Freie Universität Berlin
Institut für Mathematik
Berlin, den 14. September 2011

A posteriori Optimierung von Parametern in stabilisierten Finite–Elemente–Methoden für Konvektions-Diffusions-Gleichungen

Masterarbeit

von

Ulrich Wilbrandt

Ulrich Wilbrandt
Ulrich.Wilbrandt@fu-berlin.de

Inhaltsverzeichnis

1 Einleitung **3**

2 FEM **6**
2.1 Notation und Sobolev-Räume 6
2.2 Konvektions-Diffusions-Reaktions-Gleichungen 9
2.2.1 Schwache Lösungen 10
2.3 Finite–Elemente–Methode 14
2.3.1 Galerkin–Methode 14
2.3.2 Gitter 15
2.3.3 Finite Elemente und Finite–Elemente–Räume 17
2.3.4 Finite–Elemente–Methode 21
2.4 Streamline Upwind Petrov-Galerkin–Methode 21
2.4.1 Analysis der SUPG–Methode 24

3 A Posteriori Parameterbestimmung **33**
3.1 Allgemeine Herangehensweise 33
3.2 Berechnung des Gradienten 34
3.3 BFGS-Verfahren 36
3.3.1 Abstiegsrichtung 37
3.3.2 Schrittweite 38
3.3.3 Approximation der Hessematrix 40
3.3.4 L-BFGS 41

4 Diskussion von möglichen Funktionalen **45**
4.1 Residuale Funktionale 45
4.2 Min/Max-Prinzipien 48

5 Numerisches Beispiel **50**
5.1 Problembeschreibung 50
5.2 Ergebnisse 52
5.3 Auswertung 63

6 Ausblick **65**

1 Einleitung

In dieser Arbeit werden Konvektions-Diffusions-Gleichungen betrachtet. Sie modellieren vielfältige Prozesse in Natur und Technik. Man stelle sich beispielsweise einen schnell fließenden Fluss vor, der an einer Stelle durch eine Substanz verunreinigt wird. Die zu untersuchende Frage ist, wie genau sich diese Substanz ausbreitet. Hierbei spielen zwei physikalische Prozesse eine wichtige Rolle. Zum Einen wird die Substanz von der Strömung des Flusses mitgetragen. Dieser Vorgang heißt Konvektion. Es ist jedoch auch zu beobachten, dass eine solche Substanz Gebiete erreicht, in die es die Konvektion allein nicht getragen hätte. Der Grund dafür ist Diffusion. Sie bezeichnet das Vermischen von Teilchen, zum Beispiel aufgrund von verschieden Konzentrationen. Dies ist auch der Grund warum sich Milch in Kaffee vollständig verteilt, auch wenn nicht gerührt wird. Dass trotzdem die meisten Kaffeetrinker, nachdem sie ihre Milch hinzugegeben haben, umrühren, liegt an den verschiedenen Größenordnungen der beiden genannten Prozesse. Die Diffusion ist in diesen Beispielen sehr viel kleiner als die Konvektion, man sagt das Problem ist konvektionsdominant. Die Namensgebung der Konvektions-Diffusion-Gleichungen stammt größtenteils aus der Strömungslehre. Gleichungen dieses Typs modellieren jedoch noch weitere physikalische Phänomene, beispielsweise die Elektronenkonzentration in Halbleiterbauelementen.

Ein wichtiges charakteristisches Merkmal von Lösungen von Konvektions-Diffusions-Gleichungen sind sogenannte Grenzschichten. Dies sind Teile des Gebiets, in denen die Lösung sehr große Gradienten aufweist. Das numerische Lösen dieser Gleichungen ist im konvektionsdominanten Fall eine große Herausforderung. Im Allgemeinen ist es nicht möglich mit Standardmethoden, zum Beispiel Finiten Differenzen oder Finiten Elementen, akzeptable Ergebnisse zu erzielen. Der Grund ist, dass diese Verfahren versuchen, die wichtigsten Merkmale der Lösung aufzulösen. Das kann jedoch nicht funktionieren, da die Grenzschichten viel feiner sind als die verwendeten Gitter. Damit kommen die Verfahren nicht zurecht. Sie liefern oft Lösungen, die durch starke unphysikalische Oszillationen gekennzeichnet sind und daher als unbrauchbar eingestuft werden. Vielmehr sind angepasste Verfahren notwendig, die auch für stark dominierende Konvektion noch verlässliche Resultate liefern.

Ein vor etwa 30 Jahren durch Brooks und Hughes, [BH82, HB79], entwickeltes Verfahren ist die „streamline upwind Petrov–Galerkin“-Methode (SUPG). Um die numerische Lösung zu stabilisieren, wird zusätzliche Diffusion in Stromlinienrichtung hinzugefügt. Daher heißt dieses Verfahren auch Stromlinien–Diffusions–Methode. Diese Art der Stabilisierung führt auf jeder Gitterzelle eines der Rechnung zu Grunde liegenden Gitters einen zu wählenden Parameter ein. Die genaue Wahl dieser sogenannten SUPG-Parameter ist nicht vorgegeben. Aus der Analysis ist lediglich eine asymptotische Vorgabe bekannt. Außerdem gibt es eine Standardwahl, welche in einer Dimension knotenexakte Lösungen liefert, im Allgemeinen gilt dies jedoch nicht in mehreren Dimensionen. In Simulationen lässt sich beobachten, dass die Wahl der SUPG-Parameter großen Einfluss auf die berechnete Lösung hat. Werden sie zu klein gewählt, neigt die Lösung wieder zu unphysikalischen Oszillationen, im Falle zu großer SUPG-Parameter ist die Lösung stark verschmiert, das heißt die Grenzschicht ist viel breiter als in der Realität.

In dieser Arbeit wird ein Verfahren untersucht, welches die SUPG-Parameter a posteriori bestimmt. Dies geschieht, indem mit Hilfe eines geeigneten Funktionals die Qualität der Lösung gemessen und anschließend durch das Lösen einer restringierten Optimierungsaufgabe optimiert wird. Das Ziel ist, auf diese Weise Parameter zu finden, die zu einer Lösung ohne unphysikalische Oszillationen und Verschmierungen führen. Zu dieser Herangehensweise gibt es erst eine Arbeit [JKS11], welche dieses Jahr veröffentlicht wurde. Es werden einige der in [JKS11] als offen bezeichnete Fragen untersucht.

Es stellt sich heraus, dass diese Vorgehensweise grundsätzlich geeignet ist, um unerwünschte Eigenschaften der berechneten Lösung zu reduzieren. Die Parameter werden erfolgreich an die berechnete Lösung angepasst. Insbesondere werden sie in Grenzschichten durch die Optimierung vergrößert. Jedoch sind noch immer Oszillationen vorhanden. Des Weiteren ist die Rechnung sehr aufwändig und die Wahl eines geeigneten Funktionals bleibt offen.

Im folgenden zweiten Kapitel wird zunächst auf die Analysis von Konvektions-Diffusions-Gleichungen eingegangen. Anschließend wird die Finite–Elemente–Methode für diese Gleichungen detailliert eingeführt. Den Abschluss des zweiten Kapitels bildet die Definition und Analysis der SUPG-Methode. Das Kapitel 3 behandelt die a posteriori Parameterbestimmung, insbesondere die Berechnung des Gradienten des Funktionals und das Broyden–Fletcher–Goldfarb–Shanno–Verfahren (BFGS) zur Lösung des Optimierungsproblems. Zusätzlich wird noch das L-BFGS-Verfahren vorgestellt, welches geringere Speicheranforderungen hat. Das vierte Kapitel widmet sich der Diskussion verschiedener möglicher Funktionale. Hier wird zunächst eine Wahl aus [JKS11] untersucht. Anschließend wird ein Funktional eingeführt, welches das Ver-

letzten von Minimum- und Maximumprinzipien bestraft. Solche Verletzungen sind bei numerischen Lösungen oft in der Nähe von Grenzschichten zu beobachten. Im vorletzten Kapitel 5 werden an Hand eines Beispiels die Leistungsmerkmale der vorgestellten a posteriori Parameterbestimmung untersucht und ausgewertet. Das letzte Kapitel bietet einen Ausblick auf mögliche Themen weiterer Forschung.

2 Finite Elemente Methode für Konvektions-Diffusions-Reaktions-Gleichungen

2.1 Notation und Sobolev-Räume

Um Konvektions-Diffusions-Reaktions-Gleichungen vernünftig einführen zu können, braucht man passende Räume, in denen man Lösungen sucht. Es wird sich herausstellen, dass die sogenannten Sobolev-Räume zusammen mit dem Konzept der schwachen Lösungen dafür geeignet sind. Die verwendete Notation ist weitestgehend Standard, wie man sie in den meisten Büchern findet, zum Beispiel in [BS96], [Eva98] oder [RST08]. Im Folgenden ist Ω stets eine beschränkte, offene und zusammenhängende Teilmenge des $\mathbb{R}^d$, deren Rand mit $\partial\Omega$ bezeichnet wird.

Grundlage der meisten in dieser Arbeit verwendeten Räume sind die Lebesgue–Räume,

$$L^p(\Omega) = \left\{ f : \Omega \to \mathbb{R} \;\middle|\; f \text{ ist Lebesgue-messbar, } \|f\|_{L^p(\Omega)} < \infty \right\},$$

für $1 \leq p \leq \infty$ mit der Norm

$$\|f\|_{L^p(\Omega)} = \begin{cases} \left(\int_\Omega |f(\mathbf{x})|^p \, d\mathbf{x}\right)^{1/p} & \text{falls } p < \infty, \\ \operatorname{ess\,sup}_{\mathbf{x}\in\Omega} |f(\mathbf{x})| & \text{falls } p = \infty. \end{cases}$$

Zusätzlich sei noch

$$L^p_{\text{loc}}(\Omega) = \left\{ f : \Omega \to \mathbb{R} \;\middle|\; f \in L^p(V) \text{ für alle } V \subset \overline{V} \subset \Omega \right\}$$

definiert. Man identifiziert Funktionen, die fast überall übereinstimmen, sodass mit $f \in L^p(\Omega)$ streng genommen keine Funktion, sondern eine Äquivalenzklasse von Funktionen gemeint ist. Man spricht trotzdem meistens von einer Funktion $f \in L^p(\Omega)$, die dann nur fast überall definiert ist. Die Lebesgue-Räume sind mit der

gegebenen Norm vollständig, also Banachräume. Der für diese Arbeit besonders interessante Raum $L^2(\Omega)$ ist mit dem Skalarprodukt $(\cdot,\cdot)_0 : L^2(\Omega) \times L^2(\Omega) \to \mathbb{R}$, definiert durch

$$(f,g)_0 = \int_\Omega f(\mathbf{x})g(\mathbf{x})\,d\mathbf{x},$$

sogar ein Hilbertraum ist. Soll in der Definition dieses Skalarprodukts nur über eine offene Teilmengen $T \subset \Omega$ integriert werden, schreibt man auch $(\cdot,\cdot)_{0,T} : L^2(T) \times L^2(T) \to \mathbb{R}$. Weiter seien noch Räume stetig differenzierbarer Funktionen definiert, $k \in \mathbb{N} \cup \{0\}$,

$$\begin{aligned} C^k(\Omega) &= \{f : \Omega \to \mathbb{R} \mid f \text{ } k\text{-mal differenzierbar, alle } k\text{-ten Ableitungen stetig}\}, \\ C^\infty(\Omega) &= \bigcap_{k=1}^{\infty} C^k(\Omega), \\ C_c^\infty(\Omega) &= \{f \in C^\infty(\Omega) \mid \operatorname{supp} f \subset \Omega\}, \end{aligned}$$

wobei $\operatorname{supp} f = \overline{\{\mathbf{x} \in \Omega \mid f(\mathbf{x}) \neq 0\}}$ der Träger von f ist. Ist $f \in C^k(\Omega)$ und sind alle Ableitungen bis zur Ordnung k stetig bis zum Rand fortsetzbar, so schreibt man $f \in C^k(\overline{\Omega})$. Weiterhin ist $C(\Omega) := C^0(\Omega)$.

Um partielle Differentialgleichungen zu lösen, wird es wichtig sein, den Begriff der Ableitung allgemeiner zu fassen. Dies führt auf die sogenannte schwache Ableitung.

Definition 2.1.1: *Seien f, $g \in L^1_{loc}$ und $\alpha \in \mathbb{N}^d$ ein Multiindex der Ordnung $|\alpha| = \alpha_1 + \cdots + \alpha_d$. Man bezeichnet g als die α-te schwache partielle Ableitung von f, falls für alle Testfunktionen $\phi \in C_c^\infty(\Omega)$ gilt*

$$\int_\Omega f(\mathbf{x})D^\alpha\phi(\mathbf{x})\,d\mathbf{x} = (-1)^{|\alpha|}\int_\Omega g(\mathbf{x})\phi(\mathbf{x})\,d\mathbf{x}.$$

Man schreibt dann in Anlehnung an die klassische Ableitung $g = D^\alpha f$.

Man kann zeigen, dass die schwache Ableitung, falls sie existiert, eindeutig ist. Weiterhin sind klassisch differenzierbare Funktionen auch schwach differenzierbar mit der gleichen Ableitung. Die in der folgenden Definition eingeführten Sobolev-Räume beinhalten die schwach differenzierbaren Funktionen:

Definition 2.1.2: *Es seien $1 \leq p \leq \infty$ und $k \in \mathbb{N}$. Dann heißt*

$$W^{k,p}(\Omega) = \left\{ f \in L^p(\Omega) \,\middle|\, \text{für alle } \alpha \in \mathbb{N}^d, |\alpha| \leq k, \text{ existiert } D^\alpha f \text{ und } D^\alpha f \in L^p(\Omega) \right\}$$

Sobolev-Raum. Die zu $W^{k,p}(\Omega)$ gehörige Norm ist definiert als

$$\|f\|_{W^{k,p}(\Omega)} = \sum_{|\alpha|\leq k} \|D^\alpha f(\mathbf{x})\|_{L^p(\Omega)}\,.$$

Im Fall $p = 2$ sind dies Hilberträume mit dem Skalarprodukt

$$(f,g)_k = \sum_{|\alpha|\leq k} (D^\alpha f, D^\alpha g)_0.$$

Man schreibt $H^k(\Omega) = W^{k,2}(\Omega)$ und $\|\cdot\|_k = \|\cdot\|_{H^k(\Omega)}$, sowie $\|v\|_{k,T} = \|v\|_{H^k(T)}$ für Mengen $T \subset \overline{\Omega}$.

Man beachte, dass wegen der Beschränktheit von Ω der Raum $L^p(\Omega)$ eine Teilmenge des $L^1_{\text{loc}}(\Omega)$ ist, sodass wir zur Definition von $W^{k,p}(\Omega)$ die zuvor definierte schwache Ableitung verwenden können.

Neben einer Differentialgleichung werden auch stets Randbedingungen benötigt. Wenigstens auf einem Teil Γ^D des Randes $\partial\Omega$ soll die Lösung vorgegebene Werte annehmen. Dies ist im Sinne von L^p-Funktionen zunächst nicht sinnvoll definiert, da das Teilstück Γ^D zwar positves $d-1$-dimensionales Maß hat, aber als Teilmenge des $\mathbb{R}^d$ eine Nullmenge ist. Hier hilft jedoch der Spursatz weiter, siehe zum Beispiel [Alt07].

Theorem 2.1.3 (Spursatz)**:**
Es sei $\Omega \subset \mathbb{R}^d$ ein beschränktes, zusammenhängendes Gebiet mit Lipschitz–stetigem Rand. Dann existiert ein stetiger linearer Operator $T : H^1(\Omega) \to H^{1/2}(\partial\Omega) \subset L^2(\partial\Omega)$, sodass für alle bis zum Rand stetigen Funktionen $u \in C(\overline{\Omega})$ gilt

$$Tu = u|_{\partial\Omega}. \tag{2.1}$$

Des Weiteren ist T surjektiv.

Bermerkung 2.1.4: Später wird auch die Spur von Funktionen auf zwei disjunkten, relativ offenen Teilstücken Γ^D und Γ^N, mit $\partial\Omega = \overline{\Gamma^D \cup \Gamma^N}$, benötigt. Für $v \in H^{1/2}(\partial\Omega)$ sind die Einschränkungen $v \mapsto v|_{\Gamma^D} \in H^{1/2}(\Gamma^D)$ und $v \mapsto v|_{\Gamma^N} \in H^{1/2}(\Gamma^N)$ ebenfalls stetig, linear und surjektiv. Daher definiert man Spuroperatoren $T^D : H^1(\Omega) \to H^{1/2}(\Gamma^D)$ und $T^N : H^1(\Omega) \to H^{1/2}(\Gamma^N)$ als Komposition von T und diesen Einschränkungen.

Bermerkung 2.1.5: Als Bildraum des Spuroperators T wird oft auch $L^2(\partial\Omega)$ genommen. Es stellt sich heraus, dass T nur in einen Teilraum $X = \operatorname{Bild} T$ von $L^2(\partial\Omega)$ abbildet. So kann man diesen Teilraum $H^{1/2}(\partial\Omega) = X$ auch als Bild von

T definieren. Die folgenden beiden Normen auf $H^{1/2}(\partial\Omega)$ sind äquivalent

$$\|u\|_{H^{1/2}(\partial\Omega)} := \inf_{v\in H^1(\Omega), Tv=u} \|v\|_{H^1(\Omega)},$$

$$|||u|||_{H^{1/2}(\partial\Omega)} := \left(\|u\|^2_{L^2(\partial\Omega)} + \int_{\partial\Omega}\int_{\partial\Omega} \frac{|u(x)-u(y)|^2}{\|x-y\|^d}\, dxdy \right)^{1/2}.$$

Die zweite Norm $|||\cdot|||_{H^{1/2}(\Gamma^D)}$ ist die für $H^{1/2}(\Gamma^D)$ übliche Sobolev-Slobodeckij-Norm. Die Sobolev-Slobodeckij-Räume sind eine Verallgemeinerung der Hölder-Räume. Eine weitere Möglichkeit Sobolev-Räume mit gebrochenen Indizes zu definieren besteht durch Interpolation, siehe z. B. [BS96] oder [Dob10].

2.2 Konvektions-Diffusions-Reaktions-Gleichungen

Es sei $\Omega \subset \mathbb{R}^d$, $d = 2$ oder $d = 3$, ein beschränktes Lipschitzgebiet. Gesucht ist eine Lösung u, welche die folgende skalare lineare partielle Differentialgleichung zusammen mit den Randbedingungen erfüllt,

$$L(u) := -\varepsilon\Delta u + \mathbf{b}\cdot\nabla u + cu = f \quad \text{in } \Omega, \tag{2.2a}$$

$$u = u_b \quad \text{auf } \Gamma^D, \tag{2.2b}$$

$$\varepsilon\frac{\partial u}{\partial \mathbf{n}} = g \quad \text{auf } \Gamma^N. \tag{2.2c}$$

Hierbei sind $\Delta u = \sum_{i=1}^d \frac{\partial^2 u}{\partial x_i^2}$ und $\mathbf{b}\cdot\nabla u = \sum_{i=1}^d b_i \frac{\partial u}{\partial x_i}$. Der Rand $\partial\Omega = \overline{\Gamma^D \cup \Gamma^N}$ des Gebiets Ω besteht aus den zwei disjunkten relativ offenen Teilmengen Γ^D und Γ^N, die Dirichlet- bzw. Neumann-Rand genannt werden. Der Vektor $\mathbf{n}$ ist die äußere Normale an den Rand von Ω. Weiterhin beinhalte Γ^D den Einflussrand,

$$\{\mathbf{x}\in\partial\Omega : \ \mathbf{b}(x)\cdot\mathbf{n}(x) < 0\} \subset \Gamma^D, \tag{2.3}$$

und habe positives $(d-1)$-dimensionales Maß. Die Koeffizienten und Daten genügen den Bedingungen

1. $\varepsilon > 0$,
2. $\mathbf{b} \in L^\infty(\Omega, \mathbb{R}^d)$, div$\mathbf{b} \in L^\infty(\Omega)$,
3. $c \in L^\infty(\Omega)$, $c(x) \geq 0$ für fast alle $x \in \Omega$,
4. $f \in L^2$,
5. $u_b \in H^{1/2}(\Gamma^D)$,
6. $g \in L^2(\Gamma^N)$.

Es wird sich herausstellen, dass man die Existenz und Eindeutigkeit von (schwachen) Lösungen mit der Bedingung

$$c - \frac{1}{2}\text{div}\mathbf{b} \geq c_0 > 0 \tag{2.4}$$

zeigen kann. Die Größe ε in der Differentialgleichung (2.2) heißt Diffusionskoeffizient. Die vektorwertige Funktion $\mathbf{b}$ wird Konvektion, c Reaktion und f Quellterm genannt.

2.2.1 Schwache Lösungen

Zunächst wird die zu lösende Differentialgleichung (2.2) in eine schwache Form überführt. Zur Motivation sei zunächst angenommen, u ist tatsächlich eine zweimal stetig differenzierbare Lösung, $u \in C^2(\Omega)$. Dann multipliziert man die Gleichung $Lu = f$ mit einer glatten Testfunktion $v \in C^\infty(\Omega) \cap H^1(\Omega)$, die in einer Umgebung von Γ^D verschwindet, $v \in C^\infty_{\Gamma^D}(\Omega)$ mit

$$C^\infty_{\Gamma^D}(\Omega) := \left\{ v \in C^\infty(\Omega) \cap H^1(\Omega) : \begin{array}{l} \text{es gibt eine offene Umgebung } U \subset \mathbb{R}^d \text{ von} \\ \Gamma^D, \text{ sodass gilt } v(\mathbf{x}) = 0 \text{ für alle } \mathbf{x} \in U \cap \Omega. \end{array} \right\}$$

Anschließend wird über das Gebiet Ω integriert,

$$\int_\Omega -\varepsilon\Delta u(\mathbf{x})v(\mathbf{x}) + \mathbf{b}(\mathbf{x}) \cdot \nabla u(\mathbf{x})v(\mathbf{x}) + c(\mathbf{x})u(\mathbf{x})v(\mathbf{x})\, d\mathbf{x} = \int_\Omega f(\mathbf{x})v(\mathbf{x})\, d\mathbf{x}.$$

Den Diffusionsterm integriert man einmal partiell und erhält

$$\begin{aligned} a(u,v) &:= \int_\Omega \varepsilon\nabla u(\mathbf{x}) \cdot \nabla v(\mathbf{x}) + \mathbf{b}(\mathbf{x}) \cdot \nabla u(\mathbf{x})v(\mathbf{x}) + c(\mathbf{x})u(\mathbf{x})v(\mathbf{x})\, d\mathbf{x} \\ &= \int_\Omega f(\mathbf{x})v(\mathbf{x})\, d\mathbf{x} + \int_{\Gamma^N} g(\mathbf{x})v(\mathbf{x})\, d\mathbf{x} =: l(v). \end{aligned}$$

Der Raum $C^\infty_{\Gamma^D}(\Omega)$, aus dem diese Testfunktion stammt, wird nun abgeschlossen bezüglich der $H^1(\Omega)$-Norm,

$$V := \overline{C^\infty_{\Gamma^D}(\Omega)}^{H^1(\Omega)}. \tag{2.5}$$

Durch Approximation kann demnach als Testraum V gewählt werden und die Gleichung $a(u,v) = l(v)$ bleibt sinnvoll für $v \in V$. Der Raum V beinhaltet Funktionen,

die im Spursinne auf Γ^D verschwinden und liegt damit zwischen $H_0^1(\Omega)$ und $H^1(\Omega)$,

$$V = \left\{ v \in H^1(\Omega) \,\middle|\, T^D v = 0 \right\}, \qquad H_0^1(\Omega) \subset V \subset H^1(\Omega).$$

Es sei noch bemerkt, dass unter dem Integral über den Neumann-Rand streng genommen die Spur $T^N v(\mathbf{x})$ statt $v(\mathbf{x})$ stehen müsste. Es ist jedoch üblich, diese Feinheit zur besseren Übersichtlichkeit nicht explizit mit aufzuführen. Die schwache Formulierung des ursprünglichen Problems (2.2) lautet dann: Finde $u \in H^1$, sodass $u - u_b \in V$ ist und für alle $v \in V$ gilt

$$a(u, v) = l(v). \tag{2.6}$$

Bermerkung 2.2.1: Der Ausdruck $u - u_b \in V$ ist zunächst nicht sinnvoll, da u und u_b nicht dem gleichen Raum angehören. Man benötigt eine Fortsetzung der Funktion $u_b \in H^{1/2}(\Gamma^D)$ zu einer Funktion in $H^1(\Omega)$. Dass eine solche existiert, sichert die Surjektivität des Spuroperators T^D, siehe Bemerkung 2.1.4. Diese Fortsetzung wird wieder mit u_b bezeichnet, um die Notation nicht unnötig zu erschweren.

Der Raum V erbt seine Norm von $H^1(\Omega)$. Es erweist sich als günstig stattdessen eine äquivalente Norm zu betrachten. Dazu benötigt man die wichtige Ungleichung von Poincaré und Friedrichs, siehe z. B. [Bra07].

Theorem 2.2.2 (Poincaré–Friedrichssche Ungleichung)**:**
Es sei Ω in einem d-dimensionalen Würfel der Kantenlänge s enthalten. Dann gilt für alle $v \in V$

$$\|v\|_0 \leq s \|\nabla v\|_0 . \tag{2.7}$$

Per Definition gilt auch $\|\nabla v\|_0 \leq \|v\|_1$, das heißt die Abbildung $v \mapsto |v|_1 = \|\nabla v\|_0$ ist auf V eine zu $\|v\|_1$ äquivalente Norm. Im Folgenden sei $|\cdot|_1$ die auf V betrachtete Norm.

Existenz und Eindeutigkeit

Die Existenz und Eindeutigkeit einer schwachen Lösung zu (2.6) wird das Theorem von Lax–Milgram liefern. Es ist in allen Lehrbüchern zu finden, die sich mit der (schwachen) Lösbarkeit von elliptischen Differentialgleichungen beschäftigen, als auch in den meisten einführenden Büchern zur Funktionalanalysis, z. B. [Bra07, Dob10, BS96]. Zunächst noch eine dafür wichtige Definition.

Definition 2.2.3: *Es seien X ein Hilbertraum und $b(\cdot, \cdot) : X \times X \to \mathbb{R}$ eine Abbildung. Dann heißt b bilinear, falls b in beiden Komponenten linear ist. Sie heißt*

koerziv[1]*, falls es eine Konstante $\alpha > 0$ derart gibt, dass für alle $u \in X$ gilt*

$$b(u,u) \geq \alpha \|u\|^2 .$$

Die Bilinearform b ist stetig, falls es eine Konstante C gibt, sodass für alle u, $v \in X$ gilt

$$b(u,v) \leq C \|u\| \|v\| .$$

Der Dualraum eines normierten Raumes X wird mit X' bezeichnet.

Theorem 2.2.4 (Lax–Milgram)**:**
Es seien X ein Hilbertraum, $b(\cdot,\cdot) : X \times X \to \mathbb{R}$ eine stetige und koerzive Billinearform sowie $F \in X'$. Dann existiert genau ein $u \in X$, sodass für alle $v \in X$ gilt

$$b(u,v) = F(v).$$

Dieses Theorem kann nicht direkt auf die Bilinearform a aus (2.6) angewendet werden, da a eine Abbildung von $H^1(\Omega) \times V$ nach $\mathbb{R}$ ist. Nur im Fall von homogenen Dirichlet–Randdaten, $u_b = 0$, wird die schwache Lösung u ebenfalls in V gesucht. Ist $u_b \neq 0$, kann in (2.6) $U := u - u_b \in V$ statt u gesetzt werden und man erhält

$$a(U,v) = a(u,v) - a(u_b,v) = l(v) - a(u_b,v) =: \tilde{l}(v). \tag{2.8}$$

So ist $a : V \times V$ eine Bilinearform auf V und nur die rechte Seite ist verändert. Die gesuchte Lösung u erhält man dann als $u = U + u_b$. Es bleibt zu überprüfen, dass a und $\tilde{l}$ die Voraussetzungen des Theorems von Lax–Milgram erfüllen. Dazu seien v, $w \in V$. Dann ist

$$\begin{aligned} |a(v,w)| &= \left| \int_\Omega \varepsilon \nabla v(\mathbf{x}) \cdot \nabla w(\mathbf{x}) + \mathbf{b}(\mathbf{x}) \cdot \nabla v(\mathbf{x}) w(\mathbf{x}) + c(\mathbf{x}) v(\mathbf{x}) w(\mathbf{x}) \, d\mathbf{x} \right| \\ &\leq \varepsilon \|\nabla v\|_0 \|\nabla w\|_0 + \|\mathbf{b}\|_\infty \|\nabla v\|_0 \|w\|_0 + \|c\|_\infty \|v\|_0 \|w\|_0 \\ &\leq \max\{\varepsilon, \|\mathbf{b}\|_\infty, \|c\|_\infty\} \|v\|_1 \|w\|_1 \\ &\overset{(2.7)}{\leq} C |v|_1 |w|_1 , \end{aligned}$$

wobei in der ersten Abschätzung die Cauchy–Schwarz–Ungleichung benutzt wurde und die Konstante C zusätzlich von $\operatorname{diam}\Omega$ abhängt. Somit ist a stetig auf $V \times V$.

[1] auch: X-elliptisch, elliptisch, koerzitiv, nach unten beschränkt

Weiterhin ist a koerziv, da

$$\begin{aligned} a(v,v) &= \int_\Omega \varepsilon|\nabla v(\mathbf{x})|^2 + \mathbf{b}(\mathbf{x})\cdot\nabla v(\mathbf{x})v(\mathbf{x}) + c(\mathbf{x})v(\mathbf{x})^2\,d\mathbf{x} \\ &\geq \varepsilon\,\|\nabla v\|_0^2 + \int_\Omega c(\mathbf{x})v(\mathbf{x})^2 - \frac{1}{2}\left(\nabla\cdot\mathbf{b}(\mathbf{x})\right)v(\mathbf{x})^2\,d\mathbf{x} + \frac{1}{2}\int_{\Gamma^N} v^2(\mathbf{x})\mathbf{b}(\mathbf{x})\cdot\mathbf{n}(\mathbf{x})\,d\mathbf{x} \\ &\geq \varepsilon\,|v|_1^2 + c_0\int_\Omega v(\mathbf{x})^2\,d\mathbf{x} \\ &\geq \varepsilon\,|v|_1^2\,. \end{aligned}$$

Hierbei wurde ausgenutzt, dass $\mathbf{b}(\mathbf{x})\cdot\mathbf{n}$ auf dem Neumann–Rand nicht negativ ist, siehe (2.3), sowie dass $c_0 > 0$, (2.4). Die rechte Seite $\tilde{l}$ ist ein Element aus dem Dualraum von V, da sie linear ist und

$$\begin{aligned} \tilde{l}(v) &= \int_\Omega f(\mathbf{x})v(\mathbf{x})d\mathbf{x} + \int_{\Gamma^N} g(\mathbf{x})v(\mathbf{x})d\mathbf{x} - a(u_b,v) \\ &\leq \|f\|_0\,\|v\|_0 + \|g\|_{0,\Gamma^N}\,\|v\|_{0,\Gamma^N} + \max\left\{\varepsilon, \|\mathbf{b}\|_\infty, \|c\|_\infty\right\}\|u_b\|_1\,\|v\|_1 \\ &\leq \max\left\{\|f\|_0, \|g\|_{0,\Gamma^N}\left\|T^N\right\|, \|u_b\|_1\max\left\{\varepsilon, \|\mathbf{b}\|_\infty, \|c\|_\infty\right\}\right\}\|v\|_1 \\ &\leq C\,|v|_1\,, \end{aligned}$$

wobei $\|v\|_{0,\Gamma^N}$ mit Hilfe der Operatornorm des Spuroperators T^N abgeschätzt wurde, $\|v\|_{0,\Gamma^N} = \left\|T^N v\right\|_{0,\Gamma^N} \leq \left\|T^N\right\|\|v\|_0$. Die Konstante C hängt von $\operatorname{diam}\Omega$ ab. Mit dem Theorem von Lax–Milgram ist also die Existenz und Eindeutigkeit von schwachen Lösungen zu (2.6) gesichert.

Konvektionsdominanter Fall

Das Theorem von Lax–Milgram liefert zusätzlich zur Existenz und Eindeutigkeit noch eine Abschätzung an die Lösung u. Mit den dortigen Bezeichnungen gilt

$$\|u\|_X \leq \frac{1}{\alpha}\,\|F\|_{X'}\,,$$

wobei α die Koerzivitätskonstante aus Definition 2.2.3 ist. Betrachtet man also $\varepsilon \to 0$, erhält man zwar für jedes ε eine Lösung u_ε, jedoch sind diese Lösungen nicht gleichmäßig beschränkt. Es ist also unklar wie etwa $\lim_{\varepsilon\to 0} u_\varepsilon$ zu verstehen ist.

Setzt man formal $\varepsilon = 0$, so erhält man das *reduzierte Problem*

$$L_0 u_0 := \mathbf{b} \cdot \nabla u_0 + c u_0 = f,$$

siehe Abschnitt III.1.2 in [RST08]. Diese Gleichung ist parabolisch und Lösungen haben andere Eigenschaften als Lösungen von elliptischen Problemen wie (2.2). Insbesondere ist L_0 nicht glättend. Das heißt, während zum Beispiel aus $\Delta u \in L^2(\Omega)$ unter gewissen Bedingungen an das Gebiet $u \in H^2(\Omega)$ folgt, gilt dies nicht für L_0. Ebenfalls aus [RST08, III.1.2, III.1.3] ist jedoch folgendes Resultat bekannt:

Theorem 2.2.5:
Es seien die Koeffizientenfunktionen $\mathbf{b}$ *und* c *bis zum Rand stetig differenzierbar,* $\mathbf{b}$, $c \in C^1(\overline{\Omega})$, *sowie* $c_0 = c - \frac{1}{2}\mathrm{div}\mathbf{b} > 0$. *Weiterhin seien die Dirichlet-Daten auf dem Einströmrand* $\Gamma_- := \{\mathbf{x} \in \partial\Omega : \ \mathbf{b}(x) \cdot \mathbf{n}(x) < 0\} \subset \Gamma^D$ *homogen. Dann gibt es eine eindeutige Lösung* $u_0 \in L^2(\Omega)$ *des reduzierten Problems*

$$L_0 u_0 := \mathbf{b} \cdot \nabla u_0 + c u_0 = f \quad \text{in } \Omega, \tag{2.9}$$

$$u_0 = 0 \quad \text{auf } \Gamma_-. \tag{2.10}$$

Weiterhin konvergiert die Lösung u *von* (2.6) *schwach in* $L^2(\Omega)$ *gegen die Lösung des reduzierten Problems,* $u \rightharpoonup u_0$ *für* $\varepsilon \to 0$.

Im Allgemeinen ist die Lösung u_0 des reduzierten Problems nicht schwach differenzierbar. In bestimmten Teilen des Gebiets, unter anderem außerhalb von Grenzschichten, kann man auch

$$|u(x) - u_0(x)| \leq C\varepsilon \tag{2.11}$$

zeigen, siehe [GFL+83, RST08].

2.3 Finite–Elemente–Methode

2.3.1 Galerkin–Methode

Ausgangspunkt für die Finite–Elemente–Methode (FEM) ist die schwache Formulierung (2.6). Der unendlich-dimensionale Raum V wird durch einen endlich-dimensionalen Raum $V_h \subset V$ ersetzt, wobei h ein Parameter ist (in der Regel die Gitterweite), der angibt, wie gut V durch V_h approximiert wird, also $\dim V_h \to \infty$ für $h \to 0$. Man bezeichnet V_h oft als Ansatzraum. Der Einfachheit halber seien die Dirichlet–Daten homogen, $u_b = 0$, andernfalls muss die rechte Seite wie in Gleichung (2.8) verän-

dert werden. Es wird nun eine Lösung $u_h \in V_h$ gesucht, welche für alle $v_h \in V_h$ die Bedingung

$$a(u_h, v_h) = l(v_h) \tag{2.12}$$

erfüllt. Dies wird auch als Galerkin–Verfahren bezeichnet. Das Lemma von Lax–Milgram liefert auch hier eindeutige Lösbarkeit.

Es sei nun $\{\phi_1, \phi_2, \ldots, \phi_n\}$ eine Basis von V_h. Dann muss (2.12) nicht mehr für alle Testfunktionen $v_h \in V_h$ überprüft werden, sondern nur noch für alle Basisfunktionen ϕ_i, $i = 1, \ldots, n$. Die Lösung u_h lautet in dieser Basis

$$u_h = \sum_{k=1}^{n} z_k \phi_k$$

mit den unbekannten reellen Koeffizienten z_k. Man identifiziert den n–dimensionalen Unterraum V_h mit $\mathbb{R}^n$ und sucht nun nach dem Vektor z, dessen k-te Komponente z_k ist. Die zu lösende Aufgabe ist dann: Finde $z \in \mathbb{R}^n$, sodass für alle $i = 1, \ldots, n$ gilt

$$\sum_{k=1}^{n} a(\phi_k, \phi_i) z_k = l(\phi_i).$$

Der Vektor z erfüllt demnach das lineare Gleichungssystem

$$Az = b, \tag{2.13}$$

mit der sogenannten Steifigkeitsmatrix $A = (a_{ij})$ und rechten Seite $b = (b_j)$,

$$a_{ij} = a(\phi_j, \phi_i), \qquad b_i = l(\phi_i).$$

Die Koerzivität der Bilinearform a sichert nun die positive Definitheit der Matrix A mit der gleichen Konstanten.

Bermerkung 2.3.1: Bisher wurde angenommen, der approximierende Raum V_h sei ein Teilraum von V. Man spricht dann oft von einem konformen Verfahren. Gibt man diese Einschränkung auf, gelangt man zu nichtkonformen Methoden. Beispiele sind unstetige Galerkin–Methoden und parametrische Finite Elemente.

2.3.2 Gitter

Die Basisfunktionen der FEM haben in der Regel einen kleinen Träger und sind stückweise polynomial. Für diese Konstruktion benötigt man ein Gitter.

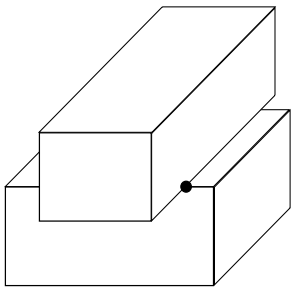
Definition 2.3.2 (Gitter)**:** *Sei $\Omega \in \mathbb{R}^d$, $d = 2$ oder $d = 3$, ein polygonal umrandetes Gebiet. Ein Gitter*[2] *$\mathcal{T}_h$ ist eine Menge von offenen Teilgebieten $T \subset \Omega$ mit folgenden Eigenschaften:*

i) $\overline{\Omega} = \bigcup_{T \in \mathcal{T}_h} \overline{T}$.
ii) Jedes $T \in \mathcal{T}_h$ ist ein konvexer Polyeder.
iii) Je zwei verschiedene T_i, $T_j \in \mathcal{T}_h$ sind disjunkt, $T_i \cap T_j = \emptyset$.

Die Teilgebiete $T \in \mathcal{T}_h$ werden auch Gitterzellen genannt. Die Gitterweite h sei definiert als die längste Kante des Gitters.

Formal ist jede Gitterzelle T das Innere der konvexen Hülle von Punkten $x_1, \ldots, x_n \in \Omega$. Diese Punkte heißen Eckpunkte. Weiter wird die Verbindung zwischen diesen Punkten Kanten genannt, falls diese Teil des Randes der Gitterzelle ist. Im Fall $d = 3$ bilden (mindestens) drei Eckpunkte eine Fläche der Gitterzelle, falls die von diesen Punkten aufgespannte Ebene Teil des Randes dieser Gitterzelle ist.

In dieser Arbeit wird die Menge Ω stets als Lipschitzgebiet angenommen. Oft wird in der Literatur behauptet, dass polygonal berandete Gebiete bereits solche sind. Dies ist jedoch nur in zwei Dimensionen richtig, wie Beispiel 6.5 aus [Dob10] zeigt: Hierbei besteht $\Omega \in \mathbb{R}^3$ aus zwei Quadern, wie in der nebenstehenden Abbildung. Es ist jedoch nicht möglich eine Umgebung um den markierten Punkt zu finden, in der sich der Rand als Graph einer Lipschitz–stetigen Funktion schreiben lässt. Jedoch bleiben Sobolev-Ungleichung und Spurabbildung für Gebiete, die eine innere Kegelbedingung erfüllen, gültig, siehe [Dob10], Anmerkungen 6.12(ii) und 6.16.(ii).



Definition 2.3.3 (zulässige Gitter)**:** *Ein Gitter $\mathcal{T}_h$ aus der vorherigen Definition heißt zulässig, falls für je zwei verschiedene Gitterzellen T_i, $T_j \in \mathcal{T}_h$ gilt: Der Schnitt $\overline{T}_i \cap \overline{T}_j$ ist entweder*

- *genau eine vollständige gemeinsame Fläche (falls $d = 3$),*
- *genau eine vollständige gemeinsame Kante,*
- *genau ein gemeinsamer Eckpunkt,*
- *oder leer.*

Es werden hier ausschließlich zulässige Gitter betrachtet. Es ist jedoch möglich, auch auf nicht zulässigen Gittern zu rechnen. Dann hat man oft sogenannte hängende Kno-

[2] auch: Triangulierung

ten. Um sicherzustellen, dass dann der Ansatz- und Testraum v_h noch ein Unterraum von V ist, muss man zusätzliche Bedingungen einführen.

In zwei Raumdimensionen sind Gitter aus Dreiecken und Vierecken üblich. In drei Dimensionen werden häufig Tetraeder, Quader oder Prismen verwendet. Solche Gitter werden in der Regel von Gittergeneratoren erzeugt. Oft bietet es sich auch an, ein sehr grobes Gitter mehrfach zu verfeinern.

Um Konvergenz von Finiten Element Methoden zu beweisen, wird es wichtig sein, dass die Gitterzellen nicht zu sehr entartet sind. Das bedeutet, dass sie zum Beispiel nicht zu flach werden. Eine Präzisierung dieser Einschränkung ist der Begriff der Regularität eines Gitters

Definition 2.3.4: *Es sei $\mathcal{T}_h$ ein zulässiges Gitter. Zu jeder Gitterzelle $T \in \mathcal{T}_h$ seien h_T dessen Durchmesser und ϱ_T der Radius des größten Innenkreises, also*

$$h_T = \sup\{|x-y| \mid x, y \in T\}, \qquad \varrho_T = \sup\{r \mid \exists x \in T : B_r(x) \subset T\}.$$

Das Gitter $\mathcal{T}_h$ wird regulär *genannt, falls es eine Konstante C gibt, die nicht von der Gitterweite h oder ε abhängt, sodass für alle Gitterzellen T gilt*

$$\frac{h_T}{\varrho_T} \leq C.$$

Bermerkung 2.3.5: Der Begriff regulär wird in der Literatur nicht einheitlich verwendet. Man findet Varianten wie formregulär, uniform oder auch quasiuniform. Hier sind die Definitionen ebenfalls nicht einheitlich.

2.3.3 Finite Elemente und Finite–Elemente–Räume

Zu einem Gitter werden zunächst auf jeder Gitterzelle Finite Elemente definiert, die dann zu einem Finite–Elemente–Raum zusammen geführt werden. Wir folgen [BS96] und [Cia02].

Definition 2.3.6 (Finites Element)**:** *Es seien*

(i) $T \subset \mathbb{R}^d$ *ein Gebiet mit stückweise glattem Rand,*
(ii) $\mathcal{P}$ *ein endlich-dimensionaler Funktionenraum auf* T *und*
(iii) $\mathcal{N} = \{N_1, \ldots, N_k\}$ *eine Basis von* $\mathcal{P}'$.

Dann nennen wir das Tripel $(T, \mathcal{P}, \mathcal{N})$ ein Finites Element.

Typische Funktionale $\Phi \in \mathcal{N}$ sind Funktionsauswertungen an einem Punkt $\Phi(v) = v(\mathbf{x})$, Auswertungen von Ableitungen $\Phi(v) = \partial_i v(\mathbf{x})$, $\Phi(v) = \partial_{ij} v(\mathbf{x})$ oder auch Integralwerte, etwa $\Phi(v) = \int_T v \, d\mathbf{x}$. Sind ausschließlich Funktionsauswertungen an verschiedenen Punkten beteiligt, so spricht man von Lagrange–Elementen, werden auch Ableitungen ausgewertet von Hermite–Elementen.

In dieser Arbeit soll stets die nodale Basis $\{\phi_1, \ldots, \phi_k\}$ von $\mathcal{P}$ betrachtet werden. Das heißt es gilt $N_i(\phi_j) = \delta_{ij}$. In diesem Fall ist der lokale Interpolationsoperator $\mathcal{I}_T$ definiert durch

$$\mathcal{I}_T w := \sum_{i=1}^{k} N_i(w)\phi_i,$$

wobei $w \in C^m(\overline{\Omega})$ ist und $m \in \mathbb{N}$ die Ordnung der höchsten partiellen Ableitung, die in einem der Funktionale $N_i(w)$ auftritt. Hier wird implizit angenommen, dass die Funktionale N_i auf $C^m(\overline{\Omega})$ statt nur auf $\mathcal{P}$ definiert sind. Zu einer Triangulierung $\mathcal{T}_h$ kann man dann einen globalen Interpolationsoperator $\mathcal{I}_{\mathcal{T}_h}$ stückweise definieren,

$$\mathcal{I}_{\mathcal{T}_h} w\big|_T = \mathcal{I}_T w.$$

Der Finite–Elemente–Raum $W_{\mathcal{T}_h}$ zu einem Gitter $\mathcal{T}_h$ ist definiert als

$$W_{\mathcal{T}_h} = \left\{ \mathcal{I}_{\mathcal{T}_h} w \;\middle|\; w \in C^m(\overline{\Omega}) \right\}. \tag{2.14}$$

Die so definierte Interpolante $\mathcal{I}_{\mathcal{T}_h} w$ von w muss im Allgemeinen nicht stetig sein. Dazu müssen die Funktionale N_i entsprechend gewählt werden. Umfasst ein Finite–Elemente–Raum den Raum der r-mal bis zum Rand stetig differenzierbaren Funktionen $C^r(\overline{\Omega})$, so wird er C^r-Finite–Elemente–Raum genannt.

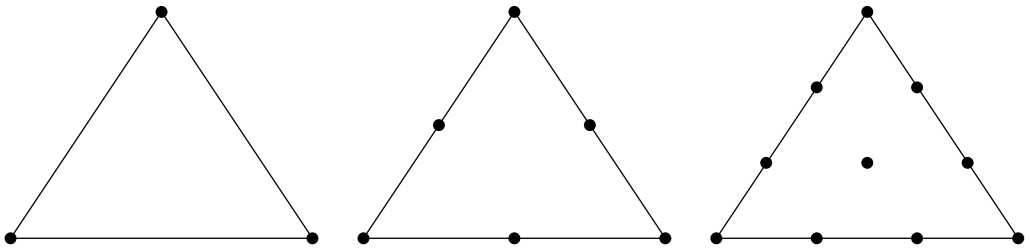
Wie die Funktionale N_i gewählt werden können, um C^0-Finite–Elemente–Räume zu konstruieren, soll durch einige Standardbeispiele verdeutlicht werden. Weitere sind unter anderem zu finden in [BS96, Kapitel 3].

Beispiel 2.3.7 (Lagrange–Elemente auf Dreiecken)**:** Es sei $\Omega \in \mathbb{R}^d$, $d = 2$, mit einem Dreiecksgitter $\mathcal{T}_h$ zum Beispiel wie in Abbildung 2.2 links zu sehen. Zu jedem Dreieck $T \in \mathcal{T}_h$ definiere $\mathcal{P}_k$ als den Raum der Polynome vom Grad höchstens $k \in \mathbb{N}$ mit Dimension $m = \dim \mathcal{P}_k = \frac{1}{2}(k+1)(k+2)$ und $\mathcal{N}_k = \{N_1, \ldots, N_m\}$ als die Funktionsauswertungen an den Knoten

$$\left\{ p_\alpha = \sum_{j=1}^{3} \frac{\alpha_j}{|\alpha|} x_j \;\middle|\; \alpha \text{ ist ein Multiindex der Ordnung } |\alpha| = k \right\},$$

wobei x_1, x_2 und x_3 die Eckpunkte des Dreiecks T sind. Die Knoten für die Fälle

$k = 1, 2, 3$ sind in Abbildung 2.1 dargestellt. Dann sind alle $(T, \mathcal{P}_k, \mathcal{N}_k)$ Finite



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$

Abbildung 2.1: Knoten der Lagrange–Elemente für verschiedene Polynomgrade k.

Elemente. Die nodale Basis zu jedem Finiten Element besteht aus Polynomen, die an genau einem Knoten den Wert Eins haben und an allen anderen Knoten verschwinden.

Der zu diesem Gitter gehörige Finite–Elemente–Raum besteht aus stetigen Funktionen, die auf jeder Gitterzelle ein Polynom vom Grad k sind. Basisfunktionen werden in der Regel ebenfalls nodal gewählt, das heißt so, dass jede Basisfunktion an genau einem Knoten den Wert Eins hat an allen anderen Knoten des Gitters den Wert Null, siehe Abbildung 2.2 rechts für den Fall $k = 1$. So stimmen diese Basisfunktionen dann auch auf jedem Finiten Element mit einer lokalen Basisfunktion überein. Eine analoge Konstruktion kann auch in mehr als zwei Dimensionen auf Simplizes durchgeführt werden und die entstehenden Finite–Elemente–Räume werden P^k-Finite–Elemente–Räume genannt.

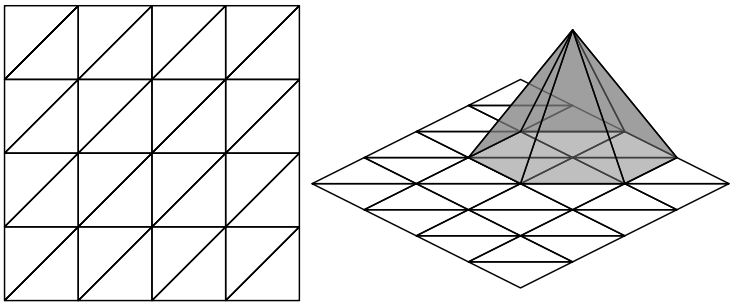


Abbildung 2.2: Ein einfaches Gitter und eine P^1–Basisfunktion.

Beispiel 2.3.8: Es sei wieder $\Omega \subset \mathbb{R}^2$, jedoch mit einem Rechtecksgitter $\mathcal{T}_h$. Auf jedem Rechteck T wird ein $(k+1)^2$-dimensionaler Polynomraum definiert, der alle Polynome $p : \mathbb{R}^2 \to \mathbb{R}$ beinhaltet, für die sowohl $x \mapsto p(x, y)$ (für alle y mit

$(x,y) \in T$) als auch $y \mapsto p(x,y)$ (für alle x mit $(x,y) \in T$) Polynome vom Grad höchstens k sind. Die Funktionale N_i seien wieder Funktionsauswertungen an den Knoten

$$\left\{ p_\alpha = \sum_{j=1}^{4} \frac{\alpha_j}{|\alpha|} x_j \;\middle|\; \alpha \text{ ist ein Multiindex der Ordnung } |\alpha| = k \right\},$$

wobei x_1, x_2, x_3, und x_4 die Eckpunkte von T sind. Hier kann man ebenfalls für den Finite–Elemente–Raum eine (globale) nodale Basis wählen. Auch in mehreren Dimensionen heißt dieser Raum Q^k-Finite–Elemente–Raum.

Für die Analysis wird es später wichtig sein, dass der Interpolationsoperator $\mathcal{I}_T$ auf jeder Gitterzelle T Funktionen ausreichend gut approximiert. Dies besagt das folgende Theorem 4.4.4 aus [BS96]

Theorem 2.3.9:
Sei $(T, \mathcal{P}, \mathcal{N})$ ein Finites Element mit folgenden Eigenschaften

(i) T ist sternenförmig[3] bezüglich der größten enthaltenen Kugel (mit Radius ϱ_T),
(ii) $\mathcal{P}_k \subset \mathcal{P} \subset W^{k+1,\infty}$,
(iii) $\mathcal{N} \subset \left(C^m(\overline{T})\right)'$.

Es seien weiter $k+1-m-d/2 > 0$ und $0 \le i \le k+1$. Dann gilt für alle $v \in H^{k+1}(T)$

$$|v - \mathcal{I}_T v|_{i,T} \le C h_T^{k+1-i} |v|_{k+1,T}, \tag{2.15}$$

wobei C nur von k, h_T/ϱ_T, $\mathcal{P}$ und $\mathcal{N}$ abhängt.

Eine solche Abschätzung wird ebenfalls in [BS96], 4.4.24, für den globalen Interpolationsoperator auf einem regulärem Gitter bewiesen. Beachte dort auch Bermerkung 4.4.27. Hier erhält man Abschätzungen der Form

$$\|v - \mathcal{I}_{\mathcal{T}_h} v\|_s \le C h^{k+1-s} |v|_{k+1}, \tag{2.16}$$

wobei $s \le r+1$ ist und der betrachtete Finite–Elemente–Raum ein C^r-Finite–Elemente–Raum ist.

[3] Eine Menge T heißt sternenförmig bezüglich einer anderen Menge B, falls gilt: Für alle $x \in T$ ist die konvexe Hülle von $\{x\} \cup B$ Teilmenge von T. Dies ist hier keine Einschränkung, da Gitterzellen als konvex vorausgesetzt waren und daher die erste Eigenschaft des Theorems erfüllen.

2.3.4 Finite–Elemente–Methode

Es sei nun $\mathcal{T}_h$ ein vorgegebenes Gitter, $W_h := V_{\mathcal{T}_h}$ ein zugehöriger Finite–Elemente–Raum und $V_h = W_h \cap V$ ein Finite–Elemente–Raum, dessen Elemente auf dem Dirichlet–Rand Γ^D verschwinden. Weiter sei $u_{b,h} \in W_h$ eine Finite–Elemente–Funktion, welche die Randdaten u_b approximiert. Dann lautet die zu lösende Aufgabe: Finde $u_h \in W_h$, sodass $u_h - u_{b,h} \in V_h$ ist und für alle $v \in V_h$ gilt

$$a(u_h, v_h) = l(v_h). \tag{2.17}$$

Ein entscheidender Vorteil der Finiten Elemente Methode ist ihre große Flexibilität. Man kann auch schwierige Gebiete mit (ausreichend vielen) Gitterzellen triangulieren, wobei sich verschiedene geometrische Formen (z. B. Drei- und Vierecke), Polynomgrade und Größen von Gitterzellen kombinieren lassen.

2.4 Streamline Upwind Petrov-Galerkin–Methode

Die „streamline upwind Petrov-Galerkin“–Methode (SUPG) ist das am weitesten verbreitete Verfahren zur Stabilisierung konvektionsdominanter Probleme vom Typ (2.2). Gleichermaßen wird auch der Name „Streamline Diffusion Finite Element Method“ (SDFEM) beziehungsweise Stromlinien–Diffusions–Methode verwendet.

In konvektionsdominanten Anwendungen stellt sich das numerische Lösen als äußerst schwierig heraus. Lösungen haben oft Grenzschichten von der Dicke ε. Diese können, ganz besonders in mehreren Dimensionen, nicht mehr mit vertretbarem Aufwand von einem Gitter aufgelöst werden. Standard–Galerkin–Diskretisierungen neigen zu starken unphysikalischen Oszillationen, siehe beispielsweise Abbildung 5.3 auf Seite 51 im vorletzten Kapitel. Man versucht auf vielfältige Weise Lösungsstrategien zu entwickeln, die ein solches Verhalten nicht zeigen.

Die meisten dieser Strategien entstanden aus der Beobachtung heraus, dass man das Problem gut lösen kann, wenn nur genug Diffusion vorhanden ist. Man fügt sogenannte künstliche Diffusion hinzu und löst ein anderes Problem mit neuer Diffusion $\tilde{\varepsilon} = \varepsilon + h$, wobei h eine Größe ist, die vom Gitter aufgelöst werden kann, also zum Beispiel die Gitterweite. Die resultierenden Lösungen oszillieren nicht mehr, besitzen allerdings auch keine scharfe Grenzschicht, man sagt die Lösung ist verschmiert. Das Ziel vieler Methoden ist nur gerade so viel künstliche Diffusion wie nötig hinzuzufü-

gen. Oft geschieht dies anisotrop, also beispielsweise in Richtung der Konvektion $\mathbf{b}$.

Ausgangspunkt ist die Finite–Elemente–Methode auf einem Gitter $\mathcal{T}_h$. Die SUPG–Methode fügt der zu lösenden Gleichung (2.17) künstliche Diffusion hinzu, indem eine zusätzliche Bilinearform $d(y_h;\cdot,\cdot) : H^1(\Omega) \times H^1(\Omega) \to \mathbb{R}$,

$$d(y_h; u_h, v_h) = \sum_{T\in\mathcal{T}_h} \int_T y_T \left(-\varepsilon\Delta u_h + \mathbf{b}\cdot\nabla u_h + cu_h\right)(\mathbf{b}\cdot\nabla v_h)\, d\mathbf{x}, \tag{2.18}$$

addiert wird. Zur besseren Übersicht wurde die Abhängigkeit der Funktionen u_h, $\mathbf{b}$, c und v_h vom Ort $\mathbf{x}$ nicht explizit mit aufgeführt. Hier ist $y_h \in Y_h \subset L^\infty(\Omega)$ eine Parameterfunktion, die auf jeder Gitterzelle T konstant ist, $y_T = y_h\big|_T$. Die künstliche Diffusion (2.18) ist das Integral der linken Seite der starken Formulierung (2.2) multipliziert mit $\mathbf{b}\cdot\nabla v_h$ und der Parameterfunktion y_h. Offenbar ist der Laplace-Term Δu_h für $u_h \in H^1$ nicht wohldefiniert. Die betrachteten Finite–Elemente–Funktionen sind jedoch innerhalb einer Gitterzelle glatt, z. B. Polynome, sodass dies einzeln auf jeder Gitterzelle möglich ist. Für P^1- oder Q^1-Finite Elemente auf Rechtecken ist der Laplace-Term ohnehin Null. Um die SUPG-Methode konform[4] zu halten, wird auf der rechten Seite ebenfalls ein Term addiert. Dazu definiere die lineare Abbildung $\tilde{l}(y_h;\cdot) : V_h \to \mathbb{R}$, durch

$$\tilde{l}(y_h; v_h) = \int_\Omega y_h f(\mathbf{x})\mathbf{b}(\mathbf{x})\cdot\nabla v_h(\mathbf{x})\, d\mathbf{x} = \sum_{T\in\mathcal{T}_h}\int_T y_T f(\mathbf{x})\mathbf{b}(\mathbf{x})\cdot\nabla v_h(\mathbf{x})\, d\mathbf{x}. \tag{2.19}$$

Die SUPG–Methode ist dann: Finde $u_h \in W_h$, sodass $u_h - u_{b,h} \in V_h$ ist und für alle $v \in V_h$ gilt

$$a_{\mathrm{SD}}(y_h; u_h, v_h) = l_{\mathrm{SD}}(y_h; v_h), \tag{2.20}$$

mit

$$a_{\mathrm{SD}}(y_h; u_h, v_h) := a(u_h, v_h) + d(y_h; u_h, v_h) \quad \text{und} \quad l_{\mathrm{SD}}(y_h; v_h) := l(v_h) + \tilde{l}(y_h; v_h).$$

Dieses Verfahren hängt offenbar von der zu wählenden Parameterfunktion y_h ab. Diese sollte groß genug sein, um die Lösung zu stabilisieren, also um unphysikalische Oszillationen zu verhindern, und andererseits klein genug um Grenzschichten zu ermöglichen. Im Allgemeinen ist nicht bekannt, wie genau sie zu wählen ist. Zu große Werte von y_h, d. h. zu viel künstliche Diffusion, führen zu verschmierten Lösungen. Dann hat die Lösung keine schmalen Grenzschichten, sondern breite Übergänge ohne schnell variierende Gradienten. Ist der Stabilisierungsparameter jedoch zu klein, entstehen unphysikalische Oszillationen. Beides ist in der Praxis nicht zufriedenstellend.

[4] Zur genauen Definition von Konformität siehe Seite 24, Analysis der SUPG–Methode

Eine Standardwahl für die Parameterfunktion ist in [JK07] vorgeschlagen und lautet

$$y_T = \frac{h_T}{2p|\mathbf{b}|}\xi\left(Pe_T\right) \quad \text{mit} \quad \xi(\alpha) = \coth\alpha - \frac{1}{\alpha}, \quad Pe_T = \frac{|\mathbf{b}|h_T}{2p\varepsilon}, \tag{2.21}$$

wobei h_T der Durchmesser der Gitterzelle T in Stromrichtung $\mathbf{b}$ ist, sowie p der Polynomgrad des Finiten Elements auf T. Diese Wahl der Parameterfunktion liefert in einer Dimension knotenexakte Lösungen. Dies trifft im Allgemeinen jedoch nicht in mehreren Dimensionen zu. In der Literatur findet man für y_T auch oft die Bezeichnung SD–Parameter.

Bermerkung 2.4.1: Die SUPG–Methode fügt der Gleichung nur in Stromlinienrichtung Diffusion hinzu. Dies sieht man in zwei Raumdimensionen, indem man zu $\mathbf{b} = (b_1, b_2)^\top \neq 0$ ein dazu senkrechtes Vektorfeld $\mathbf{b}^\perp := (-b_2, b_1)^\top$ definiert[5] und den Gradienten einer Funktion u in zwei Anteile aufteilt:

$$\nabla u = \frac{1}{\|\mathbf{b}\|^2}\left(\left(\mathbf{b}\cdot\nabla u\right)\mathbf{b} + \left(\mathbf{b}^\top\cdot\nabla u\right)\mathbf{b}^\perp\right).$$

Der Anteil

$$\sum_{T\in\mathcal{T}_h} y_T\left(\mathbf{b}\cdot\nabla u_h, \mathbf{b}\cdot\nabla v_h\right)_0$$

der zusätzlichen Bilinearform $d(y_h; u_h, v_h)$ verhält sich also wie künstliche Diffusion, allerdings nur in Richtung der Konvektion. Die anderen Anteile von $d(y_h;\cdot,\cdot)$ sowie die Änderung der rechten Seite $\tilde{l}(y_h;\cdot)$ dienen lediglich der Konformität der SUPG–Methode.

Im Spezialfall $c = 0$ und stückweise linearen Finiten Elementen auf Dreiecken gilt $\Delta u_h = 0$ und damit

$$a_{\mathrm{SD}}(y_h; u_h, v_h) = \varepsilon\left(\nabla u_h, \nabla v_h\right)_0 + \left(\mathbf{b}\cdot\nabla u_h, v_h\right)_0 + \sum_{T\in\mathcal{T}_h} y_T\left(\mathbf{b}\cdot\nabla u_h, \mathbf{b}\cdot\nabla v_h\right)_{0,T}.$$

Diese Bilinearform entspricht dem Differentialoperator $-\varepsilon\Delta u - y_h\|\mathbf{b}\|^2 u_{\mathbf{bb}} + \mathbf{b}\cdot\nabla u$ mit $u_{\mathbf{bb}} = \frac{\partial^2 u}{\partial\mathbf{b}^2}$.

[5] Das Vektorfeld $\mathbf{b}^\perp$ wird auch „crosswind direction“ genannt.

2.4.1 Analysis der SUPG–Methode

Existenz und Eindeutigkeit der Lösung

Dieser Abschnitt folgt [RST08, III.3.2.1].

Die SUPG-Methode ist konform in folgendem Sinne: Angenommen u sei für alle $T \in \mathcal{T}_h$ eine hinreichend reguläre Lösung der starken Formulierung

$$-\varepsilon\Delta u + \mathbf{b}\cdot\nabla u + cu = f \text{ in } L^2(T). \tag{2.22}$$

Dann gilt $a_{\mathrm{SD}}(y_h; u, v_h) = l_{\mathrm{SD}}(y_h; v_h)$ für alle $v_h \in V_h$. Die SUPG-Lösung u_h erfüllt ebenfalls diese Gleichung, sodass für alle $v_h \in V_h$ gilt

$$a_{\mathrm{SD}}(y_h; u - u_h, v_h) = 0. \tag{2.23}$$

Dies nennt man *Galerkin–Orthogonalität* oder auch Projektionseigenschaft und erleichtert die Analysis der SUPG–Methode.

Es sei nun $\mathcal{T}_h$ ein reguläres Gitter zu $\Omega \subset \mathbb{R}^d$ gemäß Definition 2.3.4. Weiter sei $W_h = W_{\mathcal{T}_h} \subset H^1(\Omega)$ ein Finite–Elemente–Raum auf $\mathcal{T}_h$ der aus stückweise polynomialen Funktionen besteht. Dann gibt es eine Konstante c_{inv}, sodass für alle $v_h \in W_h$ und alle $T \in \mathcal{T}_h$ die inverse Ungleichung

$$\|\Delta v\|_{0,T} \leq \frac{c_{\mathrm{inv}}}{h_T} |v|_{1,T} \tag{2.24}$$

gilt, wobei c_{inv} nicht von $h_T = \operatorname{diam} T$ abhängt, siehe zum Beispiel [Bra07, Lemma II.6.8].

Die eindeutige Lösbarkeit der SUPG–Methode wird das Theorem von Lax–Milgram, 2.2.4, liefern. Die Bilinearform muss demnach stetig und koerziv sein sowie die rechte Seite beschränkt. Um dies zu beweisen, wird der betrachtete Raum mit einer anderen Norm, der sogenannten SUPG– oder SD–Norm, versehen,

$$|\!|\!| v |\!|\!|_{\mathrm{SD}} := \left(\varepsilon |v|_1^2 + c_0 \|v\|_0^2 + \sum_{T\in\mathcal{T}_h} y_T \|\mathbf{b}\cdot\nabla v\|_{0,T}^2 \right)^{1/2}. \tag{2.25}$$

Hierbei ist $0 < c_0 \leq c - \frac{1}{2}\nabla\cdot\mathbf{b}$, siehe (2.4). Offenbar ist diese Norm sowohl abhängig von den Koeffizienten der Differentialgleichung, als auch von den Parametern y_T. Sind

diese richtig gewählt, können die Voraussetzungen des Theorems von Lax–Milgram gezeigt werden:

Lemma 2.4.2: *Auf jeder Gitterzelle $T \subset \mathcal{T}_h$ erfülle die Parameterfunktion y_h die Ungleichungen*

$$0 < y_T \leq \frac{1}{2} \min\left(\frac{c_0}{c_T^2}, \frac{h_T^2}{\varepsilon c_{inv}^2}\right), \tag{2.26}$$

mit $c_T := \|c\|_{L^\infty(T)}$. Dann ist die Bilinearform a_{SD} koerziv bezüglich der SUPG–Norm, das heißt für alle $v_h \in W_h$ gilt

$$a_{SD}(y_h; v_h, v_h) \geq \frac{1}{2} |\!|\!| v_h |\!|\!|_{SD}^2.$$

Bermerkung 2.4.3: Wird eine P^1- oder Q^1–Finite–Elementen–Lösung auf Rechtecken gesucht, verschwindet der Laplace-Term und (2.24) ist trivialer Weise erfüllt. In diesem Fall wird nur $0 < y_T \leq c_0/(2c_T^2)$ statt (2.26) gefordert.

Beweis von Lemma 2.4.2. Sei $v_h \in W_h$. Dann gilt mit der Koerzivität der Bilinearform a bezüglich $|\cdot|_1$

$$\begin{aligned} a_{\mathrm{SD}}(y_h; v_h, v_h) &= a(v_h, v_h) + d(y_h; v_h, v_h) \\ &\geq \varepsilon \, |v_h|_1^2 + c_0 \, \|v_h\|_0^2 \\ &\quad + \sum_{T \in \mathcal{T}_h} y_T \, \|\mathbf{b} \cdot \nabla v_h\|_{0,T}^2 + \sum_{T \in \mathcal{T}_h} y_T \, (-\varepsilon \Delta v_h + c v_h, \mathbf{b} \cdot \nabla v_h)_{0,T} \\ &= |\!|\!| v_h |\!|\!|_{\mathrm{SD}}^2 + \sum_{T \in \mathcal{T}_h} y_T \, (-\varepsilon \Delta v_h + c v_h, \mathbf{b} \cdot \nabla v_h)_{0,T} \, . \end{aligned}$$

Der letzte Term ist beschränkt durch die SUPG–Norm von v_h:

$$\begin{aligned}
&\left|\sum_{T\in\mathcal{T}_h} y_T\left(-\varepsilon\Delta v_h + cv_h, \mathbf{b}\cdot\nabla v_h\right)_{0,T}\right| \\
&\qquad\leq \sum_{T\in\mathcal{T}_h} y_T\left(\varepsilon\left\|\Delta v_h\right\|_{0,T}\left\|\mathbf{b}\cdot\nabla v_h\right\|_{0,T} + \left\|cv_h\right\|_{0,T}\left\|\mathbf{b}\cdot\nabla v_h\right\|_{0,T}\right) \\
&\qquad\leq \sum_{T\in\mathcal{T}_h} y_T\left(\varepsilon^2\left\|\Delta v_h\right\|_{0,T}^2 + c_T^2\left\|v_h\right\|_{0,T}^2 + \frac{1}{2}\left\|\mathbf{b}\cdot\nabla v_h\right\|_{0,T}^2\right) \\
&\qquad\overset{(2.24)}{\leq} \sum_{T\in\mathcal{T}_h} y_T\left(\varepsilon^2\frac{c_{\text{inv}}^2}{h_T^2}\left|v_h\right|_{1,T}^2 + c_T^2\left\|v_h\right\|_{0,T}^2 + \frac{1}{2}\left\|\mathbf{b}\cdot\nabla v_h\right\|_{0,T}^2\right) \\
&\qquad\leq \sum_{T\in\mathcal{T}_h}\left(\frac{\varepsilon}{2}\left|v_h\right|_{1,T}^2 + \frac{c_0}{2}\left\|v\right\|_{0,T}^2 + \frac{1}{2}y_T\left\|\mathbf{b}\cdot\nabla v_h\right\|_{0,T}^2\right) \\
&\qquad= \frac{1}{2}|\!|\!| v_h |\!|\!|_{\text{SD}},
\end{aligned}$$

wobei in der letzten Abschätzung die Voraussetzung des Lemmas verwendet wurde: $y_T\varepsilon^2\frac{c_{\text{inv}}^2}{h_T^2} \leq \frac{1}{2}\varepsilon$ und $y_T c_T^2 \leq \frac{1}{2}c_0$. Insgesamt folgt die Behauptung. □

Lemma 2.4.4: *Es seien die Voraussetzungen des vorherigen Lemmas, 2.4.2, erfüllt. Dann sind die Bilinearform $a_{SD}(y_h;\cdot,\cdot)$ und die rechte Seite l_{SD} stetig, das heißt es gibt eine Konstante C, sodass für alle v_h, $w_h \in W_h$ gilt*

$$a_{SD}(y_h; v_h, w_h) \leq C|\!|\!| v_h |\!|\!|_{SD}|\!|\!| w_h |\!|\!|_{SD}, \qquad l_{SD}(v_h) \leq C|\!|\!| v_h |\!|\!|_{SD}.$$

Beweis. Die beiden Anteile $a_{\text{SD}}(y_h;\cdot,\cdot) = a(\cdot,\cdot) + d(y_h;\cdot,\cdot)$ werden separat abgeschätzt. Es seien v_h, $w_h \in W_h$. Dann gilt mit $\sqrt{\varepsilon}\left|v_h\right|_1 \leq |\!|\!| v_h |\!|\!|_{\text{SD}}$ und $\left\|v_h\right\|_0 \leq C|\!|\!| v_h |\!|\!|_{\text{SD}}$

$$\begin{aligned}
a(v_h, w_h) &= \varepsilon\left(\nabla v_h, \nabla w_h\right)_0 + \left(\mathbf{b}\cdot\nabla v_h, w_h\right)_0 + \left(cv_h, w_h\right)_0 \\
&\leq |\!|\!| v_h |\!|\!|_{\text{SD}}|\!|\!| w_h |\!|\!|_{\text{SD}} + \left\|\mathbf{b}\cdot\nabla v_h\right\|_0\left\|w_h\right\|_0 + \left\|c\right\|_\infty\left\|v_h\right\|_0\left\|w_h\right\|_0 \\
&\leq |\!|\!| v_h |\!|\!|_{\text{SD}}|\!|\!| w_h |\!|\!|_{\text{SD}} + C\left\|\mathbf{b}\cdot\nabla v_h\right\|_0|\!|\!| w_h |\!|\!|_{\text{SD}} + C|\!|\!| v_h |\!|\!|_{\text{SD}}|\!|\!| w_h |\!|\!|_{\text{SD}} \\
&\leq C|\!|\!| v_h |\!|\!|_{\text{SD}}|\!|\!| w_h |\!|\!|_{\text{SD}},
\end{aligned}$$

da $\left\|\mathbf{b}\cdot\nabla v_h\right\|_0 \leq \left\|\mathbf{b}\right\|_\infty\varepsilon^{-1/2}|\!|\!| v_h |\!|\!|_{\text{SD}}$ gilt. Man kann den Term $\left\|\mathbf{b}\cdot\nabla v_h\right\|_0$ auch anders

abschätzen:

$$\begin{aligned}\|\mathbf{b}\cdot\nabla v_h\|_0 &= \sum_{T\in\mathcal{T}_h}\frac{\sqrt{y_T}}{\sqrt{y_T}}\|\mathbf{b}\cdot\nabla v_h\|_{0,T}\\ &\leq \max_{T\in\mathcal{T}_h}\left\{\frac{1}{\sqrt{y_T}}\right\}\sum_{T\in\mathcal{T}_h}\sqrt{y_h\|\mathbf{b}\cdot\nabla v_h\|_{0,T}^2}\\ &\leq C\left(\sum_{T\in\mathcal{T}_h} y_h\|\mathbf{b}\cdot\nabla v_h\|_{0,T}^2\right)^{1/2}\\ &\leq C|\!|\!|v_h|\!|\!|_{\mathrm{SD}}.\end{aligned}$$

Daher hängt die Konstante C von $1/c_0$ und $\min\left\{\|\mathbf{b}\|_\infty\varepsilon^{-1/2}, \max_{T\in\mathcal{T}_h}\left\{y_T^{-1/2}\right\}\right\}$ ab. Der Stabilisierungsterm d lässt sich mit Hilfe der Voraussetzungen ebenfalls abschätzen:

$$\begin{aligned}d(y_h;v_h,w_h) &\leq \left|\sum_{T\in\mathcal{T}_h} y_T\left(-\varepsilon\Delta v_h+\mathbf{b}\cdot\nabla v_h+cv_h,\mathbf{b}\cdot\nabla w_h\right)_{0,T}\right|\\ &\leq \left[\left(\sum_{T\in\mathcal{T}_h} y_T\varepsilon^2\|\Delta v_h\|_{0,T}^2\right)^{1/2}+\left(\sum_{T\in\mathcal{T}_h} y_T\|\mathbf{b}\cdot\nabla v_h\|_{0,T}^2\right)^{1/2}\right.\\ &\quad\left.+\left(\sum_{T\in\mathcal{T}_h} y_T c_T^2\|v_h\|_{0,T}^2\right)^{1/2}\right]\left(\sum_{T\in\mathcal{T}_h} y_T\|\mathbf{b}\cdot\nabla w_h\|_{0,T}^2\right)^{1/2}.\end{aligned}$$

Die vier Wurzelterme lassen sich gegen die jeweilige SUPG–Norm abschätzen. Der erste und dritte Term werden genau so wie im vorherigen Lemma jeweils gegen $\frac{1}{2}|\!|\!|v_h|\!|\!|_{\mathrm{SD}}$ unter Verwendung von (2.24) und (2.26) abgeschätzt. Die Terme, die die Konvektion $\mathbf{b}$ beinhalten, werden direkt gegen die SUPG–Norm abgeschätzt. Damit folgt

$$d(y_h;v_h,w_h)\leq 2|\!|\!|v_h|\!|\!|_{\mathrm{SD}}|\!|\!|w_h|\!|\!|_{\mathrm{SD}}.$$

Die Stetigkeit der rechten Seite l_{SD} wird ähnlich gezeigt:

$$\begin{aligned} l_{\mathrm{SD}}(v_h) &= (f, v_h)_0 + \sum_{T\in\mathcal{T}_h} y_T \, (f, \mathbf{b}\cdot\nabla v_h)_{0,T} \\ &\leq \|f\|_0 \, \|v_h\|_0 + \|f\|_0 \, \|y_h\|_\infty \left(\sum_{T\in\mathcal{T}_h} y_T \, \|\mathbf{b}\cdot\nabla v_h\|_{0,T}^2 \right)^{1/2} \\ &\leq C \, \|f\|_0 \, |\!|\!| v_h |\!|\!|_{\mathrm{SD}}. \end{aligned}$$

Die Konstante C hängt zusätzlich von $\|y_h\|_\infty$ ab. □

Werden die beiden vorherigen Lemmata kombiniert, so folgt für die Lösung u_h von (2.20) wegen

$$|\!|\!| u_h |\!|\!|_{\mathrm{SD}}^2 \leq 2a_{\mathrm{SD}}(y_h; u_h, u_h) = 2l_{\mathrm{SD}}(u_h) \leq 2C \, \|f\|_0 \, |\!|\!| u_h |\!|\!|_{\mathrm{SD}}$$

die *a priori* Abschätzung

$$|\!|\!| u_h |\!|\!|_{\mathrm{SD}} \leq 2C \, \|f\|_0 \, .$$

Fehlerabschätzung, Konvergenzordnung

Dieser Abschnitt folgt [RST08] Seite 305 und 306.

Es soll nun der Fehler $|\!|\!| u - u_h |\!|\!|_{\mathrm{SD}}$ zwischen der Lösung u von (2.6) und u_h von (2.20) untersucht werden. Dabei werden die beiden Terme $|\!|\!| \mathcal{I}_{\mathcal{T}_h} u - u_h |\!|\!|_{\mathrm{SD}}$ und $|\!|\!| u - I_{\mathcal{T}_h} u |\!|\!|_{\mathrm{SD}}$ einzeln abgeschätzt.

Lemma 2.4.5: *Es seien u die Lösung von (2.6) und u_h die Lösung von (2.20) auf einem regulären Gitter $\mathcal{T}_h$. Dann gilt unter der zusätzlichen Annahme $u \in H^{k+1}(\Omega)$*

$$|\!|\!| \mathcal{I}_{\mathcal{T}_h} u - u_h |\!|\!|_{SD} \leq C \left(\sum_{T\in\mathcal{T}_h} \left(\varepsilon + y_T + (h_T + h_T^2)(1 + y_T) + h_T^2 y_T^{-1} \right) h_T^{2k} \, |u|_{k+1,T}^2 \right)^{1/2} . \tag{2.27}$$

Beweis. Lemma 2.4.2 und die Galerkin Orthogonalität (2.23) liefern

$$\frac{1}{2} |\!|\!| \mathcal{I}_{\mathcal{T}_h} u - u_h |\!|\!|_{\mathrm{SD}}^2 \leq a_{\mathrm{SD}}(y_h; \mathcal{I}_{\mathcal{T}_h} u - u_h, \mathcal{I}_{\mathcal{T}_h} u - u_h) = a_{\mathrm{SD}}(y_h; \mathcal{I}_{\mathcal{T}_h} u - u, \mathcal{I}_{\mathcal{T}_h} u - u_h).$$

Die einzelnen Terme der rechten Seite werden nun separat behandelt:

$$\begin{aligned}\varepsilon\left(\nabla(\mathcal{I}_{\mathcal{T}_h}u-u),\nabla(\mathcal{I}_{\mathcal{T}_h}u-u_h)\right)_0 &\leq \varepsilon\left|\mathcal{I}_{\mathcal{T}_h}u-u\right|_1\left|\mathcal{I}_{\mathcal{T}_h}u-u_h\right|_1\\ &\leq \varepsilon^{1/2}\left|\mathcal{I}_{\mathcal{T}_h}u-u\right|_1\left|\!\left|\!\left|\mathcal{I}_{\mathcal{T}_h}u-u_h\right|\!\right|\!\right|_{\mathrm{SD}}\\ &\overset{(2.16)}{\leq} C\varepsilon^{1/2}h^k\left|u\right|_{k+1}\left|\!\left|\!\left|\mathcal{I}_{\mathcal{T}_h}u-u_h\right|\!\right|\!\right|_{\mathrm{SD}},\end{aligned}$$

$$\begin{aligned}&(\mathbf{b}\cdot\nabla(\mathcal{I}_{\mathcal{T}_h}u-u)+c(\mathcal{I}_{\mathcal{T}_h}u-u),\mathcal{I}_{\mathcal{T}_h}u-u_h)_0\\ &\quad=((c-\nabla\cdot\mathbf{b})(\mathcal{I}_{\mathcal{T}_h}u-u),\mathcal{I}_{\mathcal{T}_h}u-u_h)_0-(\mathcal{I}_{\mathcal{T}_h}u-u,\mathbf{b}\cdot\nabla(\mathcal{I}_{\mathcal{T}_h}u-u_h))_0\\ &\quad\leq\left\|c-\nabla\cdot\mathbf{b}\right\|_\infty\left\|\mathcal{I}_{\mathcal{T}_h}u-u\right\|_0\left\|\mathcal{I}_{\mathcal{T}_h}u-u_h\right\|_0\\ &\qquad+\sum_{T\in\mathcal{T}_h}\left\|\mathcal{I}_{\mathcal{T}_h}u-u\right\|_{0,T}\left\|\mathbf{b}\cdot\nabla\left(\mathcal{I}_{\mathcal{T}_h}u-u_h\right)\right\|_{0,T}\\ &\quad\leq C\left(\sum_{T\in\mathcal{T}_h}\left\|\mathcal{I}_{\mathcal{T}_h}u-u\right\|_{0,T}^2\right)^{1/2}\left|\!\left|\!\left|\mathcal{I}_{\mathcal{T}_h}u-u_h\right|\!\right|\!\right|_{\mathrm{SD}}\\ &\qquad+\left(\sum_{T\in\mathcal{T}_h}y_T^{-1}\left\|\mathcal{I}_{\mathcal{T}_h}u-u\right\|_{0,T}^2\right)^{1/2}\left(\sum_{T\in\mathcal{T}_h}y_T\left\|\mathbf{b}\cdot\nabla\left(\mathcal{I}_{\mathcal{T}_h}u-u_h\right)\right\|_{0,T}^2\right)^{1/2}\\ &\quad\overset{(2.15)}{\leq} Ch^k\left(\sum_{T\in\mathcal{T}_h}h_T^2\left(1+y_T^{-1}\right)\left|u\right|_{k+1,T}^2\right)^{1/2}\left|\!\left|\!\left|\mathcal{I}_{\mathcal{T}_h}u-u_h\right|\!\right|\!\right|_{\mathrm{SD}}.\end{aligned}$$

Die Terme, die von der Stabilisierung herrühren, werden ebenfalls mit der Interpo-

lationsfehlerungleichung (2.15) abgeschätzt.

$$
\begin{aligned}
&\sum_{T\in\mathcal{T}_h} y_T \left(-\varepsilon\Delta\left(\mathcal{I}_{\mathcal{T}_h}u-u\right)+\mathbf{b}\cdot\nabla\left(\mathcal{I}_{\mathcal{T}_h}u-u\right)+c\left(\mathcal{I}_{\mathcal{T}_h}u-u\right),\mathbf{b}\cdot\nabla\left(\mathcal{I}_{\mathcal{T}_h}u-u_h\right)\right)_{0,T}\\
&\leq C\sum_{T\in\mathcal{T}_h} y_T^{1/2}\left(\varepsilon h_T^{k-1}+h_T^k+h_T^{k+1}\right)|u|_{k+1,T}\, y_T^{1/2}\left\|\mathbf{b}\cdot\nabla\left(\mathcal{I}_{\mathcal{T}_h}u-u_h\right)\right\|_{0,T}\\
&\leq C\left(\sum_{T\in\mathcal{T}_h} y_T h_T^{2k}\left(\varepsilon h_T^{-1}+1+h_T\right)^2|u|_{k+1,T}^2\right)^{1/2}\\
&\quad\times\left(\sum_{T\in\mathcal{T}_h} y_T\left\|\mathbf{b}\cdot\nabla\left(\mathcal{I}_{\mathcal{T}_h}u-u_h\right)\right\|_{0,T}^2\right)^{1/2}\\
&\leq C\left(\sum_{T\in\mathcal{T}_h} y_T h_T^{2k}\left(\varepsilon h_T^{-1}+1+h_T\right)^2|u|_{k+1,T}^2\right)^{1/2}\left|\!\left|\!\left|\mathcal{I}_{\mathcal{T}_h}u-u_h\right|\!\right|\!\right|_{\mathrm{SD}}\\
&\leq C\left(\sum_{T\in\mathcal{T}_h} h_T^{2k}\left(\varepsilon+y_T+h_T(1+y_T)+h_T^2(1+y_T)\right)|u|_{k+1,T}^2\right)^{1/2}\left|\!\left|\!\left|\mathcal{I}_{\mathcal{T}_h}u-u_h\right|\!\right|\!\right|_{\mathrm{SD}}.
\end{aligned}
$$

In der letzten Ungleichung wurde $\varepsilon y_T \leq Ch_T^2$ verwendet, siehe Gleichung (2.26). Werden alle bisherigen Abschätzungen wieder zusammen geführt, erhält man die Behauptung

$$
\left|\!\left|\!\left|\mathcal{I}_{\mathcal{T}_h}u-u_h\right|\!\right|\!\right|_{\mathrm{SD}} \leq C\left(\sum_{T\in\mathcal{T}_h} h_T^{2k}\left(\varepsilon+y_T+(h_T+h_T^2)(1+y_T)+h_T^2y_T^{-1}\right)|u|_{k+1,T}^2\right)^{1/2}.
$$

□

Die beste mögliche Konvergenzrate erreicht man, indem die Terme ε, y_T, $y_T^{-1}h_T^2$ und $(h_T+h_T^2)(1+y_T)$ gegeneinander ausbalanciert werden. Dies wird durch folgende Wahl der Parameter auf jeder Gitterzelle $T\in\mathcal{T}_h$ erreicht,

$$
y_T=\begin{cases} y_0h_T & \text{falls } Pe_T>1\\ y_1h_T^2/\varepsilon & \text{falls } Pe_T\leq 1,\end{cases} \tag{2.28}
$$

mit der lokalen Péclet–Zahl

$$
Pe_T:=\frac{\|\mathbf{b}\|_{\infty,T}\,h_T}{2\varepsilon},
$$

die angibt, ob das Problem konvektionsdominant ($Pe_T > 1$) oder diffusionsdominant ($Pe_T \leq 1$) ist. Die Koeffizienten y_0 und y_1 sind vom Nutzer zu wählen.

Theorem 2.4.6:
Seien die Voraussetzungen des vorherigen Lemmas erfüllt, die Parameterfunktion y_h wie in (2.28) gewählt sowie die Gitterweite hinreichend klein, $h < 1$. Dann gilt

$$\|\!|u - u_h|\!\|_{SD} \leq C \left(\varepsilon^{1/2} + h^{1/2}\right) h^k \left|u\right|_{k+1}. \tag{2.29}$$

Die Konstante C hängt nicht von ε oder dem Gitter ab.

Beweis. Zunächst wird der Fehler $\|\!|u - u_h|\!\|_{\mathrm{SD}}$ in die Anteile der Interpolation und der Diskretisierung unterteilt,

$$\|\!|u - u_h|\!\|_{\mathrm{SD}} \leq \|\!|u - \mathcal{I}_{\mathcal{T}_h} u|\!\|_{\mathrm{SD}} + \|\!|\mathcal{I}_{\mathcal{T}_h} u - u_h|\!\|_{\mathrm{SD}}.$$

Den Diskretisierungsfehler $\|\!|\mathcal{I}_{\mathcal{T}_h} u - u_h|\!\|_{\mathrm{SD}}$ wird mit Hilfe des vorherigen Lemmas beschränkt. Hierzu werden die Parameter y_T gemäß (2.28) gewählt und die höheren Terme in h_T durch h_T abgeschätzt, da $h_T < 1$. Dann folgt

$$\|\!|\mathcal{I}_{\mathcal{T}_h} u - u_h|\!\|_{\mathrm{SD}} \leq C \left(\sum_{T \in \mathcal{T}_h} h_T^{2k} \left(\varepsilon + h_T\right) \left|u\right|_{k+1,T}^2 \right)^{1/2} \leq C h^k \left(\varepsilon^{1/2} + h_T^{1/2}\right) \left|u\right|_{k+1}.$$

Der Interpolationsfehler wird mit (2.16) und (2.15) abgeschätzt,

$$\begin{aligned}
&\|\!|u - \mathcal{I}_{\mathcal{T}_h} u|\!\|_{\mathrm{SD}} \\
&\qquad \leq \left(\varepsilon \left|u - \mathcal{I}_{\mathcal{T}_h} u\right|_1^2 + c_0 \left\|u - \mathcal{I}_{\mathcal{T}_h} u\right\|_0^2 + \sum_{T \in \mathcal{T}_h} y_T \left\|\mathbf{b} \cdot \nabla \left(u - \mathcal{I}_{\mathcal{T}_h} u\right)\right\|_{0,T}^2 \right)^{1/2} \\
&\qquad \leq C \left(\varepsilon h^{2k} \left|u\right|_{k+1}^2 + c_0 h^{2(k+1)} \left|u\right|_{k+1}^2 + \sum_{T \in \mathcal{T}_h} y_T h^{2k} \left\|\mathbf{b}\right\|_{\infty,T} \left|u\right|_{k+1,T}^2 \right)^{1/2} \\
&\qquad \leq C h^k \left(\varepsilon^{1/2} + h_T^{1/2}\right) \left|u\right|_{k+1}.
\end{aligned}$$

Hierbei wurde $y_T \leq C h_T$ und wieder $h_T^2 < h_T$ verwendet. Zusammen ergibt dies die Behauptung. □

Bermerkung 2.4.7: Die Standardwahl (2.21) ist von der Form (2.28). In Abbildung 2.3 wurde $|b| = p = 1$, $\varepsilon = 1/500$ gesetzt. Im Allgemeinen sind h_T, Pe_T und damit

auch y_h abhängig vom Ort $\mathbf{x}$.

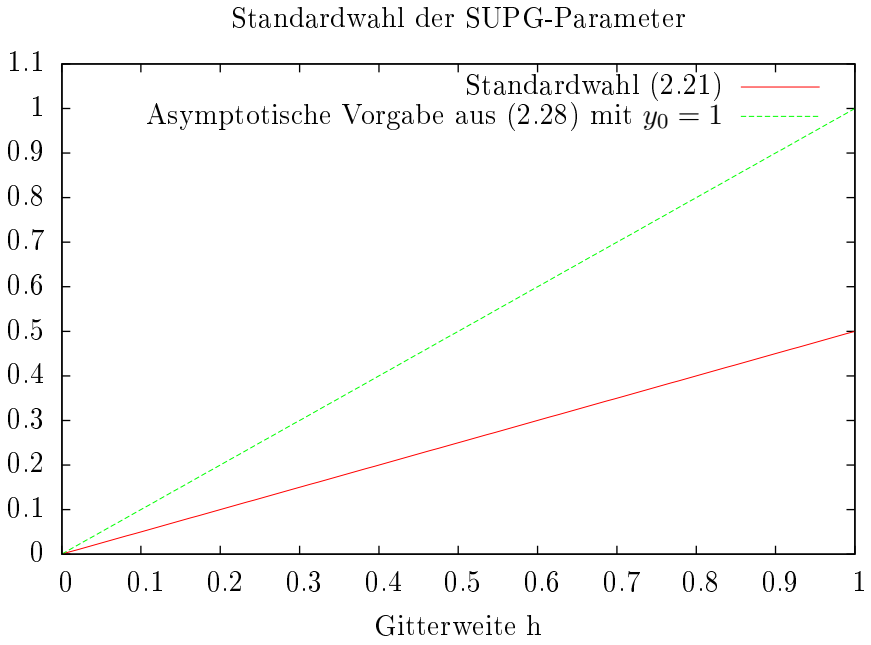


Abbildung 2.3: Die Standardwahl der SUPG-Parameter y_h aus (2.21) für verschiedene Gitterweiten h und $|b| = p = 1$ sowie $\varepsilon = 1/500$.

3 A Posteriori Parameterbestimmung

3.1 Allgemeine Herangehensweise

Die Standard-Wahl (2.21) der SUPG–Parameter hängt nicht von der Lösung ab. Das heißt auch, dass das lokale Verhalten der Lösung, z. B. Grenzschichten, unberücksichtigt bleibt. Im Fall eines gleichmäßigen Gitters und konstanter Konvektion sind beispielsweise alle Parameter gleich. Das ist nicht angebracht. Vielmehr scheint es sinnvoll, die Parameter der SUPG–Methode abhängig von einer berechneten Lösung zu wählen. Dazu soll nun die Lösung der SUPG–Methode als eine Funktion der Parameter gesehen werden. Das Ziel wird sein, diese Lösung innerhalb einer Menge zulässiger Parameter zu optimieren. Dies geschieht bezüglich eines Funktionals, welches die Qualität der Lösung misst oder abschätzt und noch gewählt werden muss.

Die Menge der zulässigen Parameter sei $D_h \subset Y_h \cong \mathbb{R}^N$, wobei N die Anzahl der Gitterzellen des Gitters $\mathcal{T}_h$ ist. Um die Koerzivität und damit die eindeutige Lösbarkeit der SUPG–Methode zu erhalten, wird

$$D_h := \prod_{T \in \mathcal{T}_h} \left[0, \frac{1}{2} \min \left(\frac{c_0}{c_T^2}, \frac{h_T^2}{\varepsilon c_{\text{inv}}^2} \right) \right]$$

gesetzt, vergleiche Lemma 2.4.2. Es sei $S_h : D_h \to V_h$ der Operator, der das SUPG–Problem (2.20) löst und anschließend die Randdaten subtrahiert. Das heißt zu gegebenen Parametern $y_h \in D_h$ löst die Finite–Elemente–Funktion $u_h = S_h(y_h) + u_{b,h}$ die Gleichung (2.20) für alle Testfunktionen $v_h \in V_h$. Nach Definition der Menge der zulässigen Parameter D_h ist dieser Operator wohldefiniert.

Es sei nun $I_h : V_h \to \mathbb{R}$ ein Funktional, das die Qualität einer diskreten Lösung u_h repräsentiert. Explizite Formeln für I_h werden später diskutiert. Das Ziel ist, einen Satz Parameter $y_h \in D_h$ zu finden, der das Funktional

$$\Phi_h(y_h) := I_h \left(S_h(y_h) \right) \tag{3.1}$$

minimiert. Offenbar ist dieses Problem nichtlinear und wird daher iterativ gelöst. Numerische Methoden zur nichtlinearen Optimierung benötigen in der Regel Informationen zu ersten Ableitungen oder sogar zur Hessematrix.

Bemerkung 3.1.1: Diese Herangehensweise ist nicht auf die SUPG–Methode beschränkt. Sie lässt sich auch auf andere Verfahren mit Parametern, die vom Nutzer bestimmt werden sollen, anwenden.

3.2 Berechnung des Gradienten

Im Folgenden sei stets angenommen, dass sowohl I_h als auch S_h Fréchet-differenzierbar sind mit Ableitungen $DI_h : V_h \to V_h'$ und $DS_h : D_h \to \mathcal{L}(Y_h, V_h)$. Dann ist auch das zu minimierende Funktional Φ_h Fréchet-differenzierbar und dessen Ableitung ergibt sich durch die Ketteregel:

$$D\Phi_h : D_h \to Y_h', \quad y_h \mapsto D\Phi_h(y_h) = DI_h\left(S_h(y_h)\right) DS_h(y_h). \tag{3.2}$$

Ist die Dimension des Raumes Y_h sehr groß, so kann diese Ableitung nicht mehr mit vertretbarem Aufwand direkt berechnet werden. Für jede Komponente des Vektors y_h müsste ein System der Größe $\dim V_h$ gelöst werden. Um die Ableitung $D\Phi_h$ effizient zu berechnen, wird stattdessen ein duales Problem gelöst werden.

Betrachte den Operator $R_h : V_h \times D_h \to V_h'$, der für alle w_h, $v_h \in V_h$ und $y_h \in D_h$ das Residuum auswertet:

$$\left(R_h\left(w_h, y_h\right)\right)(v_h) := a_{\mathrm{SD}}(y_h; w_h + u_{b,h}, v_h) - l_{\mathrm{SD}}(y_h; v_h).$$

Per Definition des Lösungsoperators S_h gilt für alle $y_h \in D_h$

$$R_h\left(S_h(y_h), y_h\right) = 0. \tag{3.3}$$

Weiter sei R_h als Fréchet differenzierbar angenommen mit partiellen Ableitungen $\partial_w R_h : V_h \times D_h \to \mathcal{L}(V_h, V_h')$ und $\partial_y R_h : V_h \times D_h \to \mathcal{L}(Y_h, V_h')$. Differenziert man nun Gleichung (3.3) nach y_h, erhält man für alle $y_h \in D_h$

$$D_y R_h\left(S_h(y_h), y_h\right) = \partial_w R_h\left(S_h(y_h), y_h\right) DS_h(y_h) + \partial_y R_h\left(S_h(y_h), y_h\right) = 0. \tag{3.4}$$

Weiterhin sei nun angenommen, dass eine Abbildung $\psi_h : D_h \to V_h$ existiert, sodass

für alle $y_h \in D_h$ und alle $\tilde{y}_h \in Y_h$ gilt

$$\left(D\Phi_h(y_h)\right)(\tilde{y}_h) = -\left[\left(\partial_y R_h\left(S_h(y_h), y_h\right)\right)(\tilde{y}_h)\right](\psi_h(y_h)). \tag{3.5}$$

Zusammen mit Gleichung (3.4) ergibt dies für alle $y_h \in D_h$ und alle $\tilde{y}_h \in Y_h$

$$\begin{aligned}\left(D\Phi_h(y_h)\right)(\tilde{y}_h) &= \left[\left(\partial_w R_h\left(S_h(y_h), y_h\right) DS_h(y_h)\right)(\tilde{y}_h)\right](\psi_h(y_h)) \\ &= \left[\left(\left(\partial_w R_h\right)'\left(S_h(y_h), y_h\right)\right)\psi_h(y_h)\right]\left(DS_h(y_h)\right)(\tilde{y}_h),\end{aligned}$$

mit dem adjungierten Operator $\left(\partial_w R_h\right)'(w_h, y_h) : V_h'' = V_h \to V_h'$ definiert für alle v_h, $\tilde{v}_h$ durch

$$\left(\left(\partial_w R_h\right)'(w_h, y_h)\, v_h\right)(\tilde{v}_h) = \left(\partial_w R_h\left(w_h, y_h\right)\tilde{v}_h\right)(v_h). \tag{3.6}$$

Eine weitere Darstellung[1] der gesuchten Ableitung $D\Phi_h$ folgt direkt aus Gleichung (3.2): Für alle $y_h \in D_h$ und alle $\tilde{y}_h \in Y_h$ gilt

$$\left(D\Phi_h(y_h)\right)(\tilde{y}_h) = \left(DI_h\left(S_h(y_h)\right) DS_h(y_h)\right)(\tilde{y}_h) = \left(DI_h\left(S_h(y_h)\right)\right)\left(DS_h(y_h)(\tilde{y}_h)\right). \tag{3.7}$$

Diese beiden Darstellungen sind identisch, falls ψ_h folgendes Problem für alle $y_h \in D_h$ löst:

$$\left(\partial_w R_h\right)'\left(S(y_h), y_h\right)\psi_h(y_h) = DI_h\left(S(y_h)\right). \tag{3.8}$$

Die Ableitung $D\Phi_h$ ist dann für alle $y_h \in D_h$ gegeben durch

$$D\Phi_h(y_h) = -\left(\partial_y R_h\right)'\left(S(y_h), y_h\right)\psi_h(y_h),$$

mit dem adjungierten Operator $\left(\partial_y R_h\right)' : V_h \times Y_h \to \mathcal{L}(V_h, Y_h')$, der für alle v_h, $w_h \in V_h$ und alle y_h, $\tilde{y}_h \in Y_h$ definiert ist als

$$\left(\left(\partial_y R_h\right)'(w_h, y_h)v_h\right)(\tilde{y}_h) = \left(\left(\partial_y R_h\right)(w_h, y_h)\tilde{y}_h\right)(v_h). \tag{3.9}$$

Beispiel 3.2.1 (Anwendung auf die SUPG–Methode)**:** Es wurde bereits der Operator $R_h : V_h \times D_h \to V_h'$ betrachtet, der das Residuum auswertet. Er ist definiert durch

$$\begin{aligned}\left(R_h\left(w_h, y_h\right)\right)(v_h) &:= a_{\mathrm{SD}}(y_h; w_h + u_{b,h}, v_h) - l_{\mathrm{SD}}(y_h; v_h) \\ &= a(w_h + u_{b,h}, v_h) + d(y_h; w_h + u_{b,h}, v_h) - l(v_h) - \tilde{l}(y_h; v_h)\end{aligned}$$

[1] Tatsächlich ist dies nur eine andere Notation, da hier (lineare) Operatoren hintereinander ausgeführt werden. Die Klammerung dient lediglich der Übersichtlichkeit.

für alle w_h, $v_h \in V_h$ und $y_h \in D_h$. Die partiellen Ableitungen $\partial_w R_h : V_h \times D_h \to \mathcal{L}(V_h, V_h')$ und $\partial_y R_h : V_h \times D_h \to \mathcal{L}(Y_h, V_h')$ von R_h sind für alle w_h, v_h, $\tilde{v}_h \in V_h$ und y_h, $\tilde{y}_h \in D_h$ gegeben durch

$$\begin{aligned}\left(\partial_w R_h\left(w_h, y_h\right)\tilde{v}_h\right)(v_h) &= a_{\mathrm{SD}}(y_h; \tilde{v}_h, v_h),\\ \left(\partial_y R_h\left(w_h, y_h\right)\tilde{y}_h\right)(v_h) &= d(\tilde{y}_h; w_h + u_{b,h}, v_h) - \tilde{l}(\tilde{y}_h; v_h),\end{aligned}$$

da sowohl a_{SD} linear in der zweiten Komponente ist als auch $d - \tilde{l}$ linear in y_h. Die Hilfsfunktion $\psi : D_h \to V_h$ löst für alle $v_h \in V_h$ die Gleichung

$$a_{\mathrm{SD}}\left(y_h; \tilde{v}_h, \psi_h(y_h)\right) = \Big(DI_h\left(S_h(y_h)\right)\Big)(v_h).$$

Damit ist die gesuchte Ableitung für alle y_h, $\tilde{y}_h \in D_h$ gegeben durch

$$\left(D\Phi_h(y_h)\right)(\tilde{y}_h) = -d\left(\tilde{y}_h; S_h(y_h), \psi_h(y_h)\right) + \tilde{l}\left(\tilde{y}_h; \psi_h(y_h)\right).$$

3.3 BFGS-Verfahren

Mit den Bezeichnungen aus den vorherigen Abschnitten lautet die Minimierungsaufgabe

$$\min_{y_h \in D_h} \Phi_h(y_h). \tag{3.10}$$

Restringierte Optimierungsaufgaben dieser Art werden numerisch iterativ gelöst. Das heißt ausgehend von einem Startwert $y_h^{(0)}$ werden weitere Iterierte $y_h^{(k)}$ berechnet, die im besten Falle gegen eine gesuchte Lösung konvergieren. Wie in Abschnitt 3.2 beschrieben, können Gradienten $D\Phi_h\big(y_h^{(k)}\big)$ berechnet werden. Die Bestimmung der Hessematrix ist jedoch zu aufwändig. In einem solchen Fall ist das Broyden–Fletcher–Goldfarb–Shanno–Verfahren (BFGS) das am meisten verwendete, weil es am Allgemeinen besser ist als alle anderen Verfahren, siehe [Luk11] und [NW06].

Als Startwert kann prinzipiell jeder Vektor $y_h \in D_h$ gewählt werden, jedoch ist es von Vorteil, Informationen über das Problem mit einfließen zu lassen. Da bereits bekannt ist, dass die nicht stabilisierte Galerkin–Methode zu unphysikalischen Oszillationen führt, ist die Wahl $y_h^{(0)} = 0$ nicht sinnvoll. Als Startwert wird stattdessen auf jeder Gitterzelle der Standardparameter aus (2.21) gesetzt. Um von einer Iterierten $y_h^{(k)}$ zur nächsten $y_h^{(k+1)}$ zu kommen, wird zunächst eine Suchrichtung $p_k \in \mathbb{R}^N$ und anschließend eine Schrittweite $\alpha_k > 0$ ermittelt. Diese Prozedur wird wiederholt, bis eine geeignete Abbruchbedingung erfüllt ist. Das allgemeine Vorgehen wird in

folgendem Algorithmus illustriert.

wähle $y_h^{(0)} \in \mathbb{R}^N$
for $k = 0, 1, \ldots$ **do**
  Bestimme Suchrichtung p_k
  Bestimme Schrittweite $\alpha_k > 0$
  setze $y_h^{(k+1)} = y_h^{(k)} + \alpha_k p_k$
  Prüfe Abbruchkriterium
end for

Algorithmus 3.1: Allgemeine Vorgehensweise

3.3.1 Abstiegsrichtung

Ausgehend von einer Iterierten $y_h^{(k)}$ ist eine Richtung p gesucht, in der das zu minimierende Funktional (lokal) abnimmt. Eine solche heißt Abstiegsrichtung und ist charakterisiert durch die Ungleichung $p^\top D\Phi_h\big(y_h^{(k)}\big) < 0$. Zu jeder positiv definiten Matrix B ist $p = -B^{-1}D\Phi_h\big(y_h^{(k)}\big)$ eine Abstiegsrichtung. Wählt man $B = \mathrm{Id}$, so erhält man das Gradientenverfahren welches im Allgemeinen sehr langsam konvergiert. Wird B als Hessematrix $D^2\Phi_h\big(y_h^{(k)}\big)$ gewählt, erhält man das lokal quadratisch konvergente Newton–Verfahren. Die Auswertung der Heesematrix ist jedoch sehr aufwändig. Stattdessen wird eine Approximation B_k verwendet, die mit jedem Iterationsschritt angepasst wird und ebenfalls positiv definit ist. Diese Verfahren werden Quasi–Newton–Verfahren genannt.

Unter der Annahme, dass Φ_h hinreichend oft differenzierbar ist, gibt es nach dem Satz von Taylor ein $t \in (0, 1)$ mit

$$\Phi_h\left(y_h^{(k)} + p\right) = \Phi_h\left(y_h^{(k)}\right) + p^\top D\Phi_h\left(y_h^{(k)}\right) + \frac{1}{2}p^\top D^2\Phi_h\left(y_h^{(k)} + tp\right)p.$$

Es sei nun B_k eine positiv definite Matrix. Sie soll die Hessematrix approximieren, wie genau wird im Abschnitt 3.3.3 behandelt. Definiere das quadratische Modell

$$m_k(p) := \Phi_h\left(y_h^{(k)}\right) + p^\top D\Phi_h\left(y_h^{(k)}\right) + \frac{1}{2}p^\top B_k p.$$

Dann ist $m_k(p) \approx \Phi_h\big(y_h^{(k)} + p\big)$ und die Richtung p, die m_k minimiert, also bei der

die Ableitung Dm_k verschwindet, ist die Abstiegsrichtung

$$p = -B_k^{-1} D\Phi_h \left(y_h^{(k)}\right). \tag{3.11}$$

3.3.2 Schrittweite

Ist die Abstiegsrichtung p_k gemäß (3.11) gefunden, soll die Schrittweite $\alpha > 0$ so bestimmt werden, dass

$$\varphi(\alpha) := \Phi_h \left(y_h^{(k)} + \alpha p_k\right) \tag{3.12}$$

minimiert wird. Dies ist ein eindimensionales Problem. Trotzdem ist das Finden des globalen Minimierers von (3.12) zu aufwändig, das heißt die Funktion Φ_h und ihre Ableitung werden zu oft ausgewertet. Stattdessen wird eine Schrittweite gesucht, die zu einer angemessenen Reduktion der Zielfunktion Φ_h führt. Um zu erreichen, dass diese Reduktion nicht zu klein ist, verlangt man

$$\varphi(\alpha) = \Phi_h \left(y_h^{(k)} + \alpha p_k\right) \leq \Phi_h \left(y_h^{(k)}\right) + c_1 \alpha D\Phi_h \left(y_h^{(k)}\right)^\top p_k, \tag{3.13}$$

wobei $c_1 \in (0,1)$ eine zu wählende Konstante ist. Dies ist jedoch für alle hinreichend kleinen Schrittweiten α erfüllt. Da zu kleine Schrittweiten sehr teuer sind, soll dies durch eine weitere Bedingung verhindert werden,

$$\varphi'(\alpha) = D\Phi_h \left(y_h^{(k)} + \alpha p_k\right)^\top p_k \geq c_2 D\Phi_h \left(y_h^{(k)}\right)^\top p_k = c_2 \varphi'(0). \tag{3.14}$$

Die Konstante $c_2 \in (c_1, 1)$ ist wieder zu wählen. Bedingung (3.14) besagt also, dass die Steigung von φ bei α bis auf eine Konstante c_2 größer ist als bei Null. Typische Werte für die Konstanten c_1 und c_2 sind $c_1 = 10^{-4}$ und $c_2 = 0.9$. Die beiden Forderungen (3.13) und (3.14) werden die *Wolfe*–Bedingungen genannt. Das ursprüngliche Ziel war, die Schrittweite α so zu wählen, dass φ minimiert wird, also so, dass φ' klein im Betrag wird. Die zweite Bedingung (3.14) beschränkt φ' nach unten. Um es auch nach oben zu beschränken, werden in Simulationen oft die *starken Wolfe*–Bedingungen gefordert,

$$\Phi_h \left(y_h^{(k)} + \alpha p_k\right) \leq \Phi_h \left(y_h^{(k)}\right) + c_1 \alpha D\Phi_h \left(y_h^{(k)}\right)^\top p_k, \tag{3.15a}$$

$$\left| D\Phi_h \left(y_h^{(k)} + \alpha p_k\right)^\top p_k \right| \geq c_2 \left| D\Phi_h \left(y_h^{(k)}\right)^\top p_k \right|. \tag{3.15b}$$

Die erste dieser beiden Bedingungen entspricht (3.13).

wähle $\overline{\alpha} > 0$, $\rho \in (0,1)$
setze $\alpha = \overline{\alpha}$
while $\Phi_h\big(y_h^{(k)} + \alpha p_k\big) > \Phi_h\big(y_h^{(k)}\big) + c_1 \alpha D\Phi_h\big(y_h^{(k)}\big)^\top p_k$
do
    setze $\alpha = \alpha \cdot \rho$
end while
setze $\alpha_k = \alpha$

Algorithmus 3.2: Algorithmus zur Bestimmung der Schrittweiten α_k.

Bermerkung 3.3.1: Lemma 3.1 aus [NW06] besagt, dass es Schrittweiten gibt, die die starken Wolfe–Bedingungen erfüllen, wenn das zu minimierende Funktional Φ_h stetig differenzierbar und auf der Menge $\left\{ y_h^{(k)} + \alpha_k p_k \mid \alpha > 0 \right\}$ nach unten beschränkt ist, wobei p_k eine Abstiegsrichtung ist. Dies wird allerdings nur für den Fall bewiesen, dass Φ_h auf dem gesamten $\mathbb{R}^N$ definiert ist. Dies ist hier nicht der Fall. Der Beweis kann trotzdem so geführt werden, wenn man beispielsweise zusätzlich

$$\Phi_h(y) > \Phi_h\left(y_h^{(0)}\right) \qquad \text{für alle } y \in \partial D_h \tag{3.16}$$

fordert. Ob dies hier erfüllt ist, ist nicht klar. Insbesondere in Bereichen, in denen die Lösung glatt ist, ist der Einfluss der Parameter auf das Funktional eher gering. So ist das Erfülltsein der (starken) Wolfe–Bedingungen in diesem Fall offen.

Bermerkung 3.3.2: Beim Bestimmen der Schrittweite wird zuerst eine Standardwahl versucht und dann getestet, ob diese die (starken) Wolfe–Bedingungen erfüllt. Falls dies nicht der Fall ist, wird eine geeignete andere Schrittweite gewählt und wieder müssen die (starken) Wolfe–Bedingungen überprüft werden. Jede getestete Schrittweite α beinhaltet dabei das Assemblieren und Lösen eines linearen Systems um $S_h\big(y_h^{(k)} + \alpha p_k\big)$ zu berechnen. Das Testen der Bedingung (3.15a) erfordert zusätzlich die Auswertung von $\Phi_h\big(y_h^{(k)} + \alpha p_k\big)$. Damit Bedingung (3.15b) getestet werden kann, muss noch $D\Phi_h\big(y_h^{(k)} + \alpha p_k\big)$ ausgewertet werden. Dies bedeutet ein erneutes Assemblieren und Lösen des adjungierten Problems (3.8). Um dies zu vermeiden wird in den in Abschnitt 5 beschriebenen Rechnungen der *Backtracking*-Algorithmus aus [NW06, Algorithmus 3.1] verwendet. Dieser ist in Algorithmus 3.2 auf Seite 39 beschrieben. Hier wird die vorgegebene initiale Schrittweite $\overline{\alpha}$ so lange um den Faktor ρ verkleinert, bis Bedingung (3.15a) erfüllt ist. Diese Vorgehensweise verhindert auch, dass die Schrittweite zu klein gewählt wird, da

$$\alpha_k = \max\left\{ \rho^n \overline{\alpha} \;\middle|\; n \in \mathbb{N} \cup \{0\},\ \text{Bedingung (3.15a) ist für } \rho^n\overline{\alpha} \text{ statt } \alpha \text{ erfüllt} \right\}.$$

3.3.3 Approximation der Hessematrix

Zu Beginn der Minimierung von Φ_h ist keine Information zur Hessematrix verfügbar. Daher wird die erste Iterierte $y_h^{(1)}$ durch das Gradientenverfahren berechnet, das heißt $B_0 = \mathrm{Id}$. Sei nun angenommen, die Iterierten $y_h^{(k)}$ und $y_h^{(k+1)} = y_h^{(k)} + \alpha_k p_k$ sind bereits berechnet. Dann ist das neue quadratische Modell

$$m_{k+1}(p) = \Phi_h\left(y_h^{(k+1)}\right) + p^\top D\Phi_h\left(y_h^{(k+1)}\right) + \frac{1}{2} p^\top B_{k+1} p \approx \Phi_h\left(y_h^{(k+1)} + p\right),$$

wobei die Approximation B_{k+1} der Hessematrix noch zu bestimmen ist. Dabei soll $m_{k+1}(p)$ möglichst viele Eigenschaften von $\Phi_h\big(y_h^{(k+1)} + p\big)$ haben. Per Konstruktion gelten bereits $m_{k+1}(0) = \Phi_h\big(y_h^{(k+1)}\big)$ und $Dm_{k+1}(0) = D\Phi_h\big(y_h^{(k+1)}\big)$. Nun wird noch gefordert, dass sich der Gradient von m_{k+1} auch bei der alten Iterierten $y_h^{(k)}$ wie der Gradient von Φ_h verhält, das heißt man fordert

$$Dm_{k+1}(-\alpha_k p_k) = D\Phi_h\left(y_h^{(k)}\right). \tag{3.17}$$

Dies ist äquivalent zu

$$\begin{aligned} Dm_{k+1}(-\alpha_k p_k) = D\Phi_h\left(y_h^{(k+1)}\right) - \alpha_k B_{k+1} p_k &= D\Phi_h\left(y_h^{(k)}\right) \\ \iff \quad \alpha_k B_{k+1} p_k &= D\Phi_h\left(y_h^{(k+1)}\right) - D\Phi_h\left(y_h^{(k)}\right) =: r_k \\ \iff \quad B_{k+1} s_k &= r_k \end{aligned}$$

mit $s_k := \alpha_k p_k = y_h^{(k+1)} - y_h^{(k)}$. Die letzte Gleichung wird Sekantengleichung genannt. Damit diese Bedingung erfüllt ist, muss B_{k+1} den Vektor s_k auf r_k abbilden. Da B_{k+1} positiv definit sein soll, kann dies nur funktionieren, wenn die sogenannte Krümmungsbedingung $s_k^\top r_k > 0$ gilt. Erfüllt die Schrittweite α_k Bedingung (3.14) und ist p_k eine Abstiegsrichtung, so folgt

$$s_k^\top r_k = \alpha_k \left(D\Phi_h\left(y_h^{(k+1)}\right) - D\Phi_h\left(y_h^{(k)}\right)\right)^\top p_k \geq \alpha_k (c_2 - 1) D\Phi_h\left(y_h^{(k)}\right)^\top p_k > 0,$$

da sowohl $c_2 - 1 < 0$ als auch $D\Phi_h\big(y_h^{(k)}\big)^\top p_k < 0$ ist. Das bedeutet, es gibt eine Matrix B_{k+1}, die s_k auf r_k abbildet. Diese ist jedoch nicht eindeutig bestimmt, da die Bedingungen Symmetrie und positive Definitheit nur $N(N-1)/2$ der N^2 Freiheitsgrade der Matrix B_{k+1} bestimmen. Um Eindeutigkeit sicher zu stellen, sollen nun Informationen aus den vorherigen Iterationen verwendet werden. Beispielsweise kann man fordern, dass B_{k+1} nahe an der symmetrischen und positiv definiten Matrix

B_k ist,

$$B_{k+1} = \underset{B=B^\top, Bs_k=r_k}{\arg\min} \|B - B_k\|.$$

Hierbei ist die zu verwendende Matrixnorm noch zu wählen. Diese Herangehensweise führt zur DFP-Formel, benannt nach Davidon, Fletcher und Powell, siehe [NW06, Abschnitt 6.1]. Da zur Berechnung der Abstiegsrichtung p_{k+1} die Inverse von B_{k+1} benötigt wird, ist es sinnvoll eine Bedingung an ebendiese zu stellen, um B_{k+1} eindeutig zu bestimmen. Sei $H_k = B_k^{-1}$ für alle k. Dann wird im BFGS-Verfahren folgende Bedingung an H_{k+1} gestellt

$$H_{k+1} = \underset{H=H^\top, Hr_k=s_k}{\arg\min} \|H - H_k\|.$$

Wählt man als Matrixnorm hier eine gewichtete Frobenius–Norm, so ergibt sich eine explizite Formel für H_{k+1}, siehe [NW06, Seite 140],

$$H_{k+1} = \left(\mathrm{Id} - \varrho_k s_k r_k^\top\right) H_k \left(\mathrm{Id} - \varrho_k r_k s_k^\top\right) + \varrho_k s_k s_k^\top \tag{3.18}$$

mit $\varrho_k = 1/(r_k^\top s_k) > 0$. Tatsächlich ist auch H_{k+1} symmetrisch und positiv definit, wenn dies H_k war. Dies sieht man durch folgende Umformung ($z \in \mathbb{R}^N \setminus \{0\}$)

$$\begin{aligned} z^\top H_{k+1} z &= z^\top \left(\mathrm{Id} - \varrho_k s_k r_k^\top\right) H_k \left(\mathrm{Id} - \varrho_k r_k s_k^\top\right) z + \varrho_k z^\top s_k s_k^\top z \\ &= z^\top \left(\mathrm{Id} - \varrho_k r_k s_k^\top\right)^\top H_k \left(\mathrm{Id} - \varrho_k r_k s_k^\top\right) z + \varrho_k \left(z^\top s_k\right)^2 \\ &= w^\top H_k w + \varrho_k \left(z^\top s_k\right)^2 \\ &> 0, \end{aligned}$$

mit $w = \left(\mathrm{Id} - \varrho_k r_k s_k^\top\right) z$. Die strikte Ungleichheit gilt, da w und $z^\top s_k$ nicht gleichzeitig Null sein können. Der BFGS-Algorithmus ist in Quelltext 3.3 gezeigt.

3.3.4 L-BFGS

Dieser Abschnitt folgt [NW06, Abschnitt 7.2].

Bei großen Optimierungsproblemen ist das Speichern der Approximationen B_k an die Hessematrix sehr aufwändig, da B_k dicht besetzt ist selbst wenn die exakte Hessematrix dies nicht ist. Das Gleiche gilt auch für die Inverse $H_k = B_k^{-1}$. Im Beispiel 3.2.1 entspricht die Dimension des Optimierungsproblems der Anzahl der Gitterzel-

wähle $y_h^{(0)} \in \mathbb{R}^N$ gemäß (2.21)
wähle Approximation H_0 der Inversen der Hessematrix
for $k = 0, 1, \ldots$ **do**
    Bestimme Richtung $p_k = -H_k D\Phi_h\big(y_h^{(k)}\big)$
    Bestimme Schrittweite $\alpha_k > 0$, die die (starken)Wolfe–Bedingungen erfüllt
    setze $y_h^{(k+1)} = y_h^{(k)} + \alpha_k p_k$
    Prüfe Abbruchkriterium
    Berechne H_{k+1} gemäß (3.18)
end for

Algorithmus 3.3: BFGS-Algorithmus.

len. Um den Speicheraufwand zu reduzieren, wird das sogenannte L-BFGS-Verfahren verwendet. Das L steht hier bei für „limited memory“.

Um die Matrix H_k aus H_{k-1} zu berechnen, werden nach Gleichung (3.18) lediglich die Vektoren r_k und s_k benötigt. Angenommen es sind neben H_0 noch alle Vektoren r_i und s_i für $i = 0, \ldots, k-1$ gegeben, so kann H_k rekursiv berechnet werden. Die Idee ist nun, dass nur jeweils die letzten m Vektoren signifikant zur Approximation an die Inverse der Hessematrix beitragen. Hierbei ist m eine zu wählende natürliche Zahl. Die vorherigen $k-m$ Vektorenpaare werden zwecks Reduktion des Speicheraufwands nicht mehr berücksichtigt. Durch wiederholtes Anwenden von (3.18) mit den letzten m Vektorenpaaren ergibt sich dann die L-BFGS-Approximation

$$\begin{aligned} H_k = &\left(V_{k-1}^\top \cdot \ldots \cdot V_{k-m}^\top\right) H_0 \left(V_{k-m} \cdot \ldots \cdot V_{k-1}\right) \\ &+ \sum_{i=0}^{m-1} \varrho_{k-m+i} \left(V_{k-1}^\top \cdot \ldots \cdot V_{k-m+i+1}^\top\right) s_{k-m+i} s_{k-m+i}^\top \left(V_{k-m+i+1} \cdot \ldots \cdot V_{k-1}\right) \end{aligned} \tag{3.19}$$

mit

$$V_k = \mathrm{Id} - \varrho_k r_k s_k^\top, \quad s_k = \alpha_k p_k = y_h^{(k+1)} - y_h^{(k)} \text{ und } r_k = D\Phi_h\left(y_h^{(k+1)}\right) - D\Phi_h\left(y_h^{(k)}\right). \tag{3.20}$$

Im Rahmen des BFGS-Verfahrens muss lediglich eine Matrix-Vektor-Multiplikation mit H_k berechnet werden. Dies ist durch Algorithmus 3.4 effizient möglich. Die Multiplikation mit der initialen Approximation der Inversen der Hessematrix H_0 ist hierbei losgelöst vom Rest des Algorithmus. Es ist daher möglich, H_0 ebenfalls mit jedem

setze $q = D\Phi_h\big(y_h^{(k)}\big)$
wähle Approximation H_0 der Inversen der Hessematrix
for $i = k-1, \ldots, k-m$ **do**
  setze $\alpha_i = \varrho_i s_i^\top q$
  setze $q = q - \alpha_i r_i$
end for
setze $t = H_0 q$
for $i = k-m, \ldots, k-1$ **do**
  setze $\beta = \varrho_i r_i^\top t$
  setze $t = t + s_i(\alpha_i - \beta)$
end for

Algorithmus 3.4: Algorithmus zur Berechnung des Produkts $t = H_k D\Phi_h\big(y_h^{(k)}\big)$ nach (3.19) bei gegebenen Vektoren $s_{k-m}, \ldots, s_{k-1}$ und $r_{k-m}, \ldots, r_{k-1}$.

Iterationsschritt zu variieren. In [NW06] wird vorgeschlagen H_k^0 als Vielfaches der Einheitsmatrix zu wählen, $H_k^0 = \gamma_k \,\mathrm{Id}$ mit

$$\gamma_k = \frac{s_{k-1}^\top r_{k-1}}{r_{k-1}^\top r_{k-1}}, \qquad k \geq 1. \tag{3.21}$$

Mit dieser Wahl des Parameters γ wird versucht eine Skalierung zu finden, die der Inversen der Hessematrix in gewisser Weise nahe ist, siehe unter der Überschrift „Implementation“ in [NW06, Abschnitt 6.1]. Die erste Matrix H_0^0 wird in der Regel als Einheitsmatrix gesetzt. Der L-BFGS-Algorithmus ist in 3.5 dargestellt.

Statt einer $N \times N$-Matrix H_k werden in jedem Iterationsschritt des L-BFGS-Verfahrens nur m Vektorpaare gespeichert. In Anwendungen hat sich gezeigt, dass $m \leq 100$ gute Ergebnisse erzielt. In [NW06] wird $3 \leq m \leq 20$ vorgeschlagen.

wähle Startwert $y_h^{(0)} \in \mathbb{R}^N$ gemäß(2.21)
for $k = 0, 1, \ldots$ **do**
  wähle H_k^0 mit (3.21)
  berechne $p_k = -H_k D\Phi_h\big(y_h^{(k)}\big)$ mit Algorithmus 3.4
  berechne α_k gemäß Algorithmus 3.2
  berechne $x_{k+1} = x_k + \alpha_k p_k$
  if $k > m$
    lösche s_{k-m} und r_{k-m}
  end if
  berechne $s_k = x_{k+1} - x_k$
  berechne $r_k = D\Phi_h\big(y_h^{(k+1)}\big) - D\Phi_h\big(y_h^{(k)}\big)$
end for

Algorithmus 3.5: L-BFGS-Algorithmus

4 Diskussion von möglichen Funktionalen

Die SUPG–Methode hat den Nachteil, dass auf jeder Gitterzelle ein Parameter zu wählen ist. Wie genau diese Wahl auszusehen hat, ist nicht bekannt. Jedoch ist klar, dass das Verhalten der Lösung in die Wahl der Parameter eingehen muss. Beispielsweise treten Grenzschichten oft an Ausströmrändern mit Dirichlet–Randbedingungen auf, sodass andere Parameterwahlen in diesen Teilen des Gebiets verbesserte Ergebnisse erzielen können, siehe [Kno09]. Im Allgemeinen ist bekannt, dass die Wahl der Parameter großen Einfluss auf die numerische Lösung hat. Die a posteriori Parameterbestimmung ist ein Weg, diese Parameter abhängig von der berechneten Lösung zu bestimmen. Die hier auftretenden Funktionale sind allerdings noch nicht definiert worden. Statt also gute Parameter zu suchen, werden jetzt gute Funktionale gesucht. Welche genau vorteilhaft sind, ist ebenfalls nicht bekannt. Die Forschungen zu diesem Thema stehen erst am Anfang.

Der Einfachheit halber wird nur der zweidimensionale Fall, $d = 2$, betrachtet.

4.1 Residuale Funktionale

Das zu minimierende Funktional $\Phi_h : D_h \to \mathbb{R}$, $\Phi_h = I_h \circ S_h$, soll eine Aussage über die Qualität einer berechneten Lösung liefern. Das Ziel ist es, scharfe Grenzschichten in numerische Lösungen zu ermöglichen, jedoch gleichzeitig unphysikalische Oszillationen zu vermeiden. Eine naheliegende Wahl sind Residuen. Das heißt man wählt

$$\Phi_h(y_h) = \| -\varepsilon \Delta S_h(y_h) + \mathbf{b} \cdot \nabla S_h(y_h) + c S_h(y_h) - f \| ,$$

wobei der Laplace–Term wieder elementweise zu verstehen ist und die Wahl der Norm noch offen ist. Sinnvoll erscheinen zum Beispiel die L^p-Normen, $p \in [1, \infty]$. Die Motivation für residuale Funktionale ist, dass im Falle kleiner Residuen die starke Form (2.2) der Gleichung gut erfüllt ist und somit auch die Lösung als gut anzusehen ist.

Bermerkung 4.1.1: Die gleiche Fragestellung ergibt sich auch bei der Suche nach Fehlerschätzern, die unter anderem bei adaptiven Gitterverfeinerungen verwendet werden können. Hierbei wird versucht die Gitterzellen zu identifizieren, in denen der Fehler am größten ist, um diese Gitterzellen dann zu verfeinern. Im Fall dominierender Konvektion versagen jedoch die gebräuchlichsten Fehlerschätzer, siehe z. B. [Joh00].

Eine naheliegende Wahl sind a posteriori Fehlerschätzer für Konvektions-Diffusions-Reaktions-Gleichungen. Von R. Verfürth aus [Ver05] ist folgender Vorschlag, ($w_h \in W_h$)

$$\tilde{I}_h(w_h) = \sum_{T\in\mathcal{T}_h} \alpha_T^2 \left\| -\varepsilon\Delta w_h + \mathbf{b}\cdot\nabla w_h + c w_h - f \right\|_{0,T}^2 + \sum_{E\subset\partial T} \varepsilon^{1/2}\alpha_E \left\| R_E(w_h) \right\|_{0,E}^2 , \tag{4.1}$$

wobei mit E die Kanten einer Gitterzelle T bezeichnet sind und

$$R_E(w_h) = \begin{cases} -\left[|\varepsilon \mathbf{n}_E\cdot\nabla w_h|\right]_E & \text{, falls } E\not\subset\partial\Omega, \\ g - \varepsilon\mathbf{n}_E\cdot\nabla w_h & \text{, falls } E\subset\Gamma^N, \\ 0, & \text{, falls } E\subset\Gamma^D, \end{cases}$$

$$\alpha_T = \min\left\{ \operatorname{diam}(T)\varepsilon^{1/2}, c_0^{-1/2} \right\},$$

$$\alpha_E = \min\left\{ \operatorname{diam}(E)\varepsilon^{1/2}, c_0^{-1/2} \right\}.$$

Hierbei ist $\mathbf{n}_E$ ein für jede Kante E festgelegter Normalenvektor[1] mit Norm Eins und $[|v_h|]$ ist der Sprung einer Funktion v_h über E, das heißt für $x\in E$ gilt

$$\left[|v_h|\right](x) := \begin{cases} \lim_{t\searrow 0}\left(v_h(x+t\mathbf{n}_E) - v_h(x-t\mathbf{n}_E)\right) & \text{, falls } E\not\subset\partial\Omega, \\ \lim_{t\searrow 0}\left(-v_h(x-t\mathbf{n}_E)\right) & \text{, falls } E\subset\partial\Omega. \end{cases}$$

Dieser Fehlerschätzer erwies sich in [JKS11] als nicht brauchbar. Dies liegt an zweierlei Problemen. Zum einen können Grenzschichten nicht mit vertretbarem Aufwand aufgelöst werden, sodass selbst knotenexakte Lösungen dort stark von Null verschiedene Residuen haben. Der globale Fehlerschätzer (4.1) wird also durch die Beiträge einiger weniger Gitterzellen dominiert, in denen eine Reduktion des Residuums eventuell nicht mehr oder nur in geringem Maße möglich ist. Da eine Minimierung des Funktionals Φ_h aber zunächst die größten Beiträge reduzieren würde, werden auch die Residuen außerhalb von Grenzschichten nicht reduziert. Um dies zu vermeiden,

[1] An inneren Kanten ist die Orientierung nicht wichtig, am Rand zeigt der Normalenvektor nach außen.

wurde in [JKS11] vorgeschlagen, die Gitterzellen an Dirichlet–Rändern nicht mit in den Fehlerschätzer einzubeziehen. Ein weiteres Problem ist, dass Oszillationen an Grenzschichten oft in der Richtung senkrecht zur Konvektion (crosswind) auftreten. In [JKS11] wurde versucht, diese durch die Minimierung eines Funktionals zu unterdrücken, welches die crosswind-Ableitung enthält. Das in dieser Arbeit verwendete Funktional aus [JKS11] ist

$$I_h(w_h) = \sum_{\substack{T\in\mathcal{T}_h \\ \overline{T}\cap\overline{\Gamma^D}=\emptyset}} \|-\varepsilon\Delta w_h + \mathbf{b}\cdot\nabla w_h + cw_h - f\|_{0,T}^2 + \left\|\phi\left(\left|\mathbf{b}^\perp\cdot\nabla w_h\right|\right)\right\|_{L^1(T)}, \tag{4.2}$$

mit

$$\mathbf{b}^\perp(\mathbf{x}) = \begin{cases} \frac{1}{|\mathbf{b}(\mathbf{x})|}\left(b_2(\mathbf{x}), -b_1(\mathbf{x})\right) & \text{, falls } \mathbf{b}(\mathbf{x})\neq 0, \\ 0 & \text{, falls } \mathbf{b}(\mathbf{x}) = 0, \end{cases}$$

$$\phi(x) = \begin{cases} \sqrt{x} & \text{, falls } x \geq 1, \\ \frac{1}{2}\left(5x^2 - 3x^3\right) & \text{, falls } x < 1. \end{cases}$$

Im Vergleich zu (4.1) fehlen hier auch die Beiträge über die Kanten. In [JKS11] wird beschrieben, dass deren Einfluss vernachlässigbar ist. Die Funktion ϕ ist so gewählt, dass I_h Fréchet-differenzierbar ist,

$$\begin{aligned} DI_h(w_h)v_h = &\sum_{\substack{T\in\mathcal{T}_h \\ \overline{T}\cap\overline{\Gamma^D}=\emptyset}} 2\left(-\varepsilon\Delta w_h + \mathbf{b}\cdot\nabla w_h + cw_h - f, -\varepsilon\Delta v_h + \mathbf{b}\cdot\nabla v_h + cv_h\right)_{0,T} \\ &+ \int_T \left[D_{w_h}\phi\left(\left|\mathbf{b}^\perp\cdot\nabla w_h\right|\right)v_h\right](\mathbf{x}) \end{aligned}$$

mit

$$\begin{aligned} &\left[D_{w_h}\phi\left(\left|\mathbf{b}^\perp\cdot\nabla w_h\right|\right)v_h\right](\mathbf{x}) = \\ &\quad\left(5\left(\mathbf{b}^\perp\cdot\nabla w_h(\mathbf{x})\right) - \frac{9}{2}\operatorname{sign}\left(\mathbf{b}^\perp\cdot\nabla w_h(\mathbf{x})\right)\left(\mathbf{b}^\perp\cdot\nabla w_h(\mathbf{x})\right)^2\right)\mathbf{b}^\perp\cdot\nabla v_h(\mathbf{x}) \end{aligned}$$

falls $\left|\mathbf{b}^\perp\cdot\nabla w_h(\mathbf{x})\right| < 1$ und

$$\left[D_{w_h}\phi\left(\left|\mathbf{b}^\perp\cdot\nabla w_h\right|\right)v_h\right](\mathbf{x}) = \frac{1}{2}\frac{\operatorname{sign}\left(\mathbf{b}^\perp\cdot\nabla w_h(\mathbf{x})\right)}{\sqrt{|\mathbf{b}^\perp\cdot\nabla w_h(\mathbf{x})|}}\mathbf{b}^\perp\cdot\nabla v_h(\mathbf{x})$$

falls $\left|\mathbf{b}^\perp\cdot\nabla w_h(\mathbf{x})\right| \geq 1$.

4.2 Funktionale zur Einhaltung von Minimum- und Maximumprinzipien

In dieser Arbeit werden Funktionale untersucht, die gewisse a priori Informationen über die Lösung erfordern. Solche Informationen sind in Anwendungen manchmal bekannt.

Beispiel 4.2.1: Für eine Lösung u der Gleichung (2.6) gilt:

$$u_b \geq 0 \text{ und } a(u,v) \geq 0 \text{ für alle } v \in H_0^1(\Omega) \text{ mit } v \geq 0 \Longrightarrow u \geq 0,$$

siehe [Dob10, Abschnitt 7.5] für eine genaue Formulierung. Das heißt unter gewissen Voraussetzungen bleibt eine Lösung positiv. Im Spezialfall $c = 0$ kann sogar $\sup_\Omega |u| = \sup_{\partial\Omega} |u|$ gezeigt werden. Auch Maximum- und Minimumprinzipien für klassische Lösungen findet man in der Literatur, zum Beispiel in [Bra07], [Dob10] oder [Eva98].

Es sei angenommen, dass die Lösung u des stetigen Problems (2.2) a priori zwischen den Grenzen $M_{\min}$ und $M_{\max}$ liegt, also

$$\text{für fast alle } x \in \Omega \text{ gilt } M_{\min} \leq u(x) \leq M_{\max}.$$

Dies können zum Beispiel physikalische Grenzen sein, etwa wenn u die Konzentration eines gelösten Stoffes ist. Dann ist sicher, dass u nur Werte zwischen Null und Eins annimmt. Ein weiteres Beispiel ist die Temperaturverteilung in einem Fluid. Hier ist ohne zusätzliche Wärmequellen (also $c = 0$ und $f \leq 0$) sicher, dass die Temperatur im Innern des Gebiets nicht höher ist als am Rand. Aus der Sicht eines Anwenders sind Verletzungen dieser Grenzen oft nicht akzeptabel. Daher sollte auch eine numerische Lösung sie respektieren. In der Praxis wird dies oft erreicht, indem die berechnete Lösung u_h durch Projektionen der Form

$$u_h \mapsto \max\{u_h, M_{\min}\} \qquad \text{und} \qquad u_h \mapsto \min\{u_h, M_{\max}\}$$

einfach auf den richtigen Bereich eingeschränkt wird. Diese Vorgehensweise scheint jedoch wenig plausibel. Insbesondere wird auf diese Weise im Allgemeinen die Massenerhaltung verletzt. Im Rahmen der a posteriori Parameterbestimmung kann mit einem geeignet gewählten Funktional das Verletzen von Minimum- und Maximumprinzipien bestraft werden. Dazu wird ein weiteres Funktional $F_h : V_h \to \mathbb{R}$ definiert, welches dem zu minimierenden Funktional Φ_h hinzugefügt wird. Es soll also

$$\Phi_h(y_h) = \beta_0 I_h\left(S_h(y_h)\right) + F_h\left(S_h(y_h)\right) \tag{4.3}$$

minimiert werden. Der frei wählbare Parameter β_0 bietet die Möglichkeit die beiden Anteile I_h und F_h des Funktionals zu gewichten. Es sei nun $F_h(u_h) = \|f_1(u_h)\|_0^2 + \|f_2(u_h)\|_0^2$ mit $f_i : V_h \to L^2(\Omega)$, $i = 1, 2$, fast überall definiert durch

$$\big(f_1(u)\big)(\mathbf{x}) = \beta_1 \exp\left(\alpha_1 g(u(\mathbf{x}))^2\right) - 1, \qquad \big(f_2(u)\big)(\mathbf{x}) = \beta_2 g\left(u(\mathbf{x})\right)^{1+\alpha_2}.$$

Dabei sind α_1, α_2, β_1 und β_2 positive Konstanten und $g : \mathbb{R} \to \mathbb{R}$ ist nur außerhalb des Intervalls $[M_{\min}, M_{\max}]$ von Null verschieden,

$$g(y) = \begin{cases} M_{\min} - y & \text{if } y \le M_{\min}, \\ 0 & \text{if } M_{\min} \le y \le M_{\max}, \\ y - M_{\max} & \text{if } y \ge M_{\min}. \end{cases}$$

Offenbar ist $F(u_h) = 0$, falls u_h nur Werte in $[M_{\min}, M_{\max}]$ annimmt. Für die a posteriori Parameterbestimmung wird noch die Fréchet–Ableitung von F_h benötigt. Diese existiert und ist für alle u_h, $v_h \in V_h$ gegeben durch

$$\begin{aligned} DF_h(u_h)v_h &= 2\left(f_1(u_h), Df_1(u_h)v_h\right)_0 + 2\left(f_2(u_h), Df_2(u_h)v_h\right)_0 \\ &= 2\int_\Omega \beta_1\left(\exp\left(\alpha_1 g(u)^2\right) - 1\right) 2\beta_1\alpha_1 g(u)g'(u)\exp\left(\alpha_1 g(u)^2\right) v\, d\mathbf{x} \\ &\quad + 2\int_\Omega \beta_2 g(u)^{1+\alpha_2}\beta_2(1+\alpha_2)g(u)^{\alpha_2}g'(u)v\, d\mathbf{x}, \end{aligned}$$

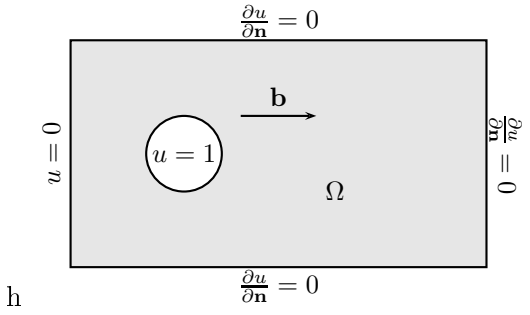
wobei auf die Abhängigkeit der Funktionen u und v von der Variable $\mathbf{x}$ aus Gründen der Übersichtlichkeit verzichtet wurde.

5 Numerisches Beispiel

Im Allgemeinen ist es nicht trivial geeignete Testbeispiele zu finden, deren Lösung bekannt ist und charakteristisches Verhalten zeigt, insbesondere Grenzschichten. Wird eine solche Lösung vorgegeben und werden die Randdaten und rechte Seite f entsprechend gesetzt, so hängt auch f von der Diffusion ε ab. Das heißt auch f kann Grenzschichten haben. Integrale vom Typ $(f, v_h)_0$ sind dann sehr schwer numerisch zu berechnen. Es werden sehr genaue Quadraturformeln benötigt, um zu verhindern, dass Quadraturfehler alle weiteren Fehlerquellen dominieren. Das verlängert die Zeit der Berechnung der rechten Seite erheblich und kann dazu führen, dass die Methode ineffizient wird. Dies gilt besonders in drei oder mehr Dimensionen.

5.1 Problembeschreibung

In dieser Arbeit soll ein Beispiel aus [Hem96] als Testbeispiel dienen. Weitere wurden in [JKS11] und [Luk11] betrachtet. Es seien $\Omega = (-3, 8) \times (-3, 3) \setminus B_1(0)$ mit $B_1(0) = \left\{(x, y) \,\middle|\, x^2 + y^2 \leq 1\right\}$, $\varepsilon = 10^{-6}$, $\mathbf{b} = (1, 0)^\top$ und $c = f = 0$, sowie folgende Randbedingungen vorgegeben:



h

Abbildung 5.1: Beschreibung des Beispiels von Hemker

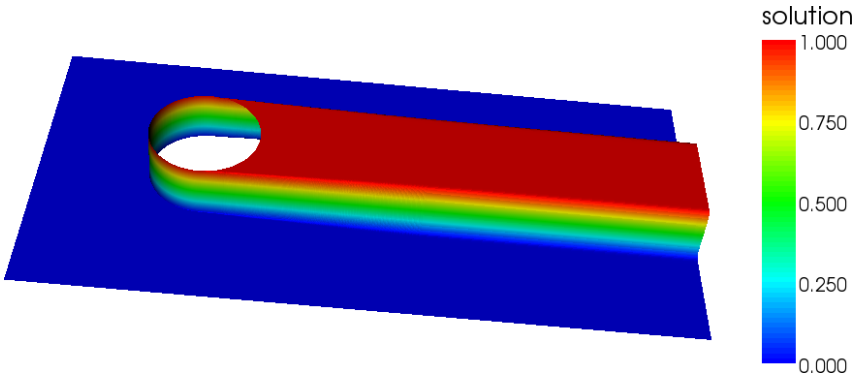


Abbildung 5.2: Exakte Lösung des Beispiels von Hemker

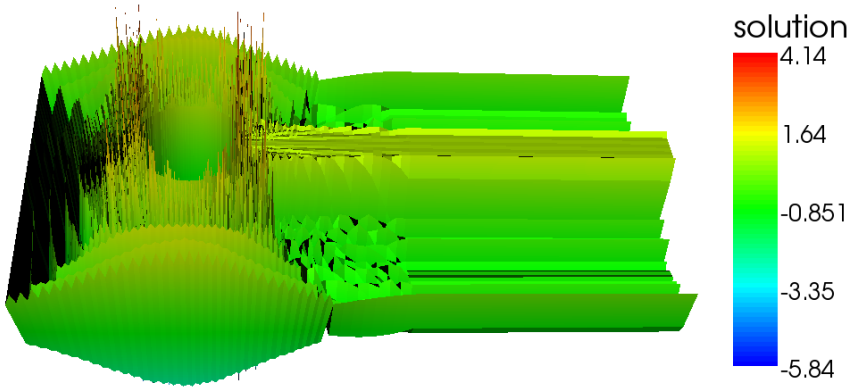


Abbildung 5.3: Lösung u des Beispiels von Hemker mit Galerkin–Verfahren, Level 4

$$
\begin{aligned}
u &= 0 \quad \text{auf } \{(x,y) \in \partial\Omega \,|\, x = -3\}, \\
u &= 1 \quad \text{auf } \left\{(x,y) \in \partial\Omega \,\middle|\, x^2 + y^2 = 1\right\}, \\
\frac{\partial u}{\partial \mathbf{n}} &= 0 \quad \text{auf allen anderen Teilen des Randes } \partial\Omega.
\end{aligned}
$$

Dies ist in Abbildung 5.1 verdeutlicht. Die Lösung kann mit Hilfe von modifizierten Bessel–Funktionen explizit angeben werden, siehe [Hem96]. Sie besitzt zwei innere Grenzschichten entlang der Konvektion $\mathbf{b}$ ausgehend vom oberen und unteren Rand des Einheitskreises sowie eine Randgrenzschicht bei $\left\{(x,y) \,\middle|\, x^2 + y^2 = 1, x < 0\right\} \subset \partial\Omega$. In Abbildung 5.2 ist die Lösung zu sehen. Zur Illustration ist in Abbildung 5.3 noch die Lösung des nicht stabilisierten Galerkin–Verfahrens zu sehen. Die angesprochenen unphysikalischen Oszillationen sind zu erkennen. Auf gröberen Gittern sind sie noch deutlicher.

Bermerkung 5.1.1: Hier ist $c - \frac{1}{2}\text{div}\mathbf{b} = 0$, sodass (2.4) nur mit $c_0 = 0$ erfüllt ist. In diesem Fall kann die eindeutige Lösbarkeit in einer schwächeren Norm als der SUPG–Norm gezeigt werden, wenn statt (2.26) nur $0 < y_T \leq h_T^2/(\varepsilon c_{\text{inv}}^2)$ gefordert wird. Sofern P^1- oder Q^1-Finiten Elemente auf Rechtecken verwendet werden, entfällt diese Forderung komplett, da in diesem Falle (2.24) trivialerweise erfüllt ist.

5.2 Ergebnisse

Alle Rechnungen wurden mit dem Programmpaket MooNMD, [JM04], durchgeführt. Hierbei kam ein direkter Löser auf verschieden Gitterlevel zum Einsatz. Die Anzahl der Level war sechs, wobei das gröbste in Abbildung 5.4 zu sehen ist und alle weiteren durch uniforme Verfeinerung daraus hervorgingen. Die krummlinigen Randteile wurden mit isoparametrischen Finiten Elementen approximiert. Das feinste Gitter hat 188416 Gitterzellen und 189536 Gitterpunkte. Alle Berechnungen wurden nur für Q^1-Finite Elemente durchgeführt, da diese in Anwendungen am wichtigsten sind. In [JKS11] wurden auch Q^2- und Q^3-Finite Elemente eingesetzt.

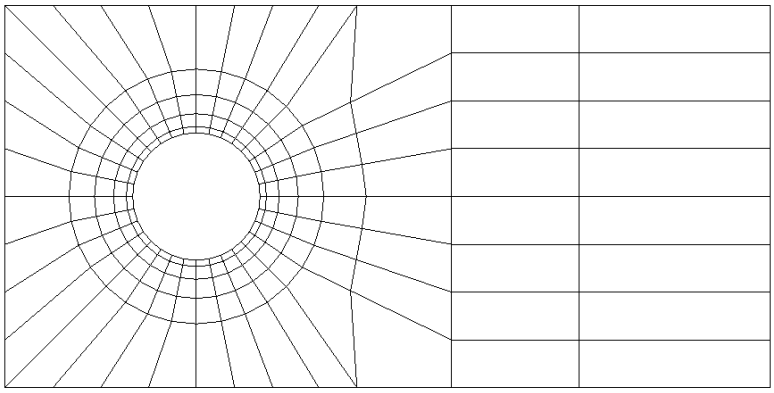
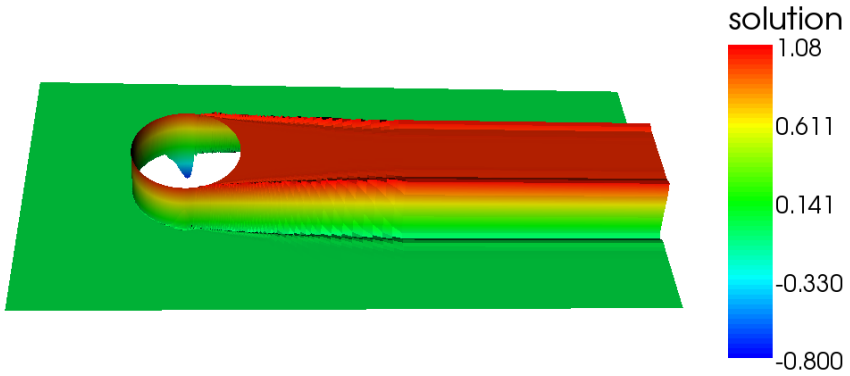


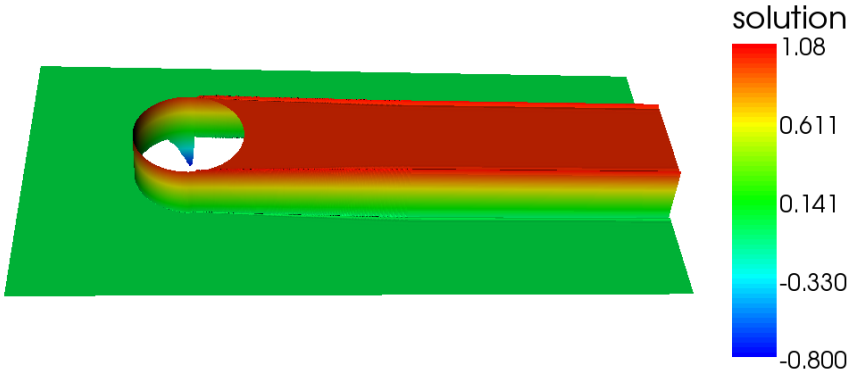
Abbildung 5.4: Gitter auf Level 1 mit 184 Gitterzellen und 219 Gitterpunkten.

Als Referenz dient die berechnete SUPG-Lösung von (2.20) mit der Paramterwahl (2.21). Diese ist in Abbildung 5.5, Seite 53, zu sehen. Es sind einerseits unphysikalische Oszillationen entlang der Grenzschicht, als auch Verletzungen des Minimum- und des Maximumprinzips deutlich erkennbar.

Es kamen die folgenden vier Funktionale zum Einsatz:



(a) SUPG-Lösung, Level 4



(b) SUPG-Lösung, Level 6

Abbildung 5.5: SUPG-Lösung

1. Das residuale Funktional $\Phi_h(y_h) = I_h\big(S_h(y_h)\big)$ aus (4.2),
2. das Funktional $\Phi_h(y_h) = \big\|f_1\big(S_h(y_h)\big)\big\|_0^2$ aus (4.3), d. h. $\beta_0 = \beta_2 = 0$,
3. das Funktional $\Phi_h(y_h) = \big\|f_2\big(S_h(y_h)\big)\big\|_0^2$ aus (4.3), d. h. $\beta_0 = \beta_1 = 0$,
4. das Funktional $\Phi_h(y_h) = \beta_0 I_h\big(S_h(y_h)\big) + F_h\big(S_h(y_h)\big)$ mit $\beta_0 = \beta_1 = \beta_2 = 1$.

Außerdem wurden die Parameter α_1 und α_2 in (4.3) stets auf Eins gesetzt.

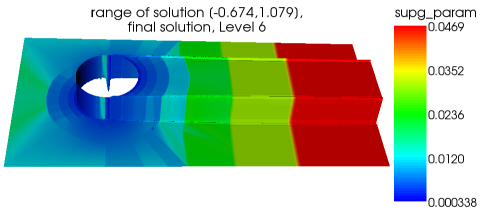
In den Simulationen wurde die Minimierung des Funktionals Φ_h mit Hilfe des L-BFGS-Algorithmus 3.5 auf Seite 44 realisiert. Als Abbruchkriterium wurde

$$\frac{\Phi_h\left(y_h^{(k-10)}\right) - \Phi_h\left(y_h^{(k)}\right)}{\Phi_h\left(y_h^{(k-10)}\right)} \leq d_{\min}$$

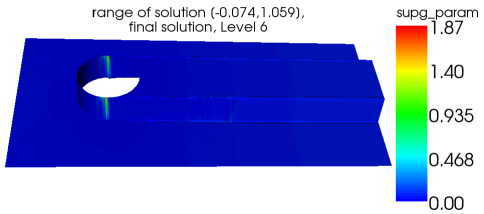
verwendet, wobei $d_{\min} = 10^{-4}$ gesetzt wurde. Die Schrittweite wurde mit dem Backtracking-Algorithmus auf Seite 39 ermittelt. Hier wurde die initiale Schrittweite $\overline{\alpha}$ auf Eins und $\rho = 0.5$ gesetzt. Weiterhin betrug die Anzahl m der zu speichernden Vektorpaare für das L-BFGS-Verfahren 100.

In Tabelle 5.1 sind für jedes der sechs Level die Rechenzeit, die Anzahl der Iterationen, die Rechenzeit pro Iteration sowie der benötigte Speicher eingetragen. Der benötigte Speicher ist bei allen vier untersuchten Funktionalen praktisch identisch. Das war auch zu erwarten, da die zu lösenden Probleme in jedem Fall gleich groß sind. Aus diesem Grund sind auch die Rechenzeiten pro Iteration vergleichbar für die Fälle 1 und 4. Im Fall 2 ist die Rechenzeit pro Iteration auf dem feinsten Gitter stark erhöht. Dies lag an sehr kleinen Schrittweiten, sodass zuvor viele Schrittweiten getestet wurden. Jeder dieser Tests erfordert das Assemblieren und Lösen eines linearen Systems sowie die Auswertung des Funktionals, siehe auch Bemerkung 3.3.2. Als weiteres Abbruchkriterium wurde in Fall 2, Level 5, und Fall 3, Level 1 und 6, die maximale Anzahl der Iterationen auf 10 000 beschränkt.

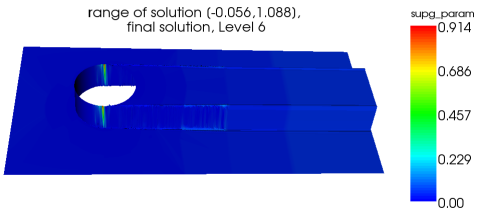
Für die einzelnen untersuchten Funktionale sind in Abbildungen 5.7-5.14 die Lösungen zu bestimmten Iterierten auf dem Level 4 dargestellt. Alle vier Funktionale führen zu einer starken Vergrößerung der SUPG-Paramter oberhalb und unterhalb des Kreises und entlang der inneren Grenzschicht.



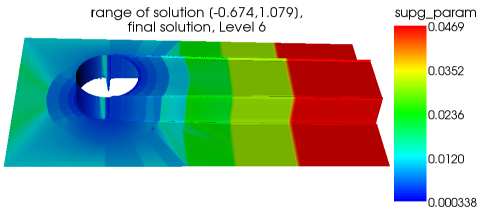
(a) Fall 1.



(b) Fall 2.



(c) Fall 3.

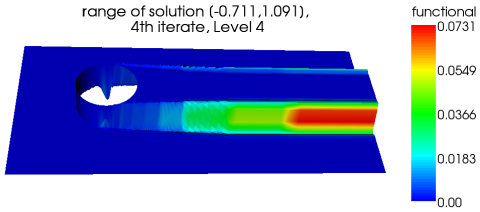


(d) Fall 4.

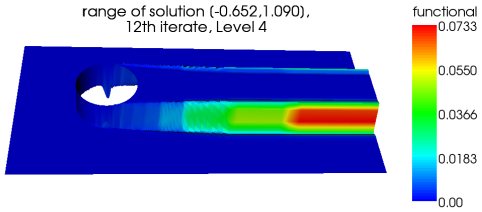
Abbildung 5.6: Berechnete Lösungen auf dem feinsten Gitter 6 für die vier untersuchten Funktionale, gefärbt nach dem SUPG-Parameter y_h (auf P^1-Finite–Elemente projiziert).

	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6
Freiheitsgrade	219	806	3084	12056	47664	189536
SUPG						
Rechenzeit	0.018	0.076	0.231	0.816	3.095	15.89
Fall 1						
Rechenzeit	11.95	77.42	367.6	981.1	4483	46386
Iterationen	492	685	732	377	535	1109
Rechenzeit / Iteration	0.024	0.113	0.502	2.602	8.380	41.82
Speicher	1.146	1.845	3.390	9.511	33.88	118.2
Fall 2						
Rechenzeit	3.763	134.0	1051	5809	70824	83580
Iterationen	113	794	1138	1427	10000	209
Rechenzeit / Iteration	0.033	0.169	0.924	4.071	7.082	399.9
Speicher	1.146	1.845	3.390	9.511	33.88	118.2
Fall 3						
Rechenzeit	214.4	516.4	668.5	3343	16930	399843
Iterationen	10000	1293	219	764	1017	10000
Rechenzeit / Iteration	0.021	0.399	3.053	4.375	16.65	39.98
Speicher	1.146	1.845	3.390	9.511	33.88	118.2
Fall 4						
Rechenzeit	23.64	83.41	509.7	1560	5882	60305
Iterationen	951	612	1095	669	647	1463
Rechenzeit / Iteration	0.025	0.136	0.465	2.332	9.093	41.22
Speicher	1.146	1.845	3.390	9.511	33.88	118.2

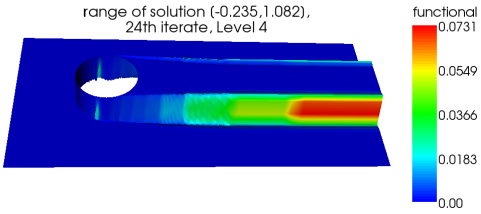
Tabelle 5.1: Vergleich der Leistungsmerkmale für die vier untersuchten Funktionale und der SUPG-Methode. Rechenzeit in Sekunden und Speicher in Megabyte (MB).



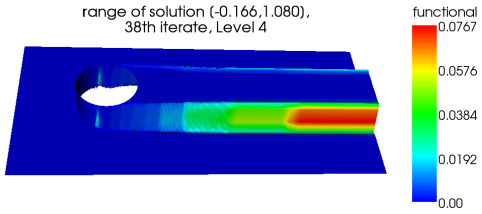
(a) Level 4, 4. Iterierte.



(b) Level 4, 12. Iterierte.

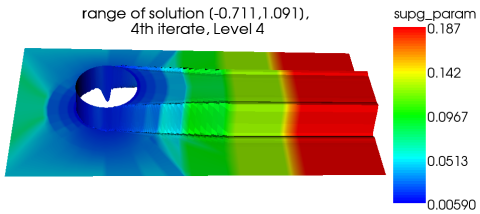


(c) Level 4, 24. Iterierte.

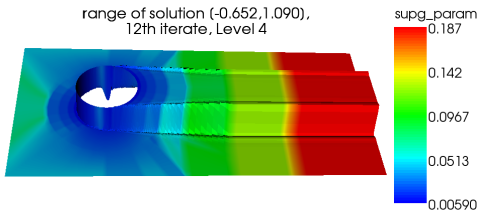


(d) Level 4, 38. Iterierte.

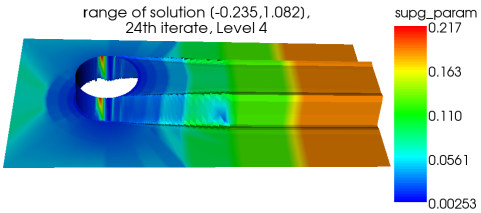
Abbildung 5.7: Fall 1, Minimierung des residualen Funktionals $\Phi_h(y_h) = I_h\big(S_h(y_h)\big)$ aus (4.2). Bild der Lösung, Farben nach Größe des Funktionals auf jeder Gitterzelle (auf P^1-Finite–Elemente projiziert).



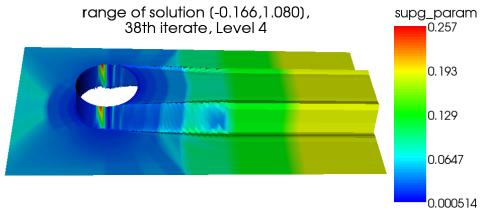
(a) Level 4, 4. Iterierte.



(b) Level 4, 12. Iterierte.

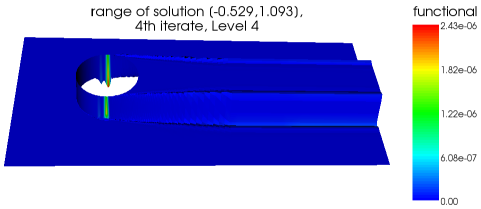


(c) Level 4, 24. Iterierte.

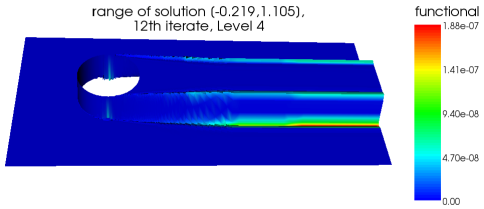


(d) Level 4, 38. Iterierte.

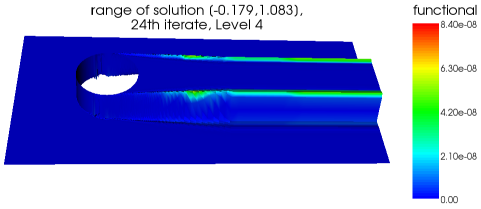
Abbildung 5.8: Fall 1, Minimierung des residualen Funktionals $\Phi_h(y_h) = I_h\big(S_h(y_h)\big)$ aus (4.2). Bild der Lösung, Farben nach Größe des Paramters y_T auf jeder Gitterzelle (auf P^1-Finite–Elemente projiziert).



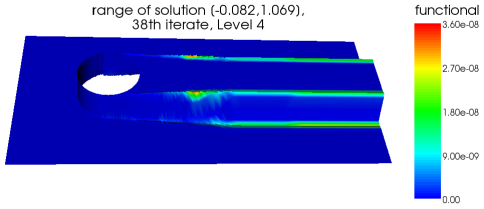
(a) Level 4, 4. Iterierte.



(b) Level 4, 12. Iterierte.

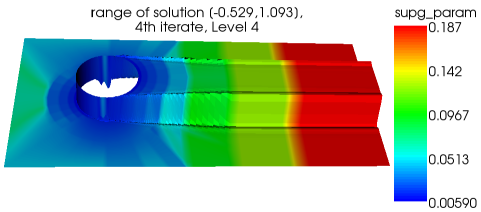


(c) Level 4, 24. Iterierte.

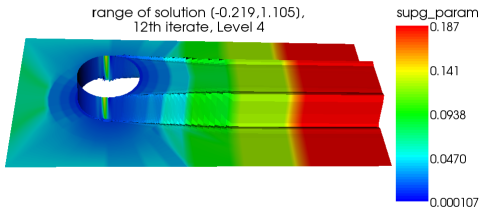


(d) Level 4, 38. Iterierte.

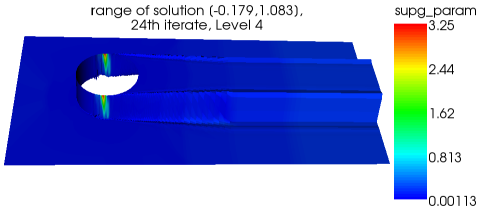
Abbildung 5.9: Fall 2, Minimierung des Funktionals $\Phi_h(y_h) = \left\| f_1\big(S_h(y_h)\big) \right\|_0^2$. Bild der Lösung, Farben nach Größe des Funktionals auf jeder Gitterzelle (auf P^1-Finite–Elemente projiziert).



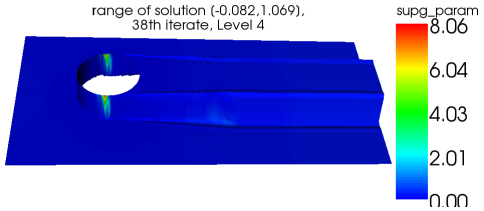
(a) Level 4, 4. Iterierte.



(b) Level 4, 12. Iterierte.

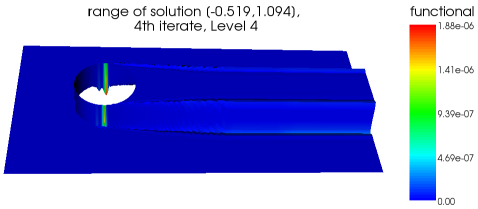


(c) Level 4, 24. Iterierte.

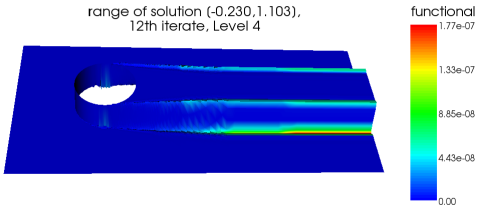


(d) Level 4, 38. Iterierte.

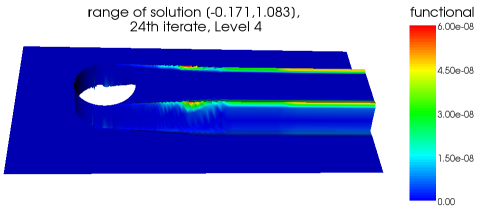
Abbildung 5.10: Fall 2, Minimierung des Funktionals $\Phi_h(y_h) = \left\| f_1\big(S_h(y_h)\big) \right\|_0^2$. Bild der Lösung, Farben nach Größe des Paramters y_T auf jeder Gitterzelle (auf P^1-Finite–Elemente projiziert).



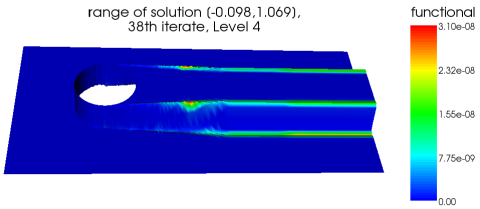
(a) Level 4, 4. Iterierte.



(b) Level 4, 12. Iterierte.

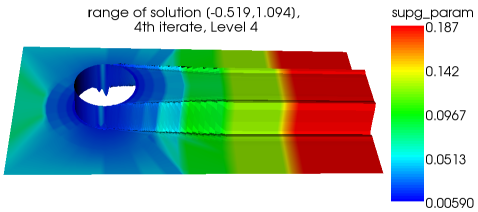


(c) Level 4, 24. Iterierte.

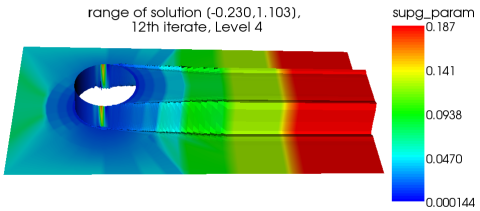


(d) Level 4, 38. Iterierte.

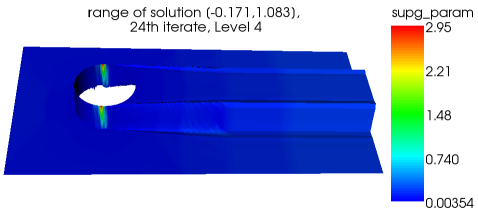
Abbildung 5.11: Fall 3, Minimierung des Funktionals $\Phi_h(y_h) = \left\| f_2\big(S_h(y_h)\big) \right\|_0^2$. Bild der Lösung, Farben nach Größe des Funktionals auf jeder Gitterzelle (auf P^1-Finite–Elemente projiziert).



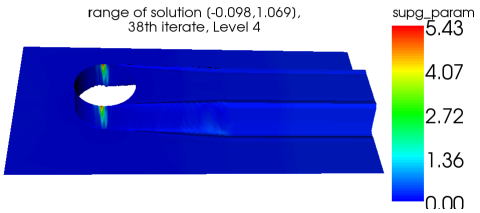
(a) Level 4, 4. Iterierte.



(b) Level 4, 12. Iterierte.

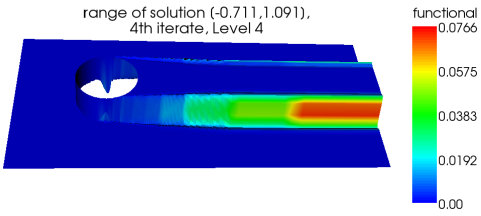


(c) Level 4, 24. Iterierte.

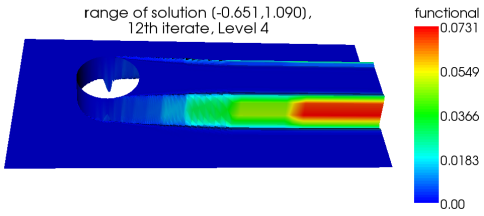


(d) Level 4, 38. Iterierte.

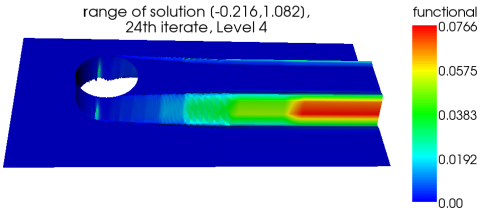
Abbildung 5.12: Fall 3, Minimierung des Funktionals $\Phi_h(y_h) = \left\| f_2\big(S_h(y_h)\big) \right\|_0^2$. Bild der Lösung, Farben nach Größe des Paramters y_T auf jeder Gitterzelle (auf P^1-Finite–Elemente projiziert).



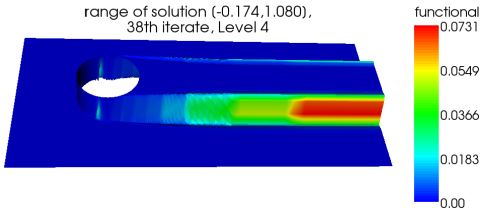
(a) Level 4, 4. Iterierte.



(b) Level 4, 12. Iterierte.

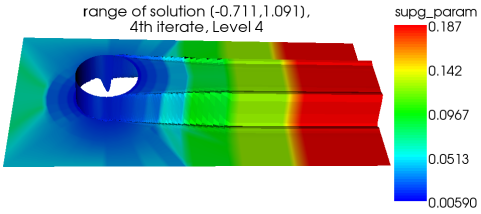


(c) Level 4, 24. Iterierte.

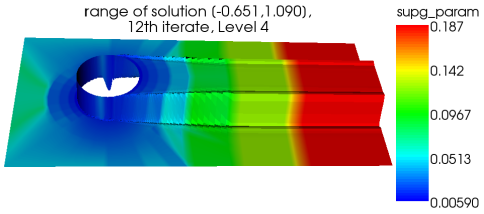


(d) Level 4, 38. Iterierte.

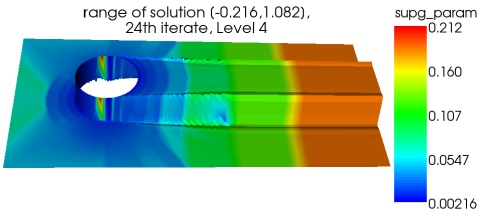
Abbildung 5.13: Fall 4, Minimierung des Funktionals $\Phi_h(y_h) = \beta_0 I_h\big(S_h(y_h)\big) + F_h\big(S_h(y_h)\big)$ mit $\beta_0 = \beta_1 = \beta_2 = 1$. Bild der Lösung, Farben nach Größe des Funktionals auf jeder Gitterzelle (auf P^1-Finite–Elemente projiziert).



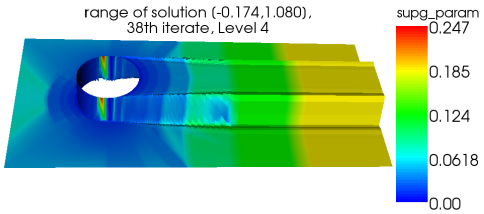
(a) Level 4, 4. Iterierte.



(b) Level 4, 12. Iterierte.

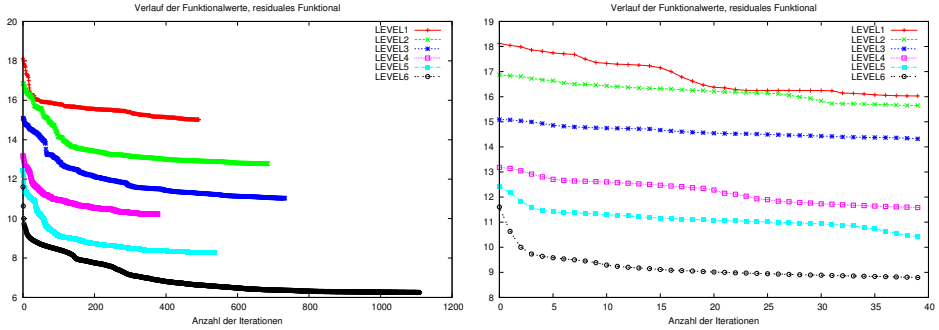


(c) Level 4, 24. Iterierte.



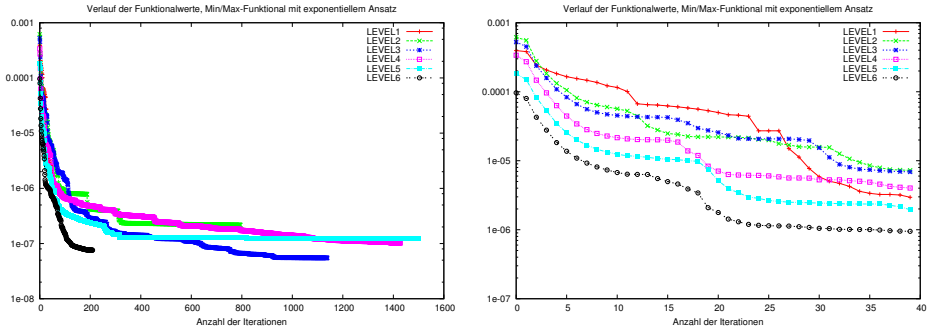
(d) Level 4, 38. Iterierte.

Abbildung 5.14: Fall 4, Minimierung des Funktionals $\Phi_h(y_h) = \beta_0 I_h\big(S_h(y_h)\big) + F_h\big(S_h(y_h)\big)$ mit $\beta_0 = \beta_1 = \beta_2 = 1$. Bild der Lösung, Farben nach Größe des Paramters y_T auf jeder Gitterzelle (auf P^1-Finite–Elemente projiziert).



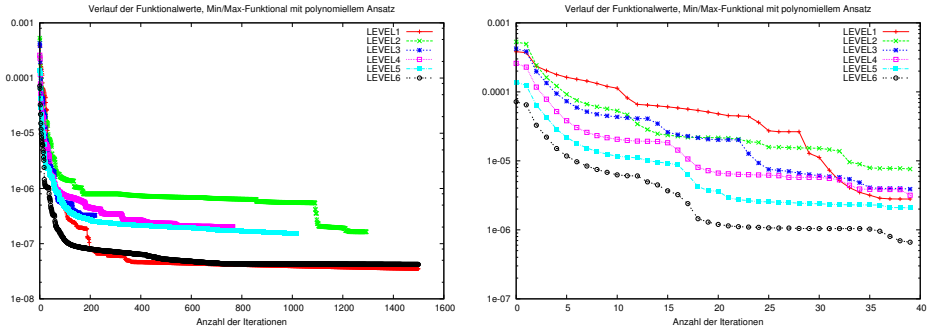
(a) Alle Iterationen. (b) Die ersten 40 Iterationen.

Abbildung 5.15: Fall 1: Verlauf des Funktionals.



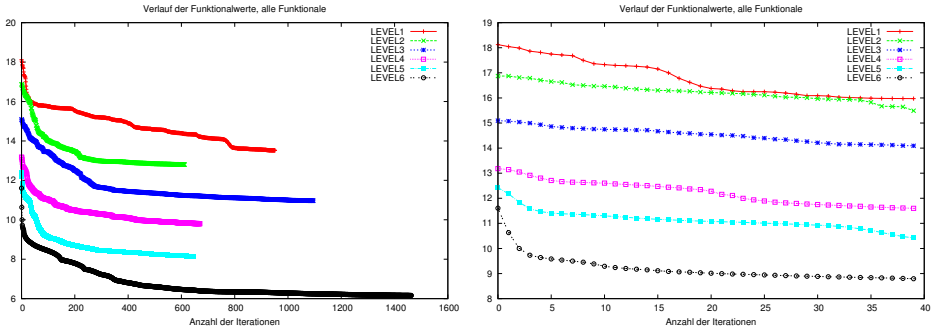
(a) Die ersten 1500 Iterationen. (b) Die ersten 40 Iterationen.

Abbildung 5.16: Fall 2: Verlauf des Funktionals (logarithmische Skala).



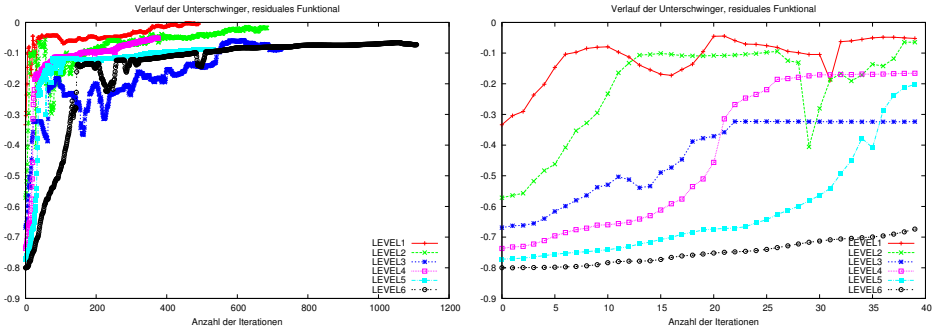
(a) Die ersten 1500 Iterationen. (b) Die ersten 40 Iterationen.

Abbildung 5.17: Fall 3: Verlauf des Funktionals (logarithmische Skala).



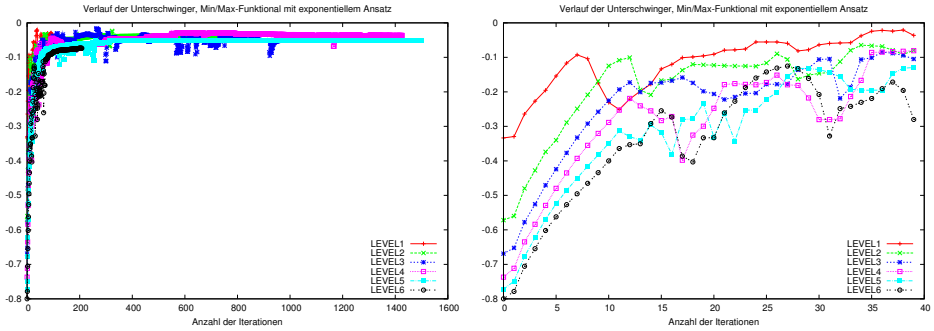
(a) Alle Iterationen. (b) Die ersten 40 Iterationen.

Abbildung 5.18: Fall 4: Verlauf des Funktionals.



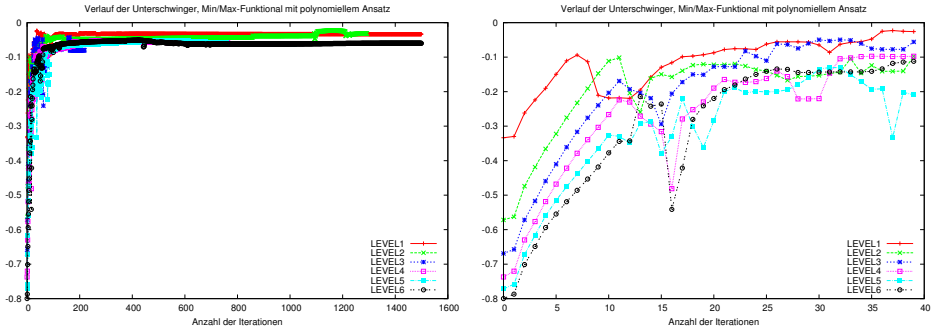
(a) Alle Iterationen. (b) Die ersten 40 Iterationen.

Abbildung 5.19: Fall 1: Verlauf von $\min u_h$, Unterschwinger.



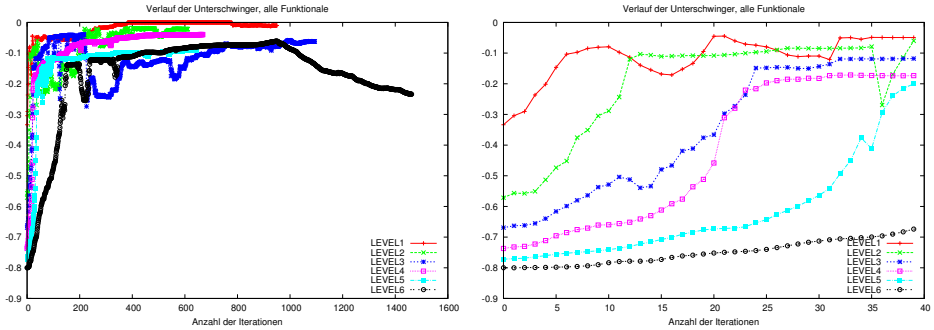
(a) Die ersten 1500 Iterationen. (b) Die ersten 40 Iterationen.

Abbildung 5.20: Fall 2: Verlauf von $\min u_h$, Unterschwinger.



(a) Die ersten 1500 Iterationen. (b) Die ersten 40 Iterationen.

Abbildung 5.21: Fall 3: Verlauf von $\min u_h$, Unterschwinger.



(a) Alle Iterationen. (b) Die ersten 40 Iterationen.

Abbildung 5.22: Fall 4: Verlauf von $\min u_h$, Unterschwinger.

5.3 Auswertung

Fall 1

Im Fall 1 des residualen Funktionals wurden die Ergebnisse aus [JKS11] bestätigt. Die berechnete Lösung hat weniger Über- und Unterschwinger als die SUPG-Lösung. Die Parameter y_h werden im Laufe der Optimierung vor allem oberhalb und unterhalb des Kreises sowie von dort in Stromlinienrichtung entlang der inneren Grenzschicht vergrößert. In allen anderen Bereichen ist die Lösung glatt und die Standardwahl (2.21) bleibt im Wesentlichen unangetastet. Gegenüber den Fällen 2 und 3 ist die Minimierung mit diesem residualen Funktional einfacher. Das heißt es werden in der Regel weniger Schrittweiten getestet, sodass die Rechenzeit pro Iterationen besonders auf feinen Gittern geringer ist.

Fall 2 und 3

Die Funktionale zur Bestrafung von Verletzungen des Minimum- und Maximumprinzips konnten in den ersten Iterationen das Funktional schnell reduzieren, beachte die logarithmische Skala in Abbildung 5.16 und 5.17, jedoch ist besonders auf den feinen Gittern die Konvergenz sehr langsam. Die Qualität der Lösung ist in beiden Fällen vergleichbar, obwohl der exponentialle Ansatz, Fall 2, den SUPG-Parameter stärker erhöhte, siehe Abbildung 5.6 (b) und (c). Es ist allerdings nicht klar, ob andere Wahlen der Parameter α_1, α_2, β_1 und β_2 den exponentiellen (Fall 2) beziehungsweise den polynomiellen Ansatz (Fall 3) noch entscheidend verbessern können. Erste Tests verliefen negativ. Es bleibt außerdem festzuhalten, dass diese beiden Funktionale grobe Über- und Unterschwinger zuverlässig detektieren, siehe Abbildung 5.9 und 5.11. So werden dann auch in den ersten ungefähr Zehn Iterationen die Unterschwinger deutlich reduziert. Dennoch bleiben auch nach vielen Iterationen noch Unterschwinger, siehe links in Abbildung 5.20 und 5.21, die nur geringfügig kleiner ausfallen, als beim residualen Funktional I_h.

Fall 4

Die Ergebnisse mit dem kombinierten Funktional ähneln in den ersten Iterationen stark denen des residualen Funktionals, sodass anzunehmen ist, dass der residuale

Anteil hier dominiert. Dies ist auch nicht ganz überraschend, da sogar eine knotenexakte Lösung u_h zu einem nicht verschwindenen residualen Funktional führt, jedoch ist $F_h(u_h)$ in diesem Fall Null. Dieser Aspekt sollte auch bei der Suche nach anderen geeigneten Funktionalen berücksichtigt werden. Vermutlich muss die gegenseitige Gewichtung der einzelnen Anteile des Funktionals durch die Paramter β_0, β_1 und β_2 noch verbessert werden. Der Grund für die Vergrößerung der Unterschwinger ab etwa 1000 Iterationen bleibt unklar, siehe Abbildung 5.22 links.

6 Ausblick

Die a posteriori Optimierung von Parametern liefert verbesserte Ergebnisse gegenüber der SUPG-Methode mit dem Standardparameter (2.21). Welches Funktional die Qualität geeignet bewertet, wird Gegenstand weiteren Forschung sein. Das vorgeschlagene Funktional zur Bestrafung von Verletzungen des Minimum- und Maximumprinzips kann selbige im Vergleich mit dem residualen Funktional nicht deutlich reduzieren. Jedoch bleibt es interessant, da es Unterschwinger schneller, das heißt in weniger Iterationen, verkleinert.

Um die unphysikalischen Oszillationen an der inneren und der Randgrenzschicht zu reduzieren, ist der vorgestellte Ansatz der a posteriori Optimierung von Parametern ebenfalls geeignet. Trotzdem konnten diese Oszillationen nicht vollständig entfernt werden. Ein Funktional, welches Oszillationen lokal misst, könnte hier Abhilfe schaffen. Vorstellbar sind etwa Bestrafungen von Vorzeichenwechsel des Gradienten. Solch ein Funktional ist jedoch nicht mehr lokal, das heiß es müssen zur Berechnung immer mehrere benachbarte Gitterzellen (Patches) betrachtet werden. Dies ist aufwändig, insbesondere in der Implementation.

In Bereichen, in denen die Lösung glatt ist, werden die Parameter durch die a posteriori Optimierung nicht wesentlich verändert. Daher scheint es sinnvoll diese Parameter zunächst zu identifizieren und anschließend von der Optimierung auszuschließen um die Effizienz dieser Vorgehensweise zu erhöhen.

Die Rechenzeit der a posteriori Optimierung der SUPG-Paramter ist noch sehr groß. Sie ist im Fall 1 des residualen Funktionals um den Faktor 2600 größer als bei der SUPG-Lösung. Hier besteht noch viel Verbesserungsbedarf.

Literaturverzeichnis

[Alt07] Alt, Hans W.: *Lineare Funktionalanalysis: Eine Anwendungsorientierte Einführung.* Springer, 2007

[BH82] Brooks, Alexander N. ; Hughes, Thomas J. R.: Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. In: *Computer Methods in Applied Mechanics and Engineering* 32 (1982), S. 199–259

[Bra07] Braess, Dietrich: *Finite Elemente - Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie.* Springer, 2007

[BS96] Brenner, Susanne C. ; Scott, L. R.: *The Mathematical Theory of Finite Element Methods.* Springer, 1996

[Cia02] Ciarlet, Philippe G.: *The finite element method for elliptic problems.* SIAM, 2002

[Dob10] Dobrowolski, Manfred: *Angewandte Funktionalanalysis.* Springer, 2010

[Eva98] Evans, Lawrence C.: *Partial Differential Equations.* American Mathematical Society, 1998

[GFL+83] Goering, Herbert ; Felgenhauer, Andreas ; Lube, Gert ; Roos, Hans-Görg ; Tobiska, Lutz: *Singularly pertubed differential equations.* Akademie Verlag, 1983

[HB79] Hughes, Thomas J. R. ; Brooks, Alexander N.: A multidimensional upwind scheme with no crosswind diffusion. In: Hughes, Thomas J. R. (Hrsg.): *Finite element methods for convection dominated flows* Bd. 34. AMD,ASME, 1979, S. 19–35

[Hem96] Hemker, Piet W.: A singularly perturbed model problem for numerical computation. In: *Journal Computational and Applied Mathematics* 76 (1996), S. 277–285

[JK07] John, Volker ; Knobloch, Petr: On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations: Part I - a review. In: *Computer Methods in Applied Mechanics and Engineering* 196 (2007), S. 2197–2215

[JKS11] John, Volker ; Knobloch, Petr ; Savescu, Simona B.: A posteriori optimization of parameters in stabilized methods for convection-diffusion problems - Part I. In: *Computer Methods in Applied Mechanics and Engineering* 200 (2011), S. 2916–2929

[JM04] John, Volker ; Matthies, Gunar: MooNMD – a program package based on mapped finite element methods. In: *Computing and Visualization in Science* 6 (2004), S. 163–170

[Joh00] John, Volker: A numerical study of a posteriori error estimators for convection-diffusion equations. In: *Computer Methods in Applied Mechanics and Engineering* 190 (2000), S. 757–781

[Kno09] Knobloch, Petr: On the choice of the SUPG parameter at outflow boundary layers. In: *Preprint No. MATH-knm-2007/3* 31 (2009), S. 369–389

[Luk11] Lukáš, Petr: *Adaptive choice of parameters in stabilization methods for convection-diffusion equations*, Charles University in Prague, Diplomarbeit, 2011

[NW06] Nocedal, Jorge ; Wright, Stephen J.: *Numerical Optimization.* 2. Auflage. Springer, 2006

[RST08] Roos, Hans-Görg ; Stynes, Martin ; Tobiska, Lutz: *Robust Numerical Methods for Singularly Perturbed Differential Equations.* Springer, 2008

[Ver05] Verfürth, Rüdiger.: Robust a posteriori error estimates for stationary convection-diffusion equations. In: *SIAM J. Numer. Anal.* 43 (2005), S. 1766–1782

Erklärung

Hiermit versichere ich, diese Masterarbeit selbstständig und nur unter Verwendung der im Literaturverzeichnis angegebenen Hilfsmittel angefertigt zu haben.

Berlin, den 14. September 2011