Lageoptimierung nach dem Maximin-Kriterium

Diplomarbeit
vorgelegt von
Frank Follert

nach einem Thema von Prof. Dr. Günter Hotz
im Fachbereich 14, Informatik, der Universität des Saarlandes

Hiermit versichere ich an Eides Statt, daß ich diese Arbeit nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.

Saarbrücken, im Juli 1994

Inhaltsverzeichnis

Einleitung

Als Probleme der *Lageoptimierung (location problems)* bezeichnet man geometrische Optimierungsaufgaben, die nach der bestmöglichen Position eines geometrischen Objekts, des *Zielobjekts*, in einer vorgegebenen festen Objektumgebung, der Menge der *Weltobjekte*, fragen. Zur Bewertung der zulässigen Positionen des zu plazierenden Zielobjekts dient dabei ein Gütemaß, das in geeigneter Weise den Abstand des Zielobjektes zu der Menge der Weltobjekte modelliert.

Problemstellungen dieser Art besitzen eine reiche Geschichte. Bereits im Jahr 1857 formulierte Sylvester die Aufgabe, den Mittelpunkt des kleinsten, eine gegebene Menge von Punkten der Ebene umschließenden Kreises zu bestimmen. Diese Aufgabe kann man als Lageoptimierungsproblem für den Kreismittelpunkt interpretieren: finde die Position, in der der Mittelpunkt den kleinstmöglichen Abstand zum entferntesten der vorgegebenen Punkte aufweist.

Das Interesse an Fragen dieser Art erwuchs zum einen aus der Eleganz der Problemstellungen und der ihnen innewohnenden komplexen Struktur, war zum anderen aber auch durch die Fülle der möglichen Anwendungen motiviert. So sind Wirtschaftsunternehmen bestrebt, ihren Entscheidungen eine wissenschaftliche Basis zu geben und beispielsweise Standorte von Niederlassungen nach mathematischen Kriterien zu bewerten. Will man etwa in einer Stadt eine neue Niederlassung gründen, erscheint es wünschenswert, sie so zu plazieren, daß ihre Entfernung zum nächstgelegenen Konkurrenzunternehmen maximiert wird, um so ein größtmögliches Einzugsgebiet zu schaffen. Ein mathematisches Modell dieser Situation ist die Aufgabe, den Mittelpunkt des größtmöglichen, in eine Punktmenge einbeschriebenen Kreises zu bestimmen. Plant man weiterhin, ein zentrales Lager für alle Niederlassungen einzurichten, das kürzestmögliche Lieferzeit zu der entferntesten Niederlassung erzielen soll, wird man zum obengenannten Sylvesterschen Problem geführt. Solche und ähnliche Fragestellungen werden in einer Teildisziplin der Unternehmensforschung (*Operations Research*), der Standortoptimierung (*facility location*), diskutiert.

Doch sind wirtschaftswissenschaftliche Fragestellungen nicht die einzigen Quellen von Problemen der Lageoptimierung. Ausgangspunkt unserer Beschäftigung mit diesem Themenkreis war eine durch neue Operationstechniken der Neurochirurgie motivierte geometrische Optimierungsaufgabe. Diese und weitere mögliche Anwendungen, z.B. aus der mathematischen Approximationstheorie und der Robotik, werden im Verlauf der Arbeit besprochen.

Offensichtlich ist die Komplexität von Problemen der Lageoptimierung sowie die Struktur von Lösungsstrategien in entscheidender Weise von der Wahl des zu optimierenden Gütekriteriums abhängig. Die eingangs genannten Beispiele des größten leeren und des kleinsten umschließenden Kreises illustrieren zwei zueinander duale Kriterien: das Minimax-Kriterium, das minimalen Abstand des Zielobjekts zum entferntesten Objekt der Umgebung fordert, sowie das Maximin-Kriterium, welches maximale Distanz zum nächstgelegenen Weltobjekt zum Ziel hat. Die Wahl eines dieser Gütekriterien ist dann angebracht, wenn der Abstand zu den Umgebungsobjekten in Extremlage die Qualität der Position des Zielobjekts bestimmt.

Angeregt durch die genannten Beispielanwendungen interessieren wir uns insbesondere für Situationen, in denen die Weltobjekte als Hindernisse interpretiert werden, zwischen denen ein Objekt so eingefügt werden soll, daß sein Freiraum maximal wird, sowie für solche, in denen das Zielobjekt eine Störungsquelle repräsentiert, deren Wechselwirkung mit den Objekten der Umgebung minimiert werden soll. In beiden Fällen erscheint es sinnvoll, das Maximin-Kriterium als Optimalitätsbedingung zu wählen. Ziel dieser Arbeit wird daher sein, die Komplexität von Lageoptimierungsproblemen nach dem Maximin-Kriterium zu studieren und effiziente Algorithmen für diese und verwandte Probleme zu entwickeln. Darüber hinaus werden wir an geeigneter Stelle Optimierungsaufgaben nach dem dualen Minimax-Kriterium diskutieren. Wir werden bei unseren Betrachtungen weitgehend von den Anwendungen abstrahieren, jedoch den meisten Kapiteln einen einführenden Abschnitt mit Anwendungsbeispielen voranstellen.

Algorithmen zur Lageoptimierung lassen sich der *algorithmischen Geometrie*, einem Teilbereich der Theoretischen Informatik, zuordnen. Das Ziel der algorithmischen Geometrie ist es, effiziente Techniken zur Bestimmung, Darstellung und Manipulation geometrischer Objekte zu entwickeln und zu analysieren. Dabei stehen, wie in der Theorie der Algorithmen üblich, Aussagen über die asymptotische Komplexität im Vordergrund. Als Ergebnis intensiver Forschungen der letzten Jahre verfügt das Gebiet über einen umfangreichen spezifischen Satz von Techniken und Paradigmen, von denen wir einige in dieser Arbeit vorstellen werden. Einen ausführlichen Überblick geben die Monographien von Mehlhorn [Meh84], Preparata und Shamos [PS85], sowie Edelsbrunner [Ed87]. Motiviert waren diese Untersuchungen durch eine Vielzahl von Problemen aus so unterschiedlichen Anwendungsgebieten wie Robotik, VLSI-Entwurf, CAD, Computergraphik, Operations Research etc., denen gemeinsam ist, daß sie durch geometrische Modelle adäquat repräsentiert werden können. Obwohl diese Probleme auch einer rein algebraischen Behandlung zugänglich sind, wählt man meist die geometrische Sichtweise, da diese den Vorteil der unmittelbaren Anschaulichkeit bietet und darüber hinaus die zugrundeliegenden geometrischen Strukturen häufig einen geeigneten Ansatzpunkt zur Realisierung eines effizienten Algorithmus bilden. Auch wir werden uns in dieser Arbeit von der geometrischen Anschauung leiten lassen, uns jedoch zum Nachweis der

Korrektheit der Verfahren meist einer algebraischen Formulierung bedienen.

Die Arbeit gliedert sich wie folgt:

In *Kapitel 1* führen wir grundlegende Begriffe ein und legen ein Berechnungsmodell für die vorgestellten Algorithmen fest. In *Kapitel 2* formulieren wir die allgemeine Problemstellung und zeigen Verbindungen zu der MaxMin-Aufgabe, einem Optimierungsproblem über Funktionenscharen, auf. Weiterhin diskutieren wir erste Lösungsansätze und grenzen die untersuchte Problematik von verwandten Fragestellungen ab. In *Kapitel 3* entwickeln wir effiziente Algorithmen zur Lageoptimierung von Punkten auf der reellen Achse und Ursprungsgeraden in der Ebene bei gewichteten Abstandsmaßen und zeigen ihre Optimalität. Die Algorithmen basieren auf einer Transformation der Problemstellung in eine MaxMin-Aufgabe von Funktionen in einer Variablen. Ausgehend von der Theorie der Davenport-Schinzel Sequenzen leiten wir MaxMin-Algorithmen für einige Funktionenklassen her und stellen diese erzielten unteren Schranken gegenüber. In *Kapitel 4* beschreiben wir einen Algorithmus zur optimalen Plazierung eines Punktes inmitten von gewichteten Punkten der Ebene, der geringere Zeitkomplexität als bisher bekannte Verfahren aufweist. Der Algorithmus folgt einer Strategie, die Methoden der algorithmischen Geometrie (Minimierungsdiagramme) mit Techniken der nichtlinearen Optimierung (Kuhn-Tucker-Bedingungen) verknüpft. In *Kapitel 5* übertragen wir diese Strategie auf Algorithmen zur Lageoptimierung von Zielobjekten im Raum, deren Lage durch einen Punkt der Einheitskugel bestimmt werden kann. Auf diese Weise können wir die bezüglich des Maximin-Kriteriums bestmögliche Lage eines Punktes auf der Einheitskugel und einer Ursprungsgeraden, eines Ursprungsstrahls sowie einer Ursprungsebene im Raum effizient berechnen. Darüber hinaus erläutern wir, wie man in analoger Weise die hierzu dualen Minimax-Probleme lösen kann. Wir stützen uns hierbei auf Resultate zur Komplexität und Berechnung von Minimierungsdiagrammen aus [SS90] und [Sh93]. Die konkreten in diesem Kapitel untersuchten Fragestellungen wurden jedoch unseres Wissens bislang in der Literatur nicht diskutiert. *Kapitel 6* gibt eine kurze Zusammenfassung und enthält abschließende Bemerkungen.

An dieser Stelle möchte ich mich bei Herrn Prof. Dr. Günter Hotz für die Vergabe des interessanten Themas und Herrn Dr. Elmar Schömer für die intensive Betreuung und wertvolle Anregungen bei der Anfertigung der Arbeit bedanken.

Des weiteren danke ich allen, die zum Gelingen der Arbeit beigetragen haben, insbesondere meinen Kommilitonen Jörg Sauer und Christoph Schmitz für fruchtbare Diskussionen und das Korrekturlesen der Arbeit.

Mein größter Dank gilt jedoch meiner Familie, ohne deren Unterstützung und Verständnis ich mich nicht in dieser Weise meinem Studium hätte widmen können.

Kapitel 1

Grundlagen

1.1 Mathematische Grundlagen

In diesem Abschnitt werden wir einige Begriffe und Notationen einführen, die zur exakten Formulierung und Behandlung von Problemen der algorithmischen Geometrie unerlässlich sind. Da die meisten der Begriffe dem Leser vertraut sein werden, verzichten wir auf ausführliche formale Definitionen und geben der Darstellung aufzählenden Charakter. Wir folgen hierbei im wesentlichen [Ed87].

Geometrische Objekte lassen sich als Punktmengen im d-dimensionalen Raum $\mathbb{R}^d$ darstellen. Wir nehmen hierzu ein festes kartesisches Bezugskoordinatensystem an. Einen Punkt $x = (x_1, x_2 \ldots, x_d)$ des $\mathbb{R}^d$ kann man dann mit dem Vektor identifizieren, der im Ursprung beginnt und im Punkt x endet: dem Ortsvektor von x. Wir erklären zwei fundamentale Operationen auf Vektoren. Für zwei Vektoren u und v des $\mathbb{R}^d$ heißt die Zahl

$$<u, v> = \sum_{i=1}^{d} u_i v_i$$

das *Skalarprodukt* von u und v. Sind u, v Vektoren des $\mathbb{R}^3$ und e_x, e_y, e_z die Einheitsvektoren, so heißt der Vektor

$$w = u \times v = \begin{vmatrix} e_x & e_y & e_z \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{vmatrix}$$

das *Vektorprodukt* von u und v.

Wir charakterisieren nun einige einfache Beziehungen zwischen Punkten.

Sei $P = \{p_0, p_1, \ldots, p_k\}$ eine Menge von endlich vielen Punkten des $\mathbb{R}^d$. Ein Punkt $x \in \mathbb{R}^d$ heißt *Linearkombination* von P, falls es reelle λ_i $(0 \leq i \leq k)$ gibt, so daß

$$x = \sum_{i=0}^{k} \lambda_i p_i.$$

Gilt weiterhin

$$\sum_{i=0}^{k} \lambda_i = 1,$$

so bezeichnet man x auch als *affine Kombination* von P. Falls schließlich auch

$$0 \leq \lambda_i \leq 1 \qquad \text{für alle } \ 0 \leq i \leq k$$

gilt, heißt x *Konvexkombination* von P.

P heißt *linear* bzw. *affin unabhängig*, falls es keinen Punkt $p_i \in P$ gibt, der lineare oder affine Kombination von $P \setminus \{p_i\}$ ist.

Die Menge der linearen, affinen oder konvexen Kombinationen von P nennt man *lineare Hülle* [kurz $\text{lin}(P)$], *affine Hülle* [$\text{aff}(P)$] oder *konvexe Hülle* [$\text{conv}(P)$] von P. Eine Punktmenge S, die lineare (affine, konvexe) Hülle einer endlichen Punktmenge P ist, heißt die von P erzeugte lineare (affine, konvexe) Menge. Auf diese Weise lassen sich auch einfache unendliche Punktmengen endlich beschreiben.

Die linearen Mengen sind gerade die Vektorunterräume des $\mathbb{R}^d$. Ein *k-dimensionaler affiner Teilraum (k–flat)* ($0 \leq k \leq d$) ist definiert als affine Hülle einer Menge von $k+1$ affin unabhängigen Punkten. Eine Menge von k Punkten $\{p_1, p_2, \ldots, p_k\}$ des $\mathbb{R}^d$ ist genau dann affin unabhängig, wenn die Menge der $k-1$ Vektoren $\{p_2-p_1, p_3-p_1, \ldots, p_k-p_1\}$ linear unabhängig ist. Eine affiner Teilraum ist somit ein um einen konstanten Vektor verschobener linearer Teilraum (Vektorunterraum) des $\mathbb{R}^d$. Er nimmt im $\mathbb{R}^3$ für $k=0$ die Gestalt eines Punktes, für $k=1$ die einer Geraden und für $k=2$ die einer Ebene an.

Man nennt einen affinen Teilraum der Dimension $d-1$ auch *Hyperebene*. Eine Hyperebene H laßt sich alternativ mit einem Vektor $v \neq 0$ (dem *Normalenvektor*) und einer reellen Zahl a beschreiben: H ist die Menge aller Punkte x mit

$$\langle x, v \rangle = a$$

Auf dem Vektorraum $\mathbb{R}^d$ läßt sich mit Hilfe des Skalarprodukts eine Norm $|x| = \sqrt{\langle x, x \rangle}$, die *euklidische Norm*, definieren.

Der daraus abgeleitete *euklidische Abstand* zweier Punkte p und q des $\mathbb{R}^d$ ergibt sich zu

$$\delta(p,q) = |p-q| = \sqrt{\sum_{i=1}^{d}(p_i-q_i)^2}.$$

Der $\mathbb{R}^d$ bildet zusammen mit dem euklidischen Abstand δ als Metrik einen *metrischen Raum*. Der euklidische Abstand zweier Punktmengen P und Q ist gegeben durch

$$\delta(P,Q) = \inf_{p \in P, q \in Q} \delta(p,q)$$

So hat zum Beispiel der Punkt p von der Hyperebene $H : (<x, v> = a)$ den Abstand

$$\delta(p, H) = \left| \frac{1}{|v|}(<p, v> - a) \right|$$

Für einen Punkt $p \in \mathbb{R}^d$ und eine reelle Zahl $\rho > 0$ heißt die Menge $\{x \in \mathbb{R}^d \mid \delta(x, p) < \rho\}$ der *offene Ball* mit Mittelpunkt p und Radius ρ. Man nennt eine Menge $S \subset \mathbb{R}^d$ *beschränkt*, falls es einen offenen Ball gibt, der S umfaßt, ansonsten heißt sie *unbeschränkt*. Eine Menge S heißt *offen*, falls sie für jeden ihrer Punkte p einen offenen Ball mit Mittelpunkt p enthält, sie heißt *abgeschlossen*, wenn ihr Komplement $\mathbb{R}^d \setminus S$ offen ist. Wir nennen die größte offene Teilmenge von S ihren *offenen Kern* $int(S)$, die kleinste abgeschlossene Obermenge von S ihre *abgeschlossene Hülle* $cl(S)$ und $\partial S = cl(S) \setminus int(S)$ den *Rand* von S. Man bezeichnet eine Menge S als *kompakt*, wenn sie abgeschlossen und beschränkt ist. Wir schreiben $\mathrm{CH}(S)$ für den Rand $\partial(conv(S))$ der konvexen Hülle einer Punktmenge S.

Zwei Punkte p und q einer Menge $S \subset \mathbb{R}^d$ heißen *zusammenhängend in S*, falls es für jede positive reelle Zahl ϵ eine Punktfolge $p = r_0, r_1, \ldots, r_n = q$ $(r_i \in S,\ 0 \leq i \leq n)$ gibt, so daß $\delta(r_i, r_{i+1}) < \epsilon$ für alle $0 \leq i \leq n-1$. Die Menge S heißt *zusammenhängend*, falls je zwei Punkte aus S in S zusammenhängend sind. Die *Zusammenhangskomponenten* (kurz ZHK) von S sind die maximal (bezüglich Mengeninklusion) zusammenhängenden Teilmengen von S. S heißt *einfach zusammenhängend*, falls jede ZHK ihres Komplements $\mathbb{R}^d \setminus S$ unbeschränkt ist.

1.2 Berechnungsmodell

Um ein vernünftiges Maß für die Komplexität der in den folgenden Kapiteln entwickelten Algorithmen zu erhalten, müssen wir ein dem Problembereich angemessenes *Berechnungsmodell* definieren. Ein solches Berechnungsmodell legt die erlaubten Elementaroperationen der Algorithmen sowie die Kosten dieser Operationen fest. In der algorithmischen Geometrie wählt man als Berechnungsmodell meist die *real random-access-machine (real RAM)* [PS85]. Die real RAM kann in jeder Speicherzelle eine reelle Zahl aufnehmen und führt die arithmetische Operationen $(+, *, -, /, \sqrt[k]{})$ und Vergleichsoperationen auf reellen Zahlen sowie indirekte Adressierung zu Einheitskosten durch.

Zur Herleitung von *unteren Schranken* für Berechnungsprobleme der algorithmischen Geometrie betrachtet man meist zunächst ein verwandtes Entscheidungsproblem. Ein angemessenes Berechnungsmodell für Entscheidungsprobleme ist der *algebraischen Entscheidungsbaum* [Be83]. Der algebraische Entscheidungsbaum hat zwei wesentliche Vorzüge: zum einen bietet er ein hinreichend realistisches Modell für auf realen Rechnern ablaufende Programme (mit Ausnahme der indirekten Adressierung), zum anderen ist er der mathematischen Analyse weit genug zugänglich, um nichttriviale Aussagen über die Komplexität von Problemen in diesem Modell zu erlauben.

Es folgt die formale Festlegung des Modells.

Definition 1 *Ein* algebraischer Entscheidungsbaum T *für eine Eingabe* $s = (s_1, \ldots, s_n)$ *ist ein Baum von Grad* ≤ 2 *zusammen mit einer Funktion, die*

a.) jedem Blatt eine Ausgabe 'Ja' oder 'Nein' zuweist,

b.) jedem Knoten v *mit einem Kind* (einfacher Knoten) *eine Variable* $Y(v)$ *und einen Befehl der Form*

$$\begin{aligned} Y(v) &= Y(v_1)\, op\, Y(v_2) \quad \textit{oder} \\ Y(v) &= c\; op\; Y(v_1) \quad \textit{oder} \\ Y(v) &= \sqrt[k]{Y(v_1)} \qquad (k \leq d\ (d\ \textit{fest})) \end{aligned}$$

zuweist, wobei $op \in \{+, -, *, /\}$*,* v_i *ein einfacher Vorgänger von* v *in* T *oder* $Y(v_i) \in \{s_1, \ldots, s_n\}$ *und* $c \in \mathbb{R}$ *ist,*

c.) jedem Knoten mit zwei Kindern (Verzweigungsknoten) *einen Vergleich der Form* $Y(v_1)$ *rel* 0 *zuweist, wobei* v_1 *ein einfacher Vorgänger von* v *ist oder* $Y(v_1) \in \{s_1, \ldots, s_n\}$ *und* $rel \in \{\geq, >, =\}$.

Die Rechnung eines algebraischen Entscheidungsbaums für eine Eingabe $s \in \mathbb{R}^n$ geschieht nach folgendem Schema :

Das Programm durchläuft, beginnend mit der Wurzel, einen Pfad $P(s)$ durch den Baum. An jedem einfachen Knoten des Pfades wird die zugehörige arithmetische Operation durchgeführt, an jedem Verzweigungsknoten wird gemäß des Ergebnisses des Vergleichs verzweigt. Wird schließlich ein Blatt erreicht, so antwortet das Programm mit der Marke ('Ja' oder 'Nein') des Blattes.

Ein Programmlauf entspricht also einem Pfad im algebraischen Entscheidungsbaum.

Man sagt, ein algebraischer Entscheidungsbaum entscheidet das **Element-von-Problem** für eine Menge $W \subseteq \mathbb{R}^n$, wenn er ihre charakteristische Funktion

$$f : W \to \{\text{'Ja', 'Nein'}\}; \qquad f(x) = \text{'Ja'} \iff x \in W$$

berechnet.

Löst ein Programm ein Entscheidungsproblem P über Eingaben aus $\mathbb{R}^n$, so entscheidet der zugehörige algebraische Entscheidungsbaum das Element-von-Problem der Menge der akzeptierten Eingaben.

Die Komplexität des Element-von-Problems einer Menge $W \subseteq \mathbb{R}^n$ ist durch die minimale Tiefe eines algebraischen Entscheidungsbaums gegeben, der dieses Problem entscheidet. Welche Mindesttiefe ein algebraischer Entscheidungsbaum haben muß, um dies leisten zu können, hängt entscheidend von der topologischen Struktur der Menge W ab. Auf dieser Tatsache beruht das fundamentale Ergebnis von Ben-Or [Be83], das Grundlage für alle folgenden Sätze über untere Schranken sein wird.

Notation: Wir bezeichnen mit $\#V$ die Anzahl der Zusammenhangskomponenten der Menge V.

Satz 1 (Ben-Or) *Sei T ein algebraischer Entscheidungsbaum, der das Element-von-Problem für $W \subseteq \mathbb{R}^n$ entscheidet. Bezeichnet h die Tiefe von T, dann gilt*

$$h \geq \frac{\log \max(\#W, \#(\mathbb{R}^n - W)) - n \log 3}{\log 6}$$

Zerfällt also die Menge W der akzeptierten Eingaben zu einem Entscheidungsproblem P beispielsweise in $n!$ Zusammenhangskomponenten, so ist

$$\frac{\log n! - n \log 3}{\log 6} = \Omega(n \log n)$$

eine untere Schranke für die Zeitkomplexität von P.

Kapitel 2

Allgemeine Problemstellung

2.1 Definitionen

Nachdem wir im vorangegangenen Kapitel die Basis für unsere Betrachtungen gelegt haben, wenden wir uns nun der formalen Beschreibung der Problemstellung zu. Wie bereits in der Einleitung ausgeführt, untersuchen wir in dieser Arbeit Probleme der Lageoptimierung. Allgemein gefaßt besteht die Aufgabe eines Algorithmus zur Lageoptimierung darin, ein geometrisches Zielobjekt derart in eine vorgegebene Objektumgebung, die Menge der Weltobjekte im *Objektraum* einzufügen, daß seine Lage in Bezug auf ein Gütekriterium optimal wird. Eine große Anzahl von Problemstellungen läßt sich unter dieses Schema subsumieren; natürlicherweise klassifiziert man sie entsprechend der erlaubten Gestalt von Welt- und Zielobjekten, sowie gemäß der Wahl des Optimalitätskriteriums.

Unsere Untersuchungen konzentrieren sich auf den $\mathbb{R}^d$ als Objektraum; als Weltobjekte lassen wir Punkte des $\mathbb{R}^d$ zu, als Zielobjekte werden wir $k-$dimensionale affine Teilräume des $\mathbb{R}^d$ betrachten.

Die Distanz des Zielobjekts zu einem der Weltpunkte wird durch ein Abstandsmaß dist beschrieben. Die Position des Zielobjekts im Objektraum wird genau dann optimal, wenn sie bezüglich dist ein Maximin-Kriterium erfüllt: die kleinste auftretende Distanz zu einem der Weltpunkte ist maximal unter allen zulässigen Positionen des Zielobjekts.

Als Abstandsmaß dist wählen wir meist den euklidischen Abstand δ. Es zeigt sich jedoch, daß der entwickelte Ansatz auch auf andere allgemeinere, beispielsweise gewichtete Abstandsfunktionen übertragbar ist. Wir nennen diese Funktionen auch dann Abstandsfunktionen, wenn sie die Metrikaxiome Nichtnegativität, Symmetrie und Gültigkeit der Dreiecksungleichung nicht erfüllen. Im wesentlichen fordern wir nur Stetigkeit auf dem Definitionsbereich.

Formal gefaßt lautet die

ALLGEMEINE PROBLEMSTELLUNG

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\}$, $p_i = (x_{i1}, \ldots, x_{id})$ von n Punkten des $\mathbb{R}^d$, denen positive reelle Gewichte $W = \{w_1, \ldots, w_n\}$ zugeordnet sind.

Gesucht: Ein bezüglich S und W optimaler k-dimensionaler affiner Teilraum F^* des $\mathbb{R}^d$ ($k < d$ fest). Optimal heißt hierbei, daß F^* das Kriterium

$$\text{Maximin:} \qquad \max_{F} \min_{1 \leq i \leq n} w_i \mathrm{dist}(p_i, F)$$

erfüllt, d. h. F^* maximiert den minimalen gewichteten Abstand dist zu S.

Fordert man zusätzlich, daß F einen vorgegebenen Punkt $a \in \mathbb{R}^d$, den *Anker*, enthält, spricht man von einem *verankerten Problem*.

Wir werden uns vornehmlich mit der optimalen Plazierung von affinen Teilräumen im zwei- und dreidimensionalen euklidischen Raum befassen.

Die Beschreibung eines $k-$dimensionalen affinen Teilraums F kann durch Angabe einer $(k+1)$-elementigen affin unabhängigen Punktmenge P geschehen, die F erzeugt: $F = \mathrm{aff}(P)$. Bei verankerten Problemen enthalte diese Menge den Anker a. Jeder dieser Teilräume des $\mathbb{R}^d$ kann also durch einen Punkt im $\mathbb{R}^{d(k+1)}$ dargestellt werden. Diese Koordinaten sind jedoch meist nicht unabhängig; durch geschickte Wahl der zur Lagefestlegung des Zielobjekts dienenden Parameter kann deren Anzahl häufig reduziert werden. Die minimale zur Beschreibung der Lage des Zielobjekts notwendige Anzahl von reellen Parametern bezeichnen wir als die Anzahl der *Freiheitsgrade* des Zielobjekts. Ein $k-$Teilraum F mit f Freiheitsgraden ist nun als Punkt p_F des *Konfigurationsraums* $K \subset \mathbb{R}^f$ darstellbar. Ist die so erhaltene Menge K der möglichen Lagen von F nicht kompakt, schränken wir sie auf einen kompakten zulässigen Bereich $M \subset K$ ein, um eine endliche Lösung des Maximierungsproblems zu garantieren. Das modifizierte Problem lautet dann

$$\text{Maximin:} \qquad \max_{F:\, p_F \in M} \min_{1 \leq i \leq n} w_i \mathrm{dist}(p_i, F)$$

Wir betrachten die Lageparameter p_{F^*} des optimalen Teilraums F^*, d. h. seinen Koordinatenvektor im Konfigurationsraum, als Lösung des Problems. Den zu dieser Lage gehörigen optimalen Abstand kann man dann in linearer Zeit bestimmen, falls - wie wir stets voraussetzen werden - alle Abstandsfunktionen in konstanter Zeit ausgewertet werden können. Gibt es mehrere optimale Lösungen, erwarten wir eine beliebige dieser Lösungen.

Neben dieser Fragestellung untersuchen wir das Optimierungsproblem nach dem dualen Minimax-Kriterium

$$\text{Minimax:} \qquad \min_{F:\, p_F \in M} \max_{1 \leq i \leq n} w_i \mathrm{dist}(p_i, F)$$

d. h. ein optimaler Teilraum F^* minimiert den maximalen gewichteten Abstand zur Punktmenge S.

2.2 Funktionale Problemfassung

Bislang haben wir die Fragestellung als geometrisches Optimierungsproblem formuliert. Abstrahiert man aber von der geometrischen Natur des Problems, gelangt man zu einer Fassung der Problemstellung, die eine Einordnung in einen allgemeineren Rahmen erlaubt. Hierzu ordnen wir jedem Weltpunkt $p_i \in S$ $(1 \leq i \leq n)$ die *Distanzfunktion*

$$f_i : M \to \mathbb{R}, \qquad f_i(p_F) := w_i \mathrm{dist}(p_i, F)$$

zu, die den Konfigurationsraumkoordinaten p_F des affinen Teilraums F dessen Abstand zum Weltpunkt p_i zuordnet. Unser Ausgangsproblem läßt sich nun als Optimierungsproblem über der Funktionenschar $\mathcal{F} = \{f_1, \ldots, f_n\}$ beschreiben.

Dazu benötigen wir die

Definition 2 *Sei $\mathcal{F} = \{f_1, \ldots, f_n\}$ eine Schar stetiger reellwertiger Funktionen über einem gemeinsamen Definitionsbereich $Def \subset \mathbb{R}^d$. Dann heißt das punktweise Minimum der Funktionenschar*

$$f^u : \ Def \to \mathbb{R}; \ f^u(x) := \min_{1 \leq i \leq n} f_i(x)$$

die untere Einhüllende *von $\mathcal{F}$ (kurz $f^u[\mathcal{F}]$), und das punktweise Maximum der Funktionenschar*

$$f^o : \ Def \to \mathbb{R}; \ f^o(x) := \max_{1 \leq i \leq n} f_i(x)$$

die obere Einhüllende *von $\mathcal{F}$ (kurz $f^o[\mathcal{F}]$).*

Bezeichnet $\mathcal{F}$ die Schar der Distanzfunktionen, wird das Ausgangsproblem zu einer Teilklasse des allgemeineren

PROBLEM : [MaxMin einer Funktionenschar]

Gegeben: Eine endliche Schar $\mathcal{F} = \{f_1, \ldots, f_n\}$ stetiger, reellwertiger Funktionen über einem gemeinsamen Definitionsbereich $Def \subseteq \mathbb{R}^d$.

Gesucht: Die Abszisse des globalen Maximums der unteren Einhüllenden, falls ein solches existiert.

In diesem Fall läßt sich das duale Minimax-Problem auch als

PROBLEM : [MinMax einer Funktionenschar]

Gegeben: Eine endliche Schar $\mathcal{F} = \{f_1, \ldots, f_n\}$ stetiger, reellwertiger Funktionen über einem gemeinsamen Definitionsbereich $Def \subseteq \mathbb{R}^d$.

Gesucht: Die Abszisse des globalen Minimums der oberen Einhüllenden, falls ein solches existiert.

formulieren.

Die Einhüllenden $f^u[\mathcal{F}]$ und $f^o[\mathcal{F}]$ sind aufgrund der Stetigkeit der Funktionen aus $\mathcal{F}$ ebenfalls stetig. Ein klassisches Resultat von Weierstraß besagt nun

Satz 2 (Weierstraß) *Eine stetige Funktion nimmt über einer kompakten Menge ihre Extrema an.*

Ist der gemeinsame Definitionsbereich *Def* der Funktionenschar kompakt wie in der allgemeinen Problemstellung gefordert, haben beide Probleme eine endliche Lösung.

Kennt man nun effiziente Algorithmen, die das MaxMin- (MinMax-)Problem für eine, die Schar der Abstandsfunktionen umfassende Klasse von Funktionen lösen, kann man diese unmittelbar zur Lösung der entsprechenden Positionierungsprobleme nutzen. Dies muß jedoch nicht der günstigste Weg sein. Wir werden insbesondere am Beispiel einiger Minimax-Probleme sehen, daß in manchen Fällen durch Ausnutzen geometrischer Eigenschaften effizientere Verfahren gewonnen werden können.

2.3 Numerische vs. kombinatorische Behandlung

Alle in dieser Arbeit vorgestellten Instanzen der allgemeinen Problemstellung sind einer kombinatorischen Behandlung zugänglich, d. h. man kann einen real RAM-Algorithmus angeben, der nach endlicher Zeit mit der exakten Lösung terminiert. Dessen ungeachtet wurde eine Reihe von Algorithmen entwickelt, um solche und ähnliche Probleme auf numerischem Wege approximativ zu lösen. Der Vorteil numerischer Verfahren liegt meist in der vergleichsweise einfachen Implementierbarkeit solcher Strategien und der Tatsache, daß dennoch hinreichend genaue Lösungen erzielt werden. Bei dem gegebenen Problemtyp treten jedoch Komplikationen auf, welche die Verwirklichung dieses Ansatzes erschweren.

Klassische Abstiegsverfahren versagen, da die zu optimierende Funktion, die untere (obere) Einhüllende der Schar der Distanzfunktionen, nicht überall differenzierbar ist, auch wenn dies für alle Funktionen der Schar zutrifft: an den Stellen, an welchen sich zwei oder mehr Distanzfunktionen auf dem Graphen der Einhüllenden schneiden, ist deren Gradient im allgemeinen nicht erklärt.

Selbst wenn der Gradient in jedem Iterationsschritt des Abstiegsverfahrens existiert, kann Konvergenz gegen ein lokales Optimum nicht garantiert werden. Darüber hinaus liegen die lokalen Optima häufig gerade in den "Knickstellen"des Graphen von f^u, so daß klassische Abbruchbedingungen der Form "$|\nabla f^u(x_k)|$ hinreichend klein"keinen Sinn machen [Lem89].

Diesen Komplikationen kann man auf zweierlei Weise begegnen [Lem86], [Lem89]:

- Ersetzen der Funktion f^u durch eine glatte Approximation.
- Anwenden spezieller Techniken der Theorie der nichtglatten Optimierung, wie z. B. Subgradientenverfahren.

Alle diese Methoden fallen unter das Schema der iterativen Abstiegsverfahren und können als solche nur Konvergenz gegen lokale Optima garantieren.

Da die obere Einhüllende konvexer Funktionen ebenfalls konvex ist, tritt bei MinMax-Problemen mit konvexen Funktionen höchstens ein lokales Minimum der oberen Einhüllenden auf, das gleichzeitig globales Minimum ist. Abstiegsverfahren führen also hier zum Erfolg. Dies gilt jedoch nicht bei MaxMin-Problemen, da dort selbst für konvexe Abstandsfunktionen zahlreiche lokale Maxima auftreten. (Dieser augenfällige Unterschied zwischen den Problemklassen MinMax und MaxMin kann als Ursache der im Fortgang der Untersuchungen gefundenen Komplexitätsdifferenzen zwischen den beiden Klassen gedeutet werden.) Ähnliches geschieht bei MinMax-Problemen, wenn Konvexität der Abstandsfunktionen nicht länger gefordert wird. Wir können also von Abstiegsverfahren im allgemeinen nur suboptimale Lösungen erwarten. Die einzig erfolgversprechenden numerischen Techniken sind damit Multistartverfahren.

2.4 Verwandte Problemstellungen

Zur Abgrenzung der betrachteten Problemklasse beschreiben wir einige verwandte Probleme in Form einer Übersicht und stellen bekannte Ergebnisse bezüglich ihrer Komplexität zusammen.

2.4.1 Tschebyscheff-Approximation

Das Tschebyscheffsche Approximationsproblem ergibt sich als Spezialfall der allgemeinen Problemstellung, wenn man den Abstand der Weltpunkte vom k-dimensionalen affinen Teilraum nicht, wie wir meist vorgehen werden, entlang des Lotes der Punkte auf den Raum mißt (orthogonaler Abstand), sondern als Absolutdifferenz der $x1$-Koordinaten bestimmt (vertikaler Abstand). Auch werden in diesem Spezialfall nur $(d-1)$-dimensionale affine Teilräume, also Hyperebenen betrachtet. Dieses Verfahren wird zur linearen Approximation von Funktionen, deren Wert nur an endlich vielen Stützstellen bekannt ist, angewandt.

Problem 1 : [Tschebyscheff-Approximation]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\}$, $p_i = (x_{i1}, \ldots, x_{id})$ von n Punkten des $\mathbb{R}^d$ mit positiven Gewichten $W = \{w_1, \ldots, w_n\}$.

Gesucht: Eine bezüglich S und W optimale Hyperebene $H^* : \sum_{j=1}^{d-1} a_j x_j + a_d = x_d$. Das Optimalitätskriterium für H^* lautet

$$\text{Tschebyscheff:} \qquad \min_H \max_{1 \leq i \leq n} w_i \left| \sum_{j=1}^{d-1} a_j x_{ij} + a_d - x_{id} \right|$$

Die Berechnung der Tschebyscheff-Approximation läßt sich auf das Lösen eines linearen Programms in Dimension $d+1$ zurückführen und kann mit den in Kapitel 3 skizzierten Methoden in linearer Zeit geschehen, wenn die Dimension d fest ist [Me84].

2.4.2 Optimierung nach dem Minisum-Kriterium

Will man nicht den maximalen Abstand zur Menge der Weltpunkte sondern vielmehr die Summe aller (gewichteten) Distanzen minimieren, erhält man ein *Minisum-Problem.* Das Optimalitätskriterium lautet hier

$$\text{Minisum:} \qquad \min_F \sum_{i=1}^{n} w_i \delta(p_i, F)$$

Dieses Problem besitzt ein weites Spektrum von Anwendungen im Bereich Operations Research. Es stellt sich zum Beispiel in der Ausprägung $d = 2,\ k = 0$ (F ist also ein Punkt) bei der Frage nach dem optimalen Standpunkt eines Lagers das verschiedene Fabriken beliefern soll. Modellieren die Weltpunkte die Fabrikstandorte, der zu plazierende Zielpunkt das Lager sowie die Gewichte w_i den Materialbedarf der Fabriken, so minimiert die Lösung des zugehörigen Minisum-Problems die Transportkosten, sofern diese proportional zu Transportweg und Gütermenge wachsen.

Das Minisum-Problem ist eine klassische Optimierungsaufgabe und wurde bereits im 17. Jahrhundert von Fermat für drei Weltpunkte und einen Zielpunkt in der Ebene gestellt und von Toricelli teilweise gelöst. In der Operations Research Literatur wurde es unter der Bezeichnung *Minisum-Weber-Problem* eingehend behandelt. Neuere Untersuchungen in [Ba86] und [Ba88] zeigen, daß im Falle $d = 2,\ k = 0$ und $n \geq 5$ Problemkonstellationen existieren, deren Lösung nicht mehr durch Radikale beschrieben werden kann. Damit ist das Problem einer Lösung im real RAM-Modell nicht zugänglich und kann nur durch numerische oder symbolische Verfahren gelöst werden.

Dies gilt jedoch nicht für Hyperebenen ($k = d - 1$) [KM93]. Für $1 \leq k < d - 1$ ist das Problem ungelöst, es erscheint aber ähnlich schwierig wie im Fall $k = 0$ (vgl. [KM93] für eine umfassende Bibliographie zu diesem Themenkreis).

2.4.3 Weitere Arbeiten

Aonuma et al. [AIIT90] plazieren ein gegebenes konvexes Polygon P so in einem anderen Polygon Q, daß der minimale Abstand zu dessen Rand maximal wird. Bezeichnen

m und n die Eckenzahl der beiden Polygone P und Q, so benötigt ihr Algorithmus Zeit $O(m^4 n \lambda_{16}(n) \log(mn)$, wenn sowohl Translation als auch Rotation des Polygons P erlaubt ist. (Eine Definition der Funktion $\lambda_s(n)$ wird in Kapitel 3 gegeben.) Ist nur Translation erlaubt, kann man die gesuchte Plazierung in Zeit $O(mn \log(mn))$ finden.

Huttenlocher et. al. [HKS91] untersuchen das Problem, die optimale Plazierung eines punktförmigen Objekts bezüglich einer Menge von Punkten der Ebene zu finden, die in verschiedene Klassen aufgeteilt ist. Betrachtet man die dem Objekt jeweils nächstgelegenen Punkte einer jeden Klasse, so besteht die Aufgabe darin, die Distanz zu dem am weitesten entfernten dieser Punkte zu minimieren. Gibt es m Klassen mit insgesamt n Punkten, so berechnet ihr Algorithmus die optimale Lösung in Zeit $O(mn \log(mn))$.

Kapitel 3

Zielobjekte mit einem Freiheitsgrad

In diesem Kapitel befassen wir uns, um mit der Problemstellung vertraut zu werden und erste Lösungsansätze zu studieren, mit Positionierungsaufgaben für Zielobjekte, die nur einen Freiheitsgrad besitzen. Zur Beschreibung des zu plazierenden $k-$dimensionalen affinen Teilraums genügt also ein reeller Parameter.

Die jeweils betrachteten Instanzen der allgemeinen Problemstellung lassen sich eindeutig durch die Angabe der Dimension d des Objektraumes, die Dimension k der zu plazierenden affinen Varietät sowie durch die Festlegung, ob es sich um ein verankertes bzw. gewichtetes Problem handelt, bestimmen.

3.1 Größte Lücke

Wir beginnen mit der Probleminstanz $k = 0, \; d = 1$, dem *Maximin-Punkt auf der reellen Achse.* Im ungewichteten Fall ist dies die Aufgabe, einen Punkt x^* so zwischen vorgegeben Weltpunkten $S = \{x_1, \ldots, x_n\}$ auf der reellen Achse zu positionieren, daß der Abstand zum nächstgelegenen Weltpunkt maximiert wird. Um eine endliche Lösung zu erzwingen, schränken wir den zulässigen Bereich M auf die konvexe Hülle $\mathrm{conv}(S)$ der Punktmenge - dies ist gerade das Intervall $[x_{\min}, x_{\max}]$ - ein. Es ist das grundlegendste aller Maximin-Probleme, erlaubt jedoch bereits nichttriviale Aussagen über die Komplexität von Problemen dieses Typs.

Geometrisch gesehen ist x^* der Mittelpunkt des größten, von zwei benachbarten Punkten aus S begrenzten Intervalls. Das Problem, die Breite dieser größten Lücke zu bestimmen, ist Ausgangspunkt für unsere Komplexitätsbetrachtungen.

Problem 2 : [Größte Lücke (Maxgap)]

Gegeben: Eine Menge $S = \{x_1, \ldots, x_n\}$ von n reellen Zahlen.

Gesucht: Maximale Differenz zweier aufeinanderfolgender Zahlen aus S.

(Zwei Zahlen x_i und x_j aus S heißen *aufeinanderfolgend*, falls $x_i \geq x_j$ und es kein x_k mit $x_i \geq x_k \geq x_j$ gibt.)

Kennt man den optimalen Punkt x^*, kann man die Breite der größten Lücke in Zeit $O(n)$ bestimmen, d. h. Maxgap kann in linearer Zeit auf unser Ausgangsproblem reduziert werden. Eine superlineare untere Schranke für (P2) überträgt sich damit unmittelbar auf das Problem des Maximin-Punktes. Um eine solche untere Schranke zu erhalten, folgen wir der Technik von Ben-Or und betrachten ein verwandtes Entscheidungsproblem.

PROBLEM 3 : [Gleichförmige Lücken (Uniform Gap)]

Gegeben: Eine Menge $S = \{x_1, \ldots, x_n\}$ von n reellen Zahlen und ein $\epsilon \in \mathbb{R}$.

Entscheide: Sind alle Differenzen von aufeinanderfolgenden Zahlen aus S gleich ϵ ?

Lemma 1 ([LW86]) *Im algebraischen Entscheidungsbaummodell benötigt man zur Lösung von (P3) Zeit* $\Omega(n \log n)$.

Beweis: Ein Algorithmus, der (P3) löst, muß entscheiden können, ob ein $x \in \mathbb{R}^n$ in der Menge $W = \{(x_1, \ldots, x_n) \mid \exists \text{ Permutation } \sigma, \text{ so daß } x_{\sigma(i)} + \epsilon = x_{\sigma(i+1)},\ i = 1, \ldots, n-1\}$. enthalten ist. Für jede Permutation σ stellt die Menge $W_\sigma = \{(x_1, \ldots, x_n) \mid x_{\sigma(i)} + \epsilon = x_{\sigma(i+1)},\ i = 1, \ldots, n-1\}$ eine Gerade dar. Da diese $n!$ Geraden für $\epsilon > 0$ disjunkt sind, zerfällt W in $n!$ Zusammenhangskomponenten. Die Behauptung folgt dann unmittelbar aus Satz 1. ∎

Aus der unteren Schranke für dieses Hilfsproblem können wir die gewünschte Komplexitätsaussage ableiten.

Lemma 2 ([LW86]) *Im algebraischen Entscheidungsbaummodell benötigt man zur Lösung von (P2) Zeit* $\Omega(n \log n)$.

Beweis: Wir reduzieren (P3) auf (P2). Für eine beliebige Instanz S und ϵ von (P3), berechnen wir mit einem Algorithmus für (P2) die größte Lücke $maxgap$ zwischen aufeinanderfolgenden Punkten aus S. Ist $maxgap \neq \epsilon$, antworten wir mit 'Nein', ansonsten berechnen wir in linearer Zeit das Maximum x_{max} sowie das Minimum x_{min} von S. Gilt $x_{max} - x_{min} = (n-1)\epsilon$, so lautet die Antwort 'Ja', sonst 'Nein'. ∎

Einen Algorithmus für (P2), der diese Zeitschranke erreicht und mithin optimal ist, kann man ohne Schwierigkeiten angeben: sortiere die Menge S aufsteigend, laufe in linearer Zeit über S und bestimme das größte, von zwei benachbarten Punkten begrenzte Intervall. Diese Methode versagt jedoch, wenn wir multiplikative Gewichte $W = \{w_1, \ldots, w_n\}$ zulassen.

Problem 4 : [Maximin-Punkt auf der reellen Achse]

Gegeben: Eine Menge $S = \{x_1, \ldots, x_n\}$ von n reellen Zahlen, denen positive reelle Gewichte $W = \{w_1, \ldots, w_n\}$ zugeordnet sind.

Gesucht: Ein Punkt $x^* \in \text{conv}(S)$, der

$$\min_{1 \leq i \leq n} w_i \delta(x, x_i)$$

maximiert.

Obwohl die optimale Lage des Punktes x^* weiterhin von zwei Weltpunkten bestimmt wird, müssen diese im Unterschied zum ungewichteten Fall nicht mehr benachbart sein. Ein trivialer Algorithmus zählt alle $\binom{n}{2}$ Punktepaare aus S auf, berechnet zu jedem Punktepaar x_i, x_j die Menge X_{ij} der äquidistanten Punkte, d. h. die (i.a. zwei) Lösungen der Gleichung $w_i \delta(x_i, x) = w_j \delta(x_j, x)$. Dann prüft er für jede Lösung $x_{ij} \in X_{ij}$ in linearer Zeit, ob ein Punkt x_k $(k \neq i, j)$ existiert, der zu x_{ij} einen kleineren gewichteten Abstand aufweist als x_i und x_j. Ist dies der Fall, verwirft er x_{ij}. Aus den verbleibenden x_{ij} $(1 \leq i \neq j \leq n)$ wählt er dasjenige aus, das $w_i \delta(x_i, x_{ij})$ maximiert. Dieser Ansatz benötigt offenbar Zeit $O(n^3)$.

Es stellt sich unmittelbar die Frage, ob ein Algorithmus existiert, der die optimale Zeitkomplexität $O(n \log n)$ besitzt. Wir werden die Frage in Abschnitt 3.5 positiv beantworten können.

Legt man ein erweitertes Berechnungsmodell zugrunde, kann man einen Algorithmus angeben, der eine Lösung von (P2) in linearer Zeit findet [Go75]. Dieses Berechnungsmodell erlaubt zusätzlich zu den Operationen der real RAM die Verwendung der nichtanalytischen *unteren Gaußklammer (floor function)* $\lfloor . \rfloor$.

Der Algorithmus teilt das Intervall zwischen x_{min} und x_{max} in $n-1$ gleichgroße Teilintervalle, die Fächer. Die verbleibenden $n-2$ Punkte werden ihrem Zahlwert entsprechend in die Fächer einsortiert. (Das Fach, in welches ein Punkt fällt, kann unter Verwendung der unteren Gaußklammer in konstanter Zeit bestimmt werden.) Dabei merkt sich der Algorithmus den höchsten und niedrigsten Wert eines jeden Fachs i $(1 \leq i \leq n-1)$ in den Variablen $HIGH[i]$ und $LOW[i]$. Da $n-2$ Punkte auf $n-1$ Fächer verteilt werden, bleibt mindestens ein Fach leer. Die größte Lücke hat damit mindestens die Größe eines Fachs; sie läßt sich durch einen Lauf über die Fächer, der die LOW und $HIGH$-Werte untersucht, ermitteln.

3.2 Verankerte Gerade in der Ebene

Um die Behandlung verankerter Probleme zu vereinfachen, legen wir das Koordinatensystem stets so, daß der Anker mit dem Ursprung zusammenfällt. Die entsprechend transformierten Koordinaten der Weltpunkte können unmittelbar in linearer Zeit bestimmt werden. Wir betrachten also o.B.d.A. in Zukunft nur solche verankerten Probleme, in denen der zu plazierende affine Teilraum den Ursprung enthalten soll (und damit einen linearen Teilraum darstellt).

Die einfachste nichttriviale verankerte Problemstellung ist der Fall $d = 2,\ k = 1$, das

PROBLEM 5 : [Maximin-Ursprungsgerade in der Ebene]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\},\ p_i = (x_i, y_i)$ von n Punkten der euklidischen Ebene und eine Menge $W = \{w_1, \ldots, w_n\}$ von positiven reellen Gewichtsfaktoren.

Gesucht: Eine Ursprungsgerade $g^* : a^*x + b^*y = 0$, die den minimalen gewichteten euklidischen Abstand zur Punktmenge S

$$\min_{1 \leq i \leq n} w_i \delta(g, p_i) = \min_{1 \leq i \leq n} \frac{w_i}{\sqrt{a_g^2 + b_g^2}} (|a_g x_i + b_g y_i|)$$

unter allen Ursprungsgeraden $g : a_g x + b_g y = 0$ maximiert. (a_g und b_g sind die Komponenten eines Normalenvektors von g).

Sind alle Gewichtsfaktoren gleich, erlaubt das Problem folgende anschauliche Deutung: suche das breiteste Band B, dessen Symmetrieachse durch den Ursprung verläuft, und dessen offener Kern $int(B)$ keinen Punkt einer vorgegeben Punktmenge S enthält. Ein Beispiel einer Probleminstanz mit optimaler Lösung zeigt die Abbildung 3.1.

Man kann sich der Gewichtsfaktoren entledigen, indem man zur Menge der transformierten Weltpunkte $S_W = \{(w_i x_i, w_i y_i)\}$ übergeht. Wir können unsere Betrachtungen also o.B.d.A. auf ungewichtete Probleminstanzen beschränken.

Wie in Kapitel 2 ausgeführt, kann man zu jedem geometrischen Maximin-Problem eine äquivalente funktionale Problemfassung finden. Wir wollen dies an diesem Beispiel explizit durchführen. Dazu bemerken wir, daß wir die Lage der Geraden g durch einen einzigen Parameter $k \in [-1, 1]$ beschreiben können, die Abszisse ihres Schnittpunkts mit dem Einheitskreis. Sei also $g_k : \sqrt{1 - k^2}x - ky = 0$. Jedem Punkt $p_i \in S$ ordnen wir eine Funktion f_i über dem Konfigurationsraum $[-1, 1]$ zu, die seinen Abstand zur Gerade g_k in Abhängigkeit von k beschreibt:

$$f_i(k) = \delta(p_i, g_k) = |\sqrt{1 - k^2}x_i - ky_i|$$

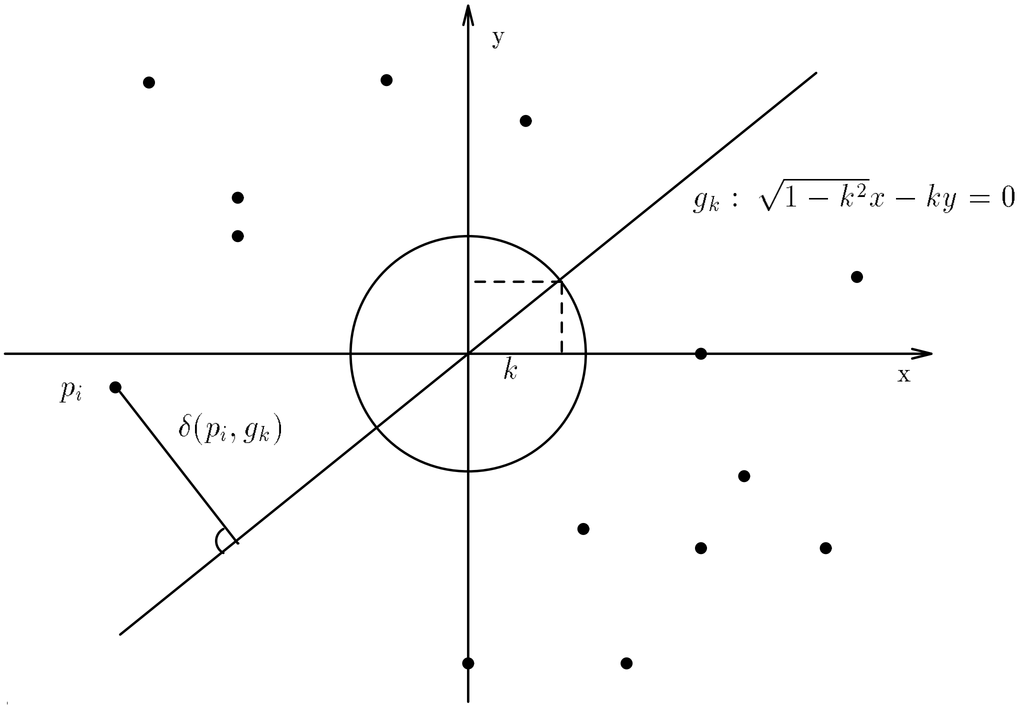

Abbildung 3.1: Punktmenge mit Maximin-Ursprungsgerade

Unsere Aufgabe besteht nun im wesentlichen darin, daß MaxMin-Problem für die Funktionenschar $\mathcal{F} = \{f_1, \ldots, f_n\}$ zu lösen. Der Umstand, daß die f_i nicht überall differenzierbar sind, kompliziert die Betrachtungen unnötig; wir glätten daher die Abstandsfunktionen durch Quadrieren: $F_i(k) := f_i^2(k) = (\sqrt{1-k^2}x_i - ky_i)^2$. Diese Transformation hat keinen Einfluß auf die Lage des Optimums, da sie als streng monotone Abbildung (über $\mathbb{R}_0^+$) die relative Ordnung der Funktionswerte erhält.

3.3 Die MaxMin-Aufgabe

Wir haben gesehen, wie sich geometrische Maximin-Aufgaben mit einem Freiheitsgrad als MaxMin-Probleme über Funktionenscharen in einer Variablen formulieren lassen. In diesem Abschnitt werden wir effiziente Algorithmen zur Lösung solcher Probleme diskutieren und untere Schranken für einige Funktionenklassen zeigen.

Bemerkung: Alle im folgenden erzielten Resultate und entwickelten Techniken lassen sich wegen der Identität $\max f(x) = -\min(-f(x))$ unmittelbar auf obere Einhüllende und das duale MinMax-Problem übertragen.

3.3.1 Lage der Maxima der unteren Einhüllenden

Die MaxMin-Aufgabe fragt nach der Abszisse des Maximums der unteren Einhüllenden von $\mathcal{F}_I = \{f_i \mid i \in I\}$ (I bezeichne hier und im folgenden eine endliche Indexmenge). An welchen Stellen kann dieses Maximum auftreten? Diese Frage beantwortet das

Lemma 3 *Sei $\mathcal{F}_I = \{f_i \mid i \in I\}$ eine Schar reellwertiger, stetig differenzierbarer Funktionen über einer kompakten Menge $T \subset \mathbb{R}$.*

Dann gibt es ein $x^* \in T$ und ein $j \in I$ mit

$$f_j(x^*) = \max_{x \in T} f^u[\mathcal{F}](x)$$

Ferner erfüllt x^* eine der Bedingungen

(i.) $\dfrac{\partial f_j}{\partial x}(x^*) = 0$

(ii.) $f_j(x^*) = f_k(x^*)$ *wobei* $k \in I \setminus \{j\}$

(iii.) $x^* \in \partial T$

Beweis: Mit den $f_i \in \mathcal{F}_I$ ist auch $f^u[\mathcal{F}_I]$ eine stetige Funktion und nimmt daher nach dem Satz von Weierstraß über einer kompakten Menge ihr Maximum an. Weiterhin kann die Funktion f^u auf $int(T)$ nur dort maximal werden, wo ihre Ableitung verschwindet oder nicht erklärt ist. Im ersten Fall ist Bedingung (i.) erfüllt, der zweite Fall impliziert (ii.). Punkt (iii.) berücksichtigt Maxima auf dem Rand des Definitionsbereichs. ∎

Die Menge der Punkte aus T, die eine der Bedingungen $(i.)-(iii.)$ des Lemmas erfüllen, umfaßt alle lokalen Extrema der unteren Einhüllenden f^u; wir nennen sie *kritische Punkte* auf T.

Unser Ziel ist es nun, ein effizientes Verfahren zu entwickeln, das es erlaubt, die kritischen Punkte rasch zu finden, um aus ihnen den optimalen Punkt x^* heraussuchen zu können. Wir nehmen hierzu an, daß folgende Operationen in konstanter Zeit durchgeführt werden können:

- bestimme den Funktionswert einer Funktion $f_i \in \mathcal{F}$ an der Stelle $x \in \mathbb{R}$
- bestimme alle Schnittpunkte zweier Funktionen $f_i, f_j \in \mathcal{F}$
- bestimme alle *stationären Punkte* (Punkte mit verschwindender Ableitung) von $f_i \in \mathcal{F}$.

Ein naiver Algorithmus zählt alle stationären Punkte der Funktionen aus $\mathcal{F}$ sowie ihre paarweisen Schnitte auf, prüft, welche von ihnen auf der unteren Einhüllenden liegen, fügt zu den Punkten, die dies erfüllen, die Funktionswerte von f^u an den Randpunkten des Definitionsbereichs hinzu und antwortet mit dem Maximum dieser Menge.

Ist sowohl die Anzahl der stationären Punkte jeder Funktion als auch die Anzahl der Schnitte zweier beliebiger Funktionen aus $\mathcal{F}$ durch eine Konstante beschränkt, benötigt diese Methode Laufzeit $O(n^3)$.

Viele dieser Berechnungen sind jedoch überflüssig: alle Punkte die nicht auf der unteren Einhüllenden liegen, scheiden von vornherein aus. Diese Beobachtung legt eine modifizierte Strategie nahe:

1. Berechne eine geeignete Darstellung der unteren Einhüllenden.

2. Bestimme deren lokale Maxima und suche aus diesen das globale Maximum heraus.

3.3.2 Davenport-Schinzel-Sequenzen

Befassen wir uns zunächst mit dem ersten Schritt der erläuterten Strategie, der effizienten Berechnung und Repräsentierung der unteren Einhüllenden $f^u[\mathcal{F}]$ einer Funktionenschar $\mathcal{F} = \{f_1, \ldots, f_n\}$ über einem Intervall Int.

Da die Laufzeit effizienter Algorithmen entscheidend von der Komplexität der zu erzeugenden Strukturen abhängt, beginnen wir mit der Analyse der kombinatorischen Komplexität von unteren Einhüllenden .

Der Wert der unteren Einhüllenden $f^u[\mathcal{F}]$ in einem Punkt $x \in \mathbb{R}$ wird durch eine oder mehrere Funktionen $f_i \in \mathcal{F}$ bestimmt, deren Funktionswert an dieser Stelle minimal unter allen Funktionen aus $\mathcal{F}$ ist: $f^u(x) = f_i(x) \leq f_j(x)\ \forall j \neq i$; wir nennen diese Funktionen f_i *aktiv* in x.

Die Schar $\mathcal{F}$ habe im folgenden die Eigenschaft, daß sich je zwei Funktionen $f_i, f_j \in \mathcal{F}$ $(i \neq j)$ in höchstens $s \in \mathbb{N}$ Punkten schneiden.

Ihre untere Einhüllende f^u induziert dann in natürliche Weise eine Partition des reellen Intervalls Int in

- eine Menge von k maximal (bezüglich Mengeninklusion) zusammenhängenden Teilintervallen Int_l $(1 \leq l \leq k)$, in denen genau eine Funktion $f_{i_k} \in \mathcal{F}$ aktiv ist

- eine Menge von Grenzen dieser Intervalle, den *Knickstellen* von f^u, in denen zwei oder mehr Funktionen aus $\mathcal{F}$ aktiv sind.

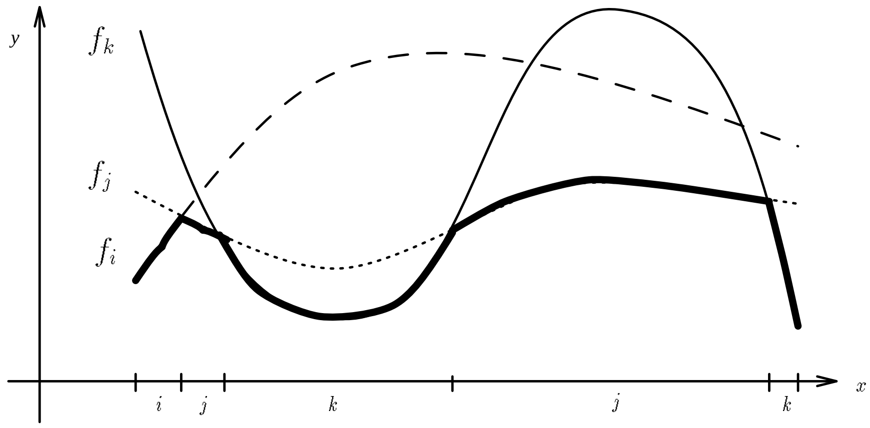

Abbildung 3.2: Untere Einhüllende dreier Funktionen mit Minimierungssequenz $ijkjk$

Seien die disjunkten Intervalle Int_l $(1 \leq l \leq k)$ von links nach rechts aufsteigend numeriert, dann heißt die Liste $(i_1, i_2, \ldots, i_k)$ der zugehörigen Indizes i_l die *Minimierungssequenz* von $\mathcal{F}$ (kurz MS[$\mathcal{F}$]). Abbildung 3.2 veranschaulicht diese Begriffe.

Zusammen mit den jeweiligen Intervallgrenzen charakterisiert die Minimierungssequenz die untere Einhüllende vollständig: sie gibt an, wie sich der Graph von f^u aus Teilstücken der Graphen von f_i $(1 \leq i \leq n)$ zusammensetzt. Die Länge dieser Beschreibung ist durch k gegeben; wir fragen nun, wie groß k in Abhängigkeit von n und s werden kann. Dazu betrachtet man folgende kombinatorische Struktur [DS65]:

Definition 3 *Seien n, s natürliche Zahlen.*
*Eine Folge $U = (u_1, \ldots, u_m)$ natürlicher Zahlen heißt $(n,s)-$*Davenport-Schinzel-Sequenz *(kurz DS(n, s)) wenn sie folgende Eigenschaften erfüllt:*

(i.) $1 \leq u_i \leq n$ *für alle* i.

(ii.) Für alle $i < m$ *gilt* $u_i \neq u_{i+1}$.

(iii.) Es gibt keine $s + 2$ *Indizes* $1 \leq i_1 < i_2 < \ldots < i_{s+2} \leq m$, *so daß*
$u_{i_1} = u_{i_3} = u_{i_5} = \ldots = a$,
$u_{i_2} = u_{i_4} = u_{i_6} = \ldots = b$, *und* $a \neq b$.

Informal gesprochen handelt es sich bei einer $DS(n,s)$ um eine endliche Sequenz natürlicher Zahlen $\leq n$, die keine Subsequenz der Form $a..b..a..b..$ der Länge $s + 2$ aufweist.

Der Zusammenhang zwischen Davenport-Schinzel-Sequenzen und unteren Einhüllenden ist gegeben durch den

Satz 3 *Sei $\mathcal{F} = \{f_1, \ldots, f_n\}$ eine Schar von n reellwertigen, stetigen Funktionen über einem Intervall $Int \subseteq I\!R$, sodaß für alle $1 \leq i < j \leq n$ gilt: $f_i(x) = f_j(x)$ hat höchstens s Lösungen. Dann bildet die Minimierungssequenz von $\mathcal{F}$ eine DS(n, s).*
Umgekehrt kann man zu jeder DS(n, s) eine Funktionenschar $\mathcal{F}$ angeben, die obige Bedingungen erfüllt und deren Minimierungssequenz gleich der gegebenen DS(n, s) ist.

Beweis: "$\Rightarrow$" Die Minimierungssequenz von $\mathcal{F}$ erfüllt offensichtlich die Punkte (i.) und (ii.) der Definition 3. Enthielte MS[$\mathcal{F}$] eine Subfolge $i..j..i..j..$ der Länge $s+2$, so müßten sich f_i und f_j im Widerspruch zur Voraussetzung $(s+1)$-mal schneiden.

"$\Leftarrow$" vgl. [At83] und [At85]

∎

Definition 4 *Sei $|U|$ die Länge der Davenport-Schinzel-Sequenz U.*

$\lambda_s(n) := max\ \{|U| : U \text{ ist eine } DS(n,s)\}$.

Eine erste einfache obere Schranke für $\lambda_s(n)$ läßt sich mit Hilfe von Satz 3 zeigen:
Sei U eine DS(n, s) maximaler Länge, $\mathcal{F}$ die nach Satz 3 existierende Funktionenschar, deren Minimierungssequenz durch U gegeben ist. Es kann nun höchstens $\binom{n}{2} \cdot s$ Schnittpunkte von Funktionen aus $\mathcal{F}$ über Int geben, die, falls sie alle auf dem Graph von f^u liegen würden, Int in höchstens $\binom{n}{2} \cdot s + 1$ Teilintervalle Int_l teilen könnten. Also sind $k = |U|$ und somit auch $\lambda_s(n)$ durch $O(n^2)$ beschränkt.

Diese Schranke ist jedoch bei weitem nicht scharf. Die Tatsache, die Davenport-Schinzel-Sequenzen besonders interessant macht, ist, daß sie um fast eine Größenordnung langsamer wachsen, als die naive Schranke vermuten läßt.
Die exakte Bestimmung der Komplexität von $\lambda_s(n)$ ist Gegenstand weitreichender Untersuchungen, z. B. in [DS65], [Sz74], [At85], [Sh88a] und [ASS]. Um die Ergebnisse dieser Autoren zu erläutern benötigen wir die

Definition 5
Bezeichne $g^{(n)}$ die n-fache Komposition der Funktion g.
Die Funktionenfolge $\{A_k\}_{k=1}^{\infty}$, $A_k : I\!N \rightarrow I\!N$, wird induktiv definiert durch

$$\begin{aligned} A_1(n) &= 2n \\ A_k(n) &= A_{k-1}^{(n)}(1) \quad \forall\, k \geq 2 \end{aligned}$$

Die Funktion $A(n) := A_n(n)$ heißt Ackermannfunktion.

Ist $g : I\!N \to I\!N$ eine streng monotone Funktion, so nennt man die Funktion

$$\gamma : I\!N \to I\!N \quad \gamma(n) = \min\{s \geq 1 : g(s) \geq n\}$$

ihre Umkehrfunktion.
Man bezeichnet die Umkehrfunktionen von $A(n)$ und $A_k(n)$ als $\alpha(n)$ und $\alpha_k(n)$.

Die ersten Funktionen der Folge $\{A_k\}_{k=1}^{\infty}$ und ihre Umkehrfunktionen sind gegeben durch

$$\begin{array}{ll} A_1(n) = 2n & \alpha_1(n) = \left\lceil \frac{n}{2} \right\rceil \\ A_2(n) = 2^n & \alpha_2(n) = \lceil \log(n) \rceil \\ A_3(n) = 2^{2^{\cdot^{\cdot^{2}}}} & \alpha_3(n) = \log^*(n) \end{array}$$

wobei der "Exponententurm" von $A_3(n)$ n Zweien enthält.

Die Ackermannfunktion $A(n)$ wächst extrem schnell über alle Grenzen, ihre Umkehrfunktion $\alpha(n)$ strebt zwar ebenfalls mit steigendem n monoton gegen unendlich, tut dies jedoch so langsam, daß sie für alle praktischen Zwecke als konstant angesehen werden kann: $\alpha(n) \leq 4$ für alle $n \leq A(4)$ und dies ist ein Exponententurm aus 65536 Zweien.

Die wichtigsten bekannten Schranken für $\lambda_s(n)$ werden nun zusammengefaßt im

Satz 4 (Sharir et al.) *Es gilt*

1. $\lambda_1(n) = n; \lambda_2(n) = 2n - 1$
2. $\lambda_3(n) = \Theta(n\alpha(n))$
3. $\lambda_4(n) = \Theta(n2^{\alpha(n)})$
4. $\lambda_s(n) = O(n \log^*(n))$

$\lambda_s(n)$ wächst also nur wenig schneller als linear.

Abschließend zeigen wir ein technisches Lemma, das einige der im folgenden ausgeführten Beweise verkürzt. Es beschreibt einen Zusammenhang zwischen Davenport-Schinzel Sequenzen gleicher Ordnung s aber verschiedener Symbolzahl n.

Lemma 4 *Sei k eine natürliche Zahl. Dann gilt*

$$\lambda_s(kn) = O(\lambda_s(n)),$$

wobei der Proportionalitätsfaktor von k und s abhängt.

Beweis: Zeige zunächst

$$\exists\, c \in \mathbb{N} : \quad \lambda_s(2n) \leq c \cdot \lambda_s(n)$$

Sei $\mathcal{F}$ eine Schar von $2n$ stetigen Funktionen, die sich paarweise höchstens s-mal schneiden und deren untere Einhüllende $f^u[\mathcal{F}]$ die maximal mögliche Komplexität $\lambda_s(2n)$ aufweist. Teile $\mathcal{F}$ in zwei gleichgroße Teilscharen $\mathcal{F}_1, \mathcal{F}_2$. l_1 (l_2) bezeichne die Länge der Minimierungssequenz von $f^u[\mathcal{F}_1]$ ($f^u[\mathcal{F}_2]$). Nach Satz 3 und Definition 4 gilt $l_1, l_2 \leq \lambda_s(n)$. Unser Ziel ist es, $\lambda_s(2n)$ durch $\lambda_s(n)$ nach oben abzuschätzen.

$f^u[\mathcal{F}]$ ergibt sich als untere Einhüllende von $f^u[\mathcal{F}_1]$ und $f^u[\mathcal{F}_2]$. Projiziert man die Schnittpunkte auf den Graphen von $f^u[\mathcal{F}_1]$ und $f^u[\mathcal{F}_2]$ auf die x-Achse, zerfällt diese in höchstens $l_1 + l_2 \leq 2\lambda_s(n)$ Intervalle. Im Innern eines solchen Intervall Int sind die Funktionen $f_i \in \mathcal{F}_1$ mit $f_i(x) = f^u[\mathcal{F}_1](x)$ und $f_j \in \mathcal{F}_2$ mit $f_j(x) = f^u[\mathcal{F}_2](x)$ eindeutig bestimmt. f_i und f_j schneiden sich höchstens s-mal. Damit hat die Minimierungssequenz von $f^u[\mathcal{F}]$ auf Int maximale Länge $s+1$. Summiert man über alle Intervalle, ist die Länge der Minimierungssequenz durch $(s+1)\cdot 2\lambda_s(n)$ beschränkt. Wir erhalten also

$$\begin{aligned}
\lambda_s(2n) &\leq 2(s+1)\lambda_s(n) \\
&\leq c \cdot \lambda_s(n) \quad \text{mit} \qquad c = 2(s+1)
\end{aligned}$$

Durch wiederholtes Anwenden dieser Abschätzung findet man

$$\lambda_s(kn) \leq c^{\lceil \log k \rceil} \lambda_s(n)$$

∎

3.3.3 Erweiterung auf partielle Funktionen

Das Konzept der unteren Einhüllenden läßt sich unmittelbar auf partielle Funktionen übertragen. $f^u[\mathcal{F}]$ ist dann über der Vereinigung der Definitionsbereiche erklärt und ergibt sich in jedem Punkt als Minimum der an dieser Stelle definierten Funktionen. Wir interessieren uns wie im vorhergehenden Abschnitt dafür, aus wievielen Teilstücken der Graphen der Funktionen $f_i \in \mathcal{F}$ sich der Graph von $f^u[\mathcal{F}]$ höchstens zusammensetzt oder mit anderen Worten für die maximal mögliche Länge der Minimierungssequenz von $f^u[\mathcal{F}]$.

Fordern wir, daß die Definitionsbereiche der Funktionen zusammenhängend sind, können wir für die Komplexität der unteren Einhüllenden ein ähnliches Resultat wie in Abschnitt 3.3.2 erhalten.

Satz 5 ([At83]) *Sei $\mathcal{F} = \{f_1, \ldots, f_n\}$ eine Schar von n stetigen, reellwertigen Funktionen, von denen jede über einem Intervall $Def_i \subseteq I\!R$ definiert ist. Gilt für jedes Indexpaar $1 \leq i < j \leq n$, daß die Gleichung $f_i(x) = f_j(x)$ höchstens s Lösungen hat, ist die Länge der Minimierungssequenz von $f^u[\mathcal{F}]$ durch $\lambda_{s+2}(n)$ beschränkt.*

Beweis: Betrachtet man die alternierenden Indexsubfolgen der Form $i..j..i..j..$, die zwei Funktionen $f_i, f_j \in \mathcal{F}$ in der Minimierungssequenz von f^u erzeugen kann, so kann deren Länge $s+3$ nicht übersteigen: im gemeinsamen Definitionsbereich $Def_i \cap Def_j$ ist ihre Länge wegen der geforderten Schnitteigenschaft auf $s+1$ beschränkt, in der symmetrischen Differenz ihrer Definitionsbereiche $(Def_j \setminus Def_i) \cup (Def_j \setminus Def_i)$ können noch zwei weitere Indizes auftreten. Damit ist die Minimierungssequenz von $f^u[\mathcal{F}]$ eine DS$(n, s+2)$. ∎

3.3.4 Effiziente Berechnung von unteren Einhüllenden

Wir beschreiben einen Algorithmus aus [At83] (siehe auch [Sh88b]) zur Berechnung der unteren Einhüllenden einer Schar $\mathcal{F}$ von n stetigen, jeweils über einem Intervall definierten Funktionen, die sich paarweise höchstens s-mal schneiden.

Der Algorithmus beruht auf der *Divide & Conquer* - Strategie: zuerst teilt man die Schar $\mathcal{F}$ in zwei (etwa) gleichgroße Mengen $\mathcal{F}_1 = \{f_1, \ldots, f_{\lfloor \frac{n}{2} \rfloor}\}$ und $\mathcal{F}_2 = \{f_{\lfloor \frac{n}{2}+1 \rfloor}, \ldots, f_n\}$ auf, berechnet dann rekursiv die unteren Einhüllenden der beiden Teilscharen $f^u[\mathcal{F}_1]$ und $f^u[\mathcal{F}_2]$, und fügt aus diesen beiden unteren Einhüllenden das gesuchte Resultat $f^u[\mathcal{F}] = \min\{f^u[\mathcal{F}_1], f^u[\mathcal{F}_2]\}$ zusammen.

Wir repräsentieren die untere Einhüllende mit Hilfe ihrer Minimierungssequenz, wobei jeder Index mit den Grenzen des zugehörigen Intervalls versehen wird.

Der *Mischschritt* geschieht wie folgt: jede der beiden unteren Einhüllenden induziert eine Partition der reellen Achse. Wir vereinigen diese Aufteilung und verfeinern sie weiter, indem wir die Abszissen der Schnittpunkte von $f^u[\mathcal{F}_1]$ und $f^u[\mathcal{F}_2]$ hinzufügen. Für jedes der neu entstandenen Intervalle können wir leicht entscheiden, welche der Funktionen $f^u[\mathcal{F}_1]$ und $f^u[\mathcal{F}_2]$ auf diesem definiert sind, und falls dies für beide zutrifft, welche auf dem Intervall aktiv ist. Daraus läßt sich die gesuchte Struktur konstruieren.

Die Laufzeit des Mischschritts wächst proportional zur Länge seiner Ein- und Ausgabe. Beide sind wegen

$$2\lambda_{s+2}(\frac{n}{2}) < \lambda_{s+2}(n)$$

von der Größenordnung $O(\lambda_{s+2}(n))$. Die Laufzeit $t(n)$ des Gesamtalgorithmus genügt somit der Rekursionsformel

$$t(n) = 2t(n/2) + O(\lambda_{s+2}(n))$$

mit der Lösung $t(n) = O(\lambda_{s+2}(n) \log n)$.

Für den Fall totaler Funktionen über einem gemeinsamen Definitionsbereich läßt sich durch analoges Vorgehen eine Laufzeit von $O(\lambda_s(n) \log n)$ erzielen.

3.3.5 Der MaxMin-Algorithmus

Nachdem wir für Schritt 1 der MaxMin-Strategie einen effizienten Algorithmus gefunden haben, liegt die Realisierung von Schritt 2 nahe.

Lemma 5 *Erfüllt eine Schar $\mathcal{F} = \{f_1, \ldots, f_n\}$ stetig differenzierbarer, über einem gemeinsamen kompakten Intervall $Int \subset I\!R$ erklärter Funktionen die in Abschnitt 3.3.1 gestellten Bedingungen und schneiden sich je zwei Funktionen $f_i, f_j \in \mathcal{F}$ höchstens s-mal, so können wir das MaxMin-Problem für $\mathcal{F}$ in Zeit $O(\lambda_s(n) \log n)$ lösen.*

Beweis: Nach Berechnung der unteren Einhüllenden f^u ermittelt der Algorithmus die Funktionswerte ihrer stationären Punkte, bestimmt die Funktionswerte der Knickstellen und der Randpunkte des Definitionsbereichs Int und antwortet mit der Abszisse des größten dieser Werte. Die Laufzeit des zweiten Schritts unserer Strategie, die Berechnung der kritischen Punkte, wächst proportional zur Länge der Minimierungssequenz von $\mathcal{F}$; der erste Schritt, die Berechnung von f^u, dominiert mithin die Laufzeit und die behauptete Schranke folgt. ∎

Eine Funktionenklasse, für die wir die geforderten Operationen mit den Elementaroperationen unseres Berechnungsmodells in konstanter Zeit realisieren können, sind die Polynomfunktionen vom Grad ≤ 4. Für Polynomfunktionen höheren Grades ist die Schnitt- und Extremaberechnung im allgemeinen nicht mehr durch geschlossene Wurzelausdrücke zu bewältigen, sondern erfordert numerische Verfahren.

Der Algorithmus ist in gleicher Weise auch auf andere Funktionenklassen anwendbar; je nach Art der Funktionen kann es aber nötig sein, den Satz der zulässigen Elementaroperationen entsprechend zu erweitern (etwa um trigonometrische Funktionen etc.).

3.4 Untere Schranken

Um die Güte des entwickelten Algorithmus zu beurteilen, leiten wir eine untere Schranke für ein verwandtes Problem her.

Problem 6 : [Maximale Ordinate der unteren Einhüllenden quadratischer Funktionen]

Gegeben: Eine endliche Schar $\mathcal{F}_I = \{f_i \mid i \in I\}$ von Polynomfunktionen vom Grad ≤ 2. (Als Beschreibung der Funktionen f_i seien Tripel (a_i, b_i, c_i) gegeben, so daß $f_i(x) = a_i x^2 + b_i x + c_i$ für alle $x \in \mathbb{R}$ gilt.)

Gesucht: Ordinate eines Maximums der unteren Einhüllenden von $\mathcal{F}_I$ über

(a.) einem kompakten Intervall $Int \subset \mathbb{R}$

(b.) $\mathbb{R}$, falls es existiert

Lemma 6 *Im algebraischen Entscheidungsbaummodell benötigt man zur Lösung von (P6) Zeit $\Omega(n \log n)$.*

Beweis: Reduziere (P2) auf (P6). Aus einer Instanz $S = \{x_1, \ldots, x_n\}$ des Problems der größten Lücke konstruieren wir die Funktionenschar $\mathcal{F}_I = \{(x - x_i)^2,\ 1 \leq i \leq n\}$. Wir bestimmen in linearer Zeit x_{min} und x_{max} und maximieren in Variante (a.) $f^u[\mathcal{F}_I]$ über $[x_{min}, x_{max}]$.

Im Variante (b.) erweitern wir $\mathcal{F}_I$ um die durch $(x_{\min}, 0)$ bzw. $(x_{\max}, 0)$ und den Schnittpunkt der beiden äußeren Parabeln verlaufenden Geraden zu

$$\mathcal{F}_J = \mathcal{F}_I \cup \{\frac{(x_{max} - x_{min})}{2}(x - x_{min}), \frac{(x_{min} - x_{max})}{2}(x - x_{max})\}$$

und maximieren $f^u[\mathcal{F}_J]$ über $\mathbb{R}$. Sei y^* die so gewonnene Maximumsordinate; die Größe der maximalen Lücke ergibt sich dann zu $2\sqrt{y^*}$. ∎

Da sich die Maximumsordinate aus der Maximumsabszisse in linearer Zeit bestimmen läst, überträgt sich die gefundene untere Schranke unmittelbar auf das MaxMin-Problem für quadratische Funktionen. Unser MaxMin-Algorithmus ist also für quadratische Funktionen optimal. Die untere Schranke gilt natürlich auch für alle Funktionenklassen, die die quadratischen Funktionen umfassen, so auch für die Polynomfunktionen vom Grad $d \leq 4$. Obiger Algorithmus, der das MaxMin-Problem für solche Polynomfunktionen in Zeit $O(\lambda_4(n) \log n)$ lösen kann, ist damit bis auf einen Faktor $O(2^{\alpha(n)})$ optimal.

Erlaubt man Elementaroperationen der Form "ermittle sämtliche Nullstellen eines Polynoms vom Grad $\leq d$ (d fest)", so erfüllen alle Polynome bis zu einem beliebigen festen Grad d die auf Seite 24 geforderten Bedingungen. Ein entsprechend erweiterter Algorithmus löst dann das MaxMin-Problem für Polynome vom Grad $\leq d$ in Zeit $O(\lambda_d(n) \log n)$. Erstaunlicherweise gelten untere Schranken, die mit Hilfe des Satzes von Ben-Or für algebraische Entscheidungsbäume gewonnen wurden, selbst dann, wenn man solche Operationen zulässt [Be83]. Somit ist auch in diesem Fall Optimalität bis auf einen Faktor $O(\log^*(n))$ erreicht.

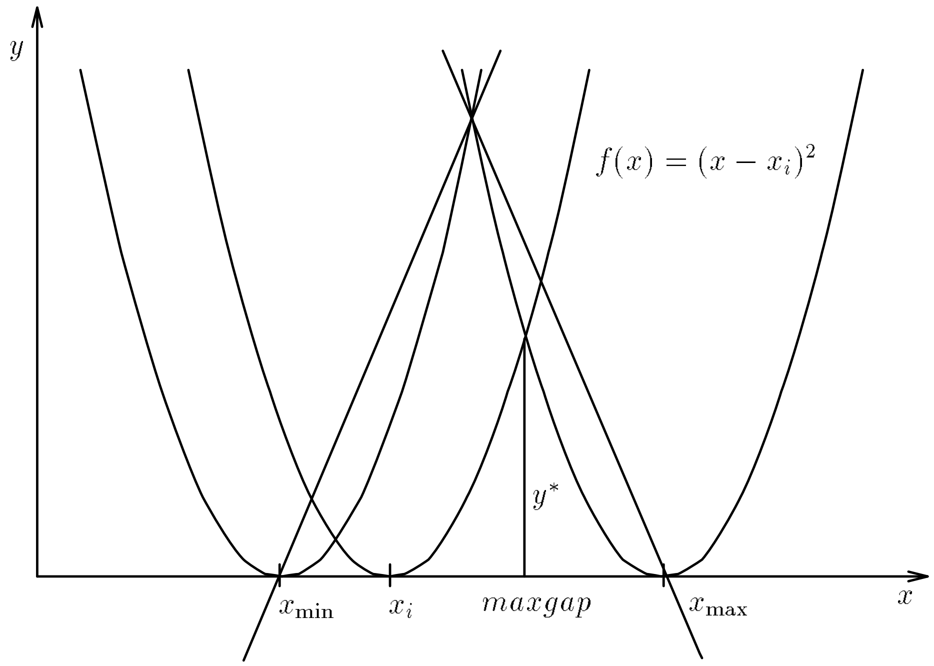

Abbildung 3.3: Reduktion von Maxgap auf die MaxMin-Aufgabe quadratischer Funktionen

Es stellt sich natürlich die Frage, ob man das Resultat auch auf lineare Funktionen ausweiten kann. Sei $\mathcal{F}_I = \{f_i \mid i \in I\}$ mit $f_i(x) = a_i x + b_i$ eine endliche Schar linearer Funktionen. Die Berechnung der Abszisse des Maximums der unteren Einhüllenden $f^u[\mathcal{F}_I]$ ist äquivalent zur Lösung des *linearen Programms*

$$\begin{array}{rl} \text{Maximiere} & y \\ \text{u.d.N.} & a_i x + b_i - y \leq 0 \quad \forall i \in I, \end{array}$$

einer speziellen Variante des allgemeinen

PROBLEM 7 : [lineares Programm in 2 Variablen unter n Nebenbedingungen]

Gegeben: Eine lineare Zielfunktion $ax+by$ sowie n lineare Nebenbedingungen der Form $g_i(x, y) = a_i x + b_i y + c_i \leq 0,\ 1 \leq i \leq n$.

Maximiere $ax + by$

u.d.N. $g_i(x, y) = a_i x + b_i y + c_i \leq 0 \quad \forall 1 \leq i \leq n$

Megiddo nutzt in einem sehr geschickten *Prune & Search* - Algorithmus die Tatsache aus, daß man nicht die gesamte zulässige Region $R = \{(x, y) \in \mathbb{R}^2 \mid g_i(x, y) \leq 0,\ \forall\, 1 \leq i \leq n\}$ explizit berechnen muß, um den optimalen Lösungspunkt zu finden. Dieser liegt

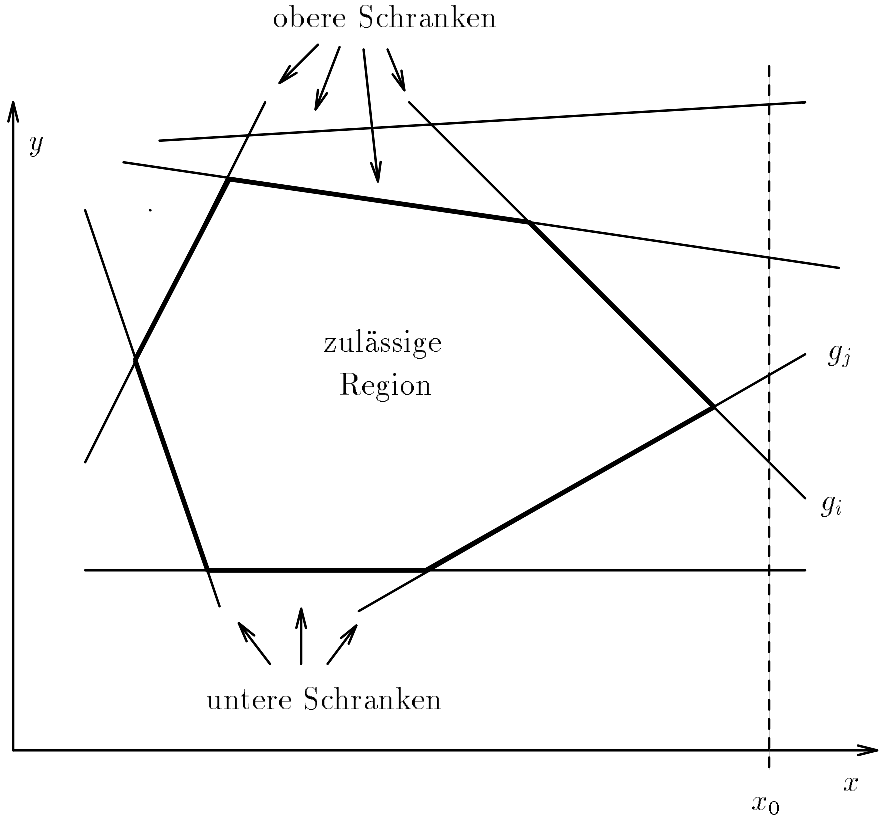

Abbildung 3.4: Im Beispielfall liegt die zulässige Region links von x_0.

stets in einer Ecke der zulässigen Region R, die als Schnitt von n Halbebenen die Form eines konvexen Polygons aufweist.

Der Algorithmus führt eine iterative Suche nach dem optimalen Eckpunkt durch. In jeder Iteration wird ein konstanter Bruchteil der Nebenbedingungen als redundant verworfen, bis nur die beiden Nebenbedingungen übrigbleiben, die den optimalen Eckpunkt bestimmen. (Eine Nebenbedingung heißt redundant, wenn sie nicht durch den optimalen Eckpunkt verläuft.) Da die Laufzeit jeder Iteration proportional zur Anzahl der noch berücksichtigten Nebenbedingungen ist, benötigt der gesamte Algorithmus lineare Zeit.

Wir illustrieren nun, wie die redundanten Nebenbedingungen gefunden werden können. Dazu nehmen wir an, daß (wie in unserem Ausgangsproblem) die zu maximierende Zielfunktion die Funktion $f(x) = y$ ist (Dies kann man durch lineare Transformation des Problems erreichen). Wir teilen die Nebenbedingungen je nach Vorzeichen von b_i in obere ($b_i > 0$) und untere ($b_i < 0$) Schranken des zulässigen Bereichs. Man kann nun zu einer vorgegeben Abszisse x_0 entscheiden, ob der optimale Eckpunkt rechts oder links von der Gerade $x = x_0$ liegt (oder die zulässige Region unbeschränkt oder leer ist).

Hierzu bestimmt man den minimalen Wert der oberen und den maximalen Wert der unteren Schranken in x_0. Ist zum Beispiel die kleinste obere Schranke $(-a_i x_0 - c_i)/b_i$ kleiner als die größte untere Schranke $(-a_j x_0 - c_j)/b_j$ und gilt für ihre Steigungen $-a_i/b_i < -a_j/b_j$, dann liegt die zulässige Region links von x_0, falls sie nicht leer ist (Abbildung 3.4).

Wir wollen nun in jedem Iterationsschritt den Wert von x_0 so wählen, daß, unabhängig davon, ob die Suche rechts oder links von x_0 weitergeführt wird, ein konstanter Bruchteil der Nebenbedingungen als redundant erkannt werden kann. Der Schlüssel hierzu ist folgende Beobachtung: Schneiden sich zwei obere Schranken in einem Punkt mit Abszisse x, ist die eine rechts von x, die andere links von x sicher redundant; analoges gilt für untere Schranken. Dies legt folgende Strategie nahe. Fasse die oberen Schranken beliebig paarweise zusammen, verfahre analog mit den unteren Schranken und berechne den Schnittpunkt eines jeden Schrankenpaars. Wählt man nun x_0 als Median der Abszissen aller dieser Schnittpunkte, so kann in jedem Iterationsschritt etwa ein Viertel der Nebenbedingungen als redundant verworfen werden: liegt etwa der optimale Eckpunkt rechts von x_0, so kann man aus jedem Paar, dessen Schnittpunkt $\tilde{x}$ links von x_0 liegt, die Nebenbedingung streichen, die rechts von $\tilde{x}$ redundant ist.

Die Prune-and-Search Technik kann man in ähnlicher Weise auf mehrdimensionale lineare Programme anwenden. Zusammenfassend ergibt sich der

Satz 6 (Megiddo) *Ein lineares Programm mit n Nebenbedingungen kann in optimaler Zeit $\Theta(n)$ gelöst werden, wenn die Dimension d fest ist.*

Wir sehen also, daß die Divide-and-Conquer-Technik zur Lösung des MaxMin-Problems linearer Funktionen nicht optimal ist.

Setzt man sich aber das Ziel, die untere Einhüllende einer Schar linearer Funktionen explizit zu konstruieren, ist die Situation eine andere. Verlangt man von einem Algorithmus, daß er wie das Divide-and-Conquer Verfahren die Minimierungssequenz der Funktionen von links nach rechts ausgibt, kann man keine bessere Laufzeit als $\Omega(n \log n)$ erwarten, da ein solcher Algorithmus zum Sortieren reeller Zahlen benutzt werden kann [PS85]. (Man ordnet z. B. jeder Zahl x_i die Gerade $y = 2x_i x - x_i^2$ - dies ist die Tangente an den Graphen von $y = x^2$ im Punkt (x_i, x_i^2) - zu.)

Die lineare Beschreibungskomplexität der unteren Einhüllenden legt die Frage nahe, ob eine andere Datenstruktur existiert, die eine effizientere Berechnung zuläßt. Eine solche sollte zumindest die Anfrage, welche der Funktionen in der Minimierungssequenz auftreten, in linearer Zeit unterstützen. Es zeigt sich jedoch, daß bereits diese Forderung eine untere Schranke von $\Omega(n \log n)$ impliziert.

Zum Beweis dieser Behauptung formulieren wir zunächst ein Hilfsproblem.

PROBLEM 8 : [Extrempunkte]

Gegeben: Eine Menge $S = \{p_i \mid i \in I\}$, $p_i = (x_i, y_i)$ von n Punkten in der Ebene.

Gesucht: Die Menge der Indizes $J \subseteq I$ der Punkte p_j, für die gilt

$$\text{conv}(S) \neq \text{conv}(S \setminus \{p_j\})$$

(Die Extrempunkte sind gerade die Ecken des Polygonzuges $\text{CH}(S)$.)

Ben-Or zeigt in [Be83], daß (P8) im algebraischen Entscheidungsbaummodell Komplexität $\Theta(n \log n)$ besitzt. Wir zitieren es, da man es in linearer Zeit auf das

Problem 9 : [untere Einhüllende linearer Funktionen]

Gegeben: Eine Schar $\mathcal{F}_I = \{f_i \mid i \in I\}$ von n linearen Funktionen. (Als Beschreibung der Funktionen f_i seien Paare (a_i, b_i) gegeben, so daß $f_i(x) = a_i x + b_i$ für alle $x \in \mathbb{R}$ gilt.)

Gesucht: Die Menge der Indizes $J \subseteq I$ der f_j, die in der Minimierungssequenz von $\mathcal{F}$ auftreten.

reduzieren kann. Damit gilt

Lemma 7 *Im algebraischen Entscheidungsbaummodell benötigt man zur Lösung von (P9) Zeit* $\Omega(n \log n)$.

Beweis: Sei S eine Instanz von (P8). Wir legen den Ursprung des Koordinatensystems so, daß er in $\text{conv}(S)$ enthalten ist (z. B. in den Schwerpunkt dreier Punkte aus S). Dies ist in linearer Zeit möglich. Betrachte nun zu allen Vektoren $u = (u_x, u_y) \in \mathbb{R}^2$ die Ursprungsgerade g_u mit Normalenvektor u. Für alle Punkte p_i bestimmt das Skalarprodukt $<p_i, u> = x_i u_x + y_i u_y$ den vorzeichenbehafteten, mit $1/|u|$ gewichteten Abstand von p_i zur Gerade g_u, d. h.

$$\delta(g_u, p_i) = \frac{1}{|u|} |<p_i, u>|$$

Liegt p_i auf $\text{CH}(S)$, so existiert eine Tangente an $\text{CH}(S)$ durch p_i. Ist u ein Normalenvektor dieser Tangente, und liegt p_i in der durch die Ursprungsgerade g_u beschränken Halbebene, in die u weist, so maximiert p_i den vorzeichenbehafteten Abstand zu g_u unter allen Punkten aus S. Da die Umkehrung ebenso gilt, folgt

$$\exists u \in \mathbb{R}^2 : \ \frac{1}{|u|} <p_i, u> \ \geq \ \frac{1}{|u|} <p_j, u> \ \forall j \in I \iff p_i \in \text{CH}(S)$$

Die linke Seite ist äquivalent zu

$$x_i \geq x_j \ \forall j \in I \ \vee \ x_i \leq x_j \ \forall j \in I \qquad (A)$$

oder

$$\exists x \in \mathbb{R} \quad x_i \cdot x + y_i \geq x_j \cdot x + y_j \; \forall j \in I \quad (B)$$

oder

$$\exists x \in \mathbb{R} \quad x_i \cdot x - y_i \geq x_j \cdot x - y_j \; \forall j \in I \quad (C)$$

(A) beschreibt die Fälle $u_y = 0$, (B) die Fälle $u_y > 0$, (C) die Fälle $u_y < 0$ (nach Division durch $|u_y|$). Definiert man nun die Funktionenscharen $\mathcal{F}_+ = \{f_i^+ \mid i \in I\}$ mit $f_i^+(x) = -(x_i \cdot x + y_i)$ und $\mathcal{F}_- = \{f_i^- \mid i \in I\}$ mit $f_i^-(x) = -(x_i \cdot x - y_i)$ (durch Negieren der Ausdrücke erhalten wir wie gewünscht Aussagen über untere Einhüllende), so gilt:

Ein Punkt p_i liegt genau dann auf CH(S), wenn er x-extremal ist (Fall A) oder die Funktion f_i^+ auf $f^u[\mathcal{F}_+]$ (Fall B) bzw. die Funktion f_i^- auf $f^u[\mathcal{F}_-]$ (Fall C) auftritt, d. h. in mindestens einem Punkt $x \in \mathbb{R}$ aktiv ist.

Die Eckpunkte der CH(S) zeichnen sich nun dadurch aus, daß sie eine der Ungleichungen der Fälle (A) - (C) in einem $x \in \mathbb{R}$ strikt erfüllen. Da dies wegen der Stetigkeit der Funktionen auch in einer Umgebung von x gilt, existiert ein nichtentartetes Intervall, in dem die Funktion f_i^- (bzw. f_i^+) als einzige aktiv ist; sie tritt damit im Minimierungsdiagramm der entsprechenden Funktionenschar auf.

Antwortet ein Algorithmus zur Lösung von (P9) angesetzt auf $\mathcal{F}_+$ mit der Menge J^+ und angesetzt auf $\mathcal{F}_-$ mit J^-, so ist $J^- \cup J^+$ die gesuchte Indexmenge der Extrempunkte. ∎

Der Beweis illustriert die enge Beziehung zwischen unteren Einhüllenden und konvexen Hüllen, die sich auch in höheren Dimensionen wiederfindet und von verschiedenen Autoren, darunter [SS90], diskutiert wird.

3.5 Anwendung der Resultate

In diesem Abschnitt werden wir unter Verwendung der gefundenen Verfahren effiziente Algorithmen für die Probleme des Kapitelanfangs angeben und ihre Optimalität zeigen.

Zur Berechnung des **Maximin-Punktes auf der reellen Achse (P4)** betrachten wir die Schar der zugehörigen quadrierten Abstandsfunktionen $F_i(x) = w_i^2(x - x_i)^2$. Diese erfüllen als Parabeln die Bedingungen des Abschnitts 3.3.1, besitzen keine lokalen Maxima und schneiden sich paarweise in ≤ 2 Punkten. Folglich kann das MaxMin-Problem dieser Funktionenschar und mit diesem (P4) in optimaler Zeit $O(\lambda_2(n) \log n) = O(n \log n)$ gelöst werden.

Wenden wir uns nun wieder der **Maximin-Ursprungsgerade in der Ebene (P5)** zu. Die quadrierten Abstandsfunktionen $F_i(k) = (\sqrt{1-k^2}x_i - ky_i)^2$ weisen im allgemeinen

(d. h. für $x_i, y_i \neq 0$) einen Tiefpunkt bei

$$k = sign\left(\frac{y_i}{x_i}\right)\sqrt{\frac{x_i^2}{x_i^2+y_i^2}}$$

und einen Hochpunkt bei

$$k = sign\left(\frac{-x_i}{y_i}\right)\sqrt{\frac{y_i^2}{x_i^2+y_i^2}}$$

auf. Geometrisch sind diese beiden Extremlagen dadurch charakterisiert, daß im ersten Fall die Ursprungsgerade durch den Punkt p_i verläuft und im zweiten Fall senkrecht auf dessen Ortsvektor steht. Die Graphen zweier Abstandsfunktionen F_i und F_j schneiden sich in höchstens zwei Punkten, deren Abszissen gerade die Lösungen der Gleichung

$$(\sqrt{1-k^2}x_i - ky_i)^2 - (\sqrt{1-k^2}x_j - ky_j)^2 = 0$$

sind. Gilt $|x_i| \neq |x_j|$ finden wir

$$k_{1,2} = sign\left(\frac{y_i \pm y_j}{x_i \pm x_j}\right)\sqrt{\frac{(x_i \pm x_j)^2}{(x_i \pm x_j)^2 + (y_i \pm y_j)^2}}$$

Für $|x_i| = |x_j|$ ist der definierte der beiden Terme $k_{1,2}$ sowie $k = 0$ Schnittstelle. Geometrisch gesehen ist eine Ursprungsgerade genau dann von zwei Punkten p_i und p_j gleichweit entfernt, wenn sie durch den Mittelpunkt der Strecke $\overline{p_i p_j}$ oder parallel zu ihr verläuft. Abbildung 3.5 veranschaulicht die gezeigten Extremal- und Schnitteigenschaften.

Die asymptotische Laufzeit des Divide-and-Conquer Algorithmus beträgt somit $O(n \log n)$.

Wir zeigen nun, daß der gefundene Algorithmus optimal ist. Der Beweis verläuft in ähnlicher Weise wie beim Maxgap-Problem: wir beginnen mit einem Entscheidungsproblem.

Problem 10 : [Gleichförmige Lücken auf dem Einheitskreis mit Randpunkten]

Gegeben: Eine Menge $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ von n Punkten auf dem ersten Quadranten des Einheitskreises und ein $\epsilon \in \mathbb{R}$.

Entscheide: Sind die euklidischen Abstände zwischen benachbarten Punkten aus der Menge $T = S \cup \{(0,1), (1,0)\}$ alle gleich ϵ ? (Zwei Punkte heißen benachbart, wenn ihre Abszissen aufeinanderfolgen.)

Lemma 8 *Im algebraischen Entscheidungsbaummodell benötigt man zur Lösung von (P10) Zeit* $\Omega(n \log n)$.

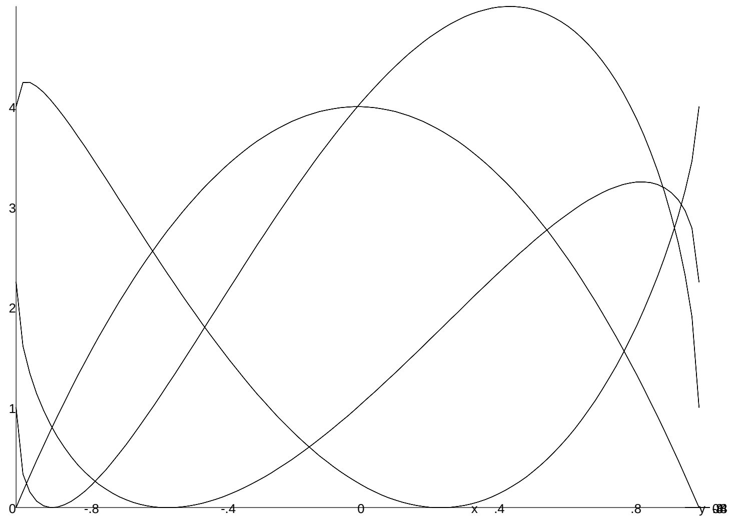

Abbildung 3.5: Abstandsfunktionen für vier Punkte mit optimaler Lösung bei $k \approx 0.8$

Der Beweis lehnt sich an eine Beweisidee in [LW86] an.

Beweis: Betrachte zunächst die Menge $W = \{(x_1, \ldots, x_n) \mid (x_1, \ldots, x_n) \in [0,1]^n$ und $\exists$ Permutation σ, so daß $x_{\sigma(i)} < x_{\sigma(i+1)}$ und

$$(x_{\sigma(i+1)} - x_{\sigma(i)})^2 + (\sqrt{1 - x^2_{\sigma(i+1)}} - \sqrt{1 - x^2_{\sigma(i)}})^2 = \epsilon^2 \quad \forall\, 1 \leq i < n\}$$

Gilt $0 < \epsilon \leq 2\sin[\pi/4(n-1)]$, so bildet für jede Permutation σ die Menge $W_\sigma = \{(x_1, \ldots, x_n) \mid (x_1, \ldots, x_n) \in [0,1]^n$ und $x_{\sigma(i)} < x_{\sigma(i+1)}$ sowie

$$(x_{\sigma(i+1)} - x_{\sigma(i)})^2 + (\sqrt{1 - x^2_{\sigma(i+1)}} - \sqrt{1 - x^2_{\sigma(i)}})^2 = \epsilon^2 \quad \forall\, 1 \leq i < n\}.$$

eine Zusammenhangskomponente von W. Ein Algorithmus zur Lösung von (P10) entscheidet das Element-von-Problem einer Menge V, die eine echte Teilmenge der Menge W bildet: es sind genau die Tupel $(x_1, \ldots, x_n)$ in V enthalten, die neben den Eigenschaften der Tupel aus W und die zusätzliche Bedingungen

$$x^2_{\sigma(1)} + (1 - \sqrt{1 - x^2_{\sigma(1)}})^2 = \epsilon^2 \text{ und } (1 - x_{\sigma(n)})^2 + 1 - x^2_{\sigma(n)} = \epsilon^2$$

erfüllen. Aus $V \subset W$ folgt unmittelbar $\#V \geq \#W = n!$ und mit Satz 1 die Behauptung. ∎

Dieses Entscheidungsproblem bildet die Basis einer untere Schranke des

Problem 11 : [Größte Lücke auf dem Einheitskreis mit Randpunkten]

Gegeben: Eine Menge $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ von n Punkten auf dem ersten Quadranten des Einheitskreises.

Gesucht: Maximaler euklidischer Abstand zweier benachbarter Punkte der Menge $T = S \cup \{(0,1), (1,0)\}$.

Beweis: Wir reduzieren (P10) auf (P11). Sei (S, ϵ) eine Instanz des (P10). Ein Algorithmus zu (P11) bestimmt die größte Lücke $maxgap$ zwischen den Punkten aus T. Ist $\epsilon \neq maxgap$ antworten wir mit 'Nein'. Ansonsten tragen wir, beginnend im Punkt $(0,1)$ den Abstand ϵ auf dem ersten Quadranten des Einheitskreis $n+1$-mal wie mit einem Zirkel ab. Genauer gesagt bestimmen wir, startend mit $m = (0,1)$, den Schnittpunkt s des Kreises um m mit Radius ϵ mit dem Einheitskreis im ersten Quadranten, der größere Abszisse als m hat. Existiert kein solcher, antworten wir mit 'Nein'. Ansonsten setzen wir $m = s$ und wiederholen den Vorgang. Liegt nach $n+1$ Iterationen s im Punkt $(1,0)$, antworten wir mit 'Ja', ansonsten mit 'Nein'. ∎

(P11) reduzieren wir nun auf das Problem der Maximin-Ursprungsgerade (P5).

Aus einer Instanz $S = \{p_i \mid i \in I\}$, $p_i = (x_i, y_i)$ des (P11) konstruiert man eine Eingabemenge T des (P5), indem man alle Punkte aus S und deren Spiegelpunkte bezüglich der x-Achse, der y-Achse und des Ursprungs mit den Schnittpunkten des Einheitskreises mit den Koordinatenachsen vereinigt.

$$T = S \cup \bigcup_{i \in I} \{(x_i, -y_i), (-x_i, y_i), (-x_i, -y_i)\} \cup \{(0,1), (1,0), (0,-1), (-1,0)\}$$

Ein Algorithmus zur Lösung von (P11) berechnet nun die euklidische Maximin-Ursprungsgerade $g^* : a^*x + b^*y = 0$ dieser $\leq 4n+4$ Punkte. Aus dieser Geraden kann man die größte Lücke auf dem Einheitskreis mit Randpunkten leicht in linearer Zeit bestimmen: suche einen Punkt aus T, der minimalen Abstand zur Geraden g^* hat. Das Doppelte dieses Abstandes ist die gesuchte größte Lücke.

Wir fassen unsere Ergebnisse zusammen zum

Lemma 9 *Im algebraischen Entscheidungsbaummodell hat das Problem der Maximin-Ursprungsgerade (P5) Komplexität* $\Theta(n \log n)$.

3.6 Varianten der Problemstellungen

Maximin-Ursprungsgerade bei additiv gewichteter Abstandsfunktion

Eine einfache Problemtransformation wie bei multiplikativen Gewichten ist im Fall *additiv gewichteter* Abstandsfunktionen

$$\mathrm{dist}_A(g, p_i) = \delta(g, p_i) - r_i$$

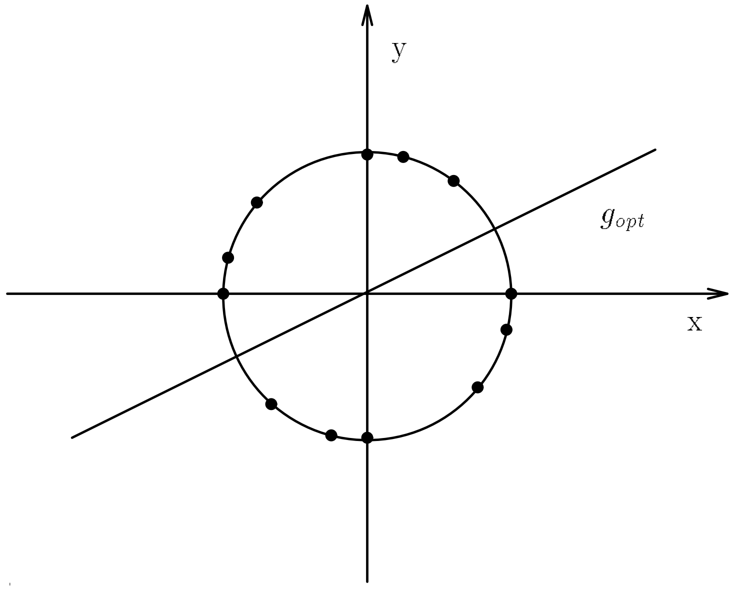

Abbildung 3.6: Reduktion von (P11) auf (P5)

mit $R = \{r_1, \ldots, r_n\} \subset \mathbb{R}$ nicht bekannt. Wir können jedoch die Divide-and-Conquer Technik auch auf diese Funktionenschar anwenden. Die Extrema dieser Funktionen liegen offensichtlich an den gleichen Stellen wie im ungewichteten Fall; sie können sich aber paarweise in bis zu 4 Punkten schneiden. Dies impliziert einen Algorithmus mit Laufzeit $O(\lambda_4(n) \log n) = O(n2^{\alpha(n)} \log n)$. Sind alle Gewichte r_i positiv und interpretiert man die Weltpunkte als Mittelpunkte von Kreisen mit Radius r_i, so modelliert dieses additiv gewichtete Maximin-Problem die Aufgabe, jene Ursprungsgerade zu finden, die den minimalen Abstand zu einer Menge von Kreisen maximiert.

Ein weiteres verallgemeinertes Abstandsmaß ist der *Laguerre-Abstand*

$$\mathrm{dist}_L(g, p_i) = \delta^2(g, p_i) - r_i^2,$$

wobei $R = \{r_1, \ldots, r_n\}$ wiederum eine Menge additiver reeller Gewichte darstellt. Der Vorteil dieser modifizierten Abstandsfunktion ist, daß sie die gleichen Extremal- und Schnitteigenschaften aufweist wie der ungewichtete Abstand $\delta(g, p_i)$: die Funktionen besitzen ein lokales Maximum und schneiden sich paarweise in höchstens zwei Punkten. Somit ergibt sich für den Divide-and-Conquer Algorithmus wie im ungewichteten Fall die optimale Laufzeit $O(n \log n)$. Der Laguerre-Abstand erlaubt im Fall $\delta^2(g, p_i) > r_i^2$ eine anschauliche geometrische Interpretation: Sei l der Fußpunkt des Lotes von p_i auf g, K der Kreis um p_i mit Radius r_i und b der Berührpunkte einer der (beiden) Tangenten durch l an den Kreis K. Dann ist $\mathrm{dist}_L(g, p_i)$ nach dem pythagoreischen Lehrsatz gleich $|lb|^2$.

Maximin-Ursprungsstrahl

Nachdem wir verschiedene gewichtete Abstandsfunktionen betrachtet haben, verlassen wir kurzzeitig den Rahmen der allgemeinen Problemstellung und modifizieren die Gestalt des Zielobjekts: wir optimieren die Lage eines Strahls, dessen Endpunkt im Ursprung liegt.

Problem 12 : [Maximin-Ursprungsstrahl]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\}$, $p_i = (x_i, y_i)$ von n Punkten der euklidischen Ebene.

Gesucht: Ein Ursprungsstrahl s_u^* mit Richtungsvektor $u^* \in \mathbb{R}^2$, der den minimalen euklidischen Abstand zur Punktmenge S maximiert.

Der euklidische Abstand eines Punktes $p_i \in S$ zu s_u ist gegeben durch

$$\delta(p_i, s_u) = \begin{cases} \delta(p_i, g_u) = \frac{1}{|u|}|u_y x_i - u_x y_i| & \text{falls } <p_i, u> \geq 0 \\ \delta(p_i, 0) = \sqrt{x_i^2 + y_i^2} & \text{sonst} \end{cases}$$

wobei g_u für die Gerade mit Richtungsvektor u steht. Wir betrachten nun den Fall, daß der Strahl in die rechte offene Halbebene $x > 0$ fällt. (Die Behandlung der linken offenen Halbebene $x < 0$ geschieht völlig symmetrisch, die der Grenzlage $x = 0$ ist trivial und erfordert nur konstante Zeit.) Dann gilt $u_x > 0$ und wir können die Steigung $m = u_y/u_x$ einführen, um ein Problem in einer Variablen zu erhalten. Der Graph einer jeden Abstandsfunktion $f_i(m)$ setzt sich (falls $y_i \neq 0$) aus zwei Teilen zusammen, welche sich im Punkt $m = -x_i/y_i$ berühren:

$$f_i(m) = \begin{cases} \dfrac{|mx_i - y_i|}{\sqrt{m^2+1}} & \text{falls } \quad (m \geq -x_i/y_i \ \wedge \ y_i > 0) \\ & \qquad\qquad \vee \ (m < -x_i/y_i \ \wedge \ y_i < 0) \\ \sqrt{x_i^2 + y_i^2} & \text{sonst} \end{cases}$$

Teilen wir nun jedes f_i in diese zwei über disjunkten Definitionsbereichen erklärten Teilfunktionen auf, erhalten wir insgesamt $2n$ Funktionen, von denen jede über einem einseitig unbegrenzten Intervall definiert ist. Die Anzahl der paarweisen Schnittpunkte dieser Funktionen ist durch 2 begrenzt. Die n konstanten Funktionen sind in jedem Punkt extremal; es genügt somit, ihre Schnittpunkte mit anderen Funktionen zu betrachten. Die Maxima der variablen Funktionen liegen gerade in den Punkten $m = -x_i/y_i$. Der Divide-and-Conquer-Algorithmus aus 3.3.4 kann die untere Einhüllende dieser Funktionenschar in Zeit $O(\lambda_4(2n) \log n) = O(\lambda_4(n) \log n)$ berechnen. Da sich je zwei partielle Funktionen zu einer über der gesamten reellen Achse definierten Funktion zusammenfügen, ist die MaxMin-Aufgabe dieser Funktionenschar wohldefiniert und aufgrund des bisher Gesagten in Zeit $O(\lambda_4(n) \log n)$ zu lösen.

Berücksichtigt man zusätzlich, daß alle Teilfunktionen über einem einseitig unbegrenzten Intervall definiert sind, kann man diese Schranke etwas verbessern. Die maximale Länge einer alternierenden Subfolge $i..j..i..$ der Minimierungssequenz dieser Funktionenschar ist durch 4 und nicht wie im Fall allgemeiner Lage der Intervalle durch 5 begrenzt ist. Somit ist die Minimierungssequenz eine DS$(3, 2n)$ und der Algorithmus läuft in Zeit $O(\lambda_3(n) \log n) = O(n\alpha(n) \log n)$.

Die gezeigten Beispiele verdeutlichen die Allgemeinheit des Ansatzes: zur Lösung eines eindimensionalen MaxMin-Problems genügt es, die Lage der lokalen Maxima sowie die Anzahl der Schnittpunkte je zweier Funktionen der Schar der Distanzfunktionen zu bestimmen, sofern diese den Anforderungen aus Abschnitt 3.3.1 genügen.

Das Problem der Maximin-Ursprungsgerade und seine Varianten sind bisher in der Literatur nicht behandelt worden. Houle und Maciel [HM88] betrachten das nicht verankerte Problem und geben einen $O(n^2)$ Algorithmus an, der den breitesten Korridor durch eine Punktmenge findet.

3.7 Duale Problemstellung

In diesem Abschnitt diskutieren wir solche Probleme, die im Sinne von Kapitel 2 dual zu den bisher untersuchten sind: wir optimieren nicht länger nach dem Maximin-, sondern dem Minimax-Kriterium. Die diesen Problemstellungen entsprechende funktionale Fassung, in der nach dem Minimum der oberen Einhüllenden der Schar der Abstandsfunktionen gefragt wird, hatten wir mit MinMax bezeichnet. Wie bereits in der Bemerkung eingangs des Kapitels festgestellt, kann man zur Lösung von MinMax die gleichen Techniken heranziehen, die beim dualen MinMax erfolgreich waren.

Daß dieses Vorgehen jedoch nicht optimal sein muß, zeigt schon das zu Maxgap duale Mincover-Problem auf der reellen Achse: der Mittelpunkt des kleinsten eine Punktmenge S überdeckenden Intervalls ist natürlich $(x_{max} + x_{min})/2$ und in linearer Zeit zu finden, wohingegen ein Divide-and-Conquer Algorithmus Zeit $O(n \log n)$ benötigen würde.

Die Tatsache, daß bereits die Komplexität der einfachsten Maximin-Aufgabe die ihrer dualen Minimax-Aufgabe übersteigt, ist ein erster Beleg für die in Abschnitt 2.3 ausgesprochene Vermutung, daß ein Maximin-Problem inhärent mindestens so schwierig ist, wie das duale Minimax-Problem.

Ein Beispiel, indem in beiden Fällen gleiche Komplexität auftritt, sind die Minimax- und Maximin-Ursprungsgeraden. Lee und Wu zeigen in [LW86] durch Reduktion des Problems der größten Lücke auf dem Einheitskreis eine untere Schranke von $\Omega(n \log n)$ für das Problem der Minimax-Ursprungsgerade.

Läßt man Geraden in beliebiger Lage zu, gelangt man zum

Problem 13 : [Minimax-Gerade in der Ebene]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\}$, $p_i = (x_i, y_i)$ von n Punkten der euklidischen Ebene und eine Menge $W = \{w_1, \ldots, w_n\}$ von positiven reellen Gewichten.

Gesucht: Eine Gerade $g^* : a^*x + b^*y + c^* = 0$, die den maximalen gewichteten euklidischen Abstand zur Punktmenge S minimiert.

Im ungewichteten Fall bezeichnet man das Doppelte der maximalen, zwischen der optimalen Geraden g^* und S auftretenden Distanz als die *Weite der Punktmenge S*. Im $d-$dimensionalen Raum definiert man die Weite einer Menge S analog als minimale Breite einer von zwei parallelen Hyperebenen begrenzten Schicht, die S enthält.

Lee und Wu beschreiben einen Algorithmus zur Bestimmung der ungewichteten Minimax-Geraden in der Ebene mit optimaler Zeitkomplexität $\Theta(n \log n)$. Dieser beruht auf der Beobachtung, daß die optimale Gerade stets parallel zu einer Kante der konvexen Hülle der Menge S verläuft und drei Punkte aus S maximale Distanz zu ihr haben. Nach Konstruktion der konvexen Hülle genügt ein Umlauf, um alle Kandidatengeraden zu ermitteln. (Im verankerten Fall liefert auch die Divide-and-Conquer Technik das Resultat in optimaler Zeit.)

Ein Zusammenhang zwischen der Konstruktion von konvexen Hüllen und der Weite von Punktmengen ist auch in höheren Dimensionen gegeben. Dies gilt sogar dann, wenn man die Punkte mit multiplikativen Gewichten versieht. Man kennt Algorithmen zur Berechnung der Weite einer multiplikativ gewichteten Punktmenge in d Dimensionen, deren Laufzeit von der Zeit zur Berechnung konvexer Hüllen in d Dimensionen dominiert wird [HIIRY90].

Kapitel 4

Punktförmige Zielobjekte in der Ebene

4.1 Motivation

In diesem Kapitel stellen wir uns die Aufgabe, ein punktförmiges Objekt bezüglich einer Menge von Weltpunkten in der euklidischen Ebene optimal zu positionieren. Ausgangspunkt unserer Betrachtungen ist wiederum die Optimierung nach dem Maximin-Kriterium (dies ist also die Instanz $d = 2,\ k = 0$ der allgemeinen Problemstellung), wir werden aber auch auf das duale Minimax-Kriterium eingehen. In der Operations Research Literatur ist diese Problematik unter der Bezeichnung *maximin (minimax) single facility location* bekannt; wir haben sie bereits in der Einleitung als einführendes Beispiel vorgestellt. Motiviert durch die Fülle der in der Praxis auftretenden Anwendungen, die sich adäquat durch diese Modelle beschreiben lassen, existiert eine Reihe von Beiträgen zu diesem Stichwort (vgl. [LMW88], [SM92]) Die Wahl des Maximin-(Minimax-)Kriteriums ist immer dann angebracht, wenn eine gute Problemlösung nicht durch den mittleren Abstand zu den Weltpunkten bestimmt wird, sondern der kleinste (größte) auftretende Abstand entscheidend ist. Eine Aufstellung praktischer Anwendungsbeispiele verdeutlicht diese Beobachtung:

1. Maximin-Kriterium

   (a) Plazieren eines neuen Unternehmens, das Wettbewerb mit bereits bestehenden, ähnlichen Unternehmen weitestgehend vermeiden will.

   (b) Plazieren einer Lärm- oder Schadstoffquelle, um schädliche Einwirkungen auf umliegende Regionen zu minimieren.

   (c) Plazieren von Maschinen innerhalb einer Fabrik, so daß die Belastung und Gefährdung der Belegschaft so gering wie möglich wird.

2. Minimax-Kriterium

   (a) Plazieren von Notfalleinrichtungen (Rettungsstationen, Feuerwehren), so daß kürzestmögliche Rettungswege erreicht werden können.

   (b) Plazieren eines Liefer- oder Reparaturservices, der schnellstmögliche Einsatzzeiten garantieren soll.

   (c) Plazieren eines Radiosenders, so daß die nötige Sendeleistung minimiert wird.

In allen Fällen gibt der euklidische Abstand ein hinreichend genaue Beschreibung der Distanz zwischen Welt- und Zielpunkt, so daß wir uns bei den folgenden Überlegungen auf dieses Abstandsmaß beschränken können.

In der Praxis treten jedoch Situationen auf, in denen die Weltpunkte nicht uniform zu behandeln sind, sondern ihnen unterschiedlich große Bedeutung zuzuordnen ist. So erzielen die Unternehmen im ersten Aufzählungspunkt sicherlich unterschiedliche Umsätze oder die Wahrscheinlichkeit eines Notfalls in Punkt 2.(a) variiert von Stadtgebiet zu Stadtgebiet. Um dennoch ein ausreichend genaues Bild der Realität zeichnen zu können, ordnen wir den Weltpunkten positive Gewichtsfaktoren zu, die die unterschiedliche Bedeutung der Weltpunkte widerspiegeln.

Somit gelangen wir zur formalen Fassung des

PROBLEM 14 : [Maximin-Punkt in der Ebene]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\}$; $p_i = (x_i, y_i)$, von Punkten in der reellen Ebene $\mathbb{R}^2$, eine Menge $W = \{w_1, \ldots, w_n\}$ von positiven, reellen Gewichtungsfaktoren, sowie eine kompakte Region $M \subset \mathbb{R}^2$.

Gesucht: Ein Punkt $p^* = (x^*, y^*) \in M$, der

$$\min_{1 \leq i \leq n} w_i \sqrt{(x - x_i)^2 + (y - y_i)^2}$$

maximiert.

Die Lagebeschreibung des Punktes kann durch seine kartesischen Koordinaten erfolgen; Objektraum und Konfigurationsraum fallen hier zusammmen. Die Einschränkung auf einen endlichen zulässigen Bereich ist erforderlich, um eine endliche Lösung zu erzwingen; die Kompaktheit von M wurde gefordert, da sie zusammen mit der Stetigkeit der Abstandsfunktionen die Existens eines Maximums impliziert und das Problem somit wohldefiniert ist.

4.2 Größter leerer Kreis

Im ungewichteten Fall hat das Problem eine anschauliche geometrische Interpretation. Man sucht nach dem Mittelpunkt $p^* \in M$ des größten Kreises K der keinen Punkt von S umfaßt (d. h. $int(K) \cap S = \emptyset$). In der Literatur findet man für diese Aufgabe meist die Bezeichnung *(largest empty circle)*.

Sofern keine andere Festlegung getroffen wird, sei die zulässige Region M die konvexe Hülle $c$conv(S) der Menge der Weltpunkte.

Ein trivialer Ansatz zur Lösung dieses Problems beruht auf der Beobachtung, daß auf dem optimalen Kreis 2 oder 3 Punkte der Menge S liegen, und darüber hinaus im ersten Fall der Mittelpunkt p^* des Kreises auf den Rand CH(S) der konvexen Hülle fallen muß (vgl. Abbildung 4.1).

Ein einfacher Algorithmus zählt alle $\binom{n}{2}$ Punktepaare und $\binom{n}{3}$ Punktetripel nacheinander auf und prüft, ob die durch diese Punkte bestimmten Kreise keinen Punkt aus S umfassen. (Nach Vorberechnung von CH(S) in Zeit $O(n \log n)$ kann dies in linearer Zeit pro Kreis geschehen.) Der Mittelpunkt des größten unter den zulässigen Kreisen ist die gesuchte Lösung. Der Algorithmus benötigt Zeit $O(n^4)$. Daß dies bei weitem nicht die bestmögliche Lösung des Problems ist, zeigt ein klassisches Resultat von Toussaint [To83].

Satz 7 (Toussaint) *Sind alle Gewichte gleich und wählt man die konvexe Hülle conv(S) der Punktmenge S als zulässige Region M, so läßt sich (P14) in optimaler Zeit $\Theta(n \log n)$ lösen.*

Der Lösungsansatz von Toussaint bildet den Ausgangspunkt unserer späteren Betrachtungen. Aus diesem Grund werden wir uns mit den zugrundeliegenden Konzepten eingehend befassen.

Der Algorithmus basiert auf einer fundamentalen Datenstruktur der algorithmischen Geometrie, dem *Voronoidiagramm* für Punktmengen. (vgl. [Au87] für eine umfassende Übersicht der möglichen Anwendungen des Voronoidiagramms).

Das Voronoidiagramm einer Punktmenge S (kurz VD$[S]$) teilt die Ebene gemäß der *Nächster-Nachbar-Regel* auf: jedem Punkt wird der Teil der Ebene zugeordnet, dessen Punkte näher zu ihm als zu den anderne Punkten von S liegen. Die folgende formalere Definition gibt Aufschluß über die Struktur des Diagramms.

Betrachtet man zwei Punkte p_i und p_j der Menge S, so ist die Menge der Punkte, die näher zu p_i als zu p_j liegt, die durch die Mittelsenkrechte der beiden Punkte begrenzte offene Halbebene $H(p_i, p_j)$. Die Punkte der Ebene, die näher zu p_i als zu einem anderen Punkt aus S liegen, bilden als Schnitt von $n-1$ Halbebenen ein konvexes Polygon,

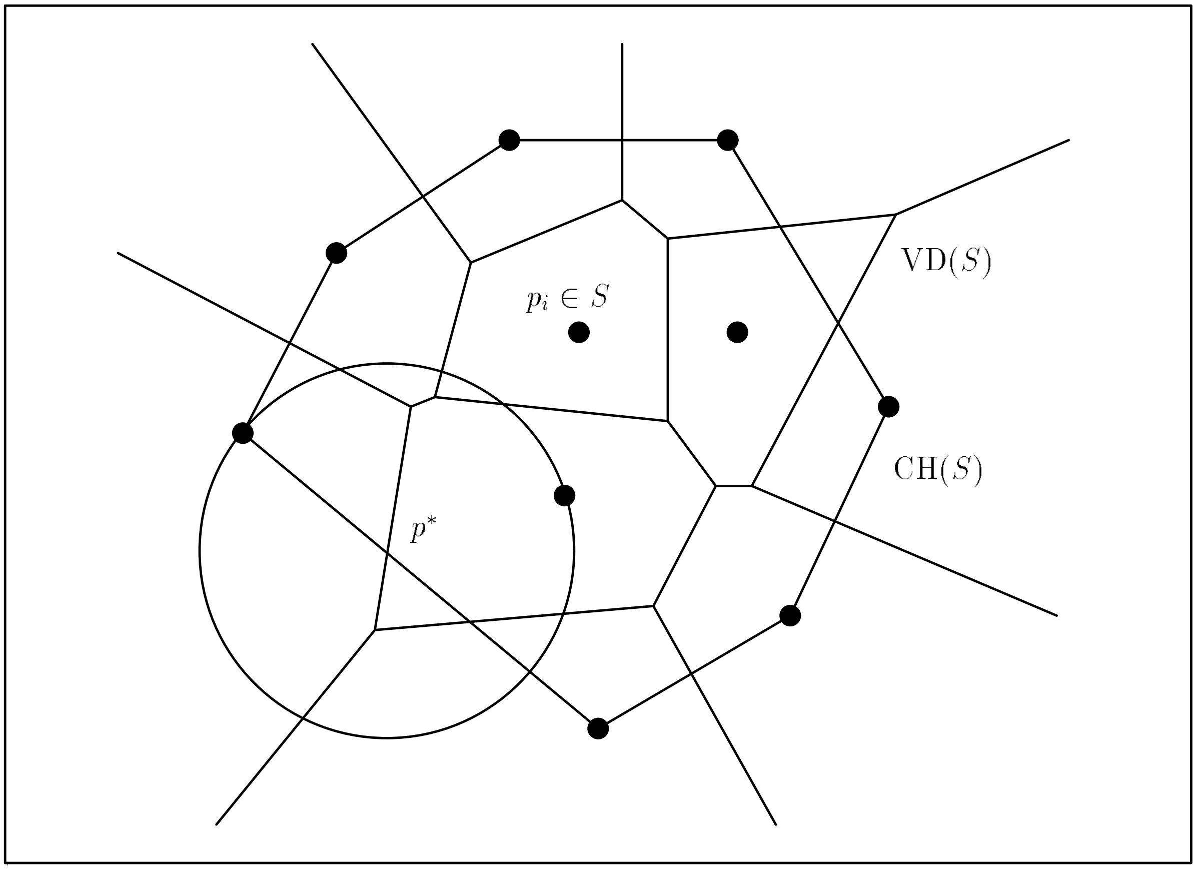

Abbildung 4.1: Punktmenge S mit Voronoidiagramm und größtem leeren Kreis

das Voronoi-Polygon $V_i = \cap_{i \neq j} H(p_i, p_j)$. Die n Voronoi-Polygone definieren eine Aufteilung der Ebene in Voronoiregionen, -kanten und -ecken, das Voronoidiagramm von S (VD[S]).

Wir beschreiben nun, wie der größte leere Kreis einer Punktmenge S effizient mit Hilfe ihres Voronoidiagramms gefunden werden kann.

Das Voronoidiagramm der Punktmenge S zerteilt ihre konvexe Hülle conv(S) in n konvexe Polygone, wobei in jedes der Polygone genau ein Punkt der Menge S fällt. Wir bezeichnen das Polygon, das den Punkt p_i enthält, mit Pol_i. (Es gilt natürlich $Pol_i \subset V_i$) Für alle $s \in Pol_i$ ist p_i der nächstgelegene Punkt aus S. (Für $s \in \partial Pol_i$ können mehrere Punkte aus S gleichweit entfernt sein.) Wählt man also einen Punkt $s \in Pol_i$ als Mittelpunkt eines Kreises, der keinen Punkt von S umfassen soll, wird dessen maximal möglicher Durchmesser durch den euklidischen Abstand $\delta(p_i, s)$ bestimmt. Betrachtet man diesen Abstand als Funktion von s über Pol_i, so nimmt dieser als konvexe Funktion über einem konvexen Polygon maximale Werte in den Ecken des

Pol_i an. Somit liegt auch der optimale Kreismittelpunkt in einer solchen Ecke. Jede Ecke eines Polygons Pol_i der Partition ist aber entweder Voronoiknoten des VD(S) oder Schnittpunkt des VD(S) mit CH(S).

Eine Darstellung des VD(S) als *double-connected-edge-list (DCEL)* [PS85] kann in optimaler Zeit $O(n \log n)$ berechnet werden. Daraus kann man alle $O(n)$ Voronoiknoten des Diagramms in linearer Zeit ablesen. Die einfache Struktur des VD(S) ermöglicht es, den Schnitt des VD(S) mit der konvexen Hülle CH(S) effizient zu bestimmen. Hierzu fügen wir dem VD(S) konzeptuell eine weitere Kante hinzu, die es im unendlichen umschließt. Ausgehend von einem beliebigen Schnittpunkt des VD(S) und der CH(S) folgen wir den Kanten des VD(S) entgegen dem Uhrzeigersinn wie den Wänden eines Labyrinths, bis wir auf einen weiteren Schnittpunkt mit CH(S) treffen und von dort einen neuen Umlauf beginnen. Auf diese Weise lassen sich alle Schnittpunkte in Zeit $O(n)$ finden. Schließlich genügt lineare Zeit, um unter den insgesamt $O(n)$ Kandidateneckpunkten den optimalen Kreismittelpunkt herauszufinden. Einzelheiten des Verfahrens kann der Leser der ausführlichen Beschreibung in [PS85] entnehmen.

Die Optimalität des Resultats folgt unmittelbar aus der unteren Schranke des Maxgap-Problems (P2), da dies einfach eine eindimensionale Instanz des Problems des größten leeren Kreises darstellt.

Verallgemeinert man das Konzept des Voronoidiagramms in naheliegender Weise auf Punktmengen im $\mathbb{R}^3$, so kann man einen analogen Algorithmus zur Berechnung der größten leeren Kugel, deren Mittelpunkt in die konvexe Hülle einer gegebenen dreidimensionalen Punktmenge fällt, formulieren.

4.3 Minimierungsdiagramme

Die Verwendung des Voronoidiagramms im vorstehenden Algorithmus weist uns den Weg, wie die Technik des vorangegangenen Kapitels auf Probleme mit zwei Freiheitsgraden zu verallgemeinern ist. Das Voronoidiagramm für Punktmengen ist die geeignete Datenstruktur, um die lokalen Maxima der unteren Einhüllenden der Schar der Abstandsfunktionen $f_i(x) = \delta(p_i, x)$ effizient zu lokalisieren und als solche ist es ein zweidimensionales Äquivalent zu den in Kapitel 3 eingeführten Minimierungssequenzen. Will man nun zweidimensionale Maximin-Probleme von größerer Allgemeinheit betrachten, gilt es, das Konzept des Voronoi-Diagramms entsprechend zu erweitern. Kennzeichnende Charakteristik der zu entwickelnden Datenstruktur muß es sein, daß sie die untere Einhüllende einer Schar reeller Funktionen in zwei Variablen hinreichend genau repräsentiert und sie ein schnelles Auffinden der lokalen Extrema unterstützt.

4.3.1 Definitionen

Wir gehen in Analogie zum eindimensionalen Fall von der Projektion des Graphen der unteren Einhüllenden in die reelle Ebene aus. Man kann diese Projektion auf einfache Weise veranschaulichen. Färbt man die Graphen der Funktionen einer Schar unterschiedlich ein und betrachtet die (undurchsichtigen) Flächen von einem Blickpunkt mit Ordinate $-\infty$, setzt sich das Bild aus einer Vielzahl von Flächenstücken verschiedener Färbung zusammen. Resultat der Projektion ist also eine Partition der Ebene in disjunkte Bereiche. Diese Beobachtung wollen wir nun formal beschreiben. Zu diesem Zweck übertragen wir zunächst den Begriff der aktiven Funktion aus dem zweiten Kapitel.

Definition 6 *Sei $\mathcal{F}_I = \{f_i \mid i \in I\}$ eine endliche Schar stetiger, reellwertiger Funktionen, die über $Def \subseteq \mathbb{R}^2$ erklärt sind. Eine Funktion $f_i \in \mathcal{F}$ heißt im Punkt $x \in Def$* aktiv*, falls $f_i(x) = f^u[\mathcal{F}](x)$.*

Ordnet man nun jedem x des Definitionsbereichs Def die in x aktiven Funktionen zu, zerfällt Def in Gebiete gleicher Markierung.

Definition 7 ([ES86]) *Die Funktion $A_\mathcal{F} : Def \rightarrow \wp(I)$*

$$A_\mathcal{F}(x) = \{i \in I \mid f_i(x) = f^u(x)\}$$

ordnet jedem x die Indexmenge der in x aktiven Funktionen zu. Sie definiert eine Äquivalenzrelation $\rho_\mathcal{F}$ auf Def, in der Weise, daß für $x, y \in Def$ gilt: $x\rho_\mathcal{F}y \iff A_\mathcal{F}(x) = A_\mathcal{F}(y)$. Die durch $\rho_\mathcal{F}$ induzierte Partition von Def heißt verallgemeinertes Voronoi-Diagramm *von $\mathcal{F}$ (kurz vVD[$\mathcal{F}$]); ihre Äquivalenzklassen heißen Voronoizellen:*

$$V_T = \{x \in Def \mid A_\mathcal{F}(x) = T\}$$

Notation: Wir schreiben statt $V_{\{i\}}$ auch V_i.

Daß diese Struktur in der Tat eine Verallgemeinerung des Voronoi-Diagramms für Punktmengen darstellt und ihr Name somit gerechtfertigt ist, sieht man leicht, wenn man als Funktionenschar $\mathcal{F}_I$ die Schar der Distanzfunktionen $f_i(x) = \delta(p_i, x),\ i \in I$ zu einer Punktmenge $S = \{p_i \mid i \in I\}$ betrachtet. In diesem Fall sind VD[S] und vVD[$\mathcal{F}_I$] identisch.

Ein beliebiges vVD kann offensichtlich bis zu $2^n - 1$ Voronoizellen besitzen. Die Voronoizellen V_T sind nicht notwendigerweise zusammenhängend, vielmehr kann eine Voronoizelle in unendlich viele ZHK zerfallen. So besteht das vVD der Funktionen $f_1(x, y) = \sin(x)$ und $f_2(x, y) = \cos(x)$ aus den drei Zellen $V_1, V_2, V_{\{1,2\}}$, jede einzelne von ihnen zerfällt jedoch in unendlich viele ZHK.

Um ein realistisches Maß für die Komplexität solcher Zerlegungen zu erhalten, erscheint es somit angemessener, nicht die Voronoizellen selbst, sondern ihre ZHK zu betrachten.

Definition 8 ([SS90]) *Sei $\mathcal{F}$ eine Schar stetiger Funktionen über Def $\subseteq I\!R^2$. Die Aufteilung von Def in die Zusammenhangskomponenten des vVD[$\mathcal{F}$] heißt* Minimierungsdiagramm *von $\mathcal{F}$ (kurz MD[$\mathcal{F}$]). Die* kombinatorische Komplexität *eines MD ($\mathcal{K}$(MD)) ist die Anzahl dieser ZHK.*

Solche Minimierungsdiagramme können wie gesehen unbeschränkte Komplexität haben.

Um degenerierte Konfigurationen auszuschließen und Diagramme zu erhalten, die die Ebene ähnlich wie die uns vertrauten Voronoidiagramme von Punktmengen in Regionen, Kanten und Knoten zerteilen, stellen wir zusätzliche Forderungen an die Funktionen.

Definition 9 ([SS90]) *Erfüllt eine Schar $\mathcal{F}_I$ reellwertiger, stetig differenzierbarer Funktionen über Def $\subset I\!R^2$ die Bedingungen*

(i.) Der Schnitt zweier Funktionen ist kein 2-dimensionales Flächenstück.

(ii.) Der Schnitt dreier Funktionen ist keine 1-dimensionale Kurve.

(iii.) Es exisistert kein Punkt, in dem sich vier Funktionen schneiden.

(iv.) Das Vorzeichen von $f_i - f_j$ wechselt beim Übergang über die Schnittlinie von f_i und f_j.

so sagt man, sie befinden sich in allgemeiner Lage.

Befindet sich die Funktionenschar $\mathcal{F}$ in allgemeiner Lage, so ist ihr Minimierungsdiagramm MD[$\mathcal{F}$] eine Aufteilung des gemeinsamen Definitionsbereichs *Def* in (bezüglich Mengeninklusion) maximal zusammenhängende

- *Regionen*, in denen nur eine Funktion aktiv ist,
- *Kanten*, auf denen zwei Funktionen aktiv sind,
- *Knoten*, in denen drei Funktionen aktiv sind.

Die Kanten und Knoten des MD einer Funktionenschar in allgemeiner Lage bilden einen planaren Graph mit Knotengrad 3. (Um der Standarddefinition eines planaren Graphen zu genügen, fügen wir konzeptuell einen Knoten im Unendlichen ein, in welchem sich alle unendlichen Kanten treffen.) Für solche Graphen kennt man den

Satz 8 (Euler) *Teilt ein planarer Graph die Ebene in f Regionen, e Kanten und v Knoten, so gilt die Relation*

$$v - e + f = 2$$

Ist der Knotengrad des Graphen ≥ 3, so sind die Größen v, f und e paarweise proportional.

Somit genügt die Betrachtung einer Kenngröße, z. B. der Knotenanzahl, um Aussagen über die asymptotische Komplexität eines MD machen zu können. Ist also z. B. die Anzahl der möglichen Schnittpunkte dreier Funktionen aus $\mathcal{F}$ durch eine Konstante k beschränkt, kann das MD[$\mathcal{F}$] nur $k\binom{n}{3}$ Knoten besitzen und für die Komplexität des Diagramms folgt die obere Schranke $\mathcal{K}(\mathrm{MD}) = O(n^3)$.

Ausschlaggebend für die geometrische Gestalt des Minimierungsdiagramms einer Funktionenschar $\mathcal{F}$ in allgemeiner Lage ist die Form der Kanten des Diagramms, die unter der gegebenen Voraussetzung die Gestalt von Kurvenstücken annehmen. Definiert man den *Separator* sep_{ij} zweier Funktionen $f_i, f_j \in \mathcal{F}_I$ als die Menge

$$sep_{ij} = \{x \in Def \mid f_i(x) = f_j(x)\}$$

wird deutlich, daß jede Kante des MD ein maximal zusammenhängendes Teilstück des Separators zweier Funktionen aus $\mathcal{F}$ ist, auf dem diese beiden Funktionen aktiv sind. Man kann daher Aussagen über die Gestalt der Kanten des MD machen, indem man die die Separatoren sep_{ij} definierenden Kurvengleichungen $f_i(x) - f_j(x) = 0$ analysiert.

4.3.2 Ordnungsäquivalente Funktionenscharen

Für die Berechnung von MD ist der Begriff der Ordnungsäquivalenz zweier Funktionenscharen von großem Nutzen.

Definition 10 ([ES86]) *Seien $\mathcal{F} = \{f_i \mid i \in I\}$ und $\mathcal{G} = \{g_i \mid i \in I\}$ Scharen reellwertiger, stetiger Funktionen über $Def \subseteq \mathbb{R}^2$, dann heißen $\mathcal{F}$ und $\mathcal{G}$ genau dann* ordnungsäquivalent, *wenn für alle $x \in Def$ und für alle $i, j \in I$ gilt:*

$$sign(f_i(x) - f_j(x)) = sign(g_i(x) - g_j(x))$$

Seine Bedeutung zeigt das

Lemma 10 *Sind $\mathcal{F}_I$ und $\mathcal{G}_I$ ordnungsäquivalente Funktionenscharen, so gilt*

$$vVD[\mathcal{F}] = vVD[\mathcal{G}], \qquad \text{und damit} \qquad MD[\mathcal{F}] = MD[\mathcal{G}]$$

Beweis: Das vVD einer Funktionenschar ist durch die von der Funktion $A(x)$ induzierten Relation ρ bestimmt. Es genügt daher zu zeigen: $A_{\mathcal{F}}(x) = A_{\mathcal{G}}(x)$ für alle $x \in Def$. Sei $x \in Def$ beliebig, $i \in A_{\mathcal{F}}(x)$. Dies impliziert $f_i(x) = f^u[\mathcal{F}](x) \leq f_j(x)\ \forall j \neq i$. Aus der Ordnungsäquivalenz folgt $g_i(x) \leq g_j(x)\ \forall j \neq i$, also $g_i(x) = g^u[\mathcal{G}](x)$ und damit $i \in A_{\mathcal{G}}(x)$. Die Rückrichtung zeigt man analog. ∎

Will man also das Minimierungsdiagramm einer Funktionenschar berechnen, kann man sie - beispielsweise um die Rechnung zu vereinfachen - zunächst in eine ordnungsäquivalente Schar transformieren oder durch eine solche ersetzen, ohne das Resultat zu verfälschen.

Der erste Schritt zur gewünschten Datenstruktur ist getan: Minimierungsdiagramme spiegeln die Struktur der unteren Einhüllenden hinreichend genau wider. Es bleibt die Frage, wie man MD effizient berechnet und repräsentiert. Die Antwort hierauf hängt entscheidend von Eigenschaften der betrachteten Funktionenklasse ab; für einige Funktionenklassen werden wir an geeigneter Stelle effiziente Verfahren erläutern.

4.4 Nichtlineare Optimierung

4.4.1 Mathematische Programme

Ziel unserer Überlegungen ist noch immer, eine effiziente Lösungsmethode des in Kapitel 1 besprochenen

PROBLEM : [MaxMin einer Funktionenschar]

Gegeben: Eine Schar $\mathcal{F} = \{f_1, \ldots, f_n\}, f_i : Def \subseteq \mathbb{R}^d \to \mathbb{R}$ stetig differenzierbarer Funktionen.

Gesucht: Abszisse des Maximums der unteren Einhüllenden $f^u[\mathcal{F}]$, sofern es existiert.

für höhere Dimensionen, insbesondere den Fall $d = 2$ zu finden. Einen ersten Schritt haben wir im letzten Abschnitt getan: das Minimierungsdiagramm bietet, sofern es in einer geeigneten Datenstruktur repräsentiert ist, ausreichende Information über die Struktur der unteren Einhüllende zweidimensionaler Funktionenscharen. Im nächsten Schritt müssen wir eine Technik entwickeln, die es erlaubt, die lokalen Maxima der unteren Einhüllenden leicht zu identifizieren, um aus diesen das globale Maximum zu bestimmen. In der eindimensionalen Probleminstanz genügte es, die lokalen Funktionsextrema, die Randpunkte des Definitionsbereichs, sowie die Schnittstellen der Funktionen auf der unteren Einhüllenden aufzuzählen. Dieses Verfahren scheitert für Funktionen in zwei Variablen, da die Schnitte der Graphen zweier Funktionen eindimensionale Gestalt annehmen, wenn wir allgemeine Lage der Funktionen voraussetzen. Weiterhin ist bei den von uns betrachteten partiellen Funktionen der Definitionsbereich durch eine eindimensionale Randlinie begrenzt. Wir müssen also, um unsere enumerative Strategie verallgemeinern zu können, die möglichen Lagen der Extrema weiter einschränken. Die hierfür nötigen Werkzeuge stellt uns die Theorie der *mathematischen Programmierung* zur Verfügung.

Definition 11 *Ein Problem der Form*

PROBLEM : [mathematisches Programm]

> Gegeben: $m+1$ reellwertige Funktionen $f, g_i : \mathbb{R}^d \to \mathbb{R}$ $(1 \le i \le m)$.
>
> Gesucht: Ein Punkt $x^* \in \mathbb{R}^d$, der unter allen Punkten des zulässigen Bereichs $M = \{x \in \mathbb{R}^d \mid g_i(x) \le 0\}$ die Funktion f maximiert.

heißt Optimierungsaufgabe (mathematisches Programm)*. f heißt dabei* Zielfunktion*, die Funktionen g_i* Nebenbedingungen*, x^** Lösung *der Optimierungsaufgabe.*

Sind keine Nebenbedingungen vorgegeben, ist der zulässige Bereich M also der gesamte $\mathbb{R}^d$, spricht man von einem *unrestringierten Problem.*

Man findet auch die abkürzender Schreibweise

$$\begin{array}{rl} & \max f(x) \\ \text{u.d.N.} & g_i(x) \le 0 \quad 1 \le i \le m \end{array}$$

Aufgrund der Identität $\min_{x \in M} f(x) = -\max_{x \in M}(-f(x))$ lassen sich Minimierungsaufgaben in Maximierungsaufgaben transformieren. Weiterhin sind Nebenbedingungen in Gleichungsform, wie $h(x) = 0$ äquivalent zu den Ungleichungen $-h(x) \le 0,\ h(x) \le 0$.

Wir sind bereits einer speziellen Klasse von Optimierungsaufgaben, den linearen Programmen, in denen sowohl die Zielfunktion als auch die Nebenbedingungen lineare Funktionen sind, begegnet, als wir in Abschnitt 3.4 untere Schranken des MaxMin-Problems in einer Variablen diskutierten.

Das MaxMin-Problem einer Funktionenschar $\mathcal{F} = \{f_1, \ldots, f_n\},\ f_i : \mathbb{R}^d \to \mathbb{R}$ läßt sich unmittelbar als unrestringierte Optimierungsaufgabe in d Dimensionen mit $f^u[\mathcal{F}]$ als Zielfunktion beschreiben. Es erweist sich jedoch als vorteilhaft, es mit der Optimierungsaufgabe in $d+1$ Variablen

$$\begin{array}{rl} & \max v \\ \text{u.d.N.} & f_i(x) \ge v \quad 1 \le i \le n \end{array}$$

zu identifizieren, wobei $v \in \mathbb{R}$ eine neue Variable ist, die den Funktionswert der unteren Einhüllenden repräsentiert.

4.4.2 Der Satz von Kuhn und Tucker

Zur Bestimmung der Lage der lokalen Maxima zitieren wir einen zentralen Satz der Theorie der nichtlinearen Programmierung.

Man bezeichnet zu einem Punkt $x^* \in M$ die Menge

$$A(x^*) = \{i \mid 1 \leq i \leq m,\ g_i(x^*) = 0\}$$

als Indexmenge der in x^* *bindenden Nebenbedingungen.*

Satz 9 (Kuhn, Tucker) *Es seien $f, g_i : \mathbb{R}^d \to \mathbb{R}$ $(1 \leq i \leq m)$ stetig differenzierbare Funktionen, und $M = \{x \in \mathbb{R}^d \mid g_i(x) \leq 0\}$ der zulässige Bereich. x^* sei lokaler Maximalpunkt von f über M, und die Optimierungsaufgabe* $\max_{x \in M} f(x)$ *sei regulär in x^*.*

Dann gibt es einen Vektor $u^* = (u_1^*, \ldots, u_m^*) \in \mathbb{R}^m$ mit

$$\begin{array}{rcll} \nabla f(x^*) - \sum_{i=1}^m u_i^* \nabla g_i(x^*) & = & 0 & \quad (1) \\ g_i(x^*) & \leq & 0 \quad \forall\, 1 \leq i \leq m & \quad (2) \\ \sum_{i=1}^m u_i^* g_i(x^*) & = & 0 & \quad (3) \\ u_i^* & \geq & 0 \quad \forall\, 1 \leq i \leq m & \quad (4) \end{array}$$

Beweis: vgl. etwa [Ho79]. ∎

Die Bedingungen (1) - (4) bezeichnet man häufig als *Kuhn-Tucker-Bedingungen*, die Komponenten des Vektors u^* heißen *Lagrange-Multiplikatoren.*

Die Regularitätsbedingung schließt degenerierte Konfigurationen aus; hinreichende Bedingung für die Regularität der Optimierungsaufgabe in einem $x \in M$ ist z.B. die Existenz eines Vektors $w \in \mathbb{R}^d$ mit $<w, \nabla g_i(x)> < 0$ für alle $i \in A(x)$ *(Mangasarian-Fromovitz-Vektor).*

Definieren wir nun zu der Schar $\mathcal{F}$ die Funktionen $f, g_i : \mathbb{R}^{d+1} \to \mathbb{R}$, $f(x,v) = v$, $g_i(x,v) = v - f_i(x)$, so ist dem MaxMin-Problem von $\mathcal{F}$ zugeordnete Optimierungsaufgabe in jedem $x \in M$ regulär, da $d = (0,..,0,-1) \in \mathbb{R}^{d+1}$ für alle $x \in M$ ein Mangasarian-Fromovitz-Vektor ist. Sei $(x^*, v^*) \in M$ eine Lösung dieser Optimierungsaufgabe. Aus den Kuhn-Tucker-Bedingungen folgt dann:

Es gibt $u_i^* \geq 0,\ 1 \leq i \leq n$, so daß

$$\begin{array}{rcll} \sum_{i=1}^n u_i^* \nabla f_i(x^*) & = & 0 & \text{aus (1)} \\ \sum_{i=1}^n u_i^* & = & 1 & \text{aus (1)} \\ f_i(x^*) & \geq & v^* & \text{aus (2)} \end{array}$$

Da nach Definition $g_i(x^*) = 0 \iff i \in A(x^*)$, folgt mit (2), (3) und (4) $u_i^* = 0$ für alle $i \notin A(x^*)$. Man kann daher (3) auch schreiben als

$$u_i^*[v - f_i(x^*)] = 0 \; \forall \; 1 \leq i \leq n$$

Wir erinnern an die Definition einer konvexen Menge in der Einleitung und einer aktiven Funktion im vorangegangenen Abschnitt und fassen zusammen im

Korollar 1 ([Lem89]) *Ist $x^* \in M$ lokales Maximum der unteren Einhüllenden einer Schar $\mathcal{F}_I = \{f_i \mid i \in I\}$ stetig differenzierbarer Funktionen, so ist der Nullvektor Konvexkombination der Gradienten der in x^* aktiven Funktionen.*

$$0 \in conv(\{\nabla f_i(x^*) \mid i \in A_{\mathcal{F}}(x^*)\})$$

Ist in x nur eine Funktion - sagen wir f_i - aktiv, so ergibt sich als Aussage des Korollars die bekannte notwendige Extremalbedingung $\nabla f_i(x) = 0$.

Sind in x zwei Funktionen aktiv - z. B. f_i, f_j - liegt x auf der Projektion des Schnitts der Funktionsgraphen von f_i und f_j, d. h. auf dem Separator sep_{ij}. x erfüllt dann wie alle lokalen Maxima, die auf sep_{ij} liegen, das Gleichungssystem

$$u_i \nabla f_i(x) + u_j \nabla f_j(x) = 0 \;\wedge\; f_i(x) = f_j(x)$$

mit geeigneten Lagrange-Multiplikatoren $u_i, u_j \geq 0,\; u_i + u_j = 1$.

Dies ist äquivalent zu

$$\nabla f_j(x) + \lambda \nabla (f_i - f_j)(x) \;\wedge\; 0 \leq \lambda \leq 1 \;\wedge\; f_i(x) = f_j(x)$$

Wir werden jedoch zur Bestimmung der Extrema auf dem Separator zweier Funktionen häufig auf einen etwas variierten Ansatz zurückgreifen, um die Rechnung zu vereinfachen. Wir benötigen hierzu den

Satz 10 *Es seien $f, h_j : I\!R^d \to I\!R$ $(1 \leq j \leq k, k < d)$ stetig differenzierbare Funktionen, $S = \{x \in I\!R^d \mid h_j(x) = 0, 1 \leq j \leq k\}$. x^* sei lokaler Extremalpunkt von f über S und die Gradienten $\nabla h_j(x^*)$ $1 \leq j \leq k$ seien linear unabhängig.*

Dann gibt es einen Vektor $v^* = (v_1^*, \ldots, v_k^*) \in I\!R^k$ mit

$$\begin{aligned} \nabla f(x^*) + \sum_{j=1}^{k} v_j^* \nabla h_j(x^*) &= 0 \\ h_j(x^*) &= 0 \quad \forall 1 \leq j \leq k \end{aligned}$$

Definiert man die *Lagrangefunktion* $L : \mathbb{R}^{d+k} \to \mathbb{R}$,

$$L(x_1, \ldots, x_d, v_1, \ldots, v_k) = f(x) + \sum_{j=1}^{k} v_j h_j(x),$$

so ist also notwendige Bedingung für die Existenz eines lokalen Extremums der Funktion f im Punkt $x^* \in S$ das Verschwinden des Gradienten der Lagrangefunktion in einem Punkt (x^*, v^*)

$$\nabla L(x^*, v^*) \stackrel{!}{=} 0$$

sofern die Gradienten $\nabla h_j(x^*)$ linear unabhängig sind.

Seien nun $f_i, f_j \in \mathcal{F}$ stetig differenzierbare Funktionen in zwei reellen Variablen und $g(x,y)$ eine stetig differenzierbare Funktion mit der Eigenschaft

$$g(x,y) = 0 \iff f_i(x,y) - f_j(x,y) = 0$$

anders gesagt, $g(x,y) = 0$ ist Kurvengleichung von sep_{ij}. Dann kann man alle lokalen Extrema auf dem Separator durch Nullsetzen des Gradienten der Lagrangefunktion

$$L(x,y,\lambda) = f_i(x,y) + \lambda g(x,y)$$

d. h. durch Lösen des Gleichungssystems

$$\begin{aligned}
\frac{\partial}{\partial x}[f(x,y) + \lambda g(x,y)] &= 0 \\
\frac{\partial}{\partial y}[f(x,y) + \lambda g(x,y)] &= 0 \\
g(x,y) &= 0
\end{aligned}$$

mit den Unbekannten x, y und λ bestimmen, wenn man die Fälle $\nabla g(x,y) = 0$ gesondert betrachtet.

Wir haben nun die Grundlagen der Strategie eines MaxMin-Algorithmus für eine Schar $\mathcal{F}$ stetiger Funktionen in zwei Variablen gelegt. Er setzt sich aus drei Phasen zusammen

1. Berechne und repräsentiere das Minimierungsdiagramm von $\mathcal{F}$ effizient.
2. Bestimme die Menge der *kritischen Punkte* des MD[$\mathcal{F}$].
3. Wähle unter den kritischen Punkten denjenigen, der die untere Einhüllende maximiert.

Das Verfahren ist korrekt, falls die untere Einhüllende ihr Maximum annimmt und die kritischen Punkte eine Obermenge der lokalen Maxima der unteren Einhüllenden darstellen. Nach Korollar 1 kann man diejenigen Punkte x^* des Definitionsbereichs als kritisch ansehen, die $0 \in \mathrm{conv}(\{\nabla f_i(x^*) \mid i \in A_{\mathcal{F}}(x^*)\})$ erfüllen. Häufig führt auch ein etwas groberes Verfahren zum Ziel: Wir nehmen alle Knoten des MD, die potentiellen Extrema auf den Separatoren, die stationären Punkte der Funktionen aus $\mathcal{F}$ sowie die Extrema auf den Randlininien des Definitionsbereichs (sofern dieser begrenzt ist) in die Menge der kritischen Punkte auf.

Treten auf der unteren Einhüllenden unendlich viele kritische Punkte mit gleichem Funktionswert auf (z.B. sind alle Punkte einer konstanten Funktion stationär und somit kritisch), ersetzen wir sie durch einen beliebigen dieser Punkte, den *Referenzpunkt*. Dies ist zulässig, da wir im Falle gleichwertiger Lösungen eine beliebige dieser Lösungen erwarten.

Bemerkung: Korollar 1 gilt gleichermaßen für lokale Minima der oberen Einhüllenden einer Schar stetig differenzierbarer Funktionen. Somit lassen sich MinMax-Aufgaben mit gleichem Ansatz lösen.

4.4.3 Äquivalente Funktionenscharen

In Abschnitt 4.3.2 haben wir gesehen, daß ordnungsäquivalente Funktionenscharen gleiche Minimierungsdiagramme erzeugen. Schränkt man diese Relation weiter ein, gelangt man zu Funktionen, deren untere Einhüllende an gleicher Stelle maximal werden.

Definition 12 *Seien $\mathcal{F}_I = \{f_i \mid i \in I\}$ und $\mathcal{G}_I = \{g_i \mid i \in I\}$ Scharen reellwertiger, stetiger Funktionen über $Def \subseteq \mathbb{R}^2$, dann heißen $\mathcal{F}_I$ und $\mathcal{G}_I$* äquivalent, *genau dann, wenn für alle $x, y \in Def$ und für alle $i, j \in I$ gilt:*

$$sign(f_i(x) - f_j(y)) = sign(g_i(x) - g_j(y))$$

Lemma 11 *Seien $\mathcal{F}_I$ und $\mathcal{G}_I$ äquivalente Funktionenscharen über $Def \subseteq \mathbb{R}^2$. Dann hat das MaxMin-Problem für $\mathcal{F}$ und $\mathcal{G}$ die gleiche Lösung, falls eine solche existiert. (Gleiches gilt auch für das MinMax-Problem).*

Beweis: Wir führen den Beweis für das MaxMin-Problem. Sei x^* Lösung des MaxMin-Problems für $\mathcal{F}$, d. h. Abszisse des globalen Maximums von f^u und $i \in A_{\mathcal{F}}(x^*)$ beliebig Dann gilt

$$\begin{array}{rcll} f_i(x^*) & \geq & f_k(y) & \forall y \in Def,\ \forall k \in A_{\mathcal{F}}(y) \\ g_i(x^*) & \geq & g_k(y) & \forall y \in Def,\ \forall k \in A_{\mathcal{F}}(y) \quad \text{da } \mathcal{F} \text{ und } \mathcal{G} \text{ äquivalent} \\ g_i(x^*) & \geq & g_k(y) & \forall y \in Def,\ \forall k \in A_{\mathcal{G}}(y) \quad \text{da MD}[\mathcal{F}] = \text{MD}[\mathcal{G}] \text{ nach Lemma 10} \end{array}$$

und x^* ist Abszisse des globalen Maximums von $g^u[\mathcal{G}]$. ∎

Äquivalente Funktionenscharen sind zum Beispiel solche, die durch Konkatenation mit einer streng monotonen Funktion ineinander überführt werden können: $\mathcal{F}_I = \{f_i \mid i \in$

$I\}$ ist äquivalent zu $\mathcal{G}_I = \{g_i \mid i \in I\}$ mit $g_i = h \circ f_i$, wenn h streng monoton steigend auf $f_i(Def)$ $\forall i \in I$.

Sind also die Funktionswerte aller Funktionen f_i einer Schar $\mathcal{F}_I$ nicht negativ, so ist $\mathcal{F}_I$ äquivalent zu $\mathcal{F}_I^2 = \{f_i^2 \mid i \in I\}$. Diesen Umstand werden wir häufig ausnutzen, da die quadrierten Funktionen oft einfacher zu handhaben sind.

4.5 Gewichteter Maximin-Punkt in der Ebene

In diesem Abschnitt werden wir die gewonnenen Ergebnisse anwenden, um das eingangs des Kapitels gestellte Problem (P14) auch im gewichteten Fall lösen zu können.

Die Aufgabenstellung läßt sich äquivalent als MaxMin-Problem über der auf dem zulässigen Bereich M erklärten Schar $\mathcal{F} = \{f_1, \ldots, f_n\}$ der quadrierten Abstandsfunktionen formulieren, wobei

$$f_i(x, y) = w_i^2 \delta^2(p_i, (x, y)) = w_i^2((x - x_i)^2 + (y - y_i)^2)$$

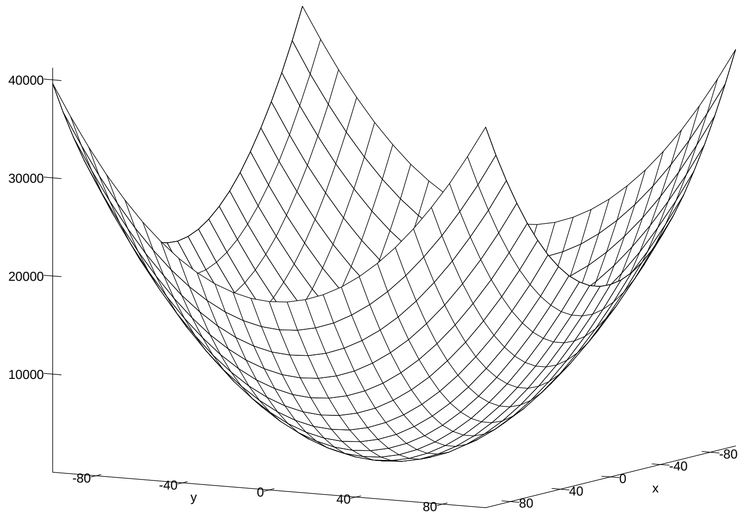

Abbildung 4.2: Der Graph der Distanzfunktion ist ein Rotationsparaboloid

Ausgehend von der auf Seite 57 skizzierten Strategie entwickeln wir eine erste Grundstruktur des Algorithmus.

Schritt 1 Berechne das Minimierungsdiagramm MD[$\mathcal{F}$]

Schritt 2 Bestimme die möglichen lokalen Maxima von $f^u[\mathcal{F}]$ im Innern $int(M)$ der zulässigen Region M mit Hilfe des MD[$\mathcal{F}$].

Schritt 3 Bestimme die möglichen lokalen Maxima von $f^u[\mathcal{F}]$ auf ∂M.

Schritt 4 Wähle aus den in den Schritten 2 und 3 gefundenen Punkten den mit maximaler Ordinate aus.

4.5.1 Struktur und Berechnung des Minimierungsdiagramms

Das in **Schritt 1** gesuchte Minimierungsdiagramm MD[$\mathcal{F}$] ist das von Aurenhammer und Edelsbrunner in [AE84] betrachtete Voronoidiagramm für gewichtete Punkt/-mengen.

Gestalt der Separatoren

Die Kanten des gesuchten Minimierungsdiagramm MD[$\mathcal{F}$] sind Teilstücke der Separatoren der Funktionenschar. Um eine erste Vorstellung vom Aufbau des Minimierungsdiagramms zu gewinnen, ermitteln wir die Gestalt der Separatoren. Der Separator der Punkte p_i und p_j ist definitionsgemäß die Lösungsmenge der Gleichung

$$\begin{array}{rrcl} & f_i(x,y) & = & f_j(x,y) \\ \Longleftrightarrow & w_i^2[(x-x_i)^2+(y-y_i)^2] & = & w_j^2[(x-x_j)^2+(y-y_j)^2] \end{array}$$

Die Lösungen bilden für $w_i > w_j$ einen Kreis mit Mittelpunkt

$$m = \frac{w_i^2 p_i - w_j^2 p_j}{w_i^2 - w_j^2}$$

und Radius

$$r = \delta(p_i,p_j)\frac{w_i w_j}{w_i^2 - w_j^2}$$

den *Kreis des Apollonius* der Punkte p_i und p_j. Gilt $w_i > w_j$ umfaßt dieser Kreis den Punkt p_i, nicht jedoch den Punkt p_j; er entartet für $w_i = w_j$ zur Mittelsenkrechten der beiden Punkte. Abbildung 4.3 gibt eine anschauliche Begründung dieses Sachverhalts.

Die Graphen der multiplikativ gewichteten Abstandsfunktionen sind Rotationsparaboloide; ihre Schnittlinie ist eine ebene Ellipse, deren Projektion in die xy-Ebene einen Kreis ergibt.

Die Punkte des $\mathbb{R}^3$, die zu zwei Punkten p_i und $p_j \in S$ gleichen gewichteten Abstand besitzen, bilden eine Kugel, die im Fall $w_i > w_j$ den Punkt p_i umschließt und im Fall $w_i = w_j$ zu einer Ebene entartet, die *Äquidistanzkugel* von p_i und p_j. Wir können sep_{ij} auch als Schnitt der Äquidistanzkugel mit der xy-Ebene interpretieren.

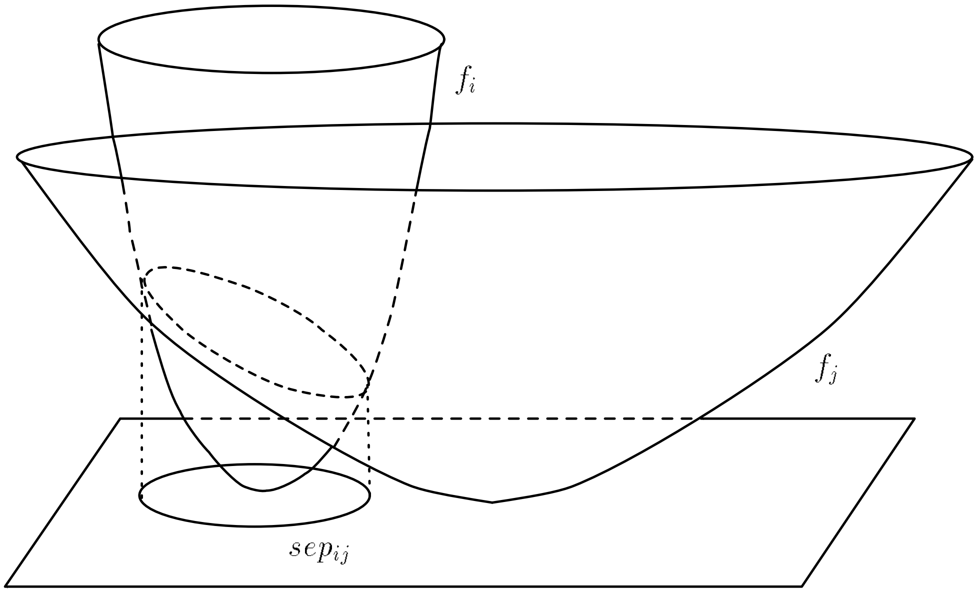

Abbildung 4.3: Graphen und Separator zweier Abstandsfunktionen

Komplexität des Minimierungsdiagramms

Die Kanten des MD[$\mathcal{F}$] sind wie gesehen Kreisbögen. Abbildung 4.4 zeigt ein Beispiel eines solchen multiplikativ gewichteten Voronoidiagramms. Im Unterschied zum ungewichteten Voronoidiagramm sind die einzelnen Regionen nicht mehr konvex, manche sind sogar nicht einfach zusammenhängend. Weiterhin kann ein Punkt p_i mehrere nicht zusammenhängende Zellen erzeugen, zu deren Punkte er bezüglich der gewichteten Abstandsfunktion nächstgelegener Punkt ist (Im Funktionenbild ist f_i einzige aktive Funktion.) Damit können wir nicht wie im ungewichteten Fall von einer linearen Größe des Diagramms ausgehen; tatsächlich existieren Probleminstanzen, in denen das MD die Ebene in $O(n^2)$ Zusammenhangskomponenten zerteilt [AE84].

Der Algorithmus von Aurenhammer und Edelsbrunner

In **Schritt 1** verwenden wir den in [AE84] beschriebenen Algorithmus, der das MD für multiplikativ gewichtete Punkte in optimaler Zeit $O(n^2)$ berechnet. Er bettet das planare Diagramm mit Hilfe einer geometrischen Transformation - der *Inversion* - in den dreidimensionalen Raum ein. Da die Inversion die Äquidistanzkugeln in Halbräume abbildet, besteht der resultierende Zellkomplex aus Polyedern und ist somit einfacher zu beschreiben und zu berechnen als das ursprüngliche Diagramm. Der Zellkomplex wird inkrementell aufgebaut, indem die den gewichteten Punkten zugeordneten Ebenen nacheinander eingefügt werden. Im nächsten Schritt schneidet man den Zellkomplex

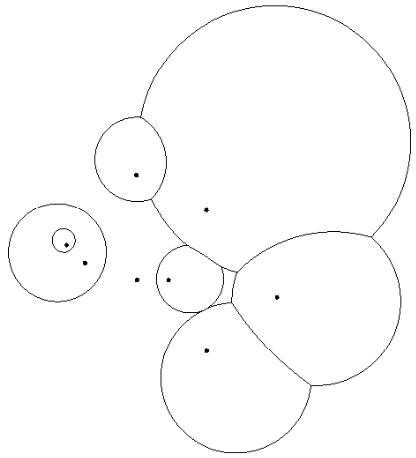

Abbildung 4.4: Multiplikativ gewichtetes Voronoidiagramm für 8 Punkte

mit einer Kugel, dem Bild der xy-Ebene unter der Inversion. Invertiert man schließlich diesen Schnitt erhält man das gewünschte Minimierungsdiagramm in der Ebene.

Aurenhammer erzielt in [Au87] mit einer anderen Technik, die auf Powerdiagrammen beruht, die gleiche optimale Laufzeit.

Aus diesem Minimierungsdiagramm benötigen wir die Informationen:

- Die Koordinaten der Knoten v_i des MD zusammen mit den ≥ 3 Indizes der Weltpunkte, die zu v_i gleichen gewichteten Abstand haben.
- Eine Liste der Kanten des MD, gekennzeichnet durch die Indizes der beiden Weltpunkte, auf deren Separator die Kante liegt (dies ermöglicht eine algebraische Beschreibung der Kurve z. B. durch ihren Mittelpunkt und Radius) ihren Anfangs- und Endpunkt v_i und v_j (diese fallen bei geschlossenen Kurven zusammen), sowie ein Bit, das anzeigt, in welcher der beiden Halbebenen, in die die Gerade $\overline{v_i v_j}$ die Ebene zerteilt, das Kurvenstück liegt.

All diese Informationen können in Zeit proportional zur Größe des MD aus der im obigen Algorithmus erzeugten Datenstruktur entnommen werden.

4.5.2 Lage der Maxima der unteren Einhüllenden

Zur effizienten Durchführung von **Schritt 2** müssen wir zunächst die Lage der lokalen Maxima $p^* = (x^*, y^*)$ der Funktion $f^u[\mathcal{F}]$ studieren.

Wir unterscheiden in Anlehnung an Korollar 1 drei Fälle:

FALL 1: p^* liegt in einer Region von MD$[\mathcal{F}]$.

Sei $A_{\mathcal{F}}(p^*) = \{i\}$, d. h. f_i ist einzige in p^* aktive Funktion. Notwendige Bedingung für die Extremalität von p^* ist dann $\nabla f_i(p^*) = 0$. Einziger stationärer Punkt von f_i ist jedoch das lokale Minimum p_i, ein Widerspruch. In den Regionen des MD sind also keine lokalen Maxima zu finden.

FALL 2: p^* liegt auf einer Kante von MD$[\mathcal{F}]$

Sei $A_{\mathcal{F}}(p^*) = \{i, j\}$ und damit $f_i(p^*) = f_j(p^*)$. Notwendige Bedingung für Maximalität von p^* ist

$$\exists\, 0 \leq \lambda \leq 1 : \quad \nabla f_j(p^*) + \lambda(\nabla f_i(p^*) - \nabla f_j(p^*)) = 0$$

Um die Rechnung zu vereinfachen, legen wir das Koordinatensystem so, daß p_j im Ursprung und p_i auf der x-Achse zu liegen kommt. Wir erhalten die Distanzfunktionen

$$f_j(x, y) = w_j^2(x^2 + y^2) \quad f_i(x, y) = w_i^2((x - x_i)^2 + y^2)$$

und suchen die Lösungen (x^*, y^*, λ^*) des Gleichungssystems

$$\begin{aligned}
\frac{\partial f_j}{\partial x}(p^*) + \lambda[\frac{\partial f_i}{\partial x}(p^*) - \frac{\partial f_j}{\partial x}(p^*)] &= 0 \\
\frac{\partial f_j}{\partial y}(p^*) + \lambda[\frac{\partial f_i}{\partial y}(p^*) - \frac{\partial f_j}{\partial y}(p^*)] &= 0 \\
f_j(p^*) - f_i(p^*) &= 0
\end{aligned}$$

unter der Nebenbedingung $0 \leq \lambda \leq 1$.

Mit der Ersetzung $a = w_i^2 / w_j^2 > 0$ ist das Gleichungssystem äquivalent zu

$$\begin{aligned}
x + \lambda[(a - 1)x - a x_i] &= 0 \qquad (1) \\
y + \lambda(a - 1)y &= 0 \qquad (2) \\
(x^2 + y^2) - a[(x - x_i)^2 + y^2] &= 0 \qquad (3)
\end{aligned}$$

Für $a = 1$ (d. h. $w_i = w_j$) erhalten wir $x = x_i/2, y = 0, \lambda = 1/2$ als Lösung.

Ansonsten ist Gleichung (2) gleichwertig mit

$$y = 0 \;\vee\; \lambda = \frac{-1}{a - 1}$$

FALL 2.1 $\lambda = \dfrac{-1}{a-1}$

Aus (1) folgt

$$\begin{array}{rrcl} & x - \frac{1}{a-1}(a(x-x_i)-x) & = & 0 \\ \Longleftrightarrow & \frac{a}{a-1}x_i & = & 0 \end{array}$$

im Widerspruch zu $a > 0$ und $x_i \neq 0$, da $p_i \neq p_j$.

FALL 2.2 $y = 0$

In (3) eingesetzt folgt

$$(1-a)x^2 + 2ax_ix - ax_i^2 = 0$$

mit den Lösungen

$$x_{1,2} = \frac{\pm\sqrt{a}}{1 \pm \sqrt{a}} x_i$$

Mit (1) ergibt sich

$$\lambda_{1,2} = \frac{1}{1 \pm \sqrt{a}}$$

Da für jedes $a > 0$ der Multiplikator $\lambda_2 = \frac{1}{1-\sqrt{a}}$ nicht wie gefordert im Intervall $[0,1]$ liegt, ist diese Lösung kein kritischer Punkt. Es bleibt also als einziges potentielles Maximum der Punkt $p^* = (\frac{\sqrt{a}}{1+\sqrt{a}}x_i, 0)$ (die Lösung $(x_i/2, 0)$ des Spezialfalles $a = 1$ wird durch diese Betrachtung ebenfalls abgedeckt). Man sieht jedoch leicht, daß p^* lokales Minimum von $f_i(x^*, y)$ und $f_j(x^*, y)$ als Funktionen in y ist und somit nicht lokales Maximum von f^u sein kann. Folglich sind auch auf den Kanten des MD keine lokalen Maxima zu finden.

FALL 3: p^* ist ein Knoten von MD$[\mathcal{F}]$

Wir betrachten sämtliche $O(n^2)$ Punkte dieser Art als potentielle Maxima, da Überlegungen wie in den vorherigen Fällen nicht zu einer signifikanten Einschränkung der Anzahl der Kandidaten führen.

Gestalt des zulässigen Bereiches

Der zulässigen Region M geben wir exemplarisch erst die Form eines einfachen Polygons mit k Ecken, später die Gestalt einer Ellipse. Der Ansatz ist aber auch auf andere geometrische Figuren übertragbar.

M ist ein einfaches Polygon mit k Ecken. Um entscheiden zu können, welche der bisher gefundenen potentiellen Maxima - dies sind die $O(n^2)$ Knoten des MD - in der zulässigen Region, dem Polygon M, liegen, erzeugen wir in Zeit $O(k \log k)$ eine Datenstruktur, die Anfragen dieser Art in Zeit $O(\log k)$ beantworten kann (vgl. z.B. [C84b]). Schritt 2 benötigt also Zeit $O(n^2 \log k + k \log k)$).

In **Schritt 3** finden wir die lokalen Maxima auf ∂M wie folgt: Sei $\overline{uv}$ eine Kante des Polygons M. Wir definieren einen eindimensionalen Querschnitt durch $\mathcal{F}$, indem wir den Definitionsbereich der Funktionen auf die Gerade uv

$$g_{uv}: \ (v_y - u_y)x - (v_x - u_x)y + v_x u_y - u_x v_y = 0$$

einschränken. Die lokalen Maxima von $f^u[\mathcal{F}]$ findet man durch Lösen der MaxMin-Aufgabe der der Schar $\mathcal{G} = \{g_1, \ldots, g_n\}$ der Querschnittsfunktionen über der Kante $\overline{uv}$. Es sind dies im Fall $u_x \neq v_x$ die Funktionen

$$g_i(x) = w_i^2 \left[(x - x_i)^2 + \left(\frac{v_y - v_x}{v_x - u_x} x + \frac{v_x u_y - u_x v_y}{v_x - u_x} - y_i \right)^2 \right]$$

über dem Intervall $[u_x, v_x]$, im Fall $u_y \neq v_y$ die Funktionen

$$g_i(y) = w_i^2 \left[\left(\frac{v_x - u_x}{v_y - u_y} y + \frac{u_x v_y - v_x u_y}{v_y - u_y} - x_i \right)^2 + (y - y_i)^2 \right]$$

über dem Intervall $[u_y, v_y]$. Da es sich in beiden Fällen um quadratische Funktionen in einer Variablen handelt, kann dies mit den Methoden aus Kapitel 2 in Zeit $O(n \log n)$ pro Kante geschehen. Summiert man über alle Kanten erhält man als Gesamtlaufzeit von Schritt 3 $O(kn \log n)$.

M ist eine Ellipse. Ist die zulässige Region eine Ellipse (vgl. Anhang), kann man mit Hilfe der sie beschreibenden Gleichung zweiten Grades $g(x, y) = 0$ in konstanter Zeit prüfen, ob ein kritischer Punkt $p^* = (x^*, y^*)$ im Innern von M liegt.

Schritt 3: Auch die lokalen Maxima auf ∂M lassen sich schneller finden. Ein solches Maximum kann unterschiedlichen Ursprungs sein:

1. Es ist lokales Maximum eines der Distanzparaboloide $f_i(x, y) = w_i^2((x - x_i)^2 + (y - y_i)^2)$ über ∂M.

2. Es ist der Schnitt einer Kante des MD mit ∂M

Wir bestimmen die Maxima aus (1.) wiederum mit der Methode der Lagrangemultiplikatoren. Zur Vereinfachung legen wir das Koordinatensystem so, daß die Symmetrieachsen der Ellipse auf die Koordinatenachsen fallen; alle Punkte der Ellipse erfüllen dann eine Gleichung der Form

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

mit reellen Formparametern a, b. Kritische Punkte sind solche, an denen der Gradient der Lagrangefunktion

$$L(x, y, \lambda) = w_i^2[(x - x_i)^2 + (y - y_i^2)] + \lambda[\frac{x^2}{a^2} + \frac{y^2}{b^2} - 1]$$

verschwindet, sie sind also Lösungen des Gleichungssystems

$$\begin{aligned} 2w_i^2(x - x_i) + \lambda\frac{2}{a^2}x &= 0 \\ 2w_i^2(y - y_i) + \lambda\frac{2}{b^2}y &= 0 \\ \frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 &= 0 \end{aligned}$$

Falls $x_i = 0$ ($y_i = 0$ symmetrisch) finden wir im allgemeinen Fall ($a^2 \neq b^2$) die Lösungen $(0, \pm b)$ und

$$\left(\pm\sqrt{a^2 - \frac{a^2b^2y_i^2}{(a^2-b^2)^2}}, \frac{b^2y_i}{b^2-a^2}\right)$$

Gilt $x_i, y_i \neq 0$ impliziert das Gleichungssystem eine Gleichung 4.Grades in λ

$$\left(\frac{aw_i^2x_i}{a^2w_i^2+\lambda}\right)^2 + \left(\frac{bw_i^2y_i}{b^2w_i^2+\lambda}\right)^2 - 1 = 0$$

und erlaubt somit maximal 4 Lösungen.

Es gibt also pro Weltpunkt höchstens 4 kritische Punkte, in denen ein Maximum angenommen werden könnte, und diese Punkte können in konstanter Zeit gefunden werden. Für jeden kritischen Punkt prüfen wir in linearer Zeit, ob er tatsächlich auf der unteren Einhüllenden liegt. Summiert über alle n Distanzfunktionen kostet dies Zeit $O(n^2)$.

Der Schnitt jeder der $O(n^2)$ Kanten des MD mit ∂M kann als Schnitt von Kegelschnitten durch Lösen einer Gleichung 4. Grades in konstanter Zeit ermittelt werden.

Wir erhalten also alle $O(n^2)$ potentiellen Maxima aus Schritt 2 und 3 in Zeit $O(n^2)$.

In **Schritt 4** ermittelt man in Zeit proportional zur Anzahl der Kandidatenpunkte den optimalen Punkt.

Wir fassen unsere Ergebnisse zusammen im

Satz 11 *Gegeben sei das Problem des gewichteten Maximin-Punktes in der Ebene nach Beschreibung (P14). Hat die Begrenzungsregion M die Form*

(i.) eines einfachen Polygons mit k Ecken, so kann eine Lösung in Zeit $O(n^2 \log k + kn \log n + k \log k)$ gefunden werden.

(ii.) einer Ellipse, so kann eine Lösung in Zeit $O(n^2)$ gefunden werden.

Wählt man als Begrenzungsregion die konvexe Hülle der Punktmenge P, so ist die Kantenanzahl durch n beschränkt und man erhält eine Laufzeit von $O(n^2 \log n)$.

Teil (i.) des Satzes verallgemeinert ein Resultat aus [To83]. Toussaint zeigt dort, daß der ungewichtete Maximin-Punkt in der Ebene in Zeit $O(n^2 \log n)$ gefunden werden kann, sofern die zulässige Region die Gestalt eines einfachen Polygons mit $O(n)$ Ecken besitzt. Eine Reihe von Beiträgen der Operations Research Literatur widmet sich dem Problem des gewichteten Maximin-Punktes in der Ebene. Techniken der algorithmischen Geometrie wurden dabei jedoch nicht eingesetzt. Der laut [SM92] einzige bekannte exakte Algorithmus [Mel80] zählt in Zeit $O(n^4)$ die von den Kuhn-Tucker-Bedingungen bestimmten kritischen Punkte auf. Weiterhin wurden in [MC85a], [MC85b] und [MC86] Heuristiken für das Maximin-Problem vorgeschlagen, deren Laufzeiten nach Angabe der Autoren in der Praxis quadratisch mit der Eingabe wachsen; ein formaler Beweis dieser Behauptung wurde jedoch weder für den Worst Case noch für den Average Case geführt. Diese Heuristiken teilen zunächst die zulässige Region in vertikale Streifen der Breite b und bestimmen in jedem Streifen die Schnittpunkte der Separatoren aller Punktetripel, deren Abstand einen Schwellenwert r nicht übersteigt. Aus diesen Schnittpunkten wird der Beste ausgewählt. Die Güte der gefundenen Lösung hängt natürlich in entscheidender Weise von der geschickten Wahl der Parameter b und r ab.

4.6 Duale Problemstellung

Das zum Maximin-Problem der Ebene duale Problem ist gegeben durch

Problem 15 : [Minimax-Punkt in der Ebene (1-center problem)]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\}$, $p_i = (x_i, y_i)$, von Punkten der reellen Ebene $\mathbb{R}^2$ und eine Menge $W = \{w_1, \ldots, w_n\}$ von positiven, reellen Gewichtungsfaktoren.

Gesucht: Ein Punkt $p^* = (x^*, y^*) \in \mathbb{R}^2$, der

$$\max_{1 \le i \le n} w_i \sqrt{(x - x_i)^2 + (y - y_i)^2}$$

minimiert.

Im ungewichteten Fall ist der Lösungspunkt p^* Mittelpunkt des kleinsten Kreises, der alle Punkte aus S enthält *(smallest enclosing circle)*. Es handelt sich also um das bereits in der Einleitung zitierte klassische Sylvestersche Problem.

Man kann dieses Problem nach Quadrieren der Abstandsfunktionen durch Einführen einer neuen Variable v als mathematisches Programm formulieren

$$\begin{array}{ll} & \min v \\ \text{u.d.N.} & v \ge (x_i - x)^2 + (y_i - y)^2 \quad \forall p_i \in S \end{array}$$

Formt man die Nebenbedingungen um

$$v \geq -2x_i x - 2y_i y + (x_i^2 + y_i^2) + (x^2 + y^2) \quad \forall p_i \in S$$

sieht man, daß sie sich aus dem quadratischen Term $x^2 + y^2$ und einem linearen Term $g_i(x,y) = -2x_i x - 2y_i y + x_i^2 + y_i^2$ zusammensetzen. Da sowohl $x^2 + y^2$ als auch alle g_i konvexe Funktionen sind, sind auch alle Nebenbedingungen als Summen konvexer Funktionen konvex. Die durch die Nebenbedingungen definierte Oberfläche $v = g_i(x,y) + x^2 + y^2$ ähnelt der Oberfläche eines Polyeders: sie setzt sich ebenfalls aus Flächen, Kanten und Eckpunkten zusammen. Die "Flächen" sind hier Teilstücke von Paraboloiden, von denen je zwei sich in einem Parabelbogen (einer "Kante") schneiden. Ein "Eckpunkt" ist Schnittpunkt dreier oder mehrerer solcher "Kanten".

Der Prune-and-Search-Algorithmus von Megiddo zur Eliminierung redundanter Nebenbedingungen in dreidimensionalen linearen Programmen kann nun auf dieses spezielle quadratische Programm in modifizierter Form angewendet werden. Dabei ist darauf zu achten, daß der Schnittpunkt der nichtredundanten Nebenbedingungen im Gegensatz zum linearen Fall nicht notwendigerweise den optimalen Lösungspunkt bestimmt; es genügt jedoch, die von diesem Eckpunkt ausgehenden "Kanten" auf Minima zu untersuchen, um die optimale Lösung zu finden.

Die für den ursprünglichen Prune-and-Search-Algorithmus ermittelte Zeitschranke $O(n)$ bleibt auch für den modifizierten Algorithmus gültig und wir erhalten das Resultat

Satz 12 (Megiddo) *Sind alle Gewichtsfaktoren gleich, so läßt sich das Problem des Minimax-Punktes in der Ebene in optimaler Zeit $\Theta(n)$ lösen.*

Erlaubt man unterschiedliche Gewichtsfaktoren w_i, ist die Anwendung der Prune-and-Search-Technik nicht länger möglich. Cole beschreibt in [C84a] einen Algorithmus, der (P15) in Zeit $O(n \log n)$ löst.

Kapitel 5

Zielobjekte im Raum mit zwei Freiheitsgraden

Im vergangenen Kapitel haben wir uns mit Positionierungsaufgaben befaßt, bei denen sowohl der Objekt- als auch der Konfigurationsraum Teilräume der euklidischen Ebene $\mathbb{R}^2$ waren. Wir wenden uns nun Problemstellungen zu, in denen die Weltpunkte eine endliche Teilmenge des $\mathbb{R}^3$ bilden und die Lage des zu plazierenden $k-$dimensionalen affinen Teilraums ($0 \leq k \leq 2$) durch einen Punkt auf der Einheitskugel festgelegt werden kann. Punkte der Einheitskugel lassen sich bereits mit zwei reellen Parametern - etwa ihren Längen- und Breitenwinkeln - spezifizieren; es handelt sich demgemäß um Positionierungsaufgaben mit zwei Freiheitsgraden in einem dreidimensionalen Objektraum. Die Problemstellung legt nahe, die Einheitskugel unmittelbar als Konfigurationsraum anzusehen. Da jedoch die Beschreibung komplexer geometrischer Objekte wie etwa der Separatoren der Abstandsfunktionen auf der Einheitskugel recht mühsam ist, werden wir häufig zu einer transformierten Darstellung des Konfigurationsraums und damit der Abstandsfunktionen übergehen, um die Rechnungen und die Konstruktion der Algorithmen zu erleichtern. Wir gliedern das Kapitel entsprechend der Dimension k des Zielobjekts in drei Abschnitte.

5.1 Größter leerer Kreis auf der Einheitskugel

5.1.1 Motivation

Erinnern wir uns an die das (P14) aus Kapitel 4 motivierenden Fragestellungen, erkennen wir, daß die Weltpunkte des (P14) häufig zur Modellierung von geographischen Objekten auf der Erdoberfläche, beispielsweise von Städten, dienen. Dennoch haben wir die Positionierungaufgabe (P14) als ebenes Problem formuliert. Diese Annäherung einer Kugeloberfläche durch die euklidischen Ebene ist nur dann hinreichend genau, wenn die Objektabstände sehr klein gegenüber dem Kugelradius sind. Betrachten wir jedoch Probleminstanzen, in denen die Kugelgestalt nicht vernachlässigt werden kann, wie etwa Städte auf verschiedenen Kontinenten, versagt der bisherige Algorithmus und das Problem stellt sich von neuem. Dieser Abschnitt ist daher dem Maximin-Problem auf der Kugel gewidmet. Es ist der Fall $k = 0$ der oben skizzierten Problembeschreibung, wobei wir den Objektraum ebenfalls auf die Einheitskugel einschränken.

Wir beginnen, indem wir den Leser mit wichtigen Begriffen der Kugelgeometrie vertraut machen und gleichzeitig einige nützliche Notationen einführen.

5.1.2 Kugelgeometrie

Für die $(d-1)$-dimensionale Einheitskugel, dies ist die Menge der Punkte des $\mathbb{R}^d$ mit Einheitsabstand zum Ursprung, verwenden wir die Notation S^{d-1}.

Sind p und q linear unabhängige Punkte des $\mathbb{R}^3$, so bezeichne E_{pq} die Ursprungsebene, die p und q enthält

$$\begin{aligned} E_{pq}: \quad & <p \times q, x> = 0 \\ \Longleftrightarrow \quad & (p_2q_3 - q_2p_3)x_1 + (p_3q_1 - p_1q_3)x_2 + (p_1q_2 - p_2q_1)x_3 = 0 \end{aligned}$$

Wir bezeichnen die Ursprungsebene mit dem von Null verschiedenen Normalenvektor p mit E^p

$$\begin{aligned} E^p: \quad & <p, x> = 0 \\ \Longleftrightarrow \quad & p_1x_1 + p_2x_2 + p_3x_3 = 0 \end{aligned}$$

Eine Ursprungsebene E schneidet die Einheitskugel S^2 in einem Kreis, dem *Großkreis* K_E. Sind p, q linear unabhängige Punkte der S^2, so bezeichnen wir den durch eine Ebene E_{pq} erzeugten Großkreis, der die beiden Punkte p und q enthält, als K_{pq}, den durch die Ebene E^p erzeugten Großkreis mit Mittelpunkt p als K^p. Ist p der Nordpol der Einheitskugel so heißt $K^p = \{s \in S^2 \mid s_3 = 0\}$ der *Äquator* der Einheitskugel; er spaltet S^2 in die *obere Hemisphäre* $S^2_+ = \{s \in S^2 \mid s_3 > 0\}$ und die *untere Hemisphäre* $S^2_- = \{s \in S^2 \mid s_3 < 0\}$.

Die kürzeste Entfernung zweier Punkte p und q auf S^2 (ihre *orthodrome Entfernung*) ist die Länge des kürzeren Bogens des Großkreises K_{pq}, auch *Geodätische* genannt. Man bezeichnet diesen Abstand als *Großkreisdistanz* $\delta_G(p,q)$. Offensichtlich ist die größtmögliche Großkreisdistanz zweier Punkte der Einheitskugel π. Solche Punkte bilden die gegenüberliegenden Schnittpunkte einer Ursprungsgeraden mit der Einheitskugel, man nennt sie *Antipoden*.

Ein Kugelkreis $K(p,r)$ ist die Menge aller Punkte, die von einem Punkt $p \in S^2$ gleichen Großkreisabstand r haben: $K(p,r) = \{s \in S^2 \mid \delta_G(s,p) = r\}$. Es gilt $K(p,r) = K(\mathit{Antipode}(p), \pi - r)$. Legt man einen der beiden Antipoden als Mittelpunkt fest, kann man den Flächeninhalt eines Kugelkreises als die Fläche der Kugelkappe definieren, die den gewählten Mittelpunkt p enthält. Ein Kugelkreis ist stets als Schnitt einer Ebene mit der Einheitskugel darstellbar.

5.1.3 Problemstellung

Mit Hilfe dieser Begriffe können wir das Problem formal fassen; wir formulieren es o.B.d.A für die Einheitskugel.

PROBLEM 16 : [Größter leerer Kreis auf der Einheitskugel]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\}$, $p_i = (x_i, y_i, z_i)$ von Punkten der Einheitskugel S^2.

(Man beachte: es ist $z_i = \pm\sqrt{1 - x_i^2 - y_i^2}$ falls $p_i \in S^2_\pm$)

Gesucht: Einen Punkt $p^* \in S^2$, der

$$\min_{1 \leq i \leq n} \delta_G(p_i, p)$$

unter allen Punkten p der Einheitskugel maximiert.

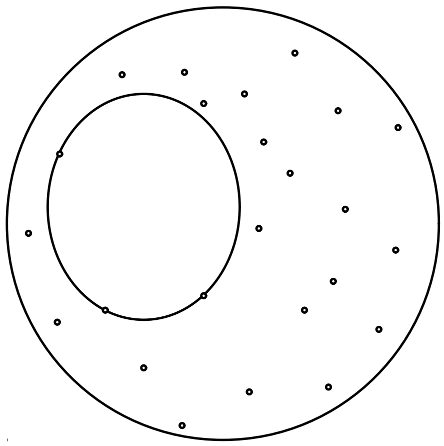

Abbildung 5.1: Größter leerer Kreis auf der Einheitskugel

Anschaulich formuliert suchen wir den Mittelpunkt p^* des hinsichtlich des Flächeninhalts größtmöglichen Kugelkreises $K(p^*)$, der keinen der Weltpunkte umfaßt, d.h. $(int(K(p^*)) \cap S = \emptyset)$.

Die Großkreisdistanz haben wir bei dieser Problemstellung als das intuitiv naheliegende Abstandsmaß gewählt. Sie ist jedoch einer algebraischen Formulierung nur schwer zugänglich. Insbesondere können wir unsere Ergebnisse bezüglich der Komplexität von Maximin-Problemen nicht übertragen, wenn wir trigonometrische Funktionen zur Berechnung der Großkreisdistanz verwenden.

Zwischen euklidischem Abstand δ und Großkreisabstand δ_G besteht jedoch die einfache Beziehung

$$\delta = 2 \sin \frac{\delta_G}{2}$$

die sich auf elementargeometrischem Weg herleiten läßt.

Wichtig für uns ist die Beobachtung, daß die Transformationsfunktion $2 \sin(x/2)$ auf dem Intervall $[0, \pi]$, auf dem δ_G variiert, streng monoton wächst.

Definieren wir nun wie gewohnt die Schar der quadrierten Distanzfunktionen $\mathcal{G}$ bezüglich des Abstandsmaßes δ_G

$$\mathcal{G} = \{g_1, \ldots, g_n\};\ g_i : S^2 \to \mathrm{I\!R},\ g_i(p) = \delta_G^2(p_i, p)$$

so ist diese nach obiger Beobachtung äquivalent zur Funktionenschar

$$\mathcal{H} = \{h_i, \ldots, h_n\};\ h_i : S^2 \to \mathrm{I\!R},\ h_i(p) = \delta^2(p_i, p)$$

Nach Lemma 11 ist damit ist auch der Lösungspunkt zu beiden Abstandsmaßen identisch. Wir können also zum euklidischen Abstandsmaß übergehen, ohne die Qualität des Problems zu verändern.

Wir suchen nun, wie im Kapitelvorspann angedeutet, eine geeignete Darstellung des *Konfigurationsraums*. Alle Punkte p der oberen Hemisphäre S^2_+ lassen sich durch zwei Parameter p_1, p_2 mit $0 \leq p_1^2 + p_2^2 \leq 1$ beschreiben: $p = (p_1, p_2, \sqrt{1 - p_1^2 - p_2^2})$. Um uns der Beschränkungen der Parameter zu entledigen und ein ebenes Problem zu erhalten, projizieren wir die Punkte der oberen Hemisphäre S^2_+ auf die Tangentialebene, die die Einheitskugel S^2 in ihrem Nordpol berührt: die Projektionsebene $E_\pi : x_3 = 1$. Wir ordnen dabei einem Punkt $p = (x, y, \sqrt{1 - x^2 - y^2})$ den Punkt

$$p' = \pi(p) = (x/\sqrt{1 - x^2 - y^2}, y/\sqrt{1 - x^2 - y^2}, 1)$$

zu. Diese bijektive Abbildung π zwischen S^2_+ und E_π heißt *Zentralprojektion* vom Koordinatenursprung, da je zwei korrespondierende Punkte die Durchstoßpunkte eines Ursprungsstrahls mit der obere Hemisphäre bzw. der Projektionsebene darstellen.

Nutzen wir diese modifizierte Darstellung, lassen sich die möglichen Lösungspunkte aus S^2_+ als Punkte einer Ebene, der Projektionsebene E_π beschreiben. Die Einschränkung auf die obere Hemisphäre können wir umgehen, indem wir mehrere Projektionen in verschiedene Richtungen durchführen und die Ergebnisse kombinieren. Offensichtlich reichen 6 solcher Projektionen auf Tangentialebenen, die Einheitsvektoren als Normalenvektoren besitzen und die Einheitskugel wie ein Würfel umschließen, aus, um jeden Punkt der Kugel zusammen mit einer Umgebung des Punktes auf mindestens einer Tangentialebene darstellen zu können. Alternativ kann man die Kugeloberfläche auf Tangentialebenen an Nord- und Südpol projizieren und die Punkte des Äquators gesondert betrachten.

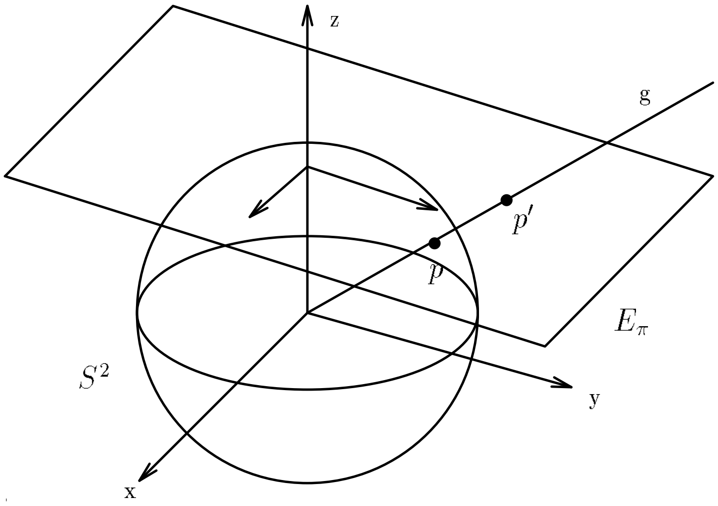

Abbildung 5.2: Zentralprojektion der oberen Hemisphäre

Betrachten wir zunächst nur die Projektionsebene $E_\pi : x_3 = 1$. Jeder Weltpunkt $p_i = (x_i, y_i, z_i)$ erklärt eine Abstandsfunktion in den Ortskoordinaten x und y der ProjektionsebeneE_π, die projizierte quadrierte Abstandsfunktion

$$f_i(x,y) = h_i(\pi^{-1}(x,y,1)) = \delta^2(p_i, \pi^{-1}(x,y,1)) = \left(x_i - \frac{x}{N}\right)^2 + \left(y_i - \frac{y}{N}\right)^2 + \left(z_i - \frac{1}{N}\right)^2$$

mit der Abkürzung $N = \sqrt{x^2+y^2+1}$ für den Normierungsfaktor. Den Koordinaten eines Punktes $p' = (x, y, 1)$ der Ebene E_π wird der quadrierte Abstand seines Urbildes p unter der Zentralprojektion π vom Weltpunkt p_i zugeordnet. (Wir betrachten der Einfachheit halber wieder den quadrierten euklidischen Abstand und verweisen auf Lemma 11).

Lösen wir das MaxMin-Problem für die Funktionenschar $\mathcal{F} = \{f_1, \ldots, f_n\}$, so haben wir die Plazierung des größtmöglichen leeren Kugelkreises gefunden, dessen Mittelpunkt auf der oberen Hemisphäre liegt. Finden wir diese optimalen Plazierungen auch für die anderen Projektionsrichtungen, bildet die Beste der Teillösungen die Lösung des Ausgangsproblems.

Wenden wir uns also dem genannten MaxMin-Problem zu. Unser Ansatz entspricht der Strategie des vorangegangen Kapitels: berechne das Minimierungsdiagramm der Funktionenschar, bestimme effizient eine Obermenge der lokalen Maxima der unteren Einhüllenden und finde unter diesen das globale Optimum.

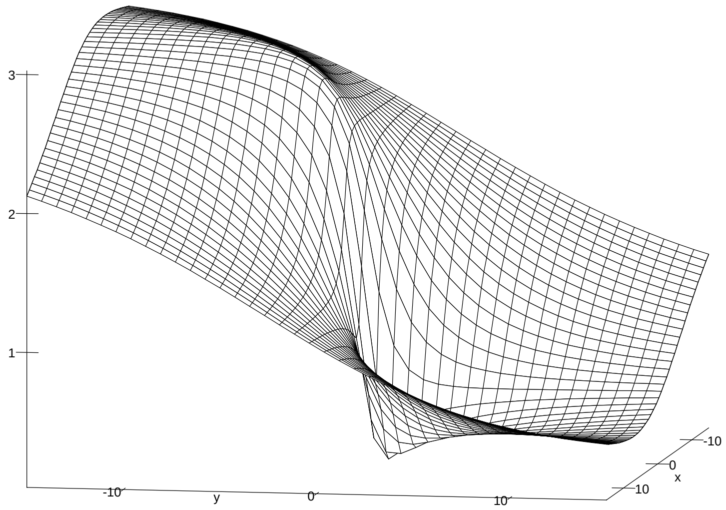

Abbildung 5.3: Graph der Distanzfunktion eines Punktes der oberen Hemisphäre

5.1.4 Struktur und Berechnung des MD

Um einen effizienten Algorithmus zur Berechnung des MD angeben zu können, ist es nötig, Klarheit über dessen Struktur und Komplexität zu gewinnen.

Gestalt der Separatoren

Die Menge der zu zwei Punkten p und q des Raumes äquidistanten Punkte bildet eine Ebene E^{p-q} mit Normalenvektor $p-q$, die durch den Mittelpunkt der Strecke $\overline{pq}$, den Punkt $\frac{1}{2}(p+q)$ verläuft. Liegen p und q auf der Einheitskugel, enthält die Ebene den Ursprung. Damit bildet die Menge der zu p, q äquidistanten Punkte auf S^2 einen Großkreis; seine Projektion auf die Ebene E_π ergibt eine Gerade. Diese Gerade ist Schnittlinie der beiden Ebenen $E_{p-q} : <p-q, x> = 0$ und $E_\pi : x_3 = 1$. Der Separator der Abstandsfunktionen h_i, h_j zweier Weltpunkte p_i und p_j ist also der Großkreis $K_{p_i - p_j}$, als Gleichung des Separators sep_{ij} der projizierten Abstandsfunktionen f_i und f_j finden wir:

$$sep_{ij} : (x_i - x_j)x + (y_i - y_j)y + z_i - z_j = 0,$$

eine Gerade in der Projektionsebene. Die Gleichung folgt auf analytischem Wege auch unmittelbar durch Gleichsetzen der Terme $f_i(x, y)$ und $f_j(x, y)$.

Komplexität des MD

Da alle Kanten des MD Liniensegmente sind, besitzt das MD der Schar der Distanzfunktionen eine ähnliche Gestalt wie das bekannte Voronoidiagramm für Punktmengen unter euklidischer Metrik. Ein Resultat von Sharir und Schwartz gibt Aufschluß über die Komplexität solcher MD.

Satz 13 ([SS90]) *Sei $\mathcal{F} = \{f_1, \ldots, f_n\}$ eine Schar von n stetig differenzierbaren, reellwertigen Funktionen über $I\!R^2$ in allgemeiner Lage. Erfüllen die Funktionen die Bedingungen*

(i.) Der Separator sep_{ij} zweier verschiedener Funktionen f_i, f_j aus $\mathcal{F}$ ist entweder leer oder zusammenhängend, wobei jeder Schnitt mit einer Geraden der Form $x = x_0$ aus genau einem Punkt besteht; darüber hinaus schneiden sich die Querschnittfunktionen $f_i(x_0, y)$ und $f_j(x_0, y)$ transversal in diesem Punkt.

(ii.) Je drei verschiedene Funktionen f_i, f_j und $f_k \in \mathcal{F}$ schneiden sich in höchstens einem Punkt.

hat ihr Minimierungsdiagramm MD[$\mathcal{F}$] Komplexität $O(n)$.

Liegen je 4 Punkte aus S nicht auf einem Kugelkreis, erfüllen die Abstandsfunktionen die geforderten Bedingungen, da die Separatoren wie gesehen Geraden sind. (Senkrechte Separatoren der Form $x = x_0$ lassen sich konzeptuell stets durch Drehen des Koordinatensystems vermeiden, so daß der Satz auch in diesem Fall gilt.)

Algorithmus zur Berechnung des MD

Die einfache Struktur des MD ermöglicht einen Algorithmus, der das Diagramm in optimaler Zeit $O(n \log n)$ konstruiert und eine Datenstruktur aufbaut, die Point Location im MD in optimaler Zeit $O(\log n)$ unterstützt.

Satz 14 ([SS90]) *Sei $\mathcal{F}$ eine Funktionenschar, die den Bedingungen aus Satz 13 genügt. Erlauben die Funktionen die Durchführung der Operationen*

(i.) Entscheide für ein Funktionenpaar $f_i, f_j \in \mathcal{F}$, ob eine der Funktionen an jeder Stelle kleineren Wert als die zweite aufweist, oder bestimme die Seite von sep_{ij}, auf der $f_i < f_j$.

(ii.) Finde für zwei Funktionen $f_i, f_j \in \mathcal{F}$ und gegebenes $x_0 \in I\!R$ den (einzigen) Schnittpunkt ihrer Querschnittfunktionen $f_i(x_0, y)$ und $f_j(x_0, y)$, falls er existiert.

(iii.) Berechne den (einzigen) Schnittpunkt dreier Funktionen $f_i, f_j, f_k \in \mathcal{F}$, falls er existiert.

in konstanter Zeit, so kann das MD[$\mathcal{F}$] in Zeit $O(n \log n)$ berechnet werden.

Der Algorithmus ist eine Verallgemeinerung des klassischen Voronoidiagramm-Algorithmus von Shamos aus [Sh78] und basiert wie dieser auf dem Divide-and-Conquer Konzept. Nach Aufteilung der Funktionenschar $\mathcal{F}$ in zwei etwa gleichgroße Teilscharen werden deren Minimierungsdiagramme rekursiv berechnet und an einer Nahtstelle zum MD[$\mathcal{F}$] zusammengefügt. Die Effizienz dieses Verfahrens beruht auf der Tatsache, daß man die Aufteilung stets so wählen kann, daß die Nahtstelle eine x-monotone Kette bildet.

Der Algorithmus erzeugt zur Darstellung des MD folgende *Datenstruktur*. Die Knoten des MD sind nach aufsteigender Abszisse in einem balancierten Suchbaum gespeichert und jeweils mit den Indizes der (drei) dort aktiven Funktionen markiert. Teilt man das MD an den Abszissen seiner Knoten in vertikale Streifen, kann man in jedem dieser Streifen die lineare Ordnung "liegt oberhalb von" auf den Kantenstücken definieren, da sie sich im Innern eines Streifens nicht schneiden. Jedem Knoten ordnet man nun einen balancierten Suchbaum zu, in den die Kantenstücke des ihm nachfolgenden Streifens gemäß obiger Ordnung eingefügt wurden. Alle Kantenstücke werden zudem mit den (beiden) Indizes der sie erzeugenden Funktionen markiert. Durch zweifache Binärsuche kann man nun jedem Punkt p der Ebene in Zeit $O(\log(n))$ die Indexmenge der in p aktiven Funktionen zugeordnet werden. Mit dieser Information findet man den Funktionswert $f^u(p)$ in konstanter Zeit.

5.1.5 Lage der Extrema der unteren Einhüllenden

Wie bereits beobachtet, können lokale Extrema der unteren Einhüllenden als lokale Extrema der Abstandsfunktion, als lokales Extremum auf einem Separator, als Schnittpunkt dreier Funktionen der Funktionenschar oder als Randextremum auftreten. Die in diesem Abschnitt definierten Abstandsfunktionen h_i sind sämtlich auf der kompakten Einheitskugel erklärt; somit entfällt die Problematik der Randextrema. Extrema der dritten Art können leicht mit Hilfe der Minimierungsdiagramme gefunden werden. Bleiben noch die erstgenannten Fälle. Die Techniken zum Aufspüren solcher Extrema kennen wir bereits: Nullsetzen des Gradienten bzw. Aufstellen der Kuhn-Tucker-Bedingungen. Dazu verwenden wir jedoch nicht die ursprüngliche, über der Einheitskugel definierten Funktionen h_i , sondern die projizierten Distanzfunktionen f_i. Um zu garantieren, daß alle Extrema gefunden werden sind im ungünstigsten Fall die Projektionen in alle 6 Achsenrichtungen zu berücksichtigen; die Zahl kann man jedoch häufig durch Symmetrie- und Analogiebetrachtungen verringern.

Die Korrektheit dieses Vorgehens zeigt das

Lemma 12 (Extremallemma) *Seien $S, T \subset I\!R^d$ und es existiere eine bijektive, stetige Abbildung $\pi : S \to T$. Sei $h : S \to I\!R$ eine stetige Funktion und sei $f : T \to I\!R$ definiert durch $f := h \circ \pi^{-1}$. Dann ist $x \in S$ genau dann lokales Maximum (Minimum) von h, wenn $\pi(x) \in T$ lokales Maximum (Minimum) von f ist.*

Beweis: Sei $x^* \in S$ lokales Maximum von h, d. h. es gibt eine Umgebung $U(x^*)$ mit $h(x^*) \geq h(x)\ \forall x \in U(x^*) \cap S$. Sei $y^* = \pi(x^*)$; y^* existiert und ist eindeutig bestimmt aufgrund der Bijektivität von π. Aus der Stetigkeit von π^{-1} folgt die Existenz einer Umgebung $V(y^*)$ mit der Eigenschaft $\pi^{-1}(y) \in U(x^*)\,\forall y \in V(y^*) \cap T$. Sei $y \in V(y^*)$ beliebig. Es gilt

$$\begin{aligned} f(y) &= h(\pi^{-1}(y)) = h(x) \\ &\leq h(x^*) && \text{da } x \in U(x^*) \\ &= h(\pi^{-1}(y^*)) = f(y^*) \end{aligned}$$

und die Maximalität von y^* auf $V(y^*)$ ist gezeigt. Der Beweis der Rückrichtung verläuft analog, die Betrachtung lokaler Minima ebenfalls. ∎

Anschaulich besagt dies, daß eine bijektive, stetige Deformation des Definitionsbereichs einer Funktion - in unserem Fall eine Projektion - nichts an der Extremaleigenschaft der Funktionswerte ändert.

Doch selbst dieses modifizierte Verfahren birgt rechnerische Komplikationen. Hier erweist sich folgende Überlegung als außerordentich nützlich. Beschreiben wir die Lage der Extrema als Schnittpunkte geometrischer Objekte, ist diese Charakterisierung von der Lage des Koordinatensystems unabhängig. Dies bedeutet insbesondere, daß wir das Koordinatensystem in eine in Bezug auf die vorgegebenen Weltpunkte günstige Lage drehen können, um dann mit Hilfe des Extremallemmas die Lage der (potentiellen) Extrema zu bestimmen.

Extrema der Distanzfunktion

Unsere geometrische Anschauung legt nahe, daß die Abstandsfunktion h_i zu einem Punkt $p_i \in S^2$ ein lokales Minimum im Punkt p_i selbst aufweist, in dem Antipoden des Punktes p_i maximal wird, ansonsten aber keine lokalen Extrema besitzt. Eine einfache Überlegung bestätigt diese Vermutung. Zu jedem Punkt q der nicht p_i oder dessen Antipode ist, betrachte man den eindeutig bestimmten Großkreis $K_{p_i q}$. Wandert man nun auf diesem Großkreis von p_i über q auf $Antipode(p_i)$ zu, wächst der Wert der Abstandsfunktion gemäß Definition streng monoton; h_i kann also in q nicht extremal werden. Dieses Ergebnis kann man auch durch Nullsetzen des Gradienten der projizierten Abstandsfunktion f_i gewinnen. Um die gesamte Kugeloberfläche auf Extrema untersuchen zu können, legen wir das Koordinatensystem so, daß p_i nacheinander auf den Nordpol, Südpol und Äquator der Kugel fällt und projizieren jeweils auf die Tangentialebene $E_\pi : x_3 = 1$. Dabei bezeichne p_i' den transformierten Punkt.

FALL 1 $p_i' = (0, 0, 1)$ [Nordpol]

Für die projizierte Abstandsfunktion und ihren Gradienten ergibt sich mit der Abkürzung $N = \sqrt{x^2 + y^2 + 1}$

$$f_i = 2 - \frac{2}{N} \qquad \nabla f_i = \frac{2}{N^3}\begin{pmatrix} x \\ y \end{pmatrix}$$

Der Gradient verschwindet nur in (0,0), einem Minimum. Dies ist der Punkt p_i' selbst.

FALL 2 $p_i' = (0, 0, -1)$ [Südpol]

$$f_i = 2 + \frac{2}{N} \quad \nabla f_i = \frac{-2}{N^3}\binom{x}{y}$$

Der Gradient verschwindet nur in (0,0), einem Maximum. Dies ist der Antipode des Punktes p_i'.

FALL 3 $p_i' = (0, 1, 0)$ [Äquator]

$$f_i = 2 - \frac{2y}{N} \quad \nabla f_i = \frac{2}{N^3}\binom{xy}{-x^2 - 1}$$

Wir finden keine stationären Punkte. Dies gilt aus Symmetriegründen auch für alle anderen Äquatorlagen.

Die Extrema der Abstandsfunktion h_i liegen also in den Schnittpunkten der Ursprungsgeraden $0p_i$ mit der Einheitskugel.

Existiert jedoch neben p_i ein weiterer Weltpunkt p_j, kann das Maximum von h_i bereits nicht mehr auf $h^u[\mathcal{H}]$ auftreten, da h_j an dieser Stelle sicher einen geringeren Wert aufweist. Da wir $|S| \geq 2$ annehmen können (der Fall $|S| = 1$ ist trivial), werden wir diese Maxima nicht weiter berücksichtigen.

Extrema auf den Separatoren

Wir haben gesehen, daß der Separator der Abstandsfunktionen h_i und h_j zu den beiden Punkten p_i und p_j der Großkreis $K^{p_i - p_j}$ ist. Die geometrische Anschauung läßt uns vermuten, daß die lokalen Extrema der Abstandsfunktion h_i über dem Separator in den Schnittpunkten des Separators $K^{p_i - p_j}$ mit dem Großkreis $K_{p_i p_j}$, der die Punkte p_i und p_j enthält, zu finden sind. Diese Vermutung bestätigen wir auf analytischem Wege durch Anwendung des Extremallemmas mit Hilfe der projizierten Abstandsfunktionen und Separatoren. Um den gesamten Separator auf lokale Extrema untersuchen zu können, legen wir das Koordinatensystem zunächst so, daß p_i auf den Kugelnordpol (-südpol) und p_j in die xz-Ebene fällt (Fälle 1 und 2), und schließlich derart, daß p_i auf dem Äquator und p_j in der xy-Ebene zu liegen kommt und projizieren auf E_π. Alle anderen Fälle können aufgrund von Symmetriebetrachtungen entfallen.

FALL 1 $p_i' = (0, 0, 1), p_j' = (x_j', 0, z_j') \quad [x_j' \neq 0]$

Für die Abstandsfunktion und ihren Gradienten erhalten wir

$$f_i = 2 - \frac{2}{N} \quad \nabla f_i = \frac{2}{N^3}\binom{x}{y}$$

Die Separatorgleichung ergibt sich zu

$$sep_{ij} : x'_j x + z'_j - 1 = 0$$

Nullsetzen des Gradienten der Lagrangefunktion führt zu dem Gleichungssystem

$$\begin{aligned} \frac{2x}{N^3} + x'_j \lambda &= 0 \\ \frac{2y}{N^3} &= 0 \\ x'_j x + z'_j - 1 &= 0 \end{aligned}$$

mit der Lösung $(\frac{1-z'_j}{x'_j}, 0)$, dem Bildpunkt eines Schnittpunktes der Großkreise $K^{p_i - p_j}$ und $K_{p_i p_j}$.

FALL 2 $p'_i = (0,0,-1), p'_j = (x'_j, 0, z'_j) \quad [x'_j \neq 0]$

Analog zu Fall 1 impliziert die Methode der Lagrangemultiplikatoren, ein Gleichungssystem, dessen einzige Lösung $(\frac{-z'_j - 1}{x'_j}, 0)$ ebenfalls Bildpunkt eines Schnittpunkts der Großkreise $K^{p_i - p_j}$ und $K_{p_i p_j}$ ist.

FALL 3 $p'_i = (0,1,0), p'_j = (x'_j, y'_j, 0) \quad [x'_j \neq 0]$

$$f_i = 2 - \frac{2y}{N} \quad \nabla f_j = \frac{2}{N^3}\begin{pmatrix} xy \\ -x^2 - 1 \end{pmatrix}$$

$$sep_{ij} : x'_j x + (y'_j - 1)y = 0$$

Das zugehörige Gleichungssystem

$$\begin{aligned} \frac{2xy}{N^3} + x'_j \lambda &= 0 \\ \frac{-2(x^2+1)}{N^3} + (y'_j - 1)\lambda &= 0 \\ x'_j x + (y'_j - 1)y &= 0 \end{aligned}$$

hat keine Lösungen. Es existieren also keine weiteren lokalen Extrema.

Die beiden gefundenen Extrema sind wie alle Schnittpunkte zweier Großkreise Antipoden. Sie halbieren die beiden durch p_i und p_j getrennten Bögen des $K_{p_i p_j}$, wobei das den kürzeren Bogen teilende Extremum ein Minimum, der andere ein Maximum darstellt.

Weiterhin können lokale Extrema der unteren Einhüllenden auch in Punkten auftreten, in denen sich die Distanzfunktionen h_i, h_j und h_k dreier Punkte p_i, p_j und p_k schneiden. In solchen Punkten schneiden sich auch deren Separatoren, die Großkreise $K^{p_i - p_j}, K^{p_i - p_k}$ und $K^{p_j - p_k}$ und dieser Schnittpunkt erscheint, sofern der Schnitt auf der unteren Einhüllenden liegt, bei geeigneter Projektionsrichtung als Knoten des MD der projizierten Distanzfunktionen.

Wir haben nun für alle kritischen Punkte, d. h. möglichen lokalen Maxima der unteren Einhüllenden von $\mathcal{H}$ Charakterisierungen angegeben, die sich nur auf die Lage der Weltpunkte auf der Kugel beziehen, dabei aber unabhängig von der Lage der Projektionsebenen sind: ein Extrempunkt kann als Schnitt einer Geraden mit der Einheitskugel bzw. als Schnittpunkt zweier oder dreier Großkreise beschrieben werden. Dies erlaubt, die Lage der Maxima der projizierten Distanzfunktionen $\mathcal{F}$ bei beliebiger Lage der Projektionsebenen effizient zu bestimmen.

5.1.6 Der MinMax-Algorithmus

Unser Algorithmus zur Lösung des (P16) verfährt nun nach folgender Strategie

Schritt 1 Projiziere die obere Hemisphäre auf die nördliche Tangentialebene $E_\pi : x_3 = 1$ und berechne das Minimierungsdiagramm der projizierten Abstandsfunktionen MD[$\mathcal{F}_1$]. (Man beachte, daß sich der Begriff "obere Hemisphäre" nur auf die mögliche Lage des gesuchten Optimums p^* bezieht; jeder der n Weltpunkte trägt unabhängig von seiner Lage eine Distanzfunktion f_i zu $\mathcal{F}_1$ bei.)

Schritt 2 Bestimme die Lage der kritischen Punkte der unteren Einhüllenden von $\mathcal{F}_1$ mit Hilfe des MD[$\mathcal{F}_1$].

Schritt 3 Projiziere die untere Hemisphäre auf die südliche Tangentialebene $E_\sigma : x_3 = -1$ und berechne das Minimierungsdiagramm der projizierten Abstandsfunktionen MD[$\mathcal{F}_2$].

Schritt 4 Bestimme die Lage der kritischen Punkte auf der unteren Einhüllenden von $\mathcal{F}_2$ mit Hilfe des MD[$\mathcal{F}_2$].

Schritt 5 Bestimme das Maximum der unteren Einhüllenden der Abstandsfunktionen auf dem Äquator.

Schritt 6 Finde unter den Kandidatenpunkten aus den Schritten 2, 4 und 5 das globale Optimum p^*.

Wir fügen nun die notwendigen Details ein und analysieren die Laufzeit der einzelnen Schritte.

Die projizierten Abstandsfunktionen können in **Schritt 1** aus den Punktkoordinaten in linearer Zeit ermittelt werden. Sie erfüllen die Bedingungen der Sätze 13 und 14, sofern die Weltpunkte allgemeine Lage einnehmen; ihr MD läßt sich demnach mit dem Algorithmus von Schwartz/Sharir in Zeit $O(n \log n)$ berechnen.

In **Schritt 2** bestimmt man für alle $O(n)$ Kanten des MD jeweils in konstanter Zeit das Maximum des zugehörigen Separators und testet in Zeit $O(\log n)$ pro Maximum mittels Point Location im MD, ob das Maximum auf der unteren Einhüllenden der Funktionenschar $\mathcal{F}_1$ auftritt. Diejenigen Maxima, für die dies zutrifft, bilden zusammen mit den Knoten des MD die $O(n)$ kritischen Punkte der unteren Einhüllenden von $\mathcal{F}_1$.

Die **Schritte 3** und **4** verlaufen völlig analog zu den Schritten 1 und 2.

Den bestmöglichen Punkt auf dem Äquator kann man unter Verwendung des MaxMin-Algorithmus für Funktionen in einer Variablen aus Kapitel 3 finden. Der Abstand eines Äquatorpunktes $p = (x, y, 0)$ vom Weltpunkt $p_i = (x_i, y_i, z_i)$ läßt sich als Funktion in x über dem Intervall $[-1, 1]$ beschreiben

$$\delta^2(p_i, p) = \begin{cases} f_i^+(x) = 2 - 2xx_i - 2\sqrt{1-x^2}y_i & \text{falls } y \geq 0 \\ f_i^-(x) = 2 - 2xx_i + 2\sqrt{1-x^2}y_i & \text{falls } y < 0 \end{cases}$$

Das MaxMin-Problem der Funktionenscharen $\mathcal{F}^+ = \{f_i^+ \mid 1 \leq i \leq n\}$ und $\mathcal{F}^- = \{f_i^- \mid 1 \leq i \leq n\}$ kann in Zeit $O(n \log n)$ gelöst werden, da die Funktionen beider Scharen nur konstante Anzahl kritischer Punkte aufweisen und sich paarweise in höchstens 2 Punkten schneiden.

Zusammenfassend finden wir den

Satz 15 *Das größte leere Kreis auf der Einheitskugel (P16) kann in Zeit* $O(n \log n)$ *berechnet werden, sofern keine 4 Weltpunkte auf einem Kugelkreis liegen.*

5.1.7 Duale Problemstellung

Der Algorithmus von Schwartz/Sharir ist gleichermaßen geeignet, die Projektion der oberen Einhüllenden der Schar der Abstandsfunktionen in die xy-Ebene, ihr *Maximierungsdiagramm*, zu berechnen. Weiterhin haben wir die mögliche Lage der kritischen Punkte der oberen Einhüllenden bereits untersucht. Die Schritte 1-4 des obigen Algorithmus können also auf einfache Weise modifiziert werden, um das globale Optimum der oberen und unteren Hemisphäre zu finden. Darüber hinaus bereitet auch die verbleibende eindimensionale MinMax-Aufgabe über dem Äquator keine Schwierigkeiten; sie kann genau wie ihre duale MaxMin-Aufgabe durch Anwenden der Divide-and-Conquer-Strategie aus Kapitel 3 gelöst werden.

Da alle erforderlichen Operationen den gleichen Zeitschranken genügen wie die entsprechenden des dualen Maximin-Problems folgt der

Satz 16 *Der kleinste umschließende Kreis auf der Einheitskugel kann in Zeit* $O(n \log n)$ *berechnet werden, sofern keine 4 Weltpunkte auf einem Kugelkreis liegen.*

5.2 Verankerte Gerade im Raum

5.2.1 Motivation

Ausgangspunkt unserer Beschäftigung mit Maximin-Optimierungsaufgaben war eine komplexe Problematik aus der neurochirurgischen Praxis: die Planung stereotaktischer Gehirnoperationen. Stereotaxie ist eine Operationstechnik, bei der eine Sonde durch eine kleine Öffnung in der Schädeldecke des Patienten geradlinig bis zu einem vorgegebenen Zielpunkt im Gehirn eingeführt wird. Diese Sonde kann dann je nach Operationszweck dazu dienen, Gewebeproben zu diagnostischen Zwecken zu entnehmen, Hämatome zu entfernen oder radioaktive Präparate zur Bestrahlung eines Tumors zu implantieren. Die Sonde wandert durch einen nadelförmigen Kanal, welcher Teil eines am Schädel des Patienten fixierten mechanischen Apparates ist, der die exakte Ausrichtung des Kanals auf den Zielpunkt erlaubt. Die Koordinaten des Zielpunktes werden hierzu mit modernen bildgebenden Verfahren wie Computertomographie oder Magnetresonanz relativ zum stereotaktischen Apparat ermittelt. Um die Gefährdung des Patienten zu minimieren und Gehirnschäden auszuschließen, ist es notwendig, den Eintrittspunkt der Sonde so zu wählen, daß ihre Trajektorie weder größere Blutgefäße oder wichtige Hirnnerven tangiert, noch wesentliche Hirnareale wie die motorische Rinde verletzt. Man minimiert somit das Operationsrisiko, wenn man einen Zugang sucht, der den minimalen Abstand des Sondenkanals zur Menge der gefährdeten Objekte maximiert. Modelliert man den Sondenkanal durch eine Gerade (Strahl), die zu vermeidenden Objekte durch (gewichtete) Punktmengen, gelangt man unmittelbar zum Problem der verankerten Maximin-Geraden in drei Dimensionen. Die Gewichte können dabei entweder die Ausdehnung der Objekte widerspiegeln, oder deren unterschiedliche Bedeutung repräsentieren.

5.2.2 Problemstellung

Gegenstand der Untersuchungen dieses Abschnitts wird also der verankerte Unterfall $d = 3,\ k = 1$ der allgemeinen Problemstellung sein. Wir notieren ihn als

PROBLEM 17 : [Maximin-Ursprungsgerade im Raum]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\},\ p_i = (x_i, y_i, z_i)$ von n Punkten des $\mathbb{R}^3$ mit positiven Gewichtsfaktoren $W = \{w_1, \ldots, w_n\}$.

Gesucht: Eine Ursprungsgerade $g_u^* : x = \alpha u^*$ mit Richtungsvektor u^*, die den minimalen gewichteten euklidischen Abstand zur Punktmenge S

$$\min_{1 \leq i \leq n} \delta(p_i, g_u)$$

maximiert.

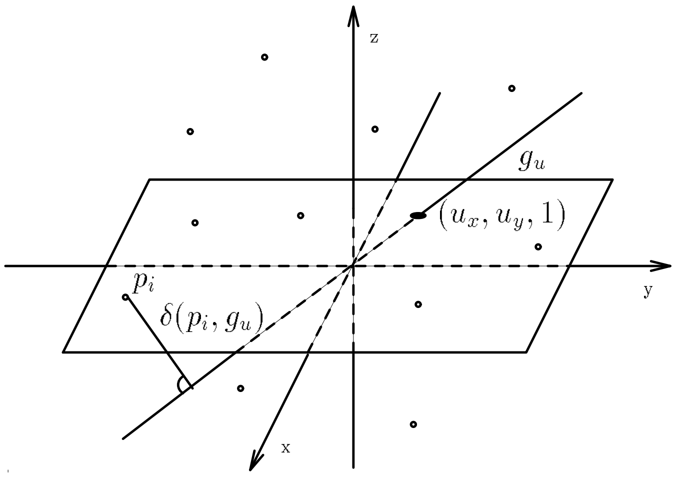

Abbildung 5.4: Maximin-Ursprungsgerade im $\mathbb{R}^3$

Die zweidimensionale Variante dieses Problems haben wir zusammen mit einigen Verallgemeinerungen gleichsam als Vorbetrachtung bereits in Kapitel 2 diskutiert. In Analogie zur Vorgehensweise im zweidimensionalen Fall können wir uns durch Übergang zur transformierten Punktmenge $S_W = \{(w_i x_i, w_i y_i, w_i z_i) \mid 1 \leq i \leq n\}$ der multiplikativen Gewichtsfaktoren entledigen. Wir beschränken unsere Betrachtungen demgemäß auf ungewichtete Probleminstanzen.

Da von zwei linear abhängigen Punkten stets der ursprüngsnähere den geringeren Abstand zu einer Ursprungsgeraden aufweist und der ursprungsfernere daher nichts zur Problemlösung beiträgt, nehmen wir an, daß alle Weltpunkte paarweise linear unabhängig sind.

Funktionale Fassung

Wir beginnen wie gewohnt, indem wir das Problem als MaxMin-Aufgabe über einem geeignet gewählten Konfigurationsraum formulieren.

Der euklidische Abstand eines Punktes $p_i = (x_i, y_i, z_i)$ von einer Ursprungsgeraden g_u mit Richtungsvektor $u = (u_x, u_y, u_z)$ ist gegeben durch

$$\begin{aligned}\delta(g_u, p_i) &= \frac{1}{|u|}|p_i \times u| \\ &= \sqrt{\frac{(y_i u_z - z_i u_y)^2 + (z_i u_x - x_i u_z)^2 + (x_i u_y - y_i u_x)^2}{u_x^2 + u_y^2 + u_z^2}}\end{aligned}$$

Obwohl der Objektraum des Problems dreidimensional ist, handelt es sich um eine Optimierungsaufgabe mit nur zwei Freiheitsgraden: kennt man den Längen- und Breitenwinkel eines Punktes, in dem die Gerade die Einheitskugel S^2 durchstößt, ist die Lage der Geraden eindeutig bestimmt. Die Wahl der Einheitskugel als Konfigurationsraum erscheint somit naheliegend. Wir gehen jedoch mit ähnlicher Begründung wie in Abschnitt 5.1 einen anderen Weg und beschreiben die Gerade mit Hilfe der Koordinaten ihres Schnittpunktes mit der Ebene $E_\pi : x_3 = 1$. Dies ist genau dann möglich, wenn die Gerade nicht in der Äquatorebene der Einheitskugel liegt; dieses Randproblem stellen wir zunächst zurück.

Nun definiert jeder Punkt $p_i \in S$ eine Abstandsfunktion in den Koordinaten x und y des Konfigurationsraums E_π, die das Quadrat seines Abstands zur Ursprungsgerade durch $(x, y, 1)$ beschreibt

$$\begin{aligned} f_i(x,y) &= \delta^2(p_i, g_{(x,y,1)}) \\ &= \frac{(y_i - z_i y)^2 + (z_i x - x_i)^2 + (x_i y - y_i x)^2}{x^2 + y^2 + 1} \end{aligned}$$

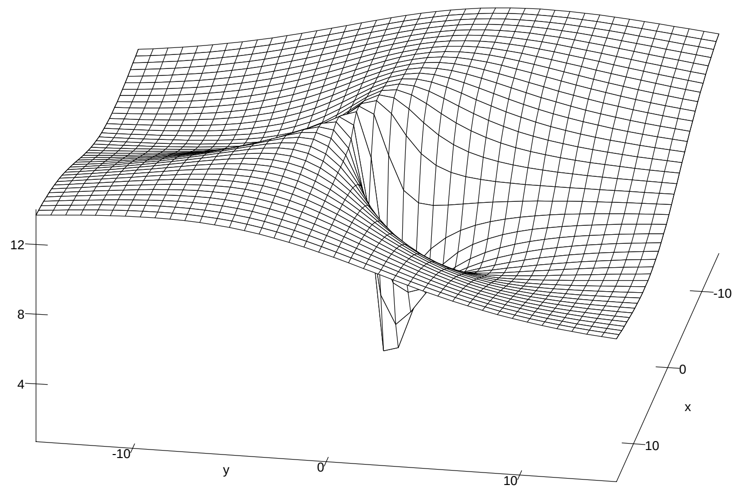

Abbildung 5.5: Graph einer Abstandsfunktion f_i

Unsere Aufgabe wird sein, das MaxMin-Problem der Funktionenschar $\mathcal{F} = \{f_1, \ldots, f_n\}$ über der Projektionsebene E_π zu lösen.

5.2.3 Struktur und Berechnung des MD

Unserem allgemeinen Lösungsschema folgend, wenden wir uns zunächst dem Minimierungsdiagramm der Funktionenschar $\mathcal{F}$ zu. Insbesondere werden wir in diesem Abschnitt Aussagen über seine Struktur und Komplexität treffen können. Die hierbei gewonnenen Informationen sind für einen effizienten Algorithmus zur Konstruktion des MD von fundamentaler Bedeutung.

Gestalt der Separatoren

Die Graphen der Abstandsfunktionen des (P17) haben keine einfache Gestalt wie etwa die Rotationsparaboloide des (P14) aus Kapitel 3. Abbildung 5.5 zeigt ein Beispiel eines solchen Graphen. Wir werden die Separatorgleichungen daher nicht auf rein algebraischem Weg herleiten, sondern eine Vorbetrachtung voranstellen, um die geometrische Anschauung zu stützen.

Lemma 13 *Seien $p_i, p_j \in I\!R^3$ zwei linear unabhängige Punkte.*

Haben die beiden Punkte unterschiedlichen Abstand zum Ursprung ($|p_i| \neq |p_j|$), so bildet die Schar der zu p_i und p_j äquidistanten Ursprungsgeraden einen elliptischen Kegel um den Ursprung. Bei gleichem Ursprungsabstand ($|p_i| = |p_j|$) zerfällt die von den äquidistanten Ursprungsgeraden erzeugte Fläche in die Ursprungsebenen $E^{p_i - p_j}$ und $E^{p_i + p_j}$, deren Normalenvektoren $p_i - p_j$ und $p_i + p_j$ senkrecht aufeinander stehen.

Beweis: Gleichheit der Abstandsfunktionen $f_i(u) = f_j(u)$ impliziert

$$\begin{aligned} 0 &= (y_i^2 + z_i^2 - y_j^2 - z_j^2)x^2 + (x_i^2 + z_i^2 - x_j^2 - z_j^2)y^2 \\ &\quad +(x_i^2 + y_i^2 - x_j^2 - y_j^2)z^2 - 2(x_i z_i - x_j z_j)xz \\ &\quad -2(x_i y_i - x_j y_j)xy - 2(y_i z_i - y_j z_j)yz \end{aligned}$$

wobei $u = (x, y, z)$. Dies ist die Gleichung einer Fläche zweiten Grades im Raum (vgl. Anhang).

FALL 1 $|p_i| \neq |p_j|$

Wir drehen das Koordinatensystem so, daß p_i auf die z-Achse und p_j in die xz-Ebene fallen. Seien $p_i' = (0, 0, z_i'), p_j' = (x_j', 0, z_j')$ die transformierten Punktkoordinaten. Die Fläche wird dann im neuen Koordinatensystem durch die Gleichung

$$0 = (z_i'^2 - z_j'^2)x^2 + (z_i'^2 - x_j'^2 - z_j'^2)y^2 - x_j'^2 z^2 + 2x_j' z_j' xz$$

beschrieben. Dies ist nach Division durch $-(z_i'^2) \neq 0$ mit den Ersetzungen $a = x_j'/z_i'$ und $b = z_j'/z_i'$ äquivalent zu

$$0 = (b^2 - 1)x^2 + (a^2 + b^2 - 1)y^2 + a^2 z^2 - 2abxz$$

Als Invarianten der Fläche (vgl. Anhang) erhalten wir

$$\Delta = 0,\ \delta = -a^2(a^2 + b^2 - 1),\ S = 2(a^2 + b^2 - 1)$$

Aus der linearen Unabhängigkeit von p_i und p_j folgt $a \neq 0$. Aus $|p_i| \neq |p_j|$ folgt $1 \neq a^2 + b^2$ und damit $a^2 + b^2 - 1 \neq 0$. Dies impliziert $\delta \neq 0$.

Weiter gilt $S \cdot \delta = -2a^2(a^2 + b^2 - 1)^2 < 0$ und wir können die Fläche gemäß Tabelle A.2 als elliptischen Kegel klassifizieren.

FALL 2 $|p_i| = |p_j|$

Wir drehen das Koordinatensystem derart, daß p_i und p_j in die xy-Ebene fallen und die y-Achse die Strecke $\overline{p_i p_j}$ halbiert. Sind $p_i' = (x_i', y_i', 0)$ und $p_j' = (x_j', y_j', 0)$ die transformierten Punktkoordinaten, so gilt $|x_i'| = |x_j'|$ und $|y_i'| = |y_j'|$. Die Fläche erfüllt im transformierten KS die Gleichung

$$(x_i' y_i' - x_j' y_j')xy = 0$$

$x_i' y_i' = x_j' y_j'$ impliziert im Widerspruch zur Voraussetzung die lineare Abhängigkeit von p_i und p_j. Einzige Lösungen sind damit die Ebenen $x = 0$ und $y = 0$, die mit den Ebenen $E^{p_i - p_j}$ bzw. $E^{p_i + p_j}$ identisch sind. $<p_i - p_j, p_i + p_j> = 0$ folgt unmittelbar aus $|p_i| = |p_j|$. ∎

Notation: Wir bezeichnen die Schar der zu zwei Punkten p_i und p_j äquidistanten Ursprungsgeraden, ihren *Äquidistanzkegel*, mit $kegel_{ij}$, und dessen Schnitt mit der Projektionsebene $E_\pi : x_3 = 1$ als Separator sep_{ij}.

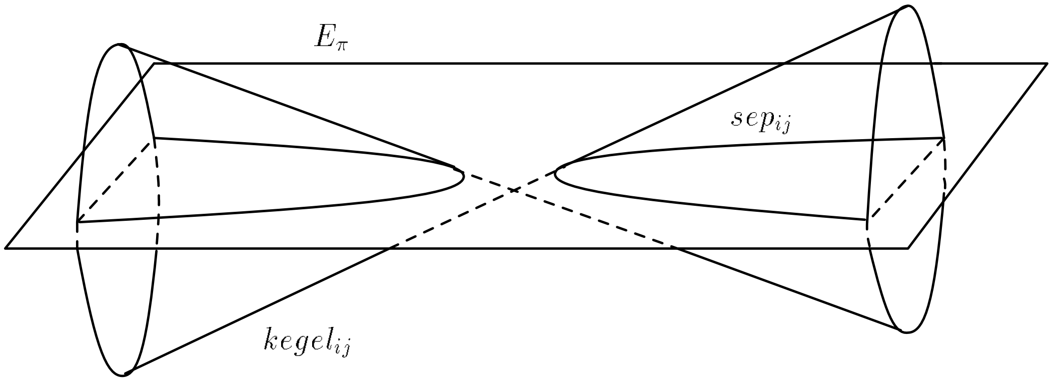

Abbildung 5.6: Äquidistanzkegel und Separator zweier Weltpunkte

Man sieht leicht, wie diese Notation mit der Definition von sep_{ij} aus Kapitel 3 einhergeht: sep_{ij} ist der Separator der über dem Konfigurationsraum E_π definierten Abstandsfunktionen $f_i(x, y)$ und $f_j(x, y)$. Er genügt der Gleichung

$$\begin{aligned} 0 \;=\; & (y_i^2 + z_i^2 - y_j^2 - z_j^2)x^2 + (x_i^2 + z_i^2 - x_j^2 - z_j^2)y^2 \\ & -2(x_i z_i - x_j z_j)x - 2(y_i z_i - y_j z_j)y \\ & -2(x_i y_i - x_j y_j)xy + x_i^2 + y_i^2 - x_j^2 - y_j^2 \end{aligned}$$

und ist somit ein Kegelschnitt (vgl. Anhang).

Die Kanten des Minimierungsdiagramms der Abstandsfunktionen sind also Kegelschnittbögen. Ein einzelner Separator kann die Ebene bereits in zwei nicht zusammenhängende Regionen zerteilen. Dies läßt uns vermuten, daß solche MD beträchtliche Komplexität aufweisen können. Abbildung 5.7 zeigt ein Beispiel eines solchen Minimierungsdiagramms.

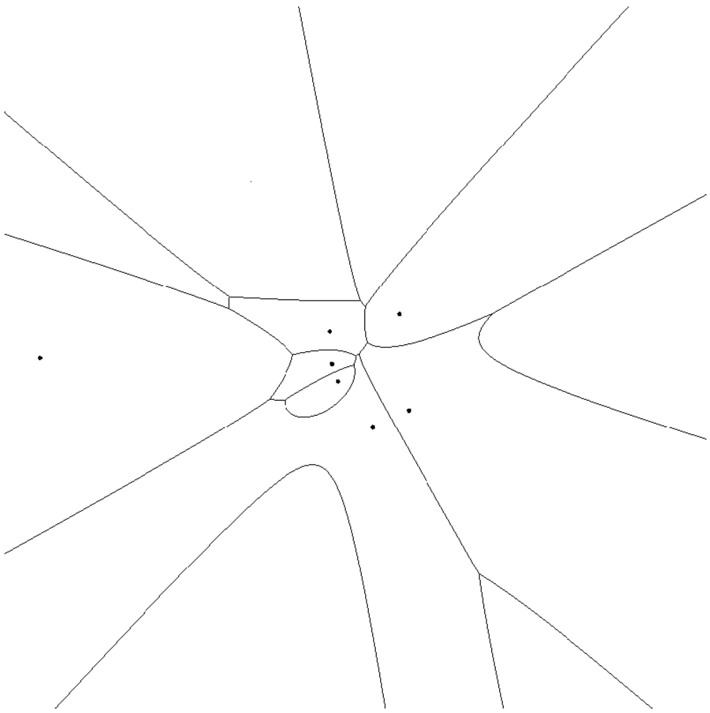

Abbildung 5.7: Minimierungsdiagramm von 10 Weltpunkten. Die Punkte markieren die Bilder der Weltpunkte unter Zentralprojektion auf E_π, soweit diese in den Bildausschnitt fallen.

Komplexität des MD

Als Grundlage unserer Komplexitätsaussagen zitieren wir aus [SS90] den

Satz 17 (Schwartz,Sharir) *Sei $\mathcal{F}_\mathcal{I} = \{f_i \mid i \in I\}$ eine endliche Schar reellwertiger, stetig differenzierbarer Funktionen über $\mathbb{R}^2$ in allgemeiner Lage. Weiterhin gelte*

(i.) Für alle $i, j \in I$, $(i \neq j)$ und jedes $x_0 \in \mathbb{R}$ hat die Gleichung $f_i(x_0, y) = f_j(x_0, y)$ höchstens zwei Lösungen.

Wir bezeichnen die kleinere der beiden Lösungen mit $r_{ij}^-(x_0)$*, die größere mit* $r_{ij}^+(x_0)$*. Existiert nur eine Lösung, so setzen wir* $r_{ij}^-(x_0) = r_{ij}^+(x_0)$*, ansonsten sind beide Ausdrücke undefiniert. Eine Sprungstelle oder ein Randpunkt des Definitionsbereichs von* r_{ij}^- $[r_{ij}^+]$ *heißt* singulärer Punkt *von* r_{ij}^- $[r_{ij}^+]$.

(ii.) Für alle $i \neq j$ *haben die Funktionen* $r_{ij}^-(x)$ *und* $r_{ij}^+(x)$ *höchstens* t *singuläre Punkte.*

(iii.) Für alle $i \neq j \neq k \in I$ *haben die Gleichungen* $f_i(x,y) = f_j(x,y) = f_k(x,y)$ *höchstens* s *Lösungen.*

Dann ist $\mathcal{K}(MD[\mathcal{F}])$ *beschränkt durch* $O(n\lambda_{s+2}(n))$*, wobei der Proportionalitätsfaktor von* s *und* t *abhängt.*

Beweis: Die Funktionen aus $\mathcal{F}$ nehmen nach Voraussetzung allgemeine Lage ein; die Knoten und Kanten ihres MD bilden daher wie bereits in Kapitel 4 bemerkt einen planaren Graphen. Nach dem Eulerschen Satz beschränkt eine obere Schranke für die Anzahl der Knoten eines planaren Graphen bis auf einen konstanten Faktor auch die Komplexität des gesamten Graphen. Es genügt also, die maximal mögliche Anzahl von Knoten des MD[$\mathcal{F}$] abzuschätzen.

Wir erläutern zunächst einige Begriffe. Man sagt, zwei Funktionen f_i, f_j stehen über einem Intervall $Int \subset \mathbb{R}$ in *eindeutiger Lagebeziehung* zueinander, wenn eine der Beziehungen $f_i < f_j$ oder $f_i > f_j$ über dem gesamten Intervall gilt. Stetige Funktionen, die sich nur in endlich vielen Punkten schneiden, stehen in den durch ihre Schnittpunkte begrenzten offenen endlichen oder unendlichen Teilintervallen ihres gemeinsamen Definitionsbereichs in eindeutiger Lagebeziehung. Für festes $x_0 \in \mathbb{R}$ stehen somit auch $f_i(x_0, y)$ und $f_j(x_0, y)$ als Funktionen von y in den von $r_{ij}^-(x_0)$, $r_{ij}^+(x_0)$ begrenzten Intervallen in eindeutiger Lagebeziehung, falls sowohl r_{ij}^+ als auch r_{ij}^- an der Stelle x_0 definiert sind.

Wir führen eine neue Notation ein, um den Definitionsbereich der r_{ij} zu erweitern und die Doppeldeutigkeit $r_{ij}^\pm = r_{ji}^\pm$ aufzulösen.
$\phi_{ij}^-(x_0)$ bezeichne das größte $y_0 \in \mathbb{R} \cup \{+\infty, -\infty\}$, so daß $f_i > f_j \; \forall y < y_0$ gilt, $\phi_{ij}^+(x_0)$ bezeichne das kleinste $y_0 \in \mathbb{R} \cup \{+\infty, -\infty\}$, so daß $f_i > f_j \; \forall y > y_0$ erfüllt ist. Hat beispielsweise ϕ_{ij}^- an der Stelle x_0 einen endlichen Wert, so ist dieser gleich $r_{ij}^-(x_0)$ und $\phi_{ji}^- = -\infty$.

Sei nun $p = (x_0, y_0)$ ein Knoten des MD, d. h. es gibt drei verschiedene Indizes $i, j, k \in I$, so daß $f_i(p) = f_j(p) = f_k(p) = \min_{l \in I} f_l(p)$. p liegt nun entweder auf dem Graphen von r_{ij}^- oder r_{ij}^+ und damit stehen f_i und f_j entweder auf $]-\infty, y_0[$ oder auf $]y_0, \infty[$ in eindeutiger Lagebeziehung. Analoge Aussagen lassen sich für die Paare i, k und j, k machen. Wir haben also drei solcher Lageeziehungen und mindestens zwei von ihnen gelten auf demselben Intervall ($]-\infty, y_0[$ oder $]y_0, \infty[$).

Es ergeben sich zwei Fallgruppen:

FALL A: Alle Lagebeziehungen gelten auf demselben Intervall.

Betrachten wir zunächst den Fall $f_i > f_j,\ f_i > f_k$ auf $]y_0, \infty[$. Dann gilt $\phi^+_{ij}(x_0) = \phi^+_{ik}(x_0) = y_0$. Nehmen wir an, es gäbe einen Index $l \in I$, so daß $\phi^+_{il}(x_0) < y_0$. Dies impliziert nach Definition von ϕ^+_{il} $f_i > f_l$ auf $]\phi^+_{il}, \infty[$ und damit auf $[y_0, \infty[$. Daraus folgt $f_l(p) < f_i(p) = \min_{h \in I} f_h(p)$, ein Widerspruch (siehe Abbildung 5.8).

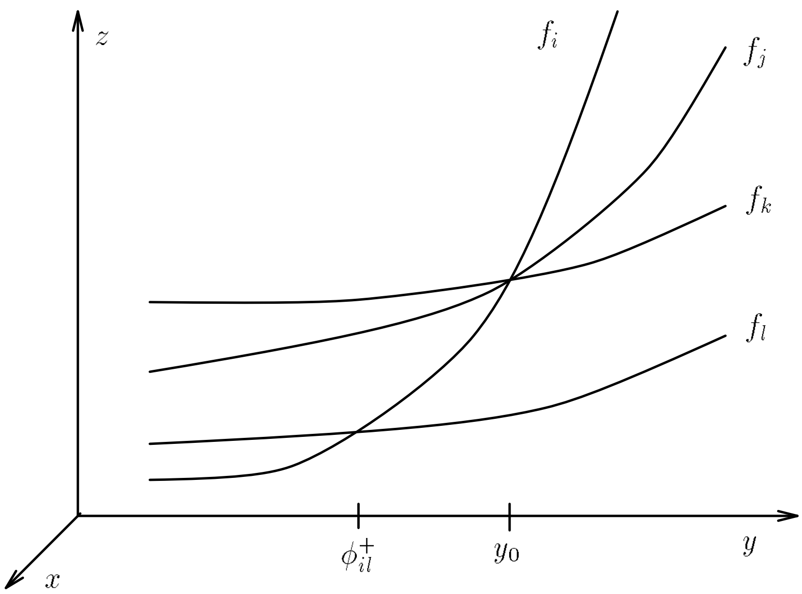

Abbildung 5.8: $\phi^+_{il} < y_0$ führt zum Widerspruch $f_l(p) < \min_{h \in I} f_h(p)$.

Wir können also sagen

$$\phi^+_{ij}(x_0) = \phi^+_{ik}(x_0) = \min_{l \in I} \phi^+_{il}(x_0) \quad (a)$$

Im Falle $f_i > f_j$ und $f_i > f_k$ auf $]-\infty, y_0[$ erhält man auf analoge Weise das Resultat

$$\phi^-_{ij}(x_0) = \phi^-_{ik}(x_0) = \max_{l \in I} \phi^-_{il}(x_0) \quad (b)$$

Alle weiteren Fälle dieser Gruppe lassen sich durch Umbenennen in einen der beiden oben ausgeführten Fälle überführen.

Erfüllt ein Punkt $p = (x_0, y_0)$ eine der beiden Bedingungen (a) oder (b), so heißt er *Dreifachschnitt vom Typ* A_i. Unser Ziel ist es nun, die maximal mögliche Anzahl solcher Dreifachschnitte abzuschätzen.

Dazu betrachten wir für ein festes $i \in I$ die untere Einhüllende $\psi^+_i(x)$ der Funktionen $\phi^+_{ij}(x)$ $(j \in I \setminus \{j\})$. Alle Punkte, die Bedingung (a) genügen, treten als Schnittpunkte zweier Funktionen auf dieser unteren Einhüllenden auf; ihre Anzahl ist also direkt durch die Komplexität von $\psi_i^+(x)$ gegeben. Um Satz 5 aus Abschnitt 3.3.3 anwenden zu können, müssen wir die Funktionen ϕ_{ij}^+ durch solche ersetzen, die den Voraussetzungen des Satzes genügen. Nach den Voraussetzungen für die Funktionen aus $\mathcal{F}_I$ zerfällt der Graph einer Funktion ϕ_{ij}^+ in höchstens $t+1$ Zusammenhangskomponenten; wir können folglich jedes ϕ_{ij}^+ durch höchstens $t+1$ stetige, partielle Funktionen darstellen, die jeweils über einem Intervall erklärt sind (Ist der Funktionswert von ϕ_{ij}^+ über einem der Intervalle $\pm\infty$, so benutzen wir als Ersatzfunktion die konstante Funktion $\pm(C+j)$, wobei C hinreichend groß gewählt wird).

Gilt $\phi_{ij}^+(x) = \phi_{ik}^+(x) = y$ für ein endliches y, so ist der Punkt (x, y) einer der höchstens s Schnittpunkte der Funktionen f_i, f_j und f_k. Folglich können sich auch die Teilfunktionen von ϕ_{ij}^+ und ϕ_{ik}^+ paarweise maximal in s Punkten schneiden. Mit Satz 5 erhalten wir $\lambda_{s+2}((t+1)n)$ als obere Schranke der Komplexität von ψ_i^+. Nach Lemma 4 gilt $\lambda_{s+2}((t+1)n) = O(\lambda_{s+2}(n))$. Durch ein völlig symmetrisches Argument zeigt man, daß auch die obere Einhüllende ψ_i^- der Funktionen ϕ_{ij}^- aus höchstens $O(\lambda_{s+2}(n))$ Teilstücken zusammengesetzt ist.

Damit ist die Anzahl der Schnittpunkte auf den Graphen von ψ_i^+ und ψ_i^- durch $O(\lambda_{s+2}(n))$ begrenzt. Es kann also auch nur $O(\lambda_{s+2}(n))$ Dreifachschnitte vom Typ A_i geben. Summiert man über alle Indizes $i \in I$, ergibt sich die Gesamtanzahl der Dreifachschnitte vom Typ A zu $O(n\lambda_{s+2}(n))$.

FALL B: Zwei der drei Lagebeziehungen gelten auf einem, die dritte auf dem anderen Intervall.

Nehmen wir an, auf dem Intervall $]y_0, \infty[$ gelten zwei Lagebeziehungen. f_i sei die (notwendigerweise existierende) Funktion, die in diesen beiden Beziehungen auftritt. Im Falle $f_i > f_j,\ f_i > f_k$ können wir die Argumentation aus Fall A übernehmen: es liegt ein Dreifachschnitt vom Typ A_i vor. Dasselbe trifft im Fall $f_j > f_i,\ f_i > f_k$ zu, nachdem wir die Indizes i und j vertauschen. Wir müssen also nur noch die Konstellation $f_i < f_j,\ f_i < f_k$ auf $[y_0, \infty[$ und z. B. $f_j < f_k$ ($f_j > f_k$ analog) auf $]-\infty, y_0[$ untersuchen.

Aus $f_k > f_i$ auf $]y_0, \infty[$ folgt entsprechend der Betrachtung im Fall A $\phi_{ki}^+(x_0) = \min_{l \in I} \phi_{kl}^+(x_0)$. p liegt also auf dem Graphen der unteren Einhüllenden $\psi_k^+ = \min_{l \in I} \phi_{kl}^+$. Ebenso folgt aus $f_k > f_j$ auf $]-\infty, y_0[$, daß p auf dem Graphen der oberen Einhüllenden ψ_k^- liegt. Alle Punkte, die auf dem Schnitt der beiden Graphen von ψ_k^+ und ψ_k^- liegen, heißen *Dreifachschnitt vom Typ B_k*.

Wieviele dieser Dreifachschnitte kann es geben? Betrachte für ein festes k die Partition der x-Achse in Intervalle, die durch die singulären Punkte der Funktionen ϕ_{kl}^+, ϕ_{kl}^-, und alle Dreifachschnitte vom Typ A_k induziert wird. Man erhält $O(n + \lambda_{s+2}(n)) = O(\lambda_{s+2}(n))$ Intervalle. Im Innern eines solchen Intervalls Int wird ψ_k^+ nur von genau einer Funktion angenommen, d. h es gibt genau ein $j \in I$ mit $\phi_{kj}^+(x) = \psi_k^+(x)$. Gleiches gilt für ψ_k^-: es gibt genau ein $i \in I$ mit $\phi_{ki}^-(x) = \psi_k^-(x)$. Damit ist jede Abszisse x_0

eines Schnittpunkts der Graphen von ψ_k^+, ψ_k^- über dem Innern von Int Lösung der Gleichung $\phi_{kj}^+(x_0) = \phi_{ki}^-(x_0)$. Folglich ist jeder dieser Schnittpunkte $p = (x_0, \psi_k^+(x_0))$ eine der höchstens s Lösungen der Gleichungen $f_i(p) = f_j(p) = f_k(p)$. Insgesamt gibt es also höchstens $O(s\lambda_{s+2}(n))$ Dreifachschnitte vom Typ B_k, also maximal $O(n\lambda_{s+2}(n))$ Dreifachschnitte vom Typ B. ∎

Wir zeigen nun, daß die Schar der Distanzfunktionen $\mathcal{F}$ den Voraussetzungen des Satzes genügt. Zur Erfüllung der Bedingung der allgemeinen Lage setzen wir dabei voraus, daß keine 4 Weltpunkte auf einem Zylinder liegen, dessen Achse durch den Ursprung verläuft.

Zunächst stellen wir fest, daß $\mathcal{F}$ ordnungsäquivalent zur Funktionenschar $\mathcal{F}^t = \{f_1^t, \ldots, f_n^t\}$ mit

$$\begin{aligned} f_i^t(x,y) &= (x^2+y^2+1)f_i(x,y) \\ &= (y_i - z_i y)^2 + (z_i x - x_i)^2 + (x_i y - y_i x)^2 \end{aligned}$$

ist. Wir können uns daher wegen $\mathrm{MD}[\mathcal{F}] = \mathrm{MD}[\mathcal{F}^t]$ (Lemma 10) auf die transformierte Schar $\mathcal{F}^t$ beziehen. Fixieren wir ein $x_0 \in \mathbb{R}$, so hat die Gleichung $f_i^t(x_0,y) - f_j^t(x_0,y) = 0$ für alle $p_i, p_j \in S$ als quadratische Gleichung in y höchstens 2 Lösungen und (i.) ist gezeigt.

Die Funktionen r_{ij}^+ und r_{ij}^- weisen genau in den Punkten Singularitäten auf, in denen die Diskriminante der obigen quadratischen Gleichung verschwindet. Da die Diskriminante durch ein quadratisches Polynom in x gegeben ist, kann dies in höchstens $t = 2$ Punkten geschehen und (ii.) ist gezeigt.

Zum Nachweis der Schnitteigenschaft (iii.) zitieren wir aus [C59] den

Satz 18 (Bezout) *N Polynomgleichungen vom Grad $n_1, n_2, \ldots n_N$ in N Variablen haben im allgemeinen $n_1 n_2 \ldots n_N$ gemeinsame Lösungen. Gibt es mehr gemeinsame Lösungen, so sind es unendlich viele.*

Im Falle $N = 2$ besagt der Satz, daß sich zwei algebraische Kurven $f(x,y) = 0$ und $g(x,y) = 0$ vom Grad n bzw. m im allgemeinen in nm Punkten schneiden.

Die Separatoren sind wie gesehen Kegelschnitte und schneiden sich daher als algebraische Kurven vom Grad zwei im allgemeinen in $s = 4$ Punkten. Da $f_i(x_0,y_0) = f_j(x_0,y_0) = f_k(x_0,y_0)$ impliziert, daß sich die Separatoren sep_{ij} und sep_{ik} in (x_0,y_0) schneiden, ist die Behauptung gezeigt.

Wir folgern daraus

Lemma 14 *Die Komplexität des Minimierungsdiagramms der Schar der Distanzfunktionen $\mathcal{F}$ ist durch $O(n\lambda_6(n))$ beschränkt, sofern $\mathcal{F}$ allgemeine Lage einnimmt.*

Der Algorithmus von Schwartz und Sharir

Nachdem wir die Komplexität des Minimierungsdiagramms abgeschätzt haben, erläutern wir nun einen effizienten Algorithmus zur Berechnung eines MD[$\mathcal{F}$] des in Satz 17 beschriebenen Typs.

Wir setzen hierbei voraus, daß einige Operationen zu Funktionen aus $\mathcal{F}$ in konstanter Zeit ausgeführt werden können. Dazu zählt

(i.) Bestimme alle Schnittpunkte dreier verschiedener Funktionen f_i, f_j, f_k.

(ii.) Bestimme alle Unstetigkeitsstellen und Punkte vertikaler Tangente der Kurve sep_{ij}

(iii.) Bestimme bei gegebenem $x_0 \in \mathbb{R}$ die ≤ 2 Nullstellen des Terms $f_i(x_0, y) - f_j(x_0, y)$.

Wir möchten bereits an dieser Stelle bemerken, daß die Schar $\mathcal{F}$ der Abstandsfunktionen diese Bedingungen erfüllt: Die Schnittpunkte in (i.) können durch Lösen von Gleichungen 4. Grades gefunden werden, bei (ii.) und (iii.) genügen gar quadratische Gleichungen.

Der Algorithmus setzt sich aus zwei Phasen zusammen: in der ersten Phase wird eine Obermenge der Knoten des MD bestimmt, aus denen in der zweiten Phase die tatsächlichen Knoten des MD herausgesucht werden, wobei gleichzeitig eine Darstellung des MD aufgebaut wird, die effizientes Point Location unterstützt.

Zur Realisierung von **Phase 1** greifen wir auf ein Resultat des Beweises zu Satz 17 zurück: jeder Knoten des MD tritt als Knickstelle einer der Funktionen ψ_i^+, ψ_i^- oder als Schnittpunkt der Graphen von ψ_i^+ und ψ_i^- für ein $i \in I$ auf. Diese Funktionen lassen sich jedoch als Einhüllende eindimensionaler partieller Funktionen mit Hilfe der Methoden aus Kapitel 3 effizient darstellen. Für jedes $i \in I$ bestimmen wir also in Zeit $O(\lambda_{s+2}(n) \log n)$ die Knickpunkte der Funktionen ψ_i^+ und ψ_i^-, verfeinern die so gewonnene Aufteilung der x-Achse, indem wir alle singulären Punkte der Funktionen $r_{ij}^{\pm}$ hinzufügen und berechnen die Schnitte der Funktionen ψ_i^+, ψ_i^- auf jedem der so entstehenden Intervalle. Wir erhalten so n aufsteigend sortierte Listen von potentiellen Knoten des MD die wir in Zeit $O(n\lambda_{s+2}(n) \log n)$ analog zum Mischschritt in Mergesort zu einer Liste S vereinigen.

Phase 2 verfährt nach dem *Sweepline-Paradigma*.
Bei Sweepline-Algorithmen führt man ein statisches zweidimensionales Problem - in unserem Fall die Berechnung des Minimierungsdiagramms - auf ein äquivalentes dynamisches eindimensionales Problem zurück. Man überstreicht dabei vorgegebene geometrische Objekte in der Ebene mit einer vertikalen Linie, der *Sweepline*, von links nach rechts. Die zu einem Zeitpunkt von der Sweepline geschnittenen Objekte heißen *aktiv*.

Der Schnitt der Sweepline mit den Objekten ändert sich im allgemeinen kontinuierlich, an einigen Stationen der Sweepline kann jedoch eine Diskontinuität auftreten : ein inaktives Objekt wird aktiv, ein aktives Objekt inaktiv oder die Struktur des Schnittes ändert sich signifikant. Die Grundidee des Sweepline-Paradigmas ist nun, nur diese diskontinuierlichen Übergänge zu betrachten. Dazu benötigt man im wesentlichen zwei Datenstrukturen:

X-Struktur ist ein aufsteigend sortiertes Verzeichnis der Stationen rechts von der Sweepline, an denen sich der Schnitt der Sweepline mit den Objekten der Ebene diskontinuierlich ändern kann. Dieses Verzeichnis kann auch während des Sweeps erweitert werden.

Y-Struktur repräsentiert in angemessener Weise den Schnitt der Sweepline mit den betrachteten geometrischen Objekten. Angemessen heißt, daß die Y-Struktur zu jedem Zeitpunkt alle zur Fortsetzung des Sweeps notwendigen Informationen beinhaltet.

In einem Iterationsschritt des Algorithmus bewegt man die Sweepline zum nächsten Haltepunkt und streicht ihn aus der X-Struktur, führt die notwendigen Änderungen an der Y-Struktur durch und fügt gegebenenfalls neue Haltepunkte unter Berücksichtigung der aufsteigenden Anordnung in die X-Struktur ein.

In unserem Fall sind die Objekte der Ebene die Kanten des MD. Eine vom Separator sep_{ij} erzeugte Kante setzt sich im allgemeinen aus zwei Ästen zusammen: einem oberen Ast, der die Kantenpunkte enthält, die auf dem Graphen von r_{ij}^{+} liegen, sowie einem unteren Ast, dessen Punkte im Graphen von r_{ij}^{-} enthalten sind. Ein solcher Ast ist nicht notwendigerweise zusammenhängend; einer der beiden Äste kann auch leer sein. Bei gegebener Lage $x = x_0$ der Sweepline kann man auf den aktiven, d. h. von der Sweepline geschnittenen Ästen anhand der Schnittpunktsordinaten eine lineare Ordnung definieren. Wir wählen nun als Y-Struktur einen höhenbalancierten binären Suchbaum, der die jeweils aktiven oberen [unteren] Äste als Tripel $(i, j, +)$ $[(i, j, -)]$ gemäß obiger Ordnung speichert. Damit können wir die in einem Haltepunkt notwendigen Einfüge- und Streicheoperationen in logarithmischer Zeit durchführen. Ein Knoten der Y-Struktur mit der Beschriftung $(i, j, +)$ $[(i, j, -)]$ steht also für den Schnittpunkt der Sweepline $x = x_0$ mit dem Separator sep_{ij} im Punkt (x_0, r_{ij}^{+}) $[(x_0, r_{ij}^{-})]$; wir nennen ihn *Kantenschnittpunkt* des Knotens.

Schränkt man den Definitionsbereich der Funktionen aus $\mathcal{F}$ auf die aktuelle Lage der Sweepline ein, erhält man eine Schar von Querschnittfunktionen in einer Variablen, die sich nach Eigenschaft (i.) des Satzes 17 paarweise höchstens zweimal schneiden. Die Komplexität der unteren Einhüllenden der Querschnittfunktionen und damit die maximale Knotenanzahl der Y-Struktur ist nach Satz 4 durch $O(n)$ beschränkt.

Der Status der Sweepline kann sich nur dann ändern, wenn sie auf einen Knoten des MD, eine Unstetigkeitsstelle einer Kante oder einen Kantenpunkt mit vertikaler Tangente trifft. Sei nun T die Menge der $O(n^2)$ Unstetigkeitsstellen und Punkte vertikaler

Tangente der Separatoren sep_{ij}. Wir berechnen und sortieren die Menge T nach aufsteigender Abszisse in Zeit $O(n^2 \log n)$ und vereinigen sie mit S zur X-Struktur. Zu jedem Punkt der X-Struktur speichern wir zusätzlich die Indizes der sich in diesem Punkt schneidenden Funktionen.

Zur Initialisierung der Y-Struktur wählen wir ein beliebiges $x_0 \in \mathbb{R}$, das kleiner als alle Werte aus $S \cup T$ ist. Die Berechnung der unteren Einhüllenden der Querschnittfunktionen $f_i(x_0, y)$ stellt ein eindimensionales Problem dar und kann somit mit den Methoden aus Kapitel 3 in Zeit $O(n \log n)$ geschehen. Man liest nun alle Knickstellen ab und fügt sie entsprechend obiger Konvention in einen Suchbaum ein.

In einer Iteration des Algorithmus sei $p_0 = (x_0, y_0)$ der nächste Haltepunkt. Wir unterscheiden zwei Fälle.

Fall 1 $p_0 \in S$

Seien i, j, k die dem Punkt p_0 zugeordneten Indizes. Damit p_0 als Knoten im MD auftritt, muß die Y-Struktur mindestens einen mit zwei dieser drei Indizes (z. B. i, j) beschrifteten Knoten enthalten, dessen Kantenschnittpunkt p_0 ist. Anschaulich bedeutet dies, daß i und j in Minimierungssequenzen der Querschnittfunktionen zu x-Werten, die nur wenig kleiner als x_0 sind, benachbart auftreten und die Sweepline sep_{ij} in Punkten schneidet, die für $x \to x_0$ gegen p_0 streben. Ist dies nicht der Fall, verwirft der Algorithmus p_0 und die Y-Struktur bleibt unverändert.

Ansonsten können zwei Fälle auftreten:

Fall 1.1 Es existieren sogar zwei (notwendigerweise benachbarte) Baumknoten, die obige Bedingung erfüllen.

Seien sie o.B.d.A mit (i, j) und (j, k) beschriftet. Dann erscheinen die Indizes i, j, k in dieser Reihenfolge in den oben bezeichneten Minimierungssequenzen. Bewegt sich die Sweepline über x_0 hinweg, verschwindet j aus der MS. Wir aktualisieren die Y-Struktur, indem wir die beiden Knoten entfernen und den Knoten (i, k) mit der korrekten Markierung ($+$ oder $-$) hinzufügen.

Fall 1.2 Existiert nur ein solcher Knoten (i, j), wird der Index k in MS für x wenig größer als x_0 zwischen i und j erscheinen. Wir aktualisieren die Y-Struktur, indem wir den Knoten entfernen und die Knoten (i, k), (k, j) mit korrekter Markierung hinzufügen.
In beiden Fällen (1.1) und (1.2) benötigt man Zeit $O(\log n)$.

Fall 2 $p_0 \in T$

Wir betrachten nur den Fall, daß p_0 Kurvenpunkt mit vertikaler Tangente ist. Seien i, j die dem Punkt p_0 zugeordneten Indizes. Damit p_0 als Punkt der Kurve sep_{ij} im MD auftritt, müssen mindestens zwei benachbarte Baumknoten in der Y-Struktur existieren, die beide mit i (analog j) beschriftet sind. Dies bedeutet, daß i (bzw. j) in MS zu x-Werten auftritt, die wenig kleiner als x_0 sind. Ist dies nicht der Fall verwirft der Algorithmus p_0 und die Y-Struktur bleibt unverändert.

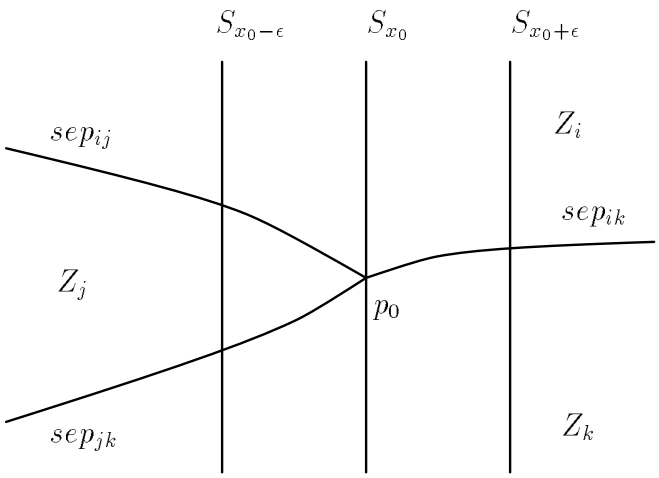

Abbildung 5.9: Fall 1.1 des Sweepline-Algorithmus

Fall 2.1 Sind diese beiden Knoten $(i, j, +)$ und $(i, j, -)$, so schneidet die Sweepline sep_{ij} für $x = x_0 - \epsilon$ (ϵ hinreichend klein) zweimal, ist Tangente an sep_{ij} in x_0 und schneidet sep_{ij} nicht mehr für $x = x_0 + \epsilon$ Wir streichen also die Knoten $(i, j, \pm)$ aus der Y-Struktur.

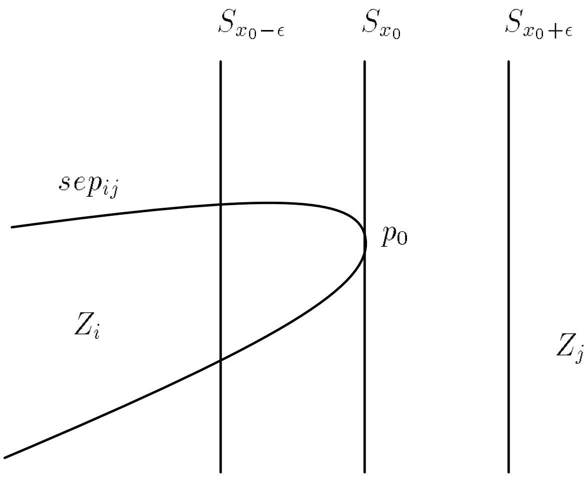

Abbildung 5.10: Fall 2.1 des Sweepline-Algorithmus

Fall 2.2 Andernfalls seien $(k, i), (i, l)$ $(k, l \in I)$ solche Baumknoten. Liegt y_0 zwischen den Kantenschnittpunkten der Baumknoten $(k, i), (i, l)$, so liegt der umgekehrte Fall vor : die Sweepline ist Tangente an sep_{ij} in $x = x_0$ und schneidet sep_{ij} für $x = x_0 + \epsilon$. Wir fügen die Knoten $(i, j, -), (i, j, +)$ in die Y-Struktur ein.
Auch in Fall 2 genügt logarithmische Zeit.

Da die Sweepline höchstens $O(n\lambda_{s+2}(n))$ Haltepunkte überstreicht, benötigt Phase 2 des Algorithmus ebenso wie Phase 1 Zeit $O(n\lambda_{s+2}(n)\log n)$. Der Algorithmus kann nun während des Sweeps eine Datenstruktur ähnlich der des Abschnitts 5.1.4 aufbauen, die Point Location im MD in logarithmischer Zeit unterstützt. Dazu fügt er die Haltepunkte der Sweepline, an denen die Y-Struktur modifiziert wurde (dies sind die Abszissen der Knoten und Punkte vertikaler Tangente des MD) in einen balancierten Suchbaum ein und ordnet jedem die zu diesem Zeitpunkt aktuelle Y-Struktur zu. Diese Datenstruktur benötigt $O(n^2\lambda_{s+2}(n))$ Speicherplatz; man kann sie jedoch beispielsweise mit den in [C84b] beschriebenen Techniken in eine Datenstruktur der Größe $O(n\lambda_{s+2}(n))$ (d.h. linear in der Größe des Graphen) transformieren, ohne Laufzeitverluste bei Point Location hinnehmen zu müssen.

5.2.4 Lage der Extrema der unteren Einhüllenden

Extrema der Distanzfunktion

Die geometrische Anschauung läßt vermuten, daß die Distanzfunktion

$$f_i(x,y) = \frac{1}{x^2+y^2+1}[(y_i - z_i y)^2 + (z_i x - x_i)^2 + (x_i y - y_i x)^2)]$$

zu einem Punkt p_i nur in ihrer Nullstelle, dies ist der Schnittpunkt der Geraden $0p_i$ mit E_π, ein lokales Minimum aufweist. Maximal wird der Abstand genau dann, wenn die Ursprungsgerade in der Ebene E^{p_i} mit Normalenvektor p_i enthalten ist. Die Konfigurationsraumkoordinaten solcher Ursprungsgeraden erfüllen die Schnittgleichung der Ebenen E^{p_i} und E_π, die Geradengleichung $x_i x + y_i y + z_i = 0$.

Wir bestätigen diese Vermutung auf analytischem Weg. Um die unübersichtliche Berechnung des Gradienten zu vermeiden, wenden wir das Extremallemma aus Abschnitt 5.1.5 an. Dazu gehen wir von der auf der Einheitskugel definierten Abstandsfunktion aus und projizieren diese auf die polare Tangentialebene in einem bezüglich p_i geeignet ausgerichteten Koordinatensystem. Zum Aufspüren aller Extrema genügen im vorliegenden Fall die beiden Projektionsrichtungen

Fall 1 $p_i' = (0,0,z_i')$ liegt auf der z-Achse

Dann ist

$$f_i(x,y) = z_i'^2 \frac{x^2+y^2}{x^2+y^2+1} \qquad \nabla f_i(x,y) = \frac{2z_i'^2}{(x^2+y^2+1)^2}\begin{pmatrix} x \\ y \end{pmatrix}$$

mit $(0,0)$ als einzigem stationären Punkt, einem Minimum. Dies ist der Durchstoßpunkt der Geraden $0p_i$ mit der Ebene $E_\pi : x_3 = 1$.

Fall 2 $p_i' = (0,y_i',0)$ liegt auf der y-Achse

Dann ist

$$f_i(x,y) = y_i'^2 \frac{x^2+1}{x^2+y^2+1} \qquad \nabla f_i(x,y) = \frac{2y_i'^2}{(x^2+y^2+1)^2} \begin{pmatrix} xy^2 \\ -y(x^2+1) \end{pmatrix}$$

Alle Punkte der Geraden $y = 0$ sind lokale Maxima der Funktion. Dies ist die Schnittgerade der Ebenen E^{p_i} und E_π.

Da die Charakterisierung der Extrema durch geometrische Objekte unabhängig von der Lage des Koordinatensystems erfolgt ist, kann ihre Lage in jedem KS effizient berechnet werden.

Extrema auf den Separatoren

Eine überzeugende geometrische Begründung der Lage der Extrema auf den Separatoren erweist sich als schwierig; hier tritt die Stärke der algebraischen Methode zutage. Vor diesem Hintergrund kann man die Verwendung der algebraischen Methoden auch bei einfachen Konstellationen als Vorübung für diesen schwierigeren Fall ansehen. Dennoch wollen wir die geometrische Anschauung des Lesers soweit führen, daß ihn die gefundenen Ergebnisse nicht überraschen. Der $kegel_{ij}$ zweier Weltpunkte p_i, p_j wird durch die Ebene $E_{p_ip_j}$ in symmetrische Hälften geteilt. Weiterhin kann für $|p_i| < |p_j|$ die Ebene E^{p_i}, nicht aber E^{p_j} Schnittgeraden mit $kegel_{ij}$ aufweisen. Existieren diese Schnittgeraden, haben sie unter allen Geraden des $kegel_{ij}$ maximalen Abstand zu p_i und p_j. Wir erwarten dann minimalen Abstand für die beiden Geraden $E_{p_ip_j} \cap kegel_{ij}$. Schneidet E^{p_i} den $kegel_{ij}$ nicht, erwarten wir, daß eine der Geraden $E_{p_ip_j} \cap kegel_{ij}$ maximalen, die andere minimalen Abstand zu p_i und p_j unter allen Geraden des $kegel_{ij}$ aufweist.

Daß diese Vermutung über die Lage der Extrema zutrifft und keine weiteren existieren, zeigt das

Lemma 15 *Gegeben seien zwei linear unabhängige Punkte p_i und p_j.*

Die Extrema ihrer Abstandsfunktionen f_i und f_j auf dem Separator von p_i und p_j (sep_{ij}) liegen

- *in den Schnittpunkten der Ursprungsebene $E_{p_ip_j}$ mit sep_{ij}*

sowie

- *im Fall $|p_i| < |p_j|$ in den Schnittpunkten der Ursprungsebene E^{p_i} mit sep_{ij}, sofern ein solcher Schnittpunkt existiert.*
- *im Fall $|p_i| = |p_j|$ im Kreuzungspunkt von sep_{ij}, d. h. dem Schnittpunkt der Ebenen $E^{p_i - p_j}$, $E^{p_i + p_j}$ und E_π.*

Beweis:

FALL 1 $|p_i| < |p_j|$

Wir richten das Koordinatensystem so aus, daß p_j auf die z-Achse und p_i in die xz-Ebene fällt. Division aller Koordinaten durch $|p_j| \neq 0$ ändert als geometrische Ähnlichkeitstransformation an der Extremaleigenschaft der Funktionen nichts. Die Koordinaten der so transformierten Punkte seien $p'_j = (0, 0, 1)$ und $p'_i = (x'_i, 0, z'_i)$. Aus der linearen Unabhängigkeit folgt $x'_i \neq 0$, aus $|p_i| < |p_j|$ folgt $x'^2_i + z'^2_i < 1$ und damit $z'^2_i < 1$.

Unser Ziel ist nun, sämtliche Lösungen der Optimierungsaufgabe

$$\begin{array}{rl} \text{maximiere} & f_i(x,y) = \dfrac{x^2+y^2}{x^2+y^2+1} \\[2ex] \text{u.d.N.} & (x,y) \in sep_{ij} \\ \iff & g(x,y) := (f_j(x,y) - f_i(x,y))(x^2+y^2+1) = 0 \end{array}$$

zu charakterisieren. Es gilt

$$\begin{aligned} g(x,y) &= (z'^2_i - 1)x^2 + (x'^2_i + z'^2_i - 1)y^2 - 2x'_i z'_i x + x'^2_i \\ &= kx^2 + ly^2 + mx + n \end{aligned}$$

mit den Ersetzungen

$$\begin{array}{rclrcl} k &=& z'^2_i - 1 & l &=& x'^2_i + z'^2_i - 1 \\ m &=& -2x'_i z'_i & n &=& x'^2_i \end{array}$$

Wir betrachten die zugehörige Lagrangefunktion

$$L(x,y,\lambda) = f_i(x,y) + \lambda g(x,y)$$

In jedem Extrempunkt verschwindet der Gradient der Lagrangefunktion.

$$\begin{array}{rclcll} \dfrac{\partial L}{\partial x} &=& \dfrac{2x}{(x^2+y^2+1)^2} + \lambda[2kx+m] &=& 0 & (1) \\[2ex] \dfrac{\partial L}{\partial y} &=& \dfrac{2y}{(x^2+y^2+1)^2} + 2\lambda ly &=& 0 & (2) \\[2ex] \dfrac{\partial L}{\partial \lambda} &=& kx^2 + ly^2 + mx + n &=& 0 & (3) \end{array}$$

Aus (2) folgt

$$y = 0 \ \vee \ l\lambda = \frac{-1}{(x^2+y^2+1)^2}$$

FALL 1.1 $y = 0$

Nach Einsetzen in (3) erhalten wir

$$(z_i'^2 - 1)x^2 - 2x_i'z_i'x + x_i'^2 = 0$$

mit den Lösungen

$$x_{1,2} = \frac{x_i'}{z_i' \pm 1}$$

Die potentiellen Extrema $(\frac{x_i'}{z_i' \pm 1}, 0)$ sind die Schnittpunkte von $E_{p_i p_j}$ mit sep_{ij}.

FALL 1.2 $l\lambda = \dfrac{-1}{(x^2 + y^2 + 1)^2}$

Nach Voraussetzung gilt $x_i'^2 + z_i'^2 \neq 1$ gilt $l \neq 0$ Einsetzen in (1) ergibt

$$\frac{2x}{(x^2 + y^2 + 1)^2} - \frac{(2kx + m)}{l(x^2 + y^2 + 1)^2} = 0$$

mit der Lösung

$$x = \frac{m}{2(l - k)} = -\frac{z_i'}{x_i'}$$

Für die y-Koordinate gilt nach (3)

$$y^2 = \frac{z_i'^2 - (x_i'^2 + z_i'^2)^2}{x_i'^2(x_i'^2 + z_i'^2 - 1)}$$

und man erhält Schnittpunkte, wenn $z_i'^2 \leq (x_i'^2 + z_i'^2)^2$. Es sind dies die Schnittpunkte von $E^{p_i} \cap E_\pi : x_i'x + z_i' = 0$ mit sep_{ij}.

Eine kleine Rechnung zeigt, daß der Separator bei dieser Lage des KS stets die Gestalt einer Ellipse annimmt, so daß diese Betrachtung bereits alle möglichen Lagen der Geraden im Äquidistanzkegel auf Extremalität untersucht.

FALL 2 $|p_i| = |p_j|$

Die Fläche der Äquidistanzgeraden zerfällt in diesem Fall wie in Lemma 13 gezeigt in zwei Ebenen. Um alle möglichen Lagen der Ursprungsgeraden auf Extremalität zu testen, müssen wir zwei Ausrichtungen des KS untersuchen.

FALL 2.1 $p_i' = (0, 0, 1), p_j' = (x_i', 0, z_i')$

Wir suchen die Extrema der Funktion

$$f_i(x, y) = \frac{x^2 + y^2}{x^2 + y^2 + 1}$$

auf dem Separator

$$sep_{ij} : (z_i'^2 - 1)x^2 - 2x_i'z_i'x + x_i'^2 = 0$$

der in die parallelen Geraden

$$g_{1,2} : (z'_i \pm 1)x - x'_i = 0$$

zerfällt. Wir erhalten mit den Abkürzungen aus Fall 1 die Gleichungssysteme

$$\begin{aligned}
\frac{2x}{(x^2+y^2+1)^2} + \lambda(2kx+m) &= 0 \\
\frac{2y}{(x^2+y^2+1)^2} &= 0 \\
(z'_i \pm 1)x - x'_i &= 0
\end{aligned}$$

mit den Lösungen $x_{1,2} = \frac{x'_i}{z'_i \pm 1}$, $y = 0$, den Schnitten von $E_{p_i p_j}$ mit sep_{ij}

FALL 2.2 $p'_i = (0,1,0), p'_j = (x'_i, y'_i, 0)$

Wir suchen die Extrema der Funktion

$$f_i(x,y) = \frac{x^2+1}{x^2+y^2+1}$$

auf dem Separator

$$sep_{ij} : (1 - y_i'^2)x^2 - x_i'^2 y^2 + 2x'_i y'_i xy = 0$$

dem Geradenpaar

$$g_{1,2} : (-y'_i \pm 1)x + x'_i y = 0$$

Wir erhalten die Gleichungssysteme

$$\begin{aligned}
\frac{2xy^2}{(x^2+y^2+1)^2} + \lambda(-y'_i \pm 1) &= 0 \\
\frac{-2y(x^2+1)}{(x^2+y^2+1)^2} + \lambda x'_i &= 0 \\
(-y'_i \pm 1)x + x'_i y &= 0
\end{aligned}$$

mit der einzigen Lösung $(0,0)$, dem Kreuzungspunkt des Separators sep_{ij}, d.h. dem Schnittpunkt der Ebenen $E^{p_i - p_j}, E^{p_i + p_j}$ mit der Ebene E_π. ∎

5.2.5 Der MaxMin-Algorithmus

Unter Verwendung der bisher erzielten Ergebnisse sind wir nun in der Lage, einen effizienten Algorithmus zur Lösung des MaxMin-Problems der Funktionenschar $\mathcal{F}$ zu konstruieren. Sein Aufbau folgt der bewährten Grundstruktur: Berechnung des MD, Enumerieren der kritischen Punkte, einer Obermenge der lokalen Maxima und Auswählen des globalen Optimums.

Zunächst ermitteln wir, wieviele kritische Punkte zu berücksichtigen sind.

1. Jeder der $\binom{n}{2}$ Separatoren trägt nach Lemma 15 höchstens 4 kritische Punkte bei; insgesamt kann es also $O(n^2)$ kritische Punkte dieses Typs geben.

2. Jede der Abstandsfunktionen nimmt ihr Maximum auf einer Geraden an. Es genügt aber, jeweils einen beliebigen Referenzpunkt dieser Maximalgeraden als kritischen Punkt auszuwählen. Ist nämlich die Abstandsfunktion auf einem Abschnitt ihrer Maximalgeraden aktiv, der den Referenzpunkt nicht enthält, ist dieser Abschnitt mindestens auf einer Seite von einer Kante des MD begrenzt und sein Endpunkt wird unter Punkt (1.) berücksichtigt.

3. Die untere Einhüllende kann ihr Maximum darüber hinaus auch in einem der $O(n\lambda_6(n))$ Knoten des MD annehmen.

Wir finden also $O(n^2 + n + n\lambda_6(n)) = O(n\lambda_6(n))$ potentielle Maxima, aus denen wir mit Hilfe des MD das globale Maximum heraussuchen.

Unser Algorithmus wird jedoch nicht wie bisher das gesamte MD vorberechnen. Vielmehr bestimmen wir zunächst die $O(n^2)$ kritischen Punkte aus den Punkten (1.) und (2.) der Aufzählung und fügen sie in Zeit $O(n^2 \log n)$ nach aufsteigender x-Koordinate sortiert in eine Liste V ein. Während der Sweepline-Algorithmus das MD bestimmt, können wir für alle Punkte $p \in V$, die in den Streifen zwischen zwei Haltepunkten der Sweepline fallen, mittels Point Location den Index der in p aktiven Funktion und damit den Wert der unteren Einhüllenden f^u an der Stelle p bestimmen. Da jeder Knoten k des MD mit den Indizes der sich dort auf dem Graphen von f^u schneidenden Funktionen markiert ist, kann der Wert $f^u(k)$ in konstanter Zeit berechnet werden. Mit dieser Information können wir in Zeit $O(n\lambda_6(n))$ aus den Kandidatenpunkten denjenigen auswählen, der den Funktionswert der unteren Einhüllenden maximiert.

Aufgrund dieser Vorgehensweise muß der Algorithmus lediglich die jeweils aktuelle Y-Struktur kennen und braucht die vorangegangenen Instanzen nicht zu speichern. Da mit Hilfe der benutzten Datenstruktur Point Location in Zeit $O(\log n)$ möglich ist, benötigt der Algorithmus insgesamt Zeit $O(n^2 \log n + n\lambda_6(n) \log n) = O(n\lambda_6(n) \log n)$.

Bislang haben wir die Geraden in Grenzlage, d. h. in der Äquatorebene $E_{Equ} : x_3 = 0$ der Einheitskugel, ausgeklammert. Zudem war die Behandlung des MaxMin-Problems nicht vollständig, da das Maximum der unteren Einhüllenden über der Ebene E_π nicht notwendigerweise definiert sein muß und zur Bestimmung des Supremums Grenzwertbetrachtungen erforderlich werden können. Aufgrund der Stetigkeit des Abstandsmaßes nehmen die Funktionen im Limes jedoch gerade die Funktionswerte auf dem Äquator an, so daß bei Berücksichtigung der optimalen Äquatorgeraden die Aufgabe korrekt gelöst wird.

Die optimale Äquatorgerade ist bestimmt durch den Lösungspunkt der eindimensionalen MaxMin-Aufgabe der über $[-1, 1]$ erklärten Funktionenschar $\mathcal{G} = \{g_1, \ldots, g_n\}$ mit

$$g_i(x) = (\sqrt{1 - x^2}x_i - xy_i)^2 + z_i^2$$

Sie beschreibt den quadrierten Abstand des Punktes $p_i = (x_i, y_i, z_i)$ von der Geraden g_x, welche die Einheitskugel im Äquatorpunkt $(x, \sqrt{1-x^2}, 0)$ durchstößt. Diese Aufgabe entspricht dem in Kapitel 3 besprochenen Problem der Maximin-Ursprungsgeraden unter Laguerre-Distanz und kann wie dort gezeigt in Zeit $O(n \log n)$ gelöst werden.

Zusammenfassend erhalten wir den

Satz 19 *Liegen keine 4 Weltpunkte auf einem Zylinder, dessen Achse durch den Nullpunkt verläuft, kann (P17) in Zeit $O(n\lambda_6(n) \log n)$ gelöst werden.*

5.2.6 Duale Problemstellung

Motivation

Ein zentraler Teilbereich der Robotik ist mit der Planung kollisionsfreier Bewegungen von Robotern in Hindernisumgebungen befaßt. Eine wesentliche Teilaufgabe hierbei ist, bei gegebener Position des Roboters und der Hindernisse zu entscheiden, ob eine Kollision der Roboterteile untereinander oder mit den Objekten der Umgebung auftritt. Solche Kollisionstests können sehr rechenintensiv sein, wenn die Objekte und Roboter komplexe und unregelmäßige Gestalt aufweisen. Häufig begegnet man diesem Problem, indem man einfache geometrische Körper wie Kugeln oder Quader bestimmt, die die einzelnen Roboterteile umfassen und die (einfacheren) Kollisionstests für diese Objekte durchführt. Verläuft der Kollisionstest für die Hüllkörper negativ, kann man schließen, daß auch die umhüllten Objekte nicht kollidieren. Verläuft der Test jedoch positiv, kann man keine Aussage treffen und muß den komplexeren Kollisionstest durchführen. Um das Auftreten dieses Falles so weit wie möglich zu vermeiden, wählt man als Hüllkörper stets den kleinstmöglichen, der das vorgegebene Objekt enthält. Für längliche Objekte, wie beispielsweise Roboterarme, erscheinen Zylinder als geeignete Hüllkörper.

Problemstellung

Sind die Koordinaten der Objektecken S bekannt, ist das Problem, den die Punktmenge umhüllenden Zylinder mit kleinstem Querschnitt zu bestimmen, dessen Achse durch einen vorgegebenen Punkt (etwa den Schwerpunkt der Menge) verläuft, äquivalent zu dem

PROBLEM 18 : [Minimax-Ursprungsgerade im Raum]

Gegeben: Eine Menge $S = \{p_1, \ldots, p_n\}$, $p_i = (x_i, y_i, z_i)$ von n Punkten des $\mathbb{R}^3$.

Gesucht: Eine Ursprungsgerade $g_u^* : x = \alpha u^*$ mit Richtungsvektor u^*, die den maximalen euklidischen Abstand zur Punktmenge S

$$\max_{1 \leq i \leq n} \delta(p_i, g_u)$$

minimiert.

Ähnlich wie beim (P16) in Abschnitt 5.1 haben wir auch hier bereits in der Diskussion des Maximin-Problems die wesentlichen Vorarbeiten zur Lösung des dualen Minimax-Problems geleistet. Der Algorithmus von Schwartz/Sharir kann gleichermaßen Minimierungs- wie (in analoger Weise definierte) Maximierungsdiagramme berechnen, da sich die entscheidenden Schnitteigenschaften der Abstandsfunktionen beim Übergang $f \to -f$ nicht verändern. Weiterhin haben wir die Lage der kritischen Punkte der oberen Einhüllenden bei der Diskussion der lokalen Extrema der Funktionen und auf den Separatoren ermittelt; auch sie erfüllen die Lagekriterien des Lemma 15. Schließlich kann auch die optimale Gerade der Äquatorebene ähnlich wie die Minimax-Ursprungsgerade in der Ebene in Zeit $O(n \log n)$ gefunden werden. Wir erhalten also in Analogie zum dualen Problem einen Algorithmus mit Laufzeit $O(n\lambda_6(n) \log n)$.

5.2.7 Varianten der Problemstellungen

Um der in Abschnitt 5.2.1 angeführten Beispielanwendung besser entsprechen zu können, betrachten wir nun einige Varianten des im vorletzten Abschnitt diskutierten Problems. Anhand dieser variierten Probleme werden wir die Allgemeinheit des verfolgten Ansatzes demonstrieren, wobei wir uns auf neueste Resultate zur Komplexität von Minimierungsdiagrammen stützen können.

Die ebenen Instanzen der hier angesprochenen Probleme haben wir im Zusammenhang mit der Maximin-Ursprungsgerade in der Ebene in Kapitel 3 diskutiert.

Maximin-Ursprungsgerade bei additiv gewichteter Abstandsfunktion

Betrachten wir additiv gewichtete Abstandsfunktionen im Sinne des *Laguerre-Abstands*. Der Menge der Weltpunkte werden reelle Gewichte $R = \{r_1, \ldots, r_n\}$ zugeordnet; diese induzieren Abstandsfunktionen der Form

$$\mathrm{dist}_L(g, p_i) = \delta^2(g, p_i) - r_i^2.$$

Das MaxMin-Problem der Laguerre-Abstandsfunktionen kann auf die gleiche Weise wie das der ungewichteten Funktionen gelöst werden. Die Separatoren haben noch immer die Gestalt von Kegelschnitten; sie können nun aber für manche Funktionenpaare leer sein. Der Algorithmus von Schwartz/Sharir kann also auch in diesem gewichteten Fall zur Berechnung des MD dienen, sofern die Abstandsfunktionen allgemeine Lage einnehmen. Lemma 15, das die Lage der kritischen Punkte auf den Separatoren beschreibt, bleibt gleichermaßen gültig. Das Problem der optimalen Äquatorgeraden ist mit Hilfe des Divide-and-Conquer Algorithmus aus 3.3.4 in Zeit $O(n \log n)$ zu lösen. Insgesamt ergibt sich die gleiche asymptotische Laufzeit wie im ungewichteten Fall: $O(n\lambda_6(n) \log n)$.

Maximin-Ursprungsstrahl im Raum

Ersetzt man die Ursprungsgerade durch einen *Ursprungsstrahl*, können die Tangentialebenen in Nord- und Südpol der Einheitskugel zusammen mit deren Äquator als Konfigurationsraum dienen. Dabei wird die Lage eines Ursprungsstrahls durch die Koordinaten seines Schnittpunktes mit einem der obengenannten Objekte repräsentiert.

Der Abstand eines Punktes p_i vom Ursprungsstrahl s_u mit Richtungsvektor u ist von seiner Lage bezüglich der Ebene E^u abhängig: liegt p_i in dem Halbraum, in den u zeigt, so ist sein Abstand zu s_u durch die Länge seines Lotes auf die Gerade g_u mit Richtungsvektor u gegeben; liegt p_i in dem von s_u abgewandten Halbraum, wählen wir seine Entfernung zum Ursprung als Abstandsmaß.

Diese Abstandsdefinition induziert eine Schar $\mathcal{G}_I = \{g_i \mid i \in I\}$ von (quadrierten) Abstandsfunktionen in den Koordinaten x und y der nördlichen Tangentialebene E_π : $x_3 = 1$

$$g_i(x,y) = \begin{cases} \delta^2(p_i, g_{(x,y,1)}) & \text{falls} \quad <(x,y,1), p_i> > 0 \\ <p_i, p_i> & \text{sonst} \end{cases}$$

Die bisher beschriebenen Algorithmen zur Konstruktion von Minimierungsdiagrammen versagen bei Funktionen dieses Typs. Hier kommt uns jedoch ein weitreichendes Resultat zu Hilfe [Sh93].

Satz 20 (Sharir) *Sei $\mathcal{F}_I = \{f_i | i \in I\}$ eine Schar partieller reeller Polynomfunktionen in zwei Variablen mit konstantem maximalen Grad b, die folgende Bedingungen erfüllen:*

(i.) Für alle $i \in I$ ist der Definitionsbereich von f_i eine zusammenhängende Region, die durch algebraische Kurvenbögen mit maximalem Grad b begrenzt wird.

(ii.) Die Funktionen befinden sich in allgemeiner Lage.

(iii.) Die Graphen dreier Funktionen f_i, f_j, f_k schneiden sich in höchstens s Punkten. (Aus dem Satz von Bezout und (ii.) folgt $s \leq b^3$)

Dann hat das Minimierungsdiagramm von $\mathcal{F}$ Komplexität $O(n^{2+\epsilon})$ für jedes $\epsilon > 0$, wobei der Proportionalitätsfaktor von ϵ, s und b abhängt. Weiterhin existiert in einem entsprechenden algebraischen Berechnungsmodell ein Algorithmus, der MD[$\mathcal{F}$] in randomisierter erwarteter Zeit $O(n^{2+\epsilon})$ für jedes $\epsilon > 0$ berechnet.

Das algebraische Berechnungsmodell muß die üblichen Elementaroperationen wie Berechnung der Schnittpunkte dreier Funktionen etc. in konstanter Zeit zur Verfügung stellen.

Der Algorithmus stellt das MD als planaren Graphen mittels seiner Knoten und Kanten dar; es ist dabei zu beachten, daß die Kanten des MD nicht nur als Projektion einer Schnittlinie zweier Funktionen sondern auch als Projektion einer Randlinie des Graphen einer Funktion auftreten können. Entsprechend repräsentieren Knoten des MD nicht notwendigerweise Schnittpunkte dreier Funktionen.

Aus der vom Algorithmus erzeugten Darstellung des MD kann in Zeit $O(n^{2+\epsilon})$ eine Datenstruktur aufgebaut werden, die Anfragen nach den in einem Punkt der Ebene aktiven Funktionen in Zeit $O(\log n)$ unterstützt ([C84b]).

Um Satz 20 anwenden zu können, transformieren wir $\mathcal{G}$ zunächst in die ordnungsäquivalente Funktionenschar $\mathcal{G}^t = \{g_i^t \mid i \in I\}$, wobei

$$\begin{aligned} g_i^t(x,y) &= (x^2+y^2+1)\, g_i^2(x,y) \\ &= \begin{cases} (y_i - z_i y)^2 + (x_i - z_i x)^2 + (y_i x - x_i y)^2 & \text{falls} \quad <(x,y,1),p_i> \; > 0 \\ (x^2+y^2+1)<p_i,p_i> & \text{sonst} \end{cases} \end{aligned}$$

Jedes g_i^t zerfällt an der Trenngeraden $x_i x + y_i y + z_i = 0$ in zwei partielle Polynomfunktionen. Verlangt man, daß keine 4 Punkte gleichen Abstand von einem Ursprungsstrahl haben und $|p_i| \neq |p_j|$ für alle $i \neq j \in I$ gilt, erfüllen diese $2n$ Polynomfunktionen die Voraussetzungen des Satzes 20 und ihr Minimierungsdiagramm, eine Verfeinerung des MD$[\mathcal{G}^t]$ (= MD$[\mathcal{G}]$), kann in randomisierter erwarteter Zeit $O(n^{2+\epsilon})$ berechnet werden.

An welchen Stellen kann $g^u[\mathcal{G}]$ extremal werden? Der Graph von $g_i, i \in I$ setzt sich aus einer konstanten und einer variablen Teilfläche zusammen, die sich über der Trenngeraden, die der Maximalgeraden der Funktion $\delta^2(p_i, g_{(x,y,1)})$ entspricht, berühren. Im variablen Teil weisen die Funktionen ein lokales Minimum in ihrer Nullstelle auf, sofern sie eine solche besitzen; sie werden auf der Trenngeraden zum konstanten Teil maximal. Wir wählen einen beliebigen Punkt der Maximalgerade als kritischen Punkt. Die Extrema auf den Separatoren zweier variabler Teilfunktionen erfüllen offensichtlich die Lagebedingungen des Lemma 15. Sie sind wie die Knoten des MD kritische Punkte. Weiterhin fügen wir aus jedem Separator zwischen einer konstanten und einer variablen Teilfunktionen sowie aus dem Definitionsbereich der konstanten Teilfunktionen je einen beliebigen Referenzpunkt in die Menge der kritischen Punkte ein.

Für jeden der $O(n^{2+\epsilon})$ kritischen Punkte der nördlichen Tangentialebene bestimmt man durch Anfrage an das berechnete MD eine in diesem Punkt aktive Funktion. Dies kostet Zeit $O(\log n)$ pro Punkt. Mit dieser Information kann man den kritischen Punkten, der g^u maximiert, durch einfache Maximumsuche finden. Da $O(\log n)$ für alle $\epsilon > 0$ asymptotisch langsamer wächst als $O(n^\epsilon)$ benötigt diese Phase insgesamt Zeit $O(n^{2+\epsilon})$. Analog verfährt man mit der südlichen Tangentialebene.

Schneidet ein Ursprungsstrahl den Äquator in $(x, \pm\sqrt{1-x^2}, 0)$, läßt sich das Quadrat seines Abstandes zum Punkt p_i als Funktion von $x \in [-1,1]$ in der Form

$$h_i^{\pm}(x) = \begin{cases} (\pm\sqrt{1-x^2}\,x_i - y_i x)^2 + z_i^2 & \text{falls} \quad x_i x + y_i y > 0 \\ x_i^2 + y_i^2 + z_i^2 & \text{sonst} \end{cases}$$

schreiben. Der optimale Strahl der Äquatorebene kann nun durch Lösen der MaxMin-Aufgaben der Funktionsscharen $\mathcal{H}^+ = \{h_i^+ \mid i \in I\}$ und $\mathcal{H}^- = \{h_i^- \mid i \in I\}$ gefunden werden. Wir verweisen auf die Argumentation auf Seite 42 und schließen, daß dies in Zeit $O(\lambda_3(n) \log n)$ geschehen kann.

Zusammengefaßt folgt, daß der optimale Ursprungsstrahl in randomisierter erwarteter Zeit $O(n^{2+\epsilon})$ gefunden werden kann.

5.3 Verankerte Ebene im Raum

5.3.1 Problemstellung

Im letzten Abschnitt dieses Kapitels wenden wir uns der Aufgabe zu, eine Ursprungsebene in Bezug auf eine Menge von (gewichteten) Weltpunkten optimal nach dem Maximin-Kritrium zu plazieren. Es ist dies der Unterfall $k = 2$ der in der Kapiteleinleitung dargelegten Fragestellung.

Wir studieren also das

PROBLEM 19 : [Maximin-Ursprungsebene im Raum]

Gegeben: Eine Menge $S = \{p_1, \dots, p_n\}$, $p_i = (x_i, y_i, z_i)$ von n Punkten des $\mathbb{R}^3$ mit positiven Gewichtungsfaktoren $W = \{w_1, \dots, w_n\}$.

Gesucht: Eine Ursprungsebene E^{v^*} mit Normalenvektor v^*, die den minimalen euklidischen Abstand zur Punktmenge S

$$\min_{1 \le i \le n} \delta(p_i, E^v)$$

maximiert.

Mit ähnlicher Begründung wie in 5.2.2 können wir o.B.d.A annehmen, daß es sich um ein ungewichtetes Problem handelt und alle Punkte paarweise linear unabhängig sind.

Funktionale Fassung

Der Abstand eines Punktes p_i von der Ebene E^v ist gegeben durch

$$\delta(p_i, E^v) = \frac{1}{|v|} |{<}p_i, v{>}| = \sqrt{\frac{(x_i v_1 + y_i v_2 + z_i v_3)^2}{v_1^2 + v_2^2 + v_3^2}}$$

Die Lage einer Ursprungsebene E^v ist eindeutig durch einen Schnittpunkt ihrer Normalengeraden g_v mit der Einheitskugel festgelegt. Es erweist sich gleichwohl als vorteilhaft, wiederum die Tangentialebene im Nordpol der Einheitskugel zusammen mit deren Äquator als Konfigurationsraum zu wählen.

Betrachten wir zunächst die nördliche Tangentialebene $E_\pi : x_3 = 1$. Schneidet die Normalengerade g_v der Ursprungsebene E^v die Ebene E_π im Punkt $(x, y, 1)$, beschreibt die Funktion

$$f_i(x, y) := \delta^2(p_i, E^{(x,y,1)}) = \frac{(x_i x + y_i y + z_i)^2}{x^2 + y^2 + 1}$$

das Quadrat des Abstandes von E^v und p_i. Abbildung 5.11 zeigt den Graph einer solchen Funktion.

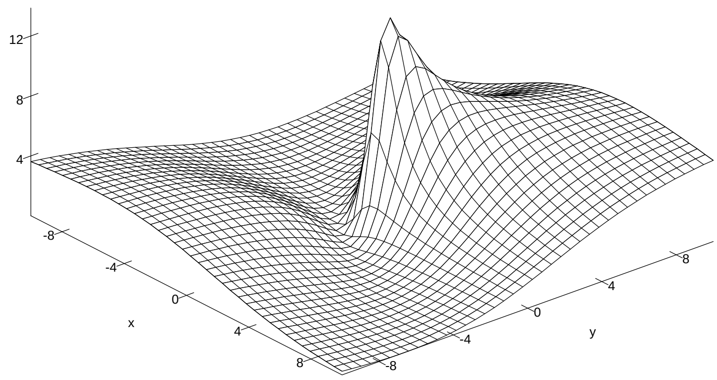

Abbildung 5.11: Graph einer Abstandsfunktion

Ist $E^{(x,y,1)}$ optimale Ursprungsebene, maximiert (x, y) die untere Einhüllende $f^u[\mathcal{F}]$ der Schar der Abstandsfunktionen $\mathcal{F} = \{f_1, \ldots, f_n\}$. Es gilt also, das MaxMin-Problem der Schar $\mathcal{F}$ zu lösen.

5.3.2 Struktur und Berechnung des MD

Gestalt der Separatoren

Die Menge der Normalenvektoren v, deren zugehörige Ebenen gleichweit von zwei vorgegebenen Punkten p_i und p_j entfernt sind, die also $\delta(p_i, E^v) = \delta(p_j, E^v)$ erfüllen, sind Lösungen des Gleichungssystems

$$|x_i v_1 + y_i v_2 + z_i v_3| = |x_j v_1 + y_j v_2 + z_j v_3|$$

Dies sind gerade die Ebenen

$$E^{p_i-p_j} : <p_i - p_j, v> = 0 \quad \text{und} \quad E^{p_i+p_j} : <p_i + p_j, v> = 0$$

Der Separator der Funktionen f_i und f_j entspricht dem Schnitt der beiden Ebenen $E^{p_i-p_j}$ und $E^{p_i+p_j}$ mit der Tangentialebene $E_\pi : x_3 = 1$, dem Geradenpaar

$$sep_{ij} : (x_i \pm x_j)x + (y_i \pm y_j)y + (z_i \pm z_j) = 0$$

Erfüllen die Funktionen aus $\mathcal{F}$ die Bedingung der allgemeinen Lage, haben also beispielsweise keine 4 Weltpunkte gleichen Abstand zu einer Ursprungsebene, so genügt $\mathcal{F}$ den Voraussetzungen des Satzes 17 : eine vertikale Gerade $x = x_0$ schneidet jeden Separator höchstens zweimal und die Separatoren schneiden sich paarweise in höchstens 4 Punkten. MD[$\mathcal{F}$] kann also in Zeit $O(n\lambda_6(n)\log n)$ berechnet werden.

5.3.3 Lage der Extrema der unteren Einhüllenden

Extrema der Distanzfunktion

Der Abstand einer Ursprungsebene E^v zu einem Punkt p_i wird offensichtlich dann maximal, wenn sich E^v mit E^{p_i} deckt, und minimal, wenn p_i in E^v enthalten ist, also $p_i \perp v$. Dies impliziert ein lokales Maximum von f_i im Punkt $(x_i/z_i, y_i/z_i)$ sowie lokale Minima auf der Geraden $x_i x + y_i y + z_i = 0$. Eine Anwendung des Extremallemmas zeigt, daß keine weiteren lokalen Extrema existieren. Es genügt, zwei ausgezeichnete Lagen des Koordinatensystems zu untersuchen.

Sei $M = x^2 + y^2 + 1$.

Fall 1 $p'_i = (0, 0, z'_i)$

$$f_i(x,y) = \frac{z'^2_i}{M} \qquad \nabla f_i = \frac{-2z'^2_i}{M^2}\begin{pmatrix} x \\ y \end{pmatrix}$$

$(0,0)$ ist einziger stationärer Punkt, ein Maximum. Dieser Punkt entspricht dem Schnittpunkt der Geraden mit Richtungsvektor p_i mit der Ebene E_π.

Fall 2 $p'_i = (0, y'_i, 0)$

$$f_i(x,y) = \frac{y'^2_i y^2}{M} \qquad \nabla f_i(x,y) = \frac{2y'^2_i}{M^2}\begin{pmatrix} -xy^2 \\ y(x^2+1) \end{pmatrix}$$

Stationär sind die Punkte der Geraden $y = 0$, sämtlich lokale Minima. Diese Gerade entspricht der Schnittgeraden der Ebenen E^{p_i} und E_π.

Extrema auf den Separatoren

Wir suchen nun die Extrema der Funktion $f_i(x,y) = \frac{1}{M}(x_i x + y_i y + z_i)^2$ auf dem Separator sep_{ij} der Funktionen f_i und f_j, dem Geradenpaar

$$sep_{ij} : (x_i \pm x_j)x + (y_i \pm y_j) + (z_i \pm z_j) = 0$$

Eine geometrisch-anschauliche Begründung für die Lage der Extrema ist nicht ersichtlich; wir sind einmal mehr auf algebraische Techniken angewiesen.

Lemma 16 *Gegeben seien zwei linear unabhängige Punkte p_i und p_j.*

Die Extrema ihrer Abstandsfunktionen f_i und f_j auf dem Separator von p_i und p_j (sep_{ij}) liegen

- *in den Schnittpunkten der Ursprungsebene $E_{p_i p_j}$ mit sep_{ij}*
- *im Kreuzungspunkt von sep_{ij}, d. h. dem Schnittpunkt der Ebenen $E^{p_i - p_j}$, $E^{p_i + p_j}$ und E_π.*

Beweis: Definiert man die Abstandsfunktionen auf der Einheitskugel, bilden ihre Separatoren Paare sich schneidender Großkreise. Um ein solches Paar auf Extrema der Abstandsfunktion untersuchen zu können projizieren wir sie in drei verschiedenen Ausrichtungen des Koordinatensysstems auf die polare Tangentialebene. Seien p'_i, p'_j die transformierten Punktkoordinaten; wir können p'_i o.B.d.A. als normiert voraussetzen, da Division durch $|p_i|^2$ die Extremaleigenschaften von f_i unberührt läßt.

FALL 1 $p'_i = (0,0,1), p'_j = (x'_j, 0, z'_j) \quad [x'_j \neq 0]$

Gesucht sind die Extrema der Funktion

$$f_i(x,y) = \frac{1}{M} \quad \nabla f_i(x,y) = \frac{-2}{M^2}\begin{pmatrix} x \\ y \end{pmatrix}$$

unter der Nebenbedingung

$$sep_{ij} : x'_j x + z'_j \mp 1 = 0$$

Die Methode der Lagrangemultiplikatoren führt auf die Gleichungssysteme

$$\begin{aligned} \frac{-2x}{M^2} + x'_j\lambda &= 0 \\ \frac{-2y}{M^2} &= 0 \\ x'_j x + z'_j \mp 1 &= 0 \end{aligned}$$

mit den Lösungen $\left(\frac{-z'_j \pm 1}{x'_j}, 0\right)$, den Schnittpunkten von $E_{p_i p_j}$ mit sep_{ij}.

FALL 2 $p_i' = (0,1,0), p_j' = (0, y_j', z_j') \quad [y_i' \neq 0]$

Die Extrema der Funktion

$$f_i(x,y) = \frac{y^2}{M} \quad \nabla f_i(x,y)\frac{2y}{M^2}\begin{pmatrix} -xy \\ x^2+1 \end{pmatrix}$$

unter der Nebenbedingung

$$sep_{ij} : (y_j' \mp 1)y + z_j' = 0$$

sind Lösung eines der Gleichungssysteme

$$\begin{aligned} \frac{-2xy}{M^2} &= 0 \\ \frac{2y(x^2+1)}{M^2} + \lambda(y_j' \mp 1) &= 0 \\ (y_j' \mp 1)y + z_j' &= 0 \end{aligned}$$

Es sind dies die Punkte $(0, \frac{-z_j'}{y_j' \mp 1})$, die ebenfalls Schnittpunkte von $E_{p_i p_j}$ mit sep_{ij} darstellen.

FALL 3 $p_i' = (0,1,0), p_j' = (x_j', y_j', 0) \quad [x_j' \neq 0]$

f_i hat auf dem Separator

$$sep_{ij} : x_j' x + (y_j' \pm 1)y = 0$$

Extrema nur in den Lösungspunkten der Gleichungssysteme

$$\begin{aligned} \frac{-2xy^2}{M^2} + \lambda x_j' &= 0 \\ \frac{2y(x^2+1)}{M^2} + \lambda(y_j \mp 1) &= 0 \\ x_j' x + (y_j' \mp 1) &= 0 \end{aligned}$$

Wir finden als einzige Lösung $(0,0)$, den Kreuzungspunkt des Separators sep_{ij}. ∎

5.3.4 Der MaxMin-Algorithmus

Kritische Punkte der nördlichen Tangentialebene sind also die $O(n)$ Maxima der Abstandsfunktion, die $O(n^2)$ potentiellen Extrema aus Lemma 16, sowie die $O(n\lambda_6(n))$ Knoten des MD. Gestaltet man den MaxMin-Algorithmus analog zu dem des Abschnitts 5.2.5, läßt sich der f^u maximierende kritische Punkt in Zeit $O(n\lambda_6(n)\log n)$ bestimmen.

Widmen wir uns nun den Ebenen, deren Normalenvektoren in die Äquatorebene der Einheitskugel fallen. Die bezüglich des MaxMin-Kriteriums beste dieser Ebenen findet man durch Lösen der MaxMin-Aufgabe der Funktionenschar $\mathcal{G} = \{g_1, \ldots, g_n\}$ mit

$$g_i : [-1,1] \to \mathbb{R};\ g_i(x) = (x_i x + y_i\sqrt{1-x^2})^2$$

g_i ordnet der Ebene E, deren Normalenvektor die Einheitskugel in $(x, \sqrt{1-x^2}, 0)$ schneidet, den quadrierten Abstand von p_i zu E zu. Dieses Problem ist äquivalent zu (P5) aus Kapitel 3 und kann mit den dort ausgeführten Techniken in Zeit $O(n \log n)$ gelöst werden.

Aus obiger Diskussion folgt unmittelbar

Satz 21 *Haben keine 4 Weltpunkte gleichen Abstand von einer Ursprungsebene, so kann (P19) im Zeit $O(n\lambda_6(n)\log n)$ gelöst werden.*

Kapitel 6

Zusammenfassung und Ausblick

Wir haben in dieser Arbeit eine Strategie entwickelt, die es erlaubt, Lageoptimierungsprobleme nach dem Minimax-Kriterium für affine Teilräume des $\mathbb{R}^3$ mit bis zu zwei Freiheitsgraden zu lösen. Die Grundidee, sämtliche lokale Maxima der unteren Einhüllenden der Schar der Abstandsfunktionen aufzuzählen, konnte durch Kombination von Techniken der algoritmischen Geometrie und der nichtlinearen Optimierung effizient umgesetzt werden. Auf diese Weise konnten wir für das Problem des gewichteten Maximin-Punktes der Ebene bestehende Zeitschranken verbessern, und für die in Kapitel 5 betrachteten Probleme erstmals Lösungsalgorithmen angeben. Für einige Positionierungsprobleme mit einem Freiheitsgrad konnte die Optimalität der Algorithmen gezeigt werden. Die allgemeine Strategie läßt sich auch auf duale Minimax-Probleme übertragen. Die meisten der vorgestellten Algorithmen erscheinen durchaus implementierbar, was die praktische Bedeutung der Ergebnisse etwa im Vergleich zu numerischen Ansätzen unterstreicht.

Die betrachteten Probleme wurden durch eine Vielzahl möglicher Anwendungen motiviert; einige davon geben Anstoß, Verallgemeinerungen der erzielten Ergebnisse anzustreben. So erscheint es im Zusammenhang mit stereotaktischen Operationen (vgl. Abschnitt 5.2.1) wünschenswert, komplexere Hindernisse wie etwa Kugeln oder Polyeder als Weltobjekte zuzulassen, um eine präzisere Modellierung der Realität zu ermöglichen. Ist andererseits der (gewichtete) euklidische Abstand kein ausreichend genaues Maß, um die Wechselwirkung zwischen Ziel- und Weltobjekten zu repräsentieren, stellt sich die Frage, inwieweit der Ansatz auf andere Metriken oder verallgemeinerte Abstandsfunktionen übertragbar ist.

Das bei Kollisionsberechnungen in der Robotik nützliche Problem, den minimalen, eine gegebene Punktmenge umfassenden zylindrischen Hüllkörper zu finden ist äquivalent zum Problem der Minimax-Geraden im Raum. Dieses ist mit den Problemen der Maximin-Geraden und der Maximin-Ebene im Raum offen geblieben; laut [KM93] kennt man für keines dieser Probleme kombinatorische Algorithmen. In allen Fällen besitzen die Zielobjekte mehr als zwei Freiheitsgrade. Dies regt Überlegungen an, ob die allgemeine Strategie auf solche höherdimensionalen Probleme ausgeweitet werden kann. Zur Zeit kennt man zwar nichttriviale Komplexitätsresultate für höherdimensionale Minimierungsdiagramme von Polynomfunktionen aus [Sh93], deren Umsetzung in konstruktive Algorithmen ist jedoch für $d > 2$ noch nicht gelungen.

Anhang A

Kurven und Flächen zweiter Ordnung

A.1 Kurven zweiter Ordnung

Als Kurven zweiter Ordnung (*Kegelschnitte*) bezeichnet man Kurven, deren kartesische Koordinaten einer Gleichung der Form

$$ax^2 + 2bxy + cy^2 + 2dx + 2ey + f = 0$$

genügen. Die Größen

$$\Delta = \begin{vmatrix} a & b & d \\ b & c & e \\ d & e & f \end{vmatrix}, \qquad \delta = \begin{vmatrix} a & b \\ b & c \end{vmatrix}, \qquad S = a + c$$

sind *Invarianten der Kurve*, d. h. sie bleiben bei Transformation des Koordinatensystems in Form einer Translation oder Rotation erhalten.

Kurven zweiter Ordnung können in Gestalt eines Kreises, einer Ellipse, Hyperbel, Parabel oder als Geradenpaar erscheinen. Der Name Kegelschnitt rührt daher, daß sich alle diese Kurven als Schnittlinie eines geraden Kreiskegels mit einer Ebene darstellen lassen.

Die Gestalt eines Kegelschnitts läßt sich aus seinen Invarianten bestimmen. Den Zusammenhang beschreibt Tabelle A.1 [BS93].

	Invarianten δ	Δ	*Gestalt der Kurve*
Mittelpunktskurven	$\delta > 0$	$\Delta \neq 0$	*Ellipse* reell für $\Delta \cdot S < 0$ imaginär für $\Delta \cdot S > 0$
		$\Delta = 0$	Paar *imaginärer Geraden*
	$\delta < 0$	$\Delta \neq 0$	*Hyperbel*
		$\Delta = 0$	Paar *sich schneidender Geraden*
Parabolische Kurven	$\delta = 0$	$\Delta \neq 0$	*Parabel*
		$\Delta = 0$	*Geradenpaar* parallele Geraden für $d^2 - af > 0$ Doppelgerade für $d^2 - af = 0$ imaginäre Geraden für $d^2 - af < 0$

Tabelle A.1: Klassifizierung der Kurven 2. Ordnung

A.2 Flächen zweiter Ordnung

Als Flächen zweiter Ordnung bezeichnet man Flächen im Raum, deren kartesische Koordinaten einer Gleichung der Form

$$a_{11}x^2 + a_{22}y^2 + a_{33}z^2 + 2a_{12}xy + 2a_{23}yz + 2a_{31}zx + 2a_{14}x + 2a_{24}y + 2a_{34}z + a_{44} = 0$$

genügen.

Mit der Identität $a_{ik} = a_{ki}$ sind die Größen

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{vmatrix}, \qquad \delta = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix},$$

$$S = a_{11} + a_{22} + a_{33}, \qquad T = a_{22}a_{33} + a_{33}a_{11} + a_{11}a_{22} - a_{23}^2 - a_{31}^2 - a_{12}^2$$

Invarianten der Fläche, d. h. sie bleiben bei Koordinatentransformation erhalten. Flächen zweiter Ordnung können in Gestalt eines Ellipsoids, Paraboloids, Hyperboloids, Zylinders oder eines Ebenenpaares auftreten. Kennt man die Invarianten der Fläche, kann man ihre Form anhand von Tabelle A.2 [BS93] bestimmen.

Mittelpunktsflächen ($\delta \neq 0$)

	$S \cdot \delta > 0, T > 0$	$S \cdot \delta < 0$ oder $T < 0$
$\Delta < 0$	*Ellipsoid* $\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1$	zweischaliges *Hyperboloid* $\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = -1$
$\Delta > 0$	imaginäres *Ellipsoid* $\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = -1$	einschaliges *Hyperboloid* $\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 1$
$\Delta = 0$	imaginärer *Kegel* $\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 0$	*Kegel* $\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 0$

Paraboloide, Zylinder und Ebenenpaare ($\delta = 0$)

	$\Delta < 0, T > 0$	$\Delta > 0, T < 0$
$\Delta \neq 0$	elliptisches *Paraboloid* $\frac{x^2}{a^2} + \frac{y^2}{b^2} = \pm z$	hyperbolisches *Paraboloid* $\frac{x^2}{a^2} - \frac{y^2}{b^2} = \pm z$
$\Delta = 0$	*Zylinderfläche* imaginärer elliptischer Zylinder für $T > 0$ parabolischer Zylinder für $T = 0$ hyperbolischer Zylinder für $T < 0$ oder die Fläche zerfällt in ein (reelles, imaginäres) *Ebenenpaar*	

Tabelle A.2: Klassifizierung der Flächen 2. Ordnung

Literaturverzeichnis

[ASS] P. Agarwal, M. Sharir und P. Shor, Sharp upper and lower bounds on the length of Davenport-Schinzel sequences, eingereicht bei *J. Combin. Theory* Ser. A,

[AIIT90] H. Aonuma, H, Imai, K. Imai und T. Tokuyama, Maximin location of convex objekts in a polygon and related dynamic Voronoi diagrams, *Proc. ACM* (1990), 225-234

[At83] M. Athalla, Dynamic computational geometry, *Proc. 24th IEEE Sympos. Found. Comp. Sci.* (1983), 92-99

[At85] M. Athalla, Some dynamic computational geometry problems, *Comp. and Maths. with Appl.* 11 (1985), 1171-1181

[AE84] F. Aurenhammer und H. Edelsbrunner, An optimal algorithm for constructing the weighted Voronoi diagram in the plane, *Pattern Recognition* 17 (1984), 251-257

[Au87] F. Aurenhammer, Power diagrams: properties, algorithms and applications, *SIAM J. Comput.* 16 (1987), 78-96

[Au91] F. Aurenhammer, Voronoi diagrams - a survey of a fundamental geometric data structure, *ACM computing surveys* (1991), 345-405

[Ba86] C. Bajaj, Proving geometic algorithms nonsolvability: an application of factoring polynomials *J. Symbolic Comput.* 2 (1986), 99 -102

[Ba88] C. Bajaj, The algebraic degree of geometric optimization problems, *Discrete Comput. Geom.* 3 (1988), 257-265

[Be83] M. Ben-Or, Lower bounds for algebraic computation trees, *Proc. 15th ACM Symp. Theo. Comp.* (1993), 498-507

[BS93] I. Bronstein und K. Semendjajew, *Taschenbuch der Mathematik* Harri Deutsch (1993)

[C84a] R. Cole, Slowing down sorting networks to obtain faster sorting algorithms, *25th IEEE Sympos. Found. Comp. Sci.* (1984), 255-260

[C84b] R. Cole, Searching and storing similar lists, *J. of Algorithms* 7 (1986), 202-220

[C59] J. Coolidge, *A treatise on algebraic plane curves* (1959), Dover

[DS65] H. Davenport und A. Schinzel, A combinatorial problem connected with differential equations, *Amer. J. Math.* 87 (1965), 684-694

[Ed87] H. Edelsbrunner, *Algorithms in combinatorial geometry*, Springer-Verlag (1987)

[ES86] H. Edelsbrunner und R. Seidel, Voronoi diagrams and arrangements, *Discrete Comput. Geom.* 1 (1986), 25-44

[Go75] T. Gonzales, Algorithms on sets and related problems, Tech. Report, Dept. of Computer Science, University of Oklahoma (1975)

[GS88] L. Guibas und J. Stolfi, Ruler, compass and computer: the design and analysis of geometric algorithms. In R. Earnshaw (ed.) *Theoretical foundations of computer graphics and CAD*, Nato ASI series, vol. F40

[Ho79] R. Horst, *Nichtlineare Optimierung*, Hanser Verlag (1979)

[HM88] M. Houle und A. Maciel, Finding the widest empty corridor through a set of points. In G. Toussaint (ed.) *Snapshots of computational and discrete geometry*, School of Computer Science, McGill University, Montreal (1988)

[HIIRY90] M. Houle, H. Imai, K. Imai, J.Roberts and P. Yamamoto, Orthogonal weighted linear L_1- and L_∞- approximation and applications, eingereicht bei *Discrete Appl. Math.* (1990)

[HKS91] D. Huttenlocher, K. Kedem und M. Sharir, The upper envelope of Voronoi surfaces and its applications, *7th Symp. Comput. Geometry* (1991), 194-203

[KM93] N.M. Korneenko und H. Martini, Hyperplane Approximation and related topics. In J.Pach (ed.) *New trends in combinatorial and computational geometry* (1993)

[LP84] D.T. Lee und F. Preparata, Computational geometry - a survey, *IEEE Transactions on computers* 33 (1984), 1072-1101

[Lee86] D.T.Lee, Geometric location problems and their complexity, *LNCS* 233 (1986), 154-167

[LW86] D.T.Lee und Y.F. Wu, Geometric complexity of some location problems, *Algorithmica* 1 (1986), 193-211

[Lem86] C. Lemarechal, An introduction to the theory of nonsmooth optimization, *Optimization* 17 (1986), 827-858

[Lem89] C. Lemarechal, Nondifferentiable optimization. In G. Nemhauser (ed.) *Handbooks in operations research and management science*, North-Holland (1989)

[LMW88] R.F. Love, J.G.Morris und G.O.Wesolowsky, *Facility location: models and methods*, North-Holland (1988).

[Me84] N. Megiddo, Linear programming in linear time when the dimension is fixed, *J ACM* 31 (1984), 114-127

[Meh84] K. Mehlhorn, *Data Structures and algorithms 3: Multi-dimensional sorting and computational geometry*, Springer-Verlag (1984)

[Mel80] E. Melachrinoudis, *The Maximin single facility location problem using an euclidian metric*, PhD Thesis, Dept. of Industrial Engeneering and Operations research, Univerity of Massachusetts, Amherst, MA (1980)

[MC85a] E. Melachrinoudis und T. Cullinane, A heuristic approach to the single facility maximin location problem. *Int. J. Prod. Res.* 23 (1985), 523-532

[MC85b] E. Melachrinoudis und T. Cullinane, Locating an undesirable facility within a geographic region using the maximin criterion. J. Reg. Sci. 25 (1985), 115-127

[MC86] E. Melachrinoudis und T. Cullinane, Locating an obnoxious facility within a polygonal region. *Annals of operations research* 6 (1986) 137-145

[PG92] C. Pandu Rangan und R. Govindan, An $O(n \log n)$ algorithm for a maximin location problem. *Discrete Applied Mathematics* 36 (1992), 203-205

[PS85] F. Preparata und M. Shamos, *Computational geometry: an introduction* Springer-Verlag (1985)

[SS90] J.T.Schwartz und M. Sharir, On the two-dimensional Davenport-Schinzel problem, *J. Symbolic Computation* 10 (1990), 371-393

[Sh78] M. Shamos, *Computational geometry*, PhD. thesis, Dept. of Computer Science, Yale university (1978)

[Sh88a] M. Sharir, Improved lower bounds on the length of general Davenport-Schinzel sequences, *Combinatorica* 8 (1988), 117-124

[Sh88b] M. Sharir, Davenport-Schinzel sequences and their geometric applications. In R. Earnshaw (ed.) *Theoretical foundations of computer graphics and CAD*, Nato ASI series vol. F40 (1988)

[Sh93] M. Sharir, Almost tight upper bounds for lower envelopes in higher dimensions, *Proc 34th IEEE Sympos. Found. Comp. Sci.*

[SM92] J. Smith und P. Winter, Computational geometry and topological network design. In D. Du, F.Hwang (eds.) *Computing in euclidian geometry*, World Scientific Publ. Co. (1992)

[Sz74] E. Szemeredi, On a problem by Davenport and Schinzel, *Acta Arithmetica* 25 (1974), 213-224

[To83] G. Toussaint, Computating largest empty circles with location constraints, *Int. J. Comput. and Inform. Sci.* 12 (1983), 347-358

[Y90] F.F. Yao, Computational geometry. In J. van Leeuwen (ed.) *Handbook of Theor. Computer Science* vol. A (1990), Elsevier