

Übungsblatt 4

Vorlesung Algorithmen II im WS 12/13

Ausgabe 29. November 2012

Besprechung 11. Dezember 2012

Problem 1: Schnitte von Strecken

Sei S eine Menge von n Strecken. Der Algorithmus aus der Vorlesung kann in $O(n \log n)$ Zeit testen, ob es in S ein sich schneidendes Streckenpaar gibt.

- Zeigen oder widerlegen Sie: Es wird immer der linkeste Schnittpunkt gefunden.
- Geben Sie einen Algorithmus an, der *alle* Schnitte zwischen Streckenpaaren in S in $O((n + k) \cdot \log n)$ Zeit berechnet und ausgibt, wobei k die Anzahl der Schnittpunkte ist.
- Zeigen Sie, dass Ihr Algorithmus mit $O(n + k)$ Speicherplatz auskommt. Ist es möglich, den nötigen Speicherplatz auf $O(n)$ zu reduzieren?

Problem 2: Schnitte von Polygonen

Seien P und Q zwei einfache Polygone. Als *Schnitt* $P \cap Q$ von P und Q bezeichnen wir die Menge der Punkte, die sowohl im Inneren von P als auch im Inneren von Q liegen. Gehen Sie wie folgt vor, um einen Algorithmus mit $O((n + k) \cdot \log n)$ Laufzeit zur Berechnung von $P \cap Q$ zu entwerfen, wobei k die Anzahl der Schnittpunkte zwischen Polygonkanten ist. Sie dürfen dabei annehmen, dass P und Q keine Eckpunkte gemeinsam haben und dass kein Eckpunkt auf einer Polygonkante liegt.

- Machen Sie sich klar, dass $P \cap Q$ aus dem inneren mehrerer Polygone bestehen kann.
- Geben Sie einen Algorithmus mit Laufzeit $O(n \log n)$ an, der entscheidet welcher der folgenden drei Fälle auftritt.
 - Der Schnitt $P \cap Q$ ist leer.
 - P ist in Q enthalten oder umgekehrt.
 - Der Schnitt $P \cap Q$ ist nicht leer und weder gleich P noch gleich Q .
- In den ersten beiden Fällen ist nichts weiter zu tun, um den Schnitt der beiden Polygone zu berechnen. Entwickeln Sie im Folgenden einen Sweep-Line Algorithmus, der den Schnitt $P \cap Q$ für den dritten Fall berechnet. Gehen Sie wie folgt vor.
 - Verwenden Sie als Haltepunkte sowohl die Eckpunkte der beiden Polygone, als auch die Schnittpunkte zwischen Polygonkanten. Welche Fälle können bei Haltepunkten auftreten?
 - Welche Daten sollen als Sweep-Line Zustand gespeichert werden? Dabei sollte zumindest für jeden Punkt auf der aktuellen Sweep-Line bekannt sein, ob er in $P \cap Q$ liegt oder nicht.
 - Wie ändert sich der Sweep-Line Zustand für die in Punkt 1. identifizierten Fälle.
 - Müssen Sie zusätzliche Informationen mitführen, um am Ende den Schnitt $P \cap Q$ als Menge von Polygonen auszugeben?

Problem 3: Konvexe Hüllen von Polygonen

- (a) Sei P ein einfaches (nicht notwendigerweise konvexes) Polygon mit n Eckpunkten. Geben Sie einen Algorithmus mit Laufzeit $O(n)$ zur Berechnung der konvexen Hülle von P an.
- (b) In der Vorlesung wurde gezeigt, dass die Laufzeit zur Berechnung der konvexen Hülle einer Punktmenge mit n Punkten in $\Theta(n \log n)$ liegt. Es gibt also keinen Algorithmus, der asymptotisch echt weniger als $n \log n$ Schritte benötigt. Warum ist das kein Widerspruch zu Teilaufgabe (a)?

Problem 4: Punkte auf der konvexen Hülle

Gegeben sei eine Punktmenge Q . Zeigen Sie folgende Aussagen.

- (a) Seien $p, q \in Q$ zwei Punkte, sodass p der am weitesten von q entfernte Punkt ist. Dann ist p ein Eckpunkt der konvexen Hülle $H(Q)$.
- (b) Ein Punkt $p \in Q$ ist ein Eckpunkt der Konvexen Hülle $H(Q)$ genau dann, wenn es eine Gerade g durch p gibt, sodass alle Punkte $Q \setminus \{p\}$ auf der gleichen Seite von g liegen.