

Offline Bewegungsplanung: Reine Translation

Elmar Langetepe
University of Bonn

Idee: Divide and Conquer

Idee: Divide and Conquer

- Konvexer Roboter

Idee: Divide and Conquer

- Konvexer Roboter
- Trianguliere alle Polygone: T_1, \dots, T_l

Idee: Divide and Conquer

- Konvexer Roboter
- Trianguliere alle Polygone: T_1, \dots, T_l
- Vereinigung aller CT_i

Idee: Divide and Conquer

- Konvexer Roboter
- Trianguliere alle Polygone: T_1, \dots, T_l
- Vereinigung aller CT_i
- Divide and Conquer

Idee: Divide and Conquer

- Konvexer Roboter
- Trianguliere alle Polygone: T_1, \dots, T_l
- Vereinigung aller CT_i
- Divide and Conquer
- Mergen!!!

Idee: Divide and Conquer

- Konvexer Roboter
- Trianguliere alle Polygone: T_1, \dots, T_l
- Vereinigung aller CT_i
- Divide and Conquer
- Mergen!!!
- Komplexität des Ergebnisses

Umweg: Pseudokreise **Def. 2.10**

Umweg: Pseudokreise **Def. 2.10**

- Jordan-Kurve:

Umweg: Pseudokreise **Def. 2.10**

- Jordan-Kurve: geschlossene Kurve, teilt Ebene in zwei Gebiete

Umweg: Pseudokreise **Def. 2.10**

- Jordan-Kurve: geschlossene Kurve, teilt Ebene in zwei Gebiete
- Paar von Pseudokreisen:

Umweg: Pseudokreise **Def. 2.10**

- Jordan-Kurve: geschlossene Kurve, teilt Ebene in zwei Gebiete
- Paar von Pseudokreisen:
 - Durch Jordankurven berandete Mengen A , B

Umweg: Pseudokreise **Def. 2.10**

- Jordan-Kurve: geschlossene Kurve, teilt Ebene in zwei Gebiete
- Paar von Pseudokreisen:
 - Durch Jordankurven berandete Mengen A , B
 - Ränder haben entweder höchstens zwei Kreuzungen

Umweg: Pseudokreise **Def. 2.10**

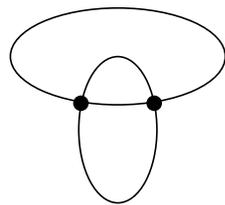
- Jordan-Kurve: geschlossene Kurve, teilt Ebene in zwei Gebiete
- Paar von Pseudokreisen:
 - Durch Jordankurven berandete Mengen A , B
 - Ränder haben entweder höchstens zwei Kreuzungen
 - Oder Ränder haben höchstens einen Berührungspunkt

Umweg: Pseudokreise **Def. 2.10**

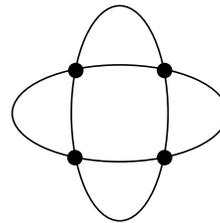
- Jordan-Kurve: geschlossene Kurve, teilt Ebene in zwei Gebiete
- Paar von Pseudokreisen:
 - Durch Jordankurven berandete Mengen A , B
 - Ränder haben entweder höchstens zwei Kreuzungen
 - Oder Ränder haben höchstens einen Berührungspunkt
- Verhalten sich wie Kreise

Umweg: Pseudokreise **Def. 2.10**

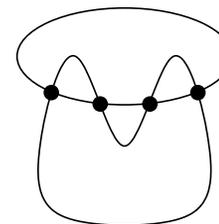
- Jordan-Kurve: geschlossene Kurve, teilt Ebene in zwei Gebiete
- Paar von Pseudokreisen:
 - Durch Jordankurven berandete Mengen A , B
 - Ränder haben entweder höchstens zwei Kreuzungen
 - Oder Ränder haben höchstens einen Berührungspunkt
- Verhalten sich wie Kreise



(i)



(ii)



(iii)

Konvexe Pseudokreise: **Lem. 2.11**

Konvexe Pseudokreise: **Lem. 2.11**

- Konvexe Mengen A, B

Konvexe Pseudokreise: **Lem. 2.11**

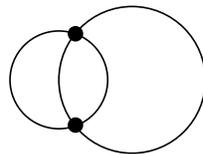
- Konvexe Mengen A, B
- A, B Paar von Pseudokreisen $\Leftrightarrow A \setminus B$ und $B \setminus A$ (weg)-zusammenhängend

Konvexe Pseudokreise: **Lem. 2.11**

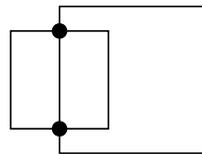
- Konvexe Mengen A, B
- A, B Paar von Pseudokreisen $\Leftrightarrow A \setminus B$ und $B \setminus A$ (weg)-zusammenhängend
- Konvexität ist wichtig (\Leftarrow)

Konvexe Pseudokreise: **Lem. 2.11**

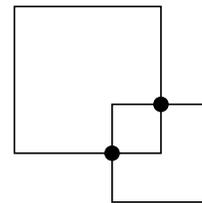
- Konvexe Mengen A, B
- A, B Paar von Pseudokreisen $\Leftrightarrow A \setminus B$ und $B \setminus A$ (weg)-zusammenhängend
- Konvexität ist wichtig (\Leftarrow)



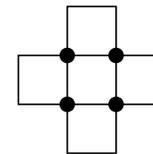
(i)



(ii)



(iii)



(iv)

Fahrplan!

Fahrplan!

- Komplexität: Vereinigung einer Menge von Pseudokreisen

Fahrplan!

- Komplexität: Vereinigung einer Menge von Pseudokreisen
- Trianguliere alle Polygone: T_1, \dots, T_l

Fahrplan!

- Komplexität: Vereinigung einer Menge von Pseudokreisen
- Trianguliere alle Polygone: T_1, \dots, T_l
- Konvexer Roboter

Fahrplan!

- Komplexität: Vereinigung einer Menge von Pseudokreisen
- Trianguliere alle Polygone: T_1, \dots, T_l
- Konvexer Roboter
- CT_i, CT_j sind Pseudokreise

Fahrplan!

- Komplexität: Vereinigung einer Menge von Pseudokreisen
- Trianguliere alle Polygone: T_1, \dots, T_l
- Konvexer Roboter
- CT_i, CT_j sind Pseudokreise
- Divide and Conquer

Fahrplan!

- Komplexität: Vereinigung einer Menge von Pseudokreisen
- Trianguliere alle Polygone: T_1, \dots, T_l
- Konvexer Roboter
- CT_i, CT_j sind Pseudokreise
- Divide and Conquer
- Mergen!!!

Fahrplan!

- Komplexität: Vereinigung einer Menge von Pseudokreisen
- Trianguliere alle Polygone: T_1, \dots, T_l
- Konvexer Roboter
- CT_i, CT_j sind Pseudokreise
- Divide and Conquer
- Mergen!!!
- Komplexität der Vereinigung

Familie polygonaler Pseudokreise **Th. 2.13**

Familie polygonaler Pseudokreise **Th. 2.13**

- A_1, A_2, \dots, A_k paarweise Paar v. Pseudokreisen

Familie polygonaler Pseudokreise **Th. 2.13**

- A_1, A_2, \dots, A_k paarweise Paar v. Pseudokreisen
- polygonal, mit insgesamt n Ecken

Familie polygonaler Pseudokreise **Th. 2.13**

- A_1, A_2, \dots, A_k paarweise Paar v. Pseudokreisen
- polygonal, mit insgesamt n Ecken
- $\partial \cup A_i$ hat Komplexität $O(n)$

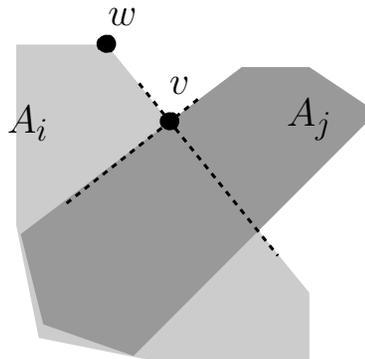
Familie polygonaler Pseudokreise **Th. 2.13**

- A_1, A_2, \dots, A_k paarweise Paar v. Pseudokreisen
- polygonal, mit insgesamt n Ecken
- $\partial \cup A_i$ hat Komplexität $O(n)$
- Zählargument, klassisch

Beweis: Th. 2.13

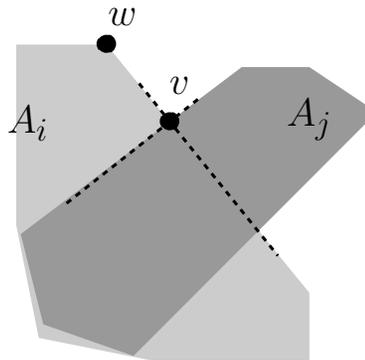
Beweis: Th. 2.13

- Knoten w von A_i auf $\partial \cup A_i$: Insgesamt: $O(n)$ viele



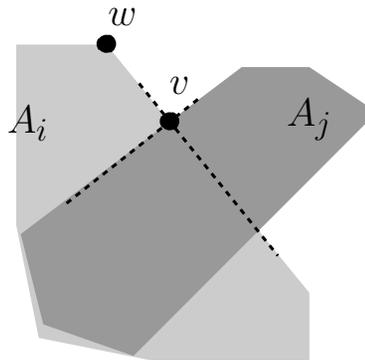
Beweis: Th. 2.13

- Knoten w von A_i auf $\partial \cup A_i$: Insgesamt: $O(n)$ viele
- Schnitt v von A_j, A_i auf $\partial \cup A_i$ zuordnen,



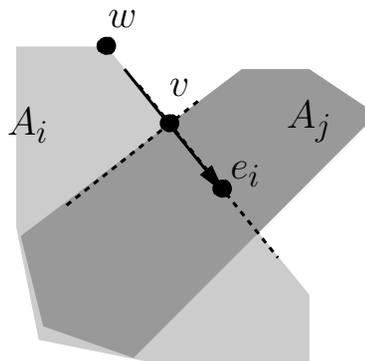
Beweis: Th. 2.13

- Knoten w von A_i auf $\partial \cup A_i$: Insgesamt: $O(n)$ viele
- Schnitt v von A_j, A_i auf $\partial \cup A_i$ zuordnen, verfolge Kante e_i :



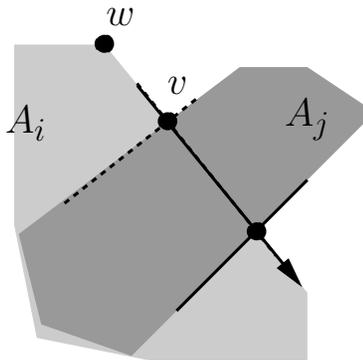
Beweis: Th. 2.13

- Knoten w von A_i auf $\partial \cup A_i$: Insgesamt: $O(n)$ viele
- Schnitt v von A_j, A_i auf $\partial \cup A_i$ zuordnen, verfolge Kante e_i :
 - Endet in A_j (innerhalb): Zähle Endpunkt von e_i



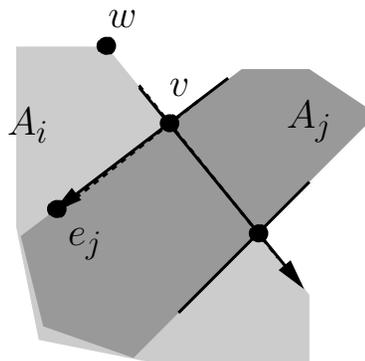
Beweis: Th. 2.13

- Knoten w von A_i auf $\partial \cup A_i$: Insgesamt: $O(n)$ viele
- Schnitt v von A_j, A_i auf $\partial \cup A_i$ zuordnen, verfolge Kante e_i :
 - Endet in A_j (innerhalb): Zähle Endpunkt von e_i
 - Geht durch A_j durch: Zähle Endpunkt von e_j (innerhalb)



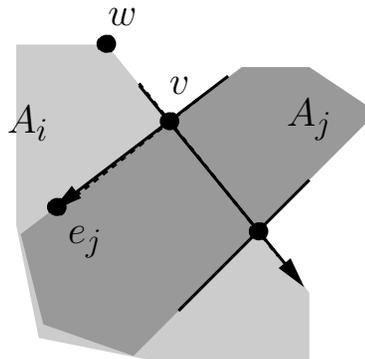
Beweis: Th. 2.13

- Knoten w von A_i auf $\partial \cup A_i$: Insgesamt: $O(n)$ viele
- Schnitt v von A_j, A_i auf $\partial \cup A_i$ zuordnen, verfolge Kante e_i :
 - Endet in A_j (innerhalb): Zähle Endpunkt von e_i
 - Geht durch A_j durch: Zähle Endpunkt von e_j (innerhalb)



Beweis: Th. 2.13

- Knoten w von A_i auf $\partial \cup A_i$: Insgesamt: $O(n)$ viele
- Schnitt v von A_j, A_i auf $\partial \cup A_i$ zuordnen, verfolge Kante e_i :
 - Endet in A_j (innerhalb): Zähle Endpunkt von e_i
 - Geht durch A_j durch: Zähle Endpunkt von e_j (innerhalb)
 - Nur zweimal belastbar, nach Aussen verfolgen



Spezielle Pseudokreise **Lem. 2.12**

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$,

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- $A = P_1 \oplus R$ und $B = P_2 \oplus R$

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- $A = P_1 \oplus R$ und $B = P_2 \oplus R$
- Paar konvexer Pseudokreise

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- $A = P_1 \oplus R$ und $B = P_2 \oplus R$
- Paar konvexer Pseudokreise

Beweis später, Benutzung:

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- $A = P_1 \oplus R$ und $B = P_2 \oplus R$
- Paar konvexer Pseudokreise

Beweis später, Benutzung:

- Triangulation der Szene: Dreiecke T_1, \dots, T_l

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- $A = P_1 \oplus R$ und $B = P_2 \oplus R$
- Paar konvexer Pseudokreise

Beweis später, Benutzung:

- Triangulation der Szene: Dreiecke T_1, \dots, T_l
- $CT_i = T_i \oplus -R$ Familie von Pseudokreisen

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- $A = P_1 \oplus R$ und $B = P_2 \oplus R$
- Paar konvexer Pseudokreise

Beweis später, Benutzung:

- Triangulation der Szene: Dreiecke T_1, \dots, T_l
- $CT_i = T_i \oplus -R$ Familie von Pseudokreisen
- Komplexität für Divide and Conquer

Komplexität Minkowski-Summe **Lem. 2.15**

- i) P_1, P_2 konvex, $P_1 \oplus P_2$ konvex, Komplexität $\Theta(m + n)$.
- ii) Nur P_2 konvex, $P_1 \oplus P_2$ Komplexität $\Theta(mn)$.
- iii) Kein P_i konvex, $P_1 \oplus P_2$ Komplexität $\Theta(m^2n^2)$.

Müssen nicht disjunkt sein!!!

Komplexität Minkowski-Summe **Lem. 2.15**

- i) P_1, P_2 konvex, $P_1 \oplus P_2$ konvex, Komplexität $\Theta(m + n)$.
- ii) Nur P_2 konvex, $P_1 \oplus P_2$ Komplexität $\Theta(mn)$.
- iii) Kein P_i konvex, $P_1 \oplus P_2$ Komplexität $\Theta(m^2n^2)$.

Müssen nicht disjunkt sein!!!

Beweis!!

Komplexität Minkowski-Summe **Lem. 2.15**

- i) P_1, P_2 konvex, $P_1 \oplus P_2$ konvex, Komplexität $\Theta(m + n)$.
- ii) Nur P_2 konvex, $P_1 \oplus P_2$ Komplexität $\Theta(mn)$.
- iii) Kein P_i konvex, $P_1 \oplus P_2$ Komplexität $\Theta(m^2n^2)$.

Müssen nicht disjunkt sein!!!

Beweis!! i) bereits gezeigt ($\Omega(n + m)$)

ii) Nur P_2 konvex: $O(mn)$

- Trianguliere P_1 ($|P_1| = n$): T_1, \dots, T_{n-2}

ii) Nur P_2 konvex: $O(mn)$

- Trianguliere P_1 ($|P_1| = n$): T_1, \dots, T_{n-2}
- $P_1 = \bigcup_{i=1}^{n-2} T_i$, Inneres disjunkt

ii) Nur P_2 konvex: $O(mn)$

- Trianguliere P_1 ($|P_1| = n$): T_1, \dots, T_{n-2}
- $P_1 = \bigcup_{i=1}^{n-2} T_i$, Inneres disjunkt
- Distributivität ausnutzen:

ii) Nur P_2 konvex: $O(mn)$

- Trianguliere P_1 ($|P_1| = n$): T_1, \dots, T_{n-2}
- $P_1 = \bigcup_{i=1}^{n-2} T_i$, Inneres disjunkt
- Distributivität ausnutzen:

$$P_1 \oplus P_2 = \left(\bigcup_{i=1}^{n-2} T_i \right) \oplus P_2 = \bigcup_{i=1}^{n-2} \underbrace{(T_i \oplus P_2)}_{O(m) \text{ Kanten}} .$$

ii) Nur P_2 konvex: $O(mn)$

- Trianguliere P_1 ($|P_1| = n$): T_1, \dots, T_{n-2}
- $P_1 = \bigcup_{i=1}^{n-2} T_i$, Inneres disjunkt
- Distributivität ausnutzen:

$$P_1 \oplus P_2 = \left(\bigcup_{i=1}^{n-2} T_i \right) \oplus P_2 = \bigcup_{i=1}^{n-2} \underbrace{(T_i \oplus P_2)}_{O(m) \text{ Kanten}} .$$

- Lemma 2.12: Familie v. Pseudokreisen, insg. $O(mn)$ Kanten

ii) Nur P_2 konvex: $O(mn)$

- Trianguliere P_1 ($|P_1| = n$): T_1, \dots, T_{n-2}
- $P_1 = \bigcup_{i=1}^{n-2} T_i$, Inneres disjunkt
- Distributivität ausnutzen:

$$P_1 \oplus P_2 = \left(\bigcup_{i=1}^{n-2} T_i \right) \oplus P_2 = \bigcup_{i=1}^{n-2} \underbrace{(T_i \oplus P_2)}_{O(m) \text{ Kanten}} .$$

- Lemma 2.12: Familie v. Pseudokreisen, insg. $O(mn)$ Kanten
- Theorem 2.13: Vereinigung hat Komplexität $O(mn)$

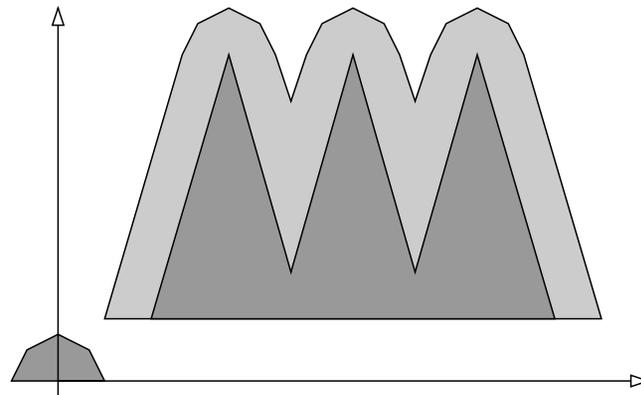
ii) Nur P_2 konvex: $\Omega(mn)$

ii) Nur P_2 konvex: $\Omega(mn)$

Untere Schranken, konstruktiv

ii) Nur P_2 konvex: $\Omega(mn)$

Untere Schranken, konstruktiv



iii) Kein P_i konvex: $O((mn)^2)$

iii) Kein P_i konvex: $O((mn)^2)$

- Trianguliere P_1 ($|P_1| = n$) und P_2 ($|P_2| = m$):
 T_1, \dots, T_{n-2} und T'_1, \dots, T'_{m-2}

iii) Kein P_i konvex: $O((mn)^2)$

- Trianguliere P_1 ($|P_1| = n$) und P_2 ($|P_2| = m$):

T_1, \dots, T_{n-2} und T'_1, \dots, T'_{m-2}

- $P_1 = \bigcup_{i=1}^{n-2} T_i, P_2 = \bigcup_{i=1}^{m-2} T'_i$

iii) Kein P_i konvex: $O((mn)^2)$

- Trianguliere P_1 ($|P_1| = n$) und P_2 ($|P_2| = m$):

T_1, \dots, T_{n-2} und T'_1, \dots, T'_{m-2}

- $P_1 = \bigcup_{i=1}^{n-2} T_i, P_2 = \bigcup_{i=1}^{m-2} T'_i$

- Distributivität ausnutzen:

$$P_1 \oplus P_2 = \bigcup_{i=1}^{n-2} \bigcup_{j=1}^{m-2} \underbrace{T_i \oplus T'_j}_{O(1) \text{ Kanten}} .$$

iii) Kein P_i konvex: $O((mn)^2)$

- Trianguliere P_1 ($|P_1| = n$) und P_2 ($|P_2| = m$):

T_1, \dots, T_{n-2} und T'_1, \dots, T'_{m-2}

- $P_1 = \bigcup_{i=1}^{n-2} T_i, P_2 = \bigcup_{i=1}^{m-2} T'_i$

- Distributivität ausnutzen:

$$P_1 \oplus P_2 = \bigcup_{i=1}^{n-2} \bigcup_{j=1}^{m-2} \underbrace{T_i \oplus T'_j}_{O(1) \text{ Kanten}} .$$

- Gesamtzahl der Kanten: $O(nm)$

iii) Kein P_i konvex: $O((mn)^2)$

- Trianguliere P_1 ($|P_1| = n$) und P_2 ($|P_2| = m$):

T_1, \dots, T_{n-2} und T'_1, \dots, T'_{m-2}

- $P_1 = \bigcup_{i=1}^{n-2} T_i, P_2 = \bigcup_{i=1}^{m-2} T'_i$

- Distributivität ausnutzen:

$$P_1 \oplus P_2 = \bigcup_{i=1}^{n-2} \bigcup_{j=1}^{m-2} \underbrace{T_i \oplus T'_j}_{O(1) \text{ Kanten}} .$$

- Gesamtzahl der Kanten: $O(nm)$, jede mit jeder: Paare $O((mn)^2)$

iii) Kein P_i konvex: $O((mn)^2)$

- Trianguliere P_1 ($|P_1| = n$) und P_2 ($|P_2| = m$):

T_1, \dots, T_{n-2} und T'_1, \dots, T'_{m-2}

- $P_1 = \bigcup_{i=1}^{n-2} T_i, P_2 = \bigcup_{i=1}^{m-2} T'_i$

- Distributivität ausnutzen:

$$P_1 \oplus P_2 = \bigcup_{i=1}^{n-2} \bigcup_{j=1}^{m-2} \underbrace{T_i \oplus T'_j}_{O(1) \text{ Kanten}} .$$

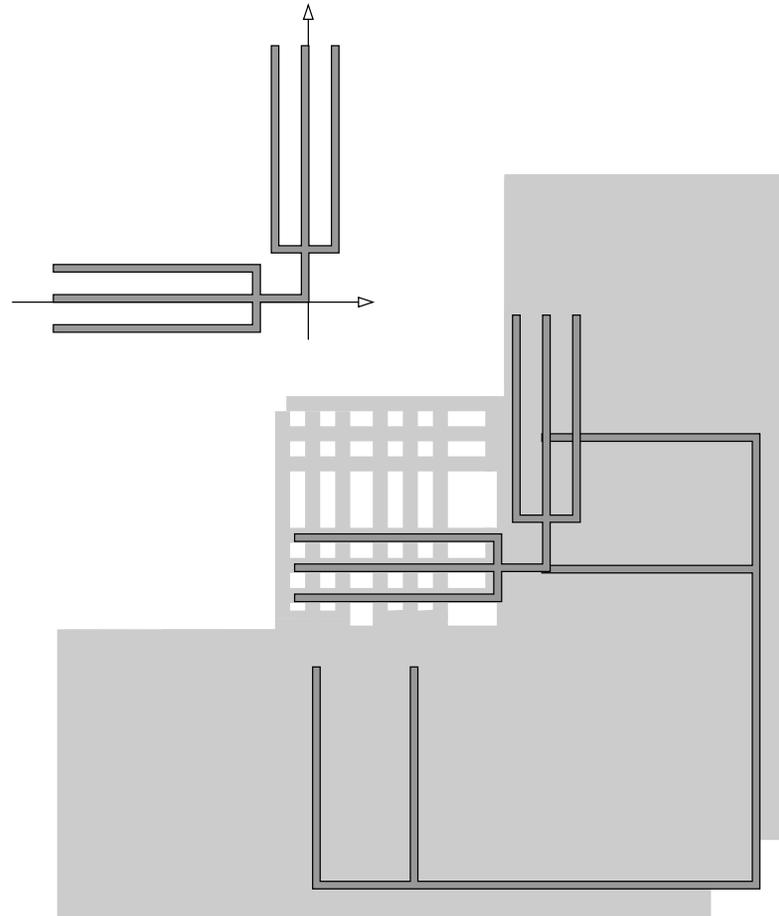
- Gesamtzahl der Kanten: $O(nm)$, jede mit jeder: Paare $O((mn)^2)$

- Pseudokreise?

iii) Kein P_i konvex: $\Omega((mn)^2)$

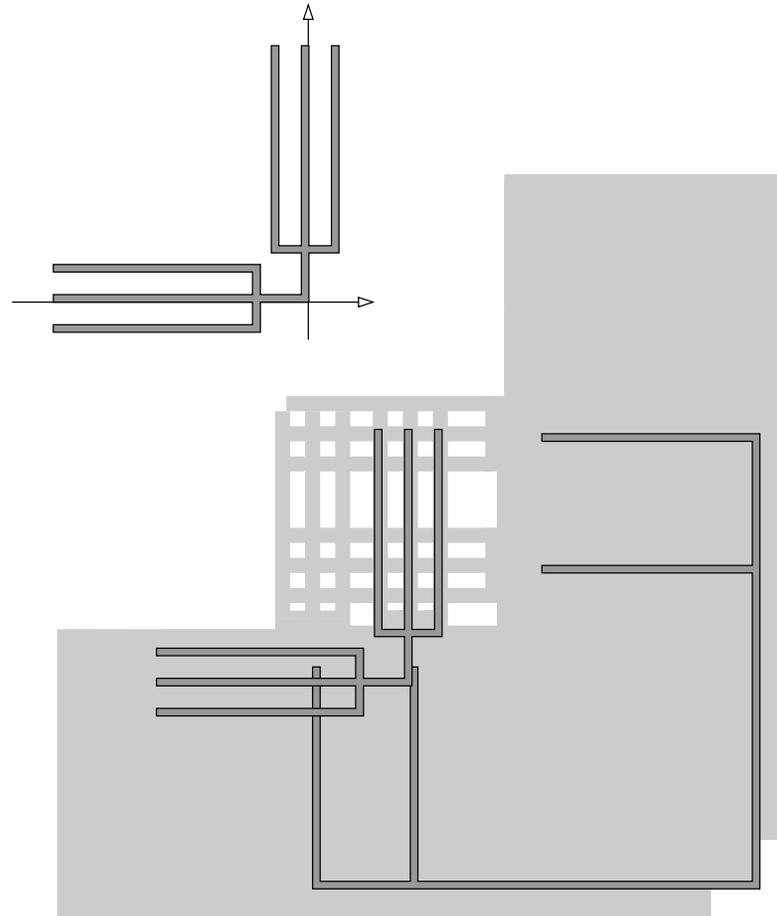
iii) Kein P_i konvex: $\Omega((mn)^2)$

Untere Schranken, konstruktiv



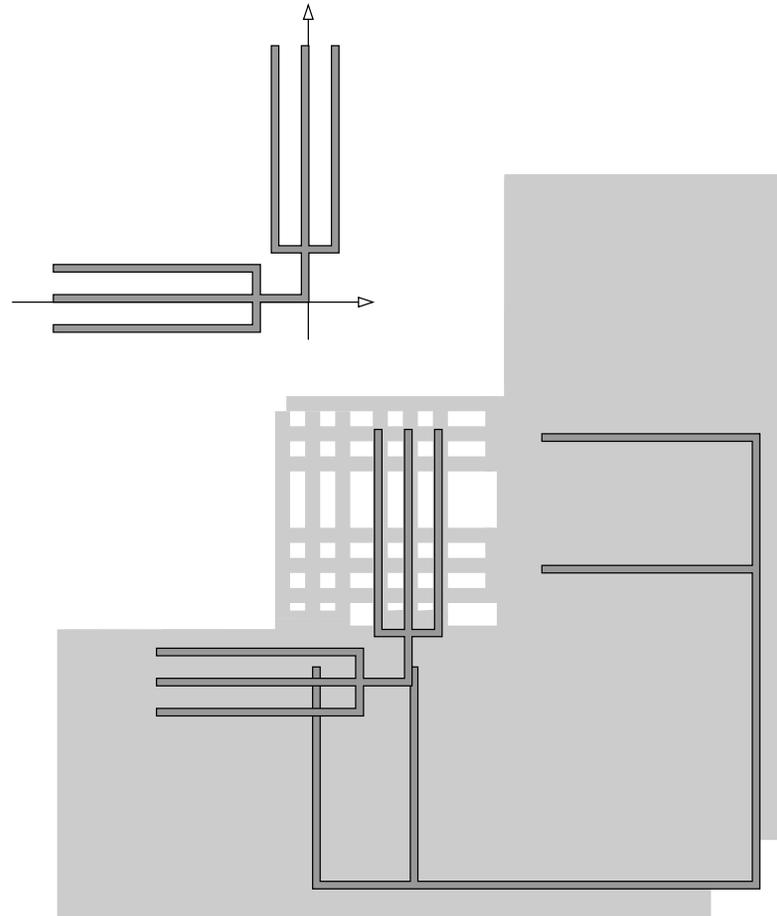
iii) Kein P_i konvex: $\Omega((mn)^2)$

Untere Schranken, konstruktiv



iii) Kein P_i konvex: $\Omega((mn)^2)$

Untere Schranken, konstruktiv



Spezielle Pseudokreise **Lem. 2.12**

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$,

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex

Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- $A = P_1 \oplus R$ und $B = P_2 \oplus R$

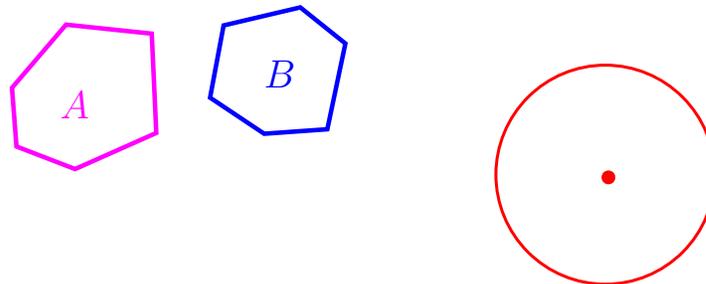
Spezielle Pseudokreise **Lem. 2.12**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- $A = P_1 \oplus R$ und $B = P_2 \oplus R$
- Paar konvexer Pseudokreise

Extremalpunkte $\dot{P}_1 \cap \dot{P}_2 = \emptyset$

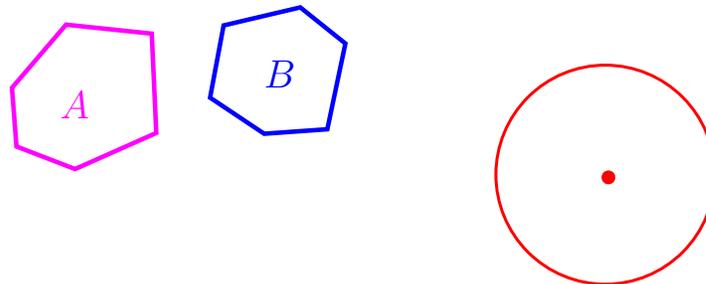
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen



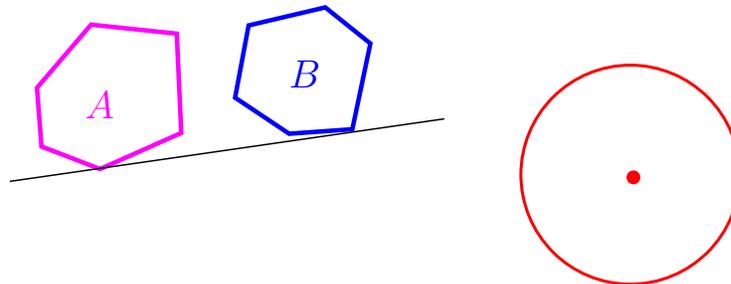
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



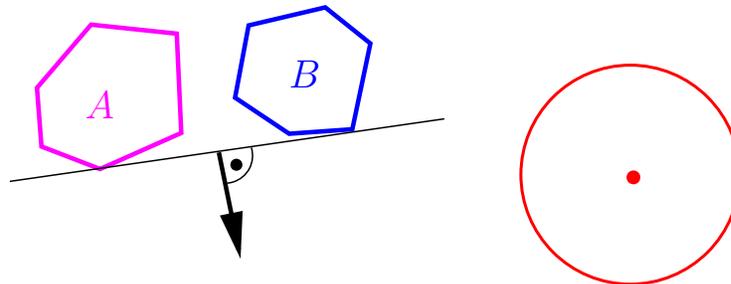
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



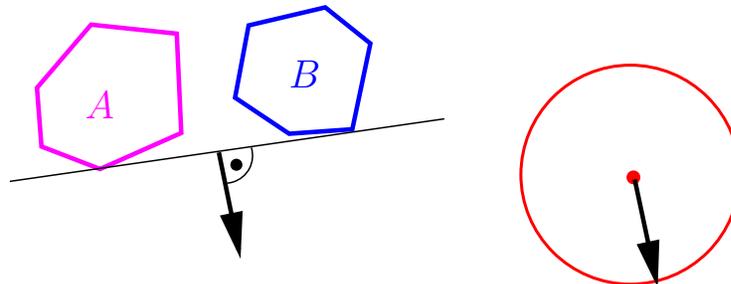
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



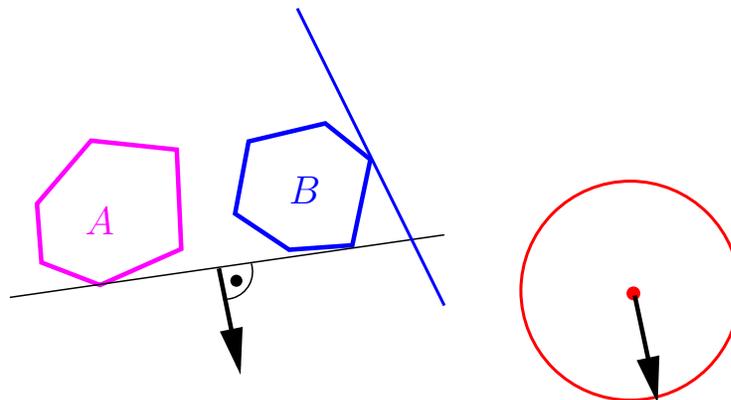
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



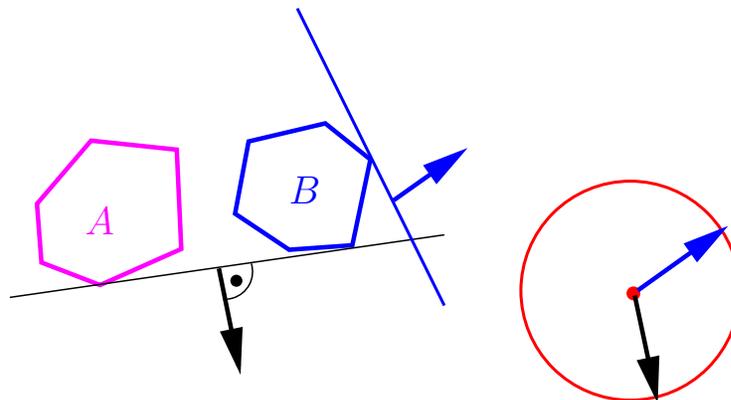
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



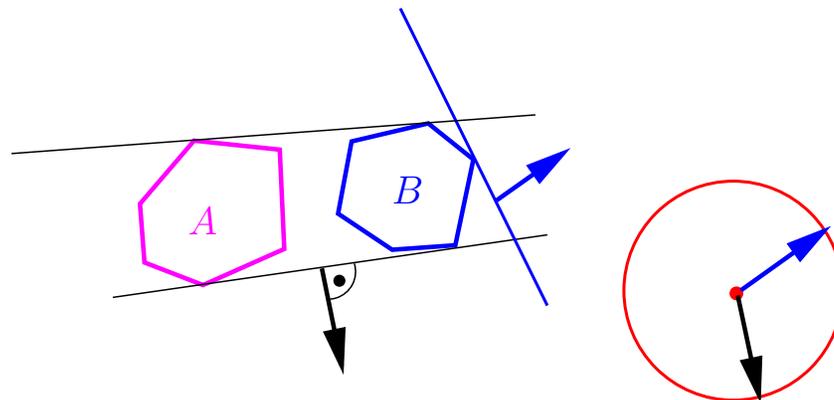
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



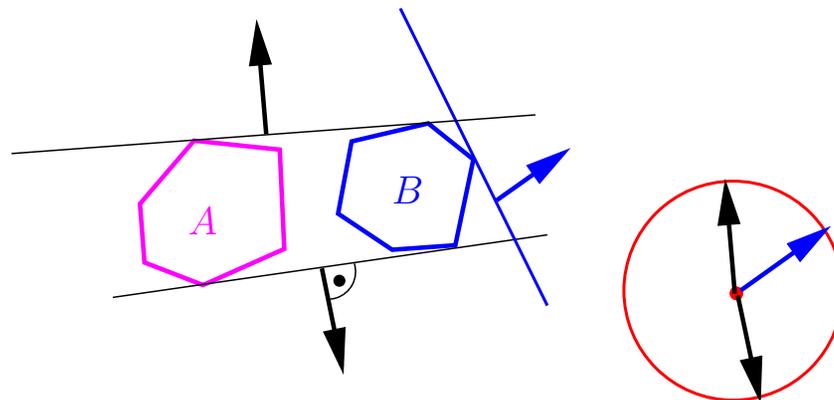
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



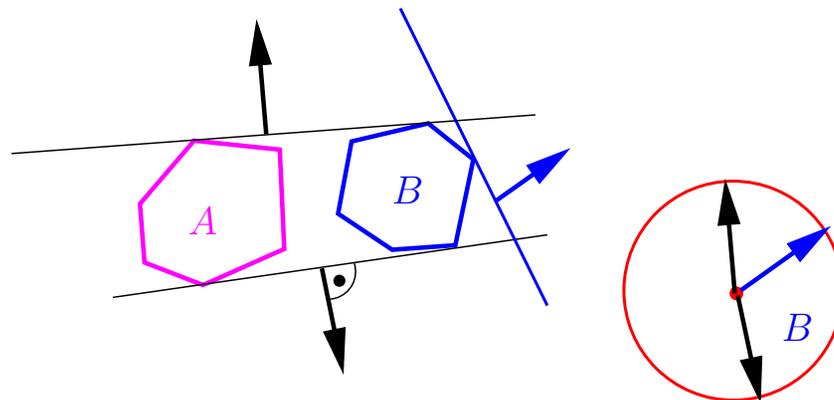
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



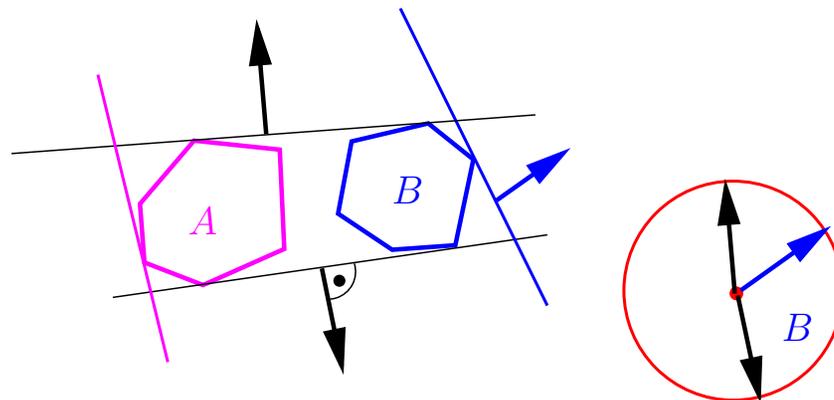
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



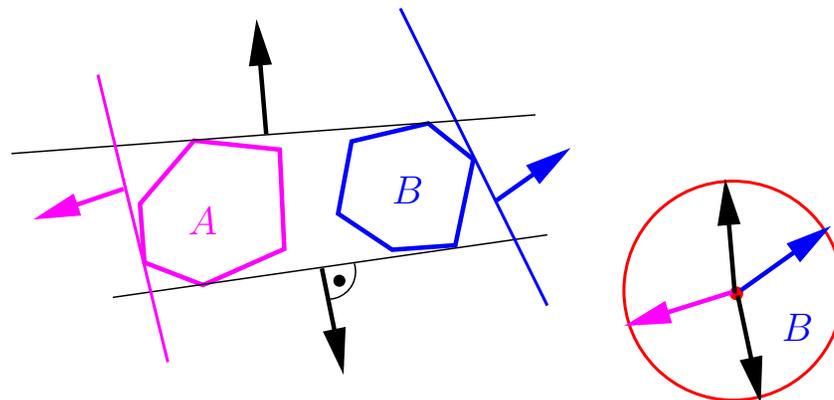
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



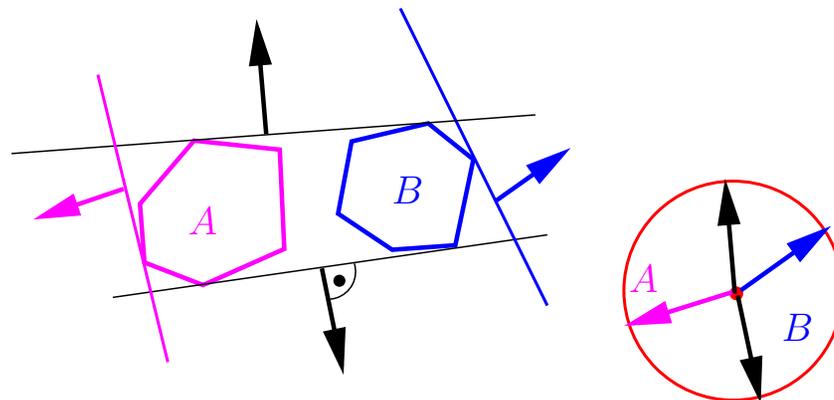
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab,



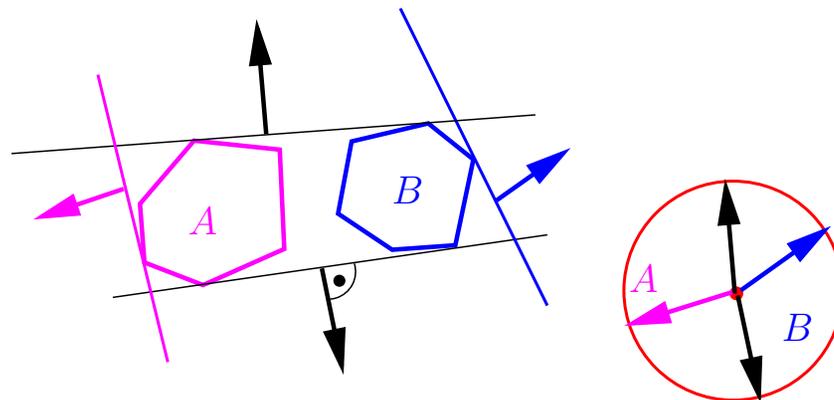
Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab, disjunkt!!



Extremalpunkte $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$

- Bem. 2.8 b)/Skript: Abb. 2.9
- Extremalpunkte $P_1 \cup P_2$ bezüglich aller Richtungen
- Wechseln sich zweimal ab, disjunkt!!



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$,

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$
- Zwei Fälle:

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$
- Zwei Fälle:
 - i) $Z_i = A$

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$
- Zwei Fälle:
 - i) $Z_i = A \Rightarrow A \cap B = \emptyset$,

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$
- Zwei Fälle:
 - i) $Z_i = A \Rightarrow A \cap B = \emptyset$, es ex. $a \in A$ mit $a \in ch(A \cup B)$,

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$
- Zwei Fälle:
 - i) $Z_i = A \Rightarrow A \cap B = \emptyset$, es ex. $a \in A$ mit $a \in ch(A \cup B)$,
denn A, B sind konvex

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$
- Zwei Fälle:
 - i) $Z_i = A \Rightarrow A \cap B = \emptyset$, es ex. $a \in A$ mit $a \in ch(A \cup B)$,
denn A, B sind konvex
 - ii) $Z_i \subset A$

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$
- Zwei Fälle:
 - i) $Z_i = A \Rightarrow A \cap B = \emptyset$, es ex. $a \in A$ mit $a \in ch(A \cup B)$,
denn A, B sind konvex
 - ii) $Z_i \subset A \Rightarrow$

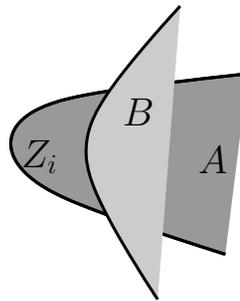
$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- P_1, P_2 konvex, $\overset{\circ}{P}_1 \cap \overset{\circ}{P}_2 = \emptyset$, R konvex
- Konvexität: klar!
- Symmetrie: Zeige $A \setminus B$ ist zusammenhängend
- Zusammenhangskomponenten Z_i von $A \setminus B$: Es gibt nur eine!
- 1) Jedes Z_i hat Punkte auf Rand von $ch(A \cup B)$
- Zwei Fälle:
 - i) $Z_i = A \Rightarrow A \cap B = \emptyset$, es ex. $a \in A$ mit $a \in ch(A \cup B)$,
denn A, B sind konvex
 - ii) $Z_i \subset A \Rightarrow \exists b \in B$ mit $b \in \partial Z_i$

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

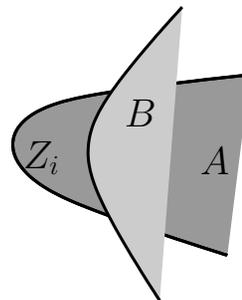
$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

Fall ii) $Z_i \subset A$



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

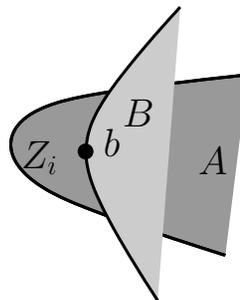
Fall ii) $Z_i \subset A$ z.z.: $\exists a \in Z_i$ und $a \in ch(A \cup B)$



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

Fall ii) $Z_i \subset A$ z.z.: $\exists a \in Z_i$ und $a \in ch(A \cup B)$

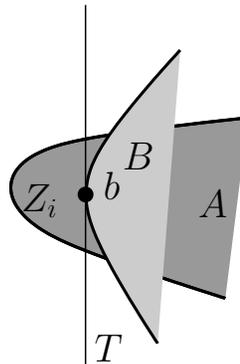
- $\exists b \in B$ mit $b \in \partial Z_i$



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

Fall ii) $Z_i \subset A$ z.z.: $\exists a \in Z_i$ und $a \in ch(A \cup B)$

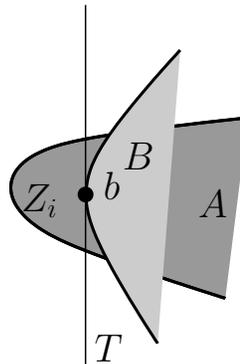
- $\exists b \in B$ mit $b \in \partial Z_i$
- Tangente T durch b an B ,



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

Fall ii) $Z_i \subset A$ z.z.: $\exists a \in Z_i$ und $a \in ch(A \cup B)$

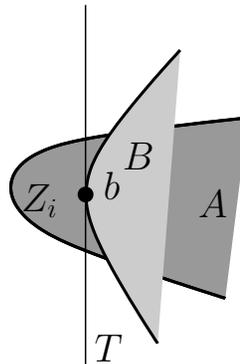
- $\exists b \in B$ mit $b \in \partial Z_i$
- Tangente T durch b an B , konvex



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

Fall ii) $Z_i \subset A$ z.z.: $\exists a \in Z_i$ und $a \in ch(A \cup B)$

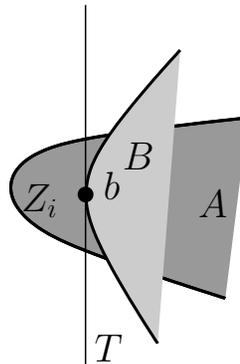
- $\exists b \in B$ mit $b \in \partial Z_i$
- Tangente T durch b an B , konvex
- B vollst. auf einer Seite,



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

Fall ii) $Z_i \subset A$ z.z.: $\exists a \in Z_i$ und $a \in ch(A \cup B)$

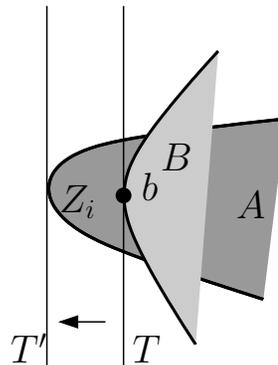
- $\exists b \in B$ mit $b \in \partial Z_i$
- Tangente T durch b an B , konvex
- B vollst. auf einer Seite, Punkte von A auf der anderen!!



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

Fall ii) $Z_i \subset A$ z.z.: $\exists a \in Z_i$ und $a \in ch(A \cup B)$

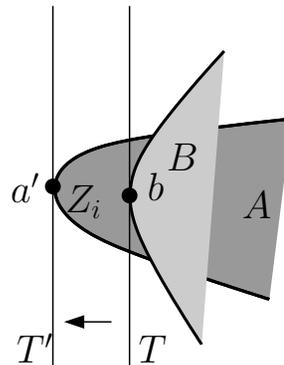
- $\exists b \in B$ mit $b \in \partial Z_i$
- Tangente T durch b an B , konvex
- B vollst. auf einer Seite, Punkte von A auf der anderen!!
- Verschiebe T parallel, bis Rand von A erreicht



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

Fall ii) $Z_i \subset A$ z.z.: $\exists a \in Z_i$ und $a \in ch(A \cup B)$

- $\exists b \in B$ mit $b \in \partial Z_i$
- Tangente T durch b an B , konvex
- B vollst. auf einer Seite, Punkte von A auf der anderen!!
- Verschiebe T parallel, bis Rand von A erreicht
- $a' \in Z_i$ und $a' \in ch(A \cup B)$



$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- 2) Zeige $A \setminus B$ ist zusammenhängend
- Gerade gezeigt:

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- 2) Zeige $A \setminus B$ ist zusammenhängend
- Gerade gezeigt: 1) Jedes Z_i von $A \setminus B$ hat Rand von $ch(A \cup B)$

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- 2) Zeige $A \setminus B$ ist zusammenhängend
- Gerade gezeigt: 1) Jedes Z_i von $A \setminus B$ hat Rand von $ch(A \cup B)$
- Annahme: Unabhängige Z_1 und Z_2 existieren

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

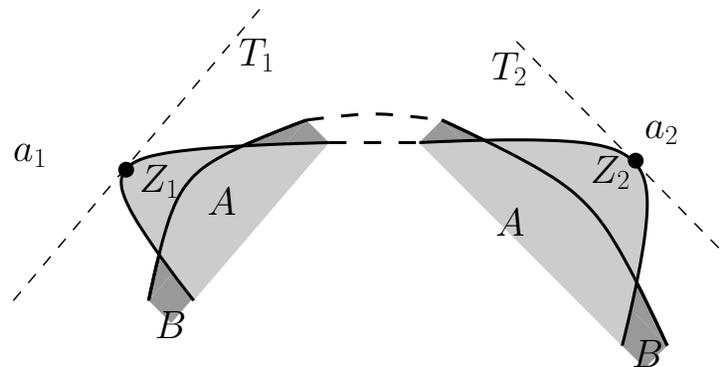
- 2) Zeige $A \setminus B$ ist zusammenhängend
- Gerade gezeigt: 1) Jedes Z_i von $A \setminus B$ hat Rand von $ch(A \cup B)$
- Annahme: Unabhängige Z_1 und Z_2 existieren
- $a_1 \in Z_1$ und $a_2 \in Z_2$, $a_1, a_2 \in ch(A \cup B)$

$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- 2) Zeige $A \setminus B$ ist zusammenhängend
- Gerade gezeigt: 1) Jedes Z_i von $A \setminus B$ hat Rand von $ch(A \cup B)$
- Annahme: Unabhängige Z_1 und Z_2 existieren
- $a_1 \in Z_1$ und $a_2 \in Z_2$, $a_1, a_2 \in ch(A \cup B)$
- Situation!!

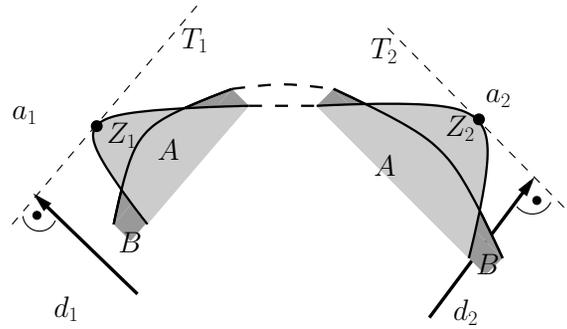
$A = P_1 \oplus R$ und $B = P_2 \oplus R$ **konvexe Pseudokreise**

- 2) Zeige $A \setminus B$ ist zusammenhängend
- Gerade gezeigt: 1) Jedes Z_i von $A \setminus B$ hat Rand von $ch(A \cup B)$
- Annahme: Unabhängige Z_1 und Z_2 existieren
- $a_1 \in Z_1$ und $a_2 \in Z_2$, $a_1, a_2 \in ch(A \cup B)$
- Situation!!



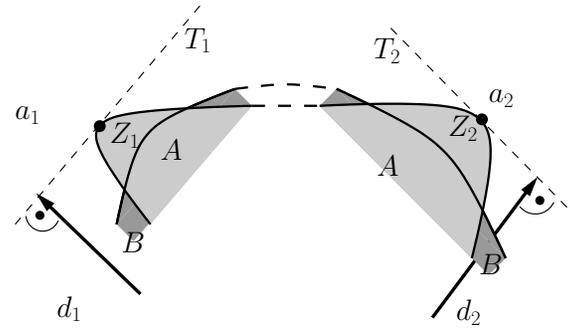
$A \setminus B$ ist zusammenhängend

$A \setminus B$ ist zusammenhängend



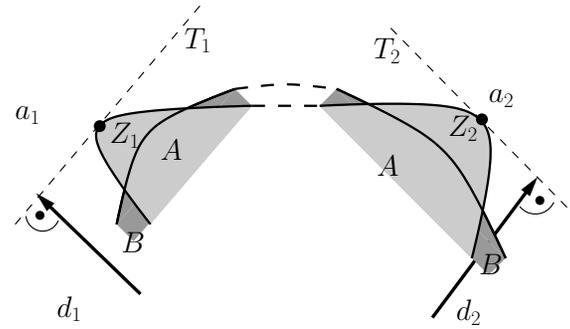
- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2

$A \setminus B$ ist zusammenhängend



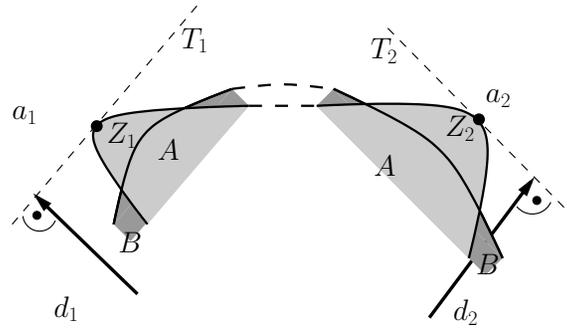
- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a):

$A \setminus B$ ist zusammenhängend



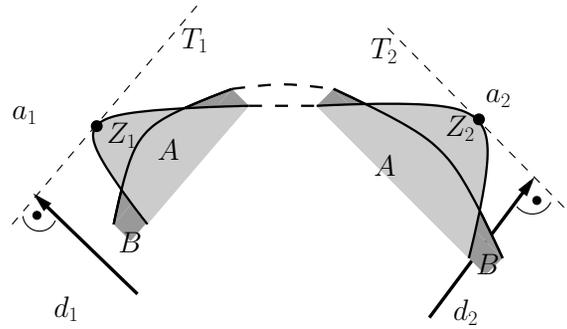
- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a): P_1 extremer als P_2 in d_1 und d_2

$A \setminus B$ ist zusammenhängend



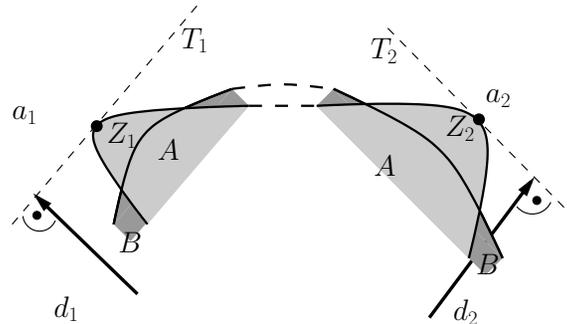
- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a): P_1 extremer als P_2 in d_1 und d_2
- Bem. 2.8 b):

$A \setminus B$ ist zusammenhängend



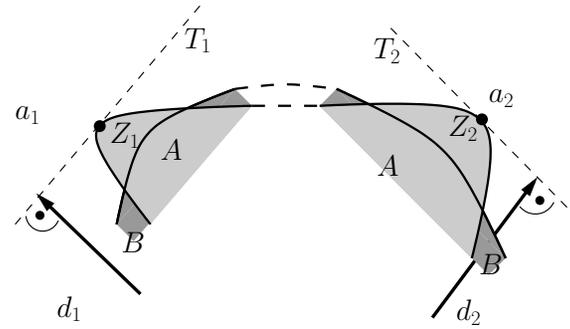
- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a): P_1 extremer als P_2 in d_1 und d_2
- Bem. 2.8 b): P_1 extr. als P_2 zw. allen d_1 und d_2 (eine Richtung!)

$A \setminus B$ ist zusammenhängend



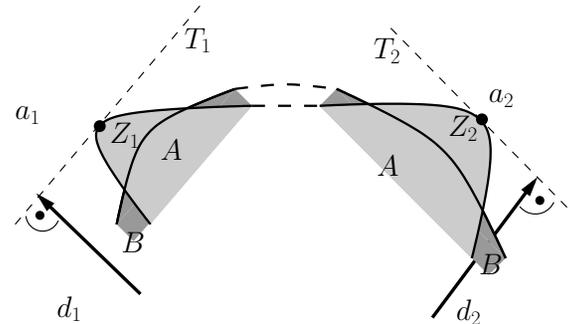
- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a): P_1 extremer als P_2 in d_1 und d_2
- Bem. 2.8 b): P_1 extr. als P_2 zw. allen d_1 und d_2 (eine Richtung!)
- Bem. 2.8 a):

$A \setminus B$ ist zusammenhängend



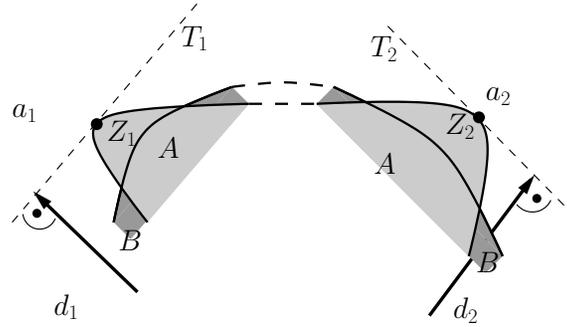
- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a): P_1 extremer als P_2 in d_1 und d_2
- Bem. 2.8 b): P_1 extr. als P_2 zw. allen d_1 und d_2 (eine Richtung!)
- Bem. 2.8 a): $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ zwischen allen d_1 und d_2 (eine Richtung!)

$A \setminus B$ ist zusammenhängend



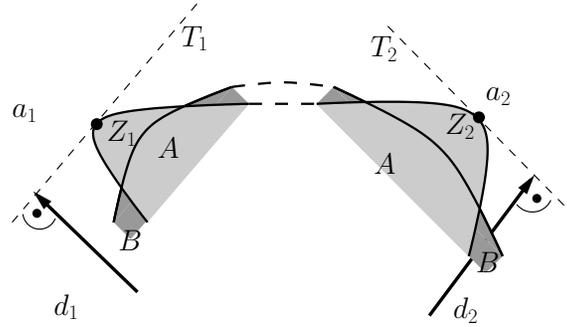
- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a): P_1 extremer als P_2 in d_1 und d_2
- Bem. 2.8 b): P_1 extr. als P_2 zw. allen d_1 und d_2 (eine Richtung!)
- Bem. 2.8 a): $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ zwischen allen d_1 und d_2 (eine Richtung!)
- Zwischen allen d_1 und d_2 (eine Richtung!) bleibt Z_i bestehen!!

$A \setminus B$ ist zusammenhängend



- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a): P_1 extremer als P_2 in d_1 und d_2
- Bem. 2.8 b): P_1 extr. als P_2 zw. allen d_1 und d_2 (eine Richtung!)
- Bem. 2.8 a): $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ zwischen allen d_1 und d_2 (eine Richtung!)
- Zwischen allen d_1 und d_2 (eine Richtung!) bleibt Z_i bestehen!!
- Es gibt nur eine Komponente!!

$A \setminus B$ ist zusammenhängend



- $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ in d_1 und d_2
- Bem. 2.8 a): P_1 extremer als P_2 in d_1 und d_2
- Bem. 2.8 b): P_1 extr. als P_2 zw. allen d_1 und d_2 (eine Richtung!)
- Bem. 2.8 a): $A = P_1 \oplus R$ extremer als $B = P_2 \oplus R$ zwischen allen d_1 und d_2 (eine Richtung!)
- Zwischen allen d_1 und d_2 (eine Richtung!) bleibt Z_i bestehen!!
- Es gibt nur eine Komponente!! $A \setminus B$ ist zusammenhängend!!

Folgerungen!! **Theorem 2.16**

Folgerungen!! **Theorem 2.16**

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten

Folgerungen!! **Theorem 2.16**

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten
- Komplexität von C_{verb} in $O(mn)$,

Folgerungen!! **Theorem 2.16**

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten
- Komplexität von C_{verb} in $O(mn)$, Beweis!!

Folgerungen!! **Theorem 2.16**

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten
- Komplexität von C_{verb} in $O(mn)$, Beweis!!
- Trianguliere alle P_j ,

Folgerungen!! **Theorem 2.16**

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten
- Komplexität von C_{verb} in $O(mn)$, Beweis!!
- Trianguliere alle $P_j, T_1, T_2, \dots, T_l$ mit $O(n)$ Kanten

Folgerungen!! **Theorem 2.16**

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten
- Komplexität von C_{verb} in $O(mn)$, Beweis!!
- Trianguliere alle $P_j, T_1, T_2, \dots, T_l$ mit $O(n)$ Kanten
- $CT_i = T_i \oplus -R(0, 0)$ Familie von Pseudokreisen **Lem. 2.12**,

Folgerungen!! **Theorem 2.16**

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten
- Komplexität von C_{verb} in $O(mn)$, Beweis!!
- Trianguliere alle $P_j, T_1, T_2, \dots, T_l$ mit $O(n)$ Kanten
- $CT_i = T_i \oplus -R(0, 0)$ Familie von Pseudokreisen **Lem. 2.12**,
 $O((m + 3)n)$ Kanten **Lem. 2.15**

Folgerungen!! Theorem 2.16

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten
- Komplexität von C_{verb} in $O(mn)$, Beweis!!
- Trianguliere alle $P_j, T_1, T_2, \dots, T_l$ mit $O(n)$ Kanten
- $CT_i = T_i \oplus -R(0, 0)$ Familie von Pseudokreisen **Lem. 2.12**,
 $O((m + 3)n)$ Kanten **Lem. 2.15**

$$C_{\text{verb}} = \underbrace{\bigcup_i T_i \oplus -R(0, 0)}_{\text{Komplexität } O(mn)}.$$

Folgerungen!! Theorem 2.16

- Konvexer Roboter R mit $|R| = m$, polygonale Szene, n Kanten
- Komplexität von C_{verb} in $O(mn)$, Beweis!!
- Trianguliere alle $P_j, T_1, T_2, \dots, T_l$ mit $O(n)$ Kanten
- $CT_i = T_i \oplus -R(0, 0)$ Familie von Pseudokreisen **Lem. 2.12**, $O((m + 3)n)$ Kanten **Lem. 2.15**

$$C_{\text{verb}} = \underbrace{\bigcup_i T_i \oplus -R(0, 0)}_{\text{Komplexität } O(mn)}.$$

Benutze **Theorem 2.13**

Folgerungen!! **Theorem 2.16**

Folgerungen!! **Theorem 2.16**

- Algorithmus 2.2: Berechnung von C_{verb}

Folgerungen!! **Theorem 2.16**

- Algorithmus 2.2: Berechnung von C_{verb}
 - Unterteile Hindernisse in gleichgroße Teilmengen von Dreiecken CP_1 und CP_2

Folgerungen!! Theorem 2.16

- Algorithmus 2.2: Berechnung von C_{verb}
 - Unterteile Hindernisse in gleichgroße Teilmengen von Dreiecken CP_1 und CP_2
 - Berechne rek. $C_{\text{verb}}(CP_1)$ und $C_{\text{verb}}(CP_2)$, je $O(mn)$ Kanten

Folgerungen!! Theorem 2.16

- Algorithmus 2.2: Berechnung von C_{verb}
 - Unterteile Hindernisse in gleichgroße Teilmengen von Dreiecken CP_1 und CP_2
 - Berechne rek. $C_{\text{verb}}(CP_1)$ und $C_{\text{verb}}(CP_2)$, je $O(mn)$ Kanten
 - Berechne Vereinigung $C_{\text{verb}}(CP_1) \cup C_{\text{verb}}(CP_2)$ in Zeit $O(mn \log(mn))$ mit Sweep (Alg. Geom.)

Folgerungen!! Theorem 2.16

- Algorithmus 2.2: Berechnung von C_{verb}
 - Unterteile Hindernisse in gleichgroße Teilmengen von Dreiecken CP_1 und CP_2
 - Berechne rek. $C_{\text{verb}}(CP_1)$ und $C_{\text{verb}}(CP_2)$, je $O(mn)$ Kanten
 - Berechne Vereinigung $C_{\text{verb}}(CP_1) \cup C_{\text{verb}}(CP_2)$ in Zeit $O(mn \log(mn))$ mit Sweep (Alg. Geom.)

Laufzeit insgesamt: $O(mn \log^2 mn)$

Laufzeitanalyse: **Theorem 2.16**

Laufzeitanalyse: **Theorem 2.16**

Arrangement mit $O(nm)$ Kanten!!

Laufzeitanalyse: **Theorem 2.16**

Arrangement mit $O(nm)$ Kanten!!

$$T(m, n) \leq 2T\left(m, \frac{n}{2}\right)$$

Laufzeitanalyse: **Theorem 2.16**

Arrangement mit $O(nm)$ Kanten!!

$$T(m, n) \leq 2T\left(m, \frac{n}{2}\right) \text{ (Rek.)}$$

Laufzeitanalyse: **Theorem 2.16**

Arrangement mit $O(nm)$ Kanten!!

$$T(m, n) \leq 2T\left(m, \frac{n}{2}\right) \text{ (Rek.)} + C \times mn \log mn$$

Laufzeitanalyse: **Theorem 2.16**

Arrangement mit $O(nm)$ Kanten!!

$$T(m, n) \leq 2T\left(m, \frac{n}{2}\right) \text{ (Rek.)} + C \times mn \log mn \text{ (Merge)}$$

Laufzeitanalyse: **Theorem 2.16**

Arrangement mit $O(nm)$ Kanten!!

$$\begin{aligned} T(m, n) &\leq 2T\left(m, \frac{n}{2}\right) \text{ (Rek.)} + C \times mn \log mn \text{ (Merge)} \\ &\leq 2\left(2T\left(m, \frac{n}{4}\right) + C\frac{mn}{2} \log \frac{mn}{2}\right) + C \times mn \log mn \end{aligned}$$

Laufzeitanalyse: **Theorem 2.16**

Arrangement mit $O(nm)$ Kanten!!

$$\begin{aligned} T(m, n) &\leq 2T\left(m, \frac{n}{2}\right) \text{ (Rek.)} + C \times mn \log mn \text{ (Merge)} \\ &\leq 2\left(2T\left(m, \frac{n}{4}\right) + C\frac{mn}{2} \log \frac{mn}{2}\right) + C \times mn \log mn \\ &\vdots \quad \vdots \end{aligned}$$

Laufzeitanalyse: **Theorem 2.16**

Arrangement mit $O(nm)$ Kanten!!

$$\begin{aligned} T(m, n) &\leq 2T\left(m, \frac{n}{2}\right) \text{ (Rek.)} + C \times mn \log mn \text{ (Merge)} \\ &\leq 2\left(2T\left(m, \frac{n}{4}\right) + C\frac{mn}{2} \log \frac{mn}{2}\right) + C \times mn \log mn \\ &\quad \vdots \quad \vdots \\ &\leq nT(m, 1) + Cmn \sum_{i=0}^{\log n} \log \frac{mn}{2^i} \end{aligned}$$

Laufzeitanalyse: **Theorem 2.16**

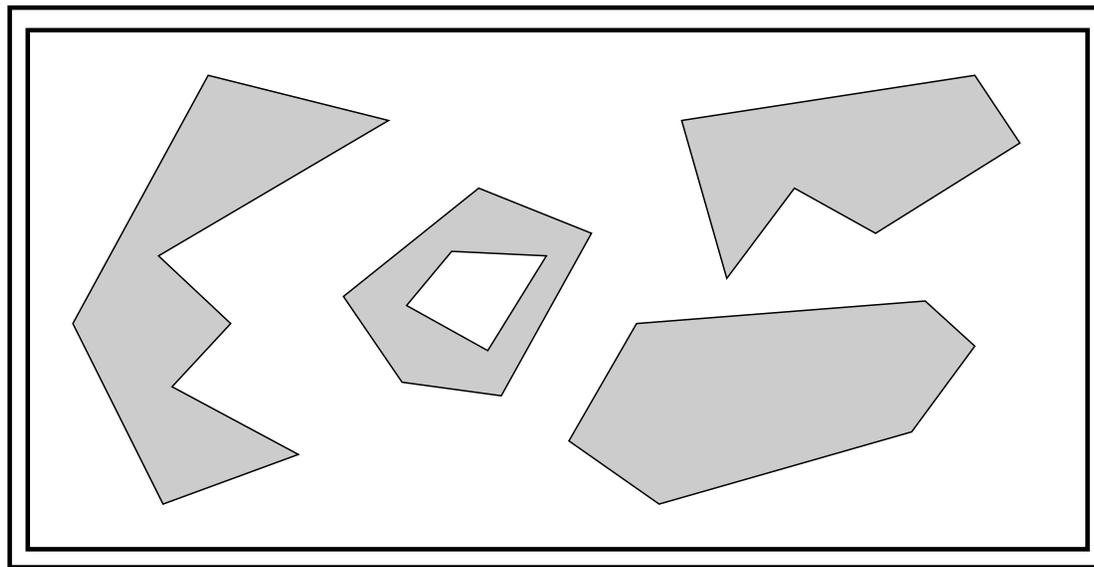
Arrangement mit $O(nm)$ Kanten!!

$$\begin{aligned} T(m, n) &\leq 2T\left(m, \frac{n}{2}\right) \text{ (Rek.)} + C \times mn \log mn \text{ (Merge)} \\ &\leq 2\left(2T\left(m, \frac{n}{4}\right) + C\frac{mn}{2} \log \frac{mn}{2}\right) + C \times mn \log mn \\ &\quad \vdots \quad \vdots \\ &\leq nT(m, 1) + Cmn \sum_{i=0}^{\log n} \log \frac{mn}{2^i} \\ &\in O(mn \log^2 mn) \end{aligned}$$

C_{frei} verwenden! Roadmap!!

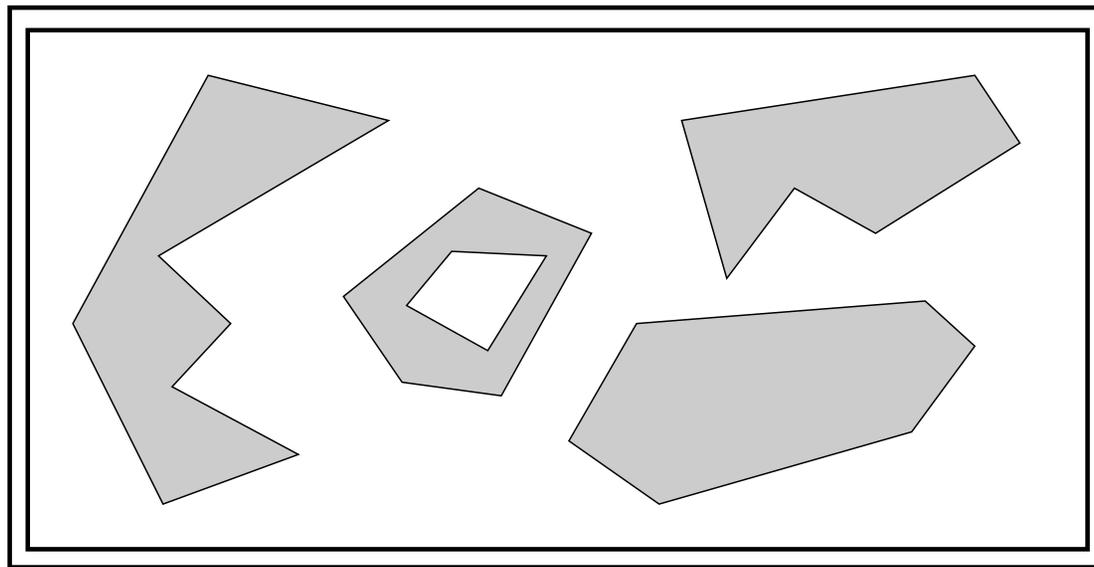
C_{frei} verwenden! Roadmap!!

- Zerlegung in Trapezen (Seidel)



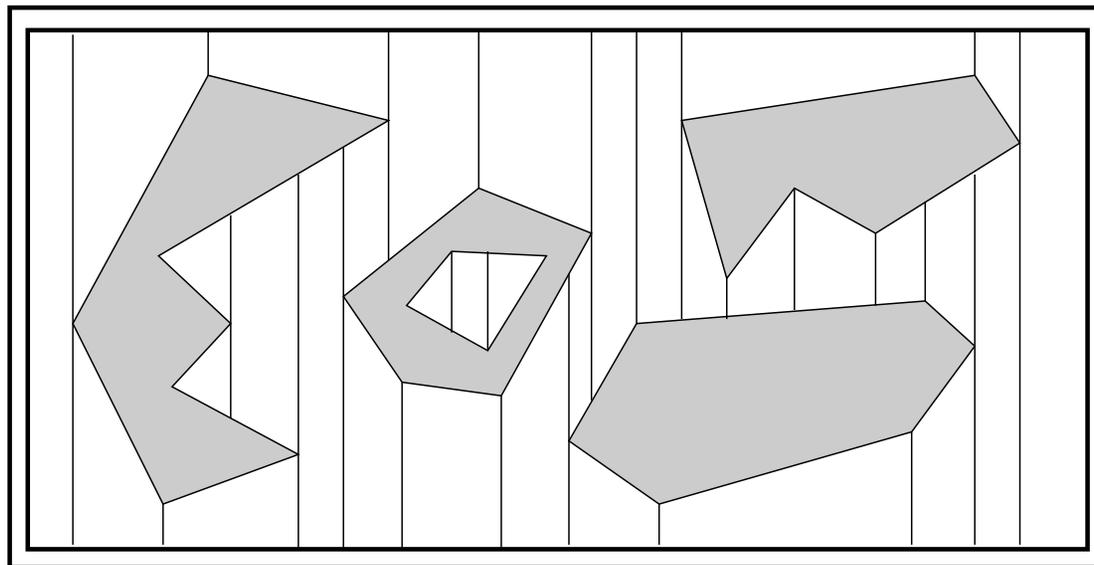
C_{frei} verwenden! Roadmap!!

- Zerlegung in Trapezen (Seidel)
- Benachbarte Trapeze verbinden (Schwerpunkt/Mittelpunkt)



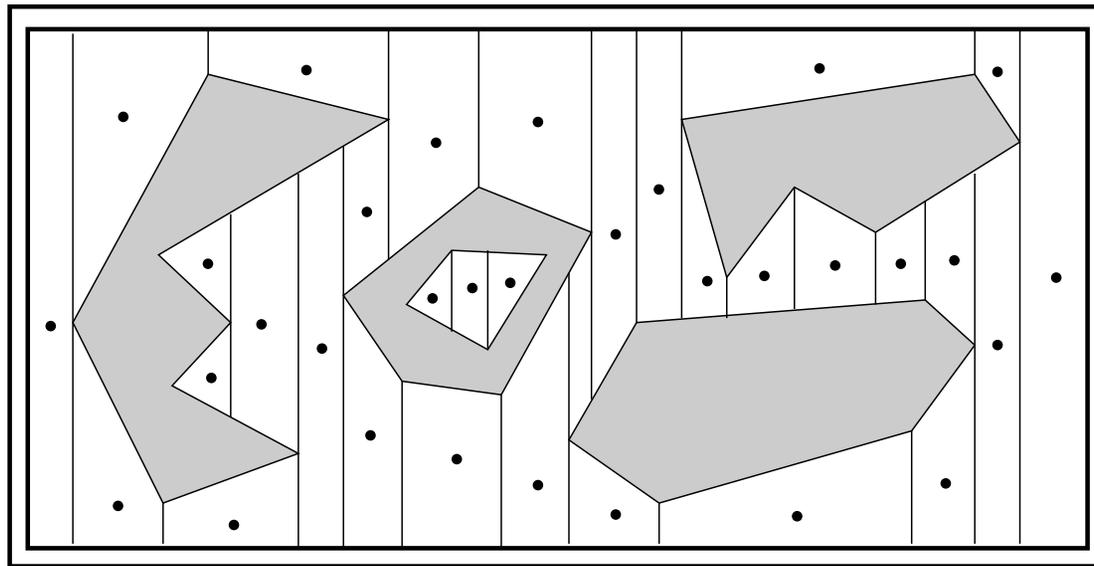
C_{frei} verwenden! Roadmap!!

- Zerlegung in Trapezen (Seidel)
- Benachbarte Trapeze verbinden (Schwerpunkt/Mittelpunkt)



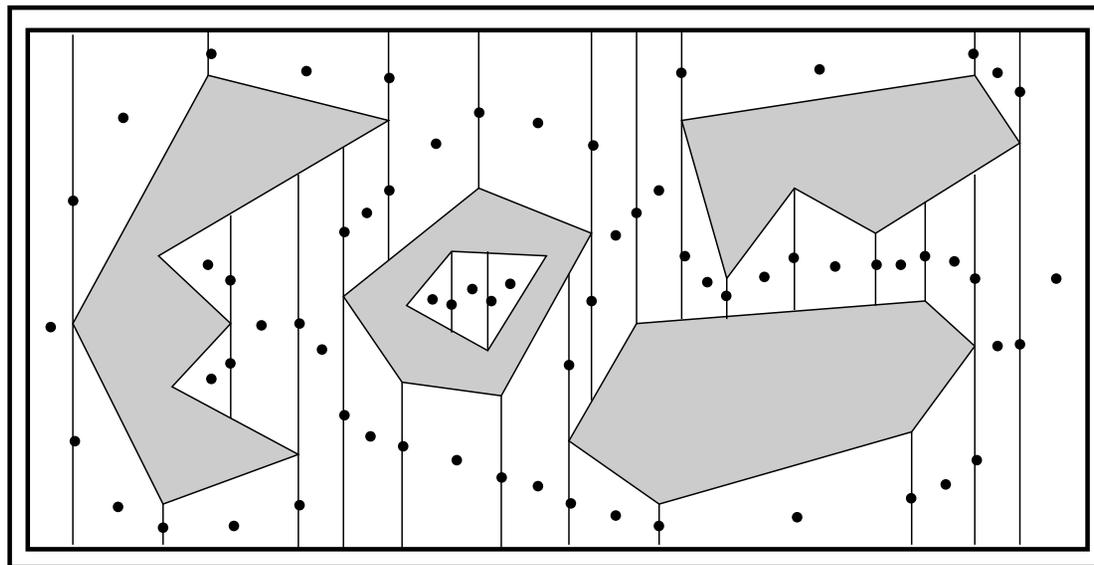
C_{frei} verwenden! Roadmap!!

- Zerlegung in Trapezen (Seidel)
- Benachbarte Trapeze verbinden (Schwerpunkt/Mittelpunkt)



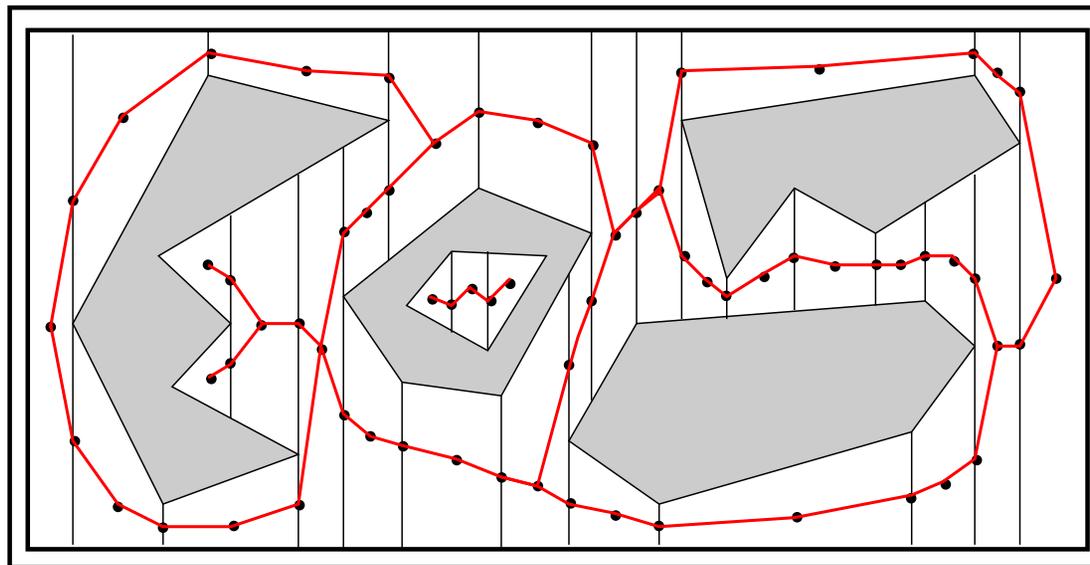
C_{frei} verwenden! Roadmap!!

- Zerlegung in Trapezen (Seidel)
- Benachbarte Trapeze verbinden (Schwerpunkt/Mittelpunkt)



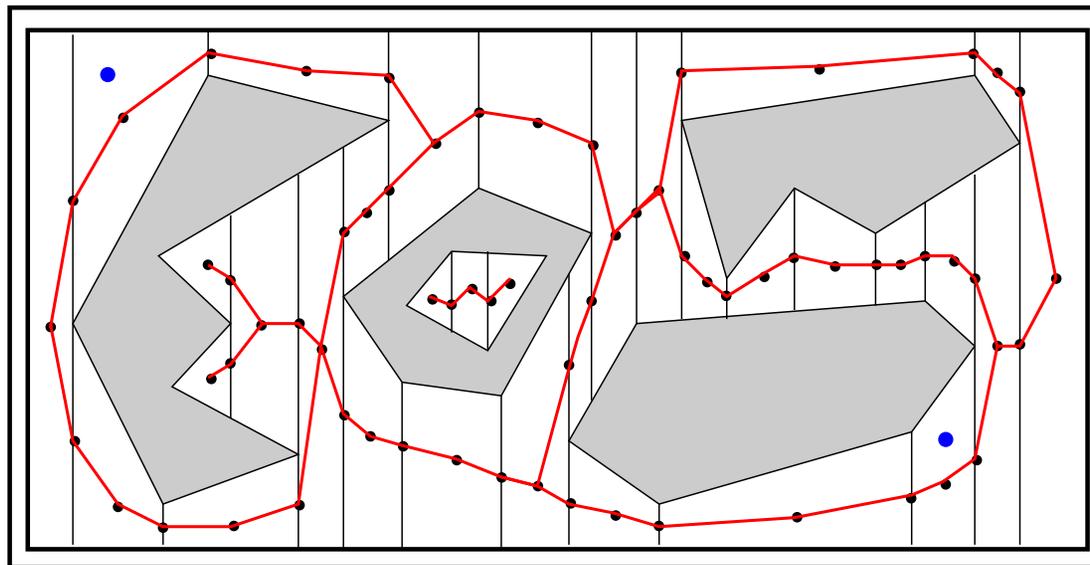
C_{frei} verwenden! Roadmap!!

- Zerlegung in Trapezen (Seidel)
- Benachbarte Trapeze verbinden (Schwerpunkt/Mittelpunkt)
- Zusammenhangsgraph, Roadmap



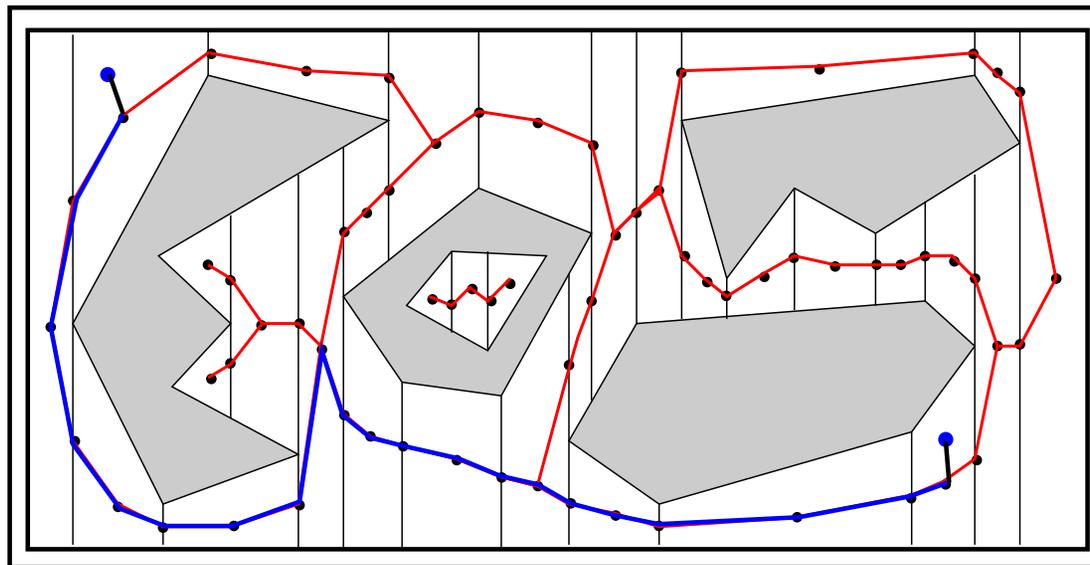
C_{frei} verwenden! Roadmap!!

- Zerlegung in Trapezen (Seidel)
- Benachbarte Trapeze verbinden (Schwerpunkt/Mittelpunkt)
- Zusammenhangsgraph, Roadmap
- Lokalisation s und t und verbinden (BFS)



C_{frei} verwenden! Roadmap!!

- Zerlegung in Trapezen (Seidel)
- Benachbarte Trapeze verbinden (Schwerpunkt/Mittelpunkt)
- Zusammenhangsgraph, Roadmap
- Lokalisation s und t und verbinden (BFS)



Algorithmus 2.3: C_{frei} gegeben!

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point-Location Struktur,

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point-Location Struktur, (randomisiert!!) $O(mn \log(mn))$

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point-Location Struktur, (randomisiert!!) $O(mn \log(mn))$
- Roadmap-Konstruktion: Verbindung adjazenter Trapeze,

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point–Location Struktur, (randomisiert!!) $O(mn \log(mn))$
- Roadmap-Konstruktion: Verbindung adjazenter Trapeze, $O(mn)$

Query:

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point-Location Struktur, (randomisiert!!) $O(mn \log(mn))$
- Roadmap-Konstruktion: Verbindung adjazenter Trapeze, $O(mn)$

Query:

- Lokalisierere s und t Trapeze,

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point-Location Struktur, (randomisiert!!) $O(mn \log(mn))$
- Roadmap-Konstruktion: Verbindung adjazenter Trapeze, $O(mn)$

Query:

- Lokalisierere s und t Trapeze, $O(\log(mn))$

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point-Location Struktur, (randomisiert!!) $O(mn \log(mn))$
- Roadmap-Konstruktion: Verbindung adjazenter Trapeze, $O(mn)$

Query:

- Lokalisierere s und t Trapeze, $O(\log(mn))$
- Finde Weg der Trapeze mit BFS,

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point-Location Struktur, (randomisiert!!) $O(mn \log(mn))$
- Roadmap-Konstruktion: Verbindung adjazenter Trapeze, $O(mn)$

Query:

- Lokalisierere s und t Trapeze, $O(\log(mn))$
- Finde Weg der Trapeze mit BFS, $O(mn)$

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point–Location Struktur, (randomisiert!!) $O(mn \log(mn))$
- Roadmap-Konstruktion: Verbindung adjazenter Trapeze, $O(mn)$

Query:

- Lokalisierere s und t Trapeze, $O(\log(mn))$
- Finde Weg der Trapeze mit BFS, $O(mn)$

Roadmap!! Einfach aber effizient!!

Algorithmus 2.3: C_{frei} gegeben!

Vorbereitung:

- Trapezzerlegung C_{frei} /Point-Location Struktur, (randomisiert!!) $O(mn \log(mn))$
- Roadmap-Konstruktion: Verbindung adjazenter Trapeze, $O(mn)$

Query:

- Lokalisierere s und t Trapeze, $O(\log(mn))$
- Finde Weg der Trapeze mit BFS, $O(mn)$

Roadmap!! Einfach aber effizient!! Probabilistisch!

Ergebnis: **Theorem 2.17**

Ergebnis: **Theorem 2.17**

Translationsbewegungen eines konvexen, polygonalen Roboters mit m Ecken in einer Umgebung mit polygonalen Hindernissen mit insgesamt n Ecken können nach $O(mn \log^2(mn))$ (randomisierter) Vorbereitungszeit in Zeit $O(mn)$ geplant werden.

Kreisförmiger! Voronoi Diagramme 2.1.1

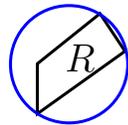
Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter



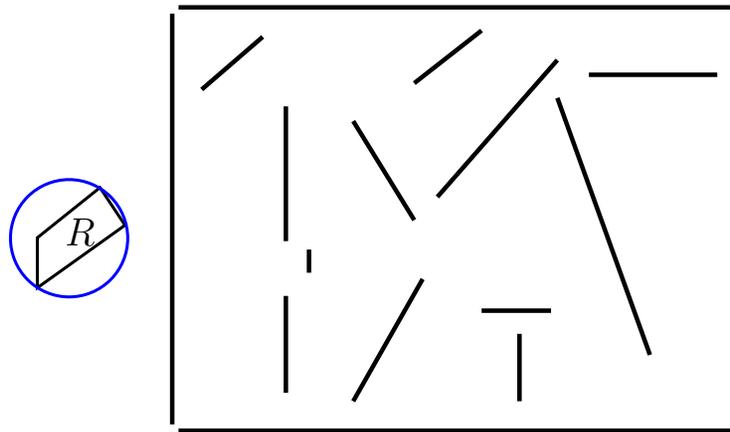
Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter
- Voronoi Diagramm der Segmente der Hindernisse



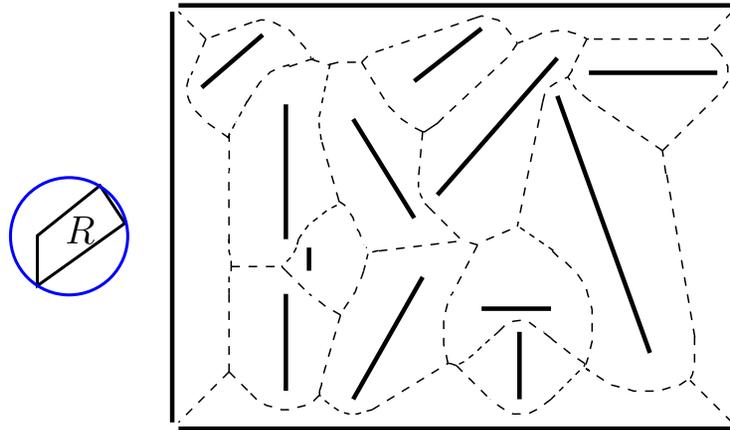
Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter
- Voronoi Diagramm der Segmente der Hindernisse
- Weg auf Bisektoren:



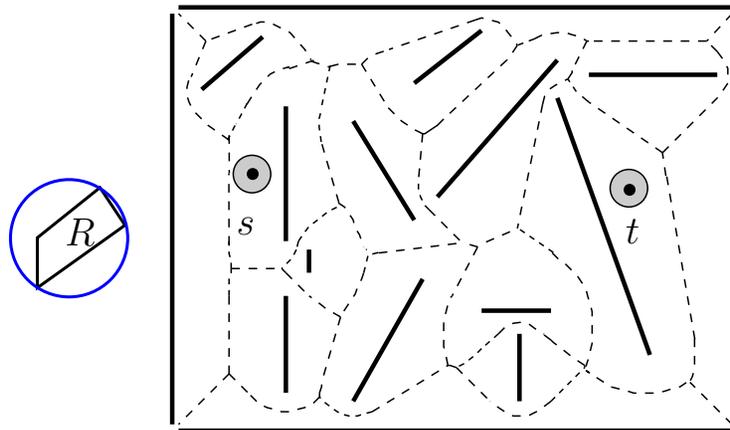
Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter
- Voronoi Diagramm der Segmente der Hindernisse
- Weg auf Bisektoren:



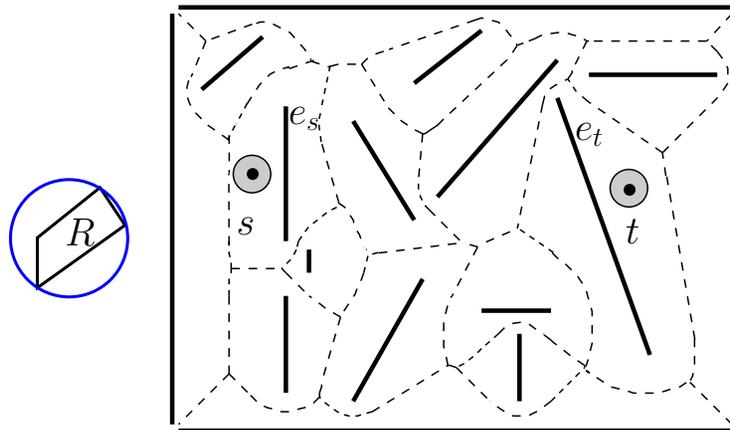
Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter
- Voronoi Diagramm der Segmente der Hindernisse
- Weg auf Bisektoren:



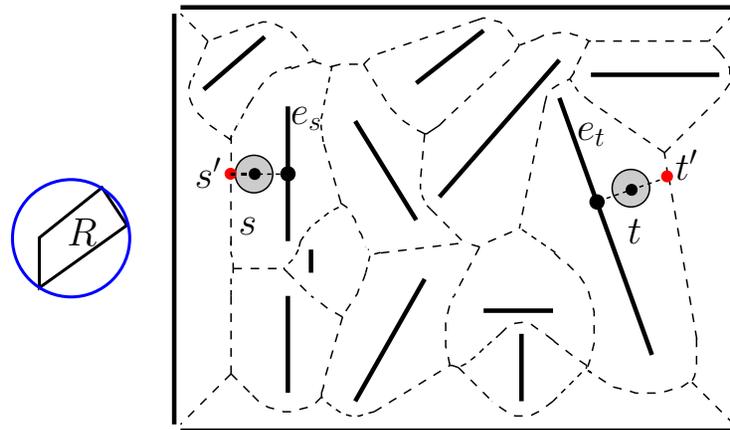
Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter
- Voronoi Diagramm der Segmente der Hindernisse
- Weg auf Bisektoren:



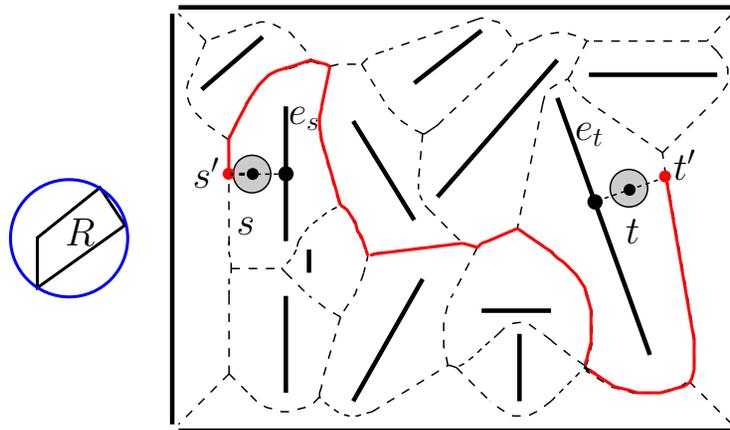
Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter
- Voronoi Diagramm der Segmente der Hindernisse
- Weg auf Bisektoren:



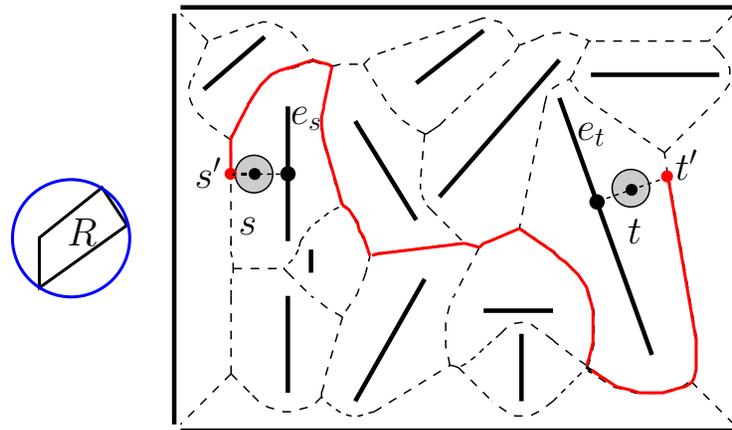
Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter
- Voronoi Diagramm der Segmente der Hindernisse
- Weg auf Bisektoren:

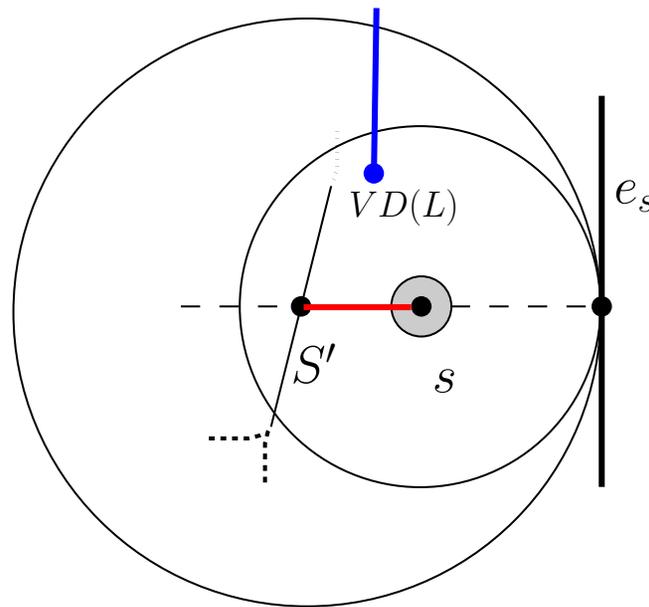


Kreisförmiger! Voronoi Diagramme 2.1.1

- Verwende kleinsten Kreis um Roboter
- Voronoi Diagramm der Segmente der Hindernisse
- Weg auf Bisektoren: Möglichst großer Abstand

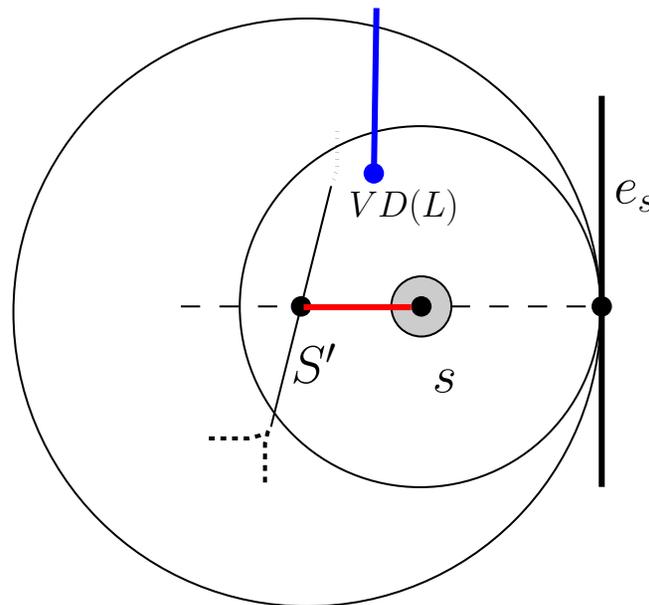


Start s' kann stets angelaufen werden



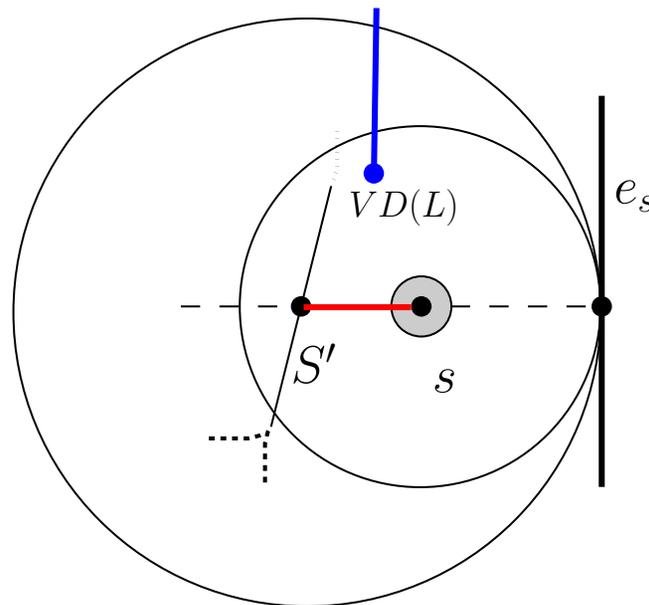
Start s' kann stets angelaufen werden

- s in Region von e_s ,



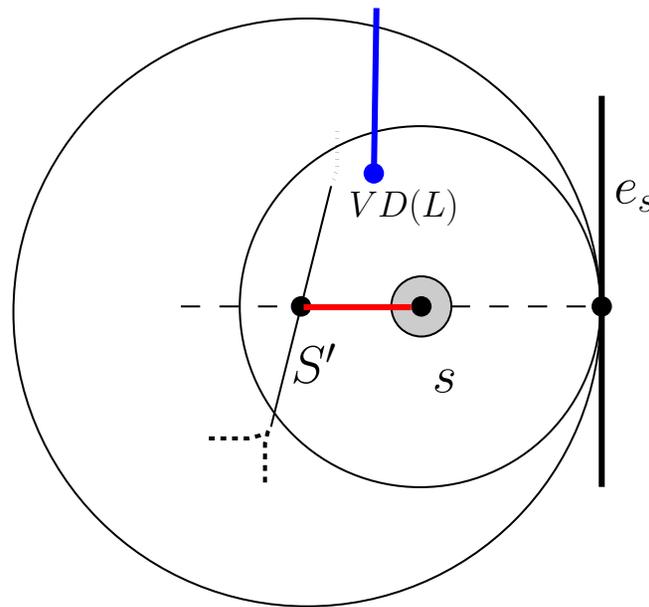
Start s' kann stets angelaufen werden

- s in Region von e_s , Kreis frei



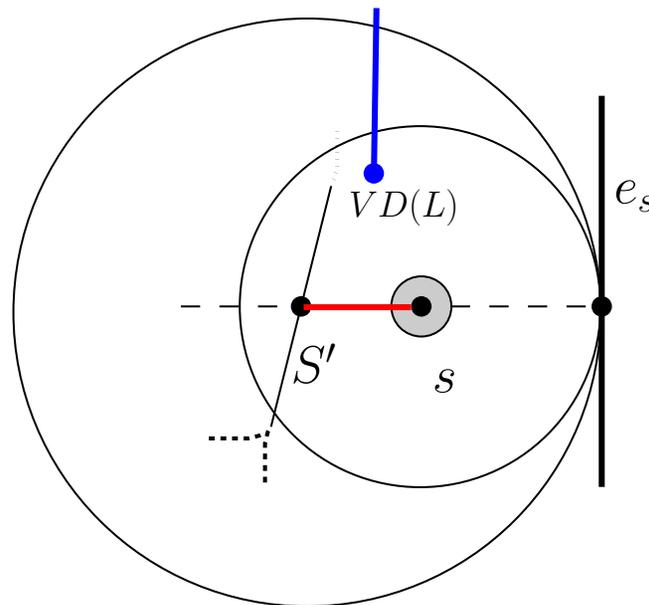
Start s' kann stets angelaufen werden

- s in Region von e_s , Kreis frei
- Kürzester Weg zu e_s , Strahl Richtung Bisector



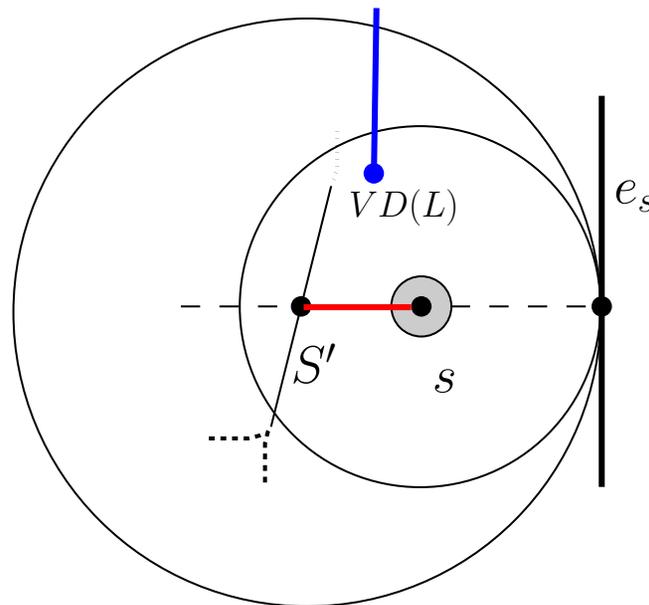
Start s' kann stets angelaufen werden

- s in Region von e_s , Kreis frei
- Kürzester Weg zu e_s , Strahl Richtung Bisector
- Trifft Bisector bei s' ,



Start s' kann stets angelaufen werden

- s in Region von e_s , Kreis frei
- Kürzester Weg zu e_s , Strahl Richtung Bisector
- Trifft Bisector bei s' , Weg ist frei!!



Algorithmus 2.1: Kreisförmiger Roboter

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente:

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung:

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Query für zwei Punkte s, t :

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Query für zwei Punkte s, t :

- Lokalisierere s, t im VD:

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Query für zwei Punkte s, t :

- Lokalisierere s, t im VD: $O(\log n)$

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Query für zwei Punkte s, t :

- Lokalisierere s, t im VD: $O(\log n)$
- s gehört zu e_s , t gehört zu e_t

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Query für zwei Punkte s, t :

- Lokalisierere s, t im VD: $O(\log n)$
- s gehört zu e_s , t gehört zu e_t
- Bestimme mit Strahlen Punkt t' und s' auf $VD(L)$:

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Query für zwei Punkte s, t :

- Lokalisierere s, t im VD: $O(\log n)$
- s gehört zu e_s , t gehört zu e_t
- Bestimme mit Strahlen Punkt t' und s' auf $VD(L)$: $O(1)$

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Query für zwei Punkte s, t :

- Lokalisierere s, t im VD: $O(\log n)$
- s gehört zu e_s , t gehört zu e_t
- Bestimme mit Strahlen Punkt t' und s' auf $VD(L)$: $O(1)$
- Berechne (BFS) in $VD(L)$ Pfad von s' nach t' mit zul. Mindestabstand oder berichte, dass es keinen gibt:

Algorithmus 2.1: Kreisförmiger Roboter

Vorbereitung:

- Konstruiere das Voronoi–Diagramm $VD(L)$, Kanten der Hindernisse als Liniensegmente: $O(n \log n)$
- Point–Location Trapezzerlegung: $O(n \log n)$

Query für zwei Punkte s, t :

- Lokalisierere s, t im VD: $O(\log n)$
- s gehört zu e_s , t gehört zu e_t
- Bestimme mit Strahlen Punkt t' und s' auf $VD(L)$: $O(1)$
- Berechne (BFS) in $VD(L)$ Pfad von s' nach t' mit zul. Mindestabstand oder berichte, dass es keinen gibt: $O(n)$