Optimale Hüllkörper für Objekte mit gekrümmten Oberflächen

Joachim Reichel

31. Oktober 2001

Diplomarbeit

nach einem Thema von Priv. Doz. Dr. Elmar Schömer in der Naturwissenschaftlich-Technischen Fakultät I, Fachrichtung 6.2 – Informatik, der Universität des Saarlandes, Saarbrücken.

Hiermit erkläre ich an Eides Statt, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Saarbrücken, im Oktober 2001

Inhaltsverzeichnis

Danksagung

Mein Dank gilt Herrn Priv. Doz. Dr. Elmar Schömer für die Vergabe dieser Diplomarbeit. Herr Schömer hat mich während der ganzen Zeit hervorragend betreut und hatte für die auftretenden Probleme stets ein offenes Ohr.
Ferner möchte ich mich bei Christian Lennerz und Thomas Warken für die gute Zusammenarbeit bedanken.
Ganz besonders bedanken möchte ich mich bei meinen Eltern, die mich während meines Studiums stets unterstützt haben.

Kapitel 1

Einleitung

In der heutigen Welt spielt hochentwickelte Technik eine bedeutende Rolle. Produkte werden immer komplizierter, gleichzeitig wird die Entwicklung dieser Produkte immer kosten- und zeitintensiver. Zusätzlich sinkt aufgrund des technologischen Fortschritts die Lebensdauer der Produkte. Daher steht für die Entwicklung neuer Produkte nur sehr wenig Zeit zu Verfügung. Dies alles ist nur durch den Einsatz von Computern beim Entwurf der Produkte möglich *(CAD - Computer Aided Design)*.
Wünschenswert wäre auch eine Verkürzung der Testphase neuer Produkte. Üblicherweise werden von neuen Produkten zunächst Prototypen angefertigt, die dann umfangreichen Tests unterworfen werden. Gegebenenfalls muss dann der Entwurf verändert und ein neuer Prototyp hergestellt werden. Beim Konzept des *Virtual Prototyping* werden keine realen Prototypen mehr hergestellt, statt dessen werden die Tests im Computer simuliert. So können Entwurfsfehler frühzeitig erkannt und zeit- und kostenaufwendige Tests an Prototypen vermieden werden. Beispielsweise kann der Entwurf eines neuen Autos auf dessen Verhalten im (virtuellen) Windkanal getestet werden. Ebenso sind virtuelle Crash-Tests denkbar.
Eine andere Möglichkeit ist die Durchführung von Untersuchungen in einer *Virtual-Reality*-Umgebung. Dabei interagiert eine mit Datenhandschuh und 3D-Brille ausgestattete Testperson mit dem Modell des Autos in einer vom Computer generierten Welt.
Für eine Virtual-Reality-Umgebung ist die wirklichkeitsgetreue Simulation der realen Welt äußerst wichtig. Ein Hauptaspekt ist hierbei die *Dynamiksimulation*, also die Vorhersage der Bewegungen von Objekten unter dem Einfluss externer Faktoren. Beispielsweise muss in einer Virtual-Reality-Umgebung der Einfluss der Schwerkraft auf die Objekte berücksichtigt werden. Zum anderen muss sichergestellt werden, dass sich zwei massive Objekte nicht durchdringen können.

Als *Hüllkörper* bezeichnen wir einfache geometrische Körper, z. B. Kugeln, Zylinder oder Quader, die das eigentliche Objekt einschließen. Hüllkörper haben große Bedeutung bei der Beschleunigung vieler Algorithmen in der Dynamiksimulation. Ein wesentlicher Bestandteil der Dynamiksimulation ist die Kollisionserkennung. Die statische Kollisionserkennung beantwortet die Frage, ob sich zwei Objekte zu einem bestimmten Zeitpunkt durchdringen oder nicht. Die dynamische Kollisionserkennung berechnet für zwei gegebene Objekte mit ihren Trajektorien den frühesten Zeitpunkt einer Berührung. Bei beiden Arten der Kollisionserkennung spielen Abstandsberechnungen eine zentrale Rolle. Diese Abstandsberechnungen sind für Körper mit gekrümmten Oberflächen und Begrenzungskanten wesentlich aufwendiger als für Polyeder.

Der Rechenaufwand lässt sich durch den Einsatz von Hüllkörpern stark verringern. Beispielsweise sind zwischen Objekten mit disjunkten Hüllkörpern keine Berührungen möglich. Dieser Vorteil lässt sich durch Verwendung von sogenannten *Hüllkörperhierarchien* noch weiter steigern. Dabei wird die Flächenmenge eines Körpers rekursiv unterteilt und jeweils in Hüllkörper eingeschlossen.

Da sich jede gekrümmte Fläche beliebig fein durch Polygonnetze approximieren lässt, stellt sich die Frage, warum man überhaupt direkt mit gekrümmten Flächen arbeiten sollte. Viele CAD-Systeme verarbeiten neben Polyedern auch Kugeln, Kegel, Zylinder und Tori. Aufgrund des Datenaustausches mit CAD-Systemen ist es wünschenswert, direkt auf diesen Datenstrukturen zu arbeiten. Zum anderen erhöht eine gute Approximation erheblich die Beschreibungskomplexität eines Objektes und verschlechtert so die Laufzeit vieler Algorithmen. Der größte Nachteil liegt jedoch darin, dass sich die Objekteigenschaften durch die Approximation verändern können. Man denke beispielsweise an den Zylinder eines Motors. Der Zylinderkolben kann sich im dicht umschließenden Zylindergehäuse frei auf- und abbewegen. Je nach Approximation der zylindrischen Flächen ist es möglich, dass sich der Zylinderkolben nicht mehr bewegen lässt. Im schlimmsten Fall durchdringen sich jetzt Zylinderkolben und Gehäuse, die zuvor disjunkt waren.

NURBS (Non-Uniform Rational B-Splines) sind ein mathematisches Modell zur Beschreibung gekrümmter Kurven und Flächen. In diesem Modell besitzen die Kurven und Flächen eine explizite Darstellung als rationale Funktionen in einer bzw. zwei Variablen. Im Gegensatz zu quadratischen Flächen gibt es im Allgemeinen keine einfache implizite Darstellung. Als Spezialfall enthalten NURBS auch Bézierkurven und -flächen und nichtrationale Splines. NURBS sind sehr flexibel und werden in vielen kommerziellen CAD-Systemen verwendet. Dies ist ein weiterer Grund für den Einsatz dieses Modells, da so die verlustbehaftete Konvertierung zwischen verschiedenen Darstellungsformen entfällt.

Für Polyeder gibt es viele Algorithmen zur schnellen Berechnung von Hüllkörpern. Beispielsweise berechnet der randomisierte Algorithmus von Welzl die kleinste einschließende Kugel für einen Polyeder [Wel91]. Die erwartete Laufzeit ist linear in der Anzahl der Eckpunkte des Polyeders. O'Rourke hat in [ORo85] einen kombinatorischen Algorithmus beschrieben, der den kleinsten einschließenden Quader für einen Polyeder berechnet. Dieser Algorithmus hat eine Laufzeit von $O(n^3)$, wobei n wieder die Anzahl der Eckpunkte des Polyeders bezeichnet.

Wir beschreiben in dieser Arbeit ein numerisches Verfahren, um volumenminimale Kugeln und Quader als Hüllkörper für Objekte mit gekrümmten Oberflächen zu berechnen. Die dabei auftretenden Teilprobleme werden als Nullstellenprobleme univariater und bivariater Polynome formuliert. Ferner geben wir für eine Teilmenge der quadratischen Kurven und Flächen eine explizite Lösung dieser Teilprobleme an. Wir benutzen lokale und globale Optimierungsverfahren, um die Hüllkörper mit minimalem Volumen zu berechnen. Der Gesamtalgorithmus ist linear in der Anzahl der Flächen, Kanten und Eckpunkte des Objekts und in der Anzahl der Iterationen des Optimierungsverfahrens.

Gliederung

Im zweiten Kapitel erklären wir zunächst kurz die in dieser Arbeit verwendete Notation. Danach wenden wir uns den NURBS zu. Wir erklären zunächst den Begriff des Knotenvektors und der B-Spline-Basisfunktion, bevor wir dann NURBS-Kurven und NURBS-Flächen definieren. Anschließend geben wir einen Überblick über die Intervallarithmetik und sprechen kurz die erweiterte Intervallarithmetik an. Wir werden die Intervallarithmetik verwenden, um den Wertebereich einer Funktion auf einem Intervall abzuschätzen. Danach erklären wir das ein- und mehrdimensionale Intervall-Newton-Verfahren. Diese Variante des gewöhnlichen Newton-Verfahrens ermöglicht die Bestimmung sämtlicher Nullstellen einer differenzierbaren Funktion. Das Verfahren berücksichtigt außerdem die für numerische Algorithmen typischen Rundungsfehler. Insbesondere werden für die Nullstellen garantierte Einschließungen berechnet. Den Abschluss dieses Kapitels bildet ein Beispiel, das den Einsatz des Intervall-Newton-Verfahrens im Zusammenhang mit NURBS demonstriert. Unter *Rückrechnung* verstehen wir das Problem, für einen Punkt auf einer gegebenen NURBS-Kurve oder NURBS-Fläche die zugehörigen Parameterwerte zu bestimmen. Dieses Problem werden wir als Nullstellenproblem formulieren und mit Hilfe des Intervall-Newton-Verfahrens lösen.
Im dritten Kapitel behandeln wir die Berechnung der optimalen Hüllkörper. Nach einer Abgrenzung der Aufgabenstellung wenden wir uns zunächst der Kugel als Hüllkörper zu. Dabei treten zwei Teilprobleme auf: Abstandsberechnungen und Enthaltenseinstests. Diese werden als Nullstellenprobleme formuliert und können mit dem vorgestellten Intervall-Newton-Verfahren gelöst werden. Für einige spezielle Kurven- und Flächentypen werden wir außerdem explizite Formeln zur Lösungen dieser Probleme angeben. Wir zeigen anschließend, dass die zu minimierende Zielfunktion konvex ist und erklären die verwendeten lokalen Optimierungsverfahren. Schließlich zeigen wir noch einige Möglichkeiten auf, den Algorithmus zu beschleunigen.
Im Fall der Quader als Hüllkörpertyp beschreiben wir zunächst, wie die Orientierung der Quader im Raum spezifiziert wird. Analog zu den Abstandsproblemen im Fall der Kugeln werden hier Ausdehnungen des Objekts in vorgegebenen Richtungen berechnet. Die zu minimierende Zielfunktion ist in diesem Fall allerdings nicht konvex, so dass neben lokalen auch globale Optimierungsverfahren zum Einsatz kommen.
Im letzten Abschnitt dieses Kapitels präsentieren wir einige Ergebnisse dieser Arbeit in Form von Beispielszenen. Wir zeigen außerdem, dass im Fall der Quader der Einsatz eines lokalen Optimierungsverfahrens zur Minimierung der nicht-konvexen Zielfunktion seine Berechtigung hat.
Das letzte Kapitel bietet einen Ausblick über Verbesserungs- und Erweiterungsmöglichkeiten der hier vorgestellten Algorithmen.

Kapitel 2

Grundlagen

In diesem Kapitel wollen wir die mathematischen Grundlagen für die Lösung der später auftretenden Nullstellenprobleme erläutern. Nach der Erklärung der in dieser Arbeit verwendeten Notation wollen wir zunächst die Begriffe *NURBS-Kurve* und *NURBS-Fläche* definieren. Danach führen wir die Intervallarithmetik ein. Anschließend erläutern wir kurz das herkömmliche Newton-Verfahren und stellen dann das Intervall-Newton-Verfahren vor. Besonderes Gewicht liegt dabei auf verschiedenen Varianten des mehrdimensionalen Intervall-Newton-Verfahrens. Das Kapitel schließt mit einem ausführlichen Beispiel für den Einsatz des Intervall-Newton-Verfahrens zur Lösung von Nullstellen-Problemen.

2.1 Notation

Wir wollen in diesem Abschnitt kurz die in dieser Arbeit verwendete Notation beschreiben.

- Für **Skalare** werden kursive Kleinbuchstaben verwendet, also z. B. i, m, n, u, v, α, λ.

- **Vektoren** werden kursiv und fettgedruckt dargestellt, z. B. $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{x}$, $\boldsymbol{y}$ oder $\boldsymbol{z}$. Mit Ausnahme von einigen Bezeichnungen im Zusammenhang mit NURBS werden nur Kleinbuchstaben verwendet.

- Für **Matrizen** verwendet wir aufrechte Großbuchstaben im Fettdruck wie zum Beispiel $\mathbf{A}$ oder $\mathbf{M}$. Die Einheitsmatrix wird mit $\mathbf{I}$ bezeichnet.

- **Intervalle** werden durch kursive Großbuchstaben dargestellt. Beispielsweise sind A, B, C, X, Y und Z Intervalle.

- **Intervallvektoren** und **Intervallmatrizen** werden wie gewöhnliche Vektoren und Matrizen dargestellt. Aus dem Zusammenhang sollte stets klar sein, ob es sich bei den Komponenten um Skalare oder Intervalle handelt.

- Für **Mengen** verwenden wir kalligraphische Buchstaben, also z. B. $\mathcal{M}$, $\mathcal{X}$ oder $\mathcal{Y}$.

2.2 NURBS

NURBS (Non-Uniform Rational B-Splines) sind ein mathematisches Modell zur Beschreibung gekrümmter Kurven und Flächen. Eine umfassende Darstellung zum Thema NURBS findet sich in [PT97]. Dort sind auch die Beweise für die aufgeführten Eigenschaften zu finden.
Zunächst werden wir den Begriff der *B-Spline-Basisfunktion* einführen und dann die Begriffe *NURBS-Kurve* und *NURBS-Fläche* definieren.

2.2.1 B-Spline-Basisfunktionen

Definition 2.1. *Eine nichtabsteigende Folge* $\boldsymbol{U}$ *von* r *reellen Zahlen* u_i, $i = 0, \ldots r-1$, *heißt* Knotenvektor der Länge r.

$$\boldsymbol{U} = \{u_0, \ldots, u_{r-1}\} \quad \textit{mit} \quad u_i \leq u_{i+1} \textit{ für } 0 \leq i < r-1$$

Die Werte u_i *heißen* Knotenpunkte. *Das halboffene Intervall* $[u_i, u_{i+1})$ *heißt* i-te Knotenspanne. *Ein Knoten* u *hat* Multiplizität k, *wenn es genau* k *Werte* $u_i, \ldots, u_{i+k-1}$ *gibt, so dass* $u = u_i = \ldots = u_{i+k-1}$ *gilt.*

Wir verwenden hier die für Knotenvektoren übliche Schreibweise mit geschweiften Klammern, obwohl es sich nicht um Mengen, sondern um Vektoren handelt.

Definition 2.2. *Sei* $\boldsymbol{U} = \{u_0, \ldots, u_{r-1}\}$ *ein Knotenvektor der Länge* r. *Die* i-te B-Spline-Basisfunktion p-ten Grades *über dem Knotenvektor* $\boldsymbol{U}$ *wird mit* $N_{i,p}(u)$ *bezeichnet und ist definiert als*

$$\begin{aligned} N_{i,0}(u) &= \begin{cases} 1, \text{ falls } u_i \leq u < u_{i+1} \\ 0, \text{ sonst} \end{cases} && 0 \leq i < r-1 \\ N_{i,p}(u) &= \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) && 0 \leq i < r-p-1 \end{aligned}$$

Der in dieser rekursiven Definition möglicherweise auftretende Quotient $\frac{0}{0}$ soll als 0 definiert werden.
In Abbildung 2.1 sind die B-Spline-Basisfunktionen ersten Grades über dem Knotenvektor $\boldsymbol{U} = \{0, 0, \frac{1}{2}, 1, 1\}$ zu sehen. Abbildung 2.2 zeigt die Basisfunktionen zweiten Grades über dem Knotenvektor $\boldsymbol{U} = \{0, 0, 0, \frac{1}{2}, \frac{1}{2}, 1, 1, 1\}$.

Definition 2.3. *Ein Knotenvektor der Länge* r *für eine Basisfunktion* p*-ten Grades heißt* nicht-periodisch *oder* offen, *wenn jeweils die ersten und letzten* $p+1$ *Einträge gleich sind.*

$$\boldsymbol{U} = \{\underbrace{a, \ldots, a}_{p+1}, u_{p+1}, \ldots, u_{(r-1)-(p+1)}, \underbrace{b, \ldots, b}_{p+1}\}$$

Wir betrachten in dieser Arbeit ausschließlich nicht-periodische Knotenvektoren. Außerdem sollen die ersten $p+1$ Einträge stets den Wert 0, die letzten $p+1$ Einträge stets den Wert 1 haben.

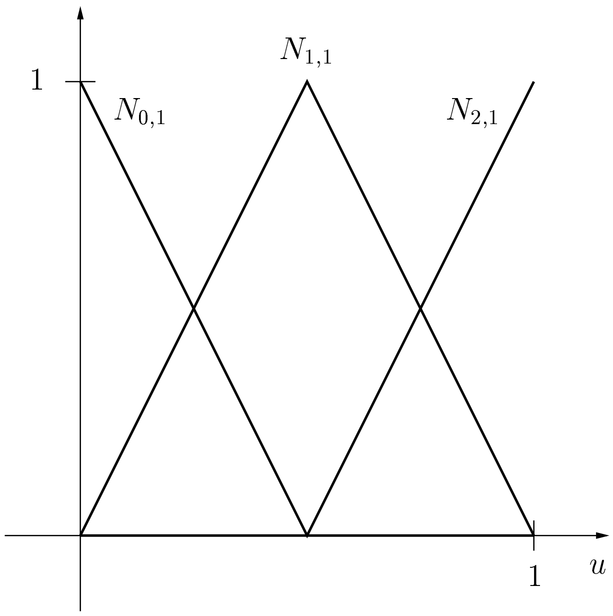

Abbildung 2.1: B-Spline-Basisfunktionen ersten Grades

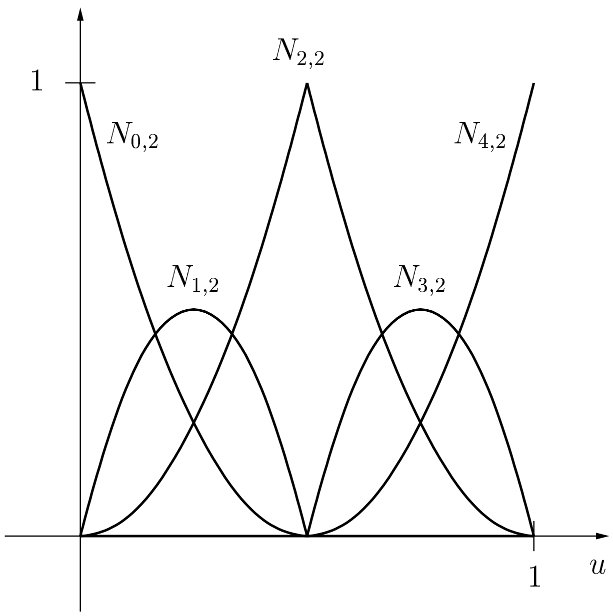

Abbildung 2.2: B-Spline-Basisfunktionen zweiten Grades

Wir wollen nun einige für unsere Zwecke wichtigen Eigenschaften der B-Spline-Basisfunktionen nennen:

- **Lokaler Träger:** Alle Basisfunktionen verschwinden außerhalb bestimmter Knotenspannen. Es gilt:
$$N_{i,p}(u) = 0 \quad \text{für alle } u \notin [u_i, u_{i+p+1})$$
- **Nichtnegativität:** Alle Basisfunktionen nehmen nur nichtnegative Werte an:
$$N_{i,p}(u) \geq 0 \quad \text{für alle } u \in [0,1], \;\; 0 \leq i < r-p-1$$
- **Teilung der Eins:** Die Menge aller Basisfunktionen p-ten Grades teilt die Eins:
$$\sum_{j=0}^{r-p-2} N_{j,p}(u) = 1 \quad \text{für alle } u \in [0,1]$$
- **Differenzierbarkeit:** Im Innern der Knotenspannen sind die Basisfunktionen beliebig oft stetig differenzierbar, wobei ab der $(p+1)$-ten Ableitungen alle Ableitungen verschwinden. An einem Knoten der Multiplizität k sind die Basisfunktionen $(p-k)$-mal stetig differenzierbar.

  Für die k-te Ableitung der Basisfunktion $N_{i,p}(u)$ gilt:
$$N_{i,p}^{(k)}(u) = \frac{p}{u_{i+p} - u_i} N_{i,p-1}^{(k-1)}(u) - \frac{p}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}^{(k-1)}(u)$$

2.2.2 NURBS-Kurven

Definition 2.4. *Eine* NURBS-Kurve p-ten Grades *ist definiert durch*

$$\boldsymbol{C}(u) = \frac{\sum_{i=0}^{n-1} N_{i,p}(u)\, w_i\, \boldsymbol{P}_i}{\sum_{i=0}^{n-1} N_{i,p}(u)\, w_i} \quad \textit{für} \quad 0 \leq u \leq 1\,. \tag{2.1}$$

Die Punkte $\boldsymbol{P}_i \in \mathbb{R}^3$ *heißen* Kontrollpunkte, *die Werte* $w_i > 0$ *heißen* Gewichte. *Mit* $N_{i,p}(u)$ *werden die B-Spline-Basisfunktionen* p*-ten Grades über einem Knotenvektor* $\boldsymbol{U} = \{u_0, \dots, u_{r-1}\}$ *der Länge* $r = n + p + 1$ *bezeichnet.*

Abbildung 2.3 zeigt eine NURBS-Kurve, die einen Halbkreis darstellt. Die NURBS-Kurve hat den Grad $p = 2$ und besitzt $n = 5$ Kontrollpunkte. Für die Gewichte gilt $w_0 = w_2 = w_4 = 1$ und $w_1 = w_3 = \frac{1}{2}\sqrt{2}$. Der zugrunde liegende Knotenvektor ist $\boldsymbol{U} = \{0, 0, 0, \frac{1}{2}, \frac{1}{2}, 1, 1, 1\}$. Die Werte der zugehörigen Basisfunktionen sind in Abbildung 2.2 dargestellt.
Die für unsere Zwecke wichtigsten Eigenschaften der NURBS-Kurven sind:

- **Interpolation der Endpunkte:** Die Kurve $\boldsymbol{C}(u)$ beginnt im Punkt $\boldsymbol{P}_0$ und endet im Punkt $\boldsymbol{P}_{n-1}$.
$$\boldsymbol{C}(0) = \boldsymbol{P}_0 \quad \text{und} \quad \boldsymbol{C}(1) = \boldsymbol{P}_{n-1}$$

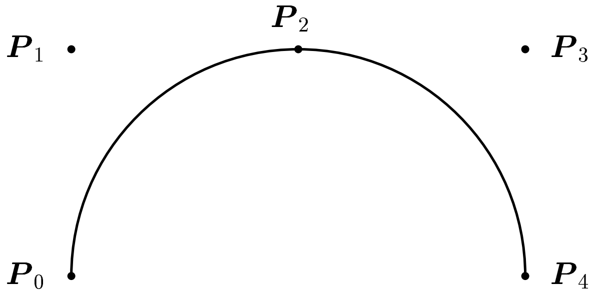

Abbildung 2.3: Ein Halbkreis als NURBS-Kurve

- **Affine Invarianz:** Um eine NURBS-Kurve einer affinen Transformation zu unterwerfen, genügt es, die affine Transformation auf die Kontrollpunkte anzuwenden.

- **Konvexe Hülle:** Der Punkt $\boldsymbol{C}(u)$ mit $u \in [u_i, u_{i+1})$ liegt in der konvexen Hülle der Kontrollpunkte $\boldsymbol{P}_{i-p}, \ldots, \boldsymbol{P}_i$.

- **Differenzierbarkeit:** Im Innern der Knotenspannen ist die Funktion $\boldsymbol{C}(u)$ beliebig oft stetig differenzierbar. An einem Knoten der Multiplizität k ist die Funktion $(p-k)$-mal stetig differenzierbar.

  Sei $\boldsymbol{C}_Z(u)$ der Zähler und $\boldsymbol{C}_N(u)$ der Nenner in Gleichung (2.1), also $\boldsymbol{C}(u) = \frac{\boldsymbol{C}_Z(u)}{\boldsymbol{C}_N(u)}$. Für die k-te Ableitung der Funktion $\boldsymbol{C}(u)$ gilt dann:

$$\boldsymbol{C}^{(k)}(u) = \frac{1}{\boldsymbol{C}_N(u)} \left(\boldsymbol{C}_Z^{(k)}(u) - \sum_{i=1}^{k} \binom{k}{i} \boldsymbol{C}_N^{(i)}(u)\, \boldsymbol{C}^{(k-i)}(u) \right)$$

Wir wollen noch den Begriff der *geschlossenen* oder *zyklischen* NURBS-Kurve einführen:

Definition 2.5. *Eine NURBS-Kurve $\boldsymbol{C}$ mit $\boldsymbol{C}(0) = \boldsymbol{C}(1)$ heißt* geschlossen *oder* zyklisch.

Zum Beispiel können ein Kreis oder eine Ellipse durch eine geschlossene NURBS-Kurve dargestellt werden.

2.2.3 NURBS-Flächen

Definition 2.6. *Eine* NURBS-Fläche (p,q)-ten Grades *ist definiert durch*

$$\boldsymbol{S}(u,v) = \frac{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} N_{i,p}(u)\, N_{j,q}(v)\, w_{i,j}\, \boldsymbol{P}_{i,j}}{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} N_{i,p}(u)\, N_{j,q}(v)\, w_{i,j}} \quad \text{für} \quad 0 \leq u, v \leq 1\,. \tag{2.2}$$

Die Punkte $\boldsymbol{P}_{i,j} \in \mathbb{R}^3$ heißen Kontrollpunkte, *die Werte $w_{i,j} > 0$ heißen* Gewichte. *Mit $N_{i,p}(u)$ werden die B-Spline-Basisfunktionen p-ten Grades über* *einem Knotenvektor* $\boldsymbol{U} = \{u_0, \ldots, u_{r-1}\}$ *der Länge* $r = n + p + 1$ *bezeichnet. Analog werden die Basisfunktionen q-ten Grades über einem Knotenvektor* $\boldsymbol{V} = \{v_0, \ldots, v_{s-1}\}$ *der Länge* $s = m + q + 1$ *mit* $N_{j,q}(v)$ *bezeichnet.*

Die Abbildung 2.4 zeigt die Hälfte eines Zylindermantels. Die NURBS-Fläche hat in u-Richtung den Grad $p = 2$ und in v-Richtung den Grad $q = 1$. Für die Gewichte gilt $w_{0,0} = w_{1,0} = w_{2,0} = w_{2,1} = w_{4,0} = w_{4,1} = 1$ und $w_{1,0} = w_{1,1} = w_{3,0} = w_{3,1} = \frac{1}{2}\sqrt{2}$. Die beiden Knotenvektoren sind $\boldsymbol{U} = \{0, 0, 0, \frac{1}{2}, \frac{1}{2}, 1, 1, 1\}$ und $\boldsymbol{V} = \{0, 0, 1, 1\}$. Die Kontrollpunkte liegen jeweils in den Ebenen der zugehörigen Halbkreisbögen.

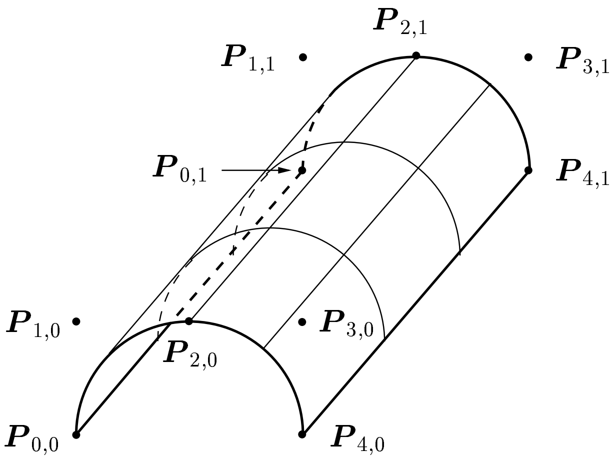

Abbildung 2.4: NURBS-Fläche

Die im vorherigen Kapitel aufgeführten Eigenschaften für NURBS-Kurven gelten sinngemäß auch für NURBS-Flächen:

- **Interpolation der Eckpunkte:** Die vier Eckpunkte der Fläche $\boldsymbol{S}(u, v)$ stimmen jeweils mit einem Kontrollpunkt überein.

$$\begin{array}{ll} \boldsymbol{S}(0,0) = \boldsymbol{P}_{0,0} & \boldsymbol{S}(1,0) = \boldsymbol{P}_{n-1,0} \\ \boldsymbol{S}(0,1) = \boldsymbol{P}_{0,m-1} & \boldsymbol{S}(1,1) = \boldsymbol{P}_{n-1,m-1} \end{array}$$

- **Affine Invarianz:** Um eine NURBS-Fläche einer affinen Transformation zu unterwerfen, genügt es, die affine Transformation auf die Kontrollpunkte anzuwenden.

- **Konvexe Hülle:** Der Punkt $\boldsymbol{S}(u, v)$ mit $u \in [u_i, u_{i+1})$ und $v \in [v_j, v_{j+1})$ liegt in der konvexen Hülle der Kontrollpunkte

$$\{\boldsymbol{P}_{k,l} \mid i - p \leq k \leq i,\ j - q \leq l \leq j\}.$$

- **Differenzierbarkeit:** Im Innern der Knotenspannen ist die Funktion $\boldsymbol{S}(u, v)$ beliebig oft stetig differenzierbar. An einem Knoten des Knotenvektors $\boldsymbol{U}$ bzw. des Knotenvektors $\boldsymbol{V}$ der Multiplizität k ist die Funktion $(p - k)$-mal bzw. $(q - k)$-mal stetig differenzierbar nach u bzw. v.

Sei $\boldsymbol{S}_Z(u,v)$ der Zähler und $\boldsymbol{S}_N(u,v)$ der Nenner in Gleichung (2.2), also $\boldsymbol{S}(u,v) = \frac{\boldsymbol{S}_Z(u,v)}{\boldsymbol{S}_N(u,v)}$. Sei außerdem $S^{(k,l)}$ eine abkürzende Schreibweise für $\frac{\partial^{(k+l)}}{\partial^k u \partial^l v} S$. Für die Ableitungen der Funktion $\boldsymbol{S}(u,v)$ gilt dann:

$$\begin{aligned}
\boldsymbol{S}^{(k,l)}(u,v) \quad = \quad & \frac{1}{\boldsymbol{S}_N(u,v)} \Bigg(\boldsymbol{S}_Z^{(k,l)}(u,v) \\
& - \sum_{i=1}^{k} \binom{k}{i} \boldsymbol{S}_N^{(i,0)}(u,v)\, \boldsymbol{S}^{(k-i,l)}(u,v) \\
& - \sum_{j=1}^{l} \binom{l}{j} \boldsymbol{S}_N^{(0,j)}(u,v)\, \boldsymbol{S}^{(k,l-j)}(u,v) \\
& - \sum_{i=1}^{k} \sum_{j=1}^{l} \binom{k}{i} \binom{l}{j} \boldsymbol{S}_N^{(i,j)}(u,v)\, \boldsymbol{S}^{(k-i,l-j)}(u,v) \Bigg)
\end{aligned} \tag{2.3}$$

Wir wollen noch den Begriff der *zyklischen* NURBS-Fläche einführen:

Definition 2.7. *Eine NURBS-Fläche* $\boldsymbol{S}$ *heißt* zyklisch in u-Richtung *bzw.* zyklisch in v-Richtung, *wenn gilt:*

$$\begin{array}{lll}
\forall v \in [0,1]: & \boldsymbol{S}(0,v) = \boldsymbol{S}(1,v) & \textit{(zyklisch in u-Richtung)} \quad \textit{bzw.} \\
\forall u \in [0,1]: & \boldsymbol{S}(u,0) = \boldsymbol{S}(u,1) & \textit{(zyklisch in v-Richtung)}\ .
\end{array}$$

Ein Zylinder lässt sich beispielsweise durch eine in *einer* Richtung zyklische NURBS-Fläche beschreiben. Bei einem Torus handelt es sich hingegen um eine in *beiden* Richtungen zyklische Fläche.

2.3 Intervallarithmetik

Wir benutzen die Intervallarithmetik, um Aussagen über den Wertebereich einer Funktion auf einem vorgegebenen Intervall zu machen. Eine ausführliche Beschreibung der Intervallarithmetik findet sich in [AH83] und [Moo69]. Wir geben nun eine Übersicht über die grundlegenden Begriffe und Eigenschaften der Intervallarithmetik.

Definition 2.8. *Die Menge* $A := \left[\underline{A}, \overline{A}\right] := \{x \in \mathbb{R} \,|\, \underline{A} \le x \le \overline{A}\}$ *mit* $\underline{A}, \overline{A} \in \mathbb{R}$ *heißt* Intervall. $\underline{A} = \inf A$ *heißt* Infimum *oder* untere Schranke *von* A, $\overline{A} = \sup A$ *heißt* Supremum *oder* obere Schranke *von* A. *Ein Intervall* A *mit* $\underline{A} = \overline{A}$ *heißt* Punktintervall. *Der Punkt* $m(A) := \frac{1}{2}(\underline{A} + \overline{A})$ *heißt* Mittelpunkt des Intervalls A.

Wir bezeichnen die Menge aller Intervalle mit $\mathbb{IR}$, die Menge aller n-dimensionalen Vektoren über $\mathbb{IR}$ mit $\mathbb{IR}^n$ und die Menge aller $n \times m$-Matrizen über $\mathbb{IR}$ mit $\mathbb{IR}^{n\times m}$. Der Mittelpunkt eines Intervallvektors oder einer Intervallmatrix ist dabei komponentenweise definiert.

Vergleiche und Teilmengenbeziehungen werden im mengentheoretischen Sinne interpretiert und sind für Vektoren und Matrizen genau dann erfüllt, wenn sie für alle ihre Komponenten erfüllt sind. Für $A, B \in \mathbb{IR}$ gilt

$$\begin{aligned} A = B \quad &\Leftrightarrow \quad \underline{A} = \underline{B} \;\wedge\; \overline{A} = \overline{B} \qquad \text{und} \\ A \subseteq B \quad &\Leftrightarrow \quad \underline{A} \geq \underline{B} \;\wedge\; \overline{A} \leq \overline{B}. \end{aligned}$$

Die Operationen $\cap$ und $\underline{\cup}$ sind für $A, B \in \mathbb{IR}$ definiert durch

$$\begin{aligned} A \cap B \;&:=\; \left[\max\left\{\underline{A}, \underline{B}\right\}, \min\left\{\overline{A}, \overline{B}\right\}\right] \qquad \text{(Schnitt)} \qquad \text{und} \\ A \,\underline{\cup}\, B \;&:=\; \left[\min\left\{\underline{A}, \underline{B}\right\}, \max\left\{\overline{A}, \overline{B}\right\}\right] \qquad \text{(Intervallhülle)}. \end{aligned}$$

Der Schnitt $A \cap B$ ist nur definiert, falls $\max\left\{\underline{A}, \underline{B}\right\} \leq \min\left\{\overline{A}, \overline{B}\right\}$ gilt. Für Vektoren und Matrizen sind die Operationen komponentenweise definiert. Man beachte, dass die Vereinigung zweier Intervalle im Allgemeinen kein Intervall ist.
Die arithmetischen Operationen auf Intervallen werden durch

$$A \circ B := \{a \circ b \mid a \in A, b \in B\}$$

definiert, wobei $A, B \in \mathbb{IR}$ und $\circ \in \{+, -, \cdot, /\}$ gilt. Das Ergebnis dieser Operationen kann explizit wie folgt berechnet werden:

$$\begin{aligned} A + B \;&=\; \left[\underline{A} + \underline{B}, \overline{A} + \overline{B}\right], \\ A - B \;&=\; \left[\underline{A} - \overline{B}, \overline{A} - \underline{B}\right], \\ A \cdot B \;&=\; \left[\min\left\{\underline{AB}, \underline{A}\overline{B}, \overline{A}\underline{B}, \overline{AB}\right\}, \max\left\{\underline{AB}, \underline{A}\overline{B}, \overline{A}\underline{B}, \overline{AB}\right\}\right], \\ A \,/\, B \;&=\; \left[\underline{A}, \overline{A}\right] \cdot \left[1/\overline{B}, 1/\underline{B}\right] \;\text{ für } 0 \notin B. \end{aligned}$$

Addition und Multiplikation sind kommutativ und assoziativ, aber nicht distributiv. Es gilt jedoch die schwächere Eigenschaft der *Subdistributivität*:

$$A \cdot (B + C) \subseteq A \cdot B + A \cdot C \qquad \text{für } A, B, C \in \mathbb{IR}.$$

Eine weitere Eigenschaft der Intervalloperationen ist die *Inklusionsisotonie*. Es gilt

$$\begin{aligned} a \in A \wedge b \in B \quad &\Rightarrow \quad a \circ b \in A \circ B \qquad \text{und} \\ A \subseteq C \wedge B \subseteq D \quad &\Rightarrow \quad A \circ B \subseteq C \circ D \end{aligned}$$

für alle $\circ \in \{+, -, \cdot, /\}$, $a, b \in \mathbb{R}$ und $A, B, C, D \in \mathbb{IR}$.

Intervallauswertung

Wir wollen die Intervallarithmetik dazu benutzen, den Wertebereich einer Funktion f auf einem Intervall X abzuschätzen. Den Wertebereich einer Funktion $f : D \to \mathbb{R}$ mit $D \subseteq \mathbb{R}$ oder $D \subseteq \mathbb{R}^n$ auf dem Intervall bzw. dem Intervallvektor $X \subseteq D$ bezeichnen wir mit $f(X)$. Mit $F(X)$ bezeichnen wir die sogenannte *Intervallauswertung* der Funktion f auf X.
Die Intervallauswertung einer Funktion erhält man, indem man alle vorkommenden Variablen durch entsprechende Intervalle und alle Operationen durch

die zugehörigen Intervalloperationen ersetzt. Konstanten werden durch entsprechende Punktintervalle ersetzt. Für den Wertebereich $f(X)$ und die Intervallauswertung $F(X)$ gilt dabei stets

$$f(X) \subseteq F(X).$$

In vielen Fällen gilt dabei echte Ungleichheit, d.h. das durch die Intervallauswertung berechnete Intervall ist echt größer als der eigentliche Wertebereich. Dieses Phänomen soll als *Überschätzung* bezeichnet werden.

Beispiel:
Gegeben sei die Funktion $f : \mathbb{R}\backslash\{0\} \to \mathbb{R}, x \mapsto x+\frac{1}{x}$ und das Intervall $X = [1, 2]$. Dann gilt für den Wertebereich $f(X) = [2, \frac{5}{2}]$. Die Intervallauswertung von f auf X liefert $F(X) = [1, 2] + [1, 1]/[1, 2] = [1, 2] + [\frac{1}{2}, 1] = [\frac{3}{2}, 3]$.

Der Grund für die Überschätzung bei der Intervallauswertung liegt in der Tatsache, dass bei der Intervallauswertung jedes Vorkommen der Variablen als eine unabhängige Instanz aufgefasst wird. So wird im obigen Beispiel der Wert 3 durch die obere Schranke 2 für den linken Summanden und die untere Schranke 1 für den Nenner des rechten Summanden erreicht. Die Überschätzung bei der Intervallauswertung ist ein nicht zu vernachlässigendes Problem beim Einsatz der Intervallarithmetik.

Erweiterte Intervallarithmetik

Bei der erweiterten Intervallarithmetik werden im Gegensatz zu der zuvor eingeführten (einfachen) Intervallarithmetik auch offene und halboffene Intervalle sowie die leere Menge und Mengen der Form $(-\infty, a] \cup [b, +\infty)$ mit $a < b$ zugelassen. Insbesondere dürfen offene Intervallgrenzen auch die Werte $+\infty$ und $-\infty$ annehmen.
Wir benötigen die erweiterte Intervallarithmetik, um beim Einsatz des Newton-Verfahrens (siehe Kapitel 2.4) Ausdrücke der Form

$$A \cap (b - C/D)$$

mit $b \in \mathbb{R}$ und $A, C, D \in \mathbb{IR}$ auswerten zu können. Insbesondere ist dabei explizit $0 \in D$ zugelassen. Die zuvor aufgeführten Relationen und Operationen für die einfache Intervallarithmetik lassen sich im wesentlichen sinngemäß auf die erweiterte Intervallarithmetik übertragen. Wir wollen die erweiterte Intervallarithmetik an dieser Stelle nicht formal definieren, sondern verweisen auf [Rat92].
Wir bemerken, dass es für den Wert des obigen Ausdrucks drei Möglichkeiten gibt:

1. Die Schnittoperation ist undefiniert, falls A und der rechte Term disjunkt sind.

2. Das Ergebnis ist ein abgeschlossenes Intervall.

3. Das Ergebnis ist die Vereinigung zweier (nicht notwendigerweise disjunkter) abgeschlossener Intervalle.

Maschinenintervallarithmetik

Alle verbreiteten Hardwarearchitekturen verwenden zur Zahlendarstellung Datenformate mit festen Wortlängen. Damit lassen sich nur endlich viele Zahlen darstellen. Wenn man mit reellen Zahlen rechnen will, muss man sich auf die Teilmenge der sogenannten *Gleitkommazahlen* beschränken. Reelle Zahlen werden dazu auf die nächstgelegene Gleitkommazahl abgebildet *(Rundung)*. Insbesondere Ergebnisse der arithmetischen Operationen müssen in der Regel gerundet werden, da die Menge der Gleitkommazahlen unter den üblichen arithmetischen Funktionen nicht abgeschlossen ist.

Im Zusammenhang mit der Intervallarithmetik verwenden wir jedoch nicht die Rundung zur nächstgelegenen Gleitkommazahl, sondern die *nach außen gerichtete Rundung*. Hierbei werden die linken Intervallgrenzen stets nach unten, die rechten Intervallgrenzen nach oben gerundet. Dadurch wird sichergestellt, dass der exakte Wert immer im Intervall enthalten ist.

2.4 Newton-Verfahren

Wir wollen nun das Intervall-Newton-Verfahren zur Bestimmung von Nullstellen differenzierbarer Funktionen vorstellen. Diese Variante des herkömmlichen Newton-Verfahrens benutzt die Intervallarithmetik, um verschiedene Nachteile des herkömmlichen Verfahrens zu vermeiden.

Ein kurzer Überblick über das herkömmliche als auch das Intervall-Newton-Verfahren findet sich in [Lec91]. Eine ausführliche Darstellung und verwandte Methoden sind in [AH83] zu finden.

Newton-Verfahren in einer Dimension

Das Newton-Verfahren ermöglicht die Approximation der Nullstellen einer stetig differenzierbaren Funktion $f : D \to \mathbb{R}$ mit $D \subseteq \mathbb{R}$. Ausgehend von einem Startwert $x^{(0)} \in \mathbb{R}$ wendet man die Iterationsvorschrift

$$x^{(i+1)} = x^{(i)} - \frac{f(x^{(i)})}{f'(x^{(i)})} \qquad i \in \mathbb{N}_0 \tag{2.4}$$

an. Als Abbruchkriterium bietet sich $|f(x^{(i)})| < \varepsilon$ oder $|x^{(i+1)} - x^{(i)}| < \varepsilon$ an. Abbildung 2.5 zeigt eine graphische Interpretation der Iterationsvorschrift. Die Konvergenz des Newton-Verfahrens ist quadratisch, wenn der Startwert $x^{(0)}$ nahe genug an einer einfachen Nullstelle von f liegt.

Ein Nachteil des Newton-Verfahrens besteht darin, dass man für eine gute Wahl des Startwertes die ungefähre Lage der Nullstellen kennen muss. Außerdem muss das Verfahren auf jede Nullstelle separat angewendet werden. Ein weiterer Nachteil ist die Tatsache, dass man keine Abschätzung über die Approximationsgüte hat. Alle diese Nachteile sind bei dem Intervall-Newton-Verfahren nicht vorhanden.

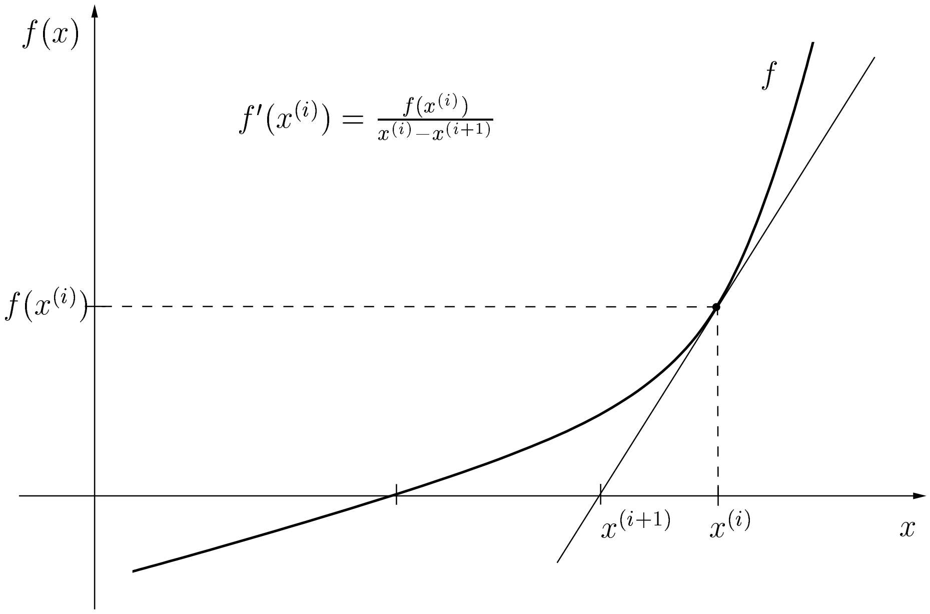

Abbildung 2.5: Newton-Schritt

Intervall-Newton-Verfahren in einer Dimension

Das Intervall-Newton-Verfahren berechnet alle Nullstellen einer stetig differenzierbaren Funktion $f : D \to \mathbb{R}$ mit $D \subseteq \mathbb{R}$ in einem Startintervall $X^{(0)} \in \mathbb{IR}$. Die Iterationsvorschrift lautet

$$X^{(i+1)} = X^{(i)} \cap \left(m(X^{(i)}) - \frac{f(m(X^{(i)}))}{F'(X^{(i)})} \right) \qquad i \in \mathbb{N}_0\,. \tag{2.5}$$

Falls die Schnittbildung die leere Menge liefert, enthält $X^{(i)}$ keine Nullstellen von f. Als Abbruchkriterium verwendet man

$$\overline{X^{(i)}} - \underline{X^{(i)}} < \varepsilon \quad \text{ oder } \quad \underline{F(X^{(i)})} > -\varepsilon \ \wedge\ \overline{F(X^{(i)})} < \varepsilon.$$

Im Fall $0 \in F'(X^{(i)})$ muss die erweiterte Intervallarithmetik verwendet werden (siehe Kapitel 2.3). Falls dabei zwei Intervalle für $X^{(i+1)}$ auftreten, werden diese anschließend getrennt weiter iteriert.
Eine graphische Darstellung der Iterationsvorschrift (2.5) ist in den Abbildungen 2.6 und 2.7 zu sehen. In jedem Iterationsschritt werden an der Stelle $m(X^{(i)})$ zwei Geraden an den Graph der Funktion f gelegt. Diese beiden Geraden haben die Steigung $\underline{F'(X^{(i)})}$ bzw. $\overline{F'(X^{(i)})}$. Die Schnittpunkte der Geraden mit der x-Achse liefern die Intervallgrenzen für ein neues Intervall, das mit $X^{(i)}$ geschnitten wird, um $X^{(i+1)}$ zu erhalten.
In Abbildung 2.6 ist der Fall $0 \notin F'(X^{(i)})$ skizziert. Der Newton-Schritt berechnet zunächst das Intervall $[\alpha, \beta]$ und nach dem Schnitt mit $X^{(i)}$ erhält man $X^{(i+1)} = [\underline{X^{(i)}}, \beta]$.

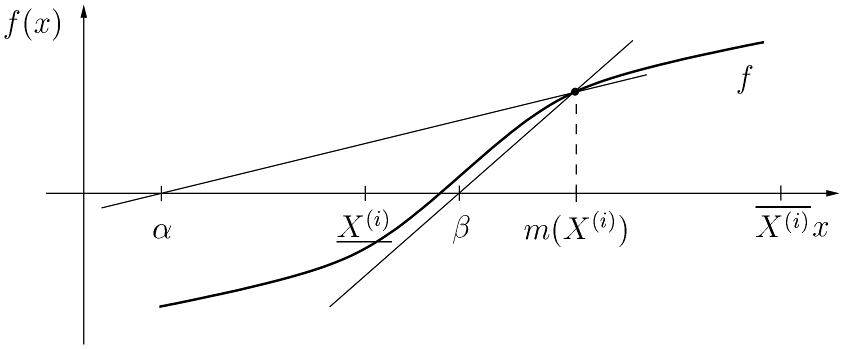

Abbildung 2.6: Intervall-Newton-Schritt mit $0 \notin F'(X^{(i)})$

Abbildung 2.7 zeigt den Fall $0 \in F'(X^{(i)})$. Die erweiterte Intervallarithmetik liefert die Menge $(-\infty, \alpha] \cup [\beta, \infty)$. Nach dem Schnitt mit $X^{(i)}$ erhält man $X^{(i+1)} = [\underline{X^{(i)}}, \alpha] \cup [\beta, \overline{X^{(i)}}]$.

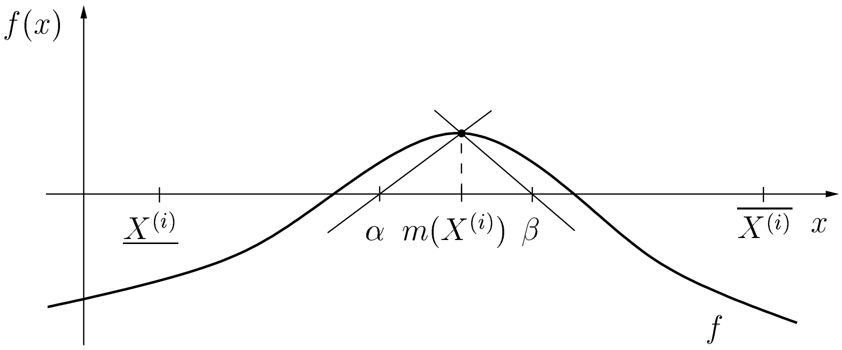

Abbildung 2.7: Intervall-Newton-Schritt mit $0 \in F'(X^{(i)})$

Newton-Verfahren in mehreren Dimensionen

Das Newton-Verfahren lässt sich ohne Probleme auf eine stetig differenzierbare Funktion $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^n$ verallgemeinern. Die Iterationsvorschrift lautet dann

$$\boldsymbol{x}^{(i+1)} = \boldsymbol{x}^{(i)} - \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{x}^{(i)})^{-1} \cdot \boldsymbol{f}(\boldsymbol{x}^{(i)}) \qquad i \in \mathbb{N}_0 \,. \tag{2.6}$$

Dabei bezeichnet $\mathbf{J}_{\boldsymbol{f}}$ die Jacobi-Matrix von $\boldsymbol{f}$:

$$\mathbf{J}_{\boldsymbol{f}} = \left(\frac{\partial \boldsymbol{f}_i}{\partial \boldsymbol{x}_j} \right)_{i,j=1..n} .$$

Um die aufwendige Invertierung der Matrix $\mathbf{J}_{\boldsymbol{f}}$ zu vermeiden, löst man stattdessen das lineare Gleichungssystem

$$\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{x}^{(i)}) \cdot (\boldsymbol{x}^{(i+1)} - \boldsymbol{x}^{(i)}) = -\boldsymbol{f}(\boldsymbol{x}^{(i)}) \tag{2.7}$$

nach $\boldsymbol{x}^{(i+1)}$ auf.

Die Nachteile des Newton-Verfahrens in einer Dimension sind auch hier vorhanden: Für eine gute Wahl des Startwertes muss man die Lage der Nullstelle kennen. Außerdem liefert das Verfahren zunächst immer nur eine Nullstelle. Auch hat man keine Abschätzung über die Approximationsgüte der Nullstellen. Bei dem Intervall-Newton-Verfahren in mehreren Dimensionen sind diese Nachteile nicht vorhanden.

Intervall-Newton-Verfahren in mehreren Dimensionen

Das Intervall-Newton-Verfahren in mehreren Dimensionen bestimmt alle Nullstellen einer stetig differenzierbaren Funktion $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^n$ in dem n-dimensionalen Quader $\boldsymbol{X}^{(0)} \in \mathbb{IR}^n$. Die Iterationsvorschrift lautet

$$\boldsymbol{X}^{(i+1)} = \boldsymbol{X}^{(i)} \cap \left(\boldsymbol{m}(\boldsymbol{X}^{(i)}) - \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{-1}\boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)}))\right) \qquad i \in \mathbb{N}_0\,. \tag{2.8}$$

Dabei ist zu beachten, dass die Jacobi-Matrix $\mathbf{J}_{\boldsymbol{f}}$ auf einem Intervallvektor auszuwerten ist. In vielen Fällen gilt dabei $0 \in \det \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})$, d. h. die Jacobi-Matrix ist nicht invertierbar. Eine Möglichkeit besteht in der Verwendung der erweiterten Intervallarithmetik und der Adjungierten von $\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})$:

$$\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{-1} = \frac{1}{\det \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})} \;\operatorname{ad} \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})\,.$$

Es gibt jedoch geeignetere Verfahren, die wir nun vorstellen wollen. Dazu formulieren wir die Iterationsvorschrift zunächst in der allgemeinen Form

$$\boldsymbol{X}^{(i+1)} = \boldsymbol{X}^{(i)} \cap \boldsymbol{op}(\boldsymbol{X}^{(i)})\,,$$

wobei $\boldsymbol{op}$ einer der folgenden Operatoren ist:

- der Newton-Operator $\boldsymbol{N}$,
- der Krawczyk-Operator $\boldsymbol{K}$,
- der Jacobi-Operator $\boldsymbol{J}$,
- der Gauss-Seidel-Operator $\boldsymbol{GS}$ oder
- der modifizierte Gauss-Seidel-Operator $\boldsymbol{MGS}$.

Dabei entspricht der *Newton-Operator* $\boldsymbol{N}$ der Iterationsvorschrift (2.8), es gilt also

$$\boldsymbol{N}(\boldsymbol{X}^{(i)}) = \boldsymbol{m}(\boldsymbol{X}^{(i)}) - \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{-1}\boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)}))\,.$$

Der *Krawczyk-Operator* $\boldsymbol{K}$ vermeidet vollständig die Inversion einer Intervallmatrix:

$$\boldsymbol{K}(\boldsymbol{X}^{(i)}) = \boldsymbol{m}(\boldsymbol{X}^{(i)}) - \boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)})) + \left(\mathbf{I} - \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})\right)\left(\boldsymbol{X}^{(i)} - \boldsymbol{m}(\boldsymbol{X}^{(i)})\right)\,.$$

Die restlichen drei Operatoren betrachten das lineare Gleichungssystem

$$\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)}) \cdot \left(\boldsymbol{X}^{(i)} - \boldsymbol{m}(\boldsymbol{X}^{(i)})\right) = -\boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)}))\,. \tag{2.9}$$

Zur Vereinfachung der Schreibweise sei im Folgenden stets $\mathbf{A}^{(i)} := \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})$, $\boldsymbol{x}^{(i)} := \boldsymbol{m}(\boldsymbol{X}^{(i)})$ und $\boldsymbol{b}^{(i)} := \boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)}))$. Das Gleichungssystem (2.9) lautet also

$$\mathbf{A}^{(i)} \cdot \left(\boldsymbol{X}^{(i)} - \boldsymbol{x}^{(i)}\right) = -\boldsymbol{b}^{(i)}.$$

Der *Jacobi-Operator* $\boldsymbol{J}$ löst die j-te Zeile des Gleichungssystems (2.9) nach $\boldsymbol{X}_j^{(i)}$ auf und verwendet den erhaltenen Term zur Berechnung von $\boldsymbol{X}_j^{(i+1)}$.

$$\begin{aligned}
\boldsymbol{Y}_j^{(i)} &= \boldsymbol{x}_j^{(i)} - \left(\frac{\boldsymbol{b}_j^{(i)} + \sum_{k \neq j} \mathbf{A}_{j,k}^{(i)}(\boldsymbol{X}_k^{(i)} - \boldsymbol{x}_k^{(i)})}{\mathbf{A}_{j,j}^{(i)}}\right) \\
\boldsymbol{J}(\boldsymbol{X}^{(i)}) &= (\boldsymbol{Y}_j^{(i)})_{j=1\ldots n}
\end{aligned}$$

Der *Gauss-Seidel-Operator* $\boldsymbol{GS}$ stellt eine Verbesserung des Jacobi-Operators dar, indem er bei der Berechnung der $\boldsymbol{Y}_j^{(i)}$ bereits die Werte $\boldsymbol{Y}_k^{(i)}$ mit $k < j$ berücksichtigt.

$$\begin{aligned}
\boldsymbol{Y}_j^{(i)} &= \boldsymbol{x}_j^{(i)} - \left(\frac{\boldsymbol{b}_j^{(i)} + \sum_{k<j} \mathbf{A}_{j,k}^{(i)}(\boldsymbol{Z}_k^{(i)} - \boldsymbol{x}_k^{(i)}) + \sum_{k>j} \mathbf{A}_{j,k}^{(i)}(\boldsymbol{X}_k^{(i)} - \boldsymbol{x}_k^{(i)})}{\mathbf{A}_{j,j}^{(i)}}\right) \\
\boldsymbol{Z}_j^{(i)} &= \boldsymbol{Y}_j^{(i)} \cap \boldsymbol{X}_j^{(i)}
\end{aligned}$$

$$\boldsymbol{GS}(\boldsymbol{X}^{(i)}) = (\boldsymbol{Z}_j^{(i)})_{j=1\ldots n}$$

Der *modifizierte Gauss-Seidel-Operator* $\boldsymbol{MGS}$ liefert noch bessere Werte als der Gauss-Seidel-Operator, allerdings ist dazu ein wesentlich höherer Rechenaufwand nötig. Dieser Mehraufwand ist gerechtfertigt, wenn die Auswertung der Funktion und deren Jacobi-Matrix im Vergleich zu den eigentlichen Rechenoperationen für das Newton-Verfahren sehr aufwendig ist. Der höhere Rechenaufwand liefert wesentlich bessere Einschließungen der Nullstellen.
Anstatt nur die l-te Zeile des Gleichungssystems (2.9) für die Bestimmung von $\boldsymbol{Y}_j^{(i)}$ zu verwenden, werden nun alle Zeilen des Gleichungssystems nach $\boldsymbol{X}_j^{(i)}$ aufgelöst. Das Auflösen der l-ten Zeile liefert den Wert $\boldsymbol{Y}_{j,l}^{(i)}$. Alle Werte $\boldsymbol{Y}_{j,l}^{(i)}$ $j = 1 \ldots n$ werden dann miteinander geschnitten.

$$\begin{aligned}
\boldsymbol{Y}_{j,l}^{(i)} &= \boldsymbol{x}_j^{(i)} - \left(\frac{\boldsymbol{b}_l^{(i)} + \sum_{k<j} \mathbf{A}_{l,k}^{(i)}(\boldsymbol{Z}_k^{(i)} - \boldsymbol{x}_k^{(i)}) + \sum_{k>j} \mathbf{A}_{l,k}^{(i)}(\boldsymbol{X}_k^{(i)} - \boldsymbol{x}_k^{(i)})}{\mathbf{A}_{l,j}^{(i)}}\right) \\
\boldsymbol{Z}_j^{(i)} &= (\boldsymbol{Y}_{j,1}^{(i)} \cap \ldots \cap \boldsymbol{Y}_{j,n}^{(i)}) \cap \boldsymbol{X}_j^{(i)}
\end{aligned}$$

$$\boldsymbol{MGS}(\boldsymbol{X}^{(i)}) = (\boldsymbol{Z}_j^{(i)})_{j=1\ldots n}$$

Die letzten drei Operatoren benutzen dabei ggfs. die erweiterte Intervallarithmetik zur Auswertung der Terme für die Variablen $\boldsymbol{Y}_j^{(i)}$ bzw. $\boldsymbol{Y}_{j,l}^{(i)}$.

Die Beziehungen zwischen den Operatoren wird durch folgendes Lemma beschrieben.

Lemma 2.9. *Sei $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^n$ eine stetig differenzierbare Funktion und $\boldsymbol{X} \in \mathbb{IR}^n$. Dann gilt:*

$$\boldsymbol{MGS}(\boldsymbol{X}) \subseteq \boldsymbol{GS}(\boldsymbol{X}) \subseteq \boldsymbol{J}(\boldsymbol{X}) \qquad \textit{und} \qquad \boldsymbol{GS}(\boldsymbol{X}) \subseteq \boldsymbol{K}(\boldsymbol{X}).$$

Beweis:
Die erste Behauptung folgt aus der Inklusionsisotonie der Intervalloperationen und der Definition der Operatoren. Ein Beweis für die zweite Behauptung findet sich in [Neu90]. □

Das (Intervall-)Newton-Verfahren kann auch verwendet werden, um die Nullstellen einer stetig differenzierbaren Funktion $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^m$ mit $m > n$ zu bestimmen. Das Gleichungssystem (2.9) besteht dann aus m Gleichungen für n Variablen. Durch Multiplikation beider Seiten mit der transponierten Jacobi-Matrix erhält man das quadratische System

$$\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{\mathrm{T}}\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)}) \cdot \Big(\boldsymbol{X}^{(i)} - \boldsymbol{m}(\boldsymbol{X}^{(i)})\Big) = -\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)})). \qquad (2.10)$$

In den Definitionen der Operatoren ist also

$$\begin{array}{ccccc} \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)}) & \text{durch} & \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{\mathrm{T}}\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)}) & \text{und} \\ \boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)})) & \text{durch} & \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)})) & \end{array}$$

zu ersetzen. Analog gilt

$$\begin{array}{rcll} \mathbf{A}^{(i)} & := & \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{\mathrm{T}}\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)}) & \text{und} \\ \boldsymbol{b}^{(i)} & := & \mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{m}(\boldsymbol{X}^{(i)})). & \end{array}$$

Durch die zusätzliche Matrixmultiplikation verbreitern sich allerdings die Intervalle in den Komponenten der Matrix $\mathbf{A}^{(i)}$ und des Vektors $\boldsymbol{b}^{(i)}$. Dies wirkt sich nachteilig auf die Konvergenzgeschwindigkeit des Verfahrens aus.
Bei Verwendung des modifizierten Gauss-Seidel-Operators kann jedoch die Multiplikation mit der transponierten Jacobi-Matrix entfallen, da dieser Operator kein quadratisches System voraussetzt. Dies ist ein wesentlicher Grund für die Effizienz des modifizierten Gauss-Seidel-Operators bei Funktionen der Form $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^m$ mit $m > n$.

Wir wollen noch ein Problem beim Einsatz des Intervall-Newton-Verfahrens ansprechen. Das herkömmliche Newton-Verfahren garantiert (quadratische) Konvergenz, vorausgesetzt, der Startvektor $\boldsymbol{x}_0$ liegt hinreichend nahe an der Nullstelle. Beim Intervall-Newton-Verfahren hängt die Konvergenzgeschwindigkeit wesentlich von der Qualität der Intervallauswertung von $\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})$ ab, d. h. je geringer die Überschätzung bei der Berechnung von $\mathbf{J}_{\boldsymbol{f}}(\boldsymbol{X}^{(i)})$, desto besser die Konvergenz.
Bei großer Überschätzung kann sogar der Fall $\boldsymbol{X}^{(i+1)} = \boldsymbol{X}^{(i)}$ eintreten. In einem solchen Fall zerlegt man den vom Intervallvektor $\boldsymbol{X}^{(i)}$ beschriebenen

n-dimensionalen Quader in mehrere kleinere Quader, die anschließend getrennt auf Nullstellen von $\boldsymbol{f}$ untersucht werden.
Eine Möglichkeit der Unterteilung besteht darin, den Intervallvektor $\boldsymbol{X}^{(i)}$ in jeder Dimension zu halbieren. Dabei einstehen 2^n neue Intervallvektoren, auf die das Intervall-Newton-Verfahren einzeln angewendet wird. Für größere Werte n erhält man mit dieser Strategie jedoch unverhältnismäßig viele Intervallvektoren. Bei einer anderen Methode wird der Vektor $\boldsymbol{X}^{(i)}$ nur in *einer* Dimension halbiert, z. B. in der Dimension, deren Komponente von $\boldsymbol{X}^{(i)}$ am größten ist. Ein besseres Kriterium zur Wahl der optimalen Halbierungsrichtung findet sich in [Rat92], ebenso wie eine Unterteilungsstrategie, die $\boldsymbol{X}^{(i)}$ in $n+1$ Teile zerlegt.

2.5 Rückrechnung

Unter *Rückrechnung* verstehen wir folgendes Problem: Gegeben sei eine NURBS-Kurve $\boldsymbol{C}$ oder eine NURBS-Fläche $\boldsymbol{S}$ und ein Punkt $\boldsymbol{P} \in \mathbb{R}^3$, der auf der Kurve bzw. der Fläche liegt. Gesucht sind die zu dem Punkt $\boldsymbol{p}$ gehörenden Parameterwerte, d.h. $u \in [0,1]$ mit $\boldsymbol{C}(u) = \boldsymbol{p}$ bzw. $(u,v) \in [0,1]^2$ mit $\boldsymbol{S}(u,v) = \boldsymbol{p}$.

Dieses Problem taucht beim Arbeiten mit NURBS sehr häufig auf. Üblicherweise arbeitet man nicht direkt mit den NURBS-Flächen selbst, sondern mit Flächenstücken, die auf einer NURBS-Fläche eingebettet sind. Abbildung 2.8 zeigt ein Flächenstück f auf einer kugelförmigen NURBS-Fläche. Das Flächenstück wird dabei durch die Angabe der Begrenzungskanten, sogenannter *Trimmkurven*, spezifiziert. Dabei handelt es sich um orientierte geschlossene Kantenzüge, die den Rand des Flächenstückes definieren. Das Flächenstück wird in diesem Zusammenhang auch als *getrimmte NURBS-Fläche* bezeichnet.

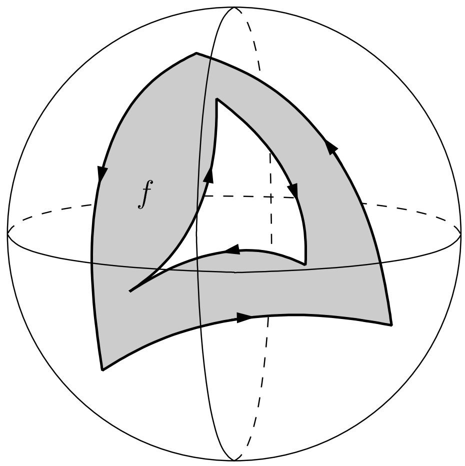

Abbildung 2.8: Eine getrimmte NURBS-Fläche

Betrachten wir nun die Visualisierung einer solchen getrimmten NURBS-Fläche. Gängige Graphikbibliotheken wie z. B. OpenGL bieten die Möglichkeit, solche

Objekte direkt zu visualisieren. Allerdings müssen dabei die Trimmkurven im Parameterraum der Fläche spezifiziert werden.
Eine einfache Methode zur Berechnung dieser Darstellung funktioniert wie folgt: Zunächst werden ausreichend viele Punkte auf den Begrenzungskanten generiert. Für jeden dieser Punkte bestimmt man die zugehörigen (u, v)-Parameter. Anschließend werden die Parameterwerte der auf den Begrenzungskanten benachbarten Punkte durch Liniensegmente verbunden. Die so erhaltenen Polygonzüge im Parameterraum approximieren die Trimmkurven der NURBS-Fläche. Die Qualität der Approximation lässt sich durch Anzahl und Lage der Punkte auf den Trimmkurven steuern. Zentrales Problem bei dieser Methode ist die Rückrechnung, also die Bestimmung der (u, v)-Parameter für einen auf der NURBS-Fläche liegenden Punkt.

Wir werden im Folgenden das Problem der Rückrechnung speziell für NURBS-Flächen diskutieren. Sämtliche Aussagen lassen sich sinngemäß auf NURBS-Kurven übertragen. Insbesondere ist das Problem für NURBS-Kurven einfacher zu lösen, da es sich dort nur um univariate Funktionen handelt.
Ein anderes, etwas aufwendigeres Verfahren zur Rückrechnung wird von Dyllong und Luther in [DL00] beschrieben. Wir bemerken außerdem, dass für das Problem der Rückrechnung für quadratische Flächen und Kurven unter bestimmten Voraussetzungen explizite Lösungen existieren (siehe [PT97], Formel (7.38)).

Es sei also eine NURBS-Fläche $\boldsymbol{S}(u, v)$ und ein Punkt $\boldsymbol{p} \in \mathbb{R}^3$ auf dieser Fläche $\boldsymbol{S}$ gegeben. Wir setzen voraus, dass die Funktion $\boldsymbol{S}$ injektiv und C^1-stetig ist. Gesucht ist nun die Parameterdarstellung von $\boldsymbol{p}$, also das Paar $(u, v) \in [0, 1]^2$ mit $\boldsymbol{S}(u, v) = \boldsymbol{p}$.
Wir geben nun verschiedene Formulierungen des Problems an. Die erste Formulierung als Nullstellenproblem ergibt sich direkt aus der Aufgabenstellung:

$$
\begin{array}{lrcl}
 & \boldsymbol{S}(u,v) & = & \boldsymbol{p} \\
\Leftrightarrow & \boldsymbol{S}(u,v) - \boldsymbol{p} & = & 0 \\
\Leftrightarrow & \boldsymbol{S}_Z(u,v) - \boldsymbol{p}\, \boldsymbol{S}_N(u,v) & = & 0
\end{array}
\tag{2.11}
$$

Dabei bezeichnen $\boldsymbol{S}_Z$ und $\boldsymbol{S}_N$ den Zähler bzw. den Nenner in der Definition von $\boldsymbol{S}$ (siehe Kapitel 2.2.3).
Der Punkt $\boldsymbol{S}(u, v)$ hat unter allen Punkten der Fläche $\boldsymbol{S}$ den geringesten Abstand zum Punkt $\boldsymbol{P}$. Das zugehörige Optimierungsproblem lautet also

$$
\min_{(u,v)\in[0,1]^2} |\, \boldsymbol{S}(u,v) - \boldsymbol{p}\, |.
$$

Aufgrund der strengen Monotonie der Wurzelfunktion kann man auch

$$
\min_{(u,v)\in[0,1]^2} (\boldsymbol{S}(u,v) - \boldsymbol{p})^2 \tag{2.12}
$$

betrachten. An einer lokalen Extremstelle verschwinden die partiellen Ableitungen. Dies lässt sich wiederum als Nullstellenproblem formulieren:

$$\begin{aligned} &\left\{\begin{array}{rcl} \frac{\partial}{\partial u}(\boldsymbol{S}(u,v)-\boldsymbol{p})^2 &=& 0 \\ \frac{\partial}{\partial v}(\boldsymbol{S}(u,v)-\boldsymbol{p})^2 &=& 0 \end{array}\right. \\ \Leftrightarrow\ &\left\{\begin{array}{rcl} (\boldsymbol{S}(u,v)-\boldsymbol{p})\ \frac{\partial}{\partial u}\boldsymbol{S}(u,v) &=& 0 \\ (\boldsymbol{S}(u,v)-\boldsymbol{p})\ \frac{\partial}{\partial v}\boldsymbol{S}(u,v) &=& 0 \end{array}\right. \end{aligned} \tag{2.13}$$

Die letzten beiden Formulierungen (2.12) und (2.13) betrachten den Abstand des Punktes $\boldsymbol{p}$ zu der Fläche $\boldsymbol{S}$. Hier ist es nicht wirklich notwendig, dass $\boldsymbol{p}$ tatsächlich auf der Fläche $\boldsymbol{S}$ liegt. Beispielsweise ist es denkbar, dass die Koordinaten des Punktes $\boldsymbol{p}$ mit Rundungsfehlern behaftet sind. Die letzten zwei Formulierungen sind damit robuster gegenüber numerischen Ungenauigkeiten. Allerdings hat sich gezeigt, dass die erste Formulierung (2.11) effizienter ist. Dies liegt an den wesentlich einfacheren Termen für die Funktion und die ersten Ableitungen. Bei komplizierteren Termen nimmt die durch die Intervallarithmetik bedingte Überschätzung stark zu und verschlechtert so das Konvergenzverhalten des Intervall-Newton-Verfahrens.
Wir werden später ein weiteres Verfahren angeben, um auch bei Verwendung der ersten Formulierung numerische Ungenauigkeiten zu behandeln.

Gesucht ist also die (einzige) Nullstelle der Funktion

$$\boldsymbol{f}:[0,1]^2 \to \mathbb{R}^3, \boldsymbol{f}(u,v) = \boldsymbol{S}_Z(u,v) - \boldsymbol{p}\ \boldsymbol{S}_N(u,v). \tag{2.14}$$

Diese Funktion ist ein stückweises bivariates Polynom vom Grad (p,q), wobei p und q den Grad der NURBS-Fläche in u-Richtung bzw. v-Richtung beschreiben. Für die partiellen Ableitung der Funktion (2.14) gilt

$$\begin{aligned} \frac{\partial}{\partial u}\boldsymbol{f}(u,v) &= \frac{\partial}{\partial u}\boldsymbol{S}_Z(u,v) - \boldsymbol{p}\,\frac{\partial}{\partial u}\boldsymbol{S}_N(u,v) \\ \frac{\partial}{\partial v}\boldsymbol{f}(u,v) &= \frac{\partial}{\partial v}\boldsymbol{S}_Z(u,v) - \boldsymbol{p}\,\frac{\partial}{\partial v}\boldsymbol{S}_N(u,v) \end{aligned}$$

Wir verwenden das Intervall-Newton-Verfahren, um die Nullstellen der Funktion $\boldsymbol{f}$ zu bestimmen. Die Jacobi-Matrix $\mathbf{J}_{\boldsymbol{f}}$ ist eine 3×2-Matrix, also insbesondere nicht quadratisch. Daher muss bei allen Operatoren, mit Ausnahme des modifizierten Gauss-Seidel-Operators $\boldsymbol{MGS}$, das Gleichungssystem (2.9) mit der transponierten Jacobi-Matrix multipliziert werden (siehe Seite 20).

Die Tabelle 2.1 zeigt für einige Beispiele eine Übersicht über die benötigten Laufzeiten. Als Beispiele werden zunächst die quadratischen Flächen Kugel, Zylinder und Kegel verwendet. Der Torus ist keine quadratische Fläche, lässt sich aber durch eine NURBS-Fläche jeweils vom Grad 2 darstellen. Mit Welle bezeichnen wir eine NURBS-Fläche vom Grad 3 in u-Richtung als auch in v-Richtung. Für deren Kontrollpunkte $\boldsymbol{P}_{i,j}$ gilt

$$\boldsymbol{P}_{i,j} = \begin{pmatrix} i \\ j \\ \sin\left(3\pi\frac{i+j}{10}\right) \end{pmatrix} \qquad 0 \le i,j \le 5.$$

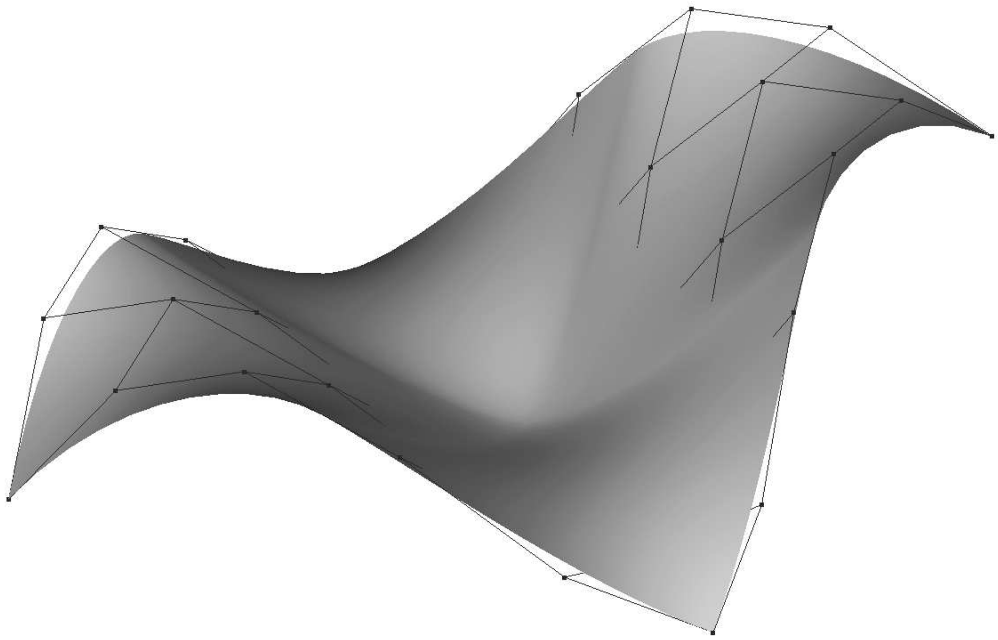

Abbildung 2.9: Wellenförmige NURBS-Fläche vom Grad 3

Alle Gewichte $w_{i,j}$ haben den Wert 1, die zugrunde liegenden Knotenvektoren sind $U = V = \{0, 0, 0, 0, \frac{1}{2}, \frac{1}{2}, 1, 1, 1, 1\}$. Eine Darstellung dieser Fläche inklusive der Kontrollpunkte ist in Abbildung 2.9 zu sehen. Benachbarte Kontrollpunkte wurden dabei durch Linien verbunden.

	Grad		Operator			
	p	q	$\boldsymbol{N}$	$\boldsymbol{K}$	$\boldsymbol{GS}$	$\boldsymbol{MGS}$
Kugel	2	2	50,4	88,3	34,8	14,3
Zylinder	2	1	12,0	37,4	21,8	7,6
Kegel	2	1	36,7	78,7	21,2	8,3
Torus	2	2	149,1	179,8	109,8	37,2
Welle	3	3	252,5	295,2	191,5	90,6

Tabelle 2.1: Laufzeiten für die Berechnung der Parameterwerte (u, v)

In der Tabelle ist die durchschnittliche Laufzeit für eine Rückrechnung beim Einsatz der jeweiligen Operatoren angegeben. Es wurden jeweils 500 Punkte auf der NURBS-Fläche zufällig ausgewählt, so dass die Parameter (u, v) gleichverteilt sind. Alle Zeiten sind in Millisekunden angegeben und wurden auf einen Pentium III mit 450 MHz Taktfrequenz und 128 MB RAM gemessen. Als Compiler wurde der gcc 2.95.3 unter Linux mit Kernel 2.2.19 verwendet.

Das Intervall-Newton-Verfahren wird abgebrochen, wenn alle Intervalllängen des Vektors $\boldsymbol{X}^{(i)}$ kleiner als 10^{-12} sind.
Die guten Werte für den Zylinder und den Kegel sind dadurch zu erklären, dass bei diesen Flächen die Basisfunktionen des Parameters v Polynome ersten Grades, also lineare Funktionen, sind. Die längsten Laufzeiten treten bei der wellenförmigen Fläche auf. Der Grund liegt darin, dass sowohl in u-Richtung als auch in v-Richtung Basisfunktionen dritten Grades verwendet werden.
Eine genauere Untersuchung zeigt außerdem, dass 75% bis 85% der Laufzeit für die Auswertung der Funktion $\boldsymbol{f}$ oder der partiellen Ableitungen $\frac{\partial}{\partial u}\boldsymbol{f}$ und $\frac{\partial}{\partial v}\boldsymbol{f}$ verwendet wird.

Wir wollen nun noch eine Möglichkeit skizzieren, um bei numerischen Ungenauigkeiten in den Ausgangsdaten geeignete Parameter u und v zu bestimmen. Seien also die Koordinaten des Punktes $\boldsymbol{p}$ leicht verfälscht (beispielsweise durch Rundung). Der Punkt $\boldsymbol{p}$ liegt also nicht mehr exakt auf der Fläche $\boldsymbol{S}$, sondern hat einen sehr geringen Abstand zu $\boldsymbol{S}$. Wünschenswert ist in einem solchen Fall die Parameterdarstellung eines Punktes, der in der Nähe von $\boldsymbol{p}$ liegt.
Wir ersetzen den Punkt $\boldsymbol{p}$ durch den achsenorientierten Würfel

$$\boldsymbol{P}_\varepsilon = \boldsymbol{p} + \begin{pmatrix} [-\varepsilon, \varepsilon] \\ [-\varepsilon, \varepsilon] \\ [-\varepsilon, \varepsilon] \end{pmatrix} \in \mathbb{R}^3 .$$

Den Würfel $\boldsymbol{P}_\varepsilon$ bezeichnen wir als ε-Verbreiterung von $\boldsymbol{p}$. Der Wert $\varepsilon > 0$ soll dabei so gewählt sein, dass sich der Würfel $\boldsymbol{P}_\varepsilon$ und die NURBS-Fläche $\boldsymbol{S}$ schneiden (vgl. Abbildung 2.10).

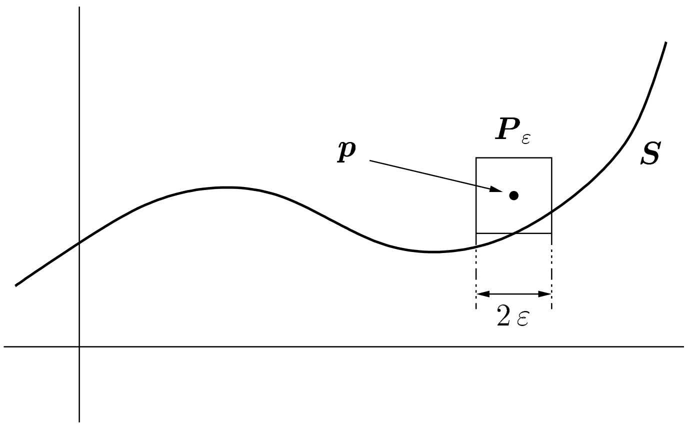

Abbildung 2.10: ε-Verbreiterung von $\boldsymbol{p}$ (Querschnitt)

Wir suchen nun ein Paar $(u,v) \in [0,1]^2$, so dass $\boldsymbol{S}(u,v)$ im Würfel $\boldsymbol{P}_\varepsilon$ liegt. Dazu betrachten wir an Stelle der Funktion (2.14) die Funktion

$$\boldsymbol{f}_\varepsilon : [0,1]^2 \to \mathbb{R}^3, \boldsymbol{f}_\varepsilon(u,v) = \boldsymbol{S}_Z(u,v) - \boldsymbol{P}_\varepsilon\, \boldsymbol{S}_N(u,v) . \qquad (2.15)$$

Dabei ist zu beachten, dass die Funktionswerte von $\boldsymbol{f}_\varepsilon$ Intervallvektoren sind. In diesem Zusammenhang verstehen wir unter einer Nullstelle ein Paar $(u,v) \in [0,1]^2$, so dass der Intervallvektor $\boldsymbol{f}_\varepsilon(u,v)$ den Nullvektor enthält.

Die Nullstellen der Funktion $\boldsymbol{f}_\varepsilon$ können ohne weiteres mit dem Intervall-Newton-Verfahren bestimmt werden. Dabei ist zu beachten, dass es nun keine isolierte Nullstelle mehr gibt. Vielmehr gibt es unendlich viele Nullstellen, die eine (in der Regel) zusammenhängende Teilmenge im Parameterraum bilden. Das Intervall-Newton-Verfahren liefert als Resultat eine Überdeckung dieser Nullstellenmenge. Aus dieser Überdeckung kann man nun ein Paar (u, v) auswählen und erhält die Parameterdarstellung eines Punktes, der in der Nähe von $\boldsymbol{p}$ liegt.

Kapitel 3

Optimale Hüllkörper

In diesem Kapitel wollen wir die Berechnung der optimalen Hüllkörper erläutern. Dazu ist zunächst notwendig, die Begriffe *Körper* und *Hüllkörper* zu präzisieren.

Definition 3.1. *Eine nichtleere Punktmenge $\mathcal{K} \subseteq \mathbb{R}^3$ heißt* Körper, *wenn gilt:*

1. *$\mathcal{K}$ ist kompakt, also abgeschlossen und beschränkt,*
2. *$\mathcal{K}$ ist zusammenhängend und*
3. *$\mathcal{K} = \overline{int(\mathcal{K})}$.*

Dabei bezeichnet $int(\mathcal{K})$ das Innere der Menge $\mathcal{K}$ und $\overline{\mathcal{M}}$ den Abschluss der Menge $\mathcal{M}$.

Die letzte Forderung besagt, dass kein Teil der Punktmenge $\mathcal{K}$ zu einer Fläche, einer Kante oder einem Punkt entartet ist. Wir bemerken außerdem, dass für unsere Zwecke die zweite Forderung eigentlich nicht notwendig ist. Wir bezeichnen daher auch nichtzusammenhängende Punktmengen als Körper, solange die anderen Forderungen erfüllt sind. Im Folgenden werden die Begriffe Körper und Objekt synonym benutzt.

Als *Hüllkörper* bezeichnen wir einfache geometrische Körper, die den eigentlichen Körper einschließen.

Definition 3.2. *Ein Körper $\mathcal{H}$ heißt* Hüllkörper *für einen Körper $\mathcal{K}$, wenn $\mathcal{H} \supseteq \mathcal{K}$ gilt.*

In dieser Arbeit verwenden wir Kugeln und beliebig orientierte Quader als Hüllkörper. Andere Möglichkeiten für Hüllkörperformen sind Zylinder und Kugelkappen (siehe [KLMP98], [Sch94]).
Einen Hüllkörper bezeichnen wir als *optimal*, wenn sein Volumen minimal ist. Wir werden später sehen, dass bei Verwendung von Kugeln der optimale Hüllkörper eindeutig bestimmt ist, während es für den kleinsten einschließenden Quader mehrere Möglichkeiten geben kann.

3.1 Kleinste einschließende Kugeln

Wir wollen nun für einen gegebenen Körper $\mathcal{K}$ die kleinste einschließende Kugel berechnen. Für das Volumen V einer Kugel mit Radius r gilt $V(r) = \frac{4}{3}\pi r^3$. Das Kugelvolumen ist demnach eine streng monoton wachsende Funktion. Um das Volumen zu minimieren, genügt es also den Kugelradius r zu minimieren.

Bei fest vorgegebenem Kugelmittelpunkt $\boldsymbol{c}$ wird der minimale Kugelradius durch den maximalen Abstand zwischen dem Punkt $\boldsymbol{c}$ und allen Punkten des Körpers $\mathcal{K}$ bestimmt (vgl. Abbildung 3.1). Die Kugel mit Radius r_1 schließt das Objekt ein, ebenso die kleinere Kugel mit Radius r_2. Die Kugel mit Radius r_3 stellt hingegen keinen Hüllkörper dar, weil sie das Objekt nicht einschließt.

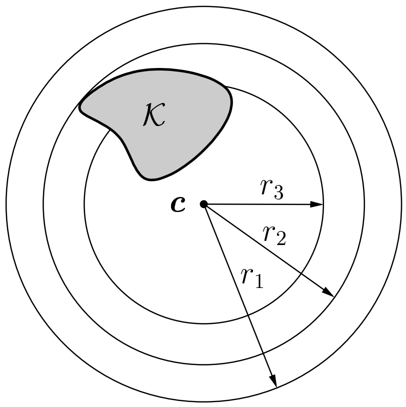

Abbildung 3.1: Kugeln mit vorgegebenem Mittelpunkt (Querschnitt)

Im folgenden Abschnitt werden wir die *elementaren Abstandsprobleme* behandeln. Wir zeigen, wie der maximale Abstand zwischen einem Punkt $\boldsymbol{p} \in \mathbb{R}^3$ und allen Punkten einer NURBS-Kurve $\boldsymbol{C}$ bzw. NURBS-Fläche $\boldsymbol{S}$ berechnet werden kann.

Da die Oberfläche des Körpers $\mathcal{K}$ in der Regel aber aus Kurven- und Flächen*stücken* besteht, müssen wir entscheiden können, ob ein "maximal entfernter Punkt" auch auf dem jeweiligen Kurven- bzw. Flächenstück liegt. Dieses Problem wird in dem zweiten Abschnitt über *Enthaltenseinstests* behandelt.

Anschließend zeigen wir, wie der maximale Abstand zwischen einem Punkt $\boldsymbol{p} \in \mathbb{R}^3$ und allen Punkten eines Körpers $\mathcal{K}$ berechnet werden kann und diskutieren einige Eigenschaften der zugrunde liegenden mathematischen Funktion. Danach wird die Verwendung lokaler Optimierungsverfahren zur Berechnung der *kleinsten* einschließenden Kugel besprochen.

Im letzten Abschnitt werden noch einige Möglichkeiten angesprochen, um die notwendigen Berechnungen zu beschleunigen.

3.1.1 Elementare Abstandsprobleme

Gegeben sei ein Punkt $\boldsymbol{p} \in \mathbb{R}^3$ und eine Kurve $\mathcal{C}$ oder eine Fläche $\mathcal{F}$ im $\mathbb{R}^3$. Die Kurve bzw. Fläche soll durch NURBS beschrieben werden können.

Gesucht sind alle lokalen Extrema der Funktion

$$f_{\boldsymbol{p}} : \mathcal{X} \to \mathbb{R},\ f_{\boldsymbol{p}}(\boldsymbol{x}) = |\,\boldsymbol{p} - \boldsymbol{x}\,| \text{ mit } \mathcal{X} = \mathcal{C} \text{ bzw. } \mathcal{X} = \mathcal{F}. \tag{3.1}$$

Aufgrund der strengen Monotonie der Wurzelfunktion sind diese identisch mit den lokalen Extrema der Funktion

$$g_{\boldsymbol{p}} : \mathcal{X} \to \mathbb{R},\ g_{\boldsymbol{p}}(\boldsymbol{x}) = f_{\boldsymbol{p}}(\boldsymbol{x})^2 = |\,\boldsymbol{p} - \boldsymbol{x}\,|^2 \text{ mit } \mathcal{X} = \mathcal{C} \text{ bzw. } \mathcal{X} = \mathcal{F}. \tag{3.2}$$

Im Folgenden soll nun erläutert werden, wie die lokalen Extrema für die verschieden Kurven- und Flächentypen berechnet werden. Für Spezialfälle wie Geraden, Ellipsen, Ebenen, Kugeln, Kegel, Zylinder und Tori geben wir explizite Lösungen an. Die allgemeinen Fälle der NURBS-Kurve und NURBS-Fläche werden als Nullstellenproblem formuliert.

Abstand Punkt – Gerade

Die Gerade sei durch $\boldsymbol{x} = \boldsymbol{r} + \lambda\, \boldsymbol{d}$ mit $|\,\boldsymbol{d}\,| = 1$ gegeben. Das globale Minimum der Funktion (3.2) ist der Lotfußpunkt von $\boldsymbol{p}$ auf die Gerade:

$$\boldsymbol{x}_{min} = \boldsymbol{r} + \boldsymbol{d}^{\mathrm{T}}(\boldsymbol{p} - \boldsymbol{r})\, \boldsymbol{d}.$$

Weitere Extrema gibt es nicht.

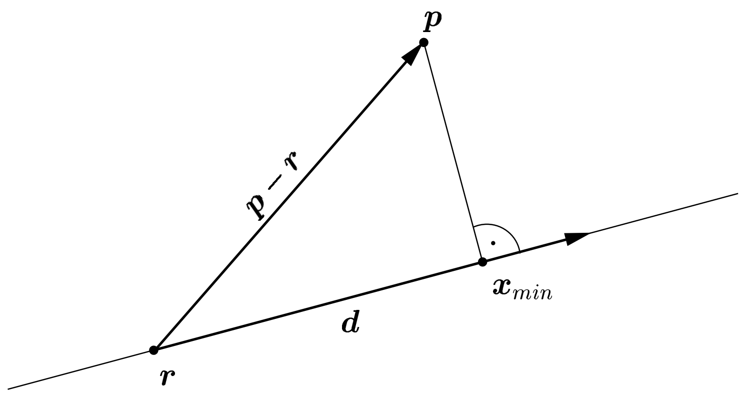

Abbildung 3.2: Abstand Punkt – Gerade

Abstand Punkt – Ellipse

Die Ellipse sei durch $\boldsymbol{x} = \boldsymbol{c} + \cos\varphi\, \boldsymbol{u} + \sin\varphi\, \boldsymbol{v}$ mit $\boldsymbol{u} \perp \boldsymbol{v}$ gegeben. Falls der Punkt $\boldsymbol{p}$ senkrecht über dem Mittelpunkt der Ellipse liegt, also $\boldsymbol{u}^{\mathrm{T}}(\boldsymbol{p} - \boldsymbol{c}) = 0$

und $\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c})=0$ gilt, dann gilt für die lokalen Extrema von (3.2):

$$\boldsymbol{x}_{min} = \begin{cases} \boldsymbol{c} \pm \boldsymbol{u}\ , & \text{falls } |\boldsymbol{u}| < |\boldsymbol{v}| \\ \boldsymbol{c} \pm \boldsymbol{v}\ , & \text{falls } |\boldsymbol{u}| > |\boldsymbol{v}| \end{cases} \qquad \boldsymbol{x}_{max} = \begin{cases} \boldsymbol{c} \pm \boldsymbol{v}\ , & \text{falls } |\boldsymbol{u}| < |\boldsymbol{v}| \\ \boldsymbol{c} \pm \boldsymbol{u}\ , & \text{falls } |\boldsymbol{u}| > |\boldsymbol{v}|\ . \end{cases}$$

Falls die Ellipse zu einem Kreis entartet, also $|\boldsymbol{u}| = |\boldsymbol{v}|$ gilt, sind alle Punkte der Kreislinie gleich weit von $\boldsymbol{p}$ entfernt.

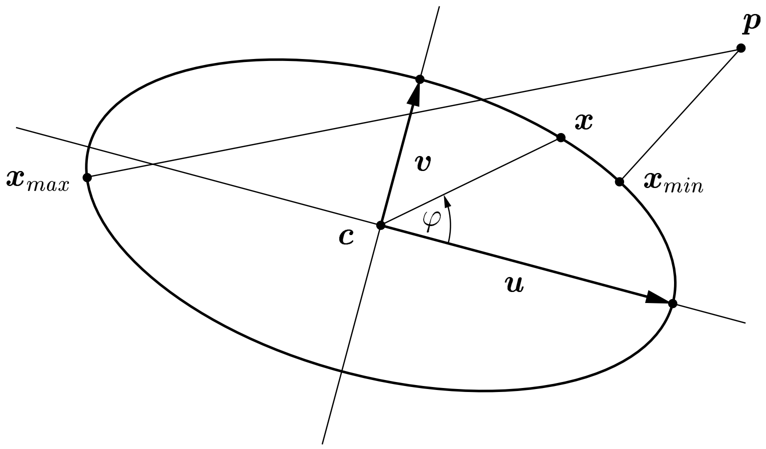

Abbildung 3.3: Abstand Punkt – Ellipse

Sei also nun o. B. d. A. $\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c}) \neq 0$ (ggfs. $\boldsymbol{u}$ durch $\boldsymbol{v}$ und $\boldsymbol{v}$ durch $-\boldsymbol{u}$ ersetzen). Um die Extrema der Funktion (3.2) zu bestimmen, betrachten wir die Ableitung nach φ:

$$\frac{d}{d\varphi}\, g_{\boldsymbol{p}}(\boldsymbol{x}) = 2\,(\boldsymbol{v}^2 - \boldsymbol{u}^2)\,\cos\varphi\,\sin\varphi + 2\,\boldsymbol{u}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c})\,\sin\varphi - 2\,\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c})\,\cos\varphi\,.$$

Durch die Substitution $\cos\varphi = \frac{1-t^2}{1+t^2}$ und $\sin\varphi = \frac{2t}{1+t^2}$ (mit $t = \tan\frac{\varphi}{2}$) erhalten wir ein Polynom vierten Grades $f(t) = \sum_{i=0}^{4} a_i t^i$ mit den Koeffizienten

$$\begin{aligned} a_4 &= \boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c}) \\ a_3 &= 2\,(\boldsymbol{u}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c}) + \boldsymbol{u}^2 - \boldsymbol{v}^2) \\ a_2 &= 0 \\ a_1 &= 2\,(\boldsymbol{u}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c}) + \boldsymbol{v}^2 - \boldsymbol{u}^2) \\ a_0 &= -\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c})\,. \end{aligned}$$

Die Nullstellen dieses Polynoms beschreiben die lokalen Extrema der Funktion (3.2). Aufgrund der Voraussetzung $\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c}) \neq 0$ gibt es keine lokalen Extrema mit $\varphi = \pi$, d. h. durch die Substitution sind keine lokalen Extrema verloren gegangen.
Falls die Ellipse zu einem Kreis entartet, also $|\boldsymbol{u}| = |\boldsymbol{v}|$ gilt, sind die lokalen Extrema durch

$$\boldsymbol{x}_{\substack{min\\max}} = \boldsymbol{c} \pm r\,\frac{\boldsymbol{p}-\boldsymbol{c}-\boldsymbol{n}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c})\boldsymbol{n}}{|\,\boldsymbol{p}-\boldsymbol{c}-\boldsymbol{n}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c})\boldsymbol{n}\,|} \qquad \text{mit } r = |\boldsymbol{u}| = |\boldsymbol{v}| \text{ und } \boldsymbol{n} = \frac{\boldsymbol{u}\times\boldsymbol{v}}{|\,\boldsymbol{u}\times\boldsymbol{v}\,|}$$

gegeben. Man beachte, dass der Nenner nicht Null wird, solange der Punkt $\boldsymbol{p}$ nicht senkrecht über dem Kreismittelpunkt liegt.

Abstand Punkt – NURBS-Kurve

Zur Bestimmung der lokalen Extrema von Funktion (3.2) betrachten wir die Ableitung nach u:

$$
\begin{aligned}
& \frac{d}{du}\big(\boldsymbol{p}-\boldsymbol{C}(u)\big)^2 = 0 \\
\Leftrightarrow\quad & \boldsymbol{C}'(u)^{\mathrm{T}}\,\big(\boldsymbol{p}-\boldsymbol{C}(u)\big) = 0 \\
\Leftrightarrow\quad & \big(\boldsymbol{C}_N(u)\,\boldsymbol{C}'_Z(u)-\boldsymbol{C}'_N(u)\,\boldsymbol{C}_Z(u)\big)^{\mathrm{T}}\,\big(\boldsymbol{p}\,\boldsymbol{C}_N(u)-\boldsymbol{C}_Z(u)\big) = 0
\end{aligned}
$$

Die lokalen Extrema sind also durch die Nullstellen der stückweise polynomialen Funktion

$$
f:[0,1]\to\mathbb{R},\ f(u)=\big(\boldsymbol{C}_N(u)\,\boldsymbol{C}'_Z(u)-\boldsymbol{C}'_N(u)\,\boldsymbol{C}_Z(u)\big)^{\mathrm{T}}\,\big(\boldsymbol{p}\,\boldsymbol{C}_N(u)-\boldsymbol{C}_Z(u)\big)
$$

gegeben. Hat die NURBS-Kurve $\boldsymbol{C}$ den Grad p, so hat die Funktion f den Grad $3p-1$. Sind die Gewichte w_i aller Kontrollpunkte gleich, so ist die Funktion $\boldsymbol{C}_N(u)$ konstant und der Grad von f reduziert sich auf $2p-1$.

Abstand Punkt – Ebene

Die Ebene sei durch $\boldsymbol{n}^{\mathrm{T}}(\boldsymbol{x}-\boldsymbol{r})=0$ mit $|\boldsymbol{n}|=1$ gegeben. Das globale Minimum der Funktion (3.2) ist der Lotfußpunkt von $\boldsymbol{p}$ auf die Ebene und berechnet sich durch

$$
\boldsymbol{x}_{min} = \boldsymbol{p} + \boldsymbol{n}^{\mathrm{T}}(\boldsymbol{r}-\boldsymbol{p})\,\boldsymbol{n}\,.
$$

Weitere Extrema gibt es nicht.

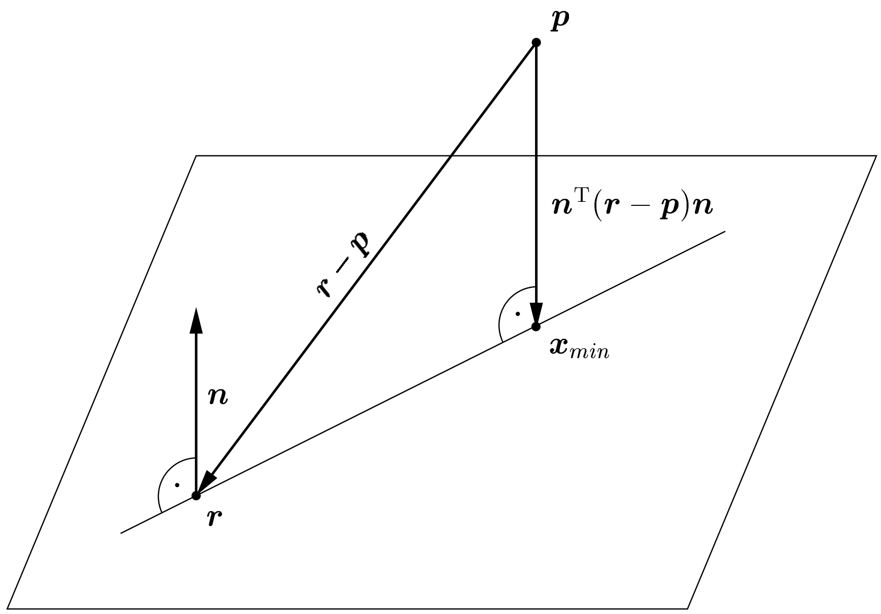

Abbildung 3.4: Abstand Punkt – Ebene

Abstand Punkt – Kugel

Die Kugel sei durch $|\boldsymbol{x} - \boldsymbol{c}| = r$ gegeben. Die lokalen Extrema der Funktion (3.2) sind die Schnittpunkte der Kugel mit der Geraden durch die Punkte $\boldsymbol{p}$ und $\boldsymbol{c}$:

$$\boldsymbol{x}_{\substack{min\\max}} = \boldsymbol{c} \pm r\,\frac{\boldsymbol{p}-\boldsymbol{c}}{|\boldsymbol{p}-\boldsymbol{c}|}.$$

Falls der Punkt $\boldsymbol{p}$ mit dem Kugelmittelpunkt $\boldsymbol{c}$ identisch ist, sind alle Punkte der Kugel gleich weit von $\boldsymbol{p}$ entfernt.

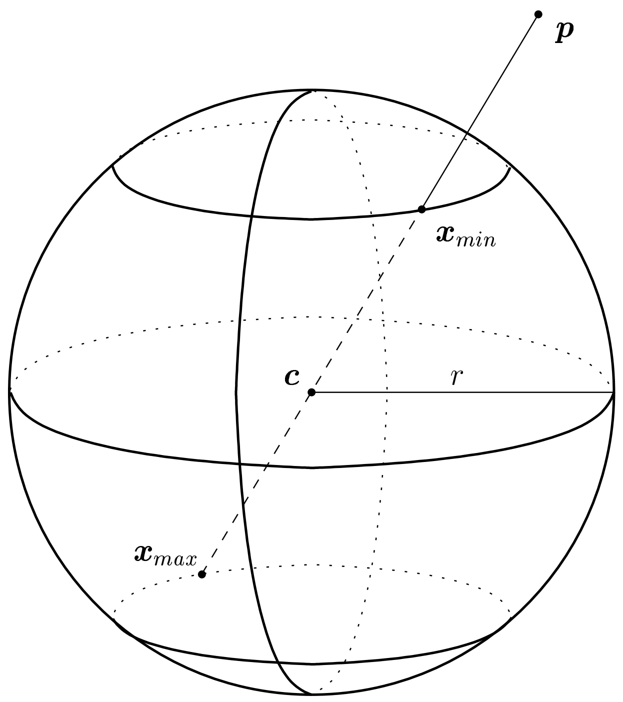

Abbildung 3.5: Abstand Punkt – Kugel

Abstand Punkt – Zylinder

Der Zylinder sei durch $\boldsymbol{x} = \boldsymbol{c} + \cos\varphi\,\boldsymbol{u} + \sin\varphi\,\boldsymbol{v} + \lambda\boldsymbol{n}$ mit $\boldsymbol{u} \perp \boldsymbol{v}$ und $\boldsymbol{n} = \frac{\boldsymbol{u}\times\boldsymbol{v}}{|\boldsymbol{u}\times\boldsymbol{v}|}$ gegeben. Die Funktion (3.2) hat keine lokalen Maxima. Alle lokalen Minima müssen offensichtlich in der Ebene durch den Punkt $\boldsymbol{p}$ mit dem Normalenvektor $\boldsymbol{n}$ liegen. Schneiden wir den Zylinder mit dieser Ebene, so erhalten wir eine Ellipse. Das Problem *Punkt – Zylinder* kann also auf das Problem *Punkt – Ellipse* reduziert werden.

Abstand Punkt – Kegel

Der Kegel sei durch $\boldsymbol{x} = \boldsymbol{a} + \lambda\,(\cos\varphi\,\boldsymbol{u} + \sin\varphi\,\boldsymbol{v} + \boldsymbol{w})$ mit $\lambda \geq 0$ gegeben, wobei die Vektoren $\boldsymbol{u}$, $\boldsymbol{v}$ und $\boldsymbol{w}$ ein Orthogonalsystem bilden.

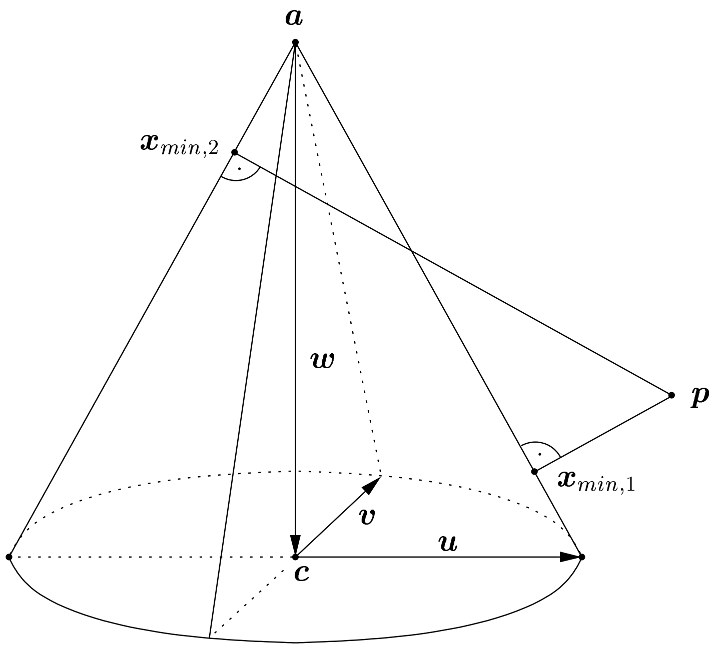

Abbildung 3.6: Abstand Punkt – Kegel

Falls der Punkt $\boldsymbol{p}$ auf der Achse des Kegels liegt, also $\boldsymbol{n} \times (\boldsymbol{p} - \boldsymbol{a}) = \boldsymbol{0}$ gilt, berechnen wir zunächst Hilfspunkte $\boldsymbol{x}'_{min}$:

$$\boldsymbol{x}'_{min} = \begin{cases} \boldsymbol{a} + \boldsymbol{w} \pm \boldsymbol{u} \ , & \text{falls } |\boldsymbol{u}| < |\boldsymbol{v}| \\ \boldsymbol{a} + \boldsymbol{w} \pm \boldsymbol{v} \ , & \text{falls } |\boldsymbol{u}| > |\boldsymbol{v}| \ . \end{cases}$$

Für jeden der beiden Hilfspunkte $\boldsymbol{x}'_{min}$ bilden wir dann die Gerade durch $\boldsymbol{x}'_{min}$ und die Kegelspitze $\boldsymbol{a}$ und lösen das Problem *Punkt – Gerade*.
Falls es sich jedoch um einen Kegel mit kreisförmigem Querschnitt handelt, also $|u| = |v|$ gilt, gibt es unendlich viele lokale Minima. Wir können ein solches Minimum $\boldsymbol{x}_{min,0}$ bestimmen, indem wir das Problem *Punkt – Gerade* für die Gerade durch den Punkt $\boldsymbol{a} + \boldsymbol{w} + \boldsymbol{u}$ und die Kegelspitze $\boldsymbol{a}$ lösen. Alle anderen Punkte ergeben sich durch Rotation von $\boldsymbol{x}_{min,0}$ um die Achse $\boldsymbol{n}$.

Sei also nun o. B. d. A. $\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p} - \boldsymbol{a}) \neq 0$ (ggfs. $\boldsymbol{u}$ durch $\boldsymbol{v}$ und $\boldsymbol{v}$ durch $-\boldsymbol{u}$ ersetzen). Um die Extrema der Funktion (3.2) zu bestimmen, betrachten wir die Ableitungen nach φ und λ:

$$\begin{aligned} \frac{\partial}{\partial\varphi} g_{\boldsymbol{p}}(\boldsymbol{x}) &= -2\,\lambda\,(\boldsymbol{p} - \boldsymbol{a} - \lambda\,(\cos\varphi\,\boldsymbol{u} + \sin\varphi\,\boldsymbol{v} + \boldsymbol{w}))\,(-\sin\varphi\,\boldsymbol{u} + \cos\varphi\,\boldsymbol{v}) \\ \frac{\partial}{\partial\lambda} g_{\boldsymbol{p}}(\boldsymbol{x}) &= -2\,(\boldsymbol{p} - \boldsymbol{a} - \lambda\,(\cos\varphi\,\boldsymbol{u} + \sin\varphi\,\boldsymbol{v} + \boldsymbol{w}))\,(\cos\varphi\,\boldsymbol{u} + \sin\varphi\,\boldsymbol{v} + \boldsymbol{w})\,. \end{aligned}$$

Die Substitution $\cos\varphi = \frac{1-t^2}{1+t^2}$ und $\sin\varphi = \frac{2t}{1+t^2}$ (mit $t = \tan\frac{\varphi}{2}$) liefert zwei bivariate Polynome in t und λ. Durch Elimination von λ erhalten wir ein

Polynom vierten Grades $f(t) = \sum_{i=0}^{4} a_i t^i$ mit den Koeffizienten

$$\begin{aligned}
a_4 &= \boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{a})\,(\boldsymbol{u}^2+\boldsymbol{w}^2)\\
a_3 &= 2\,\boldsymbol{u}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{a})\,(\boldsymbol{v}^2+\boldsymbol{w}^2)+2\,\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{a})\,(\boldsymbol{u}^2-\boldsymbol{v}^2)\\
a_2 &= 0\\
a_1 &= 2\,\boldsymbol{u}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{a})\,(\boldsymbol{v}^2+\boldsymbol{w}^2)+2\,\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{d})\,(\boldsymbol{v}^2-\boldsymbol{w}^2)\\
a_0 &= -\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{a})\,(\boldsymbol{u}^2+\boldsymbol{w}^2)\,.
\end{aligned}$$

Die Nullstellen dieses Polynoms beschreiben den Parameter φ der lokalen Extrema der Funktion (3.2). Aufgrund der Voraussetzung $\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{a}) \neq 0$ gibt es keine lokalen Extrema mit $\varphi = \pi$, d. h. durch die Substitution sind keine lokalen Extrema verloren gegangen.

Um bei gegebenem φ den Wert für λ zu bestimmen, können wir eines der beiden bivariaten Polynome nach λ auflösen. Dabei erhalten wir einen Bruchterm, der jedoch mehrere (hebbare) Singularitäten hat und für die numerische Berechnung daher ungeeignet ist. Als wesentlich genauer hat sich folgende Vorgehensweise erwiesen:
Zu einem gegebenen Wert φ wählen wir einen beliebigen Wert für λ, zum Beispiel $\lambda = 1$ und bestimmen so einen Hilfspunkt $\boldsymbol{x}'_{min}$. Anschließend lösen wir das Problem *Punkt – Gerade* für die Gerade durch $\boldsymbol{x}'_{min}$ und die Kegelspitze $\boldsymbol{a}$.

Je nach Lage des Punktes $\boldsymbol{p}$ stellt zusätzlich die Kegelspitze $\boldsymbol{a}$ ein lokales Extremum der Funktion (3.2) dar.

Abstand Punkt – Torus

Der Torus sei durch den Mittelpunkt $\boldsymbol{c}$, die Normale $\boldsymbol{n}$ mit $|\boldsymbol{n}| = 1$ und die beiden Radien R und r mit $R > r$ gegeben.

Falls der Punkt $\boldsymbol{p}$ auf der Achse des Torus liegt, also $\boldsymbol{n} \times (\boldsymbol{p}-\boldsymbol{c}) = \boldsymbol{0}$ gilt, dann gibt es unendlich viele lokale Extrema. Wir bestimmen jeweils ein lokales Minimum $\boldsymbol{x}_{min,0}$ und ein lokales Maximum $\boldsymbol{x}_{max,0}$ wie folgt:
Wir schneiden den Torus mit einer beliebigen Ebene, die den Vektor $\boldsymbol{n}$ enthält und erhalten zwei Kreise. Anschließend lösen wir für einen der beiden Kreise das Problem *Punkt – Kreis* und erhalten $\boldsymbol{x}_{min,0}$ und $\boldsymbol{x}_{max,0}$. Alle anderen Punkte $\boldsymbol{x}_{min}$ und $\boldsymbol{x}_{max}$ ergeben sich durch Rotation von $\boldsymbol{x}_{min,0}$ und $\boldsymbol{x}_{max,0}$ um die Achse $\boldsymbol{n}$.

Falls der Punkt $\boldsymbol{p}$ auf der Leitlinie des Torus liegt, also $\boldsymbol{n}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c}) = 0$ und $|\,\boldsymbol{p}-\boldsymbol{c}\,| = R$ gilt, dann erhalten wir alle lokalen Extrema der Funktion (3.2) wie folgt:
Wir schneiden den Torus mit der Ebene durch die Punkte $\boldsymbol{c}$, $\boldsymbol{c}+\boldsymbol{n}$ und $\boldsymbol{p}$ und erhalten zwei Kreise. Der Kreis mit Mittelpunkt $\boldsymbol{p}$ stellt die Minima dar. Das globale Maximum ist durch

$$\boldsymbol{x}_{max} = \boldsymbol{c} + \frac{R+r}{R}\,(\boldsymbol{c}-\boldsymbol{p})$$

gegeben.

In allen anderen Fällen berechnen sich die beiden Extrema der Funktion (3.2) durch

$$\begin{aligned}\boldsymbol{x}'_{\substack{min\\max}} &= \boldsymbol{c} \pm R\,\frac{\boldsymbol{p}-\boldsymbol{c}-\boldsymbol{n}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c})\boldsymbol{n}}{|\,\boldsymbol{p}-\boldsymbol{c}-\boldsymbol{n}^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{c})\boldsymbol{n}\,|}\\ \boldsymbol{x}_{\substack{min\\max}} &= \boldsymbol{x}'_{\substack{min\\max}} \pm r\,\frac{\boldsymbol{p}-\boldsymbol{x}'_{\substack{min\\max}}}{|\,\boldsymbol{p}-\boldsymbol{x}'_{\substack{min\\max}}\,|}\,.\end{aligned}$$

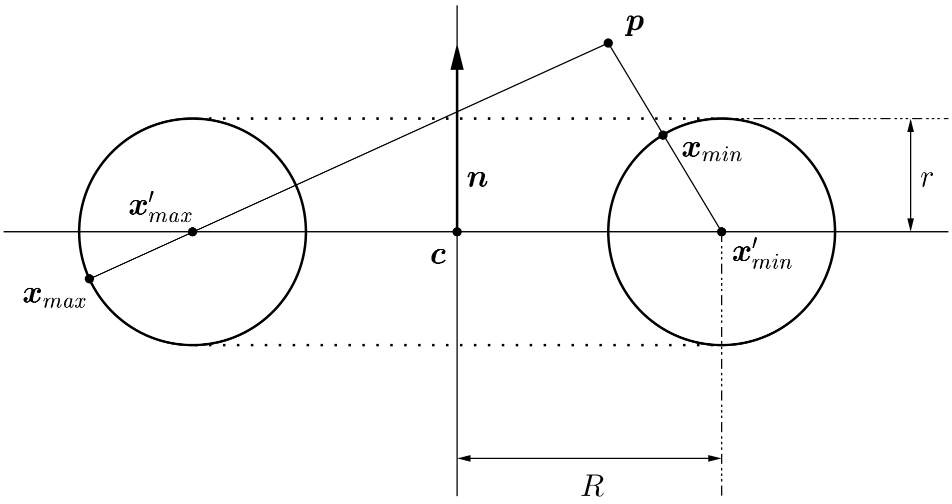

Abbildung 3.7: Abstand Punkt – Torus (Querschnitt)

Abstand Punkt – NURBS-Fläche

Der besseren Lesbarkeit wegen schreiben wir im Folgenden kurz $\boldsymbol{S}_N$ für $\boldsymbol{S}_N(u,v)$, ebenso $\boldsymbol{S}_Z$ für $\boldsymbol{S}_Z(u,v)$. Um die lokalen Extrema der Funktion (3.2) zu bestimmen, betrachten wir die partiellen Ableitungen nach u und v:

$$\begin{aligned}&\begin{cases}\frac{\partial}{\partial u}(\boldsymbol{p}-\boldsymbol{S}(u,v))^2 = 0\\ \frac{\partial}{\partial v}(\boldsymbol{p}-\boldsymbol{S}(u,v))^2 = 0\end{cases}\\ \Leftrightarrow\ &\begin{cases}\frac{\partial}{\partial u}\boldsymbol{S}(u,v)^{\mathrm{T}}\ (\boldsymbol{p}-\boldsymbol{S}(u,v)) = 0\\ \frac{\partial}{\partial v}\boldsymbol{S}(u,v)^{\mathrm{T}}\ (\boldsymbol{p}-\boldsymbol{S}(u,v)) = 0\end{cases}\\ \Leftrightarrow\ &\begin{cases}\left(\boldsymbol{S}_N\,\frac{\partial}{\partial u}\,\boldsymbol{S}_Z - \frac{\partial}{\partial u}\,\boldsymbol{S}_N\,\boldsymbol{S}_Z\right)^{\mathrm{T}}\ \left(\boldsymbol{p}\,\boldsymbol{S}_N - \boldsymbol{S}_Z\right) = 0\\ \left(\boldsymbol{S}_N\,\frac{\partial}{\partial v}\,\boldsymbol{S}_Z - \frac{\partial}{\partial v}\,\boldsymbol{S}_N\,\boldsymbol{S}_Z\right)^{\mathrm{T}}\ \left(\boldsymbol{p}\,\boldsymbol{S}_N - \boldsymbol{S}_Z\right) = 0\end{cases}\end{aligned}$$

Die lokalen Extrema sind damit durch die Nullstellen der Funktion

$$\boldsymbol{f} : [0,1]^2 \to \mathbb{R}^2,\ \boldsymbol{f}(u,v) = \begin{pmatrix}\left(\boldsymbol{S}_N\,\frac{\partial}{\partial u}\,\boldsymbol{S}_Z - \frac{\partial}{\partial u}\,\boldsymbol{S}_N\,\boldsymbol{S}_Z\right)^{\mathrm{T}}\ \left(\boldsymbol{p}\,\boldsymbol{S}_N - \boldsymbol{S}_Z\right)\\ \left(\boldsymbol{S}_N\,\frac{\partial}{\partial v}\,\boldsymbol{S}_Z - \frac{\partial}{\partial v}\,\boldsymbol{S}_N\,\boldsymbol{S}_Z\right)^{\mathrm{T}}\ \left(\boldsymbol{p}\,\boldsymbol{S}_N - \boldsymbol{S}_Z\right)\end{pmatrix}$$

gegeben. Beide Komponenten von $\boldsymbol{f}$ sind stückweise polynomiale Funktionen. Bezeichnen u und v den Grad der NURBS-Fläche $\boldsymbol{S}$, so haben die Potenzen von u und v in $\boldsymbol{f}$ höchstens den Grad $3p$ bzw. $3q$. Falls die Gewichte $w_{i,j}$ aller Kontrollpunkte gleich sind, dann ist $\boldsymbol{S}_N(u,v)$ eine konstante Funktion. Die Potenzen von u und v in $\boldsymbol{f}$ haben in diesem Fall höchstens den Grad $2p$ bzw. $2q$.

3.1.2 Enthaltenseinstests

Enthaltenseinstest Punkt – Liniensegment

Unter der Voraussetzung, dass der Punkt $\boldsymbol{p}$ auf der Geraden durch die Punkte $\boldsymbol{a}$ und $\boldsymbol{b}$ liegt, gilt:

$$\begin{aligned} & \boldsymbol{p} \text{ liegt auf dem Liniensegment } \boldsymbol{ab} \\ \Leftrightarrow \quad & (\boldsymbol{b}-\boldsymbol{a})^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{a}) \geq 0 \quad \wedge \quad (\boldsymbol{b}-\boldsymbol{a})^{\mathrm{T}}(\boldsymbol{p}-\boldsymbol{b}) \leq 0 . \end{aligned}$$

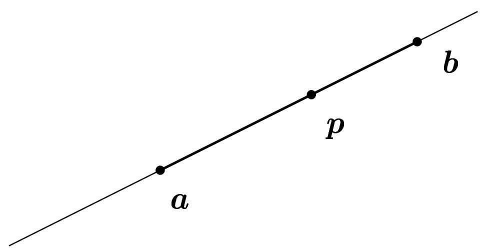

Abbildung 3.8: Enthaltenseinstest Punkt – Liniensegment

Enthaltenseinstest Punkt – Ellipsensegment

Unter der Voraussetzung, dass der Punkt $\boldsymbol{p}$ auf der zugrunde liegenden Ellipse liegt, gilt:

$$\begin{aligned} & \boldsymbol{p} \text{ liegt auf dem Ellipsensegment } \boldsymbol{ab} \\ \Leftrightarrow \quad & \varphi_a \leq \varphi_p \leq \varphi_b \quad \vee \quad \varphi_b \leq \varphi_a \leq \varphi_p \quad \vee \quad \varphi_p \leq \varphi_b \leq \varphi_a \\ & \text{mit } \varphi_x = \arctan\left(\frac{\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{x}-\boldsymbol{c})}{\boldsymbol{v}^2}, \frac{\boldsymbol{u}^{\mathrm{T}}(\boldsymbol{x}-\boldsymbol{c})}{\boldsymbol{u}^2}\right) . \end{aligned}$$

Enthaltenseinstest Punkt – NURBS-Kurvensegment

Sei also ein Punkt $\boldsymbol{p}$ auf einer NURBS-Kurve $\boldsymbol{C}$ gegeben. Das Kurvensegment sei durch die Punkte $\boldsymbol{a}$ und $\boldsymbol{b}$ spezifiziert. Wir berechnen zunächst die Parameter u_a, u_b und u_p mit

$$\boldsymbol{a} = \boldsymbol{C}(u_a)\ , \quad \boldsymbol{b} = \boldsymbol{C}(u_b) \quad \text{und} \quad \boldsymbol{p} = \boldsymbol{C}(u_p) .$$

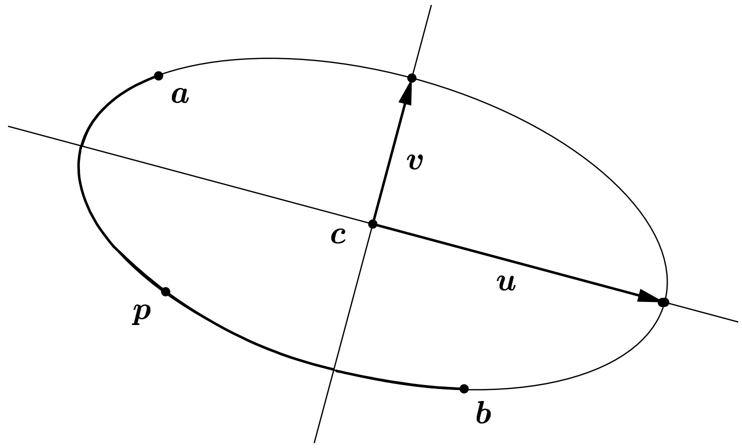

Abbildung 3.9: Enthaltenseinstest Punkt – Ellipsensegment

Eine Möglichkeit zum Berechnen dieser Werte wurde in Kapitel 2.5 gezeigt. Es gilt nun:

$$
\begin{aligned}
& \boldsymbol{p} \text{ liegt auf dem Kurvensegment } \boldsymbol{ab} \\
\Leftrightarrow \quad & u_a \leq u_p \leq u_b .
\end{aligned}
$$

Falls es sich jedoch um eine geschlossene Kurve handelt (d.h. $\boldsymbol{C}(0) = \boldsymbol{C}(1)$), gilt eine Bedingung ähnlich wie im Fall der Ellipse:

$$
\begin{aligned}
& \boldsymbol{p} \text{ liegt auf dem Kurvensegment } \boldsymbol{ab} \\
\Leftrightarrow \quad & u_a \leq u_p \leq u_b \quad \vee \quad u_b \leq u_a \leq u_p \quad \vee \quad u_p \leq u_b \leq u_a .
\end{aligned}
$$

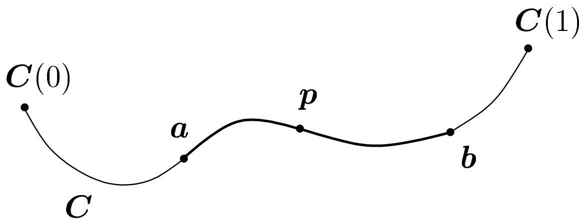

Abbildung 3.10: Enthaltenseinstest Punkt – NURBS-Kurvensegment

Enthaltenseinstest Punkt – Fläche

Um festzustellen, ob ein gegebener Punkt $\boldsymbol{p}$ auf einem Flächenstück f liegt, das auf einer NURBS-Fläche $\boldsymbol{S}$ eingebettet ist, gehen wir wie folgt vor:
Wir verfolgen, ausgehend vom Punkt $\boldsymbol{p}$, eine stetige Kurve g auf der Fläche $\boldsymbol{S}$. Die Kurve soll dabei die Begrenzungskanten von f mindestens einmal schneiden.

Der erste Schnittpunkt der Kurve g mit den Begrenzungskanten von f wird mit $\boldsymbol{s}$ bezeichnet, die zugehörige Begrenzungskante mit e.

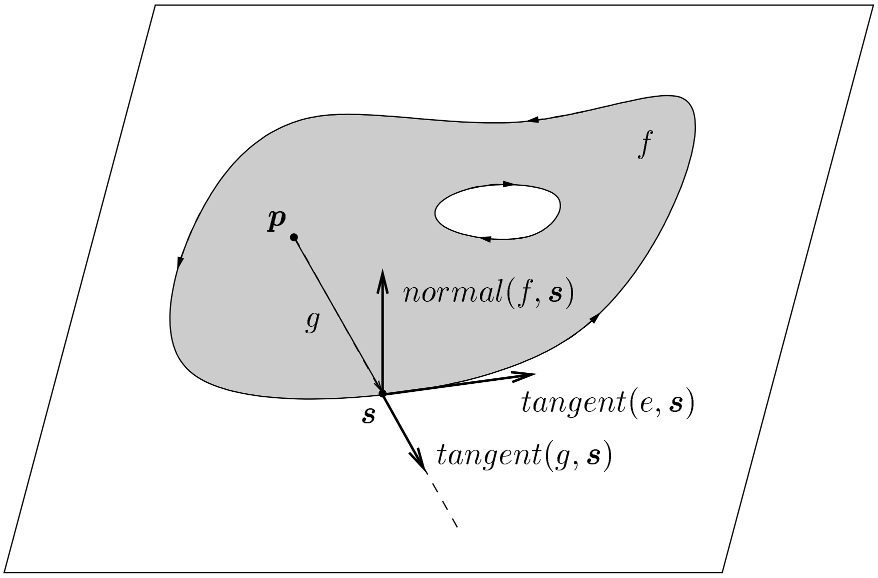

Abbildung 3.11: Enthaltenseinstest Punkt – Fläche

Unter der Voraussetzung, dass die Begrenzungskanten von f bzgl. der Flächennormalen wie in Abbildung 3.11 gezeigt orientiert sind, gilt:

$$\det(\text{tangent}(g, \boldsymbol{s}), \text{tangent}(e, \boldsymbol{s}), \text{normal}(f, \boldsymbol{s})) > 0 \quad \Rightarrow \quad \boldsymbol{p} \in f$$
$$\det(\text{tangent}(g, \boldsymbol{s}), \text{tangent}(e, \boldsymbol{s}), \text{normal}(f, \boldsymbol{s})) < 0 \quad \Rightarrow \quad \boldsymbol{p} \notin f\,.$$

Dabei bezeichnen tangent$(g, \boldsymbol{s})$ und tangent$(e, \boldsymbol{s})$ die Tangenten an die Kurve g bzw. die Begrenzungskante e im Punkt $\boldsymbol{s}$ und normal$(f, \boldsymbol{s})$ die Flächennormale im Punkt $\boldsymbol{s}$.
Im Falle $\det(\text{tangent}(g, \boldsymbol{s}), \text{tangent}(e, \boldsymbol{s}), \text{normal}(f, \boldsymbol{s})) = 0$ ist keine Aussage möglich. In einem solchen Fall müssen wir die Kurve g anders wählen (s.u.).

Bei der Berechnung des Schnittpunktes $\boldsymbol{s}$ der Kurve g und einer Begrenzungskante e treten die Schnittprobleme *Gerade – Gerade*, *Gerade – Ellipse* und *Ellipse – Ellipse* auf. Eine Lösung dieser Schnittprobleme findet sich in [Beh99]. Zusätzlich müssen wir in der Lage sein, den Schnittpunkt zweier NURBS-Kurven zu berechnen.

Schnitt zweier NURBS-Kurven

Seien $\boldsymbol{C}_1(u_1)$ und $\boldsymbol{C}_2(u_2)$ zwei NURBS-Kurven vom Grad p_1 und p_2. Die Bedingung für einen Schnittpunkt zwischen beiden Kurven lautet

$$\begin{aligned} & \boldsymbol{C}_1(u_1) = \boldsymbol{C}_2(u_2) \\ \Leftrightarrow\quad & \boldsymbol{C}_{1,Z}(u_1) \cdot \boldsymbol{C}_{2,N}(u_2) = \boldsymbol{C}_{2,Z}(u_2) \cdot \boldsymbol{C}_{1,N}(u_1) \\ \Leftrightarrow\quad & \boldsymbol{C}_{1,Z}(u_1) \cdot \boldsymbol{C}_{2,N}(u_2) - \boldsymbol{C}_{2,Z}(u_2) \cdot \boldsymbol{C}_{1,N}(u_1) = 0\,. \end{aligned}$$

Die Schnittpunkte zwischen den beiden Kurven $\boldsymbol{C}_1$ und $\boldsymbol{C}_2$ werden also durch die Nullstellen der Funktion

$$\boldsymbol{f} : [0,1]^2 \to \mathbb{R}^3, \ \boldsymbol{f}(u_1, u_2) = \boldsymbol{C}_{1,Z}(u_1) \cdot \boldsymbol{C}_{2,N}(u_2) - \boldsymbol{C}_{2,Z}(u_2) \cdot \boldsymbol{C}_{1,N}(u_1)$$

beschrieben. Die Komponenten dieser Funktion sind stückweise polynomiale Funktionen, die auftretenden Potenzen von u_1 und u_2 sind höchstens vom Grad p_1 bzw. p_2.

Berechnung der Tangenten

Um die Frage $\boldsymbol{p} \in f$ oder $\boldsymbol{p} \notin f$ zu entscheiden, müssen wir in der Lage sein, die Tangenten im Punkt $\boldsymbol{s}$ zu berechnen. Für eine Gerade ist dies trivial. Bei einer Ellipse mit Mittelpunkt $\boldsymbol{c}$ und den Halbachsen $\boldsymbol{u}$ und $\boldsymbol{v}$ ist die Tangente $\boldsymbol{t}$ im Punkt $\boldsymbol{s}$ durch

$$\boldsymbol{t} = -\frac{\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{s}-\boldsymbol{c})}{\boldsymbol{v}^2}\,\boldsymbol{u} + \frac{\boldsymbol{u}^{\mathrm{T}}(\boldsymbol{s}-\boldsymbol{c})}{\boldsymbol{u}^2}\,\boldsymbol{v}$$

gegeben. Handelt es sich um NURBS-Kurve, so berechnen wir zunächst den Parameterwert u mit $\boldsymbol{s} = \boldsymbol{C}(u)$. Die Richtung der Tangente $\boldsymbol{t}$ ist dann durch $\boldsymbol{C}'(u)$ gegeben (vgl. Abbildung 3.12).

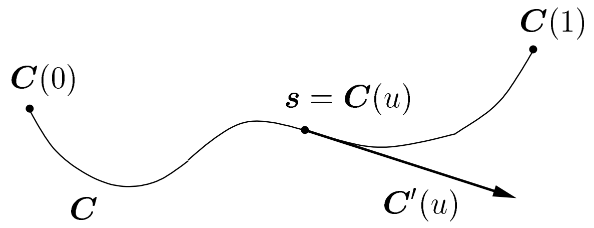

Abbildung 3.12: Tangente an eine NURBS-Kurve

Wir wollen nun auf die verschiedenen Flächentypen genauer eingehen und dabei jeweils eine einfache Möglichkeit für die Wahl der Kurve g angeben. Für den Fall, dass f auf einer Ebene, einem Zylinder oder einem Kegel eingebettet ist, wählen wir g als einen vom Punkt $\boldsymbol{p}$ ausgehenden Strahl. Die Voraussetzung, dass g die Begrenzungskanten von f schneiden soll, kann hier entfallen. Wenn es keinen Schnittpunkt gibt, liegt der Punkt $\boldsymbol{p}$ nicht auf dem Flächenstück f.

Enthaltenseinstest Punkt – Ebene

Sei $\boldsymbol{n}^{\mathrm{T}}(\boldsymbol{x} - \boldsymbol{r}) = 0$ die Ebene, auf der f eingebettet ist. Als Strahl wählen wir

$$\boldsymbol{x} = \boldsymbol{p} + \lambda\,\boldsymbol{d} \text{ mit } \lambda \geq 0 \text{ und } \boldsymbol{d}^{\mathrm{T}}\boldsymbol{n} = 0.$$

Enthaltenseinstest Punkt – Kugel

Wir wählen zunächst einen Punkt $\boldsymbol{q}$ mit $\boldsymbol{q} \neq \boldsymbol{p}$ und $\boldsymbol{q} \neq \boldsymbol{c} + (\boldsymbol{c} - \boldsymbol{p})$ auf einer Begrenzungskante von f. Als Kurve g wählen wir eine Kreisbahn durch die Punkte $\boldsymbol{p}$ und $\boldsymbol{q}$ auf der Oberfläche der Kugel. Eine Möglichkeit ist zum Beispiel die Kreisbahn in der Ebene, die durch die Punkte $\boldsymbol{p}$, $\boldsymbol{q}$ und $\boldsymbol{c}$ aufgespannt wird:

$$
\begin{aligned}
\boldsymbol{x} &= \boldsymbol{c} + \cos\varphi\, \boldsymbol{u} + \sin\varphi\, \boldsymbol{v} \\
\boldsymbol{u} &= \boldsymbol{p} - \boldsymbol{c} \\
\boldsymbol{v} &= \boldsymbol{n} \times \boldsymbol{u} \\
\boldsymbol{n} &= \frac{\boldsymbol{c} \times \boldsymbol{p} + \boldsymbol{p} \times \boldsymbol{q} + \boldsymbol{q} \times \boldsymbol{c}}{|\,\boldsymbol{c} \times \boldsymbol{p} + \boldsymbol{p} \times \boldsymbol{q} + \boldsymbol{q} \times \boldsymbol{c}\,|}.
\end{aligned}
$$

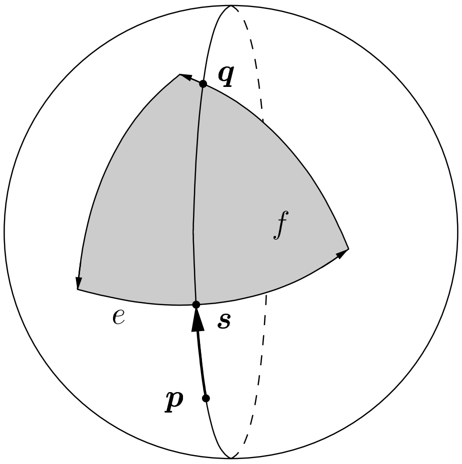

Abbildung 3.13: Enthaltenseinstest Punkt – Kugel

Enthaltenseinstest Punkt – Zylinder

Das Flächenstück f sei auf einem Zylinder $\boldsymbol{x} = \boldsymbol{c} + \cos\varphi\, \boldsymbol{u} + \sin\varphi\, \boldsymbol{v} + \lambda\, \boldsymbol{n}$ mit $\boldsymbol{u} \perp \boldsymbol{v}$ und $\boldsymbol{n} = \frac{\boldsymbol{u} \times \boldsymbol{v}}{|\boldsymbol{u} \times \boldsymbol{v}|}$ eingebettet. Als Strahl wählen wir

$$\boldsymbol{x} = \boldsymbol{p} + \lambda \boldsymbol{n} \text{ mit } \lambda \geq 0.$$

Enthaltenseinstest Punkt – Kegel

Das Flächenstück f sei auf dem Kegel $\boldsymbol{x} = \boldsymbol{a} + \mu(\cos\varphi \boldsymbol{u} + \sin\varphi \boldsymbol{v} + \boldsymbol{w})$ mit $\mu \geq 0$ eingebettet. Als Strahl wählen wir

$$\boldsymbol{x} = \boldsymbol{a} + \lambda\,(\boldsymbol{p} - \boldsymbol{a}) \text{ mit } \lambda \geq 0.$$

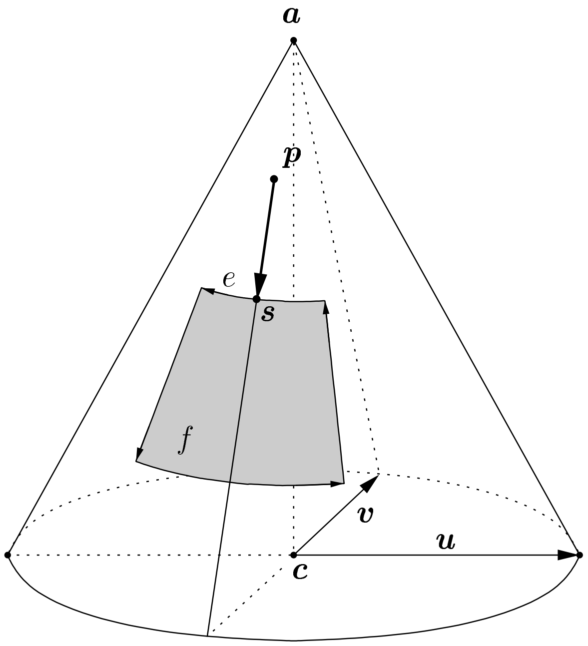

Abbildung 3.14: Enthaltenseinstest Punkt – Kegel

Enthaltenseinstest Punkt – Torus

Das Flächenstück f sei auf einem Torus mit Mittelpunkt $\boldsymbol{c}$, Normalen $\boldsymbol{n}$ und den beiden Radien R und r mit $R > r$ eingebettet. Als Verfolgungskurve g wählen wir zunächst die Kreisbahn

$$\begin{aligned}
\boldsymbol{x} &= \boldsymbol{c}_1 + \cos\varphi\, \boldsymbol{u}_1 + \sin\varphi\, \boldsymbol{v}_1 \\
\boldsymbol{c}_1 &= \boldsymbol{c} + R\, \frac{\boldsymbol{p} - \boldsymbol{c} - \boldsymbol{n}^{\mathrm{T}}(\boldsymbol{p} - \boldsymbol{c})\boldsymbol{n}}{|\, \boldsymbol{p} - \boldsymbol{c} - \boldsymbol{n}^{\mathrm{T}}(\boldsymbol{p} - \boldsymbol{c})\boldsymbol{n}\, |} \\
\boldsymbol{u}_1 &= \boldsymbol{p} - \boldsymbol{c}_1 \\
\boldsymbol{v}_1 &= \frac{\boldsymbol{n} \times (\boldsymbol{p} - \boldsymbol{c})}{|\, \boldsymbol{n} \times (\boldsymbol{p} - \boldsymbol{c})\, |} \times \boldsymbol{u}_1 .
\end{aligned}$$

Treffen wir keine der Begrenzungskanten von f, so wissen wir, dass alle Punkte der Kreisbahn entweder auf f liegen oder nicht. Wir wählen nun einen Punkt $\boldsymbol{q}$ auf einer Begrenzungskante von f und betrachten die Kreisbahn

$$\begin{aligned}
\boldsymbol{x} &= \boldsymbol{c}_2 + \cos\varphi\, \boldsymbol{u}_2 + \sin\varphi\, \boldsymbol{v}_2 \\
\boldsymbol{c}_2 &= \boldsymbol{c} + \boldsymbol{n}^{\mathrm{T}}(\boldsymbol{q} - \boldsymbol{c})\boldsymbol{n} \\
\boldsymbol{u}_2 &= \frac{|\, \boldsymbol{q} - \boldsymbol{c}_2\, |}{R}\, (\boldsymbol{c}_1 - \boldsymbol{c}) \\
\boldsymbol{v}_1 &= \boldsymbol{n} \times \boldsymbol{u}_2 .
\end{aligned}$$

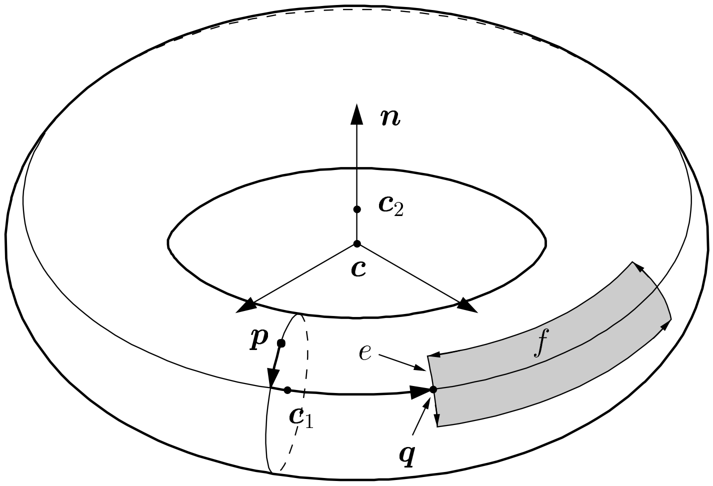

Abbildung 3.15: Enthaltenseinstest Punkt – Torus

Enthaltenseinstest Punkt – NURBS-Fläche

Sei also eine NURBS-Fläche $\boldsymbol{S}$ und ein Punkt $\boldsymbol{p}$ auf dieser Fläche gegeben. Außerdem sei ein Flächenstück f auf der Fläche $\boldsymbol{S}$ gegeben. Das Flächenstück f soll durch seine Begrenzungskanten spezifiziert sein.

Wir wählen zunächst einen Punkt $\boldsymbol{q}$ auf einer Begrenzungskante von f und berechnen die Parameterwerte $(u_{\boldsymbol{p}}, v_{\boldsymbol{p}})$ und $(u_{\boldsymbol{q}}, v_{\boldsymbol{q}})$ mit

$$\boldsymbol{S}(u_{\boldsymbol{p}}, v_{\boldsymbol{p}}) = \boldsymbol{p} \qquad \text{und} \qquad \boldsymbol{S}(u_{\boldsymbol{q}}, v_{\boldsymbol{q}}) = \boldsymbol{q}\,.$$

Abbildung 3.16 zeigt die Situation im Parameterraum der Fläche $\boldsymbol{S}$. Die grau gefärbte Punktmenge umfasst alle Parameterwerte, die zu Punkten des Flächenstücks f gehören. (Eine Berechnung dieser Parameterwerte ist *nicht* notwendig.)

Der Kantenzug $(u_{\boldsymbol{p}}, v_{\boldsymbol{p}}) - (u_{\boldsymbol{p}}, v_{\boldsymbol{q}}) - (u_{\boldsymbol{q}}, v_{\boldsymbol{q}})$ beschreibt nun die Kurve g im Parameterraum der Fläche $\boldsymbol{S}$. Beide Segmente des Kantenzuges definieren eine auf der Fläche $\boldsymbol{S}$ eingebettete NURBS-Kurve vom Grad p bzw. q. Eine mathematische Darstellung dieser Kurven lässt sich leicht berechnen. Der erste Teil der Kurve g liegt auf der NURBS-Kurve $\boldsymbol{C}_{u_{\boldsymbol{p}}}$:

$$\boldsymbol{C}_{u_{\boldsymbol{p}}}(v) = \boldsymbol{S}(u_{\boldsymbol{p}}, v) = \frac{\sum_{j=0}^{m-1} N_{j,q}(v)\, w_j\, \boldsymbol{P}_j}{\sum_{j=0}^{m-1} N_{j,q}(v)\, w_j}$$

$$\text{mit}\ \ w_j = \sum_{i=0}^{n-1} N_{i,p}(u_{\boldsymbol{p}})\, w_{i,j} \ \ \text{und}\ \ \boldsymbol{P}_j = \frac{1}{w_j} \sum_{i=0}^{n-1} N_{i,p}(u_{\boldsymbol{p}})\, w_{i,j}\, \boldsymbol{P}_{i,j}\,.$$

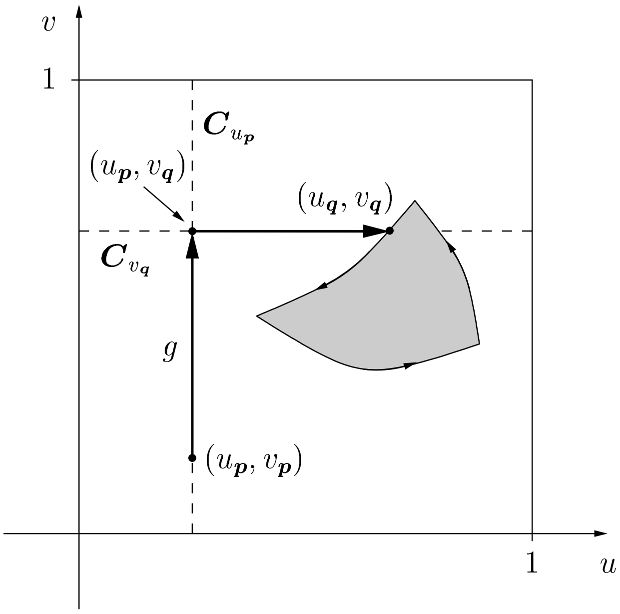

Abbildung 3.16: Enthaltenseinstest Punkt – NURBS-Fläche (Parameterraum)

Vollkommen analog liegt der zweite Teil auf der NURBS-Kurve $\boldsymbol{C}_{v_{\boldsymbol{q}}}$:

$$\boldsymbol{C}_{v_{\boldsymbol{q}}}(u) = \boldsymbol{S}(u, v_{\boldsymbol{q}}) = \frac{\sum_{i=0}^{n-1} N_{i,p}(u)\, w_i\, \boldsymbol{P}_i}{\sum_{i=0}^{n-1} N_{i,p}(u)\, w_i}$$

$$\text{mit} \quad w_i = \sum_{j=0}^{m-1} N_{j,q}(v_{\boldsymbol{q}})\, w_{i,j} \quad \text{und} \quad \boldsymbol{P}_i = \frac{1}{w_i} \sum_{j=0}^{m-1} N_{j,q}(v_{\boldsymbol{q}})\, w_{i,j}\, \boldsymbol{P}_{i,j}\,.$$

Falls der erste Teil der Kurve g auf der NURBS-Kurve $\boldsymbol{C}_{u_{\boldsymbol{p}}}$ bereits die Begrenzungskanten von f schneidet, kann die Betrachtung des zweiten Teils auf der Kurve $\boldsymbol{C}_{v_{\boldsymbol{q}}}$ entfallen. Wir bemerken außerdem, dass dieser Enthaltenseinstest auch im Falle einer zyklischen NURBS-Fläche funktioniert.

Probleme

Tritt der Fall $\det(\text{tangent}(g, \boldsymbol{s}), \text{tangent}(e, \boldsymbol{s}), \text{normal}(f, \boldsymbol{s})) = 0$ ein, können wir keine Aussage machen, ob $\boldsymbol{p} \in f$ oder $\boldsymbol{p} \notin f$ gilt. In einem solchen Fall ändern wir die Wahl der Kurve g, um ein gesichertes Ergebnis zu erhalten.
Bei dem Problem *Punkt – Ebene* ist lediglich $\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n} = 0$ gefordert, d.h. wir haben genügend Möglichkeiten für die Wahl von g.
Bei den Problemen *Punkt – Kugel* und *Punkt – Torus* können wir den Punkt $\boldsymbol{q}$ beliebig auf den Begrenzungskanten von f wählen, außerdem muss bei der Kugel der Mittelpunkt der Kreisbahn nicht mit dem Kugelmittelpunkt $\boldsymbol{c}$ übereinstimmen.
Bei einem Zylinder oder einem Kegel können wir auch eine Ellipse als Kurve g wählen. Erhalten wir keinen Schnittpunkt mit den Kanten von f, so wählen

wir — ähnlich wie im Fall des Torus oder der NURBS-Fläche — einen Punkt $\boldsymbol{q}$ auf einer Begrenzungskante von f. Dieser Punkt $\boldsymbol{q}$ kann von einem geeigneten Punkt der Ellipse durch ein Geradensegment erreicht werden.
Im Fall der allgemeinen NURBS-Fläche kann man einen anderen Punkt $\boldsymbol{q}$ auf den Begrenzungskanten wählen. Außerdem kann man den Kantenzug $(u_{\boldsymbol{p}}, v_{\boldsymbol{p}}) - (u_{\boldsymbol{p}}, v_{\boldsymbol{q}}) - (u_{\boldsymbol{q}}, v_{\boldsymbol{q}})$ durch $(u_{\boldsymbol{p}}, v_{\boldsymbol{p}}) - (u_{\boldsymbol{q}}, v_{\boldsymbol{p}}) - (u_{\boldsymbol{q}}, v_{\boldsymbol{q}})$ ersetzen.

3.1.3 Berechnung des maximalen Abstands

Gegeben sei ein Körper $\mathcal{K}$, repräsentiert durch die Flächenstücke $\mathcal{K}_f$, die Begrenzungskanten $\mathcal{K}_e$ und die Eckpunkte $\mathcal{K}_p$. Ferner sei $\boldsymbol{p} \in \mathbb{R}^3$ ein Punkt.
Mit den Methoden aus den beiden vorherigen Kapiteln können wir jetzt den maximalen Abstand $d_{\mathcal{K}}(\boldsymbol{p})$ eines Punktes $\boldsymbol{p}$ von dem Körper $\mathcal{K}$ berechnen:

$$\begin{aligned} d_{\mathcal{K}}(\boldsymbol{p}) &= \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p} - \boldsymbol{x} \,| \qquad (3.3) \\ &= \max \{ \max_{f \in \mathcal{K}_f} \max_{\boldsymbol{x} \in f} |\, \boldsymbol{p} - \boldsymbol{x} \,|,\ \max_{e \in \mathcal{K}_e} \max_{\boldsymbol{x} \in e} |\, \boldsymbol{p} - \boldsymbol{x} \,|,\ \max_{p \in \mathcal{K}_p} |\, \boldsymbol{p} - \boldsymbol{x} \,| \}. \end{aligned}$$

Die in Kapitel 3.1.1 vorgestellten Methoden zur Bestimmung der lokalen Maxima von Funktion (3.2) arbeiten mit *unbegrenzten* Flächen. Zusammen mit den Enthaltenseinstests aus Kapitel 3.1.2 werden also nur lokale Extrema im *Innern* eines Flächenstücks f berechnet. Aus diesem Grund ist es notwendig, die Begrenzungskanten eines Flächenstücks gesondert zu betrachten.
Aus dem gleichen Grund werden zunächst auch nur die lokalen Extrema im Innern einer Kante gefunden und es ist notwendig, die Endpunkte der Kanten, also die Eckpunkte des Körpers, zusätzlich zu untersuchen.

Wir zeigen nun noch einige für das weitere Vorgehen wichtige Eigenschaften der Funktion (3.3).

Lemma 3.3. *Der maximale Abstand $d_{\mathcal{K}}(\boldsymbol{p})$ eines Punktes $\boldsymbol{p} \in \mathbb{R}^3$ zu einem Körper $\mathcal{K}$ ist eine stetige Funktion.*

Beweis:
Sei $\boldsymbol{p}_0 \in \mathbb{R}^3$ beliebig, $\varepsilon > 0$ beliebig. Für alle $\boldsymbol{p} \in \mathbb{R}^3$ mit $|\, \boldsymbol{p} - \boldsymbol{p}_0 \,| < \varepsilon$ gilt dann:
1. Fall: $d_{\mathcal{K}}(\boldsymbol{p}) \geq d_{\mathcal{K}}(\boldsymbol{p}_0)$

$$\begin{aligned} |\, d_{\mathcal{K}}(\boldsymbol{p}) - d_{\mathcal{K}}(\boldsymbol{p}_0) \,| &= \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p} - \boldsymbol{x} \,| - \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p}_0 - \boldsymbol{x} \,| \\ &\leq \max_{\boldsymbol{x} \in \mathcal{K}} \big(|\, \boldsymbol{p} - \boldsymbol{p}_0 \,| + |\, \boldsymbol{p}_0 - \boldsymbol{x} \,| \big) - \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p}_0 - \boldsymbol{x} \,| \\ &= |\, \boldsymbol{p} - \boldsymbol{p}_0 \,| \\ &< \varepsilon \end{aligned}$$

2. Fall: $d_{\mathcal{K}}(\boldsymbol{p}) \leq d_{\mathcal{K}}(\boldsymbol{p}_0)$

$$\begin{aligned} |\, d_{\mathcal{K}}(\boldsymbol{p}) - d_{\mathcal{K}}(\boldsymbol{p}_0) \,| &= \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p}_0 - \boldsymbol{x} \,| - \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p} - \boldsymbol{x} \,| \\ &\leq \max_{\boldsymbol{x} \in \mathcal{K}} \big(|\, \boldsymbol{p}_0 - \boldsymbol{p} \,| + |\, \boldsymbol{p} - \boldsymbol{x} \,| \big) - \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p} - \boldsymbol{x} \,| \\ &= |\, \boldsymbol{p}_0 - \boldsymbol{p} \,| \\ &< \varepsilon \end{aligned}$$

□

Lemma 3.4. *Der maximale Abstand $d_{\mathcal{K}}(\boldsymbol{p})$ eines Punktes $\boldsymbol{p} \in \mathbb{R}^3$ zu einem Körper $\mathcal{K}$ ist im Allgemeinen keine differenzierbare Funktion.*

Beweis:
Sei $\mathcal{K}$ die Einheitskugel im $\mathbb{R}^3$. Dann gilt offensichtlich

$$d_{\mathcal{K}}(\boldsymbol{p}) = |\, \boldsymbol{p} \,| + 1 = \sqrt{p_1^2 + p_2^2 + p_3^2} + 1 .$$

Diese Funktion ist an der Stelle $\boldsymbol{p} = \boldsymbol{0}$ nicht differenzierbar. □

Hierbei handelt es sich nicht um einen Einzelfall. Selbst bei Polygonen gibt es Punkte, an denen die Funktion $d_{\mathcal{K}}(\boldsymbol{p})$ nicht differenzierbar ist.

Lemma 3.5. *Der maximale Abstand $d_{\mathcal{K}}(\boldsymbol{p})$ eines Punktes $\boldsymbol{p} \in \mathbb{R}^3$ zu einem Körper $\mathcal{K}$ ist eine konvexe Funktion.*

Beweis:
Seien $\boldsymbol{p}_1, \boldsymbol{p}_2 \in \mathbb{R}^3$ beliebig, $\lambda \in [0,1]$ beliebig. Dann gilt:

$$\begin{aligned}
& d_{\mathcal{K}}(\lambda\, \boldsymbol{p}_1 + (1-\lambda)\, \boldsymbol{p}_2) \\
= \; & \max_{\boldsymbol{x} \in \mathcal{K}} |\, (\lambda\, \boldsymbol{p}_1 + (1-\lambda)\, \boldsymbol{p}_2) - \boldsymbol{x} \,| \\
= \; & \max_{\boldsymbol{x} \in \mathcal{K}} |\, \lambda\, (\boldsymbol{p}_1 - \boldsymbol{x}) + (1-\lambda)\, (\boldsymbol{p}_2 - \boldsymbol{x}) \,| \\
\leq \; & \max_{\boldsymbol{x} \in \mathcal{K}} \big(|\, \lambda\, (\boldsymbol{p}_1 - \boldsymbol{x}) \,| + |\, (1-\lambda)\, (\boldsymbol{p}_2 - \boldsymbol{x}) \,| \big) \\
= \; & \max_{\boldsymbol{x} \in \mathcal{K}} \big(\lambda\; |\, \boldsymbol{p}_1 - \boldsymbol{x} \,| + (1-\lambda)\; |\, \boldsymbol{p}_2 - \boldsymbol{x} \,| \big) \\
\leq \; & \max_{\boldsymbol{x} \in \mathcal{K}} \lambda\; |\, \boldsymbol{p}_1 - \boldsymbol{x} \,| + \max_{\boldsymbol{x} \in \mathcal{K}} (1-\lambda)\; |\, \boldsymbol{p}_2 - \boldsymbol{x} \,| \\
= \; & \lambda\, \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p}_1 - \boldsymbol{x} \,| + (1-\lambda)\, \max_{\boldsymbol{x} \in \mathcal{K}} |\, \boldsymbol{p}_2 - \boldsymbol{x} \,| \\
= \; & \lambda\, d_{\mathcal{K}}(\boldsymbol{p}_1) + (1-\lambda)\, d_{\mathcal{K}}(\boldsymbol{p}_2)
\end{aligned}$$

□

3.1.4 Minimierung des Kugelvolumens

Wir wollen nun die kleinste einschließende Kugel für einen Körper $\mathcal{K}$ bestimmen. Damit die Kugel den Körper einschließt, muss der Kugelradius mindestens so groß wie der maximale Abstand vom Kugelmittelpunkt zur Oberfläche des Körpers sein. Dieser maximale Abstand kann wie im vorherigen Kapitel gezeigt berechnet werden. Unser Ziel ist es nun, den Kugelmittelpunkt $\boldsymbol{c}$ so zu wählen, dass dieser maximale Abstand minimal wird.
Gegeben sei also ein Körper $\mathcal{K}$. Gesucht ist ein Punkt $\boldsymbol{c} \in \mathbb{R}^3$, so dass $d_{\mathcal{K}}(\boldsymbol{c})$ minimal wird. Der Mittelpunkt der kleinsten einschließenden Kugel ist dann gerade der Punkt $\boldsymbol{c}$, der Radius r ist durch $d_{\mathcal{K}}(\boldsymbol{c})$ gegeben. Das Volumen dieser kleinsten einschließenden Kugel beträgt

$$V(\boldsymbol{c}) = \frac{4}{3}\pi \left(d_{\mathcal{K}}(\boldsymbol{c})\right)^3 . \tag{3.4}$$

Die zu minimierende Funktion $d_{\mathcal{K}}(\boldsymbol{c})$ ist nach Lemma 3.5 konvex. Zusammen mit der Beobachtung

$$\lim_{|\boldsymbol{c}|\to\infty} d_{\mathcal{K}}(\boldsymbol{c}) = \infty$$

folgt, dass die Funktion $d_{\mathcal{K}}(\boldsymbol{c})$ genau *ein* lokales Minimum hat. Aufgrund der Konvexität handelt es sich dabei um das globale Minimum.
Für die Wahl des Optimierungsverfahrens bedeutet dies, dass wir keine globalen Optimierungsverfahren brauchen, sondern dass lokale Optimierungsverfahren ausreichend sind. Da die Funktion nach Lemma 3.4 nicht differenzierbar ist, müssen wir uns auf solche Optimierungsverfahren beschränken, die ausschließlich Funktionsauswertungen benutzen und keinen Gebrauch von Gradienten oder höheren Ableitungen machen.

Als Optimierungsverfahren verwenden wir das *Downhill-Simplex-Verfahren* und die *Methode von Powell*. Eine ausführliche Beschreibung beider Verfahren und eine Implementierung findet sich in [PTVF94].
Beide Verfahren beginnen mit einem Startpunkt $\boldsymbol{x}_0$. In jedem Iterationsschritt wird der Punkt $\boldsymbol{x}_i$ durch einen besseren Punkt $\boldsymbol{x}_{i+1}$ ersetzt (also $d_{\mathcal{K}}(\boldsymbol{x}_{i+1}) < d_{\mathcal{K}}(\boldsymbol{x}_i)$). Ein Parameter `tol` dient als Abbruchkriterium: Das Verfahren terminiert, wenn es in einem Iterationsschritt nicht gelingt, den Funktionswert um den Faktor `tol` zu reduzieren.

3.1.5 Optimierungen

Zunächst werden zwei Methoden vorgestellt, um die Auswertung der Funktion (3.3) zu beschleunigen. Durch die richtige Reihenfolge bei der Auswertung der einzelnen Terme können unnötige Berechnungen vermieden werden.

- Das Abstandsproblem *Punkt – Fläche* ist wesentlich aufwendiger als das Problem *Punkt – Kante*. Dies ist hauptsächlich darauf zurückzuführen, dass bei den Enthaltenseinstests für Flächenstücke die Kurve g mit jeder Begrenzungskante von f geschnitten werden muss. Durch Vermeidung von unnötigen Enthaltenseinstests kann die Berechnung von Funktion (3.3) erheblich beschleunigt werden.

  Kommen wir in eine Situation, in der für einen Punkt $\boldsymbol{x}$ der Wert $|\boldsymbol{p} - \boldsymbol{x}|$ nicht größer als das größte bisher bekannte Maximum ist, so kann der Enthaltenseinstest entfallen. Unabhängig davon, ob der Punkt $\boldsymbol{x}$ auf dem zugehörigen Flächenstück liegt oder nicht, ändert sich das größte bisher bekannte Maximum nicht.

- Basierend auf der vorherigen Überlegung ist es günstig, wenn wir frühzeitig eine gute untere Schranke für den maximalen Abstand des Punktes $\boldsymbol{p}$ von dem Körper $\mathcal{K}$ haben. Diese erhalten wir, wenn wir die Teilterme von Funktion (3.3) in folgender Reihenfolge auswerten:

  Wir berechnen zunächst den maximalen Abstand zu den Eckpunkten $\mathcal{K}_p$ und den Kanten $\mathcal{K}_e$ des Körpers. Damit erhalten wir je nach Form des Objektes eine mehr oder weniger gute untere Schranke für den maximalen Abstand. Erst dann bestimmen wir den maximalen Abstand zu den

Flächenstücken $\mathcal{K}_f$. Durch diese Vorgehensweise können wir die Anzahl der notwendigen Enthaltenseinstests und die Laufzeit stark reduzieren.

Diesen Ansatz können wir noch stärker ausnutzen, wenn wir die Funktionsauswertungen nicht als unabhängig voneinander behandeln, sondern zuvor berechnete Werte für nachfolgende Funktionsauswertungen benutzen.

- Aus der Dreiecksungleichung folgt für zwei Punkte $\boldsymbol{p}_1, \boldsymbol{p}_2 \in \mathbb{R}^3$ (vgl. Beweis zu Lemma 3.3):

  $$d_{\mathcal{K}}(\boldsymbol{p}_2) > d_{\mathcal{K}}(\boldsymbol{p}_1) - |\,\boldsymbol{p}_1 - \boldsymbol{p}_2\,|.$$

  Unter der Voraussetzung, dass $\boldsymbol{p}_1$ und $\boldsymbol{p}_2$ nahe beieinander liegen, erhalten wir damit eine gute untere Schranke für $d_{\mathcal{K}}(\boldsymbol{p}_2)$.

  Bei der Berechnung von $d_{\mathcal{K}}(\boldsymbol{p}_1)$ speichern wir sowohl das Argument $\boldsymbol{p}_1$ als auch den Funktionswert $d_{\mathcal{K}}(\boldsymbol{p}_1)$ ab. Wenn das lokale Optimierungsverfahren nun den Wert $d_{\mathcal{K}}(\boldsymbol{p}_2)$ berechnen will, haben wir $d_{\mathcal{K}}(\boldsymbol{p}_1) - |\,\boldsymbol{p}_1 - \boldsymbol{p}_2\,|$ als untere Schranke zur Verfügung.

3.2 Kleinste einschließende Quader

In diesem Kapitel wollen wir den Algorithmus zur Berechnung des kleinsten einschließenden Quaders erläutern. Wir gehen dabei analog zum Fall der kleinsten einschließenden Kugel vor.
Im ersten Abschnitt geben wir zwei Möglichkeiten an, um die räumliche Orientierung eines Quaders zu spezifizieren. Beide Darstellungsformen kommen in unserem Algorithmus zum Einsatz.
Der zweite Abschnitt behandelt die *elementaren Ausdehnungsprobleme*. Diese Probleme stellen das Analogon zu den Abstandsproblemen bei der Berechnung der kleinsten einschließenden Kugel dar.
Anschließend erläutern wir die Berechnung des einschließenden Quaders bei vorgegebener Orientierung. Wir zeigen unter anderem, dass die für die Berechnung des kleinsten einschließenden Quaders zu minimierende Zielfunktion nicht konvex ist. Daher werden neben lokalen auch globale Optimierungsverfahren verwendet.
Im letzten Abschnitt diskutieren wir noch einige Möglichkeiten, um den Algorithmus zu beschleunigen.

3.2.1 Orientierung im Raum

Eine einschließende Kugel wurde im wesentlichen durch ihren Mittelpunkt $\boldsymbol{c}$ bestimmt. Ihr Radius ergab sich dann aus dem größten Abstand zwischen dem Punkt $\boldsymbol{c}$ und allen Punkten des Körpers $\mathcal{K}$.
Die Angabe des Mittelpunktes genügt jedoch nicht für die Beschreibung eines Quaders, hier ist zusätzlich die Orientierung im Raum wichtig. Diese kann z. B. durch Angabe der drei Achsen des Quaders spezifiziert werden (siehe

Abb. 3.17). Diese Achsen sollen im Folgenden stets mit $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ und $\boldsymbol{d}_3$ bezeichnet werden.

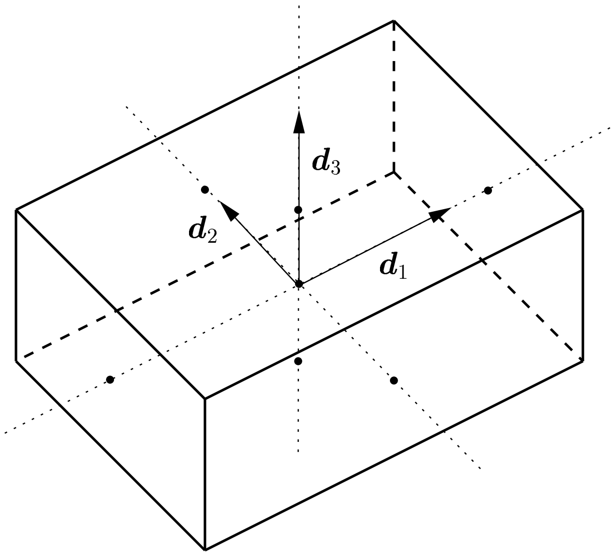

Abbildung 3.17: Orientierung eines Quaders

Die Repräsentation der Orientierung als 3-Tupel von Vektoren des $\mathbb{R}^3$ ist für die Anschauung als auch zur Lösung der elementaren Probleme (siehe nächstes Kapitel) geeignet, nicht jedoch um den *kleinsten* einschließenden Quader zu berechnen. Hierfür müssen wir *alle* möglichen Orientierungen beschreiben können. Aus Effizienzgründen soll diese Darstellung auch möglichst kompakt sein, d. h. unnötige Freiheitsgrade, komplexe Zusatzbedingungen oder Mehrdeutigkeiten sollen nicht auftreten.
Da wir mit Quadern arbeiten, bilden die Vektoren $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ und $\boldsymbol{d}_3$ stets ein Orthogonalsystem. Außerdem ist die Länge der Vektoren unwichtig, d. h. wir können uns auf Einheitsvektoren beschränken und haben ein Orthonormalsystem. Eine Möglichkeit, die Orientierung eines Orthonormalsystems zu beschreiben, sind die *Z-Y-X Eulerwinkel*.
Bei dieser Methode wird die Orientierung durch drei Winkel α, β und γ beschrieben, die folgende Bedeutung haben: Wir beginnen zunächst mit einem Orthonormalsystem $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, dessen Orientierung mit unserem Weltkoordinatensystem übereinstimmt. Zuerst drehen wir dieses System um den Winkel α um die $\boldsymbol{z}$-Achse und erhalten das System $(\boldsymbol{x}', \boldsymbol{y}', \boldsymbol{z}')$. Danach lassen wir dieses System um den Winkel β um die $\boldsymbol{y}'$-Achse rotieren und erhalten $(\boldsymbol{x}'', \boldsymbol{y}'', \boldsymbol{z}'')$. Zum Schluss drehen wir noch um den Winkel γ um die $\boldsymbol{x}''$-Achse und erhalten das endgültige Orthonormalsystem $(\boldsymbol{x}''', \boldsymbol{y}''', \boldsymbol{z}''')$. Alle drei Rotationen zusammen werden durch die Rotationsmatrix

$$R_{\alpha,\beta,\gamma} = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{pmatrix}$$

beschrieben. Für die drei Achsen $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ und $\boldsymbol{d}_3$ des Quaders gilt also

$$\begin{aligned} \boldsymbol{d}_1 &= \begin{pmatrix} \cos\alpha\cos\beta \\ \sin\alpha\cos\beta \\ -\sin\beta \end{pmatrix} \\ \boldsymbol{d}_2 &= \begin{pmatrix} \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma \\ \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma \\ \cos\beta\sin\gamma \end{pmatrix} \\ \boldsymbol{d}_3 &= \begin{pmatrix} \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ \cos\beta\cos\gamma \end{pmatrix}. \end{aligned} \tag{3.5}$$

Umgekehrt kann man aus einem Orthonormalsystem $(\boldsymbol{x}''', \boldsymbol{y}''', \boldsymbol{z}''')$ wieder die Eulerwinkel α, β und γ bestimmen, so dass

$$(\boldsymbol{x}''', \boldsymbol{y}''', \boldsymbol{z}''') = R_{\alpha,\beta,\gamma}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$$

gilt. Den genauen Rechenweg sowie weitere Darstellungsmöglichkeiten für Orientierungen im Raum finden sich in [Cr89]. Ein Resultat der Bestimmung von α, β und γ bei gegebenem Orthonormalsystem $(\boldsymbol{x}''', \boldsymbol{y}''', \boldsymbol{z}''')$ ist die Tatsache, dass man mit $(\alpha, \beta, \gamma) \in [-\pi, \pi] \times [-\frac{\pi}{2}, \frac{\pi}{2}] \times [-\pi, \pi]$ alle möglichen Orientierungen beschreiben kann.
Für unsere Zwecke genügt für das Tupel (α, β, γ) jedoch ein wesentlich kleinerer Definitionsbereich. Abbildung 3.18 zeigt einen Querschnitt eines Körpers $\mathcal{K}$ mit dem einschließenden Quader in verschiedenen Orientierungen. Dabei steht die Achse $\boldsymbol{d}_3$ stets senkrecht zur Zeichenebene. Beginnend links oben wird der Quader gegen den Uhrzeigersinn um die $\boldsymbol{d}_3$-Achse gedreht. Dabei passt sich der Quader dem Körper an, d. h. gegenüberliegende Seiten haben minimalen Abstand und schließen den Körper ein. Nach einer Drehung von $\frac{\pi}{2}$ (rechts unten) erhalten wir wieder einen Quader, der dieselbe Form hat wie zu Beginn links oben, wenn auch mit anderen Werten für α, $\boldsymbol{d}_1$ und $\boldsymbol{d}_2$.

Offenbar ändert sich die Form des Quaders nicht, wenn wir eine Achse $\boldsymbol{d}_i$ durch $-\boldsymbol{d}_i$ ersetzen. Auch ist die Reihenfolge der Achsen $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ und $\boldsymbol{d}_3$ unerheblich, die Werte können beliebig vertauscht werden. Für jeden der drei Winkel α, β und γ genügt also ein Winkelbereich von $\frac{\pi}{2}$. Wir legen uns auf $(\alpha, \beta, \gamma) \in [-\frac{\pi}{4}, \frac{\pi}{4}]^3$ fest.

3.2.2 Elementare Ausdehnungsprobleme

Gegeben sei eine Richtung $\boldsymbol{d} \in \mathbb{R}^3$ und eine Kurve $\mathcal{C}$ oder Fläche $\mathcal{F}$ im $\mathbb{R}^3$. Die Kurve bzw. Fläche soll durch NURBS beschrieben werden können.

Gesucht sind alle lokalen Extrema der Funktion

$$h_{\boldsymbol{d}} : \mathcal{X} \to \mathbb{R}, h_{\boldsymbol{d}}(\boldsymbol{x}) = \boldsymbol{d}^{\mathrm{T}}\boldsymbol{x} \text{ mit } \mathcal{X} = \mathcal{C} \text{ bzw. } \mathcal{X} = \mathcal{F}. \tag{3.6}$$

Die Fälle Gerade, Ebene, Kegel und Zylinder brauchen nicht explizit behandelt zu werden. Bei diesen Kurven- und Flächentypen gibt es stets Endpunkte (bei Kurven) bzw. Begrenzungskanten (bei Flächen), die dieselben Werte

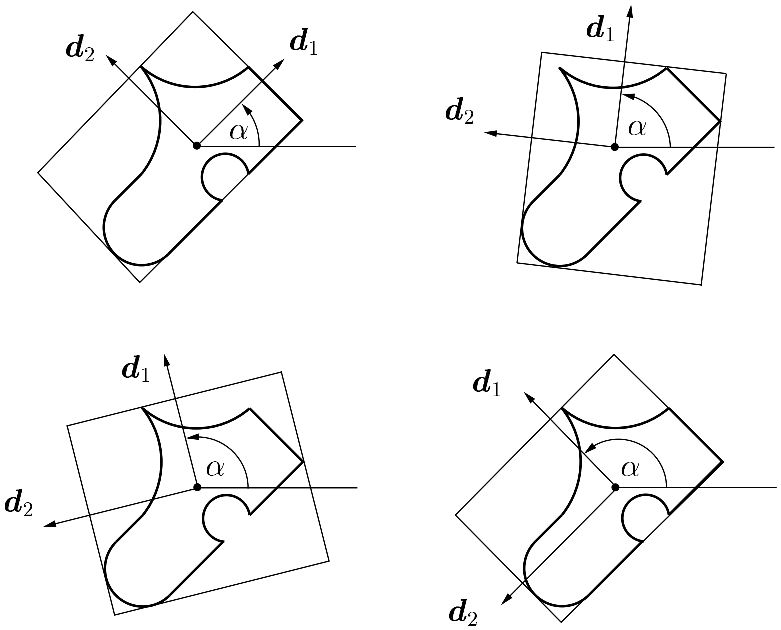

Abbildung 3.18: Drehung des Quaders um $\frac{\pi}{2}$

liefern. Für Ellipsen, Kugeln und Tori geben wir Formeln zur expliziten Berechnung der lokalen Extrema von Funktion (3.6) an. Für NURBS-Kurven und NURBS-Flächen führen wir das Problem auf ein Nullstellenproblem stückweise polynomialer Funktionen zurück.

Ausdehnung einer Ellipse

Die Ellipse sei durch $\boldsymbol{x} = \boldsymbol{c} + \cos\varphi\,\boldsymbol{u} + \sin\varphi\,\boldsymbol{v}$ mit $\boldsymbol{u} \perp \boldsymbol{v}$ gegeben. Um die Extrema der Funktion (3.6) zu bestimmen, betrachten wir die Ableitung nach φ:

$$\frac{d}{d\varphi}\,h_{\boldsymbol{d}}(\boldsymbol{x}) = -\sin\varphi\,\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u} + \cos\,\varphi\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v}.$$

Mit Hilfe von $\sin^2\varphi + \cos^2\varphi = 1$ erhalten wir

$$\cos\varphi = \pm\frac{\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u}}{\sqrt{(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})^2}} \quad \text{und} \quad \sin\varphi = \pm\frac{\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v}}{\sqrt{(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})^2}}\,.$$

Die lokalen Extrema sind dann gegeben durch

$$x_{\substack{max\\min}} = c \pm \frac{\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u}}{\sqrt{(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})^2}}\,\boldsymbol{u} \pm \frac{\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v}}{\sqrt{(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})^2}}\,\boldsymbol{v}\,.$$

Im Falle $(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})^2 = 0$ ist die Richtung $\boldsymbol{d}$ senkrecht zur Ellipsenebene, d. h. $h_{\boldsymbol{d}}(\boldsymbol{x})$ ist eine konstante Funktion.

Ausdehnung einer NURBS-Kurve

Zur Bestimmung der lokalen Extrema von Funktion (3.6) betrachten wir wieder die Ableitung nach u:

$$\begin{aligned} & \frac{d}{du}\big(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{C}(u)\big) = 0 \\ \Leftrightarrow\quad & \boldsymbol{d}^{\mathrm{T}}\boldsymbol{C}'(u) = 0 \\ \Leftrightarrow\quad & \boldsymbol{d}^{\mathrm{T}}\big(\boldsymbol{C}_N(u)\,\boldsymbol{C}'_Z(u) - \boldsymbol{C}'_N(u)\,\boldsymbol{C}_Z(u)\big) = 0 \end{aligned}$$

Die lokalen Extrema sind durch die Nullstellen der stückweise polynomialen Funktion

$$f : [0,1] \to \mathbb{R},\;\; f(u) = \boldsymbol{d}^{\mathrm{T}}\big(\boldsymbol{C}_N(u)\,\boldsymbol{C}'_Z(u) - \boldsymbol{C}'_N(u)\,\boldsymbol{C}_Z(u)\big)$$

gegeben. Bezeichnet p den Grad der NURBS-Kurve $\boldsymbol{C}$, so sind die in f auftretenden Potenzen von u vom Grad $2p-1$. Sind die Gewichte w_i aller Kontrollpunkte gleich, so ist die Funktion $\boldsymbol{C}_N(u)$ konstant und der Grad von f reduziert sich auf $p-1$.

Ausdehnung einer Kugel

Die Kugel sei durch $|\,\boldsymbol{x} - \boldsymbol{c}\,| = r$ gegeben. Die lokalen Extrema der Funktion (3.6) sind die Schnittpunkte der Kugel mit der Geraden durch den Kugelmittelpunkt $\boldsymbol{c}$ und Richtung $\boldsymbol{d}$:

$$\boldsymbol{x}_{\substack{max\\min}} = \boldsymbol{c} \pm r\,\boldsymbol{d}.$$

Ausdehnung eines Torus

Der Torus sei durch $\boldsymbol{x} = \boldsymbol{c} + (R + r\cos\psi)\,(\cos\varphi\,\boldsymbol{u} + \sin\varphi\,\boldsymbol{v}) + r\sin\psi\boldsymbol{n}$ mit $R > r$ gegeben, wobei $\boldsymbol{u}$, $\boldsymbol{v}$ und $\boldsymbol{n}$ ein Orthonormalsystem bilden.

Zur Bestimmung der lokalen Extrema betrachten wir zunächst die Ableitung nach φ:

$$\frac{d}{d\varphi}\,h_{\boldsymbol{d}}(\boldsymbol{x}) = (R + r\cos\psi)\,(-\sin\varphi\,\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u} + \cos\varphi\,\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})\,.$$

Wie bei der Ellipse erhält man

$$\cos\varphi = \pm\frac{\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u}}{\sqrt{(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})^2}} \quad \text{und} \quad \sin\varphi = \pm\frac{\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v}}{\sqrt{(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})^2}}\,.$$

Da $\boldsymbol{u}$, $\boldsymbol{v}$ und $\boldsymbol{n}$ ein Orthonormalsystem bilden, gilt:

$$\begin{aligned} (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})^2 + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n})^2 &= |\,\boldsymbol{d}\,|^2 = 1 \\ (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{u})\boldsymbol{u} + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{v})\boldsymbol{v} + (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n})\boldsymbol{n} &= \boldsymbol{d}. \end{aligned}$$

Damit erhält man für die Ableitung nach ψ:

$$\frac{d}{d\psi}\,h_{\boldsymbol{d}}(\boldsymbol{x}) = -r\sin\psi\,\sqrt{1 - (\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n})^2} + r\cos\psi\,\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n}\,.$$

Für die Nullstellen dieser Ableitung gilt

$$\cos\psi = \pm\sqrt{1-(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n})^2} \quad \text{und} \quad \sin\psi = \pm\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n}.$$

Die lokalen Extrema sind dann gegeben durch

$$x_{\substack{max\\min}} = c \pm R\,\frac{\boldsymbol{d}-(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n})\boldsymbol{n}}{\sqrt{1-(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n})^2}} \pm r\boldsymbol{d}.$$

Im Fall $\boldsymbol{d}^{\mathrm{T}}\boldsymbol{n} = 1$ ist die Richtung $\boldsymbol{d}$ parallel zur Achse des Torus. Dann liegen die lokalen Extrema auf den beiden Kreisbahnen

$$x_{\substack{max\\min}} = c \pm r\,\boldsymbol{d} + R\,(\cos\varphi\,\boldsymbol{u} + \sin\varphi\,\boldsymbol{v}).$$

Ausdehnung einer NURBS-Fläche

Zur Bestimmung der lokalen Extrema betrachten wir die partiellen Ableitungen nach u und v:

$$\begin{aligned}
&\left\{\begin{array}{l} \frac{\partial}{\partial u}\big(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{S}(u,v)\big) = 0\\ \frac{\partial}{\partial v}\big(\boldsymbol{d}^{\mathrm{T}}\boldsymbol{S}(u,v)\big) = 0 \end{array}\right.\\
\Leftrightarrow\ &\left\{\begin{array}{l} \boldsymbol{d}^{\mathrm{T}}\frac{\partial}{\partial u}\boldsymbol{S}(u,v) = 0\\ \boldsymbol{d}^{\mathrm{T}}\frac{\partial}{\partial v}\boldsymbol{S}(u,v) = 0 \end{array}\right.\\
\Leftrightarrow\ &\left\{\begin{array}{l} \boldsymbol{d}^{\mathrm{T}}\big(\boldsymbol{S}_N(u,v)\ \frac{\partial}{\partial u}\boldsymbol{S}_Z(u,v) - \frac{\partial}{\partial u}\boldsymbol{S}_N(u,v)\ \boldsymbol{S}_Z(u,v)\big) = 0\\ \boldsymbol{d}^{\mathrm{T}}\big(\boldsymbol{S}_N(u,v)\ \frac{\partial}{\partial v}\boldsymbol{S}_Z(u,v) - \frac{\partial}{\partial v}\boldsymbol{S}_N(u,v)\ \boldsymbol{S}_Z(u,v)\big) = 0 \end{array}\right.
\end{aligned}$$

Die lokalen Extrema sind also durch die Nullstellen der Funktion

$$\boldsymbol{f}:[0,1]^2\to\mathbb{R}^2,\ \boldsymbol{f}(u,v) = \left(\begin{array}{c} \boldsymbol{d}^{\mathrm{T}}\big(\boldsymbol{S}_N\ \frac{\partial}{\partial u}\boldsymbol{S}_Z - \frac{\partial}{\partial u}\boldsymbol{S}_N\ \boldsymbol{S}_Z\big)\\ \boldsymbol{d}^{\mathrm{T}}\big(\boldsymbol{S}_N\ \frac{\partial}{\partial v}\boldsymbol{S}_Z - \frac{\partial}{\partial v}\boldsymbol{S}_N\ \boldsymbol{S}_Z\big) \end{array}\right)$$

gegeben. Die Komponenten von $\boldsymbol{f}$ sind wieder stückweise polynomiale Funktionen. Bezeichnen u und v den Grad der NURBS-Fläche $\boldsymbol{S}$, so haben u und v in $\boldsymbol{f}$ höchstens den Grad $2p$ bzw. $2q$. Falls die Gewichte $w_{i,j}$ aller Kontrollpunkte gleich sind, dann ist $\boldsymbol{S}_N(u,v)$ eine konstante Funktion. Die Potenzen von u und v in $\boldsymbol{f}$ haben in diesem Fall höchstens den Grad p bzw. q.

3.2.3 Berechnung des einschließenden Quaders

Gegeben sei ein Körper $\mathcal{K}$, repräsentiert durch die Flächenstücke $\mathcal{K}_f$, die Begrenzungskanten $\mathcal{K}_e$ und die Eckpunkte $\mathcal{K}_p$. Ferner sei $\boldsymbol{d}\in\mathbb{R}^3$, $|\,\boldsymbol{d}\,| = 1$ eine Richtung.

Mit den Verfahren aus dem vorherigen Kapitel und den Enthaltenseinstests aus Kapitel 3.1.2 können wir jetzt die Ausdehnung $a_{\mathcal{K}}(\boldsymbol{d})$ eines Körpers $\mathcal{K}$ in Richtung $\boldsymbol{d}$ berechnen:

$$\begin{aligned}
a_{\mathcal{K}}(\boldsymbol{d}) &= \max_{\boldsymbol{x}\in\mathcal{K}}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x} - \min_{\boldsymbol{x}\in\mathcal{K}}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x} && (3.7)\\
\max_{\boldsymbol{x}\in\mathcal{K}}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x} &= \max\{\max_{f\in\mathcal{K}_f}\max_{\boldsymbol{x}\in f}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x},\ \max_{e\in\mathcal{K}_e}\max_{\boldsymbol{x}\in e}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x},\ \max_{p\in\mathcal{K}_p}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x}\} && (3.8)\\
\min_{\boldsymbol{x}\in\mathcal{K}}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x} &= \min\{\min_{f\in\mathcal{K}_f}\min_{\boldsymbol{x}\in f}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x},\ \min_{e\in\mathcal{K}_e}\min_{\boldsymbol{x}\in e}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x},\ \min_{p\in\mathcal{K}_p}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x}\}. && (3.9)
\end{aligned}$$

Wie bei der Berechnung des maximalen Abstands mit Funktion (3.3) ist es notwendig, nicht nur Flächenstücke, sondern auch die Begrenzungskanten und Eckpunkte des Körpers zu betrachten. Andernfalls würden wir lokale Extrema, die durch Kanten und Eckpunkte realisiert werden, übersehen.
Die Funktion (3.7) ist als Komposition stetiger Funktion selbst wieder stetig. Allerdings ist sie im Gegensatz zu Funktion (3.3) nicht konvex.

Lemma 3.6. *Die Ausdehnung $a_{\mathcal{K}}(\boldsymbol{d})$ eines Körpers $\mathcal{K}$ in einer Richtung $\boldsymbol{d} \in \mathbb{R}^3$ mit $|\,\boldsymbol{d}\,| = 1$ ist im Allgemeinen keine konvexe Funktion.*[1]

Beweis:
Wir beschränken uns der Einfachheit halber auf ein Beispiel im zweidimensionalen Fall. Das Beispiel lässt sich sofort auf den dreidimensionalen Fall übertragen (z. B. ein Prisma mit achteckiger Grundfläche und beliebiger Höhe).
Wir betrachten ein regelmäßiges Achteck, bei dem gegenüberliegende Kanten den Abstand a haben (siehe Abb. 3.19). Für $\alpha = 0$ und $\alpha = \frac{\pi}{4}$ (links oben und unten) gilt $a_{\mathcal{K}}(\boldsymbol{d}) = a$. Im Fall $\alpha = \frac{\pi}{8}$ (rechts oben) gilt aber

$$a_{\mathcal{K}}(\boldsymbol{d}) = \frac{1}{\cos\frac{\pi}{8}}\, a \approx 1.082\, a > a\,.$$

Also ist die Funktion nicht konvex. $\square$

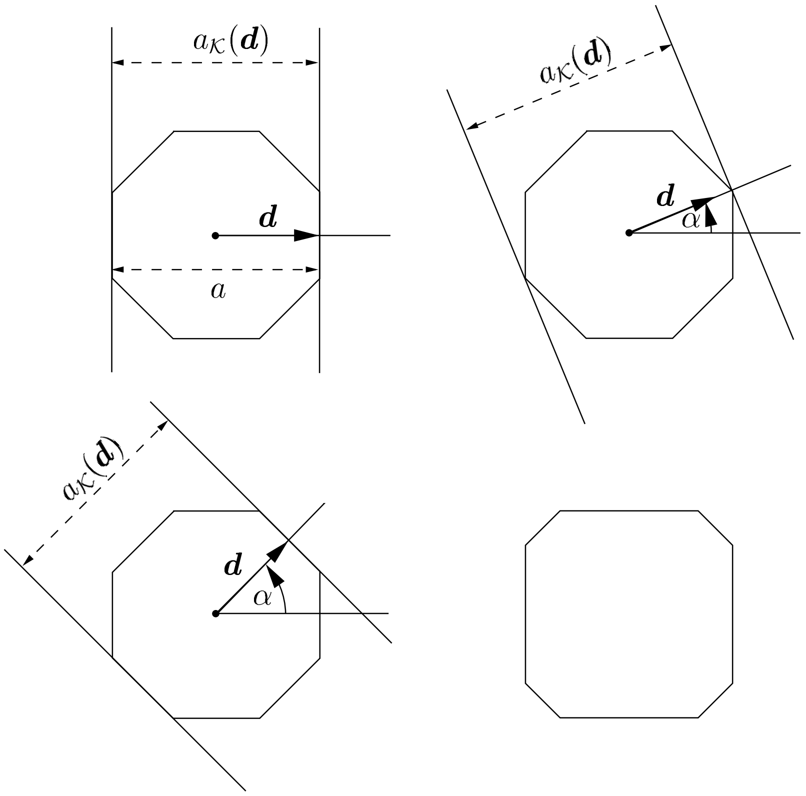

Abbildung 3.19: Ausdehnung eines Achtecks

[1] Da die Menge der Einheitsvektoren nicht konvex ist, definieren wir die Ausdehnung $a_{\mathcal{K}}(\boldsymbol{d})$ für $d \in \mathbb{R}^3 \backslash \{\boldsymbol{0}\}$ mit $|\,\boldsymbol{d}\,| \neq 1$ als $a_{\mathcal{K}}(|\,\boldsymbol{d}\,|)$. Ferner sei $a_{\mathcal{K}}(\boldsymbol{0}) = 0$.

In diesem Fall haben beide lokalen Minima denselben Wert. Durch Ändern der Seitenlängen des Achtecks erhält man sofort ein Beispiel, bei dem sich die Werte der beiden lokalen Minima unterscheiden (siehe Abb. 3.19, rechts unten).

Wir können nun bei gegebenen Orientierungsparametern α, β und γ den einschließenden Quader bestimmen. Zunächst berechnen wir die Achsen des Quaders $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ und $\boldsymbol{d}_3$ nach Formel (3.5). Danach berechnen wir jeweils für $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ und $\boldsymbol{d}_3$ die Werte $\max_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}_i^{\mathrm{T}}\boldsymbol{x}$ und $\min_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}_i^{\mathrm{T}}\boldsymbol{x}$ nach Formel (3.8) und (3.9). Die Ausdehnung $a_{\mathcal{K}}(\boldsymbol{d}_i)$ kann dann nach (3.7) berechnet werden. Ein Eckpunkt $\boldsymbol{p}$ des Quaders ist durch die Lösung des Gleichungssystems

$$\begin{aligned}
\boldsymbol{d}_1^{\mathrm{T}}\boldsymbol{p} &= \min_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}_1^{\mathrm{T}}\boldsymbol{x} \\
\boldsymbol{d}_2^{\mathrm{T}}\boldsymbol{p} &= \min_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}_2^{\mathrm{T}}\boldsymbol{x} \\
\boldsymbol{d}_3^{\mathrm{T}}\boldsymbol{p} &= \min_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}_3^{\mathrm{T}}\boldsymbol{x}
\end{aligned}$$

gegeben. Für die Kanten $\boldsymbol{a}$, $\boldsymbol{b}$ und $\boldsymbol{c}$ des Quaders gilt:

$$\begin{aligned}
\boldsymbol{a} &= a_{\mathcal{K}}(\boldsymbol{d}_1)\, \boldsymbol{d}_1 \\
\boldsymbol{b} &= a_{\mathcal{K}}(\boldsymbol{d}_2)\, \boldsymbol{d}_2 \\
\boldsymbol{c} &= a_{\mathcal{K}}(\boldsymbol{d}_3)\, \boldsymbol{d}_3\,.
\end{aligned}$$

Das Volumen des Quaders beträgt dann

$$V_{\mathcal{K}}(\alpha,\beta,\gamma) = \prod_{i=1}^{3} a_{\mathcal{K}}(\boldsymbol{d}_i)\,. \tag{3.10}$$

Funktion (3.10) ist als Komposition stetiger Funktionen ebenfalls stetig, aber nicht konvex.

Lemma 3.7. *Das Volumen $V_{\mathcal{K}}(\alpha,\beta,\gamma)$ des einschließenden Quaders für einen Körper $\mathcal{K}$ ist im Allgemeinen keine konvexe Funktion.*

Beweis:
Analog zu Beweis von Lemma 3.6 betrachten wir ein gerades Prisma mit regelmäßiger achteckiger Grundfläche und Höhe h. Wie in Abbildung 3.19 sei der Abstand zweier gegenüberliegender Seiten der Grundfläche a.
Es gibt zwei Orientierungen des Quaders, bei denen alle Seiten parallel zu Flächen des Prismas sind. In diesen beiden Fällen beträgt das Volumen des Quaders a^2h, in allen anderen Fällen ist das Volumen größer. Also ist die Funktion nicht konvex. □

3.2.4 Minimierung des Quadervolumens

Wir wollen nun für einen gegebenen Körper $\mathcal{K}$ den kleinsten einschließenden Quader bestimmen. Das Volumen dieses kleinsten einschließenden Quaders ist das globale Minimum der Funktion

$$V_{\mathcal{K}} : \left[-\tfrac{\pi}{4}, \tfrac{\pi}{4}\right]^3 \to \mathbb{R},\ V_{\mathcal{K}}(\alpha,\beta,\gamma) = \prod_{i=1}^{3} a_{\mathcal{K}}(\boldsymbol{d}_i)\,. \tag{3.11}$$

Der Quader selbst kann durch die bei der Berechnung dieser Funktion auftretenden Zwischenergebnisse wie im vorherigen Kapitel gezeigt bestimmt werden. Wie in Lemma 3.7 gezeigt, ist diese Funktion nicht konvex. Die Funktion besitzt also mehrere lokale Minima, die im Allgemeinen unterschiedliche Werte haben.

Lokale Optimierungsverfahren

Als lokale Optimierungsverfahren verwenden wir wieder das *Downhill-Simplex-Verfahren* und die *Methode von Powell*. Der Nachteil aller lokalen Optimierungsverfahren besteht darin, dass sie nur ein lokales Optimum, aber nicht unbedingt das globale Optimum finden. Bei nicht-konvexen Funktionen entscheidet dabei die Wahl des Startwertes darüber, welches der lokalen Minima gefunden wird. Um dennoch das globale Minimum zu finden, verwenden wir folgenden Trick:
Wir starten das lokale Optimierungsverfahren sehr oft, dabei verteilen wir die Startwerte gleichmäßig über den Suchraum. Das globale Minimum befindet sich dann mit hoher Wahrscheinlichkeit unter den gefundenen lokalen Minima. Erstaunlicherweise genügen bereits wenige Aufrufe des lokalen Optimierungsverfahrens, um mit recht hoher Wahrscheinlichkeit das globale Minimum zu finden (siehe Ergebnisse in Kapitel 3.3).

Globale Optimierungsverfahren

Als globales Optimierungsverfahren kommt der *Branch-and-Bound*-Algorithmus aus [Rat92] zum Einsatz. Da wir nur die Funktion (3.11) und nicht deren Ableitungen berechnen können, verwenden wir eine einfache Version mit Mittelpunktstest; der Monotonietest und der Konkavitätstest kommen nicht zum Einsatz.
Der Branch-and-Bound-Algorithmus bestimmt das globale Optimum x^* einer Funktion $f : \mathcal{X} \to \mathbb{R}$. Durch rekursive Unterteilung der Grundmenge $\mathcal{X}$ erhält man eine Abschätzung für den Wert $f(x^*)$ des globalen Optimums und kann dadurch Teile der Grundmenge von der weiteren Suche ausschließen.
Der Algorithmus besteht im wesentlichen aus drei Elementen:

- *Unterteilung des Suchraums:* Hierbei wird eine zu untersuchende Menge $\mathcal{Y} \subseteq \mathcal{X}$ in disjunkte Teilmengen $\mathcal{Y}_1, \ldots \mathcal{Y}_n$ mit $\bigcup_{i=1}^n \mathcal{Y}_i = \mathcal{Y}$ zerlegt. In unserem Fall sind alle Mengen achsenorientierte Quader, dargestellt durch Intervallvektoren. Wir zerlegen einen solchen Quader $\mathcal{Y}$ in acht achsenorientierte Quader $\mathcal{Y}_i$, indem wir das Intervall in jeder Komponente von $\mathcal{Y}_i$ halbieren.

- *Berechnung von unteren Schranken:* Für jede Teilmenge $\mathcal{Y} \subseteq \mathcal{X}$ wird eine untere Schranke $s_\mathcal{Y}$ mit $s_\mathcal{Y} \leq f(\mathcal{Y})$ benötigt. Wir benutzen wieder die Intervallarithmetik zur Berechnung einer solchen Schranke und setzen $s_\mathcal{Y} := \inf\ F(\mathcal{Y})$.

- *Berechnung von oberen Schranken:* Zusätzlich werden obere Schranken für den Wert des globalen Optimums x^* berechnet. Beispielsweise kann man für jeden auftretenden Quader $\mathcal{Y}$ den Funktionswert im Mittelpunkt des Quaders bestimmen. Von besonderem Interesse sind dabei möglichst kleine obere Schranken.

Sei nun eine obere Schranke S für den Wert $f(x^*)$ und eine untere Schranke $s_{\mathcal{Y}}$ für die Funktionswerte von f auf $\mathcal{Y}$ gegeben. Gilt nun $s_{\mathcal{Y}} > S$, so kann die weitere Betrachtung der Menge $\mathcal{Y}$ entfallen, denn es gilt

$$f(x^*) \leq S < s_{\mathcal{Y}} \leq \inf \{f(y) \mid y \in \mathcal{Y}\} ,$$

also folgt $x^* \notin \mathcal{Y}$. Der Begriff *Mittelpunktstest* ist darauf zurückzuführen, dass der Wert der oberen Schranke S in der Regel durch eine Funktionsauswertung an einem Quadermittelpunkt bestimmt wird.
Die rekursive Unterteilung des Suchraums bricht man ab, sobald die Kantenlängen eines Quaders $\mathcal{Y}$ unter eine vorgegebene Schranke fallen.

Allerdings hat sich gezeigt, dass der Branch-and-Bound-Algorithmus für dieses Problem eine sehr schlechte Laufzeit hat. Selbst für einfache Polyeder liegt die Laufzeit im Minutenbereich und steigt bei wachsender Komplexität des Körpers stark an. Der Hauptgrund liegt in den schlechten unteren Schranken $s_{\mathcal{Y}}$, so dass der Branch-and-Bound-Algorithmus nur wenige Teilmengen von $\mathcal{X}$ von der Bearbeitung ausschließen kann.
Die schlechten unteren Schranken $s_{\mathcal{Y}}$ sind durch den Einsatz der Intervallarithmetik zur Auswertung der Funktion (3.11) bedingt. Wir bezeichnen mit $d(A) = \sup A - \inf A$ die *Länge* eines Intervalls A. Ferner sei $V_{\mathcal{K},IA}$ die Intervallauswertung von Funktion (3.11). Für einen Quader $\mathcal{Y}$ mit Kantenlänge $\frac{1}{100}$ und ein Polyeder als Körper $\mathcal{K}$ haben wir für das Verhältnis zwischen Länge der Intervallauswertung und des Wertebereichs Werte von

$$\frac{d\left(V_{\mathcal{K},IA}(\mathcal{Y})\right)}{d\left(V_{\mathcal{K}}(\mathcal{Y})\right)} \approx 100$$

gemessen. Dabei lag $V_{\mathcal{K}}(\mathcal{Y})$ etwa mittig im Intervall $V_{\mathcal{K},IA}(\mathcal{Y})$. Aufgrund der starken Überschätzung beim Einsatz der Intervallarithmetik kommen die schlechten unteren Schranken $s_{\mathcal{Y}}$ zu Stande.

3.2.5 Optimierungen

Zunächst sei bemerkt, dass beide Funktionen (3.8) und (3.9) nicht getrennt, sondern parallel ausgewertet werden. Dadurch spart man viele doppelte Berechnungen bei der Bestimmung der lokalen Extrema.

Um die Berechnungen der beiden Funktionen weiter zu beschleunigen, verwenden wir dieselben Methoden wie bei der Berechnung des maximalen Abstands durch Funktion (3.3).

- Auch hier ist die Rechenzeit stark von der Anzahl der durchgeführten Enthaltenseinstests für Flächenstücke abhängig. Ein Teil dieser Enthaltenseinstests lässt sich ohne Zusatzaufwand vermeiden:

Der Enthaltenseinstest für einen Punkt $\boldsymbol{x}$ wird nur dann durchgeführt, wenn der Wert $\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x}$ kleiner als das bisher berechnete Minimum oder größer als das bisher berechnete Maximum ist. Liegt $\boldsymbol{d}^{\mathrm{T}}\boldsymbol{x}$ zwischen diesen beiden Werten, so kann der Enthaltenseinstest entfallen, ohne dass sich das Endergebnis ändert.

- Die Auswirkungen der vorherigen Verbesserung sind umso stärker, je früher wir gute Schranken für $\max_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}^{\mathrm{T}}\boldsymbol{x}$ und $\min_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}^{\mathrm{T}}\boldsymbol{x}$ haben.

  Wir betrachten bei der Berechnung von Funktion (3.8) und (3.9) zunächst die Eckpunkte $\mathcal{K}_p$ und Kanten $\mathcal{K}_e$ des Körpers und erhalten so erste brauchbare Schranken. Erst dann betrachten wir die Flächenstücke $\mathcal{K}_f$ und können dabei die meisten Enthaltenseinstests einsparen.

Im Gegensatz zur Berechnung der kleinsten einschließenden Kugel ist es uns hier nicht gelungen, aufeinanderfolgende Funktionsaufrufe zu beschleunigen.

- Um das Volumen $V_{\mathcal{K}}(\alpha, \beta, \gamma)$ des einschließenden Quaders nach (3.10) zu berechnen, sind jeweils drei Berechnungen der Werte (3.7) bis (3.9) notwendig. Dabei bilden die Argumente $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ und $\boldsymbol{d}_3$ ein Orthonormalsystem. Es war uns allerdings nicht möglich, diese Eigenschaft für eine Beschleunigung der Berechnung auszunutzen.

- Wie bei der kleinsten einschließenden Kugel wäre eine Abschätzung der Form

$$\begin{aligned} \max_{\boldsymbol{x}\in\mathcal{K}} \tilde{\boldsymbol{d}}_i^{\mathrm{T}}\boldsymbol{x} &> \max_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}_i^{\mathrm{T}}\boldsymbol{x} - \delta_{max}(\boldsymbol{d}_i, \tilde{\boldsymbol{d}}_i) \\ \min_{\boldsymbol{x}\in\mathcal{K}} \tilde{\boldsymbol{d}}_i^{\mathrm{T}}\boldsymbol{x} &< \min_{\boldsymbol{x}\in\mathcal{K}} \boldsymbol{d}_i^{\mathrm{T}}\boldsymbol{x} + \delta_{min}(\boldsymbol{d}_i, \tilde{\boldsymbol{d}}_i) \end{aligned}$$

  sehr nützlich. Auch hier ist es uns nicht gelungen, brauchbare Werte für $\delta_{max}(\boldsymbol{d}_i, \tilde{\boldsymbol{d}}_i)$ und $\delta_{min}(\boldsymbol{d}_i, \tilde{\boldsymbol{d}}_i)$ zu finden.

3.3 Ergebnisse

Zum Schluss wollen wir noch einige Beispiele präsentieren. Zur Berechnung der Hüllkörper kamen dabei ausschließlich die in dieser Arbeit vorgestellten Algorithmen zum Einsatz. Außerdem wollen wir demonstrieren, dass man auch mit lokalen Optimierungsverfahren bei der Berechnung des kleinsten einschließenden Quaders sehr gute Ergebnisse erzielen kann.
Alle Messungen wurden auf einem Pentium III mit 450 MHz Taktfrequenz und 128 MB RAM durchgeführt. Als Compiler wurde der gcc 2.95.3 unter Linux mit Kernel 2.2.19 verwendet. Die Eingabedaten liegen im SAT-Format vor. Eine Beschreibung dieses Formats findet sich in [Spa].
Die Daten für die Berechnung der kleinsten einschließenden Kugel sind in Tabelle 3.1 dargestellt, Tabelle 3.2 zeigt die entsprechenden Werte für die Berechnung

des kleinsten einschließenden Quaders. Die Spalten der beiden Tabellen haben folgende Bedeutung:

Flächenzahl: Anzahl der Flächenstücke, aus denen sich das Objekt zusammensetzt,

Funktionsaufrufe: Anzahl der Aufrufe von Funktion (3.4) bzw. (3.10) durch das Optimierungsverfahren,

Laufzeit: benötigte Zeit in Millisekunden, um die kleinste einschließende Kugel bzw. den kleinsten einschließenden Quader zu berechnen.

Die Anzahl der Funktionsaufrufe und die benötigte Laufzeit wurde für beide Optimierungsverfahren, das *Downhill-Simplex-Verfahren (DSV)* und die *Methode von Powell (MvP)*, getrennt untersucht. Als Beispiele dienen folgende vier Szenen:

`sphere6.sat`: Das Objekt besteht aus der Vereinigung von sechs Kugeln mit Radius 1 und den Mittelpunkten $(\pm\frac{1}{2}, 0, 0)$, $(0, \pm\frac{1}{2}, 0)$ und $(0, 0, \pm\frac{1}{2})$.

`smoothcube.sat`: Dieses Beispiel besteht aus einem Würfel, dessen Ecken und Kanten abgerundet wurden.

`misc.sat`: Das Objekt enthält alle besprochenen Kanten- und Flächentypen. Ein Würfel wurde mit zwei Zylindern vereinigt und hat eine zylindrische Durchbohrung erhalten. An einer Ecke wurde eine Kugel aufgesetzt, an einer anderen Ecke wurde eine kugelförmige Aussparung angebracht. Die Szene enthält außerdem einen Torus und ein Torusstück, sowie ein Zylinderstück und einen Kegelstumpf.

`rolladen.sat`: Profil einer Rolladenlamelle.

Die Objekte sind in Abbildung 3.24 bis 3.27 zusammen mit der kleinsten einschließenden Kugel bzw. dem kleinsten einschließenden Quader zu sehen.

	Flächenzahl	Funktionsaufrufe		Laufzeit	
		DSV	MvP	DSV	MvP
`sphere6.sat`	6	155	166	22,9	24,3
`smoothcube.sat`	26	168	88	44,9	27,3
`misc.sat`	26	193	454	42,7	117,7
`rolladen.sat`	99	205	228	223,8	247,4

Tabelle 3.1: Berechnung der kleinsten einschließenden Kugel

Die Ergebnisse zeigen deutlich, dass die Berechnung des kleinsten einschließenden Quaders deutlich aufwendiger ist. Die Berechnung des Quadervolumens nach Funktion (3.10) dauert wesentlich länger als die Berechnung des Kugelvolumens durch Funktion (3.4). Dies hängt damit zusammen, dass zur Berechnung des Quadervolumens die Ausdehnung des Körpers in *drei* Richtungen

	Flächenzahl	Funktionsaufrufe		Laufzeit	
		DSV	MvP	DSV	MvP
`sphere6.sat`	6	160	328	103,1	223,7
`smoothcube.sat`	26	172	89	151,2	57,0
`misc.sat`	26	238	379	159,8	271,8
`rolladen.sat`	99	169	274	504,8	757,6

Tabelle 3.2: Berechnung des kleinsten einschließenden Quaders

berechnet werden muss. Für die Berechnung des Kugelvolumens genügt dagegen *eine* Berechnung des maximalen Abstands.

Für die Berechnung des kleinsten einschließenden Quaders wurden lokale Optimierungsverfahren verwendet, obwohl die Zielfunktion nicht konvex ist. Durch sehr viele Aufrufe des Optimierungsverfahrens mit unterschiedlichen Startwerten hofft man, auch das globale Optimum zu finden. Wir wollen zeigen, dass man bereits mit wenigen Wiederholungen sehr gute Resultate erzielt.
Tabelle 3.3 zeigt einige statistische Werte der berechneten lokalen Minima bei 1000 Durchläufen. (Da die Objekte unterschiedliche Größe haben, wurden die Werte normiert, so dass das Minimum stets den Wert 1 hat.)

	Minimum	Median	Mittelwert	Maximum
`sphere6.sat`	1.000000	1.000001	1.002341	1.041835
`smoothcube.sat`	1.000000	1.000001	1.002818	1.558343
`misc.sat`	1.000000	1.000033	1.010920	1.237408
`rolladen.sat`	1.000000	1.000001	1.000648	1.157579

Tabelle 3.3: Verteilung der Werte der lokalen Minima

Aufgrund der sehr hohen Zahl von Aufrufen und der zufälligen Wahl der Startwerte können wir davon ausgehen, dass bei der Suche nach lokalen Minima auch mindestens einmal das globale Optimum gefunden wurde. Mit anderen Worten, das globale Optimum hat (normiert) den Wert 1.
Auffallend ist die Tatsache, dass sowohl der Median als auch der Mittelwert der gefundenen Minima sehr nahe am globalen Optimum liegen. Das bedeutet, dass viele Werte in der Nähe des globalen Optimums liegen und es nur sehr wenige "Ausrutscher" nach oben gibt.
Die Wahrscheinlichkeit, dass der Wert eines lokalen Minimums über dem Median liegt, beträgt $\frac{1}{2}$. Bei n Aufrufen des Optimierungsverfahrens liegen die Werte *aller* berechneten Minima mit einer Wahrscheinlichkeit von $\frac{1}{2^n}$ über dem Median. Oder umgekehrt, bei n Aufrufen erhält man für das Minimum einen Wert, der mit einer Wahrscheinlichkeit von $1 - \frac{1}{2^n}$ unter dem Median liegt. Da der Median sehr nahe am globalen Optimum liegt, genügen wenige Aufrufe des lokalen Optimierungsverfahrens, um eine sehr gute Näherung für das globale Optimum zu erhalten.

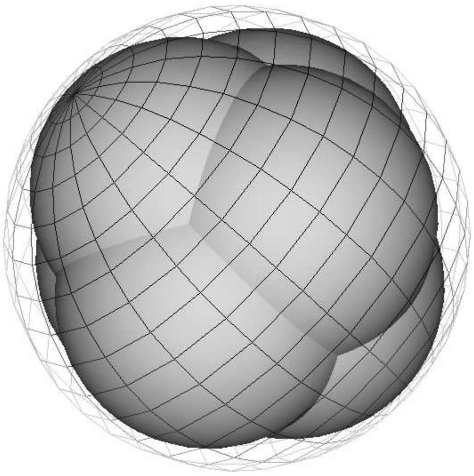

Abbildung 3.20: `sphere6.sat` - kleinste einschließende Kugel

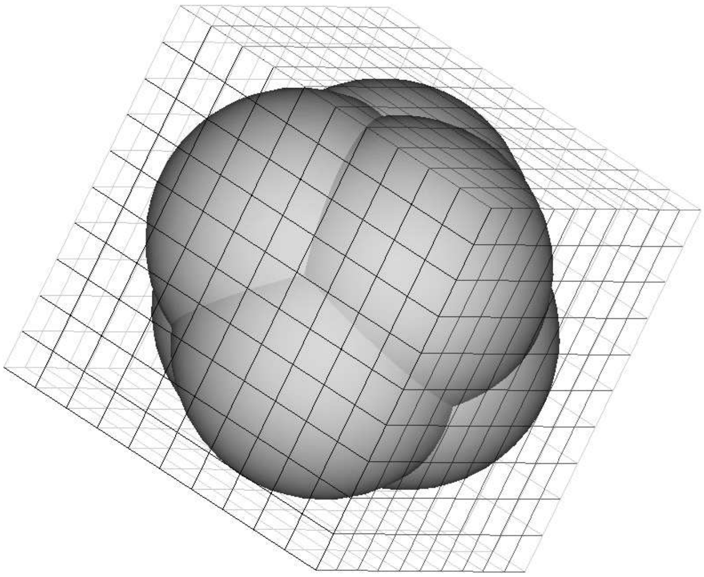

Abbildung 3.21: `sphere6.sat` - kleinster einschließender Quader

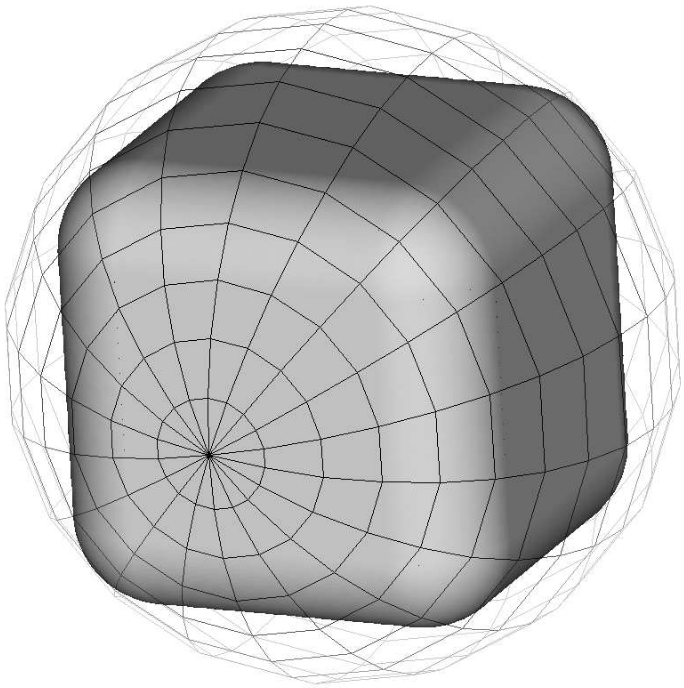

Abbildung 3.22: `smoothcube.sat` - kleinste einschließende Kugel

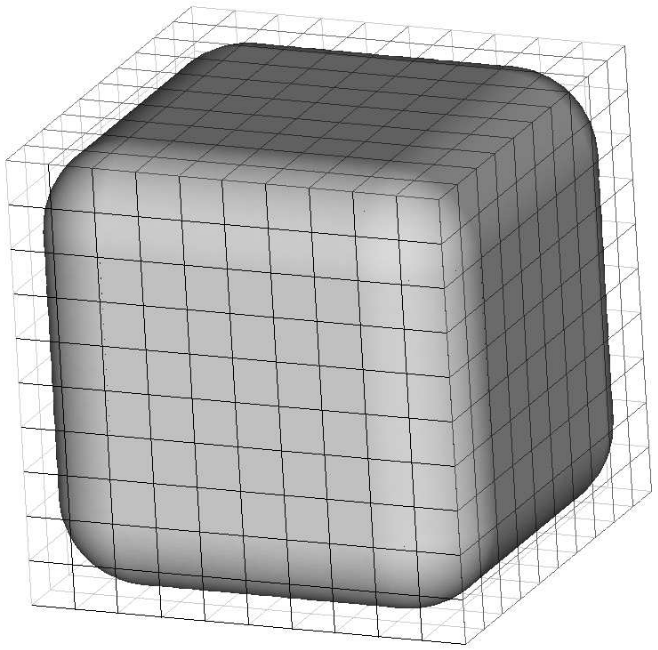

Abbildung 3.23: `smoothcube.sat` - kleinster einschließender Quader

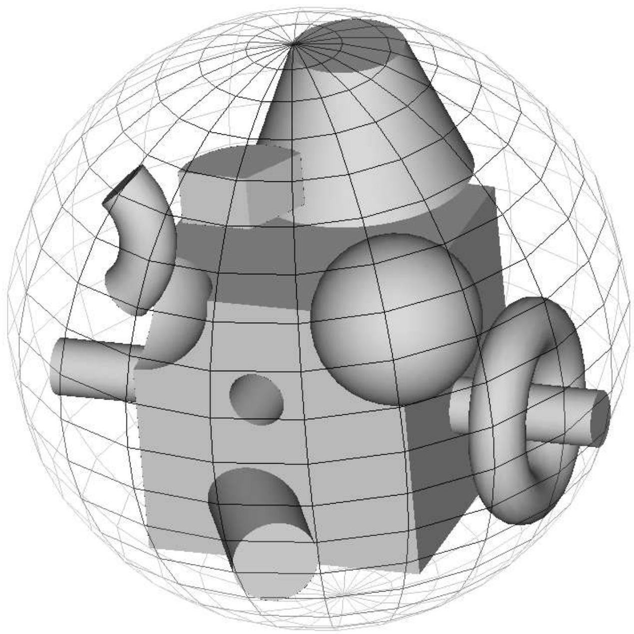

Abbildung 3.24: `misc.sat` - kleinste einschließende Kugel

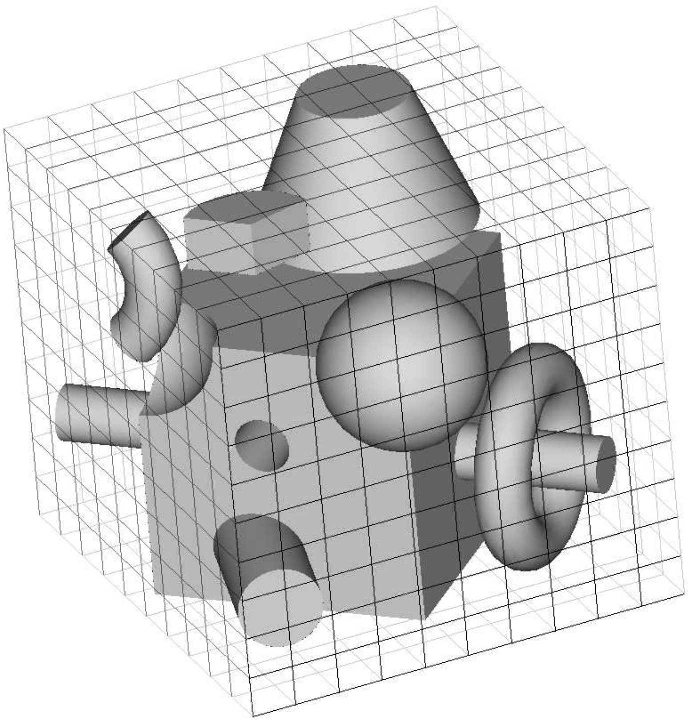

Abbildung 3.25: `misc.sat` - kleinster einschließender Quader

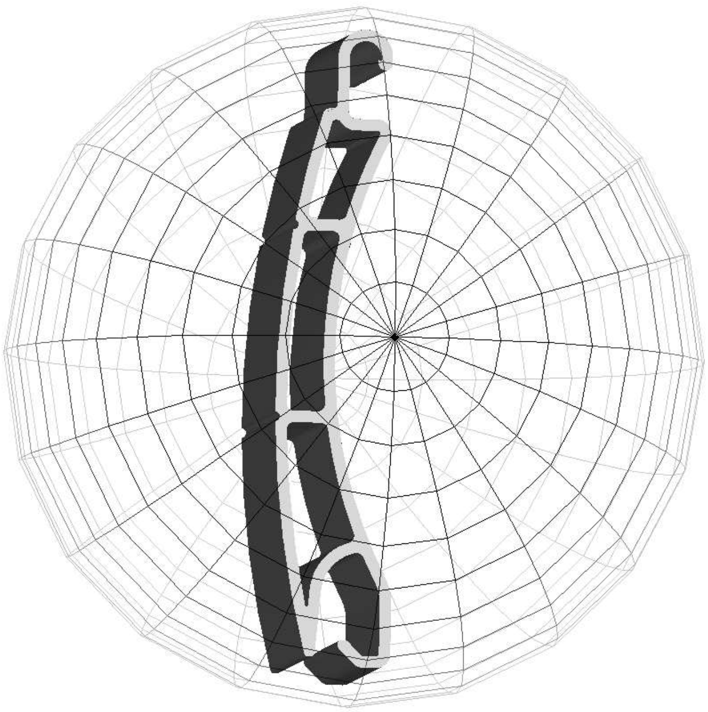

Abbildung 3.26: `rolladen.sat` - kleinste einschließende Kugel

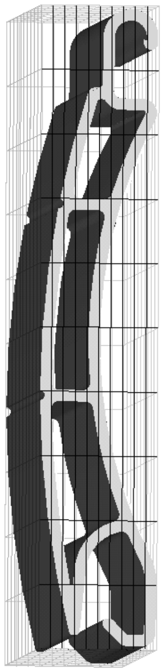

Abbildung 3.27: `rolladen.sat` - kleinster einschließender Quader

Kapitel 4

Ausblick

In dieser Arbeit haben wir einen Algorithmus zur Berechnung von optimalen Hüllkörpern für Objekte mit gekrümmten Oberflächen vorgestellt. Für spezielle Kurven- und Flächentypen haben wir explizite Lösungen für die elementaren Probleme angegeben. Für allgemeine NURBS-Kurven und -Flächen wurden diese Probleme als Nullstellenprobleme uni- und bivariater Polynome formuliert. Zur Bestimmung des Optimums wurden lokale und globale Optimierungsverfahren verwendet.

Unsere Implementierung der vorgestellten Algorithmen beschränkt sich auf die explizit behandelten Kurven- und Flächentypen. Eine solche Einschränkung bringt jedoch einige Probleme mit sich. Die Menge der Körper, die durch quadratische Flächen und Kurven begrenzt wird, ist unter den booleschen Operationen Schnitt, Vereinigung und Differenz nicht abgeschlossen. Beispielsweise ist der Schnitt zweier zylindrischer Flächen in der Regel keine quadratische Kurve mehr (siehe [Beh99]). Aus diesem Grund ist es wünschenswert, die Implementierung zumindest um NURBS-Kurven zu erweitern. Ein zweiter Erweiterungsschritt könnte allgemeine NURBS-Flächen umfassen.

Wir haben in dieser Arbeit stets gefordert, dass die Oberfläche der Objekte durch NURBS beschrieben werden kann. Dies ist in der Praxis jedoch nicht immer der Fall. So ist es denkbar, dass für eine Begrenzungskante e gar keine explizite Darstellung vorhanden ist. Die Kante wird stattdessen nur als Schnittmenge der beiden adjazenten Flächenstücke spezifiziert.
In einem solchen Fall lassen sich die elementaren Probleme als Optimierungsproblem mit Nebenbedingungen formulieren: Seien die beiden zur Kante e adjazenten Flächenstücke auf Flächen mit der impliziten Gleichung $f(\boldsymbol{x}) = 0$ bzw. $g(\boldsymbol{x}) = 0$ eingebettet. Der maximale Abstand eines Punktes $\boldsymbol{p} \in \mathbb{R}^3$ zu allen Punkten der Kante e ist dann durch die Lösung des folgenden Problems gegeben:

$$\begin{aligned} &\max_{\boldsymbol{x} \in \mathbb{R}^3} |\boldsymbol{p} - \boldsymbol{x}| \\ \text{mit}\ \ &f(\boldsymbol{x}) = 0 \\ \text{und}\ \ &g(\boldsymbol{x}) = 0 \end{aligned}$$

Die Ausdehnung einer solchen Kante e lässt sich ebenfalls als ein Optimierungsproblem mit Nebenbedingungen formulieren. Außerdem wäre es notwendig, die Routinen für die Enthaltenseinstests zu modifizieren.

Für die Berechnung des kleinsten einschließenden Quaders haben wir neben lokalen Optimierungsverfahren auch ein globales Optimierungsverfahren benutzt. Der verwendete Branch-and-Bound-Algorithmus hat sich jedoch aufgrund der langen Laufzeit für die Praxis als untauglich erwiesen. Die Ursache dafür liegt in der starken Überschätzung bei der Intervallauswertung von Funktion (3.11). Es gibt jedoch auch globale Optimierungsverfahren, die ohne Intervallarithmetik auskommen. So könnte man zum Beispiel *Simulated Annealing* [PTVF94] verwenden. Dieses Verfahren ermöglicht die Bestimmung eines "guten" lokalen Minimums für nicht-konvexe Funktionen. Es ist allerdings nicht garantiert, dass es sich hierbei wirklich um das globale Minimum handelt.

Literaturverzeichnis

[AH83] G. Alefeld, J. Herzberger: *Introduction to Interval Computations*, Academic Press, New York, 1983

[Beh99] C. Behrens: *Solid Modeling durch Boolesche Operationen*, Diplomarbeit, Technische Fakultät, Fachbereich 14 - Informatik, Universität des Saarlandes, Saarbrücken, 1999

[Cr89] J. Craig: *Introduction to Robotics, Mechanics and Control*, 2nd edition, Addison-Wesley, 1989

[DL00] E. Dyllong, W. Luther: *Distance Calculation Between a Point and a NURBS Surface*, in: Curve and Surface Design: Saint Malo 1999, pages 55-62, Vanderbilt University Press, Nashville, 2000

[KLMP98] S. Krishnan, M. Lin, D. Manocha, A. Pattekar: *Spherical Shell: A Higher Order Bounding Volume for Fast Proximity Queries*, in: Proceedings of Workshop on Algorithmic Foundations of Robotics, 1998

[Kul761] U. Kulisch: *Grundlagen des numerischen Rechnens: Mathematische Begründung der Rechnerarithmetik*, Reihe Informatik, Band 19, Bibliographisches Institut, Mannheim, 1976

[Lec91] A. P. Leclerc: *Newton's Method*, Technical Report, OSU-CISRC-01/91-TR-04, Ohio State University, Columbus, 1991

[Len00] C. Lennerz: *Impulsbasierte Dynamiksimulation starrer Körper unter Verwendung von Hüllkörperhierarchien*, Diplomarbeit, Technische Fakultät, Fachbereich 14 - Informatik, Universität des Saarlandes, Saarbrücken, 2000

[Moo69] R. E. Moore: *Intervallanalyse*, R. Oldenbourg Verlag, Mannheim, 1969

[Moo79] R. E. Moore: *Methods and Applications of Interval Analysis*, SIAM Studies in Applied Mathematics, Philadelphia, 1979

[Neu90] A. Neumaier: *Interval Methods for Systems of Equations*, Cambridge University Press, Cambridge, 1990

[ORo85] J. O'Rourke: *Finding Minimal Enclosing Boxes*, in: International Journal of Computer and Information Sciences, Volume 14, pages 183-199, 1985

[PT97] L. Piegl, W. Tiller: *The NURBS Book*, 2nd edition, Springer, 1997

[PTVF94] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery: *Numerical Recipes in C: the Art of Scientific Computing*, 2nd edition, Cambridge University Press, 1994

[Rat92] D. Ratz: *Automatische Ergebnisverifikation bei globalen Optimierungsproblemen*, Dissertation, Fakultät für Mathematik, Universität Karlsruhe (TH), 1992

[Sch94] E. Schömer: *Interaktive Montageplanung mit Kollisionserkennung*, Dissertation, Technische Fakultät, Universität des Saarlandes, Saarbrücken, 1994

[Spa] Spatial Corp.: *ACIS SAT File Format*, `http://www.spatial.com`

[War99] T. Warken: *Berechnung von Kontaktkräften für eine zwangsbasierte Dynamiksimulation*, Diplomarbeit, Technische Fakultät, Fachbereich 14 - Informatik, Universität des Saarlandes, Saarbrücken, 1999

[Wel91] E. Welzl: *Smallest Enclosing Disks (Balls and Ellipsoids)*, in: Lecture Notes in Computer Science, pages 359-370, 1991

Abbildungsverzeichnis