Approximationsschemata in Ablaufplanung, Graphentheorie und Geometrie

Geometrische Spannbäume minimalen Durchmessers

(engl.: Geometric Minimum-Diameter Spanning Tree, GMDST)

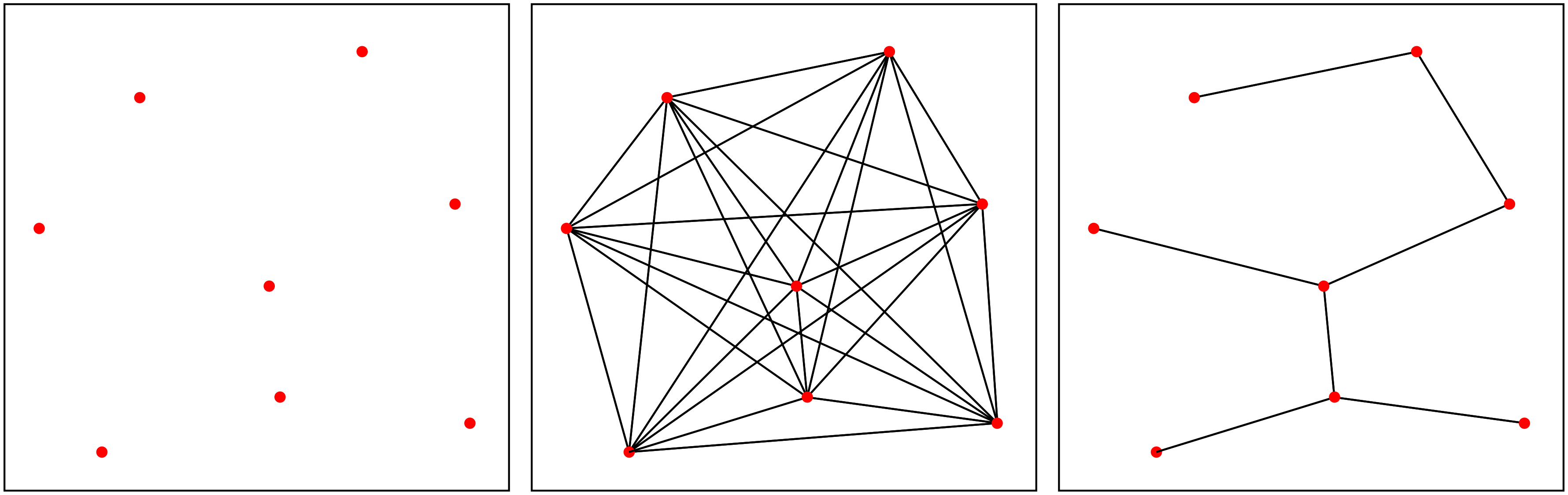
Markus Völker

SS 2004

Überblick

- Begriffsdefinitionen
- Eigenschaften von GMDSTs
- Exakter Algorithmus mit Laufzeit $\Theta(n^3)$
- ε-approximativer Algorithmus mit Laufzeit $O(n + \frac{1}{\varepsilon^3})$
- Zusammenfassung

Geometrischer Spannbaum

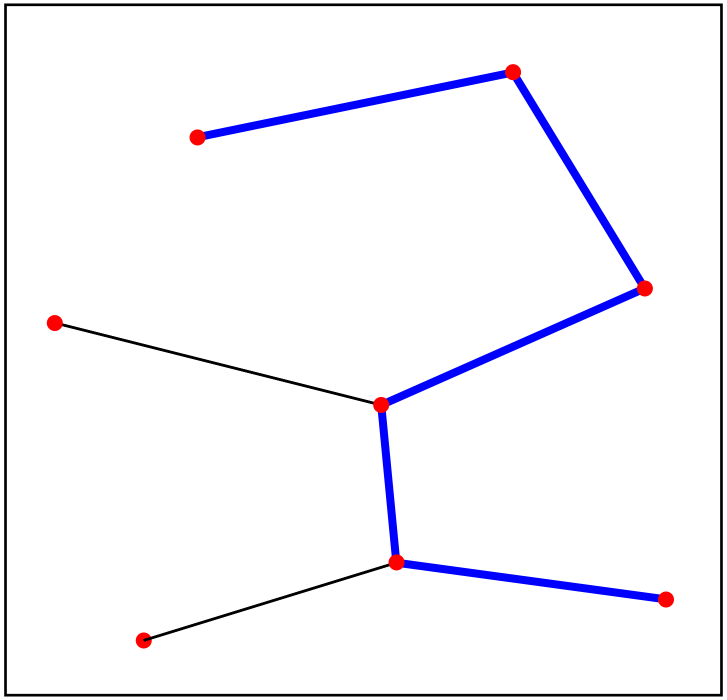


(a) Punktmenge in der euklidischen Ebene

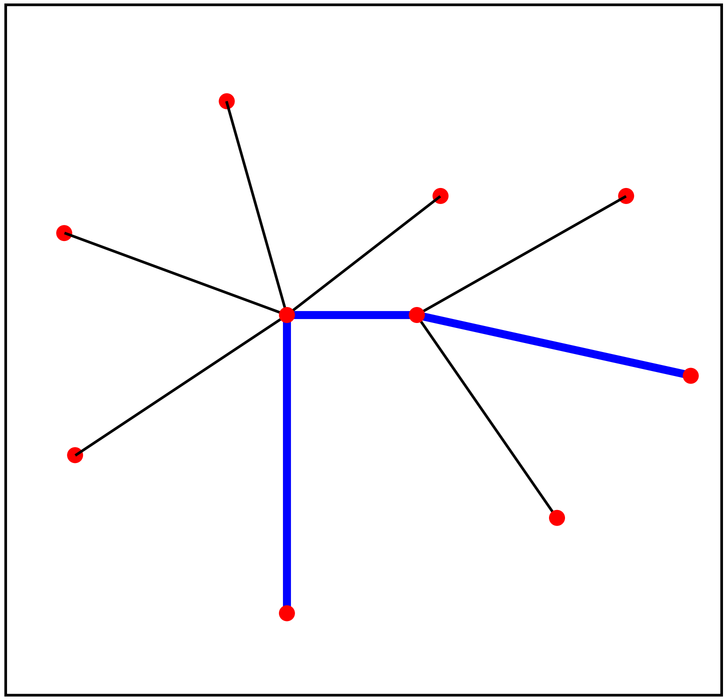
(b) durch die Punktmenge induzierter gewichteter Graph

(c) Geometrischer Spannbaum

Durchmesser eines Spannbaums und Geometrischer Spannbaum minimalen Durchmessers



(a) Durchmesser eines Spannbaums



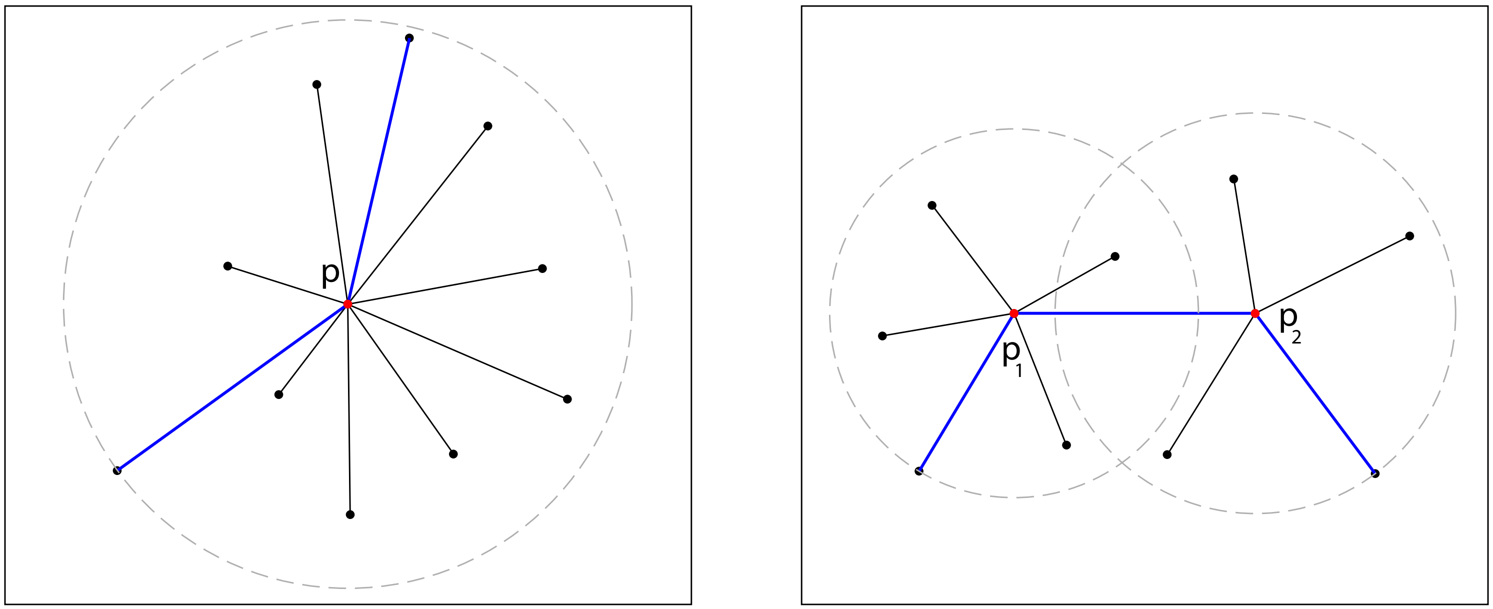
(b) Geometrischer Spannbaum mit minimalem Durchmesser

Historisches

- 1991, J. Ho, D. Lee, C. Chang, C. Wong
  - exakter Algorithmus, Laufzeit $O(n^3)$
- 2002, Timothy Chan
  - exakter Algorithmus, Laufzeit $O(n^{\frac{17}{6}})$
- 2002, J. Gudmundsson, H. Haverkort, S.-M. Park, C.-S. Shin, A. Wolff
  - ε-approximativer Algorithmus
  - Laufzeit $O(\frac{n}{\varepsilon^3} + \frac{n \log n}{\varepsilon})$, Speicherbedarf $O(\frac{n}{\varepsilon^2} + n \log n)$
- 2003, M. Spriggs, M. Keil, S. Bespamyatnikh, M. Segal, J. Snoeyink
  - ε-approximativer Algorithmus
  - Laufzeit $O(n + \frac{1}{\varepsilon^3})$, Speicherbedarf $O(n)$

Eigenschaften minimaler Spannbäume

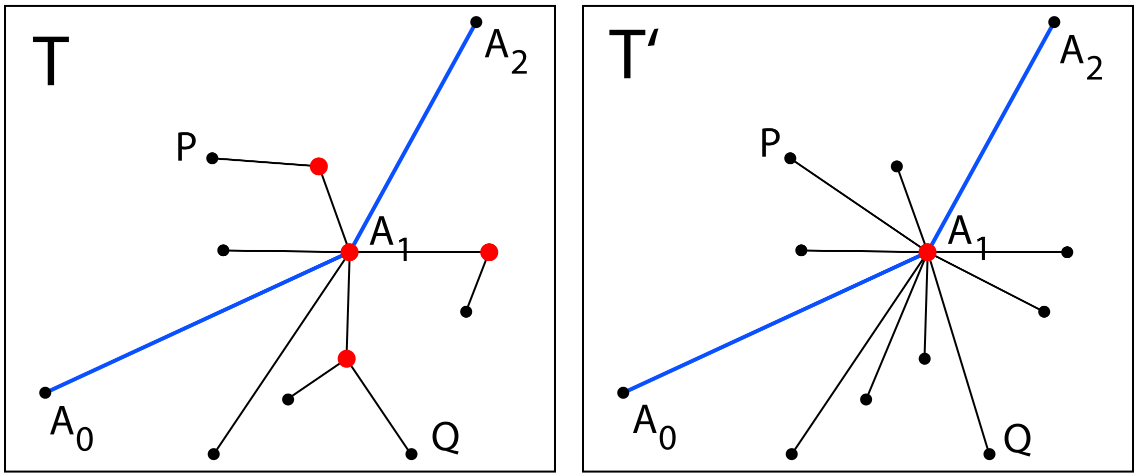
Lemma 1. *Zu jeder Punktmenge existiert ein monopolarer oder ein dipolarer GMDST.*



(a) Monopolarer GMDST (b) Dipolarer GMDST

Beweis zu Lemma 1

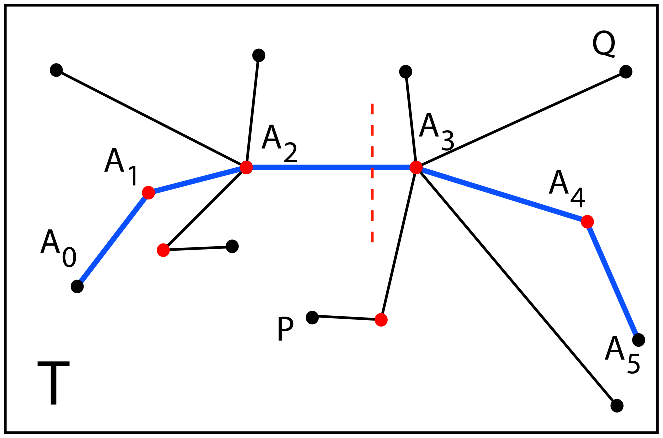
1. Fall: *es existiert ein Durchmesser mit 2 Kanten*

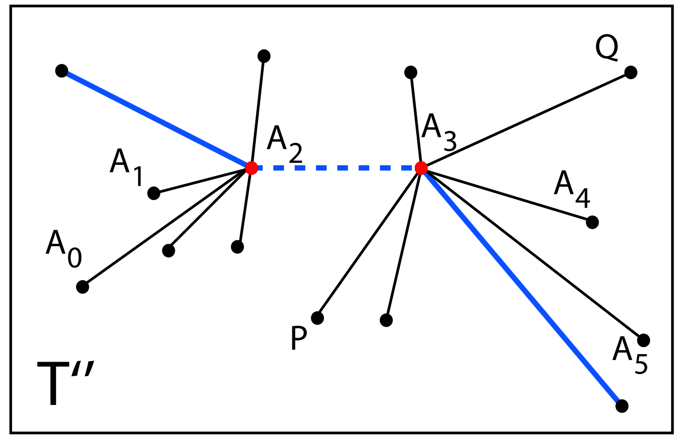


$$
\begin{aligned}
\mathrm{dist}_{\mathrm{T'}}(P,Q) &= |P,A_1| + |A_1,Q| \\
&\leq \mathrm{dist}_{\mathrm{T}}(P,A_1) + \mathrm{dist}_{\mathrm{T}}(A_1,Q) \\
&\leq |A_0,A_1| + |A_1,A_2| \\
&= \mathcal{D}_T
\end{aligned}
$$

Beweis zu Lemma 1

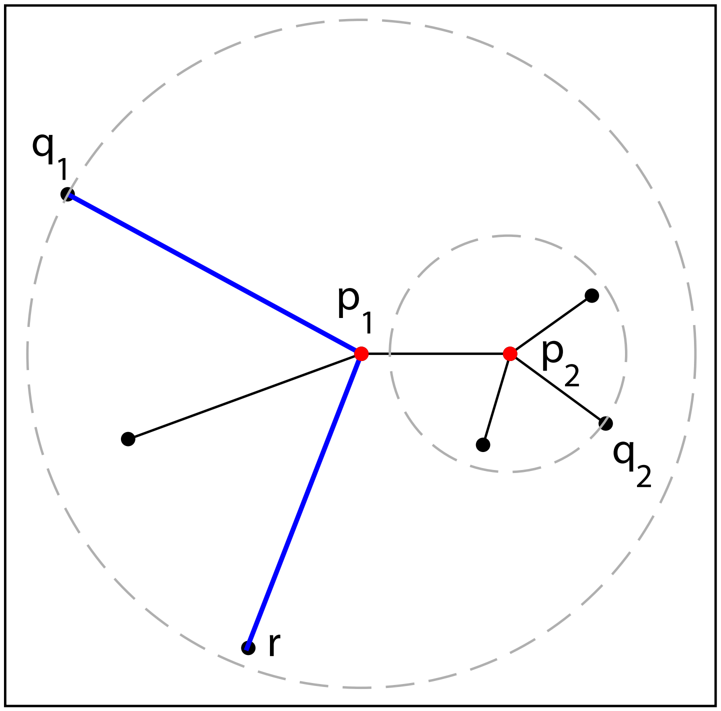
2. Fall: *alle Durchmesser besitzen mindestens 3 Kanten*

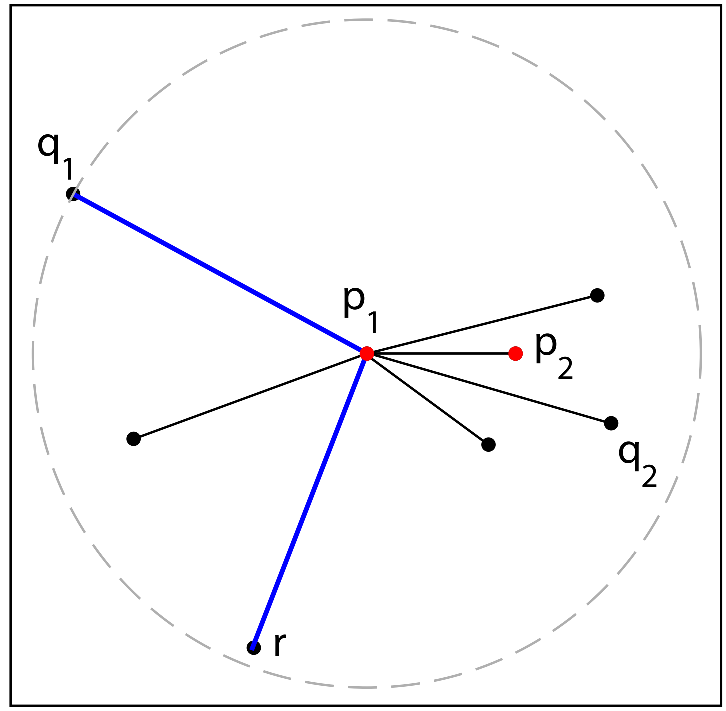




$$\begin{aligned}
\mathrm{dist}_{\mathrm{T}''}(P,Q) &= |P,A_3| + |A_3,Q| \\
&\leq \mathrm{dist}_{\mathrm{T}}(P,A_3) + \mathrm{dist}_{\mathrm{T}}(A_3,Q) \\
&\leq \mathrm{dist}_{\mathrm{T}}(A_3,A_5) + \mathrm{dist}_{\mathrm{T}}(A_3,A_5) \\
&\leq \mathrm{dist}_{\mathrm{T}}(A_0,A_3) + \mathrm{dist}_{\mathrm{T}}(A_3,A_5) \\
&= \mathcal{D}_T
\end{aligned}$$

Stabilitätsbedingung für dipolare Spannbäume





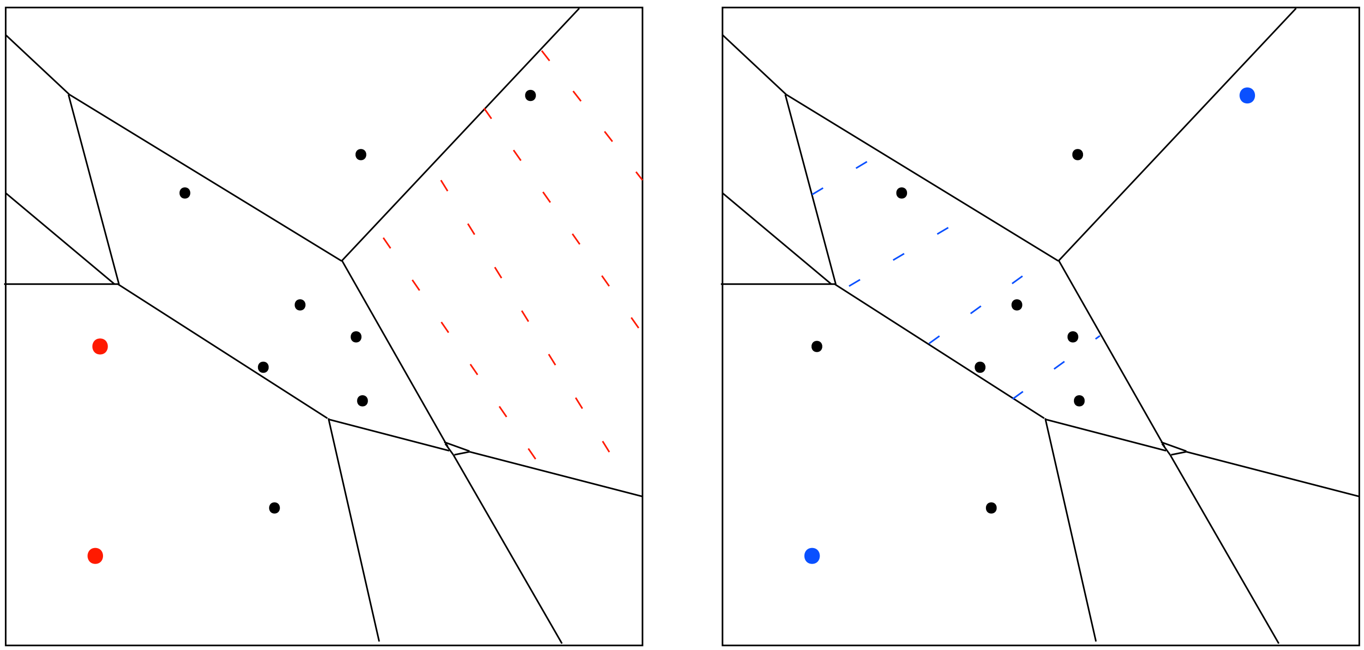
Lemma 2. *Umschließt einer der beiden Kreise alle Punkte, so existiert ein monopolarer Spannbaum, der keinen größeren Durchmesser besitzt als der dipolare. Ansonsten wird der Durchmesser durch einen Pfad mit 3 Kanten bestimmt.*

Algorithmus zur Bestimmung eines GMDST

- Gegeben: eine Punktmenge P mit n Punkten
- Bestimme monopolaren Spannbaum mit minimalem Durchmesser
- Bestimme dipolaren Spannbaum mit minimalem Durchmesser
- Der kleinere der beiden Spannbäume muss dann ein GMDST sein

Bestimmung eines monopolaren GMDSTs

- Distanz zu den beiden entferntesten Punkten minimieren
- *Second-Order Furthest-Neighbor Voronoi Diagram (SOFNVD)* in $O(n \log n)$ Zeit
- Optimaler Monopol in $O(n \log n)$ Zeit



Bestimmung eines dipolaren GMDSTs

$minSoFar \leftarrow \infty$

foreach *Punktepaar* p_1, p_2 *aus* P **do**

  sortiere restliche Punkte nach Distanz zu p_1 ins Feld L

  for $1 \leq k \leq n-2$ **do**

    verbinde p_1 mit den ersten k Punkten aus L

    verbinde p_2 mit den restlichen Punkten

    if *Stabilitätsbedingung erfüllt* **then**

      $D \leftarrow |L[k], p_1| + |p_1, p_2| + \max_{l=k+1}^{n-2} |L[l], p_2|$

      if $D < minSoFar$ **then** $minSoFar \rightarrow D$

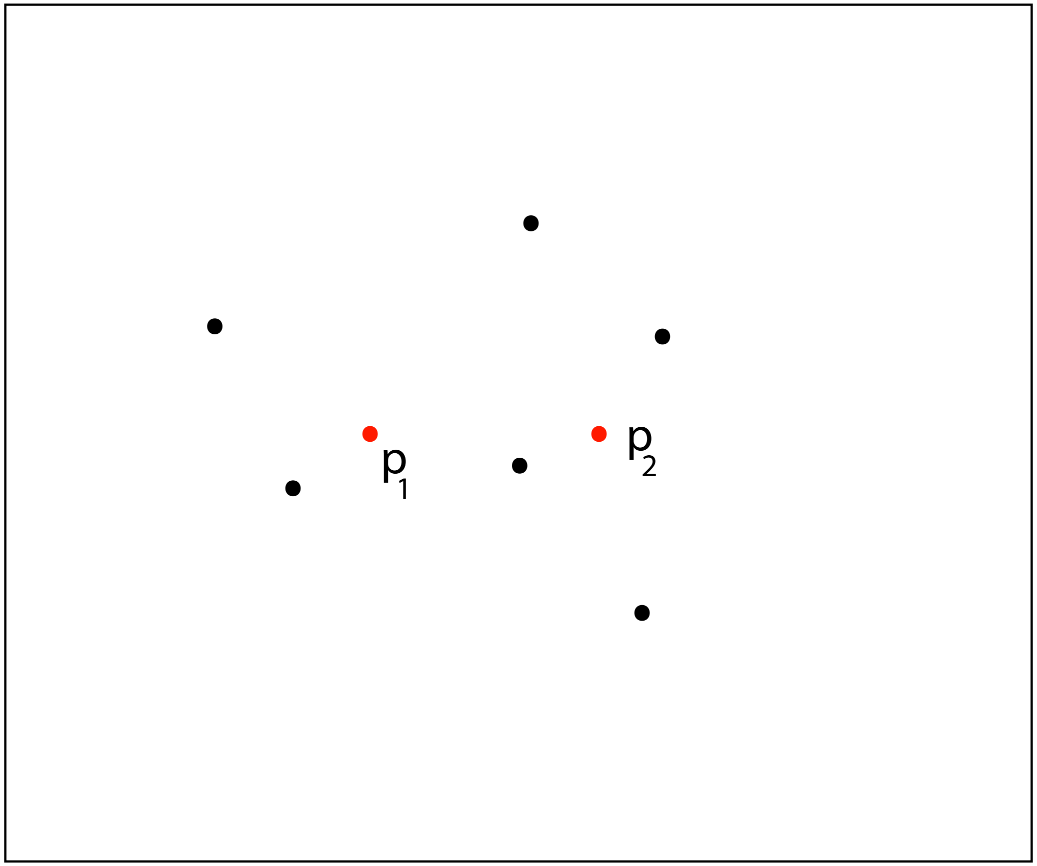
    end

  end

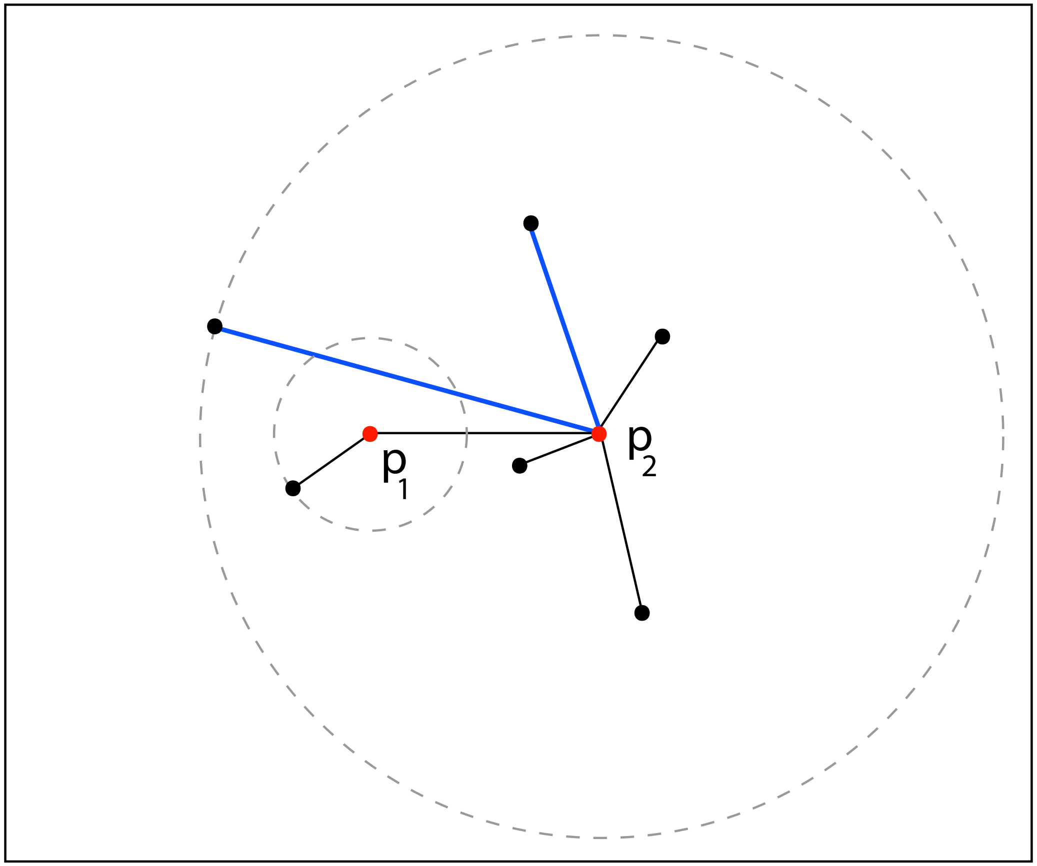
end

- in Laufzeit von $\Theta(n^3)$ möglich

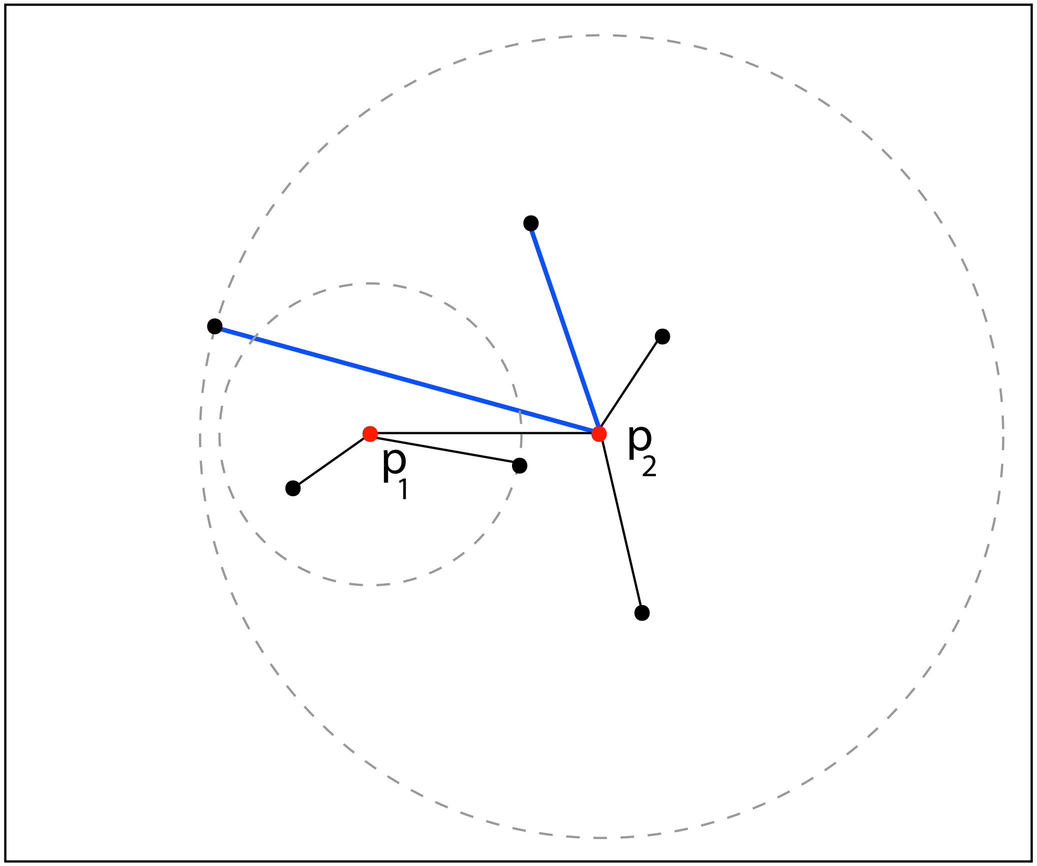
Animation zur Bestimmung des dipolaren GMDST



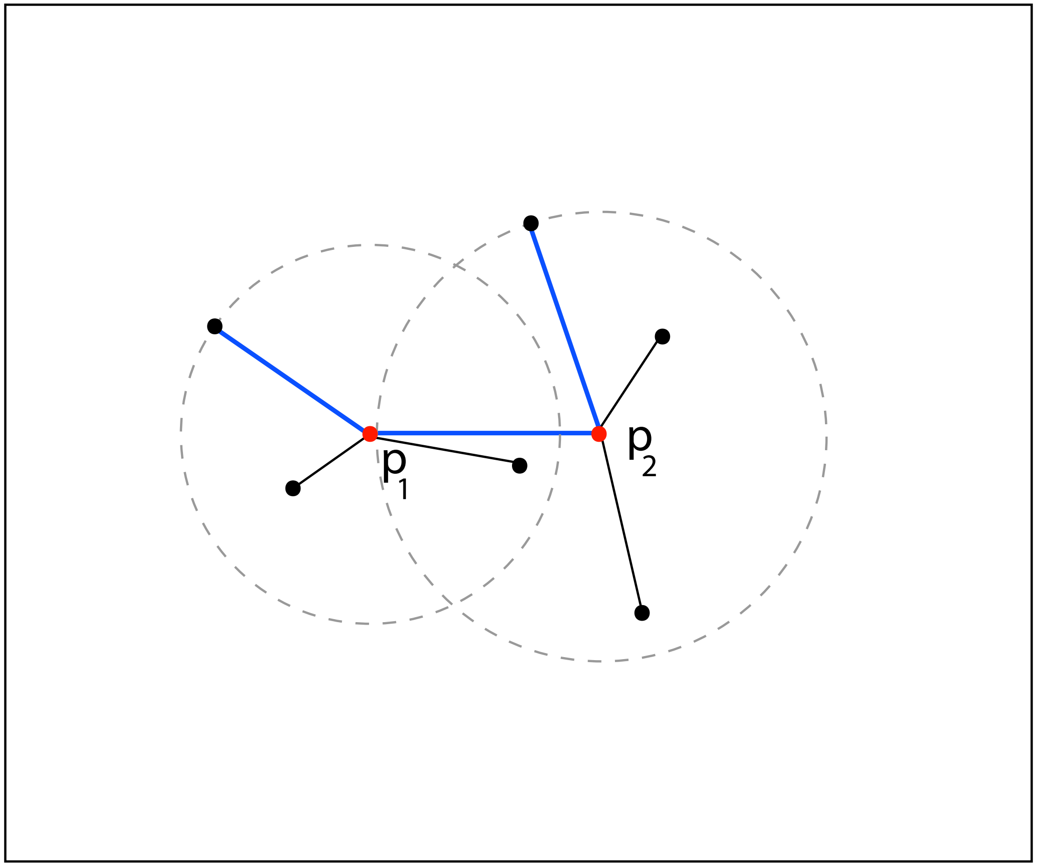
Animation zur Bestimmung des dipolaren GMDST



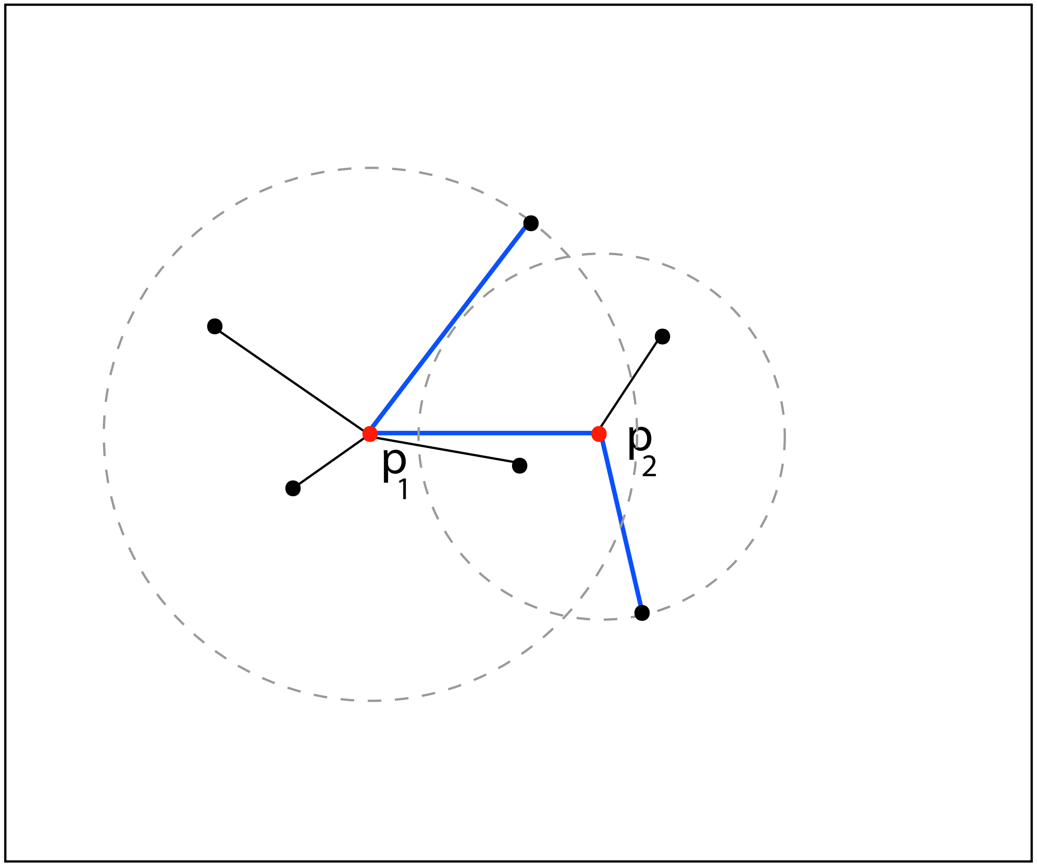
Animation zur Bestimmung des dipolaren GMDST



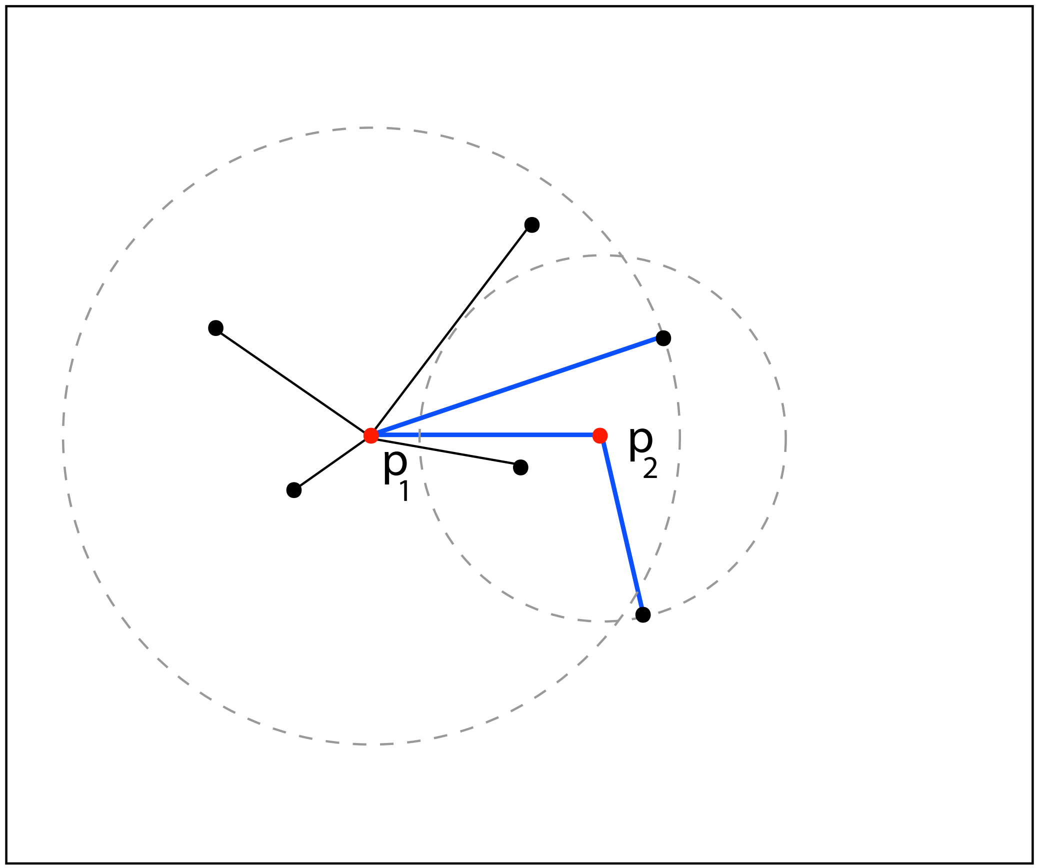
Animation zur Bestimmung des dipolaren GMDST



Animation zur Bestimmung des dipolaren GMDST



Animation zur Bestimmung des dipolaren GMDST

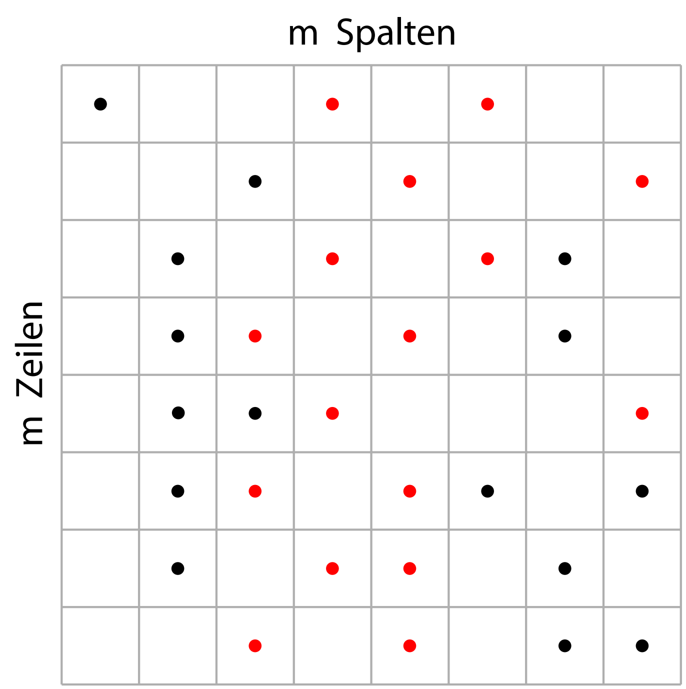


Ein Approximationsschema zur Annäherung des GMDST

- exakter Algorithmus hat Laufzeit von $\Theta(n^3)$
- *geringer* Durchmesser ausreichend $\Rightarrow$ Approximationsschema
- Durchmesser maximal um Faktor $(1+\varepsilon)$ größer als der eines GMDST
- Laufzeit $O(n+\varepsilon^{-3})$
- Speicherbedarf $O(n)$

Ein einfacheres Problem (RGMDST)

- Punktmenge auf $m \times m$-Gitter angeordnet
- Mono- oder dipolarer Spannbaum mit inneren Knoten in einer Zeile
- 2 Kandidaten in jeder Zeile für die inneren Punkte



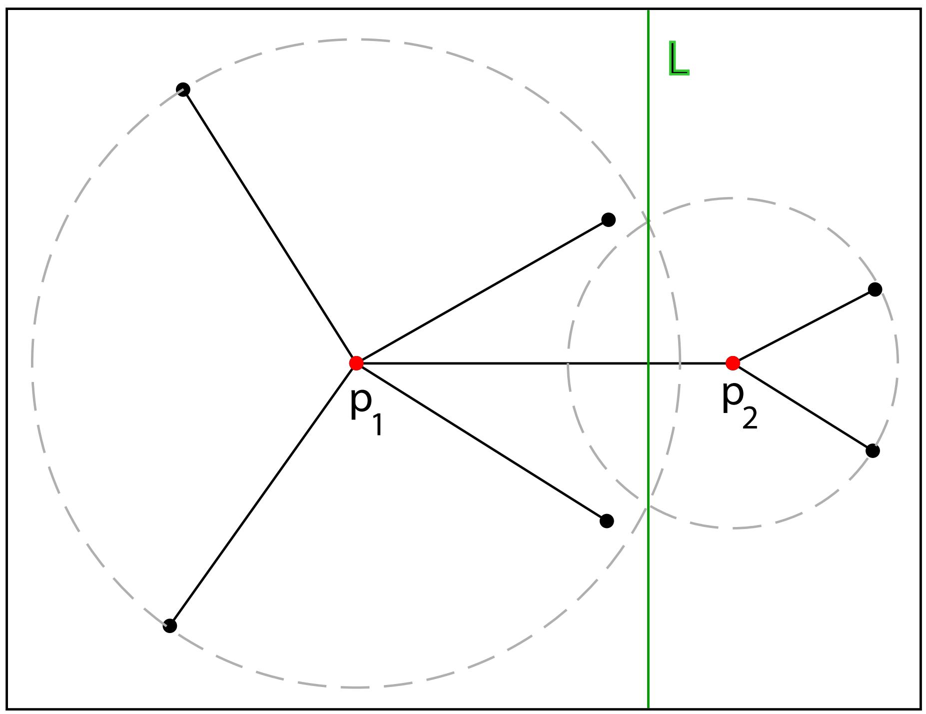
Lösen des vereinfachten Problems

Bestimmung eines optimalen monopolaren Spannbaums

- *SOFNVD* benötigt nur obere und untere beiden Punkte jeder Zeile
- *SOFNVD* der maximal $4m$ Punkte in $O(m \log m)$ Zeit
- $2m$ mögliche Monopole
- Durchmesser jedes Monopols in $O(\log m)$ Zeit
- optimaler monopolarer Spannbaum in $O(m \log m)$ Zeit

Lösen des vereinfachten Problems

Lemma 3. *Seien p_1 und p_2 innere Knoten eines dipolaren Spannbaumes und bilden $\overline{p_1 p_2}$ eine horizontale Linie. Dann existiert eine vertikale Trennlinie L, so dass der Spannbaum, der durch das Verbinden aller Punkte links von L mit p_1 und aller Punkte rechts von L mit p_2 entsteht, unter allen Spannbäumen mit inneren Knoten p_1 und p_2 minimal ist.*



Lösen des vereinfachten Problems

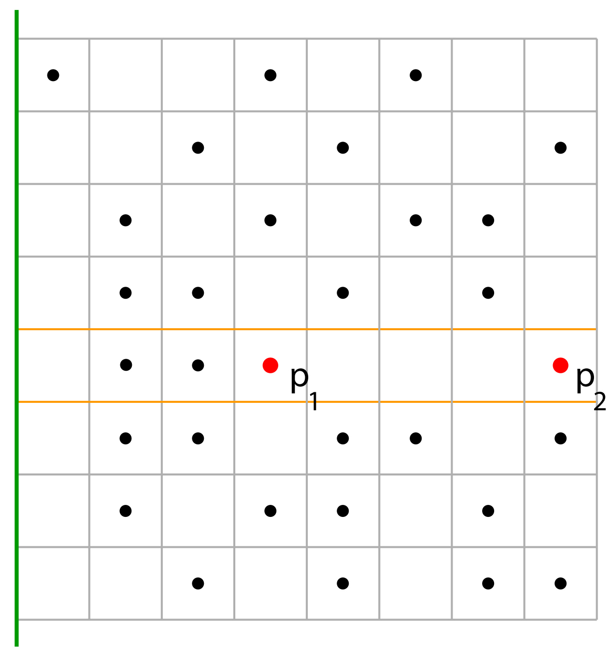
Bestimmung eines minimalen dipolaren Spannbaums

- Bestimme für jede Zeile die vertikale Trennlinie mittels Sweepline in $O(m)$ Zeit
  - $l_1(k)$: Distanz des von p_1 entferntesten Punktes links von Spalte k
  - $r_1(k)$: Distanz des von p_1 entferntesten Punktes rechts von Spalte k
  - $l_2(k)$: Distanz des von p_2 entferntesten Punktes links von Spalte k
  - $r_2(k)$: Distanz des von p_2 entferntesten Punktes rechts von Spalte k

Lösen des vereinfachten Problems

Bestimmung eines optimalen dipolaren Spannbaums

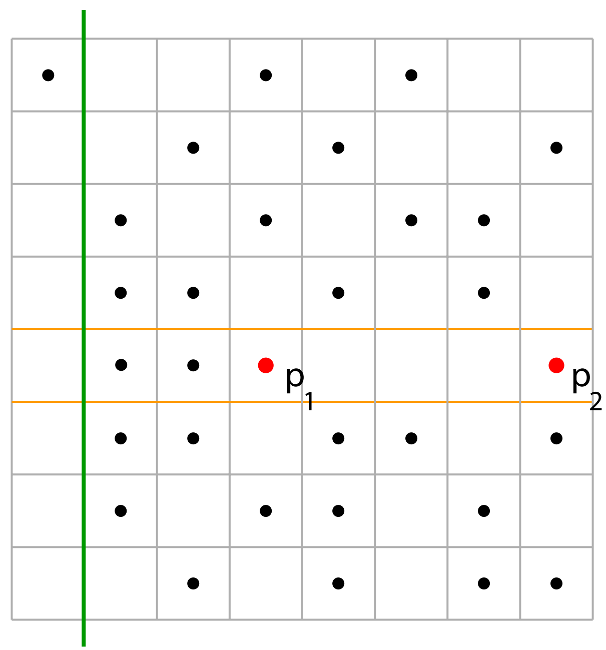
- Bestimme für jede Zeile die vertikale Trennlinie mittels Sweepline in $O(m)$ Zeit
  - $l_1(k)$: Distanz des von p_1 entferntesten Punktes links von Spalte k



Lösen des vereinfachten Problems

Bestimmung eines optimalen dipolaren Spannbaums

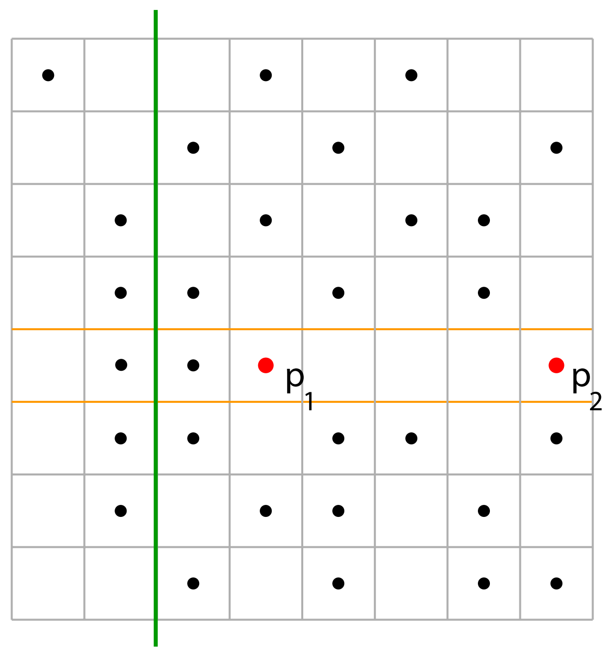
- Bestimme für jede Zeile die vertikale Trennlinie mittels Sweepline in $O(m)$ Zeit
  - $l_1(k)$: Distanz des von p_1 entferntesten Punktes links von Spalte k



Lösen des vereinfachten Problems

Bestimmung eines optimalen dipolaren Spannbaums

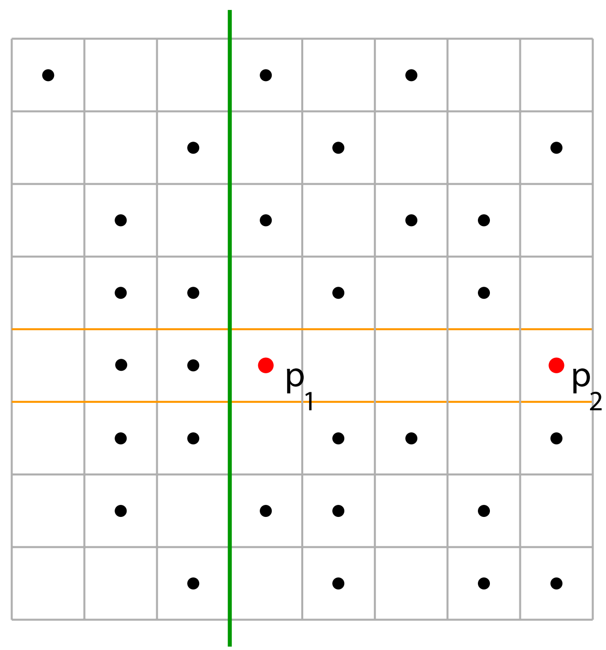
- Bestimme für jede Zeile die vertikale Trennlinie mittels Sweepline in $O(m)$ Zeit
  - $l_1(k)$: Distanz des von p_1 entferntesten Punktes links von Spalte k



Lösen des vereinfachten Problems

Bestimmung eines optimalen dipolaren Spannbaums

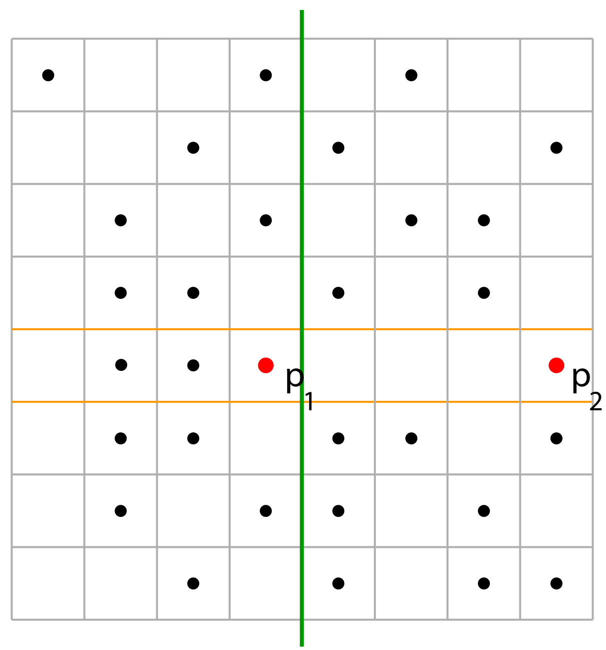
- Bestimme für jede Zeile die vertikale Trennlinie mittels Sweepline in $O(m)$ Zeit
  - $l_1(k)$: Distanz des von p_1 entferntesten Punktes links von Spalte k



Lösen des vereinfachten Problems

Bestimmung eines optimalen dipolaren Spannbaums

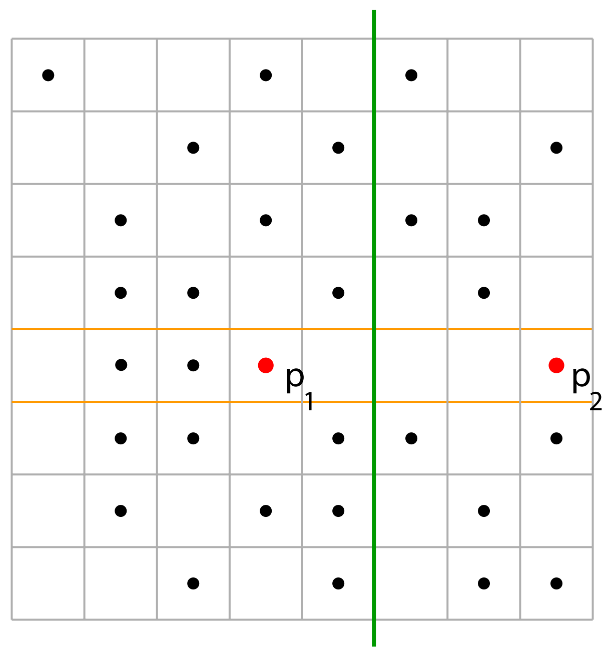
- Bestimme für jede Zeile die vertikale Trennlinie mittels Sweepline in $O(m)$ Zeit
  - $l_1(k)$: Distanz des von p_1 entferntesten Punktes links von Spalte k



Lösen des vereinfachten Problems

Bestimmung eines optimalen dipolaren Spannbaums

- Bestimme für jede Zeile die vertikale Trennlinie mittels Sweepline in $O(m)$ Zeit
  - $l_1(k)$: Distanz des von p_1 entferntesten Punktes links von Spalte k



Lösen des vereinfachten Problems

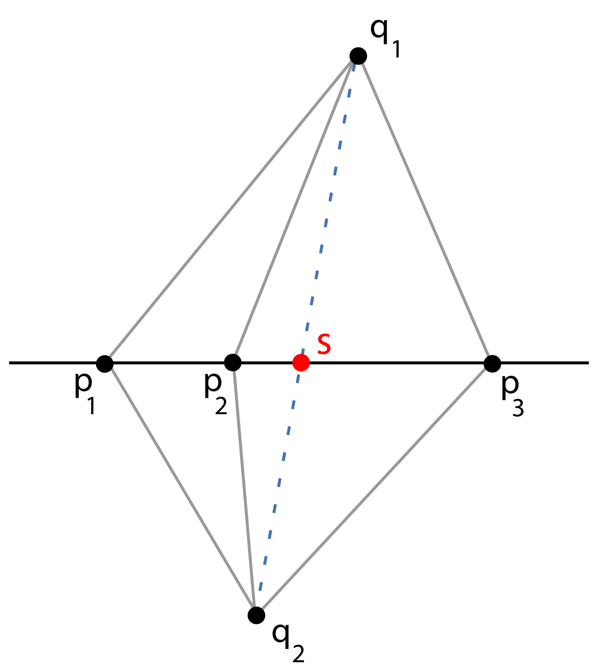
Bestimmung eines optimalen dipolaren Spannbaums

- Bestimme für jede Zeile die vertikale Trennlinie mittels Sweepline in $O(m)$ Zeit
  - $l_1(k)$: Distanz des von p_1 entferntesten Punktes links von Spalte k
  - $r_1(k)$: Distanz des von p_1 entferntesten Punktes rechts von Spalte k
  - $l_2(k)$: Distanz des von p_2 entferntesten Punktes links von Spalte k
  - $r_2(k)$: Distanz des von p_2 entferntesten Punktes rechts von Spalte k
- Stabilitätsbedingung: $l_1(k) < r_1(k+1)$ und $r_2(k+1) < l_2(k)$
- Durchmesser: $D = l_1(k) + d(p_1, p_2) + r_2(k+1)$
- optimaler Durchmesser für jede Zeile in $O(m)$ Zeit
- optimaler dipolarer Spannbaum in $O(m^2)$ Zeit

Der Steiner-Monopol

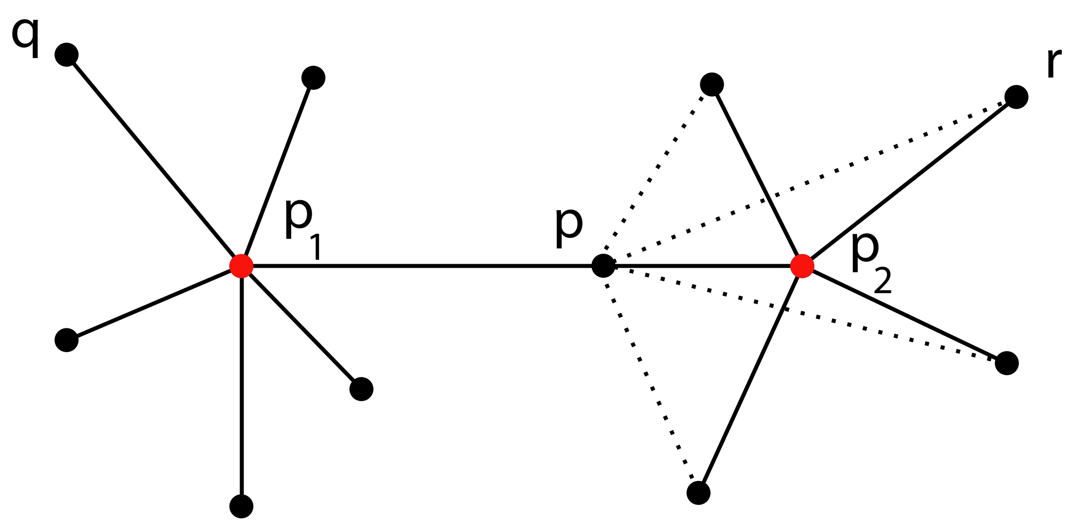
- Steiner-Monopol der Geraden: Punkt auf der Geraden, der als Monopol den Durchmesser des resultierenden Spannbaumes minimiert

- Steiner-Monopol muss in P nicht enthalten sein



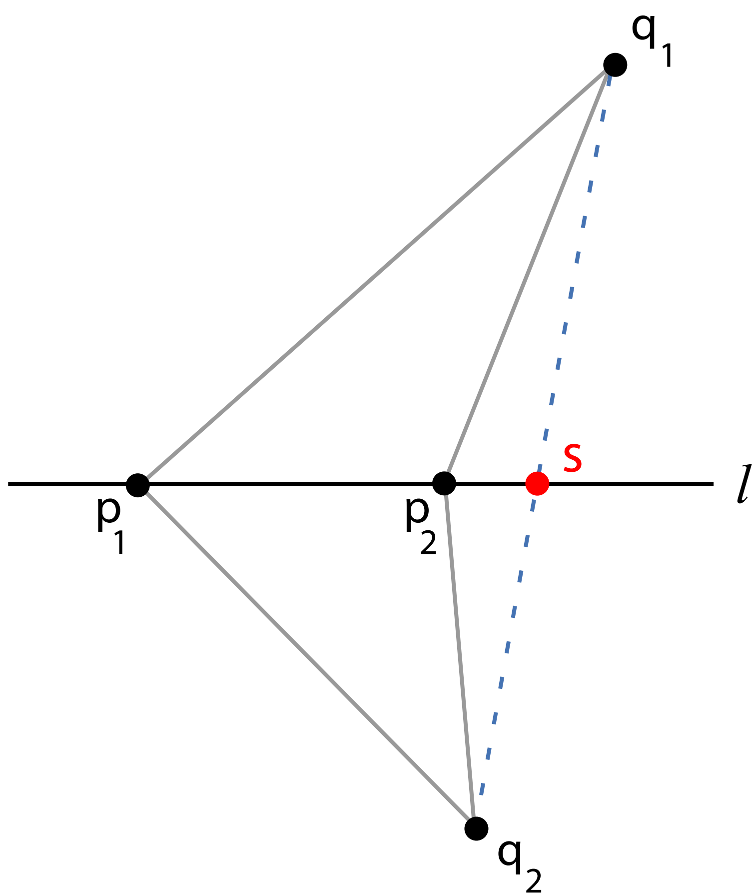
Eigenschaften optimaler Dipole

- Es existieren optimale Dipole, zwischen denen kein anderer Punkt mehr liegt



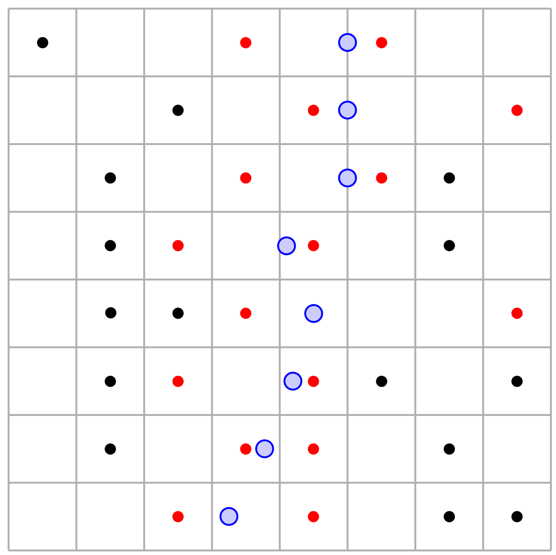
Eigenschaften optimaler Dipole

- Der Steiner-Monopol der Gerade liegt zwischen den optimalen Dipolen



Bestimmung der Kandidaten für die inneren Punkte

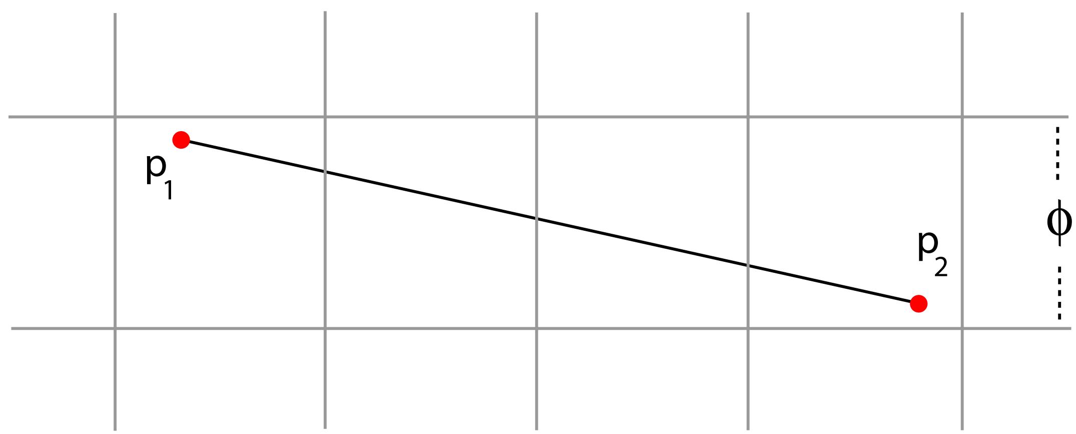
- *SOFNVD* der Punktmenge in $O(m \log m)$ Zeit
- Steiner-Monopol jeder Zeile in $O(m)$ Zeit bestimmen



- Bestimmung aller Kandidaten somit in $O(m^2)$ Zeit

Gitter-Transformation - Eigenschaften des Gitters

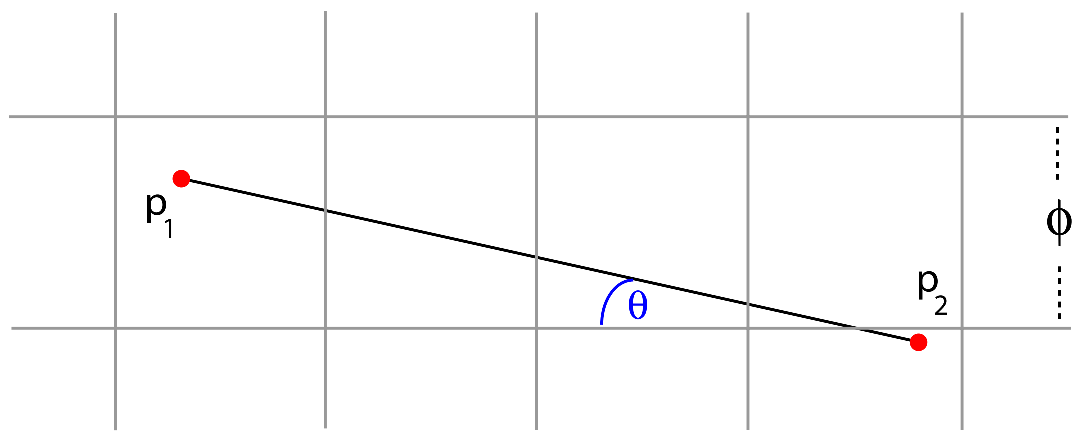
- Problem soll in Instanzen des vereinfachten Problems transformiert werden
- Ebene mit Gitter überlagern
- innere Punkte des GMDST sollen in einer Zeile liegen
- Frage: Wie muss ϕ gewählt werden?



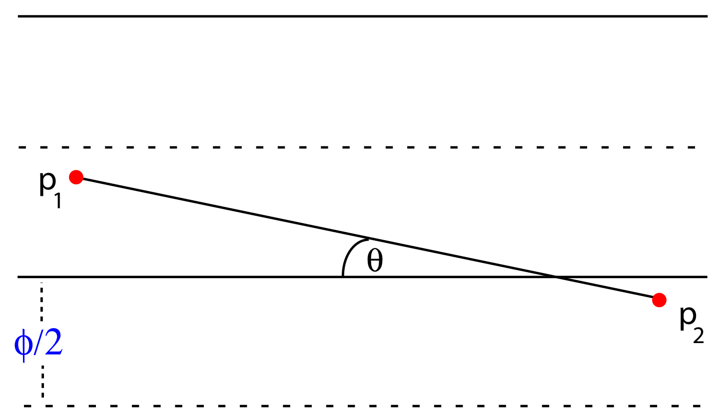
Gitter-Transformation - Eigenschaften des Gitters

- $\mathcal{D} :=$ Abstand der zwei entferntesten Punkte in P
- $\overline{p_1 p_2} \leq \mathcal{D}$ *(p_1 und p_2 sind die Dipole des GMDST)*
- wähle $\sin\theta < \frac{\phi}{\mathcal{D}}$

  $\Rightarrow p_1$ und p_2 können in Gitter mit Zeilenhöhe ϕ in einer Zeile liegen



Gitter-Transformation - Eigenschaften des Gitters



- Verlange $\sin\theta < \frac{\phi}{2\mathcal{D}}$

  $\Rightarrow$ es müssen nur 2 Gitter im Abstand $\frac{\phi}{2}$ betrachtet werden

- alle Orientierungen von p_1p_2 berücksichtigen

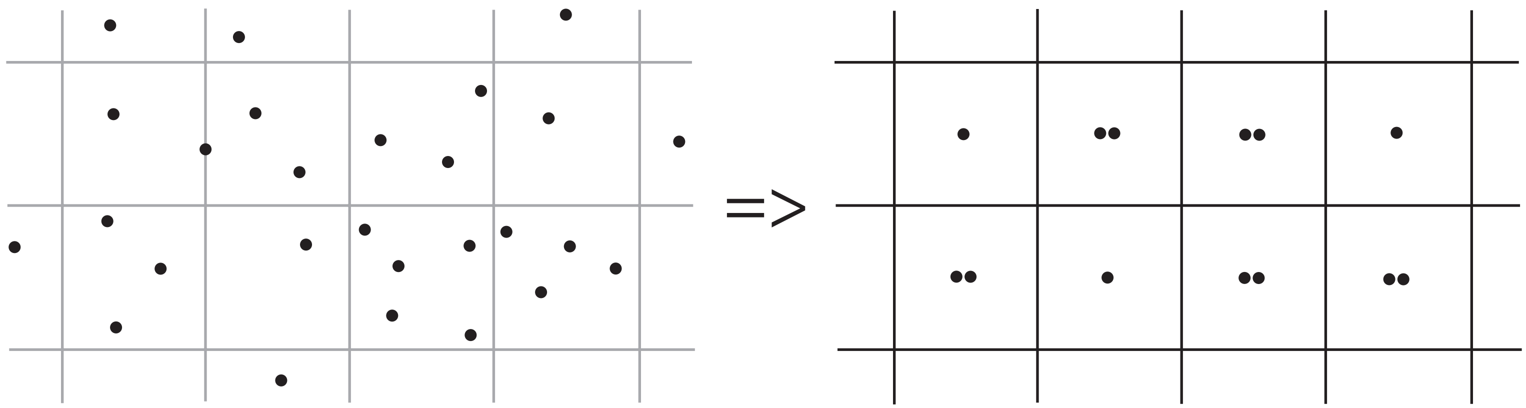
  $\Rightarrow$ mehrere Orientierungen des Gitters

- Anzahl der Gitter-Orientierungen: $\frac{\pi}{\arcsin(\phi/2\mathcal{D})} = O(\frac{\mathcal{D}}{\phi})$

Gitter-Transformation

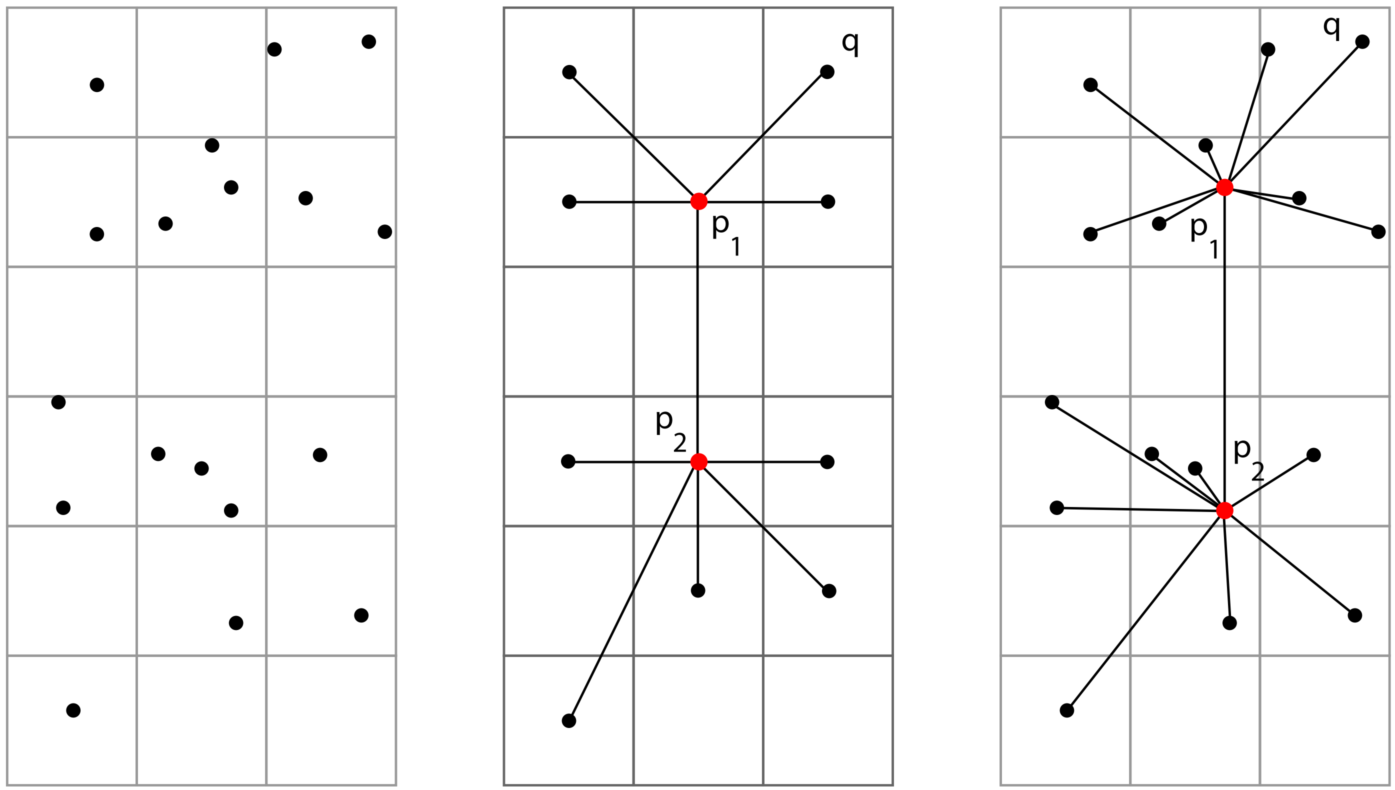
Für alle Orientierungen:

- Instanz des vereinfachten Problems erzeugen



- Vereinfachtes Problem lösen
- Spannbaum für Ursprungsproblem aus Lösung erzeugen

Rücktransformation

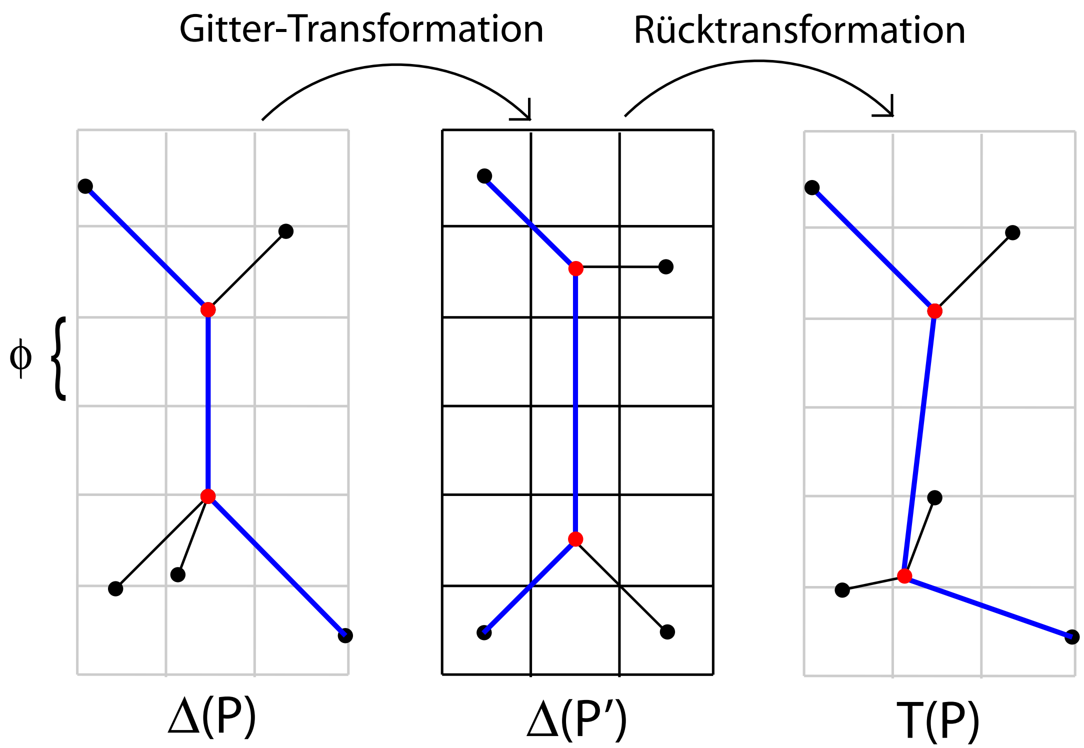


Ausgangsproblem

Lösung vereinfachtes Problem

Näherungslösung Ausgangsproblem

Güte des erzeugten Spannbaums



$$
\begin{aligned}
|\Delta(P')| &\leq |\Delta(P)| + 3\sqrt{2}\phi \\
|\mathcal{T}(P)| &\leq |\Delta(P')| + 3\sqrt{2}\phi \\
\Rightarrow |\mathcal{T}(P)| &\leq |\Delta(P)| + 6\sqrt{2}\phi
\end{aligned}
$$

Satz 1. *Gegeben eine Menge P von n Punkten in der Ebene, existiert ein Algorithmus, der für jedes $\varepsilon > 0$ einen $(1+\varepsilon)$-approximativen GMDST von P in Laufzeit $O(\varepsilon^{-3}+n)$ und mit Speicherbedarf $O(n)$ bestimmt.*

Beweis zu Satz 1

- $\varepsilon < \frac{1}{n}$

  $\Rightarrow$ Algorithmus von Ho et al.,

  $\Rightarrow O(n^3) \subseteq O(\varepsilon^{-3})$ Laufzeit

  $\Rightarrow O(n)$ Speicherbedarf

- n größer als Anzahl der Gitterquadrate

  $\Rightarrow$ initiale Gitter-Transformation

  $\Rightarrow$ maximal $2m^2$ Punkte

  $\Rightarrow$ Verdopplung des additiven Fehlers zu $12\sqrt{2}\phi$

Beweis zu Satz 1 - Güte des erzeugten Spannbaums

- $\mathcal{D} \leq |\Delta(P)|$
- Setze $\phi = \frac{\mathcal{D}\varepsilon}{12\sqrt{2}}$
- Für mindestens eine der Orientierungen gilt:

$$\frac{|\mathcal{T}(P)|}{|\Delta(P)|} \leq \frac{|\Delta(P)| + 12\sqrt{2}\phi}{|\Delta(P)|} \leq 1 + \frac{12\sqrt{2}}{|\Delta(P)|} \cdot \frac{\mathcal{D}\varepsilon}{12\sqrt{2}} \leq (1+\varepsilon)$$

Beweis zu Satz 1 - Komplexität

- Berechne $\mathcal{D}$ als maximale vertikale oder horizontale Distanz: Zeit $O(n)$
- Bestimmung von ϕ: Zeit $O(1)$
- Verwende $\mathcal{D} \times \mathcal{D}$ Bounding Box
- Anzahl der Gitterquadrate:

$$\frac{\mathcal{D} \times \mathcal{D}}{\phi^2} = \frac{\mathcal{D} \times \mathcal{D}}{(\mathcal{D}\varepsilon)^2/(12\sqrt{2})^2} = \frac{288}{\varepsilon^2} = O(\varepsilon^{-2})$$

- erste Gitter-Transformation: Zeit $O(\varepsilon^{-1} + n)$
- weitere Gitter-Transformationen: Zeit $O(\varepsilon^{-2})$

Beweis zu Satz 1 - Komplexität

- Rücktransformation des RGMDST: Zeit $O(\varepsilon^{-2} + n)$
- $O(\varepsilon^{-2})$ Laufzeit für jede Orientierung
- $O(\varepsilon^{-2} + n)$ initiale und finale Laufzeit
- Anzahl der Gitterorientierungen:

$$O\left(\frac{\mathcal{D}}{\phi}\right) = O\left(\frac{\mathcal{D} \cdot 12\sqrt{2}}{\mathcal{D}\varepsilon}\right) = O(\varepsilon^{-1})$$

$\Rightarrow$ Laufzeit $O(\varepsilon^{-3} + n)$

Zusammenfassung

- Bekannte exakte Algorithmen haben annähernd kubische Laufzeit
- Es ist keine untere Schranke des Problems bekannt
- Vorgestellter Approximationsalgorithmus ist linear in der Größe der Eingabe
- Laufzeitvorteil durch Reduktion auf $O(\varepsilon^{-1})$ einfacherer Probleme
- Lösung der vereinfachten Probleme in Zeit $O(\varepsilon^{-2})$
- Problemgröße n beeinflusst asymptotische Laufzeit nur bei der initialen Gitter-Transformation und bei der Rücktransformation
- Erstaunlicherweise dennoch Gütegarantie mit Faktor $(1 + \varepsilon)$