Delaunay Triangulation

1. Motivation

2. Triangulation ebener Punktmengen

3. Definition und Eigenschaften der Delaunay Triangulation

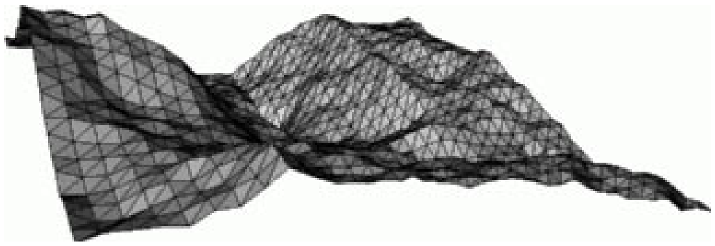
4. Berechnung der Delaunay Triangulation (randomisiert, inkrementell)

5. Analyse des Platz- und Zeitbedarfs

1. Motivation

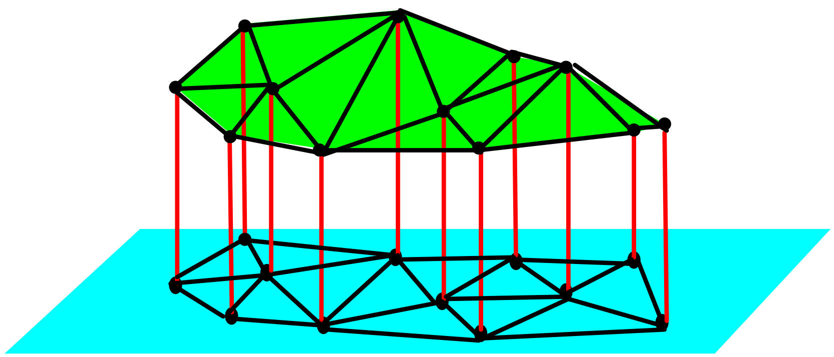
Umwandlung einer topographischen Karte

in eine perspektivische Sicht

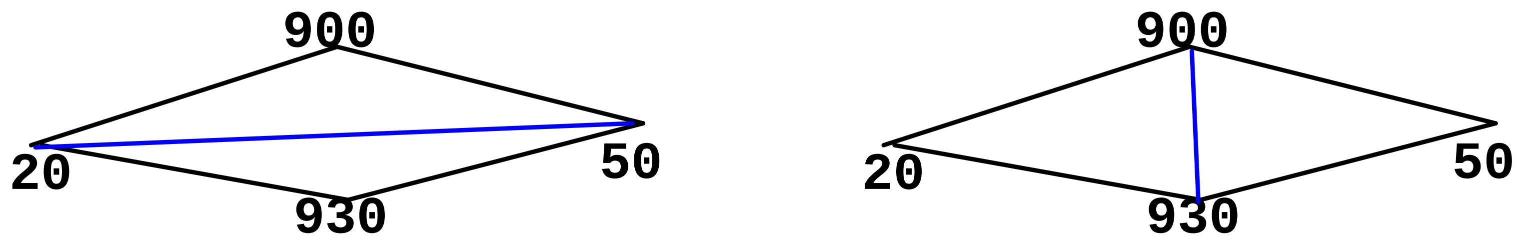


Terrains

Gegeben:
Eine Anzahl von Beispielpunkten p1,...,pn
Gesucht:
Triangulation T mit Bild von T "realistisch"



Das "Flippen" einer Kante

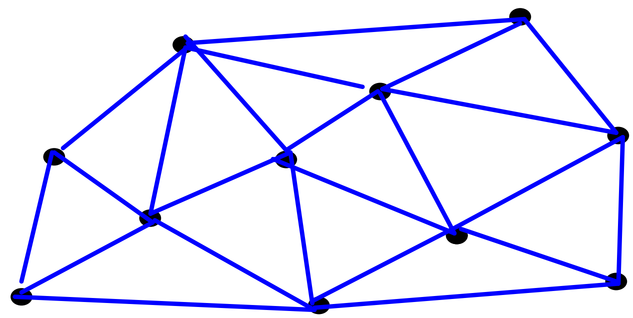


Ziel:
Maximierung des Winkels in der Triangulation

2. Triangulation ebener Punktmengen

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

Eine Triangulation T(P) ist eine planare Auf-teilung 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={p1,...,pn} 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={p1,...,pn}
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 = #Dreiecke + 1

Kante hat jeweils 2 inzidente Flächen =>

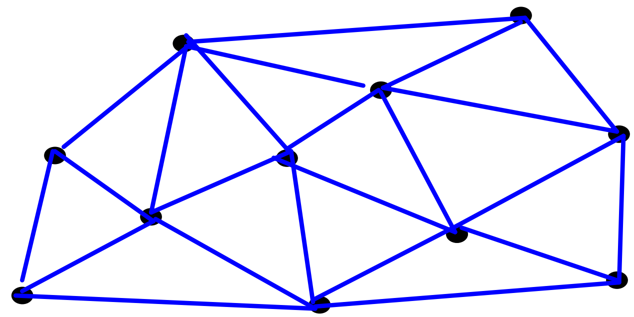
e = (3 #Dreiecke + k)/2

Euler: n-e+f = 2 =>

#Dreiecke = 2n-2-k

und
e = 3n-3-k

Winkelvektor



P Menge von n Punkten

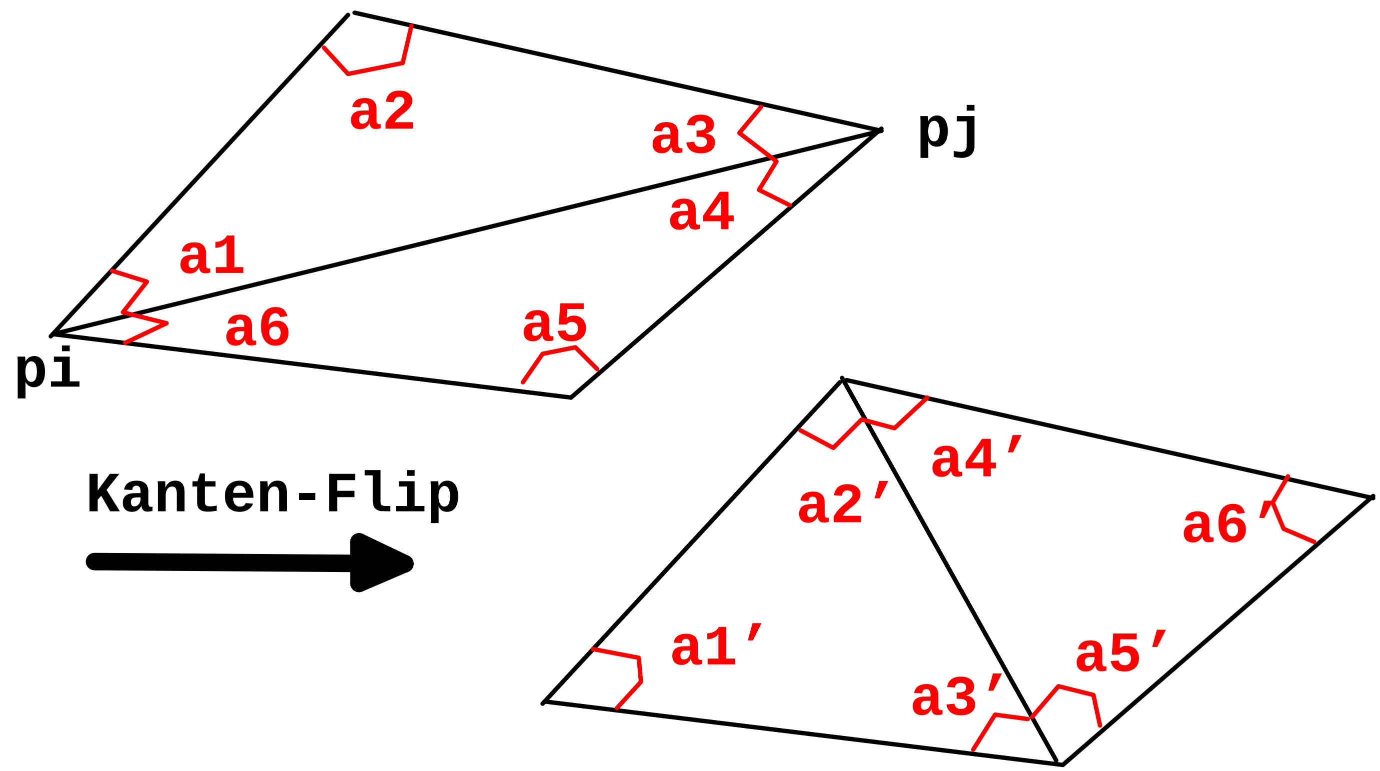
T(P) habe m Dreiecke

A(T) = {a1,...,a3m} Vektor der 3m Winkel in absteigender Größenanordnung

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

Triangulation T(P) heißt winkeloptimal, wenn A(T(P))>=A(T’(P)) für alle Triangulationen T’(P) gilt.

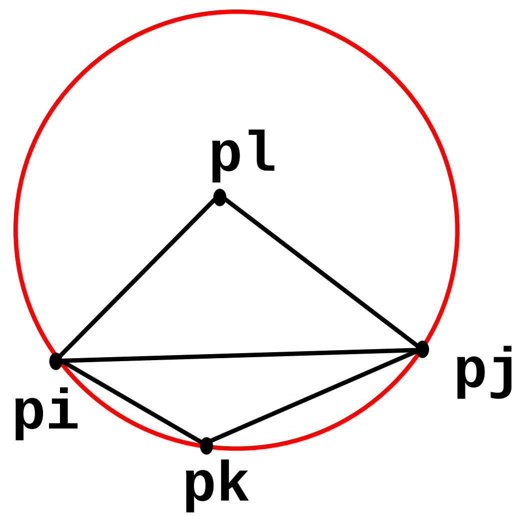
Illegale Kanten

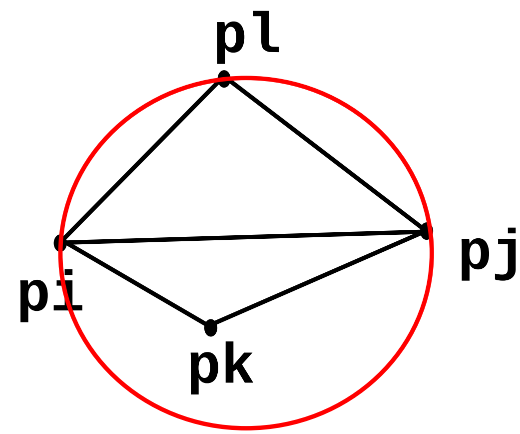


Kante $\overline{pi\ pj}$ illegal <=>
min{ai | i aus 1..6} < min {ai′ | i aus 1..6}

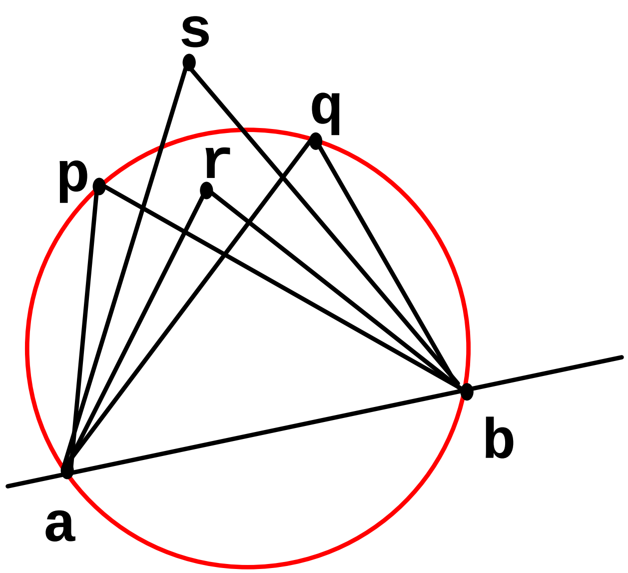
T hat illegale Kante =>
Flippen zu T′ ergibt A(T′)>A(T)

Illegalitäts-Test



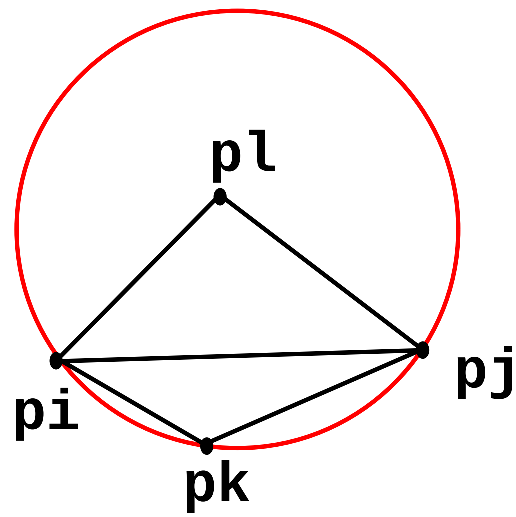


Thales Theorem

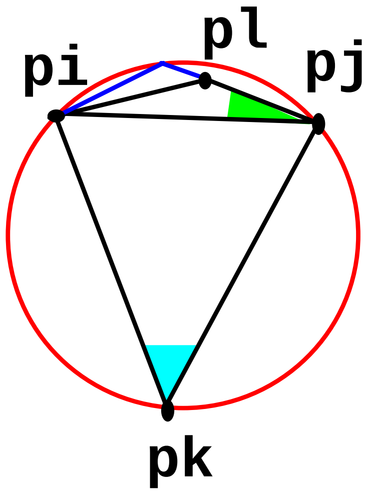


Winkel(asb) <
Winkel(aqb) = Winkel(apb) <
Winkel(arb)

Sei C Kreis um Dreieck pi,pj,pk
und pl weiterer Punkt.



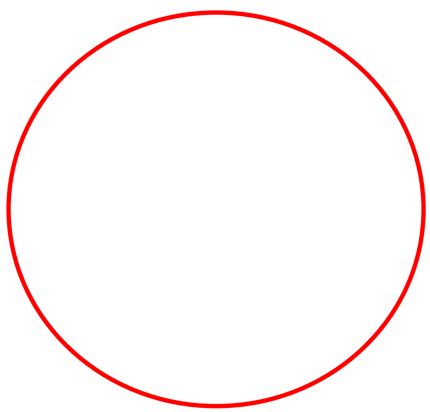
$\overline{\text{pi pj}}$ illegal <=> pl im Inneren von C



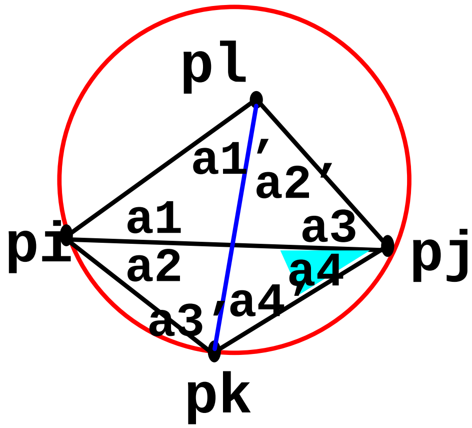
Quadrilateral mit

pl im Innern des Kreises
durch pi,pj,pk.

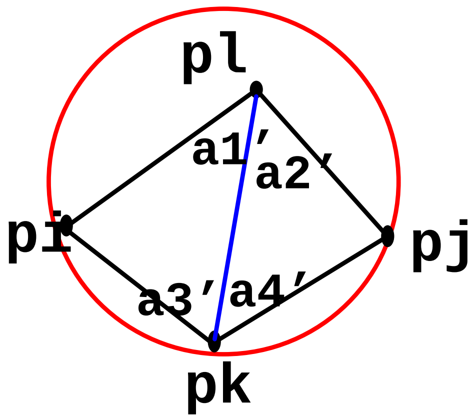
Beh.: Minimaler Winkel
nicht bei pk!

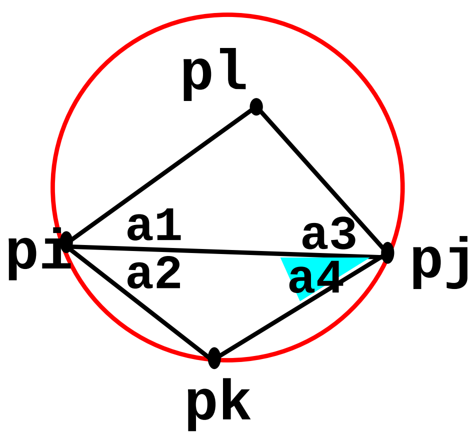


Ziel: pi pj illegal

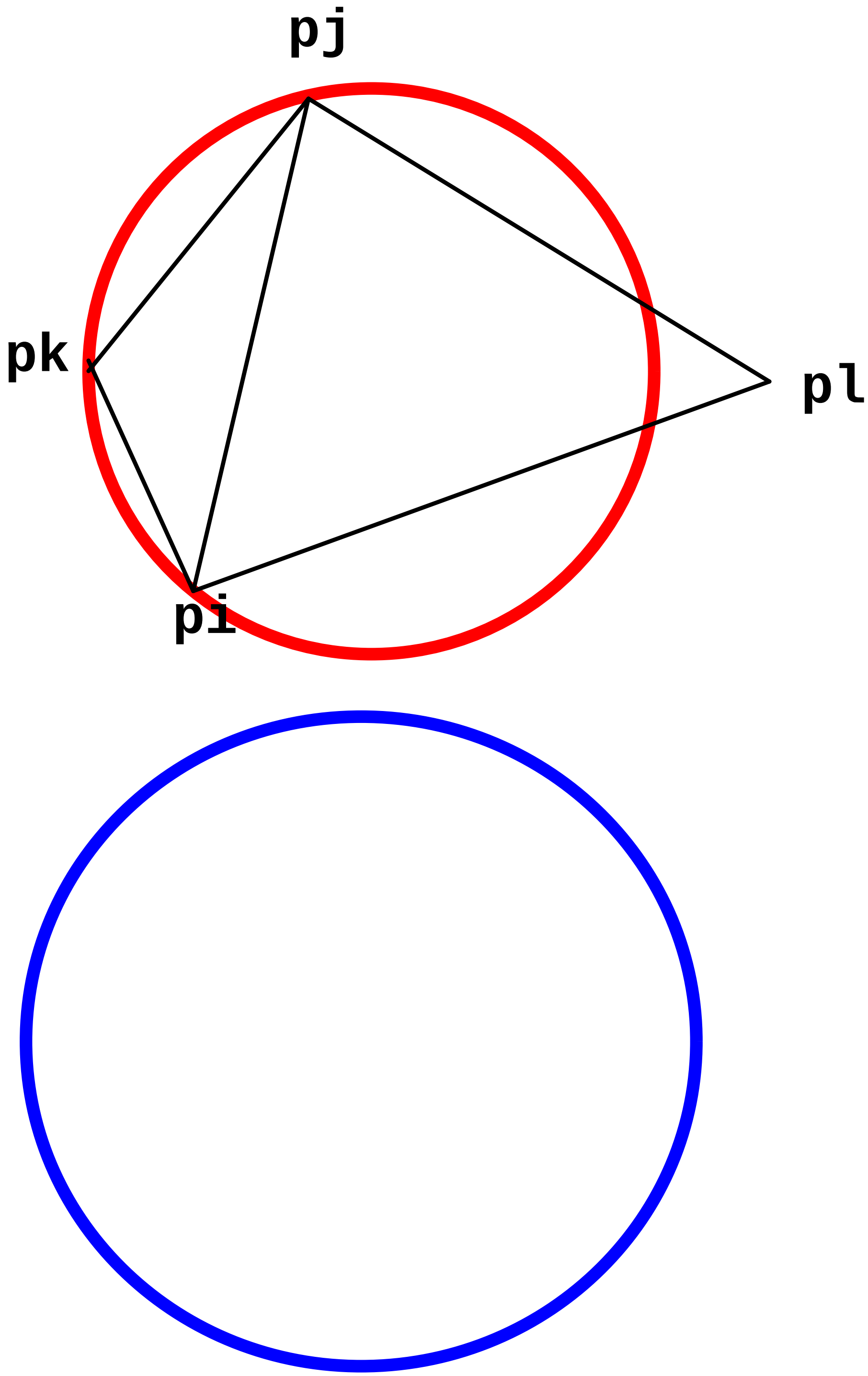


O.E. a4 minimal.





Kreiskriterium verletzt
=> Kante illegal

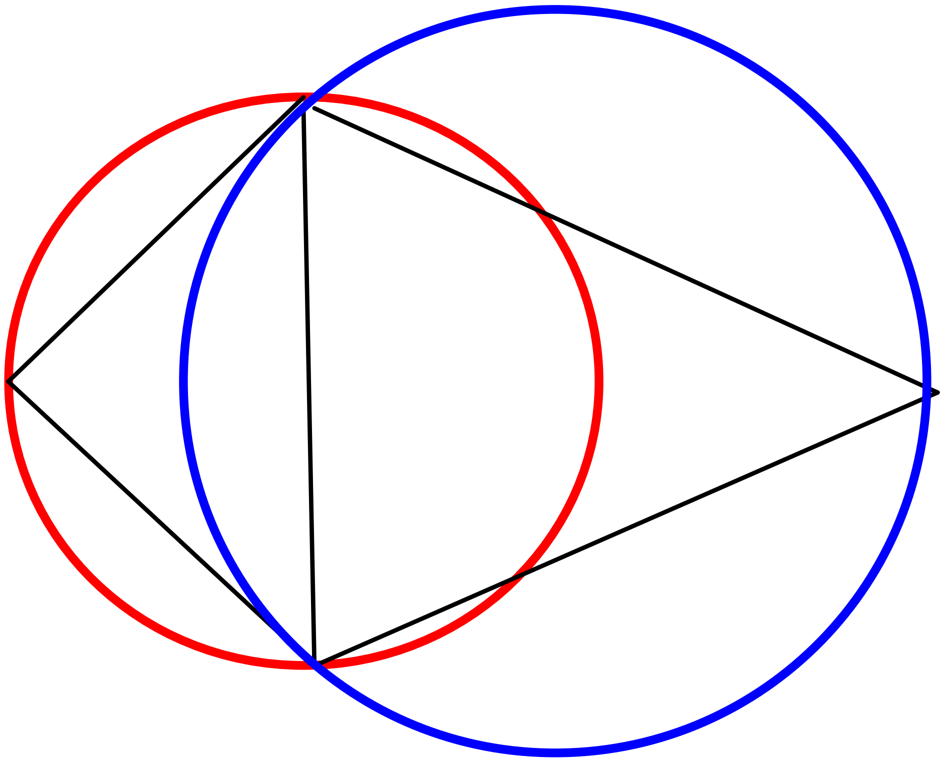


Kante $\overline{pi\ pj}$ illegal

Kante $\overline{pk\ pl}$ 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.

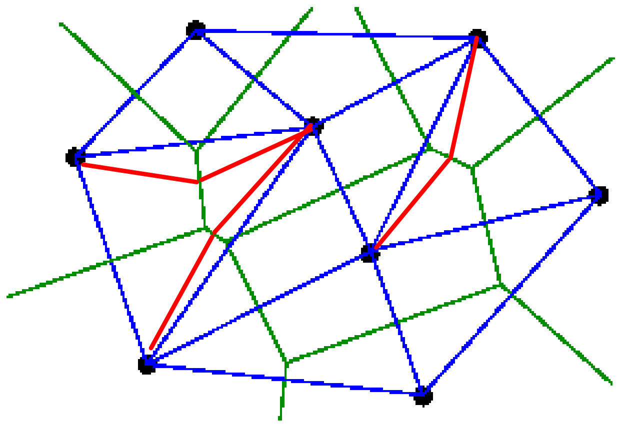
Satz

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.

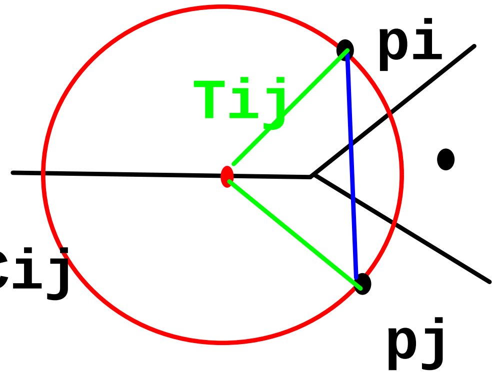
Kante zwischen zwei benachbarten Vor-Regionen



Jeder Voronoi Knoten ist Mittelpunkt eines Dreiecks der Delaunay Triangulation.

(Für Mengen von Punkten in allgemeiner Lage)

Planarität



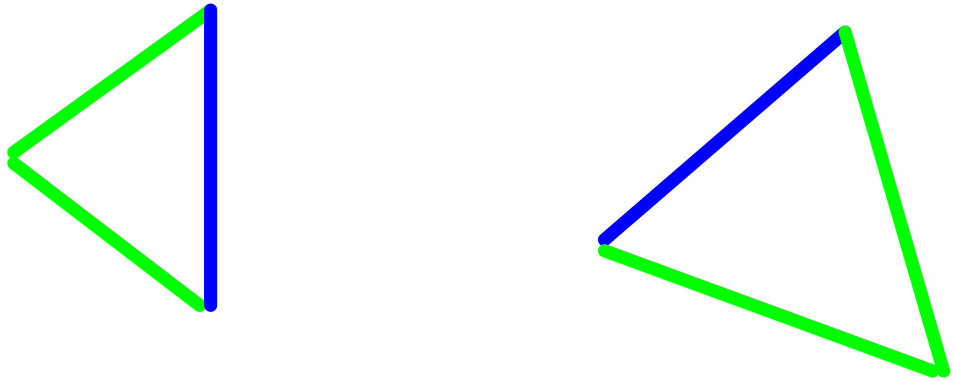
Wissen (Vor-Diagrammen)

$\overline{\text{pi pj}}$ in Delauney Graph G(P) <=>
ex. Cij, mit pi und pj und keinem pk auf dem Rand

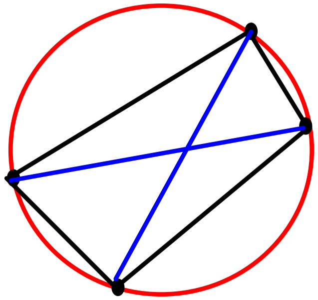
Satz: G(P) planar.

Beweis:
Ckl weiterer Kreis und Tkl weiteres Dreieck
Ckl und Cij leer => Tkl schneidet Tij nicht

=> $\overline{\text{pi pj}}$ schneidet $\overline{\text{pk pl}}$ nicht



Triangulation



Annahme:

Keine 4 Punkte auf einem Kreis

Delaunay Graph ist Triangulation

3 Punkte pi,pj,pk bilden Dreieck im
Delaunay-Graph <=>
Kreis um pi,pj,pk enthält keinen weiteren Punkt.

=> 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.

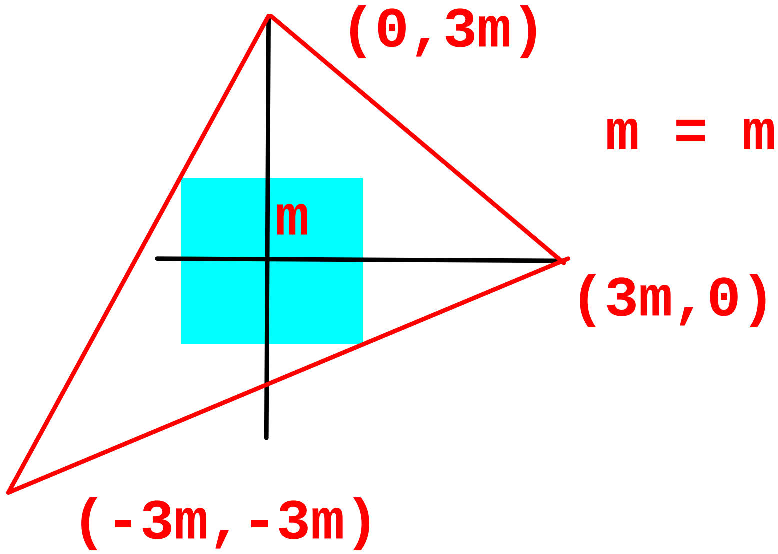
(3) DT(P) ist eine Triangulation der konvexen Hülle von P, so dass für jedes Dreick das Kreiskriterium gilt.

(4) DT(P) ist winkeloptimale Triangulation

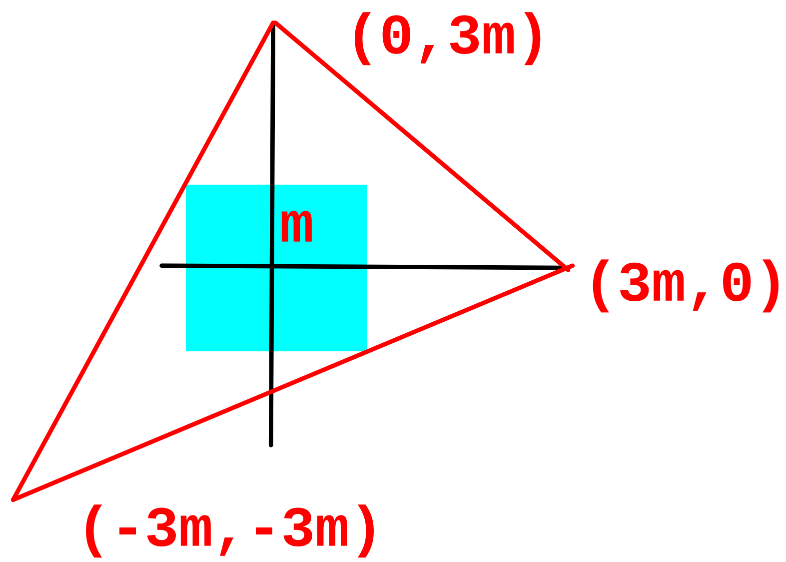
(5) Eine Kante $\overline{p_i p_j}$ ist Kante in DT(P) gdw. es einen abgeschlossenen Kreis gibt, auf dessen Rand p_i und p_j liegen und der keinen weiteren Punkt aus P in seinem Innern enthält.

3. Berechnung der Delaunay Triangulation (randomisiert, inkrementell)

Initial:
Dreieck xyz, daß alle Punkte p1,...,pn einschließt



m = max { |xi-xj|,|yi-yj| }



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
   p1,...,pn der Punkte von P

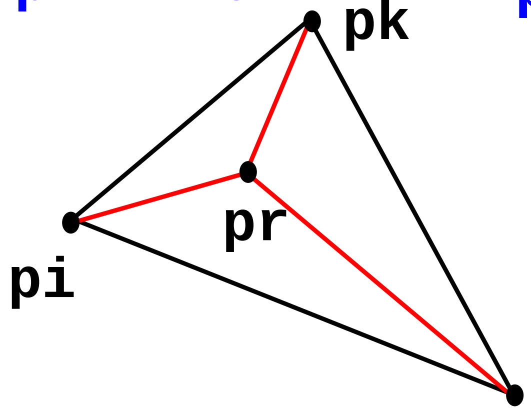
3. Für r=1 bis n:
   Füge pr zu DT(P) hinzu, d.h. finde das
   Dreick der aktuellen Triangulation, in dem
   pr liegt, füge neue Kanten ein und
   legalisiere sie.

4. Entferne alle Kanten, die mit x,y,z
   verbunden sind.

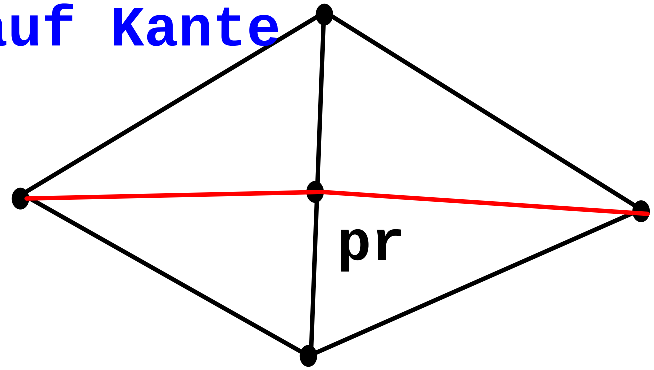
Legalisieren einer Kante

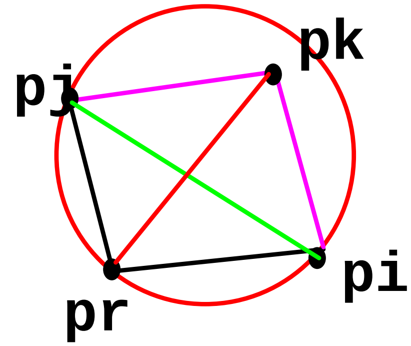
2 Fälle

pr innen



pr auf Kante





Legalize(pr, $\overline{\text{pi pj}}$)

  if ($\overline{\text{pi pj}}$ illegal)
  pk Zeuge
  Ersetze $\overline{\text{pi pj}}$ mit $\overline{\text{pr pk}}$
  Legalize(pr, $\overline{\text{pi pk}}$)
  Legalize(pr, $\overline{\text{pj pk}}$)

Algorithmus

Delaunay(P)

Ausgabe: Delaunay Triangulation der Menge P

T= xyz

for r = 1..n

Finde Dreieck in dem pr liegt

1. **Im Dreieck pi,pj,pk**

   splitte pi,pj,pk

   Legalize(pr,$\overline{pi\ pj}$),Legalize(pr,$\overline{pi\ pk}$), Legalize(pr,$\overline{pk\ pi}$)

2. **Auf Kante pi pj**

   Legalize(pr,$\overline{pi\ pl}$), Legalize($\overline{pl\ pj\ pk}$)
   Legalize(pr,$\overline{pl\ pj}$), Legalize(pr, $\overline{pk\ pi}$)

Lösche xyz mit allen Kanten zu P

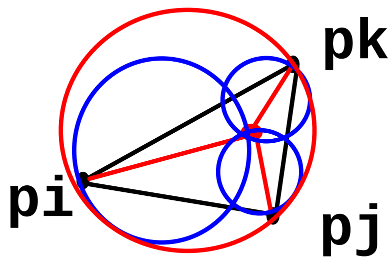
Korrektheit

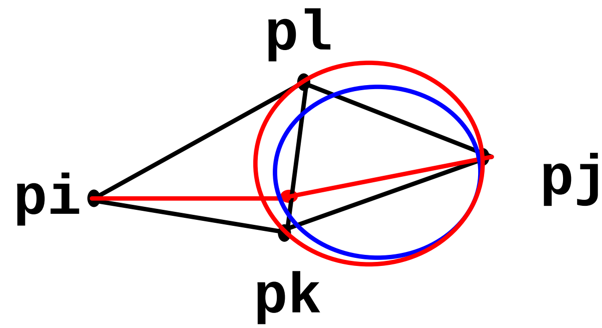
z.Z. Jede neu erzeugte Kante in Iteration r ist Kante im
Delaunay Graph von x,y,z,p1,...,pr

Nutze definierende Eigenschaft:
größter leerer Kreis

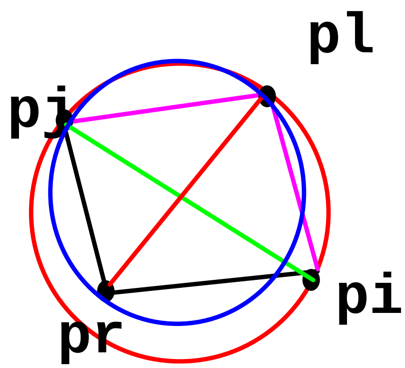
Beweisidee: "Schrumpfen des Kreises"

Vor Einfügen von pr war Kreis um pi,pj,pk leer!





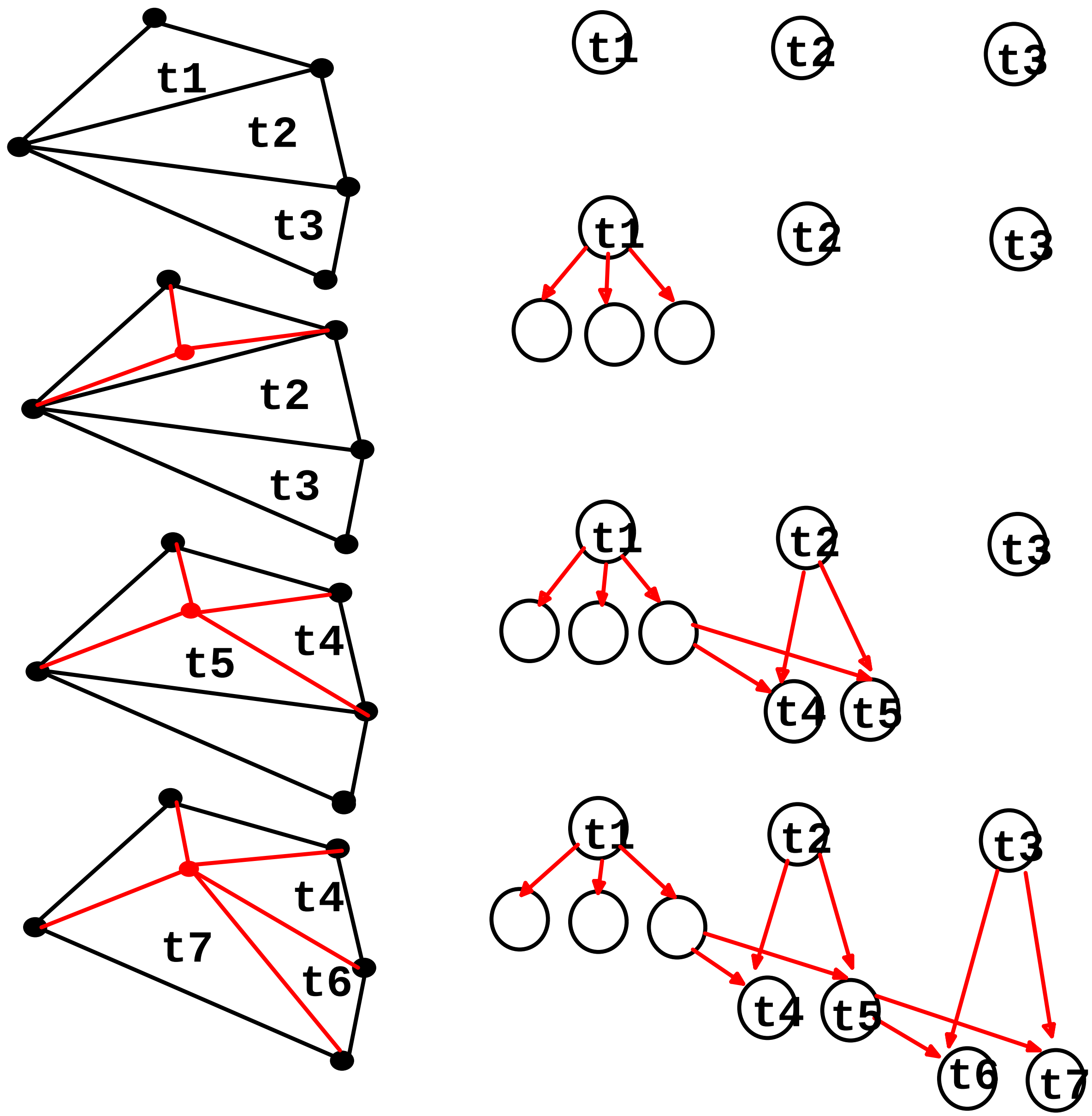
Flippen einer Kante erzeugt nur legale neue Kante:



Vor der Einfügung von pr war Kreis um pi,pj,pl leer!

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

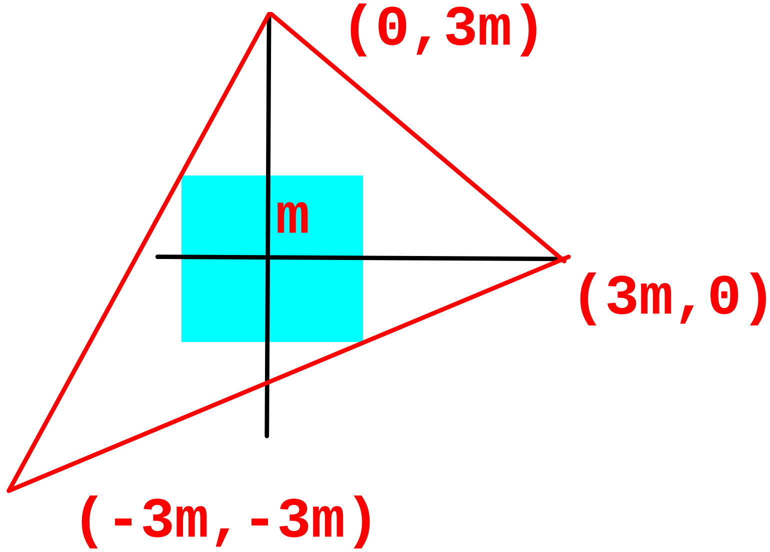
Datenstruktur zur Punktlokalisation



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.