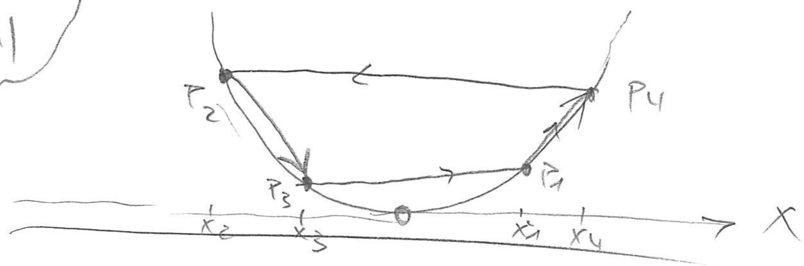
AlgGeo 4.1

Thm 4.7 Konvexe Hülle von Menge S von n Punkten im $\mathbb{R}^2$ läßt sich in Zeit $O(n \log n)$ — nach Sortieren sogar linear.

Lemma 4.8 Das ist optimal.

Bew Reduktion von Sortieren: Gegeben: $x_1, \dots, x_n \in \mathbb{R}$
Aufgabe: nach Größe anordnen

$x_i \longmapsto p_i := (x_i, x_i^2)$ in Zeit $O(n)$



$S := \{p_1, \dots, p_n\}$

Sei A Algorithmus zur Konstruktion von $ch(S)$

klar: $p_1, \dots, p_n$ sind Ecken vom $\partial ch(S)$
laufe durch die ausgegebenen Eckpunkte; bestimme den linkesten und den rechtesten, durchlaufe dieses Intervall und berichte die x-Koordinaten der Eckpunkte.
$\rightarrow$ die x_i sind sortiert! $\square$

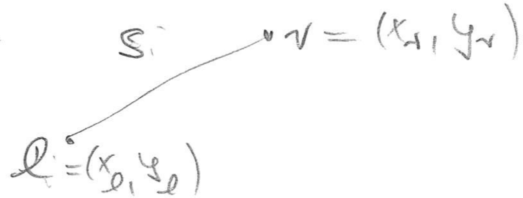
Beobachtung: (i) $ch(S) \overset{O(n)}{\longleftrightarrow}$ Sortieren ($\rightarrow$ später!)

(ii) Auch wenn die Ecken von $ch(S)$ lediglich unsortiert berichtet werden sollen, braucht man $\Omega(n \log n)$ Zeit!

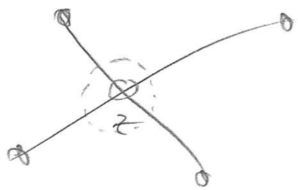
AlgGeo 4.2

2.3.2 Schnittpunkte von Liniensegmenten

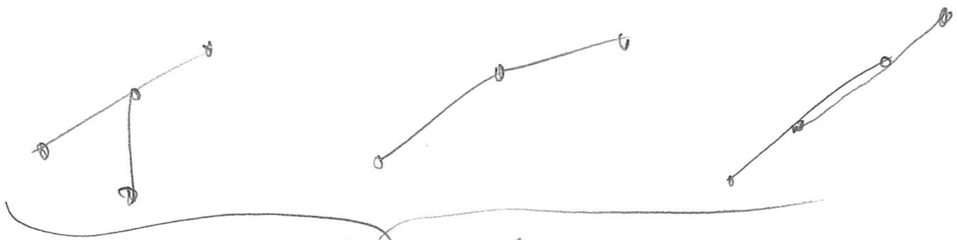
Gegeben: n Liniensegmente s_i im $\mathbb{R}^2$;



Existenzproblem: Gibt es einen echten Schnittpunkt zwischen zwei Segmenten?

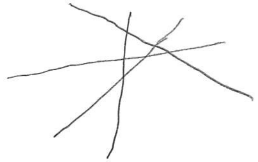


echt: z liegt im Inneren beider Segmente



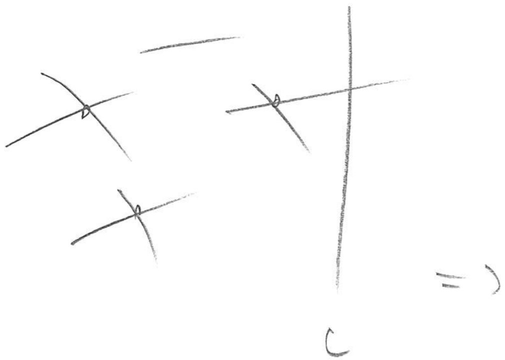
nicht echt

Aufzählungsproblem: Berichte alle k echten Schnittpunkte zwischen den Segmenten. Klar: $0 \leq k$



$\binom{n}{2} = k$

Sweep:



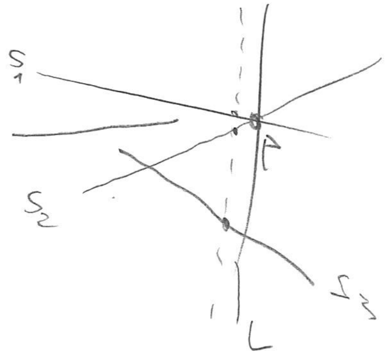
$\Rightarrow$

Verallgemeinerung vom "Kochrezept" zur Bestimmung einer Eigenschaft E von Objekten: Führe Sweep von links nach rechts durch, unterhalte geeignete Eigenschaft E', aus der sich am Ende E ergibt.

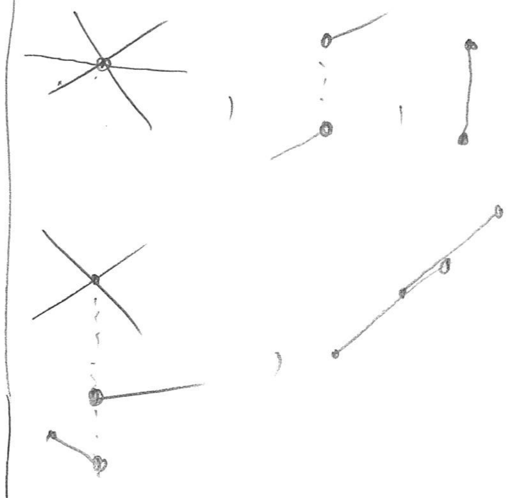
AlgGeo 4.3

Beobachtung:

Lemma 2.4 Wenn zwei Segmente s_1, s_2 echten Schnittpunkt p haben, sind sie in der Y-Ordnung längs der Sweepline L direkt benachbart, kurz bevor L den Punkt p erreicht.

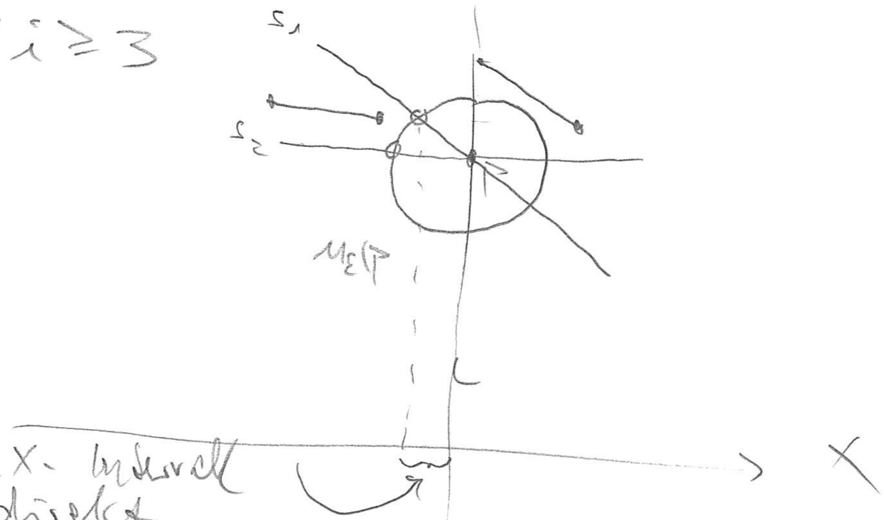


Annahme über allgemeine Lage: nicht



Bew: p kommt nur in s_1, s_2 vor $\Rightarrow \exists\, \varepsilon > 0:$

$U_\varepsilon(p) \cap s_i = \emptyset \quad \forall i \geq 3$



in diesem X-Intervall sind s_1, s_2 direkt benachbart

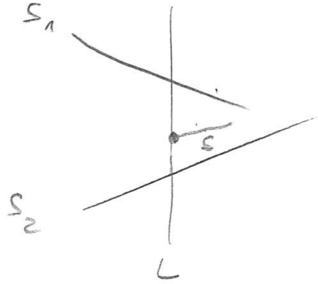
$\square$

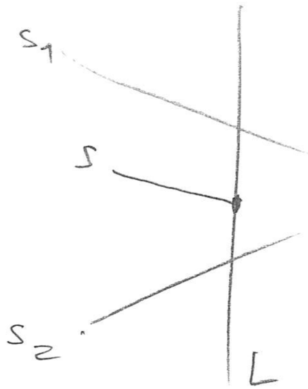
Idee: Unterhalte die von Sweepline L aktuell geschnittenen Segmente $\}\ E'$ in Y-Reihenfolge

teste direkte Nachbarn auf Schnitt

AlgGeo 4.4

Welche Events können Eigenschaft E' stören, d.h. die y-Reihenfolge der von L geschnittenen Segmente verändern?





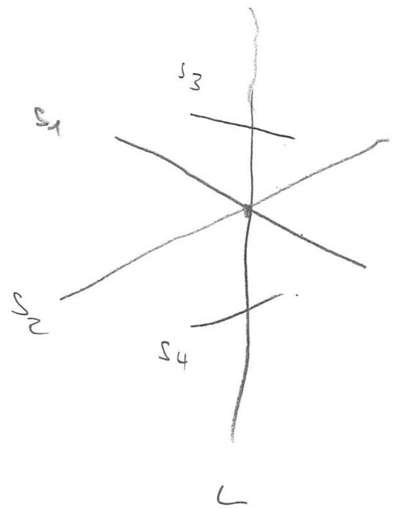
SSS: Sweep-Status-Struktur:
hier: die nach y sortierte Liste der von L geschnittenen Segmente

Event-Handling:

füge s in SSS ein
teste s_1, s und s, s_2 auf Schnitt
falls p gefunden:
Existenzproblem: "ja", Stop
Aufzählungsproblem:
berichte Schnittpunkt p
füge Schnittevent zur Zeit x_p in ES ein

entferne s aus SSS
teste s_1, s_2 auf Schnitt
⋮

ES: Event-Struktur
die nach x sortierten schon bekannten Event

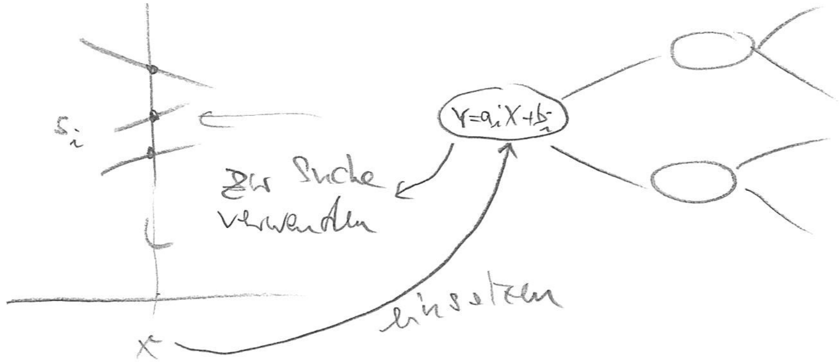


vertausche s_1, s_2 in SSS
teste s_3, s_2 und s_1, s_4 auf Schnitt
⋮

AlgGeo 4.5

Frage: Welche Operationen werden auf SSS und ES benötigt?
Mit welchen Datenstrukturen sollte man die Strukturen realisieren?

SSS: Einfügen, Entfernen, (Suchen) Nachbarn bestimmen: balancierte Bäume z.B. AVL



Platz: $O(n)$

Kosten pro Op: $O(\log n)$

ES: getmin, Einfügen, (Entfernen an bekannter Stelle) später

Priority Queue

hier: auch AVL-Baum

Platz: $O(n+k)$ $k \le n^2$

Kosten pro Op: $O(\log n)$

Oben skizzierte Prozeduren zum Event-Handling einbetten:

Initialisiere SSS $(= \emptyset)$
ES (sortierte Endpunkte der Segmente, nach x)
while ES $\neq \emptyset$
get next event e
handle e.

AlgGeo 4.6

Laufzeit? Speicherplatz?

↳ Faustformel: $\underbrace{\#\text{Events}}_{O(n+k)} \cdot \underbrace{\text{Kosten pro Event}}_{O(\log n)}$

Theorem 2.8 Bei n Liniensegmenten im $\mathbb{R}^2$ kann man in Zeit $O(n \log n)$ auf Existenz eines echten Schnittpunkts testen (und Platz $O(n)$). Das ist optimal. ✓

Theorem 2.9 Die k (echten) Schnittpunkte von n Liniensegmenten kann man in Zeit $O((n+k)\log n)$ berichten, und Platz $O(n+k)$?

optimal für $k \in O(n)$. Aber: $k \approx n^2$: $O(n^2 \log n)$ schlechter als $O(n^2)$ naiv

Speicherplatzbedarf der ES: $\leq 2n$ + Anzahl der schon entdeckten zukünftigen Schnittpunkte

Pach/Sharir: $O(n (\log n)^2)$

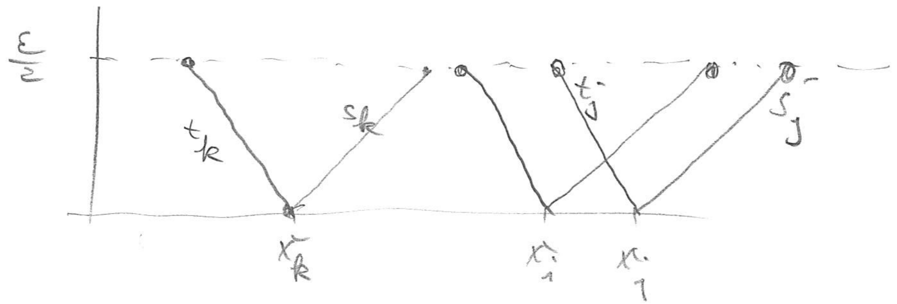
Buch Ü 2.8 : $\Omega(n \log n)$

AlgGeo 4.7

Optimalität beim Existenzproblem:

(1) Reduktion von ε-closeness: Gegeben: $x_1, \dots, x_n \in \mathbb{R}$, $\varepsilon > 0$

Frage: $\exists ?\ i \neq j : |x_i - x_j| < \varepsilon$



$x_i \longmapsto$ Segmente s_i, t_i

$s_i : (x_i, 0)\ (x_i + \frac{\varepsilon}{2}, \frac{\varepsilon}{2})$

$t_i : (x_i, 0)\ (x_i - \frac{\varepsilon}{2}, \frac{\varepsilon}{2})$

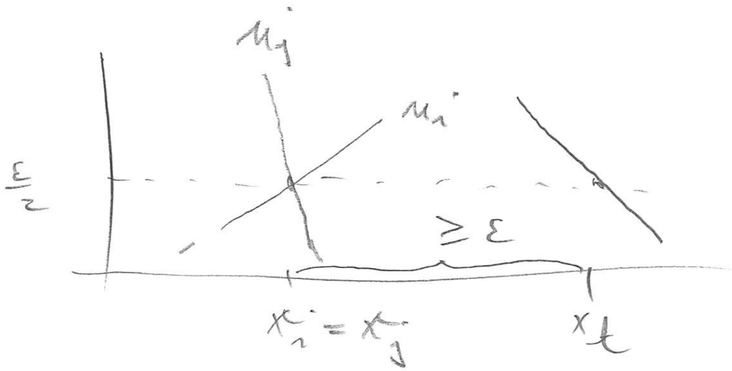
Klar: Es gibt echten Schnittpunkt $\Longleftrightarrow$ $\exists\ i \neq j : |x_i - x_j| < \varepsilon$ und $x_i \neq x_j$

(falls $x_i = x_j$: $s_i = s_j$ und $t_i = t_j$: kein echter Schnittpunkt)

(2) Falls (1) ergebnislos: Konstruiere zu x_i ein Segment u_i mit Mittelpunkt in $(x_i, \frac{\varepsilon}{2})$

Länge ε

Winkel $\frac{i}{n} \frac{\pi}{2}$

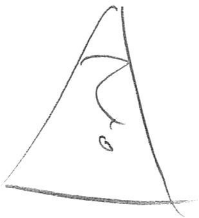


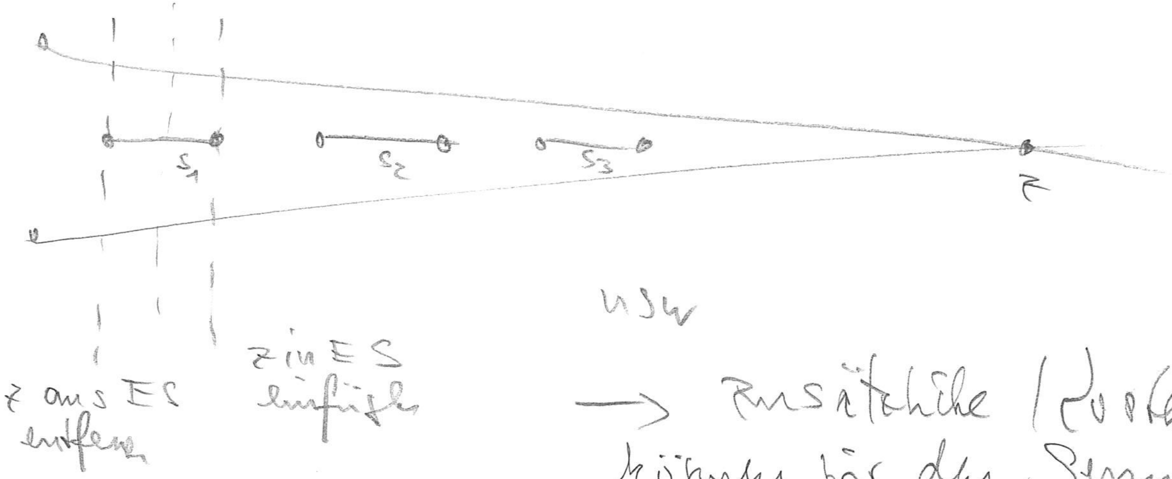
Dies existiert echter Schnittpunkt $\Longleftrightarrow$ $\exists\ i \neq j : x_i = x_j$. $\square$

AlgGeo 4.8

Speicherplatzbedarf der ES auf $O(n)$ reduzieren:

<u>Idee</u>: Speichere in ES nur die erkannten Schnittpunkte direkt längs L benachbarter Segmente: $O(n)$

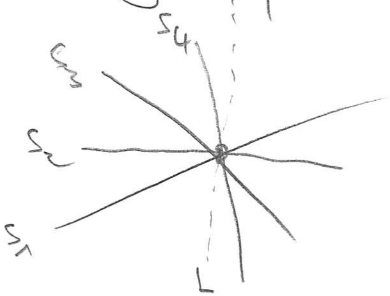




usw

→ zusätzliche Kosten können wir den Segmenten s_i in Rechnung stellen.

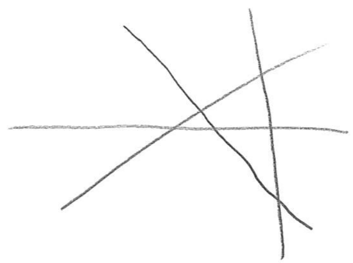
Beliebige Inputs:



n Segmente

:

Perturbationen



n^2 Schnittpunkte

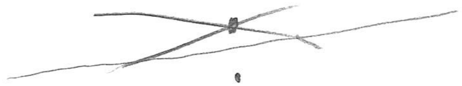
AlgGeo 4.9

besser: Schnittereignisse in ES lexikographisch ordnen

$$\ldots \underbrace{(s_1, s_2), (s_2, s_3), (s_3, s_4)}_{\text{gleichzeitig}} \ldots$$

en bloc in SSS umdrehen: $O(m)$.

1991: P. Schorn: Numerisch stabile Implementierung für beliebige Inputs: Dissertation



Forschungsgegenstand

Algorithmen-Datenbank: LEDA

CGAL