwischen 0.05 und 1 ($\epsilon^{(4)} \approx 0.48$),
3. ein Cluster um 0.07 und ein Cluster um 1 ($\epsilon^{(4)} \approx 0.64$),
4. ein Cluster um 1 ($\epsilon^{(4)} \approx 0.00$).

1.3.2 Vorkonditionierung

Im vorherigen Abschnitt haben wir gesehen, dass die Iterationszahl stark von der Eigenwertverteilung der Matrix abhängt.

Die Eigenwertverteilung der Systemmatrix des diskretisierten Oseen-Problems ist nicht so, wie im letzten Abschnitt beschrieben. Vorallem die Häufung von Eigenwerten um die Null führt zu sehr hohen Iterationszahlen. Als Ausweg bietet es sich an, das lineare Gleichungssystem durch Vorkonditionierung zu verbessern.

Unter Vorkonditionierung versteht man die Multiplikation des Systems mit einer regulären Matrix P^{-1} . Dabei substituieren wir $x = P^{-1}u$ in $Ax = b$, lösen

$$AP^{-1}u = b \tag{1.16}$$

und berechnen danach x aus $x = P^{-1}u$.

Wenn man das neue System löst, erhält man bei exakter Arithmetik die gleiche Lösung $x \in \mathbb{R}^n$, aber die Iterationszahl wird nun durch die Eigenwerte und die Kondition der Matrix AP^{-1} bestimmt.

Da die Multiplikation von A mit P^{-1} von rechts in (1.16) geschieht, wird dies auch als **Rechtsvorkonditionierung** bezeichnet. Man kann auch von links vorkonditionieren, dies hat aber

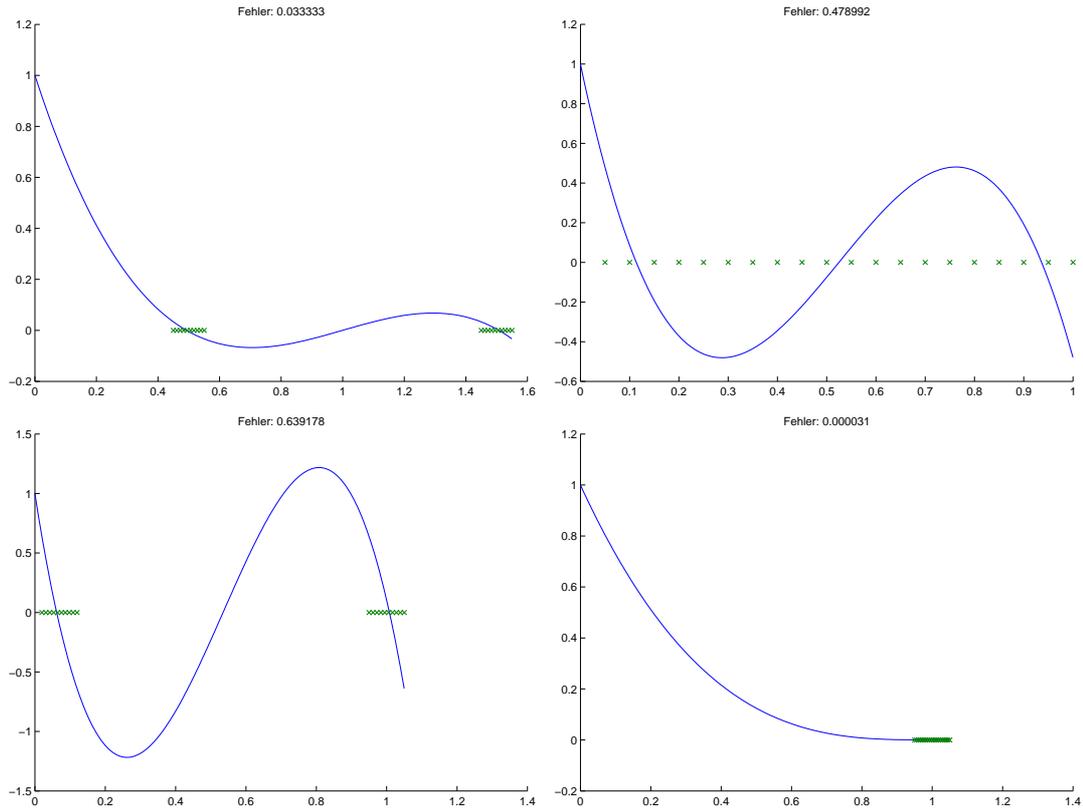


Abbildung 1.1: Minimalpolynom zu $\epsilon^{(4)}$ mit 20 Eigenwerten

einige Nachteile. So würde im GMRES-Verfahren nicht mehr das echte Residuum $\|Ax - b\|$, sondern das vorkonditionierte Residuum $\|P^{-1}(Ax - b)\|$ als Abbruchkriterium verwendet werden. Deshalb beschränken wir uns auf die Betrachtung der Rechtsvorkonditionierung.

Ziel ist es nun, P beziehungsweise P^{-1} so zu konstruieren, dass die Eigenwertverteilung von AP^{-1} besser als die von A ist. Dies gelingt, indem man P^{-1} als Näherung an A^{-1} bildet.

Wird nämlich exakt A^{-1} als Vorkonditionierer verwendet, löst man nur noch das System $Ix = b$, welches Eigenwerte und Kondition 1 hat. Damit ist $\epsilon^{(m)} = 0$ für alle m und das GMRES-Verfahren konvergiert in einem Iterationsschritt. Dies ist natürlich zu aufwändig, da man mit A^{-1} das System bereits direkt gelöst hat.

Die Idee ist nun, für P^{-1} eine günstig zu berechnende Näherung an A^{-1} zu verwenden.

Des Weiteren stellt man nun P^{-1} in der Regel nicht als Matrix auf, sondern nutzt P^{-1} nur als Operator, der ein Matrix-Vektor-Produkt mit P^{-1} ausführt. Dies reicht, da im vorkonditionierten GMRES-Verfahren nur Matrix-Vektor-Produkte mit dem Vorkonditionierer nötig sind.

Der GMRES-Algorithmus mit Vorkonditionierung und Restart sieht dann so aus:

ALGORITHMUS 3: GMRES(m) mit Rechtsvorkonditionierung (Pseudo-Code)

```

in: Matrix  $A$ , rechte Seite  $b$ , Startlösung  $x$ , Toleranz  $\rho$ ,
    Vorkonditionierer  $P^{-1}$ , Restartlänge  $m$ 
out: Näherungslösung  $x$ 

```

```

1  do
2    berechne  $r_0 := b - Ax$ ,  $\beta := \|r_0\|$ ,  $v_1 := r_0/\beta$ 
3     $g := \beta e_1$ 
4    for  $j = 1..m$ 
5       $w := AP^{-1}v_j$ 
6      //Arnoldiprozeß:
7      for  $i = 1..j$ 
8         $h_{i,j} := (w, v_i)$ 
9         $w := w - h_{i,j}v_i$ 
10     next
11      $h_{j+1,j} := \|w\|$ 
12     //  $v_{j+1}$  ist neuer zu  $v_1, \dots, v_j$  orthonormaler Basisvektor:
13      $v_{j+1} := w/h_{j+1,j}$ 
14     aktualisiere Givensrotation, so dass  $QR_j = H_j$  gilt
15     wende Givensrotation auf  $g$  an, so dass  $g = Q^T \beta e_1$  gilt
16     berechne  $y$  in  $\min_y \|R_j y - g\|$  durch Rückwärtseinsetzen
17     und speichere das Residuum  $res := \|R_j y - g\|$ 
18     //Abbruch, falls Zielresiduum bereits erreicht:
19     if ( $res < \rho$ ) goto 20
20      $x := x + P^{-1}V_j y$  //Lösung aktualisieren
21 while ( $res \geq \rho$ ) //restart, falls Lösung nicht erreicht

```

Für ein gegebenes Problem bleibt also die Aufgabe einen guten Vorkonditionierer P^{-1} zu entwickeln, so dass die Eigenwerte von AP^{-1} wenige Cluster bilden, die separiert von der Null sind.

1.3.3 Flexible Generalized Minimal Residual Method (FGMRES)

Konkret bietet sich als Löser eine Variation vom klassischen GMRES-Verfahren an, die in [Saa91] unter dem Namen „flexible GMRES“ (FGMRES) vorgestellt wird und auch in [Saa96] enthalten ist.

Im Gegensatz zum klassischen GMRES erlaubt FGMRES eine Änderung des Vorkonditionierers in jedem Iterationsschritt. Dies bietet sich bei der Lösung des Oseen-Problems besonders an, da im Vorkonditionierer häufig Systeme näherungsweise bis zu einer gewissen Genauigkeit gelöst werden und dadurch die Orthogonalität des Krylov-Unterraums nicht exakt gegeben ist. Gerade für solche Zwecke wurde FGMRES entwickelt.

Als Nachteil lässt sich nennen, dass FGMRES die doppelte Menge an Speicher für temporäre Vektoren im Vergleich zu GMRES benötigt.

Nehmen wir an, der Vorkonditionierer P^{-1} ändert sich in jeder Iteration. Wir kennzeichnen P^{-1} in Iteration j mit dem Index j : P_j^{-1} . Betrachten wir nun Algorithmus 3 und ersetzen

in Zeile 5 $w := AP^{-1}v_j$ durch $w := AP_j^{-1}v_j$, dann funktioniert die Iteration auch weiterhin, allerdings wird in Zeile 20 zur Aktualisierung der Lösung nochmal der Vorkonditionierer ausgewertet:

$$x := x + P^{-1}V_j y.$$

Hier würde ein Fehler entstehen, wenn P^{-1} nicht mit allen P_j^{-1} aus der Schleife übereinstimmt. Im FGMRES speichern wir deswegen die Vektoren $z_j := P_j^{-1}v_j$ zwischen. Damit kann sich der Vorkonditionierer beliebig ändern und die Rechnung in Zeile 20 bleibt exakt, indem wir

$$x = x + [z_1, \dots, z_m]y$$

berechnen.

Da wir jetzt v_j und z_j für $j = 1, \dots, m$ speichern müssen, benötigen wir die doppelte Menge an Speicher. Weitere Änderungen sind im FGMRES nicht notwendig. Der Quellcode sieht dann so aus:

ALGORITHMUS 4: FGMRES (Pseudo-Code)

```
in: Matrix  $A$ , rechte Seite  $b$ , Startlösung  $x$ , Toleranz  $\rho$ ,  
    Vorkonditionierer  $P_j^{-1}$ , Restartlänge  $m$   
out: Näherungslösung  $x$ 

---

  
1  do  
2    berechne  $r_0 := b - Ax$ ,  $\beta := \|r_0\|$ ,  $v_1 := r_0/\beta$   
3     $g := \beta e_1$   
4    for  $j = 1..m$   
5       $z_j := P_j^{-1}v_j$   
6       $w := Az_j$   
7      //Arnoldiprozeß:  
8      for  $i = 1..j$   
9         $h_{i,j} := (w, v_i)$   
10        $w := w - h_{i,j}v_i$   
11     next  
12      $h_{j+1,j} := \|w\|$   
13     //  $v_{j+1}$  ist neuer zu  $v_1, \dots, v_j$  orthonormaler Basisvektor:  
14      $v_{j+1} := w/h_{j+1,j}$   
15     aktualisiere Givensrotation, so dass  $QR_j = H_j$  gilt  
16     wende Givensrotation auf  $g$  an, so dass  $g = Q^T \beta e_1$  gilt  
17     berechne  $y$  in  $\min_y \|R_j y - g\|$  durch Rückwärtseinsetzen  
    und speichere das Residuum  $res := \|R_j y - g\|$   
18     //Abbruch, falls Zielresiduum bereits erreicht:  
19     if ( $res < \rho$ ) goto 21  
20   next  
21    $x := x + [z_1, \dots, z_j]y$  //Lösung aktualisieren  
22 while ( $res \geq \rho$ ) //restart, falls Lösung nicht erreicht

---


```

Kapitel 2

Problemstellung und Diskretisierung

Um ein lineares Gleichungssystem zur Berechnung zu erhalten, müssen wir die Navier-Stokes-Gleichungen linearisieren und das dadurch entstehende Oseen-Problem diskretisieren. Mit den Grundlagen aus Kapitel 1 ist es uns nun möglich, das Oseen-Problem zu behandeln.

Die Finite-Element-Diskretisierung muss gegebenenfalls stabilisiert werden, was in Abschnitt 2.4 kurz angesprochen wird.

Zuletzt bleibt noch das entstandene lineare Gleichungssystem zu analysieren. Dazu führen wir das sogenannte Stokes-Problem ein, mit dessen Hilfe wir in Abschnitt 2.5.5 Abschätzungen über Eigenwerte eines verallgemeinerten Eigenwertproblems erhalten, dass eng mit dem Oseen-Problem zusammenhängt.

2.1 Behandlung der instationären Navier-Stokes-Gleichungen

Zur numerischen Simulation einer inkompressiblen Strömung werden die instationären Navier-Stokes-Gleichungen verwendet. Dazu wird eine Geschwindigkeit $u : \Omega \times T \rightarrow \mathbb{R}^d$ und ein zugehöriger Druck $p : \Omega \times T \rightarrow \mathbb{R}$ in einem beschränkten Gebiet $\Omega \subset \mathbb{R}^d$ ($d \in \{2, 3\}$) mit Lipschitzrand $\partial\Omega$ und Zeit $T = [t_{\min}, t_{\max}]$ gesucht, die folgendes System erfüllen:

$$\begin{aligned} \frac{\partial u}{\partial t} - \nu \Delta u + (u \cdot \nabla)u + \nabla p &= f & \text{in } \Omega \times T, \\ \nabla \cdot u &= 0 & \text{in } \Omega \times T, \\ u &= g & \text{auf } \partial\Omega \times T. \end{aligned} \tag{2.1}$$

Zusätzlich werden Anfangsbedingungen benötigt. Die Anfangsbedingungen haben dabei die Form

$$u(x, t_{\min}) = u_0(x).$$

Vereinfachend betrachten wir in dieser Arbeit nur Randbedingungen 1. Art, d.h. Dirichletrandbedingungen mit $g : \partial\Omega \times T \longrightarrow \mathbb{R}^d$. Außerdem werden wir im folgenden Abschnitt nur mit homogenen Dirichletrandbedingungen rechnen, d.h.

$$g \equiv 0.$$

In der Implementierung heben wir diese Restriktion wieder auf. Dort lassen sich die Randbedingungen in die diskrete Formulierung einarbeiten, ohne dass wir auf Randintegrale oder geeignete Funktionenräume zur Randbeschreibung eingehen müssen.

In (2.1) bezeichnet $\nu > 0$ die **kinematische Viskosität** oder kurz **Viskosität** des Fluids. Die Gleichungen sind entdimensionalisiert und in Bezug auf Gebietsgröße und Geschwindigkeit normalisiert. Mit der charakteristischen Länge L und der charakteristischen Geschwindigkeit U , definieren wir die **Reynoldszahl**

$$Re = \frac{UL}{\nu},$$

die eine charakteristische Größe ist. Mit $Re \approx 1$ sind die Gleichungen diffusions-dominant, für $Re \gg 1$ konvektions-dominant. Im letzteren Fall kann es - abhängig von den Problem Daten - zu instationären Lösungen kommen, auch wenn die Problem Daten zeitunabhängig sind.

Weiter bezeichnen mit $f : \Omega \times T \longrightarrow \mathbb{R}^d$ die **äußeren Kräfte**, die auf das Fluid wirken. Die Gleichungen (2.1) stellen Erhaltungssätze aus der Physik dar: Die erste Gleichung fordert die Impulserhaltung des Fluids und die zweite Gleichung steht für die Massenerhaltung.

Die Dichte des Fluids wird vereinfachend als konstant angenommen und diese Konstante direkt in den Druck p mit hineingezogen.

Der Druck p ist in (2.1) nur bis auf eine Konstante eindeutig bestimmt, weil dort nur die Ableitungen von p auftreten. Daher fordern wir eine Normierung der Form

$$\int_{\Omega} p \, dx = 0.$$

Mit den Navier-Stokes-Gleichungen läßt sich eine Vielzahl von Strömungen simulieren. Als Beispiel seien hier Wasser-, Öl- und Gasströmungen, Innenraum-Luftströmungen und Luftumströmungen um Fahrzeuge mit Geschwindigkeiten weit unter Schallgeschwindigkeit genannt. Strömungen mit höheren Geschwindigkeiten (also zum Beispiel die Umströmung eines fliegenden Flugzeugs) lassen sich so nicht simulieren, da hier die Kompression der Luft am Flugzeug einen nicht zu vernachlässigenden Einfluß auf die Lösung hat.

2.1.1 Zeitdiskretisierung

Zur numerischen Lösung der Navier-Stokes-Gleichungen muss die Zeit T diskretisiert werden. Dazu teilen wir das Zeitintervall T in $M + 1$ diskrete Zeitpunkte $t_{\min} = t_0 < \dots < t_M = t_{\max}$ auf. Die Größe des Zeitschrittes kennzeichnen wir mit $\tau_{m+1} := t_{m+1} - t_m$.

Wir berechnen Näherungen an u und p nur zu diesen diskreten Zeitpunkten und verwenden zur Notation $u^m : \Omega \longrightarrow \mathbb{R}^d$ und $p^m : \Omega \longrightarrow \mathbb{R}$ zur Zeit t_m mit $m \in \{0, \dots, M\}$. Genauso wird auch die Koeffizientenfunktion f zeitlich diskretisiert.

Zusätzlich muss die zeitliche Ableitung $\frac{\partial u}{\partial t}$ von u diskretisiert werden. Wir beschränken uns auf einen einfachen, einseitigen Differenzenquotienten:

$$\frac{\partial u}{\partial t}(t_{m+1}) \approx \frac{u^{m+1} - u^m}{t_{m+1} - t_m} = \frac{u^{m+1}}{\tau_{m+1}} - \frac{u^m}{\tau_{m+1}}.$$

Damit erhalten wir ein stationäres System

$$\begin{aligned} \frac{u^{m+1}}{\tau_{m+1}} - \nu \Delta u^{m+1} + (u^{m+1} \cdot \nabla) u^{m+1} + \nabla p^{m+1} &= f^{m+1} + \frac{u^m}{\tau_{m+1}} && \text{in } \Omega, \\ \nabla \cdot u^{m+1} &= 0 && \text{in } \Omega, \\ u^{m+1} &= 0 && \text{auf } \partial\Omega, \end{aligned} \tag{2.2}$$

welches für jeden Zeitschritt t_m gelöst werden muss. Diese Zeitdiskretisierung ist ein implizites Euler-Verfahren beziehungsweise ein θ -Schema mit $\theta = 1$. Eine Einführung in die Zeitdiskretisierung bietet zum Beispiel [QV97] in Abschnitt 13.4.

2.1.2 Linearisierung durch Fixpunktiteration

Wir beschäftigen uns nun mit der Lösung des Systems (2.2), welches wir von der Notation noch etwas vereinfachen, indem wir konstante Ausdrücke auf der rechten Seite in die Koeffizientenfunktion f ziehen und die Notation des Iterationsschrittes weglassen:

$$\begin{aligned} -\nu \Delta u + cu + (u \cdot \nabla)u + \nabla p &= f && \text{in } \Omega, \\ \nabla \cdot u &= 0 && \text{in } \Omega, \\ u &= 0 && \text{auf } \partial\Omega. \end{aligned} \tag{2.3}$$

Das Gleichungssystem (2.3) ist im Term $(u \cdot \nabla)u$ nichtlinear. Für die Behandlung der Nichtlinearität gibt es verschiedene Ansätze. Das in u und p gekoppelte System wird nicht entkoppelt, sondern komplett gelöst. Einen Überblick über alternative Lösungsansätze für Sattelpunktprobleme bietet [BGL05].

Die Nichtlinearität lösen wir mit einer Fixpunktiteration auf. Wir starten mit einer divergenzfreien Startlösung u_0 mit homogenen Randwerten und iterieren das System

$$\begin{aligned} -\nu \Delta u_k + cu_k + (u_{k-1} \cdot \nabla)u_k + \nabla p_k &= f && \text{in } \Omega, \\ \nabla \cdot u_k &= 0 && \text{in } \Omega, \\ u_k &= 0 && \text{auf } \partial\Omega \end{aligned}$$

über k mit festem u_{k-1} und lösen dabei nach u_k, p_k . Dies sind die sogenannten verallgemeinerten Oseen-Gleichungen.

2.2 Oseen-Problem

Die Behandlung der instationären Navier-Stokes-Gleichungen reduziert sich – wie im vorherigen Kapitel gesehen – auf das häufige Lösen des *verallgemeinerten Oseen-Problems*, das

folgende Gestalt hat:

$$\begin{cases} \text{finde } (u, p) \text{ mit} \\ -\nu \Delta u + cu + (w \cdot \nabla)u + \nabla p = f & \text{in } \Omega, \\ \nabla \cdot u = 0 & \text{in } \Omega, \\ u = 0 & \text{auf } \partial\Omega. \end{cases} \quad (2.4)$$

Dabei ist w ein divergenzfreies Vektorfeld. Da der Druck p nur bis auf Konstanten bestimmt ist, fordern wir zur Drucknormierung wieder

$$\int p \, dx = 0.$$

Definition 2.1: Das Paar $(u, p) \in [C^2(\Omega) \cap C(\bar{\Omega})]^d \times C^1(\Omega)$ heißt **starke Lösung**, wenn es die **starke Formulierung** (2.4) punktweise erfüllt.

2.2.1 Herleitung der Variationsformulierung

Sei $(u, p) \in [C^2(\Omega) \cap C(\bar{\Omega})]^d \times C^1(\Omega)$ starke Lösung von (2.4) und seien $v \in C_0^\infty(\Omega)^d$ und $q \in C^\infty(\Omega)$ **Testfunktionen**, wobei v auf dem Rand verschwindet.

Zuerst multiplizieren wir die beiden Gleichungen mit den Testfunktionen v bzw. q und integrieren über das Gebiet Ω :

$$\begin{aligned} -\nu \int \Delta uv \, dx + \int cuv \, dx + \int (w \cdot \nabla)uv \, dx + \int \nabla pv \, dx &= \int fv \, dx, \\ \int (\nabla \cdot u)q \, dx &= 0. \end{aligned}$$

Nun integrieren wir partiell. Die Randintegrale der partiellen Integration verschwinden, da u und v auf dem Rand verschwinden. Die Integrale über Ω schreiben wir als L^2 -Skalarprodukt und erhalten für beliebige $(v, q) \in C_0^\infty(\Omega)^d \times C^\infty(\Omega)$:

$$\begin{aligned} \nu(\nabla u, \nabla v) + ((c + w \cdot \nabla)u, v) - (p, \nabla \cdot v) &= (f, v), \\ (q, \nabla \cdot u) &= 0. \end{aligned}$$

Zuletzt vergrößern wir den Raum der Testfunktionen durch Vervollständigung bezüglich der H^1 -Norm bzw. der L^2 -Norm und testen mit

$$v, q \in V \times Q := H_0^1(\Omega)^d \times L_*^2(\Omega).$$

Dabei sei

$$L_*^2(\Omega) := \left\{ q \in L^2(\Omega) \mid \int q \, dx = 0 \right\} \quad (2.5)$$

der normierte L^2 -Raum für den Druck.

Diese Formulierung motiviert nun die folgende Definition zur schwachen Lösung:

Definition 2.2: Das Paar $(u, p) \in V \times Q = H_0^1(\Omega)^d \times L_*^2(\Omega)$ heißt **schwache Lösung** der Variationsformulierung des Oseen Problems, wenn

$$\begin{aligned} \nu(\nabla u, \nabla v) + ((c + w \cdot \nabla)u, v) - (p, \nabla \cdot v) &= (f, v), \\ (q, \nabla \cdot u) &= 0 \end{aligned} \tag{2.6}$$

für alle $v, q \in V \times Q$ gilt.

Insbesondere sind starke Lösungen auch Lösungen im schwachen Sinn. Damit haben wir den starken Lösungsbegriff verallgemeinert, da wir nun geringere Regularitätseigenschaften an die Daten fordern müssen und auch geringere Regularität der Lösungen benötigen. Andererseits ist eine hinreichend glatte schwache Lösung wieder eine starke Lösung.

Mit den Bilinearformen

$$\begin{aligned} a : V \times V &\longrightarrow \mathbb{R}, & a(u, v) &:= \nu(\nabla u, \nabla v) + ((c + w \cdot \nabla)u, v), \\ b : V \times Q &\longrightarrow \mathbb{R}, & b(v, p) &:= -(p, \nabla \cdot v) \end{aligned}$$

schreiben wir die Variationsformulierung kurz als

$$\begin{cases} \text{finde } u, p \in V \times Q \text{ mit} \\ a(u, v) + b(v, p) = (f, v) & \forall v \in V, \\ b(u, q) = 0 & \forall q \in Q. \end{cases} \tag{2.7}$$

Satz 2.3: Das Variationsproblem (2.7) hat eine eindeutige Lösung $(u, p) \in V \times Q$.

Beweis: Die Voraussetzungen von Satz 1.22 sind zu überprüfen. Mit

$$Z = \{v \in V \mid b(v, q) = 0 \forall q \in Q\}$$

gilt für beliebige $v \in Z$ mit partieller Integration von $((w \cdot \nabla)v, v)$

$$\begin{aligned} a(v, v) &= \nu|v|_1^2 + c(v, v) + ((w \cdot \nabla)v, v) \\ &= \nu|v|_1^2 + c(v, v) - \frac{1}{2}((\nabla \cdot w)v, v) \\ &= \nu|v|_1^2 + c\|v\|^2, \end{aligned}$$

da $\nabla \cdot w = 0$ nach Voraussetzung gilt. Also ist a nach Definition 1.21 Z -elliptisch.

Nach [GR86], Corollary 2.4 existiert zu jedem $q \in Q$ ein $u \in Z^\perp$ mit

$$\nabla \cdot u = q,$$

des Weiteren gilt $\|u\|_1 \leq C\|q\|$. Daraus folgt:

$$\sup_{v \in V, v \neq 0} \frac{b(v, q)}{|v|_1} \geq \frac{b(u, q)}{|u|_1} = \frac{\|q\|^2}{|u|_1} \geq \frac{1}{C}\|q\|.$$

Damit ist die inf-sup-Bedingung erfüllt und nach Satz 1.22 gilt die Behauptung. \square

2.3 Finite-Elemente-Diskretisierung

Im folgenden Abschnitt wollen wir eine Finite-Element-Methode zur Diskretisierung vorstellen.

Mit endlichdimensionalen Unterräumen

$$V_h \subset V, \quad Q_h \subset Q$$

lautet das **Galerkin-Verfahren** zu (2.7)

$$\begin{cases} \text{finde } u, p \in V_h \times Q_h \text{ mit} \\ a(u, v) + b(v, p) = (f, v) \quad \forall v \in V_h, \\ b(u, q) = 0 \quad \forall q \in Q_h. \end{cases} \quad (2.8)$$

Kümmern wir uns zunächst um die konkrete Definition von V_h und Q_h .

2.3.1 Triangulierung

Zu einem gegebenen Gebiet $\Omega \subset \mathbb{R}^d$ mit $d \in \{2, 3\}$ und Lipschitzrand definieren wir eine Familie von Zerlegungen $\{T_h\}_h$. Jede Zerlegung T_h ist eine Menge von konvexen Teilgebieten K_i von Ω :

$$T_h = \{K_1, \dots, K_m\}$$

mit

$$\bar{\Omega} = \bigcup_i \bar{K}_i.$$

Dabei ist jedes $K \in T_h$ Bild des **Referenzelementes** \hat{K} unter einer bilinearen Transformation

$$F_K : \hat{K} \longrightarrow K \quad (2.9)$$

mit einem Einheitsquader als Referenzelement:

$$\hat{K} := (0, 1)^d. \quad (2.10)$$

Also sind die $K \in T_h$ Vierecke für $d = 2$ und Hexaeder für $d = 3$, siehe dazu Abbildung 2.1.

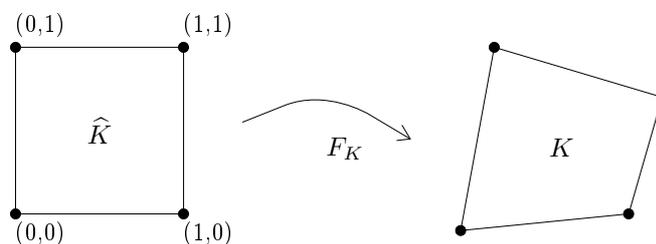


Abbildung 2.1: Transformation vom Referenzelement \hat{K}

Es bezeichne $h_K := \text{diam}(K)$ den Umkreisdurchmesser von K und analog ϱ_K den Inkreisdurchmesser. Eine Zerlegung wird mit dem maximalen Durchmesser $h = \max_{K \in T_h} h_K$ indiziert.

Wir nehmen an, dass das aus den K_i bestehende Rechteckgitter nicht degeneriert ist, insbesondere fordern wir für die Familie von Zerlegungen $\{T_h\}_h$:

1. Zulässigkeit: keine hängenden Knoten, d.h. $\overline{K_i} \cap \overline{K_j}$ ist für $i \neq j$ entweder leer, ein Eckpunkt von K_i und K_j oder eine gemeinsame Kante,
2. Isotropie: $\max_K \frac{h_K}{\varrho_K} \leq c_1$ für alle T_h , d.h. keine Degeneration von Elementen,
3. Quasi-Uniformität: $h \leq c_2 \min_K \varrho_K$.

In Abbildung 2.2 ist ein Ausschnitt einer zulässigen zweidimensionalen Zerlegung zu sehen.

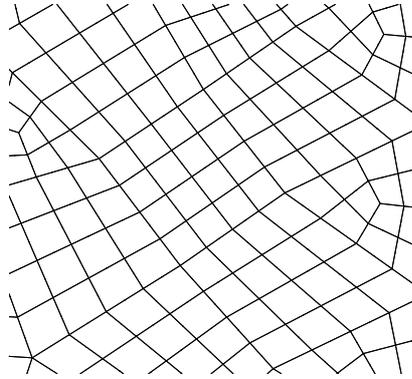


Abbildung 2.2: Ausschnitt einer zulässigen zweidimensionalen Zerlegung

2.3.2 Finite-Elemente-Räume

Zu dem Teilgebiet $K \in T_h$ definieren wir den Begriff des finiten Elements, welches die Art der Ansatzfunktionen auf diesem Teilgebiet charakterisiert und die Repräsentation einer Funktion durch Funktionswerte an den Freiheitsgraden ermöglicht.

Definition 2.4: Als **finites Element** definiert man das Tripel (K, \mathcal{P}, Σ) mit

- dem Teilgebiet $K \in T_h$,
- dem endlichdimensionalen, linearen Raum der Formfunktionen \mathcal{P} auf T_h
- und einer Basis $\Sigma \subset \mathcal{P}^*$ aus linearen Funktionalen auf \mathcal{P} , den Freiheitsgraden, mit denen sich die Elemente aus \mathcal{P} eindeutig festlegen lassen.

Den Finite-Elemente-Raum definieren wir lokal auf den einzelnen Elementen $K \in T_h$, von denen wir uns auf das Referenzelement \hat{K} zurückziehen können. Auf dem Referenzelement definieren wir einen Polynomraum wie folgt:

Definition 2.5: Wir bezeichnen mit \mathbb{Q}_k den Raum der Polynome bis zum Grad $k \in \mathbb{N}$ in jede Koordinatenrichtung auf dem Referenzelement $\hat{K} = (0, 1)^d$:

$$\mathbb{Q}_k(\hat{K}) := \text{span} \left\{ \hat{K} \longrightarrow \mathbb{R}, x \mapsto x_1^{\alpha_1} \cdots x_d^{\alpha_d} \mid 0 \leq \alpha_i \leq k \right\}. \quad (2.11)$$

Damit konstruieren wir den skalaren Finite-Elemente-Raum Q_k auf Ω :

$$Q_k := \left\{ v \in L^2(\Omega) \mid v|_K \circ F_K \in \mathbb{Q}_k(\hat{K}), K \in T_h \right\}. \quad (2.12)$$

Wir wählen eine Lagrange-Basis auf jeder Zelle K_i , wodurch wir für Funktionen eine Darstellung durch Funktionswerte an Freiheitsgraden in den Zellen erhalten. Jedes Basiselement der Lagrange-Basis hat den Funktionswert 1 an genau einem Freiheitsgrad und verschwindet an allen anderen Freiheitsgraden. Für das Q_k -Element werden die $(k+1)^d$ Freiheitsgrade auf einem regelmäßigen Gitter auf dem Referenzelement angeordnet. In Abbildung 2.3 sind einige Beispiele zu sehen.

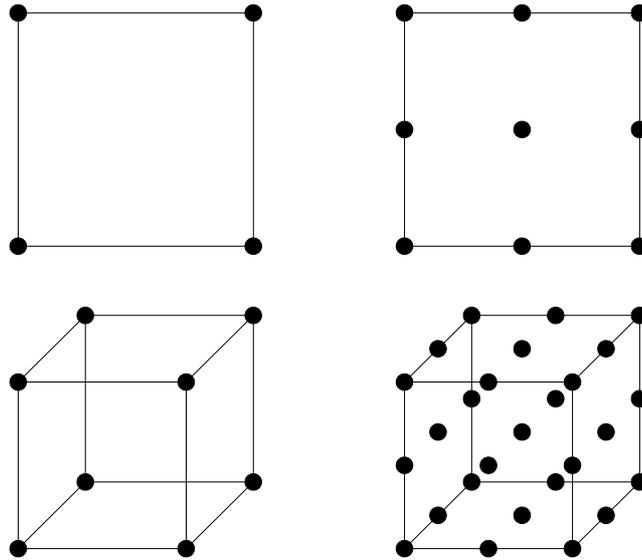


Abbildung 2.3: Anordnung der Freiheitsgrade auf dem Referenzelement für $\mathbb{Q}_1(\hat{K})$ und $\mathbb{Q}_2(\hat{K})$ in zwei und drei Dimensionen

Eine wichtige Abschätzung für Funktionen aus Q_k ist die Inverse Ungleichung:

Lemma 2.6: Sei T_h quasi-uniform, dann gibt es eine Konstante $C_{\text{inv}} > 0$ undabhängig von h , so dass für alle $v \in Q_k$ und $K \in T_h$ gilt:

$$\|v\|_{1,K} \leq C_{\text{inv}} \frac{k^2}{h_K} \|v\|_{0,K}.$$

Beweis: Siehe [GR86], Lemma A.6. \square

Mit der Definition von Q_k können wir schließlich die Räume $V_h \subset V$ und $Q_h \subset Q$ definieren:

Definition 2.7: Sei $k \geq 2$. Dann bezeichnen wir mit $Q_k - Q_{k-1}$ das sogenannte **Taylor-Hood-Paar** $V_h \times Q_h \subset V \times Q$ mit

$$\begin{aligned} V_h &:= [Q_k]^d \cap V \subset V, \\ Q_h &:= Q_{k-1} \cap Q \subset Q. \end{aligned}$$

Der Vollständigkeit halber definieren wir zusätzlich noch die Equal-Order-Elemente, auch wenn wir primär die Taylor-Hood-Elemente verwenden werden:

Definition 2.8: Sei $k \geq 1$. Dann bezeichnen wir mit $Q_k - Q_k$ das sogenannte **Equal-Order-Paar** $V_h \times Q_h \subset V \times Q$ mit

$$\begin{aligned} V_h &:= [Q_k]^d \cap V \subset V, \\ Q_h &:= Q_k \cap Q \subset Q. \end{aligned}$$

Satz 2.9: Das Taylor-Hood-Paar $Q_k - Q_{k-1}$ ist diskret inf-sup-stabil, d.h. es existiert ein $\beta > 0$ unabhängig von h mit:

$$\inf_{q \in Q_h} \sup_{v \in V_h} \frac{(q, \nabla \cdot v)}{|v|_1 \|q\|} \geq \beta > 0.$$

Beweis: Siehe [BF91] Remark 6.2. \square

Die inf-sup-Bedingung gilt aber nicht für Equal-Order-Elemente, weswegen man den Druckraum zusätzlich stabilisieren muss. Dies lässt sich beispielhaft sehr anschaulich durch Abzählargumente an sogenannten Schachbrettlösungen erkennen, siehe [Bra03], S.153ff.

Die Equal-Order-Elemente sind nur der Vollständigkeit halber definiert. Aus drei Gründen werden wir diese nur nebenbei betrachten:

Erstens lässt sich eine zusätzliche Elementordnung für die Geschwindigkeit dadurch motivieren, dass in der kontinuierlichen Formulierung Ableitungen zweiter Ordnung der Geschwindigkeit enthalten sind, aber nur Ableitungen erster Ordnung im Druck.

Zweitens verändert die notwendige Druckstabilisierung das Besetzungsschema der Systemmatrix, wodurch das Design der Vorkonditionierer schwieriger wird beziehungsweise eventuell nicht mehr funktioniert.

Drittens würde eine ausführliche Behandlung der Equal-Order-Elemente den Rahmen dieser Arbeit sprengen. Wo es möglich ist, werden wir aber auf die Unterschiede bei Verwendung von Equal-Order-Elementen hinweisen. Um das Thema der Stabilisierung, die auch für Taylor-Hood-Elemente aus anderen Gründen benötigt wird, kümmern wir uns in Abschnitt 2.4.

An dieser Stelle sei auf [LRL07] und [Löw08] bei LPS-Stabilisierung und auf [Röh07] bei residualer Stabilisierung verwiesen.

2.3.3 Matrixdarstellung des Variationsproblems

Im folgenden Abschnitt werden wir herleiten, wie man das diskretisierte Oseen-Problem als lineares Gleichungssystem assembliert.

Wir beginnen mit dem diskreten Variationsproblem des Galerkin-Verfahrens (2.8):

$$\begin{cases} \text{finde } u, p \in V_h \times Q_h \text{ mit} \\ a(u, v) + b(v, p) = (f, v) & \forall v \in V_h, \\ b(u, q) = 0 & \forall q \in Q_h. \end{cases}$$

Die diskreten Räume wählen wir als Taylor-Hood-Paar nach Definition 2.7:

$$V_h \times Q_h = [Q_k]^d \times Q_{k-1}.$$

Die Dimensionen der diskreten Räume definieren wir als

$$\begin{aligned} n_u &:= \dim V_h, \\ n_p &:= \dim Q_h. \end{aligned} \tag{2.13}$$

Sei nun $\{\phi_1, \dots, \phi_{n_u}\}$ eine Basis von V_h und $\{\psi_1, \dots, \psi_{n_p}\}$ eine Basis von Q_h . Dann existiert zu u und p eine Basisdarstellung

$$u(x) = \sum_{i=1}^{n_u} u_i \phi_i(x), \quad p(x) = \sum_{i=1}^{n_p} p_i \psi_i(x), \quad \forall x \in \Omega.$$

Damit lässt sich $u \in V_h$ mit dem Koeffizientenvektor $U = (u_1, \dots, u_{n_u})^T$ und analog $p \in Q_h$ mit $P = (p_1, \dots, p_{n_p})^T$ identifizieren.

Es reicht, im Variationsproblem nur mit den Basiselementen ϕ_i bzw. ψ_i anstelle von allen $v \in V_h$ bzw. $q \in Q_h$ zu testen, da $a(\cdot, \cdot)$ und $b(\cdot, \cdot)$ Bilinearformen sind und sich die Basisdarstellung von v und q herausziehen lässt:

$$\begin{cases} \text{finde } u, p \in V_h \times Q_h \text{ mit} \\ a(u, \phi_i) + b(\phi_i, p) = (f, \phi_i) & \forall i \in \{1, \dots, n_u\}, \\ b(u, \psi_i) = 0 & \forall i \in \{1, \dots, n_p\}. \end{cases}$$

Setzen wir nun die Basisdarstellung von u und p ein:

$$\begin{aligned} a\left(\sum_{j=1}^{n_u} u_j \phi_j, \phi_i\right) + b\left(\phi_i, \sum_{j=1}^{n_p} p_j \psi_j\right) &= (f, \phi_i) \quad \forall i \in \{1, \dots, n_u\}, \\ b\left(\sum_{j=1}^{n_u} u_j \phi_j, \psi_i\right) &= 0 \quad \forall i \in \{1, \dots, n_p\}. \end{aligned}$$

Durch Ausnutzung der Linearität der Bilinearformen erhalten wir

$$\begin{aligned} \sum_{j=1}^{n_u} u_j a(\phi_j, \phi_i) + \sum_{j=1}^{n_p} p_j b(\phi_i, \psi_j) &= (f, \phi_i) \quad \forall i \in \{1, \dots, n_u\}, \\ \sum_{j=1}^{n_u} u_j b(\phi_j, \psi_i) &= 0 \quad \forall i \in \{1, \dots, n_p\}. \end{aligned}$$

Nun definieren wir die Matrizen $A = (a_{ij}) \in \mathbb{R}^{n_u \times n_u}$, $B = (b_{ij}) \in \mathbb{R}^{n_p \times n_u}$ mit

$$a_{ij} := a(\phi_j, \phi_i) \quad \text{und} \quad b_{ij} := b(\phi_j, \psi_i)$$

und den Vektor $F = (f_i) \in \mathbb{R}^{n_u}$ mit $f_i := (f, \phi_i)$ und schreiben das System als lineares Gleichungssystem:

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} U \\ P \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix}.$$

Ein wichtiger Aspekt für die Lösung ist, dass die Systemmatrix

$$M := \begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix}$$

schwach besetzt ist, also viele Nulleinträge besitzt. Diese Nulleinträge müssen im Rechner nicht gespeichert werden.

2.3.4 Lösbarkeit des diskreten Problems

Wir nehmen nun an, dass wir das System mit einem Taylor-Hood-Paar diskretisieren. Nach [Ste90] Theorem 2.1 ist damit die diskrete inf-sup-Bedingung erfüllt. Die anderen Voraussetzungen von Satz 1.23 sind analog zu Satz 2.3 zu beweisen. Damit ist das System eindeutig lösbar

2.4 Stabilisierung

Die in Abschnitt 2.3 vorgestellte Diskretisierung lässt sich in dieser Form nur unter bestimmten Umständen einsetzen, da es zu unphysikalischen Oszillationen in der Lösung kommen kann. Dafür lassen sich zwei verschiedene Gründe nennen:

Diese Oszillationen können zum einen im Fall dominierender Konvektion, also für $\nu \ll 1$, auftreten. Zum anderen kann aufgrund der Wahl der diskreten Räume für V und Q die diskrete Inf-Sup-Bedingung (1.8) verletzt werden. In Abschnitt 2.3 haben wir zwei verschiedene Finite-Element-Paare vorgestellt. Die sogenannten Taylor-Hood-Paare $Q_k - Q_{k-1}$ erfüllen die diskrete inf-sup-Bedingung, die Equal-Order-Paare $Q_k - Q_k$ hingegen nicht.

In beiden Fällen lässt sich das System stabilisieren, wodurch die Oszillationen eliminiert oder zumindest lokalisiert werden. Dafür wird das ursprüngliche System

$$A(u, p; v, q) = L(v, q) \quad \forall (v, q) \in V \times Q$$

durch ein stabilisiertes System der Form

$$A(u, p; v, q) + A_*(u, p; v, q) = L(v, q) + L_*(v, q) \quad \forall (v, q) \in V \times Q$$

ersetzt. Dabei gilt

$$\begin{aligned} A(u, p; v, q) &= \nu(\nabla u, \nabla v) + ((w \cdot \nabla)u + c \cdot u, v) - (p, \nabla \cdot v) + (q, \nabla \cdot u), \\ L(v, q) &= (f, v). \end{aligned}$$

Wir betrachten hier zwei verbreitete Varianten zur Stabilisierung der Standard-Galerkin-Methode: residuale Stabilisierung und Stabilisierung durch lokale Projektion. In den beiden hier vorgestellten Varianten wird die Divergenz durch die sogenannte Grad-Div-Stabilisierung stabilisiert, die später aus Sicht der Löser noch wichtig wird. Die Grad-Div-Stabilisierung hat einen nicht unerheblichen Anteil an dem stabilisierenden Effekt der zwei Varianten. Alleine deshalb macht es Sinn, die Grad-Div-Stabilisierung getrennt zu verwenden, mehr dazu im folgenden Abschnitt 2.4.1.

Zu stabilisierende Größen sind die Divergenzbedingung $\nabla \cdot u = 0$ in der kontinuierlichen Formulierung und die Geschwindigkeitsgleichung bei dominierender Konvektion sowie der Druck bei Equal-Order-Elementen, welche nicht inf-sup-stabil sind.

Die Stabilisierungen haben nicht nur stabilisierende Wirkung auf die Lösung, sondern verändern die Eigenschaften des zu lösenden linearen Gleichungssystems zum Teil erheblich. Dies wirkt sich natürlich auf die Löser und Vorkonditionierer aus, was wir später analysieren möchten.

Ein wichtiger Aspekt ist die korrekte Wahl der Stabilisierungsparameter. Bei zu großen Parametern kann es zu “Überstabilisierung” kommen, wodurch die Lösung stark verfälscht wird, indem scharfe Grenzschichten in der Lösung zum Teil “verschmiert” werden. Wenn die Parameter hingegen zu klein werden, reicht die Stabilisierung unter Umständen nicht aus und Oszillationen verbleiben.

Neben der optimalen Parameterwahl für ein konkretes Problem interessiert außerdem die Abhängigkeit der Parameter von ν und h .

2.4.1 Grad-Div-Stabilisierung

Die sogenannte **Grad-Div-Stabilisierung** stabilisiert die Divergenzbedingung des Systems, indem das Residuum der Divergenz-Nebenbedingung als Strafterm hinzugefügt wird.

Konkret haben die Operatoren dabei folgende Form:

$$A_{\text{grad-div}}(u, p; v, q) = \sum_{T \in \mathcal{T}_h} \gamma_T (\nabla \cdot u, \nabla \cdot v)_T, \quad (2.14)$$

$$L_{\text{grad-div}}(v, q) = 0 \quad (2.15)$$

mit den Stabilisierungsparametern $\gamma_T > 0$.

Der Name “Grad-Div-Stabilisierung” stammt daher, dass sich der Term $(\nabla \cdot u, \nabla \cdot v)$ nach partieller Integration aus der starken Formulierung $\nabla(\nabla \cdot u)$ herleiten lässt.

Der Stabilisierungsparameter wird für inf-sup-stabile Elemente als

$$\gamma_T \in \mathcal{O}(1)$$

gewählt. Da der Druck nicht stabilisiert wird, ist die Grad-Div-Stabilisierung bei Equal-Order-Elementen nicht einzeln verwendbar. Das Wahl von γ_T lässt sich durch Anwendung des Parameterdesigns für residuale Stabilisierung (siehe Abschnitt 2.4.2) für Taylor-Hood-Elemente motivieren. Wenn man den Stabilisierungsparameter für den SUPG- und PSPG-Term auf Null

setzt, verbleibt nur die Grad-Div-Stabilisierung. Damit erhält man zusätzlich eine Fehlerabschätzung zur Referenzlösung u, p des nicht stabilisierten Systems. Mit hinreichender Regularität $(u, p) \in [H^{k+1}]^d \times H^k$ gilt mit einem geeigneten $\alpha > 0$:

$$\nu \|u - u_h\|_1^2 + c \|u - u_h\|_0^2 + \gamma \|\nabla \cdot (u - u_h)\|_0^2 + \alpha \|p - p_h\|^2 \leq C_1 h^{2k} \|u\|_{k+1}^2 + C_2 h^{2k} \|p\|_k^2. \quad (2.16)$$

Siehe dazu [LR06], Corollary 3.3.

In [OR04] findet man die getrennte Betrachtung der Grad-Div-Stabilisierung für das Stokes-Problem, auf das wir in Abschnitt 2.5.4 eingehen. Table 1 in [MLR07] verdeutlicht im Kontext der residualen Stabilisierung, dass die Grad-Div-Stabilisierung auch alleine zur Stabilisierung des Oseen-Problems verwendet werden kann beziehungsweise in der residualen Stabilisierung eine wichtige Rolle spielt.

2.4.2 Residuale Stabilisierung

Unter dem Namen der *residualen Stabilisierung* (residual based stabilization, kurz **RBS**) fassen wir die Kombination aus Divergenzstabilisierung, Stromlinien-Stabilisierung der Geschwindigkeit (SUPG) und Druckstabilisierung (PSPG) zusammen.

Die Operatoren haben hier folgende Form:

$$\begin{aligned} A_{\text{rbs}}(u, p; v, q) &= A_{\text{grad-div}}(u, p; v, q) && \text{(grad-div)} \\ &+ \sum_{T \in \mathcal{T}_h} \delta_T (-\nu \Delta u + (w \cdot \nabla)u + c \cdot u + \nabla p, (w \cdot \nabla)v)_T && \text{(SUPG)} \\ &+ \sum_{T \in \mathcal{T}_h} \delta_T^p (-\nu \Delta u + (w \cdot \nabla)u + c \cdot u + \nabla p, \nabla q)_T, && \text{(PSPG)} \\ L_{\text{rbs}}(v, q) &= \sum_{T \in \mathcal{T}_h} (f, \delta_T ((w \cdot \nabla)u + \delta_T^p \nabla q))_T. \end{aligned}$$

Die Stabilisierung ist konsistent, das heißt eine kontinuierliche Lösung erfüllt die stabilisierte Variationsformulierung. Wir haben hierbei drei Parameter: γ_T der Grad-Div-Stabilisierung, δ_T von SUPG und δ_T^p von PSPG. Das Parameterdesign muss hier getrennt für Equal-Order- und Taylor-Hood-Elemente durchgeführt werden.

Für Taylor-Hood-Elemente sei auf [MLR07] verwiesen. Die Ergebnisse sind

$$\begin{aligned} \delta_T &\sim \frac{h_T^2}{\nu + b_\infty C_F + c C_F^2 + b_\infty^2 \min(\nu^{-1} C_F^2; c^{-1})}, \\ \delta_T^p &= 0, \\ \gamma_T &= \gamma \sim \nu + b_\infty C_F. \end{aligned}$$

Dabei gilt $b_\infty := \|b\|_{L^\infty}$ und C_F ist die Friedrichskonstante aus der Friedrichs-Ungleichung.

Die Analysis liefert $0 \leq \delta_T^p \leq \delta_T$, die Druckstabilisierung ist aber nur bei Equal-Order-Elementen notwendig (siehe dazu Remark 1 in [MLR07]). Rechnungen bestätigen dieses Ergebnis. Deswegen wählen wir $\delta_T^p = 0$ für Taylor-Hood-Elemente.

Mit diesem Parameterdesign für Taylor-Hood-Elemente $Q_k - Q_{k-1}$ erhalten wir mit der zusätzlichen Regularitätsanforderung $(u, p) \in [H^{k+1}]^d \times H^k$ und einem geeignetem $\alpha > 0$ (siehe [MLR07], Lemma 2) die a-priori Fehlerabschätzung

$$|||(u - u_h, p - p_h)|||^2 \leq \sum_{T \in T_h} h_T^{2k} \left[C_1 \|u\|_{k+1, \omega(T)}^2 + C_2 \|p\|_{k, T}^2 \right]$$

bezüglich der Tripelnorm

$$|||(v, q)|||^2 := \nu |v|_1^2 + c \|v\|_0^2 + \gamma \|\nabla \cdot v\|_0^2 + \sum_{T \in T_h} \delta_T \|(b \cdot \nabla)v\|_{0, T}^2 + \alpha \|q\|_0^2 + \sum_{T \in T_h} \tau_T |q|_{1, T}^2,$$

siehe [MLR07] Theorem 5. Dabei bezeichnet $\omega(T)$ die Umgebung des Elements $T \in T_h$:

$$\omega(T) := \bigcup_{T' \in T_h, \bar{T}' \cap \bar{T} \neq \emptyset} T'.$$

Der Vollständigkeit halber sei noch das Parameterdesign für Equal-Order-Elemente gegeben, welches in [BBJL07] motiviert wird:

$$\begin{aligned} \delta_T &= \delta_T^p \sim \left(\nu h_T^{-2} + \|b\|_{L^\infty(T)} h_T^{-1} + c \right)^{-1}, \\ \gamma_T &\sim h_T^2 \delta_T^{-1}. \end{aligned}$$

2.4.3 Stabilisierung durch lokale Projektion

Die Stabilisierung durch lokale Projektion (kurz LPS) lässt sich durch einen Multiskalenansatz motivieren. Die Strömung wird in drei Skalen geteilt: kleinste, kleine und große Skalen. Die kleinsten Skalen werden im Gegensatz zu den kleinen und großen nicht aufgelöst, sondern nur deren Einfluß modelliert. Dabei wird angenommen, dass die kleinsten Skalen nur die kleinen Skalen beeinflussen. Der Einfluß wird durch zusätzliche, künstliche Diffusion auf den kleinen Skalen simuliert.

In der hier vorgestellten Variante entstehen die kleinen Skalen aus den großen Skalen durch eine regelmäßige Verfeinerung des Gitters und durch Erhöhung des Ansatzgrads der Basisfunktionen.

Das feine Gitter bildet das Rechengebiet, während die großen Skalen durch eine Projektion π in den Grobraum entstehen. Damit diese Projektion lokal ausgewertet werden kann, muss der Grobraum aus un stetigen Elementen bestehen. Der Anteil einer Funktion auf den feinen Skalen wird durch den Fluktuationsoperator $\kappa_h := I - \pi$ ermittelt. Für die vektorwertige Geschwindigkeit sei $\kappa_h^u := I - \pi^u$ passend definiert.

Wir bezeichnen mit M_h das Grobgitter oder Makrogitter. Eine Makrozelle besteht aus der Vereinigung der Gitterzellen aus T_h , die durch die Verfeinerung entstanden sind. Damit erhalten wir eine disjunkte Zerlegung von T_h . Es bezeichne h_M den Durchmesser der Makrozelle $M \in M_h$.

Die Variationsformulierung der LPS-Stabilisierung besteht aus Grad-Div-Stabilisierung,

Stromlinienstabilisierung der Geschwindigkeit und Druckstabilisierung:

$$A_{\text{Ips}}(u, p; v, q) = A_{\text{grad-div}}(u, p; v, q) + \sum_{M \in M_h} [\tau_M(\kappa_h^u((b \cdot \nabla)u), \kappa_h^u((b \cdot \nabla)v))_M + \alpha_M(\kappa_h^u(\nabla p), \kappa_h^u(\nabla q))_M].$$

Im Gegensatz zur residualen Stabilisierung ist diese Stabilisierung symmetrisch, aber nicht konsistent. Die Terme in $A_{\text{Ips}}(u, p; v, q)$ wirken als Strafterm (penalty term).

Wir betrachten hier nur das Parameterdesign für Taylor-Hood-Elemente aus [RL07], Theorem 2:

$$\begin{aligned} \mu &\sim 1, \\ \alpha_M &\sim h_M^2, \\ \tau_M &\sim \min \left\{ \frac{h_M}{\|b\|_{L^\infty}}, \frac{h_M^2}{\nu} \right\}. \end{aligned}$$

Für eine hinreichend glatte Referenzlösung $(u, p) \in [H^{k+1}(\Omega)]^d \times H^k(\Omega)$ und Daten $((w \cdot \nabla)u \in [H^k(\Omega)]^d)$ gilt damit die a-priori-Fehlerabschätzung

$$|||(u - u_h, p - p_h)|||^2 \leq \sum_{M \in M_h} \left[h_M^{2k+1} \|(b \cdot \nabla)u\|_{k, \omega(M)}^2 + h_M^{2k} \|u\|_{k+1, \omega(M)}^2 + h_M^{2k} \|p\|_{l+1, \omega(M)}^2 \right]$$

mit der Tripelnorm

$$|||(v, q)|||^2 := \nu |v|_1^2 + c \|v\|_0^2 + \alpha \|q\|_0^2 + A_{\text{Ips}}(v, q; v, q)$$

für ein geeignetes $\alpha > 0$. Genauso wie bei der residualen Stabilisierung ist die Druckstabilisierung nur für nicht stabile Elementepaare notwendig, so dass wir bei Taylor-Hood-Elementen $\alpha_M = 0$ wählen können. In diesem Zusammenhang sei auf die Diplomarbeit [Löw08] verwiesen, die in Abschnitt 6.4 diesen Sachverhalt behandelt.

2.5 Betrachtung des linearen Systems

In Abschnitt 2.3.3 haben wir die Matrixdarstellung für das nicht stabilisierte Oseen-Problem hergeleitet. Allgemeiner hat die Systemmatrix im stabilisierten Fall die Form

$$M = \begin{pmatrix} A & B_S^T \\ B & C \end{pmatrix}.$$

Bei residualer Stabilisierung ist im Gegensatz zu LPS $B \neq B_S$ durch den nichtsymmetrischen Anteil im SUPG-Teil. Bei Equal-Order-Elementen muss der Druck entweder durch LPS oder PSPG stabilisiert werden, wodurch $C \neq 0$ ist. Wir unterscheiden jetzt vier verschiedene Fälle für die Gestalt von M :

2.5. Betrachtung des linearen Systems

1. $B_S \neq B, C \neq 0$: Equal-Order-Elemente mit residualer Stabilisierung.
2. $B_S = B, C \neq 0$: Equal-Order-Elemente mit LPS-Stabilisierung.
3. $B_S \neq B, C = 0$: Taylor-Hood-Elemente mit residualer Stabilisierung.
4. $B_S = B, C = 0$: Taylor-Hood-Elemente ohne Stabilisierung, mit Grad-div-Stabilisierung oder mit LPS-Stabilisierung.

In den Abbildungen 2.4, 2.5 und 2.6 sind verschiedene Besetzungsmuster der Matrix M dargestellt. Dabei bezeichnet ein schwarzer Punkt einen Matrixeintrag ungleich Null. Zu erwähnen ist, dass in die Matrizen bereits Dirichlet-Randbedingungen eingearbeitet sind. Dadurch enthalten die Matrizen an einem Randknoten der Geschwindigkeit in der zugehörigen Zeile und Spalte nur den Diagonaleintrag. In Tabelle 2.1 sind jeweils die Anzahl der Nicht-Null-Einträge aufgelistet.

Abb.	Typ	Stab.	Einträge ungleich Null					pro Zeile
			M	A	B^T	B	C	
2.4-L	2D- $Q_2 - Q_1$	-	2212	1122	545	545	0	11.83
2.4-R	3D- $Q_2 - Q_1$	-	3275	1323	976	976	0	8.15
2.5-L	2D- $Q_2 - Q_1$	Grad-Div	2962	1872	545	545	0	15.84
2.5-M	2D- $Q_2 - Q_1$	LPS	3826	2736	545	545	0	20.46
2.5-R	2D- $Q_2 - Q_1$	RBS	2995	1872	578	545	0	16.02
2.6-L	2D- $Q_2 - Q_1$	-	5746	3074	1336	1336	0	17.36
2.6-R	2D- $Q_2 - Q_2$	LPS	7669	2736	1266	1266	2401	31.56

Tabelle 2.1: Anzahl der Nicht-Null-Einträge der Systemmatrix bei unterschiedlicher Diskretisierung

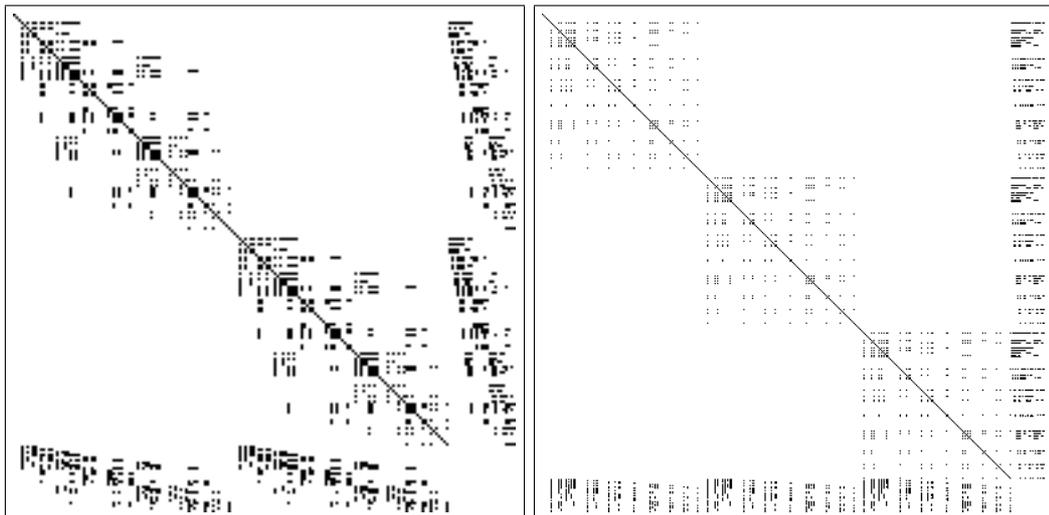


Abbildung 2.4: Besetzungsmuster von Systemmatrizen, 2D und 3D ohne Stabilisierung

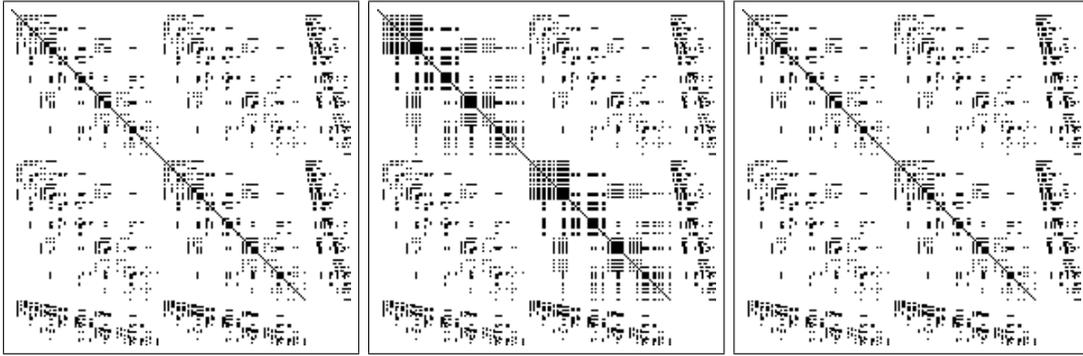


Abbildung 2.5: Besetzungsmuster von Systemmatrizen, 2D mit Grad-Div-, LPS- und RBS-Stabilisierung

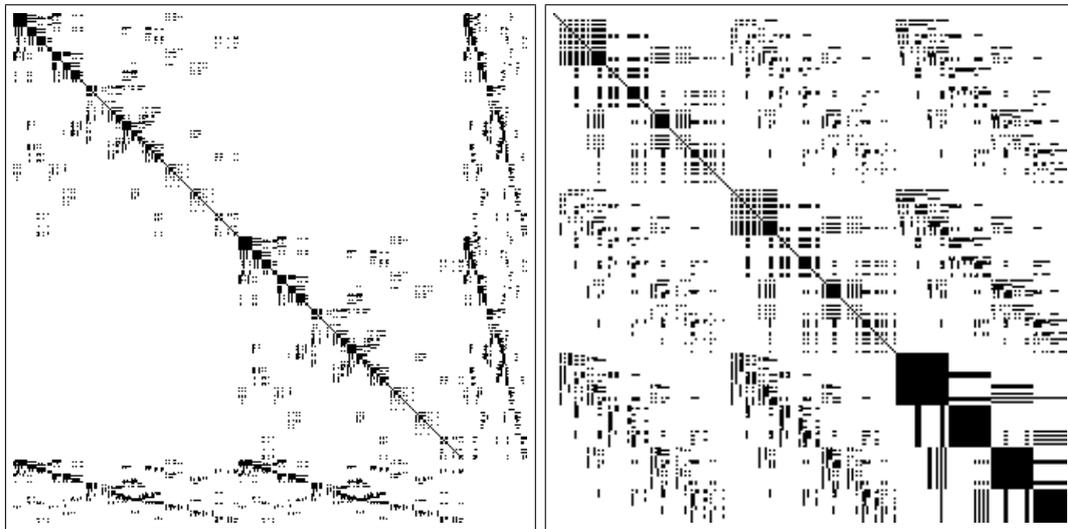


Abbildung 2.6: Besetzungsmuster von Systemmatrizen, unregelmäßiges Gitter und Equal-Order-Elemente

In Abbildung 2.4 sieht man ein unstabilisiertes System mit $Q_2 - Q_1$ -Elementen eines zweidimensionalen Problems links und eines dreidimensionalen Problems rechts. In den Besetzungsmustern erkennt man deutlich die Struktur der Matrix. Der A -Block hat jeweils Blockdiagonalgestalt mit je einem Block pro Geschwindigkeitskomponente, also zwei im zweidimensionalen Problem und drei im dreidimensionalen Problem. Die Geschwindigkeitskomponenten sind untereinander nicht gekoppelt, da das System nicht stabilisiert ist. Der C -Block rechts unten ist leer, da auch der Druck nicht stabilisiert wird. Eine Komponente des A -Blocks ist deutlich größer als die Zahl der Zeilen im B^T -Block, da für die Geschwindigkeit Q_2 -Elemente anstelle der Q_1 -Elemente für den Druck verwendet werden. Im dreidimensionalen Problem (rechts) verschiebt sich das Verhältnis der Freiheitsgrade für Geschwindigkeit und Druck noch weiter, so dass ein Großteil der Freiheitsgrade für die Geschwindigkeit verwendet wird.

In Abbildung 2.5 sieht man wieder das zweidimensionale Problem aus Abbildung 2.4, aber diesmal unterschiedlich stabilisiert: links nur mit Grad-Div-, in der Mitte LPS- und rechts mit

RBS-Stabilisierung. Man erkennt, dass die Grad-Div-Stabilisierung (die ja in allen drei Besetzungsmustern vorhanden ist) wie erwartet die unterschiedlichen Geschwindigkeitskomponenten miteinander koppelt. Im mittleren Bild sieht man deutlich, dass in den Diagonalblöcken der Geschwindigkeit zusätzliche Kopplungen entstehen. Dies liegt daran, dass durch die verwendete Makroelementtechnik auch Freiheitsgrade von nicht benachbarten Zellen gekoppelt werden. Man kann nur schlecht erkennen, dass im rechten Bild $B_S \neq B$ gilt, weil es nur sehr wenige zusätzliche Einträge in B_S^T gibt.

Betrachten wir nun Abbildung 2.6. Links sehen wir ein nicht stabilisiertes zweidimensionales Bild, aber im Gegensatz zu dem Problem in Abbildung 2.4 auf einem unregelmäßigen Gitter. Man sieht daher in den Geschwindigkeitsblöcken eine andere Verteilung der Kopplungen. Im rechten Bild sehen wir ein LPS-stabilisiertes, zweidimensionales Problem mit $Q_2 - Q_2$ -Elementen. Hier entspricht die Größe eines B -Blocks dem eines Geschwindigkeitsblocks aus dem A -Block, da beide nun gleich viele Freiheitsgrade besitzen. Durch die Grad-Div-Stabilisierung sind die Geschwindigkeitskomponenten wieder miteinander gekoppelt, genauso wie in Abbildung 2.5. Neu ist, dass wir auch den Druck stabilisieren müssen. Dies erkennt man im vorhandenen C -Block rechts unten.

2.5.1 Drucknormierung des diskreten Systems

Im kontinuierlichen Problem forderten wir die Bedingung

$$\int p \, dx = 0, \tag{2.17}$$

damit der Druck eindeutig bestimmt ist. Auch im diskreten Fall ist der Druck nicht eindeutig bestimmt. Mit der Matrixdarstellung in Abschnitt 2.3.3 ist dadurch die Systemmatrix singulär, selbst wenn Randbedingungen für die Geschwindigkeit u in die Matrix eingearbeitet werden.

Die Einarbeitung von (2.17) im diskreten Problem ist möglich, würde aber alle Druckfreiheitsgrade miteinander koppeln. Man kann die Normierung (2.17) auch anders erhalten. Dazu ersetzen wir in einem post-processing-Schritt p durch $p - \int p \, dx$ und erhalten dadurch eine normierte Lösung, für die (2.17) auch im Diskreten gilt.

Dadurch reduziert sich das Problem auf die Behandlung des Null-Eigenwerts der Systemmatrix, um das System lösen zu können. Dafür existieren mehrere Alternativen:

Man fixiert einen beliebigen Druckfreiheitsgrad im System auf den Wert 0. Dadurch erhält man einen zusätzlichen Diagonaleintrag im C -Block. Für Taylor-Hood-Elemente gilt dadurch nun aber $C \neq 0$, auch wenn es sich nur um eine kleine “Störung” handelt.

Alternativ könnte man einen penalty-Term einführen, der die Abweichung vom Integralmittel Null “bestraft”. Dieser hätte die Form einer skalierten Massematrix (siehe Abschnitt 2.5.2) im C -Block der Form $\varepsilon(\psi_i, \psi_j)$. Dabei ist $0 < \varepsilon \ll 1$ zu wählen.

Die Singularität kann außerdem durch die Wahl der Randbedingungen für u beeinflusst werden. Denn durch fehlende Dirichlet-Randbedingungen auf einem Teil des Randes, zum Beispiel am Ausströmrand, erhält man natürliche Neumann-Randbedingungen an dieser Stelle. Bei der

partiellen Integration für die schwache Formulierung erhält man die Randterme

$$\int_{\partial\Omega} v \frac{\partial u_i}{\partial n} ds - \int_{\partial\Omega} vp \cdot n ds,$$

die für $v|_{\partial\Omega} = 0$ verschwinden. Falls es aber Freiheitsgrade für u auf dem Rand gibt, so enthält das assemblierte System dadurch implizit die Bedingung

$$\frac{\partial u_i}{\partial n} - p \cdot n = 0,$$

womit der Druck eindeutig bestimmt ist.

Selbst wenn man auf dem ganzen Rand Dirichlet-Bedingungen für die Geschwindigkeit vorschreibt und keine der oben genannten Alternativen verwendet, um eine reguläre Systemmatrix zu erhalten, kann das System unter Umständen gelöst werden. Denn der GMRES-Algorithmus funktioniert auch mit einem Null-Eigenwert der Vielfachheit 1, wenn sich das Startresiduum r_0 im Bild von M befindet:

$$r_0 = Ax_0 - b \in \mathcal{R}(M). \quad (2.18)$$

Siehe dazu [Smo07], Proposition 3.3. Wobei die Bedingung (2.18) bei Verwendung eines variierenden Vorkonditionierers nicht einfach zu überprüfen ist.

In den durchgeführten Rechnungen haben wir generell mit dem singulären Problem gerechnet, ohne auf Probleme zu stoßen. Dennoch nehmen wir zur theoretischen Betrachtung an, dass B maximalen Rang hat, also

$$\dim(\text{Ker}B) = n_u - n_p \quad (2.19)$$

gilt.

2.5.2 Eigenschaften der Massematrix

Definition 2.10: *Die Matrix*

$$(M_p)_{ij} = (\psi_i, \psi_j) = \int \psi_i \psi_j dx$$

heißt **Massematrix des Druckraums**.

Satz 2.11: *Die Massematrix M_p des Druckraums ist invertierbar, symmetrisch und damit auch diagonalisierbar.*

Beweis: Die Massematrix M_p ist gegeben durch

$$(M_p)_{ij} = (\psi_i, \psi_j).$$

Das Skalarprodukt ist symmetrisch und positiv definit und damit auch M_p . Symmetrisch positiv definite Matrizen sind invertierbar und nach dem Spektralsatz (Satz 1.30) ist M_p damit auch diagonalisierbar. \square

Satz 2.12: *Es existieren $c, C > 0$ unabhängig von h mit*

$$ch^d \leq \frac{x^T M_p x}{x^T x} \leq Ch^d \quad \forall q.$$

Beweis: Siehe [ESW05] Remark 1.31. \square

2.5.3 Definition des Schurkomplements

Die LU-Zerlegung der Matrix $M = LU$ mit linker unterer Dreiecksmatrix L und rechter oberer Dreiecksmatrix U lautet:

$$M = LU = \begin{pmatrix} A & B_S^T \\ B & C \end{pmatrix} = \begin{pmatrix} I & 0 \\ BA^{-1} & I \end{pmatrix} \begin{pmatrix} A & B_S^T \\ 0 & -BA^{-1}B_S^T + C \end{pmatrix}.$$

Der rechte, untere Block der Matrix U wird bei der Vorkonditionierung des Systems eine wichtige Rolle spielen und wird in der Literatur als **Schurkomplement** der Matrix M bezeichnet:

$$S := -BA^{-1}B_S^T + C.$$

2.5.4 Spezialfall: Stokes-Problem

In diesem Abschnitt betrachten wir einen Spezialfall des Oseen-Problems. Wenn das zu simulierende Fluid sehr viskos ist - man denke zum Beispiel an Honig - lassen sich die beschreibenden Gleichungen vereinfachen. Da die viskosen Kräfte für $Re \ll 1$ dominieren, kann man die anderen Terme weglassen und man erhält:

$$\begin{cases} \text{finde } (u, p) \text{ mit} \\ -\nu \Delta u + \nabla p = f & \text{in } \Omega, \\ \nabla \cdot u = 0 & \text{in } \Omega, \\ u = 0 & \text{auf } \partial\Omega. \end{cases}$$

Durch Normierung der Gleichungen können wir die Viskosität ν aus der Gleichung eliminieren, indem die rechte Seite und der Druck p umskaliert werden. Damit erhält man die **Stokes-Gleichungen**:

$$\begin{cases} \text{finde } (u, p) \text{ mit} \\ -\Delta u + \nabla p = f & \text{in } \Omega, \\ \nabla \cdot u = 0 & \text{in } \Omega, \\ u = 0 & \text{auf } \partial\Omega. \end{cases} \quad (2.20)$$

Die Diskretisierung läuft nun analog zu Abschnitt 2.3 und man erhält das System

$$M_\infty \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}. \quad (2.21)$$

Im Gegensatz zur Systemmatrix des Oseen-Problems ist hier M_∞ symmetrisch, genauso wie das Schurkomplement der Systemmatrix, das wir genauso wie beim Oseen-Problem definieren.

Definition 2.13: Es bezeichne S_∞ das Schurkomplement von (2.21), das definiert ist als

$$S_\infty = -BA^{-1}B^T.$$

Folgender Satz liefert, dass sich das Schurkomplement durch die Massesmatrix unabhängig von h approximieren lässt.

Satz 2.14: *Es sei $S_\infty = -BA^{-1}B^T$ das Schurkomplement des Stokes-Problems bei einer inf-sup-stabilen Diskretisierung. Dann existieren, $\gamma, \Gamma \in \mathbb{R}$ unabhängig von h , so dass:*

$$\gamma^2 \leq \frac{-x^T S_\infty x}{x^T M_p x} \leq \Gamma^2 \quad \forall x \in \mathbb{R}^n.$$

Beweis: Nach Voraussetzung gilt die inf-sup-Bedingung, d.h.

$$\beta \|p\|_Q \leq \sup_{u \in V_h \setminus \{0\}} \frac{(p, \nabla \cdot u)}{\|u\|_V} \quad \forall p \in Q_h.$$

Seien P und U die zu p und u gehörigen Koeffizientenvektoren wie in Abschnitt 2.3.3. Für die Normen gilt dann im diskreten mit einer von h und ν unabhängigen Konstante $c > 0$:

$$\|p\|_Q^2 = P^T M_p P, \quad \|u\|_V^2 \geq c |u|_V^2 = c (U^T A U),$$

da $\|\cdot\|_V$ und $|\cdot|_V$ nach Satz 1.19 äquivalent sind. Die Matrix A entspricht der Diskretisierung von $(\nabla u, \nabla v)$ und ist symmetrisch und dadurch diagonalisierbar. Damit gilt:

$$\begin{aligned} \beta (P^T M_p P)^{1/2} &= \beta \|p\|_Q \\ &\leq \sup_{u \in V_h \setminus \{0\}} \frac{(p, \nabla \cdot u)}{\|u\|_V} \\ &\leq \max_{U \neq 0} \frac{P^T B U}{(U^T A U)^{1/2}} \\ &= \max_{V = A^{1/2} U} \frac{P^T B A^{-1/2} V}{(V^T V)^{1/2}} \end{aligned}$$

(was für $V = A^{-1/2} B^T P$ maximal wird)

$$\begin{aligned} \dots &= \frac{P^T B A^{-1} B^T P}{(P^T B A^{-1} B^T P)^{1/2}} \\ &= (P^T B A^{-1} B^T P)^{1/2} \\ &= -(P^T S_\infty P)^{1/2}. \end{aligned}$$

Andererseits gilt mit der Cauchy-Schwarz-Ungleichung und Satz 1.20

$$(p, \nabla \cdot u) \leq \|p\| \|\nabla \cdot u\| \leq \|p\| \|\nabla u\|$$

und in Matrixform mit $u = A^{-1} B^T p$:

$$-(P^T S_\infty P) \leq (P^T M_p P)^{1/2} (U^T A U)^{1/2} = -(P^T M_p P)^{1/2} (P^T S_\infty P)^{1/2}.$$

Daraus folgt dann

$$-(P^T S_\infty P)^{1/2} \leq (P^T M_p P)^{1/2}. \quad \square$$

2.5.5 Eigenwertbetrachtungen

Wir haben gesehen, dass sich im Stokes-Fall das Schurkomplement durch die Massematrix unabhängig von h approximieren lässt. Dies gilt auch beim Oseen-Problem, aber S ist abhängig von der Viskosität ν , im Gegensatz zur Massematrix.

Die genaue ν -Abhängigkeit des Schurkomplements ist später von Bedeutung. Indem wir uns auf den Fall vom Stokes-Problem zurückziehen, gelingt es uns unter gewissen Anforderungen die ν -Abhängigkeit zu zeigen. Unter anderem fordern wir, dass sich der A -Block in einen symmetrischen und einen schiefsymmetrischen Anteil aufteilen lässt:

Lemma 2.15: *Sei M eine Systemmatrix zum Oseen-Problem ohne Stabilisierung, mit homogenen Dirichlettrandbedingungen und mit $c = 0$. Dann lässt sich die Matrix A im oberen linken Block in die Matrizen L und N zerlegen:*

$$A = \nu L + N.$$

Dabei ist $L = L^T$ symmetrisch und $N = -N^T$ schiefsymmetrisch.

Außerdem lassen sich die Eigenwerte der Matrix

$$W := -L^{-1/2}NL^{-1}NL^{-1/2}$$

mit $C > 0$ unabhängig von h beschränken:

$$0 \leq \lambda(W) \leq C.$$

Beweis: Die Matrix A entspricht der Diskretisierung von

$$\nu(\nabla u, \nabla v) + ((b \cdot \nabla)u, v).$$

Der Term $(\nabla u, \nabla v)$ ist offensichtlich symmetrisch und wird damit in L diskretisiert. Der zweite Summand ist aufgrund der partiellen Integration schiefsymmetrisch und bildet daher die Matrix N .

Für den zweiten Teil der Behauptung definieren wir die Matrix

$$\tilde{N} := L^{-1/2}NL^{-1/2},$$

die wieder schiefsymmetrisch ist. Nach Satz 1.31 sind die Eigenwerte von $-\tilde{N}^2 = W$ reell und nicht-negativ.

Nach Definition gilt $L_{ij} = (\nabla \phi_j, \nabla \phi_i)$. Sei $u = \sum_{k=1}^{n_u} U_k \phi_k$, dann gilt:

$$\|\nabla u\|^2 = (\nabla u, \nabla u) = (U, LU) = (L^{1/2}U, L^{1/2}U) = \|L^{1/2}U\|^2.$$

Damit folgt mit Verwendung der Poincaré-Friedrichs-Ungleichung (Satz 1.18):

$$\begin{aligned} (NU, V) &= ((w \cdot \nabla)u, v) \\ &\leq \|(w \cdot \nabla)u\| \|v\| \\ &\leq C|u| \|v\| \\ &\leq C'|u| |v| \\ &= C' \|\nabla u\| \|\nabla v\| \\ &= C' \|L^{1/2}U\| \|L^{1/2}V\|. \end{aligned}$$

Also folgt:

$$\begin{aligned} \sup_{u,v} \frac{(\tilde{N}u, v)}{\|u\|\|v\|} &= \sup_{u,v} \frac{(L^{-1/2}NL^{-1/2}u, v)}{\|u\|\|v\|} \\ &= \sup_{u,v} \frac{(Nu, v)}{\|L^{1/2}u\|\|L^{1/2}v\|} \\ &\leq C'. \end{aligned}$$

Damit sind die Eigenwerte von \tilde{N} mit $C' > 0$ unabhängig von h beschränkt, siehe auch [ES86], Theorem 4 oder [OV07], Lemma 3.3. Schließlich folgt:

$$0 \leq \lambda(W) \leq (C')^2. \quad \square$$

Mit diesem Lemma können wir folgenden Satz beweisen. Die Beweisidee stammt aus [ES95], Theorem 1.

Satz 2.16: *Betrachte das verallgemeinerte Eigenwertproblem*

$$-Sp = \mu M_p p \quad (2.22)$$

mit dem Schurkomplement $S := -BA^{-1}B^T$ vom Oseen-Problem und M_p der Massematrix im Druckraum. Das Oseen-Problem ist dabei nicht stabilisiert und es gelte $c = 0$.

Dann existieren positive Konstanten C_1, C_2, C_3 unabhängig von ν und h mit

$$\begin{aligned} C_1\nu &\leq \operatorname{Re} \mu \leq C_2\nu^{-1}, \\ |\operatorname{Im} \mu| &\leq C_3\nu^{-1}. \end{aligned}$$

Beweis: Anders formuliert suchen wir Schranken für die Eigenwerte μ im Eigenwertproblem

$$-(M_p^{-1/2}SM_p^{-1/2})p = \mu p, \quad (2.23)$$

da M_p nach Satz 2.11 invertierbar und diagonalisierbar ist. Mit

$$\begin{aligned} C &:= M_p^{-1/2}B\frac{1}{2}(A^{-1} + A^{-T})B^T M_p^{-1/2}, \\ R &:= -iM_p^{-1/2}B\frac{1}{2}(A^{-1} - A^{-T})B^T M_p^{-1/2} \end{aligned}$$

zerlegen wir

$$-M_p^{-1/2}SM_p^{-1/2} = C + iR$$

mit den hermiteschen Matrizen C und R . Damit sind die Voraussetzungen für Satz 1.33 (Bendixsons Theorem) erfüllt und wir erhalten mit 1.28:

$$\min_{x \neq 0} \frac{x^*Cx}{x^*x} \leq \operatorname{Re} \mu \leq \max_{x \neq 0} \frac{x^*Cx}{x^*x} \quad \text{und} \quad |\operatorname{Im} \mu| \leq \max_{x \neq 0} \frac{x^*Rx}{x^*x}. \quad (2.24)$$

Für den Rayleigh-Quotienten im Realteil gilt mit $y = M_p^{-1/2}x$ (und da M_p symmetrisch ist):

$$\begin{aligned} \frac{x^*Cx}{x^*x} &= \frac{x^*M_p^{-1/2}B\frac{1}{2}(A^{-1} + A^{-T})B^T M_p^{-1/2}x}{x^*x} \\ &= \frac{y^*B\frac{1}{2}(A^{-1} + A^{-T})B^Ty}{y^*M_p y} \\ &= \frac{y^*B\frac{1}{2}(A^{-1} + A^{-T})B^Ty}{y^*S_\infty y} \cdot \frac{y^*S_\infty y}{y^*M_p y}. \end{aligned} \quad (2.25)$$

2.5. Betrachtung des linearen Systems

Dabei bezeichnet S_∞ das Schurkomplement des Stokes-Problems, siehe Definition 2.13. Nach Satz 2.14 ist der rechte Bruch unabhängig von h und ν , dass heißt es existieren $\gamma, \Gamma > 0$ mit:

$$\gamma^2 \leq \frac{y^* S_\infty y}{y^* M_p y} \leq \Gamma^2. \quad (2.26)$$

Zur weiteren Abschätzung nutzen wir die Konstruktion der Matrix A aus. Nach Lemma 2.15 lässt sich A schreiben als

$$A = \nu L + N$$

mit der symmetrischen Matrix L und der schiefsymmetrischen Matrix N . Die Matrix L entspricht dabei dem (1,1)-Block der Systemmatrix vom Stokes-Problem, siehe Abschnitt 2.5.4.

Nun gilt:

$$\begin{aligned} \frac{1}{2} (A^{-1} + A^{-T}) &= A^{-1} \frac{1}{2} (A + A^T) A^{-T} \\ &= (\nu L + N)^{-1} (\nu L) (\nu L - N)^{-1} \end{aligned} \quad (2.27)$$

und da $A + A^T = \nu L + N + \nu L + N^T = 2\nu L$ deswegen

$$\begin{aligned} \frac{1}{2} (A^{-1} + A^{-T}) &= (\nu L + N)^{-1} (\nu^{-1} L^{-1})^{-1} (\nu L - N)^{-1} \\ &= [\nu^{-1} L^{-1} (\nu L + N)]^{-1} (\nu L - N)^{-1} \\ &= [I + \nu^{-1} L^{-1} N]^{-1} (\nu L - N)^{-1} \\ &= [(\nu L - N)(I + \nu^{-1} L^{-1} N)]^{-1} \\ &= [\nu L - N + N - \nu^{-1} N L^{-1} N]^{-1} \\ &= [\nu L - \nu^{-1} N L^{-1} N]^{-1}. \end{aligned}$$

Jetzt ist L diagonalisierbar und damit gilt

$$\begin{aligned} \frac{1}{2} (A^{-1} + A^{-T}) &= L^{-1/2} L^{1/2} [\nu L - \nu^{-1} N L^{-1} N]^{-1} L^{1/2} L^{-1/2} \\ &= L^{-1/2} [\nu I - \nu^{-1} L^{-1/2} N L^{-1} N L^{-1/2}]^{-1} L^{-1/2}. \end{aligned}$$

Wir definieren nun $\tilde{N} := L^{-1/2} N L^{-1/2}$ und erhalten dadurch für den ersten Bruch in (2.25) mit der Substitution $z = L^{-1/2} B^T y$:

$$\begin{aligned} \frac{y^* B \frac{1}{2} (A^{-1} + A^{-T}) B^T y}{y^* S_\infty y} &= \frac{y^* B L^{-1/2} (\nu I - \nu^{-1} \tilde{N}^2)^{-1} L^{-1/2} B^T y}{y^* S_\infty y} \\ &= \frac{z^* (\nu I - \nu^{-1} \tilde{N}^2)^{-1} z}{z^* z}. \end{aligned}$$

Es sei angemerkt, dass das Maximum über $z \in \mathbb{R}^{n_u}$ größer oder gleich dem Maximum über $y \in \mathbb{R}^{n_p}$ ist, da $n_u \geq n_p$ gilt. Analog ist das Minimum über $z \in \mathbb{R}^{n_u}$ kleiner oder gleich dem Minimum über $y \in \mathbb{R}^{n_p}$. Dies zerstört die Abschätzung 2.24 jedoch nicht.

Nach Lemma 2.15 gilt die Abschätzung

$$0 \leq \lambda \left(-\tilde{N}^2 \right) \leq \delta^2$$

bzw.

$$1 \leq \lambda \left(I - \nu^{-2} \tilde{N}^2 \right) \leq 1 + \frac{\delta^2}{\nu^2}.$$

Nach Multiplikation mit ν und Bildung des Inversen erhalten wir schließlich

$$\frac{\nu}{\nu^2 + \delta^2} \leq \lambda \left(\left[\nu I - \nu^{-1} \tilde{N}^2 \right]^{-1} \right) \leq \nu^{-1}. \quad (2.28)$$

Mit $\tilde{C} := (1 + \delta^2)^{-1}$ und $\nu < 1$ lässt sich die linke Seite weiter abschätzen:

$$\tilde{C}\nu \leq \frac{\nu}{\nu^2 + \delta^2}. \quad (2.29)$$

Nun kombinieren wir (2.25), (2.26), (2.28) und (2.29) und erhalten

$$\tilde{C}\nu\gamma^2 \leq \frac{x^*Cx}{x^*x} \leq \nu^{-1}\Gamma^2.$$

Mit (2.24) folgt der erste Teil der Behauptung:

$$\tilde{C}\nu\gamma^2 \leq \operatorname{Re} \mu \leq \nu^{-1}\Gamma^2.$$

Betrachten wir nun den zweiten Rayleigh-Quotienten aus (2.24). Sei wieder $y = M_p^{-1}x$, dann gilt nach ähnlicher Rechnung wie in (2.25):

$$\frac{x^*Rx}{x^*x} = \frac{-\mathbf{i}y^*B\frac{1}{2}(A^{-1} - A^{-T})B^Ty}{y^*S_\infty y} \cdot \frac{y^*S_\infty y}{y^*M_p y}. \quad (2.30)$$

Für R rechnen wir analog zu (2.27):

$$\begin{aligned} -\frac{1}{2}(A^{-1} - A^{-T}) &= A^{-1}\frac{1}{2}(A - A^T)A^{-T} \\ &= (\nu L + N)^{-1}(N)(\nu L - N)^{-1} \\ &= L^{-1/2}(\nu I + \tilde{N})^{-1}L^{-1/2}NL^{-1/2}(\nu I - \tilde{N})^{-1}L^{-1/2} \\ &= L^{-1/2}(\nu I + \tilde{N})^{-1}\tilde{N}(\nu I - \tilde{N})^{-1}L^{-1/2} \end{aligned}$$

und erhalten für den ersten Term aus (2.30):

$$\begin{aligned} \frac{-\mathbf{i}y^*B\frac{1}{2}(A^{-1} - A^{-T})B^Ty}{y^*S_\infty y} &= \frac{\mathbf{i}y^*BL^{-1/2}(\nu I + \tilde{N})^{-1}\tilde{N}(\nu I - \tilde{N})^{-1}L^{-1/2}B^Ty}{y^*S_\infty y} \\ &= \frac{-\mathbf{i}v^*\tilde{N}v}{v^*(\nu^2 I - \tilde{N}^2)v} \end{aligned}$$

mit $v = (\nu I - \tilde{N})^{-1} L^{-1/2} B^T y$, da

$$\begin{aligned}
 v^*(\nu^2 I - \tilde{N}^2)v &= y^* B L^{-1/2} (\nu I - \tilde{N})^{-T} (\nu^2 I - \tilde{N}^2) (\nu I - \tilde{N})^{-1} L^{-1/2} B^T y \\
 &= y^* B L^{-1/2} (\nu I - \tilde{N})^{-T} (\nu I + \tilde{N}) (\nu I - \tilde{N}) (\nu I - \tilde{N})^{-1} L^{-1/2} B^T y \\
 &= y^* B L^{-1/2} (\nu I + \tilde{N})^{-1} (\nu I + \tilde{N}) L^{-1/2} B^T y \\
 &= y^* B L^{-1/2} L^{-1/2} B^T y \\
 &= -y^* S_\infty y,
 \end{aligned}$$

nach Definition 2.13 vom Schurkomplement des Stokesproblems S_∞ .

Da \tilde{N} schief-symmetrisch ist, ist $-\mathbf{i}\tilde{N}$ hermitesch und damit normal. Nach dem Spektralsatz (Satz 1.30) existiert die Zerlegung $-\mathbf{i}\tilde{N} = U\Lambda U^*$ mit einer unitären Matrix U und einer rein reellen Diagonalmatrix Λ . Damit gilt

$$\tilde{N}^2 = U\Lambda^2 U^*.$$

Daraus folgt mit $w := U^*v$:

$$\begin{aligned}
 \max_v \left| \frac{-\mathbf{i}v^* \tilde{N}v}{v^*(\nu^2 I - \tilde{N}^2)v} \right| &= \max_w \frac{|w^* \Lambda w|}{w^*(\nu^2 I + \Lambda^2)w} \\
 &\leq \max_{\lambda \in [-\delta, \delta]} \frac{|\lambda|}{\nu^2 + \lambda^2} \\
 &= \max_{\lambda \in [0, \delta]} \frac{\lambda}{\nu^2 + \lambda^2}.
 \end{aligned}$$

Als Funktion in λ ist der Bruch konkav und hat sein Maximum bei $\lambda = \nu$, d.h.

$$\max_v \left| \frac{-\mathbf{i}v^* \tilde{N}v}{v^*(\nu^2 I - \tilde{N}^2)v} \right| \leq \frac{1}{2\nu}.$$

Damit erhalten wir den zweiten Teil der Behauptung:

$$|\operatorname{Im} \mu| \leq \max_{x \neq 0} \left| \frac{x^* R x}{x^* x} \right| \leq \frac{\Gamma^2}{2\nu}. \quad \square$$

Kapitel 3

Vorkonditionierungsstrategien

In diesem Kapitel werden wir verschiedene Vorkonditionierer für das Oseen-Problem vorstellen und analysieren. Eigenwertbetrachtungen helfen uns, die Güte der Vorkonditionierer theoretisch zu untersuchen.

Wir beschränken uns auf Taylor-Hood-Elemente, weswegen keine Druckstabilisierung nötig ist und damit $C = 0$ gilt. Damit lautet die Systemmatrix nach Abschnitt 2.5

$$M = \begin{pmatrix} A & B_S^T \\ B & 0 \end{pmatrix}.$$

Das lineare Gleichungssystem lautet damit

$$M \begin{pmatrix} U \\ P \end{pmatrix} = \begin{pmatrix} A & B_S^T \\ B & 0 \end{pmatrix} \begin{pmatrix} U \\ P \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix}.$$

Weiter wird das Schurkomplement (siehe Abschnitt 2.5.3) eine wichtige Rolle spielen, welches wegen $C = 0$ die Gestalt

$$S := -B_S^T A^{-1} B$$

hat.

Wir suchen also einen Vorkonditionierer P^{-1} für

$$MP^{-1} \begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix}$$

mit einer geclusterten Eigenwertverteilung von MP^{-1} , wie in Abschnitt 1.3.2 beschrieben.

3.1 Klassifizierung der Vorkonditionierer

Alle im folgenden vorgestellten Vorkonditionierer verwenden die gleiche Blockstruktur wie die Systemmatrix M . Des Weiteren werden nur zwei verschiedene Besetzungsmuster verwendet.

In der ersten Variante sind nur die Diagonalblöcke ungleich Null, weswegen P als **Blockdiagonal-Vorkonditionierer** bezeichnet wird:

$$P_{\sphericalangle} := \begin{pmatrix} \tilde{A} & 0 \\ 0 & \tilde{S} \end{pmatrix}.$$

In der anderen Variante ist zusätzlich der rechte, obere Block besetzt. Dadurch erhält er den Namen **Blockdreiecks-Vorkonditionierer** und hat die Gestalt:

$$P_{\triangle} := \begin{pmatrix} \tilde{A} & B_S^T \\ 0 & \tilde{S} \end{pmatrix}.$$

Dabei bezeichnet \tilde{A} bzw. \tilde{S} eine Näherung an A bzw. S . Die folgenden Betrachtungen sind von der Art und Genauigkeit der Näherung unabhängig. Konkrete Vorschläge für \tilde{A} und \tilde{S} gibt es als Überblick in Abschnitt 3.2 und zum Teil in der Behandlung der konkreten Vorkonditionierer in Abschnitt 3.3.

In keinem der Vorkonditionierer wird P^{-1} als Matrix ausgerechnet, sondern nur als implizit gegebener Operator verwendet. Dies reicht aus, da FGMRES nur die Anwendung $P^{-1}v$ für beliebige Vektoren v benötigt und keinen Zugriff auf P^{-1} macht.

3.1.1 Blockdiagonal-Vorkonditionierung

Betrachten wir zunächst die Klasse der Blockdiagonal-Vorkonditionierer. Die Inverse von P_{\sphericalangle} lautet

$$P_{\sphericalangle}^{-1} = \begin{pmatrix} \tilde{A}^{-1} & 0 \\ 0 & \tilde{S}^{-1} \end{pmatrix}.$$

Durch Nachrechnen sehen wir

$$MP_{\sphericalangle}^{-1} = \begin{pmatrix} A & B_S^T \\ B & 0 \end{pmatrix} \begin{pmatrix} \tilde{A}^{-1} & 0 \\ 0 & \tilde{S}^{-1} \end{pmatrix} = \begin{pmatrix} A\tilde{A}^{-1} & B_S^T\tilde{S}^{-1} \\ B\tilde{A}^{-1} & 0 \end{pmatrix}.$$

Über die Eigenwertverteilung von $MP_{\sphericalangle}^{-1}$ lässt sich folgender Satz formulieren:

Satz 3.1: *Mit exakter Berechnung von A , d.h. $\tilde{A}^{-1} = A^{-1}$ hat $MP_{\sphericalangle}^{-1}$ die Eigenwerte*

- $\lambda = 1$ mit Vielfachheit $n_u - n_p$,
- $\frac{1}{2}(1 \pm \sqrt{1 - 4\mu_i})$ mit Vielfachheit $2n_p$.

Dabei sind μ_i , $i = 1, \dots, n_p$ die n_p Eigenwerte vom verallgemeinerten Eigenwertproblem

$$Sq_i = \mu_i \tilde{S}q_i$$

und n_u und n_p die Dimensionen vom diskreten Geschwindigkeits- und Druckraum aus Definition 2.13.

Beweis: Es gilt

$$MP_{\leftarrow}^{-1} = \begin{pmatrix} A\tilde{A}^{-1} & B_S^T\tilde{S}^{-1} \\ B\tilde{A}^{-1} & 0 \end{pmatrix} = \begin{pmatrix} I & B_S^T\tilde{S}^{-1} \\ BA^{-1} & 0 \end{pmatrix}.$$

Sei $q \in \text{Ker } B$, dann ist $(Aq, 0)^T$ Eigenvektor zum Eigenwert 1, denn

$$MP_{\leftarrow}^{-1}(Aq, 0)^T = (Aq, BA^{-1}Aq) = (Aq, 0)^T.$$

Wegen $\dim \text{Ker } B = n_u - n_p$ (siehe (2.19)) hat der Eigenwert 1 die Vielfachheit $n_u - n_p$.

Für die restlichen Eigenwerte betrachten wir das Eigenwertproblem

$$\begin{pmatrix} I & B_S^T\tilde{S}^{-1} \\ BA^{-1} & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} - \lambda \begin{pmatrix} u \\ p \end{pmatrix} = 0$$

und damit

$$\begin{aligned} (1 - \lambda)u + B_S^T\tilde{S}^{-1}p &= 0 \\ BA^{-1}u - \lambda p &= 0. \end{aligned}$$

Wir nehmen nun an, dass $\lambda \neq 1$ ist. Dann lässt sich die erste Zeile umformen zu

$$u = \frac{-1}{(1 - \lambda)} B_S^T\tilde{S}^{-1}p.$$

Einsetzen in die zweite Zeile ergibt:

$$\frac{-1}{(1 - \lambda)} BA^{-1} B_S^T\tilde{S}^{-1}p - \lambda p = 0.$$

Nach Definition ist $-BA^{-1}B_S^T = S$ und damit

$$\frac{1}{(1 - \lambda)} Sp = \lambda\tilde{S}p.$$

Einsetzen von Eigenwerten und Eigenvektoren (μ_i, q_i) der Behauptung ergibt

$$\frac{\mu_i}{(1 - \lambda)} \tilde{S}q_i = \lambda\tilde{S}q_i$$

und damit

$$\lambda(1 - \lambda) - \mu_i = 0$$

bzw.

$$\lambda = \frac{1}{2}(1 \pm \sqrt{1 - 4\mu_i}).$$

Damit haben wir genau $2n_p$ Eigenwerte der Matrix, da $i = 1, \dots, n_p$. □

3.1.2 Blockdreiecks-Vorkonditionierung

Die Inverse von P_Δ lautet wie folgt und lässt sich in folgender Weise zerlegen:

$$\begin{aligned} P_\Delta^{-1} &= \begin{pmatrix} \tilde{A}^{-1} & -\tilde{A}^{-1}B_S^T\tilde{S}^{-1} \\ 0 & \tilde{S}^{-1} \end{pmatrix} \\ &= \begin{pmatrix} \tilde{A}^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & B_S^T \\ 0 & -I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -\tilde{S}^{-1} \end{pmatrix}. \end{aligned}$$

Diese Zerlegung werden wir bei der Auswertung von $P_\Delta^{-1}v$ verwenden. Damit sind pro Auswertung nur eine Lösung $\tilde{A}^{-1}u$, eine von $\tilde{S}^{-1}p$, eine Multiplikation mit B_S^T und einige Vorzeichenoperationen notwendig.

Lemma 3.2: *Es gilt*

$$MP_\Delta^{-1} = \begin{pmatrix} A\tilde{A}^{-1} & (\tilde{A} - A)A^{-1}B_S^T\tilde{S}^{-1} \\ B\tilde{A}^{-1} & S\tilde{S}^{-1} \end{pmatrix}. \quad (3.1)$$

Beweis: Ausmultiplizieren von MP_Δ^{-1} liefert die Behauptung. \square

Lemma 3.3: *Mit exakter Berechnung von P_Δ (d.h. $\tilde{A} = A$ und $\tilde{S} = S$) hat MP_Δ^{-1} nur den Eigenwert 1 mit Vielfachheit $n_u + n_p$.*

Beweis: Mit Lemma 3.2 gilt

$$MP_\Delta^{-1} = \begin{pmatrix} I & 0 \\ BA^{-1} & I \end{pmatrix} \quad (3.2)$$

und damit folgt die Behauptung. \square

Mit exakter Berechnung von P_Δ sind demnach alle Eigenwerte gleich eins. Was das für die benötigten Iterationsschritte aussagt, zeigt folgendes Lemma:

Lemma 3.4: *Mit exakter Berechnung von P_Δ konvergiert das GMRES-Verfahren in maximal 2 Iterationen.*

Beweis: Die Matrix MP_Δ^{-1} ist nicht diagonalisierbar, deswegen ist Satz 1.39 nicht anwendbar.

Wir definieren

$$K = \begin{pmatrix} I & 0 \\ L & I \end{pmatrix} := MP_\Delta^{-1}.$$

Nehmen wir an, wir starten mit der Startlösung $x_0 = 0$. Dann lautet der Krylowraum in Iteration 2 mit $b = (b_1, b_2)^T$:

$$\begin{aligned} \mathcal{K}_m(K, b) &= \text{span}\{b, Kb, K^2b\} \\ &= \text{span}\left\{ \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \begin{pmatrix} b_1 \\ Lb_1 + b_2 \end{pmatrix}, \begin{pmatrix} b_1 \\ 2 \cdot Lb_1 + b_2 \end{pmatrix} \right\} \\ &= \text{span}\left\{ \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \begin{pmatrix} 0 \\ Lb_1 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \end{pmatrix} \right\}. \end{aligned}$$

Die Vektoren sind demnach linear abhängig und nach Satz 1.36 bricht das GMRES-Verfahren damit in Iteration 2 ab und hat bereits die korrekte Lösung gefunden. \square

Die exakte Berechnung von P_Δ ist nicht sinnvoll, weil dadurch das System bereits gelöst ist, ohne dass man ein GMRES-Verfahren verwenden muss. Gerade die direkte Berechnung von A^{-1} und S^{-1} wollten wir durch die Verwendung eines vorkonditionierten GMRES-Verfahrens vermeiden.

Die Hoffnung ist, dass wir auch mit einer guten Näherung von P_Δ geringe Iterationszahlen erhalten werden. Unser Ziel wird es deswegen sein, die Eigenwerte von MP_Δ^{-1} unabhängig von h und ν zu beschränken.

3.2 Behandlung der Teilprobleme

Wenn wir günstig zu berechnende Näherungen \tilde{A}^{-1} bzw. \tilde{S}^{-1} an A^{-1} bzw. S^{-1} finden, ist damit die effiziente Lösung des Oseen-Problems gegeben.

3.2.1 Approximation des Geschwindigkeitsblocks

Für die Berechnung von $\tilde{A}^{-1}v$ gibt es einige vielversprechende Ansätze.

Einfache Varianten sind unvollständige LU-Zerlegungen der Matrix oder die Nutzung eines iterativen Lösers mit einfachen Vorkonditionierern wie SSOR oder wieder LU-Zerlegungen, wie zum Beispiel ILU(0). Für moderate Viskositäten ν funktioniert das sehr gut, aber für kleine Viskositäten stagniert das iterative Lösungsverfahren.

Wesentlich besser arbeiten einzelne geometrische Mehrgitter V- oder W-Zykel, siehe zum Beispiel [OB06].

Bei nicht zu großen Problemen eignen sich auch direkte Lösungsverfahren, die allerdings in Bezug auf den Speicherbedarf schlechter skalieren.

Weiterhin wäre es interessant, Gebietszerlegungsverfahren zur Vorkonditionierung zu verwenden, was sich gerade bei großen Problemen anbietet. Es stellt sich heraus, dass die Approximation der Matrix A^{-1} nicht sehr exakt sein muss, wodurch das Gebietszerlegungsverfahren nicht auskonvergiert sein muss. Dies werden wir später feststellen, siehe Abschnitt 4.4.1, Tabelle 4.2.

3.2.2 Approximation des Schurkomplements

Die Approximation des Schurkomplements $S = -B_S^T A^{-1} B$ stellt sich als wesentlich schwieriger dar als die Approximation des Geschwindigkeitsblocks.

Für $\nu \approx 1$ können wir S durch die Massematrix M_p nach Satz 2.14 gut approximieren. Satz 2.16 zeigt, dass das für $\nu \ll 1$ aber nicht mehr funktioniert. Verschiedene Autoren beschäftigen sich entweder damit, S geeignet zu approximieren oder die Lösung zu umgehen. Mehr dazu im folgenden Abschnitt.

3.3 Vorkonditionierer

Zunächst stellen wir vier verschiedene Vorkonditionierer vor, die in den letzten Jahren entwickelt wurden. Davon liefert der Olshanskii-Vorkonditionierer in 3.3.4 theoretisch die besten Ergebnisse, was sich dann auch in der Praxis bestätigt. Unter anderem ist für den Olshanskii-Vorkonditionierer auch die ausführlichste theoretische Behandlung möglich, so dass explizite Schranken für die Eigenwertverteilung des vorkonditionierten Systems angegeben werden können.

In Abschnitt 3.3.5 betrachten wir noch einen fünften Vorkonditionierer, der viele Gemeinsamkeiten mit dem Olshanskii-Vorkonditionierer besitzt. Die etwas abgewandelte Konstruktion erschwert die theoretische Betrachtung, ermöglicht aber eine bessere Verwendung mit dem stabilisierten System.

3.3.1 Blockdiagonal-Vorkonditionierer von Silvester und Wathen

Silvester und Wathen ([SW93], [SW94]) schlagen den relativ leicht zu berechnenden Blockdiagonal-Vorkonditionierer

$$P_{\text{WS}} := \begin{pmatrix} \tilde{A} & 0 \\ 0 & I/\omega \end{pmatrix} \quad (3.3)$$

vor. Dabei ist $\omega \in \mathbb{R}$ ein noch zu bestimmender Parameter und \tilde{A} eine geeignete Näherung an A . Dieser Vorkonditionierer ist eigentlich für das Stokes-Problem (siehe Abschnitt 2.5.4) entwickelt und liefert keine guten Ergebnisse für kleine ν .

Satz 3.5: Sei $\tilde{A} = A$ (also exakt gelöst), habe B vollen Zeilenrang und seien (γ_i, c_i) die m Paare aus Eigenwert und zugehörigem Eigenvektor vom Schurkomplement S .

Dann sind die Eigenwerte der vorkonditionierten Matrix MP_{WS}^{-1}

- 1 mit Vielfachheit $n - m$
- $(1 \pm \sqrt{1 - 4\omega\gamma_i})/2$ mit Vielfachheit $2m$

Beweis: Das verallgemeinerte Eigenwertproblem

$$Sp_i = \mu_i \omega^{-1} Ip_i$$

hat die Eigenwerte $\mu_i = \omega\gamma_i$. Damit folgt die Behauptung nach Satz 3.1. \square

Die Eigenwerte bilden dabei drei Cluster: um 1, um $(1 + \sqrt{1 - 4\omega\gamma_i})/2$ und um $(1 - \sqrt{1 - 4\omega\gamma_i})/2$.

Zur Wahl von ω empfiehlt [dNW06] im Stokes-Fall $\omega = 1$ und für das Oseen-Problem $\omega = \nu$. Es stellt sich heraus, dass das Verfahren relativ unsensibel gegenüber der Wahl von ω ist.

Für kleine ν ist die Identität eine schlechte Näherung an das Schurkomplement. Dadurch scheint der Löser eher für Stokes geeignet, was sich in den Rechnungen bestätigt, siehe 4.4.2, Tabelle 4.4.

3.3.2 Pressure-convection-diffusion-Vorkonditionierer

Ein relativ bekannter Vorkonditionierer für das Oseen–Problem ist der sogenannte pressure-convection-diffusion-Vorkonditionierer (kurz PCD) entwickelt von Kay, Loghin und Wathen in [KLW02]. Es handelt sich um einen Blockdreiecksvorkonditionierer P_Δ :

$$P_\Delta = \begin{pmatrix} \tilde{A} & B_S^T \\ 0 & \tilde{S} \end{pmatrix}.$$

Dabei ist \tilde{A} eine Näherung an A (wie gehabt), die Approximation vom Schurkomplement S hingegen ist komplizierter. Die Autoren wählen dafür

$$\tilde{S}^{-1} := -M_p^{-1}F_pA_p^{-1}.$$

Dabei ist M_p die Massematrix im Druckraum, F_p ein Konvektions-Diffusions-Operator im Druckraum und A_p ein Laplace-Operator im Druckraum.

Das System für F_p und für A_p muss zusätzlich assembliert werden. Das Hauptproblem ist dabei, dass man sich geeignete Randbedingungen überlegen muss. In [KLW02] werden reine Neumann-Bedingungen vorgeschlagen, was damit erklärt wird, dass schließlich auch der Druck im eigentlichen System nur bis auf Konstanten bestimmt ist.

Satz 3.6: *Mit exakter Berechnung von A^{-1} und \tilde{S}^{-1} hat der Eigenwert 1 des vorkonditionierten Systems die Vielfachheit n_u und die restlichen n_p Eigenwerte entsprechen den Eigenwerten der Matrix $S\tilde{S}^{-1}$.*

Beweis: Nach Lemma 3.2 gilt

$$MP_\Delta^{-1} = \begin{pmatrix} A\tilde{A}^{-1} & (\tilde{A} - A)A^{-1}B_1^T\tilde{S}^{-1} \\ B_2\tilde{A}^{-1} & S\tilde{S}^{-1} \end{pmatrix} = \begin{pmatrix} I & 0 \\ B_2A^{-1} & S\tilde{S}^{-1} \end{pmatrix}.$$

Damit sind n_u Eigenwerte gleich 1 und die restlichen n_p genügen

$$S\tilde{S}^{-1}p = \lambda_i p. \quad \square$$

Die Qualität dieses Löser hängt also zusammengefasst davon ab, wie gut \tilde{S}^{-1} das Schurkomplement annähert. In [OV07] ist die h -Unabhängigkeit der Eigenwertverteilung bei exakter Berechnung der Hilfsprobleme bewiesen.

Bei der Analyse von numerischen Tests in [OV07] kommen die Autoren zu dem Schluß, dass es eine leichte Abhängigkeit von der Viskosität in der Größenordnung $\mathcal{O}(\nu^{-1/3})$ gibt, selbst bei exakter Berechnung der verwendeten Operatoren.

Für $c = 0$ lässt sich die Wahl von \tilde{S}^{-1} relativ anschaulich motivieren. Die starke, kontinuierliche Formulierung zum Schurkomplement lautet

$$BA^{-1}B^T \sim -\nabla \cdot (-\nu\Delta + b \cdot \nabla)^{-1}\nabla.$$

Wenn wir nun annehmen, dass die Operatoren kommutieren, so erhalten wir

$$-BA^{-1}B^T \sim (-\nu\Delta + b \cdot \nabla)^{-1}(-\Delta).$$

3.3. Vorkonditionierer

Das würde die Wahl $\tilde{S}^{-1} = -F_p A_p^{-1}$ motivieren, aber auch für die Massematrix in \tilde{S} gibt es eine Erklärung: Wenn wir uns das Stokes-Problem ansehen, würde man damit

$$S_\infty^{-1} \sim -(-\nu\Delta)(-\Delta)^{-1} = -\nu I$$

erhalten. In Satz 2.14 haben wir aber gesehen, dass die Massematrix M_p eine bessere Näherung für S_∞ ist. So erhält man schließlich die Wahl

$$\tilde{S}^{-1} = -M_p^{-1} F_p A_p^{-1}.$$

Neben dem Problem der Randbedingungen besteht eine zusätzliche Schwierigkeit darin, dass das Konvektions-Diffusions-Problem F_p für kleine ν stabilisiert werden muss.

In Tabelle 3.1 sind die FGMRES-Iterationszahlen bei PCD-Vorkonditionierung für verschiedene h und ν dargestellt. Es handelt sich um ein glattes Oseen-Problem mit Referenzlösung (siehe Abschnitt 4.2.1). Alle Teilprobleme sind exakt gelöst, das System F_p wurde nicht stabilisiert und das Abbruchkriterium ist das relative Residuum von 10^{-10} zur Startlösung Null.

1/h	ν				
	10^0	10^{-1}	10^{-2}	10^{-3}	10^{-4}
16	19	25	57	-	-
32	19	26	50	-	-
64	18	25	41	954	-
128	17	24	39	287	-
256	16	23	38	127	-

Tabelle 3.1: Iterationszahlen bei PCD-Vorkonditionierung für verschiedene h und ν

Man erkennt die geringen und h -unabhängigen Iterationszahlen für moderate ν . Für $\nu < 10^{-2}$ funktioniert der Vorkonditionierer nur für feine Gitter. Dies liegt wahrscheinlich daran, dass das System F_p nicht stabilisiert ist, was aber für kleine ν notwendig ist, solange nicht alle Details der Lösung komplett aufgelöst werden. Aber selbst für $h = 1/256$ erkennt man die unerwünschte ν -Abhängigkeit der Iterationszahlen.

3.3.3 BFBt-Vorkonditionierer

Analog zum PCD-Vorkonditionierer stellt Elman in [Elm99] eine alternative Näherung an S vor. Er wählt auch

$$P_{\text{BFBt}} = \begin{pmatrix} \tilde{A} & B^T \\ 0 & -\tilde{S} \end{pmatrix}, \quad (3.4)$$

aber diesmal (bei stetigen finiten Elementen für den Druck)

$$\tilde{S} = (BB^T)(BAB^T)^{-1}(BB^T)$$

bzw.

$$\tilde{S}^{-1} = (BB^T)^{-1}(BAB^T)(BB^T)^{-1}.$$

Die Bezeichnung BFBt ist entstanden, da in [Elm99] A mit F bezeichnet wird.

Im Gegensatz zum PCD-Vorkonditionierer sind hier keine zusätzlichen Systemmatrizen aufzustellen, des Weiteren gibt es keine Probleme mit zu bestimmenden Randbedingungen.

Laut [Elm99] existiert aber eine leichte Abhängigkeit der Iterationszahl von h und ν in der Form $\mathcal{O}(\nu^{-1/2}h^{-1/2})$.

3.3.4 Olshanskii-Vorkonditionierer

In [OB06] wird eine interessante Technik vorgestellt, die durch Manipulation des Ausgangssystems die Problematik der Approximation des Schurkomplements im Vorkonditionierer umgeht.

Das ursprüngliche System lautet

$$\begin{pmatrix} A & B_S^T \\ B & 0 \end{pmatrix} \begin{pmatrix} U \\ P \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix}. \quad (3.5)$$

Sei $W \in \mathbb{R}^{n_p \times n_p}$ zunächst eine beliebige positiv definite Matrix. Dann modifizieren wir die Systemmatrix M auf folgende Weise:

$$M_\gamma := \begin{pmatrix} A_\gamma & B_S^T \\ B & 0 \end{pmatrix} \quad (3.6)$$

mit $\gamma > 0$ und

$$A_\gamma := A + \gamma B_S^T W^{-1} B. \quad (3.7)$$

Die Gleichungssysteme (3.5) und

$$M_\gamma \begin{pmatrix} U \\ P \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix} \quad (3.8)$$

sind äquivalent, da $BU = 0$ gilt (wegen der zweiten Zeile in M beziehungsweise M_γ). Diese Veränderung des Systems ist als augmented lagrangian (AL) Methode in der Literatur bekannt und wurde bereits vorher verwendet und analysiert, siehe zum Beispiel [GG03].

Als Vorkonditionierer wählt Olshanskii den Blockdreiecks-Vorkonditionierer

$$P_{\text{Osh}} := \begin{pmatrix} \tilde{A}_\gamma & B_S^T \\ 0 & \tilde{S} \end{pmatrix},$$

aber diesmal mit einer Näherung an A_γ im $(1,1)$ -Block. Für das zu approximierende Schurkomplement wählt er:

$$\tilde{S}^{-1} := -\nu M_p^{-1} - \gamma W^{-1}. \quad (3.9)$$

Als W wählen wir die Massematrix M_p oder eine geeignete Approximation wie ein Zeilenlumping zur einfacheren Berechnung. Mit $W = M_p$ reduziert sich die Näherung des Schurkomplements auf eine umskalierte Massematrix:

$$\tilde{S}^{-1} = -(\nu + \gamma)M_p^{-1}.$$

Die Manipulation im System bewirkt, dass auch für kleinere ν die Massematrix eine gute Approximation für das Schurkomplement bleibt, was wir gleich sehen werden. Die Lösung der Gleichungen $\tilde{A}_\gamma^{-1}v$ und $M_p^{-1}q$ und die Wahl des Parameters γ wird danach behandelt.

Das exakte Schurkomplement von (3.8) hat sich natürlich gegenüber S verändert und wir bezeichnen es mit

$$S_\gamma := BA_\gamma^{-1}B_S^T = B(A + \gamma B_S^T M_p^{-1} B)^{-1} B_S^T. \quad (3.10)$$

Folgendes Lemma bringt die Inversen von S und S_γ in Zusammenhang, was wir für den Beweis über die Eigenwertverteilung benötigen werden:

Lemma 3.7: *Es gilt*

$$-S_\gamma^{-1} = -S^{-1} + \gamma M_p^{-1}$$

beziehungsweise

$$\left[B(A + \gamma B_S^T M_p^{-1} B)^{-1} B_S^T \right]^{-1} = (BA^{-1}B_S^T)^{-1} + \gamma M_p^{-1}. \quad (3.11)$$

Beweis: Sei $D^{-1} := \gamma M_p^{-1}$. Mit der Woodbury-Matrixidentität (Satz 1.25) sehen wir

$$\begin{aligned} (A + \gamma B_S^T M_p^{-1} B)^{-1} &= A^{-1} - A^{-1} B_S^T (D + BA^{-1} B_S^T)^{-1} BA^{-1} \\ &= A^{-1} - A^{-1} B_S^T D^{-1} (I + BA^{-1} B_S^T D^{-1})^{-1} BA^{-1}. \end{aligned}$$

Wir definieren $S := BA^{-1}B_S^T$ und rechnen damit

$$\begin{aligned} &B(A + \gamma B_S^T M_p^{-1} B)^{-1} B_S^T \\ &= B \left(A^{-1} - A^{-1} B_S^T D^{-1} (I + BA^{-1} B_S^T D^{-1})^{-1} BA^{-1} \right) B_S^T \\ &= S - SD^{-1} (I + SD^{-1})^{-1} S. \end{aligned}$$

Für das Produkt der Matrizen aus (3.11) gilt dadurch:

$$\begin{aligned} &\left[B(A + \gamma B_S^T M_p^{-1} B)^{-1} B_S^T \right] \left[(BA^{-1}B_S^T)^{-1} + \gamma M_p^{-1} \right] \\ &= \left[S - SD^{-1} (I + SD^{-1})^{-1} S \right] \left[S^{-1} + D^{-1} \right] \\ &= I + SD^{-1} - SD^{-1} (I + SD^{-1})^{-1} S \left[S^{-1} + D^{-1} \right] \\ &= I + SD^{-1} - SD^{-1} (I + SD^{-1})^{-1} (I + SD^{-1}) \\ &= I + SD^{-1} - SD^{-1} \\ &= I. \quad \square \end{aligned}$$

Nun können wir uns die Eigenwerte des vorkonditionierten Systems anschauen:

Satz 3.8: *Mit $W = M_p$ und exakter Lösung von \tilde{A}_γ^{-1} und \tilde{S}^{-1} hat $M_\gamma P_{\text{Olsb}}^{-1}$ den Eigenwert 1 mit Vielfachheit n . Die restlichen m Eigenwerte sind*

$$\lambda_i = \frac{\gamma + \nu}{\gamma + \mu_i^{-1}}.$$

Dabei sind (μ_i, q_i) , $i = 1, \dots, n_p$, Paare aus Eigenwerten und zugehörigen Eigenvektoren vom verallgemeinerten Eigenwertproblem

$$-Sq_i = \mu_i M_p q_i.$$

Beweis: Wir betrachten das Eigenwertproblem

$$M_\gamma P_{\text{Olsh}}^{-1} \begin{pmatrix} u \\ p \end{pmatrix} = \lambda \begin{pmatrix} u \\ p \end{pmatrix} \quad (3.12)$$

und nach Lemma 3.2 gilt

$$M_\gamma P_{\text{Olsh}}^{-1} = \begin{pmatrix} I & 0 \\ BA_\gamma^{-1} & S_\gamma \tilde{S}^{-1} \end{pmatrix}$$

mit $S_\gamma = -BA_\gamma^{-1}B_S^T$. Damit hat der Eigenwert $\lambda = 1$ die Vielfachheit n . Nehmen wir nun an, dass $\lambda \neq 1$ ist. Dann folgt aus der ersten Gleichung von (3.12) wegen $u = \lambda u$ für die Geschwindigkeitskomponente des Eigenvektors zu λ :

$$u = 0.$$

Damit lautet die zweite Zeile der Eigenwertgleichung

$$S_\gamma \tilde{S}^{-1} p = \lambda p$$

und Einsetzen ergibt:

$$-(\nu + \gamma)M_p^{-1}p = \lambda S_\gamma^{-1}p.$$

Nach Lemma 3.7 gilt dann:

$$(\nu + \gamma)M_p^{-1}p = \lambda(-S^{-1} + \gamma M_p^{-1})p.$$

Seien nun (μ_i, q_i) Paare aus Eigenwerten und Eigenvektoren vom Eigenwertproblem

$$-Sq_i = \mu_i M_p q_i.$$

Dann gilt mit $r_i = M_p q_i$

$$-S^{-1}r_i = \mu_i^{-1}M_p^{-1}r_i$$

und Einsetzen ergibt:

$$(\nu + \gamma)M_p^{-1}r_i = \lambda_i(\mu_i^{-1}M_p^{-1} + \gamma M_p^{-1})r_i.$$

Dadurch gilt für λ_i

$$(\nu + \gamma) = \lambda_i(\mu_i^{-1} + \gamma)$$

beziehungsweise

$$\lambda_i = \frac{\gamma + \nu}{\gamma + \mu_i^{-1}}. \quad \square$$

Damit können wir sogar Grenzen für die Eigenwerte bestimmen:

Satz 3.9: Für die Eigenwerte λ_i von $M_\gamma P_{\text{Olsh}}^{-1}$ ohne Stabilisierung gilt:

$$1 - \frac{1}{\gamma \operatorname{Re}(\mu_i) + 1} \leq \operatorname{Re}(\lambda_i) < 1 + \nu \gamma^{-1}$$

und

$$|\operatorname{Im}(\lambda_i)| < (1 + \nu \gamma^{-1}) \min\{\gamma |\operatorname{Im}(\mu_i)|, \frac{1}{\gamma |\operatorname{Im}(\mu_i)|}\}.$$

Beweis: Nach Satz 3.8 gilt

$$\lambda_i = \frac{\gamma + \nu}{\gamma + \mu_i^{-1}}.$$

Sei $\mu_i =: a + b\mathbf{i}$ mit $a, b \in \mathbb{R}$. Nach Satz 2.16 ist $a > 0$ und deswegen gilt:

$$\begin{aligned} \lambda_i &= \frac{\gamma + \nu}{\gamma + (a + b\mathbf{i})^{-1}} \\ &= \frac{\gamma + \nu}{\gamma + \frac{a - b\mathbf{i}}{a^2 + b^2}} \\ &= \frac{(\gamma + \nu)(a^2 + b^2)}{\gamma(a^2 + b^2) + a - b\mathbf{i}} \\ &= \frac{(\gamma + \nu)(a^2 + b^2)(\gamma(a^2 + b^2) + a) + \mathbf{i}(\gamma + \nu)(a^2 + b^2)b}{(\gamma(a^2 + b^2) + a)^2 + b^2} \\ &= \frac{(\gamma + \nu)(a^2 + b^2)(\gamma(a^2 + b^2) + a) + \mathbf{i}(\gamma + \nu)(a^2 + b^2)b}{\gamma^2(a^2 + b^2)^2 + 2a\gamma(a^2 + b^2) + a^2 + b^2} \\ &= \frac{(\gamma + \nu)(\gamma(a^2 + b^2) + a) + \mathbf{i}(\gamma + \nu)b}{\gamma^2(a^2 + b^2) + 2a\gamma + 1} \end{aligned}$$

damit ist

$$\begin{aligned} \operatorname{Re} \lambda_i &= \frac{(\gamma + \nu)(\gamma(a^2 + b^2) + a)}{\gamma^2(a^2 + b^2) + 2a\gamma + 1} \\ &\geq \frac{\gamma^2 a^2 + \gamma^2 b^2 + a\gamma}{\gamma^2 a^2 + \gamma^2 b^2 + 2a\gamma + 1} \\ &= \frac{\gamma^2 a^2 + \gamma^2 b^2 + 2a\gamma + 1}{\gamma^2 a^2 + \gamma^2 b^2 + 2a\gamma + 1} - \frac{a\gamma + 1}{\gamma^2 a^2 + \gamma^2 b^2 + 2a\gamma + 1} \\ &= 1 - \frac{a\gamma + 1}{\gamma^2 a^2 + \gamma^2 b^2 + 2a\gamma + 1} \\ &= 1 - \frac{a\gamma + 1}{(a\gamma + 1)^2 + \gamma^2 b^2} \\ &\geq 1 - \frac{1}{a\gamma + 1} \end{aligned}$$

bzw.

$$\begin{aligned} \operatorname{Re} \lambda_i &= \frac{(\gamma + \nu)(\gamma(a^2 + b^2) + a)}{\gamma^2(a^2 + b^2) + 2a\gamma + 1} \\ &= (1 + \nu \gamma^{-1}) \frac{\gamma^2(a^2 + b^2) + a\gamma}{\gamma^2(a^2 + b^2) + 2a\gamma + 1} \\ &< (1 + \nu \gamma^{-1}) \end{aligned}$$

und

$$\begin{aligned} |\operatorname{Im} \lambda_i| &= \frac{(\gamma + \nu)|b|}{\gamma^2(a^2 + b^2) + 2a\gamma + 1} \\ &= (1 + \nu\gamma^{-1}) \frac{\gamma|b|}{\gamma^2 a^2 + \gamma^2 b^2 + 2a\gamma + 1}. \end{aligned}$$

Damit ist nun entweder

$$|\operatorname{Im} \lambda_i| \leq (1 + \nu\gamma^{-1}) \frac{\gamma|b|}{1}$$

oder

$$|\operatorname{Im} \lambda_i| \leq (1 + \nu\gamma^{-1}) \frac{\gamma|b|}{\gamma^2 b^2} = (1 + \nu\gamma^{-1}) \frac{1}{\gamma|b|},$$

was zu beweisen war. \square

Wenn wir den Parameter γ geeignet wählen, lassen sich die Eigenwerte unabhängig von h und ν beschränken:

Satz 3.10: Sei $\nu \leq 1$ und $\gamma \in \mathcal{O}(\nu^{-1})$. Dann sind ohne Stabilisierung alle Eigenwerte von $M_\gamma P_{\text{Olsh}}^{-1}$ im Rechteck $[a, A] \times [-b, b]$, unabhängig von ν , h und mit $a > 0$, enthalten.

Beweis: Nach Satz 2.16 gilt

$$c\nu \leq \operatorname{Re} \mu_i \leq C\nu^{-1} \quad \text{und} \quad |\operatorname{Im} \mu_i| \leq C_2\nu^{-1}$$

und nach Satz 3.9 gelten die Abschätzungen

$$1 - \frac{1}{\gamma \operatorname{Re}(\mu_i) + 1} \leq \operatorname{Re}(\lambda_i) < 1 + \nu\gamma^{-1}$$

und

$$|\operatorname{Im}(\lambda_i)| < (1 + \nu\gamma^{-1}) \min\{\gamma|\operatorname{Im}(\mu_i)|, \frac{1}{\gamma|\operatorname{Im}(\mu_i)|}\}.$$

Wir wählen $\gamma = \alpha\nu^{-1}$ und schätzen mit $0 < \nu \leq 1$ ab:

$$\begin{aligned} \operatorname{Re}(\lambda_i) &\geq 1 - \frac{1}{\gamma \operatorname{Re}(\mu_i) + 1} \geq 1 - \frac{1}{\alpha\nu^{-1}c\nu + 1} = 1 - \frac{1}{\alpha c + 1} =: a > 0, \\ \operatorname{Re}(\lambda_i) &< 1 + \nu\gamma^{-1} = 1 + \nu\alpha\nu^{-1} = 1 + \alpha =: A, \\ |\operatorname{Im}(\lambda_i)| &< (1 + \nu^2\alpha^{-1}) \min\left\{\frac{\alpha C_2}{\nu^2}, \frac{\nu^2}{\alpha C_2}\right\} \leq (1 + \alpha^{-1}) =: b. \quad \square \end{aligned}$$

Problematisch an diesem Ergebnis ist, dass die Matrix A_γ für große γ schlecht konditioniert ist, da der Teil $B^T W^{-1} B$ mit Nullraum der Dimension $n - m$ die Matrix dominiert.

In der Praxis reicht es aber, $\gamma \in \mathcal{O}(1)$ zu wählen. Dies bestätigen auch die Rechnungen in [OB06]. Es lässt sich noch anmerken, dass Olshanskii bereits gute Ergebnisse mit Zeilenlumping von $W = M_p$ erreicht. Des Weiteren stellt Olshanskii in [OB06] einen speziellen Mehrgitter-Löser für den A_γ -Block vor.

3.3.5 Grad-Div-Vorkonditionierer

In Abschnitt 3.3.4 haben wir gesehen, wie man die Lösung des Schurkomplements im Vorkonditionierer durch eine augmented lagrangian-Technik vereinfacht. Der Grad-Div-Vorkonditionierer baut auf dieser Idee auf und weist deshalb sehr große Ähnlichkeiten zum Olshanskii-Vorkonditionierer auf.

Der veränderte A -Block im Olshanskii-Vorkonditionierer lautet:

$$A_\gamma = A + \gamma B^T W^{-1} B.$$

Mit $W = I$ addieren wir den Term $\gamma B^T B$. In der starken Formulierung entspricht $\gamma B^T B$ dem Term $\gamma \nabla(\nabla \cdot)$, genau wie bei der Grad-Div-Stabilisierung (siehe Abschnitt 2.4.1), die wie folgt schwach formuliert wird:

$$\gamma(\nabla \cdot u, \nabla \cdot v).$$

Im Diskreten ist dies nicht exakt das gleiche, da die Addition von $\gamma B^T W^{-1} B$ zu einem äquivalenten Gleichungssystem führt, während die Grad-Div-Stabilisierung die Lösung verändert.

Die Idee vom Grad-Div-Vorkonditionierer (kurz GD) ist nun, den Olshanskii-Vorkonditionierer auf das Grad-Div-stabilisierte ursprüngliche System - also mit $\gamma = 0$ - anzuwenden.

Wir verwenden auch

$$S_\gamma^{-1} := -(\nu + \gamma)M_p^{-1}.$$

Die Lösung wird also verändert, was eigentlich nicht von einem Vorkonditionierer erwünscht ist. Allerdings wirkt sich der Effekt der Stabilisierung positiv auf die Lösung aus, da Oszillationen gedämpft werden.

Ein Problem ist, dass die Wahl von γ für den Vorkonditionierer und für die Stabilisierung selbst zueinander passen müssen, um gute Ergebnisse zu erzielen. Allerdings erwarten wir in beiden Fällen, dass $\gamma \in \mathcal{O}(1)$ eine gute Wahl ist.

Eine theoretische Betrachtung der Eigenwerte ist aufgrund der Stabilisierung nur schwer möglich. Dies ist etwas unbefriedigend, aber mit zum Olshanskii-Vorkonditionierer vergleichbaren Ergebnissen kann gerechnet werden.

Ein Vorteil vom GD- gegenüber dem Olshanskii-Vorkonditionierer ist, dass die Systemmatrix direkt assemblierbar ist. Außerdem ist die Verwendung von Stabilisierung für die Qualität der Lösung sinnvoll. Wenn man bei der Verwendung des Olshanskii-Vorkonditionierers stabilisieren wollte, ist offen, wie sich die Verwendung von Grad-Div-Stabilisierung auf den Vorkonditionierer ausübt.

Der Unterschied zwischen der Grad-Div-Stabilisierung S_{GD} und dem $\gamma B^T M_p^{-1} B$ -Term lässt sich auch im Diskreten näher bestimmen. Dazu definieren wir die L^2 -Projektion π vom Divergenzraum der Geschwindigkeit

$$V_{\text{DIV}} := \text{span}\{\nabla \cdot \phi_1, \dots, \nabla \cdot \phi_{n_u}\}$$

in den Druckraum $Q_h = \text{span}\{\psi_1, \dots, \psi_{n_p}\}$:

$$\pi : V_{\text{DIV}} \longrightarrow Q_h$$

durch

$$(\pi(\nabla \cdot \phi_j), \psi_i) = (\nabla \cdot \phi_j, \psi_i) \quad \forall i = 1, \dots, n_p. \quad (3.13)$$

Wir definieren weiter eine Darstellung von π als Matrix $P = (P_{i,j})$ durch

$$\pi(\nabla \cdot \phi_j) = \sum_{i=1}^{n_p} P_{i,j} \psi_i \quad \forall j = 1, \dots, n_u.$$

Satz 3.11: *Es gilt*

$$(S_{GD} - B^T M_p^{-1} B)_{ij} = (\kappa(\nabla \cdot \phi_j), \kappa(\nabla \cdot \phi_i))$$

mit dem Fluktuationsoperator $\kappa := I - \pi$.

Beweis: Nach Abschnitt 2.3.3 und 2.4.1 gilt

$$\begin{aligned} (S_{GD})_{ij} &= (\nabla \phi_j, \nabla \cdot \phi_i), \\ (B^T)_{ij} &= (\psi_j, \nabla \cdot \phi_i), \\ (B)_{ij} &= (\psi_i, \nabla \cdot \phi_j), \\ (M_p)_{ij} &= (\psi_i, \psi_j). \end{aligned}$$

Damit gilt $B = M_p P$ wegen

$$\begin{aligned} (B)_{i,j} &= \left(\sum_{k=1}^{n_p} P_{k,j} \psi_k, \psi_i \right) \\ &= \sum_{k=1}^{n_p} P_{k,j} (\psi_k, \psi_i) \\ &= \sum_{k=1}^{n_p} (\psi_i, \psi_k) P_{k,j} \\ &= (M_p P)_{i,j}. \end{aligned}$$

Für den Eintrag i, j der Differenz der Grad-Div-Stabilisierung und dem $B^T M_p^{-1} B$ -Term gilt

$$\begin{aligned} (S_{GD} - B^T M_p^{-1} B)_{ij} &= (\nabla \cdot \phi_j, \nabla \cdot \phi_i) - (B^T M_p^{-1} B)_{ij} \\ &= (\nabla \cdot \phi_j, \nabla \cdot \phi_i) - (B^T M_p^{-1} M_p P)_{ij} \\ &= (\nabla \cdot \phi_j, \nabla \cdot \phi_i) - (B^T P)_{ij} \\ &= (\nabla \cdot \phi_j, \nabla \cdot \phi_i) - \sum_{k=1}^{n_p} (B^T)_{ik} P_{k,j} \\ &= (\nabla \cdot \phi_j, \nabla \cdot \phi_i) - \sum_{k=1}^{n_p} (P_{k,j} (\psi_k, \nabla \cdot \phi_i)) \\ &= (\nabla \cdot \phi_j, \nabla \cdot \phi_i) - (\pi(\nabla \cdot \phi_j), \nabla \cdot \phi_i) \\ &= (\nabla \cdot \phi_j - \pi(\nabla \cdot \phi_j), \nabla \cdot \phi_i) \\ &= (\nabla \cdot \phi_j - \pi(\nabla \cdot \phi_j), \nabla \cdot \phi_i - \pi(\nabla \cdot \phi_i)) \\ &= (\kappa(\nabla \cdot \phi_j), \kappa(\nabla \cdot \phi_i)), \end{aligned}$$

da

$$(\nabla \cdot \phi_j - \pi(\nabla \cdot \phi_j), \pi(\nabla \cdot \phi_i)) = 0$$

wegen (3.13). \square

Damit ist bewiesen, dass die algebraische Manipulation $B^T M_p^{-1} B$ auch durch die Grad-Div-Stabilisierung erreicht wird. Außerdem wird deutlich, dass sich die Grad-Div-Stabilisierung als Projektionsstabilisierung darstellen lässt und nicht alle Skalen stabilisiert.

Sei nun u, p eine kontinuierliche Lösung und u_h, p_h eine diskrete Lösung des Grad-Div-stabilisierten Systems. Zur Lösung u_h gehöre der Vektor U durch

$$u_h = \sum_{k=1}^{n_u} U_k \phi_k,$$

analog V zu v_h . Mit $\|\cdot\|_{\mathbb{R}}$ bezeichnen wir die euklidische Norm des \mathbb{R}^{n_u} für einen Koeffizientenvektor U und mit $\|\cdot\|_0$ die L^2 -Norm.

Damit können wir untersuchen wie sich die Differenz zwischen der Grad-Div-Stabilisierung und $B^T M_p^{-1} B$ für h gegen 0 verhält:

$$\begin{aligned} \|(S_{GD} - B^T M_p^{-1} B) U\|_{\mathbb{R}}^2 &= \sup_{\|V\|_{\mathbb{R}}=1} V^T (S_{GD} - B^T M_p^{-1} B) U \\ &= \sup_{\|V\|_{\mathbb{R}}=1} (\kappa(\nabla \cdot v_h), \kappa(\nabla \cdot u_h)) \\ &= \sup_{\|V\|_{\mathbb{R}}=1} (\kappa(\nabla \cdot v_h), \kappa(\nabla \cdot (u_h - u))) \\ &\leq \sup_{\|V\|_{\mathbb{R}}=1} \|\kappa(\nabla \cdot v_h)\|_0 \|\kappa(\nabla \cdot (u_h - u))\|_0 \\ &\leq \|\kappa\|^2 \sup_{\|V\|_{\mathbb{R}}=1} \|\nabla \cdot v_h\|_0 \|\nabla \cdot (u_h - u)\|_0 \\ &\leq \underbrace{\|\kappa\|^2}_{=:T_1} \underbrace{\sup_{\|V\|_{\mathbb{R}}=1} \|\nabla \cdot v_h\|_0}_{=:T_2} \underbrace{\|\nabla \cdot (u_h - u)\|_0}_{=:T_3}. \end{aligned}$$

Der Fluktuationsoperator κ ist stetig, damit gilt $T_1 \leq C_1$. Für T_2 rechnen wir mit der inversen Ungleichung (Lemma 2.6):

$$T_2 = \sup_{v_h} \frac{\|\nabla \cdot v_h\|_0}{\|V\|_{\mathbb{R}}} \leq C' \sup_{v_h} \frac{h^{-1} \|v_h\|_0}{\|V\|_{\mathbb{R}}}.$$

Nach [KA00], Satz 3.43 gilt mit $C > 0$ unabhängig von h

$$\|V\|_{\mathbb{R}} \geq C h^{-d/2} \|v_h\|_0$$

und damit

$$T_2 \leq C'' \sup_{v_h} h^{-1+d/2} \frac{\|v_h\|_0}{\|v_h\|_0} = C'' h^{-1+d/2} \leq C''' \text{ für } h \rightarrow 0,$$

da $d \geq 2$. Mit der a-priori-Fehlerabschätzung (2.16) für das Grad-Div-stabilisierte System gilt für hinreichend glatte u, p :

$$T_3^2 = \|\nabla \cdot (u_h - u)\|_0^2 \leq C_1 h^{2k} \|u\|_{k+1}^2 + C_2 h^{2k} \|p\|_k^2 \longrightarrow 0 \quad \text{für } h \rightarrow 0.$$

Insgesamt folgt dadurch

$$\|(S_{GD} - B^T M_p^{-1} B) U\|_{\mathbb{R}}^2 \longrightarrow 0 \quad \text{für } h \rightarrow 0.$$

Demnach verschwindet der stabilisierende Term in der Grad-Div-Stabilisierung für h gegen 0 und der Grad-Div-Vorkonditionierer und der Olshanskii-Vorkonditionierer sind asymptotisch gesehen identisch.

Es bleibt durch Rechnungen zu überprüfen, ob der GD-Vorkonditionierer die h - und ν -Unabhängigkeit vom Olshanskii-Vorkonditionierer in der Praxis teilt. Dies werden wir in Kapitel 4 bestätigen können.

Kapitel 4

Numerische Ergebnisse

In diesem Kapitel werden die numerischen Rechnungen vorgestellt, die im Zusammenhang mit dieser Arbeit erstellt wurden. Dazu werden zunächst einige Details zur Implementierung betrachtet und die Testprobleme definiert.

Neben dem Vergleich der Vorkonditionierer aus Abschnitt 3.3 steht die Analyse des GD-Vorkonditionierers im Vordergrund.

4.1 Implementierung und verwendete Bibliotheken

In Zusammenarbeit mit anderen Diplomanden ist eine Bibliothek zur Lösung von Oseen- und Navier-Stokes-Gleichungen entwickelt worden. Die Bibliothek ist in C++ entwickelt und baut auf der Finite-Elemente-Bibliothek `deal.II` (siehe [BHK07] und [BHK]) auf, die neben finiten Elementen unter anderem auch Routinen zum Laden von Gittern und Speichern von Lösungen und Routinen der linearen Algebra enthält.

Mit der von uns entwickelten Bibliothek ist es möglich, verschiedene Stabilisierungsverfahren und Löser in zwei und drei Dimensionen zu vergleichen. Besonders praktisch ist, dass sich die Auswahl und das Parameterdesign für Stabilisierung und Vorkonditionierer ohne Neukompilierung über Parameterdateien steuern lässt.

Auf den Objekten von Vektoren, dünn besetzten Matrizen und iterativen Lösern, die bereits von `deal.II` zur Verfügung gestellt wurden, ist ein Objekt-Orientiertes Framework für die Vorkonditionierer implementiert worden. So kann man Löser und Vorkonditionierer für Teilprobleme beliebig kombinieren und wiederverwenden. Zum Beispiel lässt sich der Löser für den A -Block per Parameterdatei auswählen, konfigurieren und selbst unterschiedlich vorkonditionieren.

Noch zu erwähnen ist die Art und Weise, wie Dirichletrandbedingungen in `deal.II` behandelt werden. Das System wird zunächst komplett mit Randknoten assembliert. Danach wird für jeden Freiheitsgrad, der durch die Randbedingungen fixiert wird, die Systemmatrix bearbeitet. Die zugehörige Zeile wird auf Null und auf die Diagonale ein positiver Eintrag α gesetzt. In

der rechten Seite wird in der gleichen Zeile der durch die Randbedingung vorgegebene Wert multipliziert mit α eingetragen, so dass der Freiheitsgrad nach der Lösung des Systems den korrekten Wert annimmt. Zusätzlich kann noch die zugehörige Spalte durch Gleichungsumformungen eliminiert werden, um die Kopplung dieses Freiheitsgrades mit anderen aufzuheben. Dadurch erhält man die Symmetrie der Systemmatrix, zum Beispiel zwischen B und B^T .

4.2 Testprobleme

4.2.1 Problem 1: glattes Oseen-Problem

Dieses Oseen-Problem ist auf dem zweidimensionalen Gebiet $\Omega = (0, 1)^2$ definiert und stammt aus [GLOS05], Beispiel 1. Die rechte Seite f wird zur glatten Referenzlösung

$$\begin{aligned} u &= (\sin(\pi x), -\pi y \cos(\pi x))^T, \\ p &= \sin(\pi x) \cos(\pi y) \end{aligned}$$

mit Konvektion $b = u$ ausgerechnet:

$$f = \begin{pmatrix} \nu\pi^2 \sin(\pi x) + \pi \sin(\pi x) \cos(\pi x) + c \sin(\pi x) + \pi \cos(\pi x) \cos(\pi y) \\ -\nu\pi^3 y \cos(\pi x) + \pi^2 y \sin^2(\pi x) + \pi^2 y \cos^2(\pi x) + -c\pi y \cos(\pi x) - \pi \sin(\pi x) \sin(\pi y) \end{pmatrix}.$$

Aus zwei Gründen wurde als Konvektionsrichtung die Lösung u gewählt. Einerseits wird dadurch die Lösung eines Navier-Stokes-Problems nachempfunden, bei dem in der letzten Fixpunktiteration $b = u$ gilt. Andererseits sind Probleme mit konstanter Konvektionsrichtung im Vergleich zu diesem Problem unrealistisch. So arbeitet der PCD-Vorkonditionierer zum Beispiel für Probleme mit konstanter Konvektionsrichtung effektiver als bei Problemen mit komplizierterer Strömung, siehe Beispiel 1 in [OB06].

Durch die gegebene Referenzlösung lässt sich die Genauigkeit der Diskretisierung und der Löser analysieren. Die Glattheit ermöglicht auch gut den Einsatz höherer Elementordnungen, die optimale Ordnungen in der Konvergenz liefern.

4.2.2 Problem 2: Lid-Driven Cavity

Die Lid-Driven Cavity ist ein stationäres Navier-Stokes-Problem, welches sich zweidimensional und auch dreidimensional auf $\Omega = (0, 1)^d$ formulieren lässt.

In diesem weit verbreiteten Rechenbeispiel wird die Strömung nur durch eine tangential verlaufende an einer der Seitenflächen bzw. Kanten anliegende Strömung angetrieben, die äußeren Kräfte sind Null ($f = 0$). Die Dirichlet-Randwerte lauten im dreidimensionalen Fall

$$u = \begin{cases} (1, 0, 0)^T & \text{für } (x, y, z) \in (0, 1) \times (0, 1) \times \{1\}, \\ (0, 0, 0)^T & \text{sonst.} \end{cases}$$

Im Zweidimensionalen wird einfach die y -Komponente weggelassen. Diese Cavity lässt sich "dicht" oder "undicht" definieren, je nachdem, ob die tangentiale Strömung auch auf den Kanten, also für $y \in \{0, 1\}$ oder $z \in \{0, 1\}$, gilt.

Trotz der simplen Problemstellung sind die Lösungen nicht trivial. Durch die Randbedingung bildet sich im Gebiet ein großer rotierender Wirbel. Abhängig von der Reynoldszahl bilden sich in den unteren Ecken weitere gegenläufige Wirbel. Im dreidimensionalen Fall gibt es noch zusätzliche Wirbel in der yz -Ebene.

Stationäre Lösungen gibt es in beiden Fällen nur bis zu einer gewissen kritischen Reynoldszahl, die im zweidimensionalen Fall bei ungefähr 8000 liegt, im dreidimensionalen bereits bei 1000, siehe zum Beispiel [PA95].

4.2.3 Problem 3: Beltrami-Flow

Das dritte Beispiel ist wieder ein Oseen-Problem mit Referenzlösung wie Problem 1, aber diesmal auf dem dreidimensionalen Gebiet $\Omega = (-1, 1)^3$ definiert. Mit einem Parameter $a \in \mathbb{R}$ gelte:

$$u = -a \begin{pmatrix} e^{ax} \sin(ay \pm dz) + e^{az} \cos(ax \pm dy) \\ e^{ay} \sin(az \pm dx) + e^{ax} \cos(ay \pm dz) \\ e^{az} \sin(ax \pm dy) + e^{ay} \cos(az \pm dx) \end{pmatrix}$$

und

$$p = -\frac{a^2}{2} [e^{2ax} + e^{2ay} + e^{2az} + 2 \sin(ax \pm dy) \cos(az \pm dx) e^{a(y+z)} + 2 \sin(ay \pm dz) \cos(ax \pm dy) e^{a(z+x)} + 2 \sin(az \pm dx) \cos(ay \pm dz) e^{a(x+y)}]$$

mit $f = 0$.

4.3 Implementierungsdetails zu den Vorkonditionierern

In allen nachfolgenden Rechnungen wird das FGMRES-Verfahren mit der Startlösung $x_0 = 0$ gestartet. Das Abbruchkriterium ist relativ zum Startresiduum

$$\|Ax_0 - b\|$$

gewählt. Wenn nicht anders angegeben, wird bis zu einer Genauigkeit von $TOL := 10^{-10}$ gelöst, also bis

$$r_m \leq TOL \|Ax_0 - b\| = TOL \|b\|$$

gilt. Die Restartlänge vom FGMRES-Verfahren wurde auf 100 gesetzt.

Bei der Implementierung der Vorkonditionierer wurde stärker auf die korrekte Funktionsweise als auf die tatsächliche Laufzeit geachtet. Der Vergleich der Laufzeiten der unterschiedlichen Löser macht in dieser Arbeit außerdem keinen Sinn, da kein Mehrgitter-Löser für den A -Block zur Verfügung stand und diese Lösung den Rechenaufwand dominierte.

Für den A -Block wurde ein in deal.II eingebundenes, direktes Lösungsverfahren verwendet, siehe [Dav04]. Zum Vergleich wurde an manchen Stellen auch die komplette Systemmatrix mit

diesem direkten Verfahren gelöst. Das Hauptproblem dieses Verfahrens für den A -Block ist, dass dafür sehr viel Speicher benötigt wird, wodurch große und vor allem dreidimensionale Rechnungen nicht möglich waren.

Beim WS-Vorkonditionierer wird der A -Block direkt gelöst und der Parameter $\omega = 1$ gewählt.

Für PCD-Vorkonditionierer werden die Hilfsprobleme M_p , F_p , A_p jeweils als Matrix assembliert. Das Konvektions-Diffusions-Problem F_p wurde nicht stabilisiert, was für kleine ν problematisch ist, wie wir in den Rechnungen in Abschnitt 3.3.2 bereits gesehen haben. Das Gleichungssystem für die Massematrix M_p wird genauso wie der A -Block und das A_p -Problem direkt gelöst. Da die Matrix A_p ohne Randbedingungen singular ist, wird das System mit einem zusätzlichen Term $\alpha(p, q)$ mit $\alpha = 10^{-8}$ regularisiert.

Der Olshanskii-Vorkonditionierer verwendet $W^{-1} = M_p^{-1}$ mit einem direkten Löser. Da sich A_γ nicht assemblieren lässt, weil die Matrix dicht besetzt werden würde, wird A_γ iterativ mit einem zusätzlichen GMRES-Verfahren gelöst. In jedem Iterationsschritt wird dafür $A_\gamma v = Av + \gamma B_S^T M_p^{-1} Bv$ ausgerechnet, wobei eine Lösung mit dem direkten Löser für M_p^{-1} nötig ist. Die Laufzeit lässt sich wahrscheinlich erheblich durch die Verwendung einer Näherung von M_p^{-1} zum Beispiel durch Zeilenlumping beschleunigen, wie auch in [OB06] vorgeschlagen.

Da beim GD-Vorkonditionierer die Matrix A mit Stabilisierung assembliert werden kann, lässt sich dafür dann der direkte Löser verwendet. Auch die Massematrix für die Näherung des Schurkomplements wird direkt gelöst, wenn nicht anders beschrieben.

4.4 Numerische Tests

Die numerischen Tests sind in drei Abschnitte unterteilt. Zuerst wird der GD-Vorkonditionierer analysiert, ob die h - und ν -Unabhängigkeit der FGMRES-Iterationszahlen auch in der Praxis zu bemerken ist und um die Bedeutung der Grad-Div-Stabilisierung herauszufinden. Danach werden die Vorkonditionierer aus Abschnitt 3.3 miteinander verglichen und auf h - und ν -Abhängigkeit untersucht.

Zuletzt gibt es noch einige Anwendungen des GD-Vorkonditionierers. Neben einem Navier-Stokes-Problem und einem 3D-Problem wird die Auswirkung von Stabilisierung auf die Lösergeschwindigkeit analysiert.

4.4.1 Grad-Div-Vorkonditionierer

Die erste Rechnung vergleicht die nötigen FGMRES-Iterationen für drei verschiedene Viskositäten in Abhängigkeit des Parameters der Grad-Div-Stabilisierung γ . In Abbildung 4.1 sind die notwendigen Iterationszahlen über der Wahl von γ in doppelt-logarithmischer Darstellung aufgetragen. Gerechnet wurde Problem 1 mit $Q_2 - Q_1$ -Elementen und einer Gitterfeinheit von $h = 1/64$. Andere Gittergrößen wurden auch gerechnet, liegen aber nazu deckungsgleich übereinander und sind deshalb nicht dargestellt.

Für $\nu = 1$ ist die Kurve relativ flach und selbst ohne Stabilisierung mit 19 Iterationen verwendbar. Das liegt daran, dass die Massematrix im Vorkonditionierer im Stokes-Fall eine gute

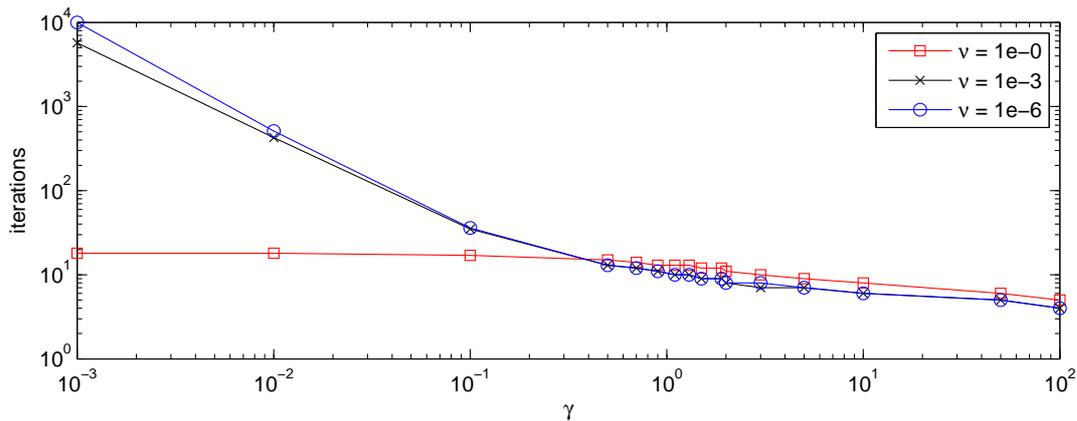


Abbildung 4.1: Iterationszahlen in Abhängigkeit von der Grad-Div-Stabilisierung, $h = 1/64$

Approximation an das Schurkomplement ist. Bereits für $\nu = 10^{-3}$ ist der Vorkonditionierer für $\gamma < 0.1$ schlecht einsetzbar. Wenn aber $\gamma \in \mathcal{O}(1)$ gewählt wird, sind die Iterationszahlen unabhängig von ν in der Größenordnung von 15 Iterationen. Für noch größere γ fallen die Iterationszahlen zwar noch leicht, weshalb aber trotzdem $\gamma \sim 1$ optimal ist, zeigt die nächste Rechnung.

In Abbildung 4.1 haben wir gesehen, dass die FGMRES-Iterationen mit wachsendem γ monoton abnehmen. Die Lösung des A_γ -Blocks im Vorkonditionierer wird dadurch aber schwieriger, da der Block von dem algebraischen Teil mit großem Kern dominiert wird. Diese Auswirkung ist exemplarisch in Abbildung 4.2 verdeutlicht. In jeder Grafik ist in Abhängigkeit von γ die Zahl der äußeren FGMRES-Iterationen, die durchschnittliche Zahl der GMRES-Iterationen für den A -Block und die Gesamtzahl der GMRES-Iterationen für den A -Block dargestellt. Von oben nach unten ist Problem 1 mit $\nu \in \{10^0, 10^{-3}, 10^{-6}\}$ gerechnet.

Die Gesamtzahl der GMRES-Iterationen für den A -Block ist eine Schätzung für den Gesamtaufwand der Lösung des Systems. Für $\nu = 1$ sind die Kurven relativ flach, das heißt die Parameterwahl für die Stabilisierung ist wenig sensibel. Der Gesamtaufwand bleibt nahezu konstant. Anders ist es für kleine ν . Hier nimmt mit größer werdendem γ die Zahl der äußeren Iteration ab, wie bereits in Abbildung 4.1 verdeutlicht. Mit wachsendem γ wird aber die Zahl der inneren GMRES-Iterationen für den A -Block monoton größer. Dadurch gibt es ein Minimum des Gesamtaufwands ungefähr bei $\gamma = 1$, wie man am Besten bei $\nu = 10^{-6}$ erkennen kann.

Es ist anzumerken, dass die in Abbildung 4.2 dargestellten Ergebnisse nur qualitativ zu werten sind. Wir können zwar dadurch das Ergebnis $\gamma \in \mathcal{O}(1)$ bestätigen, das genaue Optimum hängt aber vom inneren Löser ab und wie empfindlich dieser gegenüber der Stabilisierung ist. Wie bereits erwähnt ist aber das verwendete GMRES-Verfahren mit SSOR-Vorkonditionierung nicht optimal hierfür geeignet.

Diese Vergleichsrechnungen wurden auch für verschiedene h durchgeführt, lieferten aber nahezu deckungsgleiche Ergebnisse und sind deswegen hier nicht enthalten.

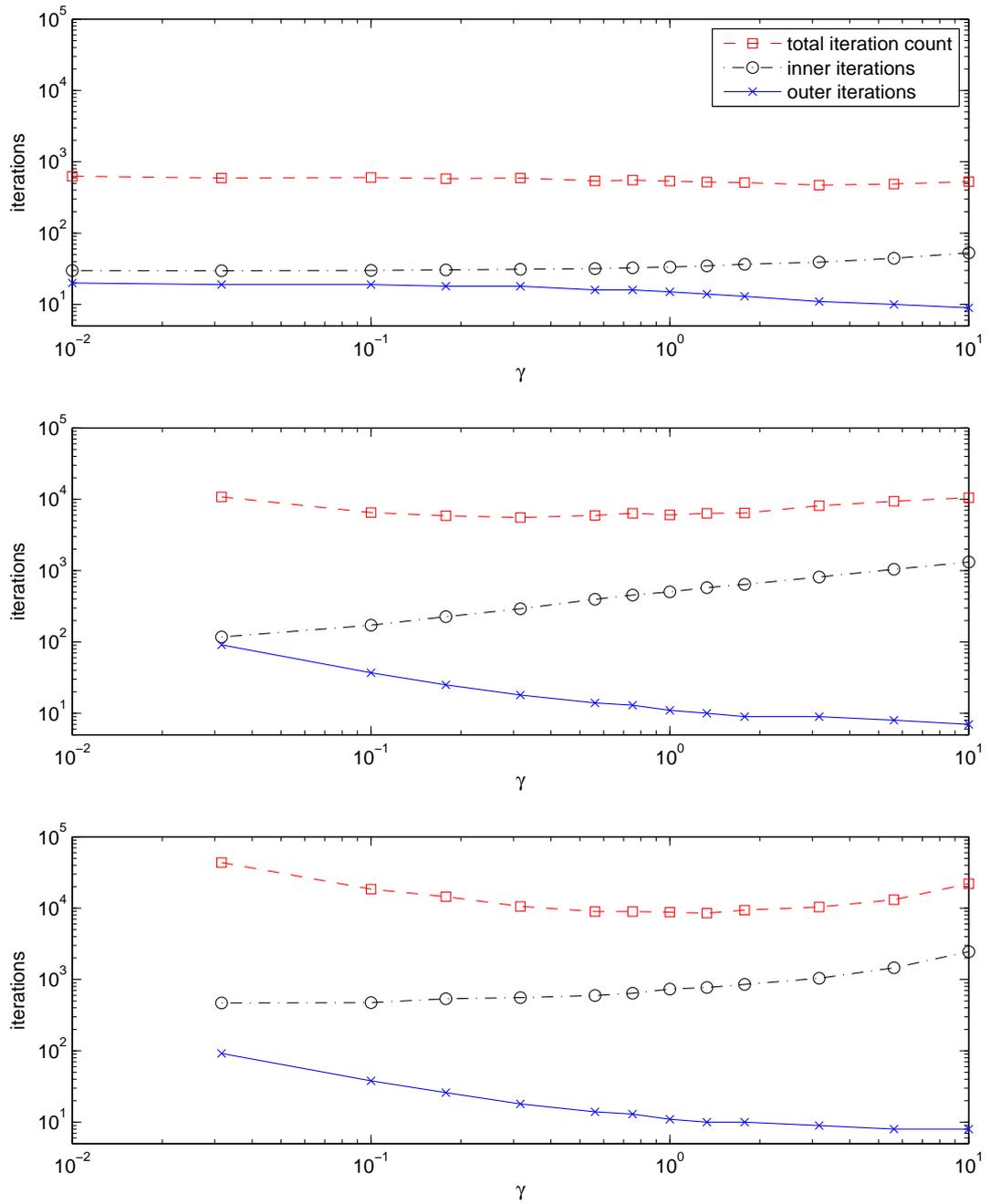


Abbildung 4.2: Vergleich: innerer, äußerer und Gesamtaufwand in Abhängigkeit von γ für $\nu \in \{10^0, 10^{-3}, 10^{-6}\}$, $h = 1/32$

1/h	ν						
	10^0	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
4	16	12	11	11	11	11	11
8	16	12	11	11	11	11	11
16	16	12	11	11	11	11	11
32	16	11	11	11	11	11	11
64	14	11	10	10	10	10	10
128	12	10	10	10	10	10	10
256	12	10	9	9	9	9	9

 Tabelle 4.1: h - and ν -Abhängigkeit

Da wir rechnerisch die Wahl $\gamma \in \mathcal{O}(1)$ bestätigt haben, können wir nun für $\gamma = 1$ die h - und ν -Abhängigkeit des GD-Vorkonditionierers untersuchen. In Tabelle 4.1 sind die FGMRES-Iterationszahlen für Problem 1 mit $Q_2 - Q_1$ -Elementen untersucht. Die Unabhängigkeit von der Viskosität und der Gitterfeinheit ist daraus deutlich zu erkennen. Die nötigen Iterationen nehmen für feinere Gitter sogar leicht ab.

Bei jeder Anwendung der GD-Vorkonditionierers müssen Systeme mit A und M_p gelöst werden. Interessanterweise ist es nicht notwendig, diese inneren Gleichungssysteme exakt zu lösen, wie Tabelle 4.2 verdeutlicht. Hier wurde Problem 1 mit GD-Vorkonditionierer mit Gitterfeinheit $h = 1/32$ und $Q_2 - Q_1$ -Elementen gerechnet. Das Zielresiduum wurde auf $TOL = 10^{-14}$ gesetzt, um einen deutlicheren Unterschied bei den Iterationszahlen zu erkennen. Für die Probleme A und M_p wurden GMRES-Verfahren mit unabhängig voneinander gewähltem Zielresiduum verwendet.

Unabhängig von ν ist die äußere Iterationszahl bereits ab einer inneren Lösergenauigkeit von 10^{-3} nahezu optimal. Dies motiviert ungenaue Lösungsverfahren wie einzelne Mehrgitter V-Zykel für den A -Block oder Zeilenlumping für M_p zu verwenden, um die Lösungsgeschwindigkeit zu erhöhen. In Abbildung 4.3 ist der Verlauf des FGMRES-Residuums von Problem 1 mit GD-Vorkonditionierung mit exaktem Löser für den A - und M_p -Block dargestellt.

In Tabelle 4.3 ist ein Vergleich der Vorkonditionierer für den A -Block im GD-Vorkonditionierer zu sehen, wenn der A -Block auch iterativ gelöst wird. Es handelt sich um Problem 1 mit $Q_2 - Q_1$ -Elementen und $h = 1/64$. Das GMRES-Residuum ist in allen Fällen auf 10^{-4} relativ zum Startresiduum gesetzt. Es ist jeweils die Gesamtzahl der notwendigen inneren Iterationen und die Gesamtzeit für verschiedene Viskositäten dargestellt. Das Verfahren wurde abgebrochen, falls in einer äußeren Iteration mehr als 10000 GMRES-Schritte notwendig waren. Man erkennt, dass neben dem direkten Löser nur SSOR-Vorkonditionierung überhaupt funktioniert. Diese Tabelle verdeutlicht, wie wichtig ein zuverlässiger Löser für den A -Block für große Probleme ist, bei denen der direkte Löser nicht mehr arbeitet.

4.4.2 Vergleichsrechnungen

Mit dem notwendigen Wissen über den GD-Vorkonditionierer können wir nun den Vergleich der verschiedenen Vorkonditionierer aus Abschnitt 3.3 durchführen.

$\nu = 10^0$		res(M_p)					
res(A_γ)	0.1	0.01	0.001	10^{-5}	10^{-9}	direkt	
0.1	91	60	56	56	56	56	
0.01	35	32	32	33	33	33	
0.001	25	23	23	23	23	23	
10^{-5}	23	21	21	21	21	21	
10^{-9}	23	21	21	21	21	21	
direkt	23	21	21	21	21	21	

$\nu = 10^{-3}$		res(M_p)					
res(A_γ)	0.1	0.01	0.001	10^{-5}	10^{-9}	direkt	
0.1	168	47	41	45	45	45	
0.01	25	20	20	20	20	20	
0.001	17	16	16	16	16	16	
10^{-5}	16	15	15	15	15	15	
10^{-9}	16	15	15	15	15	15	
direkt	16	15	15	15	15	15	

$\nu = 10^{-6}$		res(M_p)					
res(A_γ)	0.1	0.01	0.001	10^{-5}	10^{-9}	direkt	
0.1	200	65	65	55	53	54	
0.01	24	21	20	20	20	20	
0.001	17	16	16	16	16	16	
10^{-5}	16	15	15	15	15	15	
10^{-9}	16	15	15	15	15	15	
direkt	16	15	15	15	15	15	

Tabelle 4.2: Einfluss der inneren Lösergenauigkeit, $h = 1/32$

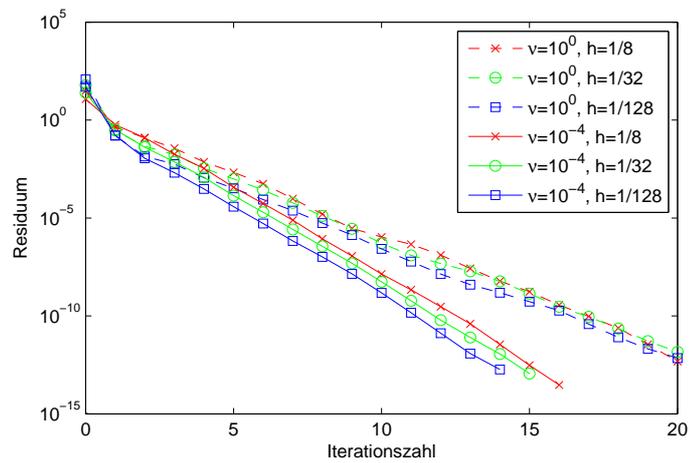


Abbildung 4.3: Residuenverlauf für verschiedene h und ν

ν	Direkt	SSOR		ILU0		Id	
	Zeit	It.	Zeit	It.	Zeit	It.	Zeit
10^0	3s	133	114s	131	125s	417	236s
10^{-3}	3s	2726	1953s	fail		6744	2861s
10^{-6}	3s	5166	4148s	fail		fail	

 Tabelle 4.3: Löser für den A -Block, $h = 1/64$

In Tabelle 4.4 findet sich eine Vergleichsrechnung der Vorkonditionierer von Problem 1 mit $h = 1/32$ für verschiedene ν . Die inneren Probleme sind jeweils exakt gelöst, um einen Eindruck über die notwendigen Iterationszahlen zu erhalten. Stabilisierung ist nur beim GD-Vorkonditionierer aktiviert.

Die Löserzeiten für den Olshanskii-Vorkonditionierer sind so hoch, da auch das A_γ -System iterativ bis zu einer hohen Genauigkeit gelöst wird. Unabhängigkeit der Iterationszahlen von ν sind nur beim GD- und Olshanskii-Vorkonditionierer zu sehen. Letzterer scheitert für kleine ν am Löser für den A_γ -Block, da dieser nicht vorkonditioniert ist.

Mit einer effizienteren Implementierung des Olshanskii-Vorkonditionierers wäre dieser wahrscheinlich gut einzusetzen gewesen. Für große ν sind alle getesteten Vorkonditionierer gut einsetzbar und arbeiten wesentlich besser als ein nicht vorkonditioniertes GMRES-Verfahren. Für kleine ν funktionieren nur der Olshanskii- und der Grad-Div-Vorkonditionierer. Letzterer liefert ν -unabhängige Iterationszahlen und Löserzeiten und schlägt damit alle anderen implementierten Verfahren.

4.4.3 Anwendung

Wir betrachten nun einige weitere Rechnungen, die mit dem GD-Vorkonditionierer durchgeführt wurden.

Um zu bestätigen, dass der Vorkonditionierer unabhängig von der verwendeten Elementordnung funktioniert, findet sich in Tabelle 4.5 eine Vergleichsrechnung. Die nötigen Iterationszahlen sind für höhere Elementordnungen exakt dieselben. Es wurde mit $\gamma = 1$ und $h = 1/16$ das Problem 1 mit unterschiedlichen Elementen gelöst.

	$\nu = 10^0$		$\nu = 10^{-2}$		$\nu = 10^{-3}$		$\nu = 10^{-4}$		$\nu = 10^{-6}$	
	Zeit	It.	Zeit	It.	Zeit	It.	Zeit	It.	Zeit	It.
GD	0.70s	14	0.49s	11	0.50s	11	0.61s	11	0.61s	11
ID	85.28s	4378	fail		fail		fail		fail	
WS	1.47s	61	19.19s	787	68.08s	2367	348.33s	12051	fail	
Olsh	84.58s	14	2386.69s	10	6509.23s	10	fail		fail	
PCD	0.69s	20	1.54s	53	fail		fail		fail	

 Tabelle 4.4: Vergleich der Vorkonditionierer, $h = 1/32$

Elementpaar	ν		
	10^0	10^{-3}	10^{-6}
$Q_2 - Q_1$	14	11	11
$Q_3 - Q_2$	14	11	11
$Q_4 - Q_3$	14	11	11
$Q_5 - Q_4$	14	11	11

Tabelle 4.5: Einfluß der Elementordnung auf Iterationszahlen, $h = 1/16$

Methode	γ	Direkt	It.	Zeit	$\ e_u\ _0$	$ e_u _1$	SP-Einträge
Grad-Div	1.00	15.4s	10	12.5s	2.155e-6	1.625e-3	5865314
LPS	0.56	16.6s	12	16.5s	1.237e-6	0.969e-3	8489250
SUPG	0.31	14.3s	28	25.3s	1.289e-6	1.003e-3	5879349
NoStab	0.00	31.8s		fail	1.066e-4	0.939e-1	3751200
Grad-Div	0.56	15.4s	12	13.5s	2.149e-6	1.625e-3	5865314
Grad-Div	0.31	15.4s	16	16.0s	2.564e-6	1.978e-3	5865314

Tabelle 4.6: Auswirkung der Stabilisierung auf GD-Vorkonditionierung (I), $h = 1/128$

Ein wichtiger Aspekt beim Lösen von Oseen-Problemen für kleine Viskositäten ist die Stabilisierung wie in Abschnitt 2.4 beschrieben. Tabelle 4.6 enthält einen Vergleich für $Q_2 - Q_1$ -Elemente auf einem unregelmäßigen Gitter mit $h = 1/128$. In der ersten Zeile ist Grad-Div-Stabilisierung mit $\gamma = 1$ aktiviert, wie es für den GD-Vorkonditionierer optimal ist. Für LPS- und RBS-Stabilisierung wurden jeweils die optimalen Parameter zur Minimierung der Fehler der Lösung gewählt. Die Fehler der Geschwindigkeit in L^2 -Norm und H^1 -Seminorm sind mit dargestellt. In Tabelle 4.7 findet sich ein Überblick für verschiedene h und ν .

Da der Grad-Div-Parameter γ für LPS- und RBS-Stabilisierung kleiner ist, als für den Löser optimal wäre, ist zum Vergleich Grad-Div-Stabilisierung mit genau diesen Parametern zusätzlich dargestellt. Man sieht, dass sich die Iterationszahlen leicht gegenüber $\gamma = 1$ erhöhen.

LPS-Stabilisierung beeinflusst den GD-Vorkonditionierer nicht, wie man auch für verschiedene h und ν in Tabelle 4.7 erkennt. Dies lässt sich damit begründen, dass durch den symmetrischen Aufbau der Stabilisierung die Blockstruktur nicht verletzt wird. Durch die größere Anzahl der Matrixeinträge (in der Spalte "SP-Einträge" dargestellt) erhöht sich allerdings die Löserzeit etwas. Die residuale Stabilisierung hingegen passt wegen $B_S \neq B$ nicht zum GD-Vorkonditionierer, was man an der deutlich höheren Iterationszahl erkennt. In Tabelle 4.7 fällt auf, dass die Iterationszahlen für $h \rightarrow 0$ besser werden. Dies lässt sich damit begründen, dass die SUPG-Stabilisierung mit h^2 skaliert und damit der unsymmetrische Anteil $B_S - B$ für $h \rightarrow 0$ verschwindet.

Um vergleichen zu können, ob sich der GD-Vorkonditionierer im Vergleich zu einem direkten Löser überhaupt lohnt, vergleichen wir die Löserzeiten für $h \rightarrow 0$. In Tabelle 4.8 sind die Zeiten gegenüber gestellt. Für den A -Block vom GD-Vorkonditionierer ist der gleiche direkte Löser verwendet worden, wie der in der Spalte "Direkt" für die ganze Matrix.

Selbst für kleine Probleme ist der GD-Vorkonditionierer genauso schnell wie der direkte Löser.

$\nu = 10^0$	γ	$h = 1/16$			$h = 1/32$			$h = 1/64$			$h = 1/128$		
		It	GD	Direct	It	GD	Direct	It	GD	Direct	It	GD	Direct
Grad-Div	1.00	14	0.11s	0.07s	14	0.70s	0.38s	13	3.22s	2.22s	12	16.24s	15.90s
LPS	0.56	16	0.18s	0.09s	16	0.98s	0.45s	15	4.25s	2.32s	14	18.52s	14.52s
SUPG	0.31	52	0.38s	0.08s	39	1.54s	0.40s	26	5.32s	2.20s	18	25.72s	18.07s
NoStab	0.00	20	0.12s	0.06s	19	0.68s	0.46s	19	3.30s	3.59s	18	14.54s	27.73s
Grad-Div	0.56	16	0.15s	0.08s	16	0.86s	0.42s	15	4.17s	2.41s	14	20.50s	17.57s
Grad-Div	0.31	17	0.17s	0.08s	17	0.94s	0.41s	16	4.36s	2.50s	15	21.67s	17.97s
$\nu = 10^{-3}$		$h = 1/16$			$h = 1/32$			$h = 1/64$			$h = 1/128$		
	γ	It	GD	Direct	It	GD	Direct	It	GD	Direct	It	GD	Direct
Grad-Div	1.00	11	0.11s	0.07s	11	0.59s	0.42s	10	3.00s	2.49s	10	16.82s	17.98s
LPS	0.56	14	0.20s	0.10s	13	0.84s	0.49s	13	4.20s	2.54s	12	19.44s	16.07s
SUPG	0.31	135	1.05s	0.07s	69	2.53s	0.41s	37	7.18s	2.57s	22	26.41s	17.87s
NoStab	0.00	1146	6.92s	0.08s	3784	118.12s	0.48s	6855	950.19s	3.41s	9138	6318.11s	24.75s
Grad-Div	0.56	14	0.14s	0.08s	13	0.65s	0.44s	12	3.34s	2.55s	12	18.48s	17.30s
Grad-Div	0.31	18	0.17s	0.08s	17	0.77s	0.44s	17	3.92s	2.52s	16	20.93s	17.98s
$\nu = 10^{-6}$		$h = 1/16$			$h = 1/32$			$h = 1/64$			$h = 1/128$		
	γ	It	GD	Direct	It	GD	Direct	It	GD	Direct	It	GD	Direct
Grad-Div	1.00	11	0.12s	0.08s	11	0.56s	0.42s	10	2.79s	2.58s	10	16.45s	17.91s
LPS	0.56	14	0.20s	0.10s	13	0.84s	0.50s	13	4.16s	2.61s	12	19.67s	15.96s
SUPG	0.31	134	1.02s	0.08s	70	2.53s	0.42s	37	7.18s	2.56s	22	22.93s	16.33s
NoStab	0.00	3586	19.10s	0.07s	fail	fail	0.44s	fail	fail	3.61s	fail	fail	30.52s
Grad-Div	0.56	14	0.12s	0.08s	13	0.56s	0.37s	12	3.17s	2.28s	12	18.53s	17.83s
Grad-Div	0.31	18	0.16s	0.08s	17	0.76s	0.43s	17	4.08s	2.48s	16	22.16s	18.04s

Tabelle 4.7: Auswirkung der Stabilisierung auf GD-Vorkonditionierung (II)

DOFs:	38k	150k	170k	530k	606k	1000k
GD	2s	12s	14s	65s	90s	151s
Direkt	2s	15s	17s	94s	fail	fail

Tabelle 4.8: Vergleich GD-Vorkondtionierer und direkter Löser

ν	h		
	1/8	1/16	1/32
10^0	28	29	28
10^{-3}	22	21	22
10^{-6}	22	21	fail

Tabelle 4.9: Iterationszahlen für Beltrami-Flow

Bei größeren Problemen wird er sogar schneller, was sich vor allem durch den geringeren Speicherbedarf erklären lässt. Die Zerlegung der gesamten Systemmatrix in Sattelpunktform scheint im direkten Löser deutlich schlechter zu funktionieren, als die des A -Blocks, der nicht viel kleiner ist als das gesamte System. Werden die zu lösenden Probleme noch größer, versagt der direkte Löser, da nicht genug Speicher für die Zerlegung vorhanden ist.

Der GD-Vorkondtionierer funktioniert wie erwartet auch mit dreidimensionalen Problemen, wie Tabelle 4.9 zeigt. Gerechnet wurde Problem 3 mit $a = \pi/4$ für verschiedene ν und h . Da im Dreidimensionalen aufgrund der Problemgröße der direkte Löser nicht mehr verwendbar ist, wurde der A -Block mit einem GMRES-Verfahren mit SSOR-Vorkondtionierung gelöst. Die Lösung scheiterte für $\nu = 10^{-6}$ und $h = 1/32$, da der Löser für den A -Block in 10000 Iterationen nicht konvergierte. Ansonsten ist auch hier die h - und ν -Unabhängigkeit zu erkennen.

4.4.4 Stationäres Navier-Stokes-Problem

Zum Abschluß der Untersuchungen testen wir den GD-Vorkondtionierer für ein nichtlineares Problem: Lid-Driven Cavity.

In Tabelle 4.10 sieht man für unterschiedliche Reynoldszahlen und zwei verschiedene Gittergrößen die notwendigen Picard-Iterationen zur Auflösung der Nicht-Linearität. Das Abbruchkriterium war die Reduktion des nicht-linearen Residuums auf 10^{-8} . Um die Konvergenz zu sichern, wurde die Fixpunktiteration in allen Fällen mit dem Faktor 0.9 gedämpft.

Für größere Reynoldszahlen nimmt die Zahl der Picard-Iterationen deutlich zu. Die FGMRES-Iterationen pro Picard-Iterationen bleiben aber weitestgehend konstant. Genauso ist die FGMRES-Iterationszahl unabhängig von h .

Die Ergebnisse in der dreidimensionalen Variante der Cavity sehen vergleichbar aus, siehe Tabelle 4.11. Hier wurde die Picard-Iteration mit dem Faktor 0.7 gedämpft, während die Reynoldszahl 100 beträgt.

Re	h=1/32			h=1/64		
	Pic.-It	\sum It.	\emptyset It.	Pic.-It	\sum It.	\emptyset It.
100	16	79	4.94	16	78	4.88
1000	30	175	5.83	30	171	5.70
5000	33	198	6.00	41	242	5.90
7500	145	841	5.80	77	458	5.95

Tabelle 4.10: 2D Lid-Driven Cavity mit GD-Vorkondtionierer

h	Pic.-It.	\sum It.	\emptyset It.
1/8	20	218	10.90
1/16	19	207	10.89
1/32	18	210	11.67

Tabelle 4.11: 3D Lid-Driven Cavity mit GD-Vorkondtionierer

4.5 Eigenwertplots

Wenn wir die Matrix M mit P^{-1} vorkonditionieren, wird die Iterationszahl durch die Eigenwertverteilung des Produkts MP^{-1} bestimmt. Die Visualisierung der Eigenwerte kann dadurch Aufschluss über einen Vorkondtionierer liefern.

Die Matrix MP^{-1} liegt nicht explizit vor, da P^{-1} nur als Operator gegeben ist. Daher ist die Berechnung der Eigenwerte nicht ohne weiteres möglich. Mit folgender Technik lässt sich mit einigem Aufwand trotzdem die Matrix MP^{-1} bilden: Wir wenden nacheinander die Einheitsvektoren $x_i \in \mathbb{R}^{n_u+n_p}$ auf den Operator P^{-1} an und erhalten damit eine dicht besetzte Darstellung von P^{-1} . Nach Matrixmultiplikation mit M erhalten wir die Matrix MP^{-1} , von der sich nun die Eigenwerte berechnen und visualisieren lassen.

In Abbildung 4.4 sieht man in der linken Spalte die Eigenwerte der Matrix M für $\nu \in \{10^0, 10^{-1}, 10^{-3}, 10^{-6}\}$. Man sieht deutlich, dass die Eigenwerte nicht geclustert sind und sich um die Null häufen, was schlechte Konvergenzgeschwindigkeit verspricht. Auf der rechten Seite sind die Eigenwerte von MP^{-1} mit dem GD-Vorkondtionierer jeweils mit der gleichen Viskosität wie links dargestellt. Hier sieht man deutlich die Clusterbildung der Eigenwerte um die Eins herum (man beachte die starke Vergrößerung des komplexen Anteils). Ein Eigenwert ist hier exakt Null, dieser entspricht dem Druckfreiheitsgrad, der nicht eindeutig bestimmt ist, siehe dazu Abschnitt 2.5.1. Des Weiteren bleibt die Eigenwertverteilung für unterschiedliche ν erhalten, deswegen ist die Zahl der FGMRES-Iterationen unabhängig von ν .

In Abbildung 4.5 und 4.6 sind die Eigenwerte jeweils links ohne Vorkonditionierung und rechts mit GD-Vorkonditionierung dargestellt. In Abbildung 4.5 werden für $\nu = 1$ in den Spalten die verschiedenen Stabilisierungen verglichen: Grad-Div-Stabilisierung mit $\gamma = 1$, System mit LPS-Stabilisierung, residuale Stabilisierung und am Ende Grad-Div-Stabilisierung mit den Stabilisierungsparametern aus LPS bzw. RBS, also $\gamma = 0.56$ und $\gamma = 0.31$. Abbildung 4.6 zeigt die gleichen Eigenwertverteilungen für $\nu = 10^{-3}$.

Für $\nu = 1$ werden die Eigenwerte durch die GD-Vorkonditionierung wieder um die Eins ge-

clustert, allerdings gibt es eine Menge an Eigenwerten, die zwischen Null und Eins verteilt liegen. Dies erklärt, warum für $\nu = 1$ die Iterationszahlen leicht höher liegen. Für $\nu = 10^{-3}$ (Abbildung 4.6) werden diese Eigenwerte geclustert, was die Iterationszahlen leicht verringert. Das durch lokale Projektion stabilisierte System (2. Zeile) hat zwar leicht andere Eigenwerte, die Struktur des vorkonditionierten System bleibt aber erhalten. Dies steht im Gegensatz zur residualen Stabilisierung, die vor allem für $\nu = 1$ das System stark verändert. Von einer Clusterung der Eigenwerte durch den Vorkonditionierer kann dort nicht gesprochen werden, was sich in den Rechnungen bestätigt, siehe zum Beispiel Tabelle 4.7. Für $\nu = 10^{-3}$ sieht die Eigenwertverteilung des nicht vorkonditionierten, residual stabilisierten Systems (Abbildung 4.6, Zeile 3 links) den anderen recht ähnlich.

In Abbildung 4.7 sind die Eigenwerte des residual stabilisierten Systems noch einmal detailliert für $\nu \in \{10^0, 10^{-2}, 10^{-3}, 10^{-6}\}$ untereinander dargestellt. Im vorkonditionierten Fall (rechte Spalte), sehen die Eigenwerte vergleichbar mit denen in Abbildung 4.4 aus, allerdings ist hier die Verteilung in der komplexen Achse ungefähr um einen Faktor 2 größer, was einer schlechteren Clusterung entspricht.

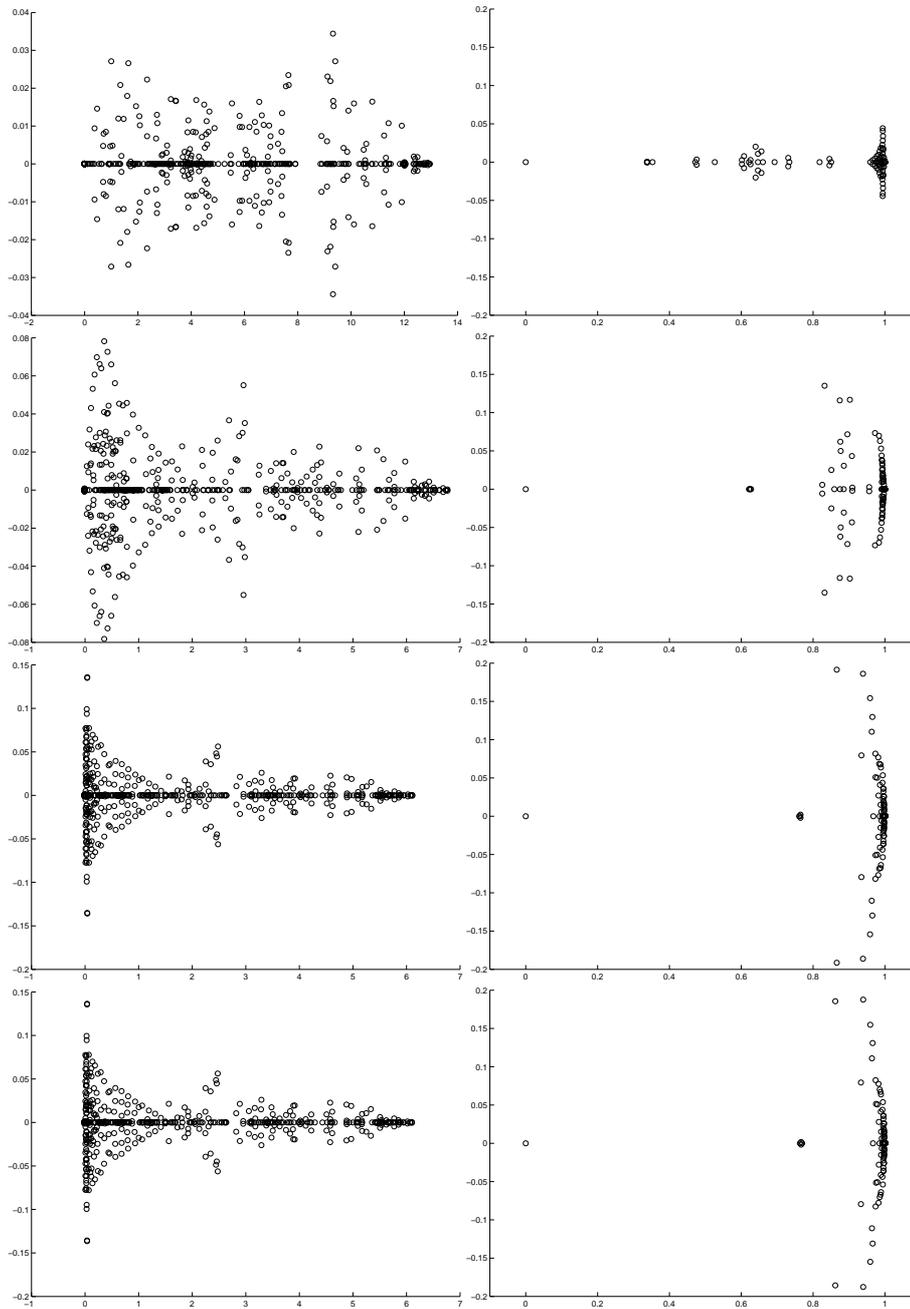


Abbildung 4.4: Eigenwerte: mit Grad-Div-Stabilisierung, ohne/mit GD-Vorkonditionierer für $\nu \in \{10^0, 10^{-1}, 10^{-3}, 10^{-6}\}$

4.5. Eigenwertplots

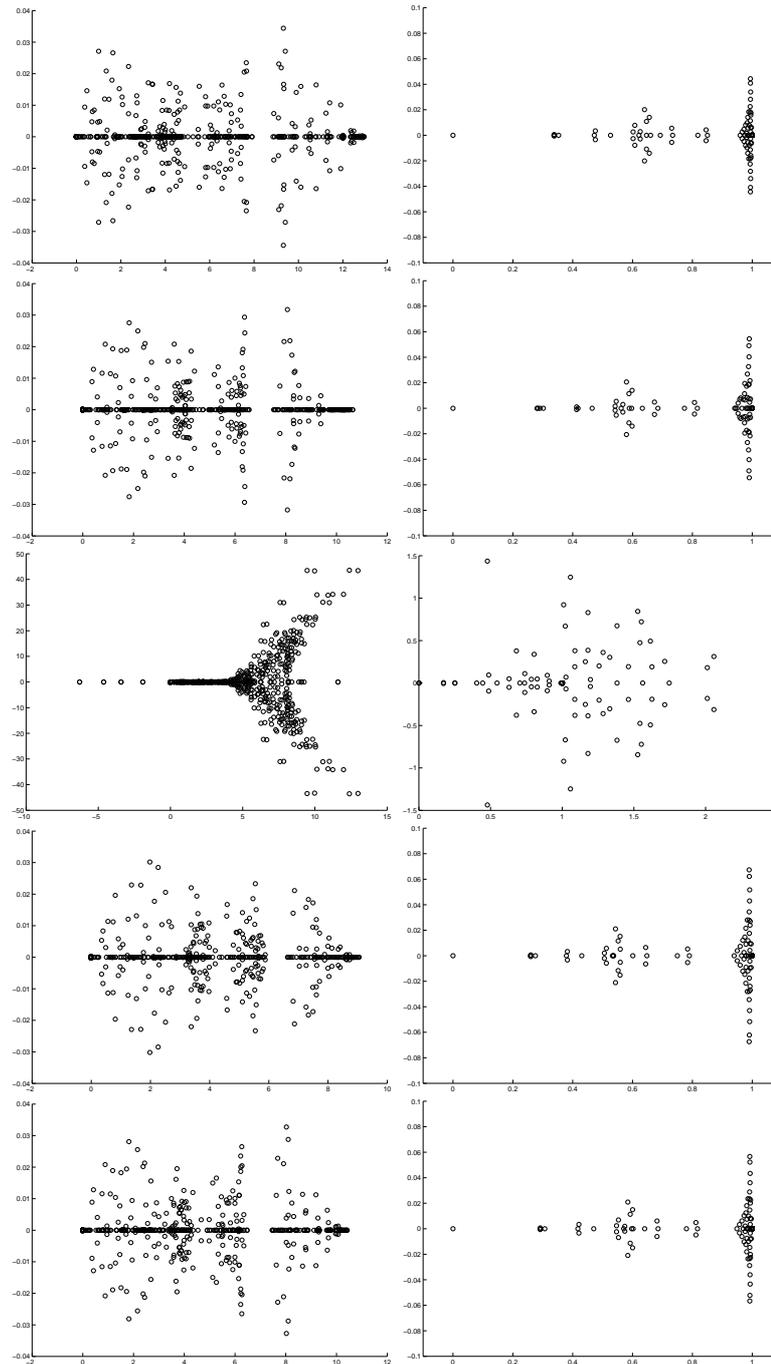


Abbildung 4.5: Eigenwerte: Vergleich von Grad-Div-, LPS-, RBS-Stabilisierung und Grad-Div-Stabilisierung mit $\gamma \in \{0.56, 0.31\}$, $\nu = 1$

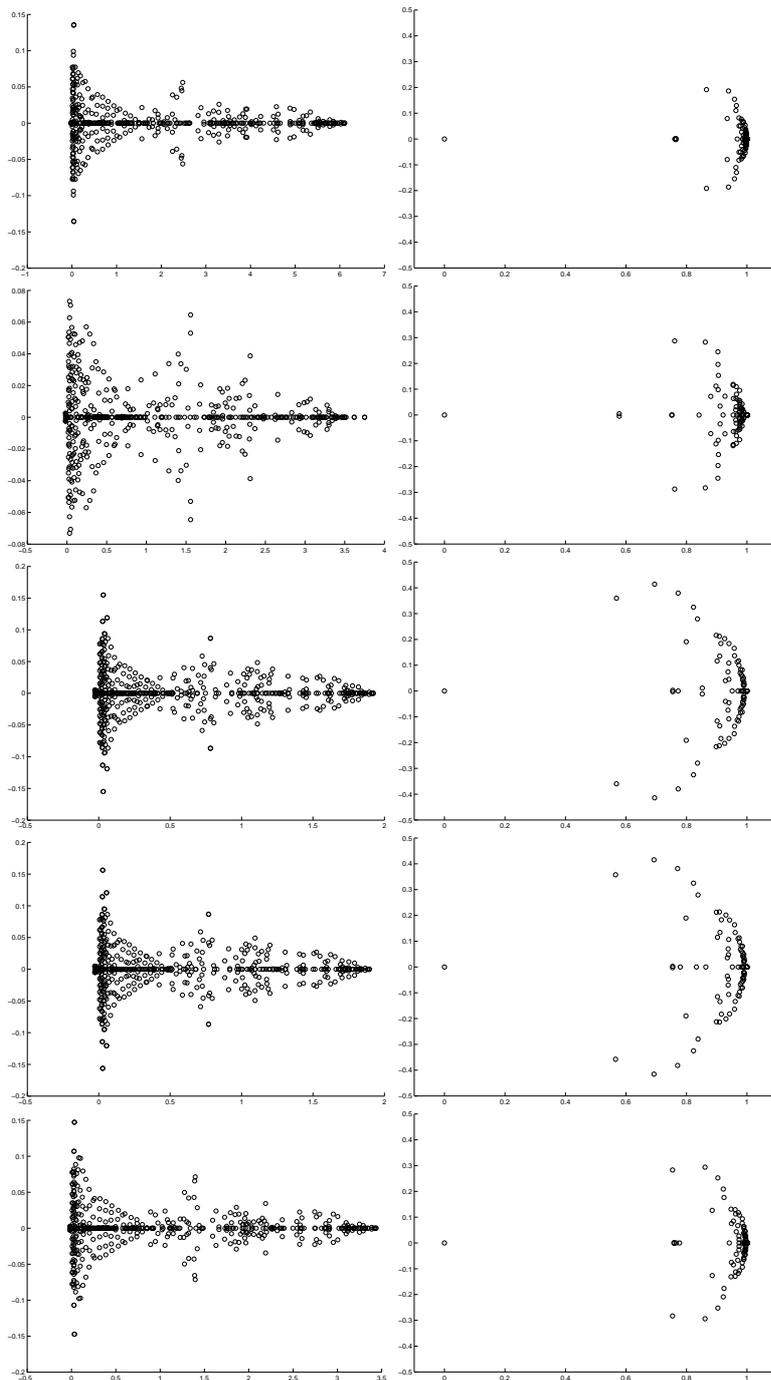


Abbildung 4.6: Eigenwerte: Vergleich von Grad-Div-, LPS-, RBS-Stabilisierung und Grad-Div-Stabilisierung mit $\gamma \in \{0.56, 0.31\}$, $\nu = 10^{-3}$

4.5. Eigenwertplots

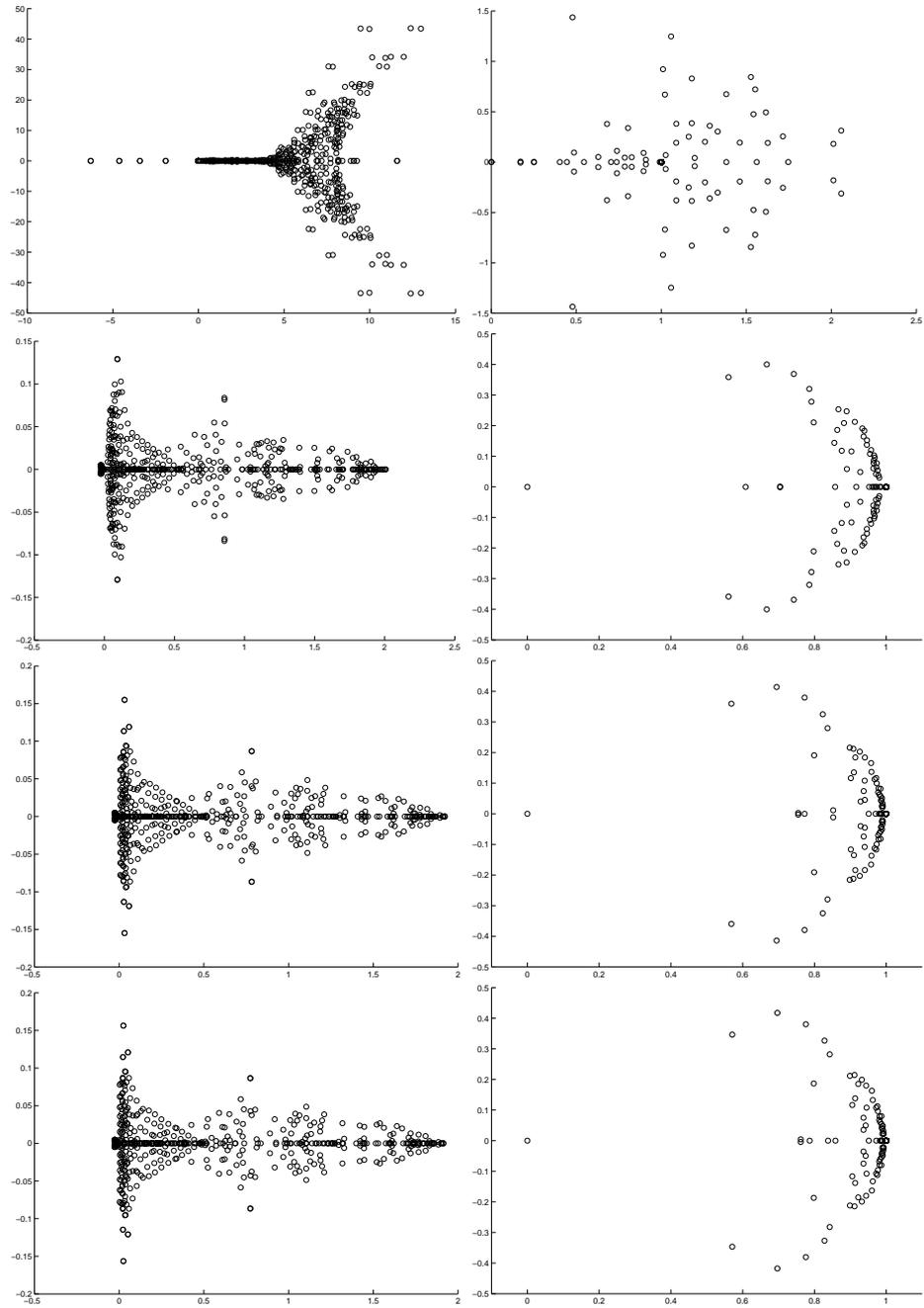


Abbildung 4.7: Eigenwerte: mit residualer Stabilisierung, ohne/mit GD-Vorkonditionierer für $\nu \in \{10^0, 10^{-2}, 10^{-3}, 10^{-6}\}$

Kapitel 5

Fazit

Der Schwerpunkt wurde in dieser Arbeit auf die Betrachtung der Vorkonditionierung bei Taylor-Hood-Elementen gelegt. Dies lässt sich vor allem damit begründen, dass die bei Equal-Order-Elementen notwendige Druckstabilisierung die Gestalt der Systemmatrix durch Einträge im C -Block verändert.

Der in dieser Arbeit vorgestellte Grad-Div-Vorkonditionierer lieferte sehr gute Ergebnisse bei Verwendung von Taylor-Hood-Elementen, insbesondere funktionierte er auch für $\nu \rightarrow 0$ ohne Schwierigkeiten. Die anderen Vorkonditionierer lieferten (ausgenommen der Olshanskii-Vorkonditionierer) alle eine gewisse h - und ν -Abhängigkeit, die eine Benutzung erschwert oder für bestimmte Probleme sogar unmöglich macht, wie die durchgeführten Rechnungen zeigen.

Aufgrund der Kombination von Stabilisierung und Vorkonditionierung beim Grad-Div-Vorkonditionierer war eine theoretische Analyse schwierig, aber durch eine Analyse der Grad-Div-Stabilisierung konnte eine Ähnlichkeit zum Olshanskii-Vorkonditionierer hergestellt werden. So lässt sich die Grad-Div-Stabilisierung in einen algebraischen und einen stabilisierenden Teil aufteilen. Die algebraische Veränderung der Systemmatrix entspricht der des Olshanskii-Vorkonditionierers und zusätzliche Stabilisierung beeinflusst den Vorkonditionierer nicht negativ. Außerdem verschwindet die Stabilisierung für $h \rightarrow 0$.

Der Olshanskii-Vorkonditionierer wurde theoretisch analysiert und die h - und ν -Unabhängigkeit im unstabilisierten Fall bewiesen. Die Rechnungen bestätigen, dass sich diese Eigenschaft auch auf den Grad-Div-Vorkonditionierer überträgt, obwohl das System stabilisiert wird.

Ein großer Vorteil des Grad-Div-Vorkonditionierers liegt darin, dass er auch mit Stabilisierung einsetzbar ist. Die Verwendung von Stabilisierung durch lokale Projektion hatte keinen Einfluß auf die Performanz des Vorkonditionierers. Aufgrund der unsymmetrischen Struktur der residualen Stabilisierung lässt sich diese leider nicht mit dem Grad-Div-Vorkonditionierer verbinden. Aber selbst wenn man nur mit Grad-Div-Stabilisierung rechnet, verbessern sich die Fehler der Lösung gegenüber ausgeschalteter Stabilisierung für kleine ν erheblich.

Parallel zu dieser Diplomarbeit wurde der Grad-Div-Vorkonditionierer mit dem FGMRES-Verfahren als Löser für die linearen Gleichungssysteme bei Verwendung von inexakten, glo-

balisierten Newton-Verfahren für die stationären Navier-Stokes-Gleichungen verwendet, siehe [Lem08]. Bei inexakten Newton-Verfahren wird die Genauigkeit, mit der die Gleichungssysteme gelöst werden, vom Algorithmus variiert. Eine exakte Lösung ist nicht notwendig, da auch eine ungenaue Abstiegsrichtung ausreicht, solange die Genauigkeit mit der Schrittweite gesteuert wird. Dadurch lagen die FGMRES-Iterationszahlen pro Newton-Iterationsschritt meist im einstelligen Bereich.

Ein Lücke bleibt bei der Behandlung des A -Blocks, der auch für andere Vorkonditionierer gelöst werden muss. Doch selbst mit dem einfachen Ansatz, dafür einen direkten Löser zu verwenden, können größere Probleme überhaupt erst und kleinere Probleme schneller als mit dem direkten Löser für das gesamte System gelöst werden.

Die Entwicklung eines Mehrgitterverfahrens für den A -Block wäre wünschenswert gewesen, zum Beispiel wie in [OB06] beschrieben. Diese relativ aufwändige Entwicklung hätte den Rahmen dieser Arbeit gesprengt. Nicht zu vernachlässigen ist in diesem Zusammenhang die Möglichkeit der Parallelisierung, weil sich nur dadurch die Problemgröße hinreichend erweitern lässt. Weiterhin wäre der Vergleich oder die Kombination von Mehrgitter- und Gebietszerlegungsverfahren ratsam.

Ein weiterer interessanter Aspekt ist die direkte Übertragung der theoretischen Ergebnisse des Olshanskii-Vorkonditionierers auf den Grad-Div-Vorkonditionierer, was nicht abschließend behandelt werden konnte. Genauso ist die Kombination von Stabilisierung durch lokale Projektion und Vorkonditionierung noch nicht befriedigend abgeschlossen. In beiden Fällen wäre eine Verallgemeinerung der Resultate des Olshanskii-Vorkonditionierers auf ein stabilisiertes System nötig gewesen.

Trotzdem lässt sich resümieren, dass mit dieser Diplomarbeit ein robuster und für andere Arbeiten gut verwendbarer Löser für das verallgemeinerte Oseen-Problem auch in Verbindung mit Stabilisierung entstanden ist. Des Weiteren sind hierdurch wichtige Grundsteine für die etwaige Entwicklung eines parallel laufenden Löser für das verallgemeinerte Oseen-Problem gelegt worden.

Danksagung

Ich bedanke mich zuallererst bei Herrn Rapin und Herrn Lube für die sehr gute und freundliche Betreuung meiner Arbeit und die Begleitung seit Beginn meines Hauptstudiums. Gerade die immer offen stehende Tür von Herrn Rapins Büro war eine sehr große Hilfe und die angeregten Diskussionen haben viel zur Entstehung dieser Arbeit beigetragen. Danke.

Außerdem möchte ich meinen Eltern danken, dass sie mir mein Studium und damit auch diese Arbeit ermöglicht haben. Danke.

Des Weiteren möchte ich meinen Freunden Benni, Johannes, Lars, Markus und Wiebke für die angenehme Atmosphäre in unserer Arbeitsgruppe danken. Ich denke die Kommunikation in Zusammenhang mit der Programmierung und den Arbeiten und die Ablenkungen außerhalb des Studiums sind nicht nur für mich eine Stütze gewesen. Danke.

Weiterhin danke ich allen, die nicht namentlich genannt sind, aber dennoch direkt oder indirekt zur Vervollständigung der Arbeit beigetragen haben.

Literaturverzeichnis

- [Alt99] H. W. Alt. *Lineare Funktionalanalysis. Eine anwendungsorientierte Einführung*. Springer-Verlag, Berlin, 1999.
- [BBC⁺94] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. SIAM, Philadelphia, PA, 1994.
- [BBJL07] M. Braack, E. Burman, V. John, and G. Lube. Stabilized Finite Element Methods for the Generalized Oseen Problem. *Comp. Meth. Appl. Mech. Engrg.* 196 4-6, pages 853–866, 2007.
- [BF91] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Springer-Verlag, New York, 1991.
- [BGL05] M. Benzi, G. H. Golub, and J. Liesen. Numerical Solution of Saddle Point Problems. *Acta Numerica*, 14:1–137, 2005.
- [BHK] W. Bangerth, R. Hartmann, and G. Kanschat. deal.II *Differential Equations Analysis Library, Technical Reference*. <http://www.dealii.org>.
- [BHK07] W. Bangerth, R. Hartmann, and G. Kanschat. deal.II — a General Purpose Object Oriented Finite Element Library. *ACM Transactions on Mathematical Software*, 33(4):27, August 2007.
- [Bra03] D. Braess. *Finite Elemente: Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer-Verlag, 3., korrigierte und erg. Aufl., 2003.
- [Bre74] F. Brezzi. On the Existence, Uniqueness and Approximation of Saddle-Point Problems arising from Lagrangian Multipliers. *R.A.I.R.O. Anal. Numer.*, 1974.
- [Dav04] T. A. Davis. Algorithm 832: UMFPACK V4.3 - An Unsymmetric-Pattern Multifrontal Method. *ACM Trans. Math. Softw.*, 30(2):196–199, 2004.
- [dNW06] A. C. de Niet and F. W. Wubs. Two Preconditioners for Saddle Point Problems in Fluid Flows. *Int. J. Num. Meth. Fluids*, 2006.
- [Elm99] H. C. Elman. Preconditioning for the Steady-State Navier–Stokes Equations with Low Viscosity. *SIAM Journal on Scientific Computing*, 20(4):1299–1316, 1999.

- [ES86] H. C. Elman and M. H. Schultz. Preconditioning by Fast Direct Methods for Nonself-Adjoint Nonseparable Elliptic Equations. *SIAM Journal on Numerical Analysis*, 23(1):44–57, feb 1986.
- [ES95] H. C. Elman and D. Silvester. Fast Nonsymmetric Iterations and Preconditioning for Navier-Stokes Equations. Numerical Analysis Report No. 263, Manchester Centre for Computational Mathematics, Manchester, England, 1995. To appear in *SIAM Journal of Scientific Computing*.
- [ESW02] H. C. Elman, D. J. Silvester, and A. J. Wathen. Performance and Analysis of Saddle Point Preconditioners for the Discrete Steady-State Navier–Stokes Equations. *Numerische Mathematik*, 90(4):665–688, 2002.
- [ESW05] H. C. Elman, D. J. Silvester, and A. J. Wathen. *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics*. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford, New York, 2005.
- [GG03] G. H. Golub and C. Greif. On Solving Block-Structured Indefinite Linear Systems. *SIAM Journal on Scientific Computing*, 24(6):2076–2092, November 2003.
- [GLOS05] T. Gelhard, G. Lube, M. A. Olshanskii, and J.-H. Starcke. Stabilized Finite Element Schemes with LBB-Stable Elements for Incompressible Flows. *J. Comput. Appl. Math.*, 177(2):243–267, 2005.
- [GR86] V. Girault and P.-A. Raviart. *Finite Element Methods for the Navier–Stokes Equations*. Springer–Verlag, New York, 1986.
- [GV96] G. H. Golub and C. F. Van Loan. *Matrix Computation*. John Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, Maryland, third edition, 1996.
- [KA00] P. Knabner and L. Angermann. *Numerik partieller Differentialgleichungen*. Springer–Verlag, Berlin, Heidelberg, New York, 2000.
- [KLW02] D. Kay, D. Loghin, and A. Wathen. A Preconditioner for the Steady-State Navier–Stokes Equations. *SIAM J. Sci. Comput.*, 24(1):237–256, 2002.
- [Lem08] W. Lemster. Globalisierte Newton-Verfahren mit Anwendung auf das Navier-Stokes-Problem. Diplomarbeit, Georg-August-Universität Göttingen, 2008.
- [Löw08] J. Löwe. Stabilisierung durch lokale Projektion für inkompressible Strömungsprobleme. Diplomarbeit, Georg-August-Universität Göttingen, 2008.
- [LR06] G. Lube and G. Rapin. Residual-Based Stabilized Higher-Order FEM for a Generalized Oseen Problem. *Math. Models & Meth. in Applied Sciences* 16, pages 1–18, 2006.
- [LRL07] G. Lube, G. Rapin, and J. Löwe. Local Projection Stabilizations for Incompressible Flows: Equal-Order vs. Inf-Sup Stable Interpolation. *submitted*, 2007.
- [LS99] L. Little and Y. Saad. Block LU Preconditioners for Symmetric and Nonsymmetric Saddle Point Problems, 1999.

- [Lub06] G. Lube. Lineare Funktionalanalysis und Anwendungen auf partielle Differentialgleichungen. Georg-August-Universität Göttingen, Skript Wintersemester 2005/2006, 2006.
- [MLR07] G. Matthies, G. Lube, and L. Röhe. Residual-Based Stabilisation of inf-sup Stable Discretisations of the Generalised Oseen Problem. *to appear*, 2007.
- [OB06] M. A. Olshanskii and M. Benzi. An Augmented Lagrangian-Based Approach to the Oseen Problem. *SIAM Journal on Scientific Computing*, 28:2095–2113, 2006.
- [OR04] M. A. Olshanskii and A. Reusken. Grad-Div Stabilization for Stokes Equations. *Math. Comput.*, 73(248):1699–1718, 2004.
- [OV07] M. A. Olshanskii and Y. Vassilevski. Pressure Schur Complement Preconditioners for the Discrete Oseen Problem. Technical report, FB Mathematik, Universität Dortmund, 2007. Ergebnisberichte des Instituts für Angewandte Mathematik, Nummer 339.
- [PA95] M. Poliashenko and C. K. Aidun. A Direct Method for Computation of Simple Bifurcations. *J. Comput. Phys.*, 121(2):246–260, 1995.
- [QV97] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer-Verlag, Berlin, second edition, 1997.
- [RL07] G. Rapin and J. Löwe. Local Projection Stabilizations for Inf-Sup Stable Finite Elements Applied to the Oseen Problem. *submitted*, 2007.
- [Röh07] L. Röhe. Residuale Stabilisierung für Finite-Elemente-Verfahren bei inkompressiblen Strömungen. Diplomarbeit, Georg-August-Universität Göttingen, 2007.
- [Saa91] Y. Saad. A Flexible Inner-Outer Preconditioned GMRES Algorithm. Technical Report 91-279, Minnesota Supercomputer Institute, University of Minnesota, 1991.
- [Saa96] Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS Publ., 1996.
- [SB93] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*, volume 12 of *Texts in applied mathematics*. Springer-Verlag, second edition, 1993.
- [Smo07] L. Smoch. Spectral Behaviour of GMRES Applied to Singular Systems. *Adv. Comput. Math.*, 27(2):151–166, 2007.
- [Ste90] R. Stenberg. Error Analysis of some Finite Element Methods for the Stokes Problem. *Mathematics of Computation*, 54(190):495–508, apr 1990.
- [SW93] D. Silvester and A. Wathen. Fast Iterative Solution of Stabilised Stokes Systems Part I: Using Simple Diagonal Preconditioners. *SIAM J. Numer. Anal.*, 30(3):630–649, 1993.
- [SW94] D. Silvester and A. Wathen. Fast Iterative Solution of Stabilised Stokes Systems Part II: Using General Block Preconditioners. *SIAM J. Numer. Anal.*, 31(5):1352–1367, 1994.
- [Zul02] W. Zulehner. Analysis of Iterative Methods for Saddle Point Problems: a Unified Approach. *Math. Comput.*, 71(238):479–505, 2002.