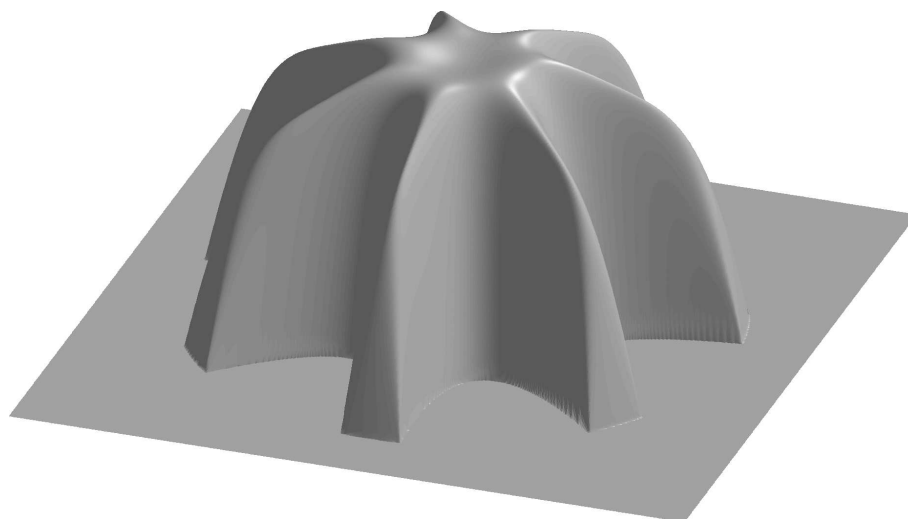


DIPLOMARBEIT

\mathcal{R} -FUNKTIONEN FÜR
FINITE ELEMENTE APPROXIMATIONEN
MIT WEB-SPLINES

Marco Boßle

26. September 2002



Betreuer: Prof. Dr. K. Höllig
2. Lehrstuhl
Mathematisches Institut A
Universität Stuttgart

Danksagung

All denjenigen, die zum Gelingen dieser Arbeit auf die unterschiedlichste Art und Weise beigetragen haben, gilt mein Dank. Insbesondere möchte ich Herrn Prof. Dr. Klaus Höllich für seine Anregung zu diesem Thema und seine unermüdliche und wertvolle Unterstützung danken. Auch den Beteiligten am web-Projekt bin ich für angeregte Diskussionen, konstruktive Kritik und produktive Vorschläge sehr dankbar. Insbesondere wären hier die Assistenten des 2. Lehrstuhls Dipl.-math. Joachim Wipper, Dipl.-math. Jörg Hörner und Dipl.-math. Christian Apprich zu nennen, aber auch Anja Streit und Andreas Kopf, die mir bei vielen kleinen Entscheidungen und Problemen geholfen haben.

Mein Dank gilt natürlich auch meiner Familie, die mich stets ohne Einschränkungen gefördert hat. Ein letzter, besonders grosser Dank geht an meine Partnerin Britta Meier für die grenzenlose Unterstützung die ich tagtäglich durch sie erfahre.

Erklärung

Hiermit versichere ich, dass ich diese Diplomarbeit selbstständig und nur unter Verwendung der im Literaturverzeichnis angegebenen Hilfsmittel angefertigt habe.

Stuttgart, den 26. September 2002

Marco Boßle

Inhaltsverzeichnis

Einleitung	7
1 Die web-Methode	9
1.1 Modellproblem	9
1.2 web-Methode	10
1.2.1 Splines	10
1.2.2 Duale Funktionale	12
1.2.3 Gewichtung und Erweiterung	12
1.3 Stabilität	16
2 Gewichts- und \mathcal{R}-Funktionen	19
2.1 Definition der Gewichtsfunktionen	19
2.2 Definition der \mathcal{R} -Funktionen	19
2.3 Eigenschaften verschiedener \mathcal{R} -Funktionen	24
2.3.1 Funktions-System \mathcal{R}_1	24
2.3.2 Funktions-System \mathcal{R}_0	25
2.3.3 Funktions-System \mathcal{R}_α	26
2.3.4 Funktions-System \mathcal{R}_0^m	29
2.3.5 Funktions-System \mathcal{R}_p	30
2.3.6 Funktions-System \mathcal{R}_C^m	33
3 web-Eigenschaften der Gewichtsfunktionen	39
3.1 Äquivalenz zur Abstandsfunktion	39
3.2 Beschränktheit des Gradienten	41
3.3 Zusammenfassung aller Eigenschaften	43
4 Grundlagen für die Implementierung	45
4.1 Quadriken und Gewichtsfunktionen	45
4.1.1 Darstellung der Quadriken	45
4.1.2 Bestimmung der definierenden Matrizen allgemeiner Quadriken	46
4.1.3 Grundformen der Quadriken und notwendige Transformationen	49
4.1.4 Gewichtsfunktion der Quadriken	50
4.1.5 Kombination der Quadriken	50
4.2 Berechnung der Randkurve	51
4.2.1 Definition der Bézierkurven	51
4.2.2 Darstellung von Quadriken mittels Bézierkurven	52
4.2.3 Bézierkurven zu Quadriken in Normalform	55
4.2.4 Schnitt von Bézierkurven	59

4.2.5	Randkurve eines kombinierten Gebietes	60
5	Umsetzung der Grundlagen	63
5.1	Implementierung der Gewichtsfunktion	63
5.2	Implementierung der Objekte	64
5.3	Implementierung der Berechnung der Randkurven	67
5.4	Der QBE	68
5.4.1	Das Hauptfenster	69
5.4.2	Das Menü Datei	69
5.4.3	Die Menüs Vereinigung, Schnitt und Differenz	70
5.4.4	Das Menü Gewichtsfunktion	70
6	Beispiele	71
6.1	Interaktives Beispiel	71
6.2	Beispiel ohne Interaktion	73
6.3	Lösung des Poisson-Problems	74
6.4	Rechenzeiten für \mathcal{R} -Funktionen	74
7	Visualisierung der Eigenschaften	77
7.1	Differenzierbarkeit in Ecken	77
7.2	Normalisiertheit	79
7.3	Sonderfälle	80
A	Kurzbeschreibung der Programme	83
A.1	Datenstrukturen	83
A.2	Programme	84

Einleitung

Die Methode der finiten Elemente ist in den Ingenieurwissenschaften und der angewandten Mathematik weit verbreitet und erfreut sich heute einer größeren Beliebtheit denn je. Sie ist die Methode, die bei der Berechnung numerischer Simulationen die breiteste Anerkennung findet. Für eine Chronologie der Finiten Elemente sei auf [Hö03] oder [Zie95] verwiesen. Doch auch heute noch ist dies ein Gebiet höchster Forschungsaktivität für Mathematiker und Ingenieure.

Diese Verfahren beruhen meist auf einer approximativen Diskretisierung des zu untersuchenden Gebietes und der anschließenden Definition von Ansatzfunktionen auf den Teilgebieten. Schon Courant [Cou43] führte Hutfunktionen ein, es folgten unter vielen Anderen Turner, Martin, Clough und Topp [TCMT94] sowie Argyris [Arg60].

Auch Splines haben seit den ersten grundlegenden Arbeiten von de Boor [dB72], Bézier [Béz66] und Casteljau [dC59] eine Entwicklung bis zum Standard in numerischer Approximation und Freiform-Design erfahren.

Höllig, Reif und Wipper ist es nun in [HRW01b] gelungen, B-Splines als Ansatzfunktionen zu verwenden. Sie konnten die entstehenden Probleme der Stabilität und Approximationsgüte überwinden. Eine der herausragenden Eigenschaften dieses web-Verfahrens ist die fehlende Notwendigkeit einer Vernetzung. Trotzdem kann das Gebiet exakt in das Verfahren einfließen, indem nicht die Träger der Finiten Elemente einer Vernetzung angepasst werden, sondern die Ansatzfunktionen selbst mit einer Gewichtsfunktion skaliert werden. Dieses Vorgehen wurde bereits von Kantorowitsch und Krylow [KK56] verwendet.

Die Anforderungen an eine solche Gewichtsfunktion sind natürlich enorm. Sie muss auf dem gesamten Rand verschwinden, im Innern des Gebietes positiv und außerhalb negativ sein. Desweiteren darf sie nicht die Differenziationseigenschaften der B-Splines verletzen.

Schon 1963 hat Rvachev in [Rva63] eine Reihe solcher Funktionen gefunden, welche die Berechnung von Gewichtsfunktionen auf booleschen Kombinationen von beliebigen Gebieten erlauben. Auch später war Rvachev auf dem Gebiet sehr aktiv, wie die weiteren grundlegenden Arbeiten [Rva67, Rva74, Rva82] und die Literaturliste in [RS95] zeigen. Eine komplette Liste der Veröffentlichungen bis 1987 kann in [Shi88] gefunden werden. Ein Grossteil dieser Werke wurde allerdings weitgehend unbeachtet der westlichen Welt in russisch veröffentlicht. Erst mit [Sha88] wurde eine englische Zusammenfassung der Ergebnisse Rvachevs präsentiert.

Diese Arbeit widmet sich nun der Analyse dieser \mathcal{R} -Funktionen. In Kapitel 1 wird die

web-Methode vorgestellt und die grundlegenden Anforderungen an eine Gewichtsfunktion ermittelt. Daraufhin folgt in Kapitel 2 eine allgemeine Untersuchung der verschiedenen Typen von \mathcal{R} -Funktionen und eine umfassende Darstellung der unterschiedlichen Eigenschaften. Kapitel 3 ist eine kurze Abhandlung der speziellen Eigenschaften, die für die web-Methode benötigt wird. Das darauffolgende Kapitel 4 legt die zusätzlich Anforderungen der web-Methode und deren Lösung dar: Der Rand der Gebiete muss dargestellt und bei der Kombinationen von Gebieten neu berechnet werden. Es werden hier mögliche Darstellungen und Algorithmen für die Randberechnung erläutert. Diese werden im nächsten Kapitel 5 soweit praktisch dargestellt, wie es für die Implementierung der \mathcal{R} -Funktionen notwendig war. Auch eine Beschreibung des Programmes, welches die Berechnung der Randkurven und Gewichtsfunktionen interaktiv vornimmt, kann hier gefunden werden. Das letzte Kapitel 6 enthält schließlich eine Visualisierung der Eigenschaften aller untersuchten \mathcal{R} -Funktionen.

Kapitel 1

Die web-Methode

Die web-Methode ist ein Verfahren der finiten Elemente, bei dem der Approximationsraum auf kardinalen B-Splines basiert. Auch wenn die Wahl von Splines als Basisfunktionen auf den ersten Blick sehr naheliegend erscheinen mag, so stellen sich bei näherer Betrachtung doch große Probleme dar.

Eine einfache Basis aus kardinalen Splines ist nicht stabil. Allerdings kann durch eine geeignete Anpassung der Basisfunktionen diese Stabilität erreicht werden. Diese Anpassung soll in diesem Kapitel erläutert werden.

Ausserdem lassen sich mit den quadratischen Tägern der B-Splines beliebige Gebiete nur sehr schlecht approximieren, was die Skalierung mit einer Gewichtsfunktion nötig macht. Die Anforderungen an diese Gewichtsfunktionen ergeben sich aus den gewünschten Approximationseigenschaften der web-Splines, die auch in diesem Kapitel beleuchtet werden.

Zunächst jedoch eine Definition, welche die Lesbarkeit dieses Kapitels verbessert.

Definition 1.1

Gegeben seien die Funktionen $f, g : K \rightarrow \mathbb{R}$. Dann gilt mit Konstanten $c_1, c_2 \in \mathbb{R}_+$

$$f \succeq g \quad :\Leftrightarrow \quad f(x) \leq c_1 g(x) \quad \forall x \in K,$$

$$f \preceq g \quad :\Leftrightarrow \quad f(x) \geq c_1 g(x) \quad \forall x \in K$$

und heißen genau dann äquivalent, wenn

$$f \asymp g \quad :\Leftrightarrow \quad (f(x) \leq c_1 g(x) \quad \text{und} \quad f(x) \geq c_2 g(x)).$$

Zwei Funktionen heißen also äquivalent, wenn sie gegenseitig als obere und untere Schranke benutzt werden können.

1.1 Modellproblem

Die web-Methode soll an dem folgenden Modellproblem erläutert werden. Es wird das *Poisson-Problem* mit *Dirichlet-Randbedingungen* auf einem beschränkten Gebiet $\Omega \in \mathbb{R}^m$ untersucht,

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{auf } \partial\Omega. \end{aligned} \tag{1.1}$$

Die Lösung wird durch eine Linearkombination

$$u_h = \sum_i a_i B_i$$

von Basisfunktionen B_i approximiert. Diese sollen auf $\partial\Omega$ verschwinden, was die Einhaltung der Randwertbedingungen impliziert. Die elliptische Bilinearform a und die Linearform ℓ des Poissonproblem sind allgemein bekannt und können beispielsweise in [Bra92] gefunden werden. Die Koeffizienten $A = \{a_i\}_i$ ergeben sich aus dem linearen *Galerkin-System*

$$\begin{aligned} a(B_k, u_h) &= \int_{\Omega} \partial_1 B_k \partial_1 u_h + \partial_2 B_k \partial_2 u_h \\ &= \int_{\Omega} \nabla B_k \nabla u_h \\ &= \sum_i \left(\int_{\Omega} \nabla B_k \nabla B_i \right) a_i \\ \ell(B_k) &= \int_{\Omega} B_k f \end{aligned}$$

also

$$\sum_i \left(\int_{\Omega} \nabla B_k \nabla B_i \right) a_i = \int_{\Omega} B_k f \quad (1.2)$$

welches im Folgenden durch $G_h A = F$ abgekürzt wird.

1.2 web-Methode

Die web-Methode basiert auf Splines, die nun definiert werden sollen.

1.2.1 Splines

Die hier präsentierten Definitionen stellen nur Spezialfälle allgemeiner Splines dar, die für die vorgestellten Verfahren allerdings vollkommen ausreichen. Für eine ausführliche Theorie über Splines und ähnliche geometrische Gebilde sei hier auf [HL89] oder [Far92] verwiesen.

Definition 1.2

Ein kardinaler B-Spline ist definiert durch die Rekursion

$$\tilde{b}^n(x) := \int_0^1 b^{n-1}(x-t) dt$$

mit

$$\tilde{b}^1(x) := \begin{cases} x & \text{für } 0 \leq x \leq 1 \\ 2-x & \text{für } 1 \leq x \leq 2 \\ 0 & \text{für sonst} \end{cases} .$$

Ein kardinaler Tensorprodukt B-Spline ist das Produkt kardinaler B-Splines

$$b^n(x) := b^{n_1}(x_1) \dots b^{n_d}(x_d).$$

Dessen skalierte Translate werden mit

$$b_{k,h}^n(x) := h^{-m/2} b^n(x/h - k), \quad k \in \mathbb{Z}^m$$

bezeichnet.

Schließlich werden die Gitterzellen als

$$G_\ell := h([0, 1]^m + \ell), \quad \ell \in \mathbb{Z}^m$$

definiert.

Der Raum \mathcal{S} heißt Splineraum und ist der Raum aller Linearkombinationen von B-Splines.

Da im web-Verfahren stets die skalierten Translate der B-Splines benötigt werden, wird im Weiteren b_k statt $b_{k,h}^n$ geschrieben. Der Parameter k gibt dabei die Verschiebung an, der Parameter h bestimmt die Größe der Gitterzelle und wird in Abschnitt 1.2.3 die Größe der Unterteilung des Gebietes Ω sein.

Bemerkung 1.3

Die B-Splines sind $(n-1)$ -mal stetig differenzierbar und sind Polynome des Grades n auf jeder Gitterzelle. Die Basisfunktionen haben einen beschränkten Träger. Ein d -dimensionaler Tensorprodukt B-Spline des Grades n hat einen Träger, der genau $(n+1)^d$ Gitterzellen umfasst.

Sei nun ein Gebiet Ω gegeben.

Definition 1.4

Die Menge der Indizes von B-Splines, deren Träger das Gebiet Ω überlappen, wird mit

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

bezeichnet.

Sei

$$L_k = \{\ell \in \mathbb{Z}^m : G_{k+\ell} \subset \text{supp } b_k \cap \Omega\}.$$

Ein B-Spline b_k heißt innerer B-Spline, wenn L_k nichtleer ist, sonst äußerer B-Spline. Die zugehörigen Teilmengen von K sind I und J ,

$$K = I \cup J.$$

Desweiteren wird für jeden inneren Spline eine Gitterzelle ausgewählt, die vollständig in Ω liegt. Die Mitte dieser Gitterzelle wird mit x_i bezeichnet, der Verschiebungsindex mit $\ell(i)$

$$x_i \in G_{i+\ell(i)}.$$

Damit werden die B-Splines nach ihren Trägern kategorisiert und erlauben somit eine Bestimmung der Splines mit einem geringen Anteil innerhalb des Gebietes. Diese sind für das unkontrollierbare Verhalten der Galerkinmatrix zuständig und müssen auf eine besondere Weise behandelt werden.

1.2.2 Duale Funktionale

Die Stabilität der Basis kann mit einem schönen Beweis gezeigt werden, der sogenannte *duale Funktionale* benutzt, die nun definiert werden sollen.

Satz 1.5

Für jedes $\ell \in \{0, \dots, n\}^m$ existiert eine stetige Funktion λ^ℓ , sodass

$$\text{supp}(\lambda^\ell) = [1/4, 3/4]^m + \ell$$

und

$$\int_{\mathbb{R}} b(x-k)\lambda^\ell = \delta_k^0.$$

Bemerkung 1.6

Die dualen Funktionale sind nicht eindeutig, insbesondere lassen sich durch andere Einschränkung des Trägers beliebige weitere finden. Die Funktionen

$$\lambda_i^\ell(x) = h^{-m/2}\lambda^\ell(x/h - i)$$

erfüllen

$$\text{supp}(\lambda_i^\ell) = h([1/4, 3/4]^m + i + \ell) =: G'_{i+\ell}$$

und

$$\int_{\mathbb{R}} b_k \lambda_i^\ell = \delta_k^i. \tag{1.3}$$

Da die B-Splines keine orthogonale Basis bilden, werden für die Beweise Funktionen benötigt, die zu ihnen orthogonal sind. Diese Eigenschaft erfüllen die dualen Funktionale.

Von ganz entscheidender Bedeutung sind die B-Splines und die zugehörigen dualen Funktionale, da sie beschränkt und zueinander orthogonal sind. Dieses Resultat wird im nächsten Abschnitt verwendet werden.

Satz 1.7

Für alle $k \in K$ und $i \in I$ sind die B-Splines b_k und die dualen Funktionale λ_i in Bezug auf die Gitterweite h beschränkt und laut 1.3 biorthogonal, d.h.

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

1.2.3 Gewichtung und Erweiterung

Da die oben definierten B-Splines stets einen quadratischen Träger haben, lassen sich mit ihnen beliebige Gebiete nur approximieren. Die Randlinie wird stets durch die Gitterzellen vorgegeben sein, was auf sehr schlechte Approximationen führt. Abbildung 1.1 zeigt einen B-Spline am Rande eines Gebietes und vermittelt einen Eindruck dieses Problems.

Eine elegante Methode ist die Multiplikation der Basisfunktionen mit einer Gewichtsfunktion (vgl. Kapitel 2.1). Dies stellt sicher, dass die finiten Elemente, mit denen approximiert wird, die Randbedingung exakt erfüllen, wie in Abbildung 1.2. Die Anforderungen an die

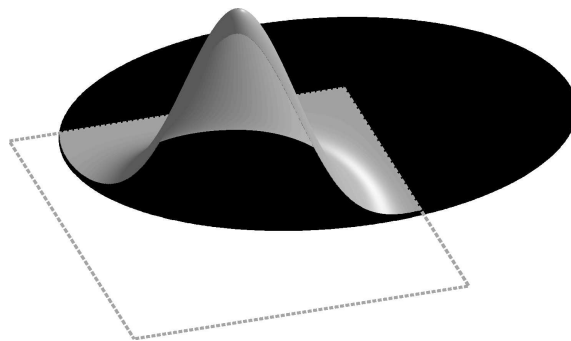


Abbildung 1.1: Ein B-Spline am Rande eines Gebietes

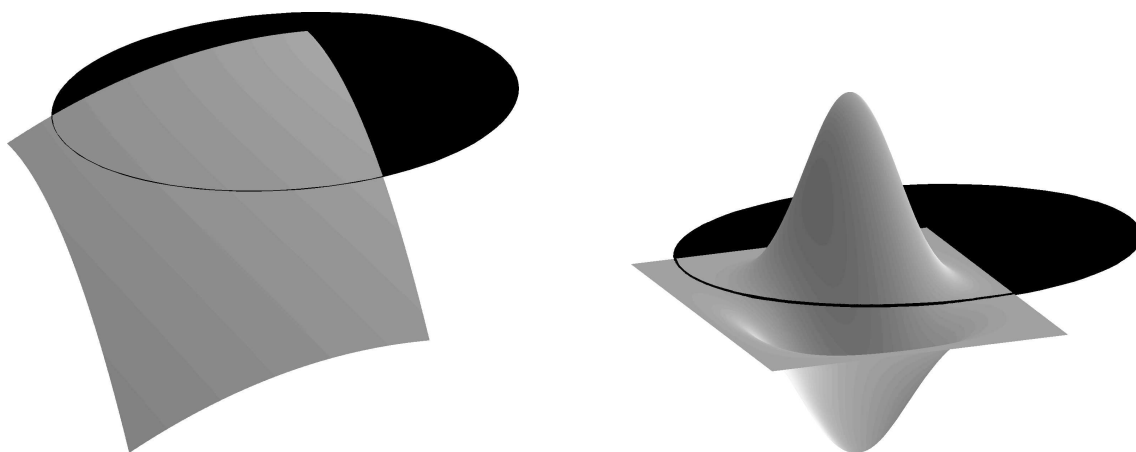


Abbildung 1.2: Gewichtsfunktion und gewichteter Spline

Eigenschaften der Gewichtsfunktion ergeben sich aus den Beweisen für die Stabilität der Basis (s. 1.3).

Einen schweren Rückschlag stellt allerdings die Tatsache dar, dass die gewichteten B-Splines (wb-Splines) auf Grund mangelnder Stabilität nicht verhindern können, dass die Galerkin-Matrix eine nicht zu kontrollierende Kondition erlangt. Dies liegt daran, dass die äußeren Splines einen extrem kleinen Anteil an dem Gebiet Ω haben können und somit sehr kleine Einträge in der Galerkin-Matrix liefern. Sie werden jedoch für die Approximationsordnung benötigt. Die Lösung dieses Problems wurde erstmals in [HRW01b] behandelt und soll an dieser Stelle kurz eingeführt werden.

Definition 1.8 (web-Spline)

Für $i \in I$ ist der gewichtete erweiterte B-Spline (web-Spline) B_i definiert durch

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

wobei x_i die Mitte der Gitterzelle $G_{i+\ell(i)}$ bezeichnet. Die Koeffizienten $e_{i,j}$ erfüllen

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

und sind so gewählt, dass alle gewichteten Polynome ωp mit Koordinatengrad $p \leq n$ im web-Raum $\mathbb{B}_h = \text{span}\{B_i : i \in I\}$ enthalten sind.

Die folgende Wahl der Koeffizienten $e_{i,j}$ erweist sich als geeignet.

Satz 1.9

Ist für alle $j \in J$

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

ein nächstgelegenes Indexarray zu j , dann erfüllen die Koeffizienten

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

die Bedingungen zur Konstruktion von web-Splines.

Der Beweis kann in [HRW01b] gefunden werden. Die Anzahl der Basisfunktionen wird also auf eine Weise eingeschränkt, die den Approximationsraum dennoch erhält. Äußere Splines werden an die inneren Splines gekoppelt, d.h. der Träger einer Basisfunktion enthält mindestens eine vollständige Gitterzelle. Die extrem kleinen Einträge in der Galerkin-Matrix können auf diese Weise also vermieden werden.

An dieser Stelle sei nochmals darauf hingewiesen, dass die Gewichtung der Basisfunktionen die Einhaltung der Randbedingung impliziert. Ohne die Gewichtung der B-Splines wäre dies nicht möglich!

Für den Stabilitätsbeweis, der im nächsten Abschnitt folgt, müssen noch die dualen Funktionale an die web-Splines angepasst werden.

Definition 1.10

Die dualen Funktionale der web-Splines lauten

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

Dies führt auf folgenden Satz, der für die Stabilität entscheidend ist.

Satz 1.11

Für $i, k \in I$ sind die web-Splines B_i und die dualen Funktionale Λ_k bezüglich der Gitterweite h gleichmäßig beschränkt und biorthogonal,

$$\|B_i\|_0 \leq 1, \quad \|\Lambda_k\|_0 \leq 1, \quad \int_{\Omega} B_i \Lambda_k = \delta_i^k.$$

Beweis:

Für die Gewichtsfunktion gilt laut (1.12) $\omega(x_i) \geq d(x_i, \partial\Omega) \geq h$, da x_i der Mittelpunkt einer ganz in Ω enthaltenen Gitterzelle ist. Zudem gilt für alle Punkte $x \in \text{supp}(B_i)$, dass $d(x, x_i) \leq h$ ist. Die Dreiecksungleichung liefert

$$\frac{d(x, \partial\Omega)}{d(x_i, \partial\Omega)} = 1 + \frac{d(x, \partial\Omega) - d(x_i, \partial\Omega)}{d(x_i, \partial\Omega)} \leq 1 + \frac{d(x, x_i)}{h} \leq 1.$$

Also gilt für den Gewichtungsfaktor der B-Splines

$$\frac{\omega(x)}{\omega(x_i)} \leq 1 + \frac{\omega(x) - \omega(x_i)}{\omega(x_i)} \leq 1,$$

woraus mit Hilfe von Satz 1.7 die Beschränktheit der B-Splines folgt:

$$\|B_i\|_0 \leq \|b_i\|_0 + \sum_{j \in J} |e_{i,j}| \|b_j\|_0 \leq 1.$$

Die Beschränktheit der dualen Funktionale ergibt sich aus den folgenden Beobachtungen. Zunächst wird x auf $\text{supp}(\Lambda_k) = G_{k+\ell(k)}$ eingeschränkt. Da G' den Durchmesser $h/2$ und den Mittelpunkt $x_{k+\ell(k)}$ besitzt, gilt $d(x, \partial\Omega) \geq h$. Also gilt auch $\omega(x_k)/\omega(x) \leq 1$, woraus wieder mit Satz 1.7 die Beschränktheit der dualen Funktionale folgt. Schließlich gilt mit $i, k \in I$ und Satz 1.7

$$\begin{aligned} \int B_i \Lambda_k &= \int \frac{\omega}{\omega(x_i)} \left(b_i - \sum_{j \in J} e_{i,j} b_j \right) \frac{\omega(x_k)}{\omega} \lambda_k \\ &= \frac{\omega(x_k)}{\omega(x_i)} \left(\int b_i \lambda_k - \sum_{j \in J} e_{i,j} \int b_j \lambda_k \right) \\ &= \frac{\omega(x_k)}{\omega(x_i)} \int b_i \lambda_k \\ &= \delta_k^i \end{aligned}$$

■

1.3 Stabilität

In diesem Abschnitt wird die Stabilität der Basis und die Beschränktheit der Galerkin-Matrix betrachtet. An dieser Stelle wird auch deutlich, dass an die Gewichtsfunktion ω außer den Eigenschaften der Definition 2.1 noch weitere Anforderungen gestellt werden müssen. An dieser Stelle werden nur diejenigen Beweise ausgeführt, in denen der Einfluss der besonderen Eigenschaften der Gewichtsfunktion zum Tragen kommt. Ausgeführte Beweise finden sich in [HRW01a]. Diese zusätzlichen Eigenschaften werden in Kapitel 3 ausführlich behandelt.

Definition 1.12 (web-Eigenschaften der Gewichtsfunktion)

Eine Gewichtsfunktion ω besitzt die web-Eigenschaften, falls sie zusätzlich zu den Eigenschaften der Definition 2.1 noch die Bedingungen

$$\forall x \in \Omega : \quad w(x) \asymp d(x, \partial\Omega) \tag{1.4}$$

und

$$\forall x \in \Omega : \quad |\nabla w(x)| \preceq 1 \tag{1.5}$$

erfüllt.

Die Ergebnissen des letzten Abschnitts lassen nun eine Stabilitätsaussage der Basis zu, die für die Gutartigkeit der Kondition der Galerkin-Matrix notwendig ist.

Satz 1.13 (Stabilität der Basis)

Eine web-Basis $\{B_i\}_{i \in I}$ nach Definition 1.8 ist bezüglich der L_2 -Norm stabil:

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

wobei $A = \{a_i\}_{i \in I}$.

Es gilt allerdings noch mehr. Auch eine Sobolev-Norm höherer Ordnung kann durch die Koeffizienten abgeschätzt werden, wie der folgende Satz zeigt.

Satz 1.14

Eine web-Basis $\{B_i\}_{i \in I}$ erfüllt

$$\left\| \sum_{i \in I} a_i B_i \right\|_1 \preceq h^{-1} \|A\| .$$

Beweis:

Eine beliebige Ableitung erster Ordnung DB_i ist nach der Produktregel die Summe zweier Ausdrücke,

$$DB_i = \left[\frac{D\omega}{\omega(x_i)} \left(b_i + \sum_{j \in J} e_{i,j} b_j \right) \right] + \left[\frac{\omega}{\omega(x_i)} \left(Db_i + \sum_{j \in J} e_{i,j} Db_j \right) \right] .$$

Diese Summe kann also nur beschränkt sein, falls (1.4) gilt. Der erste Summand ist folglich aufgrund $\omega(x_i)^{-1} \preceq h^{-1}$ und Definition 1.12 beschränkt, da der Mittelpunkt x_i der Gitterzelle den Gebietsrand nicht schneidet. Im zweiten Summanden liefert $Db_k(x) =$

$h^{-1}h^{-m/2}(Db)(x/h - k)$ ebenfalls den richtigen Exponenten von h . Die Summation der Schranken für B_i ergibt die zugehörige Abschätzung für beliebige Linearkombinationen.

■

Aus den Sätzen 1.13 und 1.14 kann nun ein entscheidender Satz hergeleitet werden, dessen Beweis wie alle anderen in [HRW01b] gefunden werden kann.

Satz 1.15

Die Konditionszahl der Galerkin-Matrix G_h des web-Raums \mathbb{B}_h bezüglich der web-Basis $\{B_i\}_{i \in I}$ ist beschränkt durch

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

An dieser Stelle soll nochmals darauf hingewiesen werden, dass das moderate Anwachsen der Kondition der Galerkin-Matrix allein von der erweiterten Basis abhängt. Ein vergleichbares Ergebnis existiert für wb-Splines nicht.

Kapitel 2

Gewichts- und \mathcal{R} -Funktionen

Dieses Kapitel betrachtet nun \mathcal{R} -Funktionen, die erstmals von Rvachev in [Rva67] eingeführt wurden. Sie stellen die Mittel zur Verfügung, Gewichtsfunktionen auf booleschen Kombinationen von Gebieten zu definieren und stellen somit ein mächtiges und sehr nützliches Hilfsmittel für die web-Splines zur Verfügung.

Viele der hier präsentierten Ergebnisse werden in den russischen Werken [Rva67, Rva74, Rva82] behandelt. Doch vor allem die in englisch verfassten Arbeiten [ST99, Sha88] dienen als Grundlage für dieses Kapitel.

2.1 Definition der Gewichtsfunktionen

Definition 2.1

Sei Ω ein abgeschlossenes Gebiet des \mathbb{R}^n . Eine Funktion $\omega : \mathbb{R}^n \rightarrow \mathbb{R}$ heißt Gewichtsfunktion auf Ω genau dann, wenn für alle $x \in \mathbb{R}^n$ gilt

$$\omega(x) < 0 \quad \text{für } x \notin \Omega \tag{2.1}$$

$$\omega(x) \geq 0 \quad \text{für } x \in \Omega. \tag{2.2}$$

Die Funktion ω heißt darüber hinaus strikte Gewichtsfunktion auf Ω genau dann, wenn zusätzlich noch

$$\omega(x) = 0 \Leftrightarrow x \in \partial\Omega \tag{2.3}$$

gilt. Die Menge der strikten Gewichtsfunktionen auf Ω wird mit \mathcal{G}_Ω bezeichnet.

Die Definitionen entsprechen der intuitiven Vorstellung, dass eine Gewichtsfunktion im Innern des Gebietes positiv und im Äußeren negativ sein soll.

2.2 Definition der \mathcal{R} -Funktionen

In diesem Abschnitt werden die \mathcal{R} -Funktionen eingeführt. Sie erlauben die Berechnung von Gewichtsfunktionen zu Gebieten, die mittels boolescher Funktionen zusammengesetzt werden. Das folgende Beispiel soll die Problemstellung verdeutlichen.

Beispiel 2.2

Gegeben seien die drei Kreise

$$K_1 : x^2 + y^2 = 1$$

$$K_2 : x^2 + (y - 1/2)^2 = 1$$

$$K_3 : (x - 1/2)^2 + y^2 = 1,$$

zu denen sich einfache Gewichtsfunktionen angeben lassen (vgl. Kapitel 4.1.4)

$$\omega_1 := 1 - x^2 - y^2$$

$$\omega_2 := 1 - x^2 - (y - 1/2)^2$$

$$\omega_3 := 1 - (x - 1/2)^2 - y^2.$$

Die drei Kreise sollen nun miteinander geschnitten werden (s. Abb. 2.1) und die Ge-

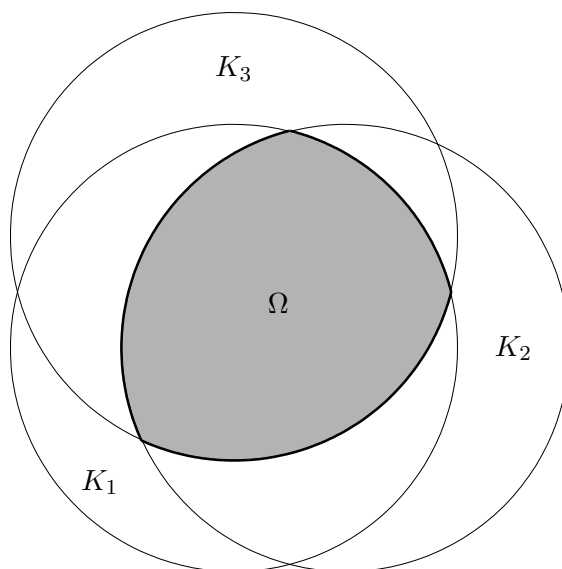


Abbildung 2.1: Die drei Kreise aus Beispiel 2.2 und ihre Schnittmenge

wichtsfunktionen auf den einzelnen Kreisen so miteinander verrechnet werden, dass eine Gewichtsfunktion für $\Omega = K_1 \cap K_2 \cap K_3$ entsteht.

Dieses Beispiel wird an mehreren Stellen dieser Arbeit erneut aufgegriffen und erweitert werden. Für die formale Behandlung dieses Problems werden nun einige Definitionen erforderlich sein.

Definition 2.3

Eine Partition Γ der reellen Zahlen ist eine Aufteilung von \mathbb{R} in m disjunkte Intervalle X_1, \dots, X_m . Es gilt also

$$\Gamma_m = \{X_1, \dots, X_m\}, \quad \bigcup_{i=1}^m X_i = \mathbb{R}.$$

Im weiteren wird der Spezialfall

$$\Gamma_2 = \{(-\infty, 0), [0, \infty)\}$$

verwendet.

Die Partition Γ_2 trennt also die positiven von den negativen reellen Zahlen. Für $n > 2$ ergeben sich dementsprechend feinere Einteilungen. Die Unterscheidung kann mit Hilfe der folgenden Definitionen einfacher dargestellt werden.

Definition 2.4

Die Indexmenge wird definiert durch

$$B_m := \{0, \dots, m\}.$$

Im Falle $m = 2$ ergibt sich die Grundmenge B_2 für boolesche Funktionen.

Den naheliegenden Zusammenhang zwischen Partition und Indexmenge ist die nun folgende Abbildung.

Definition 2.5

Die Funktion, die durch

$$\begin{aligned} S_m &: \mathbb{R} \longrightarrow B_m \\ x &\longmapsto i \quad \text{für } x \in X_i \end{aligned}$$

definiert wird, heißt Indikatorfunktion. Hervorzuheben ist auch hier der Fall

$$\begin{aligned} S_2 &: \mathbb{R} \longrightarrow B_2 = \{0, 1\} \\ x &\longmapsto \begin{cases} 0 & \text{für } x < 0 \\ 1 & \text{für } x \geq 0 \end{cases} \end{aligned}$$

Die Indikatorfunktion weist also jedem Teil der Partition einen Index zu und erlaubt somit eine einfache Unterscheidung der Teile.

Bemerkung 2.6

S_2 ist also vollständig durch das Vorzeichen seines Argumentes definiert, d.h. S_2 ändert seinen Wert nur auf den Koordinatenebenen $\{x_i = 0\}$. Sie ist also eine Signumfunktion. Der allgemeinere Fall S_n verfeinert diese Eigenschaft zusätzlich. Insbesondere ist jeder Rundungsoperator auf den positiven Zahlen eine solche Indikatorfunktion.

Mit diesen Hilfsmitteln lassen sich nun \mathcal{R} -Funktionen definieren.

Definition 2.7

Eine Funktion $R: \mathbb{R}^n \rightarrow \mathbb{R}$ heißt \mathcal{R} -Funktion genau dann, wenn eine Funktion F existiert, mit

$$F: B_m^n \longrightarrow B_m,$$

sodass

$$S_m(R(x_1, \dots, x_n)) = F(S_m(x_1), \dots, S_m(x_n)), \quad x_i \in \mathbb{R}$$

gilt. In diesem Fall heißt F die begleitende Funktion von R . Die Menge aller \mathcal{R} -Funktionen wird mit \mathcal{R} bezeichnet.

In dem oben bereits erwähnten Spezialfall ergibt sich, dass $\omega: \mathbb{R}^n \rightarrow \mathbb{R}$ genau dann eine \mathcal{R} -Funktion ist, wenn eine boolesche Funktion $F: \{0, 1\}^n \rightarrow \{0, 1\}$ existiert, sodass

$$S_2(\omega(x_1, \dots, x_n)) = F(S_2(x_1), \dots, S_2(x_n)) \quad x_i \in \mathbb{R}$$

ist. ω heißt \mathcal{R} -Funktion zur Menge Ω , falls $\Omega = \{x \in \mathbb{R}^n : S_2(\omega(x)) > 0\}$ ist.

F ist hier also eine boolesche Funktion, die $\{0, 1\}^n$ auf $\{0, 1\}$ abbildet. Im Weiteren soll allerdings auch die Funktion, welche die Gebiete Ω_1 und Ω_2 auf Ω abbildet, diese Bezeichnung tragen. Für die Kombinationsfunktion gilt also folglich

$$F : \Omega_1 \times \Omega_2 \longrightarrow \Omega := \{x : F(S_2(x)) > 0\}.$$

Hier wird F also sowohl als boolescher Operator als auch als mengentheoretischer Kombinationsoperator verwendet.

Beispiel 2.8

S_2 aus Definition 2.5 ist eine einfache \mathcal{R} -Funktion auf $\Omega = (\mathbb{R}_0^+)^n$, weil $S_2 \circ S_2 = S_2$ gilt. Eine detaillierte Behandlung von \mathcal{R} -Funktionen erfolgt in Kapitel 2.3.

Sowohl Gewichtsfunktionen als auch \mathcal{R} -Funktionen sind also an ein Gebiet gebunden. Die Gemeinsamkeiten der beiden Funktionentypen lassen sich im folgenden Satz fassen.

Satz 2.9

Gegeben seien die Gewichtsfunktionen ω_1 und ω_2 zu den Gebieten Ω_1 bzw. Ω_2 . Weiter sei $\Omega = F(\Omega_1, \Omega_2)$. Dann gilt für jede \mathcal{R} -Funktion

$$R(\omega_1, \omega_2) \in \mathcal{G}_\Omega,$$

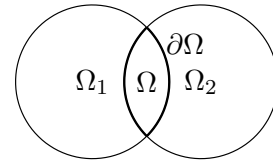
also ist jede \mathcal{R} -Funktion eine Gewichtsfunktion auf Ω . Erfüllt ein $R \in \mathcal{R}$ darüber hinaus noch (2.3), so ist R eine strikte Gewichtsfunktion.

Nicht jede \mathcal{R} -Funktion ist eine strikte Gewichtsfunktion. Eine Begründung hierfür wird in Abschnitt 7.3 illustriert.

Also liefert die Kombination von Gewichtsfunktionen mittels geeigneter \mathcal{R} -Funktionen stets wieder Gewichtsfunktionen. Es besteht damit also die Möglichkeit, Gewichte auf Kombinationen von Gebieten einfach aus den Gewichtsfunktionen der Ausgangsmengen anzugeben.

Bemerkung 2.10

Im Weiteren wird das Gebiet, zu dem ω_1 die Gewichtsfunktion ist, Ω_1 genannt, das Gebiet zu ω_2 erhält den Namen Ω_2 . Das kombinierte Gebiet heißt Ω und der Rand wird schließlich mit $\partial\Omega$ bezeichnet.



Bemerkung 2.11

Aus Satz 2.9 folgt, dass $R \in \mathcal{R}$ genau dann, wenn R auf Ω die Bedingungen (2.1) und (2.3) erfüllt und stetig ist. Damit ist das Verschwinden von R auf dem Rand von Ω impliziert.

Die \mathcal{R} -Funktionen werden allein nach ihren Werten auf der Partition Γ unterschieden. Aus diesem Grunde gibt es natürlich keine eindeutige \mathcal{R} -Funktion zu einer Partition.

Definition 2.12

Seien Q, R stetige, reelle Funktionen

$$Q, R : \mathbb{R}^n \longrightarrow \mathbb{R}$$

und die Äquivalenzrelation

$$Q \sim R \iff S_m(Q) = S_m(R)$$

gegeben. Sei M die Menge aller Funktionen, für die

$$Q, R \in M \iff Q \sim R$$

gilt. M wird dann ein Ast der \mathcal{R} -Funktionen genannt.

In dem oben erwähnten Spezialfall $n = m = 2$ ergibt sich

$$\begin{aligned} Q, R &: \mathbb{R}^2 \longrightarrow \mathbb{R} \\ Q \sim R &\iff S_2(\omega_1) = S_2(\omega_2). \end{aligned}$$

Zwei Funktionen gehören also zu demselben Ast, wenn sie dieselbe Funktion F als begleitende Funktion besitzen. Im eindimensionalen Spezialfall mit der Partition Γ_2 haben die Funktionen eines Astes also stets dasselbe Vorzeichen.

Die Anzahl der Äste ergibt sich aus folgender einfacher Überlegung.

Bemerkung 2.13

Es existieren 2^n Kombinationen für die n Argumente der booleschen Funktion F , die jeder der Kombinationen zwei Ergebnisse zuweisen kann. Also existieren 2^{2^n} verschiedene boolesche Funktionen F , somit ebensoviele Indikatorfunktionen S_m und schließlich ebensoviele Äste der \mathcal{R} -Funktionen.

\mathcal{R} -Funktionen hängen also eng mit booleschen Funktionen zusammen. Die folgende Definition erlaubt eine Kategorisierung dieser Funktionen.

Definition 2.14

Ein System G von booleschen Funktionen heißt vollständig, falls jede boolesche Funktion als Kombination von Funktionen aus G dargestellt werden kann. Ist H ein System von \mathcal{R} -Funktionen, deren begleitende Funktionen die Menge G bilden, so heißt auch H vollständig.

Es ist wohlbekannt, dass das System boolescher Funktionen

$$G = \{0, \neg X, X_1 \wedge X_2\}$$

vollständig ist. Also ist die Definition von drei \mathcal{R} -Funktionen mit diesen begleitenden Funktionen schon ausreichend. Gewöhnlich wird allerdings das System

$$G = \{0, \neg X, X_1 \wedge X_2, X_1 \vee X_2, \}$$

benutzt, wie es auch hier geschehen soll.

Definition 2.15

Sei ν der nach außen gerichtete Normalenvektor auf $\partial\Omega$. Eine Gewichtsfunktion R auf Ω heißt normalisiert zur Ordnung n , falls die Bedingungen

$$\left. \frac{\partial}{\partial \nu} R \right|_{\partial\Omega} = 1 \tag{2.4}$$

$$\left. \frac{\partial^k}{\partial \nu^k} R \right|_{\partial\Omega} = 0 \quad k = 2, \dots, n \tag{2.5}$$

auf allen Punkten erfüllt ist, in denen ein eindeutiger Normalenvektor existiert.

Die Normalisiertheit einer Funktion liefert also eine Aussage über die Ähnlichkeit mit der Abstandsfunktion $d(x, \partial\Omega)$.

2.3 Eigenschaften verschiedener \mathcal{R} -Funktionen

In diesem Abschnitt werden unterschiedliche Definitionen auf ihre \mathcal{R} -Funktionseigenschaft, Differenzierbarkeitseigenschaften und Normalisiertheit untersucht. Dabei wird stets angenommen, dass die Funktionen ω_1 und ω_2 ausreichend oft differenzierbare, von ausreichender Ordnung normalisierte Gewichtsfunktionen sind. Außerdem ist mit dem Innern des Gebietes Ω stets $\Omega \setminus \partial\Omega$ gemeint.

Bemerkung 2.16

Die Differenzierbarkeit der nun folgenden \mathcal{R} -Funktionen kann auf die Differenzierbarkeit der \mathcal{R} -Funktion in beiden Argumenten zurückgeführt werden.

$$\begin{aligned} \frac{\partial}{\partial v} R(\omega_1, \omega_2) &= v \nabla R(\omega_1, \omega_2) \\ &= \sum_{i=1}^n v_i \left(\frac{\partial}{\partial \omega_1} R(\omega_1, \omega_2) \frac{\partial}{\partial x_i} \omega_1 + \frac{\partial}{\partial \omega_2} R(\omega_1, \omega_2) \frac{\partial}{\partial x_i} \omega_2 \right), \end{aligned}$$

wobei $v \in \mathbb{R}^n$ eine beliebige Richtung darstellt. Nach den Voraussetzungen genügt für die Diskussion von $\frac{\partial}{\partial v} R(\omega_1, \omega_2)$ also eine Betrachtung der Ableitungen $\frac{\partial}{\partial \omega_1} R(\omega_1, \omega_2)$ und $\frac{\partial}{\partial \omega_2} R(\omega_1, \omega_2)$.

2.3.1 Funktions-System \mathcal{R}_1

Das einfachste (und intuitivste) \mathcal{R} -Funktionen-System sind die Funktionen in \mathcal{R}_1 . Sie spielen aufgrund mangelnder Differenziationseigenschaften eine nur untergeordnete Rolle, sollen aber den Grundgedanken der \mathcal{R} -Funktionen verdeutlichen.

Definition 2.17

Das Funktionen-System \mathcal{R}_1 ist durch

$$\mathcal{R}_1 := \{-1; -\omega_1; \min(\omega_1, \omega_2); \max(\omega_1, \omega_2)\}.$$

definiert.

Satz 2.18

Die \mathcal{R}_1 -Funktionen sind überall analytisch, bis auf die Punkte $\omega_1 = \omega_2$. Darüber hinaus sind sie normalisiert von beliebiger Ordnung.

Beweis:

Die Funktionen lassen sich umschreiben zu

$$\begin{aligned} R_1^\cup &= \max(\omega_1, \omega_2) = \frac{1}{2} (\omega_1 + \omega_2 + |\omega_1 - \omega_2|) \quad \text{und} \\ R_1^\cap &= \min(\omega_1, \omega_2) = \frac{1}{2} (\omega_1 + \omega_2 - |\omega_1 - \omega_2|). \end{aligned}$$

Es folgt also die Analytizität bis auf die Punkte, an denen der Betrag verschwindet $|\omega_1 - \omega_2| = 0 \Leftrightarrow \omega_1 = \omega_2$. Dieselbe Argumentation ergibt natürlich auch, dass R_1^\cap analytisch bis auf die Nullstellen des Betrages ist.

Die Normalisiertheit lässt sich leicht anhand der Ableitungen feststellen

$$\frac{\partial}{\partial \omega_1} R_1^\cap \Big|_{\omega_1=0, \omega_2>0} = \frac{1}{2}(1 \mp 1)$$

und

$$\left. \frac{\partial}{\partial \omega_1} R_1^{\cup} \right|_{\omega_1=0, \omega_2 < 0} = \frac{1}{2}(1 \pm 1).$$

Die Ableitung nach ω_2 ergibt wegen der Symmetrie der beiden Argumente das entsprechende Resultat. Die höheren Ableitungen verschwinden alle, somit sind die \mathcal{R}_1 -Funktionen also normalisiert von beliebiger Ordnung. ■

Die Funktionen in \mathcal{R}_1 realisieren also die Abstandsfunktion auf Ω , falls ω_1 und ω_2 die Abstandsfunktionen auf Ω_1 und Ω_2 sind. Diese Funktionen eignen sich also für einfache Abstandsberechnungen kombinierter Gebiete.

2.3.2 Funktions-System \mathcal{R}_0

Definition 2.19

Das \mathcal{R}_0 -Funktionen-System ist durch

$$R_0 := \left\{ -1; -\omega_1; \omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2}; \omega_1 + \omega_2 + \sqrt{\omega_1^2 + \omega_2^2} \right\}.$$

definiert.

Satz 2.20

Die \mathcal{R}_0 -Funktionen sind bis auf die Punkte $\omega_1 = \omega_2 = 0$ überall analytisch.

Beweis:

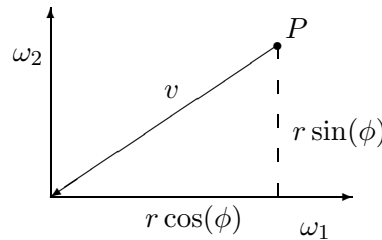


Abbildung 2.2: Skizze zur Richtungsableitung

Die Wurzelfunktion ist auf \mathbb{R}_0^+ analytisch, die restlichen Operationen auf ganz \mathbb{R} . Um die Nichtdifferenzierbarkeit der \mathcal{R}_0 -Funktionen im Ursprung zu beweisen, werden die begleitenden Funktionen zur Vereinigung und Schnitt gewählt

$$\begin{aligned} R_0^{\cup} &= \omega_1 + \omega_2 \pm \sqrt{\omega_1^2 + \omega_2^2} \\ \frac{\partial}{\partial \omega_1} R_0^{\cup} &= 1 \pm \frac{\omega_1}{\sqrt{\omega_1^2 + \omega_2^2}} \\ \frac{\partial}{\partial \omega_2} R_0^{\cup} &= 1 \pm \frac{\omega_2}{\sqrt{\omega_1^2 + \omega_2^2}} \end{aligned}$$

und mittels $\omega_1 = r \cos(\phi)$ und $\omega_2 = r \sin(\phi)$ in Polarkoordinaten umgewandelt

$$\begin{aligned}\frac{\partial}{\partial \omega_1} R_0^{\cup \cap} &= 1 \pm \frac{r \cos(\phi)}{\sqrt{r^2 \cos^2(\phi) + r^2 \sin^2(\phi)}} = 1 \pm \cos(\phi) \\ \frac{\partial}{\partial \omega_2} R_0^{\cup \cap} &= 1 \pm \sin(\phi).\end{aligned}$$

Für die Ableitung in Richtung eines Vektors $v = [\cos(\phi), \sin(\phi)]$, der vom Ursprung aus durch den Punkt P geht (s. Abb. 2.2), ergibt sich somit

$$\begin{aligned}\frac{\partial}{\partial v} R_0^{\cup \cap} &= v_1 \frac{\partial}{\partial \omega_1} R_0^{\cup \cap} + v_2 \frac{\partial}{\partial \omega_2} R_0^{\cup \cap} \\ &= \cos(\phi)(1 \pm \cos(\phi)) + \sin(\phi)(1 \pm \sin(\phi)) \\ &= \cos(\phi) + \sin(\phi) \pm 1.\end{aligned}$$

Da die Richtungsableitung also von dem Winkel abhängt, ist $R_0^{\cup \cap}$ im Nullpunkt nicht differenzierbar. ■

2.3.3 Funktions-System \mathcal{R}_α

Dieses \mathcal{R} -Funktionen-System ist durch

$$\mathcal{R}_\alpha := \left\{ -1; \quad -\omega_1; \quad \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2} \right); \right. \\ \left. \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 + \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2} \right) \right\}$$

mit einer symmetrischen Funktion $\alpha : \mathbb{R}^n \rightarrow \mathbb{R}$, $-1 < \alpha \leq 1$ definiert.

Satz 2.21

Die in \mathcal{R}_α definierten Funktionen sind wohldefinierte \mathcal{R} -Funktionen.

Beweis:

Es muss also gezeigt werden, dass der Radikant stets definiert ist, und dass die Forderungen an \mathcal{R} -Funktionen erfüllt sind.

Wohldefiniertheit

Für den Radikanten in R_α^\cap und R_α^\cup gilt

$$\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2 \geq \omega_1^2 + \omega_2^2 - 2\omega_1\omega_2 = (\omega_1 - \omega_2)^2 \geq 0.$$

\mathcal{R} -Funktionseigenschaft

Mit Hilfe von $-1 < \alpha \leq 1$ und $x + y - |x - y| = \min(x, y)$ ergeben sich die folgenden Resultate.

$x \notin \Omega$: Es gilt also $\omega_1 < 0$, $\omega_2 < 0$ und damit

$$\begin{aligned}R_\alpha^\cap(\omega_1, \omega_2) &= \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2} \right) \\ &\leq \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\omega_1\omega_2} \right) \\ &= \frac{1}{1+\alpha} (\omega_1 + \omega_2 - |\omega_1 - \omega_2|) \\ &= \frac{1}{1+\alpha} \min(\omega_1, \omega_2) \\ &< 0,\end{aligned}$$

oder $\omega_1 < 0, \omega_2 > 0$ und somit

$$\begin{aligned} R_\alpha^\cap(\omega_1, \omega_2) &= \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2} \right) \\ &\leq \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\omega_1\omega_2} \right) \\ &= \frac{1}{1+\alpha} (\omega_1 + \omega_2 - |\omega_1 - \omega_2|) \\ &= \frac{1}{1+\alpha} \min(\omega_1, \omega_2) \\ &< 0. \end{aligned}$$

$x \in \Omega$: Dies ist gleichbedeutend mit $\omega_1 > 0, \omega_2 > 0$, woraus

$$\begin{aligned} R_\alpha^\cap(\omega_1, \omega_2) &= \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2} \right) \\ &\geq \frac{1}{1+1} (\omega_1 + \omega_2 - \sqrt{\omega_1 + \omega_2 + 2\omega_1\omega_2}) \\ &= \frac{1}{2} (\omega_1 + \omega_2 - |\omega_1 + \omega_2|) \\ &= \frac{1}{2} (\omega_1 + \omega_2 - \omega_1 - \omega_2) \\ &= 0 \end{aligned}$$

folgt.

Laut Bemerkung 2.11 ist R_α^\cap also eine \mathcal{R} -Funktion. Mit denselben Hilfsmitteln erreicht man die Abschätzungen für R_α^\cup . ■

Satz 2.22

Die \mathcal{R}_α -Funktionen sind normalisiert von der Ordnung 1. Nach Definition 2.16 gilt also für eine Funktion $R \in \mathcal{R}_\alpha$

$$R|_{\partial\Omega} = 0, \quad \text{und} \quad \frac{\partial}{\partial\nu} R \Big|_{\partial\Omega} = 1.$$

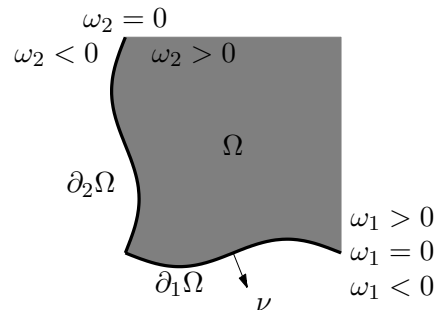


Abbildung 2.3: Skizze zur Normalisierung

Beweis:

Der Beweis wird hier für die Funktion R_α^\cup durchgeführt. Wie in der Abbildung 2.3 zu

sehen ist, existiert ein Teil des Randes $\partial_1\Omega$, auf dem $\omega_1 \equiv 0$, $\omega_2 > 0$ und insbesondere $\frac{\partial}{\partial\nu}\omega_1 = 1$, da ω_1 laut Voraussetzung auf diesem Rand normalisiert ist. Der Beweis für den Fall $\omega_2 \equiv 0$ und $\omega_1 \geq 0$ ($\partial_2\Omega$) läuft vollkommen analog.

Zunächst läßt sich R_α^\cup als Verkettung ansehen

$$R_\alpha^\cup(x, y) = R(\alpha(x, y); \omega_1, \omega_2)$$

mit

$$R: (\alpha; \omega_1, \omega_2) \mapsto \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 + \sqrt{\omega_1^2 + \omega_2^2 - \alpha\omega_1\omega_2} \right).$$

Mittels der Kettenregel und der Normalisiertheit von ω_1 ergibt sich nun

$$\begin{aligned} \left. \frac{\partial}{\partial\nu} R(\alpha; \omega_1, \omega_2) \right|_{\partial_1\Omega} &= \left. \frac{\partial}{\partial\alpha} R \frac{\partial}{\partial\nu} \alpha + \frac{\partial}{\partial\omega_1} R \frac{\partial}{\partial\nu} \omega_1 + \frac{\partial}{\partial\omega_2} R \frac{\partial}{\partial\nu} \omega_2 \right|_{\omega_1=0, \omega_2>0} \\ &= \left. \frac{\partial}{\partial\alpha} R \frac{\partial}{\partial\nu} \alpha + \frac{\partial}{\partial\omega_1} R + \frac{\partial}{\partial\omega_2} R \frac{\partial}{\partial\nu} \omega_2 \right|_{\omega_1=0, \omega_2>0} \end{aligned} \quad (2.6)$$

Die kritischen Punkte $\omega_1 = \omega_2 = 0$ werden bei der Betrachtung der Normalisiertheit außer Acht gelassen, da an diesen Stellen ohnehin kein eindeutiger Normalervektor existiert. Die Einzelbetrachtung der Ableitungen liefert dann

$$\begin{aligned} \left. \frac{\partial}{\partial\alpha} R \right|_{\omega_1=0, \omega_2>0} &= \left. \frac{\partial}{\partial\alpha} \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2} \right) \right|_{\omega_1=0, \omega_2>0} \\ &= -\frac{1}{(1+\alpha)^2} \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2} \right) + \\ &\quad + \frac{2\omega_1\omega_2}{(1+\alpha)(2\sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2})} \Big|_{\omega_1=0, \omega_2>0} \\ &= -\frac{1}{(1+\alpha)^2} (\omega_2 - |\omega_2|) \Big|_{\omega_1=0, \omega_2>0} = 0, \end{aligned} \quad (2.7)$$

bzw.

$$\begin{aligned} \left. \frac{\partial}{\partial\omega_1} R \right|_{\omega_1=0, \omega_2>0} &= \left. \frac{1}{1+\alpha} \left(1 - \frac{\omega_2}{\sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2}} \right) \right|_{\omega_1=0, \omega_2>0} \\ &= \frac{1}{1+\alpha} \left(1 - \frac{\omega_2}{|\omega_2|} \right) \Big|_{\omega_1=0, \omega_2>0} = 0 \end{aligned} \quad (2.8)$$

und schließlich

$$\begin{aligned} \left. \frac{\partial}{\partial\omega_2} R \right|_{\omega_1=0, \omega_2>0} &= \left. \frac{1}{1+\alpha} \left(1 - \frac{2\omega_1 - 2\alpha\omega_2}{2\sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2}} \right) \right|_{\omega_1=0, \omega_2>0} \\ &= \frac{1}{1+\alpha} \left(1 + \frac{2\alpha\omega_2}{2|\omega_2|} \right) \Big|_{\omega_1=0, \omega_2>0} \\ &= \frac{1}{1+\alpha} (1 + \alpha) = 1 \end{aligned} \quad (2.9)$$

Somit ergibt sich für (2.6) mit Hilfe von (2.7), (2.8) und (2.9)

$$\left. \frac{\partial}{\partial\nu} R(\alpha; \omega_1, \omega_2) \right|_{\partial_1\Omega} = 1.$$

Der Beweis für R_α^\cap verläuft vollkommen analog. ■

In [Rva82] schlägt Rvachev für die Funktion

$$\alpha = \frac{1}{1 + \omega_1^2 + \omega_2^2}$$

vor. In der Tatsache, dass der Gradient der Funktionen in \mathcal{R}_α in der Nähe des Ursprungs stark oszillieren kann, offenbart sich allerdings ein Nachteil dieser Funktionen.

2.3.4 Funktions-System \mathcal{R}_0^m

Definition 2.23

Das \mathcal{R}_0^m -Funktionen-System ist durch

$$\mathcal{R}_0^m := \left\{ -1; \quad -\omega_1; \quad \left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2} \right) (\omega_1^2 + \omega_2^2)^{m/2}; \right. \\ \left. \left(\omega_1 + \omega_2 + \sqrt{\omega_1^2 + \omega_2^2} \right) (\omega_1^2 + \omega_2^2)^{m/2} \right\}$$

definiert, wobei $m \in 2\mathbb{N}$.

Die Funktionen in \mathcal{R}_0^m stellen also eine Modifizierung der \mathcal{R}_0 -Funktionen dar.

Satz 2.24 (\mathcal{R} -Funktionseigenschaft)

Die Funktionen in \mathcal{R}_0^m sind \mathcal{R} -Funktionen.

Beweis:

Die Funktionen, deren begleitende Funktionen die Vereinigung und der Schnitt sind, gehen aus den entsprechenden Funktionen in \mathcal{R}_α durch Multiplikation mit dem Faktor $(\omega_1^2 + \omega_2^2)^{m/2}$ hervor. Dieser ist nur dann Null, wenn $\omega_1 = \omega_2 = 0$ ist, und ist ansonsten positiv. Also folgt die Behauptung mit Satz 2.21. ■

Satz 2.25

Die Funktionen in \mathcal{R}_0^m sind im Innern des Gebietes Ω analytisch und im Punkt $\omega_1 = \omega_2 = 0$ m -mal differenzierbar. Darüber hinaus ist der Gradient der Funktion stets beschränkt.

Beweis:

Wie im Beweis des Satzes 2.20 wird hier die Richtungsableitung im Nullpunkt betrachtet, indem die Ableitung im Nullpunkt durch eine Grenzwertbildung auf einer Ursprungsgeraden ermittelt wird (vgl. Abb. 2.2). Mit $\omega_1 = r \cos(\phi)$, $\omega_2 = r \sin(\phi)$ lautet R_0^m in Polarkoordinaten

$$\begin{aligned} R_0^{m \cup \cap} &= \left(\omega_1 + \omega_2 \pm \sqrt{\omega_1^2 + \omega_2^2} \right) (\omega_1^2 + \omega_2^2)^{m/2} \\ &= \left(r \cos(\phi) + r \sin(\phi) \pm \sqrt{r^2 \cos^2(\phi) + r^2 \sin^2(\phi)} \right) (r^2 \cos^2(\phi) + r^2 \sin^2(\phi))^{m/2} \\ &= r (\cos(\phi) + \sin(\phi) \pm 1) r^m \\ &= r^{m+1} (\cos(\phi) + \sin(\phi) \pm 1). \end{aligned}$$

Für diese spezielle Wahl ist die Richtungsableitung also identisch mit der Ableitung nach dem Radius (s. Abb. 2.2). Also lauten die gewünschten Ableitungen

$$\frac{\partial^n}{\partial \ell^n} R_0^{m \cup \cap} = \begin{cases} \frac{(m+1)!}{(m+1-n)!} r^{m+1-n} (\cos(\phi) + \sin(\phi) \pm 1) & \text{für } n \leq m+1 \\ 0 & \text{für } n > m+1 \end{cases}.$$

Für den Grenzwert der Ableitungen im Ursprung ergibt sich schließlich

$$\lim_{r \rightarrow 0} \left(\frac{\partial^n}{\partial \ell^n} R_0^{m \cup \cap} \right) = \begin{cases} 0 & \text{für } n < m + 1 \\ (m + 1)! (\cos(\phi) + \sin(\phi) \pm 1) & \text{für } n = m + 1 \\ 0 & \text{für } n > m + 1 \end{cases},$$

weshalb die Grenzwerte auf allen Ursprungsgeraden übereinstimmen, solange $n \leq m$. Für $n = m + 1$ existiert kein eindeutiger Grenzwert mehr, weshalb \mathcal{R}_0^m genau m -mal stetig differenzierbar ist. Allerdings zeigt der letzte Grenzwert, dass diese Ableitung und somit auch der Gradient stets beschränkt bleibt. ■

Bemerkung 2.26

Die Funktionen in \mathcal{R}_0^m sind nicht normalisiert. Schon die einfachen Gewichtsfunktionen $\omega_1(x, y) = x$ und $\omega_2(x, y) = y$ und $m = 2$ liefert die Funktion

$$R_0^{m \cap} = (x + y - \sqrt{x^2 - y^2})(x^2 + y^2),$$

welche auf dem Rand $x > 0, y = 0$ die Ableitung

$$\left. \frac{\partial}{\partial y} R_0^{m \cap} \right|_{x>0, y=0} = \left(1 - \frac{y}{\sqrt{x^2 + y^2}} \right) (x^2 + y^2) + 2y (x + y - \sqrt{x^2 - y^2}) \Big|_{x>0, y=0} = x^2$$

annimmt. Die Funktion ist also nicht normalisiert. Eine Illustration dieser Eigenschaft findet sich im Kapitel 7.

2.3.5 Funktions-System \mathcal{R}_p

Definition 2.27

Das \mathcal{R}_p -Funktionen-System ist durch

$$\mathcal{R}_p := \left\{ -1; -\omega_1; \omega_1 + \omega_2 - (\omega_1^p + \omega_2^p)^{\frac{1}{p}}; \omega_1 + \omega_2 + (\omega_1^p + \omega_2^p)^{\frac{1}{p}} \right\}$$

definiert, wobei p eine gerade natürliche Zahl ist.

Satz 2.28

Die Funktionen in \mathcal{R}_p sind wohldefinierte \mathcal{R} -Funktionen.

Beweis:

Es muss gezeigt werden, dass die Funktionen in \mathcal{R}_p auf \mathbb{R}^2 definiert sind, und dass sie die \mathcal{R} -Funktionseigenschaft erfüllen.

Wohldefiniertheit

Der Radikant ist nichtnegativ, weil p gerade ist; also ist die Wurzel immer definiert.

\mathcal{R} -Funktionseigenschaft

Auch hier werden wegen Bemerkung 2.11 nur zwei Fälle betrachtet.

$\mathbf{x} \in \Omega$: Dies ist äquivalent zu $\omega_1 > 0$ und $\omega_2 > 0$, also folgt

$$\begin{aligned} R_p^\cap &= \omega_1 + \omega_2 - (\omega_1^p + \omega_2^p)^{\frac{1}{p}} > \omega_1 + \omega_2 - ((\omega_1 + \omega_2)^p)^{\frac{1}{p}} \\ &= \omega_1 + \omega_2 - (\omega_1 + \omega_2) = 0 \\ R_p^\cup &= \omega_1 + \omega_2 + (\omega_1^p + \omega_2^p)^{\frac{1}{p}} > 0. \end{aligned}$$

$\mathbf{x} \notin \Omega$: In diesem Zusammenhang werden die Äquivalenzen

$$\begin{array}{llll} \omega_1^p & > 0 & \text{bzw.} & \omega_2^p & > 0 \\ \Leftrightarrow \omega_1^p + \omega_2^p & > \omega_2^p & & \Leftrightarrow \omega_1^p + \omega_2^p & > (-\omega_1)^p \\ \Leftrightarrow (\omega_1^p + \omega_2^p)^{\frac{1}{p}} & > \omega_2 & & \Leftrightarrow (\omega_1^p + \omega_2^p)^{\frac{1}{p}} & > -\omega_1 \\ \Leftrightarrow \omega_2 - (\omega_1^p + \omega_2^p)^{\frac{1}{p}} & < 0 & & \Leftrightarrow \omega_1 + (\omega_1^p + \omega_2^p)^{\frac{1}{p}} & > 0 \end{array}$$

benötigt. Dieser Fall entspricht nun entweder $\omega_1 < 0, \omega_2 > 0$, also

$$\begin{aligned} R_p^\cap &= \omega_1 + \underbrace{\omega_2 - (\omega_1^p + \omega_2^p)^{\frac{1}{p}}}_{<0} < 0 \\ R_p^\cup &= \omega_2 + \underbrace{\omega_1 + (\omega_1^p + \omega_2^p)^{\frac{1}{p}}}_{>0} > 0, \end{aligned}$$

oder $\omega_1 > 0, \omega_2 < 0$, was wegen der Symmetrie von \mathcal{R}_p dasselbe Ergebnis liefert, oder schließlich $\omega_1 < 0$ und $\omega_2 < 0$, was

$$\begin{aligned} R_p^\cap &= \omega_1 + \omega_2 - (\omega_1^p + \omega_2^p)^{\frac{1}{p}} < 0 \\ R_p^\cup &= \omega_1 + \omega_2 + (\omega_1^p + \omega_2^p)^{\frac{1}{p}} < \omega_1 + \omega_2 + \left(\sum_{i=0}^p \binom{p}{i} \underbrace{\omega_1^{p-i} \omega_2^i}_{>0} \right)^{\frac{1}{p}} \\ &= \omega_1 + \omega_2 + |\omega_1 + \omega_2| = 0. \end{aligned}$$

entspricht. Damit ist die Behauptung bewiesen. ■

Satz 2.29 (Ableitungen der \mathcal{R}_p -Funktionen)

Die Funktionen in \mathcal{R}_p sind für $(\omega_1, \omega_2) \neq (0, 0)$ beliebig oft differenzierbar. Ihre Ableitungen lauten

$$\frac{\partial^n}{\partial \omega_2^n} \mathcal{R}_p = \delta_n^1 + \sum_{i=1}^n (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_2^{ip-n} \alpha_{i,n},$$

wobei δ das Kroneckersymbol und $\alpha_{i,n}$ eine Konstante ist, die nur von i, p und n abhängt.

Beweis:

Wie schon bei den Beweisen zuvor wird hier nur R_p^\cup betrachtet, da R_p^\cap ohne Mehraufwand mit denselben Mitteln untersucht werden kann. Die Behauptung wird mittels vollständiger Induktion bewiesen.

$n = 1$:

$$\frac{\partial}{\partial \omega_i} R_p^\cup = 1 + (\omega_1^p + \omega_2^p)^{\frac{1-p}{p}} \omega_i^{p-1} (-1) \tag{2.10}$$

$n \Rightarrow (n+1)$:

$$\begin{aligned}
\frac{\partial^{n+1}}{\partial \omega_i^{n+1}} R_p^\cup &= \frac{\partial}{\partial \omega_i} \left(\frac{\partial^n}{\partial \omega_i^n} R_p^\cup \right) \\
&= \frac{\partial}{\partial \omega_i} \left(\sum_{i=1}^n (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_i^{ip-n} \alpha_{i,n} \right) \\
&= \sum_{i=1}^n \left((1-ip) (\omega_1^p + \omega_2^p)^{\frac{1-(i+1)p}{p}} \omega_i^{(i+1)p-(n+1)} \alpha_{i,n} + \right. \\
&\quad \left. (ip-n) (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_i^{ip-(n+1)} \alpha_{i,n} \right) \\
&= \sum_{i=2}^{n+1} (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_i^{ip-(n+1)} (1-(i-1)p) \alpha_{i,n} + \\
&\quad \sum_{i=1}^n (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_i^{ip-(n+1)} (ip-n) \alpha_{i,n}
\end{aligned}$$

Diese Aufspaltung der Summen erlaubt nun eine Definition der Koeffizienten α_{ij} .

$$\begin{aligned}
\frac{\partial^{n+1}}{\partial \omega_i^{n+1}} R_p^\cup &= (\omega_1^p + \omega_2^p)^{\frac{1-(n+1)p}{p}} \omega_i^{(n+1)p-(n+1)} \underbrace{(1-np) \alpha_{n+1,n}}_{=:\alpha_{n+1,n+1}} + \\
&\quad + \sum_{i=2}^n (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_i^{ip-(n+1)} \underbrace{((1-(i-1)p) + (ip-n)) \alpha_{i,n}}_{=:\alpha_{i,n+1}} + \\
&\quad + (\omega_1^p + \omega_2^p)^{\frac{1-p}{p}} \omega_i^{p-(n+1)} \underbrace{(p-n) \alpha_{1,n}}_{=:\alpha_{1,n+1}} \tag{2.11} \\
&= \sum_{i=1}^{n+1} (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_i^{ip-n+1} \alpha_{i,n+1}
\end{aligned}$$

Die Nichtdifferenzierbarkeit im Ursprung ergibt sich aus der Darstellung in Polarkoordinaten.

$$\begin{aligned}
\frac{\partial^n}{\partial \omega_1^n} R_p^\cup &= 1 + (r^p \cos^p(\varphi) + r^p \sin^p(\varphi))^{(1-p)/p} (r^{p-1} \sin^{p-1}(\varphi)) (-1) \\
&= 1 - r^{p-1} r^{1-p} (\cos^p(\varphi) + \sin^p(\varphi))^{(1-p)/p} (\sin^{p-1}(\varphi)) \\
&= 1 - (\cos^p(\varphi) + \sin^p(\varphi))^{(1-p)/p} (\sin^{p-1}(\varphi))
\end{aligned}$$

Dies besitzt für $r \rightarrow 0$ keinen eindeutigen Grenzwert. ■

Satz 2.30

Für den Koeffizienten $\alpha_{i,j}$ gilt $\alpha_{1,i} \neq 0$ für $i \leq p$.

Beweis:

Laut (2.11) gilt $\alpha_{1,i} = (p-i+1)\alpha_{1,i-1}$, was wegen (2.10) erst bei $i = p+1$ verschwinden kann. ■

Die Ergebnisse der letzten beiden Sätze lassen sich nun für eine Aussage über die Normalisiertheit verwenden.

Satz 2.31

Die Funktionen in \mathcal{R}_p ist normalisiert zur Ordnung $p - 1$.

Beweis:

Für die erste Ableitung ergibt sich nach (2.10) auf dem Rand

$$\left. \frac{\partial}{\partial \omega_1} R_p \right|_{\omega_1=0, \omega_2>0} = 1 + (\omega_1^p + \omega_2^p)^{\frac{1-p}{p}} \omega_1^{p-1} (-1) \Big|_{\omega_1=0, \omega_2>0}$$

Für alle $n < p$ gilt

$$\left. \frac{\partial^n}{\partial \omega_1^n} R_p \right|_{\omega_1=0, \omega_2>0} = \sum_{i=1}^n (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_2^{\overbrace{ip-n}^{\geq 1}} \alpha_{i,n} \Big|_{\omega_1=0, \omega_2>0} = 0,$$

wohingegen $n = p$ zu

$$\begin{aligned} \left. \frac{\partial^p}{\partial \omega_1^p} R_p \right|_{\omega_1=0, \omega_2>0} &= \sum_{i=1}^p (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_1^{(i-1)p} \alpha_{i,p} \Big|_{\omega_1=0, \omega_2>0} \\ &= (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \omega_1^0 \alpha_{i,p} \Big|_{\omega_1=0, \omega_2>0} \\ &\quad + \sum_{i=2}^p (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \underbrace{\omega_1^{(i-1)p}}_{=0} \alpha_{i,p} \Big|_{\omega_1=0, \omega_2>0} \\ &= (\omega_1^p + \omega_2^p)^{\frac{1-ip}{p}} \alpha_{i,p} \Big|_{\omega_1=0, \omega_2>0} \\ &= \alpha_{i,p} \omega_1^{1-p} \Big|_{\omega_1=0, \omega_2>0} \end{aligned}$$

führt. Dies ist laut Satz 2.30 ungleich Null, die Funktionen in \mathcal{R}_p ist also nur normalisiert zur Ordnung $p - 1$. ■

2.3.6 Funktions-System \mathcal{R}_C^m

Rvachev hat in [Rva67] zwar bewiesen, dass eine \mathcal{R} -Funktion nicht ausschliesslich mit Multiplikation und Addition zu realisieren ist, allerdings ist eine Wurzelberechnung auch nicht notwendig, wie das nun folgende Funktionen-System zeigt.

Definition 2.32

Die \mathcal{R}_C^m -Funktionen sind durch

$$\mathcal{R}_C^m := \left\{ -1; \quad -\omega_1; \quad (\omega_1 + \omega_2)^m \operatorname{sign}(\omega_1 + \omega_2)^{m-1} - (\omega_1 - \omega_2)^m \operatorname{sign}(\omega_1 - \omega_2)^m; \right. \\ \left. (\omega_1 + \omega_2)^m \operatorname{sign}(\omega_1 + \omega_2)^{m-1} + (\omega_1 - \omega_2)^m \operatorname{sign}(\omega_1 - \omega_2)^m \right\}$$

definiert.

Satz 2.33 (Hilfssatz für den Beweis der \mathcal{R} -Funktionseigenschaft (2.34))

Für $\omega_1, \omega_2 \in \mathbb{R}$ und $m \in \mathbb{N}$ gilt $(|\omega_1| + |\omega_2|)^m > (|\omega_1| - |\omega_2|)^m$.

Beweis:

Eine einfache Binomialdarstellung führt auf

$$\begin{aligned}
& (|\omega_1| + |\omega_2|)^m - (|\omega_1| - |\omega_2|)^m \\
&= \sum_{i=0}^m \binom{m}{i} |\omega_1|^{m-i} |\omega_2|^i - \sum_{i=0}^m \binom{m}{i} |\omega_1|^{m-i} (-|\omega_2|)^i \\
&= \sum_{i=0}^m \binom{m}{i} |\omega_1|^{m-i} |\omega_2|^i - (-|\omega_2|)^i \\
&= \sum_{i=0}^m \binom{m}{i} |\omega_1|^{m-i} |\omega_2|^i (1 - (-1)^i) \\
&= \sum_{i=0}^m \binom{m}{i} |\omega_1|^{m-i} |\omega_2|^i \begin{cases} 2 & \text{für } i \text{ ungerade} \\ 0 & \text{für } i \text{ gerade} \end{cases} \\
&> \sum_{i=0}^m \binom{m}{i} |\omega_1|^{m-i} |\omega_2|^i (0) \\
&= 0.
\end{aligned}$$

■

Satz 2.34 (\mathcal{R} -Funktionseigenschaft)

Die \mathcal{R}_C^m -Funktionen sind wohldefinierte \mathcal{R} -Funktionen.

Beweis:

Die Wohldefiniertheit ist klar. Laut 2.11 ist die Betrachtung der beiden Fälle $x \in \Omega$ und $x \notin \Omega$ hinreichend. Dieser Beweis wird wieder nur für $\mathcal{R}_C^{m \cap}$ geführt, da der Beweis für $\mathcal{R}_C^{m \cup}$ mit denselben Mitteln geführt werden kann.

$x \in \Omega$: Dies entspricht also $\omega > 0$ und $\omega_2 > 0$.

m gerade: Mit Hilfe von Satz 2.33 und $\text{sign}(x)^m = \text{sign}(x)^{m-2}$ ergibt sich

$$\begin{aligned}
\mathcal{R}_C^{m \cap}(\omega_1, \omega_2) &= (\omega_1 + \omega_2)^m \text{sign}(\omega_1 + \omega_2)^{m-1} - (\omega_1 - \omega_2)^m \text{sign}(\omega_1 - \omega_2)^m \\
&= (\omega_1 + \omega_2)^m \text{sign}(\omega_1 + \omega_2) - (\omega_1 - \omega_2)^m \\
&= (\omega_1 + \omega_2)^m - (\omega_1 - \omega_2)^m > 0
\end{aligned}$$

m ungerade: Wie bei geradem m läßt sich mit Satz 2.33 schließen, dass

$$\begin{aligned}
\mathcal{R}_C^m(\omega_1, \omega_2) &= \\
&= (\omega_1 + \omega_2)^m - \text{sign}(\omega_1 - \omega_2)(\omega_1 - \omega_2)^m \\
&\geq (\omega_1 + \omega_2)^m - (\omega_1 - \omega_2)^m \\
&> 0.
\end{aligned}$$

$x \notin \Omega$: Dieser Fall spaltet sich in die folgenden Unterfälle auf.

$\omega_1 < 0, \omega_2 > 0$:

In diesem Fall gilt zunächst natürlich $\omega_1 - \omega_2 < 0$. Die Summe wird mittels einer Fallunterscheidung untersucht.

1. Fall: $\omega_1 + \omega_2 < 0$ Mit Hilfe des Satzes 2.33 ergibt sich hier

$$\begin{aligned}
R_C^{m\cap}(\omega_1, \omega_2) &= \\
&= (\omega_1 + \omega_2)^m \operatorname{sign}(\omega_1 + \omega_2)^{m-1} - (\omega_1 - \omega_2)^m \operatorname{sign}(\omega_1 - \omega_2)^m \\
&= (\omega_1 + \omega_2)^m (-1)^{m-1} - (\omega_1 - \omega_2)^m (-1)^m \\
&= \begin{cases} |\omega_1 + \omega_2|^m (-1) - |\omega_1 - \omega_2|^m & \text{für } m \text{ gerade} \\ -(|\omega_1 + \omega_2|^m) - (-|\omega_1 - \omega_2|^m) (-1)^m & \text{für } m \text{ ungerade} \end{cases} \\
&= -(|\omega_1 + \omega_2|^m + |\omega_1 - \omega_2|^m) < 0.
\end{aligned}$$

2. Fall: $\omega_1 + \omega_2 > 0$

$$\begin{aligned}
R_C^{m\cap}(\omega_1, \omega_2) &= \\
&= (\omega_1 + \omega_2)^m \operatorname{sign}(\omega_1 + \omega_2)^{m-1} - (\omega_1 - \omega_2)^m \operatorname{sign}(\omega_1 - \omega_2)^m \\
&= (\omega_1 + \omega_2)^m - (\omega_1 - \omega_2)^m (-1)^m \\
&= \begin{cases} |\omega_1 + \omega_2|^m - (-|\omega_1 - \omega_2|)^m & \text{für } m \text{ gerade} \\ |\omega_1 + \omega_2|^m - (-|\omega_1 - \omega_2|)^m (-1)^m & \text{für } m \text{ ungerade} \end{cases} \\
&= \begin{cases} (|\omega_2| - |\omega_1|)^m - (-|\omega_1| - |\omega_2|)^m & \text{für } m \text{ gerade} \\ (|\omega_2| - |\omega_1|)^m - (|\omega_1| + |\omega_2|)^m & \text{für } m \text{ ungerade} \end{cases} \\
&= \sum_{i=0}^m \binom{m}{i} |\omega_1|^{m-i} |\omega_2|^i \begin{cases} (-1)^{m-i} - ((-1)^m) & \text{für } m \text{ gerade} \\ (-1)^{m-i} - 1 & \text{für } m \text{ ungerade} \end{cases} \\
&= \sum_{i=0}^m \binom{m}{i} |\omega_1|^{m-i} |\omega_2|^i \begin{cases} 0 & \text{für } m-i \text{ gerade} \\ -2 & \text{für } m-i \text{ ungerade} \end{cases} \\
&< 0
\end{aligned}$$

$\omega_1 > 0, \omega_2 < 0$:

Wird durch vergleichbare Abschätzungen wie oben bewiesen.

$\omega_1 < 0, \omega_2 < 0$:

Auch hier liefert die Anwendung von Satz 2.33

$$\begin{aligned}
R_C^{m\cap}(\omega_1, \omega_2) &= \\
&= (\omega_1 + \omega_2)^m \operatorname{sign}(\omega_1 + \omega_2)^{m-1} - (\omega_1 - \omega_2)^m \operatorname{sign}(\omega_1 - \omega_2)^m \\
&= (-1)^{m-1} ((\omega_1 + \omega_2)^m + (\omega_1 - \omega_2)^m) \\
&= (-1)^{m-1} ((-1)^m (|\omega_1| + |\omega_2|)^m + (-1)^m (|\omega_1| - |\omega_2|)^m) \\
&= (-1)^{2m-1} ((|\omega_1| + |\omega_2|)^m + (|\omega_1| - |\omega_2|)^m) \\
&= -((|\omega_1| + |\omega_2|)^m + (|\omega_1| - |\omega_2|)^m) \\
&< 0.
\end{aligned}$$

■

Für die Beurteilung der Normalisiertheit werden wieder die Ableitungen der Funktionen in \mathcal{R}_C^m benötigt, die im folgenden Satz zusammengefaßt werden.

Satz 2.35 (Ableitungen von \mathcal{R}_C^m)

Die Ableitungen von R_C^m lauten

$$\frac{\partial^n}{\partial \omega_1^n} R_C^{m\cap}(\omega_1, \omega_2) = \frac{m!}{(m-n)!} \begin{cases} R_1^{m-n} & \text{für } n \text{ ungerade} \\ R_C^{m-n\cap} & \text{für } n \text{ gerade} \end{cases} \quad (2.12)$$

und

$$\frac{\partial^n}{\partial \omega_2^n} R_C^{m \cap}(\omega_1, \omega_2) = \frac{m!}{(m-n)!} \begin{cases} R_2^{m-n} & \text{für } n \text{ ungerade} \\ R_C^{m-n \cap} & \text{für } n \text{ gerade} \end{cases}, \quad (2.13)$$

wobei

$$R_1^m : = (\omega_1 + \omega_2)^m \operatorname{sign}(\omega_1 + \omega_2)^m - (\omega_1 - \omega_2)^m \operatorname{sign}(\omega_1 - \omega_2)^{m-1} \quad (2.14)$$

$$R_2^m : = (\omega_1 + \omega_2)^m \operatorname{sign}(\omega_1 + \omega_2)^m + (\omega_1 - \omega_2)^m \operatorname{sign}(\omega_1 - \omega_2)^{m-1} \quad (2.15)$$

gilt.

Beweis:

Der Beweis wird hier nur für die Ableitungen nach dem ersten Argument durchgeführt. Die anderen Ableitungen ergeben sich wieder einmal durch analoge Betrachtungen. Vollständige Induktion:

$n = 1$:

Mit

$$\operatorname{sign}(x)^m = \operatorname{sign}(x)^{m-2} \quad (2.16)$$

folgt

$$\begin{aligned} \frac{\partial}{\partial \omega_1} R_C^{m \cap}(\omega_1, \omega_2) &= \\ &= m \left((\omega_1 + \omega_2)^{m-1} \operatorname{sign}(\omega_1 + \omega_2)^{m-1} - (\omega_1 - \omega_2)^{m-1} \operatorname{sign}(\omega_1 - \omega_2)^m \right) \\ &= \frac{m!}{(m-1)!} \left((\omega_1 + \omega_2)^{m-1} \operatorname{sign}(\omega_1 + \omega_2)^{m-1} - (\omega_1 - \omega_2)^{m-1} \operatorname{sign}(\omega_1 - \omega_2)^{m-2} \right) \\ &= R_1^m. \end{aligned}$$

$(n-1) \rightarrow n$:

Für ungerades n gilt mit (2.14) und (2.16)

$$\begin{aligned} \frac{\partial^n}{\partial \omega_1^n} R_C^m(\omega_1, \omega_2) &= \\ &= \frac{\partial}{\partial \omega_1} \left(\frac{\partial^{n-1}}{\partial \omega_1^{n-1}} R_C^m \right) \\ &= \frac{\partial}{\partial \omega_1} \left(\frac{m!}{(m-n+1)!} R_2^{m-n+1} \right) \\ &= \frac{m!}{(m-n+1)!} \frac{\partial}{\partial \omega_1} \left((\omega_1 + \omega_2)^{m-n+1} \operatorname{sign}(\omega_1 + \omega_2)^{m-n+1} - \right. \\ &\quad \left. (\omega_1 - \omega_2)^{m-n+1} \operatorname{sign}(\omega_1 - \omega_2)^{m-n} \right) \\ &= \frac{m!}{(m-n+1)!} (m-n+1) \left((\omega_1 + \omega_2)^{m-n} \operatorname{sign}(\omega_1 + \omega_2)^{m-n+1} - \right. \\ &\quad \left. (\omega_1 - \omega_2)^{m-n} \operatorname{sign}(\omega_1 - \omega_2)^{m-n} \right) \\ &= \frac{m!}{(m-n)!} R_1^{m-n}(\omega_1, \omega_2). \end{aligned}$$

Für gerades n ergibt (2.15) mit einer ähnlichen Rechnung

$$\frac{\partial}{\partial \omega_1} R_C^{m \cap}(\omega_1, \omega_2) = \frac{m!}{(m-n)!} R_C^{m-n \cap}(\omega_1, \omega_2).$$

■

Satz 2.36

Im Innern des Gebietes gilt $\mathcal{R}_C^m \subset C^{m-1}$.

Beweis:

Abgesehen von den Stellen $\omega_1 = \omega_2$ und $\omega_1 = -\omega_2$ sind die Funktionen natürlich analytisch. Bleibt also die Untersuchung an den Stellen $\omega_1 = \omega_2$. Hier werden die Ableitungen zweier verschiedener Richtungen verglichen.

$n < m$:

Linksseitiger Grenzwert

$$\begin{aligned} & \lim_{\substack{\omega_1 \rightarrow \omega_2 \\ \omega_1 < \omega_2}} \left(\frac{\partial^n}{\partial \omega_1^n} R_C^{m \cap}(\omega_1, \omega_2) \right) = \\ & = \lim_{\substack{\omega_1 \rightarrow \omega_2 \\ \omega_1 < \omega_2}} \frac{m!}{(m-n)!} \begin{cases} R_1^{m-n \cap}(\omega_1, \omega_2) & \text{für } n \text{ ungerade} \\ R_2^{m-n \cap}(\omega_1, \omega_2) & \text{für } n \text{ gerade} \end{cases} \\ & = \lim_{\substack{\omega_1 \rightarrow \omega_2 \\ \omega_1 < \omega_2}} \frac{m!}{(m-n)!} \begin{cases} (\omega_1 + \omega_2)^{m-n} \text{sign}(\omega_1 + \omega_2)^{m-n-1} - \\ (\omega_1 - \omega_2)^{m-n} \text{sign}(\omega_1 - \omega_2)^{m-n} & \text{für } n \text{ ungerade} \\ (\omega_1 + \omega_2)^{m-n} \text{sign}(\omega_1 + \omega_2)^{m-n} - \\ (\omega_1 - \omega_2)^{m-n} \text{sign}(\omega_1 - \omega_2)^{m-n-1} & \text{für } n \text{ gerade} \end{cases} \\ & = \frac{m!}{(m-n)!} (2\omega_1)^{m-n} \end{aligned}$$

Rechtsseitiger Grenzwert

$$\begin{aligned} & \lim_{\substack{\omega_1 \rightarrow \omega_2 \\ \omega_1 > \omega_2}} \left(\frac{\partial^n}{\partial \omega_1^n} R_C^{m \cap}(\omega_1, \omega_2) \right) = \\ & = \lim_{\substack{\omega_1 \rightarrow \omega_2 \\ \omega_1 > \omega_2}} \frac{m!}{(m-n)!} \begin{cases} R_1^{m-n \cap}(\omega_1, \omega_2) & \text{für } n \text{ ungerade} \\ R_2^{m-n \cap}(\omega_1, \omega_2) & \text{für } n \text{ gerade} \end{cases} \\ & = \frac{m!}{(m-n)!} (2\omega_1)^{m-n} \end{aligned}$$

$n = m$:

In diesem Fall ergibt der linksseitige Grenzwert

$$\begin{aligned} & \lim_{\substack{\omega_1 \rightarrow \omega_2 \\ \omega_1 < \omega_2}} \left(\frac{\partial^n}{\partial \omega_1^n} R_C^n(\omega_1, \omega_2) \right) = \\ & = n! \lim_{\substack{\omega_1 \rightarrow \omega_2 \\ \omega_1 < \omega_2}} \begin{cases} (\omega_1 + \omega_2)^0 \text{sign}(\omega_1 + \omega_2)^{-1} - \\ (\omega_1 - \omega_2)^0 \text{sign}(\omega_1 - \omega_2)^0 & \text{für } n \text{ ungerade} \\ (\omega_1 + \omega_2)^0 \text{sign}(\omega_1 + \omega_2)^0 - \\ (\omega_1 - \omega_2)^0 \text{sign}(\omega_1 - \omega_2)^{-1} & \text{für } n \text{ gerade} \end{cases} \\ & = \begin{cases} 0 & \text{für } n \text{ ungerade} \\ n! & \text{für } n \text{ gerade} \end{cases}, \end{aligned}$$

allerdings liefert der rechsseitige Grenzwert

$$\lim_{\substack{\omega_1 \rightarrow \omega_2 \\ \omega_1 > \omega_2}} \left(\frac{\partial^n}{\partial \omega_1^n} R_C^m(\omega_1, \omega_2) \right) = 0. \quad (2.17)$$

Die Grenzwerte stimmen also im Allgemeinen nicht mehr überein, also ist die Funktion nicht weiter differenzierbar. ■

Satz 2.37

Die Funktionen in \mathcal{R}_C^m sind nicht normalisiert.

Beweis:

Schon die erste Ableitung auf dem Rand von Ω liefert

$$\begin{aligned} & \frac{\partial}{\partial \omega_1} R_C^m(\omega_1, \omega_2) \Big|_{\omega_1=0, \omega_2>0} \\ &= \frac{m!}{(m-1)!} R_1^{m-1} \Big|_{\omega_1=0, \omega_2>0} \\ &= m(\omega_1 + \omega_2)^{m-1} \operatorname{sign}(\omega_1 + \omega_2)^{m-1} - (\omega_1 - \omega_2)^{m-1} \operatorname{sign} \omega_1 - \omega_2^{m-2} \Big|_{\omega_1=0, \omega_2>0} \\ &= m\omega_2^{m-1} \operatorname{sign}(\omega_2)^{m-1} - (-\omega_2)^{m-1} \operatorname{sign} -\omega_2^{m-2} \Big|_{\omega_1=0, \omega_2>0} \\ &= \begin{cases} m\omega_2^{m-1} - (-\omega_2^{m-1}) & \text{für } m \text{ gerade} \\ m\omega_2^{m-1} + \omega_2^{m-1} & \text{für } m \text{ ungerade} \end{cases} \\ &= (m+1)\omega_2^{m-1}, \end{aligned}$$

was im Allgemeinen ungleich eins ist. ■

Kapitel 3

web-Eigenschaften der Gewichtsfunktionen

Für die Anwendung der Gewichtsfunktionen aus Kapitel 2 im web-Verfahren müssen die zusätzlichen Bedingungen der Definition 1.12 erfüllt sein, damit die Stabilitätsaussagen der Sätze 1.13 und 1.14 und die Beschränktheit der Galerkin-Matrix in Satz 1.15 nicht verletzt werden.

3.1 Äquivalenz zur Abstandsfunktion

Die erste Bedingung (1.4) ist die geforderte Äquivalenz der Gewichtsfunktion zur Abstandsfunktion. In diesem Abschnitt wird dieser Zusammenhang für alle in Kapitel 2 eingeführten \mathcal{R} -Funktionen bewiesen.

Satz 3.1

Die Funktionen in \mathcal{R}_α sind äquivalent zur Abstandsfunktion.

Beweis:

Die Behauptung ergibt sich durch einfache Abschätzungen der Gewichtsfunktion.

$$\begin{aligned} R_\alpha^\cup &= \frac{1}{1+\alpha} \left(\omega_1 + \omega_2 + \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2} \right) \\ &\leq \frac{1}{1+\alpha} \left(cd + cd + \sqrt{c^2 d^2 + c^2 d^2 + 2\alpha c^2 d^2} \right) \\ &= \frac{2 + \sqrt{2 - 2\alpha}}{1 + \alpha} cd \\ &\leq cd \\ R_\alpha^\cup &\geq \frac{1}{1+\alpha} (\omega_1 + \omega_2) \\ &\geq cd \end{aligned}$$

Mit Hilfe dieses Satzes sind natürlich auch die Funktionen in \mathcal{R}_0 und \mathcal{R}_1 äquivalent zur Abstandsfunktion. ■

Satz 3.2

Die Funktionen in \mathcal{R}_p sind äquivalent zur Abstandsfunktion.

Beweis:

Auch hier führen einfache Abschätzungen innerhalb des Gebietes Ω auf

$$\begin{aligned} R^{p\cup} &= \omega_1 + \omega_2 + (\omega_1^p + \omega_2^p)^{1/p} \\ &\leq cd + cd + (c^p d^p + c^p d^p)^{1/p} \\ &\leq cd \\ R_\alpha^\cup &\geq \omega_1 + \omega_2 \\ &\geq cd. \end{aligned}$$

■

Satz 3.3

Die Funktionen in \mathcal{R}_0^m und \mathcal{R}_C^m sind nicht äquivalent zur Abstandsfunktion.

Die Nichtäquivalenz der Funktionen in \mathcal{R}_0^m und \mathcal{R}_C^m kann mit dem Satz 3.4 gezeigt werden. Dieser Satz zeigt auch einen interessanten Zusammenhang von differenzierbaren Funktionen an Ecken.

Satz 3.4

Sei $\omega \in C^2$ eine Gewichtsfunktion auf $\Omega := \{(x, y) : x \leq 0 \vee y \leq 0\}$. Dann besitzt ω in $(0, 0)$ eine dreifache Nullstelle.

Dieser Satz ist gleichbedeutend mit der Aussage, dass ω nicht äquivalent zur Abstandsfunktion ist und somit nicht normalisiert sein kann.

Beweis:

Durch die Differenzierbarkeitseigenschaften kann Ω als Taylorentwicklung um den Ursprung dargestellt werden.

$$\omega(x, y) = a_1 + a_2x + a_3y + a_4x^2 + a_5xy + a_6y^2 + R_3(x, y)$$

Die Forderung, dass ω eine Gewichtsfunktion ist ergibt zunächst $\omega(0, 0) = a_1 = 0$.

Für $x = 0$ und $y \leq 0$ folgt

$$0 \equiv a_3y + a_6y^2 + o(y^2)$$

und damit $a_3 = 0$ sowie $a_6 = 0$. Analog schließt man, dass $a_2 = a_4 = 0$. Es bleibt also

$$\omega(x, y) = a_5xy + R_3(x, y),$$

oder in Polarkoordinaten dargestellt

$$\omega(r, \varphi) = a_5r^2 \cos(\varphi) \sin(\varphi) + R_3(r, \varphi).$$

Folglich gilt

$$\omega(\varepsilon, \phi) \approx a_5\varepsilon^2 \cos(\varphi) \sin(\varphi)$$

in einer kleinen ε -Umgebung um den Ursprung. Würde also $a_5 \neq 0$ gelten, so führte dies auf einen Vorzeichenwechsel von ω bei $\varphi = 0$. Also muss auch $a_5 = 0$ sein. ■

3.2 Beschränktheit des Gradienten

Die zweite Eigenschaft, die eine Gewichtsfunktion erfüllen muss, damit die web-Basis stabil wird, ist laut Abschnitt 1.3 die Beschränktheit des Gradienten. In diesem Abschnitt wird gezeigt, dass alle in Kapitel 2 vorgestellten \mathcal{R} -Funktionen diese Forderung erfüllen.

In diesem Abschnitt wird natürlich vorausgesetzt, dass die Gradienten der Gewichtsfunktionen ω_1 und ω_2 innerhalb des Gebietes Ω beschränkt sind.

Satz 3.5

Die \mathcal{R}_0 -Funktionen besitzen einen beschränkten Gradienten, falls die Funktionen ω_2 und ω_1 einen beschränkten Gradienten besitzen.

Beweis:

Hier wird nur die Funktion R_0^\cup betrachtet, für welche

$$R_0^\cup(x, y) = R(\omega_2(x, y), \omega_1(x, y))$$

mit $R: (\omega_2, \omega_1) \mapsto \omega_2 + \omega_1 + \sqrt{\omega_2^2 + \omega_1^2}$

gilt. Die Kettenregel liefert

$$\frac{\partial}{\partial x_i} R_0^\cup = \frac{\partial}{\partial \omega_2} R \frac{\partial}{\partial x_i} \omega_2 + \frac{\partial}{\partial \omega_1} R \frac{\partial}{\partial x_i} \omega_1.$$

R hat nun die Ableitungen

$$\frac{\partial}{\partial \omega_i} R = 1 + \frac{\omega_i}{\sqrt{\omega_1^2 + \omega_2^2}}$$

welche sich mittels $\omega_i \geq 0$ im Innern leicht abschätzen lassen.

$$1 \leq \frac{\partial}{\partial \omega_i} R \leq 1 + \frac{\omega_i}{\sqrt{\omega_i^2}} = 1 + \text{sign}(\omega_i) \leq 2$$

Damit gilt

$$\begin{aligned} |\nabla R(\omega_2, \omega_1)|^2 &= \sum_{i=1}^m \left(\frac{\partial}{\partial \omega_2} R(\omega_2, \omega_1) \frac{\partial}{\partial x_i} \omega_2 + \frac{\partial}{\partial \omega_1} R(\omega_2, \omega_1) \frac{\partial}{\partial x_i} \omega_1 \right)^2 \\ &\leq \sum_{i=1}^m \left(\left| \frac{\partial}{\partial \omega_2} R(\omega_2, \omega_1) \right| \left| \frac{\partial}{\partial x_i} \omega_2 \right| + \left| \frac{\partial}{\partial \omega_1} R(\omega_2, \omega_1) \right| \left| \frac{\partial}{\partial x_i} \omega_1 \right| \right)^2 \\ &\leq \left(2 \left| \frac{\partial}{\partial x} \omega_2 \right| + 2 \left| \frac{\partial}{\partial x} \omega_1 \right| \right)^2 + \left(2 \left| \frac{\partial}{\partial y} \omega_2 \right| + 2 \left| \frac{\partial}{\partial y} \omega_1 \right| \right)^2 \\ &\leq 4 \left((c + c)^2 + (c + c)^2 \right) \\ &\leq 32c^2. \end{aligned}$$

Eine Betrachtung von R_0^\cap führt mit ähnlichen Abschätzungen zu demselben Ergebnis. ■

Satz 3.6

Die \mathcal{R}_1 -Funktionen besitzen einen beschränkten Gradienten.

Beweis:

Die Ableitungen der Funktionen in \mathcal{R}_1 sind entweder 1 oder -1 . Somit ist auch an nicht-differenzierbaren Stellen die Beschränktheit des Gradienten als Grenzwert gesichert. ■

Satz 3.7

Die \mathcal{R}^m -Funktionen besitzen einen beschränkten Gradienten.

Beweis:

Wie im vorhergehenden Beweis ist die Beschränktheit der Ableitungen hinreichend für die Beschränktheit des Gradienten.

$$\begin{aligned} \frac{\partial}{\partial \omega_i} R^{m \cap U} &= \left(1 \pm \frac{\omega_i}{\sqrt{\omega_1^2 + \omega_2^2}} \right) (\omega_1^2 + \omega_2^2)^{m/2} \\ &\quad + m \omega_i \left(\omega_1 + \omega_2 \pm \sqrt{\omega_1^2 + \omega_2^2} \right) (\omega_1^2 + \omega_2^2)^{\overbrace{m/2-1}^{\geq 0}} \\ &< c \end{aligned}$$

Dies lässt sich auf die Beschränktheit der Gewichtsfunktionen zurückführen. ■

Satz 3.8

Die \mathcal{R}^p -Funktionen besitzen einen beschränkten Gradienten.

Beweis:

Die Ableitungen der \mathcal{R}^p -Funktionen lauten

$$\begin{aligned} \frac{\partial}{\partial \omega_1} R^{p \cap U} &= 1 \pm \omega_1^{p-1} (\omega_1^p + \omega_2^p)^{1/p-1} \\ &= 1 \pm \frac{\omega_1^{p-1}}{(\omega_1^p + \omega_2^p)^{p-1/p}} \\ &= 1 \pm \frac{\omega_1^{p-1}}{\left(\sqrt[p]{\omega_1^p + \omega_2^p} \right)^{p-1}} \\ &= 1 \pm \left(\frac{\omega_1}{\sqrt[p]{\omega_1^p + \omega_2^p}} \right)^{p-1} \end{aligned}$$

Diese sind natürlich für $(\omega_1, \omega_2) \neq (0, 0)$ beschränkt. Für die Grenzwertbetrachtung werden die Gewichtsfunktionen in Polarkoordinaten dargestellt.

$$\begin{aligned} \frac{\partial}{\partial \omega_1} R^{p \cap U} &= 1 \pm \left(\frac{r \sin \varphi}{\sqrt[p]{r^p \sin^p \varphi + r^p \cos^p \varphi}} \right)^{p-1} \\ &= 1 \pm \left(\frac{r \sin \varphi}{|r| \sqrt[p]{\sin^p \varphi + \cos^p \varphi}} \right)^{p-1} \\ &= 1 \pm \left(\frac{\sin \varphi}{\sqrt[p]{\sin^p \varphi + \cos^p \varphi}} \right)^{p-1}. \end{aligned}$$

Der Betrag dieses Ausdrucks ist für $r \rightarrow 0$ zwar nicht eindeutig, aber beschränkt. ■

Satz 3.9

Die \mathcal{R}_C^m -Funktionen besitzen einen beschränkten Gradienten.

Beweis:

Ist m gerade, so lautet die Ableitung

$$\frac{\partial}{\partial \omega_i} R_C^{m \cap \cup} = \begin{cases} (\omega_1 + \omega_2)^m \pm (\omega_1 - \omega_2)^m & \text{für } (f + g) \geq 0 \\ -(\omega_1 + \omega_2)^m \pm (\omega_1 - \omega_2)^m & \text{für } (f + g) < 0 \end{cases}$$

Die Ableitungen können also keine Singularitäten bilden. Für m ungerade ergibt sich das entsprechende Ergebnis. ■

3.3 Zusammenfassung aller Eigenschaften

In den Kapiteln 2 und 3 wurden umfassend die Eigenschaften der verschiedenen \mathcal{R} -Funktionen beleuchtet. An dieser Stelle werden alle Eigenschaften nochmals zusammengefasst, um einen kompakten Überblick zu ermöglichen.

Funktions-System	\mathcal{R}_0	\mathcal{R}_1	\mathcal{R}_α	\mathcal{R}_0^m	\mathcal{R}_C^m	\mathcal{R}^p
Differenzierbarkeitsordnung in Ω	∞	0	0	∞	$m - 1$	∞
im Ursprung	0	0	0	m	$m - 1$	0
Ordnung der Normalisiertheit	1	∞	1	0	0	$(p - 1)$
Äquivalent zur Abstandsfunktion	✓	✓	✓	-	-	✓
Beschränkter Gradient	✓	✓		✓	✓	✓

Abbildung 3.1: Eigenschaften der \mathcal{R} -Funktionen

Abschließend lässt sich also sagen, dass nur die Funktionen in \mathcal{R}^m und \mathcal{R}_C^m die web-Eigenschaften nicht erfüllen und somit für das in Kapitel 1 geschilderte Problem nicht eignen. Alle anderen Funktionsfamilien besitzen die geforderten Eigenschaften.

Allerdings werden bei anderen Problemen auch andere Anforderungen gestellt. So wird in [Str02] für die Lösung des Plattenproblems Gewichtsfunktionen benötigt, die statt 1.4 die Bedingung

$$\forall x \in \Omega : w(x) \asymp d(x, \partial\Omega)^2$$

erfüllen. Hierfür wären beispielsweise die Funktionen in \mathcal{R}^m mit $m = 2$ geeignet.

Kapitel 4

Grundlagen für die Implementierung

Die Überlegungen der letzten Kapitel eignen sich für die Umsetzung eines Programms, das die Kombination von Gebieten und die daraus resultierenden Gewichtsfunktionen verwaltet und berechnet. Die Aufgabenstellung, die nun in den nächsten Kapiteln behandelt werden soll, ist konkret die Erstellung eines Editors, der die Eingabe von Gebieten und deren Kombinationen erlaubt und auf Grund dieser Daten die passenden Gewichtsfunktionen errechnet. Des Weiteren ist die Randkurve des neu entstehenden Gebietes für die Integration in Kapitel 1 von entscheidender Wichtigkeit.

Diese Arbeit betrachtet Quadriken als berandende Kurven. In diesem Kapitel wird erläutert, wie die Quadriken elegant dargestellt und verarbeitet werden können und wie die Gewichtsfunktionen auf den entstehenden Gebieten zu handhaben sind. Des Weiteren werden noch die Grundlagen geschaffen, um die Randkurven darzustellen, zu berechnen und zu verarbeiten.

4.1 Quadriken und Gewichtsfunktionen

Eine Quadrik kann auf verschiedene Weisen dargestellt werden. Hier soll nun erläutert werden, welche Darstellung gewählt wurde und wie die Randkurve dazu errechnet werden kann.

4.1.1 Darstellung der Quadriken

Eine Quadrik im euklidischen n -dimensionalen Raum lässt sich auf verschiedene Arten darstellen.

Definition 4.1 (Quadrikformen)

Die Koordinatenform einer Quadrik lautet

$$\sum_{i,j=1}^n a_{ij}x_i x_j + 2 \sum_{i=1}^n b_i x_i + c = 0, \quad (4.1)$$

wobei $a_{ij} = a_{ji}$ gelten muss. Mit $\mathbf{x} = [x_1, \dots, x_n]^T$, $A = (a_{ij})|_{i,j=1,\dots,n}$ und $b = [b_1, \dots, b_n]$ lässt sich dies in der Matrixform als

$$F(\mathbf{x}) = \mathbf{x}^T A \mathbf{x} + 2b\mathbf{x} + c = 0 \quad (4.2)$$

schreiben.

Die homogene Matrixform liefert die Zusammenfassung der Terme in (4.2) zu

$$F(\mathbf{X}) = \mathbf{X}^T \underbrace{\begin{bmatrix} A & b^T \\ b & c \end{bmatrix}}_{=:Q} \mathbf{X} = 0 \quad (4.3)$$

mit $\mathbf{X} = [x_1, \dots, x_n, 1]^T$.

Die neu entstandene Matrix Q ist natürlich symmetrisch. Die Koordinaten, die hier eingeführt wurden, entsprechen natürlich den homogenen Koordinaten aus der projektiven Geometrie.

4.1.2 Bestimmung der definierenden Matrizen allgemeiner Quadriken

Bei einer Quadrik in Normalform ist Q eine Matrix in der Grundform (vgl. Kapitel 5). Eine allgemeine Quadrik erhält man durch das Anwenden euklidischer Bewegungen auf diese Grundform. Bei der Implementierung wurde für jede Quadrik ein Satz an euklidischen Bewegungen definiert, der sich einfach aus den Eingabedaten errechnen lässt.

Für die Transformation der definierenden Matrizen werden folgende Koordinatensysteme verwendet.

Definition 4.2

Sei \tilde{Q} die Normalform der definierenden Matrix Q einer Quadrik,

$$\mathbf{X}Q\mathbf{X}^T = 0 \quad (4.4)$$

$$\tilde{\mathbf{X}}\tilde{Q}\tilde{\mathbf{X}}^T = 0, \quad (4.5)$$

$\tilde{\mathbf{X}}$ die Koordinaten des Koordinatensystems bezüglich dessen die Matrix die Normalform annimmt, \mathbf{X} die euklidischen Koordinaten und T eine Transformationsmatrix die $\tilde{\mathbf{X}}$ in \mathbf{X} überführt

$$\mathbf{X} = T\tilde{\mathbf{X}}.$$

Bemerkung 4.3

Für die Transformation der definierenden Matrix der Quadrik gilt

$$\mathbf{X}Q\mathbf{X}^T = 0$$

$$(T\tilde{\mathbf{X}})Q(T\tilde{\mathbf{X}})^T = 0$$

$$\tilde{\mathbf{X}}(TQT^T)\tilde{\mathbf{X}}^T = 0$$

woraus mit (4.5)

$$TQT^T = \tilde{Q} \quad (4.6)$$

$$\Leftrightarrow Q = T^{-1}\tilde{Q}(T^T)^{-1} \quad (4.7)$$

folgt.

Ein Punkt in homogenen Koordinaten hat die Form

$$P_{hom} = [wx \quad wy \quad w].$$

Die inhomogene Form bestimmt sich dann natürlich als

$$P_{inh} = [x \ y].$$

Die bekannten euklidischen Bewegungen in zwei Koordinaten werden hier auf homogene Koordinaten erweitert.

Definition 4.4

Die euklidischen Bewegungen werden mit T_{inh} , die projektiven mit T_{hom} bezeichnet. Die Abbildung

$$\begin{aligned} I : \mathbb{R}^3 &\rightarrow \mathbb{R}^2 \\ [x \ y \ w] &\mapsto \begin{bmatrix} x \\ y \\ w \end{bmatrix} \end{aligned}$$

wird Inhomogenisierung genannt.

Translation: Die Verschiebung um einen definierenden Vektor $v = [v_1, v_2]$

$$\mathbf{x} = V_{inh}(\tilde{\mathbf{x}}) = [\tilde{x}_1 + v_1, \tilde{x}_2 + v_2] \quad (4.8)$$

kann mittels Matrixoperation als

$$\mathbf{X} = V_{hom}(\tilde{\mathbf{X}}) := \begin{bmatrix} 1 & 0 & v_1 \\ 0 & 1 & v_2 \\ 0 & 0 & 1 \end{bmatrix} \tilde{\mathbf{X}} \quad (4.9)$$

geschrieben werden.

Rotation: Es werden hier nur Drehungen um den Ursprung betrachtet. Wie allgemein bekannt ist, lautet die Operatormatrix für den definierenden Drehwinkel ϕ

$$\mathbf{X} = R_{hom}(\tilde{\mathbf{X}}) := \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tilde{\mathbf{X}}. \quad (4.10)$$

Skalierung: Hier werden Skalierungen in beiden Dimensionsrichtungen erforderlich sein. Diese lauten für den definierenden Skalierungsfaktor α , bzw. β

$$\mathbf{X} = S_x(\tilde{\mathbf{X}}) := \begin{bmatrix} \alpha & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tilde{\mathbf{X}} \quad (4.11)$$

bzw.

$$\mathbf{X} = S_y(\tilde{\mathbf{X}}) := \begin{bmatrix} 1 & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tilde{\mathbf{X}} \quad (4.12)$$

Bemerkung 4.5

Die Skalierung des Gewichtes

$$S_z := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \gamma \end{bmatrix}$$

kann auf eine Skalierung der beiden Komponenten zurückgeführt werden:

$$\begin{aligned}
 I\left(S_z\left([x \ y \ \mathbf{w}]^T\right)\right) &= I([x \ y \ \gamma\mathbf{w}]) \\
 &= \begin{bmatrix} x & y \\ \gamma\mathbf{w} & \gamma\mathbf{w} \end{bmatrix}^T \\
 &= I\left(\begin{bmatrix} \frac{1}{\gamma}x & \frac{1}{\gamma}y & \mathbf{w} \end{bmatrix}^T\right) \\
 &= I\left(S_x S_y [x \ y \ \mathbf{w}]^T\right)
 \end{aligned}$$

Scherung: Die Scherung um den definierenden Scherwinkel ψ ist ebenfalls wohlbekannt

$$\mathbf{X} = X_{hom}(\tilde{\mathbf{X}}) = \begin{bmatrix} 1 & \tan(\psi) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tilde{\mathbf{X}} \quad (4.13)$$

Satz 4.6

Die Homogenisierung und die Transformationen eines Punktes sind kommutativ in dem Sinne, dass

$$I \circ T_{hom} = T_{inh} \circ I.$$

Beweis:

Alle oben definierten Bewegungen lassen das Gewicht unverändert, da in der dritten Zeile jeder Bewegungsmatrix der dritte Einheitsvektor steht und somit nur das Gewicht eingerechnet wird. Außer bei der Translation gilt für die Darstellung der Anwendung einer Bewegung also

$$T_{hom}(\mathbf{X}) = \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{w}x \\ \mathbf{w}y \\ \mathbf{w} \end{bmatrix} \quad (4.14)$$

bzw.

$$T_{inh}(\mathbf{x}) = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

Somit ergibt sich

$$I(T_{hom}(\mathbf{X})) = I\left(\begin{array}{c} a_{11}\mathbf{w}x + a_{12}\mathbf{w}y \\ a_{21}\mathbf{w}x + a_{22}\mathbf{w}y \\ \mathbf{w} \end{array}\right) = \begin{pmatrix} a_{11}x + a_{12}y \\ a_{21}x + a_{22}y \end{pmatrix}$$

und

$$T_{inh}(I(\mathbf{X})) = \begin{bmatrix} a_{11}x + a_{12}y \\ a_{21}x + a_{22}y \end{bmatrix}$$

■

4.1.3 Grundformen der Quadriken und notwendige Transformationen

Jede Quadrik wird durch eine symmetrische Matrix definiert. Dies impliziert sechs Freiheitsgrade.

$$Q = \begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix}$$

Allerdings lässt sich stets ein Eintrag auf eins normieren, wie sich aus der impliziten Darstellung (4.4) ergibt

$$\begin{aligned} 0 &= \mathbf{X}^T Q \mathbf{X} \\ \Leftrightarrow 0 &= \frac{1}{a} \mathbf{X}^T Q \mathbf{X} \\ \Leftrightarrow 0 &= \frac{1}{a} \mathbf{X}^T \begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix} \mathbf{X} \\ \Leftrightarrow 0 &= \mathbf{X}^T \begin{bmatrix} 1 & \tilde{b} & \tilde{c} \\ \tilde{b} & \tilde{d} & \tilde{e} \\ \tilde{c} & \tilde{e} & \tilde{f} \end{bmatrix} \mathbf{X}, \end{aligned}$$

wobei o.B.d.A. der Eintrag $a \neq 0$ als Normalisierungsfaktor ausgewählt wurde. Es ergeben sich also fünf Freiheitsgrade. [BKGH97] kann entnommen werden, dass die Determinanten

$$\Delta = \begin{vmatrix} a & b & c \\ b & d & e \\ c & e & f \end{vmatrix} \quad (4.15)$$

$$\delta = \begin{vmatrix} a & b \\ b & d \end{vmatrix} \quad (4.16)$$

Invarianten bzgl. Koordinatentransformation sind und eine Kategorisierung der Quadriken erlauben.

Für die einzelnen Quadriken ergeben sich die nun folgende Fälle.

Spezialfall Gerade Hier gilt $\Delta = 0$. Die Gerade hat nur drei Freiheitsgrade. Die Anwendung einer Rotation und einer Translation auf die definierende Matrix einer 'Einheitsgeraden'

$$y = 0 \quad \Leftrightarrow \quad Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

liefert also schon alle möglichen Geraden.

Parabel Die Normalform der Parabel lautet $y - x^2 = 0$, das bedeutet, dass die definierende Matrix die Grundform

$$Q = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 \end{bmatrix} \quad (4.17)$$

besitzt. Zusätzlich gilt $\delta = 0$ also $a_{11}a_{22} - a_{12}^2 = 0$ oder o.B.d.A. $a_{11} = a_{12}^2/a_{22}$. Es steht also ein weiteres Argument fest. Die verbleibenden vier Freiheitsgrade werden schließlich durch Drehung (1), Translation (2) und Skalierung (in einer Koordinatenrichtung)(1) festgelegt.

Ellipse Die Normalform der Ellipse lautet $1 - x^2 - y^2 = 0$, also muss die definierende Matrix

$$Q = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

lauten. Hier gilt $\delta > 0$, also kommen keine weiteren Einschränkungen zum Tragen, weshalb also fünf Freiheitsgrade abgedeckt werden müssen. Dies geschieht durch Translation (2), Drehung (1) und Skalierung in beiden Koordinatenrichtungen (2).

Hyperbel Die Normalform der Hyperbel lautet $xy - 1 = 0$, das bedeutet, dass die definierende Matrix die Grundform

$$Q = \begin{bmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (4.18)$$

hat. Ebenso wie bei der Ellipse stehen auch hier fünf Freiheitsgrade zur Verfügung, die wegen der einfacheren Verarbeitung hier mit der Translation(2), Scherung(1), Rotation(1) und Skalierung (mit demselben Skalierungsfaktor in beiden Koordinatenrichtungen) (1) gewählt werden.

4.1.4 Gewichtsfunktion der Quadriken

Nachdem nun die implizite Darstellung der Quadriken eingeführt wurde, liegt die Frage nach Gewichtsfunktionen auf der Hand. Diese lassen sich für Quadriken sehr leicht finden.

Satz 4.7 (Gewichtsfunktion einer Quadrik)

Die definierende Gleichung jeder Quadrik

$$F(x, y) = [x \ y \ 1] Q [x \ y \ 1]^T \quad (4.19)$$

ist eine strikte Gewichtsfunktion.

Beweis:

Es ist wohlbekannt, dass die Nullstellen einer Quadrik die oben genannten Figuren sind. Dabei gibt es (entartete Fälle ausgenommen) keine isolierten Nullstellen, vielmehr teilt das Nullstellengebilde den Raum in zwei Teile. Da die definierenden Gleichungen polynomial, also insbesondere stetig und die Nullstellen (nichtentartet!) stets einfach sind, folgt die Behauptung, dass F innerhalb eines Gebietes streng positiv ist und außerhalb streng negativ. ■

4.1.5 Kombination der Quadriken

Die Gewichtsfunktionen der Quadriken lassen sich nun mit den in Kapitel 2 betrachteten \mathcal{R} -Funktionen auf elegante Weise kombinieren.

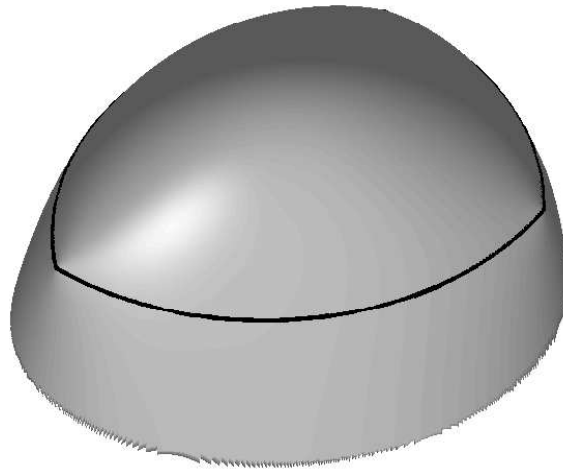


Abbildung 4.1: Die Gewichtsfunktion auf der Schnittmenge dreier Kreise

Beispiel 4.8

Wie schon im Beispiel 2.2 sollen hier drei Einheitskreise miteinander geschnitten werden. Schon dieses einfache Problem führt auf eine beachtliche Gewichtsfunktion, die in Abbildung 4.1 dargestellt wurde.

$$\begin{aligned}
 R_1 &:= 1 - x^2 - y^2 \\
 R_2 &:= 1 - x^2 - (y - 1)^2 \\
 R_3 &:= 1 - (x - 1)^2 - y^2 \\
 R_4 &:= R_1 + R_2 - \sqrt{R_1 + R_2} \\
 R_5 &:= R_3 + R_4 - \sqrt{R_3 + R_4} \\
 \Rightarrow R_5 &= 1 - 3x^2 - 3y^2 + 2x + 2y - \sqrt{-2x^2 + 2x - 2y^2 + 2y} - \\
 &\quad \sqrt{1 - 3x^2 - 3y^2 + 2x + 2y - \sqrt{-2x^2 + 2x - 2y^2 + 2y}}
 \end{aligned}$$

Insbesondere die Berechnung der Ableitungen ist alles andere als trivial und nicht für eine Automatisierung geeignet. Aus diesem Grund wird in Kapitel 5.1 ein besserer Weg für die Verwaltung gewählt.

4.2 Berechnung der Randkurve

Für die Berechnung der Integrale in der Galerkin-Matrix werden die Randkurven der Gebiete benötigt, über die integriert wird.

4.2.1 Definition der Bézierkurven

Die Randkurve des Gebietes Ω wird aus stückweise gebrochenrationalen quadratischen Bézierkurven aufgebaut. Sie stellen eine Erweiterung der Bézierkurven dar.

Definition 4.9 (Definition der Bézierkurven)

Mittels

$$b_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

wird durch

$$B(t) = \sum_{i=0}^n P_i b_i^n(t)$$

eine Bézierkurve definiert, wobei die $P_i = [P_{i1} \ P_{i2}]$ die Eckpunkte eines Kontrollpolygons darstellen. Eine gebrochenrationale Bézierkurve hat die Gestalt

$$c(t) = \frac{\sum_{i=0}^n P_i w_i b_i^n(t)}{\sum_{i=0}^n w_i b_i^n(t)} \quad (4.20)$$

mit den Gewichten $\prod_{i=0}^n w_i \neq 0$.

Bemerkung 4.10

Die Bézierkurven werden auf $t \in [0, 1]$ eingeschränkt, was durch die Wahl der Kontrollpunkte jederzeit möglich ist. Des Weiteren sind Bézierkurven und somit auch gebrochenrationale Bézierkurven invariant gegenüber Bewegungen. Schließlich ist noch zu bemerken, dass die $b_i^n(t)$ eine Basis der Bézierkurven bilden. Die Beweise hierfür - sowie weiterführende Betrachtungen, auf die hier nicht eingegangen werden kann - können in [Rei98, Rei99] oder in [HL89] bzw. [dB78] gefunden werden.

4.2.2 Darstellung von Quadriken mittels Bézierkurven

Im weiteren werden hilfreiche Zusammenhänge zwischen Quadriken und Bézierkurven behandelt, welche die gemeinsame Einführung dieser beiden Objekte rechtfertigen.

Satz 4.11

Für jede beliebige quadratische Kurve

$$\begin{aligned} C : \mathbb{R} &\rightarrow \mathbb{R}^3 \\ t &\rightarrow \begin{bmatrix} x(t) \\ y(t) \\ w(t) \end{bmatrix} \end{aligned}$$

existiert eine nichtentartete Quadrik Q , die (4.4) erfüllt.

Beweis:

Es gilt

$$C(t)^T Q C(t) = \sum_{i=0}^4 l_i(a, \dots, f) t^i,$$

wobei $l_i(a, b, c, d, e, f)$ geeignete Linearkombinationen der Einträge der definierenden Matrix Q sind. Dieses Polynom soll nun laut (4.4) Null sein, also müssen die Koeffizienten l_i allesamt verschwinden. Dies führt auf fünf Gleichungen bei sechs Unbekannten, es existiert also eine nichttriviale Lösung, welche die gesuchte Quadrik liefert. ■

Satz 4.12

Jede quadratische rationale Bézierkurve ist Teil einer Quadrik.

Beweis:

Die Bézierkurve lässt sich in homogenen Koordinaten als

$$C(t) = \begin{bmatrix} x(t) \\ y(t) \\ \mathbf{w}(t) \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^2 P_{i1} \mathbf{w}_i b_i^2(t) \\ \sum_{i=0}^2 P_{i2} \mathbf{w}_i b_i^2(t) \\ \sum_{i=0}^2 \mathbf{w}_i b_i^2(t) \end{bmatrix}$$

schreiben, denn

$$I(C(t)) = \begin{bmatrix} c_1(t) \\ c_2(t) \end{bmatrix}.$$

Jede Komponente ist also eine quadratische Kurve und mit Satz 4.11 existiert eine Quadrik, sodass (4.4) erfüllt ist. Das heißt, dass die Bézierkurve die definierende Gleichung für alle $t \in [0, 1]$ erfüllt, sie also in dem Kegelschnitt liegt. ■

Die quadratischen Bézierkurven stellen also eine Untermenge der Quadriken dar. Der folgende Satz zeigt die Umkehrung der Inklusion, weshalb sie sich besonders für deren Darstellung eignen.

Satz 4.13

Jeder Teil einer nichtentarteten Quadrik lässt sich als rationale Bézierkurve darstellen.

Beweis:

Die Quadriken können mittels Hauptachsentransformation auf eine der Normalformen $y^2 + x^2 - 1 = 0$, $xy - 1 = 0$ oder $y - x^2 = 0$ gebracht werden. Für diese können elementare Bézierkurven angegeben werden (s. Kapitel 4.2.3). Die Rücktransformation der Bézierkurven liefert die allgemeine Quadrik, da diese invariant bezüglich affinen Bewegungen ist. ■

Also ist die Menge der Quadriken gleich der Menge der durch rationale Bézierkurven darstellbaren Kurven. Die nun folgenden Sätze werden für die Umwandlung der Quadriken in Bézierkurven benötigt.

Satz 4.14 (Tangenten an Quadriken)

Die Tangente an die Quadrik mit der definierenden Matrix Q im Punkt $[x_0 \ y_0 \ \mathbf{w}_0]^T$ ist gegeben durch $[x \ y \ \mathbf{w}]Q[x_0 \ y_0 \ \mathbf{w}_0]^T$.

Beweis:

Die Randpunkte einer Quadrik erfüllen laut (4.4) die Bedingung

$$\mathbf{X}^T Q \mathbf{X} = 0. \tag{4.21}$$

Eine Tangente an die Quadrik durch den Punkt X_0 kann mittels

$$\mathbf{X} = \mathbf{X}_0 + \alpha R, \quad \alpha \in \mathbb{R} \tag{4.22}$$

mit einem geeigneten Vektor R definiert werden. Die Tangente soll die Quadrik nun in genau einem Punkt schneiden. Die Schnittpunktbedingung ((4.22) in (4.21) eingesetzt) liefert

$$\begin{aligned} & (\mathbf{X}_0 + \alpha R)^T Q (\mathbf{X}_0 + \alpha R) = 0 \\ \Leftrightarrow & \mathbf{X}_0^T Q \mathbf{X}_0 + \alpha R^T Q \mathbf{X}_0 + \alpha \mathbf{X}_0^T Q R + \alpha^2 R^T Q R = 0 \\ \Leftrightarrow & \mathbf{X}_0^T Q \mathbf{X}_0 + \alpha R^T Q \mathbf{X}_0 + \alpha \mathbf{X}_0^T Q R + \alpha^2 R^T Q R = 0 \\ \Leftrightarrow & \mathbf{X}_0^T Q \mathbf{X}_0 + 2\alpha \mathbf{X}_0^T Q R + \alpha^2 R^T Q R = 0, \end{aligned}$$

wobei die letzte Äquivalenz wegen der Symmetrie der definierenden Matrix gilt. Da \mathbf{X}_0 ein Punkt auf der Quadrik ist, gilt wegen (4.21)

$$\mathbf{X}_0^T Q \mathbf{X}_0 = 0. \quad (4.23)$$

Also folgt

$$\begin{aligned} \mathbf{X}_0^T Q \mathbf{X}_0 + 2\alpha \mathbf{X}_0^T Q R + \alpha^2 R^T Q R &= 0 \\ \Leftrightarrow \alpha(2\mathbf{X}_0^T Q R + \alpha R^T Q R) &= 0. \end{aligned}$$

Damit diese Gleichung genau eine Lösung für α besitzt, muss $\alpha = 0$ die einzige Lösung sein. Mit

$$\begin{aligned} 2\mathbf{X}_0 Q R^T + \alpha R Q R^T &= 0 \\ \Leftrightarrow \alpha &= \frac{2\mathbf{X}_0 Q R^T}{R Q R^T} \end{aligned}$$

kann dies nur für

$$\mathbf{X}_0^T Q R = 0 \quad (4.24)$$

der Fall sein.

Eine Erweiterung von (4.22) um $Q \mathbf{X}_0$ liefert gemeinsam mit (4.23) und (4.24)

$$\begin{aligned} \mathbf{X}^T Q \mathbf{X}_0 &= (\mathbf{X}_0 + \alpha R)^T Q \mathbf{X}_0^T \\ &= \mathbf{X}_0^T Q \mathbf{X}_0 + \alpha R^T Q \mathbf{X}_0 \\ &= 0. \end{aligned}$$

Somit ist die Behauptung bewiesen. ■

Das folgende Verfahren erlaubt die Berechnung der berandenden Bézierkurve. Drei Punkte der Bézierkurve müssen bekannt sein. Diese werden $c(0)$, $c(1)$ und $c(\infty)$ zugeordnet.

1. Setze $P_1 := c(0)$ und $P_3 := c(1)$
2. Bestimme P_2 als Schnittpunkt der Tangenten in $c(0)$ und $c(1)$ mittels (4.14):

$$\begin{aligned} [P_{11} \ P_{12} \ 1] Q \begin{bmatrix} P_{21} \\ P_{22} \\ 1 \end{bmatrix} &= 0 \\ [P_{31} \ P_{32} \ 1] Q \begin{bmatrix} P_{21} \\ P_{22} \\ 1 \end{bmatrix} &= 0 \end{aligned}$$

Dies lässt sich zu

$$\begin{bmatrix} P_1 & 1 \\ P_3 & 1 \end{bmatrix} Q \begin{bmatrix} P_2^T \\ 1 \end{bmatrix} = 0 \quad (4.25)$$

zusammenführen.

3. Der Grenzwert der Kurve lautet

$$\begin{aligned} c(\infty) &= \lim_{t \rightarrow \infty} c(t) \\ &= \lim_{t \rightarrow \infty} \frac{w_1 P_1 (1-t)^2 + 2w_2 P_2 t(1-t) + w_3 P_3 t^2}{w_1 (1-t)^2 + 2w_2 t(1-t) + w_3 t^2} \\ &= \frac{w_1 P_1 - 2w_2 P_2 + w_3 P_3}{w_1 - 2w_2 + w_3} \end{aligned}$$

Damit gilt

$$\begin{aligned} c(\infty)(w_1 - 2w_2 + w_3) &= w_1 P_1 - 2w_2 P_2 + w_3 P_3 \\ (c(\infty) - P_1)w_1 - 2(c(\infty) - P_2)w_2 + (c(\infty) - P_3)w_3 &= 0. \end{aligned}$$

Die Gewichte W lassen sich also als nichttriviale Lösung von

$$\begin{bmatrix} c(\infty) - P_1 \\ -2(c(\infty) - P_2) \\ c(\infty) - P_3 \end{bmatrix}^T W = 0 \quad (4.26)$$

bestimmen.

4.2.3 Bézierkurven zu Quadriken in Normalform

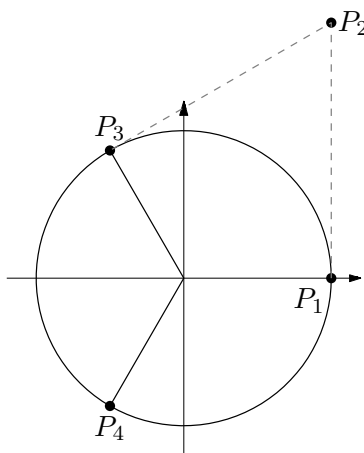


Abbildung 4.2: Unterteilung einer Ellipse in Bézierkurvenstücke

Ellipse Die Normalform der Ellipse ist der Einheitskreis. Auf diesem werden die drei Punkte als die Eckpunkte eines gleichseitigen Dreiecks gewählt (s. Abbildung 4.2). Diese in obiges Verfahren eingesetzt ergeben

1.

$$\begin{aligned} P_1 &:= c(0) = [1 \quad 0] \\ P_3 &:= c(1) = \left[-\frac{1}{2} \quad \frac{1}{2}\sqrt{3}\right] \\ P_4 &:= c(\infty) = \left[-\frac{1}{2} \quad -\frac{1}{2}\sqrt{3}\right] \end{aligned}$$

2. Diese Punkte in (4.25) eingesetzt ergibt

$$\begin{aligned} \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{1}{2}\sqrt{3} & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_2^T \\ 1 \end{bmatrix} &= 0 \\ \begin{bmatrix} -1 & 0 & 1 \\ \frac{1}{2} & -\frac{1}{2}\sqrt{3} & 1 \end{bmatrix} \begin{bmatrix} P_2^T \\ 1 \end{bmatrix} &= 0 \\ P_2 &= [1 \quad \sqrt{3}]. \end{aligned}$$

3. Alle Punkte lassen sich schließlich in (4.26) verwenden.

$$\begin{aligned} \begin{bmatrix} P_4 - P_1 \\ -2(P_4 - P_2) \\ P_4 - P_3 \end{bmatrix}^T W &= 0 \\ \begin{bmatrix} -\frac{3}{2} & -\frac{1}{2}\sqrt{3} \\ -2(-\frac{3}{2} & -\frac{3}{2}\sqrt{3}) \\ 0 & -\sqrt{3} \end{bmatrix}^T W &= 0 \\ \begin{bmatrix} -\frac{3}{2} & 3 & 0 \\ -\frac{1}{2}\sqrt{3} & 3\sqrt{3} & -\sqrt{3} \end{bmatrix} W &= 0 \\ W &= [\alpha \quad \frac{1}{2}\alpha \quad \alpha] \quad \text{mit } \alpha \in \mathbb{R} \end{aligned}$$

Hier wurde nun $\alpha = 1$, d.h.

$$W = [1 \quad \frac{1}{2} \quad 1]$$

gewählt. Wegen der Invarianz der Bézierkurven gegenüber Bewegungen bleiben die Gewichte erhalten, wenn man die Punkte zyklisch vertauscht. So ergeben sich die anderen Stücke des Kreises, indem die erste berechnete Kurve um $\frac{2}{3}\pi$ bzw. $\frac{4}{3}\pi$ um den Ursprung gedreht wird. Die Darstellung der allgemeinen Ellipse ergibt sich nun also aus der Anwendung der Transformationsmatrix in (4.5).

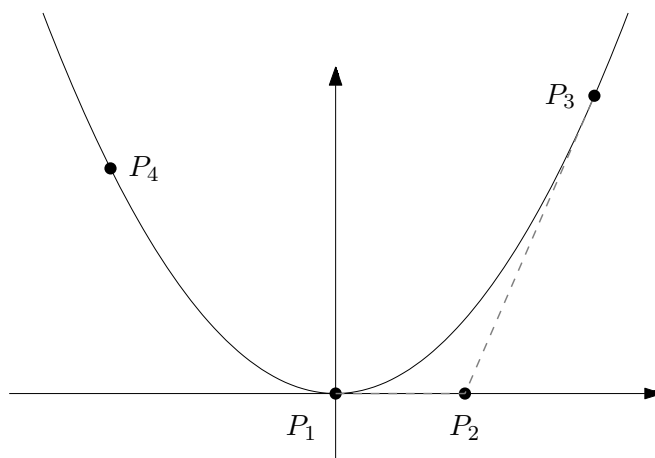


Abbildung 4.3: Unterteilung einer Parabel in Bézierkurvenstücke

Parabel Die Normalform der Parabel ist die Einheitsparabel $y = x^2$. Auf dieser werden der Scheitel und ein weiterer Punkt gewählt. Der letzte (Fern-)Punkt kann als $P_4 := [-1, 1]$ gewählt werden (s. Abbildung 4.3).

1. Die Punkte lauten also

$$\begin{aligned} P_1 &:= c(0) = [0 \ 0] \\ P_3 &:= c(1) = [P_{31} \ P_{31}^2] := [P_{31} \ P_{31}^2] \\ P_4 &:= c(\infty) = [-1 \ 1]. \end{aligned}$$

2. Der zweite Punkt ergibt sich aus (4.25)

$$\begin{aligned} \begin{bmatrix} 0 & 0 & 1 \\ P_{31} & P_{31}^2 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 \end{bmatrix} \begin{bmatrix} P_2^T \\ 1 \end{bmatrix} &= 0 \\ \begin{bmatrix} 0 & \frac{1}{2} & 0 \\ -P_{31} & \frac{1}{2} & \frac{1}{2}P_{31}^2 \end{bmatrix} \begin{bmatrix} P_2^T \\ 1 \end{bmatrix} &= 0 \\ P_2 &= [\frac{1}{2}P_{31} \ 0]. \end{aligned}$$

3. Für die Gewichte ergibt sich also

$$\begin{aligned} \begin{bmatrix} P_4 - P_1 \\ -2(P_4 - P_2) \\ P_4 - P_3 \end{bmatrix}^T W &= 0 \\ \begin{bmatrix} -1 & 1 \\ -2(-1 - \frac{1}{2}P_{31}) & -2(1 - 0) \\ -1 - P_{31} & 1 - P_{31}^2 \end{bmatrix}^T W &= 0 \\ \begin{bmatrix} -1 & 2 + P_{31} & -1 - P_{31} \\ 1 & -2 & 1 - p^2 \end{bmatrix} W &= 0, \end{aligned}$$

was auf

$$W = [(1 + P_{31})^2\beta \ (1 + P_{31})\beta \ \beta] \quad \text{mit } \beta \in \mathbb{R}$$

führt.

Hier wurde nun $\beta = 1$, d.h.

$$W = [(1 + P_{31})^2 \ (1 + P_{31}) \ 1]$$

gewählt. Den zweiten Ast der Parabel erhält man durch Spiegelung des ersten Astes, wobei dies wegen der affinen Invarianz durch die Spiegelung der Kontrollpunkte realisierbar ist. Die allgemeine Parabel ergibt sich - wie schon bei der Ellipse - durch Anwendung der Transformationsmatrix in (4.5).

Hyperbel Die Normalform der Hyperbel ist die Einheitshyperbel $y = \frac{1}{x}$. Auf dieser werden der Scheitel und ein weiterer Punkt gewählt. Der letzte (Fern-)Punkt kann als $P_4 := [-1, -1]$ gewählt werden (s. Abbildung 4.3).

1. Die Punkte lauten also

$$\begin{aligned} P_1 &:= c(0) = [1 \ 1] \\ P_3 &:= c(1) = [P_{31} \ \frac{1}{P_{31}}] \\ P_4 &:= c(\infty) = [-1 \ -1]. \end{aligned}$$

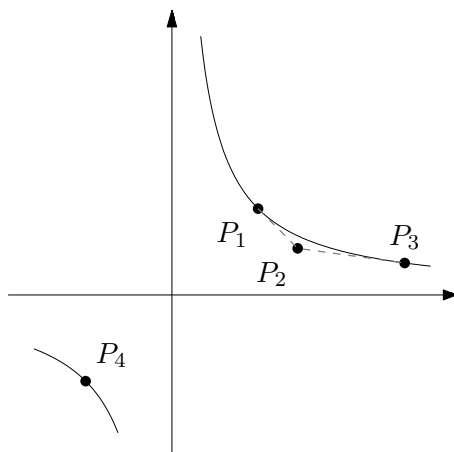


Abbildung 4.4: Unterteilung einer Hyperbel in Bézierkurvenstücke

2. Der zweite Punkt ergibt sich laut (4.25) zu

$$\begin{aligned} \begin{bmatrix} 1 & 1 & 1 \\ P_{31} & \frac{1}{P_{31}} & 1 \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} P_2^T \\ 1 \end{bmatrix} &= 0 \\ \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & -1 \\ \frac{1}{2P_{31}} & \frac{1}{2}P_{31} & -1 \end{bmatrix} \begin{bmatrix} P_2^T \\ 1 \end{bmatrix} &= 0 \\ P_2 &= \begin{bmatrix} \frac{2P_{31}}{1+P_{31}} & \frac{2}{1+P_{31}} \end{bmatrix}. \end{aligned}$$

3. Schließlich lauten die Gewichte

$$\begin{aligned} \begin{bmatrix} P_4 - P_1 \\ -2(P_4 - P_2) \\ P_4 - P_3 \end{bmatrix}^T W &= 0 \\ \begin{bmatrix} -2 & -2 \\ -2(-1 - \frac{2P_{31}}{1+P_{31}}) & -2(-1 - \frac{2}{1+P_{31}}) \\ -1 - P_{31} & -1 - \frac{1}{P_{31}} \end{bmatrix}^T W &= 0 \\ \begin{bmatrix} -2 & \frac{2}{1+P_{31}}(3P_{31} + 1) & -1 - P_{31} \\ -2 & 2 & \frac{1}{P_{31}}(P_{31} + 1) \end{bmatrix} W &= 0, \end{aligned}$$

also

$$W = \left[\frac{(1+P_{31})^2}{4P_{31}}\gamma \quad \frac{(1+P_{31})^2}{4P_{31}}\gamma \quad \gamma \right] \quad \text{mit } \gamma \in \mathbb{R}.$$

Hier wurde nun $\gamma = 4P_{31}$ gesetzt, d.h.

$$W = \left[(1+P_{31})^2 \quad (1+P_{31})^2 \quad 4P_{31} \right].$$

Wegen der Invarianz der Bézierkurven gegenüber Bewegungen bleiben die Gewichte erhalten, wenn man die Punkte am Ursprung spiegelt. Eine Darstellung der Hyperbel liefert nun eine Anwendung der Transformationsmatrix in (4.5).

4.2.4 Schnitt von Bézierkurven

Bei der Ermittlung von Randkurven von zu kombinierenden Gebieten werden natürlich der Schnitt dieser Randkurven benötigt. Die Schnittbedingung für die quadratischen rationalen Bézierkurven $c(t)$ und $d(s)$ lautet

$$c(t) = d(s) \quad (4.27)$$

Dies ist eine Gleichung in zwei Komponenten mit zwei Unbekannten. Sie ist natürlich polynomial in beiden Unbekannten und kann deshalb in jeder Komponente auf die Form

$$\begin{bmatrix} t^0 & t^1 & t^2 \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} s^0 \\ s^1 \\ s^2 \end{bmatrix} = 0 \quad (4.28)$$

gebracht werden. Darauf lässt sich nun folgendes Verfahren anwenden, wie es in [Hö98] zu finden ist.

Satz 4.15 (Resultante)

Die Polynome $p(s) = a_0(t) + a_1(t)s + a_2(t)s^2$ und $q(s) = b_0(t) + b_1(t)s + b_2(t)s^2$ haben genau dann eine gemeinsame Nullstelle, wenn die Resultante

$$r(p, q) = \det \begin{bmatrix} a_0 & a_1 & a_2 & 0 \\ 0 & a_0 & a_1 & a_2 \\ b_0 & b_1 & b_2 & 0 \\ 0 & b_0 & b_1 & b_2 \end{bmatrix} \quad (4.29)$$

verschwindet.

Der Beweis des Satzes ist in [vdW73] zu finden, die nun folgende algorithmische Vorgehensweise wird in [Hö98] beschrieben.

Satz 4.15 lässt sich auf (4.28) anwenden. Die Schnittpunkte lassen sich einfach durch Nullsetzen der Determinate in (4.29) berechnen.

Diese Ermittlung der Determinaten lässt sich noch durch Umgehung der symbolischen Berechnung verbessern. Der Grad des Polynoms r ist maximal 8, da maximal 4 Polynome vom Grad 2 multipliziert werden. Also ist das Polynom durch seine Werte an 9 Stellen bereits vollständig definiert. Die Wahl der Stellen ist natürlich beliebig, weshalb hier

$$t_k = e^{2\pi ik/16} \quad (4.30)$$

gewählt werden kann. Somit gilt

$$a_i(t) = \sum_{j=0}^2 A_{ji} t^j \quad (4.31)$$

$$a_i(t_k) = \sum_{j=0}^2 A_{ji} \exp(2\pi ijk/16) \quad (4.32)$$

$$= \sum_{j=0}^{15} A_{ji} \exp(2\pi ijk/16), \quad (4.33)$$

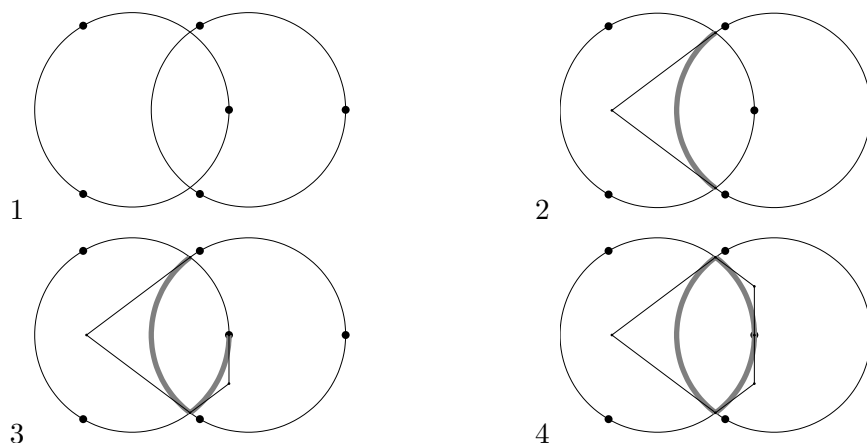


Abbildung 4.5: Sukzessive Berechnung der Randkurve beim Schnitt zweier Kreise

wobei (4.33) der diskreten Fouriertransformation des Polynoms $a_i(t)$ entspricht, mit $A_{ij} = 0$ für $j \geq 3$. Diese lässt sich also sehr einfach und schnell berechnen. Wird nun (4.33) sowohl für a_i als auch für b_i jedes k berechnet, so ergeben sich die Werte der Resultante $r(t_k)$. Die Auswertung an 16 Stellen statt nur an 9 wird wegen der einfacheren Behandlung von Punktzahlen, die einer Zweierpotenz entsprechen, gewählt. Eine Interpolation liefert nun die Koeffizienten von r zurück. Da allerdings für r und r_k derselbe Zusammenhang wie in (4.33) besteht, ist die Interpolation gleichbedeutend mit der inversen Fouriertransformation. Also ist auch dieser Schritt wieder sehr günstig durchzuführen.

Als letztes berechnet man also noch die Nullstellen von r , sortiert die nicht reellen aus und wählt die im Intervall $[0, 1]$ vorhandenen aus.

4.2.5 Randkurve eines kombinierten Gebietes

Die Randkurve eines Gebietes besteht aus mehreren Bézierkurven. Für die Berechnung einer Kombination von Kurven wird folgender Algorithmus verwendet.

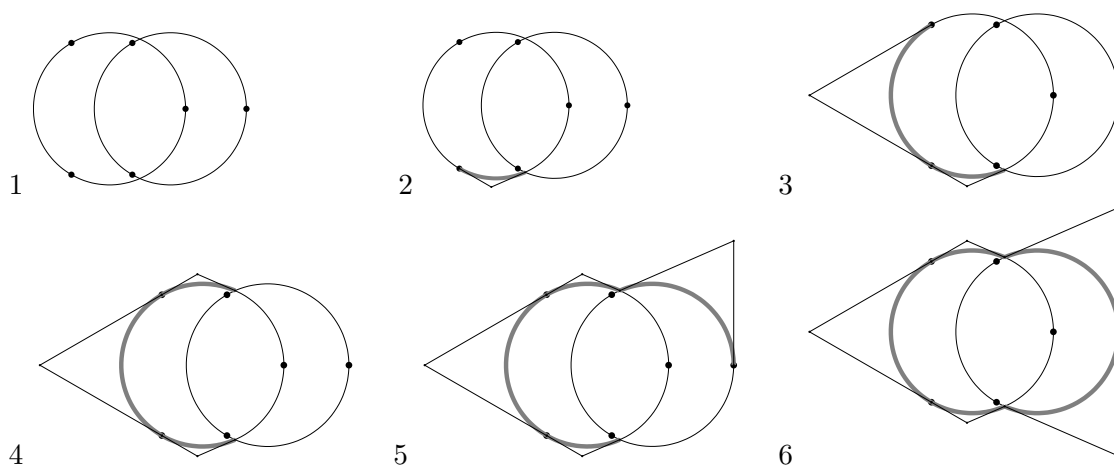


Abbildung 4.6: Sukzessive Berechnung der Randkurve bei der Vereinigung zweier Kreise

Ausgangspunkt ist ein beliebiger Schnittpunkt, der durch systematische Auswertung des Schnittes aller Bézierkurven ermittelt wird. Zu den beiden aus diesem Punkt herauslaufenden Kurvenstücken werden die Tangenten in diesem Schnittpunkt berechnet. Die Lage der Tangenten zueinander erlaubt die Wahl einer Kurve, auf der weiter gelaufen wird. Bei der Schnittbildung wird bestimmt, welcher Tangentenvektor in die linke Halbebene der durch den anderen Vektor aufgespannten Gerade zeigt. Dieser gehört zu dem Kurvenstück, auf dem weiter gelaufen wird. Die Schnittpunkte dieses Stückes mit allen anderen Bézierkurvenstücken wird ermittelt. Falls mindestens ein Schnittpunkt vorhanden ist, wird weiter verfahren wie zu Beginn. Liegt kein weiterer Schnittpunkt auf dem Kurvenstück, so muss auf das nächste Kurvenstück gesprungen werden, auf dem dann wieder die Schnittpunktbestimmung durchgeführt wird. Dies wird solange wiederholt, bis der Startpunkt wieder erreicht wird.

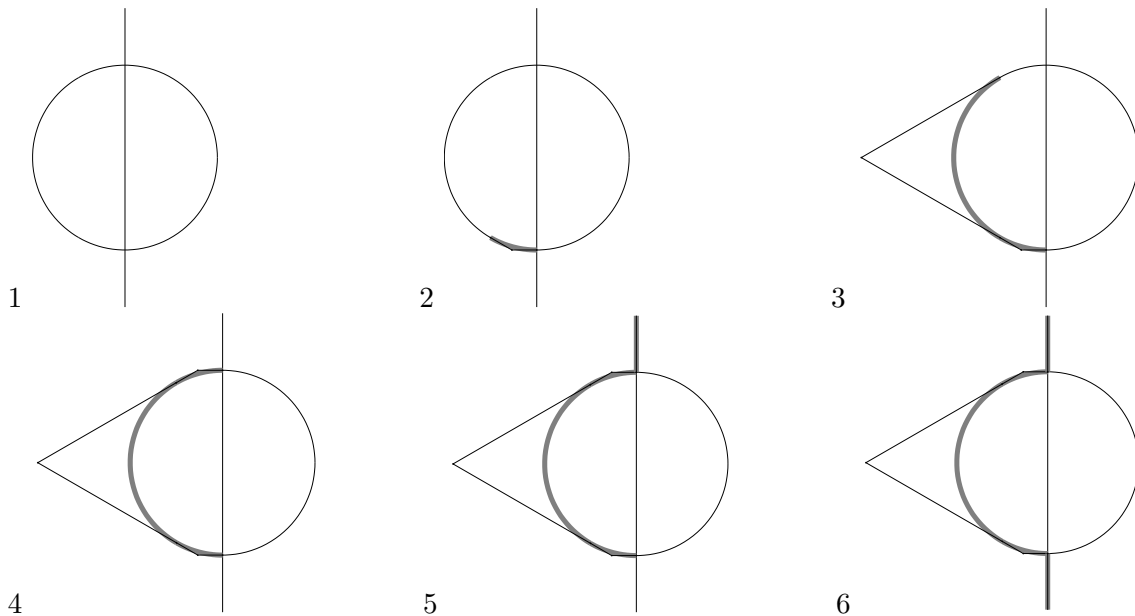


Abbildung 4.7: Sukzessive Berechnung der Randkurve bei der Vereinigung von Kreis und Gerade

In Abbildung 4.5 sind die einzelnen Schritte bei der Bestimmung des Schnittes zweier Kreise, die selbst aus drei Bézierkurven bestehen, dargestellt. Die Teilkurven sind durch die Markierungen auf den Kreisen begrenzt (vgl. Abschnitt 4.2.3). Die durch den Algorithmus festgestellten Kurvenstücke sind mitsamt Kontrollpolygon dargestellt.

In der Abbildung 4.6 ist der Algorithmus noch einmal an dem Beispiel der Vereinigung zweier Kreise dargestellt. Die Vereinigung zweier Gebiete wird mittels $A \cup B = (A^C \cap B^C)^C$ auf die Schnittbildung zurückgeführt, wobei A^C das Komplement von A ist.

Bei dieser Vorgehensweise muss noch beachtet werden, dass während der Konstruktion auch nicht berandete Gebiete entstehen können (siehe Abbildung 4.7). Falls obiger Algorithmus also auf einer Bézierkurve durch den Rand des Darstellungsfensters tritt, so wird der Algorithmus am Rand abgebrochen und die gesamte Berechnung im ursprünglichen

Startpunkt nochmals gestartet, allerdings in umgekehrter Richtung, um den 'Beginn' der Kurve, die in diesen Schnittpunkt mündet, zu berechnen.

In Abbildung 4.7 ist der Startpunkt an der Unterseite des Kreises zu erkennen, von dem aus die Randkurve nach oben entlanggelaufen wird. An der oberen Fenterseite angekommen, wird beim Startpunkt erneut rückwärts gestartet.

Kapitel 5

Umsetzung der Grundlagen

Nachdem in Kapitel 4 die Grundlagen für die Implementierung erläutert wurden, soll in diesem Kapitel die eigentliche Umsetzung dargestellt werden. So soll die Berechnung der definierenden Parameter von Quadriken aus Punkten, die sich einfach interaktiv eingeben lassen, ermöglicht werden. Auch eine Matrizenformulierung des Schnittes zweier Bézierkurven für eine effizientere Berechnung wird hier gefasst. Schließlich wird noch der QBE vorgestellt, in dem die Erkenntnisse des letzten Kapitels umgesetzt wurden.

5.1 Implementierung der Gewichtsfunktion

In der Implementierung wurde der Ansatz gewählt, dass eine Funktion erzeugt wird, welche die Gewichtsfunktion realisiert. Die Ableitungen werden mittels automatischer Differenziation weitergeführt.

Bemerkung 5.1

Allgemein gilt für die Ableitung einer \mathcal{R} -Funktion für $m > k, l$

$$\frac{\partial^n}{\partial f^n} R_m(R_k, R_l) = \frac{\partial^n}{\partial f^n} R_m \Big|_{R_k, R_l} \frac{\partial^n}{\partial f^n} R_k + \frac{\partial^n}{\partial g^n} R_m \Big|_{R_k, R_l} \frac{\partial^n}{\partial f^n} R_l. \quad (5.1)$$

Also können beliebige Ableitungen auf Ableitungen von \mathcal{R} -Funktionen mit niedrigerem Index zurückgeführt werden. Ebenso bietet sich die Möglichkeit, in jedem Schritt auch alle Ableitungen mitzuberechnen, wodurch sich Ableitungen von Funktionen mit einem höheren Index mithilfe dieser 'Ableitungstabelle' einfach berechnen lassen.

Das folgende Beispiel, das bereits in Abschnitt 2.2 eingeführt wurde, soll dies verdeutlichen. Mit der Einführung der \mathcal{R} -Funktionen steht nun ein Kalkül bereit, das die automatische Berechnung der Gewichtsfunktionen erlaubt.

Beispiel 5.2

Es wird hier also Beispiel 2.2 wieder aufgegriffen.

$$\begin{aligned} R_1 &= 1 - x^2 - y^2 \\ R_2 &= 1 - (x - 1)^2 - y^2 \\ R_3 &= 1 - x^2 - (y - 1)^2 \\ R_4 &:= R_1 + R_2 - \sqrt{(R_1 + R_2)} \\ R_5 &:= R_3 + R_4 - \sqrt{(R_3 + R_4)} \end{aligned}$$

Die ersten drei Funktionen definieren Gewichtsfunktionen auf drei Kreisen, die letzten beiden realisieren den Schnitt dieser drei. Die Ableitungen werden mittels 5.1 schrittweise berechnet, es ergibt sich also folgende Berechnungskette für die Ableitungen erster Ordnung

$$\begin{aligned}
R_1 &= 1 - x^2 - y^2 \\
\frac{\partial}{\partial x} R_1 &= -2x \\
\frac{\partial}{\partial y} R_1 &= -2y \\
\\
R_2 &= 1 - (x - 1)^2 - y^2 \\
\frac{\partial}{\partial x} R_2 &= -2x + 2 \\
\frac{\partial}{\partial y} R_2 &= -2y \\
R_3 &= R_1 + R_2 - \sqrt{R_1 + R_2} \\
\frac{\partial}{\partial x} R_3 &= \left(1 + \frac{\omega_1}{\sqrt{\omega_1^2 + \omega_2^2}}\right) \Big|_{\omega_1=R_1, \omega_2=R_2} \frac{\partial}{\partial x} R_1 + \\
&\quad \left(1 + \frac{\omega_2}{\sqrt{\omega_1^2 + \omega_2^2}}\right) \Big|_{\omega_1=R_1, \omega_2=R_2} \frac{\partial}{\partial x} R_2 \\
&= 1 + \frac{R_1}{\sqrt{R_1^2 + R_2^2}} \frac{\partial}{\partial x} R_1 + 1 + \frac{R_2}{\sqrt{R_1^2 + R_2^2}} \frac{\partial}{\partial x} R_2 \\
\frac{\partial}{\partial y} R_3 &= \left(1 + \frac{\omega_1}{\sqrt{\omega_1^2 + \omega_2^2}}\right) \Big|_{\omega_1=R_1, \omega_2=R_2} \frac{\partial}{\partial y} R_1 + \\
&\quad \left(1 + \frac{\omega_2}{\sqrt{\omega_1^2 + \omega_2^2}}\right) \Big|_{\omega_1=R_1, \omega_2=R_2} \frac{\partial}{\partial y} R_2 \\
&= 1 + \frac{R_1}{\sqrt{R_1^2 + R_2^2}} \frac{\partial}{\partial y} R_1 + 1 + \frac{R_2}{\sqrt{R_1^2 + R_2^2}} \frac{\partial}{\partial y} R_2,
\end{aligned}$$

wodurch nicht jede Ableitung symbolisch berechnet werden muss, sondern rekursives Einsetzen zum Ziel führt. Die Gewichtsfunktion wird darüber hinaus natürlich stets auf einem Gitter ausgewertet, weshalb jede der oben genannten Funktionen auf Matrizen angewandt wird (wobei jede Operation als komponentenweise Operation aufzufassen ist). Das letzte in der Kette stehende R_n ist die Gewichtsfunktion auf Ω ausgewertet an den Punkten, die in x und y vorgegeben werden.

Bemerkung 5.3

Da die Differenziation stets symbolisch durchgeführt wird, sind auch die Ergebnisse der Ableitungen von R_n bis auf die üblichen Rundungsfehler akkurat.

5.2 Implementierung der Objekte

Bei der Implementierung werden nur Quadriken verwaltet. Diese werden mittels ihrer definierenden Matrix gespeichert. Diese Matrix wird wie in Kapitel 4.1.2 geschildert ermittelt.

Für die Implementierung wurden verschiedene Bewegungen für jeden Quadriktyp gewählt, die gut mit den einzugebenden Daten im Editor harmonisieren.

Für die Berechnungen werden an dieser Stelle noch ein paar Schreibweisen definiert, um eine kompakte Darstellung zu ermöglichen.

Definition 5.4

Seien $v, w \in \mathbb{R}^d$. Das Skalarprodukt der beiden Vektoren ist das bekannte euklidische Skalarprodukt

$$\langle v, w \rangle := \sum_{i=1}^d v_i w_i$$

mit der induzierten euklidischen Norm

$$|v| := \sqrt{\langle v, v \rangle}.$$

Die orthogonale Projektion von v auf w ist durch

$$\mathcal{P}_w(v) := w \frac{\langle v, w \rangle}{|w|^2}$$

gegeben.

Schließlich sei an dieser Stelle noch der Einheitsvektor

$$E := [1 \ 0]$$

definiert.

Die Definition von $\mathcal{P}_w(v)$ entspricht der bildlichen Vorstellung der Orthogonalprojektion des Vektors v auf w .

Gerade

Eingabe zwei Punkte P_1, P_2 auf der Geraden

Berechnung der Matrizen P_1 wird als Verschiebungsvektor gedeutet, d.h. der definierende Vektor der Translationsmatrix lautet

$$v := [P_1 \ P_2].$$

Der definierende Drehwinkel ergibt sich aus der Lage des verschobenen Punktes P_2 gegenüber der x-Achse.

$$\begin{aligned} \tilde{P}_2 &= V^{-1}P_2 \\ \phi &:= \arccos\left(\frac{\langle E, V^{-1}P_2 \rangle}{|E| |V^{-1}P_2|}\right) = \arccos\left(\frac{\langle E, V^{-1}P_2 \rangle}{|V^{-1}P_2|}\right) \end{aligned}$$

wobei Winkel über π natürlich gesondert behandelt werden müssen.

Ellipse

Eingabe Mittelpunkt P_1 , Endpunkt einer Halbgeraden P_2 und ein weiterer Punkt P_3 , dessen Projektion auf die zweite Halbachse deren Endpunkt vorgibt

Berechnung der Matrizen Die Verschiebung wird wie bei der Gerade durch P_1 vorgegeben.

$$v = [P_{11} \quad P_{12}]$$

Die beiden Vektoren, die die Halbachsen darstellen, seien

$$\begin{aligned} h_1 &:= P_2 - P_1 \\ h_2 &:= P_3 - P_1. \end{aligned}$$

Die Entfernung von P_1 zu P_2 gibt die Streckung in Richtung der ersten Halbachse an.

$$\alpha = |P_1 - P_2|$$

Die Entfernung der Projektion von P_3 auf h_2 zu P_1 ergibt den definierenden Streckungsfaktor in Richtung der zweiten Halbachse.

$$\beta = |\mathcal{P}_{h_2}(P_3) - P_1|$$

Die Lage des transformierten Punktes P_3 liefert schließlich noch den Drehwinkel.

$$\phi = \arccos\left(\frac{\langle E, V^{-1}P_3 \rangle}{|E||V^{-1}P_3|}\right) = \arccos\left(\frac{\langle E, V^{-1}P_3 \rangle}{|V^{-1}P_3|}\right)$$

Parabel

Eingabe Scheitel P_1 , ein Punkt P_2 auf der Symmetrieachse und ein beliebiger Punkt P_3 auf der Parabel

Berechnung der Matrizen Die Translation ist durch den Scheitel vorgegeben.

$$v = [P_{11} \quad P_{21}]$$

Die Rotation folgt aus der Symmetrieachse

$$\phi = \arccos\left(\frac{\langle E, T^{-1}P \rangle}{|T^{-1}P|}\right)$$

und die Skalierung entspricht dem Verhältnis der Komponenten des transformierten Punktes $\tilde{P} := R^{-1}T^{-1}P_3$

$$\alpha = \frac{\tilde{P}_2}{\tilde{P}_1^2}$$

Hyperbel

Eingabe Schnittpunkt der Asymptoten P_1 , ein Punkt P_2 auf der Symmetrieachse, ein Punkt P_3 auf einer Asymptote und schließlich ein beliebiger Punkt P_4 auf der Hyperbel selbst.

Berechnung der Matrizen Die Verschiebung wird wieder einmal durch P_1 definiert.

$$v = [P_{11} \quad P_{12}]$$

Der Punkt P_3 wird so gewählt, dass er auf der rechten Seite der Symmetrieachse liegt; falls notwendig, wird P_3 an der Symmetrieachse gespiegelt.

Der definierende Rotationswinkel ist durch die Lage der Asymptote definiert.

$$\phi := \arccos \left(\frac{\langle E, V^{-1}P_3 \rangle}{|E||V^{-1}P_3|} \right) = \arccos \left(\frac{\langle E, V^{-1}P_3 \rangle}{|V^{-1}P_3|} \right)$$

Für die Scherung wird der Winkel zwischen zweiter Asymptote und y-Achse nach der Translation und Rotation benötigt. Dazu kann die Symmetrieachse herangezogen werden, da sie den Öffnungswinkel der Asymptoten halbiert.

Der Winkel zwischen Asymptote und x-Achse nach Anwendung der Verschiebung und der Rotation ist Null, da diese Bewegungen ja genau dafür gewählt wurden.

Der Winkel zwischen Symmetrieachse und x-Achse ist schließlich

$$\mu := \arccos \left(\frac{\langle E, R^{-1}V^{-1}P_2 \rangle}{|R^{-1}V^{-1}P_2|} \right).$$

Also berechnet sich der definierende Scherwinkel zu

$$\psi = \frac{\pi}{2} - 2\mu.$$

Der Skalierungsfaktor lässt sich schließlich noch aus P_4 berechnen, da er wie bei der Parabel das Verhältnis der beiden Koordinaten des transformierten Kurvenpunktes ist.

$$\alpha = \beta = \sqrt{|X^{-1}S^{-1}V^{-1}P_4|}$$

5.3 Implementierung der Berechnung der Randkurven

Die Schnittbedingung, die in (4.27) definiert wurde, lautet ausgeschrieben

$$\begin{aligned} c(t) &= \frac{b_0^2(t)w_0P_0 + b_1^2(t)w_1P_1 + b_2^2(t)w_2P_2}{b_0^2(t)w_0 + b_1^2(t)w_1 + b_2^2(t)w_2} \\ &= \frac{(t-1)^2w_0P_0 + t(t-1)w_1P_1 + t^2w_2P_2}{(t-1)^2w_0 + t(t-1)w_1 + t^2w_2} \\ &= \frac{(w_0P_0) + (w_1P_1 - w_0P_0)t + (w_2P_2 - 2w_1P_1 + w_0P_0)t^2}{(w_0) + (w_1 - w_0)t + (w_2 - 2w_1 + w_0)t^2} \\ &= \frac{g_0P + g_1Pt + g_2Pt^2}{g_0 + g_1t + g_2t^2} \end{aligned}$$

$$\text{und } d(s) = \frac{h_0Q + h_1Qt + h_2Qt^2}{h_0 + h_1t + h_2t^2},$$

mit

$$\begin{aligned}
 G &:= [g_0 \ g_1 \ g_2]^T \\
 &:= [w_0 \ -2w_0 + 2w_1 \ w_0 - 2w_1 + w_2]^T \\
 &= \begin{bmatrix} 1 & 0 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{bmatrix} \begin{bmatrix} w_0 & 0 & 0 \\ 0 & w_1 & 0 \\ 0 & 0 & w_2 \end{bmatrix} \\
 &=: F W
 \end{aligned}$$

und

$$\begin{aligned}
 H &:= [h_0 \ h_1 \ h_2]^T \\
 &:= [v_0 \ -2v_0 + 2v_1 \ v_0 - 2v_1 + v_2]^T \\
 &=: F V.
 \end{aligned}$$

Desweiteren wird noch eine Matrix der Punkte aufgestellt

$$P^i := \begin{bmatrix} P_{0i} \\ P_{1i} \\ P_{2i} \end{bmatrix},$$

womit sich je eine Komponente von (5.2) und (5.2) zu

$$c_i(t) = \frac{T F W^T P^i}{T F W^T E}$$

bzw.

$$d_i(s) = \frac{(F V^T Q^i)^T S^T}{(F V^T E)^T S^T}$$

ergibt. Dies alles in (4.27) eingesetzt führt zu

$$\begin{aligned}
 c_i(t) &= d_i(s) \\
 \frac{TFWP^i}{TFWE} &= \frac{(FVQ^i)^T S^T}{(FVE)^T S^T} \\
 TFWP^i(FVE)^T S^T &= TFWE(FVQ^i)^T S^T \\
 T((FW)(P^i E^T - EQ^{iT})(FV)^T) S^T &= 0,
 \end{aligned}$$

welches die Matrix A in (4.28) darstellt.

5.4 Der QBE

Im Rahmen dieser Diplomarbeit wurde ein Programmpaket erstellt, das die Umsetzung dieser Sachverhalte darstellt. Der Editor, der in MATLAB programmiert wurde, heißt QBE, eine Abkürzung für Quadric Boundary Editor. Er erlaubt die grafische Eingabe von Quadriken und deren Kombinationen und berechnet die zugehörigen Gewichtsfunktionen und Randkurven.

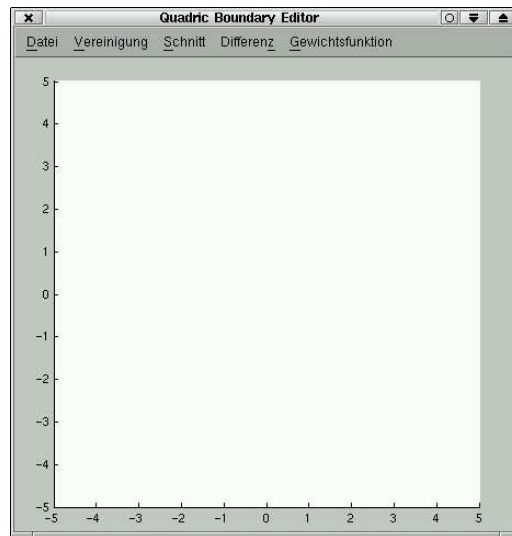


Abbildung 5.1: Das Hauptfenster des QBE

5.4.1 Das Hauptfenster

Wird der QBE aus Matlab heraus gestartet, so wird das Hauptfenster geöffnet, das in Abbildung 5.1 zu sehen ist. Das Hauptmenü enthält die Untermenüs *Datei*, *Vereinigung*, *Schnitt*, *Differenz* und *Gewichtsfunktion*, die im Folgenden erläutert werden sollen.

5.4.2 Das Menü Datei

In Abbildung 5.2 sind die Unterpunkte dieses Menüs dargestellt. Die Einträge bieten die folgenden Möglichkeiten.

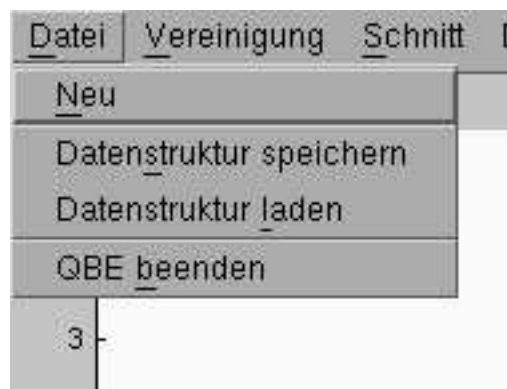


Abbildung 5.2: Das Untermenü Datei

Neu

Löscht eventuell vorhandene Gebiete und ermöglicht den Neubeginn einer Eingabe.

Datenstruktur speichern

Ein vorhandenes Gebiet kann hier gespeichert werden. Alle notwendigen Datenstruk-

turen werden in ein einzugebendes File geschrieben.

Datenstruktur laden

Eine Datei kann hier eingelesen werden und erlaubt somit die weitere Bearbeitung eines zuvor erstellten Gebietes.

QBE beenden

Beendet den QBE

5.4.3 Die Menüs Vereinigung, Schnitt und Differenz

Diese Menüs sind in Abbildung 5.3 dargestellt. Sie bieten die Möglichkeit, jeweils die Vereinigung, den Schnitt oder die Differenz des bestehenden Gebietes mit Halbebene, Ellipse, Parabel, Hyperbel bzw. mit einem vordefinierten Gebiet zu berechnen. Nach Anwählen eines dieser Punkte erscheint in der Titelleiste die weitere Vorgehensweise für die Eingabe.

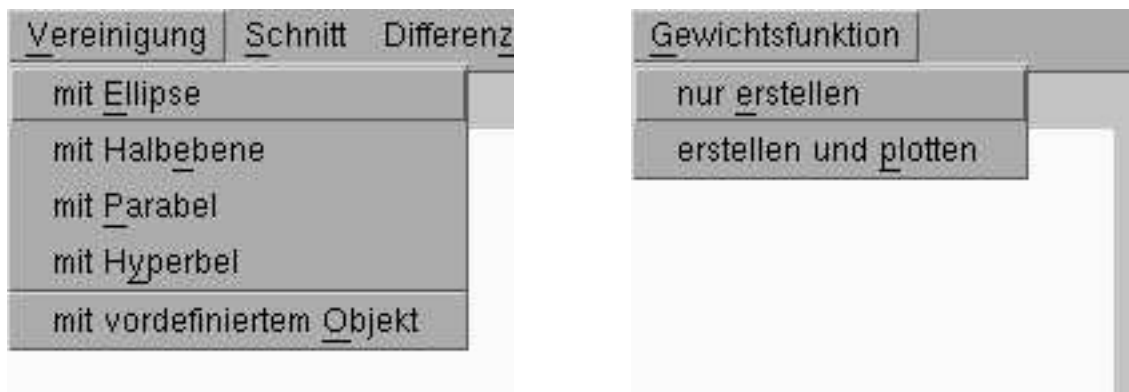


Abbildung 5.3: Das Untermenü Vereinigung als Beispiel für die Operatormenüs und das Untermenü Gewichtsfunktion

5.4.4 Das Menü Gewichtsfunktion

Nachdem ein Gebiet erstellt wurde, kann in diesem Menü die zugehörige Gewichtsfunktion erstellt werden. Abbildung 5.3 zeigt die beiden Möglichkeiten, die angeboten werden.

nur erstellen

Eine Matlab-Funktion wird erstellt (`web_weight_rfct`), welche die zu dem Gebiet passende Gewichtsfunktion an eingegebenen Stellen auswertet. Diese Funktion wird in das aktuelle Verzeichnis geschrieben und kann so immer wieder verwendet werden.

erstellen und plotten

Bei diesem Punkt wird zusätzlich noch ein Bild der Gewichtsfunktion erstellt, das in einem separaten Fenster angezeigt wird.

Kapitel 6

Beispiele

In diesem Kapitel werden zunächst die Handhabung des **QBE** erläutert. Es werden die interaktive und die skriptgesteuerte Erzeugung von Gebieten ausführlich erklärt. Im Anschluss daran folgt die Anwendung der Theorie in Kapitel 1, d.h. auf einem Gebiet wird das Poisson-Problem mit Dirichlet-Randbedingungen gelöst (siehe Abschnitt 1.1).

6.1 Interaktives Beispiel

An dieser Stelle soll exemplarisch ein Beispiel konstruiert werden, das alle geometrischen Objekte verwendet, um ein Gebiet zu konstruieren. Das Gebiet ist in Abbildung 6.1 zu sehen. Bei der Konstruktion ist zu beachten, dass zunächst der äußere Rand konstruiert und danach die Löcher hineingeschnitten werden sollten. Die Eingabe eines Loches, das aus mehreren Rändern besteht, erfolgt durch die Zwischenspeicherung dieses Loches als eigenständiges Gebiet.

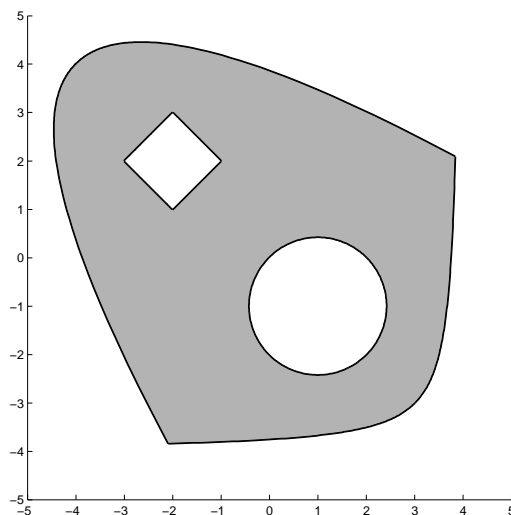


Abbildung 6.1: Das Beispielgebiet in Kapitel 6.1

Als erstes wird also das Quadrat eingegeben, welches das Loch im Gebiet darstellt.

- Zunächst wird eine Halbebene erstellt, die dann durch den Schnitt mit weiteren Halbebenen beschränkt wird. Im Menü **Vereinigung** wird dies mit dem Unterpunkt

mit **Halbebene** ermöglicht. Der **QBE** erwartet die Eingabe zweier Punkte auf der Randlinie der Halbebene, welche in diesem Beispiel bei $(-2, 1)$ und $(-1, 2)$ liegen sollen. Die Seite, auf welcher die Halbebene liegen soll, wird mittels eines Klicks in die linke obere Ecke ausgewählt.

- Das Quadrat wird durch den Schnitt mit drei weiteren Halbebenen durch die Punkte $(-2, 3)$ und $(-3, 2)$ erstellt. Im Menü **Schnitt** wird jeweils der Unterpunkt mit **Halbebene** gewählt und je zwei Punkte mit der Maus eingegeben. Die jeweils richtige Wahl der Seite führt auf das Quadrat.
- Dieses Quadrat soll nun abgespeichert werden. Dafür muss im Menü **Datei** der Unterpunkt **Datenstruktur speichern** gewählt werden, der die Speicherung der nötigen Variablen ermöglicht. In diesem Fall sollte der Dateiname **ex_square.mat** gewählt werden.

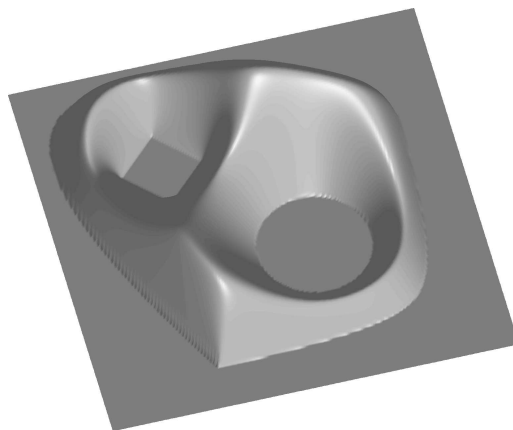


Abbildung 6.2: Gewichtsfunktion des Beispielgebietes

Nun folgt die Eingabe des Gebietes, aus dem anschließend die Löcher herausgeschnitten werden.

- Zunächst muss der Arbeitsplatz mittels **Datei, Neu** wieder gelöscht werden. Anschließend wird die Parabel eingegeben, was durch **Vereinigung, mit Parabel** realisiert wird. Nacheinander werden der Scheitelpunkt $(-4, 4)$, ein Punkt auf der Symmetrieachse $(4, -4)$ und schließlich ein Punkt der Parabel $(-2, -4)$ gewählt. Als letztes wird ein Punkt innerhalb des Gebietes ausgewählt (z.B. $(0, 0)$).
- Als nächstes folgt der Schnitt mit der Hyperbel. Nachdem der Menüpunkt **Schnitt, mit Hyperbel** angewählt wurde, können der Schnittpunkt der Asymptoten $(4, -4)$, ein Punkt auf der Symmetrieachse $(-4, 4)$, ein Punkt auf einer Asymptote $(-2, -4)$ und schließlich ein Punkt auf der Hyperbel selbst $(3, -3)$ nacheinander eingegeben werden.

Nachdem nun die Randkurve feststeht, können Löcher in das Gebiet geschnitten werden.

- Zunächst wird das elliptische Loch in das Gebiet geschnitten. Im Menü **Differenz** befindet sich der Unterpunkt mit **Ellipse**. Nach der Wahl dieses Punktes wird die Eingabe des Mittelpunktes $(1, -1)$, des Endpunktes der ersten Halbachse $(0, -2)$

und eines Punktes $(2, -2)$, dessen Projektion auf die Normale auf die erste Halbachse durch den Mittelpunkt den Endpunkt der zweiten Halbachse darstellt (vgl. Kapitel 5.2) ermöglicht. Die Wahl des Inneren der Ellipse führt auf das gewünschte gelochte Gebiet.

- Zuletzt wird das oben erstellte, quadratische Gebiet entfernt. Im Menü **Differenz** wird durch den Unterpunkt **mit vordefiniertem Gebiet** ein Dialogfenster aufgerufen, das die Eingabe eines Dateinamens verlangt (hier `ex_square.mat`). Das Ergebnis ist das gewünschte Gebiet in Abbildung 6.1.

Abschließend kann nun die Gewichtsfunktion zu diesem Gebiet erstellt und betrachtet werden. Dafür sollte im Menü **Gewichtsfunktion** der Unterpunkt **erstellen und plotten** gewählt werden. Dies erzeugt ein neues Fenster, in dem die Gewichtsfunktion wie in Abbildung 6.2 dargestellt wird. In der Darstellung ist zu erkennen, dass negative Werte der Gewichtsfunktion auf Null gesetzt werden. Andernfalls käme es aufgrund der polynomialen Abnahme der Gewichtsfunktion außerhalb des Gebietes zu betragsmäßig sehr großen Ausschlägen, was die Qualität der Darstellung stark verschlechterte.

6.2 Beispiel ohne Interaktion

Alle Funktionen des QBE können auch in Skriptform aufgerufen werden. Dazu dienen unter Anderem die Befehle `qbe_script_ell`, `...para`, `...hyp` und `...plane`. Diese Funktionen akzeptieren die definierenden Parameter, rechnen sie in definierende Matrizen um und führen die Gebietskombination und die Randberechnung durch.

Obiges Beispiel lautet als Script also:

```

1 qbe_script_init;
2 qbe_script_plane([-2,1], [-1,2], OpType_union);
3 qbe_script_plane([-1,2], [-2,3], OpType_intersec);
4 qbe_script_plane([-2,3], [-3,2], OpType_intersec);
5 qbe_script_plane([-3,2], [-2,1], OpType_intersec);
6 qbe_script_save('temp.mat');
7 qbe_script_init;
8 qbe_script_para([-4,4], -3/4*pi, [-2,-4], OpType_union);
9 qbe_script_hyp ([4,-4], pi/2, 0, [3,-3], OpType_intersec);
10 qbe_script_ell ([1,-1], sqrt(2*1^2), sqrt(2*1^2), pi/4, OpType_diff);
11 qbe_script_load('temp.mat', OpType_diff);
12 qbe_script_end;

```

Die erste Zeile ist lediglich für Initialisierungen notwendig. Die folgenden vier Zeilen definieren die Halbebenen (über zwei Punkte, die auf der trennenden Geraden liegen), die später das quadratische Loch bilden werden. Zeile sechs speichert dieses Quadrat in dem File `'temp.mat'`. Die nächste Zeile sorgt für einen Neustart. Daraufhin wird in den Zeilen acht bis zehn die äußere Randkurve mittels Parabel, Hyperbel und Ellipse konstruiert. Zeile elf schneidet das Quadrat aus dem Gebiet aus und die letzte Zeile plottet das Ergebnis.

Die Skriptfunktionen eignen sich insbesondere für exakte Eingaben, die in einer Datei abgelegt und somit einfach bearbeitet werden können. Für genauere Angaben über die Skriptfunktionen sei auf Anhang A oder auf [HAB⁺] verwiesen.

6.3 Lösung des Poisson-Problems

In diesem Abschnitt wird das Poisson-Problem mit Dirichlet-Randbedingungen gelöst, wie es in (1.1) definiert ist.

$$\begin{aligned} -\Delta u &= 1 && \text{in } \Omega, \\ u &= 0 && \text{auf } \partial\Omega. \end{aligned}$$

Als Gebiet Ω , auf dem das Problem gelöst wird, dient das oben definierte. Die Gewichtsfunktion die zum Einsatz kommt, ist die automatisch durch den QBE generierte bei der die Funktionen in \mathcal{R}_0 zum Einsatz kommen. Die Lösung wird mittels im web-Projekt erstellter Funktionen gefunden. Die Hauptfunktion lautet `web_poissongl_sol`, das auf einige Unterfunktionen zurückgreift. Für eine detailliertere Beschreibung der Funktionsweisen sei auf [HAB⁺] verwiesen.

Die Lösung ist in Abbildung 6.3 zu sehen.

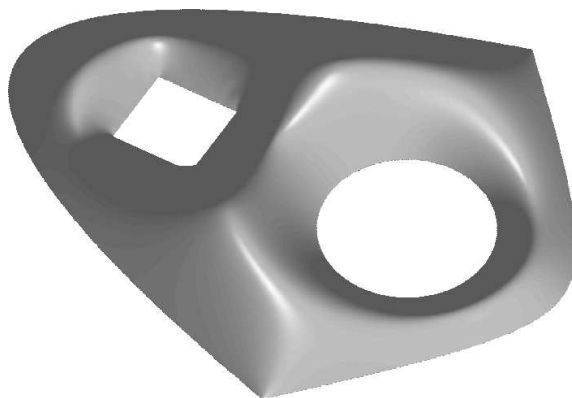


Abbildung 6.3: Lösung der Poissongleichung

Im Gegensatz zur Gewichtsfunktion in der Abbildung 6.2 hat die Lösung deutlich entschärfte Grate. Dies entspricht erstens der Eigenschaft einer Lösung des Poisson-Problems und ist zweitens eine Folge davon, dass Splines als Finite Elemente gewählt wurden.

6.4 Rechenzeiten für \mathcal{R} -Funktionen

Die \mathcal{R} -Funktionen aus Kapitel 2 wurden implementiert und getestet. Die Laufzeiten dieser Funktionen werden in diesem Abschnitt präsentiert.

Alle Berechnungen wurden in MATLAB (Version 6.1.0.450 (R12.1)) durchgeführt. Als Rechner diente ein AMD Athlon XP1800+ Prozessor mit 1024MB Hauptspeicher.

Definiert wurden die Gebiete

$$\Omega_n := \bigcup_{i=1}^n \{x \in \mathbb{R}^2 : |x - (i, 0)| \leq 1\},$$

also die Vereinigung von Einheitskreisen mit Mittelpunkten auf ganzzahligen Abschnitten einer Koordinatenachse. Diese Gebiete wurden durch sukzessive Kombination der Kreise gebildet und in dem Quadrat

$$G := \{x \in \mathbb{R}^2 : -1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1\}$$

ausgewertet.

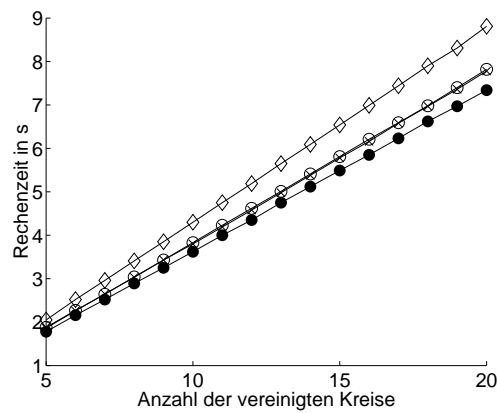


Abbildung 6.4: Absolute Rechenzeiten für verschiedene \mathcal{R} -Funktionen

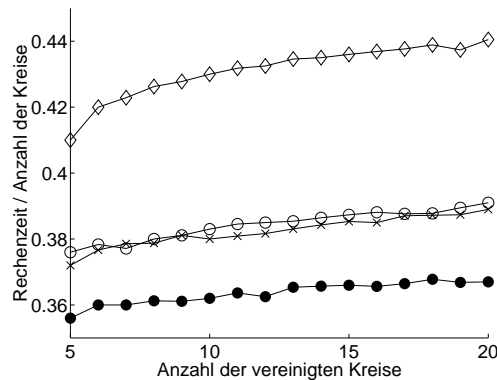


Abbildung 6.5: relative Rechenzeiten

Abbildung 6.4 zeigt die absoluten Rechenzeiten. Dabei steht \bullet für Funktionen aus \mathcal{R}_0 , \times steht für \mathcal{R}_m mit $m = 2$, \circ für \mathcal{R}_p , $p = 2$ und schließlich \diamond für \mathcal{R}_C^m , $m = 2$. Der erwartete lineare Verlauf ist leicht zu erkennen. Es zeigt sich ein leichter Rechenzeitvorteil der \mathcal{R}_0 -Funktionen, gegenüber den \mathcal{R}^m - und \mathcal{R}_p -Funktionen, die praktisch gleichauf liegen. Der Unterschied beträgt Durchschnittlich etwa 5%. Größer, nämlich ca. 18%, ist der Unterschied zwischen den Rechenzeiten der \mathcal{R}_0 -Funktionen und denen aus \mathcal{R}_C^m . Diese

Berechnungen wurden für $m = p = 2$ durchgeführt.

In Abbildung 6.5 ist der Unterschied der Rechenzeiten nochmals deutlicher zu sehen. Hier wurden nur die Steigungen der Geradenstücke geplottet. Sie zeigen die Nähe von \mathcal{R}^m zu \mathcal{R}_p und die Unterschied zwischen den anderen Funktionsfamilien noch deutlicher. Der leichte Anstieg der Graphen in diesem Diagramm macht deutlich, dass es sich bei den Rechenzeiten doch nicht um eine perfekt lineare Entwicklung handelt. Die geringe Steigerung kommt wahrscheinlich durch eine komplexere Speicherverwaltung bei vielen Gebieten zustande.

Kapitel 7

Visualisierung der Eigenschaften

In diesem Kapitel werden die Eigenschaften, die in den Kapiteln 2 und 3 gezeigt wurden, visualisiert. Die unterschiedlichen \mathcal{R} -Funktionen werden auf geeignete Gebiete angewandt. Dies ermöglicht beispielweise die einfache Veranschaulichung der Nichtdifferenzierbarkeit in Ecken.

7.1 Differenzierbarkeit in Ecken

An dieser Stelle soll ein einfaches Gebiet mit einer Ecke betrachtet werden (siehe Abbildung 7.1).

$$\Omega_1 := \{(x, y) \in \mathbb{R} : x > 0\}$$

$$\Omega_2 := \{(x, y) \in \mathbb{R} : y > 0\}$$

$$\Omega_q := \Omega_1 \cap \Omega_2$$

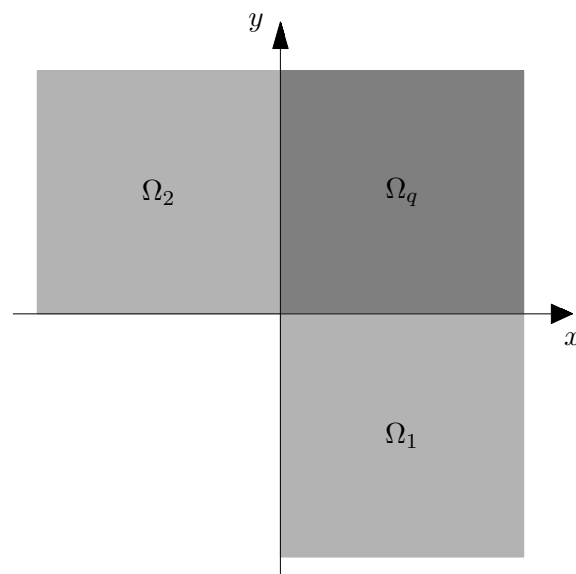


Abbildung 7.1: Das Gebiet Ω_q als Schnitt der Gebiete Ω_1 und Ω_2

Als Gewichtsfunktionen auf den Grundmengen wird

$$\omega_1(x, y) = y, \quad \omega_2(x, y) = x$$

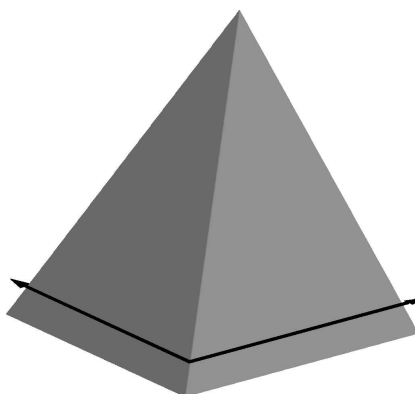
gewählt. Diese werden nun mit allen \mathcal{R} -Funktionen zu einer Gewichtsfunktion auf Ω_q kombiniert.

- **Funktions-System \mathcal{R}_1**

Das einfachste System.

$$R_1^\cap = \min(x, y).$$

Im Bild zeigt sich die Nichtdifferenzierbarkeit auf der ersten Winkelhalbierenden. Somit ist natürlich auch im Ursprung die Differenzierbarkeitsordnung gleich Null.



- **Funktions-Systeme \mathcal{R}_0 und \mathcal{R}_p**

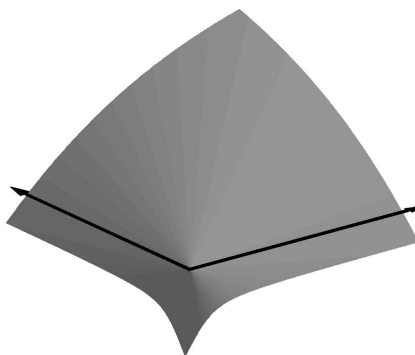
Die Funktionensysteme enthalten die Schnittfunktionen

$$R_0^\cap = x + y - \sqrt{x^2 + y^2}$$

und

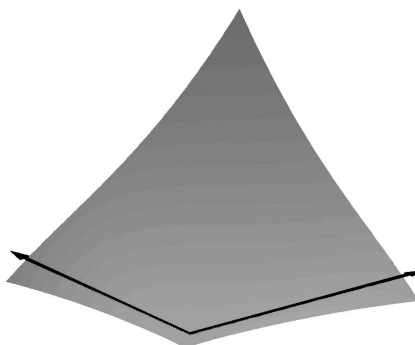
$$R_0^\cap = x + y - (x^p + y^p)^{1/p}$$

Die entstehenden Bilder ähneln sich sehr, weswegen hier nur eines abgebildet ist. Es lässt sich in der Abbildung kein Grat feststellen. Die in den Sätzen 2.20 und 2.29 bewiesene Nichtdifferenzierbarkeit im Ursprung ist leicht an der auftretenden Spitze zu erkennen.



- **Funktions-Systeme \mathcal{R}_0^m und \mathcal{R}_C^m**

Die Funktionen in \mathcal{R}_0^m sind ebenso wie diejenigen aus \mathcal{R}_C^m selbst in Ecken m -fach differenzierbar. Dies wurde in den Sätzen 2.25 und 2.36 gezeigt. In der Abbildung lässt sich in Folge dessen weder ein Grat noch eine Spitze feststellen. Die Bilder der beiden Funktionsfamilien ähneln sich wieder so sehr, dass nur eine Abbildung an dieser Stelle angeführt wird.

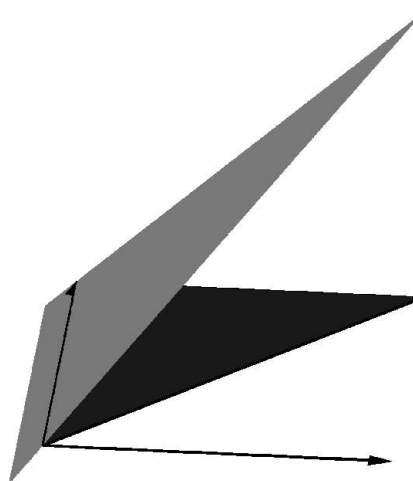


7.2 Normalisiertheit

Eine Gewichtsfunktion ist nach Definition 2.15 normalisiert, wenn die Bedingungen 2.4 und 2.5 gelten. Anschaulich kann eine Funktion, die quadratisch oder von noch höherer Ordnung gegen den Rand konvergiert, nicht normalisiert sein, da die Ableitungen nicht verschwinden würden. In den folgenden Bildern wurden die Gewichtsfunktionen an der ersten Winkelhalbierenden aufgeschnitten, damit die Abnahme zum Rand hin besser zur Geltung kommt.

- **Funktions-System \mathcal{R}_1**

Die Gewichtsfunktionen auf den ursprünglichen Gebieten Ω_1 und Ω_2 entsprechen den Abstandsfunktionen, daher ist auch ω die Abstandsfunktion auf Ω_q . Diese ist natürlich vollständig normiert (vgl. Satz 2.18). Im Bild zeigt sich dies durch den linearen Verlauf zum Rand hin.



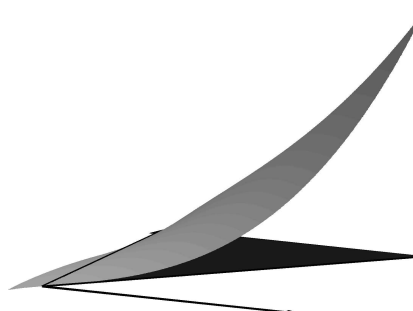
- **Funktions-System \mathcal{R}_0**

Laut Satz 2.22 sind die Funktionen in \mathcal{R}_α normiert von der Ordnung 1. Das bedeutet, dass sie linear in den Rand laufen. An Ecken überlagern sich natürlich die linearen Verläufe und ergeben auch einen linearen Verlauf in die Ecke hinein, wie in der Abbildung zu sehen ist.



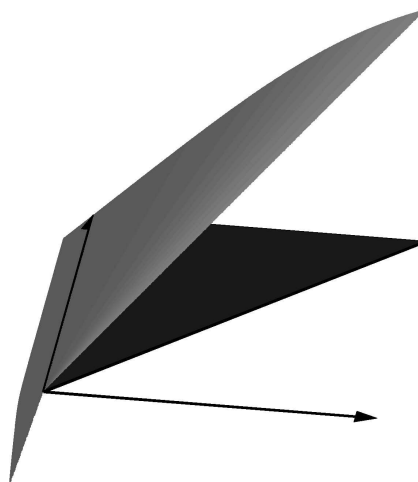
- **Funktions-System \mathcal{R}_m**

Schon in Bemerkung 2.26 wurde dieses Beispiel betrachtet und gezeigt, dass die Gewichtsfunktion nicht normalisiert ist. In der Abbildung, in der \mathcal{R}_m für $m = 2$ dargestellt wurde, ist die quadratische Abnahme in der Nähe der Ecke leicht zu erkennen.



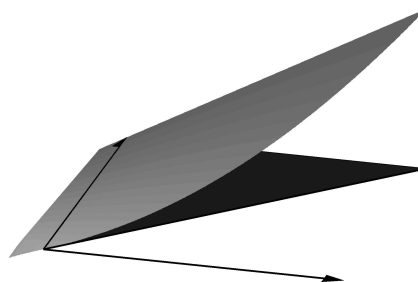
- Funktions-System \mathcal{R}_p

In der Abbildung, die \mathcal{R}_p für $p = 4$ darstellt, ist die lineare Annäherung an den Rand deutlich zu sehen, die bereits in Satz 2.31 gezeigt wurde. Die Tatsache, dass die Gewichtsfunktion zur Ordnung $p - 1 = 3$ normalisiert ist; lässt sich natürlich nicht in der Abbildung beobachten.



- Funktions-System \mathcal{R}_C^m

Die Funktionen in \mathcal{R}_C^m ähneln denen in \mathcal{R}_m weitgehend. So ist auch die in der Abbildung dargestellte Funktion ($m = 2$) nicht normalisiert, wie schon seit Satz 2.37 bekannt ist.



7.3 Sonderfälle

Bei der Konstruktion der Gewichtsfunktionen von kombinierten Gebieten sind ein paar Sonderfälle zu beachten, die in Abschnitt 4.1.4 noch nicht angesprochen wurden.

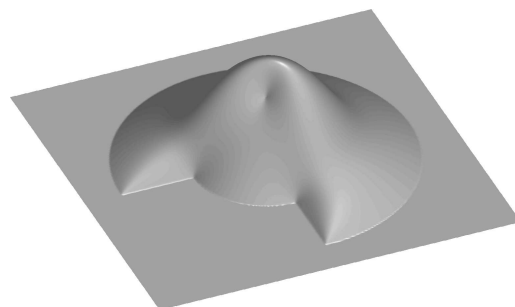
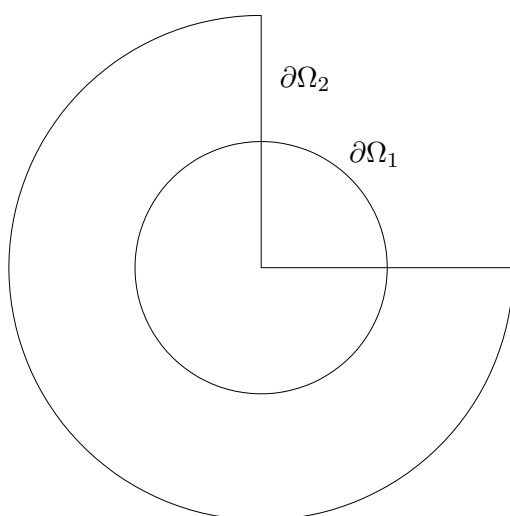


Abbildung 7.2: Ein Gebiet mit einer Nichtdifferenzierbarkeit im Innern

- In Kapitel 2 wurden die Differentiationseigenschaften der verschiedenen \mathcal{R} -Funktionen untersucht. Dabei wurde stets davon ausgegangen, dass die zu kombinierenden Gewichtsfunktionen ausreichend glatt sind (vgl. Einführung Kapitel 2.3). Andererseits wurde in Satz 2.20 gezeigt, dass eine Gewichtsfunktion, die mittels \mathcal{R}_0 erzeugt wurde, in einem Schnittpunkt der zugehörigen Gebietsränder nicht differenzierbar ist. Liegt dieser Schnittpunkt nun in einem anderen Gebiet, mit dem das bestehende vereinigt werden soll, bleibt diese nicht differenzierbare Stelle erhalten (s. Abb. 7.2).

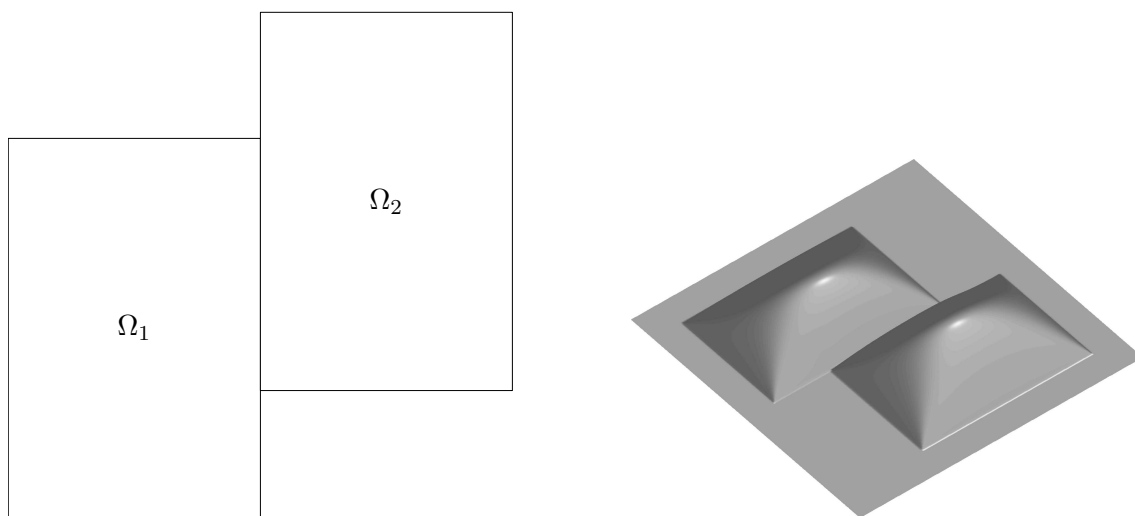


Abbildung 7.3: Die Vereinigung zweier Gebiete erzeugt Nullstellen in Innern des Gebietes

- Eine weitere in Kapitel 2 gezeigte Eigenschaft stößt bei Nichtbeachtung der Voraussetzungen auf ihre Grenzen: Werden zwei Gebiete Ω_1 , Ω_2 vereinigt, deren Schnitt nur Randpunkte beider Gebiete enthält (siehe Abbildung 7.3), dann erhält die Gewichtsfunktion auf dem vereinigten Gebiet Ω Nullstellen an Punkten, die im Innern des Gebietes liegen.

Diese inneren Nullstellen können durch zusätzliche Vereinigung mit einem geeigneten Gebiet eliminiert werden. Allerdings ist dabei die Nichtkommutativität der \mathcal{R} -Funktionen zu beachten! Wird zu obiger Vereinigung ein Kreis hinzugefügt, so bleibt, wie oben erwähnt, die Nichtdifferenzierbarkeit erhalten (siehe Abbildung 7.4 links). Die Vertauschung der Vereinigungen (Ω_1 wird zuerst mit dem Kreis vereinigt, danach mit Ω_2) liefert das gewünschte glatte Ergebnis (siehe Abbildung 7.4 rechts).

- Bei der Kombination von Gebieten ist zu beachten, dass sich diese Operationen unter Umständen vermeiden lassen. Als Beispiel diene hier die Vereinigung von zwei Kreisen. Ist der Kreis Ω_1 im Kreis Ω_2 enthalten, so ist die Vereinigung gleich Ω_2 . Diese Operation kann also schlicht übergangen werden, was deutliche Berechnungsvorteile bringen kann (siehe Abschnitt). Im QBE sind solche Vereinfachungen nicht enthalten, da sie mit aufwändigen In-Out-Test verbunden wären. Ebenso können durch wiederholte Kombination von Gebieten komplexe begleitende boolesche Funktionen entstehen, die unter Umständen vereinfachbar sind. Auf die Vereinfachung kann hier

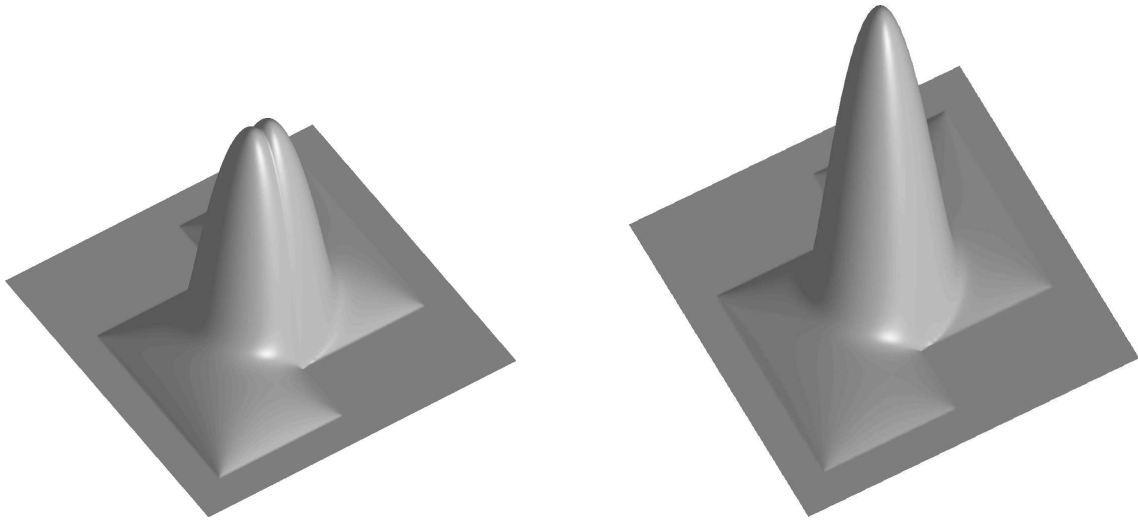


Abbildung 7.4: Gewichtsfunktionen bei unterschiedlichen Kombinationsreihenfolgen

nicht eingegangen werden, allerdings findet sich in [Bon77] eine gute Referenz für dieses Problem.

Anhang A

Kurzbeschreibung der Programme

In diesem Abschnitt werden die wichtigsten Programme des QBE-Paketes kurz vorgestellt. Sie stellen eine Umsetzung der \mathcal{R} -Funktionen dar und wurden auf der Grundlage der in Kapitel 4 und 5 vorgestellten Zusammenhänge implementiert. Eine ausführlichere Beschreibung findet sich in [HAB⁺].

A.1 Datenstrukturen

Die zentralen Datenstrukturen sind:

WEB_BOUNDARY

Dies ist eine $(n \times 1)$ -cell-Variable zur Speicherung der Gebietsränder. n ist die Anzahl der Randkurven des Gebiets.

WEB_BOUNDARY{k} ist ein $(m \times 1)$ -struct-Array. m ist die Anzahl der Bézier-Segmente. Die Struktur besteht aus den Einträgen:

degree: double (1×1) : Grad der Bézierkurve

points: double $((\text{degree} + 1) \times 3)$: Kontrollpunkte, immer in rationaler Form angegeben. In jeder Zeile steht ein homogener Kontrollpunkt in der Form (wx, wy, w) . Die Randkurven sind mathematisch positiv orientiert, d.h. das berandete Gebiet liegt auf der linken Seite der Kurven.

WEB_SET

Dies ist eine (1×1) -struct-Variable zur Speicherung von Gebieten und deren Kombinationen.

Die struct-Variable umfasst die Felder:

set: $1 \times n$ cell: Beschreibung der Grundgebiete

op: $m \times 3$ double: Beschreibung der Kombinationen

WEB_SET.set ist $(1 \times n)$ -Cell von symmetrischen (3×3) -Matrizen. Jedes Element Q der Cell repräsentiert den Rand einer Quadrik in der Form $XQX^T = 0$ mit $X = [x \ y \ 1]$ (siehe auch 4.1.1).

WEB_SET.op ist $(m \times 3)$ -Matrix. Jede Zeile der Matrix entspricht dem Tripel (Operator, Argument 1, Argument 2), wobei Operator aus der Menge (OpType_union, OpType_intersec, OpType_compl OpType_diff) gewählt werden muss. Ist ein Ar-

gument negativ, so bezeichnet es eines der Grundgebiete, ist das Argument positiv, so bezeichnet es ein Kombinationsgebiet.

A.2 Programme

Die wichtigsten Programme des QBE-Pakets:

qbe

Diese Funktion ist zwar der Start des QBE, erstellt allerdings lediglich das Hauptfenster und initialisiert das Menü (vgl. Kapitel 5.4), in dem alle Befehle zur Verfügung gestellt werden.

qbe_add_set

Diese Funktion ermöglicht die Kombination des aktuellen Gebietes mit einer anzugebenden Quadrik.

Die Quadrik wird einfach in `WEB_SET` aufgenommen und die Operatormatrix entsprechend angepasst. Nur im Falle der Hyperbel wird diese zuerst noch mit einer Halbebene geschnitten, damit nur ein Ast der Hyperbel berücksichtigt wird.

Diese Funktion wird von allen Quadrik-Eingabe-Funktionen (`qbe_ellipse` usw.) aufgerufen.

qbe_bez_bound_comb

Die Kombination von Randkurven ist die weitaus komplexere Aufgabe als die Berechnung der Gewichtsfunktion. Diese Funktion ist das Herz dieser Aufgabe.

Sie realisiert den Algorithmus in Abschnitt 4.2.5. Von den Spezialfällen einmal abgesehen werden hier alle Kurven miteinander geschnitten (mit Hilfe von `qbe_bez_sec`) und die resultierenden Schnittpunkte nacheinander entlanggelaufen.

Eine Vielzahl von Sonderfällen müssen allerdings auch berücksichtigt werden, so zum Beispiel die Handhabung von nicht geschlossenen Flächen oder das Einfügen von Löchern.

qbe_bez_sec

Die Funktion `qbe_bez_sec` berechnet die Parameterwerte aller Schnittpunkte zweier quadratischer rationaler Bezierkurven.

Bei zwei nichtlinearen Kurven wird das Gleichungssystem $c(t) = d(s)$ wie in Abschnitt 4.2.4 umgeformt und mittels `qbe_quad_solve` gelöst.

qbe_bound_plot

Plottet ein Gebiet, das durch eine Kurve aus quadratischen Bezierkurvenstücken berandet ist.

Zunächst werden auf der Kurve Auswertungspunkte ermittelt, dann wird getestet, ob die Kurve geschlossen ist. Falls dies nicht zutreffen sollte, so werden die Randpunkte des augenblicklichen Fensters hinzugefügt, sodass ein 'unbeschränktes' Gebiet gezeichnet wird.

qbe_complement

In der Funktion `qbe_add_set` wird aus Gründen der Vereinfachung nur die Schnittbildung realisiert. Alle anderen Funktionen lassen sich mittels dieser und der Komplementbildung darstellen. Die Funktion `qbe_complement` errechnet das Komplement

eines Gebietes, indem die Operationen auf den Grundgebieten angepasst werden und die berandende Kurve umgekehrt wird.

`qbe_ellipse`, `qbe_halfplane`, `qbe_hyperbola`, `qbe_parabola`

Diese Funktionen werden direkt aus dem Menü des QBE aufgerufen. Sie realisieren die interaktive Eingabe von Quadriken, wie sie in Abschnitt 5.2 beschrieben ist. Die Funktionen realisieren genau diese Transformationen um die Darstellung allgemeiner Quadriken zu erhalten.

`qbe_io_diag`

Wird bei den Funktionen `qbe_ellipse` usw. das bestehende Gebiet mit genau einer Quadrik kombiniert, so ermöglicht die Funktion `qbe_io_diag` die Speicherung eines Gebietes. Somit können in einem Zwischenschritt Gebiete gespeichert werden, die dann später mit einem anderen, ebenfalls bereits kombinierten Gebiet verrechnet werden können. Dies erweitert also die Kombinationsmöglichkeiten des QBE von sukzessiven Berechnungen auf beliebige Kombinationen.

`qbe_params_to_def...`

Diese Funktionen ermöglichen die Berechnung der definierenden Matrizen aus den definierenden Parametern. Die Umrechnung erfolgt nach den in Abschnitt 5.2 erläuterten Zusammenhängen.

`qbe_plot_weight`

Erzeugt einen Plot der Funktion, die in `web_weight_rfct` abgelegt ist. Negative Werte werden als Null gezeichnet. Die Gewichtsfunktionen in dieser Arbeit (z.B. Abbildung 6.1) wurden alle mit diesem Programm erstellt.

`qbe_predef_obj`

Diese Funktion ermöglicht das Einlesen eines zuvor abgespeicherten Gebietes. Dieses wird mit dem bisher bestehenden Gebiet kombiniert. Daher stellt die Funktion eine Erweiterung von `qbe_ellipse` usw. dar, welche eine Umsetzung von beliebigen Kombinationen ermöglicht.

Das vordefinierte Objekt wird eingelesen, der Struktur `WEB_SET` hinzugefügt und schließlich wird die neu entstandene Randkurve aus aktuellem und vordefinierten Gebiet berechnet.

`qbe_quad_solve`

Die Funktion `qbe_quad_solve` ermittelt die reellen Lösungen einer biquadratischen Gleichung in zwei Dimensionen, wie sie z.B. in (4.27) gegeben ist.

Die Gleichung wird mittels Satz 4.15 und den folgenden Bemerkungen gelöst, indem die Resultante berechnet und ausgewertet wird.

`qbe_script_init`

Die Funktionenfamilie `qbe_script...` ermöglicht die scriptgesteuerte Erzeugung von Gebieten. `qbe_script_init` erfüllt dabei die notwendigen Initialisierungsaufgaben und sollte vor Benutzung der weiteren Funktionen dieser Familie aufgerufen werden.

`qbe_script_ell`, `qbe_script_hyp`, `qbe_script_para`, `qbe_script_plane`

Diese Funktionen stellen die scriptgesteuerte (interaktionsfreie) Erstellung von Gebieten und deren Gewichtsfunktionen dar. Durch Angabe der definierenden Parameter (siehe Abschnitt 4.1.3) und einem booleschen Operator wird die entsprechende Quadrik dem bisherigen Gebiet hinzugefügt.

qbe_script_end

Diese Funktion erstellt mittels `qbe_plot_weight` einen Plot des Gebietes, das zuvor mit obenstehenden Funktionen erstellt wurde. Es bildet somit den natürlichen Abschluss der `qbe_script_...` Funktionen.

web_weight_rfct_create

Das eigentliche Kernstück der Erstellung der Gewichtsfunktion eines Gebietes. Die definierenden Matrizen der Quadriken werden aus der Struktur `WEB_SET` ausgelesen und nach dem Verfahren in Abschnitt 5.1 in eine Tabelle eingetragen, welche die sukzessive Berechnung der Gewichtsfunktion und ihrer Ableitungen ermöglicht.

Literaturverzeichnis

- [Arg60] J.H. Argyris. *Energy Theorems and Structural Analysis*. Butterworths, 1960.
- [Béz66] P. Bézier. Définition numérique des courbes et surfaces I. *Automatisme*, XI:625–632, 1966.
- [BKGH97] I.N. Bronstein, K.A.Semendjajew, G.Musiol, and H.Mühlig. *Taschenbuch der Mathematik*. Verlag Harry Deutsch, 1997.
- [Bon77] U. Bong. *Boolsche Algebra*. Herder, 1977.
- [Bra92] D. Braess. *Finite Elemente*. Springer-Verlag, 1992.
- [Cou43] R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bull. Amer. Math. Soc.*, 49:1–23, 1943.
- [dB72] C. de Boor. On calculating with b-splines. *J. Approx. Theory*, 6:50–62, 1972.
- [dB78] C. de Boor. *A practical guide to splines*. Applied Mathematical Sciences. Springer-Verlag, 1978.
- [dC59] P. de Casteljaou. Courbes et surfaces a pôles. Technical report, André Citroën Automobiles SA, 1959.
- [Far92] G.E. Farin. *Curves and surfaces for computer aided geometric design: a practical guide*. Academic Press, Inc., 1992.
- [HAB⁺] K. Höllig, C. Apprich, M. Boßle, W. Geis, J. Hörner, A. Kopf, B. Russig, A. Streit, and J. Wipper. WEB-Projekt-Dokumentation. Universität Stuttgart, Mathematisches Institut A.
- [HL89] J. Hoschek and D. Lasser. *Grundlagen der geometrischen Datenverarbeitung*. B.G. Teuber Stuttgart, 1989.
- [Hö98] K. Höllig. *Grundlagen der Numerik*. Math Text, Zavelstein, 1998.
- [Hö03] K. Höllig. Finite Element Methods with B-Splines. *Frontiers in Applied Mathematics 26*, SIAM, 2003. to appear.
- [HRW01a] K. Höllig, U. Reif, and J. Wipper. Error estimates for the web-method. In T. Lyche and L.L. Schumaker, editors, *Mathematical Methods for Curves and Surfaces: Oslo 2000*, pages 195–209. Vanderbilt University Press, 2001.
- [HRW01b] K. Höllig, U. Reif, and J. Wipper. Weighted extended b-spline approximation of Dirichlet problems. *SIAM J. Numer. Anal.*, 39(2):442–462, 2001.

- [KK56] L.W. Kantorowitsch and W.I. Krylow. *Näherungsmethoden der Höheren Analysis*. VEB Deutscher Verlag der Wissenschaften, Berlin, 1956.
- [Rei98] U. Reif. Geometrische Datenverarbeitung. Vorlesungsskript, Universität Stuttgart, 1998.
- [Rei99] U. Reif. Splineapproximation. Vorlesungsskript, Universität Stuttgart, 1999.
- [RS95] V.L. Rvachev and T.I. Sheiko. *R-functions in boundary value problems in mechanics*. *Applied Mechanics Reviews*, 48(4):151–188, 1995.
- [Rva63] V.L. Rvachev. Analytical description of some geometric objects. *Dokl. AS USSR*, 153(4):765–768, 1963.
- [Rva67] V.L. Rvachev. *Geometric Applications of Logic Algebra*. Naukova Dumka, 1967. russisch.
- [Rva74] V.L. Rvachev. *Methods of Logic Algebra in Mathematical Physics*. Naukova Dumka, 1974. russisch.
- [Rva82] V.L. Rvachev. *Theory of R-Functions and Some Applications*. Naukova Dumka, 1982. russisch.
- [Sha88] V. Shapiro. Theory of *R-functions* and applications: A primer. Technical Report CPA88-3, Cornell Programmable Automation, Sibley School of Mechanical Engineering, Ithaca, NY, 1988.
- [Shi88] A.K. Shidlovsky, editor. *Vladimir Logvinovich Rvachev*. Bibliography of scientists of UkSSR. Naukova Dumka, 1988. russisch.
- [ST99] V. Shapiro and I. Tsukanov. Implicit functions with guaranteed differential properties. *Proceedings of the Fifth ACM Symposium on Solid Modeling and Applications*, 1999.
- [Str02] A. Streit. Web-Spline-Approximation der Plattengleichung. Master's thesis, Universität Stuttgart, Mathematisches Institut A, 2002.
- [TCMT94] M.J. Turner, R.W. Clough, H.C. Martin, and L.C. Topp. Stiffness and deflection analysis of complex structures. *J. Aeronaut. Sci.*, 23(9):805–823, 854, 1994.
- [vdW73] B.L. vander Waerden. *Algebra I*. Springer, 1973.
- [Zie95] O.C. Zienkiewicz. Origins, milestones and directions of the finite element method. *Arch. Comp. Methods Eng.*, 2:1–48, 1995.