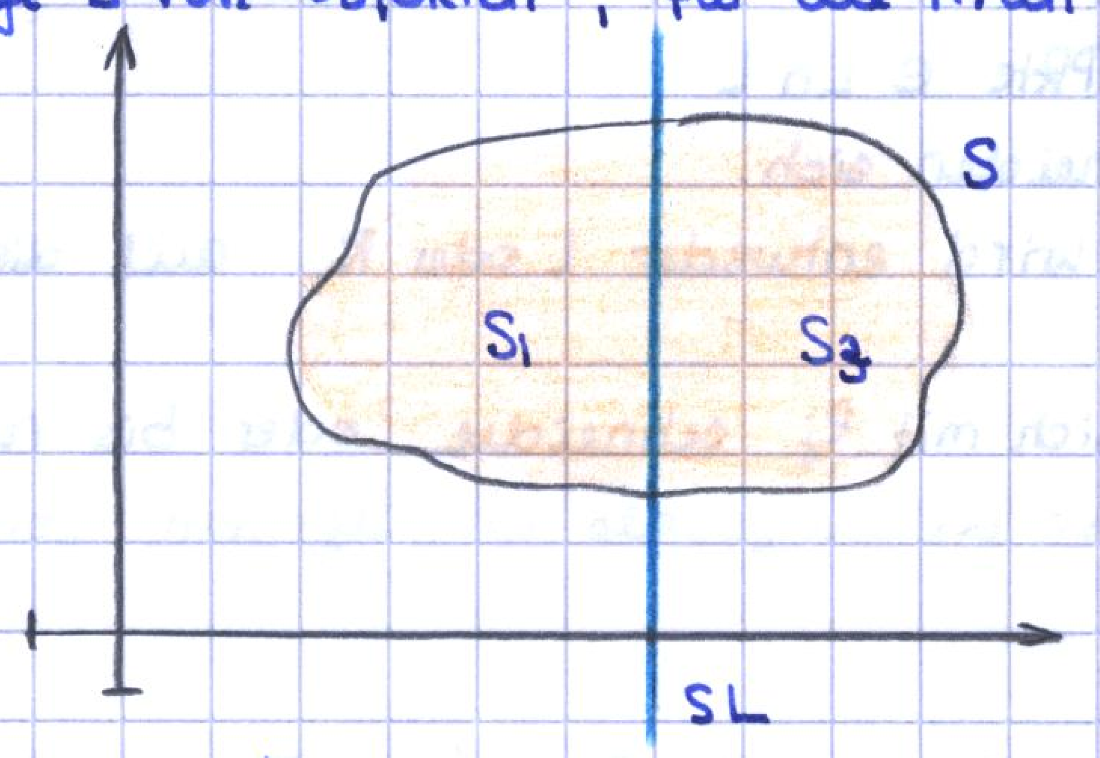
Kapitel III: Das Plane Sweep Verfahren.

3.1. Einführung:

3.1.1. Idee:

Allg. Ansatz zu Lsg. geom. Probleme in der Ebene, die inkrementell durch schrittweises Betrachten der Eingabeobjekte in einer bestimmten Reihenfolge gelöst werden können.
dh. meistens von links nach rechst (xy-Ordnung).
Genauer: Man schiebt / bewegt eine sog. Sweepline, also eine senkrechte Gerade SL über die Menge S von Objekten, für die man ein Problem lösen möchte.



Im Allg. zerlegen wir S in drei Mengen:
- S_1 links von SL
  (Teilproblem für S ist gelöst).
- S_3 rechts von SL
  Objekte, die wir noch nicht kennen.
- $S_2 = S \cap SL$
  alle Objekte die von SL geschnitten werden.

3.1.2 Bem: S_2 stellt den Teil der Eingabe dar, der noch relevant ist zur Berechnung restlicher Lsg (für S_3)
S_2 ist eine dynamische 1-dim. Menge.
⇒ Reduzierung eines statischen 2-dim. Problems auf ein dynamisches 1-dim. Problem.
(die dynam. 1-dim. Probleme werden balancierte Suchbäume sein.).

3.1.3. Bem:
Wir haben spezielle Varianten des SL-Verfahrens kennengelernt:
- inkrementelle konv. Hülle
  1.3.8. Grahams Scan für beide Hüllen gleichzeitig
- Triangulierung.
  Übung.

3.2. Line Segment Intersection.

Schnitt von Geradensegmenten / Strecken.

3.2.1. Problem:
Geg: Menge S von n Segmenten
Ges: alle Schnittpkte.
(Bzw. Unterteilung der Ebene in Graph)
Grundoperation: $s_i \cap s_j \;\forall s_i, s_j \in S$
(dies geht in konst. Zeit $O(1)$ mit orientation-Test).

3.2.2. Triviale Lsg:
Teste alle Paare nacheinander
⇒ Laufzeit $O(n^2)$.

3.2.3. Ziel: outputsensitiver Algorithmus, dh. Laufzeit hängt von Zahl der Schnittpkte ab.

3.2.4. Idee für einen Plane Sweep Algorithmus: