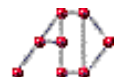


Delaunay Triangulation

1. Motivation
2. Triangulation ebener Punktmenngen
3. Definition und Eigenschaften der Delaunay Triangulation
4. Berechnung der Delaunay Triangulation (randomisiert, inkrementell)
5. Analyse des Platz- und Zeitbedarfs

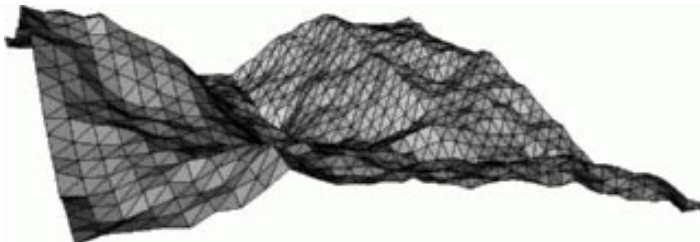


1. Motivation

Umwandlung einer topographischen Karte



in eine perspektivische Sicht



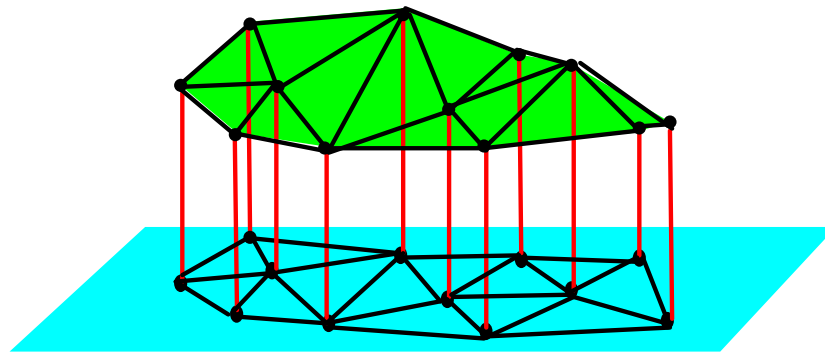
Terrains

Gegeben:

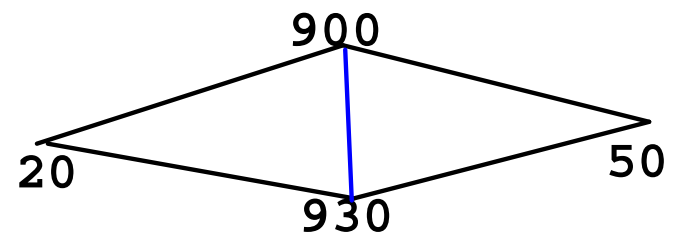
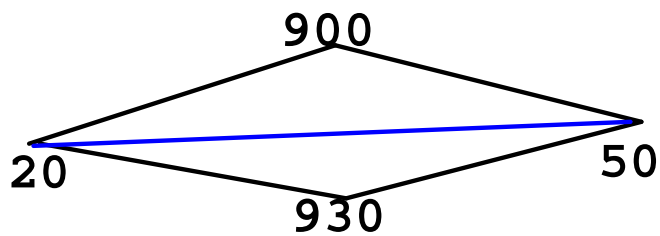
Eine Anzahl von Beispielpunkten p_1, \dots, p_n

Gesucht:

Triangulation T mit Bild von T "realistisch"



Das "Flippen" einer Kante



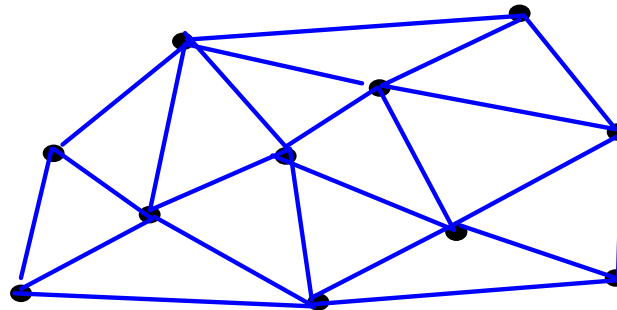
Ziel:

Maximierung des Winkels in der Triangulation

2. Triangulation ebener Punktmengen

Gegeben: Menge P von n Punkten in der Ebene (nicht alle kollinear).

Eine **Triangulation** $T(P)$ ist eine planare Aufteilung der konvexen Hülle von P in Dreiecke mit Eckpunkten aus P .



$T(P)$ ist maximale Unterteilung.

Für eine gegebene Punktmenge gibt es nur endlich viele verschiedene Triangulationen.

Größe von Triangulationen

Satz:

Jede Triangulation einer Punktmenge $P = \{p_1, \dots, p_n\}$ besitzt $2n - 2 + k$ Dreiecke und $3n - 3 - k$ Kanten, $k = \#$ Kanten auf der konvexen Hülle

Größe von Triangulationen

Satz:

Jede Triangulation einer Punktmenge $P = \{p_1, \dots, p_n\}$ besitzt $2n - 2 + k$ Dreiecke und $3n - 3 - k$ Kanten, $k = \#$ Kanten auf der konvexen Hülle

Beweis:

Dreieck 3 Kanten, äußere Fläche k Kanten

$$f = \# \text{Dreiecke} + 1$$

Kante hat jeweils 2 inzidente Flächen \Rightarrow

$$e = (3 \# \text{Dreiecke} + k) / 2$$

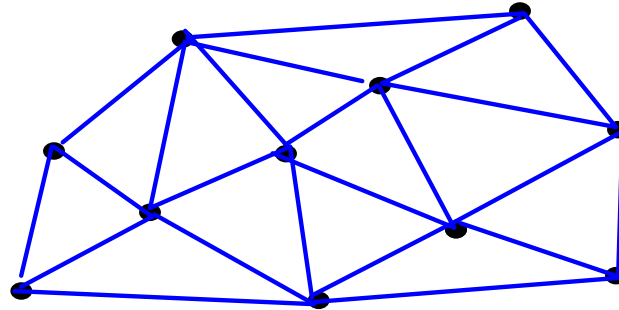
$$\text{Euler: } n - e + f = 2 \quad \Rightarrow$$

$$\# \text{Dreiecke} = 2n - 2 - k$$

und

$$e = 3n - 3 - k$$

Winkelvektor



P Menge von n Punkten

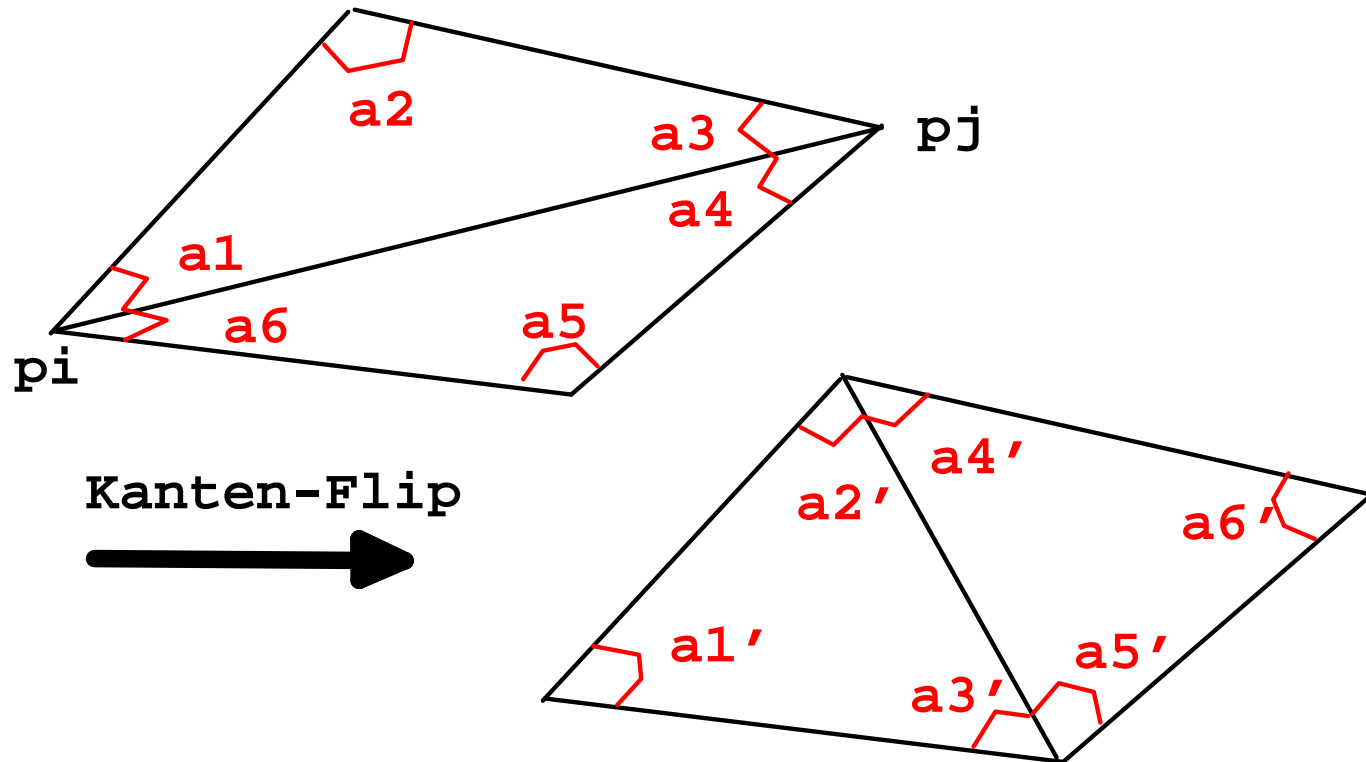
$T(P)$ habe m Dreiecke

$A(T) = \{a_1, \dots, a_{3m}\}$ Vektor der $3m$ Winkel in absteigender Größenanordnung

Triangulationen lassen sich bzgl. $A(T)$ lexikographisch anordnen!

Triangulation $T(P)$ heißt **winkeloptimal**, wenn $A(T(P)) \geq A(T'(P))$ für alle Triangulationen $T'(P)$ gilt.

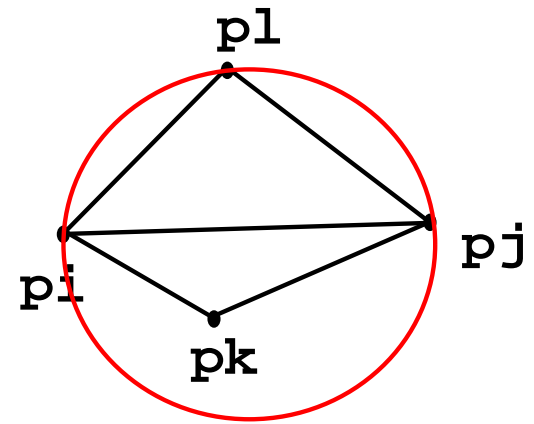
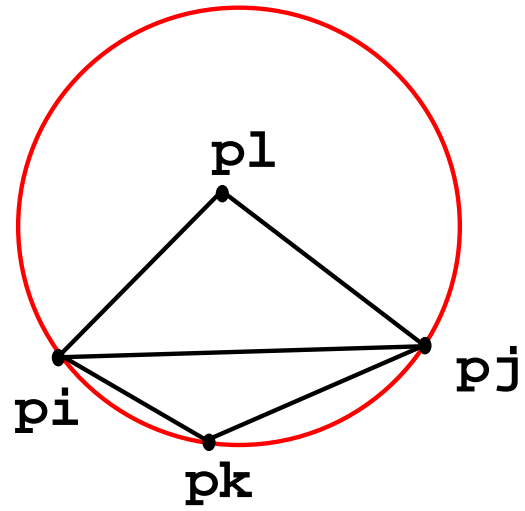
Illegale Kanten



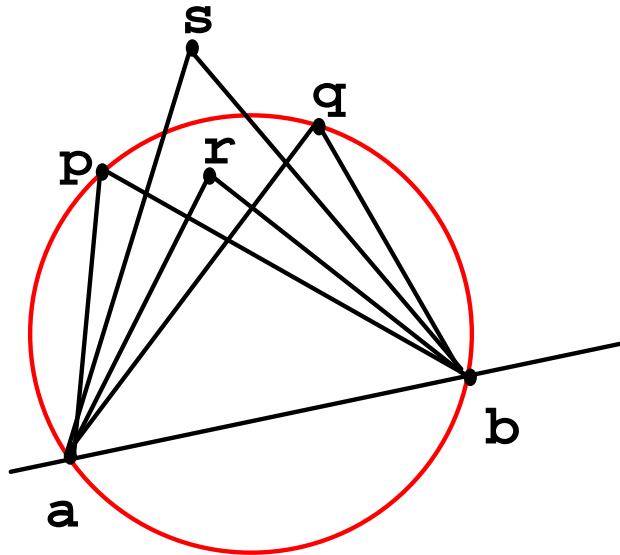
Kante $\overline{p_i p_j}$ illegal \Leftrightarrow
 $\min\{a_i \mid i \text{ aus } 1..6\} < \min\{a_i' \mid i \text{ aus } 1..6\}$

T hat illegale Kante \Rightarrow
Flippen zu T' ergibt $A(T') > A(T)$

Illegalitäts-Test

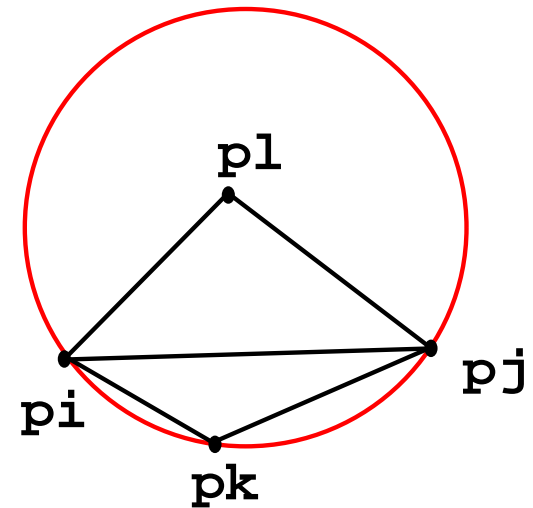


Thales Theorem



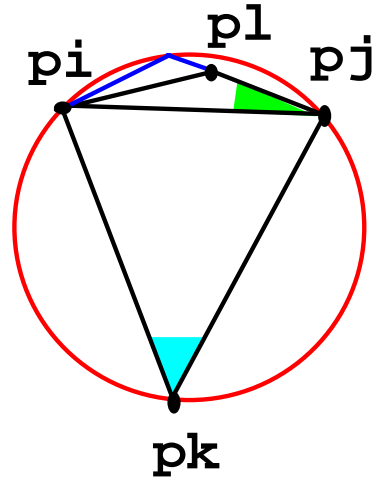
$$\begin{aligned} \text{Winkel}(asb) &< \\ \text{Winkel}(aqb) &= \text{Winkel}(apb) < \\ \text{Winkel}(arb) & \end{aligned}$$

Sei C Kreis um Dreieck p_i, p_j, p_k
und p_l weiterer Punkt.

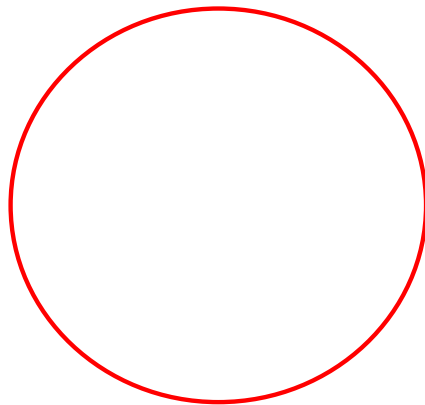


$p_i p_j$ illegal $\Leftrightarrow p_l$ im Inneren von C

Quadrilateral mit
 p_l im Innern des Kreises
durch p_i, p_j, p_k .

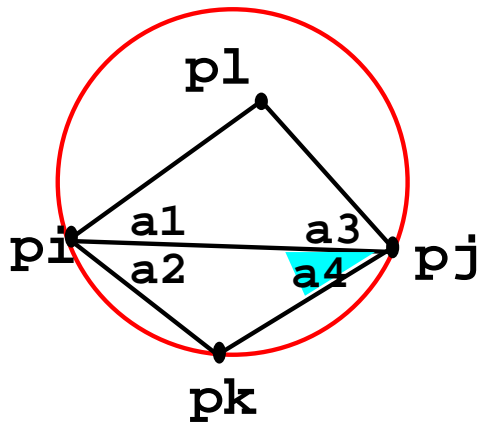
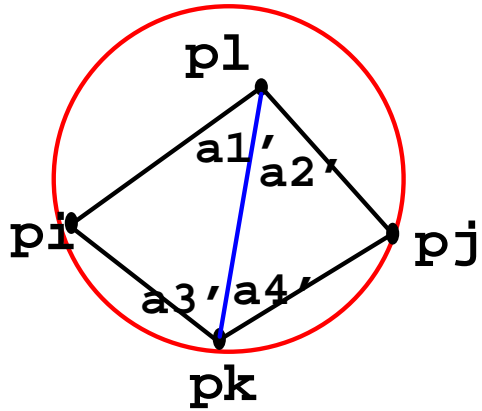
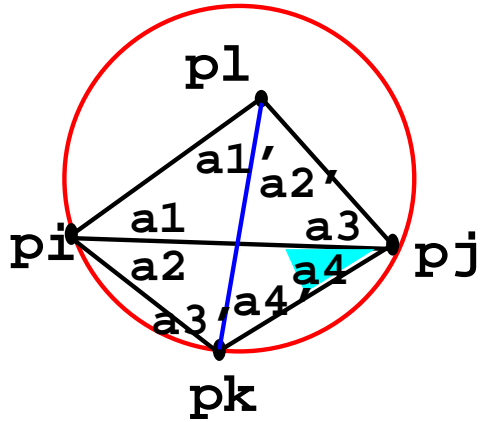


Beh.: Minimaler Winkel
nicht bei p_k !



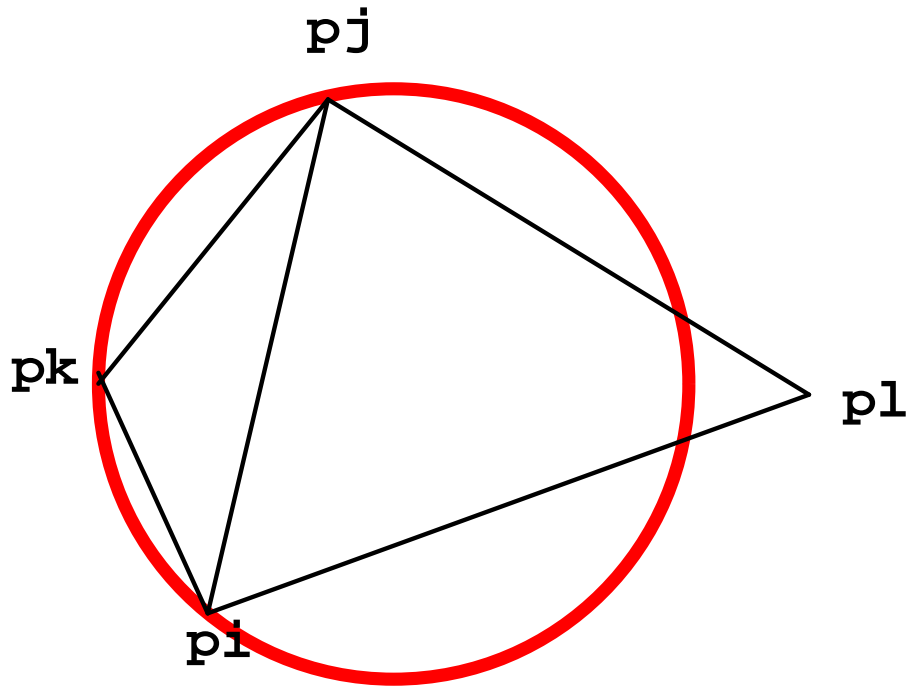
Ziel: $p_i p_j$ illegal

O.E. a_4 minimal.

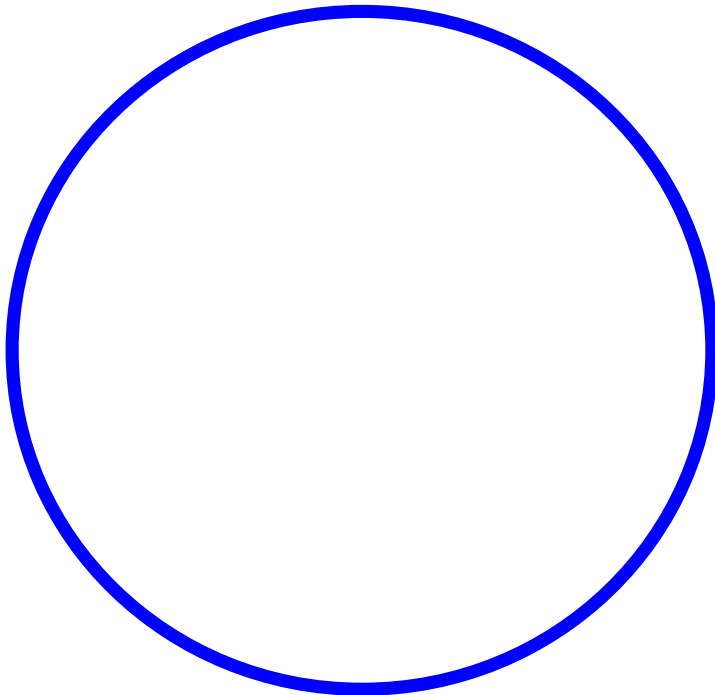


Kreiskriterium verletzt
=> Kante illegal

Kante $\overline{p_i p_j}$ illegal

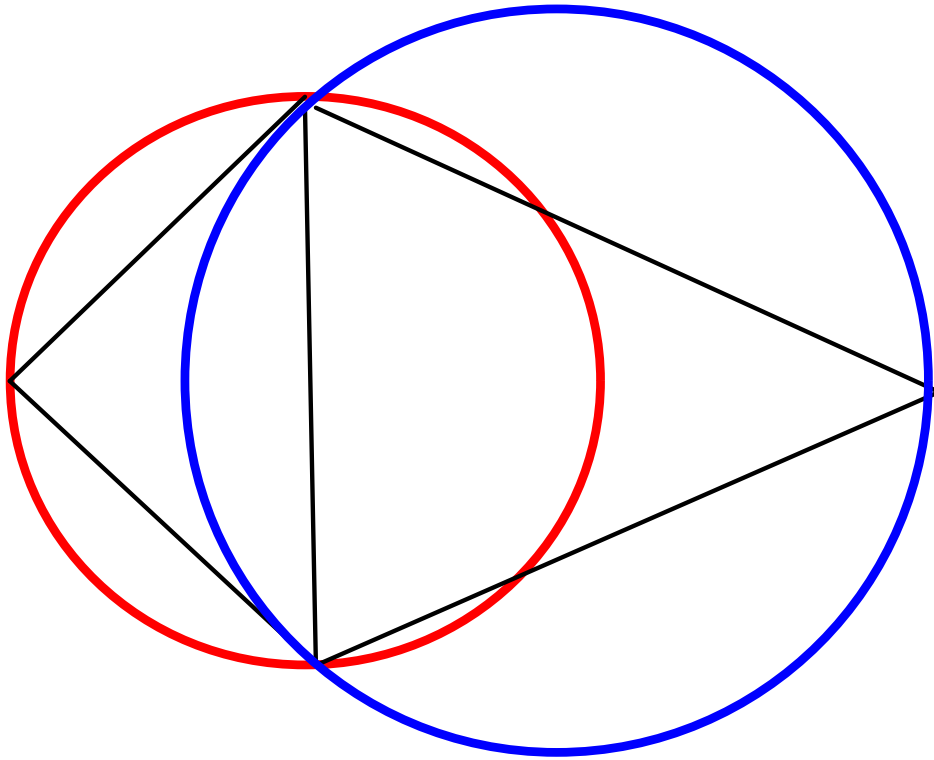


Kante $\overline{p_k p_l}$ illegal



Kreiskriterium
verletzt

Kreiskriterium



Eine Triangulation erfüllt das Kreiskriterium gdw. der Umkreis eines jeden Dreiecks enthält keinen weiteren Punkt im Innern.

Satz

Eine Triangulation $T(P)$ einer Menge von Punkten enthält keine illegale Kante gdw. nirgendwo das Kreiskriterium verletzt ist.

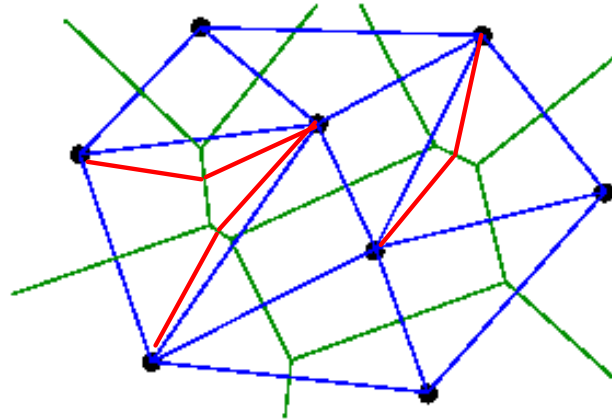
Satz

Jede Triangulation $T(P)$ einer Punktmenge P kann in endlich vielen Schritten in eine winkeloptimale Triangulation verwandelt werden.

2. Definition und Eigenschaften der Delaunay Triangulation

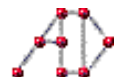
Die Delaunay Triangulation ist das "straight - line dual" des Voronoi Diagramms.

Kante zwischen zwei benachbarten Vor-Regionen

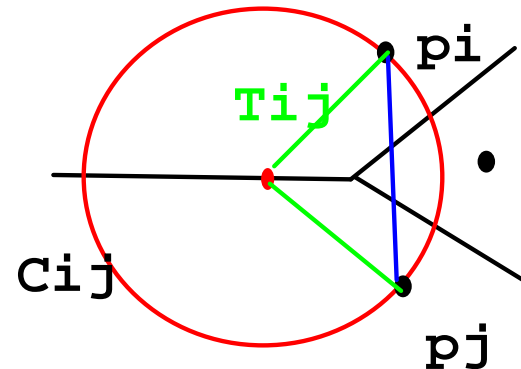


Jeder Voronoi Knoten ist Mittelpunkt eines Dreiecks der Delaunay Triangulation.

(Für Mengen von Punkten in allgemeiner Lage)



Planarität



Wissen (Vor-Diagrammen)

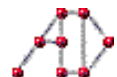
$p_i p_j$ in Delauney Graph $G(P) \Leftrightarrow$
ex. C_{ij} , mit p_i und p_j und keinem p_k
auf dem Rand

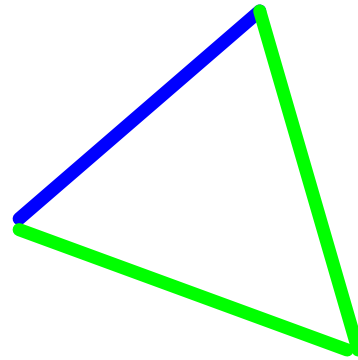
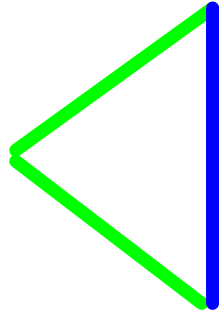
Satz: $G(P)$ planar.

Beweis:

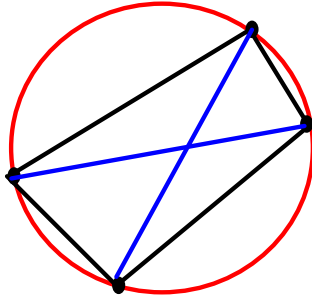
C_{kl} weiterer Kreis und T_{kl} weiteres Dreieck
 C_{kl} und C_{ij} leer $\Rightarrow T_{kl}$ schneidet T_{ij} nicht

$\Rightarrow \overline{p_i p_j}$ schneidet $\overline{p_k p_l}$ nicht





Triangulation



Annahme:

Keine 4 Punkte auf einem Kreis

Delaunay Graph ist Triangulation

3 Punkte p_i, p_j, p_k bilden Dreieck im
Delaunay-Graph \Leftrightarrow

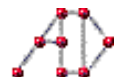
Kreis um p_i, p_j, p_k enthält keinen weiteren Punkt.

\Rightarrow Jede Delaunay-Triangulation ist legal

Ziel:

Winkeloptimale Triangulation ist Delaunay.

Jede Delaunay-Triangulation ist Winkeloptimal.



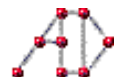
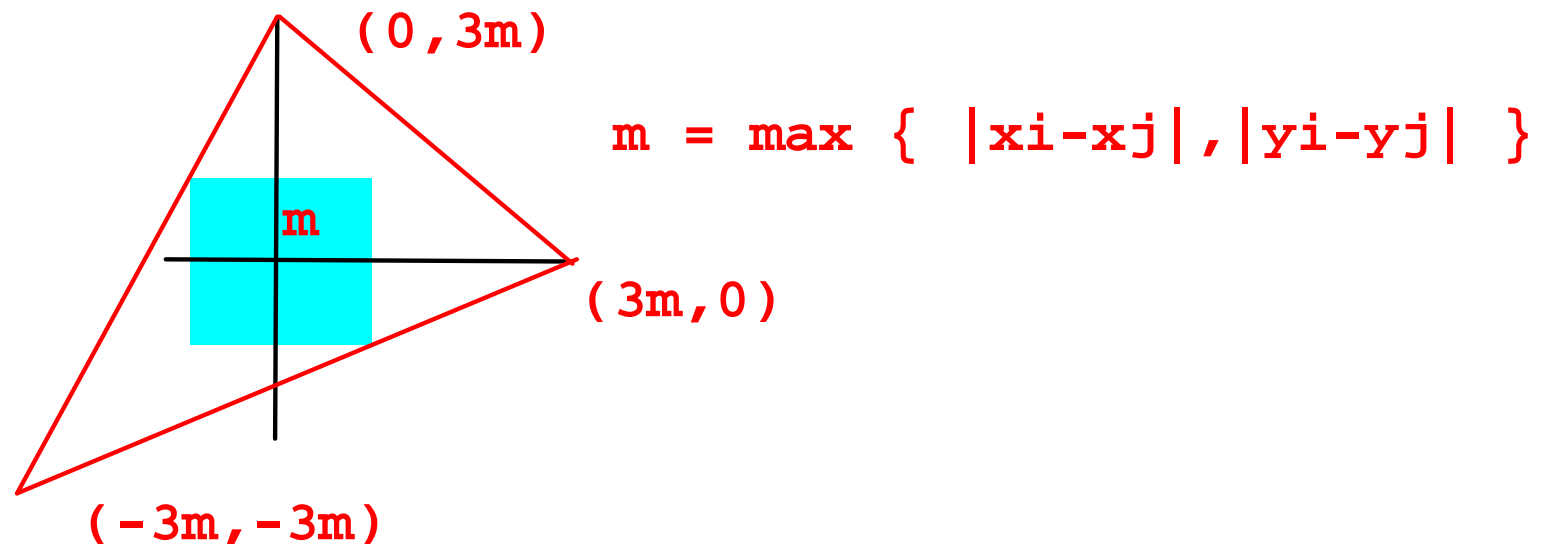
Äquivalente Charakterisierungen der Delaunay Triangulation $DT(P)$ einer Punktmenge P

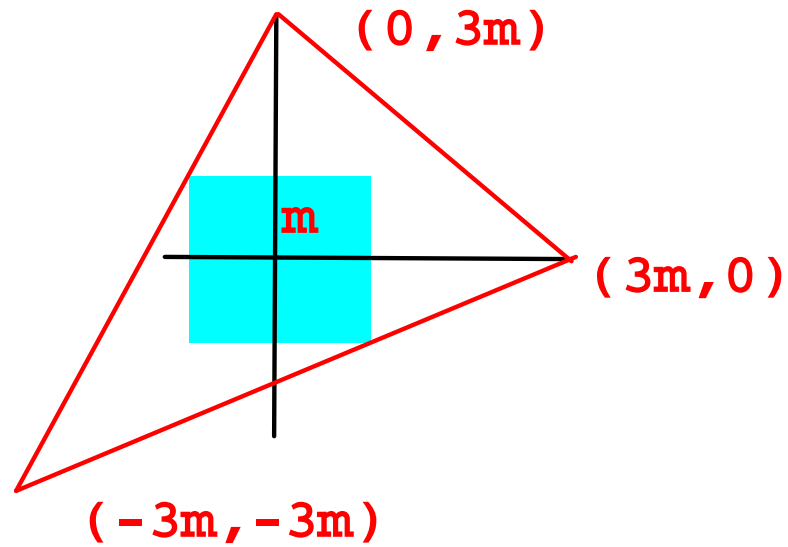
- (1) $DT(P)$ ist straight-line-dual von $VD(P)$
- (2) $DT(P)$ ist eine Triangulation der konvexen Hülle von P , so dass alle Kanten legal sind.
- (3) $DT(P)$ ist eine Triangulation der konvexen Hülle von P , so dass für jedes Dreieck das Kreiskriterium gilt.
- (4) $DT(P)$ ist winkeloptimale Triangulation
- (5) Eine Kante $\overline{p_i p_j}$ ist Kante in $DT(P)$ gdw. es einen abgeschlossenen Kreis gibt, auf dessen Rand p_i und p_j liegen und der keinen weiteren Punkt aus P in seinem Innern enthält.

3. Berechnung der Delaunay Triangulation (randomisiert, inkrementell)

Initial:

Dreieck xyz , daß alle Punkte p_1, \dots, p_n einschließt





Ausgangsdreieck so gross, dass jeder Kreis um drei Punkte aus P keinen Eckpunkt des Ausgangsdreiecks enthält.

Algorithmus DT(P)

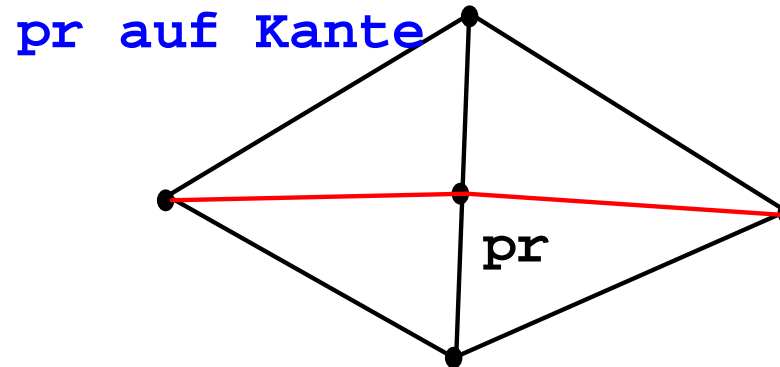
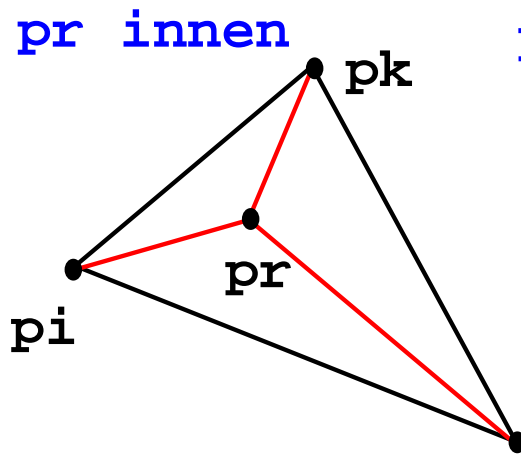
Seien

x, y, z Punkte, sodass P im Ausgangsdreieck T mit diesen Eckpunkten enthalten ist.

1. Initialisiere $DT(P)$ durch T
2. Berechne eine zufällige Permutation p_1, \dots, p_n der Punkte von P
3. Für $r=1$ bis n :
Füge p_r zu $DT(P)$ hinzu, d.h. finde das Dreieck der aktuellen Triangulation, in dem p_r liegt, füge neue Kanten ein und legalisiere sie.
4. Entferne alle Kanten, die mit x, y, z verbunden sind.

Legalisieren einer Kante

2 Fälle



Legalize(pr, pi pj)

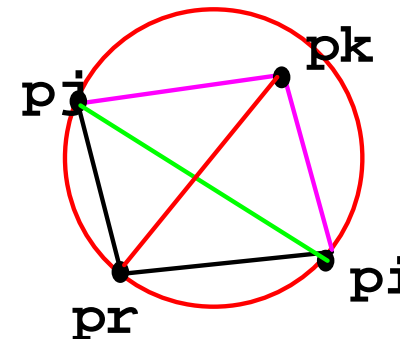
if (pi pj illegal)

pk Zeuge

Ersetze pi pj mit pr pk

Legalize(pr, pi pk)

Legalize(pr, pj pk)



Algorithmus

Delaunay(P)

Ausgabe: Delaunay Triangulation der Menge P

T= xyz

for r = 1..n

Finde Dreieck in dem pr liegt

1. Im Dreieck p_i, p_j, p_k

splitte p_i, p_j, p_k

Legalize($\overline{p_i p_j}$), Legalize($\overline{p_i p_k}$),
Legalize($\overline{p_k p_i}$)

2. Auf Kante $p_i p_j$

Legalize($\overline{p_i p_l}$), Legalize($\overline{p_l p_j p_k}$)
Legalize($\overline{p_l p_j}$), Legalize($\overline{p_k p_i}$)

Lösche xyz mit allen Kanten zu P



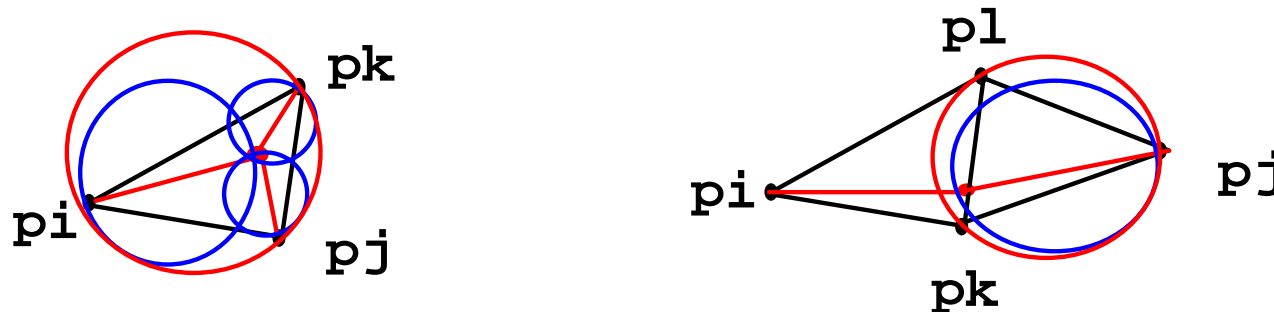
Korrektheit

z.Z. Jede neu erzeugte Kante in Iteration r
ist Kante im
Delaunay Graph von x, y, z, p_1, \dots, p_r

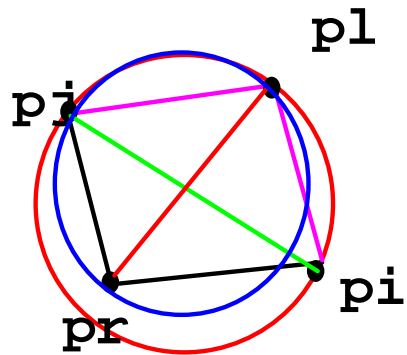
Nutze definierende Eigenschaft:
größter leerer Kreis

Beweisidee: "Schrumpfen des Kreises"

Vor Einfügen von p_r war Kreis um p_i, p_j, p_k
leer!



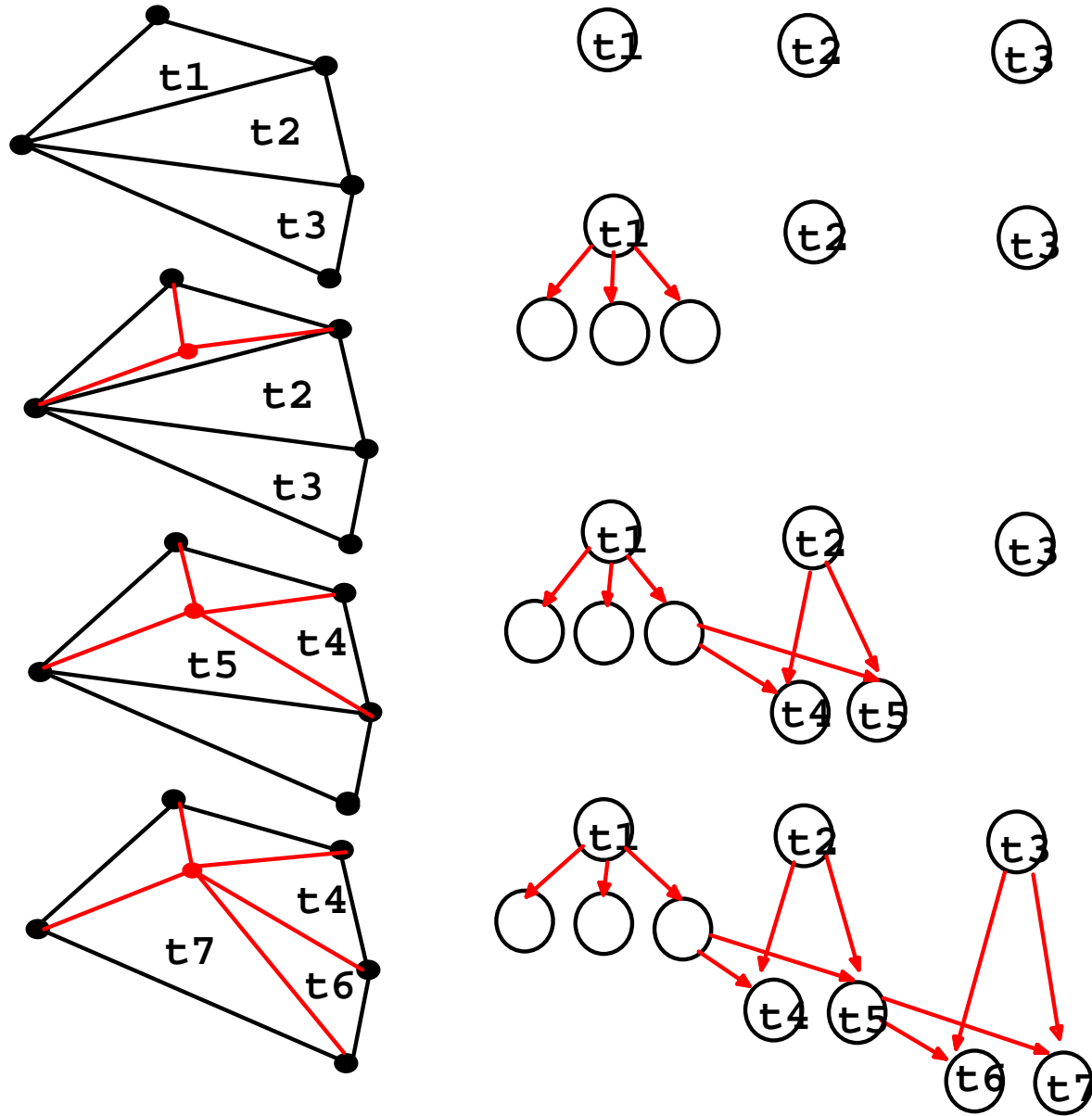
Flippen einer Kante erzeugt nur legale neue Kante:



Vor der Einfügung von p_r
war Kreis um p_i, p_j, p_l leer!

Beobachtung: Kantenflip nach Einfügen von p_r
erzeugt nur mit p_r inzidente Kanten.

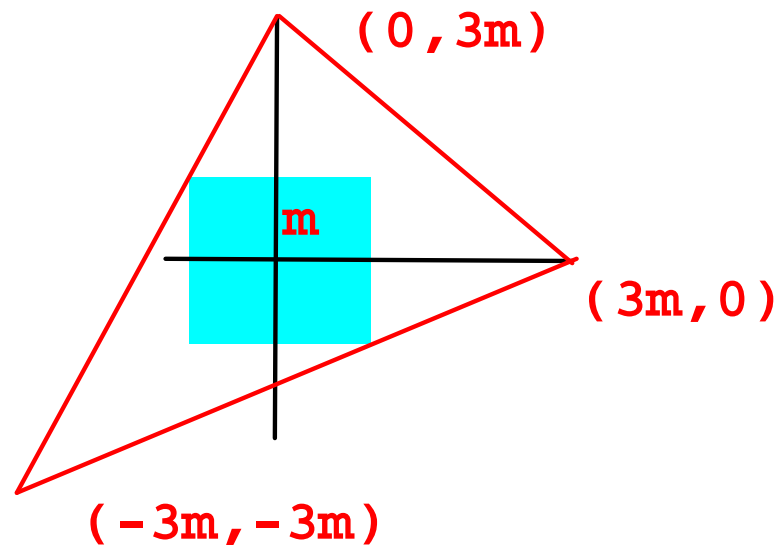
Datenstruktur zur Punktlokalisierung



5. Analyse des Algorithmus zur Konstruktion der Delaunay Triangulation $DT(P)$

Satz:

Der Erwartungswert für die Anzahl der vom Algorithmus zur Konstruktion von $DT(P)$ erzeugten Dreiecke für eine Menge P von n Punkten ist höchstens $9n+1$.



Satz

Die Delaunay Triangulation $DT(P)$ für eine Menge P von n Punkten kann in erwarteter Zeit $O(n \log n)$ und erwartetem Platz $O(n)$ berechnet werden.