10. Liniensegmente und Sichtbarkeit

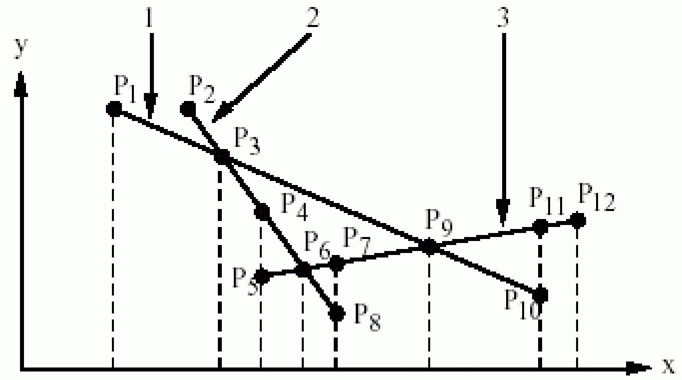
dieses Kapitel: Beispiel für einen einfachen Algorithmus mit komplexer Analyse

vgl. Hinrichs 2001

- Sichtbarkeitsproblem in der Computergraphik:
  Gegeben eine Konfiguration von Objekten im dreidimensionalen Raum und einen Betrachterstandpunkt, was ist sichtbar?
- Ziel: Vermitteln eines Eindrucks der Komplexität des Sichtbarkeitsproblems
- Problem:
  Gegeben n Liniensegmente in der Ebene, bestimme Folge der Teilsegmente, die für einen Betrachter bei $y = -\infty$ sichtbar ist.

also "nur" 2D-Variante des Problems !

Beispiel:
Gegeben sind die Endpunkte (P_1, P_{10}), (P_2, P_8), (P_5, P_{12}) der Liniensegmente 1, 2 und 3

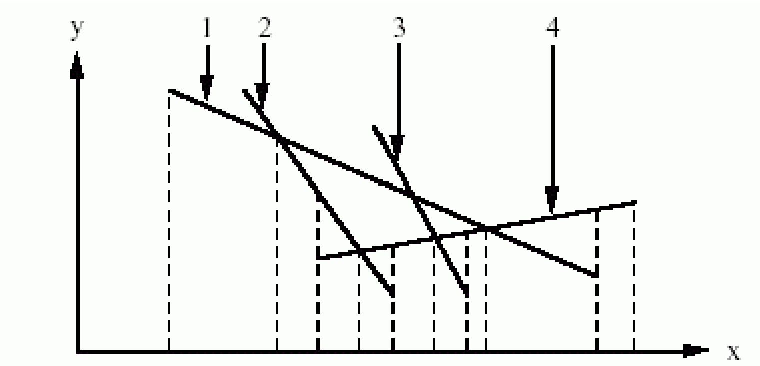


Gesucht ist die Liste der sichtbaren Teilsegmente, wobei jedes Teilsegment beschrieben wird durch ihre Endpunkte und die Nummer des zugehörigen Liniensegmentes:
(P_1, P_3, 1), (P_3, P_4, 2), (P_5, P_6, 3), (P_6, P_8, 2), (P_7, P_9, 3), (P_9, P_{10}, 1), (P_{11}, P_{12}, 3)

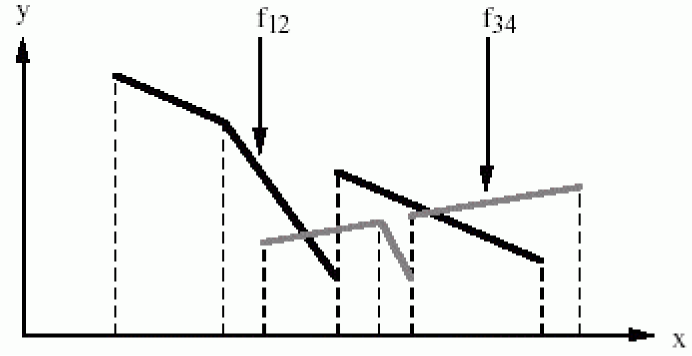
- 1. Idee:
  Für jedes der n^2 geordneten Paare (L_i, L_j) von Liniensegmenten entferne aus L_i das Teilsegment, das durch L_j verdeckt wird.
  Verwalte für jedes L_i eine Folge der sichtbaren Teilsegmente (höchstens n). Das Auffinden der Endpunkte von L_j in dieser liste benötigt O(log n) Zeit.
  ⇒ Gesamtzeitaufwand: $O(n^2 \cdot \log n)$

- 2. Idee:
  Direkte Anwendung von Plane-Sweep, ähnlich wie beim Liniensegmentschnitt, wobei ein sichtbares Teilsegment festgelegt wird durch das bzgl. der $<_L$ Ordnung kleinste Segment.
  ⇒ Gesamtzeitaufwand: $O((n+s)\cdot \log n)$, wobei s die Gesamtanzahl der Schnittpunkte der Liniensegmente bezeichnet.

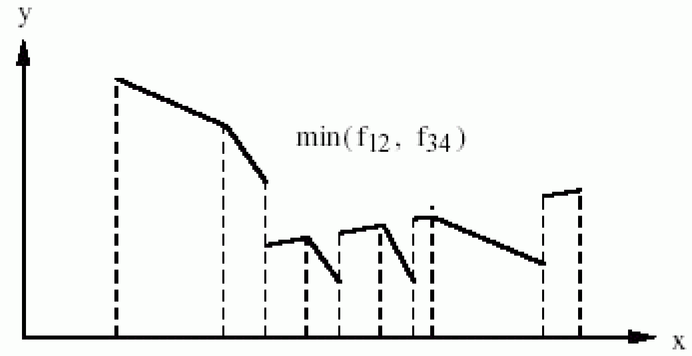
- 3. Idee:
  Divide & Conquer liefert einen effizienteren Algorithmus
  Für n = 0 oder n = 1 ist die Lösung trivial.
  Für n > 1 wird die Menge der Segmente in zwei etwa gleich große Teilmengen (beliebig) aufgeteilt, das Sichtbarkeitsproblem für beide Teilmengen rekursiv gelöst und die beiden Lösungen zu einer Lösung des Gesamtproblems zusammengefügt:



Aufteilung in Teilmengen {1, 2} und {3, 4} und Lösung des Problems für beide Teilmengen:



Zusammenfügen (*Merge*) der beiden Teillösungen zu einer Lösung des Gesamtproblems durch Berechnung des Minimums zweiter stückweise linearen, nicht notwendigerweise stetigen Funktionen:



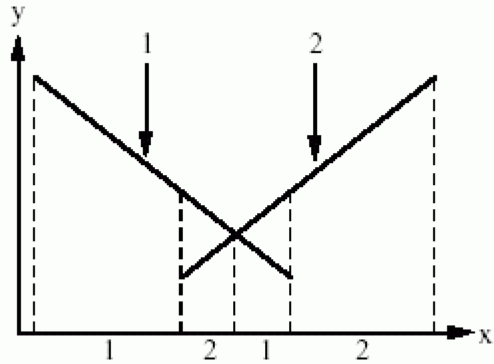
Zeitaufwand O(v) für das Zusammenfügen, wenn v die Anzahl der Teilsegmente bezeichnet.

Zeitkomplexität des Divide & Conquer Algorithmus:
Tiefe der Rekursion: O(log n)
Alle Merge-Schritte auf einer bestimmten Rekursionsstufe benötigen O(V) Zeitaufwand, wenn V die Gesamtanzahl aller sichtbaren Teilsegmente auf dieser Stufe bezeichnet.
⇒ Gesamtzeitaufwand: O(V·log n)

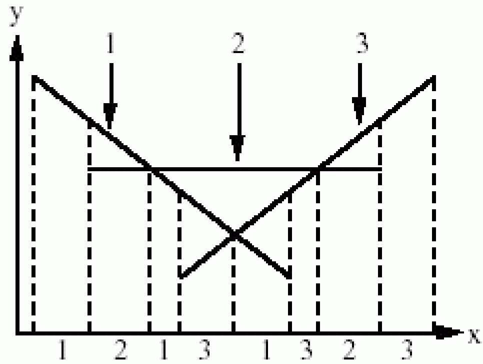
Wie groß kann V werden?

Sei V(n) die Anzahl der sichtbaren Teilsegmente in einer gegebenen Konfiguration von n Liniensegmenten, d.h. die Größe der Ausgabe bei der Sichtbarkeitsberechnung.

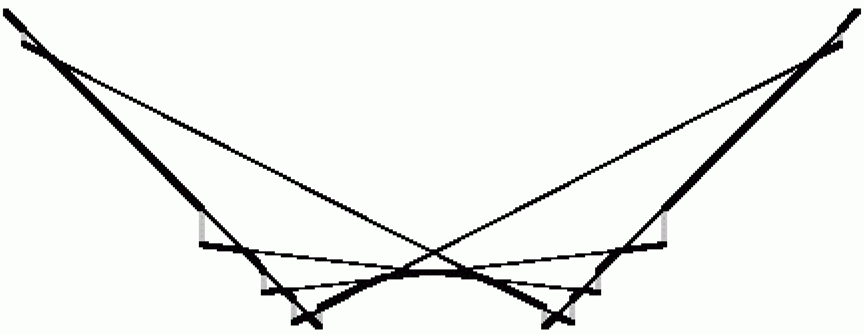
$V(2) = 4$



$V(3) = 8$

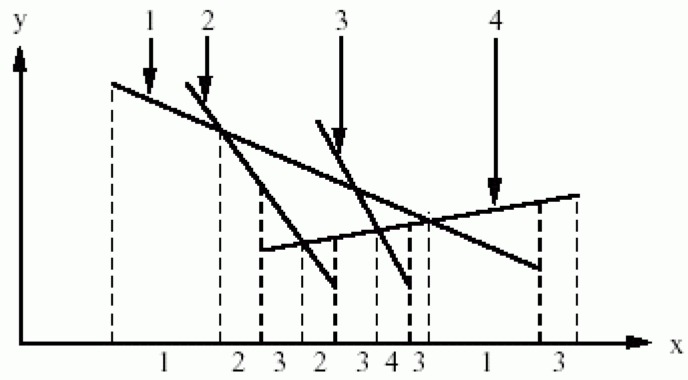


Vermutung: $V(n) = 5 \cdot n - 8$



$V(n) \in O(n)$ gilt nicht!!!

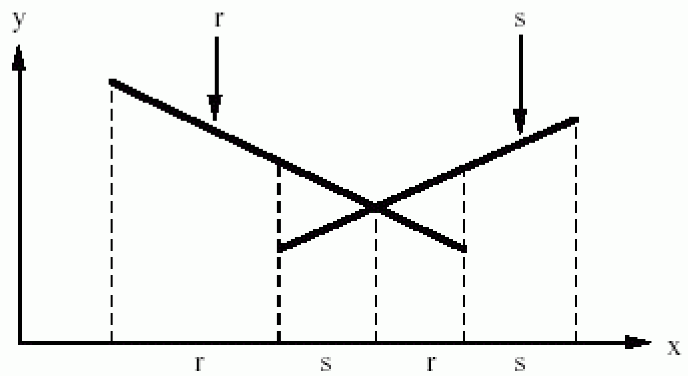
Transformation des Sichtbarkeitsproblems in ein kombinatorisches Zeichenreihenproblem:
Numeriere die gegebenen Liniensegmente von 1 bis n durch. Läuft man entlang der x-Achse vom am weitesten links liegenden Endpunkt eines Segments zum am weitesten rechts liegenden Endpunkt und notiert das jeweils sichtbare Segment, so erhält man eine Zahlenfolge:



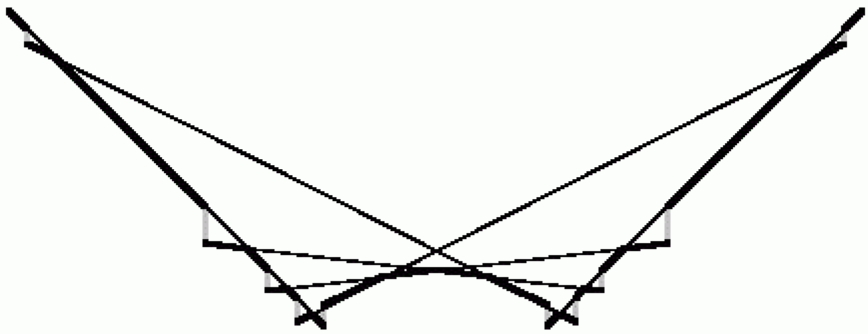
Eine Konfiguration von n Liniensegmenten liefert eine Zahlenfolge $u_1, u_2, \ldots , u_m$ mit den folgenden Eigenschaften:

1. $1 \leq u_i \leq n$ for $1 \leq i \leq m$
2. $u_i \neq u_{i+1}$ for $1 \leq i \leq m-1$
3. Es gibt keine 5 Indizes $1 \leq a < b < c < d < e \leq m$, so daß $u_a = u_c = u_e = r$ und $u_b = u_d = s$, $r \neq s$.

Eigenschaft 3 entspricht den geometrischen Eigenschaften zweier sich schneidender Liniensegmente: Sieht man r, s, r, s (möglicherweise separiert durch andere Segmente), so kann man r nicht wiedersehen, da sich sonst r und s mehr als einmal schneiden müßten:



Beispiel:



zugehörige Folge

1, 2, 1, 3, 1, … , 1, n–1, 1, n–1, n–2, n–3, … , 3, 2, n, 2, n, 3, n, … , n, n–2, n, n–1, n

Folgen mit den Eigenschaften 1 - 3 heißen *Davenport-Schinzel* Folgen und wurden intensiv im Zusammenhang mit linearen Differentialgleichungen studiert.

Hart und Sharir haben gezeigt, daß die maximale Länge einer DavenportSchinzel Folge über n Buchstaben gerade $k \cdot n \cdot \alpha(n)$ ist, wobei k eine Konstante und $\alpha(n)$ die Inverse der Ackermannschen Funktion A(n) ist:

$A(n) = A_n(n)$, wobei

$A_k(1) = 2$ für $k \geq 1$

$A_1(n) = A_1(n-1) + 2$ für $n \geq 2$

$A_k(n) = A_{k-1}(A_k(n-1))$ für $k \geq 2$

$\alpha(n)$ wird definiert durch $\alpha(n) = \min\{m: A(m) \geq n\}$.

$\alpha(n) \leq 3$ für $n \leq 16$

$\alpha(n) \leq 4$ für $n \leq \underbrace{2^{2^{2^{\cdots^{2}}}}}_{65536-\text{mal}}$ n

$\lim_{n \to \infty} \alpha(n) = \infty$

Wiernik und Sharir haben gezeigt, daß es für jede Davenport-Schinzel Folge eine entsprechende Konfiguration von Liniensegmenten gibt, die dieser Folge entspricht.

$\Rightarrow$ 1-1-Beziehung zwischen Davenport-Schinzel Folgen und dem 2-dimensionalen Sichtbarkeitsproblem
Größe der Ausgabe kann superlinear sein

$\Rightarrow$ $V(n) \in \Theta(n \cdot \alpha(n))$
$\Rightarrow$ Gesamtzeitaufwand des Divide & Conquer Algorithmus: $O(\alpha(n) \cdot n \cdot \log n)$

Referenzen:

S. Hart, M. Sharir:
Nonlinearity of Davenport-Schinzel sequences and of generalized path compression schemes,
Combinatorica 6, 151 – 177 (1986)

A. Wiernik, M. Sharir:
Planar realizations of nonlinear Davenport-Schinzel sequences by segments,
Discrete and Computational Geometry 3(1), 15 – 47 (1988)