



Vorlesung Algorithmische Geometrie

Geradenarrangements und Dualität von Punkten und Geraden

INSTITUT FÜR THEORETISCHE INFORMATIK · FAKULTÄT FÜR INFORMATIK

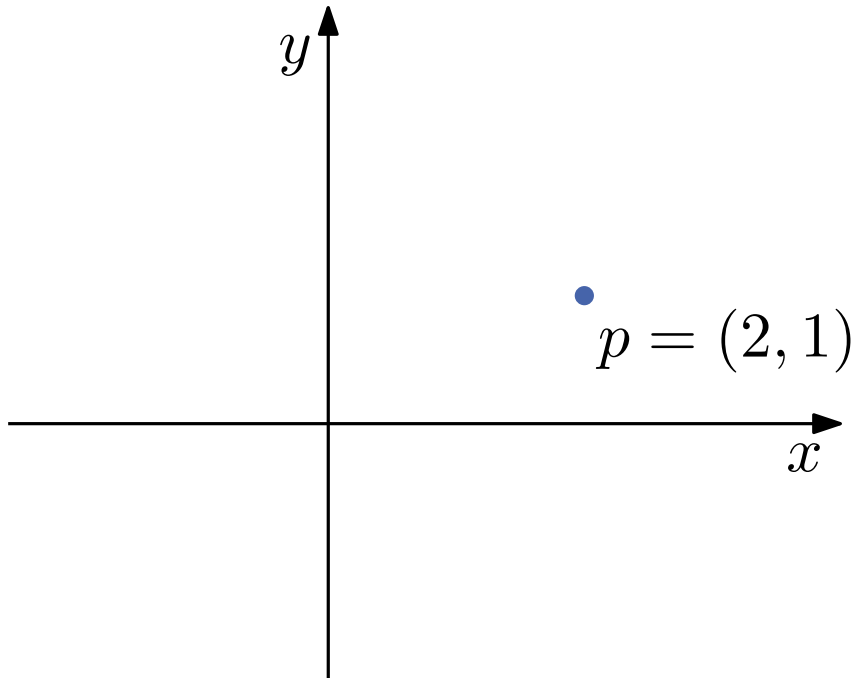


Martin Nöllenburg
17.06.2014



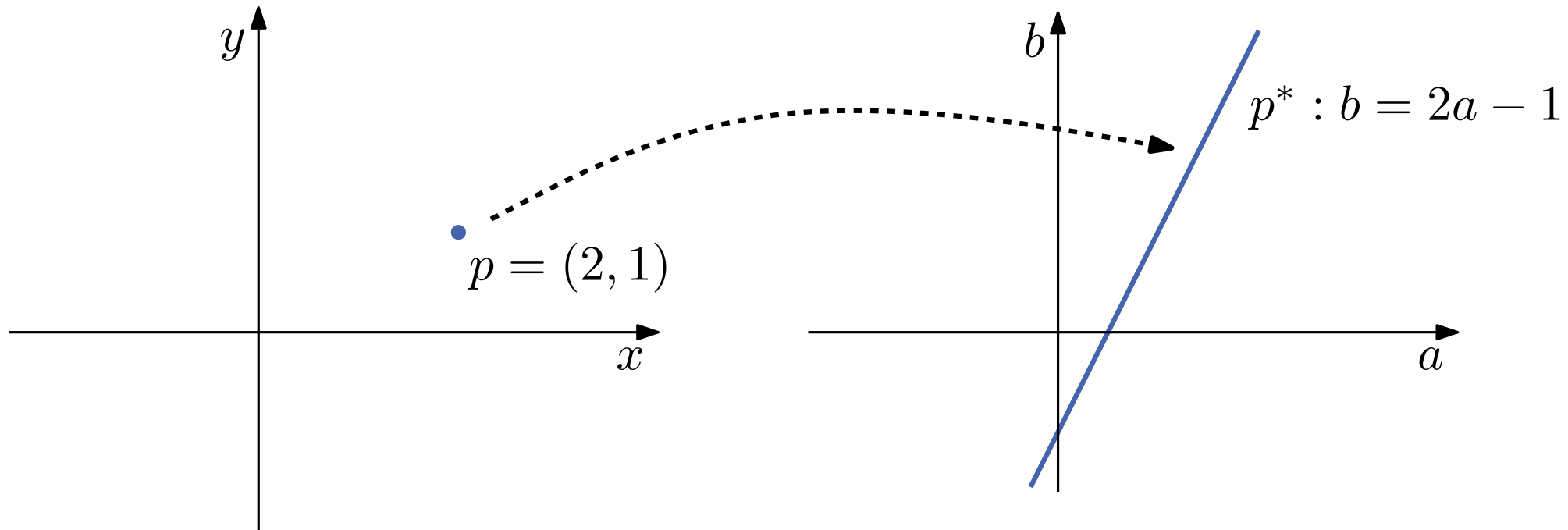
Dualitätsabbildung

Bisher haben wir Dualität für planare Graphen bzw. Voronoi-Diagramme und Delaunay-Triangulierungen gesehen. Auch Punkte und Geraden können **dual** zueinander sein.



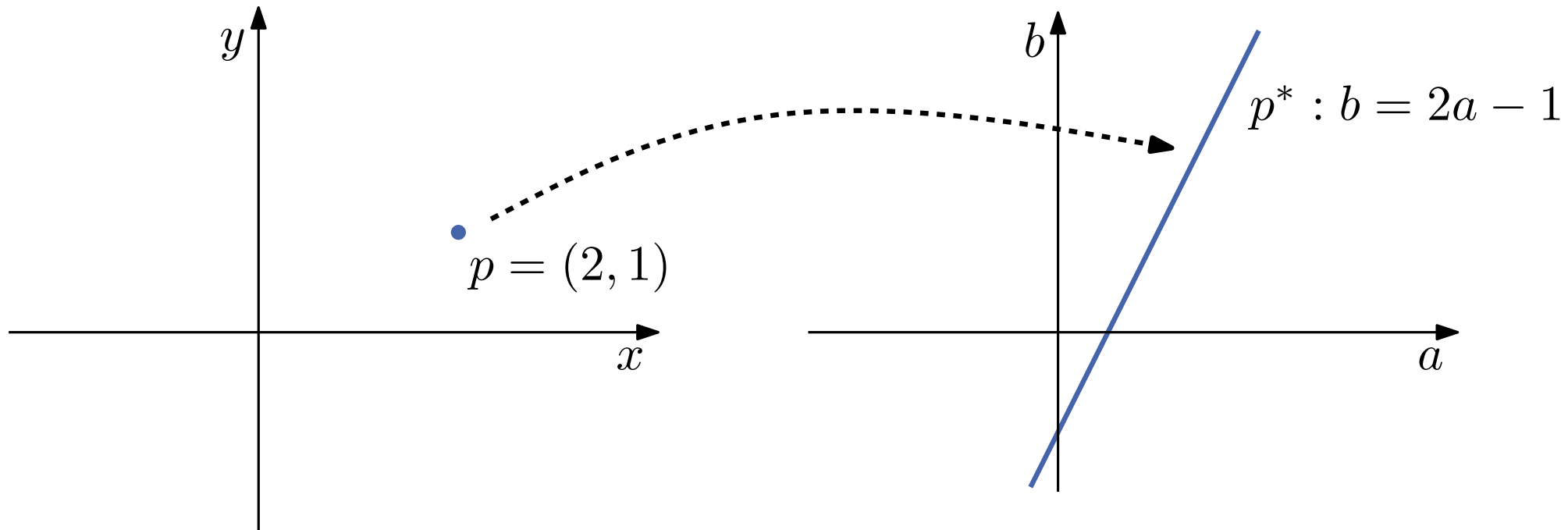
Dualitätsabbildung

Bisher haben wir Dualität für planare Graphen bzw. Voronoi-Diagramme und Delaunay-Triangulierungen gesehen. Auch Punkte und Geraden können **dual** zueinander sein.



Dualitätsabbildung

Bisher haben wir Dualität für planare Graphen bzw. Voronoi-Diagramme und Delaunay-Triangulierungen gesehen. Auch Punkte und Geraden können **dual** zueinander sein.

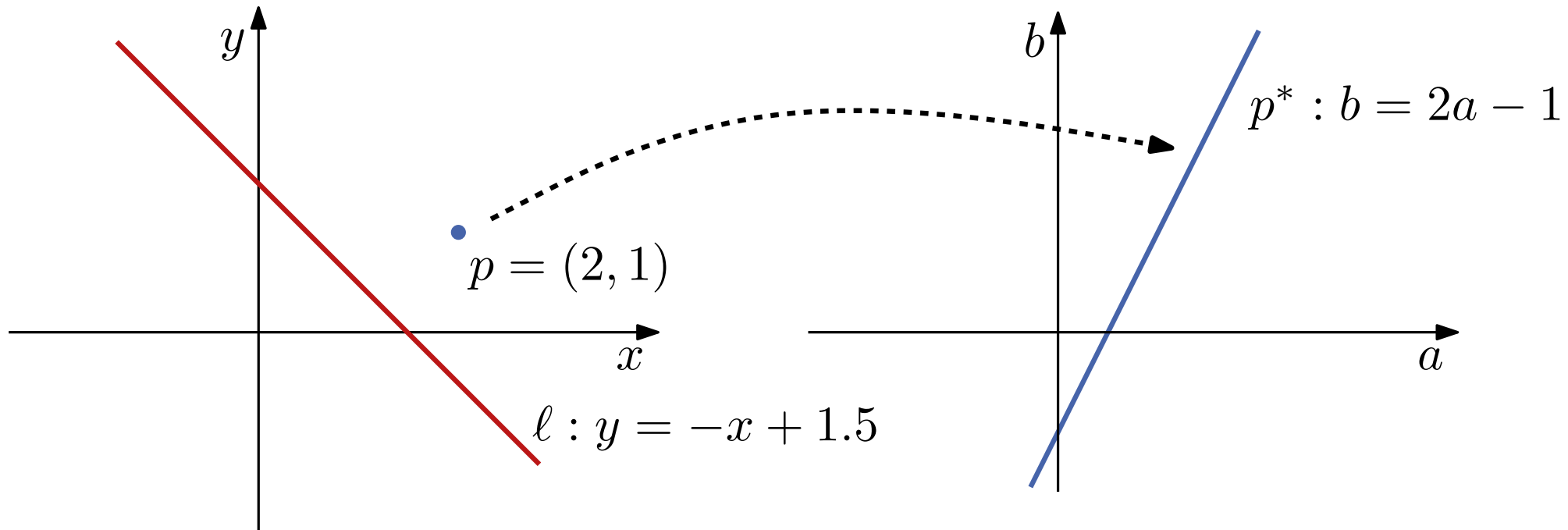


Def.: Die Dualitätsabbildung $(\cdot)^*$ ist definiert durch

$$p = (p_x, p_y) \quad \mapsto \quad p^* : b = p_x a - p_y$$

Dualitätsabbildung

Bisher haben wir Dualität für planare Graphen bzw. Voronoi-Diagramme und Delaunay-Triangulierungen gesehen. Auch Punkte und Geraden können **dual** zueinander sein.

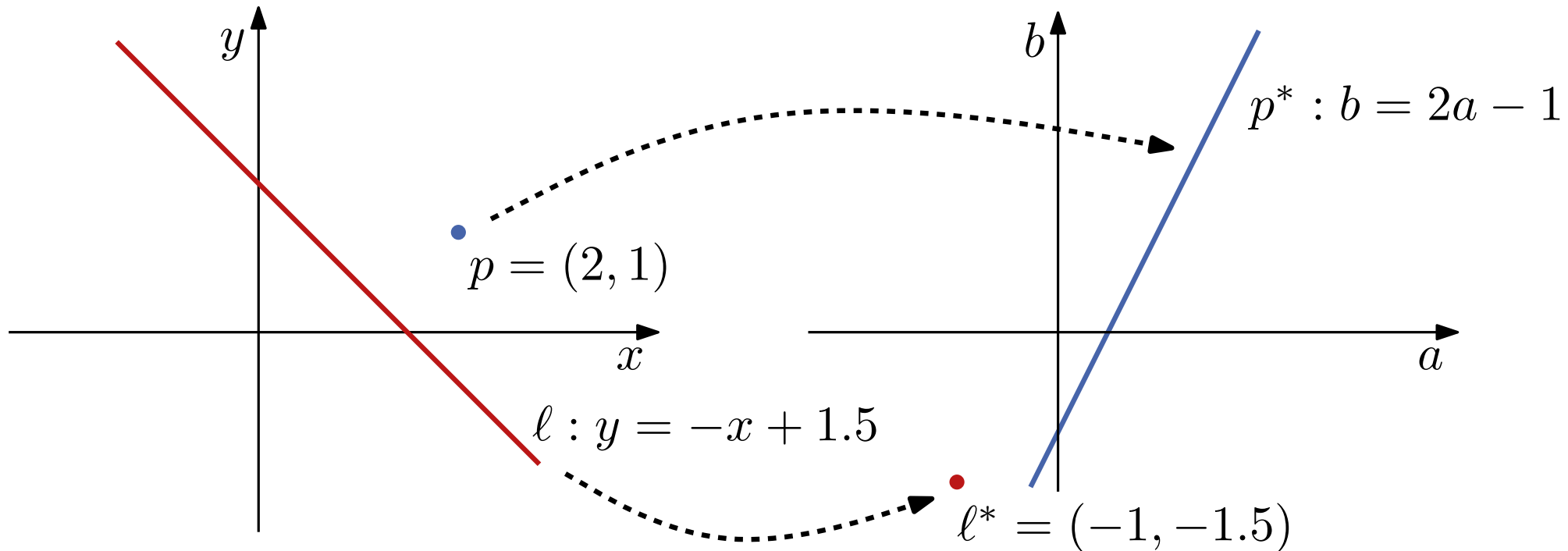


Def.: Die Dualitätsabbildung $(\cdot)^*$ ist definiert durch

$$p = (p_x, p_y) \quad \mapsto \quad p^* : b = p_x a - p_y$$

Dualitätsabbildung

Bisher haben wir Dualität für planare Graphen bzw. Voronoi-Diagramme und Delaunay-Triangulierungen gesehen. Auch Punkte und Geraden können **dual** zueinander sein.

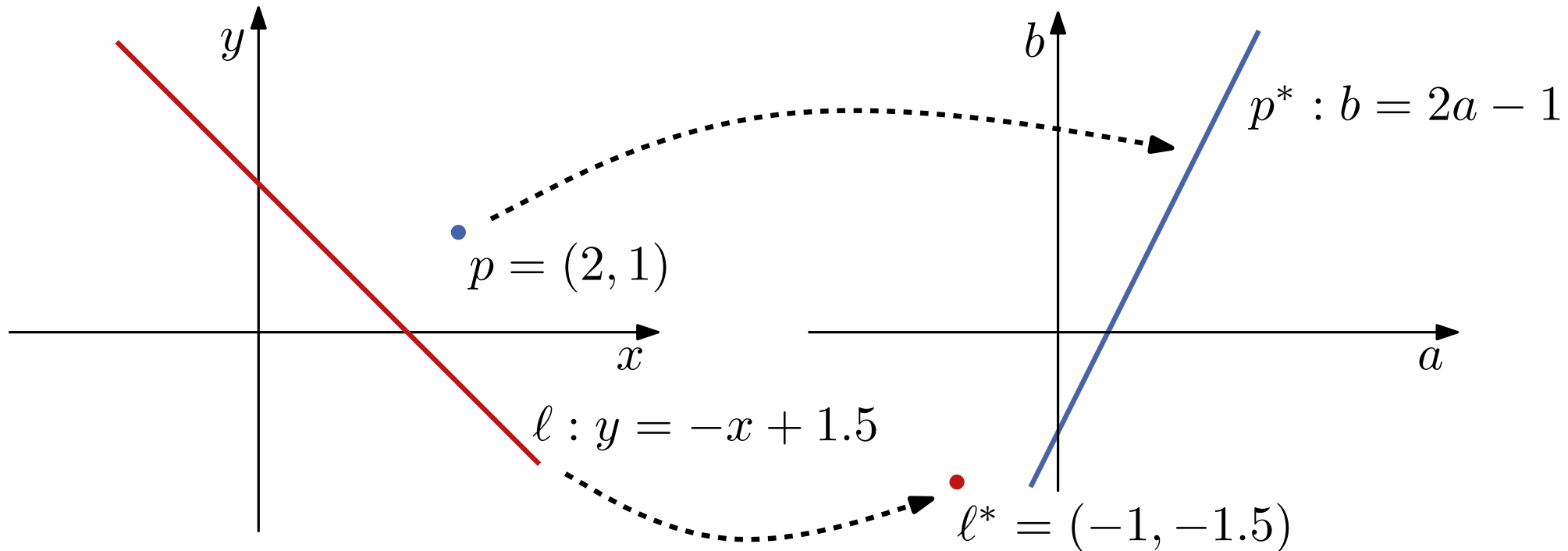


Def.: Die Dualitätsabbildung $(\cdot)^*$ ist definiert durch

$$p = (p_x, p_y) \quad \mapsto \quad p^* : b = p_x a - p_y$$

Dualitätsabbildung

Bisher haben wir Dualität für planare Graphen bzw. Voronoi-Diagramme und Delaunay-Triangulierungen gesehen. Auch Punkte und Geraden können **dual** zueinander sein.



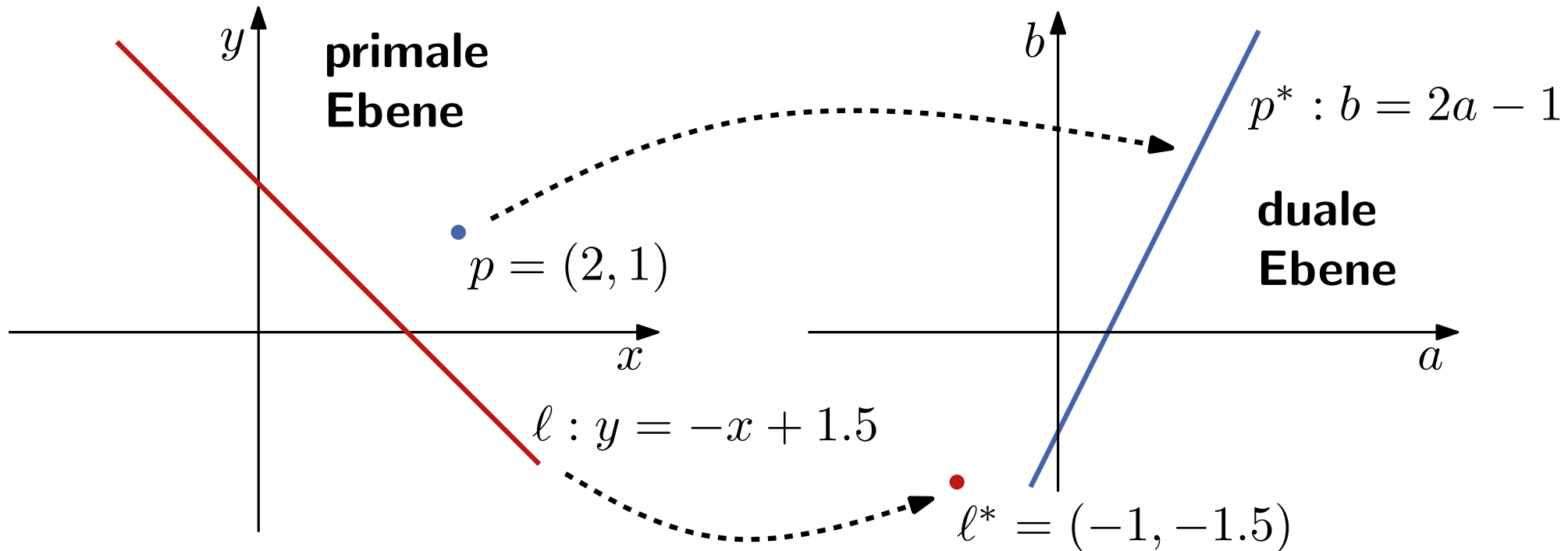
Def.: Die Dualitätsabbildung $(\cdot)^*$ ist definiert durch

$$p = (p_x, p_y) \quad \mapsto \quad p^* : b = p_x a - p_y$$

$$\ell : y = m x + c \quad \mapsto \quad \ell^* = (m, -c)$$

Dualitätsabbildung

Bisher haben wir Dualität für planare Graphen bzw. Voronoi-Diagramme und Delaunay-Triangulierungen gesehen. Auch Punkte und Geraden können **dual** zueinander sein.



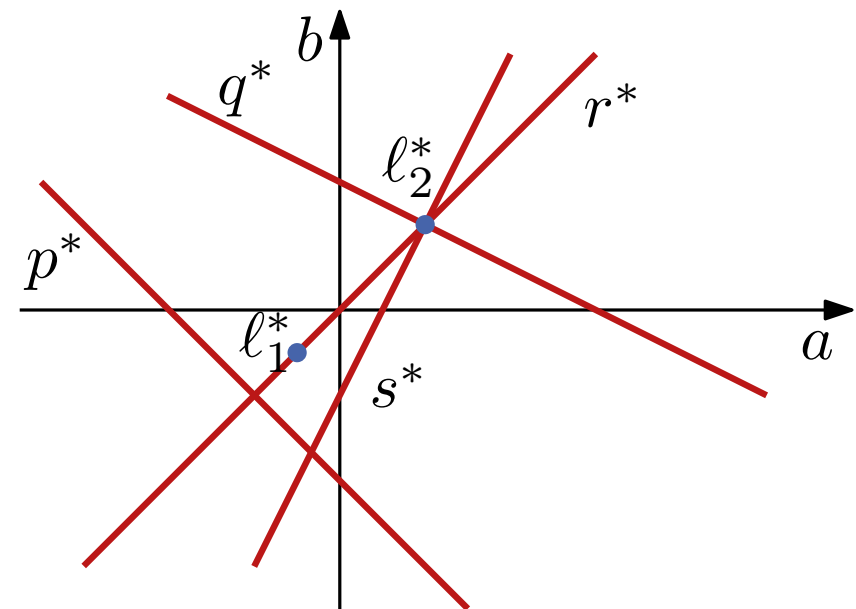
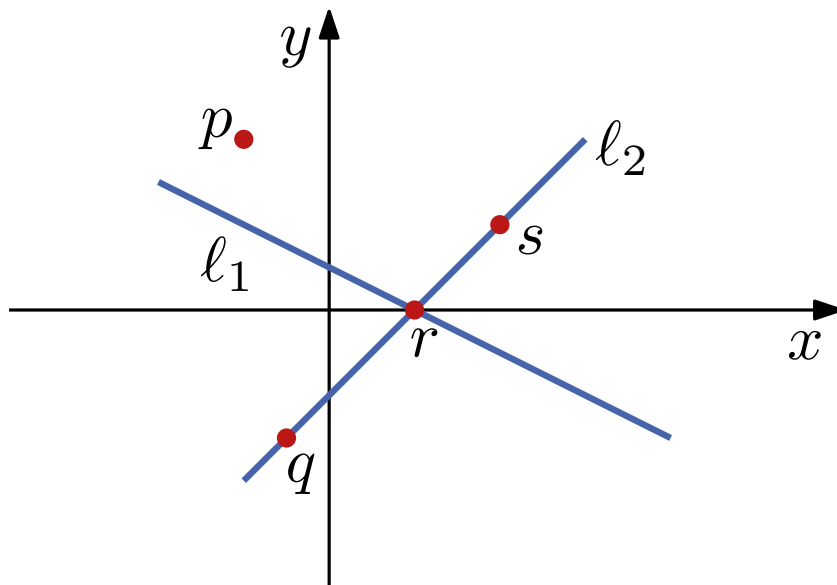
Def.: Die Dualitätsabbildung $(\cdot)^*$ ist definiert durch

$$p = (p_x, p_y) \quad \mapsto \quad p^* : b = p_x a - p_y$$

$$\ell : y = m x + c \quad \mapsto \quad \ell^* = (m, -c)$$

Lemma 1: Es gelten die folgenden Eigenschaften

- $(p^*)^* = p$ und $(l^*)^* = l$
- p liegt unter/auf/über $l \Leftrightarrow p^*$ läuft über/auf/unter l^*
- l_1 und l_2 schneiden sich in r
 $\Leftrightarrow r^*$ geht durch l_1^* und l_2^*
- q, r, s kollinear
 $\Leftrightarrow q^*, r^*, s^*$ schneiden sich in gemeinsamem Punkt



Lemma 1: Es gelten die folgenden Eigenschaften

- $(p^*)^* = p$ und $(l^*)^* = l$
- p liegt unter/auf/über $l \Leftrightarrow p^*$ läuft über/auf/unter l^*
- l_1 und l_2 schneiden sich in r
 $\Leftrightarrow r^*$ geht durch l_1^* und l_2^*
- q, r, s kollinear
 $\Leftrightarrow q^*, r^*, s^*$ schneiden sich in gemeinsamem Punkt

Wie sieht das duale Objekt zu einer Strecke $s = \overline{pq}$ aus?

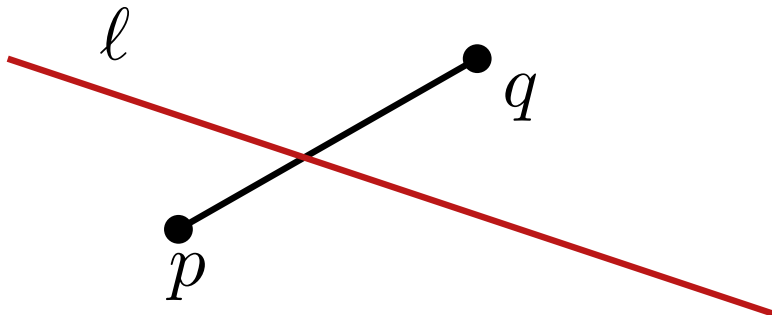
Welche duale Beziehung gilt für eine Gerade l , die s schneidet?

Lemma 1: Es gelten die folgenden Eigenschaften

- $(p^*)^* = p$ und $(l^*)^* = l$
- p liegt unter/auf/über $l \Leftrightarrow p^*$ läuft über/auf/unter l^*
- l_1 und l_2 schneiden sich in r
 $\Leftrightarrow r^*$ geht durch l_1^* und l_2^*
- q, r, s kollinear
 $\Leftrightarrow q^*, r^*, s^*$ schneiden sich in gemeinsamem Punkt

Wie sieht das duale Objekt zu einer Strecke $s = \overline{pq}$ aus?

Welche duale Beziehung gilt für eine Gerade l , die s schneidet?

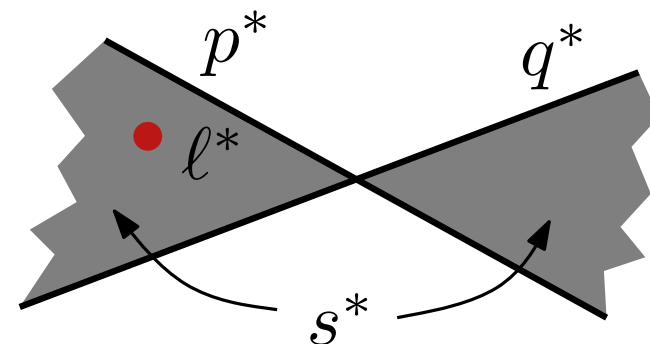
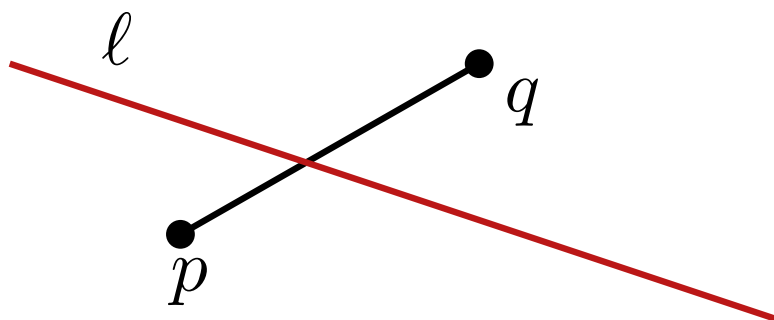


Lemma 1: Es gelten die folgenden Eigenschaften

- $(p^*)^* = p$ und $(l^*)^* = l$
- p liegt unter/auf/über $l \Leftrightarrow p^*$ läuft über/auf/unter l^*
- l_1 und l_2 schneiden sich in r
 $\Leftrightarrow r^*$ geht durch l_1^* und l_2^*
- q, r, s kollinear
 $\Leftrightarrow q^*, r^*, s^*$ schneiden sich in gemeinsamem Punkt

Wie sieht das duale Objekt zu einer Strecke $s = \overline{pq}$ aus?

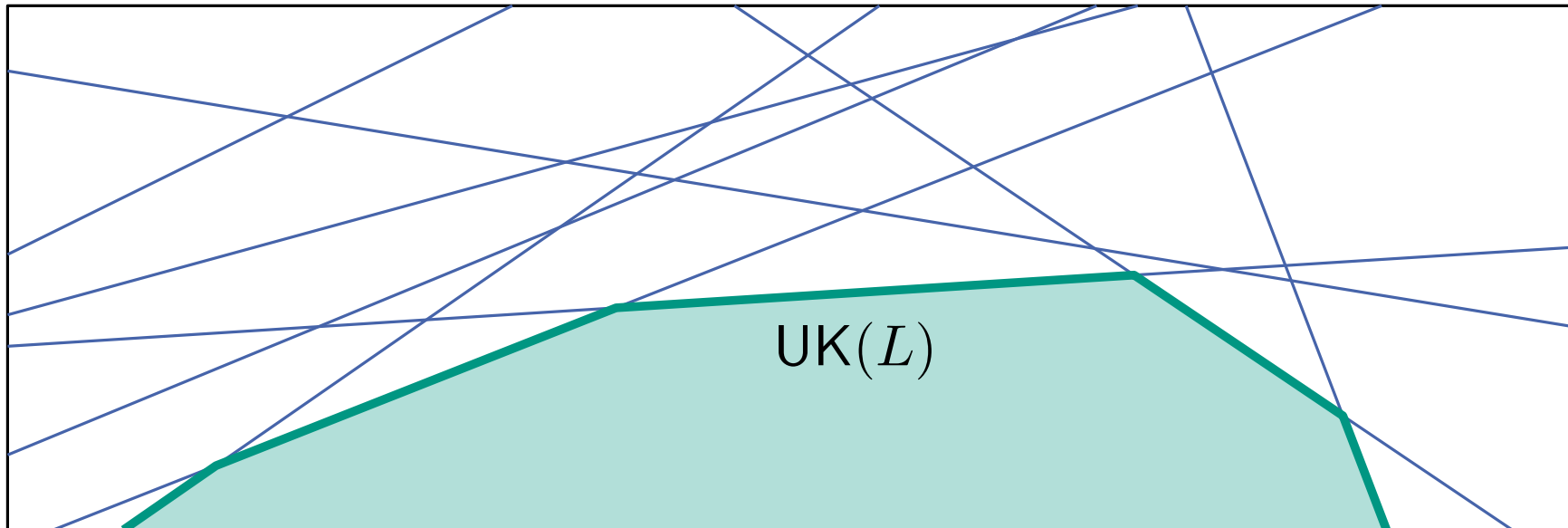
Welche duale Beziehung gilt für eine Gerade l , die s schneidet?



Dualität macht geometrische Probleme weder leichter noch schwerer; sie bietet einen anderen (oft hilfreichen) Blickwinkel!

Wir werden zwei Beispiele genauer betrachten:

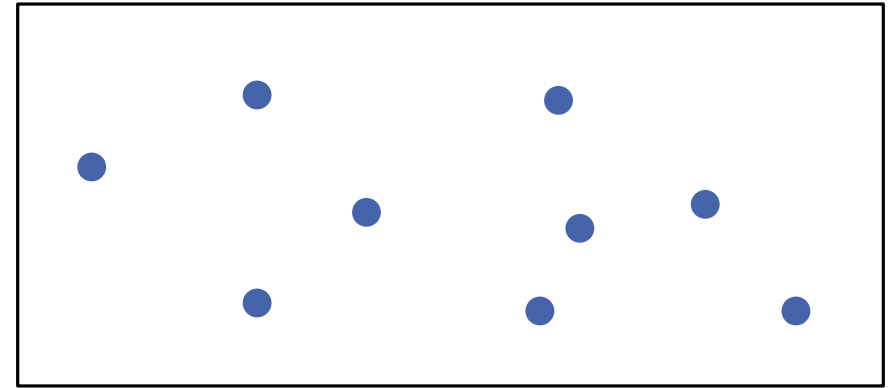
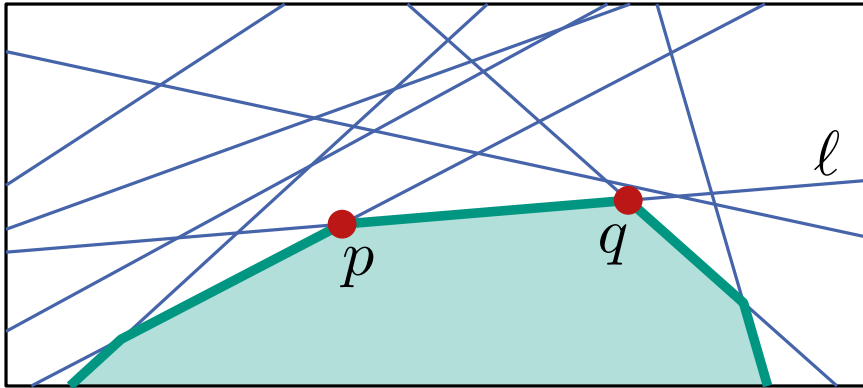
- untere/obere Konturen von Geradenarrangements
- kleinstes Dreieck in einer Punktmenge



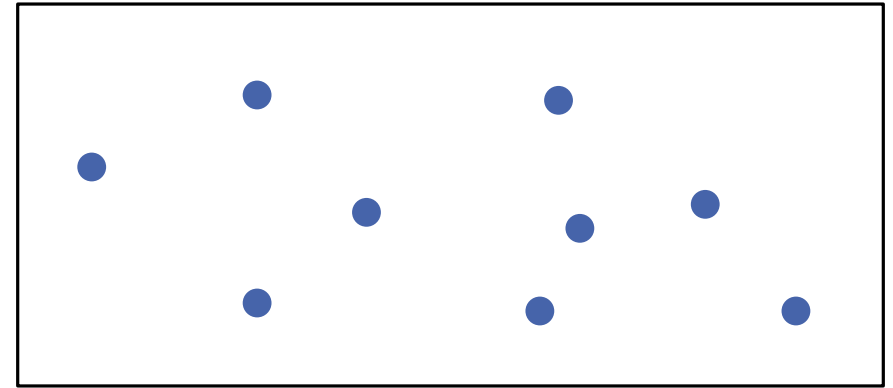
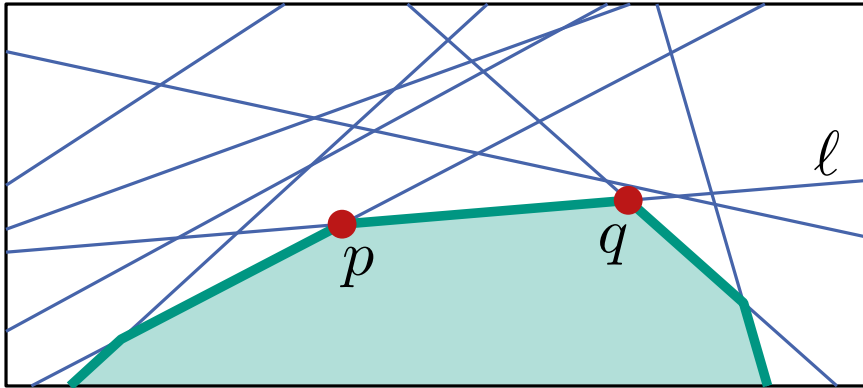
Def.: Für eine Menge L von Geraden ist die untere Kontur $UK(L)$ von L die Menge aller Punkte aus $\cup_{\ell \in L} \ell$, die nicht oberhalb einer Geraden aus L liegen.

Möglichkeiten zur Berechnung der unteren Kontur:

- Algorithmus `IntersectHalbplanes` aus 4. VL
- Betrachte das duale Problem für $L^* = \{\ell^* \mid \ell \in L\}$

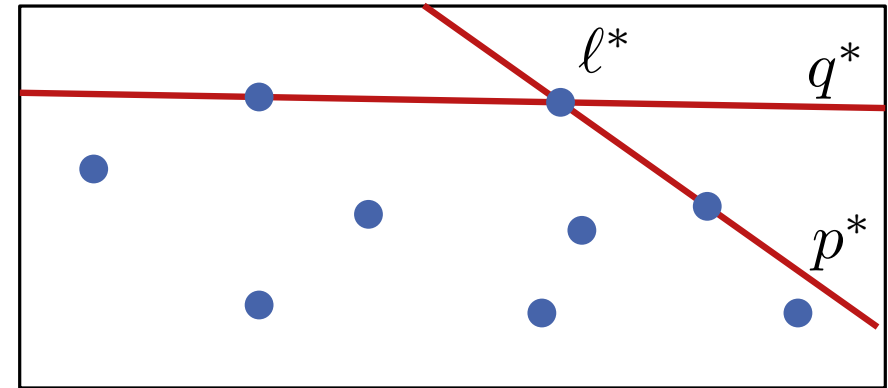
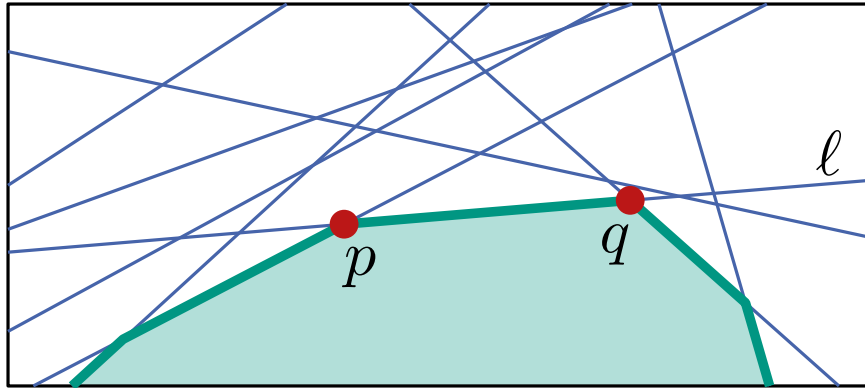


Wann erscheint ℓ als Strecke \overline{pq} auf $UK(L)$?



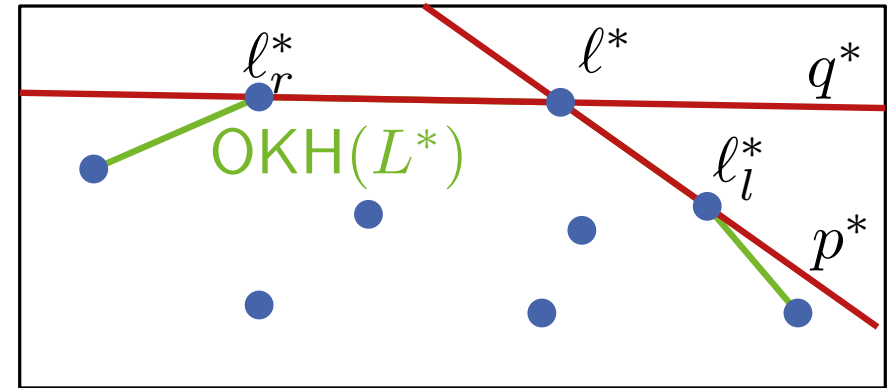
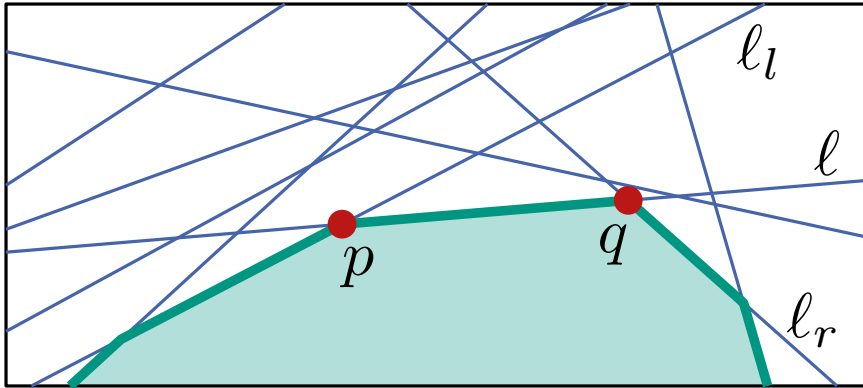
Wann erscheint ℓ als Strecke \overline{pq} auf $UK(L)$?

- p und q liegen unterhalb aller Geraden in L



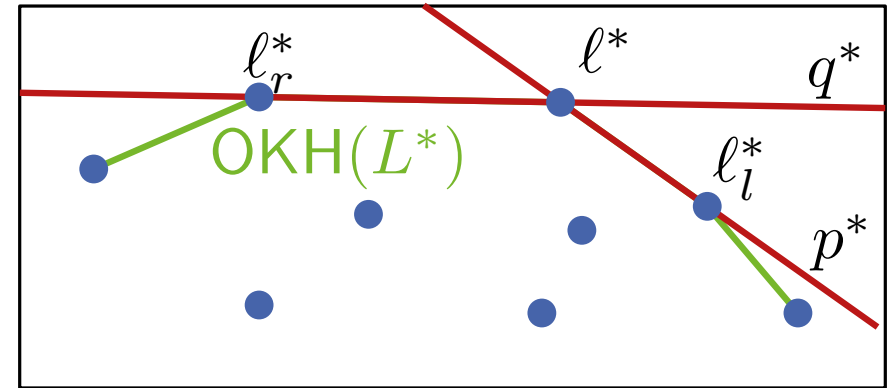
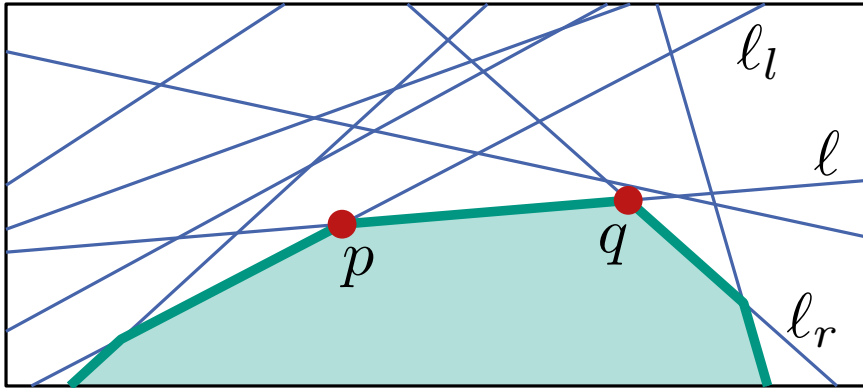
Wann erscheint l als Strecke \overline{pq} auf $UK(L)$?

- p und q liegen unterhalb aller Geraden in L
- p^* und q^* liegen oberhalb aller Punkte aus L^*



Wann erscheint ℓ als Strecke \overline{pq} auf $UK(L)$?

- p und q liegen unterhalb aller Geraden in L
- p^* und q^* liegen oberhalb aller Punkte aus L^*
 \Rightarrow liegen benachbart auf oberer konvexer Hülle $OKH(L^*)$
- Schnittpunkt von p^* und q^* ist ℓ^* , Knoten von $OKH(L^*)$



Wann erscheint ℓ als Strecke \overline{pq} auf $UK(L)$?

- p und q liegen unterhalb aller Geraden in L
- p^* und q^* liegen oberhalb aller Punkte aus L^*
 \Rightarrow liegen benachbart auf oberer konvexer Hülle $OKH(L^*)$
- Schnittpunkt von p^* und q^* ist ℓ^* , Knoten von $OKH(L^*)$

Lemma 2: Die Geraden auf $UK(L)$ von rechts nach links entsprechen den Knoten auf $OKH(L^*)$ von links nach rechts.

Berechnung der Kontur

- Algorithmus zur Berechnung der oberen konvexen Hülle mit Laufzeit $O(n \log n)$ (s. erste VL zu konvexer Hülle)

Berechnung der Kontur

- Algorithmus zur Berechnung der oberen konvexen Hülle mit Laufzeit $O(n \log n)$ (s. erste VL zu konvexer Hülle)
- Primale Geraden zu Punkten auf $OKH(L^*)$ in umgekehrter Reihenfolge bilden $UK(L)$

Berechnung der Kontur

- Algorithmus zur Berechnung der oberen konvexen Hülle mit Laufzeit $O(n \log n)$ (s. erste VL zu konvexer Hülle)
- Primale Geraden zu Punkten auf $OKH(L^*)$ in umgekehrter Reihenfolge bilden $UK(L)$
- Analog: obere Kontur von $L \hat{=} \text{untere konvexe Hülle von } L^*$

Berechnung der Kontur

- Algorithmus zur Berechnung der oberen konvexen Hülle mit Laufzeit $O(n \log n)$ (s. erste VL zu konvexer Hülle)
- Primale Geraden zu Punkten auf $OKH(L^*)$ in umgekehrter Reihenfolge bilden $UK(L)$
- Analog: obere Kontur von $L \hat{=} \text{untere konvexe Hülle von } L^*$

Wie lässt sich das nutzen um den Schnitt von n Halbebenen zu berechnen?

- Algorithmus zur Berechnung der oberen konvexen Hülle mit Laufzeit $O(n \log n)$ (s. erste VL zu konvexer Hülle)
- Primale Geraden zu Punkten auf $OKH(L^*)$ in umgekehrter Reihenfolge bilden $UK(L)$
- Analog: obere Kontur von $L \hat{=} \text{untere konvexe Hülle von } L^*$

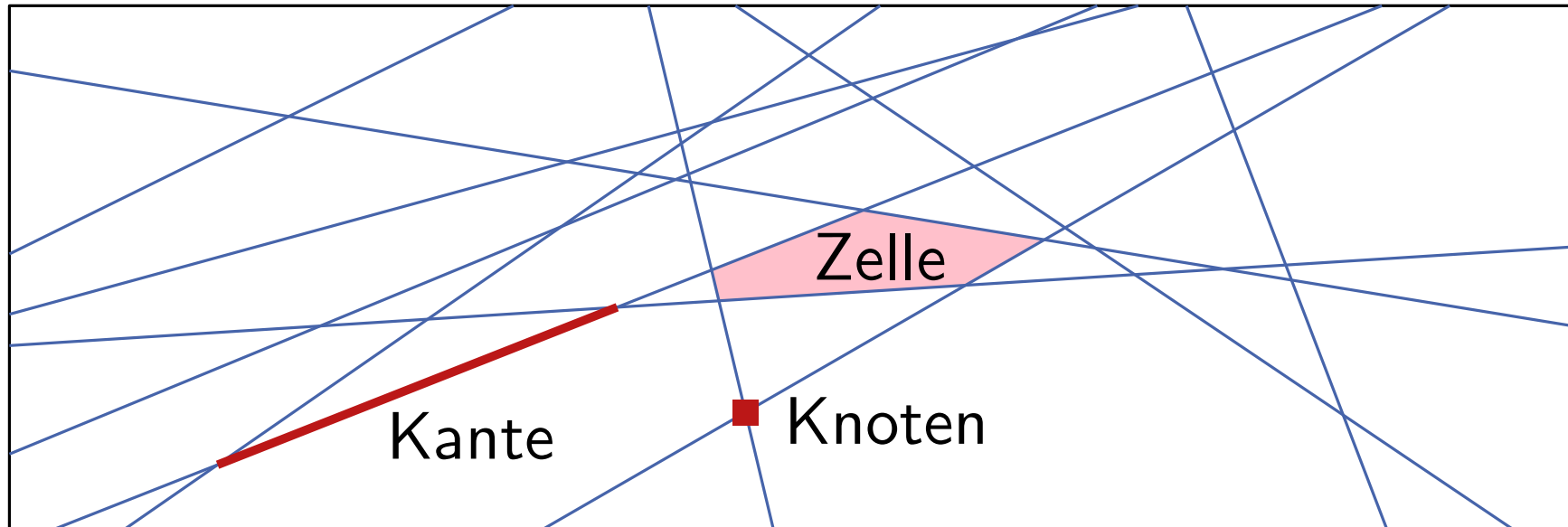
Wie lässt sich das nutzen um den Schnitt von n Halbebenen zu berechnen?

- untere Kontur der von oben begrenzenden Geraden
- obere Kontur der von unten begrenzenden Geraden
- Schnitt der beiden konvexen Regionen (s. 4. VL)
- insgesamt $O(n \log n)$ Laufzeit

Zwischenfrage:

Wie testet man ob n Punkte in allgemeiner Lage sind?

Wie findet man eine maximal große Menge kollinearere Punkte?



Def.: Eine Menge L von Geraden definiert eine Unterteilung $\mathcal{A}(L)$ der Ebene (das **Geradenarrangement**) in Knoten, Kanten und Zellen (tlws. unbeschränkt).
 $\mathcal{A}(L)$ heißt **einfach**, wenn keine drei Geraden durch einen Punkt gehen und keine zwei Geraden parallel sind.

Komplexität von $\mathcal{A}(L)$

Die kombinatorische Komplexität von $\mathcal{A}(L)$ ist die Zahl der Knoten, Kanten und Zellen. Es gilt:

Satz 1: Sei $\mathcal{A}(L)$ ein einfaches Arrangement von n Geraden. Dann hat $\mathcal{A}(L)$ $\binom{n}{2}$ Knoten, n^2 Kanten und $\binom{n}{2} + n + 1$ Zellen.

Komplexität von $\mathcal{A}(L)$

Die kombinatorische Komplexität von $\mathcal{A}(L)$ ist die Zahl der Knoten, Kanten und Zellen. Es gilt:

Satz 1: Sei $\mathcal{A}(L)$ ein einfaches Arrangement von n Geraden. Dann hat $\mathcal{A}(L)$ $\binom{n}{2}$ Knoten, n^2 Kanten und $\binom{n}{2} + n + 1$ Zellen.

Datenstruktur für $\mathcal{A}(L)$:

- füge bounding box aller Knoten hinzu (s. Übung)
→ planar eingebetteter Graph G
- doppelt-verkettete Kantenliste für G

Komplexität von $\mathcal{A}(L)$

Die kombinatorische Komplexität von $\mathcal{A}(L)$ ist die Zahl der Knoten, Kanten und Zellen. Es gilt:

Satz 1: Sei $\mathcal{A}(L)$ ein einfaches Arrangement von n Geraden. Dann hat $\mathcal{A}(L)$ $\binom{n}{2}$ Knoten, n^2 Kanten und $\binom{n}{2} + n + 1$ Zellen.

Datenstruktur für $\mathcal{A}(L)$:

- füge bounding box aller Knoten hinzu (s. Übung)
→ planar eingebetteter Graph G
- doppelt-verkettete Kantenliste für G

Kennen wir schon eine Möglichkeit um $\mathcal{A}(L)$ zu berechnen?

Konstruktion von $\mathcal{A}(L)$

Input: Geraden $L = \{\ell_1, \dots, \ell_n\}$

Output: DCEL \mathcal{D} für $\mathcal{A}(L)$

initialisiere \mathcal{D} für bounding box B der Knoten von $\mathcal{A}(L)$

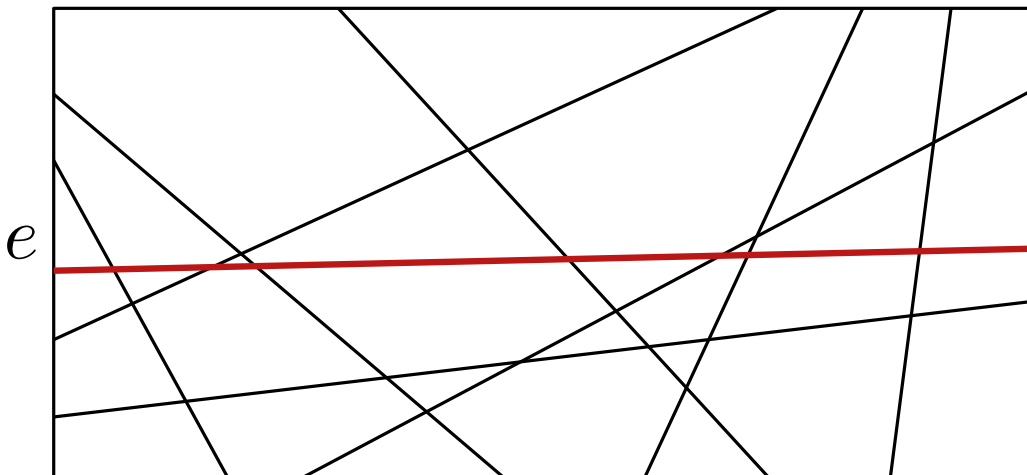
for $i \leftarrow 1$ **to** n **do**

 finde linken Schnittpunkt von ℓ_i mit Kante e von B

$f \leftarrow$ innere Zelle inzident zu e

while $f \neq$ äußere Zelle **do**

 zerteile f , update \mathcal{D} und setze f auf nächste Zelle



Konstruktion von $\mathcal{A}(L)$

Input: Geraden $L = \{\ell_1, \dots, \ell_n\}$

Output: DCEL \mathcal{D} für $\mathcal{A}(L)$

initialisiere \mathcal{D} für bounding box B der Knoten von $\mathcal{A}(L)$

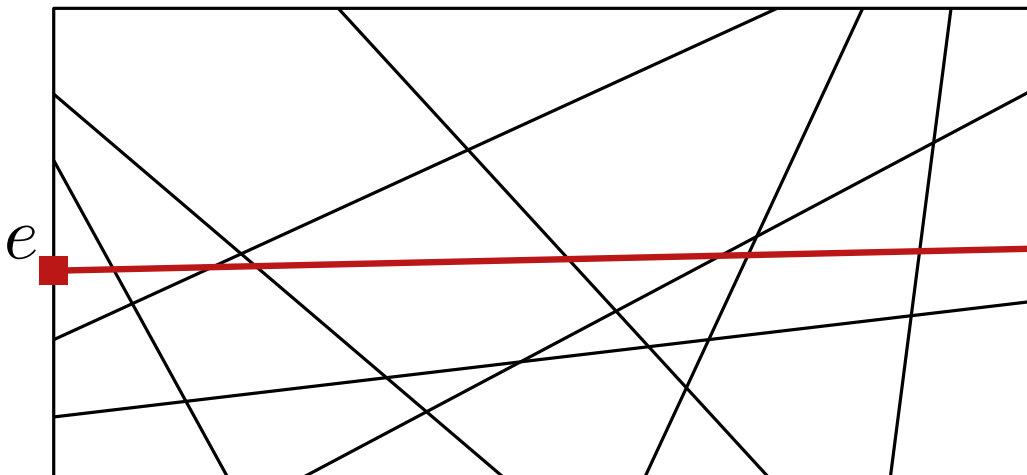
for $i \leftarrow 1$ **to** n **do**

 finde linken Schnittpunkt von ℓ_i mit Kante e von B

$f \leftarrow$ innere Zelle inzident zu e

while $f \neq$ äußere Zelle **do**

 zerteile f , update \mathcal{D} und setze f auf nächste Zelle



Konstruktion von $\mathcal{A}(L)$

Input: Geraden $L = \{\ell_1, \dots, \ell_n\}$

Output: DCEL \mathcal{D} für $\mathcal{A}(L)$

initialisiere \mathcal{D} für bounding box B der Knoten von $\mathcal{A}(L)$

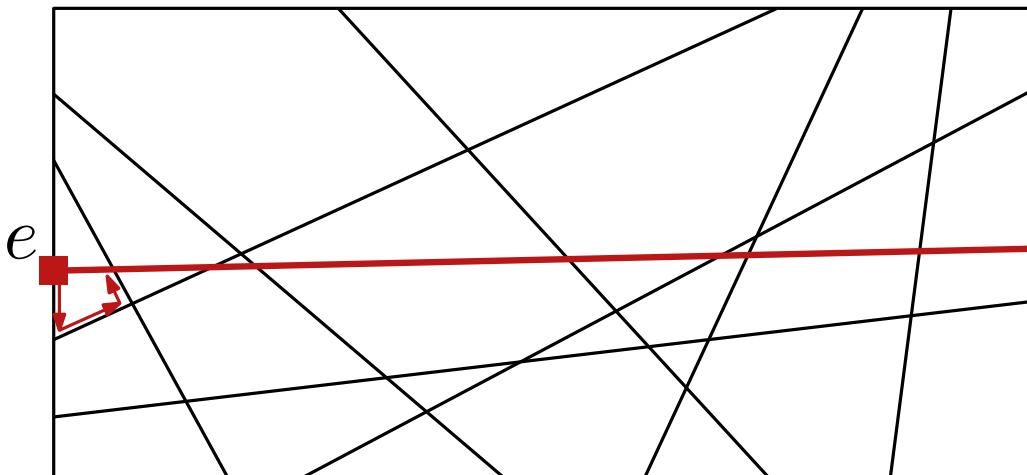
for $i \leftarrow 1$ **to** n **do**

 finde linken Schnittpunkt von ℓ_i mit Kante e von B

$f \leftarrow$ innere Zelle inzident zu e

while $f \neq$ äußere Zelle **do**

 zerteile f , update \mathcal{D} und setze f auf nächste Zelle



Konstruktion von $\mathcal{A}(L)$

Input: Geraden $L = \{\ell_1, \dots, \ell_n\}$

Output: DCEL \mathcal{D} für $\mathcal{A}(L)$

initialisiere \mathcal{D} für bounding box B der Knoten von $\mathcal{A}(L)$

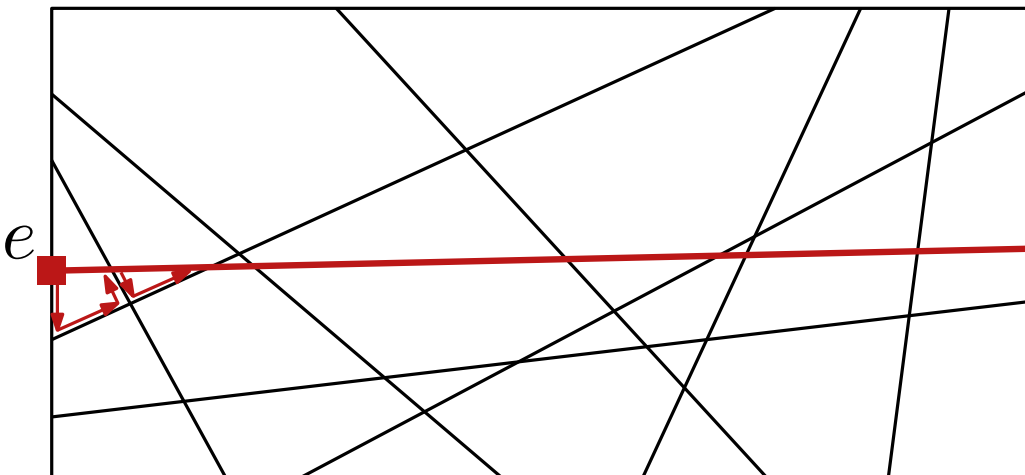
for $i \leftarrow 1$ **to** n **do**

 finde linken Schnittpunkt von ℓ_i mit Kante e von B

$f \leftarrow$ innere Zelle inzident zu e

while $f \neq$ äußere Zelle **do**

 zerteile f , update \mathcal{D} und setze f auf nächste Zelle



Konstruktion von $\mathcal{A}(L)$

Input: Geraden $L = \{\ell_1, \dots, \ell_n\}$

Output: DCEL \mathcal{D} für $\mathcal{A}(L)$

initialisiere \mathcal{D} für bounding box B der Knoten von $\mathcal{A}(L)$

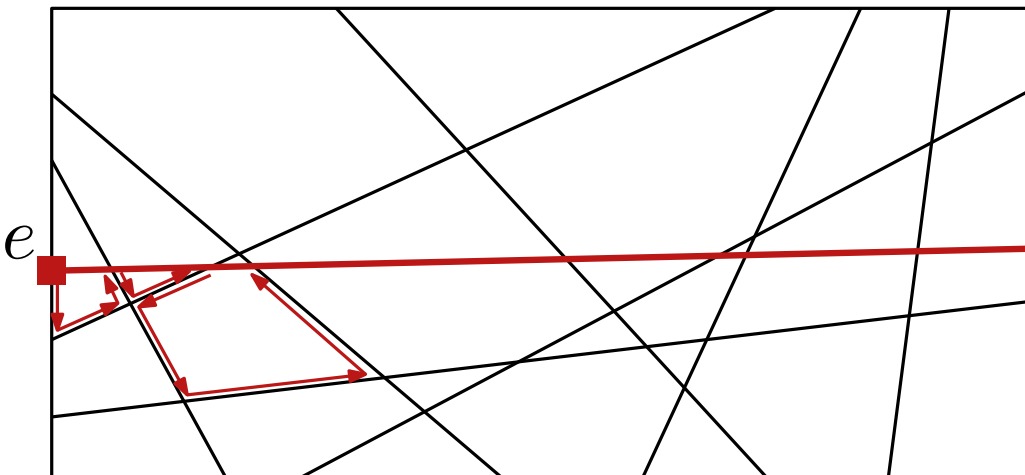
for $i \leftarrow 1$ **to** n **do**

 finde linken Schnittpunkt von ℓ_i mit Kante e von B

$f \leftarrow$ innere Zelle inzident zu e

while $f \neq$ äußere Zelle **do**

 zerteile f , update \mathcal{D} und setze f auf nächste Zelle



Konstruktion von $\mathcal{A}(L)$

Input: Geraden $L = \{\ell_1, \dots, \ell_n\}$

Output: DCEL \mathcal{D} für $\mathcal{A}(L)$

initialisiere \mathcal{D} für bounding box B der Knoten von $\mathcal{A}(L)$

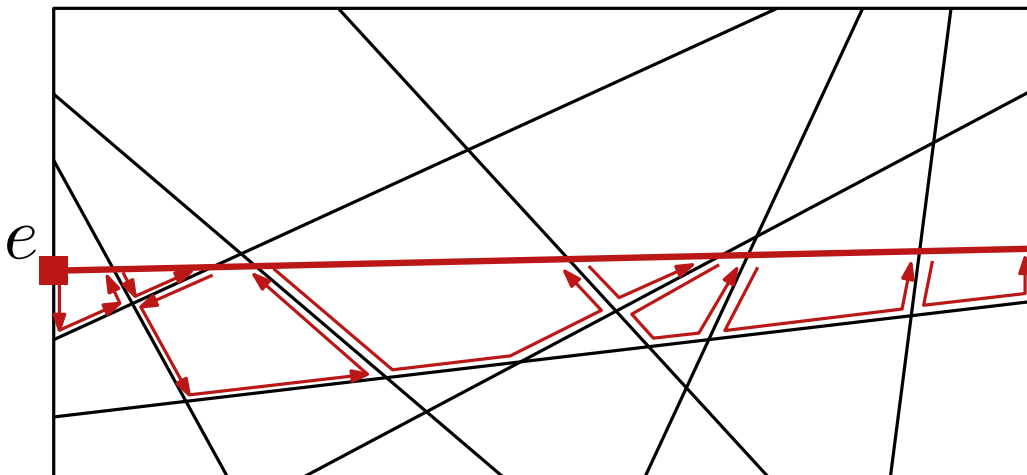
for $i \leftarrow 1$ **to** n **do**

 finde linken Schnittpunkt von ℓ_i mit Kante e von B

$f \leftarrow$ innere Zelle inzident zu e

while $f \neq$ äußere Zelle **do**

 zerteile f , update \mathcal{D} und setze f auf nächste Zelle



Konstruktion von $\mathcal{A}(L)$

Input: Geraden $L = \{\ell_1, \dots, \ell_n\}$

Output: DCEL \mathcal{D} für $\mathcal{A}(L)$

initialisiere \mathcal{D} für bounding box B der Knoten von $\mathcal{A}(L)$

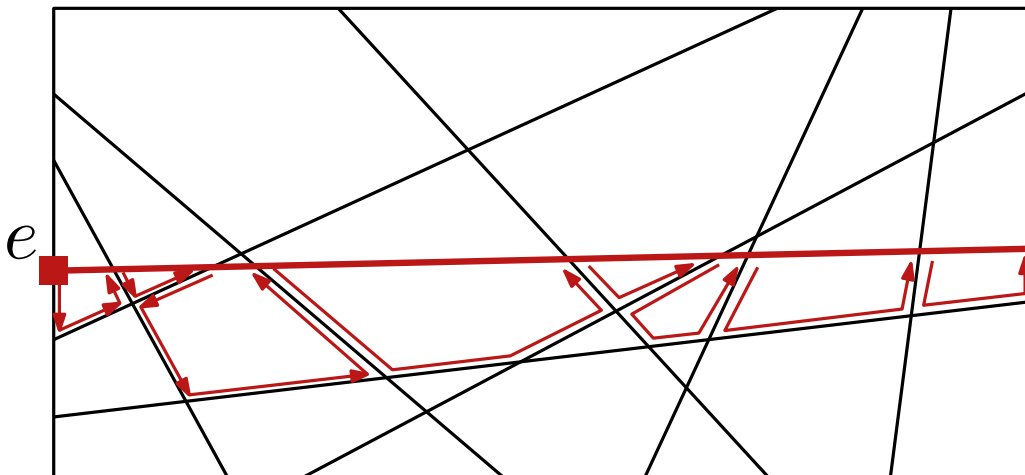
for $i \leftarrow 1$ **to** n **do**

 finde linken Schnittpunkt von ℓ_i mit Kante e von B

$f \leftarrow$ innere Zelle inzident zu e

while $f \neq$ äußere Zelle **do**

 zerteile f , update \mathcal{D} und setze f auf nächste Zelle



Laufzeit?

Konstruktion von $\mathcal{A}(L)$

Input: Geraden $L = \{\ell_1, \dots, \ell_n\}$

Output: DCEL \mathcal{D} für $\mathcal{A}(L)$

initialisiere \mathcal{D} für bounding box B der Knoten von $\mathcal{A}(L)$

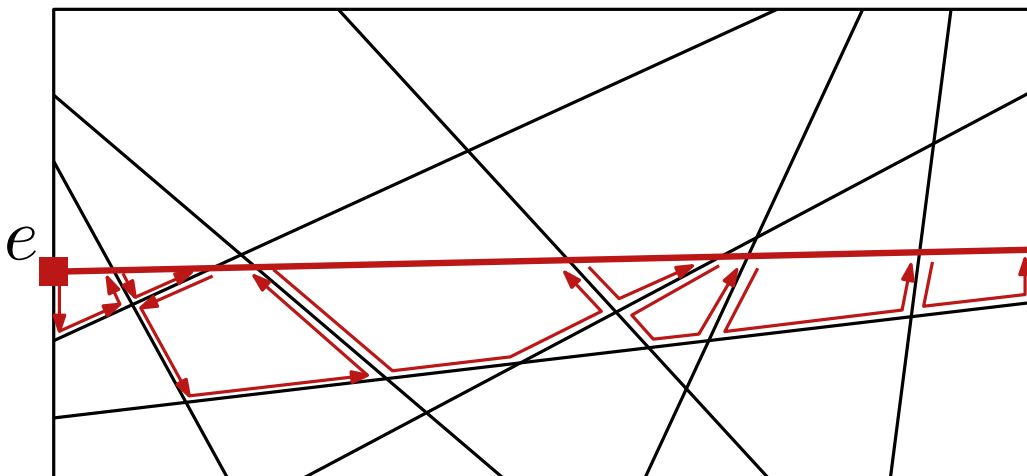
for $i \leftarrow 1$ **to** n **do**

 finde linken Schnittpunkt von ℓ_i mit Kante e von B

$f \leftarrow$ innere Zelle inzident zu e

while $f \neq$ äußere Zelle **do**

 zerteile f , update \mathcal{D} und setze f auf nächste Zelle



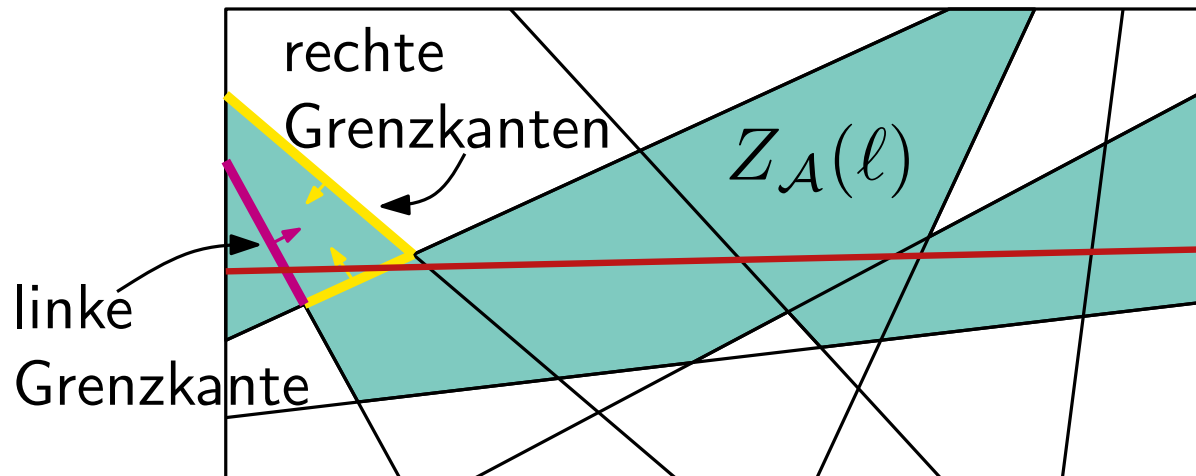
Laufzeit?

- bounding box: $O(n^2)$
- Startpunkt ℓ_i : $O(i)$
- **while**-Schleife:
 $O(|\text{roter Pfad}|)$

Def.: Für ein Arrangement $\mathcal{A}(L)$ und eine Gerade $\ell \notin L$ ist die **Zone** $Z_{\mathcal{A}}(\ell)$ die Menge aller Zellen von $\mathcal{A}(L)$, deren Abschluss ℓ schneidet.

Zonensatz

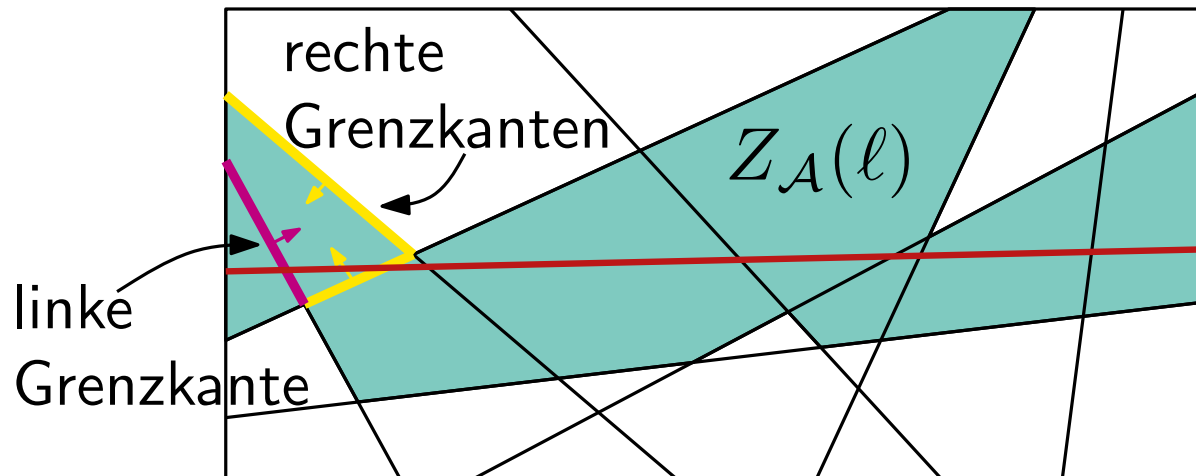
Def.: Für ein Arrangement $\mathcal{A}(L)$ und eine Gerade $\ell \notin L$ ist die **Zone** $Z_{\mathcal{A}}(\ell)$ die Menge aller Zellen von $\mathcal{A}(L)$, deren Abschluss ℓ schneidet.



Wie viele Zellen
enthält $Z_{\mathcal{A}}(\ell)$?

Wie viele Kanten
enthält $Z_{\mathcal{A}}(\ell)$?

Def.: Für ein Arrangement $\mathcal{A}(L)$ und eine Gerade $\ell \notin L$ ist die **Zone** $Z_{\mathcal{A}}(\ell)$ die Menge aller Zellen von $\mathcal{A}(L)$, deren Abschluss ℓ schneidet.

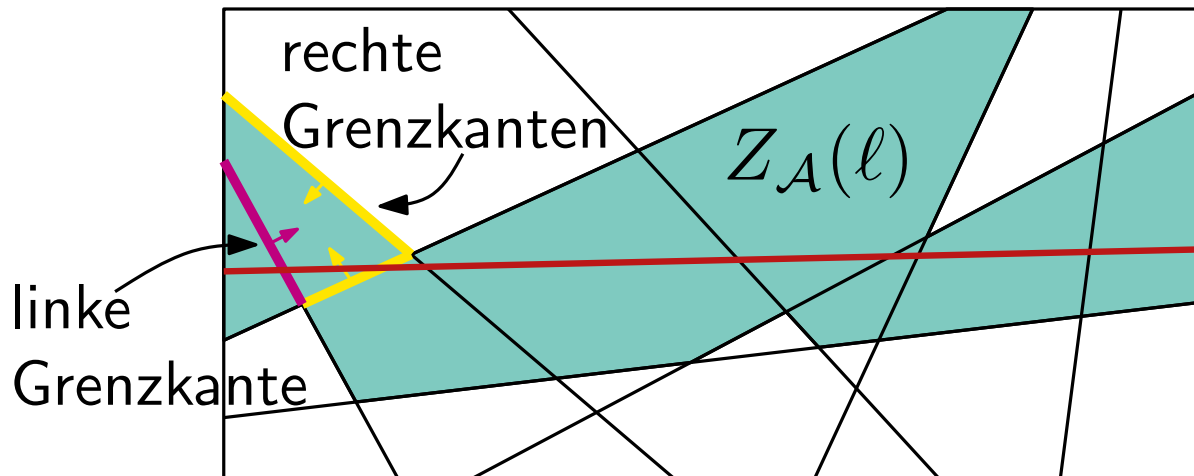


Wie viele Zellen
enthält $Z_{\mathcal{A}}(\ell)$?

Wie viele Kanten
enthält $Z_{\mathcal{A}}(\ell)$?

Satz 2: Für ein Arrangement $\mathcal{A}(L)$ von n Geraden in der Ebene und eine Gerade $\ell \notin L$ besteht $Z_{\mathcal{A}}(\ell)$ aus höchstens $6n$ Kanten.

Def.: Für ein Arrangement $\mathcal{A}(L)$ und eine Gerade $\ell \notin L$ ist die **Zone** $Z_{\mathcal{A}}(\ell)$ die Menge aller Zellen von $\mathcal{A}(L)$, deren Abschluss ℓ schneidet.



Wie viele Zellen enthält $Z_{\mathcal{A}}(\ell)$?

Wie viele Kanten enthält $Z_{\mathcal{A}}(\ell)$?

Satz 2: Für ein Arrangement $\mathcal{A}(L)$ von n Geraden in der Ebene und eine Gerade $\ell \notin L$ besteht $Z_{\mathcal{A}}(\ell)$ aus höchstens $6n$ Kanten.

Satz 3: Das Arrangement $\mathcal{A}(L)$ einer Menge von n Geraden kann in $O(n^2)$ Zeit und Platz konstruiert werden.

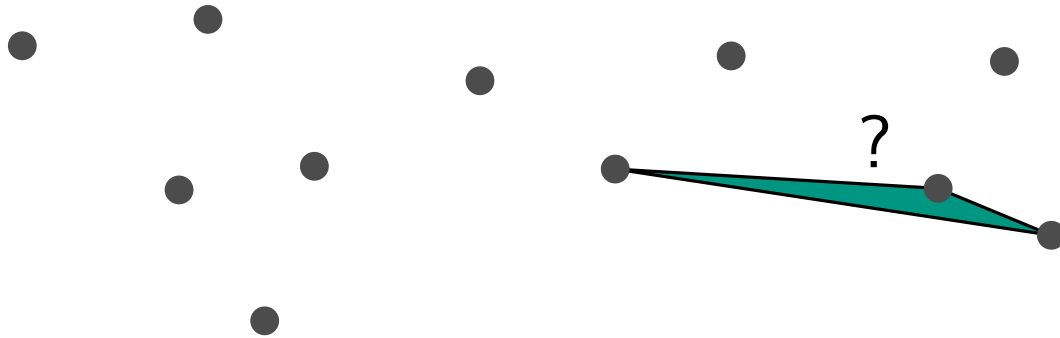
Zwischenfrage:

Wie testet man ob n Punkte in allgemeiner Lage sind?

Wie findet man eine maximal große Menge kollinearere Punkte?

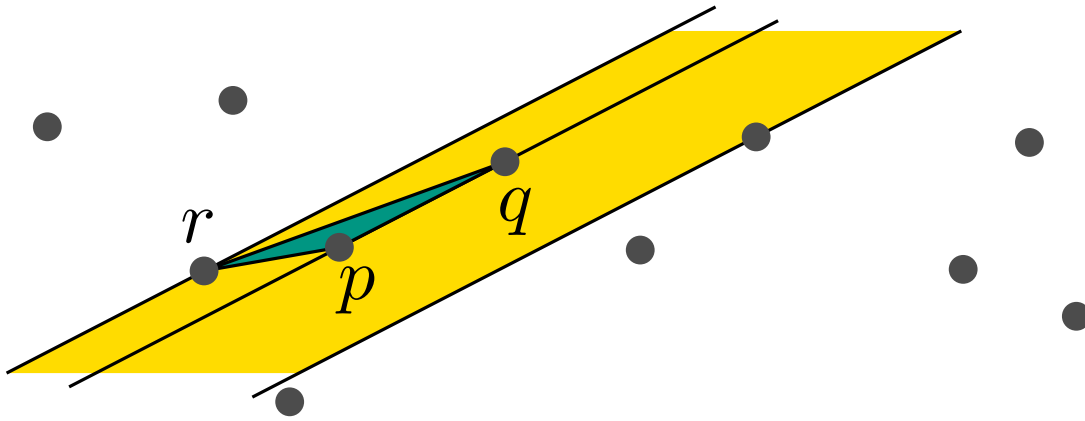
Kleinstes Dreieck

Gegeben eine Menge P von n Punkten in \mathbb{R}^2 , finde ein flächenminimales Dreieck Δpqr mit $p, q, r \in P$.



Kleinstes Dreieck

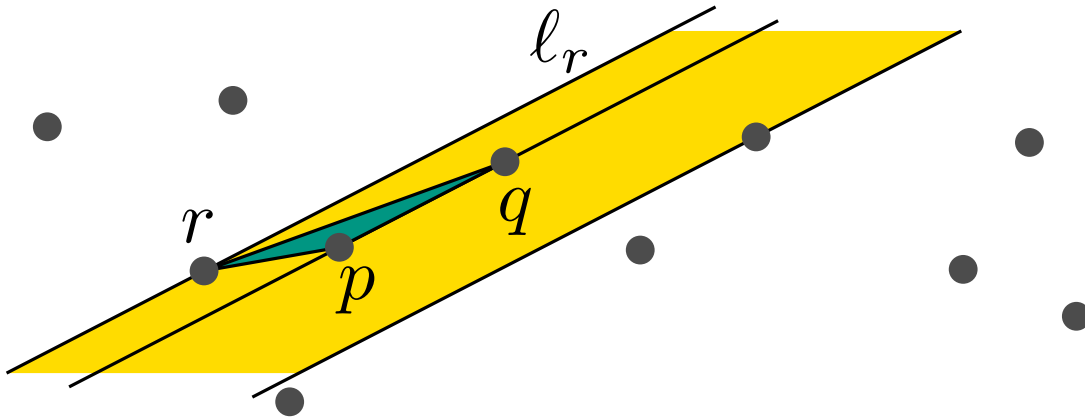
Gegeben eine Menge P von n Punkten in \mathbb{R}^2 , finde ein flächenminimales Dreieck Δpqr mit $p, q, r \in P$.



Seien $p, q \in P$. Der Punkt $r \in P \setminus \{p, q\}$, der Δpqr minimiert, liegt auf dem Rand des größten leeren Korridors entlang pq .

Kleinstes Dreieck

Gegeben eine Menge P von n Punkten in \mathbb{R}^2 , finde ein flächenminimales Dreieck Δpqr mit $p, q, r \in P$.

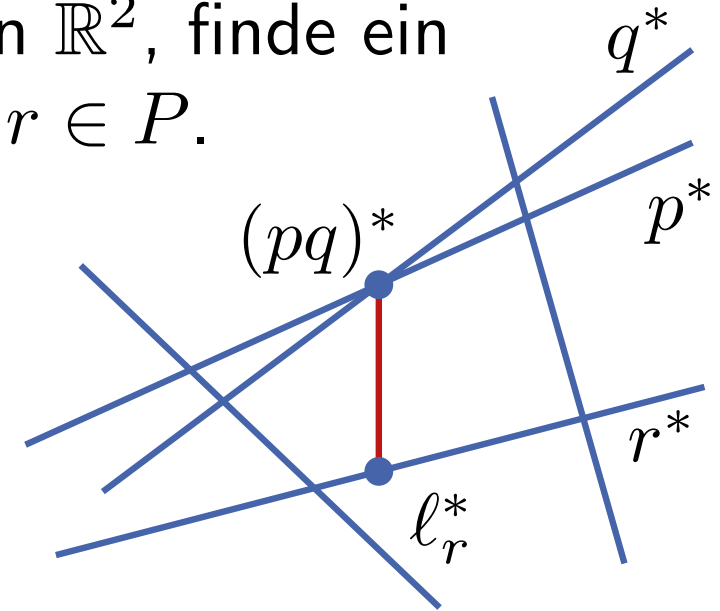
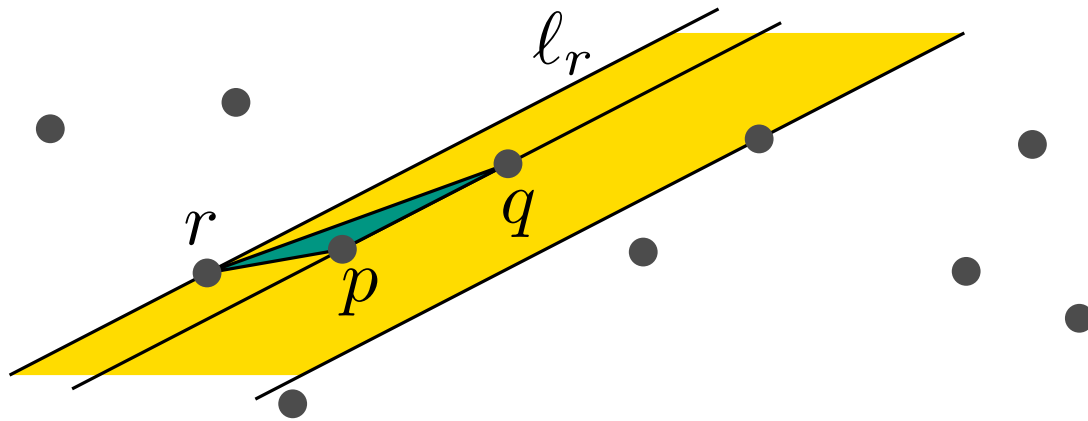


Seien $p, q \in P$. Der Punkt $r \in P \setminus \{p, q\}$, der Δpqr minimiert, liegt auf dem Rand des größten leeren Korridors entlang pq .

Zwischen pq und der Geraden l_r durch r parallel zu pq liegt kein weiterer Punkt aus P .

Kleinstes Dreieck

Gegeben eine Menge P von n Punkten in \mathbb{R}^2 , finde ein flächenminimales Dreieck Δpqr mit $p, q, r \in P$.

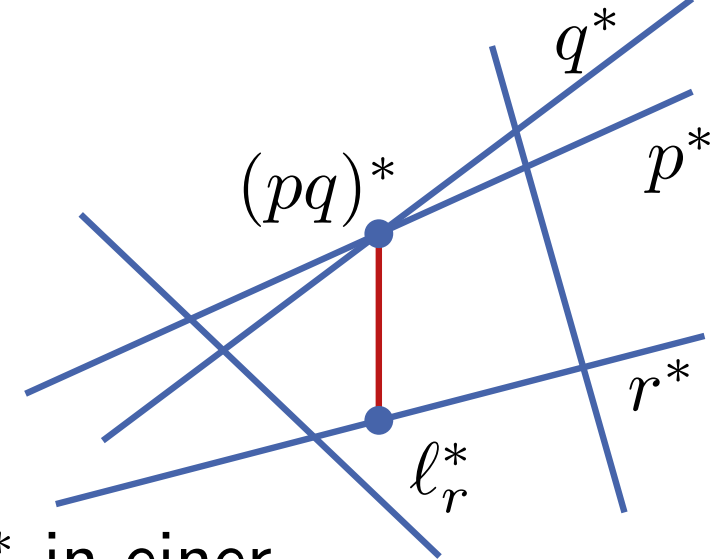
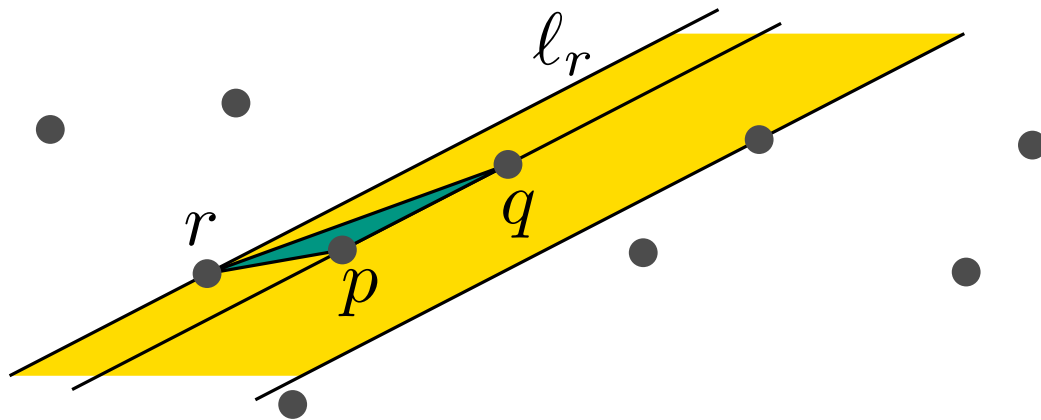


Seien $p, q \in P$. Der Punkt $r \in P \setminus \{p, q\}$, der Δpqr minimiert, liegt auf dem Rand des größten leeren Korridors entlang pq .

Zwischen pq und der Geraden l_r durch r parallel zu pq liegt kein weiterer Punkt aus P .

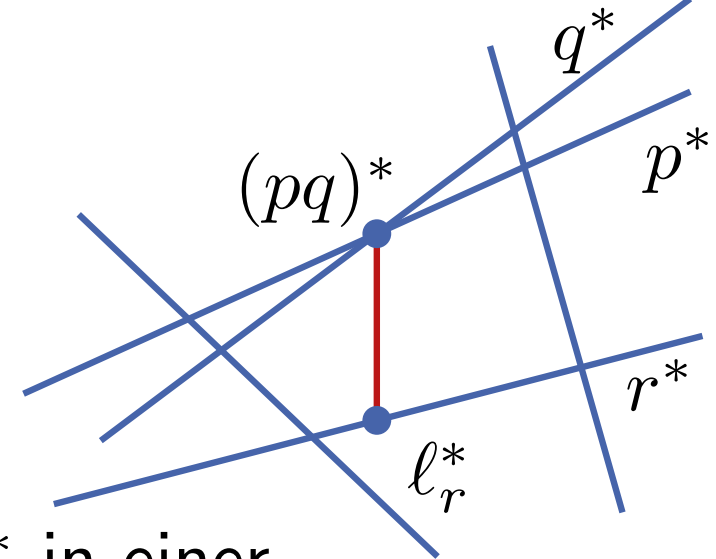
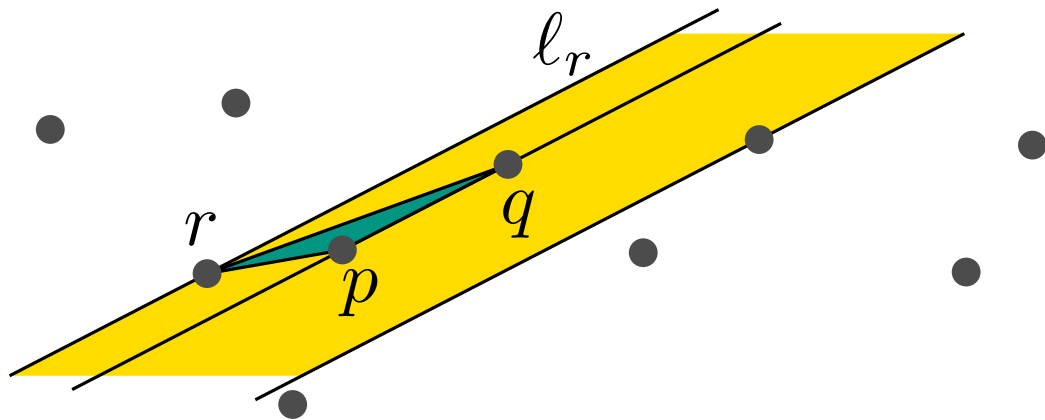
- Im Dualen:**
- l_r^* liegt auf r^*
 - l_r^* und $(pq)^*$ haben gleiche x -Koordinate
 - keine Gerade $p^* \in P^*$ schneidet $\overline{l_r^* (pq)^*}$

Berechnung im Dualen



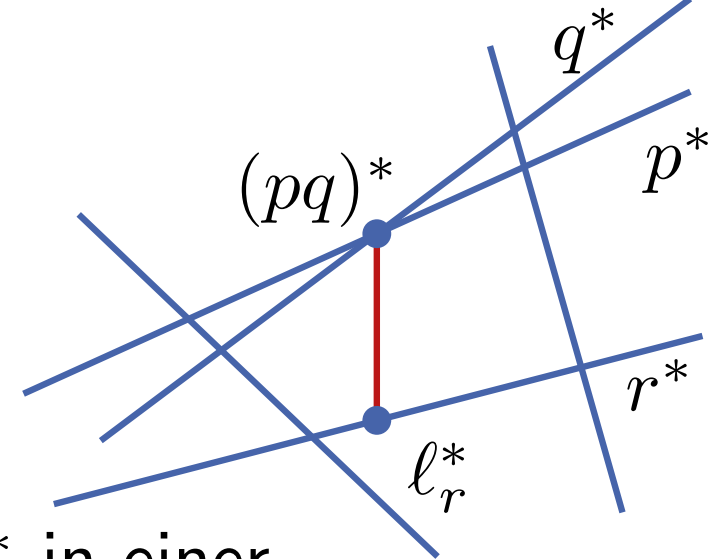
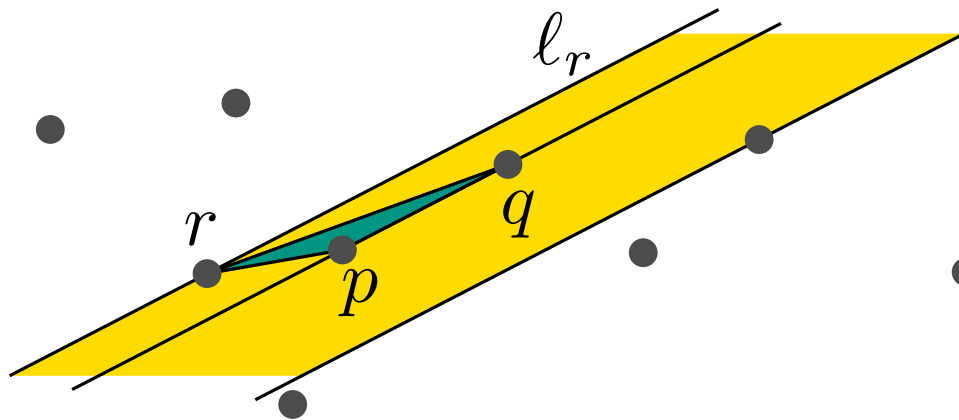
- l_r^* liegt vertikal über oder unter $(pq)^*$ in einer gemeinsamen Zelle von $\mathcal{A}(P^*) \Rightarrow$ nur zwei Kandidaten

Berechnung im Dualen



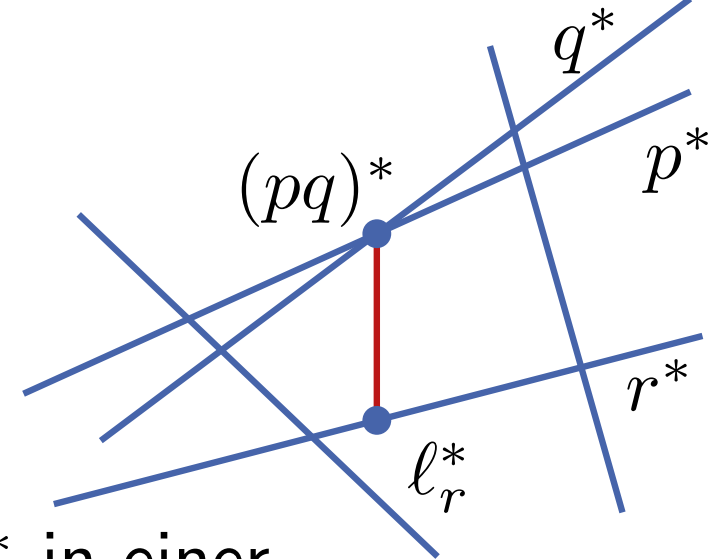
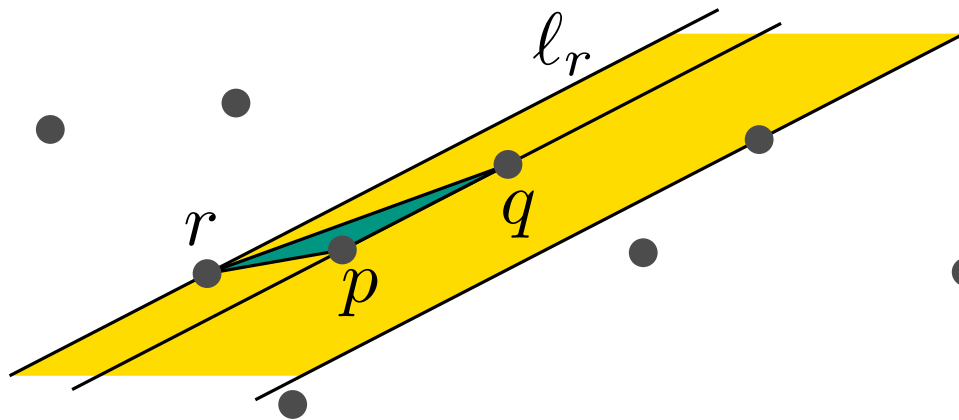
- l_r^* liegt vertikal über oder unter $(pq)^*$ in einer gemeinsamen Zelle von $\mathcal{A}(P^*) \Rightarrow$ nur zwei Kandidaten
- Berechne in $O(n^2)$ Zeit das Arrangement $\mathcal{A}(P^*)$

Berechnung im Dualen



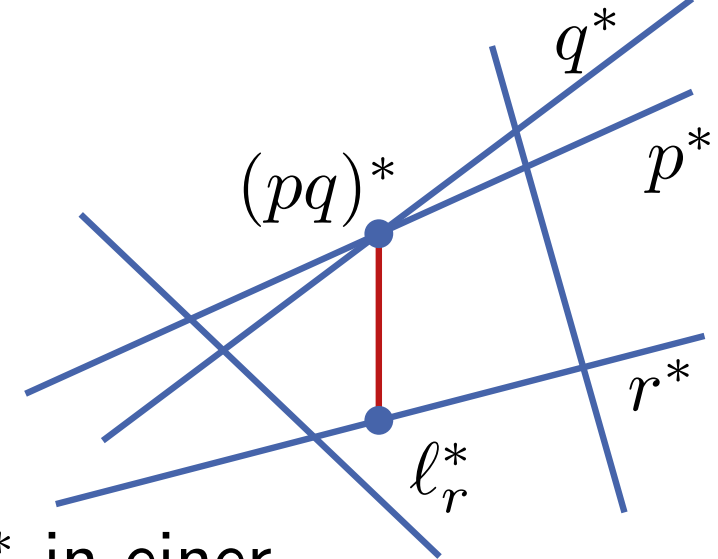
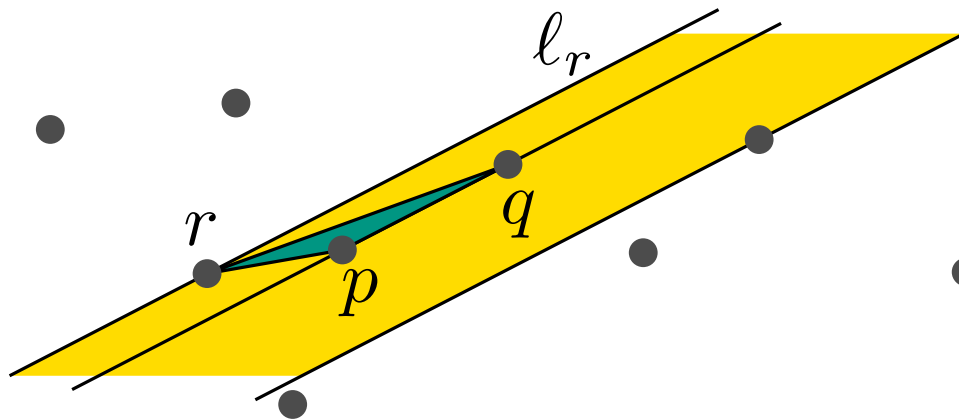
- l_r^* liegt vertikal über oder unter $(pq)^*$ in einer gemeinsamen Zelle von $\mathcal{A}(P^*) \Rightarrow$ nur zwei Kandidaten
- Berechne in $O(n^2)$ Zeit das Arrangement $\mathcal{A}(P^*)$
- Berechne in jeder Zelle die vertikalen Nachbarn der Knoten
→ Zeit linear in Zellenkomplexität **wie?**

Berechnung im Dualen



- l_r^* liegt vertikal über oder unter $(pq)^*$ in einer gemeinsamen Zelle von $\mathcal{A}(P^*) \Rightarrow$ nur zwei Kandidaten
- Berechne in $O(n^2)$ Zeit das Arrangement $\mathcal{A}(P^*)$
- Berechne in jeder Zelle die vertikalen Nachbarn der Knoten
→ Zeit linear in Zellenkomplexität **wie?**
- für alle $O(n^2)$ Kandidaten-Tripel $(pq)^*r^*$ berechne in $O(1)$ Zeit Fläche Δpqr

Berechnung im Dualen



- l_r^* liegt vertikal über oder unter $(pq)^*$ in einer gemeinsamen Zelle von $\mathcal{A}(P^*) \Rightarrow$ nur zwei Kandidaten
- Berechne in $O(n^2)$ Zeit das Arrangement $\mathcal{A}(P^*)$
- Berechne in jeder Zelle die vertikalen Nachbarn der Knoten
→ Zeit linear in Zellenkomplexität **wie?**
- für alle $O(n^2)$ Kandidaten-Tripel $(pq)^*r^*$ berechne in $O(1)$ Zeit Fläche Δpqr
- findet Minimum in insgesamt $O(n^2)$ Zeit

- Zwei Diebe haben eine Kette aus Diamanten und Smaragden gestohlen und möchten sie fair teilen, ohne sie zu sehr zu zerstören. Wieviele Schnitte sind nötig?

- Zwei Diebe haben eine Kette aus Diamanten und Smaragden gestohlen und möchten sie fair teilen, ohne sie zu sehr zu zerstören. Wieviele Schnitte sind nötig?

Satz 4: Seien S, D zwei endliche Punktmengen in der Ebene. Dann gibt es eine Gerade ℓ , die gleichzeitig S und D halbiert.

- Zwei Diebe haben eine Kette aus Diamanten und Smaragden gestohlen und möchten sie fair teilen, ohne sie zu sehr zu zerstören. Wieviele Schnitte sind nötig?

Satz 4: Seien S, D zwei endliche Punktmenge in der Ebene. Dann gibt es eine Gerade ℓ , die gleichzeitig S und D halbiert.

- Gegeben n Strecken in der Ebene, finde eine maximale stabbing-Gerade, d.h. eine Gerade, die möglichst viele Strecken schneidet.

Dualität ist ein sehr nützliches Werkzeug in der algorithmischen Geometrie!

Dualität ist ein sehr nützliches Werkzeug in der algorithmischen Geometrie!

Kann man die Dualität auch in höheren Dimensionen nutzen?

Dualität ist ein sehr nützliches Werkzeug in der algorithmischen Geometrie!

Kann man die Dualität auch in höheren Dimensionen nutzen?

Ja, auch zwischen d -dimensionalen Punkten und Hyperebenen gibt es eine entsprechende inzidenz- und ordnungserhaltende Dualitätsabbildung.

Dualität ist ein sehr nützliches Werkzeug in der algorithmischen Geometrie!

Kann man die Dualität auch in höheren Dimensionen nutzen?

Ja, auch zwischen d -dimensionalen Punkten und Hyperebenen gibt es eine entsprechende inzidenz- und ordnungserhaltende Dualitätsabbildung.

Wie steht es um höherdimensionale Arrangements?

Dualität ist ein sehr nützliches Werkzeug in der algorithmischen Geometrie!

Kann man die Dualität auch in höheren Dimensionen nutzen?

Ja, auch zwischen d -dimensionalen Punkten und Hyperebenen gibt es eine entsprechende inzidenz- und ordnungserhaltende Dualitätsabbildung.

Wie steht es um höherdimensionale Arrangements?

Das Arrangement von n Hyperebenen im \mathbb{R}^d hat Komplexität $\Theta(n^d)$. Eine Verallgemeinerung des Zonensatzes führt zu einem $O(n^d)$ -Zeit Algorithmus.