

Offline Bewegungsplanung: Kritische Orientierungen

Elmar Langetepe
University of Bonn

Weitere Vorgehensweise!

Weitere Vorgehensweise!

- Bis jetzt:

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch
 - Nötige Änderungen:

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch
 - Nötige Änderungen: Anzahl $O(mn\lambda_6(mn))$ Orientierungen,

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch
 - Nötige Änderungen: Anzahl $O(mn\lambda_6(mn))$ Orientierungen, Aktualisierung für alle diese: $O(mn\lambda_6(mn) \log(mn))$

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch
 - Nötige Änderungen: Anzahl $O(mn\lambda_6(mn))$ Orientierungen, Aktualisierung für alle diese: $O(mn\lambda_6(mn) \log(mn))$
 - Leider nur Obermenge berechnet:

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch
 - Nötige Änderungen: Anzahl $O(mn\lambda_6(mn))$ Orientierungen, Aktualisierung für alle diese: $O(mn\lambda_6(mn) \log(mn))$
 - Leider nur Obermenge berechnet: Kandidaten (z.B. Krit. Platz.)

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch
 - Nötige Änderungen: Anzahl $O(mn\lambda_6(mn))$ Orientierungen, Aktualisierung für alle diese: $O(mn\lambda_6(mn) \log(mn))$
 - Leider nur Obermenge berechnet: Kandidaten (z.B. Krit. Platz.)
 - Eventuell nicht frei!

Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch
 - Nötige Änderungen: Anzahl $O(mn\lambda_6(mn))$ Orientierungen, Aktualisierung für alle diese: $O(mn\lambda_6(mn) \log(mn))$
 - Leider nur Obermenge berechnet: Kandidaten (z.B. Krit. Platz.)
 - Eventuell nicht frei!
- Aufgabe:

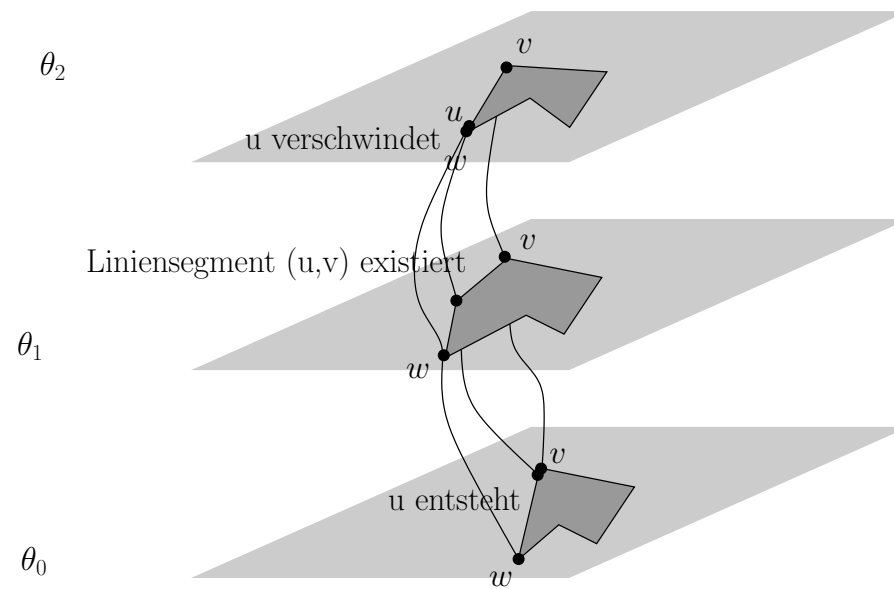
Weitere Vorgehensweise!

- Bis jetzt:
 - Graph V^θ für feste Orientierung
 - Zusammenhang in V^θ und ∂C_{frei}^θ sind identisch
 - Nötige Änderungen: Anzahl $O(mn\lambda_6(mn))$ Orientierungen, Aktualisierung für alle diese: $O(mn\lambda_6(mn) \log(mn))$
 - Leider nur Obermenge berechnet: Kandidaten (z.B. Krit. Platz.)
 - Eventuell nicht frei!
- Aufgabe:
 - Alle diese Informationen in einer Struktur für die Bahnplanung abspeichern, zusammenfassen

– Dabei: Die freien herausfiltern

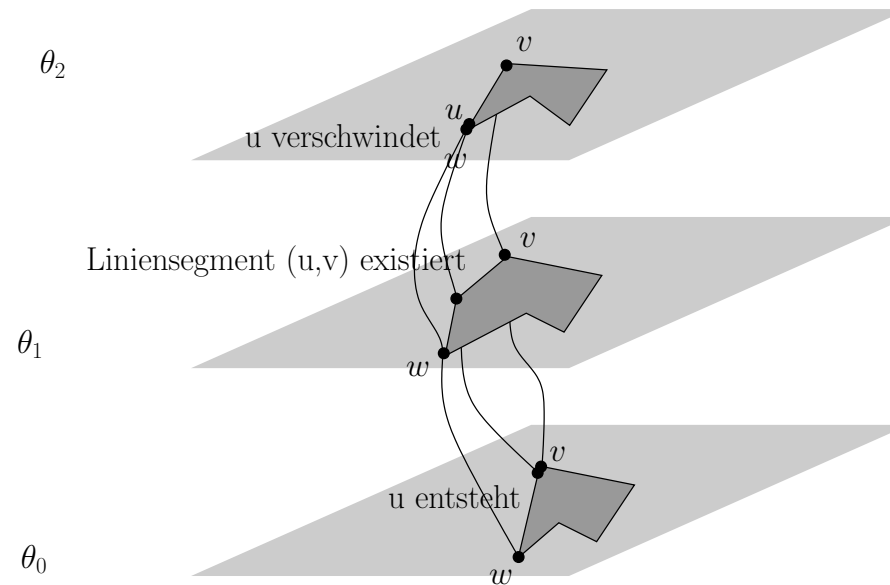
- Dabei: Die freien herausfiltern
- Bahnplanung mit Struktur

Struktur: Kantengraph **Def.: 2.43**



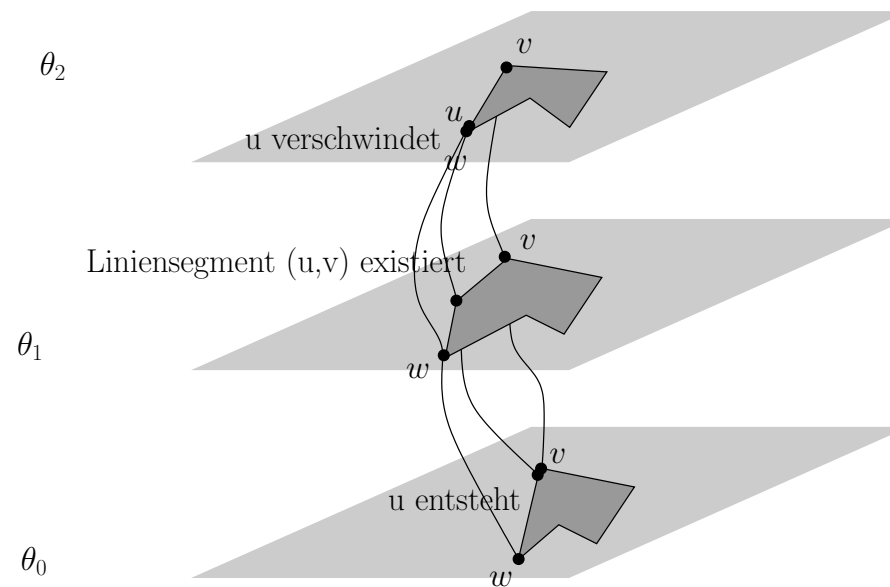
Struktur: Kantengraph **Def.: 2.43**

- Wie lange überlebt ein Knoten in C_{frei}^θ



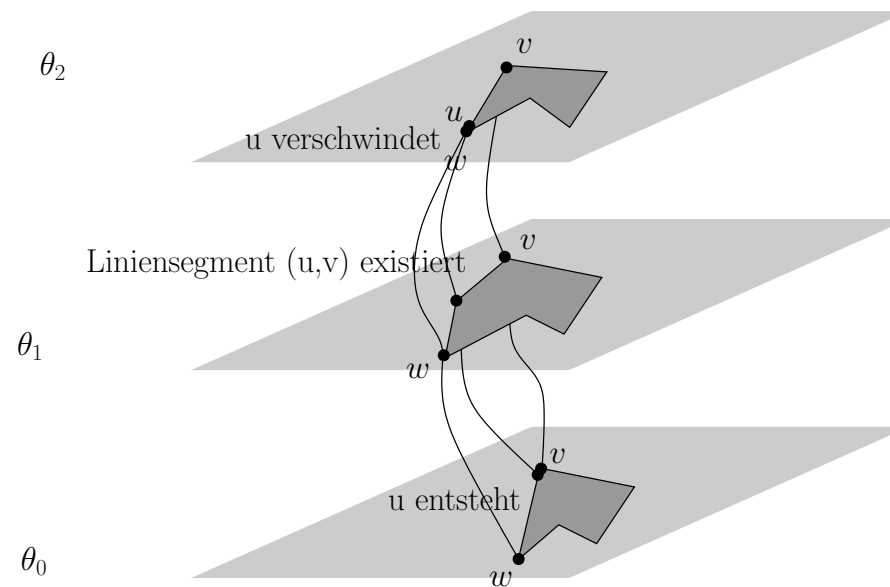
Struktur: Kantengraph **Def.: 2.43**

- Wie lange überlebt ein Knoten in C_{frei}^θ
- Wie lange überlebt eine Kante zwischen Knoten in C_{frei}^θ



Struktur: Kantengraph **Def.: 2.43**

- Wie lange überlebt ein Knoten in C_{frei}^θ
- Wie lange überlebt eine Kante zwischen Knoten in C_{frei}^θ
- Bilden eine gekrümmte Fläche in C_{frei} ?



Formal: Kantengraph **Def.: 2.42**

Formal: Kantengraph Def.: 2.42

- θ_0 nicht-kritische Orientierung, u Knoten in V^{θ_0} (Beschriftung)

Formal: Kantengraph Def.: 2.42

- θ_0 nicht-kritische Orientierung, u Knoten in V^{θ_0} (Beschriftung)
- Lebensdauer von u : max. Intervall $L(u) = (\theta_1, \theta_2) \ni \theta_0$, Knoten u kommt in jedem Knotengraph V^θ , $\theta \in L(u)$, vor

Formal: Kantengraph Def.: 2.42

- θ_0 nicht-kritische Orientierung, u Knoten in V^{θ_0} (Beschriftung)
- Lebensdauer von u : max. Intervall $L(u) = (\theta_1, \theta_2) \ni \theta_0$, Knoten u kommt in jedem Knotengraph V^θ , $\theta \in L(u)$, vor
- Kantengraph E :

Formal: Kantengraph Def.: 2.42

- θ_0 nicht-kritische Orientierung, u Knoten in V^{θ_0} (Beschriftung)
- Lebensdauer von u : max. Intervall $L(u) = (\theta_1, \theta_2) \ni \theta_0$, Knoten u kommt in jedem Knotengraph V^θ , $\theta \in L(u)$, vor
- Kantengraph E :
 - erweiterte Knoten $(u, L(u))$

Formal: Kantengraph Def.: 2.42

- θ_0 nicht-kritische Orientierung, u Knoten in V^{θ_0} (Beschriftung)
- Lebensdauer von u : max. Intervall $L(u) = (\theta_1, \theta_2) \ni \theta_0$, Knoten u kommt in jedem Knotengraph V^θ , $\theta \in L(u)$, vor
- Kantengraph E :
 - erweiterte Knoten $(u, L(u))$
 - Knoten $(u, L(u))$ und $(v, L(v))$ bilden Kante e in E $:\Leftrightarrow$ es ex. $\theta \in L(u) \cap L(v)$ mit V^θ hat Kante (u, v)

Formal: Kantengraph Def.: 2.42

- θ_0 nicht-kritische Orientierung, u Knoten in V^{θ_0} (Beschriftung)
- Lebensdauer von u : max. Intervall $L(u) = (\theta_1, \theta_2) \ni \theta_0$, Knoten u kommt in jedem Knotengraph $V^\theta, \theta \in L(u)$, vor
- Kantengraph E :
 - erweiterte Knoten $(u, L(u))$
 - Knoten $(u, L(u))$ und $(v, L(v))$ bilden Kante e in $E \Leftrightarrow$ es ex. $\theta \in L(u) \cap L(v)$ mit V^θ hat Kante (u, v)
 - Lebensdauer Kante e : max. Intervall $L(e) = (\theta_1, \theta_2) \ni \theta_0$, Kante (u, v) kommt in jedem Knotengraph $V^\theta, \theta \in L(e)$ vor

Formal: Kantengraph Def.: 2.42

- θ_0 nicht-kritische Orientierung, u Knoten in V^{θ_0} (Beschriftung)
- Lebensdauer von u : max. Intervall $L(u) = (\theta_1, \theta_2) \ni \theta_0$, Knoten u kommt in jedem Knotengraph V^θ , $\theta \in L(u)$, vor
- Kantengraph E :
 - erweiterte Knoten $(u, L(u))$
 - Knoten $(u, L(u))$ und $(v, L(v))$ bilden Kante e in E $:\Leftrightarrow$ es ex. $\theta \in L(u) \cap L(v)$ mit V^θ hat Kante (u, v)
 - Lebensdauer Kante e : max. Intervall $L(e) = (\theta_1, \theta_2) \ni \theta_0$, Kante (u, v) kommt in jedem Knotengraph V^θ , $\theta \in L(e)$ vor
 - $e = ((u, L(u)), (v, L(v)))$ und $L(e)$ ergibt $(e, L(e))$

Berechnung der Krit. Orient. 2.3.3

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* :

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$
2. Zyklische Tripel (O_j, O_k, O_l) mit Krit. Platz.

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$
2. Zyklische Tripel (O_j, O_k, O_l) mit Krit. Platz.
3. Krit.Platzierung vom Typ: Ecke/Ecke plus Ecke/Kante

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$
2. Zyklische Tripel (O_j, O_k, O_l) mit Krit. Platz.
3. Krit.Platzierung vom Typ: Ecke/Ecke plus Ecke/Kante
4. Orient.: Kante des erweiterten Hindernisse ist waagerecht

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$
2. Zyklische Tripel (O_j, O_k, O_l) mit Krit. Platz.
3. Krit.Platzierung vom Typ: Ecke/Ecke plus Ecke/Kante
4. Orient.: Kante des erweiterten Hindernisse ist waagerecht
5. Orient.: Konv. Ecke von C_{frei}^θ liegt unterhalb and. Ecken

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$
 2. Zyklische Tripel (O_j, O_k, O_l) mit Krit. Platz.
 3. Krit.Platzierung vom Typ: Ecke/Ecke plus Ecke/Kante
 4. Orient.: Kante des erweiterten Hindernisse ist waagerecht
 5. Orient.: Konv. Ecke von C_{frei}^θ liegt unterhalb and. Ecken
- 1 und 2 zu T^+ : Obermenge von Krit. Platz. i)-ii) (Alg. 2.8)

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$
 2. Zyklische Tripel (O_j, O_k, O_l) mit Krit. Platz.
 3. Krit.Platzierung vom Typ: Ecke/Ecke plus Ecke/Kante
 4. Orient.: Kante des erweiterten Hindernisse ist waagerecht
 5. Orient.: Konv. Ecke von C_{frei}^θ liegt unterhalb and. Ecken
- 1 und 2 zu T^+ : Obermenge von Krit. Platz. i)-ii) (Alg. 2.8)
 - 3 war Übungsaufgabe (und Alg. 2.9): Krit. Platz. iii)-iv)

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$
 2. Zyklische Tripel (O_j, O_k, O_l) mit Krit. Platz.
 3. Krit.Platzierung vom Typ: Ecke/Ecke plus Ecke/Kante
 4. Orient.: Kante des erweiterten Hindernisse ist waagerecht
 5. Orient.: Konv. Ecke von C_{frei}^θ liegt unterhalb and. Ecken
- 1 und 2 zu T^+ : Obermenge von Krit. Platz. i)-ii) (Alg. 2.8)
 - 3 war Übungsaufgabe (und Alg. 2.9): Krit. Platz. iii)-iv)
 - 4 und 5: Obermenge von Krit. Orient. v) und vi), Alg. folgt aus Beschreibung vorab

Berechnung der Krit. Orient. 2.3.3

Def: Menge T^* : Berechnet in $O(mn\lambda_6(mn)\log(mn))$

1. Eckpunkte der unteren Konturen $UK(A(O_i))$
 2. Zyklische Tripel (O_j, O_k, O_l) mit Krit. Platz.
 3. Krit.Platzierung vom Typ: Ecke/Ecke plus Ecke/Kante
 4. Orient.: Kante des erweiterten Hindernisse ist waagerecht
 5. Orient.: Konv. Ecke von C_{frei}^θ liegt unterhalb and. Ecken
- 1 und 2 zu T^+ : Obermenge von Krit. Platz. i)-ii) (Alg. 2.8)
 - 3 war Übungsaufgabe (und Alg. 2.9): Krit. Platz. iii)-iv)
 - 4 und 5: Obermenge von Krit. Orient. v) und vi), Alg. folgt aus Beschreibung vorab

Problem: Orientierungen aus T^* sind eventuell nicht frei!!

T^* Obermenge der gültigen Krit. Orient.

T^* Obermenge der gültigen Krit. Orient.

Menge T^* : Bereits berechnet in $O(mn\lambda_6(mn)\log(mn))$

T^* Obermenge der gültigen Krit. Orient.

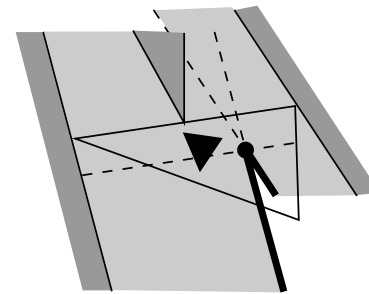
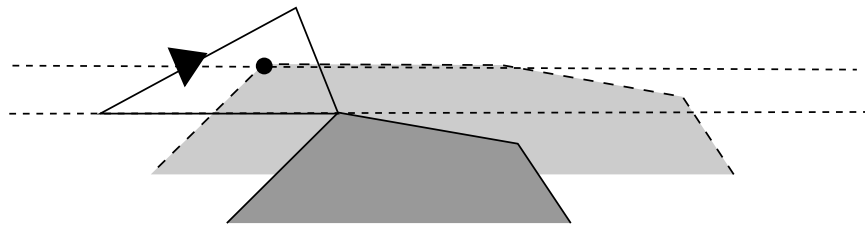
Menge T^* : Bereits berechnet in $O(mn\lambda_6(mn)\log(mn))$

Beispiele: 1. Krit. Platzierung! 4. Waagerechte Kante! Nicht frei!

T^* Obermenge der gültigen Krit. Orient.

Menge T^* : Bereits berechnet in $O(mn\lambda_6(mn)\log(mn))$

Beispiele: 1. Krit. Platzierung! 4. Waagerechte Kante! Nicht frei!



Idee des Filterns

Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?

Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5) : (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher $(\theta_1 = \theta_2 - \epsilon)$,

Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5) : (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2 , O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten,

Idee des Filterns

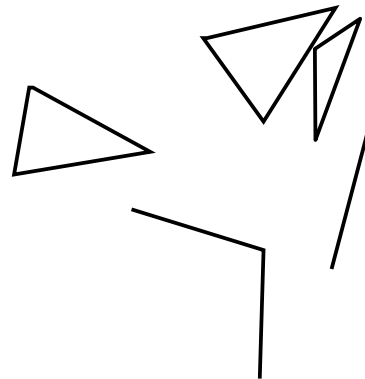
- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5) : (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2 , O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen

Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5) : (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2 , O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O :

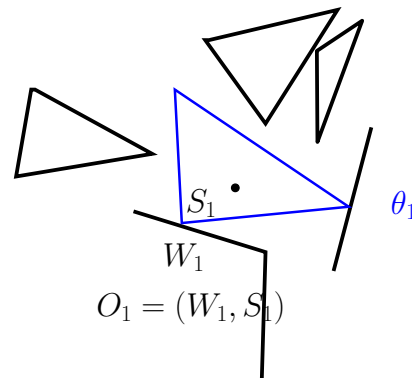
Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!



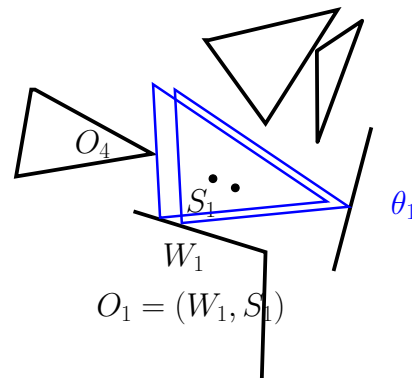
Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!



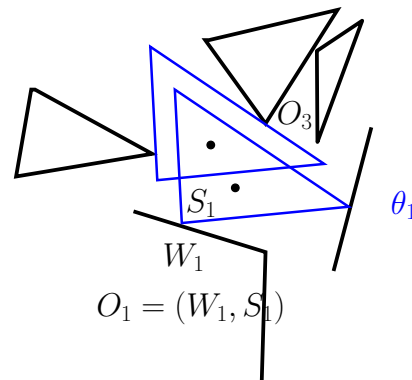
Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!



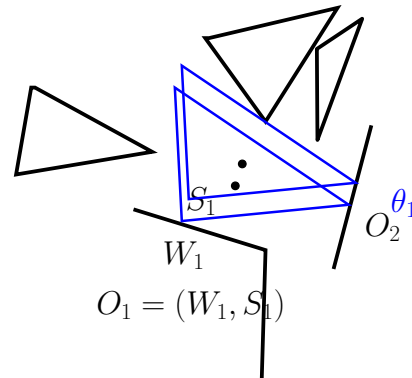
Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!



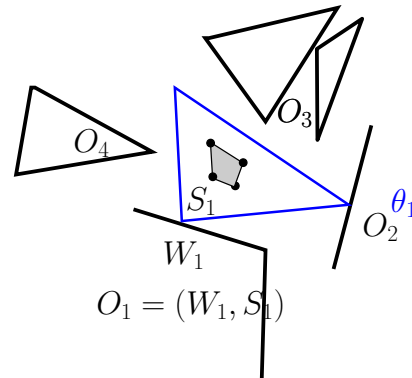
Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!



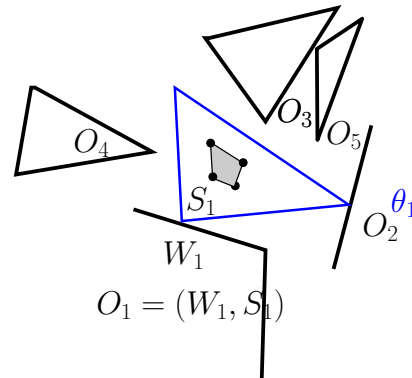
Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!



Idee des Filterns

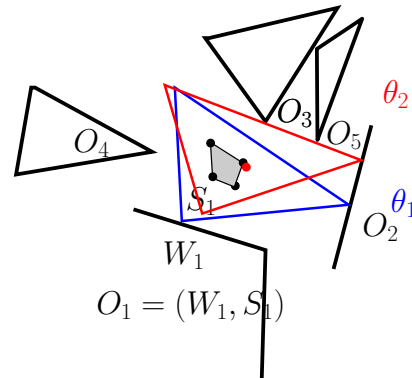
- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!



$$\begin{aligned}
 L(O_1)(\theta_1) &= \{O_2, O_4\} \\
 L(O_2)(\theta_1) &= \{O_1, O_3\} \\
 L(O_3)(\theta_1) &= \{O_2, O_4\} \\
 L(O_4)(\theta_1) &= \{O_1, O_3\} \\
 L(O_5)(\theta_1) &= \{\}
 \end{aligned}$$

Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!



$$L(O_1)(\theta_1) = \{O_2, O_4\}$$

$$L(O_2)(\theta_1) = \{O_1, O_3\}$$

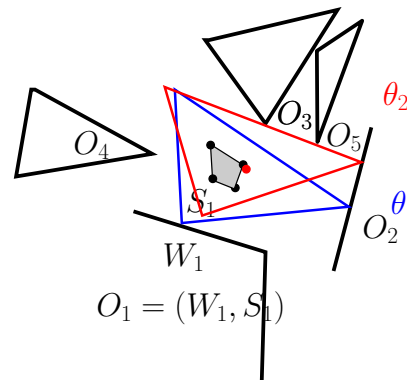
$$L(O_3)(\theta_1) = \{O_2, O_4\}$$

$$L(O_4)(\theta_1) = \{O_1, O_3\}$$

$$L(O_5)(\theta_1) = \{\}$$

Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!



$$L(O_1)(\theta_1) = \{O_2, O_4\}$$

$$L(O_2)(\theta_1) = \{O_1, O_3\}$$

$$L(O_3)(\theta_1) = \{O_2, O_4\}$$

$$L(O_4)(\theta_1) = \{O_1, O_3\}$$

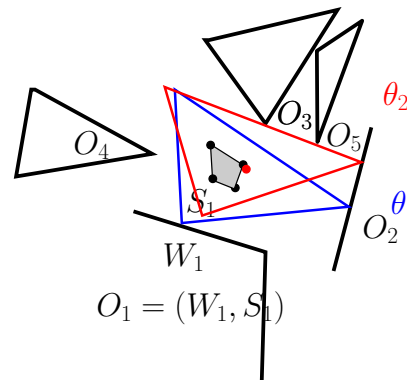
$$L(O_5)(\theta_1) = \{\}$$

(O_2, O_3, O_5)

Frei?

Idee des Filterns

- Krit. Platzierung θ_2 mit drei Kontakten (O_2, O_3, O_5): (Halb)Frei?
- **Bereits in Komponente**: Kurz vorher ($\theta_1 = \theta_2 - \epsilon$), O_2, O_3 und O_5 erzeugten mind. einen Knoten von $C_{frei}^{\theta_1}(V^{\theta_1})$, mutiert zu neuem Knoten, oder zwei Knoten verschmelzen
- Kontakt O : Listen L_O der Knotenerzeugung!
- Beispiel!!
- Test: Liste von L_{O_2} oder L_{O_3} oder L_{O_5} enthält andere



$$L(O_1)(\theta_1) = \{O_2, O_4\}$$

$$L(O_2)(\theta_1) = \{O_1, O_3\}$$

$$L(O_3)(\theta_1) = \{O_2, O_4\}$$

$$L(O_4)(\theta_1) = \{O_1, O_3\}$$

$$L(O_5)(\theta_1) = \{\}$$

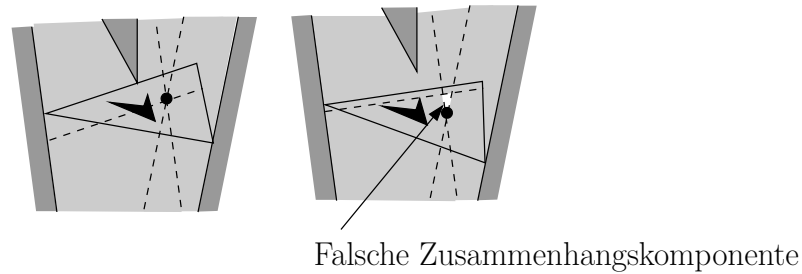
(O_2, O_3, O_5)

Frei?

Problem: Start Komponente

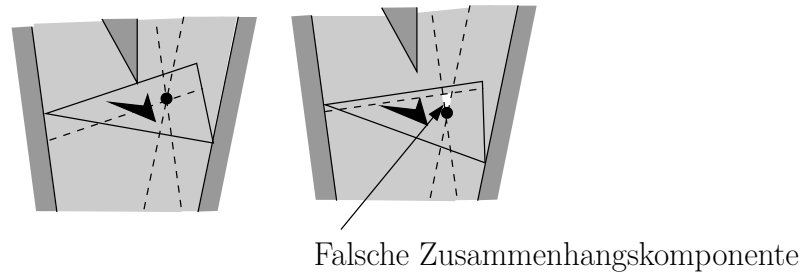
Problem: Start Komponente

- Beim Entstehen gibt es keine Listen und gar keinen Kontakt!



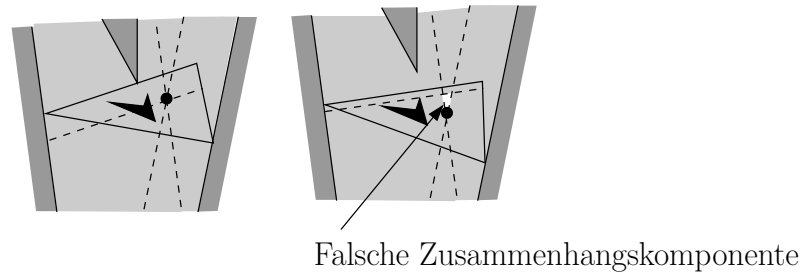
Problem: Start Komponente

- Beim Entstehen gibt es keine Listen und gar keinen Kontakt!
- Idee: Nur die Komponente, die Startpunkt s enthält: Z_s



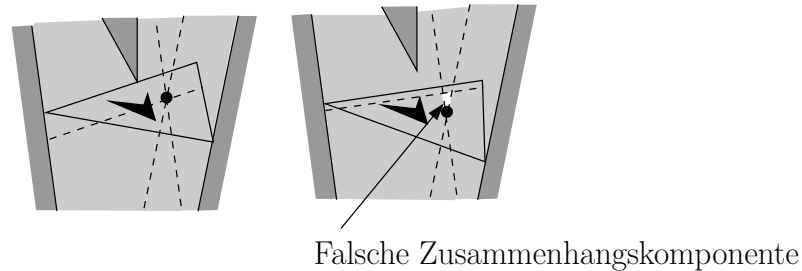
Problem: Start Komponente

- Beim Entstehen gibt es keine Listen und gar keinen Kontakt!
- Idee: Nur die Komponente, die Startpunkt s enthält: Z_s
- Zu Beginn: Zshg. Teil von C_{frei}^θ und V^θ



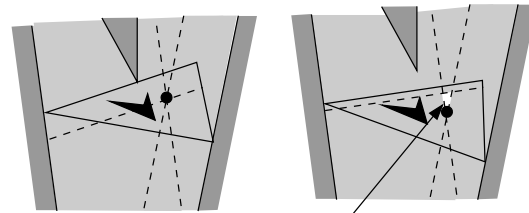
Problem: Start Komponente

- Beim Entstehen gibt es keine Listen und gar keinen Kontakt!
- Idee: Nur die Komponente, die Startpunkt s enthält: Z_s
- Zu Beginn: Zshg. Teil von C_{frei}^θ und V^θ
- Wann können andere korr. Komponenten dazu kommen?
Vereinigung!



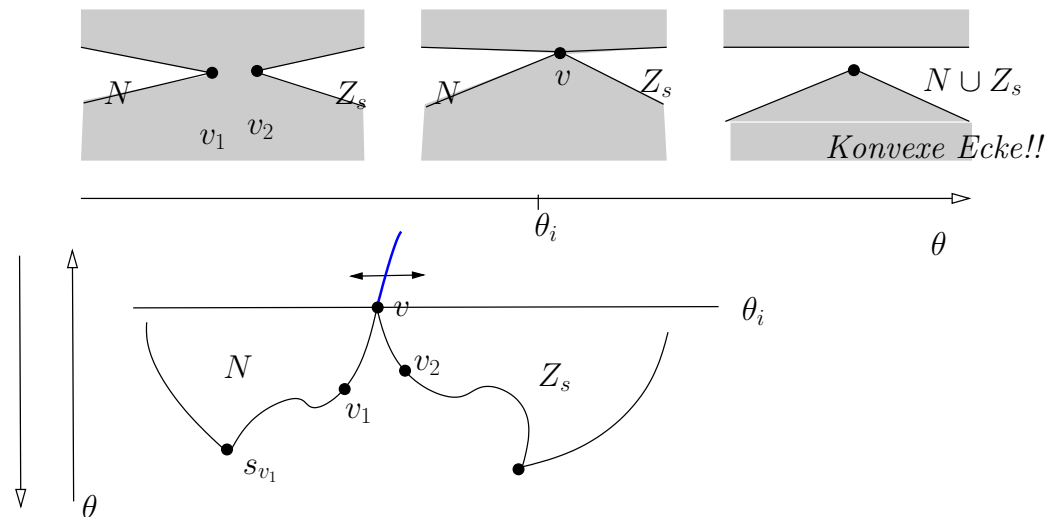
Problem: Start Komponente

- Beim Entstehen gibt es keine Listen und gar keinen Kontakt!
- Idee: Nur die Komponente, die Startpunkt s enthält: Z_s
- Zu Beginn: Zshg. Teil von C_{frei}^θ und V^θ
- Wann können andere korr. Komponenten dazu kommen?
Vereinigung!
- Krit. Orientierung Typ iv)!!



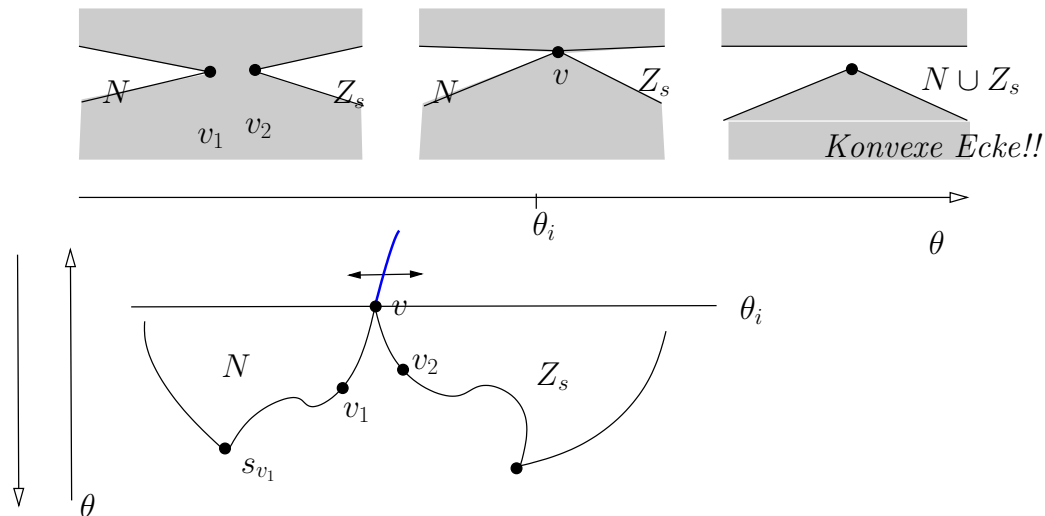
Falsche Zusammenhangskomponente

Vereinigung von Komponenten!



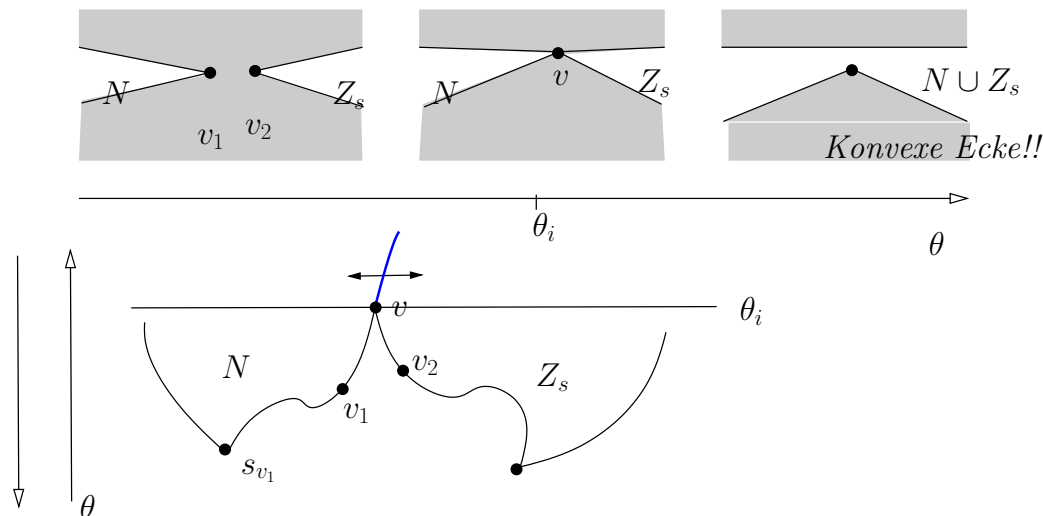
Vereinigung von Komponenten!

- Ecke/Ecke, Kante/Ecke: Konv. Ecke in C_{verb}^θ entsteht!



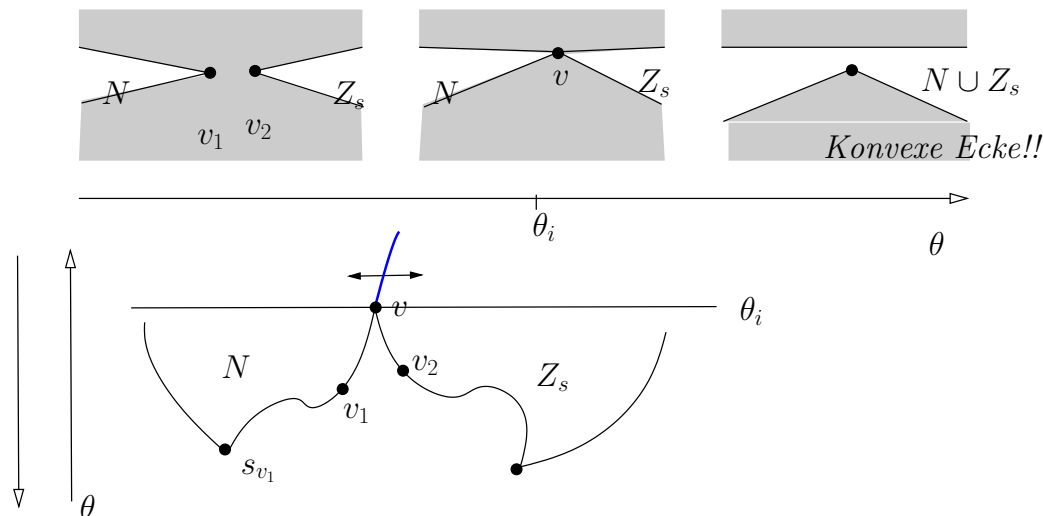
Vereinigung von Komponenten!

- Ecke/Ecke, Kante/Ecke: Konv. Ecke in C_{verb}^θ entsteht!
- Die freien Platz. dieser Art berechnen: Alg. 2.9



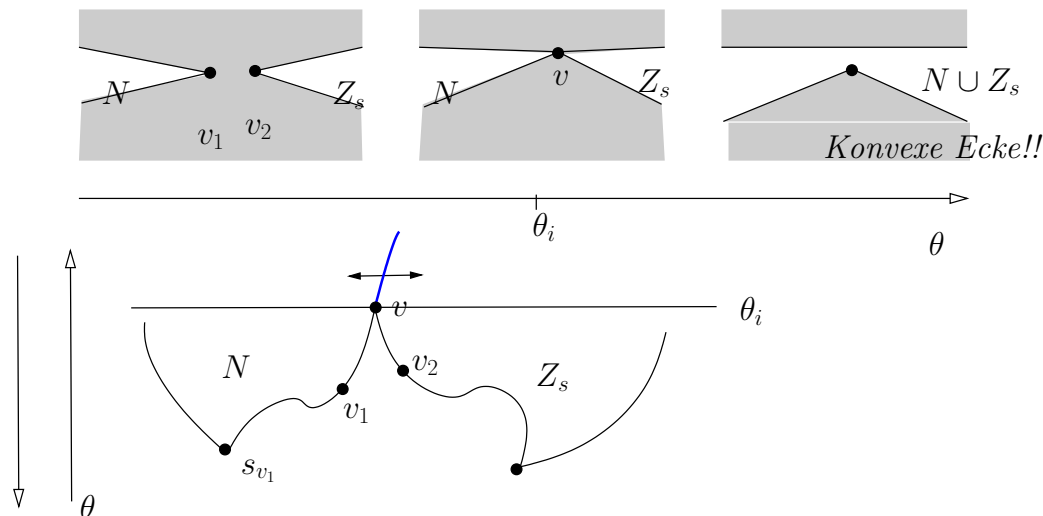
Vereinigung von Komponenten!

- Ecke/Ecke, Kante/Ecke: Konv. Ecke in C_{verb}^θ entsteht!
- Die freien Platz. dieser Art berechnen: Alg. 2.9
- $T' \subseteq T^*$: Freie Platzierung beginn. konvexer Ecke v



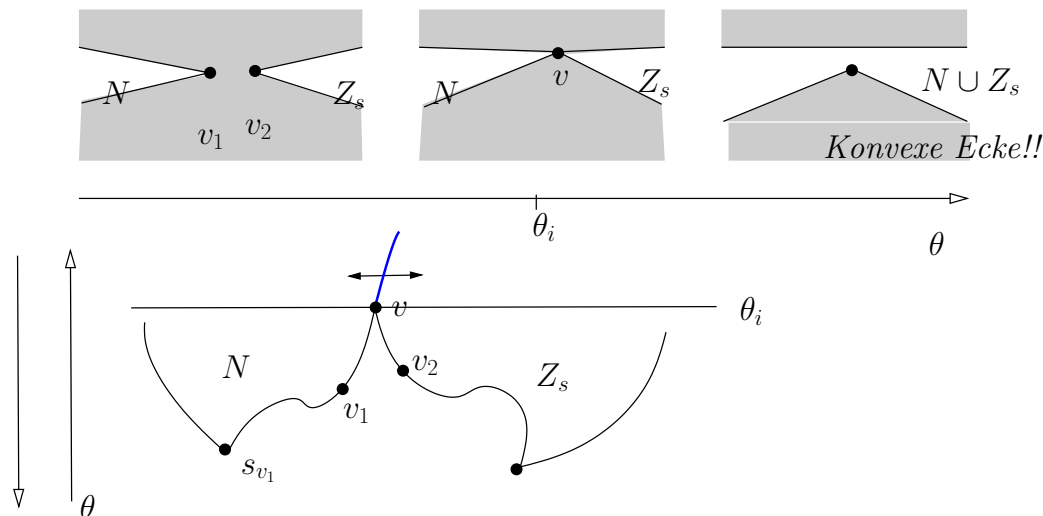
Vereinigung von Komponenten!

- Ecke/Ecke, Kante/Ecke: Konv. Ecke in C_{verb}^θ entsteht!
- Die freien Platz. dieser Art berechnen: Alg. 2.9
- $T' \subseteq T^*$: Freie Platzierung beginn. konvexer Ecke v
- Komponente, die irgendwann die konvexe Ecke enthält muss irgendwo entstanden sein

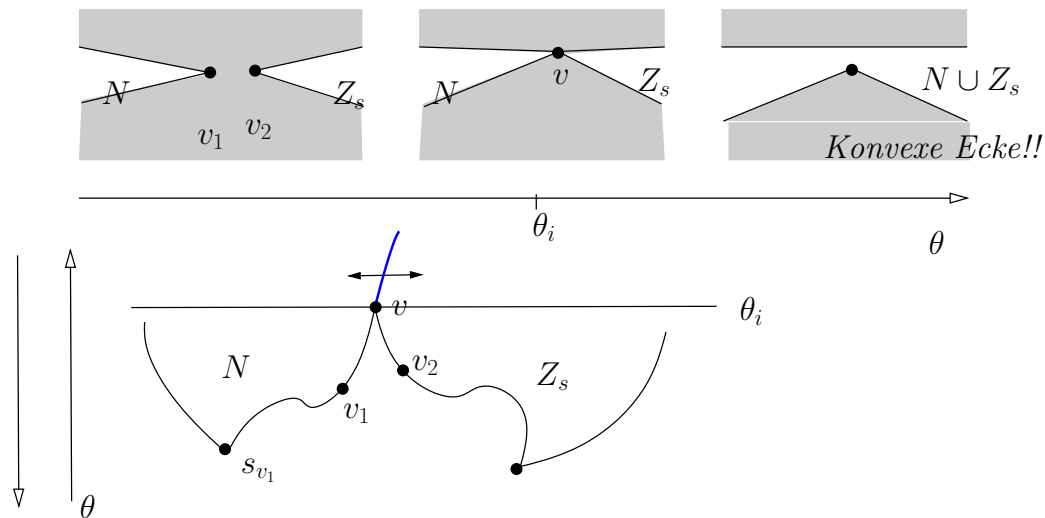


Vereinigung von Komponenten!

- Ecke/Ecke, Kante/Ecke: Konv. Ecke in C_{verb}^θ entsteht!
- Die freien Platz. dieser Art berechnen: Alg. 2.9
- $T' \subseteq T^*$: Freie Platzierung beginn. konvexer Ecke v
- Komponente, die irgendwann die konvexe Ecke enthält muss irgendwo entstanden sein
- Entstehungszeitpunkte s_v bestimmen: T''

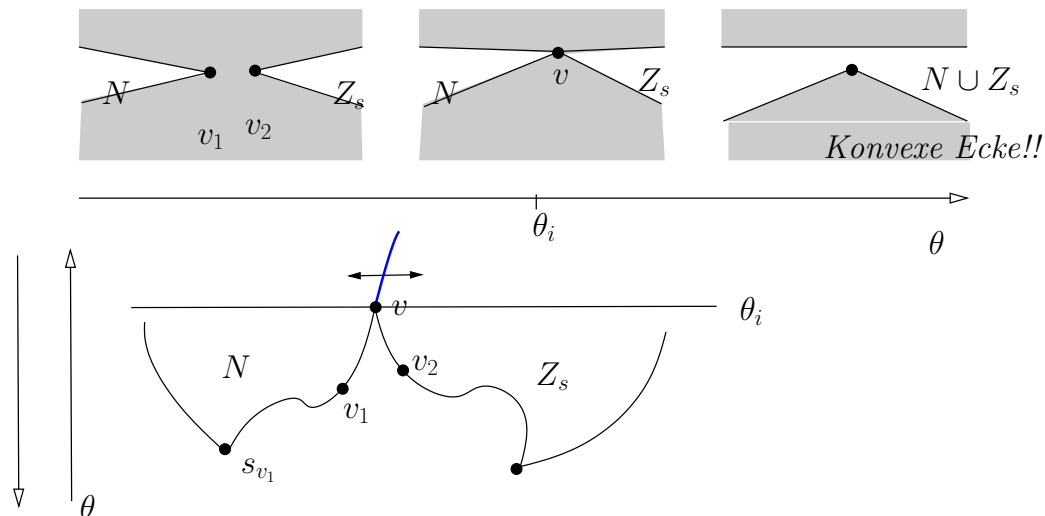


Nochmals Vorgehensweise!



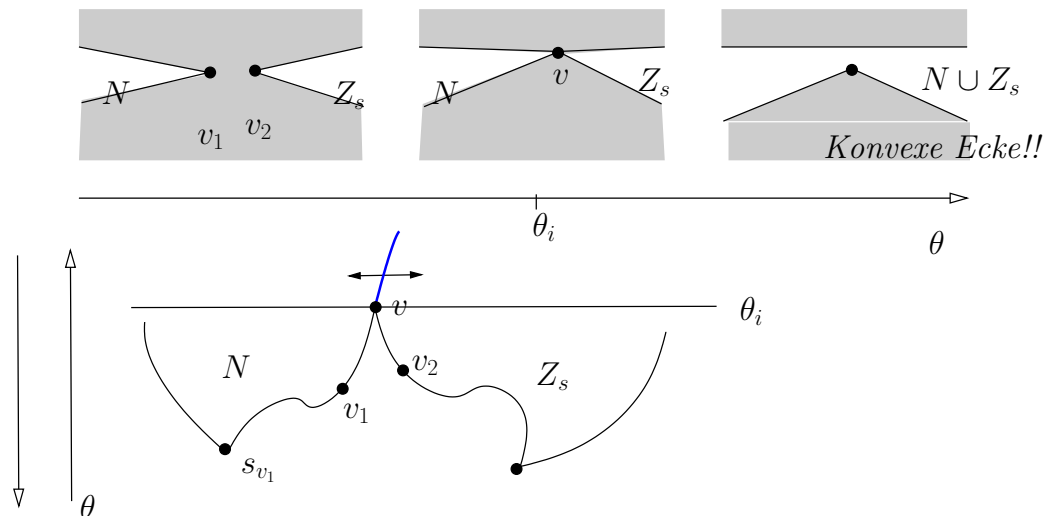
Nochmals Vorgehensweise!

- Zuerst alle Zeitpunkte, wo freie konv. v Ecke entsteht: T'



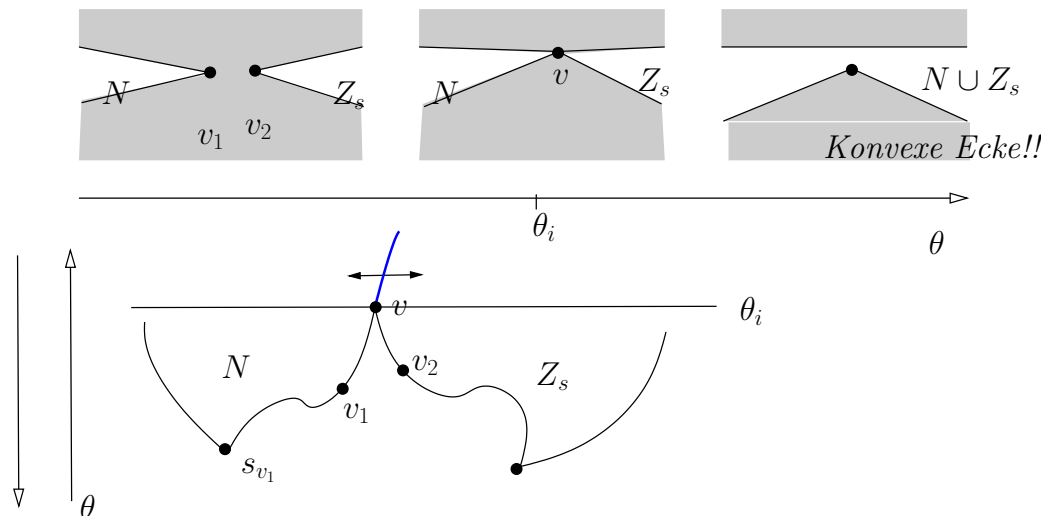
Nochmals Vorgehensweise!

- Zuerst alle Zeitpunkte, wo freie konv. v Ecke entsteht: T'
- Alg. 2.9!



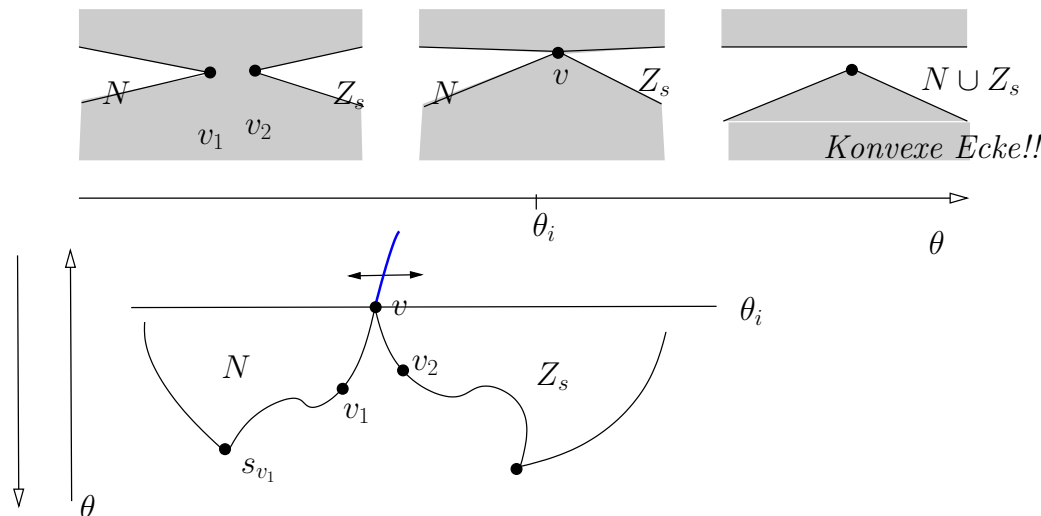
Nochmals Vorgehensweise!

- Zuerst alle Zeitpunkte, wo freie konv. v Ecke entsteht: T'
- Alg. 2.9!
- Rückwärtsdurchlauf: Wo entstanden die Komponenten für Element v aus T' ?



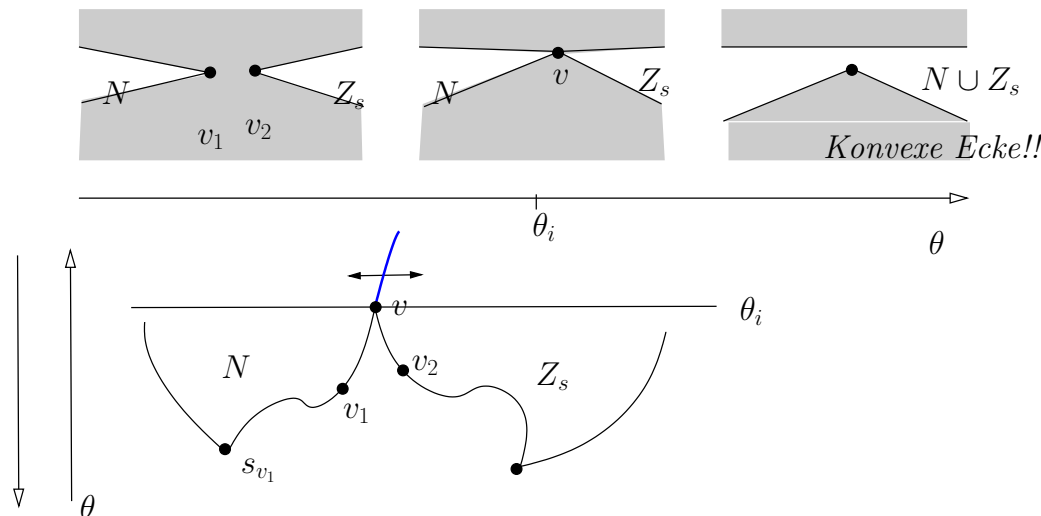
Nochmals Vorgehensweise!

- Zuerst alle Zeitpunkte, wo freie konv. v Ecke entsteht: T'
- Alg. 2.9!
- Rückwärtsdurchlauf: Wo entstanden die Komponenten für Element v aus T' ? Alg. 2.10



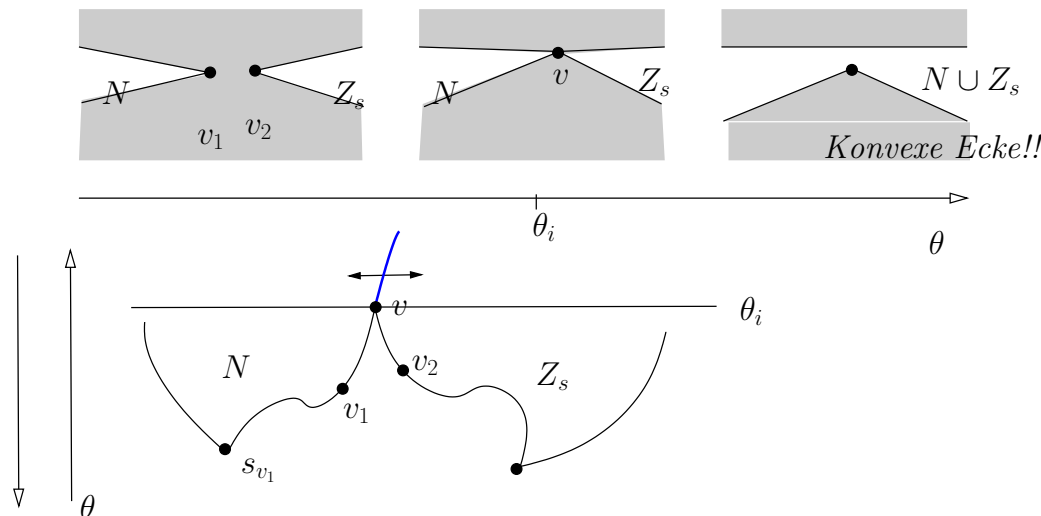
Nochmals Vorgehensweise!

- Zuerst alle Zeitpunkte, wo freie konv. v Ecke entsteht: T'
- Alg. 2.9!
- Rückwärtsdurchlauf: Wo entstanden die Komponenten für Element v aus T' ? Alg. 2.10
- Alle diese Entstehungszeitpunkte s_v : T''



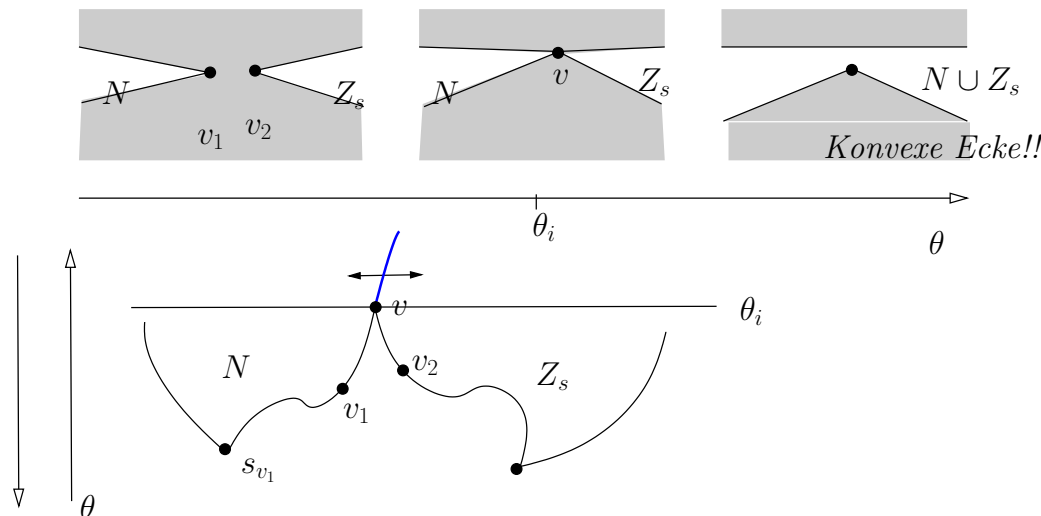
Nochmals Vorgehensweise!

- Zuerst alle Zeitpunkte, wo freie konv. v Ecke entsteht: T'
- Alg. 2.9!
- Rückwärtsdurchlauf: Wo entstanden die Komponenten für Element v aus T' ? Alg. 2.10
- Alle diese Entstehungszeitpunkte s_v : T''
- Vorwärtsdurchlauf: Komp. mit aufbauen,

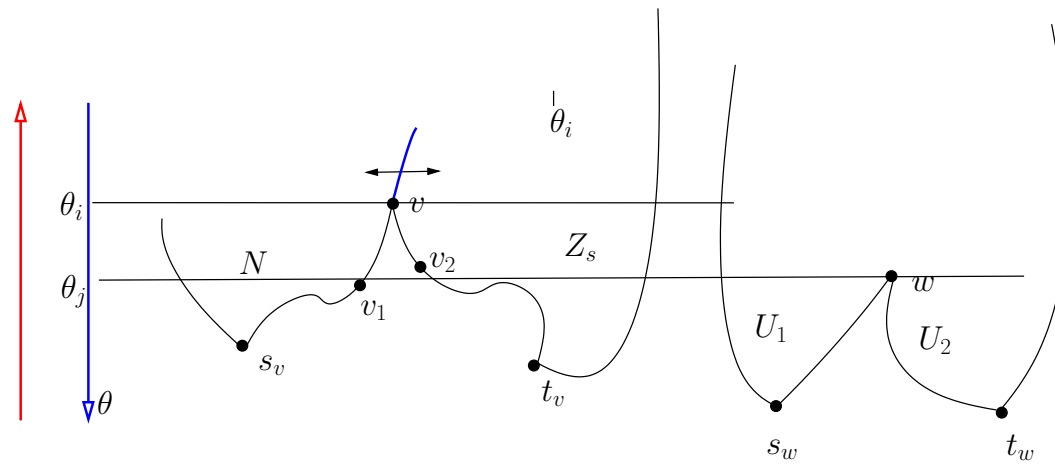


Nochmals Vorgehensweise!

- Zuerst alle Zeitpunkte, wo freie konv. v Ecke entsteht: T'
- Alg. 2.9!
- Rückwärtsdurchlauf: Wo entstanden die Komponenten für Element v aus T' ? Alg. 2.10
- Alle diese Entstehungszeitpunkte s_v : T''
- Vorwärtsdurchlauf: Komp. mit aufbauen, korrekt!

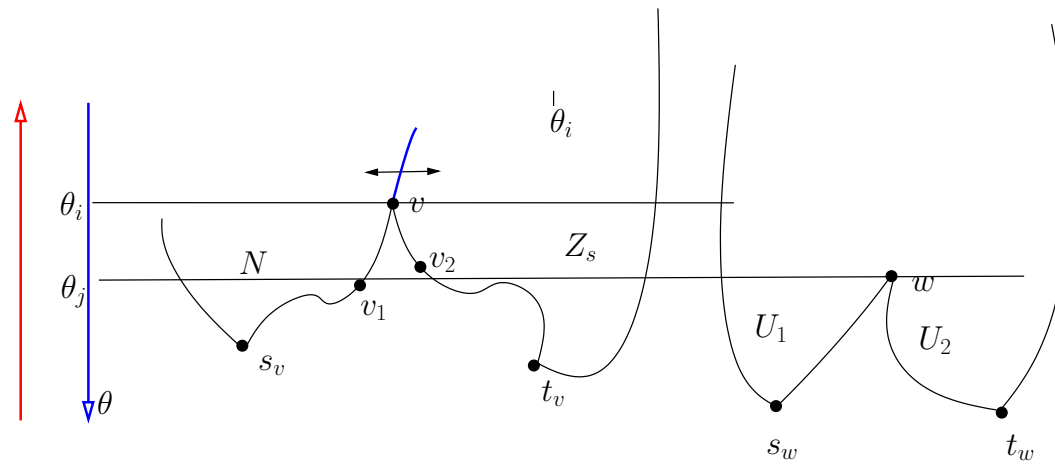


Was bedeutet das?!



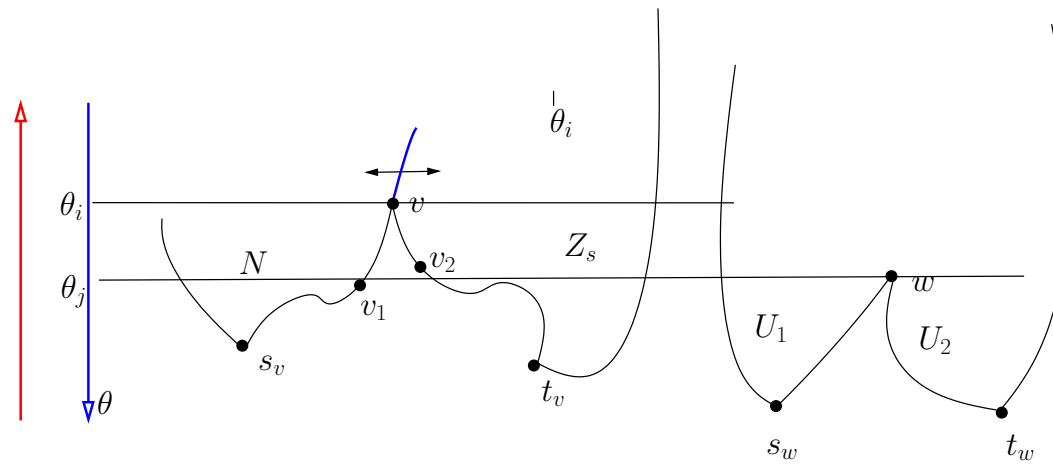
Was bedeutet das?!

- Neben korrekten Komp. von θ_0 bauen wir noch die Komponenten auf, die mit Elementen aus T'' starten



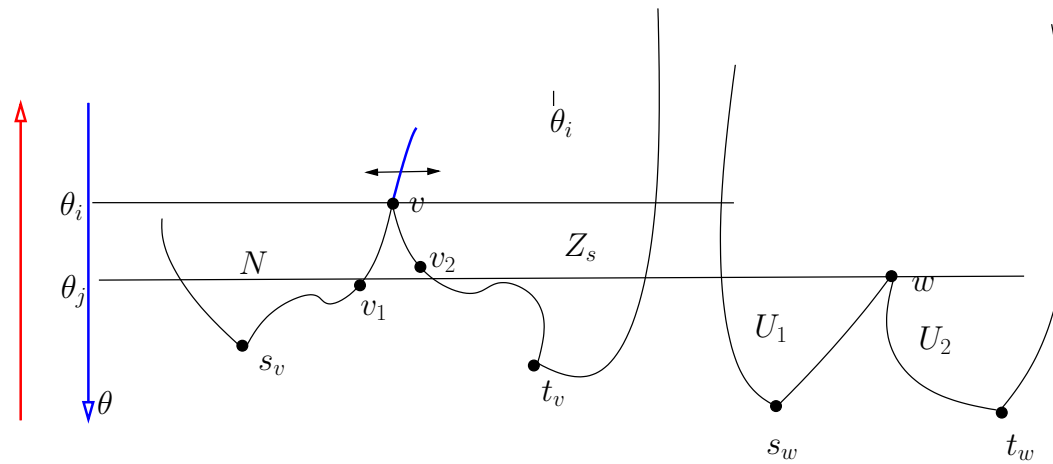
Was bedeutet das?!

- Neben korrekten Komp. von θ_0 bauen wir noch die Komponenten auf, die mit Elementen aus T'' starten
- Diese sind auch korrekt



Was bedeutet das?!

- Neben korrekten Komp. von θ_0 bauen wir noch die Komponenten auf, die mit Elementen aus T'' starten
- Diese sind auch korrekt
- Können evtl. mit Z_s zusammenfallen



Freie Konv. Ecken T' : Alg. 2.9

Freie Konv. Ecken T' : Alg. 2.9

Output: $T' \subseteq T^*$: krit. Orient., konv. Ecke von C_{verb}^θ ((iii)/(iv))

Freie Konv. Ecken T' : Alg. 2.9

Output: $T' \subseteq T^*$: krit. Orient., konv. Ecke von C_{verb}^θ ((iii)/(iv))

Intervalle : $O((mn)^2)$ verbotene Winkelintervalle (gleich!)

Freie Konv. Ecken T' : Alg. 2.9

Output: $T' \subseteq T^*$: krit. Orient., konv. Ecke von C_{verb}^θ ((iii)/(iv))

Intervalle : $O((mn)^2)$ verbotene Winkelintervalle (gleich!)

Ber. Vereinigung Intervalle durch Sort.: $O((mn)^2 \log(mn))$

Freie Konv. Ecken T' : Alg. 2.9

Output: $T' \subseteq T^*$: krit. Orient., konv. Ecke von C_{verb}^θ ((iii)/(iv))

Intervalle : $O((mn)^2)$ verbotene Winkelintervalle (gleich!)

Ber. Vereinigung Intervalle durch Sort.: $O((mn)^2 \log(mn))$

T' enthält die Endpunkte der vereinigten Intervalle.

Freie Konv. Ecken T' : Alg. 2.9

Output: $T' \subseteq T^*$: krit. Orient., konv. Ecke von C_{verb}^θ ((iii)/(iv))

Intervalle : $O((mn)^2)$ verbotene Winkelintervalle (gleich!)

Ber. Vereinigung Intervalle durch Sort.: $O((mn)^2 \log(mn))$

T' enthält die Endpunkte der vereinigten Intervalle.

Nicht jedesmal entsteht neuer Zusammenhang!!

Freie Konv. Ecken T' : Alg. 2.9

Output: $T' \subseteq T^*$: krit. Orient., konv. Ecke von C_{verb}^θ ((iii)/(iv))

Intervalle : $O((mn)^2)$ verbotene Winkelintervalle (gleich!)

Ber. Vereinigung Intervalle durch Sort.: $O((mn)^2 \log(mn))$

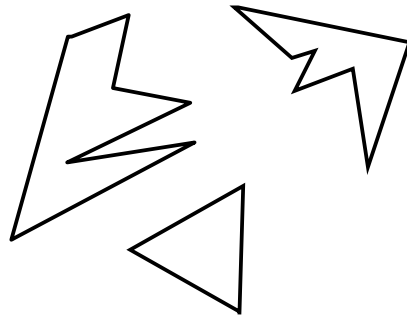
T' enthält die Endpunkte der vereinigten Intervalle.

Nicht jedesmal entsteht neuer Zusammenhang!!

Freie Konv. Ecken T' : Alg. 2.9

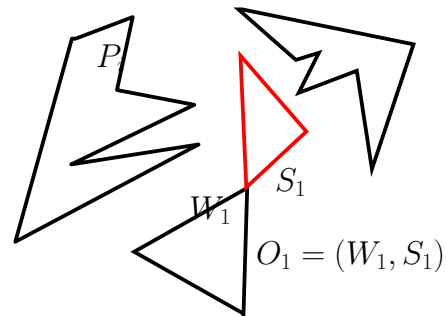
Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :



Freie Konv. Ecken T' : Alg. 2.9

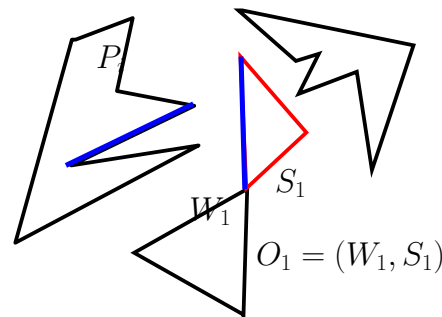
Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :



Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :

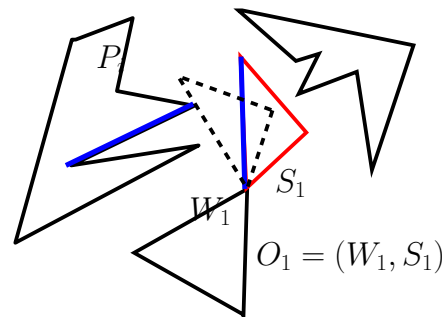
- Für jede Kante H_j eines Hindernisses P_i und jede Kante R_l von R . Ber. durch Rotation von R um v Intervall der Orient., bei denen R_l das Hindernis P_i schneidet.



Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :

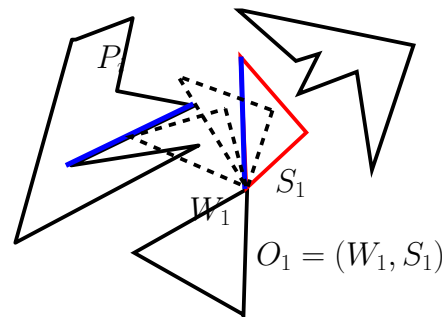
- Für jede Kante H_j eines Hindernisses P_i und jede Kante R_l von R . Ber. durch Rotation von R um v Intervall der Orient., bei denen R_l das Hindernis P_i schneidet.



Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :

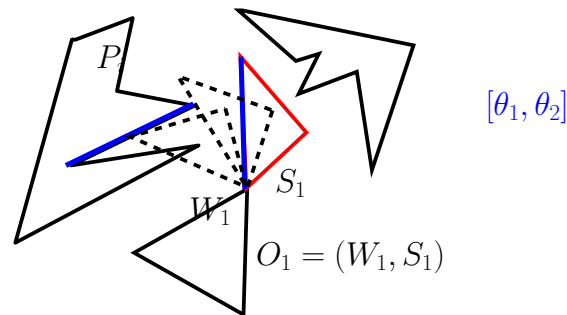
- Für jede Kante H_j eines Hindernisses P_i und jede Kante R_l von R . Ber. durch Rotation von R um v Intervall der Orient., bei denen R_l das Hindernis P_i schneidet.



Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :

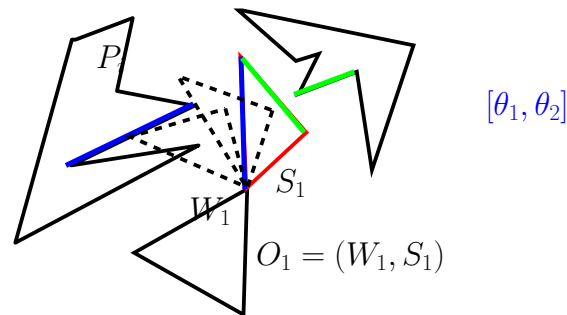
- Für jede Kante H_j eines Hindernisses P_i und jede Kante R_l von R . Ber. durch Rotation von R um v Intervall der Orient., bei denen R_l das Hindernis P_i schneidet.



Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :

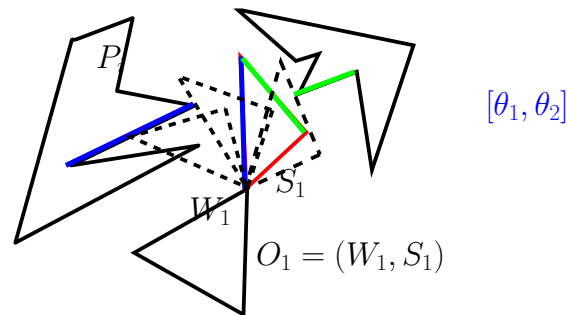
- Für jede Kante H_j eines Hindernisses P_i und jede Kante R_l von R . Ber. durch Rotation von R um v Intervall der Orient., bei denen R_l das Hindernis P_i schneidet.



Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :

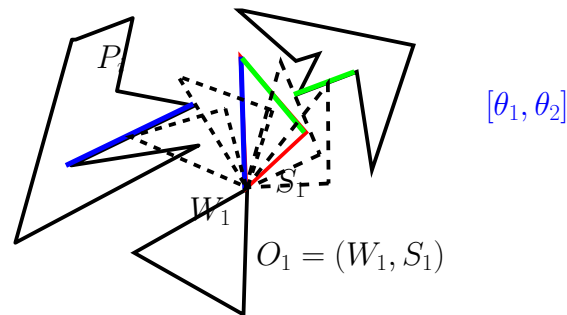
- Für jede Kante H_j eines Hindernisses P_i und jede Kante R_l von R . Ber. durch Rotation von R um v Intervall der Orient., bei denen R_l das Hindernis P_i schneidet.



Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :

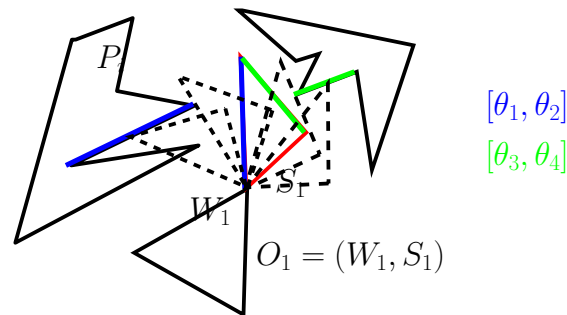
- Für jede Kante H_j eines Hindernisses P_i und jede Kante R_l von R . Ber. durch Rotation von R um v Intervall der Orient., bei denen R_l das Hindernis P_i schneidet.



Freie Konv. Ecken T' : Alg. 2.9

Für ein Ecke/Ecke Kontaktpaar, Kontaktecke v :

- Für jede Kante H_j eines Hindernisses P_i und jede Kante R_l von R . Ber. durch Rotation von R um v Intervall der Orient., bei denen R_l das Hindernis P_i schneidet.
- Insgesamt $O(mn)$ verbotene Winkelintervalle in Zeit $O(mn)$.



Freie Konv. Ecken T' : Alg. 2.9

Freie Konv. Ecken T' : Alg. 2.9

- $O(mn)$ Intervalle für ein Ecke/Ecke Paar

Freie Konv. Ecken T' : Alg. 2.9

- $O(mn)$ Intervalle für ein Ecke/Ecke Paar
- Vereinigung der Intervalle durch Sortieren in $O(mn \log mn)$

Freie Konv. Ecken T' : Alg. 2.9

- $O(mn)$ Intervalle für ein Ecke/Ecke Paar
- Vereinigung der Intervalle durch Sortieren in $O(mn \log mn)$
- Ränder ergeben Platzierungen

Freie Konv. Ecken T' : Alg. 2.9

- $O(mn)$ Intervalle für ein Ecke/Ecke Paar
- Vereinigung der Intervalle durch Sortieren in $O(mn \log mn)$
- Ränder ergeben Platzierungen
- $O((mn)^2 \log mn)$ für alle Ecke/Ecke Paare

Int. freie Orient. T'' : Alg. 2.10

Int. freie Orient. T'' : Alg. 2.10

- Idee:

Int. freie Orient. T'' : Alg. 2.10

- Idee: Verfolge die Orientierungen aus T' bis Komponente entsteht

Int. freie Orient. T'' : Alg. 2.10

- Idee: Verfolge die Orientierungen aus T' bis Komponente entsteht
- Output:

Int. freie Orient. T'' : Alg. 2.10

- Idee: Verfolge die Orientierungen aus T' bis Komponente entsteht
- Output: $T'' \subseteq T^*$: alle kritischen Orientierungen, lokal entsteht neue *interessante* Komponente

Int. freie Orient. T'' : Alg. 2.10

- Idee: Verfolge die Orientierungen aus T' bis Komponente entsteht
- Output: $T'' \subseteq T^*$: alle kritischen Orientierungen, lokal entsteht neue *interessante* Komponente
- Datenstruktur:

Int. freie Orient. T'' : Alg. 2.10

- Idee: Verfolge die Orientierungen aus T' bis Komponente entsteht
- Output: $T'' \subseteq T^*$: alle kritischen Orientierungen, lokal entsteht neue *interessante* Komponente
- Datenstruktur: Für jedes Kontaktpaar O unterhalte nach θ sortierte Liste $L_O(\theta)$: Kontakte O' die mit O Knoten in C_{frei}^θ erzeugen.

Int. freie Orient. T'' : Alg. 2.10

- Idee: Verfolge die Orientierungen aus T' bis Komponente entsteht
- Output: $T'' \subseteq T^*$: alle kritischen Orientierungen, lokal entsteht neue *interessante* Komponente
- Datenstruktur: Für jedes Kontaktpaar O unterhalte nach θ sortierte Liste $L_O(\theta)$: Kontakte O' die mit O Knoten in C_{frei}^θ erzeugen.
- Initialisierung:

Int. freie Orient. T'' : Alg. 2.10

- Idee: Verfolge die Orientierungen aus T' bis Komponente entsteht
- Output: $T'' \subseteq T^*$: alle kritischen Orientierungen, lokal entsteht neue *interessante* Komponente
- Datenstruktur: Für jedes Kontaktpaar O unterhalte nach θ sortierte Liste $L_O(\theta)$: Kontakte O' die mit O Knoten in C_{frei}^θ erzeugen.
- Initialisierung: Ber. $C_{\text{frei}}^{\theta_0}$ für Startorientierung θ_0 , initialisiere Listen $L_O(\theta_0)$ Kontaktpaaren O, O' , die in $C_{\text{frei}}^{\theta_0}$ Knoten erzeugen.

Alg. 2.10

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$:

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein:

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:
 - O, O', O'' die zugeh. Kontaktpaare:

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:
 - O, O', O'' die zugeh. Kontaktpaare: Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:
 - O, O', O'' die zugeh. Kontaktpaare: Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$
 - Suche erfolglos:

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:
 - O, O', O'' die zugeh. Kontaktpaare: Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$
 - Suche erfolglos: ignoriere θ_i .

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:
 - O, O', O'' die zugeh. Kontaktpaare: Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$
 - Suche erfolglos: ignoriere θ_i .
 - Falls z.B. O' in $L_O(\theta_i)$ (oder O in $L_{O'}(\theta_i)$) (θ_i ist frei)

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:
 - O, O', O'' die zugeh. Kontaktpaare: Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$
 - Suche erfolglos: ignoriere θ_i .
 - Falls z.B. O' in $L_O(\theta_i)$ (oder O in $L_{O'}(\theta_i)$) (θ_i ist frei) ber. lok. Änderungen in C_{frei}^θ , akt. Listen $L_O(\theta_i), L_{O'}(\theta_i), L_{O''}(\theta_i)$

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:
 - O, O', O'' die zugeh. Kontaktpaare: Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$
 - Suche erfolglos: ignoriere θ_i .
 - Falls z.B. O' in $L_O(\theta_i)$ (oder O in $L_{O'}(\theta_i)$) (θ_i ist frei) ber. lok. Änderungen in C_{frei}^θ , akt. Listen $L_O(\theta_i), L_{O'}(\theta_i), L_{O''}(\theta_i)$
 - Falls bei θ_i Komp. verschw.:

Alg. 2.10

Bearbeite nach **fallendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Falls $\theta_i \in T'$: O_1, O'_1 die Kontaktp. von v_1 und O_2, O'_2 von v_2 .
Füge ein: O'_1 in $L_{O_1}(\theta_i)$, O_1 in $L_{O'_1}(\theta_i)$, O'_2 in $L_{O_2}(\theta_i)$, O_2 in $L_{O'_2}(\theta_i)$
- Falls $\theta_i \notin T'$:
 - O, O', O'' die zugeh. Kontaktpaare: Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$
 - Suche erfolglos: ignoriere θ_i .
 - Falls z.B. O' in $L_O(\theta_i)$ (oder O in $L_{O'}(\theta_i)$) (θ_i ist frei) ber. lok. Änderungen in C_{frei}^θ , akt. Listen $L_O(\theta_i), L_{O'}(\theta_i), L_{O''}(\theta_i)$
 - Falls bei θ_i Komp. verschw.: Füge θ_i in T'' ein.

Was bisher geschah!!

Was bisher geschah!!

Kritische Orientierungen T^*

$$O(mn \lambda_6(mn) \log(mn))$$

Was bisher geschah!!

Kritische Orientierungen T^* $O(mn \lambda_6(mn) \log(mn))$
Orientierungen T' (konv. Eck. von C_{verb}^θ) $O(m^2 n^2 \log(mn))$

Was bisher geschah!!

Kritische Orientierungen T^*	$O(mn \lambda_6(mn) \log(mn))$
Orientierungen T' (konv. Eck. von C_{verb}^θ)	$O(m^2 n^2 \log(mn))$
Orientierungen T'' (wicht. Komp.)	$O(mn \lambda_6(mn) \log(mn))$

Was bisher geschah!!

Kritische Orientierungen T^*	$O(mn \lambda_6(mn) \log(mn))$
Orientierungen T' (konv. Eck. von C_{verb}^θ)	$O(m^2 n^2 \log(mn))$
Orientierungen T'' (wicht. Komp.)	$O(mn \lambda_6(mn) \log(mn))$
T aus T^*, T' und T''	$O(mn \lambda_6(mn) \log(mn))$

Was bisher geschah!!

Kritische Orientierungen T^*	$O(mn \lambda_6(mn) \log(mn))$
Orientierungen T' (konv. Eck. von C_{verb}^θ)	$O(m^2 n^2 \log(mn))$
Orientierungen T'' (wicht. Komp.)	$O(mn \lambda_6(mn) \log(mn))$
T aus T^*, T' und T''	$O(mn \lambda_6(mn) \log(mn))$
T enthält alles was wir brauchen!	

Was bisher geschah!!

Kritische Orientierungen T^*	$O(mn \lambda_6(mn) \log(mn))$
Orientierungen T' (konv. Eck. von C_{verb}^θ)	$O(m^2 n^2 \log(mn))$
Orientierungen T'' (wicht. Komp.)	$O(mn \lambda_6(mn) \log(mn))$
T aus T^*, T' und T''	$O(mn \lambda_6(mn) \log(mn))$
T enthält alles was wir brauchen!	
Alle sortiert!!	

Was bisher geschah!!

Kritische Orientierungen T^*	$O(mn \lambda_6(mn) \log(mn))$
Orientierungen T' (konv. Eck. von C_{verb}^θ)	$O(m^2 n^2 \log(mn))$
Orientierungen T'' (wicht. Komp.)	$O(mn \lambda_6(mn) \log(mn))$
T aus T^*, T' und T''	$O(mn \lambda_6(mn) \log(mn))$
T enthält alles was wir brauchen!	
Alle sortiert!!	

Danach: Mit T den notwendigen Kantengraph aufbauen!!

Freie Orient. T Alg. 2.11

Freie Orient. T Alg. 2.11

- Idee:

Freie Orient. T Alg. 2.11

- Idee: Aus T^* die gültigen Orient. mit T' und T'' und Listen

Freie Orient. T Alg. 2.11

- Idee: Aus T^* die gültigen Orient. mit T' und T'' und Listen
- Output:

Freie Orient. T Alg. 2.11

- Idee: Aus T^* die gültigen Orient. mit T' und T'' und Listen
- Output: T : Menge aller gültigen kritischen Orientierungen

Freie Orient. T Alg. 2.11

- Idee: Aus T^* die gültigen Orient. mit T' und T'' und Listen
- Output: T : Menge aller gültigen kritischen Orientierungen
- Datenstruktur:

Freie Orient. T Alg. 2.11

- Idee: Aus T^* die gültigen Orient. mit T' und T'' und Listen
- Output: T : Menge aller gültigen kritischen Orientierungen
- Datenstruktur: Für jedes Kontaktpaar O unterhalte nach θ sort. Liste $L_O(\theta)$: Kontakte O' die zusammen mit O Knoten in C_{frei}^θ erzeugen.

Freie Orient. T Alg. 2.11

- Idee: Aus T^* die gültigen Orient. mit T' und T'' und Listen
- Output: T : Menge aller gültigen kritischen Orientierungen
- Datenstruktur: Für jedes Kontaktpaar O unterhalte nach θ sort. Liste $L_O(\theta)$: Kontakte O' die zusammen mit O Knoten in C_{frei}^θ erzeugen.
- Initialisierung:

Freie Orient. T Alg. 2.11

- Idee: Aus T^* die gültigen Orient. mit T' und T'' und Listen
- Output: T : Menge aller gültigen kritischen Orientierungen
- Datenstruktur: Für jedes Kontaktpaar O unterhalte nach θ sort. Liste $L_O(\theta)$: Kontakte O' die zusammen mit O Knoten in C_{frei}^θ erzeugen.
- Initialisierung: Ber. $C_{\text{frei}}^{\theta_0}$ für Startorientierung θ_0 , init. Listen $L_O(\theta_0)$ mit Kontaktpaaren O, O' , die in $C_{\text{frei}}^{\theta_0}$ Knoten erzeugen.

Freie Orient. T Alg. 2.11

- Idee: Aus T^* die gültigen Orient. mit T' und T'' und Listen
- Output: T : Menge aller gültigen kritischen Orientierungen
- Datenstruktur: Für jedes Kontaktpaar O unterhalte nach θ sort. Liste $L_O(\theta)$: Kontakte O' die zusammen mit O Knoten in C_{frei}^θ erzeugen.
- Initialisierung: Ber. $C_{\text{frei}}^{\theta_0}$ für Startorientierung θ_0 , init. Listen $L_O(\theta_0)$ mit Kontaktpaaren O, O' , die in $C_{\text{frei}}^{\theta_0}$ Knoten erzeugen.
- Listen werden für drei Kontakte verwendet!!

Freie Orient. T Alg. 2.11

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:
 - Falls $\theta_i \notin T''$: ignoriere θ_i .

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:
 - Falls $\theta_i \notin T''$: ignoriere θ_i .
 - Ansonsten aktualisiere die Listen $L_O(\theta_i)$.

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:
 - Falls $\theta_i \notin T''$: ignoriere θ_i .
 - Ansonsten aktualisiere die Listen $L_O(\theta_i)$.
- Falls nein:

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:
 - Falls $\theta_i \notin T''$: ignoriere θ_i .
 - Ansonsten aktualisiere die Listen $L_O(\theta_i)$.
- Falls nein:
 - O, O', O'' die zugeh. Kontaktpaare.

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:
 - Falls $\theta_i \notin T''$: ignoriere θ_i .
 - Ansonsten aktualisiere die Listen $L_O(\theta_i)$.
- Falls nein:
 - O, O', O'' die zugeh. Kontaktpaare. Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$.

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:
 - Falls $\theta_i \notin T''$: ignoriere θ_i .
 - Ansonsten aktualisiere die Listen $L_O(\theta_i)$.
- Falls nein:
 - O, O', O'' die zugeh. Kontaktpaare. Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$.
 - Falls z.B. O' in $L_O(\theta_i)$ (oder O in $L_{O'}(\theta_i)$) (θ_i ist frei):

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:
 - Falls $\theta_i \notin T''$: ignoriere θ_i .
 - Ansonsten aktualisiere die Listen $L_O(\theta_i)$.
- Falls nein:
 - O, O', O'' die zugeh. Kontaktpaare. Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$.
 - Falls z.B. O' in $L_O(\theta_i)$ (oder O in $L_{O'}(\theta_i)$) (θ_i ist frei): ber. lok. Änderungen in C_{frei}^θ ,

Freie Orient. T Alg. 2.11

Bearbeite nach **steigendem Winkel** alle krit. Orient. $\theta_i \in T^*$:

- Teste, ob bei θ_i evtl. neue Komponente entsteht.
- Falls ja:
 - Falls $\theta_i \notin T''$: ignoriere θ_i .
 - Ansonsten aktualisiere die Listen $L_O(\theta_i)$.
- Falls nein:
 - O, O', O'' die zugeh. Kontaktpaare. Suche nach O oder O' in $L_{O''}(\theta_i)$, nach O oder O'' in $L_{O'}(\theta_i)$, nach O' oder O'' in $L_O(\theta_i)$.
 - Falls z.B. O' in $L_O(\theta_i)$ (oder O in $L_{O'}(\theta_i)$) (θ_i ist frei): ber. lok. Änderungen in C_{frei}^θ , akt. Listen $L_O(\theta_i), L_{O'}(\theta_i), L_{O''}(\theta_i)$.

- Falls Suche erfolglos (z.B. nur zwei Kontaktpaare),

- Falls Suche erfolglos (z.B. nur zwei Kontaktpaare), teste $\theta_i \in T'$.

- Falls Suche erfolglos (z.B. nur zwei Kontaktpaare), teste $\theta_i \in T'$. Falls ja:

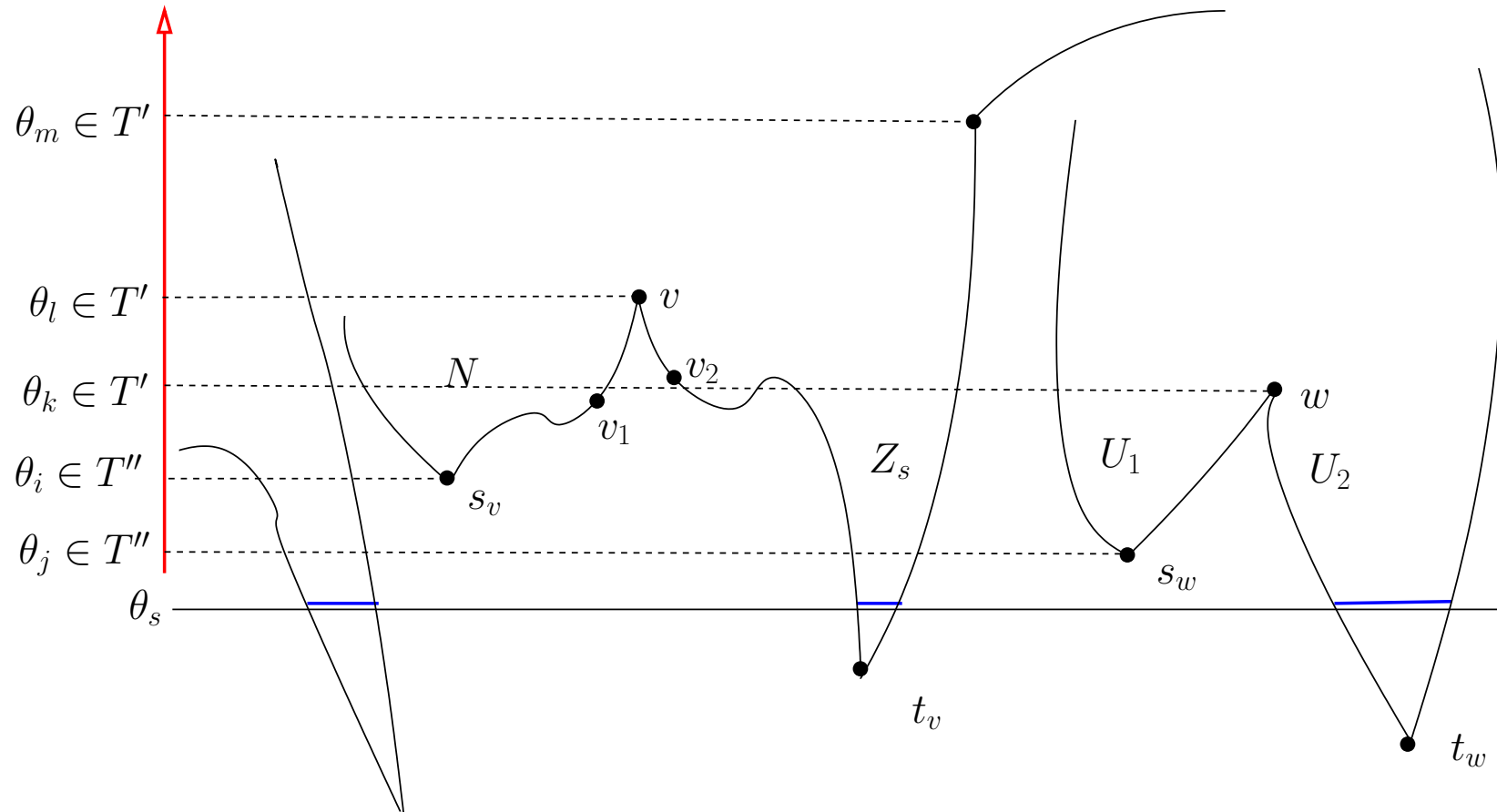
- Falls Suche erfolglos (z.B. nur zwei Kontaktpaare), teste $\theta_i \in T'$. Falls ja: Es entsteht konvexe Ecke von C_{verb}^θ , aber keine neue Komponente, θ_i ist gültig.

- Falls Suche erfolglos (z.B. nur zwei Kontaktpaare), teste $\theta_i \in T'$. Falls ja: Es entsteht konvexe Ecke von C_{verb}^θ , aber keine neue Komponente, θ_i ist gültig. Falls nein:

- Falls Suche erfolglos (z.B. nur zwei Kontaktpaare), teste $\theta_i \in T'$. Falls ja: Es entsteht konvexe Ecke von C_{verb}^θ , aber keine neue Komponente, θ_i ist gültig. Falls nein: ignoriere θ_i .

- Falls Suche erfolglos (z.B. nur zwei Kontaktpaare), teste $\theta_i \in T'$. Falls ja: Es entsteht konvexe Ecke von C_{verb}^θ , aber keine neue Komponente, θ_i ist gültig. Falls nein: ignoriere θ_i .
- T besteht aus allen nicht ignorierten Orientierungen aus T^* .

Veranschaulichung!!



Insgesamt!!

Kritische Orientierungen T^*	$O(mn \lambda_6(mn) \log(mn))$
Orientierungen T' (konv. Eck. von C_{verb}^θ)	$O(m^2 n^2 \log(mn))$
Orientierungen T'' (wicht. Komp.)	$O(mn \lambda_6(mn) \log(mn))$
T aus T^*, T' und T''	$O(mn \lambda_6(mn) \log(mn))$

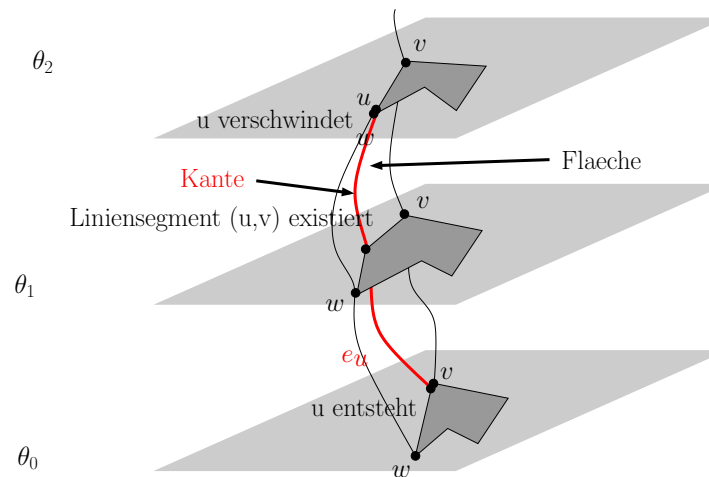
T enthält alles was wir brauchen!

Damit läßt sich sukzessive V^θ bestimmen, sukzessive der Kantengraph aufbauen

Zeit $O(mn \lambda_6(mn) \log(mn))$

Aufbau des Kantengraphen: Alg. 2.12

- erweiterte Knoten $(u, L(u))$: Kante in C_{frei}
- erweiterter Knoten $(v, L(v))$: Kante in C_{frei}
- erw. Kante $e = ((u, L(u)), (v, L(v)))$ mit $L(e)$: Fläche in C_{frei}



Aufbau des Kantengraphen: Alg. 2.12

Aufbau des Kantengraphen: Alg. 2.12

- Berechne Menge T aller kritischen Orientierungen, sortiert nach aufsteigendem θ !

Aufbau des Kantengraphen: Alg. 2.12

- Berechne Menge T aller kritischen Orientierungen, sortiert nach aufsteigendem θ ! Zeit: $O(nm\lambda_6(mn) \log mn)$,

Aufbau des Kantengraphen: Alg. 2.12

- Berechne Menge T aller kritischen Orientierungen, sortiert nach aufsteigendem θ ! Zeit: $O(nm\lambda_6(mn) \log mn)$,
 $|T| = O(nm\lambda_6(mn))$

Aufbau des Kantengraphen: Alg. 2.12

- Berechne Menge T aller kritischen Orientierungen, sortiert nach aufsteigendem θ ! Zeit: $O(nm\lambda_6(mn) \log mn)$,
 $|T| = O(nm\lambda_6(mn))$
- Wähle $\theta_0 \notin T$ und berechne V^{θ_0} .

Aufbau des Kantengraphen: Alg. 2.12

- Berechne Menge T aller kritischen Orientierungen, sortiert nach aufsteigendem θ ! Zeit: $O(nm\lambda_6(mn) \log mn)$,
 $|T| = O(nm\lambda_6(mn))$
- Wähle $\theta_0 \notin T$ und berechne V^{θ_0} .
- Initialisiere E mit den Kanten und Knoten aus V^{θ_0} , dabei bleiben die Lebensdauerintervalle $L(u)$ zunächst undefiniert.

Aufbau des Kantengraphen: Alg. 2.12

- Berechne Menge T aller kritischen Orientierungen, sortiert nach aufsteigendem θ ! Zeit: $O(nm\lambda_6(mn) \log mn)$,
 $|T| = O(nm\lambda_6(mn))$
- Wähle $\theta_0 \notin T$ und berechne V^{θ_0} .
- Initialisiere E mit den Kanten und Knoten aus V^{θ_0} , dabei bleiben die Lebensdauerintervalle $L(u)$ zunächst undefiniert. (Wir sind mitten in einem Intervall!!)

Aufbau des Kantengraphen: Alg. 2.12

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive:

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :
 - Bestimme die Änderungen in V^θ : Übergang von $\theta = \theta_i - \varepsilon$ nach $\theta = \theta_i + \varepsilon$

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :
 - Bestimme die Änderungen in V^θ : Übergang von $\theta = \theta_i - \varepsilon$ nach $\theta = \theta_i + \varepsilon$
 - Entferne alle verschwindenden Knoten und Kanten $v_1, \dots, v_r, e_1, \dots, e_s$ aus V^θ ,

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :
 - Bestimme die Änderungen in V^θ : Übergang von $\theta = \theta_i - \varepsilon$ nach $\theta = \theta_i + \varepsilon$
 - Entferne alle verschwindenden Knoten und Kanten $v_1, \dots, v_r, e_1, \dots, e_s$ aus V^θ , setze θ_i als obere Grenze der Lebensdauer von $L(v_j)$ bzw. $L(e_k)$ ein

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :
 - Bestimme die Änderungen in V^θ : Übergang von $\theta = \theta_i - \varepsilon$ nach $\theta = \theta_i + \varepsilon$
 - Entferne alle verschwindenden Knoten und Kanten $v_1, \dots, v_r, e_1, \dots, e_s$ aus V^θ , setze θ_i als obere Grenze der Lebensdauer von $L(v_j)$ bzw. $L(e_k)$ ein
 - Füge alle neu entstandenen Knoten und Kanten $v_1^*, \dots, v_{r^*}, e_1^*, \dots, e_{s^*}$ in V^θ und E ein,

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :
 - Bestimme die Änderungen in V^θ : Übergang von $\theta = \theta_i - \varepsilon$ nach $\theta = \theta_i + \varepsilon$
 - Entferne alle verschwindenden Knoten und Kanten $v_1, \dots, v_r, e_1, \dots, e_s$ aus V^θ , setze θ_i als obere Grenze der Lebensdauer von $L(v_j)$ bzw. $L(e_k)$ ein
 - Füge alle neu entstandenen Knoten und Kanten $v_1^*, \dots, v_r^*, e_1^*, \dots, e_s^*$ in V^θ und E ein, setze θ_i als untere Grenze der Lebensdauer von $L(v_j^*)$ bzw. $L(e_k^*)$ ein

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :
 - Bestimme die Änderungen in V^θ : Übergang von $\theta = \theta_i - \varepsilon$ nach $\theta = \theta_i + \varepsilon$
 - Entferne alle verschwindenden Knoten und Kanten $v_1, \dots, v_r, e_1, \dots, e_s$ aus V^θ , setze θ_i als obere Grenze der Lebensdauer von $L(v_j)$ bzw. $L(e_k)$ ein
 - Füge alle neu entstandenen Knoten und Kanten $v_1^*, \dots, v_r^*, e_1^*, \dots, e_s^*$ in V^θ und E ein, setze θ_i als untere Grenze der Lebensdauer von $L(v_j^*)$ bzw. $L(e_k^*)$ ein
- Nach 2π Durchlauf bei θ_0 :

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :
 - Bestimme die Änderungen in V^θ : Übergang von $\theta = \theta_i - \varepsilon$ nach $\theta = \theta_i + \varepsilon$
 - Entferne alle verschwindenden Knoten und Kanten $v_1, \dots, v_r, e_1, \dots, e_s$ aus V^θ , setze θ_i als obere Grenze der Lebensdauer von $L(v_j)$ bzw. $L(e_k)$ ein
 - Füge alle neu entstandenen Knoten und Kanten $v_1^*, \dots, v_r^*, e_1^*, \dots, e_s^*$ in V^θ und E ein, setze θ_i als untere Grenze der Lebensdauer von $L(v_j^*)$ bzw. $L(e_k^*)$ ein
- Nach 2π Durchlauf bei θ_0 : Trage noch fehlenden Lebensdauern nach,

Aufbau des Kantengraphen: Alg. 2.12

- Sukzessive: Für Orientierung θ_i in T :
 - Bestimme die Änderungen in V^θ : Übergang von $\theta = \theta_i - \varepsilon$ nach $\theta = \theta_i + \varepsilon$
 - Entferne alle verschwindenden Knoten und Kanten $v_1, \dots, v_r, e_1, \dots, e_s$ aus V^θ , setze θ_i als obere Grenze der Lebensdauer von $L(v_j)$ bzw. $L(e_k)$ ein
 - Füge alle neu entstandenen Knoten und Kanten $v_1^*, \dots, v_r^*, e_1^*, \dots, e_s^*$ in V^θ und E ein, setze θ_i als untere Grenze der Lebensdauer von $L(v_j^*)$ bzw. $L(e_k^*)$ ein
- Nach 2π Durchlauf bei θ_0 : Trage noch fehlenden Lebensdauern nach, genauer: fasse zwei Knoten der Art $(u, (\theta_1, \text{undef.}))$ und $(u, (\text{undef.}, \theta_2))$ zu Knoten $(u, (\theta_1, \theta_2))$ zusammen

Aufbau des Kantengraphen: Alg. 2.12

Aufbau des Kantengraphen: Alg. 2.12

Analyse:

Aufbau des Kantengraphen: Alg. 2.12

Analyse:

Alle Änderungen in $O(mn\lambda_6(mn) \log mn)$,

Aufbau des Kantengraphen: Alg. 2.12

Analyse:

Alle Änderungen in $O(mn\lambda_6(mn) \log mn)$, bereits gezeigt

Aufbau des Kantengraphen: Alg. 2.12

Analyse:

Alle Änderungen in $O(mn\lambda_6(mn) \log mn)$, bereits gezeigt

Kantengraph hat Komplexität $O(mn\lambda_6(mn))$

Aufbau des Kantengraphen: Alg. 2.12

Analyse:

Alle Änderungen in $O(mn\lambda_6(mn) \log mn)$, bereits gezeigt

Kantengraph hat Komplexität $O(mn\lambda_6(mn))$

Anfangs: $O(mn)$,

Aufbau des Kantengraphen: Alg. 2.12

Analyse:

Alle Änderungen in $O(mn\lambda_6(mn) \log mn)$, bereits gezeigt

Kantengraph hat Komplexität $O(mn\lambda_6(mn))$

Anfangs: $O(mn)$, dann kommen noch $O(mn\lambda_6(mn))$ Änderungen dazu

Aufbau des Kantengraphen: Alg. 2.12

Analyse:

Alle Änderungen in $O(mn\lambda_6(mn) \log mn)$, bereits gezeigt

Kantengraph hat Komplexität $O(mn\lambda_6(mn))$

Anfangs: $O(mn)$, dann kommen noch $O(mn\lambda_6(mn))$ Änderungen dazu

Aufbau: $O(mn\lambda_6(mn) \log mn)$!

Bahnplanung mit Kantengraph

Bahnplanung mit Kantengraph

- Bewegung von Orientierung (x_1, y_1, θ_1) zu (x_2, y_2, θ_2) ?

Bahnplanung mit Kantengraph

- Bewegung von Orientierung (x_1, y_1, θ_1) zu (x_2, y_2, θ_2) ?
- Kantengraph repräsentiert den Rand von C_{frei}

Bahnplanung mit Kantengraph

- Bewegung von Orientierung (x_1, y_1, θ_1) zu (x_2, y_2, θ_2) ?
- Kantengraph repräsentiert den Rand von C_{frei}
- Erstmal auf den Rand!!

Bahnplanung mit Kantengraph

- Bewegung von Orientierung (x_1, y_1, θ_1) zu (x_2, y_2, θ_2) ?
- Kantengraph repräsentiert den Rand von C_{frei}
- Erstmal auf den Rand!!
- **Def. 2.44** Abbildung auf den Rand

Bahnplanung mit Kantengraph

- Bewegung von Orientierung (x_1, y_1, θ_1) zu (x_2, y_2, θ_2) ?
- Kantengraph repräsentiert den Rand von C_{frei}
- Erstmal auf den Rand!!
- **Def. 2.44** Abbildung auf den Rand
- $\Phi : C_{frei} \longrightarrow$ Knoten von E

Bahnplanung mit Kantengraph

- Bewegung von Orientierung (x_1, y_1, θ_1) zu (x_2, y_2, θ_2) ?
- Kantengraph repräsentiert den Rand von C_{frei}
- Erstmal auf den Rand!!
- **Def. 2.44** Abbildung auf den Rand
- $\Phi : C_{frei} \longrightarrow$ Knoten von E
- $(x, y, \theta) \longmapsto$ Knoten $(u, L(u))$

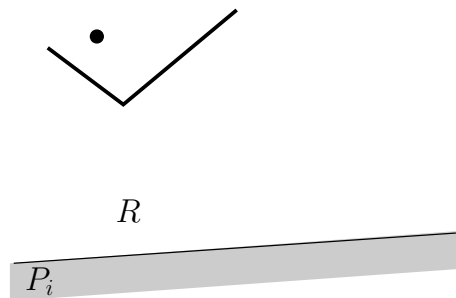
Bahnplanung mit Kantengraph

- Bewegung von Orientierung (x_1, y_1, θ_1) zu (x_2, y_2, θ_2) ?
- Kantengraph repräsentiert den Rand von C_{frei}
- Erstmal auf den Rand!!
- **Def. 2.44** Abbildung auf den Rand
- $\Phi : C_{frei} \longrightarrow$ Knoten von E
- $(x, y, \theta) \longmapsto$ Knoten $(u, L(u))$
- Entspricht Kante e_u von C_{frei} auf dem Rand der Zusammenhangskomponente Z von C_{frei} mit $(x, y, \theta) \in Z$

Bahnplanung mit Kantengraph

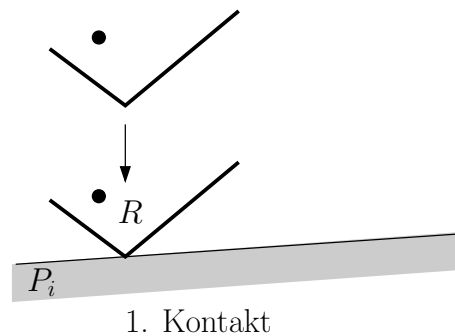
- Bewegung von Orientierung (x_1, y_1, θ_1) zu (x_2, y_2, θ_2) ?
- Kantengraph repräsentiert den Rand von C_{frei}
- Erstmal auf den Rand!!
- **Def. 2.44** Abbildung auf den Rand
- $\Phi : C_{frei} \longrightarrow$ Knoten von E
- $(x, y, \theta) \longmapsto$ Knoten $(u, L(u))$
- Entspricht Kante e_u von C_{frei} auf dem Rand der Zusammenhangskomponente Z von C_{frei} mit $(x, y, \theta) \in Z$
- Konstruktiv!!

Def. 2.44 Abbildung



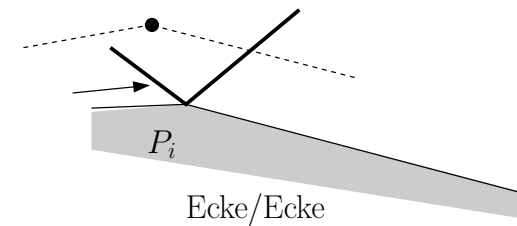
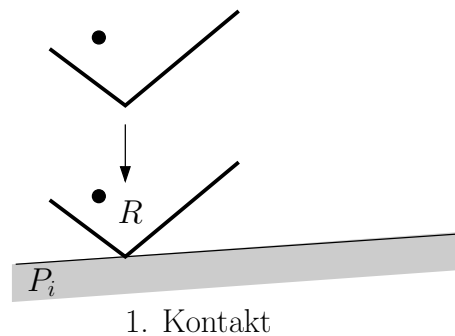
Def. 2.44 Abbildung

- Bewegung von R nach unten: Kontakt I



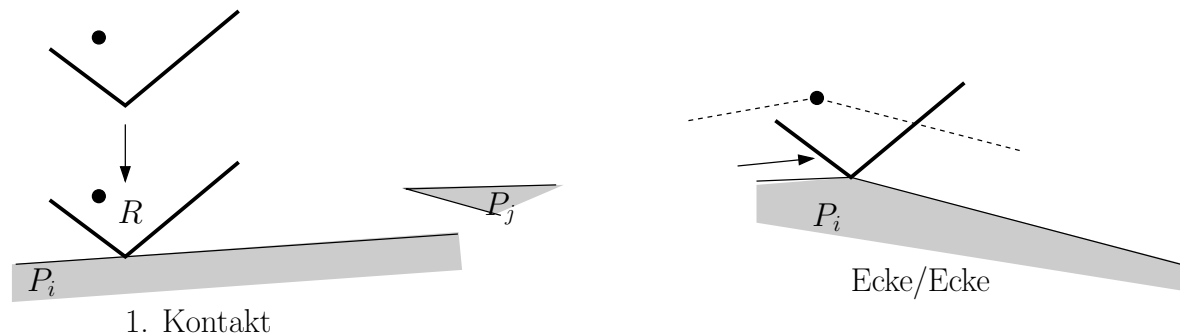
Def. 2.44 Abbildung

- Bewegung von R nach unten: Kontakt I
- Bereits Ecke/Ecke oder Mehrfach-Kontakt: Fertig



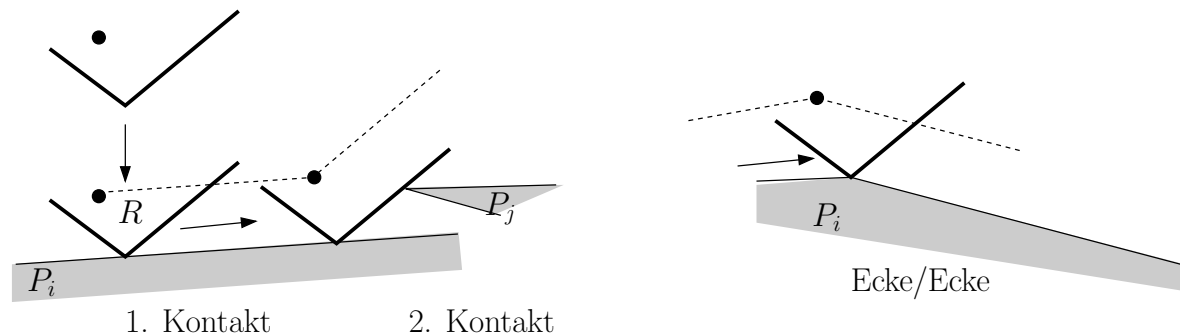
Def. 2.44 Abbildung

- Bewegung von R nach unten: Kontakt I
- Bereits Ecke/Ecke oder Mehrfach-Kontakt: Fertig
- Sonst Bewegung nach rechts, bis Knoten in V^θ



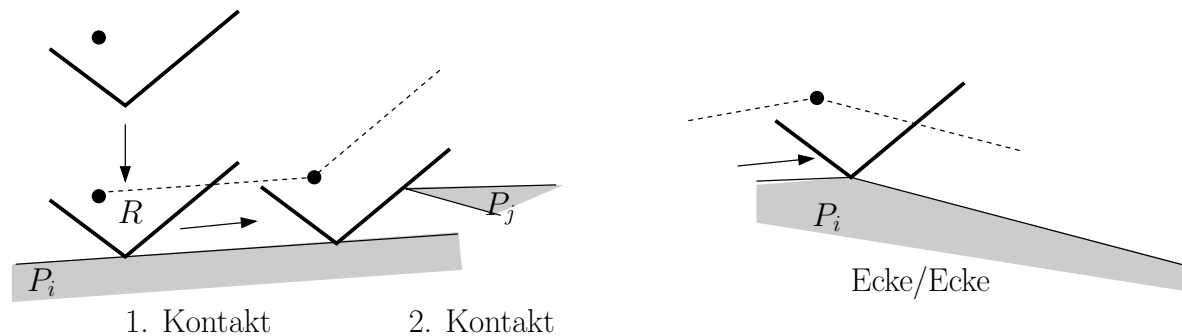
Def. 2.44 Abbildung

- Bewegung von R nach unten: Kontakt I
- Bereits Ecke/Ecke oder Mehrfach-Kontakt: Fertig
- Sonst Bewegung nach rechts, bis Knoten in V^θ



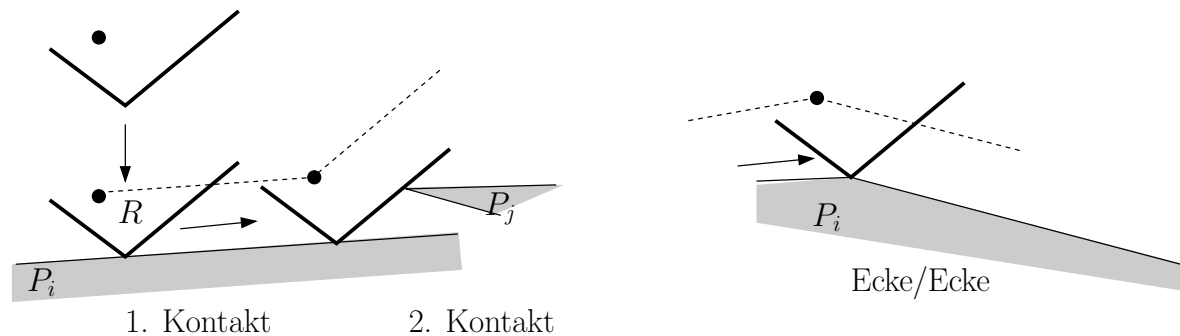
Def. 2.44 Abbildung

- Bewegung von R nach unten: Kontakt I
- Bereits Ecke/Ecke oder Mehrfach-Kontakt: Fertig
- Sonst Bewegung nach rechts, bis Knoten in V^θ
- Erweiterter Knoten $(u, L(u))$ in E



Def. 2.44 Abbildung

- Bewegung von R nach unten: Kontakt I
- Bereits Ecke/Ecke oder Mehrfach-Kontakt: Fertig
- Sonst Bewegung nach rechts, bis Knoten in V^θ
- Erweiterter Knoten $(u, L(u))$ in E
- In $O(nm)$ berechnen



Bahnplanung mit Kantengraph

Bahnplanung mit Kantengraph

- Bewegung von Orientierung $s = (x_s, y_s, \theta_s)$ zu $t = (x_t, y_t, \theta_t)$?

Bahnplanung mit Kantengraph

- Bewegung von Orientierung $s = (x_s, y_s, \theta_s)$ zu $t = (x_t, y_t, \theta_t)$?
- Kollisionsfrei?

Bahnplanung mit Kantengraph

- Bewegung von Orientierung $s = (x_s, y_s, \theta_s)$ zu $t = (x_t, y_t, \theta_t)$?
- Kollisionsfrei?
- Existiert Weg von $\Phi(s)$ zu $\Phi(t)$ in E ?

Bahnplanung mit Kantengraph

- Bewegung von Orientierung $s = (x_s, y_s, \theta_s)$ zu $t = (x_t, y_t, \theta_t)$?
- Kollisionsfrei?
- Existiert Weg von $\Phi(s)$ zu $\Phi(t)$ in E ?
- Korrektheit

Bahnplanung mit Kantengraph

- Bewegung von Orientierung $s = (x_s, y_s, \theta_s)$ zu $t = (x_t, y_t, \theta_t)$?
- Kollisionsfrei?
- Existiert Weg von $\Phi(s)$ zu $\Phi(t)$ in E ?
- Korrektheit

Theorem 2.45:

Bahnplanung mit Kantengraph

- Bewegung von Orientierung $s = (x_s, y_s, \theta_s)$ zu $t = (x_t, y_t, \theta_t)$?
- Kollisionsfrei?
- Existiert Weg von $\Phi(s)$ zu $\Phi(t)$ in E ?
- Korrektheit

Theorem 2.45: s, t zwei freie Plazierungen von R .

Bahnplanung mit Kantengraph

- Bewegung von Orientierung $s = (x_s, y_s, \theta_s)$ zu $t = (x_t, y_t, \theta_t)$?
- Kollisionsfrei?
- Existiert Weg von $\Phi(s)$ zu $\Phi(t)$ in E ?
- Korrektheit

Theorem 2.45: s, t zwei freie Plazierungen von R .

Es gibt eine kollisionsfreie Bewegung von s nach t genau dann, wenn $\Phi(s)$ und $\Phi(t)$ Knoten derselben Zusammenhangskomponente Z des Kantengraphen E sind.

Beweis: Theorem 2.45

Beweis: Theorem 2.45

Lemma 2.46:

Beweis: Theorem 2.45

Lemma 2.46:

Zur Zusammenhangskomponente W von C_{frei} gehört eindeutig eine Zusammenhangskomponente Z von E mit:

Beweis: Theorem 2.45

Lemma 2.46:

Zur Zusammenhangskomponente W von C_{frei} gehört eindeutig eine Zusammenhangskomponente Z von E mit:

Knoten von $Z \iff$ Kanten in ∂W .

Beweis: Theorem 2.45

Lemma 2.46:

Zur Zusammenhangskomponente W von C_{frei} gehört eindeutig eine Zusammenhangskomponente Z von E mit:

Knoten von $Z \longleftrightarrow$ Kanten in ∂W .

Zusammenhangskomponenten korrelieren