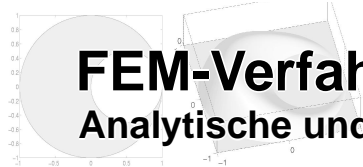


Algebraische Gewichtsfunktion
(nach Kantorowitsch und Krylow)

Geglättete Abstandsfunktion
(Plateau-Funktion)



FEM-Verfahren mit web-Spline-Basis

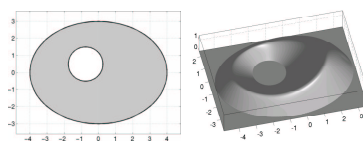
Analytische und numerische Behandlung geeigneter Gewichtsfunktionen

Algebraische Gewichtsfunktion auf einem
sichelförmigen Gebiet

Wissenschaftliche Arbeit von Winfried Geis

$$w(x, y) = (a^2 - x^2 - y^2)(x^2 - ax + y^2)$$

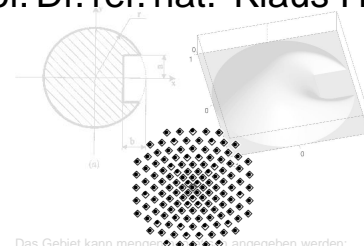
Integraldarstellung der
Gewichtsfunktion



$$w(X)^{-1} = \int_{\partial\Omega_P} \frac{ds_P}{\|X - P(s)\|^2} \quad \text{mit } X \in \mathbb{R}^2$$

R-Funktionen nach Rvachev

Betreuung:
Prof. Dr. rer. nat. Klaus Hölbig



Das Gebiet kann mangel angegeben werden:

$$\Omega = w_1 \cap (w_2 \cap w_3)$$

$$w_1 = \frac{1}{2r}(r^2 - x^2 - y^2) \geq 0$$

$$w_2 = x - r + b \geq 0$$

$$w_3 = a^2 - y^2 \geq 0$$

Mathematisches Institut A
Universität Stuttgart
20. Dezember 2001

Zusammenfassung

In technischen Anwendungen treten häufig Probleme der mathematischen Physik auf, die mittels analytischer Methoden entweder gar nicht, oder nur sehr umständlich gelöst werden können. Da ein reales Problem ohnehin nur innerhalb gewisser Fehlertoleranzen betrachtet werden kann, eignen sich Näherungsmethoden zur Lösung von Differentialgleichungen wie z.B. das Finite Differenzen- oder die Finite Element-Verfahren (beide lassen sich sehr effizient mit dem Computer umsetzen). Durchgesetzt haben sich in der Praxis die Finiten Elemente, die in Anlehnung an die klassischen Konzepte von Ritz und Galerkin von den Ingenieuren Argyris, Martin und Clough begründet wurden.

Im ersten Teil dieser Arbeit werde ich zunächst einen kurzen Überblick über Standarddiskretisierungsverfahren geben - Ritz-Galerkin-Verfahren und das Verfahren der Finiten Elemente (FEM) - um dann auf die **web-Methode** zu sprechen zu kommen, eine neue Methode, deren Vorteile zum einen in der einfachen Gittergenerierung (die web-Methode benutzt im Gegensatz zu Standardmethoden ein gleichmäßiges Gitter von Quadraten), zum anderen in der oft schnelleren Konvergenz im Vergleich zu Standard FEM-Verfahren liegt. Anders als die bekannten FEM-Verfahren arbeitet die web-Methode nicht mit stückweise linearen Hutfunktionen bzw. Polynomen höheren Grades, sondern mit kubischen B-Splines, welche mit einer geeigneten Abschneidefunktion w multipliziert werden, wobei *geeignet* bedeutet, dass sie zum einen den Gebietsrand möglichst gut approximiert, so dass gilt:

$$\begin{aligned} w(X) &\equiv 0 \quad \forall X \in \partial\Omega \\ \text{hier} \quad w(X) &< 0 \quad \forall X \in \Omega^{\text{G}} \\ w(X) &> 0 \quad \forall X \in \overset{\circ}{\Omega} \end{aligned} \tag{1}$$

zum anderen, dass sie glatt über das Gebiet gespannt ist. Bedingung (1) ist im Allgemeinen nicht notwendig, in unserem Fall aber recht nützlich, da sie auf eine glatte Fortsetzung ausserhalb des Gebietes Ω führt. Eine mögliche Gewichtsfunktion

$$w(X)^{-1} = \int_{\partial\Omega} \frac{ds_p}{\|X - P(s)\|^2}$$

wird daher - dies ist der zentrale Teil dieser Arbeit - auf ihre Brauchbarkeit hinsichtlich der web-Methode überprüft werden.

Im praktischen Teil meiner Diplomarbeit wird es dann darum gehen, die Gewichtsfunktion an einzelnen Stellen auszuwerten, und diese mittels Quasiinterpolanten zu approximieren. Dabei wird es wichtig sein, die nötigen Integrationen durch effiziente Verfahren mit einer möglichst hohen Genauigkeit (vor allem in den Randpunkten) durchzuführen. Das Hauptproblem liegt dabei darin, dass der Kehrwert, über den die Gewichtsfunktion definiert ist, in den Randpunkten eine Polstelle aufweist, wobei gerade in diesen Punkten eine hohe Genauigkeit erwünscht ist. Dies erfordert eine Anpassung des numerischen Integrationsverfahrens.

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 1 | Standarddiskretisierungsverfahren | 8 |
| 1.1 | Rayleigh-Ritz-Galerkin-Verfahren | 8 |
| 1.1.1 | Vorbemerkungen | 8 |
| 1.1.2 | Das eigentliche Verfahren | 12 |
| 1.2 | Standard Finite Element Methoden | 14 |
| 1.2.1 | FEM mit Dreiecksgittern | 15 |
| 1.2.2 | FEM mit Vierecksgittern | 20 |
| 2 | Die web-Methode | 24 |
| 2.1 | Splines - Eine kurze Einführung | 24 |
| 2.1.1 | Suche nach einer geeigneten Basis - die abgebrochenen Potenzen | 25 |
| 2.1.2 | B-Splines | 28 |
| 2.2 | Die Methode | 33 |
| 2.2.1 | Einführung | 33 |
| 2.2.2 | Stabilität der B-Spline-Basis | 35 |
| 2.2.3 | Weighted Extended B-Splines | 38 |
| 2.2.4 | Stabilität und Approximationsordnung | 40 |
| 2.2.5 | Die Rolle der Gewichtsfunktion für die web-Methode | 41 |
| 2.2.6 | R-Funktionen nach Rvachev | 42 |
| 2.2.7 | Geglättete Abstandsfunktion | 44 |
| 3 | Die Gewichtsfunktion $w(X)$ | 46 |
| 3.1 | Annahmen und geometrisches Modell | 46 |
| 3.2 | Einige Hilfssätze | 47 |
| 3.3 | Glattheitseigenschaften | 50 |
| 3.4 | Auswertung der Ableitung auf dem Rand ($\nabla u \neq 0$ auf $\partial\Omega$) | 53 |
| 4 | Numerische Umsetzung | 57 |
| 4.1 | Theoretische Vorarbeit | 57 |
| 4.1.1 | Bézier-Technik für Kurven | 57 |

| | |
|---|-----------|
| <i>INHALTSVERZEICHNIS</i> | 5 |
| 4.2 Die Vorgehensweise im Überblick | 62 |
| 4.3 Programmcode und Erläuterungen | 63 |
| 4.3.1 Markierung der randnahen Punkte | 63 |
| 4.3.2 Berechnung der Gewichtsfunktion auf den Gitterpunkten | 68 |
| 4.3.3 Quasi-Interpolation der diskreten Punkte | 76 |
| Literaturverzeichnis | 83 |
| Erklärung | 85 |

Liste der Programme und Programmteile

| | | |
|---|---|----|
| 1 | Initialisierung: Entwurf eines Gitters und Markierung randnaher Punkte | 66 |
| 2 | Integration und glatte Fortsetzung des Gewichts | 73 |
| 3 | Romberg-Integration (web_romberg_invdif) | 74 |
| 4 | Steuerung für geschachtelten Romberg (web_weight_qifc_schachtel) . . | 75 |
| 5 | Quasi-Interpolation und Berechnung der Gewcihtsfunktion an vorgegebenen Werten, In-Out-Test | 81 |

Symbolerklärung

| | |
|-------------------------------------|--|
| $\text{dist}(X)$ | Geglättete Abstandsfunktion (hier oft als euklidischer Abstand verwendet) |
| $w(X)$ | Gewichtsfunktion über einem Gebiet Ω ; hier meist $w(X)^{-1} = \int_{\partial\Omega} \frac{ds_P}{\ X-P(s)\ ^2}$ |
| Ω | Gebiet |
| $\partial\Omega$ | Rand des Gebietes Ω |
| $\overset{\circ}{\Omega}$ | Inneres des Gebietes Ω |
| $\mathbf{K} := (\mathbf{K}_{ij})$ | Steifigkeitsmatrix |
| V | Raum der Testfunktionen |
| V_n | endlich-dimensionaler Unterraum von V |
| $\{\phi_1, \dots, \phi_n\}$ | Basis des Raumes V_n , bestehend aus Testfunktionen |
| $a(u, v)$ | Bilinearform zur Variationsformulierung von $Lu = f \Rightarrow a(u, v) = (f, v)$ |
| H^m | Sobolev-Raum mit Funktionen u in $L_2(\Omega)$, die schwache Ableitungen $\partial^\alpha u$ für alle $ \alpha \leq m$ besitzen. |
| H_0^m | $u \in H^m$ mit kompaktem Träger |
| $\ \cdot\ _{H^1}$ | H^1 -Norm |
| $ \cdot _{H^1}$ | H^1 -Halbnorm |
| (\cdot, \cdot) | Euklidisches Skalarprodukt |
| $f \preceq g$ | $\exists c : f \leq cg$ mit c unabhängig von h . Analog \succeq |
| $f \asymp g$ | $f \preceq g$ und $f \succeq g$ |
| $b_{j,m}$ | B-Spline Basisfunktion vom Grad m mit dem Träger $[u_j, u_{j+m+1})$ |
| $S_{n,\mathcal{U}}(\Omega)(\Omega)$ | Spliner Raum über Ω zur Knotenfolge $\mathcal{U} = (u_j)_j$ mit Grad n |
| L | Differentialoperator zweiter Ordnung, z.B. $L(v) := -\text{div}(p\nabla v) + qv$ |
| \mathcal{C}_0^∞ | Raum der Testfunktionen |
| $\ v\ _E$ | $\ v\ _E := \sqrt{a(v, v)} \quad \forall v \in V$, die Energienorm |

Kapitel 1

Standarddiskretisierungsverfahren

1.1 Rayleigh-Ritz-Galerkin-Verfahren

1.1.1 Vorbemerkungen

Bevor wir mit dem eigentlichen Verfahren beginnen können, muss noch etwas Vorarbeit geleistet werden. Betrachten wir die Differentialgleichung

$$\begin{aligned} Lu &= f \quad \text{auf } \Omega \\ u &= g \quad \text{auf dem Rand } \partial\Omega \\ p, q &\in \mathcal{C}(\bar{\Omega}) \quad \text{mit } p(x) > 0, q(x) \geq 0 \end{aligned}$$

wobei der Differentialoperator durch

$$L(v) := -\operatorname{div}(p\nabla v) + qv$$

gegeben ist. Diese Formulierung beinhaltet die am häufigsten behandelten Modellfälle Laplace-, Poisson und Helmholtz-Gleichung. u bildet vom Definitionsbereich Ω nach ab. Im Allgemeinen kann man sich darauf beschränken, Probleme der Art

$$\begin{aligned} Lu &= f \quad \text{auf } \Omega \\ u &= 0 \quad \text{auf dem Rand } \partial\Omega \end{aligned}$$

mit homogenen Randdaten zu betrachten. Führt man nämlich eine beliebig glatte Fortsetzung \tilde{g} der inhomogenen Randdaten ein, so kann u auch geschrieben werden als

$$u = \tilde{u} + \tilde{g} \quad \text{mit } \tilde{g} = g \quad \forall \quad x \in \partial\Omega.$$

Eingesetzt erhält man:

$$\begin{aligned} & -\operatorname{div}(p\nabla u) + qu = f \\ \Leftrightarrow & -\operatorname{div}(p\nabla(\tilde{u} + \tilde{g})) + q(\tilde{u} + \tilde{g}) = f \\ \Leftrightarrow & -\operatorname{div}(p\nabla\tilde{u}) + q\tilde{u} = \underbrace{f + \operatorname{div}(p\nabla\tilde{g}) - q\tilde{g}}_{=\tilde{f}} \quad \text{mit } \tilde{u} = 0 \quad \forall \quad x \in \partial\Omega \end{aligned}$$

Dies wird für die weiteren Betrachtungen von erheblichem Vorteil sein.

Das L^2 -Skalarprodukt sei nun wie gewohnt definiert durch $(u, v) := \int_{\Omega} uv$. Gilt $Lu = f$ mit $u = 0$ auf dem Rand, so wird u auch die Gleichung

$$(Lu, v) = (f, v) \quad \forall \quad v \in V \quad (1.1)$$

erfüllen. V sei dabei eine Teilmenge des Raumes der Testfunktionen C_0^∞ .

$$V := H_0^1 = \{v : v \in H^1(\Omega) \quad \wedge \quad v(x) \equiv 0 \quad \forall \quad x \in \partial\Omega\}.$$

Das folgende Lemma zeigt, dass (1.1) sich als symmetrisches Variationsproblem schreiben lässt.

Lemma 1.1.1. *L ist ein symmetrischer Operator auf dem Definitionsbereich V , und es gilt*

$$(u, L(v)) = (L(u), v) \quad \forall \quad u, v \in V$$

Beweis: Beweisen lässt sich dies mittels partieller Integration. Im Mehrdimensionalen entspricht dies der Verwendung des Satzes von Green (nebenbei bemerkt sollte dafür der Rand des Gebietes Ω , $\partial\Omega$, stückweise glatt sein und die Kegelbedingung erfüllen, das heißt: die Innenwinkel der Gebietsecken seien positiv, sodass man einen Kegel mit positivem Scheitelwinkel so in Ω verschieben kann, dass er die Ecken berührt).

$$\begin{aligned} (v, Lu) &= (v, f) \\ &= \int_{\Omega} v [-\operatorname{div}(p\nabla u) + qu] \, dx \\ &= \int_{\Omega} -v \operatorname{div}(p\nabla u) \, dx + \int_{\Omega} vqu \, dx \\ &= \int_{\Omega} \nabla v p \nabla u \, dx - \underbrace{\int_{\partial\Omega} \frac{\partial u}{\partial n} v \, dF}_{=0} + \int_{\Omega} vqu \, dx, \quad \text{wobei} \quad \frac{\partial u}{\partial n} := \mathbf{n}(\nabla \mathbf{u}) \\ &= \int_{\Omega} [p \nabla v \nabla u + quv] \, dx \end{aligned} \quad (1.2)$$

$$= (Lu, v) \quad \text{wegen Symmetrie} \quad (1.3)$$

□

Die rechte Seite (1.3) ist für alle $v \in V$ definiert und liefert uns eine symmetrische Bilinearform:

$$a(u, v) := \int_{\Omega} [p \nabla u \nabla v + quv] \, dx. \quad (1.4)$$

Damit erhält man die schwache Formulierung (unter Voraussetzung der Symmetrie äquivalent zur Variationsformulierung $J(u) = \frac{1}{2}a(u, u) - F(u) \rightarrow \min$) des Problems

$$a(u, v) = (f, v) = F(v) \quad \forall \quad v \in V \quad (1.5)$$

mit der Bilinearform a und dem Funktional F im Dualraum V' .

Um die Existenz und Eindeutigkeit der Lösung des Problems (1.5) zeigen zu können (dies ist gleichbedeutend mit der positiven Definitheit von $a(.,.)$), benötigen wir die Poincaré-Friedrichs-Ungleichung:

Theorem 1.1.1 (Poincaré-Friedrichs-Ungleichung). *Sei Ω in einem n -dimensionalen Würfel W der Kantenlänge s enthalten. Dann ist*

$$\|v\|_{L_2} \leq s |v|_1 \quad \forall \quad v \in H_0^1(\Omega)$$

beziehungsweise
$$\int_{\Omega} v^2 dx = s \int_{\Omega} \sum_{i=1}^n \left| \frac{\partial v}{\partial x_i} \right|^2 dx$$

Beweis: (vgl. [1] S.29)

$C_0^\infty(\Omega)$ liegt dicht in $H_0^1(\Omega)$. Es genügt daher, die Ungleichung für $v \in C_0^\infty$ zu zeigen. Wir haben angenommen, dass $\Omega \subset W := \{(x_1, x_2, \dots, x_n); 0 < x_i < s\}$ und v auf $W \setminus \Omega$ verschwindet. Es gilt:

$$v(x_1, x_2, \dots, x_n) = v(0, x_2, \dots, x_n) + \int_0^{x_1} \partial_1 v(t, x_2, \dots, x_n) dt \quad (1.6)$$

Der Randterm verschwindet nach Voraussetzung, und die Cauchy-Schwarz'sche Ungleichung liefert

$$\begin{aligned} |v(x)|^2 &\leq \int_0^{x_1} 1^2 dt \int_0^{x_1} |\partial_1 v(t, x_2, \dots, x_n)|^2 dt \\ &\leq \underbrace{\int_0^s 1^2 dt}_{=s} \int_0^{x_1} |\partial_1 v(t, x_2, \dots, x_n)|^2 dt \end{aligned}$$

Integriere nun die Ungleichung. Da die rechte Seite unabhängig von x_1 ist und damit wie eine Konstante behandelt werden kann, folgt

$$\begin{aligned} \int_0^s |v(x)|^2 dx_1 &\leq \int_0^s \left(s \int_0^{x_1} |\partial_1 v(t, x_2, \dots, x_n)|^2 dt \right) dx_1 \\ &= s^2 \int_0^s |\partial_1 v(x_1, x_2, \dots, x_n)|^2 dx_1 \\ &= s^2 \int_0^s |\partial_1 v(x)|^2 dx_1 \end{aligned}$$

Schließlich wird über alle Koordinaten integriert.

$$\begin{aligned} \int_W |v|^2 dx &\leq s^2 \int_W |\partial_1 v|^2 dx \\ &\leq \sum_{1 \leq j \leq n} \int_W |\partial_j v|^2 dx \\ &= s^2 |v|_1^2 \end{aligned}$$

□

Damit haben wir das Handwerkszeug, um die positive Definitheit der Bilinearform zeigen zu können. Aus der Definition wissen wir, dass

$$Lu = -\operatorname{div}(p\nabla u) + qu.$$

Setze in die Bilinearform (1.4) ein, wobei $p_0 := \min\{p(x), \forall x \in \Omega\}$ und k eine lediglich von der Gebietsgröße abhängige Konstante sein soll.

$$\begin{aligned} |a(u, u)| &= \int_{\Omega} p |\nabla u|^2 + qu^2 dx \\ &= \int_{\Omega} p \nabla u^2 dx + \int_{\Omega} qu^2 dx \\ &\geq \int_{\Omega} p \nabla u^2 dx \geq p_0 \int_{\Omega} \nabla u^2 dx \\ &= p_0 \int_{\Omega} \sum_{i=1}^n \left(\frac{\partial u}{\partial x_i} \right)^2 dx \\ &\geq \frac{p_0}{k} \int_{\Omega} u^2 dx = \frac{p_0}{k} \|u\|_{L_2}^2 \quad (\text{Poincaré-Friedrichs}) \end{aligned}$$

Die Bilinearform ist also V -elliptisch. Zeige nun die Stetigkeit von a , also dass $|a(u, v)| \leq k \|u\| \|v\|$, wobei aufgrund der Stetigkeit eine obere Schranke p_{∞} für p und q_{∞} für q existiert. Es sei außerdem $s := \max(p_{\infty}, q_{\infty})$.

$$\begin{aligned} |a(u, v)| &= \left| \int_{\Omega} p \nabla u \nabla v + \int_{\Omega} quv \right| \\ &\leq p_{\infty} \left| \int_{\Omega} \nabla u \nabla v \right| + q_{\infty} \left| \int_{\Omega} uv \right| \\ &\leq s (\|\nabla u\|_2 \|\nabla v\|_2 + \|u\|_2 \|v\|_2) \quad (\text{Cauchy-Schwarz}) \\ &= s (|u|_{H^1} |v|_{H^1} + \|u\|_{H^0} \|v\|_{H^0}) \\ &\leq s (\|u\|_{H^1} \|v\|_{H^1} + \|u\|_{H^1} \|v\|_{H^1}) \\ &= 2s \|u\|_{H^1} \|v\|_{H^1} \end{aligned}$$

Damit ist gezeigt, dass die Bilinearform stetig ist. Das Funktional $F(v)$ lässt sich dann wie folgt abschätzen:

$$\begin{aligned} F(u) &= \int_{\Omega} fu \quad dx \\ |F(u)| &= \left| \int_{\Omega} fu \right| \\ &\leq \|f\|_2 \|u\|_2 \quad (\text{Cauchy-Schwarz}) \\ &\leq \underbrace{\|f\|_2}_{=C} \tilde{c} |u|_{H^1} \quad (\text{Poincaré-Friedrichs-Ungleichung}) \\ &\leq C \|u\|_{H^1} \end{aligned}$$

F ist also auch ein stetiges Funktional, und somit sind die Voraussetzungen für das Lemma von Lax-Milgram gegeben. Damit folgt die Existenz und Eindeutigkeit der Lösung.

Lemma 1.1.2 (Lax-Milgram). *Sei $(V, (\cdot, \cdot))$ ein Hilbertraum, $a(\cdot, \cdot)$ eine positiv definite Bilinearform und $F \in V'$ ein lineares Funktional. Dann gibt es eine eindeutige Lösung $u \in V$, sodass gilt:*

$$a(u, v) = F(v) \quad \forall v \in V$$

Der Beweis dieser Lemmas kann in den meisten Standard-FEM-Werken (siehe auch [1], [2], [4]) nachgelesen werden. Da z.B. das Riesz'sche Darstellungstheorem verwendet wird, welches aus Gründen der Vollständigkeit wiederum bewiesen werden müsste, verzichte ich an dieser Stelle auf den Beweis, da dies zum einen der Systematik nicht zuträglich ist, zum anderen zum Verständnis der folgenden Kapitel nichts Wesentliches beizutragen vermag.

1.1.2 Das eigentliche Verfahren

Der Raum H_0^1 ist ein unendlich-dimensionaler Raum. D.h. die Beziehung $a(u, v) = (f, v)$ ist einer numerischen Berechnung nicht zugänglich. Deshalb werden wir jetzt eine Diskretisierung durchführen, also nur endlich viele Testfunktionen einsetzen. Dies lässt sich auch für eine große Anzahl noch recht gut mit dem Computer berechnen.

Nehmen wir nun einen endlich-dimensionalen Funktionenraum $V_n \subset V$. Die Idee des Verfahrens ist, ein $u_s \in V_n$ zu suchen, sodaß

$$a(u_s, v) = (f, v) \quad \forall v \in V_n \tag{1.7}$$

Konkret berechnet wird dies, indem man sich zunächst eine Basis aus Testfunktionen wählt (ϕ_1, \dots, ϕ_n) . Die Lösung der Gleichung (1.7) wird dabei in der Form $u_s = \sum_{j=1}^n U_j \phi_j$ gesucht. Sei weiter

$$\begin{aligned} K_{ij} &= a(\phi_j, \phi_i) \\ F_i &= (f, \phi_i) \quad i = 1 \dots n \\ \mathbf{U} &= (U_j), \end{aligned}$$

wobei $\mathbf{K} = (K_{ij})$ die sogenannte Steifigkeitsmatrix definiert. Dann ist die Lösung des LGS

$$\mathbf{KU} = \mathbf{F} \tag{1.8}$$

äquivalent zur Lösung des mit Gleichung (1.7) verbundenen Problems, wie unschwer nachzurechnen ist.

Theorem 1.1.2. *Sei $f \in L^2(\Omega)$, dann hat die Gleichung (1.7) eine eindeutige Lösung. Wobei anzumerken ist, dass im endlichdimensionalen Fall - wie hier - Eindeutigkeit und Existenz äquivalent zueinander sind.*

Beweis:. Betrachte das zu (1.7) äquivalente System (1.8).

Annahme: Es gibt ein $\tilde{u} \in V$, $\tilde{u} \neq 0$ mit zugehörigem \tilde{U} (dem Koeffizientenvektor), sodass gilt

$$\mathbf{K}\tilde{U} = \mathbf{0}$$

Sei $\tilde{u} = \sum_j \tilde{U}_j \phi_j$. Aus der Äquivalenz von (1.7) zu (1.8) folgt, dass

$$\begin{aligned} a(\tilde{u}, \phi_j) &= 0 \quad \forall j \quad \text{und damit} \\ \tilde{U}_j a(\tilde{u}, \phi_j) &= 0 \\ &= a(\tilde{u}, \tilde{U}_j \phi_j) \\ \sum_j a(\tilde{u}, \tilde{U}_j \phi_j) &= a(\tilde{u}, \tilde{u}) = 0 \\ a(\tilde{u}, \tilde{u}) &= \int_{\Omega} (\nabla \tilde{u})^2 \\ &\Rightarrow \nabla \tilde{u} = \mathbf{0} \end{aligned} \tag{1.9}$$

\tilde{u} ist also konstant. Da wir aber nach Voraussetzung wissen, dass $\tilde{u}|_{\partial\Omega} \equiv 0$ folgt damit, dass $\tilde{u} \equiv 0$. Damit ist die Annahme widerlegt, und es folgt die Existenz und Eindeutigkeit der Lösung. \square

Trotz Eindeutigkeit stellt sich aber die Frage, ob die erhaltene Lösung u_s überhaupt Sinn ergibt, sprich, ob die Lösung u_s den Fehler bezüglich einer zum Raum V verträglichen Norm minimiert. Dies sollen die folgenden Betrachtungen klären.

Der Fehler der diskreten Lösung $u - u_s$ erfüllt eine Orthogonalitätsrelation. Es gilt

$$a(u - u_s, v) = 0 \quad \forall \quad v \in V_n$$

Definition 1.1.1. Die Norm

$$\|v\|_E := \sqrt{a(v, v)} \quad \forall \quad v \in V$$

wird *Energienorm* genannt.

Die Energienorm ist zu der Norm auf V äquivalent. Sie erfüllt als Skalarproduktnorm die Cauchy-Schwarz-Ungleichung:

$$\begin{aligned} |a(u, v)| &\leq \|u\|_E \|v\|_E \quad \forall \quad u, v \in V \\ &\leq C \|u\|_E \|v\|_E \quad C \text{ konstant} \end{aligned}$$

Damit können wir das folgende Lemma beweisen.

Lemma 1.1.3. Die diskrete Lösung u_s des Problems minimiert den Fehler zur exakten Lösung u in der Energienorm.

Beweis: (siehe auch [2]).

Es sei u_s die diskrete Näherungslösung zu u .

$$\begin{aligned}\|u - u_s\|_E^2 &= a(u - u_s, u - u_s) \\ &= a(u - u_s, u - v) + a(u - u_s, v - u_s) \quad \text{Orthogonalität im zweiten Glied} \\ &= a(u - u_s, u - v) \\ &\leq \|u - u_s\|_E \|u - v\|_E\end{aligned}$$

Für $\|u - u_s\|_E = 0$ ist die Ungleichung trivial. Ansonsten ergibt sich daraus die Abschätzung

$$\begin{aligned}\|u - u_s\|_E &\leq \|u - v\|_E \quad \forall \quad v \in V_n \\ \|u - u_s\|_E &\leq \inf\{\|u - v\|_E : v \in V_n\} \\ \inf\{\|u - v\|_E : v \in V_n\} &\leq \|u - u_s\|_E \quad \text{da } u_s \in V_n \\ \|u - u_s\|_E &= \min\{\|u - v\|_E : v \in V_n\}\end{aligned}$$

Dabei darf hier \inf durch \min ersetzt werden, denn u_s wird ja tatsächlich angenommen. \square

Wir haben nun Existenz und Eindeutigkeit der diskreten Lösung gezeigt, und wissen, dass sie den Fehler minimiert. Was noch fehlt ist eine vernünftige Fehlerabschätzung. Nach dem Lemma von Céa gilt:

Lemma 1.1.4. Die Bilinearform a sei V -elliptisch ($H_0^m(\Omega) \subset V \subset H^m(\Omega)$) und u bzw. u_h seien die Lösungen des Variationsproblems in V bzw. in V_n . Dann ist

$$\|u - u_h\|_{H^m} \leq \frac{C}{\alpha} \inf_{v_h \in V_n} \|u - v_h\|_{H^m}$$

wobei α die Konstante aus der V -Elliptizitätsbedingung und C aus der Stetigkeitsbedingung ist.

Das heißt, dass es in der Regel von nicht geringer Wichtigkeit ist, welchen Unterraum V_n man wählt. In der Praxis wird meistens der Raum der Polynome verwendet (von diesen wissen wir, dass sie jede stetige Funktion beliebig gut approximieren können). Dabei wird weniger der Polynomgrad in die Höhe getrieben, sondern man verwendet stückweise Polynome oder Polynome niedrigen Grades - ein Hauptmerkmal der FEM.

1.2 Standard Finite Element Methoden

Die Finite Element-Methode (FEM) beruht wie schon erwähnt auf der zuvor besprochenen Rayleigh-Ritz-Galerkin-Methode, wobei Funktionen mit genügend kleinem Träger verwendet werden. Auf einige Besonderheiten werde ich in diesem Kapitel eingehen. Da es hierbei nur um das grobe Verständnis gehen soll, werde ich mich auf

den 2D-Fall beschränken. Als Modellproblem soll das Poisson-Problem zum Zuge kommen - für andere Probleme ist die Vorgehensweise recht ähnlich.

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega \\ u &= 0 \quad \text{auf } \partial\Omega \end{aligned}$$

Oftmals ist nicht ganz klar, was die Bezeichnung „Finites Element“ eigentlich bedeutet, ob also damit das einzelne Teilstück von Ω oder die Formfunktion auf diesem, oder die Kombination der beiden gemeint ist. Ich werde deshalb meist nicht auf diesen Begriff zurückgreifen, sondern versuchen, eindeutige Bezeichnungen zu wählen.

Um einen endlich-dimensionalen Unterraum der Testfunktionen zu erzeugen beginnt man bei der Finiten Element Methode damit, das Gebiet Ω in kleinere Einheiten zu unterteilen. Im zweidimensionalen Fall zerlegt man entweder in Dreiecke (Triangulierung) oder Vierecke und zwar so, dass sie folgenden Forderungen genügen:

Definition 1.2.1 (Zulässige Zerlegung). Die Zerlegung \mathcal{T} des Gebietes $\bar{\Omega}$ ist zulässig, falls gilt:

- (T 1) $\cup_i K_i = \bar{\Omega}$ mit $K_i \in \mathcal{T}$.
- (T 2) Besteht der Schnitt zweier K_i aus genau einem Punkt, dann ist dieser Punkt ein Eckpunkt beider Drei- bzw Vierecke.
- (T 3) Besteht der Schnitt zweier K_i aus mehr als einem Punkt, dann ist die Schnittmenge eine komplette Kante der jeweiligen K_i .
- (T 4) Für jedes $K \in \mathcal{T}$ gilt: K ist geschlossen und das Innere von K ist nicht leer.
- (T 5) Seien K_1 und K_2 verschieden, dann gilt: $\overset{\circ}{K}_1 \cap \overset{\circ}{K}_2 = \emptyset$.
- (T 6) Hat jedes der Elemente K_i einen Durchmesser von höchstens $2h$, dann schreibt man auch \mathcal{T}_h anstatt \mathcal{T} .
- (T 7) Bezeichnet h_K den halben Durchmesser eines Elements, so heißt eine Zerlegung \mathcal{T}_h quasiuniform, wenn es eine Zahl κ gibt, so dass jedes Element K von \mathcal{T}_h einen Kreis mit Radius $\rho_K = \frac{h_K}{\kappa}$ enthält.
- (T 8) Ersetzt man im vorherigen Fall h_K durch h so erhält man Uniformität der Zerlegung.

Hat man nun das Gebiet unterteilt, so gibt es unzählige Möglichkeiten, auf den einzelnen Teilstücken Ansatzfunktionen zu erklären. Dazu aber im Folgenden mehr.

1.2.1 FEM mit Dreiecksgittern

Eine Möglichkeit ist, wie schon gesagt, das vorliegende Gebiet Ω in Dreiecke zu unterteilen - also zu triangulieren. Eine besonders einfache Methode, die den Anforderungen an eine FEM-Methode gerecht wird wurde von Courant im Jahr 1943 gefunden.

Beispiel 1.2.1 (Courant). Die einfachste Möglichkeit, Basiselemente (sprich Dreieck gepaart mit Basisfunktion) zu finden, ist, die Knotenpunkte einfach in die Ecken der Dreiecke zu setzen – wir erzeugen dabei sogenannte C^0 -Elemente, also Elemente, die zumindest stetige Übergänge zum nächsten Element darstellen können. Betrachten wir den einfachen Fall, dass

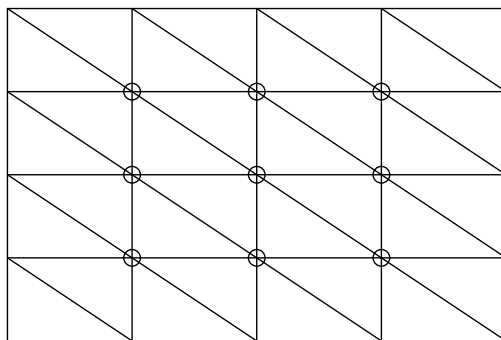


Abbildung 1.1: Gleichmäßige Triangulierung eines Gebietes Ω mit Schrittweite h . Die Knotenpunkte sind umringelt hervorgehoben.

Ω ein Rechteck darstellt, dann lässt sich sehr einfach sogar eine gleichmäßige Triangulierung mit fester Schrittweite h vornehmen. In einem solchen Fall wird der Raum der zugehörigen Testfunktionen auch gerne mit einem h im Index gekennzeichnet

$$S_h := \{v \in C(\Omega); v \text{ ist in jedem Dreieck linear, und } v = 0 \text{ auf } \partial\Omega\}$$

Da die Ansatzfunktionen linear sind, ist ein v schon durch die drei Eckpunkte eines Elements eindeutig festgelegt, denn mit $v \in S_h$ hat v die Form $v(x, y) := a + bx + cy$. Ist N die Anzahl der inneren Gitterpunkte (x_i, y_i) , dann ist eine mögliche Basis durch $\{\Psi_i\}_{i=1}^N$ gegeben mit

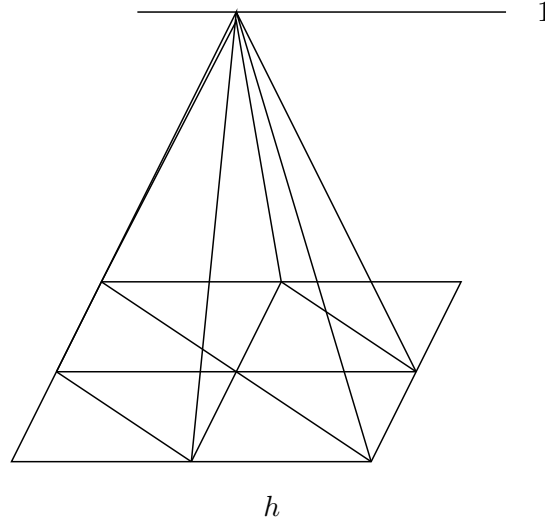
$$\Psi_i(x_j, y_j) = \delta_{ij}$$

Wobei die jeweilige Basisfunktion linear zum nächsten Knotenpunkt hin abfällt. Stellt man sich eine lokale Steifigkeitsmatrix für ein einzelnes Element auf, so ergibt sich mit $h := 1$ nach Berechnung der $a(\Psi_i, \Psi_j)$ die Matrix $\begin{pmatrix} -1 & -1 & -1 \\ -1 & 4 & -1 \\ -1 & -1 & -1 \end{pmatrix}$.

In der Praxis wird etwas anders verfahren als in diesem Fall. Dieses „knotenorientierte“ Verfahren wie es in dem obigen Beispiel verwendet wird, benötigt zu viel Rechenzeit, da zunächst die für den Punkt wesentlichen Dreiecke herausgesucht werden müssen.

Deshalb geht man im Normalfall „elementorientiert“ vor. Das heißt, man berechnet für jedes Element $K_j \in \mathcal{T}$ den Beitrag zur Steifigkeitsmatrix (also bei m Knoten eine $m \times m$ -Untermatrix der Steifigkeitsmatrix), indem man das Dreieck K_j auf ein Referenzdreieck K_{ref} transformiert:

$$\begin{aligned} Q_j : K_{ref} &\rightarrow K_j \\ \zeta &\mapsto x = Q_j(\zeta) \end{aligned}$$

Abbildung 1.2: Einzelne Basisfunktion Ψ_i

Der Beitrag den das Element K_j liefert kann dabei angegeben werden mit

$$\frac{\mu(K_j)}{\mu(K_{ref})} \int_{K_{ref}} \sum_{k,l} a_{kl}(Q_j^{-1})_{k,k'}(Q_j^{-1})_{l',l} \partial_{k'} N_i \partial_{l'} N_j \, d\zeta$$

μ gibt dabei den jeweiligen Flächeninhalt an. Funktionen aus der bisherigen Basis fallen dabei mit den sogenannten Formfunktionen N_i , also den genormten Basisfunktionen des Referenzdreiecks zusammen.

Damit kann aber eine neue Formulierung für den endlich-dimensionalen Raum der Ansatzfunktionen angegeben werden:

$$S^l(\Omega, \mathcal{T}) = \{u \in H^l(\Omega) : u|_{K_j} = s_j(Q_j^{-1}(x)), s_j \in S(K_{ref})\}$$

Um weitere Freiheitsgrade zu erhalten, und damit auch Elemente zu erzeugen, die \mathcal{C}^1 - bzw. \mathcal{C}^n -Bedingungen erfüllen können, werden zusätzliche Auswertungen (Normalenableitungen, n -te Ableitungen, zusätzliche Punkte) entweder an den Ecken oder an anderen Punkten des Dreiecks vorgenommen. Die folgende Tabelle (1.4), soll einen kleinen Überblick geben.

Um nun eine Systemmatrix aufstellen zu können, sind $M \cdot s$ Matricelementberechnungen erforderlich, wenn wir das Gebiet in M Elemente und jeweils s lokale Freiheitsgrade unterteilen. Den Polynomgrad beliebig in die Höhe zu treiben, wird also nicht genügen, um die Näherungslösung zu verbessern. Statt dessen wird in bestimmten Teilgebieten des Gebietes Ω eine Verfeinerung des Netzes vorgenommen. Gründe für eine lokale Verfeinerung könnten sein:

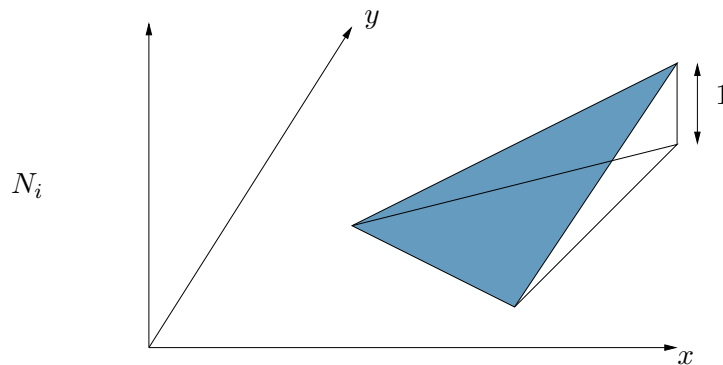


Abbildung 1.3: Formfunktion über einem Dreiecks-Patch

- Treten im Gebiet einspringende Ecken auf, so können schon die ersten Ableitungen (diese bestimmen ja mit, wie gut die analytische Lösung angenähert wird) sehr große Werte annehmen. Durch lokale Verfeinerung kann jedoch auch in Bereichen mit hohem Gradienten der Fehler klein gehalten werden.

Beispiel 1.2.2 (Einspringende Ecke vgl. [1]). Wir betrachten ein Gebiet im \mathbb{R}^2 mit einspringender Ecke. Das Gebiet sei definiert als:

$$\Omega = \{(x, y) \in \mathbb{R}^2; x^2 + y^2 < 1, x < 0 \text{ oder } y > 0\}$$

Identifiziert man den \mathbb{R}^2 mit der Gauss'schen Zahlenebene, so ist $w(z) := z^{\frac{2}{3}}$ analytisch in Ω

$$\begin{aligned} w(z) &= z^{\frac{2}{3}} \\ &= \left(r e^{i\phi} \right)^{\frac{2}{3}} \\ &= r^{\frac{2}{3}} \left(\cos \left(\frac{2}{3} \phi \right) + i \sin \left(\frac{2}{3} \phi \right) \right) \end{aligned}$$

und der Imaginärteil $u(z) := \operatorname{Im} w(z)$ Lösung der Randwertaufgabe

$$\begin{aligned} \Delta u &= 0 \quad \forall \quad x, y \in \Omega \\ u(re^{i\phi}) &= r^{\frac{2}{3}} \sin \left(\frac{2}{3} \phi \right) \quad 0 \leq \phi \leq \frac{3\pi}{2} \\ u &= 0 \quad u|_{\partial\Omega}. \end{aligned}$$

- Vorgabe des Funktionswertes
- ⊙ Vorgabe von Funktionswert und den 1. Ableitungen
- ⊗ Vorgabe von Funktionswert, 1. und 2. Ableitungen
- ⊥ Vorgabe der Normalableitung

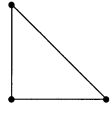
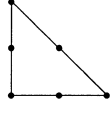
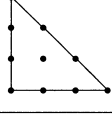
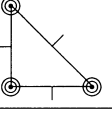
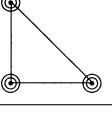
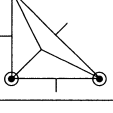
| | |
|---|---|
|  | Lineares Dreieckselement \mathcal{M}_0^1 $u \in C^0(\Omega)$ $\Pi_{\text{ref}} = \mathcal{P}_1, \quad \dim \Pi_{\text{ref}} = 3$ |
|  | Quadratisches Dreieckselement \mathcal{M}_0^2 $u \in C^0(\Omega)$ $\Pi_{\text{ref}} = \mathcal{P}_2, \quad \dim \Pi_{\text{ref}} = 6$ |
|  | Kubisches Dreieckselement \mathcal{M}_0^3 $u \in C^0(\Omega)$ $\Pi_{\text{ref}} = \mathcal{P}_3, \quad \dim \Pi_{\text{ref}} = 10$ |
|  | Argyris Dreieck $u \in C^1(\Omega)$ $\Pi_{\text{ref}} = \mathcal{P}_5, \quad \dim \Pi_{\text{ref}} = 21$ |
|  | Bell Dreieck $u \in C^1(\Omega)$ $\Pi_{\text{ref}} \subset \mathcal{P}_5, \quad \partial_\nu u _{\partial T_i} \in \mathcal{P}_3, \quad \dim \Pi_{\text{ref}} = 18$ |
|  | Hsieh-Clough-Tocher-Element $u \in C^1(\Omega)$ $T = \bigcup_{i=1}^3 K_i, \quad u _{K_i} \in \mathcal{P}_3, \quad \dim \Pi_{\text{ref}} = 12$ |

Abbildung 1.4: Verschiedene Dreieckselemente [1]

Denn es gilt:

$$\begin{aligned}
 x &= r \cos \phi \\
 y &= r \sin \phi \\
 \Delta u(x(r, \phi), y(r, \phi)) &= \Delta U(r, \Phi) \quad \text{mit} \\
 \Delta U &= U_{rr} + \frac{1}{r^2} U_{\phi\phi} + \frac{1}{r} U_r \\
 &= -\frac{2}{9} r^{-\frac{4}{3}} \sin\left(\frac{2}{3}\phi\right) - \frac{4}{9} r^{-\frac{4}{3}} \sin\left(\frac{2}{3}\phi\right) + \frac{2}{3} r^{-\frac{4}{3}} \sin\left(\frac{2}{3}\phi\right) \\
 &= 0
 \end{aligned}$$

Dass die Randbedingungen eingehalten werden, ist klar, da mit dem Vorfaktor $\frac{2}{3}$ lediglich auf den Dreiviertelskreis „skaliert“ wird. Aber schon diese harmonische Lösungsfunktion verhält sich im einspringenden Eckpunkt $\mathbf{0}$ nicht mehr gutmütig, denn wir erhalten durch Ableitung $w'(z) = \frac{2}{3} z^{-\frac{1}{3}}$. Für $z \rightarrow \mathbf{0}$ ist also ∇u nicht mehr beschränkt.

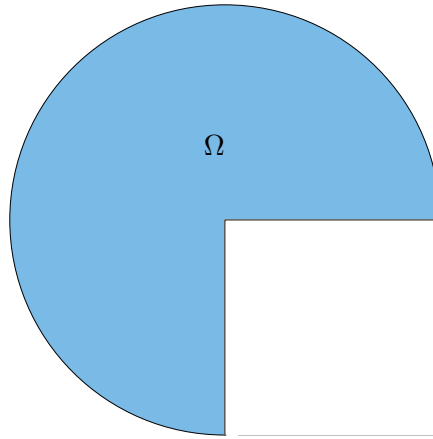


Abbildung 1.5: Einheitskreis mit einspringender Ecke

- An einem bestimmten Punkt des Gebietes möchte man besonders genaue Werte für die Näherungslösung erzielen.

Üblicherweise geschieht der Verfeinerungsvorgang durch automatisierte Netzgeneratoren. Wie hoch der Vernetzungsaufwand ist - und somit auch die Rechenzeit - hängt hierbei wesentlich von der „Glattheit“ des Gebietes ab (genauer gesagt von der Glattheit des Gebietsrandes). Glattheit soll hierbei nicht nur die geometrische Glattheit des Gebietes beinhalten. Änderungen der Randbedingungen können ähnliche Effekte wie einspringende Ecken verursachen und sind deshalb genauso zu beachten.

1.2.2 FEM mit Vierecksgittern

Zerlegt man ein Gebiet Ω in Vierecke, dann wird statt der Polynomfamilie

$$\mathcal{P}_t := \{u(x, y) = \sum_{0 \leq i+k \leq t} c_{ik} x^i y^k\} \quad \text{mit} \quad t = \deg(u)$$

die Polynomfamilie mit Tensorprodukten angesetzt

$$\mathcal{Q}_t := \{u(x, y) = \sum_{0 \leq i, k \leq t} c_{ik} x^i y^k\}$$

deren Dimension mit \mathcal{P}_t^1 (also Polynome in nur einer Variablen) zusammenhängt über

$$\dim \mathcal{Q}_t = (\dim(\mathcal{P}_t^1))^2.$$

Das einfachste Beispiel für ein solches Viereckselement ist ein Rechteck dessen Seiten parallel zu den Koordinatenachsen verlaufen. Wertet man nur an den Eckpunkten aus, so erhält man Ansatzfunktionen aus dem Raum \mathcal{Q}_1 . Wir können dann für die Funktion auf dem Rechteckselement ein 4×4 LGS ansetzen mit

$$u(x, y) = a + bx + cy + dxy \tag{1.10}$$

Enlang der Kante ist das ansonsten quadratische Polynom linear, denn wir können dort jeweils eine Variable konstant setzen. Durch die Gradreduktion am Rand ist es möglich, stetige Übergänge zum Nachbarn zu realisieren, da immer noch genügend Freiheitsgrade für die Bedingung der eigenen Zelle übrigbleiben.

So reibungslos wie es auf den ersten Blick erscheint kann die Lösung auf dem Viereck aber nicht immer berechnet werden. Nehmen wir an, unser kleines Teilstück wäre eine Raute z.B. in Form eines um 45° gedrehten Quadrates. Dann besitzt das angesetzte lineare Gleichungssystem keine Lösung.

Beispiel 1.2.3. Die Werte der Raute seien gewählt wie in Abbildung (1.6). Dann können wir das zugehörige lineare Gleichungssystem aufstellen:

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 2 \\ 1 & 1 & 2 & 2 \\ 1 & 0 & 1 & 0 \end{pmatrix}}_{:=A} \mathbf{c} = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 2 \end{pmatrix}$$

$$\Leftrightarrow \det(A) = 0$$

Damit aber hat das System keine Lösung mehr.

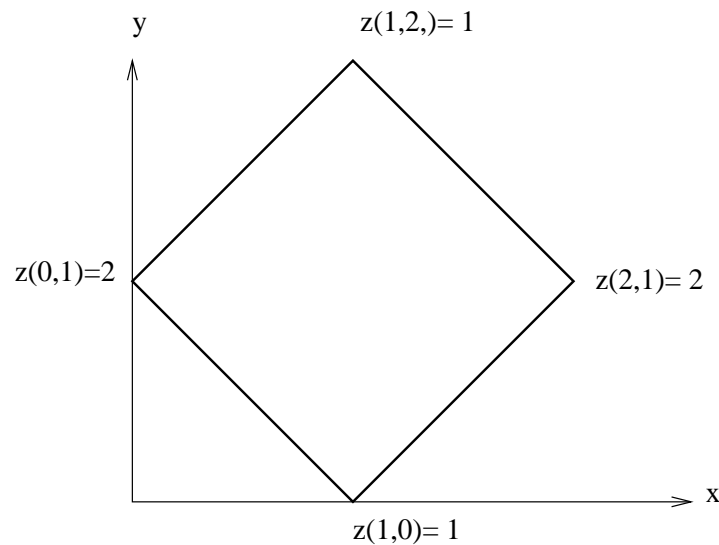


Abbildung 1.6: Raute mit den Eckpunkten $(1, 0)$, $(2, 1)$, $(1, 2)$ und $(0, 1)$ wobei abwechselnd die Werte 1 und 2 angenommen werden.

Da wir uns also nie sicher sein können, ob wir eine Ansatzfunktion finden, wird analog der Vorgehensweise bei Dreieckselementen ein Referenzviereck definiert, und dieses mitsamt den Referenz-Ansatzfunktionen auf das tatsächliche Viereck abgebildet. Handelt es sich bei dem Gitterviereck um ein Parallelogramm, so genügt uns dafür eine affine Abbildung.

Transformation eines Referenzvierecks auf ein Parallelogramm:

Zwei sich schneidende Seiten können jeweils parametrisiert werden als

$$\alpha_1 x + \beta_1 y = \text{const}$$

$$\alpha_2 x + \beta_2 y = \text{const}.$$

Transformiert man das kartesische Koordinatensystem auf das durch die zwei Schenkel aufgespannte Koordinatensystem, so muss gelten:

$$\zeta(x, y) = \alpha_1 x + \beta_1 y$$

$$\eta(x, y) = \alpha_2 x + \beta_2 y.$$

Damit lautet der lineare Ansatz (siehe (1.10)):

$$u(x, y) = a + b\zeta(x, y) + c\eta(x, y) + d\zeta(x, y)\eta(x, y)$$

Und wieder haben wir die Bedingung erfüllt, dass die Einschränkung auf den Rand linear ist; wir haben also den Raum

$$S = \{v \in C^0(\bar{\Omega}); \text{ für jedes Gitterelement } K \text{ gilt: } \eta|_K \in \mathcal{P}^2, \text{ und } v|_{\partial K} \text{ linear}\}$$

Da es im Normalfall nicht genügt, Parallelogramme zuzulassen, sondern allgemeinere Gebilde das Gebiet Ω besser ausfüllen können, greift man zum Instrument der Isoparametrischen Abbildung. Dabei wird das Referenzquadrat auf ein beliebiges Viereck abgebildet. Will man den Gebietsrand genau überdecken, so können dabei im Gebiet auch krummlinig berandete Viereckselemente auftreten. Da hierbei die Transformation aber nicht trivialerweise folgt, soll hier stellvertretend der Fall eines geradlinig berandeten Vierecks betrachtet werden. Ist $T_{\text{ref}} = [0, 1]^2$ das Einheitsquadrat, so

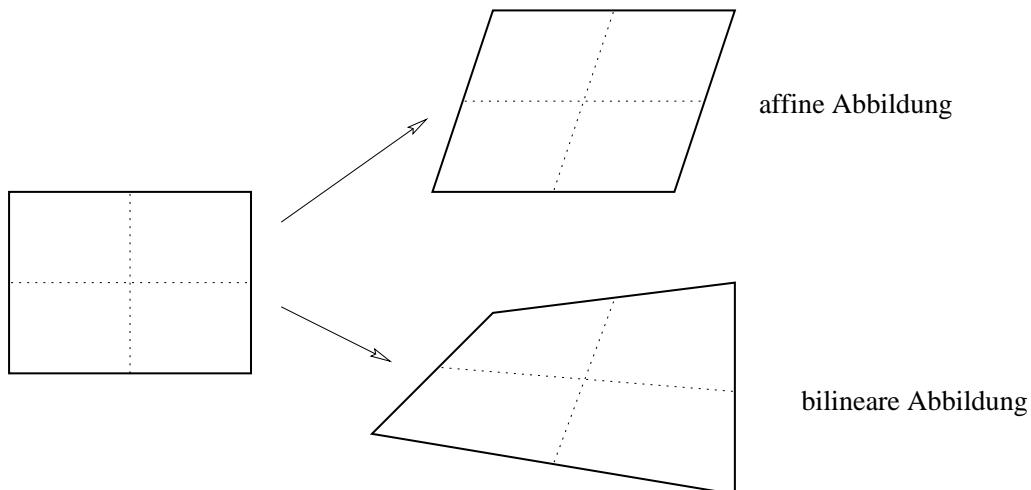


Abbildung 1.7: Isoparametrische Vierecke mit geradlinigen Kanten.

müssen die Koordinaten für jeden Eckpunkt transformiert werden. Wir haben damit

im Zweidimensionalen 8 freie Parameter zur Verfügung. Die Transformation erfolgt wie im affinen Fall.

Wie bei den Dreieckselementen, so besteht auch hier die Möglichkeit, durch Erhöhung des Polynomgrades mehr Freiheitsgrade zu gewinnen. Wenn wir uns wieder auf den einfachen Fall eines Rechteckes zurückziehen (und das dürfen wir ruhig tun, da wir ja jedes beliebige Viereck zurücktransformieren können), dann ist zum Beispiel die Serendipity-Klasse mit Auswertungspunkten in den Ecken und den Seitenmitten (manchmal wird auch der Rechtecksmittelpunkt noch hinzugenommen) eine der wichtigsten Vertreterinnen. Nur um zu zeigen, dass die Graderhöhung immer sehr große Terme erzeugt, möchte ich hier noch den allgemeinen Ansatz für die Funktion u auf einem Viereckselement K aufschreiben:

$$\begin{aligned} u(x, y) = & a + bx + cy + dxy \\ & + e(x^2 - 1)(y - 1) + f(x^2 - 1)(y + 1) \\ & + g(x - 1)(y^2 - 1) + h(x + 1)(y^2 - 1) \end{aligned}$$

In der praktischen Anwendung hat sich aber ganz klar das Konzept der Triangulierung durchgesetzt, hauptsächlich weil es noch keine hinreichend allgemeinen Algorithmen zur Konstruktion von Hexaedernetzen in drei Dimensionen gibt.

Kapitel 2

Die web-Methode

2.1 Splines - Eine kurze Einführung

Polynome haben lokal sehr gute Approximationseigenschaften. Bei einer großen Anzahl von Interpolationspunkten aber sind sie nicht geeignet, das Approximationsproblem hinreichend gut zu lösen. Störungen pflanzen sich über das gesamte Definitionsgebiet fort, und der hohe Polynomgrad führt an den Gebiets- und Intervallrändern bzw zwischen den Interpolationspunkten zu starken Oszillationen, wie dies schon anhand des Beispiels der Lagrange-Polynome (2.1) deutlich wird. Um die guten lo-

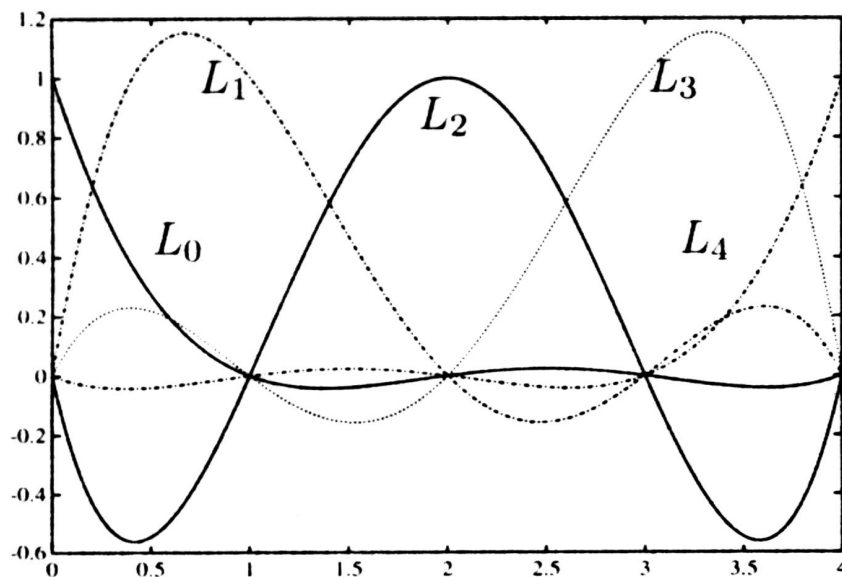


Abbildung 2.1: Lagrange-Polynome L_i für $n = 4$ und äquidistante Knoten t_i

kalen Approximationseigenschaften für Polynome niedrigen Grades dennoch nutzen zu können, teilt man das Definitionsgebiet Ω durch eine Knotenfolge u_j in Intervalle

auf, und fügt die darauf lebenden Approximationspolynome glatt an den Intervallenenden zusammen, wobei man den Grad m der Polynome vorgibt (meist $m = 2, 3$ oder 4).

Definition 2.1.1 (Spline). *Ein Spline einer reellen Veränderlichen ist eine Funktion, die stückweise auf Intervallen definiert wird, und deren Teile an den Nahtstellen nach vorgegebenen Glattheitseigenschaften zusammengesetzt werden.*

Diese Definition beinhaltet unter anderem auch Funktionen auf den einzelnen Intervallen, die keine Polynome sind. Tatsächlich aber wird in der Praxis fast ausschließlich mit Polynomen gearbeitet - wir werden dies nachfolgend genauso handhaben. Die Bezeichnung *Spline* geht im Übrigen auf Schönberg (1946) zurück. Erste Anwendung fanden diese Funktionen in der Auswertung ballistischer Tabellen.

2.1.1 Suche nach einer geeigneten Basis - die abgebrochenen Potenzen

Bevor wir eine Basis des Splineraumes $S_{n,U}(\Omega)$ (hierbei ist n der Grad des Polynoms, \mathcal{U} die Knotenfolge der u_j und Ω das Intervall bzw. Gebiet) angeben können, müssen die Glattheitsbedingungen präzisiert werden. Fragen wir uns zunächst, welche Glattheit wir an den Knoten erwarten können:

Sind zwei Polynomsegmente vom Grad n n -mal stetig differenzierbar verbunden, so stellen Sie dasselbe Polynom dar (jeder Vorfaktor eines Monomes stimmt dann mit dem entsprechenden Vorfaktor des nachfolgenden Polynomsegmentes überein). Man vereinbart deshalb: An einem einfachen Knoten ist ein Spline mit Grad n mindestens $(n - 1)$ -mal stetig differenzierbar (also ist auf alle Fälle $P_n \subset S_n$). An einem k -fachen Knoten soll ein Spline $(n - k)$ -mal stetig differenzierbar sein. Auf diesen Fall werde ich aber nicht weiter eingehen, da er für die weiteren Betrachtungen nicht von Bedeutung sein wird.

Eine Basis für den durch den Grad n und die Knotenfolge \mathcal{U} festgelegten Splineraum läßt sich konstruieren, indem man sukzessive die Freiheitsgrade auf den einzelnen Intervallen betrachtet.

Vereinbarung: Der Definitionsbereich für $S_{n,U}$ beginnt mit dem $n + 1$ -ten Knoten und hat am Ende $n + 1$ Knoten am Rand bzw. außerhalb von \mathcal{U} . Die Begründung dafür wird später noch in Theorem (2.1.1) nachgeliefert werden.

Betrachten wir nun Splines vom Grad n . Dann ist für jeden Knoten per definitionem die zugehörige abgebrochene Potenz

$$(t - u_j)_+^n := \begin{cases} (t - u_j)^n & \text{für } t \geq u_j, \\ 0 & \text{sonst} \end{cases}$$

ein Element des Splineraumes. Die abgebrochenen Potenzen erfüllen zudem die geforderten Glattheitseigenschaften an den Knotenpunkten. Bleibt also noch zu überprüfen, ob sie auch als Basis des Splineraumes geeignet sind.

Satz 2.1.1. *Die Monome und die abgebrochenen Potenzen bilden eine Basis*

$$\mathcal{B} := \{1, t, \dots, t^n, (t - u_1)_+^n, \dots, (t - u_{l-1})_+^n\} \quad (2.1)$$

des Splineraumes $S_{n,\mathcal{U}}$ (Spline vom Grad $\leq n$). Insbesondere gilt für die Dimension von $S_{n,\mathcal{U}}(\Omega)$, dass

$$\dim S_{n,\mathcal{U}} = n + l.$$

Da die ersten $n + 1$ abgebrochenen Potenzen eine Basis des Raumes \mathcal{C}^n bilden, ist die Gültigkeit der obigen Aussage äquivalent zur Basiseigenschaft der $n + l$ abgebrochenen Potenzen an den Knotenpunkten u_{-n}, \dots, u_{l-1} .

Beweis. Für den Beweis werde ich mich an die zur Basis der abgebrochenen Potenzen äquivalente gemischte Basis aus Monomen und abgebrochenen Potenzen halten. Des weiteren soll die Knotenfolge keine Vielfachheiten besitzen. Der Spline s lebt also in \mathcal{C}^{n-1} .

Zunächst zeigen wir, dass zur Konstruktion der Basis des Splineraums $S_{n,\mathcal{U}}(\Omega)$ maximal $n + l$ Freiheitsgrade zur Verfügung stehen. Gehen wir sukzessive vor, und beginnen mit dem ersten Intervall $[u_0, u_1]$, so können wir jedes beliebige Polynom mit $\deg \leq n$ wählen; damit haben wir $n + 1$ freie Parameter (Grad 0 mit eingeschlossen). Aufgrund der Glattheitsforderung $s \in \mathcal{C}^{n-1}$ sind die nachfolgenden Polynome auf den Intervallen $[u_1, u_2], \dots, [u_{l-1}, u_l]$ bis auf einen frei wählbaren Faktor durch die jeweiligen Vorgänger festgelegt. Das heißt, dass maximal noch $l - 1$ Freiheitsgrade hinzukommen können. Damit ist $\dim S_{n,\mathcal{U}}(\Omega) \leq n + l$. Damit \mathcal{B} eine Basis ist, muss noch gezeigt werden, dass die Funktionen in \mathcal{B} linear unabhängig sind. Wie in solchen Fällen üblich sei

$$s(t) := \sum_{i=0}^n a_i t^i + \sum_{i=1}^{l-1} c_i (t - t_i)_+^n = 0 \quad \forall \quad t \in [u_0, u_l] \quad (2.2)$$

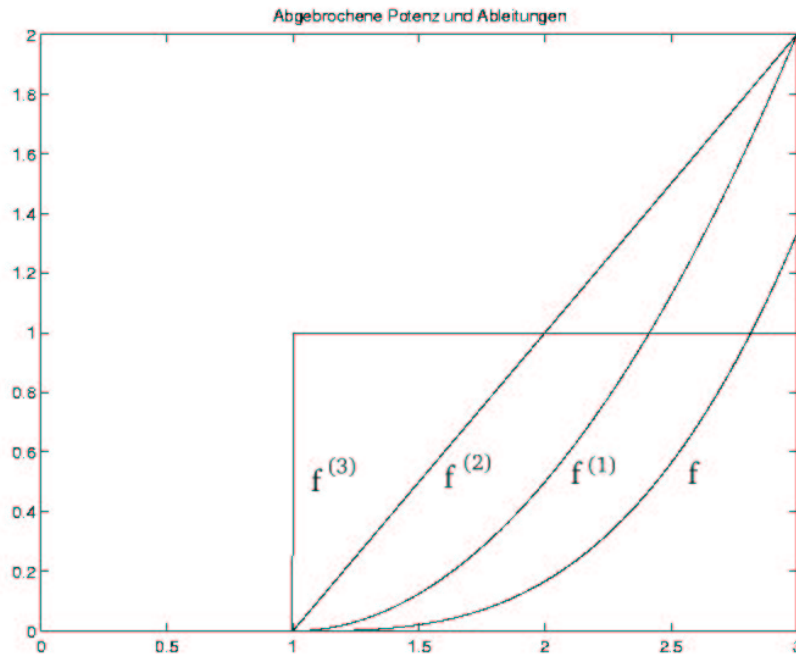
Sei $f(t^+)$ und $f(t^-)$ der rechts- bzw. linksseitige Grenzwert. Wenden wir die linearen Funktionale

$$G_i(f) := \frac{1}{n!} \left(f^{(n)}(t_i^+) - f^{(n)}(t_i^-) \right)$$

auf die Gleichung (2.2) an, so folgt für alle $i = 1, \dots, l - 1$

$$\begin{aligned} 0 &= G_i(s) \\ &= \underbrace{G_i \left(\sum_{j=0}^n a_j t^j \right)}_{=0} + \sum_{j=1}^{l-1} c_j \underbrace{G_i(t - t_j)_+^n}_{=\delta_{ij}} \\ &= c_i. \end{aligned}$$

Das Kroneckersymbol ist leicht erklärt, wenn man sich einmal ein Beispiel dazu ansieht (Abb. 2.2). Der Abbildung können wir entnehmen, dass die n -te Ableitung aus einer Treppenfunktion mit einer Stufe im Punkt u_j der durch die n Ableitungen entstandenen Stufenhöhe von $n!$ besteht. Das heißt, dass der rechtsseitige Grenzwert am Punkt u_j genau $n!$ ist, der linksseitige 0. Damit hat das Funktional G_i mit $i = j$ genau



Abbildungung 2.2: Abgebrochene Potenz $f(t) = (t - 1)_+^3$ und die Ableitungen bis zur Ordnung 3.

an diesem Punkt (und nur dort, denn links und rechts von u_j ist $f^{(n)}$ konstant) den Wert 1.

Damit müssen alle $c_i = 0$ sein, woraus folgt, dass $s(t) = \sum_{i=0}^n a_i t^i = 0$ für alle $t \in [u_0, u_l]$. Aufgrund der linearen Unabhängigkeit der Monombasis folgt damit, dass auch $a_0 = \dots = a_n$. \square

Nun ist auch klar, wie eine Splineraumbasis zu konstruieren ist:

Sei

$$u_{-n} \leq \dots \leq u_0 < u_1 \leq \dots \leq u_{l-1} < u_l \leq \dots u_{n+l}, \quad \mathcal{U} = [u_0, u_l], \quad (2.3)$$

wobei $u_j < u_{j+1}$ (wir wollen uns hier ja nur mit Knoten der Vielfachheit 1 auseinandersetzen).

Sei $p \in S_{n,\mathcal{U}}(\Omega)$ mit $p|_{[u_0, u_1]} = p_0$. Hat der Knotenpunkt u_1 nur die Vielfachheit 1, so darf erst in der n -ten Ableitung am Übergang zum nächsten Intervall eine Unstetigkeit auftreten. Das heißt, dass wir nun ein Vielfaches der abgebrochenen Potenz $(t - u_1)_+^n$ hinzufügen können.

So einfach diese Basis zu konstruieren ist, sie hat leider einen gravierenden Nachteil: Ebenso wie die Basis der Polynome besitzt sie einen globalen Charakter. Dies widerspricht aber der Grundidee der Splines, stückweise lokale Polynome zu verwenden.

Ein weiterer Nachteil liegt in der fehlenden Relation zwischen Geometrie und Koeffizienten. Daher wird nun eine neue Splineraumbasis eingeführt, die B-Spline-Basis.

2.1.2 B-Splines

Die B-Splines werden ausgehend von den charakteristischen Funktionen auf $[u_j, u_{j+1})$, die wir mit $b_{j,0}$ bezeichnen, rekursiv definiert.

$$b_{j,0}(x) := \begin{cases} 1 & \text{für } u_j \leq x < u_{j+1} \\ 0 & \text{sonst.} \end{cases}$$

Einen B-Spline vom Grad m erhält man dann durch die Rekursion

$$\begin{aligned} b_{j,m} &= w_{j,m} b_{j,m-1} + (1 - w_{j+1,m}) b_{j+1,m-1} \\ w_{j,m}(x) &:= \begin{cases} \frac{x-u_j}{u_{j+m}-u_j} & \text{für } u_j < u_{j+m} \\ 0 & \text{sonst.} \end{cases} \end{aligned} \quad (2.4)$$

Der Definition können wir entnehmen, dass genau genommen nur halboffene Intervalle verwendet werden. Dies geschieht deshalb, weil sonst die Definition des Splines im Knotenpunkt nicht eindeutig wäre. Tatsächlich aber ist es vollkommen gleich, ob wir ein abgeschlossenes oder halboffenes Intervall verwenden, denn an diesem Punkt nehmen beide Splines für $m > 0$ aufgrund der Stetigkeitsanforderung an den Spline ohnehin denselben Wert an.

Anschaulich liefert uns die Rekursion (2.4) nach dem ersten Schritt eine stückweise lineare Hutfunktion, deren Träger sich über das Intervall $[u_j, u_{j+2})$ erstreckt. Im nächsten Schritt erhalten wir dann einen quadratischen Spline usw. Bild (2.3) zeigt unter anderem das Dreiecksschema zur Berechnung der jeweiligen B-Splines. Eine Anmerkung noch zur Rekursion: Im allgemeinen Fall mehrfacher Knoten, den wir hier nicht weiter behandeln werden, kommt es vor, dass Basisfunktionen niedriger Ordnung nicht vorkommen. Unsere Knotenfolge soll aber immer einfach sein, weshalb auch für alle $u_j \in \mathcal{U}$ gilt: $\mu([u_j, u_{j+1}]) > 0$.

Die Definition der Gewichte und der $b_{0,j}$ führt dazu, dass alle $b_{j,n}$ positiv sein müssen, wobei die $b_{j,n}$ auf den einzelnen Intervallen ihres Trägers ein Polynom vom Grad n darstellt. Weiter lässt sich mittels Induktion zeigen:

Korollar 2.1.1. *Die B-Spline Basisfunktionen bilden eine Partition der Einheit.*

$$\sum_j b_{j,m}(x) = 1 \quad x \in \mathcal{U} \quad (2.5)$$

Beweis. Der Induktionsanfang ist mit $m = 1$ klar. Laut Induktionsvoraussetzung gilt: $\sum_k b_{k,m-1} = 1$.

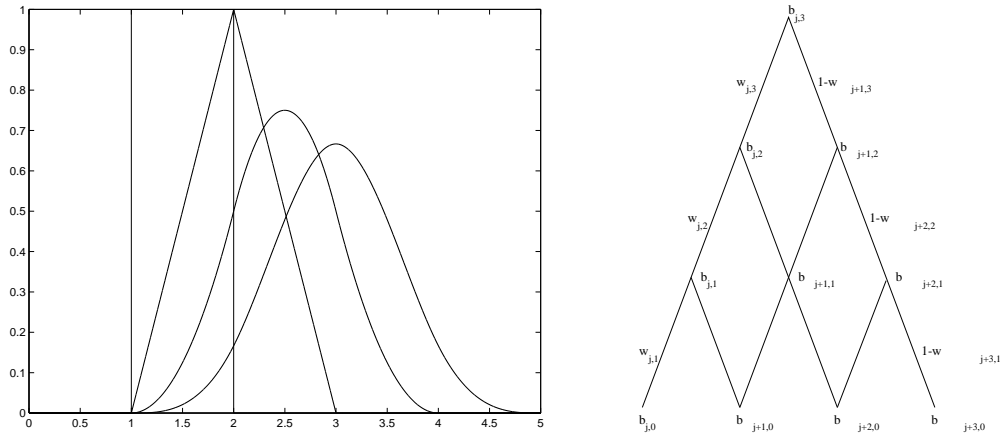


Abbildung 2.3: Rekursive Definition der B-Splines. Es sind eine charakteristische Funktion ($b_{1,0}$), eine Hutfunktion ($b_{1,1}$), eine quadratische ($b_{1,2}$) und eine kubische ($b_{1,3}$) B-Spline Basisfunktion zu sehen.

Induktionsschritt:

$$\sum_k b_{k,m}(x) = \sum_k b_{0,m}(x-k)$$

$$\text{Rekursionsformel} \Rightarrow = \sum_k \frac{x-k}{m} b_{0,m-1}(x-k) + \frac{n-x+k+1}{m} b_{1,m-1}(x-k)$$

$$= \sum_k \frac{x-k}{m} b_{0,m-1}(x-k) + \sum_k \frac{m-x+k+1}{m} b_{0,m-1}(x-(k+1))$$

$$\text{Umsummation } \tilde{k} = k+1 \quad = \sum_k \frac{x-k}{m} b_{0,m-1}(x-k) + \sum_{\tilde{k}} \frac{m-x+\tilde{k}}{m} b_{0,m-1}(x-\tilde{k})$$

$$\text{Zusammenfassen} \quad -\infty < k, \tilde{k} < \infty \quad = \sum_{\hat{k}} b_{0,m-1}(x-\hat{k})$$

$$= \sum_{\hat{k}} b_{\hat{k},m-1}(x) = 1$$

□

Was wir also bislang haben, ist eine Rekursion und die Eigenschaft der B-Spline Basisfunktionen, eine Partition der Eins zu bilden, wobei die Basiseigenschaft bis zu diesem Zeitpunkt noch nicht klar ist.

Theorem 2.1.1 (B-Spline-Basis). Die B-Splines

$$b_{j,m}, j = -m : l-1$$

bilden eine Basis für den Splineraum $S_{n,\mathcal{U}}(\Omega)$, der durch die Knotenfolge (2.3) festgelegt ist.

Beweis: Zum Beweis diese Theorems können wir praktischerweise die zuvor gezeigten Basis- und Glattheitseigenschaften der abgebrochenen Potenzen verwenden; indem wir zeigen, dass die $m + l$ B-Splines in der Lage sind, die Splineraumbasis der Dimension $m + l$ aus (2.1) darzustellen, folgt sofort die Behauptung.

Da die Verschiebung eines Summationsindex immer gewisse Fehlerquellen birgt, werde ich nachfolgend darauf verzichten, obere und untere Grenzen konkret anzugeben. Vielmehr wird der Index von $-\infty$ bis ∞ laufen. Sollte der Bedarf bestehen, die Summation auf den tatsächlich notwendigen Bereich einzuschränken, so stellt dies auch kein Problem dar, denn der Träger einer B-Spline Basisfunktion ist ja bekannt.

Die Anwendung der Rekursionsformel (2.4) auf eine Linearkombination von B-Splines liefert das Ergebnis

$$\sum_j c_j b_{j,m} = \sum_j c_{j,1} b_{j,m-1} \quad \text{mit} \quad (2.6)$$

$$c_{j,1}(x) = c_j w_{j,m}(x) + c_{j-1}(1 - w_{j,m}(x)).$$

Nun müssen wir die Koeffizienten c_j näher bestimmen. Wir setzen

$$c_j = \Psi_{j,m}(t) = (u_{j+1} - t) \cdots (u_{j+m} - t).$$

Für ein beliebiges $t \in$ gilt dann

$$\begin{aligned} & \Psi_{j,m}(t) w_{j,m}(x) + \Psi_{j-1,m}(t) (1 - w_{j,m}(x)) \\ &= (u_{j+1} - t) \cdots (u_{j+m} - t) \left(\frac{x - u_j}{u_{j+m} - u_j} \right) + (u_j - t) \cdots (u_{j+m-1} - t) \left(1 - \frac{x - u_j}{u_{j+m} - u_j} \right) \\ &= \underbrace{(u_{j+1} - t) \cdots (u_{j+m-1} - t)}_{=\Psi_{j,m-1}(t)} \left(\frac{(x - u_j)(u_{j+m} - t)}{u_{j+m} - u_j} + \left(1 - \frac{x - u_j}{u_{j+m} - u_j} \right) (u_j - t) \right) \\ &= \Psi_{j,m-1}(t) (x - t). \end{aligned}$$

Es gilt also

$$\sum_j \Psi_{j,m}(t) b_{j,m}(x) = (x - t) \sum_j \Psi_{j,m-1}(t) b_{j,m-1}(x)$$

Unter Verwendung von $\Psi_{j,0}(t) = 1$ erhalten wir nach m -facher Anwendung der obigen Umformung den als Marsden-Identität bekannten Zusammenhang

$$\begin{aligned} \sum_j \Psi_{j,m}(t) b_{j,m}(x) &= (x - t)^2 \sum_j \Psi_{j,m-2}(t) b_{j,m-2}(x) \\ &= \dots \\ &= (x - t)^{m-1} \sum_j \Psi_{j,1}(t) b_{j,1}(x) \\ &= (x - t)^m \underbrace{\sum_j 1 \cdot b_{j,0}}_{=1} \end{aligned} \quad (2.7)$$

Durch Differentiation von (2.7) nach t und Auswertung am Punkt $t = 0$ folgt, dass alle x -Potenzen mit Grad $\leq m$ durch Linearkombinationen der B-Splines darstellbar sind.

$$\begin{aligned} \frac{\partial}{\partial t} \sum_j b_{j,m}(\alpha_{j,1}t^m + \dots + \alpha_{j,m}) &= \frac{\partial}{\partial t}(x-t)^m \\ &= -m(x-t)^{m-1} = \sum_j b_{j,m}(\alpha_{j,1}(m)t^{m-1} + \dots + \alpha_{j,m-1}) \\ t=0 \quad \Rightarrow \quad x^{m-1} &= \frac{-1}{m} \sum_j b_{j,m}(x)\alpha_{j,m-1} \end{aligned}$$

Analog folgt die behauptete Eigenschaft für fortgeführte Differentiation des Ausdrucks. Liegt x innerhalb des Definitionsbereiches \mathcal{U} , so müssen zur Berechnung lediglich die B-Splines aufaddiert werden, deren Träger x beinhaltet.

Was uns noch zur Basis (2.1) fehlt, ist die Darstellung der abgebrochenen Potenzen. Für einen Knoten u_k gilt

$$\Psi_{j,m}(u_k) = 0, \quad \text{für } j = k - m : k - 1.$$

Da der Schnitt des Trägers der Basisfunktion $b_{j,m}$ für $j < k - m$ mit dem Intervall $[u_k, \infty)$ aber leer ist, folgt aus (2.7) schon die B-Splinedarstellung der abgebrochenen Potenz

$$(x - u_k)_+^m = \sum_{j \geq k} \Psi_{j,m}(u_k) b_{j,m}(x).$$

□

Bei äquidistanten Knotenfolgen, wie sie in der web-Methode auftreten, können wir durch Verschiebung eines B-Splines jeden anderen erzeugen.

$$\begin{aligned} b_{j,m}(x) &= b_{0,m}(x - jh) \\ &= b_m\left(\frac{x}{h} - j\right) \end{aligned}$$

b_m heißt der Kardinal-B-Spline. Er besitzt nach Voraussetzung die Knoten $0, 1, \dots, m+1$. Die Rekursion (2.4) hat für Kardinal-B-Splines die besonders einfache Form

$$mb_m(x) = xb_{m-1}(x) + (m+1-x)b_{m-1}(x-1)$$

Kommen wir aber zum eigentlich wichtigen Punkt - der Auswertung einer Splinekurve. Identität (2.6) liefert uns hierzu schon den entscheidenden Hinweis. Wiederholtes Anwenden der Gleichung ergibt

$$\begin{aligned} p(x) &= \sum c_j b_{j,m}(x) \\ &= \sum c_{j,m} b_{j,0}(x) \end{aligned}$$

$b_{j,0}$ ist aber gerade die charakteristische Funktion auf dem x enthaltenden Intervall $[j, j+1)$, weshalb wir uns nur noch um die Auswertung der Polynome $c_{j,m}$ kümmern müssen. Nach unserer Rekursionsformel gilt:

$$\begin{aligned} c_{j,0} &= c_j \\ c_{j,k+1} &= c_{j,k} w_{j,m-k}(x) + c_{j-1,k} (1 - w_{j,m-k}(x)) \quad k = 0 : m-1 \end{aligned}$$

Damit können aufgrund der B-Spline-Rekursion auch die Koeffizienten $c_{j,m}(x)$ in einem Dreiecksschema berechnet werden.

Uniforme multivariate B-Splines

Die uniformen B-Splines sind ein Spezialfall der bisher behandelten B-Splines. Sie zeichnen sich dadurch aus, dass sie über einer gleichmäßigen Knotenfolge definiert sind. Wie wir schon gesehen haben, können sie durch Skalierung kardinaler B-Splines konstruiert werden.

Diesen entsprechen im m die multivariaten uniformen B-Splines auf gleichmäßigen „Gittern“ der Rasterweite h .

Definition 2.1.2. Ein (normalisierter) uniformer m -variater B-Spline vom Grad n der Gitterweite h und dem Shift $k = (k_1, \dots, k_m) \in \mathbb{R}^m$ ist definiert durch

$$b_{k,h}^n(x) := h^{-\frac{m}{2}} \prod_{\nu=1}^m b^{n_\nu} \left(\frac{x_\nu}{h} - k_\nu \right), \quad x = (x_1, \dots, x_m) \in \mathbb{R}^m \quad (2.8)$$

Der Faktor $h^{-\frac{m}{2}}$ hat hierbei die Aufgabe, die L_2 -Norm des Splines von der Gitterweite unabhängig zu machen (nehme hier $\|\cdot\|^2$):

$$\begin{aligned} \int_{\mathbb{R}^m} \left(h^{-\frac{m}{2}} b \left(\frac{x}{h} - k \right) \right)^2 dx &= h^{-m} \int \left(b \left(\frac{x_1}{h} - k_1 \right) \right)^2 dx_1 \dots \\ &\quad \dots \int \left(b \left(\frac{x_m}{h} - k_m \right) \right)^2 dx_m \\ \text{Subst. } \Rightarrow &= h^{-m} \int (b(y_1))^2 h \cdot dy_1 \dots \int (b(y_m))^2 h \cdot dy_m \\ &= h^{-m} h^m \int_{\mathbb{R}^m} (b(y))^2 dy \\ &\Rightarrow \int_{\mathbb{R}^m} |b_k|^2 \asymp 1. \end{aligned}$$

Da der B-Spline nun unabhängig von der Gitterweite und dem Verschiebungsvektor ist, erhalten wir wieder das Integral über einen kardinalen B-Spline. Dieser hat aber einen kompakten Träger und ist beschränkt, weshalb auch die L_2 -Norm beschränkt ist.

Analog dem eindimensionalen Fall wird hier anstelle des kardinalen B-Splines auf dem Einheitsintervall der B-Spline b_n skaliert und verschoben in die jeweiligen Raumrichtungen. Es gilt:

$$\text{supp}(b_{k,h}^n) = kh + (nQ_\star)h, \quad Q_\star := (0, 1)^m$$

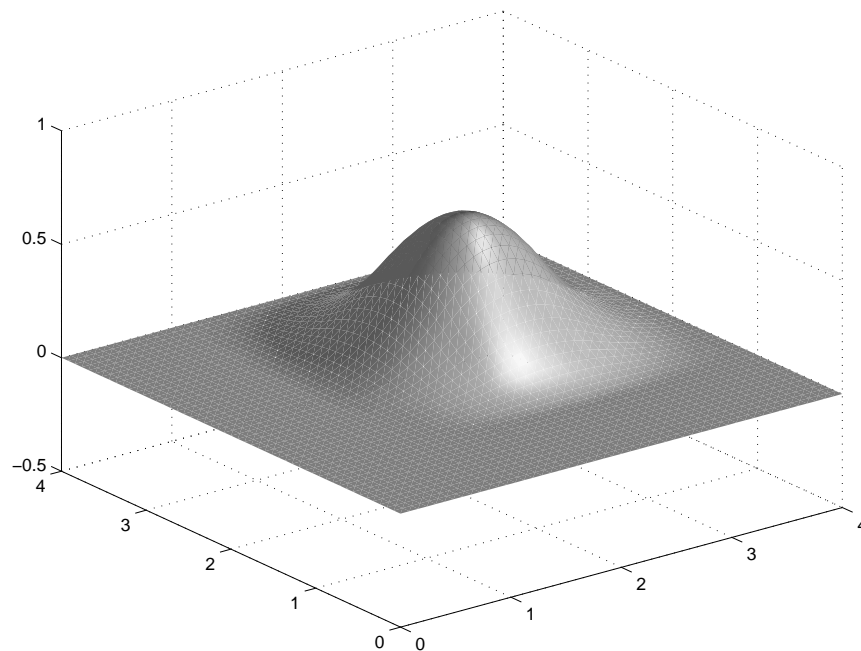


Abbildung 2.4: Kardinaler B-Spline.

2.2 Die Methode

Eine ausführlichere Behandlung der web-Methode kann [11] entnommen werden.

2.2.1 Einführung

Vorweg noch ein Wort zur Notation: um den Text lesbarer zu machen wird folgende Schreibweise eingeführt:

$$\begin{aligned} f &\preceq g \quad \text{für} \\ f &\leq cg \end{aligned}$$

mit einer Konstanten c . Gitterweiten h beeinflussen diese Konstante aber nicht! Analog ist die umgekehrte Richtung und das Zeichen \asymp definiert.

Wie schon in den vorangegangenen Kapiteln nehmen wir für die weitere Besprechung an, dass wir die Poisson-Gleichung mit homogenen Dirichlet-Randbedingungen betrachten.

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega \subset \mathbb{R}^m \\ u &= 0 \quad \text{für } u|_{\partial\Omega} \end{aligned}$$

Die schwache Formulierung führt auf das zu minimierende Funktional

$$\frac{1}{2} \int_{\Omega} \nabla v \nabla v - \int_{\Omega} f v \quad v \in V = H_0^1(\Omega) \quad (2.9)$$

Unter der Annahme, dass der Rand und die Daten f glatt sind, existiert eine Lösung u und wir können diese durch eine diskrete Lösung

$$u_h = \sum_i a_i B_i$$

approximieren. Als Basisfunktionen werden wir Tensorprodukt-B-Splines mit einem Träger von 3×3 Quadraten der Seitenlänge h verwenden. Eingesetzt in Gleichung (2.9) folgt:

$$\begin{aligned} \int u_h \nabla B_k &= \sum_i \int a_i \nabla B_i \nabla B_k \quad \text{wegen } \nabla \text{ und } \int \text{ linear} \\ &= \int f \nabla B_k \quad \text{nach Voraussetzung} \end{aligned}$$

Dieses Galerkin-System können wir kurz schreiben als:

$$G_h A = F \quad \text{mit } A = \{a_i\}.$$

Um die Randbedingung zu erfüllen, fordern wir, dass die einzelnen B_i auf dem Gebietsrand $\partial\Omega$ verschwinden.

Der durch die Diskretisierung entstandene Approximationsfehler (orthogonal zum Raum der Basisfunktionen des diskreten Unterraumes) lässt sich wegen des Lemmas von Céa (1.1.4) durch

$$\|u - u_h\|_{H^1} \preceq \inf_{v_h \in V_h} \|u - v_h\|_{H^1} \quad (2.10)$$

abschätzen. Diese lässt sich gegen die Sobolev-Norm

$$\|v\|_{l,\Omega} = \left(\sum_{|\alpha| \leq l} \int_{\Omega} |D^\alpha v|^2 \right)^{\frac{1}{2}}, \quad \|v\|_{l,\Omega} = \|v\|_{H^l}$$

abschätzen. Aus (2.10) folgt sofort, dass bei Verwendung stückweiser Polynome mit $\text{Grad} \leq n$ der Fehler durch

$$\|u - u_h\|_{H^1} \preceq h^n \quad (2.11)$$

abgeschätzt werden kann.

Ein weiterer wichtiger Punkt ist die Konditionszahl des Galerkin-Systems. Bei standard-FEM mit quasiuniformen Gebietszerlegungen ist die Kondition bezüglich der 2-Norm bei einer Gitterweite h durch

$$\text{cond}_2 G_h \sim h^{-2}$$

beschränkt. Durch diese Eigenschaft wird garantiert, dass die für gewöhnlich verwendeten iterativen Löser stabil arbeiten. Dass dies auch im Fall der web-Splines zutrifft werden die folgenden Kapitel zeigen.

Da bei komplizierten Gebieten die Gittergenerierung einen großen Teil der Rechenleistung verbraucht, wurden sogenannte gitterfreie Methoden entwickelt, welche gewichtete Finite Elemente verwenden, sodass die Lösung z.B. durch

$$u \approx wp \quad p \in$$

wobei w eine positive Gewichtsfunktion mit $w|_{\partial\Omega} = 0$ (garantiert das Einhalten der Randbedingungen) und \mathcal{P} ein linearer Raum ist. Um Missverständnissen vorzubeugen: Natürlich verwenden wir hier ein Gitter; jedoch können wir ein beliebiges quadratisches Gitter über das Gebiet legen, das heißt also, dass der Prozess der Gittergenerierung im Sinne eines an das Gebiet angepassten Gitters wegfällt.

Als Basis von \mathcal{P} bietet sich - wie oben schon erwähnt - der Raum der Splines an.

2.2.2 Stabilität der B-Spline-Basis

Mit b soll - der in (2.8) eingeführte normalisierte uniforme multivariate B-Spline bezeichnet werden. Es sei hier nochmals erwähnt, dass dieser aufgrund des Normalisierungsfaktors bezüglich der L_2 -Norm unabhängig von der Gitterweite h ist.

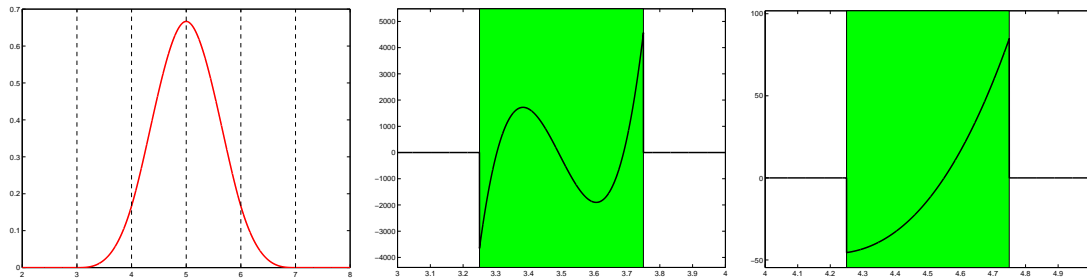


Abbildung 2.5: B-Spline und die beiden dualen Funktionale λ_0 und λ_1 im kubischen, eindimensionalen Fall

Eine elegante Methode zu zeigen, dass die aus den B-Splines gebildete Basis auf dem Definitionsgebiet \mathcal{D} stabil ist, verwendet die dualen Funktionale. Stabilität bedeutet, dass der Spline $s = \sum_{k \in \mathcal{D}} c_k b$ bezüglich der 2- bzw. L_2 -Norm durch den Koordinatenvektor C nach oben und unten beschränkt ist, also:

$$\text{const } \|C\|_{2,\mathcal{D}} \leq \left\| \sum_{k \in \mathcal{D}} b c_k \right\| \leq \|C\|_{2,\mathcal{D}}$$

wobei const lediglich vom Grad n der B-Splines abhängt.

Nach ([17] S. 142 ff) gibt es für jedes $l \in \{0, \dots, n\}^m$ eine Funktion λ^l - das duale Funktional - mit Träger $[\frac{1}{4}, \frac{3}{4}]^m + l$ so, dass gilt:

$$\int b(\cdot - k) \lambda^l = \delta_{k,0}$$

Auch diese Funktionale können wir so konstruieren, dass sie unabhängig von Gitterweite und Shift sind indem wir den Normalisierungsfaktor einführen:

$$\lambda_i^l(x) = h^{-\frac{m}{2}} \lambda^l\left(\frac{x}{h} - i\right)$$

mit Träger:

$$Q'_{i+l} = h\left(\left[\frac{1}{4}, \frac{3}{4}\right]^m + i + l\right).$$

Da l immer von der jeweilig zu betrachtenden Gitterzelle ausgeht, schreiben wir zur Verdeutlichung $l(i)$. Dieser Index beschreibt also die Gitterzelle $Q_{i+l(i)}$. Wir fordern ausserdem, dass der Träger dieser Gitterzelle vollständig im Gebiet Ω liegen muss - ansonsten wird das l verworfen.

Definition 2.2.1. Die Menge der für das Gebiet Ω relevanten B-Spline Indizes bezeichnen wir mit

$$K = \{k \in {}^m : \text{supp } b_k \cap \Omega \neq \emptyset\}.$$

Sei $L_k = \{l \in {}^m : Q_{k+l} \subset \text{supp } b_k \cap \Omega\}$; dann nennen wir b_k einen inneren B-Spline, falls L_k nichtleer ist, ansonsten äusseren B-Spline. K setzt sich damit aus den disjunkten Mengen I (Index für die inneren B-Splines) und J zusammen. Ist $i \in I$, so soll der Index $l(i)$ eines zugehörigen dualen Funktional in L_i liegen. Wenn klar ist, welches duale Funktional gemeint ist, wird folgende Schreibweise verwendet:

$$\lambda_i = \lambda_i^{l(i)}.$$

Diese Indizierung stellt sicher, dass für jeden inneren B-Spline ein normalisiertes duales Funktional existiert, dessen Träger vollständig im Gebiet liegt, und dessen Abstand zum Rand proportional zu h ist. In Bild 2.6 ist ein B-Spline mit einem dazugehörigen dualen Funktional zu sehen.

Duale Funktionale für äussere B-Splines anzugeben ist zwar möglich aber nicht sinnvoll, da ihre Norm mit kleiner werdendem Träger in Ω stark anwachsen würde. Daher verzichtet man darauf, auch die äusseren B-Splines mit dualen Funktionalen zu versehen. Die zugehörigen inneren dualen Funktionale sind aber orthogonal zu allen äusseren B-Splines, da sich ihre Träger nicht schneiden.

Theorem 2.2.1. Für alle $k \in K$ und $i \in I$ sind die B-Splines b_k und die dualen Funktionale λ_i einheitlich beschränkt in Bezug auf die Gitterweite und bi-orthogonal.

$$\|b_k\| \leq 1, \quad \|\lambda_i\| \leq 1$$

$$\int_{\Omega} b_k \lambda_i = \delta_{k,i}$$

Ein häufig verfolgtes Ziel in FE-Methoden ist, den Grad zu erhöhen, um die Approximationseigenschaften der Näherungslösung zu verbessern. Die Identität von Marsden besagt, dass Polynome vom Grad m durch Linearkombination von B-Splines $b_{j,i}$

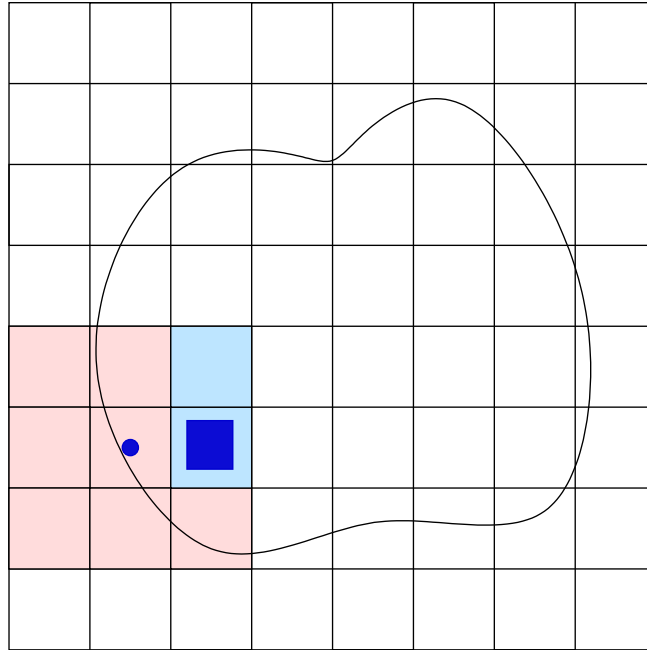


Abbildung 2.6: Schnitt eines B-Splines mit dem Gebiet Ω , wobei der Mittelpunkt des B-Spline-Trägers mit einem Punkt markiert ist. Die zugehörigen Indizes sind $L_k = \{[2, 1], [2, 2]\}$. Im Bild ist der Träger des dualen Funktional zum lokalen Index $l(i) = [2, 1]$ eingetragen.

der Ordnung $m + 1$ dargestellt werden können. Nach (2.7) gilt:

$$\sum_j \Psi_{j,m}(t) b_{j,m}(x) = (x - t)^m \underbrace{\sum_j 1 \cdot b_{j,0}}_{=1 \text{ auf } [u_j, u_{j+1}]}$$

Um alle Monome zu erhalten muss die jeweilige Linearkombination berechnet werden. Umsortieren nach den Knoten führt auf die Polynome $q(k)$:

Theorem 2.2.2. *Der Spline*

$$p = \sum_{k \in K} q(k) b_k \tag{2.12}$$

ist ein Polynom vom Grad $\leq n$ auf Ω , wenn, und nur dann, wenn q ein Polynom vom Grad $\leq n$ auf K ist.

Konsequenz dieses Theorems ist, dass wir nicht auf die äußeren B-Splines verzichten können, wenn wir den Polynomgrad aufrecht erhalten wollen.

Denn um auf einer Zelle den Polynomgrad m zu erhalten müssen alle auf den umliegenden Zellen lebenden B-Splines - also alle, deren Träger einen nichtleeren Schnitt mit der gerade betrachteten Zelle haben - existieren. Dieser Zusammenhang liefert uns also eine Vorschrift für die Verknüpfung äußerer und innerer B-Splines.

2.2.3 Weighted Extended B-Splines

Eine gewöhnliche B-Spline-Basis ist ungeeignet, homogene Randdaten beliebiger Gebietsränder zu erfüllen. Dieses Verhalten kann aber leicht erzwungen werden, indem die Basis mit einer auf dem Rand verschwindenden, zur geglätteten Abstandsfunktion äquivalenten Gewichtsfunktion multipliziert wird.

$$w(X) \sim \text{dist}(X, \partial\Omega)$$

Der Raum der gewichteten B-Splines erfüllt natürlich immer noch die Approximationsbedingung (2.11). Die Kondition der Galerkin-Matrix G_h kann aber wegen der äußeren gewichteten B-Splines, deren Träger einen nur kleinen Schnitt mit dem Gebiet besitzt, stark anwachsen. Da aufgrund der Marsden-Identität aber zur Erhaltung des Polynomgrades und damit der Approximationsgüte alle B-Splines, deren Träger einen nichtleeren Schnitt mit dem Gebiet besitzt, benötigt werden, können die äußeren B-Splines nicht einfach weggelassen werden. Vielmehr werden sie an innere, nahe gelegene Splines angehängt:

Definition 2.2.2. Sei $K = I \cup J$ die Indexmenge der relevanten B-Splines mit den Teilmengen I innerer bzw. J äußerer B-Spline-Indices. Dann ist $\forall i \in I$ der erweiterte gewichtete B-Spline (kurz: **web-Spline**) definiert durch

$$B_i = \frac{w}{w(x_i)} \left(b_i + \sum_{j \in J} e_{i,j} b_j \right)$$

$$|e_{i,j}| \leq 1, \quad e_{i,j} = 0 \quad \text{für} \quad \|i - j\| \geq 1$$

x_i sei das Zentrum einer im Innern liegenden Gitterzelle $Q_{i+l(i)}$. Die Koeffizienten $e_{i,j}$ seien dabei so gewählt, dass alle gewichteten Polynome mit Grad $\leq n$ im Spliner Raum enthalten sind.

Der Faktor $\frac{w}{w(x_i)}$ dient lediglich der Stabilisierung der Methode und bewirkt, dass die am Gebietsrand befindlichen **web-Spline** Basisfunktionen nicht mit zu kleinen Koeffizienten versehen werden. Die Tatsache, dass nur endlich viele Koeffizienten $e_{i,j}$ ungleich 0 sind - durch die Forderung $\|i - j\| \geq 1$ beschränken wir uns auf die „benachbarten“ Indices - ist garantiert, dass sich der Träger der **web-Splines** nach wie vor proportional zur Gitterweite h verhält. Liegt ein **web-Spline** weit genug im Innern des Gebiets, so werden gar keine äußeren B-Splines an ihn angehängt - es handelt sich also um einen gewöhnlichen gewichteten B-Spline. Durch die Beschränktheit der Koeffizienten wird außerdem verhindert, dass der **web-Spline** bei kleiner werdender Gitterweite unkontrolliert anwächst.

Mit Hilfe der definierten Basis sollte es nun möglich sein, die Indexmenge in Gleichung (2.12) auf die Indexmenge I reduzieren zu können. Der Grad des Polynoms $q(k)$ ist $\leq n$. Daher kann der Wert jedes Koeffizienten $q(j)$ an einem $j \in J$ durch Interpolation von $(n+1)^m$ inneren Koeffizienten $q(i)$ mit Indizes $i \in I$ mittels Lagrangepolynomen ermittelt werden. Als geeignete innere Koeffizienten $I(j)$ könnten

zum Beispiel die zu j bezüglich der Maximumnorm auf \mathbb{Z}^m nächstgelegenen $(n+1)^m$ ganzzahligen inneren Gitterindizes dienen. Setze:

$$l_{i,j}(k) = \delta_{i,k}, \quad i, k \in I(j)$$

Für festes j wird dann für $e_{i,j}$ der Wert des Lagrange-Polynoms, ausgewertet an der Stelle j , gesetzt.

$$e_{i,j} = l_{j,i}(j), \quad i \in I(j)$$

Um über den gesamten Index I summieren zu können, werden alle $e_{i,j}$ mit $i \notin I(j)$ auf den Wert 0 gesetzt. Damit gilt

$$q(j) = \sum_{i \in I} e_{i,j} q(i).$$

Eingesetzt in Gleichung (2.12):

$$\begin{aligned} p(x) &= \sum_{i \in I} q(i) b_i(x) + \sum_{j \in J} q(j) b_j(x) \\ &= \sum_{i \in I} q(i) \left[b_i(x) + \sum_{j \in J} e_{i,j} b_j(x) \right], \quad x \in \Omega \end{aligned}$$

Multiplikation mit der glatten Gewichtsfunktion führt uns wieder auf die zuvor angegebene Definition der web-Splines.

Sowohl der Index $I(j)$ als auch der Koeffizient $e_{i,j}$ sind beschränkt, falls man sich an die vorgeschlagene Wahl der inneren Koeffizienten hält; denn bei ausreichend feiner Gitterunterteilung und mit der Voraussetzung, dass der Gebietsrand glatt ist, kann der Rand lokal nahezu als Hyperfläche (hier: Gerade) angesehen werden. Damit aber ist der Hausdorff-Abstand von j zu $i(j)$ asymptotisch (für $h \rightarrow 0$) beschränkt durch $2(n+1)$. Mit dieser Notation und Wahl der Indizes kann der Koeffizient nach Skalierung als von der Gitterweite h unabhängiges Produkt univariater Lagrange-Polynome explizit angegeben werden:

Theorem 2.2.3. Für alle $j \in J$ sei

$$I(j) = \{i \in \mathbb{Z}^m : \alpha_\mu \leq i_\mu \leq \alpha_\mu + n\}$$

eine Menge nächstgelegener innerer Indices. Dann sind die Koeffizienten

$$e_{i,j} = \begin{cases} \prod_{\mu=1}^m \prod_{\substack{l=\alpha_\mu \\ l \neq i_\mu}}^{\alpha_\mu+n} \frac{j_\mu-l}{i_\mu-l} & \text{für } i \in I(j) \\ 0 & \text{sonst} \end{cases}$$

geeignet zur Konstruktion von web-Splines gemäß Definition 2.2.2. Die Beschränktheit der Koeffizienten folgt aus der Tatsache, dass die Lagrange-Koeffizienten den Wert 1 haben, und die Differenz $i - j$ beschränkt ist (Polynome weisen keine Polstelle auf).

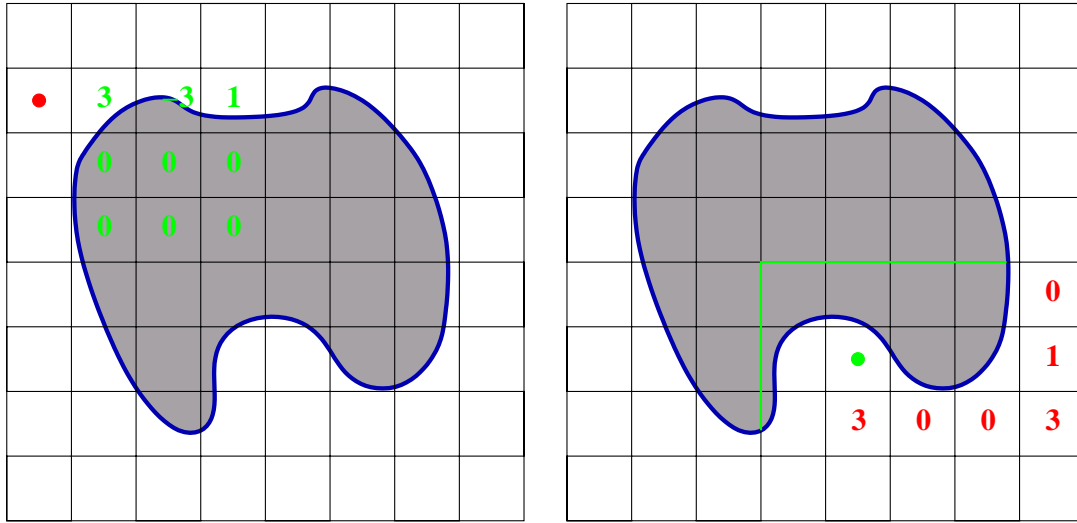


Abbildung 2.7: Links: die Koeffizienten $e_{i,j}$ eines äußeren B-Splines. Auf der rechten Seite ist der Träger eines web-Splines abgebildet; die Zahlen geben die Koeffizienten der zugehörigen äußeren B-Splines an.

Ein Beispiel für die Konstruktion der web-Splines gemäß Definition 2.2.2 und dem vorangegangenen Theorem ist in Abbildung 2.7 dargestellt.

Da der Punkt x_i , an welchem die Gewichtsfunktion aus Normierungsgründen ausgewertet werden muss innerhalb des Gebietes Ω liegen muss, fällt der hier mit einem weissen Kreis markierte Punkt nicht mit dem Mittelpunkt des inneren B-Spline-Trägers zusammen.

2.2.4 Stabilität und Approximationsordnung

An dieser Stelle möchte ich lediglich die wesentlichen Eigenschaften bezüglich Stabilität und Approximationsordnung der web-Spline-Basis aufführen. Dabei werde ich auf die Beweise (nachzulesen in [11]) verzichten, da diese sich ohnehin stark an die Beweise zur Stabilität der B-Spline-Basis anlehnen.

Stabilität

Die dualen Funktionale für die web-Basis Λ_k , $k \in I$, von b_i mit Trägern in Ω seien gegeben durch

$$\Lambda_k = \frac{w(x_k)}{w} \lambda_k \quad k \in I.$$

Dann besitzt die web-Basis die folgenden Eigenschaften:

Theorem 2.2.4. Es seien $i, k \in I$; dann sind die dualen Funktionale Λ_k und die **web-Splines** B_i bezüglich der Gitterweite h gleichmäßig beschränkt in L_2 - lassen sich also gegen eine Konstante abschätzen - und biorthogonal,

$$\|B_i\|_0 \preceq 1, \quad \|\Lambda_k\|_0 \preceq 1, \quad \int_{\Omega} B_i \Lambda_k = \delta_{i,k}.$$

Theorem 2.2.5. Die **web-Basis** ist bezüglich der L_2 -Norm stabil,

$$\left\| \sum_{i \in I} a_i B_i \right\|_0 \asymp \|A\|_2.$$

$\|\cdot\|_0$ sei dabei die L_2 -Norm, $\|A\|_2$ die Norm des Koeffizientenvektors.

Theorem 2.2.6. Das Spektrum der Galerkinmatrix G_h ist beschränkt durch

$$1 \preceq \varrho(G_h) \preceq h^{-2}.$$

Eine direkte Folgerung ist die Beschränktheit der Galerkinmatrix

$$\text{cond} G_h \preceq h^{-2}.$$

die für jedes FE-Verfahren benötigt wird, damit die iterativen Löser in einer angemessenen Zeit konvergieren.

Approximationsordnung

Sei u glatte Lösung des Modellproblems und $u_h \in$ eine durch das Galerkin-System bestimmte FE- Approximation. Dann gilt

$$\|u - u_h\|_r \preceq h^{n+1-r}.$$

2.2.5 Die Rolle der Gewichtsfunktion für die **web-Methode**

Eine nicht zu unterschätzende Rolle für die **web-Methode** spielt die Wahl der Gewichtsfunktion. Zwar tritt bei entsprechender Wahl keine qualitative Änderung ein, was die Konvergenzordnung bezüglich der h -Potenz betrifft, das heißt, der L_2 -Fehler verschwindet nach wie vor mit h^{n+1} , jedoch unterliegen die Vorfaktoren des Fehlers deutlichen Schwankungen. Für welche der Gewichtsfunktionen man sich entscheidet hängt im Wesentlichen von der Komplexität des Gebietsrandes ab.

Algebraische Funktionen

Die Verwendung algebraischer Gewichtsfunktionen für Ritz-Galerkin-Verfahren (die FEM ist ein Spezialfall) geht auf Kantorowitsch und Krylow (1956) zurück (siehe [14] S255ff).

Ist der Gebietsrand durch eine einfache algebraische Darstellungen gegeben, so kann eine Gewichtsfunktion sofort in geschlossener Form angegeben werden:

- Wenn sich der Rand als $F(x, y) = 0$ darstellen lässt, so kann $w(x, y) = \pm F(x, y)$ gesetzt werden, z.B. für den Kreis: $w(x, y) = R^2 - x^2 - y^2$.
- Für ein konvexes Polygon (u. U. auch krummlinig berandet) mit der Darstellung $a_i x + b_i y + c_i = 0$ (für krummlinig z.B. noch quadratische Terme) kann die Gewichtsfunktion angegeben werden durch

$$w(x, y) = \pm(a_1 x + b_1 y + c_1) \dots (a_m x + b_m y + c_m)$$

Im Fall eines Rechtecks mit $-a \leq x \leq a$, $-b \leq y \leq b$ hat w die Darstellung:

$$w(x, y) = (x^2 - a^2)(y^2 - b^2)$$

Ein Fall eines krummlinig berandeten „Polygons“ ist der einer Sichel, die durch Kreise mit den Radien a und $\frac{a}{2}$ gebildet wird:

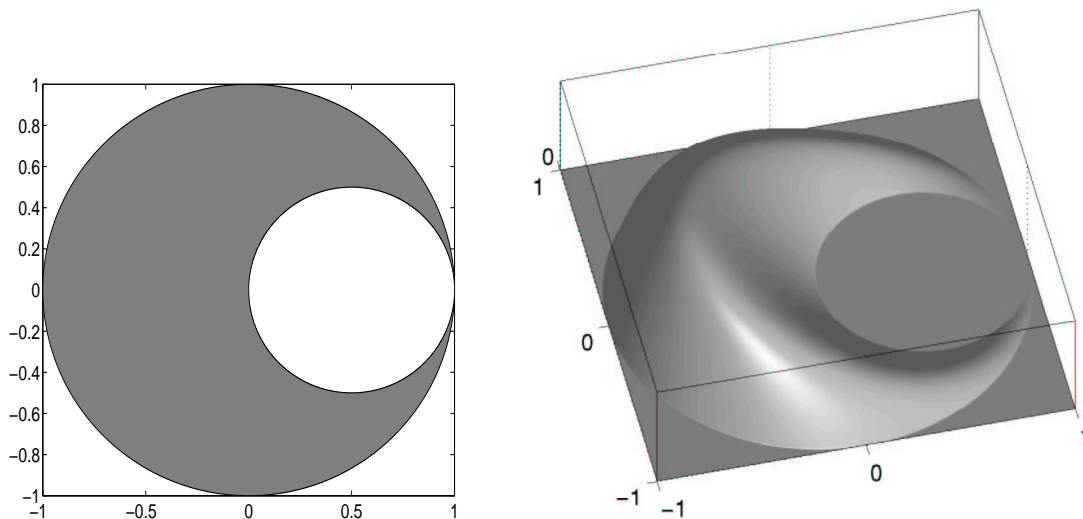


Abbildung 2.8: Beispiel einer algebraischen Gewichtsfunktion auf einem sichelförmigen Gebiet mit $w(x, y) = (a^2 - x^2 - y^2)(x^2 - ax + y^2)$

2.2.6 R-Funktionen nach Rvachev

(Siehe auch [16]).

Setzt sich das Gebiet aus Schnitten bzw Vereinigungen einfacher algebraischer Funktionen zusammen, können Rvachev-Funktionen (R-Funktionen) als Gewichtsfunktionen eingesetzt werden.

R-Funktionen sind reellwertige Funktionen, deren Vorzeichen ausschließlich vom Vorzeichen der jeweiligen Argumente abhängt. Z.B. kann die Funktion xyz nur dann negative Werte annehmen, wenn eine ungerade Anzahl der Koeffizienten ein negatives Vorzeichen hat. Solche R-Funktionen können das Verhalten boolescher Operatoren

nachbilden. Als Beispiel stelle man sich die Funktion $\min(x, y)$ vor, welche das logische UND (\wedge) nachbildet. R-Funktionen, die mit derselben logischen Funktion verwandt sind werden dabei zu Äquivalenzklassen zusammengefasst. So wie Boolesche Funktionen sind auch diese unter Komposition abgeschlossen.

Beispiel 2.2.1 (Familien von R-Funktionen). Es sei $\alpha(x, y)$ ein beliebige Funktion, so dass $-1 \leq \alpha \leq 1$. Dann gilt:

$$x \wedge_{\alpha} y \equiv \frac{1}{\alpha + 1} \left(x + y - \sqrt{x^2 + y^2 - 2\alpha xy} \right)$$

$$x \vee_{\alpha} y \equiv \frac{1}{\alpha + 1} \left(x + y + \sqrt{x^2 + y^2 - 2\alpha xy} \right)$$

Für $\alpha = 1$ nehmen die Funktionen jeweils Maximum und Minimum an, für $\alpha = 0$ erhalten wir einfachere Funktionen \wedge_0 und \vee_0 , die in einem weiteren Beispiel zur Anwendung kommen werden.

R-Funktionen können dazu verwendet werden, ein durch einfache geometrische Gebiete berandetes Objekt in einem einzigen Ausdruck w darzustellen (dies ist der Fall bei einem durch ein System von Ungleichungen berandeten Gebiet). Dabei kann sich w bei entsprechender Parametrisierung so wie der glättete Abstand vom Gebietsrand verhalten.

Die Theorie der R-Funktionen bietet einen direkten Zusammenhang zwischen geometrischer Modellierung und mengentheoretischen bzw logischen Operationen. Denn, wie gesagt, für jede logische oder mengentheoretische Operation steht uns eine reellwertige Funktion mit gewünschten Eigenschaften, wie z.B. besondere Glattheit zur Verfügung. Da die mengentheoretischen Symbole einfach durch Ersetzen der entsprechenden logischen Operatoren (und damit auch durch Einsetzen passender R-Funktionen) umgerechnet werden können, lassen sich diese Operationen mit nur geringem Rechenaufwand am Computer umsetzen.

Beispiel 2.2.2. Das Gebiet aus Abb. 2.9 kann zunächst als mengentheoretischer Ausdruck angegeben werden:

$$\Omega = w_1 \cap (w_2 \bar{\cap} w_3)$$

$$w_1 = \frac{1}{2r}(r^2 - x^2 - y^2) \geq 0$$

$$w_2 = x - r + b \geq 0$$

$$w_3 = \frac{a^2 - y^2}{2a} \geq 0$$

Dieser mengentheoretische Ausdruck kann durch Ersetzen der entsprechenden Operatoren durch ihre logischen Gegenstücke übersetzt werden in

$$w = w_1 \wedge_0 (-(w_2 \wedge_0 w_3))$$

Das heißt, die w_i werden durch Einsetzen in \wedge_0 zu einer Funktion w (hier eine zum geglätteten Abstand äquivalenten Gewichtsfunktion) verknüpft.

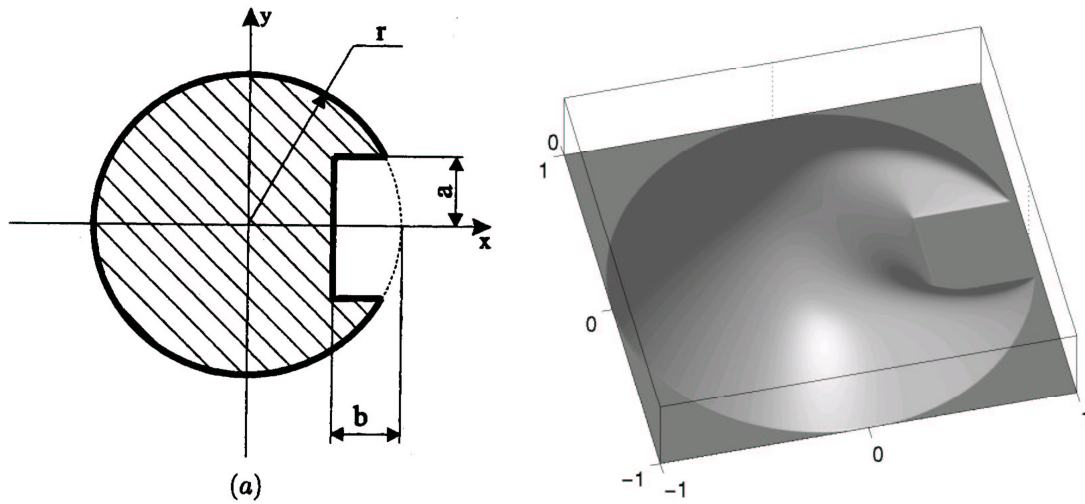


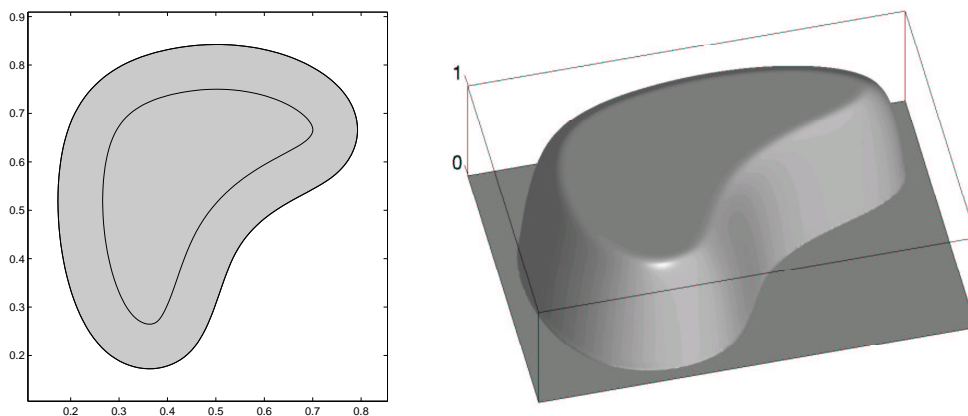
Abbildung 2.9: Gebiet und zugehörige R-Funktion

2.2.7 Geglättete Abstandsfunktion

Ist das Gebiet allgemein durch eine NURBS- oder einen Bézier-Kurve gegeben, so können weder algebraische noch R-Funktionen verwendet werden. Stattdessen wird die geglättete, im Innern in ein 1-Plateau übergehende Abstandsfunktion eingesetzt. Die Breite des Übergangs wird durch δ , die Glattheit des Überganges in das Plateau durch γ gesteuert:

$$w(x) = 1 - (\max(\delta - \text{dist}(x, \partial D), 0)/\delta)^\gamma.$$

Der Vorteil dieser Gewichtsfunktion liegt nicht nur in der Allgemeinheit der zulässi-



gen Gebiete; vielmehr können im Innern des Gebietes aufgrund des 1-Plateaus tabellarische Werte für $\int \nabla B_i \nabla B_k$ verwendet werden, was den Rechenaufwand stark

minimiert. Genau am Übergang in das Plateau handelt man sich aber leider auch einen relativ großen Fehler ein.

Ein neuer Ansatz für eine noch besser auf die Aufgabenstellung angepasste Gewichtsfunktion ist zur Verbesserung der Fehlerkonstanten notwendig, und um diesen soll es im folgenden Kapitel gehen.

Kapitel 3

Die Gewichtsfunktion $w(X)$

Im Folgenden werden einige Voraussetzungen und Bezeichnungen eingeführt, die, falls nicht ausdrücklich umdefiniert, Gültigkeit für den gesamten Theorieteil haben sollen.

3.1 Annahmen und geometrisches Modell

In diesem Kapitel werden einige geometrische Annahmen angestellt, die - bis auf die Glattheit des Gebietsrandes $\partial\Omega$ - keine wesentlichen Einschränkungen darstellen, sondern lediglich zur Übersichtlichkeit der folgenden Beweise dienen sollen.

Die Gewichtsfunktion sei gegeben durch

$$w(X)^{-1} = \int_{\partial\Omega_P} \frac{ds_P}{\|X - P(s)\|^2} \quad \text{mit } X \in \mathbb{R}^2$$

. Des weiteren ist der Rand $\partial\Omega$ durch die Kurve P parametrisiert:

$$P : \quad \longrightarrow \quad \mathbb{R}^2$$
$$s \longmapsto \begin{pmatrix} P_1(s) \\ P_2(s) \end{pmatrix}, \quad \text{wobei } s \in [0, s_{\text{Periodendauer}}]$$

Betrachte o.B.d.A. immer den Grenzwert $(x_1, x_2) = X \rightarrow (0, 0) = P(0) \in \partial\Omega$. Diese Annahme ist deshalb legitim, weil Translation und Rotation des Koordinatensystems sich weder auf das über eine geschlossene Kurve laufende Integral noch auf den Abstand eines Gebietpunktes zum Rand auswirken. Aufgrund der Glattheitseigenschaften des Randes kann dieser in einer hinreichend kleinen Umgebung U durch eine Funktion dargestellt werden:

$$\mathcal{C}^2 \ni p : \quad \rightarrow \quad \mathbb{R}^2$$
$$t \mapsto (t, p(t)) \quad \text{wobei } t \in [-\varepsilon, \varepsilon] \quad (3.1)$$

In dieser Umgebung U soll $|p'(t)| \leq \varepsilon \ll 1$ für $|t| \leq \varepsilon$ gelten. Damit ist $p(0) = 0$; $p'(0) = 0$ darf o.B.d.A. angenommen werden, denn eine Drehung des Gebietes ändert,

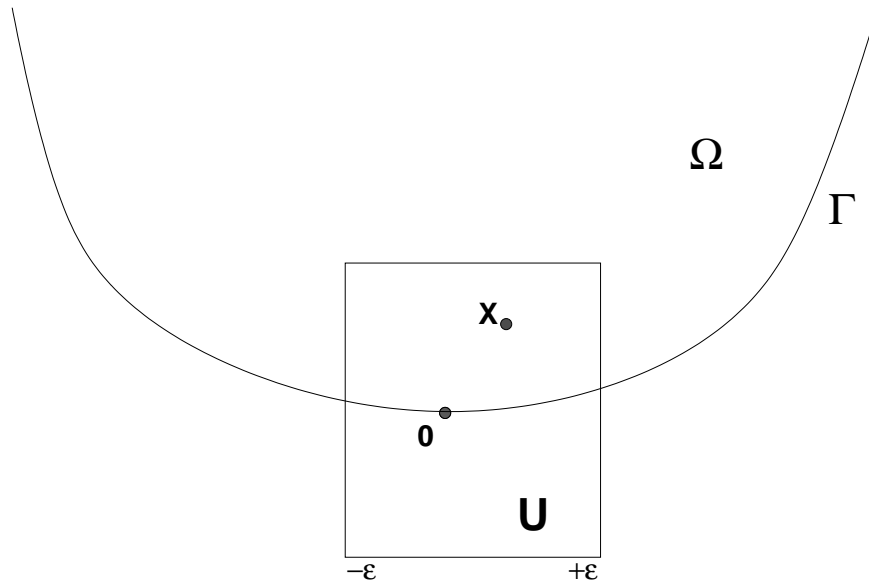


Abbildung 3.1: Gebietsrand mit Umgebung U um den Nullpunkt (nach einer Transformation des Koordinatensystems).

wie schon gesagt, nichts am Wert des Integrals (siehe auch Abb. 3.1).

3.2 Einige Hilfssätze

Lemma 3.2.1. *Sei $X \in U \cap \Omega$, dann gilt*

$$\text{dist}(X) \asymp x_2 - p(x_1) =: h(X), \quad \text{und} \quad (3.2)$$

$$\lim_{X \rightarrow 0} \frac{\text{dist}(X)}{h(X)} = 1, \quad (3.3)$$

wenn wir mit $\text{dist}(X)$ die glatte Abstandsfunktion von X zum Rand $\partial\Omega$, und mit p die in (3.1) erklärte Funktionsdarstellung in einem Gebiet U bezeichnen.

Beweis. Zeige zunächst (3.2).

O.B.d.A. sei $x_1 > 0$ (wir begründen dies wieder mit der freien Drehbarkeit des Gebietes Ω). Nach Voraussetzung gilt $|p'| < \epsilon$. Es sei $X = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, $P = \begin{pmatrix} x_1 \\ p(x_1) \end{pmatrix}$, $\tilde{P} = \begin{pmatrix} \tilde{x}_1 \\ p(\tilde{x}_1) \end{pmatrix}$, mit $\text{dist}(X) = \|X - \tilde{P}\| =: l_2$ und damit $X - \tilde{P} \perp \partial\Omega$ im Punkt \tilde{P} .

α ist der von $X - \tilde{P}$ und $X - P$ eingeschlossene Winkel. Es sei also entsprechend der

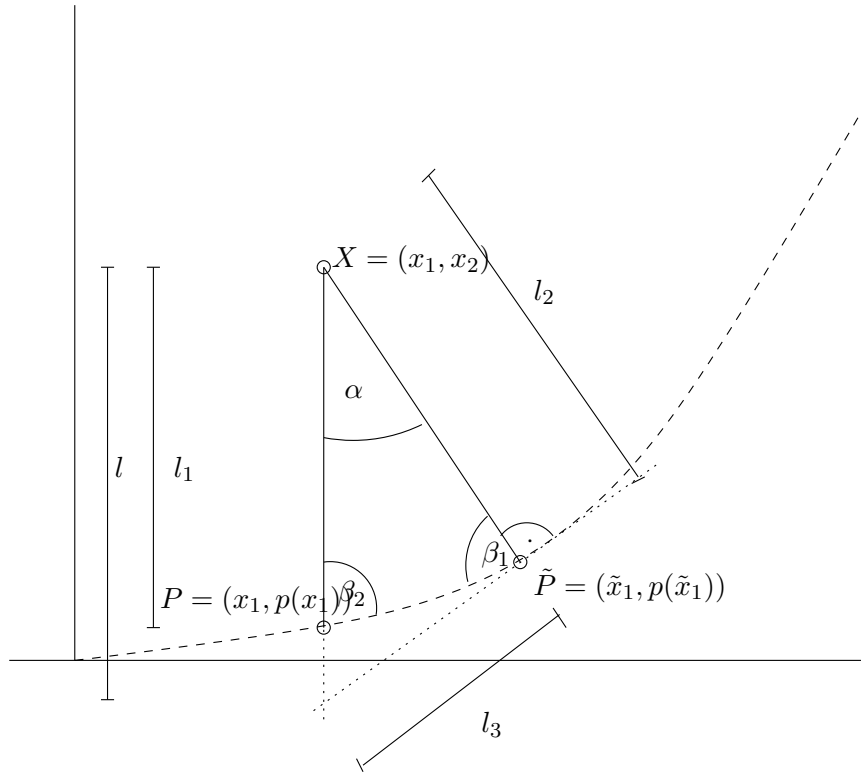


Abbildung 3.2: Skizze zu Lemma 3.2.1

Skizze 3.2

$$l_1 = \left\| \underbrace{\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} x_1 \\ p(x_1) \end{pmatrix}}_{=|x_2 - p(x_1)|} \right\|$$

$$l_2 = \left\| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} \tilde{x}_1 \\ p(\tilde{x}_1) \end{pmatrix} \right\|$$

Da p' auf dem für uns interessanten Intervall beschränkt ist, gibt es eine Schranke const_d mit $\text{dist}(X) \leq \text{const}_d(x_2 - p(x_1))$ (das gilt natürlich immer für $\text{const}_d \geq 1$, denn $\text{dist}(X)$ ist gerade der Abstand; für $\text{dist}(X) \neq (x_2 - p(x_1))$ kann const_d sogar kleiner 1 angesetzt werden).

Nun gilt es, die Rückrichtung zu zeigen. Schätze als nächstes die Distanzfunktion nach unten ab. Es gilt

$$\text{dist}(X) = l_2 \quad |x_2 - p(x_1)| = l_1$$

Da alle unsere Betrachtungen innerhalb einer ε -Umgebung um den Nullpunkt stattfinden, können wir aufgrund der Glattheit von p o.B.d.A. annehmen, dass sowohl α deutlich kleiner als 90° als auch $\beta_i \geq 1^\circ$ ist. Es gilt also:

$$\frac{\sin \beta_1}{\sin \beta_2} < \text{const}_{\tilde{d}} < \infty$$

Aus dem Sinus-Satz folgt:

$$\begin{aligned} \frac{l_1}{l_2} &= \frac{\sin \beta_1}{\sin \beta_2} < \text{const}_{\tilde{d}} \\ \Rightarrow l_1 &\leq \text{const}_{\tilde{d}} l_2 \end{aligned} \quad (3.4)$$

Damit können wir zum Beweis von (3.3) übergehen. Für die folgenden Argumente genügt es $\lim_{x_1 \rightarrow 0}$ zu betrachten. Für $x_1 \rightarrow 0$ gilt nämlich $\lim_{x_1 \rightarrow 0} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ x_2 \end{pmatrix}$. Damit ist $P = \begin{pmatrix} x_1 \\ p(x_1) \end{pmatrix} = \begin{pmatrix} 0 \\ p(0) \end{pmatrix}$. Unter Verwendung der Tatsache, dass $X - \tilde{P} \perp \partial\Omega$ im Punkt \tilde{P} erhalten wir:

$$\begin{aligned} \left\langle \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} \tilde{x}_1 \\ p(\tilde{x}_1) \end{pmatrix}, \begin{pmatrix} \tilde{x}_1 \\ p(\tilde{x}_1) \end{pmatrix} \right\rangle &= 0 \\ \left\langle \begin{pmatrix} 0 \\ x_2 \end{pmatrix} - \begin{pmatrix} \tilde{x}_1 \\ p(\tilde{x}_1) \end{pmatrix}, \begin{pmatrix} \tilde{x}_1 \\ p(\tilde{x}_1) \end{pmatrix} \right\rangle &= 0 \\ \Rightarrow -\tilde{x}_1^2 + p(\tilde{x}_1)x_2 - p(\tilde{x}_1)^2 &= 0 \\ \tilde{x}_1^2 + p(\tilde{x}_1)^2 &= p(\tilde{x}_1)x_2 \quad \forall x_2 \geq 0 \end{aligned}$$

Damit folgt $\tilde{x}_1 = 0$, $p(\tilde{x}_1) = 0$, was anschaulich auch klar ist, denn die Ableitung im Punkte 0 ist null, weshalb folglich alle Vektoren der Form $\begin{pmatrix} 0 \\ x_2 \end{pmatrix}$ senkrecht auf dem Randpunkt $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$ stehen. Damit sind aber P und \tilde{P} identisch, und es gilt $\alpha = 0$ und $l_1 = l_2$. Und insbesondere gilt dies natürlich für $\mathbf{X} \rightarrow \mathbf{0}$. \square

Lemma 3.2.2. Sei $|\tilde{s}| \leq \varepsilon$. Dann ist

$$s^2 + (1 - p'(\tilde{s})s)^2 \asymp s^2 + 1 \quad (3.5)$$

Beweis:. Setze $p'(\tilde{s}) = v \leq \varepsilon$, denn die Ableitung ist glatt und nimmt im Nullpunkt den Wert 0 an, weshalb diese Annahme in der Umgebung U zulässig ist.

Fall 1: $|s| \leq \frac{1}{2}$

$$\begin{aligned} s^2 + (1 - vs)^2 &\leq s^2 + \left(1 + \frac{\varepsilon}{2}\right)^2 \\ &\leq 4(s^2 + 1) \quad \text{mit} \quad \frac{\varepsilon}{2} \leq \frac{1}{2} \end{aligned}$$

$$\begin{aligned} s^2 + (1 - vs)^2 &\geq s^2 + \left(1 - \frac{\varepsilon}{2}\right)^2 \\ &\geq \frac{1}{4}(s^2 + 1) \end{aligned}$$

Fall 2: $|s| \geq \frac{1}{2}$

$$\begin{aligned} s^2 + (1 - vs)^2 &\geq s^2 \\ &\geq \frac{1}{5}(s^2 + 1) \end{aligned}$$

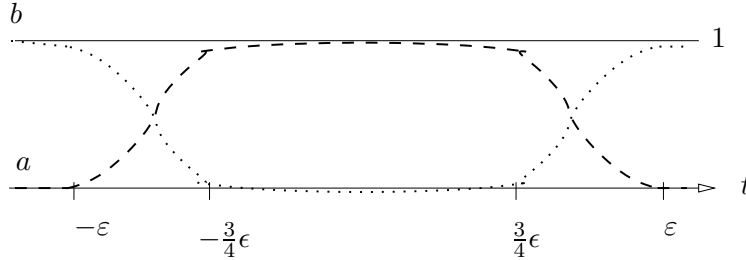
$$\begin{aligned} s^2 + (1 - vs)^2 &\leq s^2 + (2s + \frac{1}{2}s)^2 \\ &\leq 7(s^2 + 1) \end{aligned}$$

□

3.3 Glattheitseigenschaften

Theorem 3.3.1. Sei $g := \frac{\text{dist}(X)}{w(X)}$, dann ist g stetig auf $\bar{\Omega}$, und $g(X) > 0$ auf Ω .

Beweis. Als Hilfsmittel für den Beweis definiere ich noch zwei Funktionen $a(s)$ und $b(s)$ mit $s \in \mathcal{D}(\partial\Omega)$ und $a + b \equiv 1$.



Die Summe der beiden Funktionen a und b ist 1, kann also in das Integral hineinmultipliziert werden. Es soll gelten, dass $a(t) \equiv 1$ für $|t| \leq \frac{3}{4}\epsilon$.

$$g(X) = \int_{\partial\Omega_P} \frac{\text{dist}(x_1, x_2) \, ds_P}{\|(x_1, x_2) - P(s)\|^2} \quad \text{Definition von } p \text{ bei } (0, 0) \quad (3.6)$$

Und wir erhalten die äquivalente Formulierung

$$\begin{aligned} \tilde{g}(X) &= \int_{-\infty}^{\infty} \frac{(a(t) + b(t)) \text{dist}(x_1, x_2) \|(1, p'(t))\| \, dt}{\|(x_1, x_2) - (t, p(t))\|^2} \\ &= \int_{-\infty}^{\infty} \frac{(a(t) + b(t)) \text{dist}(x_1, x_2) \sqrt{1 + p'(t)^2} \, dt}{\left\| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} t \\ p(t) \end{pmatrix} \right\|^2} \end{aligned} \quad (3.7)$$

Das heißt: $g(X)$ und $\tilde{g}(X)$ liefern denselben Wert zurück.
Erklärungsbedürftig sind bei diesem Schritt zwei Dinge:

- Warum kann ich die Integrationsgrenzen so wählen? Normalerweise müsste ich diese ja korrekt mit-transformieren.
- Warum darf ich die Funktion p über ein so großes Intervall verwenden (eigentlich ist diese nur in einem genügend kleinen Intervall $[-\varepsilon, \varepsilon]$ definiert)? Letzteres wird automatisch mit der zuerst genannten Fragestellung geklärt werden

Das Integral kann mittels a und b (f bezeichne den Integranden) aufgesplittet werden in (siehe Abb 3.2)

$$\begin{aligned} \int_{\partial\Omega} f &= \int_{\partial\Omega \cap U} f + \int_{\partial\Omega \setminus U} f \\ &= \int_{|t| \leq \varepsilon} a(t)f(t)\sqrt{1+p'(t)^2} dt + \int_{\partial\Omega \setminus U} f \end{aligned} \quad (3.8)$$

Betrachten wir den b -Anteil, also den dritten Term, und klammern $\text{dist}(X)$ aus:

$$\text{dist}(X) \int_{-\infty}^{\infty} \frac{b(t)\|1, p'(t)\| dt}{\left\| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} t \\ p(t) \end{pmatrix} \right\|^2}$$

Was passiert, wenn wir $X \rightarrow \mathbf{0}$ gehen lassen (und das genau werden wir ja auch in diesem Beweis tun)? Innerhalb der Umgebung U ist p' beschränkt, damit ist es auch der Zähler des Integranden. Das heißt für $X \rightarrow \mathbf{0}$, was nichts anderes bedeutet als $\text{dist}(X) \rightarrow 0$ geht der Ausdruck gegen Null. Nun hat der Nenner zwar im Nullpunkt eine Singularität, aber dort ist $b \equiv 0$, das heißt, die Singularität braucht uns nicht zu kümmern. Der vorletzte Term geht also gegen null.

Außerhalb von U verwenden wir die ursprüngliche Darstellung von f , dies ist in Gleichung (3.8) der letzte Ausdruck. Dort gilt aber

$$g(X) = \text{dist}(X) \int_{\partial\Omega} \underbrace{\frac{ds}{\|X - P\|^2}}_{>0}$$

und damit geht der ganze Ausdruck problemlos gegen Null für $X \rightarrow \mathbf{0}$. Das heißt, dass wir uns nur noch dem a -Teil widmen müssen. Von diesem wissen wir aber, dass er außerhalb des Intervalles $[-\varepsilon, \varepsilon]$ laut Definition identisch Null ist, die etwas ~~legere~~ erscheinende Wahl der Grenzen nicht nur legitim, sondern auch vollkommen exakt ist und es gilt:

$$\lim_{X \rightarrow \mathbf{0}} \int_{\partial\Omega} f = \int a(t)f(t)\sqrt{1+p'(t)^2} dt$$

Doch zurück zum eigentlichen Beweis. Die nötigen Voraussetzungen für den Beweis sind hier nochmals stichwortartig zusammengefaßt

- Für a soll gelten: $a'(0) = 0$, $a(0) = 1$ und $a \leq 1$
- Da $p(0)$ und $p'(0)$ gleich null, folgt aus der Taylorentwicklung von $\partial\Omega$ bei 0

$$\begin{aligned} p(t) &= p(0) + p'(0)(t-0) + \frac{p''(\zeta)}{2}(t-0)^2 \quad \text{mit } \zeta \in [0, t] \\ &= 0 + 0 + \frac{p''(\zeta)}{2}(t-0)^2 \\ &= u \cdot t^2 \end{aligned}$$

Dabei gilt, dass $|u| \leq \text{const}_p$, also beschränkt, denn p ist zweimal stetig differenzierbar, und wird nur auf einem beschränkten Intervall betrachtet; damit muss auch p'' beschränkt sein, und es folgt die Behauptung.

- Die Ableitung von a ist beschränkt.
- $|X| \leq \delta$
- Es gilt $\text{dist}(X) < \text{const}_d(x_2 - p(x_1))$.

Wie schon zuvor argumentiert, können wir im Integralausdruck den b -Teil vernachlässigen. Es gilt dann:

$$g(X) = \int_{-\infty}^{\infty} \frac{\text{dist}(X)a(t)\sqrt{1+p'(t)^2} dt}{(x_1 - t)^2 + (x_2 - p(t))^2}$$

t wird nun substituiert:

$$\begin{aligned} t - x_1 &= s(x_2 - p(x_1)) \\ \Rightarrow dt &= ds(x_2 - p(x_1)) \end{aligned}$$

Eingesetzt in den Integranden folgt:

$$\begin{aligned} &\int_{-\infty}^{\infty} \frac{\overbrace{\text{dist}(X)a(s(x_2 - p(x_1)) + x_1))^{\leq 1} \sqrt{1+p'(s(x_2 - p(x_1)) + x_1)^2} (x_2 - p(x_1)) ds}{(x_1 - (x_1 + s(x_2 - p(x_1))))^2 + (x_2 - p((x_1 + s(x_2 - p(x_1))))))^2} \\ &\leq \int_{-\infty}^{\infty} \frac{\text{const}_d(x_2 - p(x_1))^2 \cdot 1 \cdot \sqrt{1+p'(s(x_2 - p(x_1)) + p(x_1))^2} ds}{(s(x_2 - p(x_1)))^2 + (x_2 - p((x_1 + s(x_2 - p(x_1))))))^2} \\ &\stackrel{\text{Abl. von } p \text{ beschr.}}{\leq} \int_{-\infty}^{\infty} \frac{\text{const}_d \text{const}_p ds}{s^2 + \frac{(x_2 - p((x_1 + s(x_2 - p(x_1))))))^2}{(x_2 - p(x_1))^2}} = \star \end{aligned}$$

Nun müssen wir uns noch um den Nenner kümmern. Entwickle p um den Punkt x_1 :

$$\begin{aligned} p(x_1 + s(x_2 - p(x_1))) &= p(x_1) + p'(w)(s(x_2 - p(x_1))) \\ w &\in [x_1, x_1 + s(x_2 - p(x_1))] \end{aligned}$$

Eingesetzt folgt ($h := x_2 - p(x_1)$):

$$\begin{aligned}
 \star &\leq \int_{-\infty}^{\infty} \frac{\text{const } ds}{s^2 + \frac{(h-p'(w)(sh))^2}{h^2}} \\
 &= \int_{-\infty}^{\infty} \frac{\text{const } ds}{s^2 + (1 - vs)^2} \\
 &\stackrel{\text{Lemma()}}{\leq} \text{const} \int_{-\infty}^{\infty} \frac{ds}{s^2 + 1} \\
 &\leq \int_{-\infty}^{\infty} \frac{ds}{s^2 + 1} \quad \text{wegen } \text{const}_d \text{const}_p \leq 1
 \end{aligned}$$

Nehmen wir die Konstanten nochmals genauer unter die Lupe, so folgt mit Lemma (3.2.1) und (3.3), dass $\lim_{\mathbf{x} \rightarrow \mathbf{0}} \text{const}_d = 1$. Außerdem wissen wir nach Voraussetzung, dass $p'(0) = 0$; damit aber gilt:

$$\begin{aligned}
 \lim_{\mathbf{x} \rightarrow \mathbf{0}} \text{const}_p &= \lim_{\mathbf{x} \rightarrow \mathbf{0}} \sqrt{1 + p'(s(x_2 - p(x_1)) + x_1)^2} \\
 &= 1 \\
 \lim_{\mathbf{x} \rightarrow \mathbf{0}} p'(w) &= 0
 \end{aligned}$$

Also gilt $\text{const} = 1$ und $\int_{-\infty}^{\infty} \frac{ds}{s^2+1}$ ist, wie wir schon gesehen haben, eine Majorante des Integranden. Mit dem Satz von der majorisierten Konvergenz (LEBESGUE) folgt die Konvergenz gegen $\underbrace{\pi a(0)}_{=1}$ (siehe auch [9] S.97ff). \square

3.4 Auswertung der Ableitung auf dem Rand ($\nabla u \neq 0$ auf $\partial\Omega$)

Wie schon im vorhergehenden Beweis für die Stetigkeit nehmen wir o.B.d.A. an, dass die Kurve durch den Nullpunkt laufe, sodass $P(t_0) = (0, 0) \in \partial\Omega$. Des weiteren sei die Kurve wieder glatt, weshalb wir die Kurve in Nullpunkt-Nähe bei Bedarf durch eine Funktion \tilde{p} ersetzen können:

$$P = \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} p_1(t_0) \\ p_2(t_0) \end{pmatrix} = \begin{pmatrix} 0 \\ \tilde{p}(0) \end{pmatrix} \quad t_{\text{Periodenbeginn}} \leq t_0 \leq t_{\text{Periodenende}}$$

Dabei soll gelten: $\tilde{p}'(0) = 0$. Aufgrund dieser Wahl von \tilde{p}' können wir den Normalenvektor im Punkt $\mathbf{0}$ mit $\vec{n} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$ angeben.

Theorem 3.4.1. Die Normalenableitung von w an einem beliebigen Kurvenpunkt ist (zeige dies hier o.B.d.A für den Nullpunkt) ungleich der Null.

Beweis. Die Richtungsableitung ist gegeben durch (verzichte aus Gründen der Über-

sichtigkeit, den Limes $\lim_{x_1, x_2 \rightarrow 0}$ innerhalb des Integrals immer aufzuschreiben):

$$\begin{aligned} Dw_n &= \left\langle \begin{pmatrix} 0 \\ -1 \end{pmatrix}, Dw(X) \right\rangle \\ &= (-1) \frac{(-1)(-1)}{\left(\int_{\partial\Omega} \frac{ds}{(x_1 - p_1)^2 + (x_2 - p_2)^2} \right)^2} \int_{\partial\Omega} \frac{-2(x_2 - p_2) ds}{((x_1 - p_1)^2 + (x_2 - p_2)^2)^2} \\ &= \frac{1}{\left(\int_{\partial\Omega} \frac{\text{dist}(X) ds}{(x_1 - p_1)^2 + (x_2 - p_2)^2} \right)^2} \int_{\partial\Omega} \frac{\text{dist}(X)^2}{((x_1 - p_1)^2 + (x_2 - p_2)^2)^2} 2(x_2 - p_2) ds = \star \end{aligned}$$

Das Hinzunehmen der dist -Funktion ist völlig legitim. Schliesslich handelt es sich hierbei um eine beliebige glatte Funktion. Damit erhalten wir aber im Nenner des ersten Faktors den schon berechneten Grenzwert zum Quadrat. Das heisst, dass wir uns im Folgenden nur auf das zweite Integral konzentrieren müssen:

$$\star = \frac{1}{\pi^2} \int_{\partial\Omega} \frac{\text{dist}(X)^2 2(x_2 - p_2) ds}{((x_1 - p_1)^2 + (x_2 - p_2)^2)^2}$$

Betrachte das Integral wieder in der Nähe des Nullpunktes. Für diese Zwecke ist es wieder sinnvoll, P als Funktion darzustellen. Des weiteren wird das Integral wieder mit $a(t) + b(t) = 1$ multipliziert, wobei der $b(t)$ -Teil wieder zu 0 wird für $\mathbf{X} \rightarrow \mathbf{0}$, weshalb ich den Term im weiteren Verlauf einfach weglassen werde.

$$\star = \frac{1}{\pi^2} \int_{-\infty}^{\infty} \frac{a(t) \sqrt{1 + p'(t)^2} 2(x_2 - p(t)) \text{dist}(X)^2 dt}{((x_1 - t)^2 + (x_2 - p(t))^2)^2} \quad (3.9)$$

Im Folgenden wird nur das übriggebliebene Integral betrachtet werden. Transformiere das Integral wie gehabt mit $t - x_1 = s(x_2 - p(x_1)) \Rightarrow dt = ds(x_2 - p(x_1))$

$$\int_{-\infty}^{\infty} \frac{a(x_1 + s(x_2 - p(x_1))) \sqrt{1 + p'(\dots)^2} \text{dist}(X)^2 (x_2 - p(x_1)) 2(x_2 - p(\dots)) ds}{((x_2 - p(x_1))^2 + (x_2 - p(x_1 + s(x_2 - p(x_1))))^2)^2} = \diamond$$

Eingesetzt folgt

$$\diamond \leq \int_{-\infty}^{\infty} \frac{\overset{\nearrow 1 \text{ für } x_1, x_2 \rightarrow 0}{a(x_1 + s(x_2 - p(x_1)))} \sqrt{\dots} \text{const}_d^2 (x_2 - p(x_1))^3 2(x_2 - p(\dots)) ds}{((x_2 - p(x_1))^2 + (x_2 - p(x_1 + s(x_2 - p(x_1))))^2)^2} = \clubsuit$$

Setze ein, dass $p(x_1 + s(x_2 - p(x_1))) = p(x_1) + p'(w)hs$ (Taylor-Entwicklung), wobei zur übersichtlicheren Darstellung die Abkürzungen $h = (x_2 - p(x_1))$ und $p'(w) = v$ verwendet werden sollen.

$$\begin{aligned} \clubsuit &\leq 2 \text{const}^2 \int_{-\infty}^{\infty} \frac{1 \sqrt{1 + p'(x_1 + s(x_2 - p(x_1)))^2} h^3 (x_2 - p(x_1) - p'(w)sh) ds}{(h^2 s^2 + (x_2 - p(x_1) - p'(w)hs)^2)^2} \\ &= 2 \text{const}^2 \int_{-\infty}^{\infty} \frac{\overset{\geq 1}{\sqrt{1 + p'(x_1 + s(x_2 - p(x_1)))^2}} h^4 (1 - vs) ds}{h^4 (s^2 + \frac{(h - vhs)^2}{h^2})^2} \\ &\leq 2 \text{const}^2 \int_{-\infty}^{\infty} \frac{1(1 - vs) ds}{(s^2 + \frac{(h - vhs)^2}{h^2})^2} \\ &= 2 \text{const}^2 \int_{-\infty}^{\infty} \frac{1 - vs ds}{(s^2 + (1 - vs)^2)^2} = \heartsuit \end{aligned}$$

Gehen wir nun wieder zur punktweisen Konvergenz über, so wird durch Limesbildung $vs \rightarrow 0$ gehen, denn $v = p'(w)$ mit $w \in [0, \varepsilon]$. Des weiteren wissen wir nach Lemma (3.2.1), dass die vorgezogene Konstante den Grenzwert 1 besitzt. Damit gilt:

$$\begin{aligned}\heartsuit &= \lim_{\mathbf{x} \rightarrow \mathbf{0}} 2\text{const} \int_{-\infty}^{\infty} \frac{1 - vs \, ds}{(s^2 + (1 - vs)^2)^2} \\ &= 2 \int_{-\infty}^{\infty} \frac{ds}{(s^2 + 1)^2} \\ &= 2 \left(\frac{1}{2} \frac{s}{s^2 + 1} + \frac{1}{2} \arctan(s) \right) \Big|_{-\infty}^{\infty} \\ &= 2 \frac{1}{2} \left(\frac{\pi}{2} - \left(-\frac{\pi}{2} \right) \right) \\ &= \pi\end{aligned}$$

Fasst man nun wieder beide Integrale (3.9) zusammen, so erhält man

$$\begin{aligned}\star &= \frac{1}{\pi^2} \pi \\ &= \frac{1}{\pi} > 0\end{aligned}$$

Es kann also sogar für jeden Randpunkt exakt die Normalenableitung mit $\frac{1}{\pi}$ angegeben werden. \square

Dieses Ergebnis kann im folgenden Beispiel problemlos reproduziert werden.

Beispiel 3.4.1 (Ein leicht zu rechnender Spezialfall). Zeige für den einfachsten Fall - den Einheitskreis - dass mit der folgenden Methode der Gradient an einem Punkt und insbesondere am Rand berechnet werden kann.

$$w(X)^{-1} = \int_{\partial\Omega} \frac{ds}{\|X - X(s)\|^2}$$

Teste die Funktion für den Einheitskreis und $X = [r, 0]$

$$\int_0^{2\pi} \frac{dt}{(\cos t - r)^2 + \sin^2 t}$$

Der Nenner hat dabei den Wert (folgt direkt aus der Berechnung über das Skalarprodukt)

$$1 - 2r \cos t + r^2$$

Setze des weiteren $z = \exp(it)$, und $dz = iz \cdot dt$, so gilt

$$\begin{aligned}
 w(X)^{-1} &= \frac{1}{i} \oint \frac{\frac{dz}{z}}{1 - r(z + \frac{1}{z}) + r^2} \\
 &= -\frac{1}{ir} \oint \frac{dz}{\underbrace{z^2 - (r + \frac{1}{r})z + 1}_{=(z-r)(z-\frac{1}{r})}} \\
 &= -\frac{2\pi}{r} \begin{cases} (r - \frac{1}{r})^{-1} & \text{für } r < 1 \\ (\frac{1}{r} - r)^{-1} & \text{für } r > 1 \end{cases} \\
 &= \pm \frac{2\pi}{1 - r^2} \tag{3.10}
 \end{aligned}$$

Leitet man nach r ab (dies entspricht der gewünschten Normalenableitung), so erhält man

$$u'(r) = \mp \frac{r}{\pi},$$

also die Ableitung $\frac{1}{\pi}$ am Rand, und dies stimmt erfreulicherweise mit dem hergeleiteten Ergebnis überein.

Kapitel 4

Numerische Umsetzung

Da zum Verständnis der numerischen Lösung wieder ein gewisses Vorwissen hilfreich sein könnte, werden auch in diesem eher praktisch orientierten Teil grundlegende Sachverhalte nochmals kurz eingeführt. Den einzelnen wichtigen Programmabschnitten wird dabei jeweils ein Kapitel gewidmet sein. Die Bernstein-Polynome und B-Splines habe ich als Vorbereitung in einem eigenen Kapitel zusammengefasst, da der Umfang den Rahmen einer kurzen Erläuterung zum Programmtext sprengen würde.

4.1 Theoretische Vorarbeit

4.1.1 Bézier-Technik für Kurven

Das im Zusammenhang mit der web-Methode aufgebaute Softwareprojekt verwendet im (vorerst) zweidimensional umgesetzten Fall Bézier-Kurven zur Beschreibung der Gebietsränder. Ein großer Vorteil der Bézier-Kurven ist z.B. der direkte Zusammenhang zwischen den Grunddaten (Bézier-Punkte) und der geometrischen Gestalt der Kurve; ein weiterer der direkt und allgemein bekannte Zusammenhang zwischen Ableitung und Funktion. Dazu aber im Folgenden mehr (siehe auch [18]).

Die Bernstein Polynome

Für die Paramterdarstellung von Kurven wir die spezielle Basis der Bernstein-Polynome verwendet. Dass für die nun zu beweisenden Sätze und Eigenschaften lediglich das Intervall $[0, 1]$ betrachtet wird nimmt diesen nicht die Allgemeingültigkeit, denn durch affine Transformation läßt sich jedes beliebige reelle Intervall $[a, b]$ auf das Einheitsintervall abbilden:

$$\lambda = \frac{t - a}{b - a}. \quad (4.1)$$

Des weiteren besagt der binomische Lehrsatz, dass

$$\begin{aligned} 1 &= (\lambda + (1 - \lambda))^n \\ &= \sum_{i=0}^n \binom{n}{i} \lambda^i (1 - \lambda)^{n-i} \end{aligned} \quad (4.2)$$

Dies lässt sich sehr leicht mittels Vollständiger Induktion verifizieren. Für $n = 0, 1$ ist die Gültigkeit offensichtlich. Kommen wir also zum Induktionsschritt:

$$\begin{aligned} \lambda + (1 - \lambda)^{n+1} &= \binom{n}{0} \lambda^0 (1 - \lambda)^{n+1} \\ &\quad + \binom{n}{1} \lambda^1 (1 - \lambda)^n + \binom{n}{0} \lambda^1 (1 - \lambda)^n \\ &\quad \dots \\ &\quad + \binom{n}{n} \lambda^n (1 - \lambda)^1 + \binom{n}{n-1} \lambda^n (1 - \lambda)^1 \\ &\quad + \binom{n}{n} \lambda^{n+1} (1 - \lambda)^0 \\ &= \sum_{i=0}^{n+1} \binom{n+1}{i} \lambda^i (1 - \lambda)^{n+1-i} \end{aligned} \quad (4.3)$$

Diese Zerlegung der Einsfunktion führt zur

Definition 4.1.1. Unter dem i -ten Bernstein-Polynom vom Grad n bezüglich des Einheitsintervalles versteht man das Polynom

$$B_i^n(\lambda) := \binom{n}{i} \lambda^i (1 - \lambda)^{n-i}, \quad (i = 0, \dots, n) \quad (4.4)$$

Das i -te Bernstein-Polynom bezüglich eines beliebigen Intervalles ist durch Einsetzen der Transformation (4.1) gegeben.

Einige wichtige Eigenschaften der Bernstein-Polynome formuliert der folgende

Satz 4.1.1. Für die Bernstein-Polynome B_i^n über dem Einheitsintervall gilt:

$$\lambda = 0 \quad \text{ist } i\text{-fache Nullstelle von } B_i^n(\lambda) \quad (4.5)$$

$$\lambda = 1 \quad \text{ist } n - i\text{-fache Nullstelle von } B_i^n(\lambda) \quad (4.6)$$

$$B_i^n(\lambda) = B_{n-i}^n(\lambda) \quad \text{Symmetrie} \quad (4.7)$$

$$(1 - \lambda) B_0^n(\lambda) = B_0^{n+1}(\lambda), \quad \lambda B_n^n(\lambda) = B_{n+1}^{n+1}(\lambda) \quad (4.8)$$

$$0 \leq B_i^n(\lambda) \leq 1 \quad \text{für } \lambda \in [0, 1] \quad (i = 0, 1, \dots, n) \quad (4.9)$$

Beweis. Die ersten beiden Eigenschaften gehen direkt aus der Definition der Bernstein-Polynome hervor. Die Symmetrieeigenschaft (4.7) lässt sich auch sehr leicht zeigen.

Denn es gilt:

$$\begin{aligned}
 B_i^n(\lambda) &= \binom{n}{i} \lambda^i (1-\lambda)^{n-i} \\
 &= \binom{n}{n-i} (1-\lambda)^{n-i} \lambda^{n-(n-i)} \\
 &= B_{n-i}^n(1-\lambda)
 \end{aligned} \tag{4.10}$$

Die beiden Eigenschaften (4.8) ergeben sich aus den Werten für n folgendermaßen:

$$\begin{aligned}
 (1-\lambda)B_0^n(\lambda) &= (1-\lambda) \binom{n}{0} (1-\lambda)^n \\
 &= \binom{n+1}{0} (1-\lambda)^{n+1} = B_0^{n+1}(\lambda) \\
 \lambda B_n^n(\lambda) &= \lambda \binom{n}{n} \lambda^n \\
 &= \binom{n+1}{n+1} \lambda^{n+1} = B_{n+1}^{n+1}(\lambda).
 \end{aligned} \tag{4.11}$$

Die im letzten Punkt proklamierte Nichtnegativität ist aufgrund der Wahl des Einheitsintervalles klar, denn dort ist sowohl λ als auch $(1-\lambda)$ größer 0. Damit sind die Bernstein-Polynome im Einheitsintervall nicht negativ. \square

Ein weiterer wichtiger Aspekt der Bernstein-Polynome ist deren Basiseigenschaft im Vektorraum \mathbb{P}_n .

Satz 4.1.2. Sei $n \in \mathbb{N}$ fest gewählt, dann bilden die Bernstein-Polynome für $i = 0, \dots, n$ eine Basis des Vektorraumes \mathbb{P}_n der reellen Polynome vom Grad n .

Beweis:. Wieder einmal setzen wir die Linearkombination

$$\sum_{i=0}^n c_i B_i^n(\lambda) = 0, \quad c_i \in \mathbb{R}, \quad \lambda \in [0,1] \tag{4.12}$$

woraus notwendigerweise für die lineare Unabhängigkeit der Bernsteinpolynome folgen muss, dass $c_i = 0$ für $(i = 0, \dots, n)$. Setzen wir $\lambda = 1$, so ist $B_i^n(1) = 0$ für $i = 0, \dots, n-1$. Da das Bernsteinpolynom $B_n^n(1) = 1$ ist folgt damit, dass $c_n = 0$ sein muss. Da die Linearkombination eine stetig differenzierbare Funktion ist, muss auch die Ableitung identisch mit der Nullfunktion sein. Da aber $B_i^n(\lambda)$ eine $n-i$ -fache Nullstelle an der Stelle $\lambda = 1$ besitzt und außerdem $B_{n-1}^{n'}(1) \neq 0$ ist, so folgt wegen

$$\begin{aligned}
 0 &= \frac{d}{d\lambda} \sum_{i=0}^n c_i B_i^n(\lambda) \Big|_{\lambda=1} \\
 &= c_{n-1} B_{n-1}^{n'}(1)
 \end{aligned} \tag{4.13}$$

dass $c_{n-1} = 0$. Analog kann Schritt für Schritt jede weitere Ableitung herangezogen werden, um zu zeigen, dass $c_n = c_{n-1} = \dots = c_0 = 0$ gilt. Da also die $(n+1)$ Bernsteinpolynome linear unabhängig sind, und zudem noch die Dimension $m = n+1$ besitzen, folgt ihre Basiseigenschaft. \square

Eine wichtige und schöne Eigenschaft der Bernstein-Polynome ist die einfache Berechenbarkeit ihrer Ableitungen.

Satz 4.1.3. *Ausgehend von der Definition gilt für die Ableitungen der Bernstein-Polynome:*

$$\frac{d}{d\lambda} B_i^n(\lambda) = \begin{cases} -nB_0^{n-1}(\lambda) & \text{für } i = 0 \\ n[B_{i-1}^{n-1}(\lambda) - B_i^{n-1}(\lambda)] & \text{für } i = 1, 2, \dots, n-1 \\ nB_{n-1}^{n-1}(\lambda) & \text{für } i = n \end{cases} \quad (4.14)$$

Beweis: Ableiten der Definition des Bernsteinpolynomes liefert unter Anwendung der Produktregel:

$$\frac{d}{d\lambda} B_i^n(\lambda) = \binom{n}{i} [i\lambda^{i-1}(1-\lambda)^{n-i} - (n-i)\lambda^i(1-\lambda)^{n-i-1}]$$

Die Spezialfälle $i = 0$ bz $i = n$ lassen sich im Prinzip sofort ablesen. Für die Übrigen ($1 \leq i \leq n-1$) gelten für die Binomialkoeffizienten:

$$\binom{n}{i} i = n \binom{n-1}{i-1}, \quad \binom{n}{i} (n-i) = n \binom{n-1}{i}$$

Damit folgt aber direkt die Behauptung. \square

Bézier-Kurven

Um Kurven in der Ebene (oder im Raum) approximieren zu können, greift man auf stückweise Polynominterpolation zurück. Da die Bernstein-Polynome eine Basis des Vektorraums \mathcal{P}_n sind, lassen sich die Raumkoordinaten je als Linearkombinationen der Bernstein-Polynome darstellen:

$$x_k(t) := \sum_{i=0}^n b_{ik} B_i^n(t; a, b) \quad k = 1, 2, \dots, d$$

d Dimension

$[a, b]$ allgemeines Parameterintervall

b_{ik} Koeffizientenvektor

Damit gelangen wir zur Bézier-Darstellung

$$\mathbf{x}(t) := (x_1(t), \dots, x_d(t))^T$$

Die Bézier-Punkte haben auch eine tatsächliche geometrische Bedeutung, denn die Koordinaten des d -dimensionalen Polynoms $P(t) \in \mathbb{R}_n^d$ bilden das sogenannte Bézier-Polygon, welches bezüglich der Kurve eine konvexe-Hülle-Eigenschaft besitzt.

Satz 4.1.4 (Konvexe Hülle). *Die Menge der Punkte der Bézier-Kurve*

$$M := \left\{ \mathbf{x}(\lambda) = \sum_{i=0}^n b_i B_i^n(\lambda) : \lambda \in [0, 1] \right\}$$

liegt in der konvexen Hülle der $n + 1$ Bézier-Punkte b_0, b_1, \dots, b_n .

Beweis: Als Definitionsbereich sei aus den schon bekannten Gründen das Einheitsintervall gewählt. Für $\lambda \in [0, 1]$ gilt nach (4.9) $0 \leq B_i^n(\lambda) \leq 1$. Da die Bernstein-Polynome eine Partition der Eins bilden, also $\sum_{i=0}^n B_i^n(\lambda) = 1$, stellt das Kurvensegment $\mathbf{x}(\lambda) = \sum_{i=0}^n B_i^n(\lambda) b_i$ eine lineare Konvexkombination der Bézier-Punkte dar. Damit aber liegt diese in der konvexen Hülle der $n + 1$ Bézier-Punkte. \square

Im Folgenden möchte ich mich mit den Kurvenenden beschäftigen, denn das erklärte Ziel ist es, mehrere Kurvensegmente mit gewünschter Glattheit oder zumindest stetig zusammenzufügen.

Satz 4.1.5. *Die Randpunkte einer Bézier-Kurve $\mathbf{x}(\lambda) = \sum_{i=0}^n b_i B_i^n(\lambda)$, $n \geq 2$ sind Anfangs- und Endpunkte derselben. Die Richtung der Tangente an die Kurve in den Randpunkten hängt lediglich vom Randpunkt selbst und seinem nächsten Nachbarn ab und stimmt mit der Richtung der Verbindungslinie der beiden Punkte überein. Entsprechend hängt die zweite Ableitung im Randpunkt von diesem und seinen zwei nächsten Nachbarn ab:*

$$\mathbf{x}(0) = b_0, \quad \mathbf{x}(1) = b_n \tag{4.15}$$

$$\mathbf{x}'(0) = n(b_1 - b_0), \quad \mathbf{x}'(1) = n(b_n - b_{n-1}) \tag{4.16}$$

$$\mathbf{x}''(0) = n(n-1)(b_2 - 2b_1 + b_0)$$

$$\mathbf{x}''(1) = n(n-1)(b_n - 2b_{n-1} + b_{n-2})$$

Beweis: (4.15) folgt direkt zum einen aus der Definition der Bernstein-Polynome, als auch aus der Tatsache heraus, dass (4.5, 4.6) λ an der Stelle 0 und 1 je eine i - bzw. $n - i$ fache Nullstelle besitzt. Übrig bleibt nur das 0-te bzw. das n -te Bézier-Polynom, welches an diesen Stellen jeweils den Wert 1 hat, und damit \mathbf{x} den Wert b_0 bzw. den Wert b_n .

Zu (4.16). Die Ableitungen berechnen sich nach (4.14) an der Stelle λ zu

$$\begin{aligned} \mathbf{x}'(\lambda) &= -nb_0 B_0^{n-1}(\lambda) + \sum_{i=1}^{n-1} nb_i [B_{i-1}^{n-1}(\lambda) - B_i^{n-1}(\lambda)] + nb_n B_{n-1}^{n-1}(\lambda) \\ &= n \sum_{i=0}^{n-1} (b_{i+1} - b_i) B_i^{n-1}(\lambda) \end{aligned}$$

Unter Verwendung von (4.5),(4.6) folgt sofort die Behauptung.
Für die zweite Ableitung ist die Argumentation analog. \square

Diese Eigenschaften versetzen uns nun in die Lage, Kurven mit gewünschter Glätte (Werte für höhere Ableitungen lassen sich nach dem oben beschriebenen Schema berechnen) zusammenzusetzen.

Zur Auswertung eines Kurvensegmentes kann der Algorithmus von *de Casteljau* verwendet werden. Ähnlich wie bei der Auswertung von B-Splines wird hier die rekursive Definition der Bernstein-Polynome angewandt, wodurch sich letztendlich die Rechenschritte auf Konvexkombinationen von Bézier-Punkten reduzieren lassen - eine Vorgehensweise, die wieder ganz einfach als Dreiecksschema dargestellt werden kann. Ich möchte es an dieser Stelle mit der Angabe dieser Beweisidee bewenden lassen, denn der Beweis ist sehr technisch, und kann in fast jedem Numerik-Lehrbuch nachgelesen werden.

4.2 Die Vorgehensweise im Überblick

Da ich im folgenden Kapitel Algorithmus und kurze theoretische Einführung immer zu einer Einheit zusammenfassen werde, möchte ich an dieser Stelle einen groben Überblick über den Gesamt Ablauf geben, um somit die wichtigsten Schritte herauszustreichen.

Die Vorgabe für das Programm war, eine Routine zu schreiben, welche innerhalb des von mehreren Mitarbeitern und Studenten des Lehrstuhls unter Leitung von Herrn Höllig entwickelten *web-Spline* Projekts verwendet werden kann. Auf Anfrage soll es den Wert (bzw. zusätzlich weitere Ableitungen) der hier behandelten Gewichtsfunktion über einem durch Bézier-Segmente definierten Gebiet liefern. Die eigentliche Auswertung geschieht mittels Quasiinterpolanten über der an diskreten Stellen ausgewerteten Gewichtsfunktion. Die Schritte im einzelnen:

1. Aufruf des Programmes *web_weight_qifc* mit oder ohne entsprechende Optionen. Das Programm überprüft, ob das Programm in einem vorigen Durchlauf schon mit den Optionen aufgerufen wurde. Wenn ja, kann sofort die Auswertung erfolgen.
2. Ist die Initialisierung noch nicht erfolgt, muss zunächst ein gleichmäßiges Gitter über einer das Gebiet umschließenden Box erzeugt werden.
3. Markiere diejenigen Gitterpunkte, welche nahe an Randpunkten des Gebietes liegen, um sie später gesondert behandeln zu können.
4. Auswertung an den Gitterpunkten. Als Integrationsverfahren (siehe Definition der Gewichtsfunktion) wird der Romberg-Algorithmus verwendet, markierte Randpunkte werden aufgrund der Gefahr einer Polstelle mittels eines geschachtelten Romberg-Algorithmus' berechnet.
5. Versee äußere Punkte mit negativem Vorzeichen, um die Gewichtsfunktion nach außen glatt fortsetzen zu können.

6. Auswertung mittels Quasiinterpolant und Speicherung der berechneten Werte in einer globalen Variablen.

4.3 Programmcode und Erläuterungen

4.3.1 Markierung der randnahen Punkte

Das Hauptproblem bei der numerischen Umsetzung stellt die Integration im Nenner der Umkehrfunktion $w(X)^{-1}$ dar. Als Integrationsverfahren wird hier die Romberg-Iteration verwendet (dies ist zwar nicht unbedingt das schnellste, aber dafür ein relativ stabiles Verfahren). Um zu verhindern, dass das Verfahren an einer Polstelle in eine nicht abbrechende Schleife verfällt, wird zunächst überprüft, welche der in der aufzustellenden Wertematrix enthaltenen diskreten Gitterpunkte nahe an der Randkurve liegen. Diese werden dann markiert und bei der Berechnung der Gewichtsfunktion mit einem geschachtelten Romberg-Verfahren behandelt. Zur Markierung der Punkte durchläuft das Programm alle Randkurven, und unterteilt diese in gleich lange Strecken (halbe Gitterweite), sprich, berechnet, nach welcher Zeit t entlang der Kurve eine Strecke von $a = \frac{\text{Gitterweite}}{2}$ zurückgelegt wurde. Da sich die Umlaufgeschwindigkeit auf der Randkurve ständig ändern kann, ist es nicht möglich, zunächst die Gesamtlänge s_{gesamt} der Kurve zu berechnen, dann durch die Rasterweite der Gitterpunkte zu teilen und im Anschluss daran den Definitionsbereich der Kurve $[t_{\text{Start}}, t_{\text{Ende}}]$ einfach durch die oben erhaltenen Anzahl der Unterteilungspunkte zu teilen.

Das Programm geht wie folgt vor:

- Abfrage der Rasterweite des Gitters, auf welchem später gerechnet wird.
- Berechnung des nächsten Punktes t_{k+1} , für den gilt:

$$s(t_{k+1}) - s(t_k) \approx \frac{\text{Rasterweite}}{2}$$

Wobei $k \in \dots$ und $t_{k+1} > t_k$ gilt, und $s(t)$ der ab dem Zeitpunkt t_0 zurückgelegte Weg ist.

- Berechnung der Kurvenpunkte an den t_i und Markierung der umliegenden Gitterpunkte. Daher auch der zur Sicherheit etwas feiner gewählte Abstand a ; dies stellt sicher dass auch tatsächlich **alle** randnahen Punkte berücksichtigt werden.

Weglängen und deren näherungsweise Berechnung

Will man die Länge eines Weges $\gamma : [a, b] \rightarrow \mathbb{R}^p$ berechnen, so ist anschaulich klar, dass man durch Zerlegung des Intervalles $Z := \{t_0, t_1, \dots, t_n\}$ von $[a, b]$ und Bestimmung der Abstände $|\gamma(t_k) - \gamma(t_{k-1})|$ je zweier aufeinander folgender Punkte und Addition derselben eine Näherung des Weges erhalten kann.

$$L(\gamma, Z) := \sum_{k=1}^n \|\gamma(t_k) - \gamma(t_{k-1})\|$$

Durch eine Verfeinerung der Zerlegung wächst auf Grund der Dreiecksungleichung der Betrag von $L(\gamma, Z)$ an. Ist L beschränkt für alle Zerlegungen, so existiert für $n \rightarrow \infty$ ein Grenzwert $\sup_Z L(\gamma, Z)$, welchen wir als die Länge des Weges ansehen.

Definition 4.3.1. Ein Weg $\gamma : [a, b] \rightarrow \mathbb{R}^p$ heisst rektifizierbar, wenn wir für alle Zerlegungen Z des Intervalles $[a, b]$ die Summe $L(\gamma, Z) := \sum_{k=1}^n |\gamma(t_k) - \gamma(t_{k-1})|$ nach oben durch eine Konstante M abschätzen können. Die Zahl $\sup_Z L(\gamma, Z)$ wird dann Länge von γ genannt.

Da unsere Ränder zumindest stückweise stetig differenzierbar sind, ist der Beweis des folgenden Satzes vollkommen ausreichend für unsere Zwecke.

Satz 4.3.1. Der Weg $\gamma : [a, b] \rightarrow \mathbb{R}^p$ sei stetig differenzierbar. Dann ist er rektifizierbar, seine Weglängenfunktion s ist stetig differenzierbar, und für alle $t \in [a, b]$ gilt:

$$\dot{s}(t) = \|\dot{\gamma}(t)\|.$$

Die Länge $L(\gamma)$ des Weges γ berechnet sich nach der Formel

$$\begin{aligned} L(\gamma) &= \int_a^b \|\dot{\gamma}(t)\| \, dt \\ &= \int_a^b \sqrt{\dot{\gamma}_1^2 + \cdots + \dot{\gamma}_p^2} \, dt \end{aligned}$$

Beweis:. Sei $Z := \{t_0, \dots, t_n\}$ eine beliebige Zerlegung des Intervalles $[a, b]$. γ und damit auch $\dot{\gamma}$ sind vektorwertig. Dann gilt mit der Dreiecksungleichung, dass

$$\begin{aligned} \|\gamma(t_k) - \gamma(t_{k-1})\| &= \left\| \int_{t_{k-1}}^{t_k} \dot{\gamma}(t) \, dt \right\| \\ &\leq \int_{t_{k-1}}^{t_k} \|\dot{\gamma}(t)\| \, dt \end{aligned}$$

Durch Summation folgt

$$\sum_{k=1}^n \|\gamma(t_k) - \gamma(t_{k-1})\| \leq \int_a^b \|\dot{\gamma}\| \, dt$$

Da die rechte Seite unabhängig von einer Zerlegung ist, folgt, dass γ rektifizierbar ist, und es gilt:

$$L(\gamma) \leq \int_a^b \|\dot{\gamma}(t)\| \, dt \tag{4.17}$$

Sei nun $t \in (a, b]$ und $h < 0$ so gewählt, dass $t + h \leq b$. Aus unseren vorherigen Überlegungen wissen wir, dass die Verbindungslinie von $\gamma(t)$ nach $\gamma(t + h)$ immer kleiner sein muss, als der auf dem Intervall $[t, t + h]$ zurückgelegte Weg. Es gilt also:

$$|\gamma(t + h) - \gamma(t)| \leq s(t + h) - s(t)$$

Unter Verwendung von (4.17) gilt somit

$$\left\| \frac{\gamma(t+h) - \gamma(t)}{h} \right\| \leq \frac{s(t+h) - s(t)}{h} \leq \frac{1}{h} \int_t^{t+h} \|\dot{\gamma}(\tau)\| d\tau$$

Bilden wir den Grenzwert $h \rightarrow 0$, so strebt der linke Term gegen $\|\dot{\gamma}(t)\|$, der rechte nach dem zweiten Hauptsatz der Differential- und Integralrechnung ebenso.

Der linksseitige Grenzwert folgt analog, weshalb s stetig differenzierbar ist. Damit folgen die Behauptungen. \square

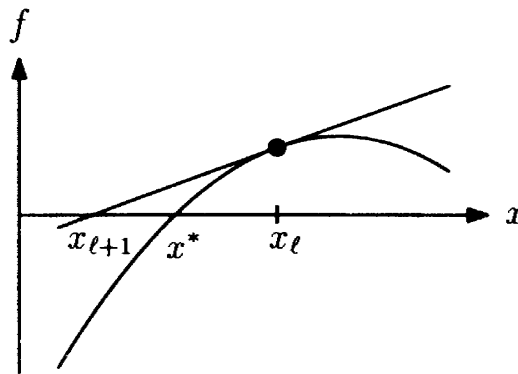


Abbildung 4.1: Newton-Verfahren

[13]

Nun geht es ja in der zu programmierenden Anwendung darum, zunächst eine ungefähre, schnell berechenbare Näherung von t_k für ein vorgegebenes t_{k-1} zu berechnen, sodass gilt:

$$s(t_k) - s(t_{k-1}) \approx \frac{\text{Rasterweite}}{2}$$

Die halbe Rasterweite verwende ich hier aus Sicherheitsgründen, damit keiner der Gitterpunkte unberücksichtigt bleibt. Eine einfache Iteration zur Näherung können wir mit Hilfe einer Taylorentwicklung um den Punkt t_{k-1} erhalten: Dabei bezeichne a die vorgegebene halbe Rasterweite und s die Weglänge mit $\dot{s} = \|\dot{\gamma}\|_2 = \sqrt{\gamma_1^2 + \dots + \gamma_n^2}$.

$$\begin{aligned} s(t_k) &= s(t_{k-1})(t_k - t_{k-1})^0 + \dot{s}(t_{k-1})(t_k - t_{k-1}) + \text{Rest} \\ \Leftrightarrow a &\approx \dot{s}(t_{k-1})(t_k - t_{k-1}) \\ \Leftrightarrow t_k &\approx \frac{a}{\dot{s}(t_{k-1})} + t_{k-1} \end{aligned}$$

Je enger hierbei die einzelnen t_i liegen, umso besser wird natürlich die Näherung.

Diese erste Näherung kann nun zur genaueren Bestimmung des Punktes t aus dem Definitionsbereich in ein Newton-Verfahren eingesetzt werden.

Die Startnäherung t_0 ist zumindest so gut, dass eine Konvergenz des Newton-Verfahrens gegen einen anderen als den gewünschten Punkt

$$t_{k+1} = t_k - \frac{f(t_k)}{f'(t_k)}$$

nicht zu befürchten ist. Da das Newton-Verfahren gegen einen t -Wert konvergiert, an welchem eine Nullstelle der betrachteten Funktion vorliegt, muss für f

$$f(t_k) = \int_{t_0}^{t_k} \dot{s}(\tau) d\tau - k \frac{\text{Rasterweite}}{2}$$

$$f'(t_k) = \dot{s}(t_k)$$

eingesetzt werden. (siehe Abb (4.1)).

Umsetzung

Programm 1: Initialisierung: Entwurf eines Gitters und Markierung randnaher Punkte

```

Deklariere die globalen Variablen WEB_WEIGHT_QIFC, WEB_GRID
Lade globale Variable WEB_BOUNDARY (Segmente, Kurven)
WEB_GRID[x,y,Zellbreite]=web_grid_init(Gebiet,Zellbreite) (Gitterinitialisierung)
Aufstellen der Punktematrix G mit  $x, y, z, m$  (Markierung)
Kurvenunterteilung in Zellbreite/2
for  $wb1 = 1$  to  $length(WEB\_BOUNDARY)$  do
  for  $wb2 = 1$  to  $length(WEB\_BOUNDARY\{wb1\})$  do
    Startnäherung  $t_{next} = \frac{Zellbreite}{2\sqrt{\sum Tangentialvektor.^2}} + t_{alt}$ 
    Suche nächsten Rasterpunkt mittels Newton
    while  $genauigkeit > eps * 10000$  do
      Romberg(web_bez_eval,0 :  $\frac{Zellbreite}{2}$  :  $t_{next}$ )
    end while
  end for
end for
Suche randnahe Punkte in der Wertematrix
for  $i = 1$  to  $length(Rasterpunkte)$  do
  Suche  $x$ -Wert größer als
   $a = \max(\text{End}(x\text{-Werte von } G) ; \text{aktueller } x\text{-Wert von Rasterpunkte}(i))$ 
   $b = \max(\text{End}(y\text{-Werte von } G) ; \text{aktueller } y\text{-Wert von Rasterpunkte}(i))$ 
   $m = m + 1$ 
   $G(4(b-1) : 4 : 4 * (b+1), a-1 : a+1) = G(4(b-1) : 4 : 4(b+1), a-1 : a+1) + 1$ 
end for

```

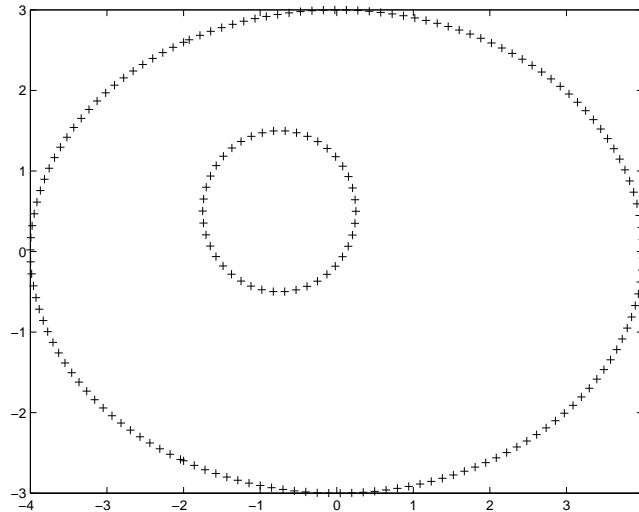
Ergebnis

Abbildung 4.2: Rasterung der Kurve

Als Auswertungsgebiet habe ich eine Ellipse mit einem eingeschriebenen Kreis ausgewählt. Dieses Beispiel werde ich bis zum Schluss verwenden, um die Veränderung in den einzelnen Schritten deutlicher herausstellen zu können.

Bild (4.2) zeigt die zunächst erfolgte Unterteilung der Kurve in $\frac{\text{zellbreite}}{2}$ -Stücke. Danach werden - wie schon erwähnt - die nächstliegenden Matrixeinträge (diese entsprechen dem über das Gebiet gelegten Gitter) markiert. Das Bild zeigt auf dem Einsniveau die markierten Matrixpunkte, ungefährliche Punkte liegen auf Nullniveau (Abb. (4.3)).

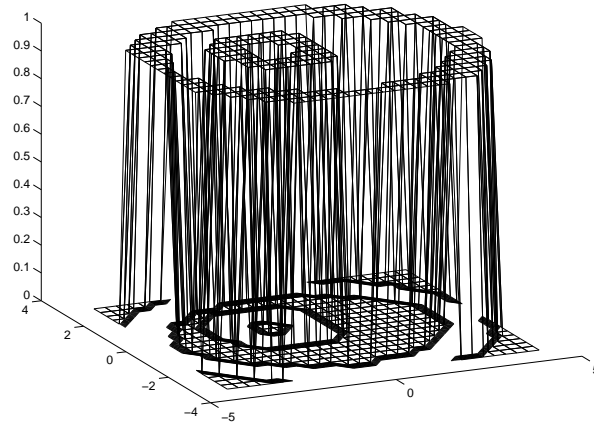


Abbildung 4.3: Markierung der Matrix

4.3.2 Berechnung der Gewichtsfunktion auf den Gitterpunkten

Nachdem die Unterteilung in randferne und randnahe Punkte erfolgt ist, kommen wir nun zur eigentlichen Berechnung der Gewichtsfunktion. Genauer gesagt: es wird die Gewichtsfunktion innerhalb, und der Betrag der glatten Fortsetzung ausserhalb des Gebietes berechnet.

Das Gebiet ist - wie schon erwähnt - durch eine Bézier-Kurve beschrieben, welche durch Kontrollpunkte und die dazugehörigen Gewichte festgelegt ist. In unserem Fall wurden die Werte in der globalen Variablen `WEB_BOUNDARY` abgelegt.

```

>> WEB_BOUNDARY

WEB_BOUNDARY =

    [1x3 struct]    [1x3 struct]

>> WEB_BOUNDARY{1}

ans =

1x3 struct array with fields:
    degree
    points

>> WEB_BOUNDARY{1}(1)

ans =

    degree: 2

    points: [3x3 double]

>> WEB_BOUNDARY{1}(3).points

ans =

    -2.0000    -2.5981     1.0000
     2.0000    -2.5981     0.5000
     4.0000         0     1.0000

```

Abbildung 4.4: Die Matlab-Ausgabe der globalen $(n \times 1)$ cell-Variable `WEB_BOUNDARY` (hier $n = 2$, also für jeden Rand ein struct-Array). Die Punkteinträge setzen sich aus den gewichteten x - und y -Werten und dem Gewicht selbst zusammen.

Die Romberg-Integration muss dabei über alle Ränder und Segmente derselben erfolgen. Ist ein Punkt als randfern markiert (also nicht markiert), so wird einfach nacheinander über die Ränder und darin in einer Schleife über die Segmente einzeln integriert und danach alle Teilsummen aufaddiert.

Im vorangegangenen Abschnitt mussten schon Abstände zur Bestimmung der kritischen Matrixeinträge berechnet werden. Dabei wurde jeweils auch mitgespeichert, in welchem Segment und auf welchem Rand das jeweilige Minimum liegt. Diese Information können wir weiterverwenden. Ist ein Wert markiert, so kann zunächst über die ungefährlichen Segmente und alle Ränder integriert werden. Danach erfolgt die Integration über das kritische Segment durch eine iterierte Schachtelung. Dabei wird

jeweils ein Intervall, welches das Minimum enthält ausgespart, und nur über den Rest integriert. Mit dem ausgesparten Intervall verfährt man dann analog. Unterschreitet der in einem Schritt berechnete Wert die vorgegebene Genauigkeit, so bricht die Iteration ab, und die Teilsummen werden wieder wie gehabt aufaddiert.

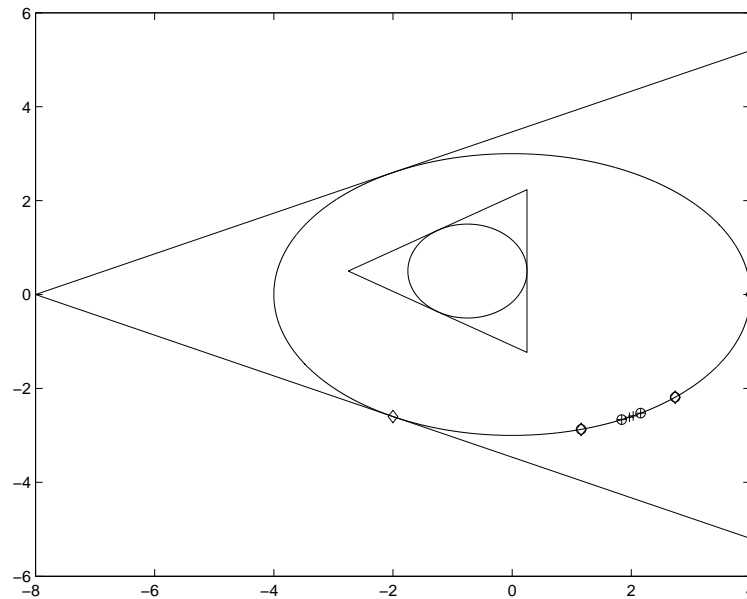


Abbildung 4.5: Gebiet mit konvexer Hülle. Die Schachtelung ist schematisch dargestellt (hier: Unterteilung im Verhältnis 2 : 1 : 2). Der erste Schritt ist durch \diamond der zweite durch \circ , der dritte durch $+$ dargestellt.

Die grosse Schwierigkeit hierbei bestand in der Behandlung von Spezialfällen, wie z.B.:

- Ein kritischer Punkt liegt direkt auf dem Rand bzw. der Abstand zum Rand ist kleiner als die gewählte Rechengenauigkeit. Dieser Fall muss abgefangen werden, da sonst die Integration nicht konvergiert (zur Erinnerung: wir berechnen $w^{-1} = \frac{ds}{\int \|X - P(s)\|^2}$).
- Der kritische Matrixeintrag befindet sich direkt auf einem Segmentrand.

Der Romberg-Algorithmus

Die Grundlage für den Romberg-Algorithmus bildet die Trapezregel zur Approximation von Integralen. Dabei wird der zu integrierende Bereich $[a, b]$ unterteilt in n äquidistante Stücke und die durch das stückweise lineare Interpolationspolynom aufgespannten Trapeze aufaddiert:

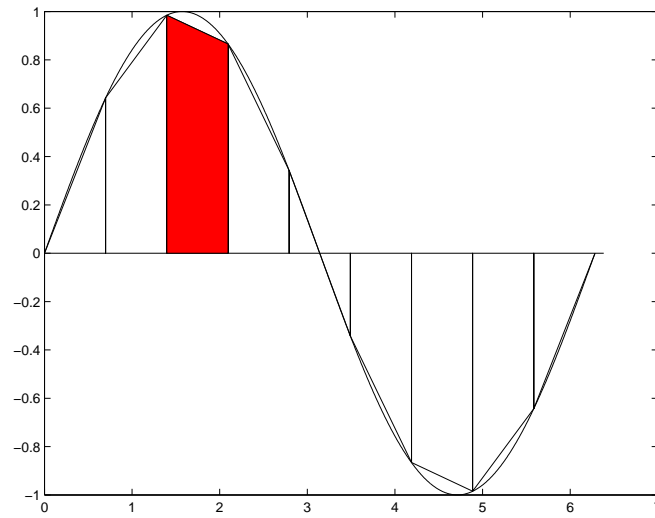


Abbildung 4.6: Trapezregel

$$\mathcal{S}f \approx \mathcal{S}(h, f, [a, b]) = h \left(\frac{1}{2}f(a) + f(a+h) + \cdots + \frac{1}{2}f(b) \right)$$

$$h = \frac{b-a}{n}$$

Theorem 4.3.1 (asymptotische Fehlerentwicklung der Trapezregel). Ist f glatt, so gilt für den Fehler der Trapezregel (Euler-McLaurin-Summenformel):

$$\begin{aligned} \Delta\mathcal{S}(h, t, [a, b]) &= \mathcal{S}f - \mathcal{S}(h, f, [a, b]) \\ &= \sum_{j=1}^{m-1} c_{2j} \left(f^{(2j-1)}(b) - f^{(2j-1)}(a) \right) h^{2j} + c_{2m} f^{(2m)}(u)(b-a) h^{2m} \end{aligned} \quad (4.18)$$

u sei dabei aus $[a, b]$ und die von f unabhängigen Konstanten c_i sind über die Bernoulli-Polynome definiert.

Für nicht-periodische Funktionen hat das Verfahren die Fehlerordnung $\mathcal{O}(h^2)$. Mit (4.18) können wir die Integralapproximation auch schreiben als:

$$\mathcal{S}(h, f, [a, b]) = \mathcal{S}f + f_1 h^2 + f h^4 + \dots$$

mit $f_j = -c_{2j} \left(f^{(2j-1)}(b) - f^{(2j-1)}(a) \right)$

Und mittels Extrapolation können wir eine verbesserte Approximation berechnen:

$$\begin{aligned}\mathcal{S}_2(h) &= \frac{4}{4-1}(4 - \mathcal{S}_1(h) - \mathcal{S}_1(2h)) \\ &= \frac{1}{3}(4(\mathcal{S}f + f_1 h^2 + f_2 h^4 + \dots) - (\mathcal{S}f + f_1(2h)^2 + f_2(2h)^4 + \dots)) \\ &= \mathcal{S}f - \frac{12}{3}f_2 h^4\end{aligned}$$

Ist f hinreichend glatt, so kann diese Iteration weitergeführt werden

$$\begin{aligned}\mathcal{S}_{i+1}(h) &= \frac{4^i \mathcal{S}_i(h) - \mathcal{S}_i(2h)}{4^i - 1} \\ &= \mathcal{S}f + \mathcal{O}(h^{2i+2})\end{aligned}$$

und die dominantesten Fehlerterme verschwinden bei jedem Schritt.

Der Romberg-Algorithmus verwendet nun genau diese Iteration. Er berechnet Schritt für Schritt Näherungen mit halbiertem Weite h mittels Trapezregel und führt nach jeder Halbierung die soeben beschriebene Extrapolation durch. Praktischerweise können zuvor schon berechnete Funktionswerte der Trapezregel wieder verwendet werden, denn durch die Halbierung von h treten diese in der folgenden Rechnung zwangsweise wieder auf. Die Approximation erfolgt, wie man der Iterationsvorschrift entnehmen kann nach einem Dreiecksschema. Das heißt also:

Will man \mathcal{S}_{j+1} berechnen und sind $\mathcal{S}_1(h_j), \dots, \mathcal{S}_j(h_j)$ ($h_j = 2^{1-j}h$) bekannt, dann muss zunächst $\mathcal{S}_1(h_{j+1})$ mittels Trapezregel berechnet werden; die Terme $\mathcal{S}_i(h_{j+1})$ $i = 2 : j+1$ folgen durch Extrapolation.

Umsetzung

```

        Programm 2: Integration und glatte Fortsetzung des Gewichts
for  $i = y - \text{Richtung}$  to end do
    for  $j = x - \text{Richtung}$  to end do
        if  $m < 1$  then
            S=web_romberg_invers(Punkt,Gebiet)
            % Romberg-Integration über alle Kurven
             $G(x, y, z) = \frac{1}{S}$ 
        else
            sneu[n,S]=web_weight_qifc_schachtel(Gebiet,Punkt)
            % Geschachtelter Romberg
            if  $n == 2$  then
                 $G(x, y, z) = 0$ 
            else
                 $G(x, y, z) = \frac{1}{S}$ 
            end if
        end if
    end for
end for
    % Glatte Fortsetzung der Werte nach außen
    inout=web_in_out(G)
     $G(3 : 4 : \text{end}, :) = \text{inout} * G(3 : 4 : \text{end}, :)$ 

```

Das Programm web_in_out ist Teil eines von **Jörg Hörner** geschriebenen Programms.

 Programm 3: Romberg-Integration (web_romberg_invdif)

```

if Aufruf ohne Segment- oder Kurvenangabe then
  for  $i = 1$  to Kurvenanzahl do
    for  $j = 1$  to Segmentanzahl do
      romberg(Inverse( $y$ ))
    end for
  end for
else
  romberg(Inverse(Kurve,Segment, $a,b$ ))
   $a, b$  Start und Endpunkt
end if

```

Ergebnis

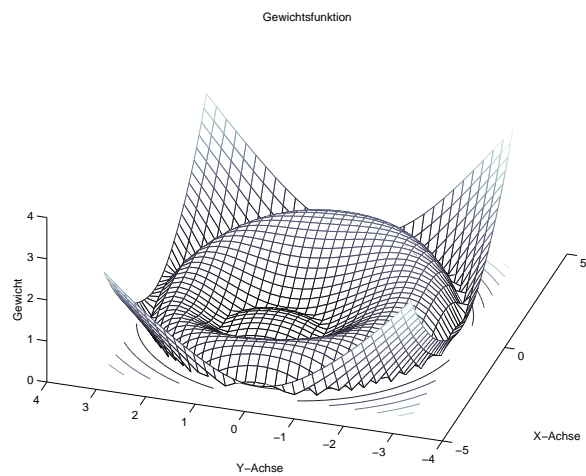


Abbildung 4.7: Funktionswerte der strikt positiven Gewichtsfunktion (auch außerhalb des Gebiets!)

Programm 4: Steuerung für geschachtelten Romberg (web_weight_qifc_schachtel)

Unterteilung des Integrationsintervalles in $\frac{1}{h}$ Teile (x_i)

$hmin = \min(Punkt - x_i)$

if $hmin < \frac{Genauigkeit}{10}$ **then**

$sneu(2) = [0, 2]$

% keine Berechnung, falls zu nahe am Rand

else

$sneu(2) = 1$

for $i = 1$ to Anzahl Kurven **do**

for $j \setminus$ kritisches Segment **do**

romberg_inv_dif(Gebiet,Kurve,Segment)

end for

end for

while Abbruchkriterium $<$ Genauigkeit **do**

$x = find(hmin \leq hmin)$

if min oder max(x) \neq Randpunkte des Segments **then**

neu=romberg(Gebiet,Kurve,kritisches Segment, $a:\min(x-h), \max(x+h) : b$)

else

neu=romberg(Gebiet,Kurve,kritisches Segment, $a+h : b$) oder

neu=romberg(Gebiet,Kurve,kritisches Segment, $a : b-h$)

end if

neues Minimum des kritischen Intervalles bestimmen

$S=S+neu$

end while

$sneu = [S, 1]$

end if

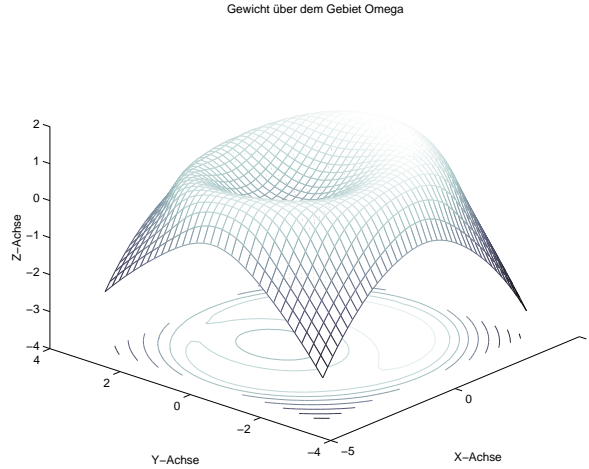


Abbildung 4.8: Glatte Forsetzung der Gewichtsfunktion nach außen durch Multiplikation mit der *in_out*-Matrix

4.3.3 Quasi-Interpolation der diskreten Punkte

Mit der hier beschriebenen Methode ist es nicht möglich, in einem vernünftigen Zeitrahmen eine für FEM-Methoden interessante Anzahl an Auswertungspunkten zu berechnen. Das macht aber nichts, denn wir sind nicht darauf angewiesen, dass die Gewichtsfunktion genau die bisher definierte Form exakt wiedergibt (dies ist nur für den Rand interessant). Vielmehr genügt es, die Gewichtsfunktion durch einen leicht zu berechnenden Quasi-Interpolanten zu approximieren.

Quasi-Interpolation

Quasi-Interpolation ist eine Approximationsmethode, die im Gegensatz zur Interpolation keine numerische Berechnung eines linearen Gleichungssystems erfordert.

Definition 4.3.2. Sei $s_{n,T}$ der Raum der Splines der Ordnung n über der nichtentarteten Knotenfolge T . Ein Quasiinterpolant ist eine beschränkte lineare Abbildung, die vom Raum der stetigen Funktionen über den Bereich $\mathcal{D}(T)$ in den Raum der Splines $S_{n,T}(\mathcal{D}(T))$ der Ordnung n übergeht. Reproduziert ein Quasi-Interpolant Polynome der Ordnung ν , dann bezeichnet man ihn als Quasi-Interpolanten der Ordnung ν :

$$Qg = \sum_j b_j^n(Q_j g), \quad Qp = p \quad \forall \quad p \in P_\nu.$$

Die Kontrollpunkte $g_j = Q_j g$ sind durch eine Folge linearer Funktionale Q_j definiert, die nur von der Einschränkung der Funktion g auf das Trägerintervall $s_j^n = \text{supp } b_j^n := [\tau_j, \tau_{j+n}]$ abhängt:

$$Q_j g = q_j g|_{s_j^n}$$

Die einfachste Möglichkeit eines solchen Q_j wäre ein Punkt-Funktional, also $Q_j g = g(t_j)$:

$$Qg = \sum_j g(t_j) b_j^n$$

Wobei $t_i \in [a, b] \cap (x_i, \dots, x_{i+m+1})$ mit der typischen Wahl:

$$t_i = \begin{cases} \frac{1}{2}(x_i + x_{i+m+1}) & \text{falls } \frac{1}{2}(x_i + x_{i+m+1}) \in [a, b] \\ a & \text{falls } \frac{1}{2}(x_i + x_{i+m+1}) < a \\ b & \text{falls } \frac{1}{2}(x_i + x_{i+m+1}) > b. \end{cases}$$

Nach [15] beträgt der Approximationsfehler für dieses Vorgehen $\|f - Q(f)\|_\infty \leq \frac{1}{2}(m+1)\delta \|f'\|_\infty$, wobei mit $Q(f)$ die Approximation, also der Quasi-Interpolant gemeint ist.

Durch eine geschicktere Wahl der Auswertungspunkte t_i kann das Ergebnis aber noch deutlich verbessert werden:

Seien μ_i die Greville Abszissen der Knotenfolge T (das Mittel der inneren Knoten)

$$\mu_j = \tau_j^* = \frac{(\tau_{j+1} + \dots + \tau_{j+m-1})}{(n-1)}$$

Dann hat der sogenannte *Schönberg*-Quasi-Interpolant die Form:

$$\begin{aligned} Q_j g &= g(\mu_j) \\ \Rightarrow Qg &= \sum_j Q_j g(\mu_j) \\ &= \sum_j b_j(t) g(\mu_j) \end{aligned}$$

Dieser Quasi-Interpolant hat die Ordnung 2 (exakt für lineare Polynome $p(t) = at + b$), aufgrund der linearen Präzision der Greville-Abszissen ($t = \sum_j b_j(t) \tau_j^*$).

$$\begin{aligned} Qp(t)(t) &= \sum_j b_j^n(a\mu_j + b) \\ &= a \sum_j b_j^n \mu_j + b \sum_j b_j^n \\ &= at + b \end{aligned}$$

Der Schönberg-Quasi-Interpolant, welcher auch in unserem Fall Anwendung finden soll, ist linear und bezüglich der ∞ -Norm beschränkt, denn

$$\begin{aligned}\|Q\|_\infty &:= \sup_{g \neq 0, g \in \mathcal{C}(D(T))} \frac{\|Qg\|_\infty}{\|g\|_\infty} \\ \|Qg\|_\infty &= \left\| \sum_j b_j^n g(\mu_j) \right\|_\infty \\ &\leq \max |g(\mu_j)| \leq \|g\|_\infty \\ \rightarrow \|Q\|_\infty &= \sup \frac{\|Qg\|_\infty}{\|g\|_\infty} \leq 1\end{aligned}$$

Für $g \equiv 1$ gilt sogar Gleichheit (B-Splines bilden eine Partition der Eins), also $\|Q\|_\infty = 1$.

Zur Bestimmung des Approximationsfehlers für den Schönberg-Quasi-Interpolanten betrachtet man dessen lineares Taylorpolynom. Für $t_0 \in [\tau_k, \tau_{k+1})$ und die Taylorentwicklung zweiter Ordnung folgt:

$$\begin{aligned}p(t) &= g(t_0) + g'(t_0)(t - t_0) \\ p(t) &= Qp(t) \quad \text{lineare Präzision} \\ \Rightarrow p(t_0) &= g(t_0)\end{aligned}$$

Approximationsfehler:

$$\begin{aligned}\Delta Q &= |g(t_0) - Qg(t_0)| \\ |Qp(t_0) - Qg(t_0)| &= |Q(p - g)(t_0)| \\ &= \sum_{j=k-m+1}^k b_j^n(t_0)(p(\mu_j) - g(\mu_j))\end{aligned}$$

$h(t) := [\tau_{k-n+1}, \tau_{k+n})$ ist der Träger all der B-Splines, die im Punkt $t \in [\tau_k, \tau_{k+1})$ nicht verschwinden; $|h(t)|$ sei die lokale Feinheit der Knotenfolge im Punkt t (also die Länge des t enthaltenden Intervalles). Da p die Entwicklung der Funktion g um den Punkt t_0 war, gilt:

$$\begin{aligned}|p(\mu_j) - g(\mu_j)| &= R_2(\mu_j) \\ &= \left| \frac{g''(t_0 + \theta(\mu_j - t_0))}{2} (\mu_j - t_0)^2 \right| \quad 0 < \theta < 1 \\ |p(\mu_j) - g(\mu_j)| &\leq \frac{1}{2} \|g''\|_{h(t_0)} |\mu_j - t_0|^2\end{aligned}$$

$\|\cdot\|_{h(t_0)}$ sei dabei eine „lokale“, also auf das t_0 enthaltende Intervall eingeschränkte Supremumsnorm.

Da $|\mu_j - t_0| \leq |h(t_0)|$, folgt für den Approximationsfehler des Quasi-Interpolanten:

$$|g(t) - Qg(t)| \leq \frac{1}{2} \|g''\|_{h(t)} |h(t)|^2 \quad (4.19)$$

Bei der Erzeugung von Quasi-Interpolanten höherer Ordnung hilft ein Blick auf die Marsden-Identität und ihre Ableitungen:

$$\begin{aligned} (t - \tau)^{n-1} &= \sum_j \psi_{j,n}(\tau) b_j^n(t) \\ \frac{\partial^{n-\nu}}{\partial \tau} (t - \tau)^{n-1} &= (-1)^{n-\nu} n(n-1) \dots \nu (t - \tau)^{\nu-1} \\ &= \sum_j b_j^n(t) \frac{\partial^{n-\nu}}{\partial \tau} \psi_{j,n}(\tau) \\ &= \sum_j b_j^n(t) \psi_{j,n}^\nu(\tau) \end{aligned}$$

Mit

$$\begin{aligned} \psi_{j,n}^\nu(\tau) &= (-1)^{n-\nu} \frac{(\nu-1)!}{(n-1)!} \partial^{n-\nu} \psi_{j,n}(\tau) \\ \psi_{j,n}(\tau) &= (u_{j+1} - \tau) \dots (u_{j+n} - \tau). \end{aligned}$$

Wenn wir wollen, dass der Quasi-Interpolant auch Polynome höherer Ordnung reproduziert, dann muss gelten

$$\begin{aligned} Q((t - \tau)^{\nu-1}) &\stackrel{!}{=} (t - \tau)^{\nu-1} \\ \Leftrightarrow \sum_j b_j^n(t) Q_j((t - \tau)^{\nu-1}) &= \sum_j b_j^n(t) \psi_{n,j}^\nu(\tau) \\ \Leftrightarrow Q_j((t - \tau)^{\nu-1}) &= \psi_{n,j}^\nu(\tau) \end{aligned}$$

$\psi_{n,j}^\nu(\tau)$ lässt sich auffassen als Polynom der Ordnung ν , abhängig von τ , und ist damit zerlegbar in ν linear unabhängige Polynome der Ordnung ν . Seien $t_{j,l}$ paarweise verschieden. Dann sei

$$\begin{aligned} \psi_{n,j}^\nu(t_{j,l}) &= \sum_{k=1}^{\nu} Q_{j,k}(t_{j,k} - t_{j,l})^{\nu-1} \quad l = 1 : \nu \\ &:= Q_j(\cdot - t_{j,l}) \end{aligned}$$

Da die $t_{i,j}$ paarweise verschieden sind, ist das daraus entstehende LGS eindeutig lösbar. Mit der Lösung der $Q_{j,k}$ erhalten wir auch eine Konstruktionsanleitung für $Q_j g$. Bei gleicher Wahl der Knotenpunkte $t_{j,k}$ wie für das Polynom der Ordnung ν gilt nun

$$Q_j g := \sum Q_{j,k} g(t_{j,k}) \quad t_{j,k} \in s_j^n.$$

Die Operatornorm des Quasi-Interpolanten (bezüglich der Supremumsnorm) erhalten wir mit:

$$\begin{aligned} |Q_j g| &= \left| \sum_k Q_{j,k} g(t_{j,k}) \right| \\ &\leq \sum_k |Q_{j,k}| |g(t_{j,k})| \\ &\leq \|g\| \sum_k |Q_{j,k}| \\ \Rightarrow \|Q_j\| &\leq \sum_k |Q_{j,k}| \\ \text{genauer } \|Q_j\|_\infty &= \sum_k |Q_{j,k}| \\ \Rightarrow \|Q_j g\|_\infty &= \left\| \sum_j b_j^n Q_j g \right\|_\infty \\ &\leq \max \|Q_j\|_\infty \|g\|_\infty \\ \Rightarrow \|Q\|_\infty &\leq \max_j \sum_k |Q_{j,k}| \end{aligned}$$

Der Fehler für diesen Quasi-Interpolanten kann analog (4.19) berechnet werden. Also

$$|g(t) - Qg(t)| \leq \frac{1}{(\nu)!} \|Q\| \|\partial^\nu g\|_{h(t)} |h(t)|^\nu$$

Der Fehler wird erheblich durch die Wahl der Stützstellen $t_{j,k}$ beeinflusst (genauer gesagt $\|Q\|$).

Eine häufig verwendete Wahl der Stützstellen sind die Intervallmittelpunkte

$$t_{j,k} = \frac{j + k - \frac{1}{2}}{h}, \quad k = 1 : \nu$$

Umsetzung

Die Auswertung des Quasi-Interpolanten ist nach Berechnung der Koeffizienten der einzelnen Basisfunktionen nur noch eine gewöhnliche Spline-Auswertung. Ein von **Jörg Hörner** für diesen Zweck geschriebenes Programm (`spl.eval`) konnte dafür komplett übernommen werden. Der struct-Array `spl` enthält Angaben über den Grad, die Koeffizienten und die Auswertungsdimension. `X`, `Y` sind die Punkte, an welchen die Auswertung erfolgen soll.

Programm 5: Quasi-Interpolation und Berechnung der Gewichtsfunction an vorgegebenen Werten, In-Out-Test

„Berechnung“ der Koeffizienten für den Schönberg-Quasiinterpolanten

$$G_{neu}(x - Werte) = G(x - Werte + 2 * h)$$

$$G_{neu}(y - Werte) = G(y - Werte + 2 * h)$$

Erweitere G_{neu} um nötige Randpunkte zur Splineauswertung (Knotenverdopplung).

`spl.eval(spl,X,Y)`

`End(G <= 0) = 0`

Anschließend wird das Ergebnis in eine globale Variable geschrieben, und kann somit von anderen Programmen des web-Projekts weiterverwendet werden.

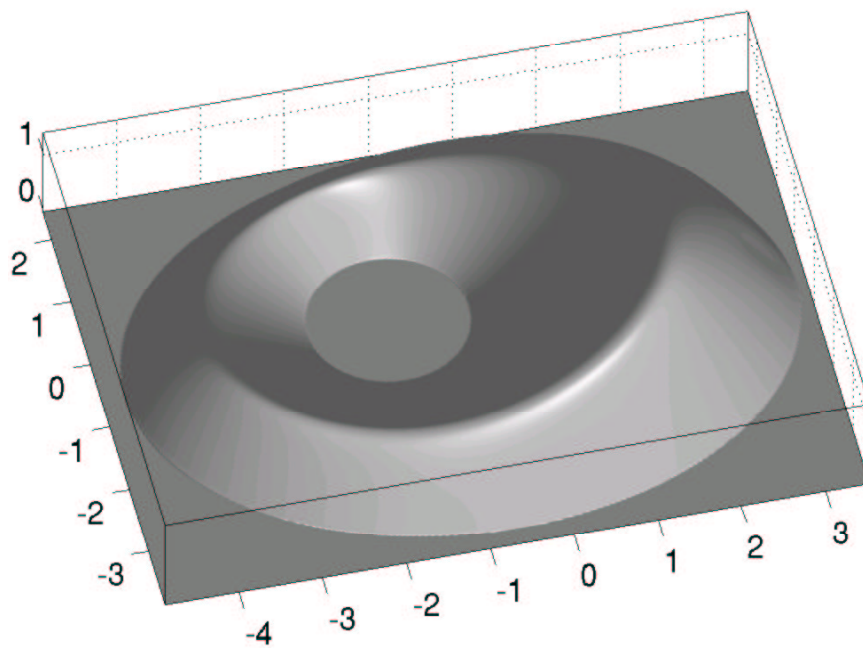
Ergebnis

Abbildung 4.9: Endgültige Version der Gewichtsfunktion, ausgewertet mittels Quasi-Interpolant



Literaturverzeichnis

- [1] D. Braess, *Finite Elemente*, Springer Verlag Berlin Heidelberg, 1997
- [2] S.C. Brenner, L.R. Scott, *The Mathematical Theory of Finite Element Methods*, Springer Verlag New York Inc., 1994
- [3] I.N. Bronstein, K.A. Semendjajew, *Teubner-Taschenbuch der Mathematik*, Teubner Verlag Stuttgart, 1996
- [4] Philippe G. Ciarlet, *The finite element method for elliptic problems*, North-Holland Publishing Company Amsterdam, 1980
- [5] Peter Deuflhard, Andreas Hohmann, *Numerische Mathematik: eine algorithmisch orientierte Einführung*, Walter de Gruyter & Co. Berlin, 1993
- [6] J. Elstrodt, *Maß- und Integrationstheorie*, Springer Verlag Berlin Heidelberg, 1996
- [7] Günther Hämmerlin, Karl-Heinz Hoffmann, *Numerische Mathematik*, Springer Verlag Berlin, 1989
- [8] Duane Hanselman, Bruce Littlefield, *Mastering Matlab. A Comprehensive Tutorial and Reference*, Prentice-Hall, New Jersey 1996
- [9] H. Heuser, *Lehrbuch der Analysis Teil 1*, B. G. Teubner Verlag Stuttgart, 1993
- [10] H. Heuser, *Lehrbuch der Analysis Teil 2*, B. G. Teubner Verlag Stuttgart, 1995
- [11] K. Höllig, U. Reif, J. Wipper, *Weighted extended b-spline approximation of Dirichlet problems*, Preprint 2000-8 Universität Stuttgart, 2000
- [12] K. Höllig, U. Reif, J. Wipper *Error Estimates for the web-Method*, Preprint 2000-16 Universität Stuttgart, 2000
- [13] Klaus Höllig *Grundlagen der Numerik*, MathText, Zavelstein, 1998
- [14] L.W. Kantorowitsch, W.I. Krylow, *Näherungsmethoden der höheren Analysis*, VEB Deutscher Verlag der Wissenschaften Berlin, 1956
- [15] Günther Nürnberger, *Approximation by Spline Functions*, Springer-Verlag Berlin 1989

- [16] V.L. Rvachev, T.I. Sheiko, V. Shapiro, I. Tsukanov, *On completeness of RFM solution structures*, Computational Mechanics 25 (2000) 305-316, Springer Verlag
- [17] Schumaker, *Spline Functions: Basic Theory*, John Wiley & Sons New York, 1981
- [18] H.R. Schwarz, *Numerische Mathematik*, Teubner Verlag Stuttgart, 1997
- [19] Ch. Schwab, *p- and hp-Finite Element Methods. Theory and Applications in Solid and Fluid Mechanics*, Clarendon Press Oxford, 1998
- [20] Stoer, *Numerische Mathematik 1*, Springer Verlag Berlin Heidelberg, 1994
- [21] Stoer, Bulirsch, *Numerische Mathematik 2*, Springer Verlag Berlin Heidelberg, 1990
- [22] G. Strang, G.J. Fix, *An analysis of the finite element method*, Prentice-Hall, 1973
- [23] D. Werner, *Funktionalanalysis*, Springer Verlag Berlin, 1997
- [24] Peter Williams, *Algorithms*, 1996
- [25] O. Zienkiewicz and R.L. Taylor, *The Finite Element Method. Fourth Edition Volume 1. Basic Formulations and Linear Problems*, McGraw-Hill International(UK), 1994

Erklärung

Hiermit erkläre ich, **Winfried Geis**, Matrikelnummer 173 169 5, die vorliegende Diplomarbeit gemäß den Bedingungen der Diplomprüfungsordnung der **mathematischen Fakultät der Universität Stuttgart** selbständig verfasst zu haben. Wo ich Hilfen, Leistungen und Ergebnisse anderer verwendet habe, ist dies kenntlich gemacht.

Stuttgart, den 11. November 2001

Winfried Geis

Danksagung

Zunächst möchte ich mich ganz herzlich bei Herrn Prof. Dr. Klaus Hölzig zum einen für das spannende Thema zum anderen für sein Engagement bei Betreuung der Arbeit bedanken.

Bedanken möchte ich mich auch bei allen Mitarbeitern und Diplomanden des zweiten Lehrstuhles für die angenehme und freundschaftliche Atmosphäre. Ganz besonderer Dank gilt dabei Jörg Hörner und Joachim Wipper, die sich für mich immer Zeit genommen, und damit nicht unwesentlich zum Gelingen dieser Arbeit beigetragen haben.

Mein Dank gilt auch Thomas Merkle, Florian Haag und dem sechsten Lehrstuhl für ihr offenes Ohr, ihren Ratschlag und ihre Diskussionsbereitschaft.

Ganz besonders danken möchte ich meinen Eltern, ohne deren weitreichende Unterstützung mein Studium so nicht möglich gewesen wäre: Danke!