AK Rechnerische Geometrie

Jan Pöschko
auf Grundlage einer Vorlesung von
Univ.-Prof. Dipl.-Ing. Dr. Franz Aurenhammer
im Wintersemester 2009/2010

2. Oktober 2012

Inhaltsverzeichnis

1 Arrangements von Geraden

$L = \{g_1, \ldots, g_n\}$, $A(L)$ mit $|L| = n$: $\binom{n}{2}$ Knoten, n^2 Kanten, $1 + n + \binom{n}{2}$ Zellen
Signatur $\sigma(p) = (\sigma_i)_i$ mit

$$\sigma_i = \begin{cases} + & p \text{ unter } g_i \\ 0 & p \text{ auf } g_i \\ - & p \text{ über } g_i \end{cases}$$

1.1 Dualität

$$p = \begin{pmatrix} a \\ b \end{pmatrix} \leftrightarrow T(p) : y = 2ax - b$$

Dualität ist eigentlich Polarität bezüglich Parabel $y = x^2$.
Eigenschafen:

1. erhält relative Lage zwischen Punkten und Geraden: p über g $\Leftrightarrow$T(g) über $T(p)$
2. erhält Inzidenzen: $s = g \cap h \Leftrightarrow T(s)$ durch Punkte $T(g)$ und $T(h)$

1.2 Kollinearität von Punkten

p, q, r kollinear $\Leftrightarrow$ $T(p)$, $T(q)$, $T(r)$ durch gemeinsamen Punkt

1.3 Levels in Arrangements

k-Level: $\{e \mid \sigma(e) \text{ hat genau } k - 1 +\}$
Größe von k-Level in Arrangement von n Geraden: 1971: $\mathcal{O}(n\sqrt{k}) \rightarrow$ 2000: $ne^{\Omega(\sqrt{\log k})}$

1.4 Konstruktion von Arrangements

direkt: pro Gerade $n - 1$ Schnitte, sortieren: $\mathcal{O}(n^2 \log n)$
plain sweep: ebenso
Einfügen: konstruiere $A(L \cup \{g\})$ aus $A(L)$; Laufzeit $\mathcal{O}(n^2)$
Zonen-Theorem: Die Anzahl der Kanten der Zellen in der Zone von g (Zellen von $A(L)$, die von g geschnitten werden) ist $\leq 6n$. (Beweis: injektive Abbildung von Rechtskanten auf Schnittpunkte von g)

1.5 Dreieck kleinster Fläche

Im kleinsten Dreieck (i, j, k) hat p_k den kleinsten Abstand von $\overline{p_i p_j}$.
Wird $v = T(\overline{p_i p_j})$ vertikal bewegt, so ist $T(p_k)$ die erste Gerade, die getroffen wird. $\Rightarrow$ Knoten v gehört zu einer Zelle in $A(T(p_1), \ldots, T(p_n))$, zu der auch $T(p_k)$ eine Kante beiträgt.
Algorithmus:

1. Dualisieren der p_i

2. Konstruiere Arrangement

3. Für alle Zellen F: für alle Knoten v von F: für alle Kanten e (nicht inzident zu v): betrachte (i, j, k), wobei $v = T(p_i) \cap T(p_j)$ und $e \subset T(p_k)$

Analyse:

$$\sum_{F \in A} d^2(F) \leq 10n^2$$

weil $\sum_{g_i} d(F) \leq 10n$ laut Zonentheorem.

1.6 Halbebenen-Suchproblem

Bestimme alle Punkte in S, die oberhalb von g liegen.

Äquivalent zum Finden aller Geraden, die in $A(T(S))$ unter $T(g)$ liegen. Speichere Zeiger von allen oberen Kanten in einer Zelle im Arrangement zu einer unteren Kante e.

Laufzeit $Q(n) = \mathcal{O}(k + \log n)$, Speicher $S(n) = \Theta(n^2)$

1.7 Partitionsbäume

Halbebenen-Suchproblem mit $S(n) = \mathcal{O}(n)$, $Q(n) = \mathcal{O}(k + n^{0.7})$

Ham-Sandwich-Theorem: für Gerade g, die S im Verhältnis $n_1 + n_2 : n_3 + n_4$ aufteilt, gibt es eine weitere Gerade g', sodass S in 4 Teile der Größen n_1, n_2, n_3, n_4 zerteilt wird. (Beweis: $\alpha(x)$: kleinster Winkel in x über g, sodass $n_1 : n_2$-Teilung stimmt; $\beta(x)$: kleinster Winkel, sodass $n_3 : n_4$ stimmt; $\alpha : 0 \to \pi$, $\beta : \pi \to 0$ stetig)

Balancierter Binärbaum; jeder Knoten hat 2 Pointer auf ersten und letzten Punkt in seinem Teilbaum.

$Q(n) \leq 3Q(\frac{n}{4}) + \mathcal{O}(1) \Rightarrow Q(n) = \mathcal{O}(n^{0.77})$

genauer: $Q(2^{k+1}) \leq Q(2^k) + Q(2^{k-1}) + c \Rightarrow Q(2^{k+1}) \leq 2^{(k+1)\,\mathrm{ld}(1+\sqrt{5})-(k+1)} - c$

Aufbau HS-Gerade zwischen S_1 und S_2 (getrennt durch y-Achse): Median-Level in $T(S_1)$ (monoton fallend in x) und $T(S_2)$ (monoton steigend); suche Schnittpunkt r; $T(r)$ ist HS-Gerade.

Berechnung des Median-Level per Datenstruktur für dynamischen Schnitt von Halbebenen. $P(n) = \mathcal{O}(n^{\frac{4}{3}+\varepsilon})$

2 Parametrische Suche

Paralleler Entscheidungsalgorithmus $\to$ serieller Optimierungsalgorithmus.

Betrachte ENT: $P(x)$ monoton, d.h. aus $P(x_0)$ folgt $P(x)$ für $x < x_0$.

OPT: finde größtes x^* mit $P(x^*)$.

Wende Algorithmus A für $P(x)$ auf (unbekannte) Lösung x^* des Optimierungsproblems an. $\to$ Entscheidungen $F_i(x^*) \leq 0$.

- Berechne Nullstellen von F_i: $x_1, \ldots, x_k$

- Wende Entscheidungsalgorithmus A auf $x_1, \ldots, x_k$ and

- Weil P monoton: $\exists^1$ Index j mit $A(x_j) = T$ und $A(x_{j+1}) = F$; somit $x^* \in I = [x_j, x_{j+1}]$.

2.1 Median-Level

betrachte Arrangement von Geraden $g_1, \dots, g_n$, Median-Level $M(x)$ = Median von $g(x), \dots, g_n(x)$;

ENT: geg. x, entscheide $M(x) \le 0$ (berechne $g_i(x)$, Median davon, teste $g_m(x) \le 0$)

OPT: bei Median-Berechnung tritt $g_i(x^*) \le g_j(x^*)$ auf; berechne also Nullstelle von

$$f_{ij} = a_i x + b_i - a_j x + b_j,$$

das ist x-Koordinate von $g_i \cap g_j$; berechne $A(x_{ij})$

2.2 Theil-Schätzer

Finde Gerade $\overline{p_j p_j}$, die den Median aller $\binom{n}{2}$ Steigungen realisiert.

$\overline{p_i p_j}$ hat k-kleinste Steigung $\Leftrightarrow x_{ij} = T(\overline{p_i p_j})$ hat k-kleinste x-Koordinate, d.h. x_{ij} ist der k-te Knoten von links im Arrangement $A(T(p_1), \dots, T(p_n))$.

ENT: geg. x, entscheide $L(x)$ = Anzahl der Knoten in A links von $x \le \frac{1}{2}\binom{n}{2}$. (berechenbar über Anzahl der Inversionen der Permutation $\pi(x)$ der g_i)

OPT: berechne $L(x_{ij})$, wobei x_{ij} ist x-Koordinate von $g_i \cap g_j$, usw.

Rechenzeit: $\mathcal{O}(n \log n \cdot n \log n)$

Parallelisieren: Bitones Sortieren; $\mathcal{O}(\log^2 n)$ Runden, pro Runde n Vergleiche; betrachte n Nullstellen in $\mathcal{O}(n)$, sortiere sie in $\mathcal{O}(n \log n)$, Binärsuche in dieser sortieren Liste mit $\mathcal{O}(\log n)$ Schritten zu $\mathcal{O}(n \log n)$ Zeit ($L(x)$ berechnen). $\Rightarrow \mathcal{O}(n \log^4 n)$ Zeit

Verbesserung durch Intervalldrittelung auf $\mathcal{O}(n \log^2 n)$ und Randomisierung

3 Rückwärtsanalyse

3.1 Mittelachse von konvexen Polygonen

$$M(P) = \{x \in P \mid \text{nächster Nachbar von } x \text{ bezüglich Rand von } P \text{ ist mehrdeutig}\}$$

teilt P in n Regionen, Baumstruktur: n Blätter (Ecken von P), $n-2$ innere Knoten, $2n-3$ Kanten

Randomisiertes Einfügen: $\mathcal{O}(|\mathrm{reg}(e)|)$ pro Kante e (merke Nachbarkanten)

$$\mathbb{E}[|\mathrm{reg}(e)|] = \frac{1}{n} \sum |\mathrm{reg}(e_i)| = \frac{1}{n} 2(2n-3) < 4 = \mathcal{O}(1)$$

3.2 Lineares Optimieren

Normalvektor z einer Hyperebene in $\mathbb{R}^d$, Menge H von n Halbräumen, $\bigcap_{h \in H} h = P$ konvexes Polytop.

Finde $x \in P$, sodass x Tangentialpunkt einer Hyperebene mit Normalvektor z an P.

Größe von P: $\Theta(n^{\lfloor \frac{d}{2} \rfloor})$
Rekursiver Algorithmus für $\mathrm{LP}(H, z)$:

- $d = 1$: P = Intervall, Optimallösung = Endpunkt
- $d = n$: P hat nur eine Ecke $\Rightarrow$ lineares Gleichungssystem in $\mathcal{O}(d^3)$
- $n > d > 1$: wähle $h \in H$ zufällig, löse $\mathrm{LP}(H \setminus \{h\}, z)$ mit Lösung w
  - $w \in h$: bereits Gesamtlösung
  - $w \notin h$ (mit Wahrscheinlichkeit $\frac{d}{n}$): Optimallösung liegt in begrenzender Hyperebene g von h; bilde $H' = (H \setminus \{h\}) \cap g$, projiziere z auf g, löse $\mathrm{LP}(H', z')$

$$T(n, d) = \mathcal{O}(d!n)$$

kann von $d!$ auf $2^{\sqrt{d}}$ gedrückt werden.

3.3 Böses Beispiel

Triangulierung von n Punkten p_j und Dreieck t, $p_j \in t$

4 Monotone Matrizen

$n \times m$-Matrix A $(n \leq m)$ monoton: aus $i_1 < i_2$ folgt $j(i_1) \leq j(i_2)$ (j: Spaltenindex des linkesten Maximums in Zeile i)

A total monoton: alle Submatrizen (äquivalent: alle 2×2-Submatrizen) monoton

$\begin{pmatrix} a & b \\ d & d \end{pmatrix}$ monoton $\Rightarrow$ aus $a < b$ folgt $c < d$

Wenn $j_1 < j_2$ und $A(i, j_1) \geq A(i, j_2)$, dann sind alle $A(x, j_2)$ tot für $1 \leq x \leq i$.

Wenn $j_1 < j_2$ und $A(i, j_1) < A(i, j_2)$, dann sind alle $A(x, j_1)$ tot für $i \leq x \leq n$.

$\Rightarrow$ Reduce-Algorithmus, der A auf $n \times n$-Matrix C reduziert, sodass Maxima nicht verloren gehen.

Zeilenmaxima und globales Maximum in $\mathcal{O}(m)$ Zeit.

4.1 Weiteste Nachbarn

in konvexem Polygon: $n \times (2n - 1)$-Matrix

$$A_{ij} = \begin{cases} j - i & \text{für } j \leq i \\ d(p_i, p_j) & i < j < i + n \\ -1 & \text{für } j \geq i + n \end{cases}$$

Beweis der Monotonie der 2×2-Submatrizen per Vierecksungleichung.

Anwendung: flächengrößtes Dreieck.

4.2 Euclidean Distance Transform

Finde zu jedem von n^2 Gitterpunkten den Abstand zu seinem nächsten Nachbarn in einer Gitterpunktmenge B.

mit monotonen Matrizen in $\mathcal{O}(n^2)$ Zeit

5 n^2-schwere Probleme

- 3SUM: geg. ganze Zahlen M; gefr. $\exists a, b, c \in M : a + b + c = 0$
- 3SUM': geg. ganze Zahlen A, B, C; gefr. $\exists a \in A, b \in B, c \in C : a + b = c$ (sortiere B, C; bilde $\forall a \in A$: $a + B$; gleiche mit C ab)
- Koll: geg. n Punkte; gefr. sind drei davon kollinear (per Arrangements)

  3SUM $\leq$ Koll: für $m \in M$ betrachte Punkt $p(m) = \binom{m}{m^3}$; $a, b, c \in M$ haben Summe 0 $\Leftrightarrow p(a), p(b), p(c)$ kollinear.
- Koll': geg. n Punkte $p = \binom{x}{y}$, $x \in \mathbb{N}$, $y \in \{0, 1, 2\}$; gefr. $\exists$ nicht-horizontale Gerade, die mehr als zwei Punkte enthält

  3SUM' $\leq$ Koll': $b \in B \to (2b, 2)$, $c \in C \to (c, 1)$, $a \in A \to (2a, 0)$.
- SEP: geg. n Objekte (Spezialfall: Liniensegmente); gefr. $\exists$ Gerade, die Objekte vermeidet und in zwei nichtleere Teilmengen zerlegt

  Koll' $\leq$ SEP: dreizeiliges Gitter mit ausgesparten Punkten
- Strips-Cover-$\triangle$: geg. Dreieck, n Streifen; gefr. wird Dreieck von Streifen abgedeckt

  SEP $\leq$ Strips-Cover-$\triangle$: $p = \binom{a}{b} \mapsto T(p) : y = 2ax - b$ (vertikales Segment $\mapsto$ Streifen, Halbstrahl $\mapsto$ Halbebene, separierende Gerade $\mapsto$ Punkt);

  $T(g)$ liegt nicht in Streifen/Halbebenen $\Leftrightarrow$ Streifen überdecken Dreieck.
- Triangles-Cover-$\triangle$ $\geq$ Strips-Cover-$\triangle$: baue Dreiecke aus Streifen.
- Hole-in-Union: geg. n Objekte; gefr. enthält ihre Vereinigung ein „Loch"

  Triangles-Cover-$\triangle$ $\leq$ Hole-in-Union: geg. Menge T von Dreiecken, bilde $t \cap \triangle$ (triangulieren, wenn nötig) $\Rightarrow$ Menge T' von Dreiecken mit $|T'| \leq 4\,|T|$ und T' überdeckt $\triangle$ (d.h. $\bigcup t'$ hat keine Löcher) $\Leftrightarrow T$ überdeckt $\triangle$.
- Area-of-Union: geg. n Dreiecke; ges. Fläche ihrer Vereinigung

  Triangles-Cover-$\triangle$ $\leq$ Area-of-Union: T' überdeckt $\triangle \Leftrightarrow$ abgedeckte Fläche ist Fläche von $\triangle$