11. Übung Informatik I

Marcus Rickert

30. Dezember 1995

Aufgabe 1

Ich benutze für diese Aufgabe die in der Vorlesung vorgeschlagene Datenstruktur, wo für jeden Punkt $p \in S$ die 'Kanten' seines Voronoigebietes als doppelt zirkulär verkettete Liste abgelegt werden. Dabei wird eine Kante als Anfangs- und Endpunkt abgespeichert, falls sie nicht entartet ist bzw. als Anfangspunkt und Richtung, falls sie zu einem Strahl entartet ist.

Außerdem enhält jede Kante einen Zeiger auf den Punkt aus S, dessen Voronoigebiet ebenfalls an diese Kante grenzt.

zu (i)

Algorithmus

- Wähle einen Punkt $p \in S$.
- Gehe die Kantenliste des Voronoigebietes dieses Punktes durch und berechne jeweils den Abstand des Punktes zu jeder Kante (egal ob Strecke oder Strahl).
- Suche hiervon linear das Minimum s_0, das bei Kante vom Nachbarpunkt p_i erhalten wurde. s_0 entspricht dem halben Abstand vom Punkt p zum nächsten Nachbar.
- p_i ist der gesuchte Nachbar.
- Wiederhole obige Schritte für alle Punkte aus S.

Laufzeit

Die Ermittlung aller Abstände eines Punktes zu allen Kanten und lineare Dursuchen der Abstände hat die Laufzeit $O(m_i)$, wobei m_i die Anzahl der Kanten des Voronoigebietes des jeweiligen Punktes ist. Laut Vorlesung hat das Voronoidiagramm höchstens $m = 3n - 6$ Kanten, so daß sich die Gesamtlaufzeit auf $O(m)$=$O(n)$ beläuft.

zu (ii)

Algorithmus

- Durchsuche die Kantenlisten der Voronoigebiete aller Punkte $p \in S$ bis das erste mal ein Strahl als Kante gefunden wird (Dieser Punkt p_0, der den Strahl als Begrenzung seines Voronoigebietes hat, muß laut Vorlesung ein Punkt der konvexen Hülle sein, da das Gebiet unbeschränkt ist.
- Wechsle mit Hilfe des Zeigers auf das benachbarte Gebiet, welches auch unbeschränkt sein muß, da es den gleichen Strahl enthält. Also ist auch der zu diesem Gebiet gehörige Punkt p_1 Element der konvexen Hülle und die Strecke (p_0, p_1) ist eine Kante der konvexen Hülle.
- Durchsuche nun die Kantenliste des neuen Punktes, bis der zweite Strahl gefunden wird, und wechsle über dessen Zeiger zum Nachbar.
- Wiederhole die obigen Schritte, bis man wieder am Punkt p_0 ankommt.

Laufzeit

Das Suchen nach dem Punkt p_0 hat Laufzeit $O(n)$. Das restliche Verknüpfen der Punkte hat ebenfalls $O(n)$, da jede Kante höchstens einmal durchsucht wird (Begründung wie in (i)).

Aufgabe 2

Tut mir leid, aber ich kann mit der Aufgabenstellung nicht sehr viel anfangen.