

Sweepline Algorithmen - Linienschnitte

Paul Jungeblut

2. Juli 2014

Sweepline - Was ist das?

Sweepline

- ▶ Häufige Methode zum Lösen geometrischer Probleme
- ▶ Gesamte Ebene wird mit einer Linie gescannt (Scanline)
- ▶ Nur an bestimmten, wichtigen Punkten (Events) muss etwas getan werden

Beispiele

- ▶ Graham-Scan, Sweepline scannt um einen Punkt rotierend
- ▶ Closest-Pair, klassisch

Problemstellung

Aufgabe

- ▶ n Strecken in der Ebene, jeweils gegeben durch die beiden Endpunkte
- ▶ **Aufgabe:** Finde alle Schnittpunkte

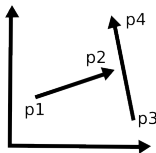
Vereinfachungen

- ▶ keine zwei End-/Schnittpunkte haben die gleiche x-Koordinate
- ▶ kein Endpunkt liegt auf einer anderen Strecke
- ▶ max. 2 Strecken schneiden sich in einem Punkt

Naiver Ansatz

Erinnerung

$$\begin{aligned} \text{Schnitt}(p_1, p_2, p_3, p_4) = \\ ccw(p_1, p_2, p_3) \cdot ccw(p_1, p_2, p_4) \leq 0 \wedge \\ ccw(p_3, p_4, p_1) \cdot ccw(p_3, p_4, p_2) \leq 0 \end{aligned}$$



Algorithmus

- ▶ Teste für je zwei Strecken, ob sie sich schneiden
- ▶ Berechne Schnittpunkt (LGS)
- ▶ Laufzeit: $O(n^2)$

Bentley–Ottmann Algorithmus

Idee

- ▶ Lasse Sweepline L von links nach rechts über die Ebene laufen.
- ▶ Zu jedem Zeitpunkt schneidet S eine Teilmenge der Strecken. Die vertikale Anordnung verändert sich dabei nur bei einem Schnittpunkt.
- ▶ Events sind
 - ▶ noch nicht gescannte Endpunkte
 - ▶ Schnittpunkte von Strecken, die in der vertikalen Anordnung nebeneinander liegen

Bentley–Ottmann Algorithmus

Algorithmus - Initialisierung

1. Erstelle Priority Queue pq für zukünftige Events, priorisiert nach x-Koordinate. pq enthält zu Beginn alle Endpunkte.
2. Erstelle Set T für vertikale Anordnung der Schnittpunkte zwischen den Strecken und der Sweepline. Sortierung nach y-Koordinate. Zu Beginn leer.

Bentley–Ottmann Algorithmus

Algorithmus - Fortsetzung

3. Solange pq nicht leer ist, entferne erstes Element aus pq .
 - ▶ **Linker Endpunkt einer Strecke s :** Füge s in T ein. Suche Strecken r und t direkt über und unter s (falls sie existieren). Falls ihr Schnittpunkt als Event in pq liegt, entferne ihn. Falls s die Strecken r oder t schneidet, füge die Schnittpunkte in pq ein.
 - ▶ **Rechter Endpunkt einer Strecke s :** Suche Strecken r und t direkt über und unter s . Falls sie sich noch schneiden, füge Schnittpunkt zu pq hinzu. Entferne s aus T .
 - ▶ **Schnittpunkt zweier Strecken s, t :** Tausche Positionen von s und t in T . Finde Strecken o und u darüber und darunter. Entferne Schnittpunkte mit diesen, füge neue ein.