

# Zusammenfassung Sweep Liniensegmente

Elmar Langetepe  
University of Bonn

# SSS: Schnittpunkte von Liniensegmenten

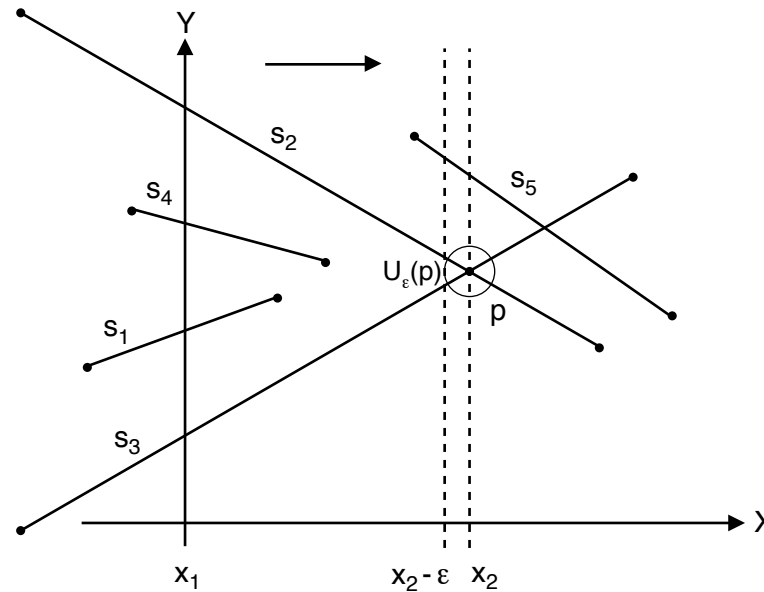
- $n$  Liniensegmente
- ● Schnittpunkte berichten
- Existenz-/Aufzählungsproblem
- Nicht-degenerierte Lage



Theorem 2.8 Man kann in Zeit  $O(n \log n)$  und mit Speicherplatz  $O(n)$  ermitteln, ob von  $n$  Liniensegmenten in der Ebene mindestens zwei einen echten Schnittpunkt haben.■

Theorem 2.10 Das Sweep-Verfahren berechnet die  $k$  echten Schnittpunkte von  $n$  Liniensegmenten in Zeit  $O((n + k) \log n)$  mit Platz  $O(n)$ .■

# SSS: Ordnung der Liniensegmente

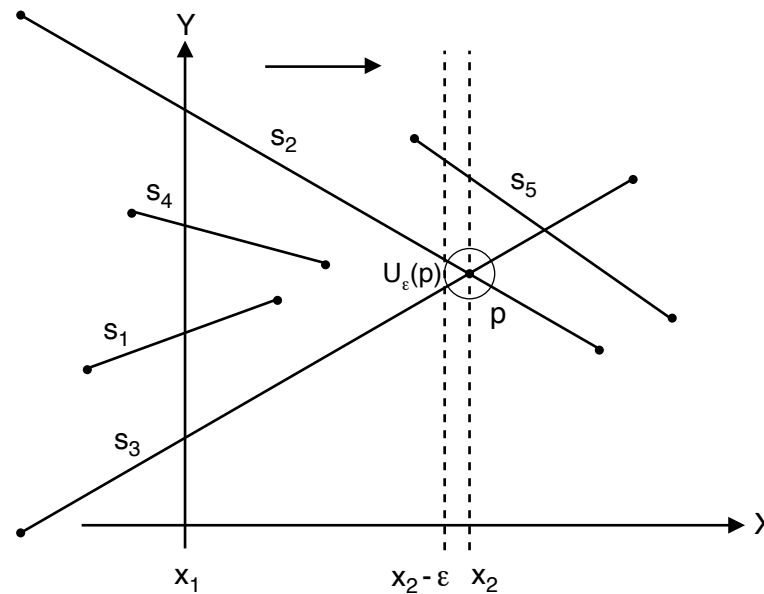


1. Neuer linker Endpunkt!
2. Neuer rechter Endpunkt!
3. Schnittpunkt

SSS: Balancierter Binärbaum mit dynamischen Schlüsseln  
(Geradengleichung!)

# SSS: Strukturelle Eigenschaft

Lemma 2.7 Haben zwei Liniensegmente einen echten Schnittpunkt zum Zeitpunkt  $t$ , dann sind die Segmente unmittelbar vor  $t$  Nachbarn in der SSS gewesen.



# Ereignisstruktur, Initialisierung

- ES: Priorityqueue, SSS balancierter Binärbaum mit Segmentschlüsseln
- Ereignisse: Schnittpunkt, Linker Endpunkt, Rechter Endpunkt
- SSS: Einfügen/Entfernen/Ordnung wechseln, Schnitttest mit Nachbarn, Einfügen Schnittereignis

(\* Initialisierung SSS und ES\*)

Initialisiere die Strukturen SSS und ES;

sortiere die  $2n$  Endpunkte nach aufsteigenden  $X$ -Koordinaten;

erzeuge daraus Ereignisse;

füge diese Ereignisse in die ES ein;

## Bearbeitung Ereignisse

(\* Sweep und Ausgabe \*)

**while**  $ES \neq \emptyset$  **do**

*Ereignis* := *NächstesEreignis*(*ES*);

**with** *Ereignis* **do**

**case** *Typ* **of**

LinkerEndpunkt:

*FügeEin*(*SSS*, *Seg*, *Zeit*);

*VSeg* := *Vorg*(*SSS*, *Seg*, *Zeit*);

*TesteSchnittErzeugeEreignis*(*VSeg*, *Seg*);

*NSeg* := *Nachf*(*SSS*, *Seg*, *Zeit*);

*TesteSchnittErzeugeEreignis*(*Seg*, *NSeg*);

RechterEndpunkt:

$VSeg := Vorg(SSS, Seg, Zeit);$

$NSeg := Nachf(SSS, Seg, Zeit);$

$Entferne(SSS, Seg, Zeit);$

$TesteSchnittErzeugeEreignis(VSeg, NSeg);$

Schnittpunkt:

Berichte  $(USeg, OSeg)$  als Paar mit Schnitt;

$Vertausche(SSS, USeg, OSeg, Zeit);$

$VSeg := Vorg(SSS, OSeg, Zeit);$

$TesteSchnittErzeugeEreignis(VSeg, OSeg);$

$NSeg := Nachf(SSS, USeg, Zeit);$

$TesteSchnittErzeugeEreignis(USeg, NSeg)$

# Analyse und Erweiterungen

- $O((n + k) \log n)$  Schritte
- Platzsparregel: Entlang jedes Segmentes nur einen Schnittpunkt
- ES hat Komplexität  $O(n)$  statt  $O(n + k)$ !
- Aufheben der Degeneriertheit, z.B. gleiche  $X$ -Koordinaten



# Degenerierte Fälle

- Intrinsisch: Im Algorithmus selbst!
  - Extrinsisch: Vorab den Input verändern!
1. Unterschiedliche  $X$ -Koordinaten erzwingen (ext.)
  2. Reihenfolge der Bearbeitung festlegen (int.)
  3. Gemeinsame Schnittpunkte entzerren (int.)

# Buchkapitel, Seiten

Kapitel 2.2.3



Seite 64 – 75 unten