

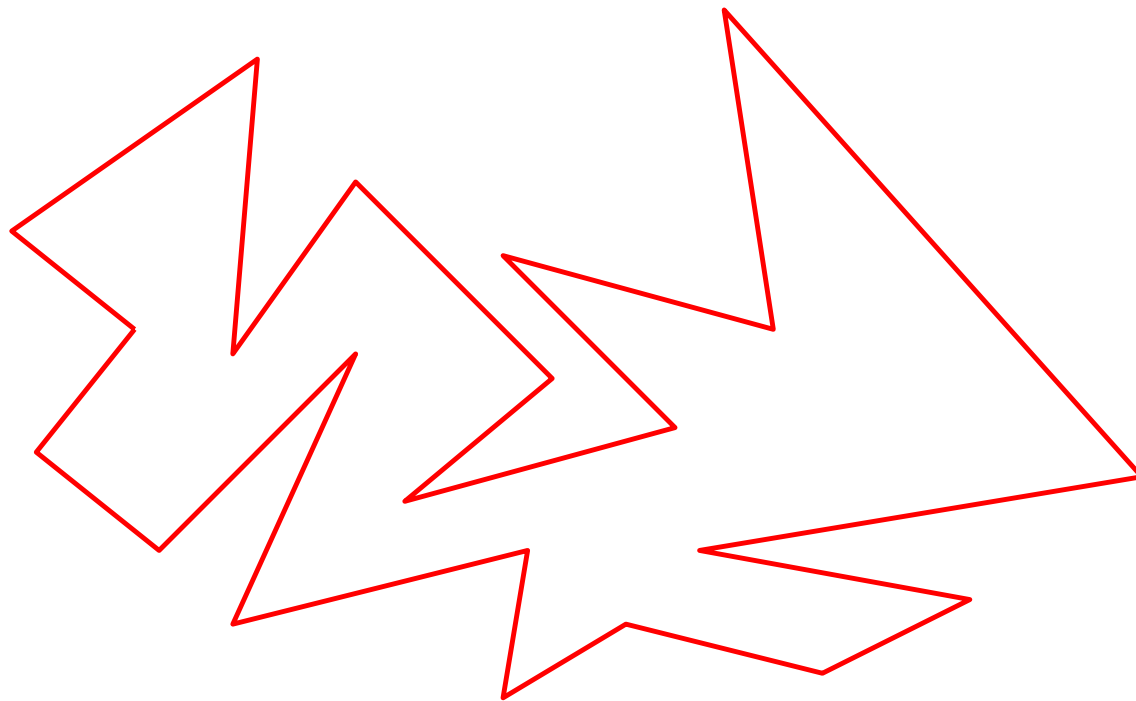
Offline Bewegungsplanung: Maximaler Durchmesser

Elmar Langetepe
University of Bonn

1.2.3 Durchmesser einfacher Polygone

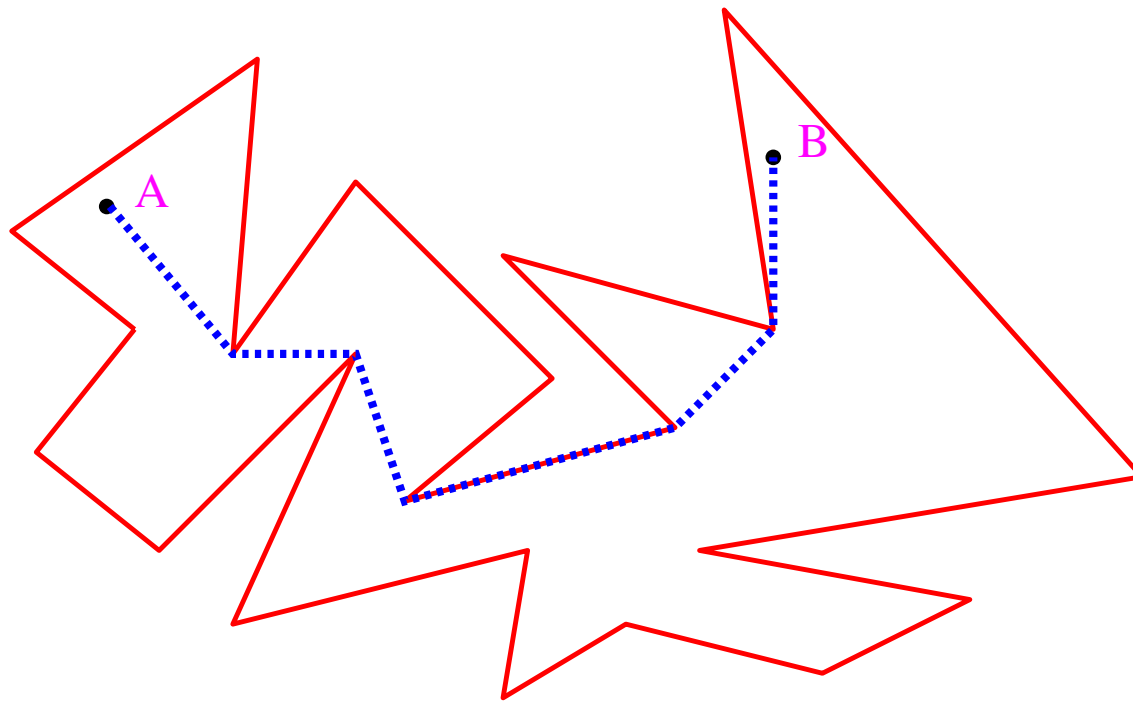
1.2.3 Durchmesser einfacher Polygone

- Einfaches Polygon P



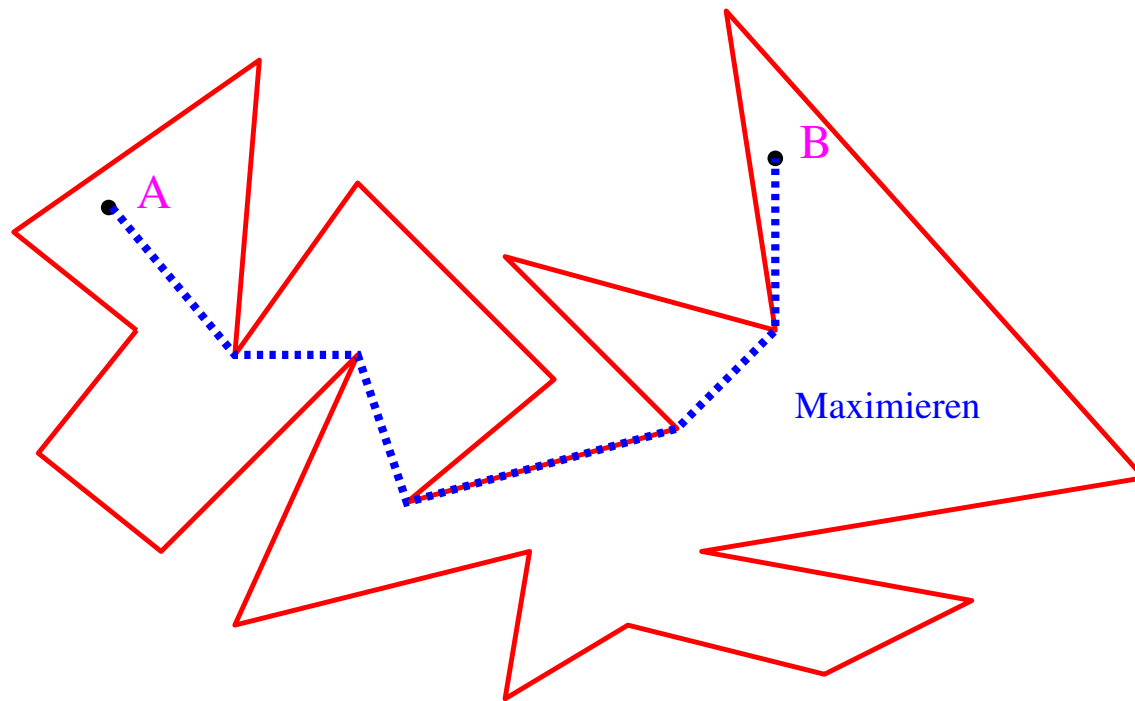
1.2.3 Durchmesser einfacher Polygone

- Einfaches Polygon P
- Längster Kürzester Weg zwischen zwei Punkten



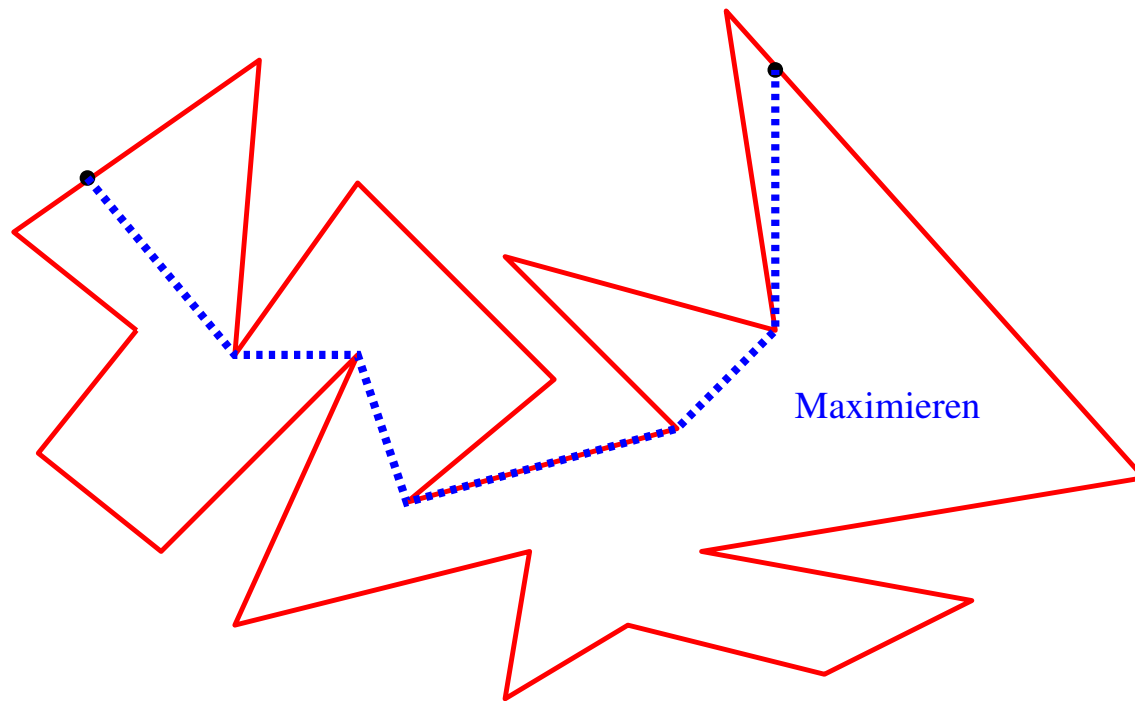
1.2.3 Durchmesser einfacher Polygone

- Einfaches Polygon P
- Längster Kürzester Weg zwischen zwei Punkten
- Endpunkte sind Ecken des Polygons



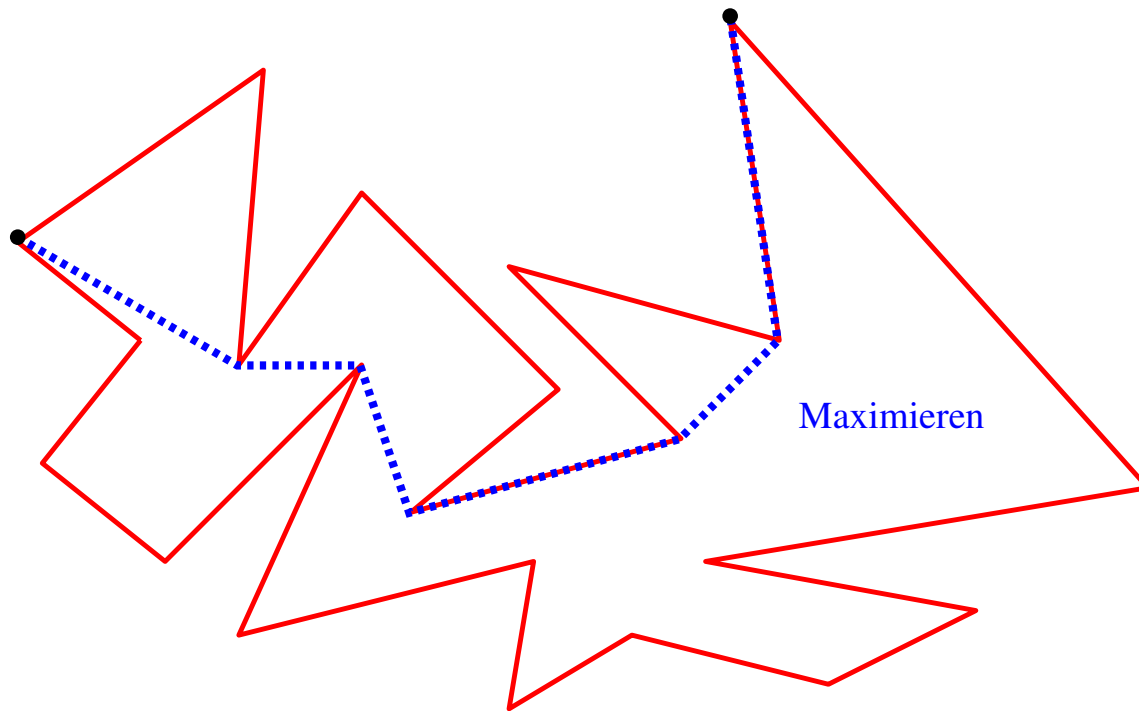
1.2.3 Durchmesser einfacher Polygone

- Einfaches Polygon P
- Längster kürzester Weg zwischen zwei Punkten
- Endpunkte sind Ecken des Polygons



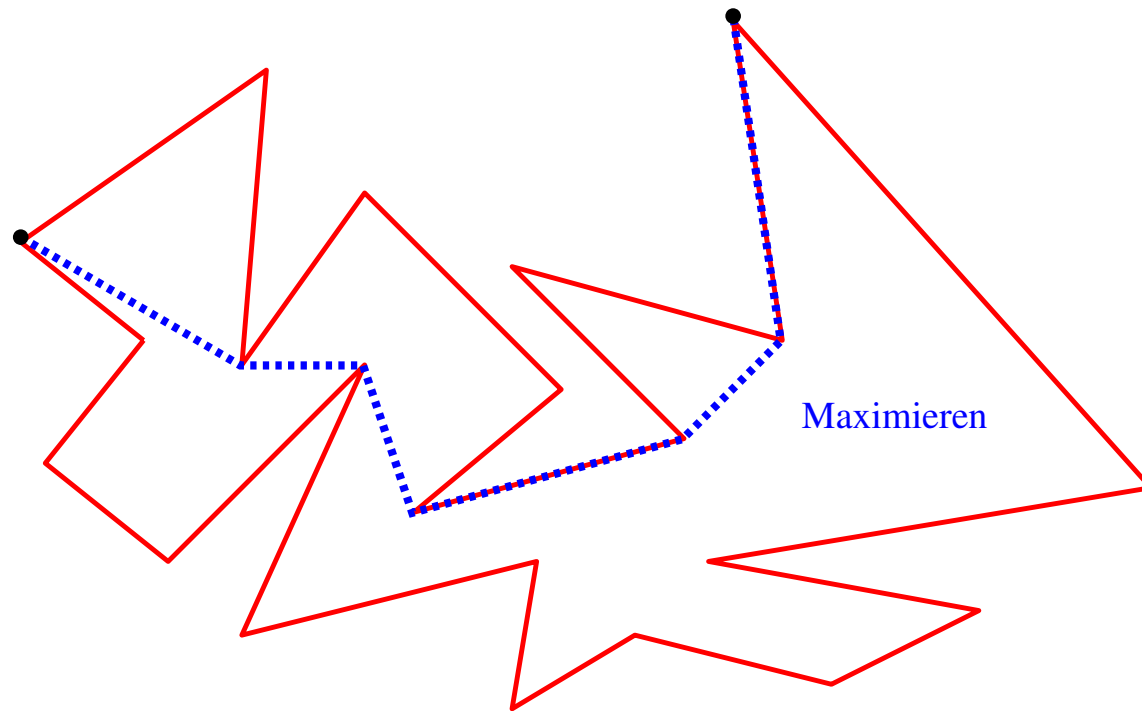
1.2.3 Durchmesser einfacher Polygone

- Einfaches Polygon P
- Längster kürzester Weg zwischen zwei Punkten
- Endpunkte sind Ecken des Polygons



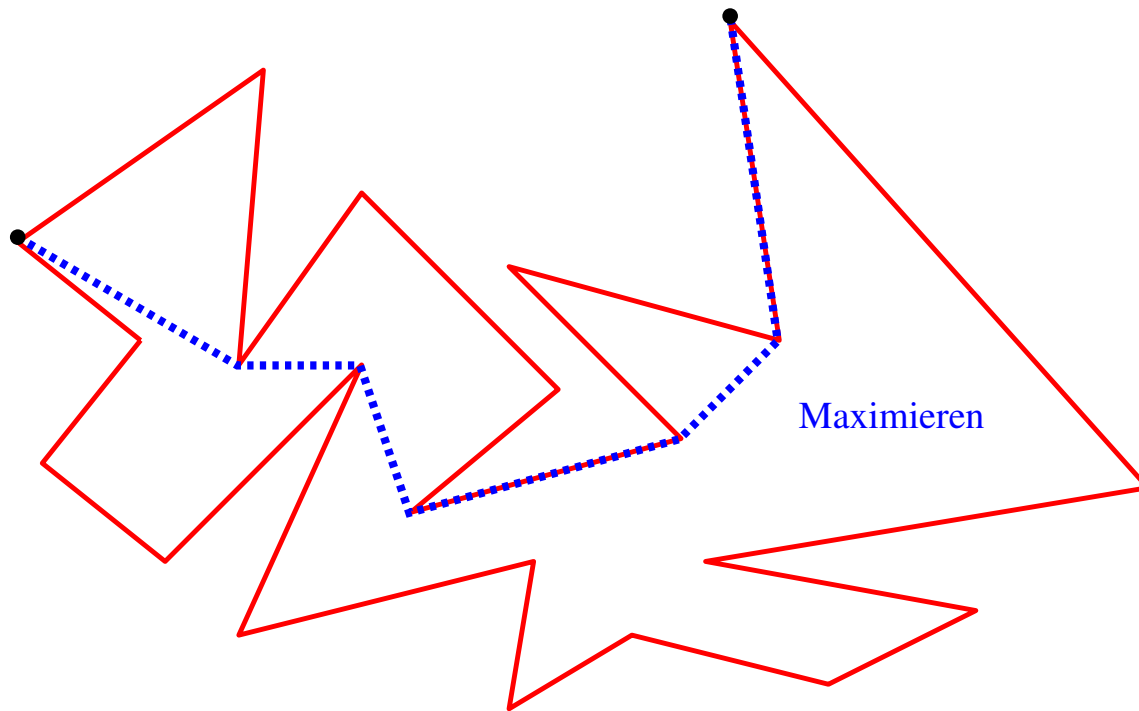
1.2.3 Durchmesser einfacher Polygone

- Einfaches Polygon P
- Längster kürzester Weg zwischen zwei Punkten
- Endpunkte sind Ecken des Polygons n^2 Kandidatenpaare



1.2.3 Durchmesser einfacher Polygone

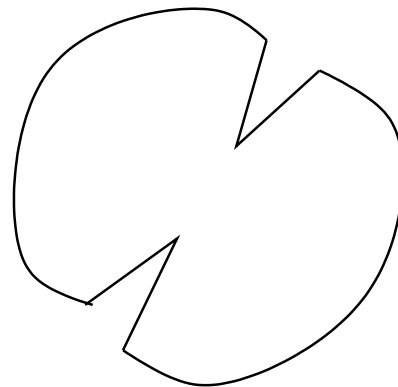
- Einfaches Polygon P
- Längster Kürzester Weg zwischen zwei Punkten
- Endpunkte sind Ecken des Polygons n^2 Kandidatenpaare
- Formal: $\max_{p_i, p_j} \text{Ecken von } P \ d(p_i, p_j)$



Idee der Berechnung:

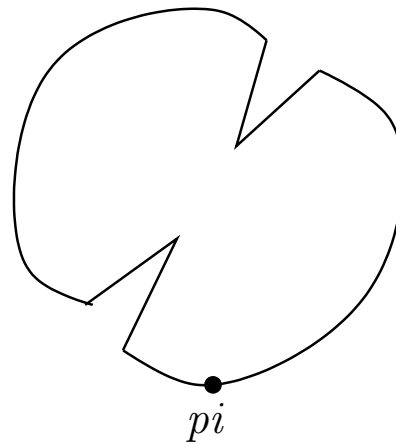
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes



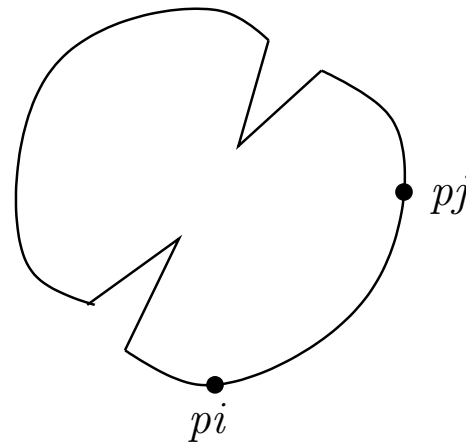
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes



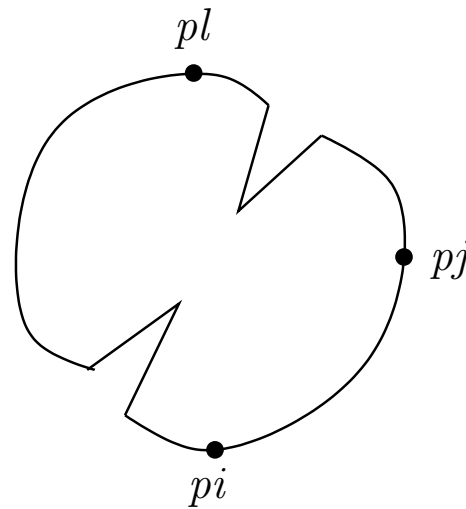
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes



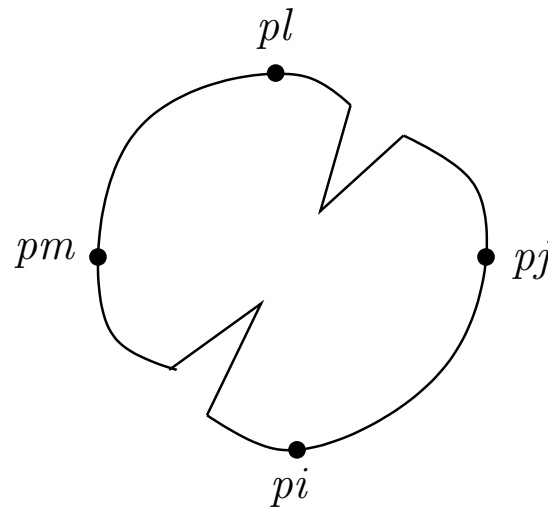
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes



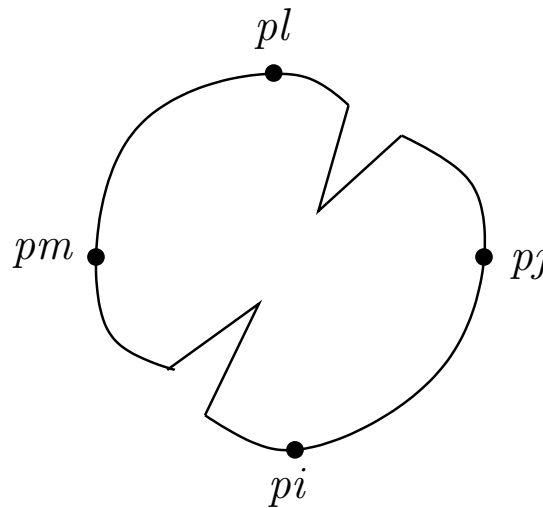
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes



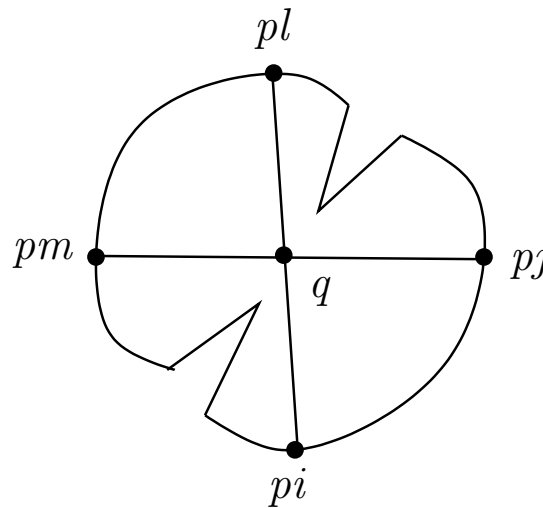
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes
- Monge Eigenschaft: $d(p_i, p_m) + d(p_j, p_l) \leq d(p_i, p_l) + d(p_j, p_m)$



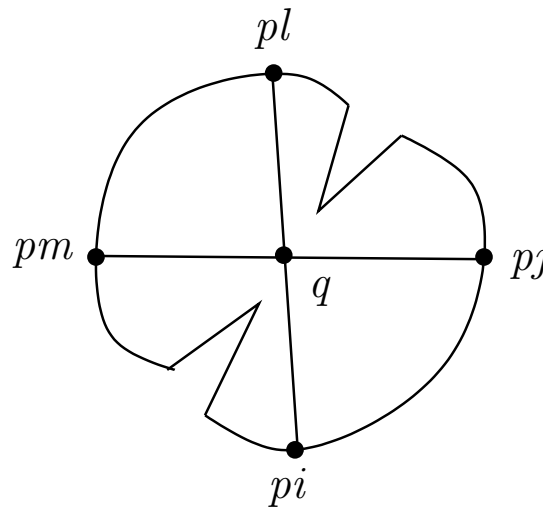
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes
- Monge Eigenschaft: $d(p_i, p_m) + d(p_j, p_l) \leq d(p_i, p_l) + d(p_j, p_m)$
- Wege $\pi(p_i, p_l)$ und $\pi(p_j, p_m)$ schneiden sich



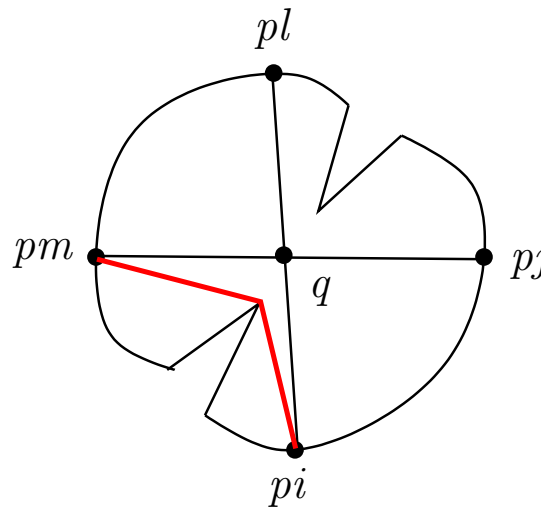
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes
- Monge Eigenschaft: $d(p_i, p_m) + d(p_j, p_l) \leq d(p_i, p_l) + d(p_j, p_m)$
- Wege $\pi(p_i, p_l)$ und $\pi(p_j, p_m)$ schneiden sich
- Dreiecksungleichungen anwenden!!



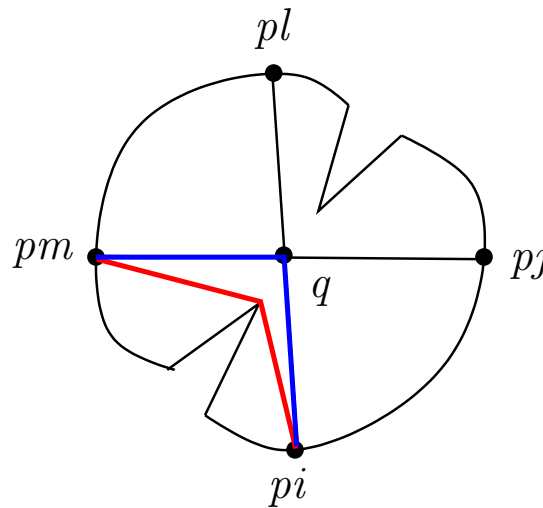
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes
- Monge Eigenschaft: $d(p_i, p_m) + d(p_j, p_l) \leq d(p_i, p_l) + d(p_j, p_m)$
- Wege $\pi(p_i, p_l)$ und $\pi(p_j, p_m)$ schneiden sich
- Dreiecksungleichungen anwenden!!



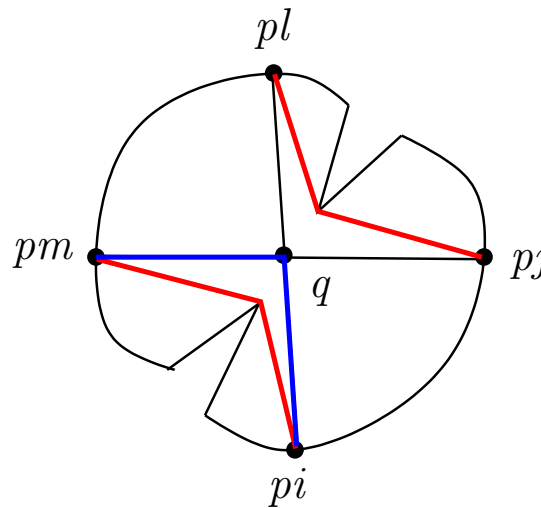
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes
- Monge Eigenschaft: $d(p_i, p_m) + d(p_j, p_l) \leq d(p_i, p_l) + d(p_j, p_m)$
- Wege $\pi(p_i, p_l)$ und $\pi(p_j, p_m)$ schneiden sich
- Dreiecksungleichungen anwenden!!



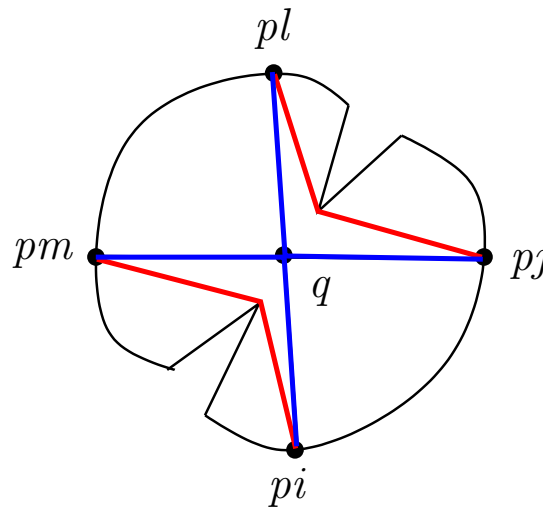
Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes
- Monge Eigenschaft: $d(p_i, p_m) + d(p_j, p_l) \leq d(p_i, p_l) + d(p_j, p_m)$
- Wege $\pi(p_i, p_l)$ und $\pi(p_j, p_m)$ schneiden sich
- Dreiecksungleichungen anwenden!!



Idee der Berechnung:

- p_i, p_j, p_l, p_m entlang des Randes
- Monge Eigenschaft: $d(p_i, p_m) + d(p_j, p_l) \leq d(p_i, p_l) + d(p_j, p_m)$
- Wege $\pi(p_i, p_l)$ und $\pi(p_j, p_m)$ schneiden sich
- Dreiecksungleichungen anwenden!!



Matrix A mit schöner Eigenschaft!

$$\begin{matrix}
 & 1 & 2 & 3 & & n-1 & n \\
 1 & 1-n & d(1,2) & d(1,3) & \dots & d(1,n-1) & d(1,n) \\
 2 & 1-n & 2-n & d(2,3) & \dots & d(2,n-1) & d(2,n) \\
 3 & 1-n & 2-n & 3-n & \dots & d(3,n-1) & d(3,n) \\
 \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\
 n-1 & 1-n & 2-n & 3-n & \dots & \dots & d(n-1,n) \\
 n & 1-n & 2-n & 3-n & \dots & \dots & 0
 \end{matrix}$$

$d(i, j)$ entspricht $d(p_i, p_j)$ und p_1, p_2, \dots, p_n Knoten entlang des Randes von P !

Def. 1.17: Monotone Matrix!

Def. 1.17: Monotone Matrix!

$n \times m$ -Matrix $A = (a_{il})$ heißt *monoton*, falls

Def. 1.17: Monotone Matrix!

$n \times m$ -Matrix $A = (a_{i\ell})$ heißt *monoton*, falls

$$\forall 1 \leq i < j \leq n, 1 \leq k < \ell \leq m : (a_{ik} < a_{i\ell} \Rightarrow a_{jk} < a_{j\ell}).$$

Def. 1.17: Monotone Matrix!

$n \times m$ -Matrix $A = (a_{il})$ heißt *monoton*, falls

$$\forall 1 \leq i < j \leq n, 1 \leq k < \ell \leq m : (a_{ik} < a_{i\ell} \Rightarrow a_{jk} < a_{j\ell}).$$

$$\begin{array}{c} \\ \end{array} \begin{array}{cc} k & \ell \\ \left(\begin{array}{ccc} a_{ik} & < & a_{i\ell} \\ & \Downarrow & \\ a_{jk} & < & a_{j\ell} \end{array} \right) \end{array}$$

Linkeste Zeilenmaxima weiter nach rechts

Linkeste Zeilenmaxima weiter nach rechts

$$\begin{pmatrix} 5 & 5 & 5 & 5 & 5 \\ 1 & 7 & 9 & 6 & 3 \\ 4 & 6 & 10 & 7 & 12 \end{pmatrix}$$

Lemma 1.19: A ist monoton!

Lemma 1.19: A ist monoton!

$$\begin{array}{c}
 1 \quad 2 \quad 3 \quad \dots \quad n-1 \quad n \\
 \begin{array}{c}
 1 \\
 2 \\
 3 \\
 \vdots \\
 n-1 \\
 n
 \end{array}
 \left(\begin{array}{cccccc}
 1-n & d(1,2) & d(1,3) & \dots & d(1,n-1) & d(1,n) \\
 1-n & 2-n & d(2,3) & \dots & d(2,n-1) & d(2,n) \\
 1-n & 2-n & 3-n & \dots & d(3,n-1) & d(3,n) \\
 \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
 1-n & 2-n & 3-n & \dots & \dots & d(n-1,n) \\
 1-n & 2-n & 3-n & \dots & \dots & 0
 \end{array} \right)
 \end{array}$$

Lemma 1.19: A ist monoton!

$$\begin{array}{c}
 1 \quad 2 \quad 3 \quad \dots \quad n-1 \quad n \\
 \begin{array}{c}
 1 \\
 2 \\
 3 \\
 \vdots \\
 n-1 \\
 n
 \end{array}
 \left(\begin{array}{cccccc}
 1-n & d(1,2) & d(1,3) & \dots & d(1,n-1) & d(1,n) \\
 1-n & 2-n & d(2,3) & \dots & d(2,n-1) & d(2,n) \\
 1-n & 2-n & 3-n & \dots & d(3,n-1) & d(3,n) \\
 \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\
 1-n & 2-n & 3-n & \dots & \dots & d(n-1,n) \\
 1-n & 2-n & 3-n & \dots & \dots & 0
 \end{array} \right)
 \end{array}$$

Beweis!!!

Theorem 1.20: Maxima für monotone Matrizen

Theorem 1.20: Maxima für monotone Matrizen

- Monotone $n \times m$ -Matrix $A = (a_{il})$, $m \geq n$

Theorem 1.20: Maxima für monotone Matrizen

- Monotone $n \times m$ -Matrix $A = (a_{il})$, $m \geq n$
- Alg.1.7 Spaltenreduktion (Spalten streichen): monotone $n \times n$ -Matrix A'

Theorem 1.20: Maxima für monotone Matrizen

- Monotone $n \times m$ -Matrix $A = (a_{il})$, $m \geq n$
- Alg.1.7 Spaltenreduktion (Spalten streichen): monotone $n \times n$ -Matrix A'
- Zeilenmaxima (A, i)

Theorem 1.20: Maxima für monotone Matrizen

- Monotone $n \times m$ -Matrix $A = (a_{il})$, $m \geq n$
- Alg.1.7 Spaltenreduktion (Spalten streichen): monotone $n \times n$ -Matrix A'
- Zeilenmaxima (A, i) aus Zeilenmaxima (A', i) gewinnen

Theorem 1.20: Maxima für monotone Matrizen

- Monotone $n \times m$ -Matrix $A = (a_{il})$, $m \geq n$
- Alg.1.7 Spaltenreduktion (Spalten streichen): monotone $n \times n$ -Matrix A'
- Zeilenmaxima (A, i) aus Zeilenmaxima (A', i) gewinnen
- Alg.1.8 Zeilenreduktion (gerade Zeilen auswählen): Monotone $n \times m$ -Matrix B

Theorem 1.20: Maxima für monotone Matrizen

- Monotone $n \times m$ -Matrix $A = (a_{il})$, $m \geq n$
- Alg.1.7 Spaltenreduktion (Spalten streichen): monotone $n \times n$ -Matrix A'
- Zeilenmaxima (A, i) aus Zeilenmaxima (A', i) gewinnen
- Alg.1.8 Zeilenreduktion (gerade Zeilen auswählen): Monotone $n \times m$ -Matrix B
- Zeilenmaxima ungerade Zeilen

Theorem 1.20: Maxima für monotone Matrizen

- Monotone $n \times m$ -Matrix $A = (a_{il})$, $m \geq n$
- Alg.1.7 Spaltenreduktion (Spalten streichen): monotone $n \times n$ -Matrix A'
- Zeilenmaxima (A, i) aus Zeilenmaxima (A', i) gewinnen
- Alg.1.8 Zeilenreduktion (gerade Zeilen auswählen): Monotone $n \times m$ -Matrix B
- Zeilenmaxima ungerade Zeilen aus Zeilenmaxima gerade Zeilen gewinnen

Theorem 1.20: Maxima für monotone Matrizen

- Monotone $n \times m$ -Matrix $A = (a_{il})$, $m \geq n$
- Alg.1.7 Spaltenreduktion (Spalten streichen): monotone $n \times n$ -Matrix A'
- Zeilenmaxima (A, i) aus Zeilenmaxima (A', i) gewinnen
- Alg.1.8 Zeilenreduktion (gerade Zeilen auswählen): Monotone $n \times m$ -Matrix B
- Zeilenmaxima ungerade Zeilen aus Zeilenmaxima gerade Zeilen gewinnen
- Rekursiv mit Spaltenreduktion (Alg. 1.7)!!

Alg. 1.7: Spaltenreduktion

Alg. 1.7: Spaltenreduktion

Invariante: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \geq a_{1,4} \quad a_{1,5}$
 $a_{2,2} \geq a_{2,3} \geq a_{2,4} \quad a_{2,5}$
 $a_{3,3} \geq a_{3,4} \quad a_{3,5}$
 $a_{4,4} \stackrel{?}{<} a_{4,5}$

Alg. 1.7: Spaltenreduktion

$$\begin{array}{cccccc} \text{Invariante:} & a_{1,1} & \geq & a_{1,2} & \geq & a_{1,3} & \geq & a_{1,4} & & a_{1,5} \\ & & & a_{2,2} & \geq & a_{2,3} & \geq & a_{2,4} & & a_{2,5} \\ & & & & & a_{3,3} & \geq & a_{3,4} & & a_{3,5} \\ & & & & & & & a_{4,4} & < & a_{4,5} \end{array}$$

Falls nein, dann:

Alg. 1.7: Spaltenreduktion

Invariante: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \geq a_{1,4} \geq a_{1,5}$
 $a_{2,2} \geq a_{2,3} \geq a_{2,4} \geq a_{2,5}$
 $a_{3,3} \geq a_{3,4} \geq a_{3,5}$
 $a_{4,4} \stackrel{?}{<} a_{4,5}$

Falls nein, dann: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \geq a_{1,4} \geq a_{1,5}$
 $a_{2,2} \geq a_{2,3} \geq a_{2,4} \geq a_{2,5}$
 $a_{3,3} \geq a_{3,4} \geq a_{3,5}$
 $a_{4,4} \geq a_{4,5}$

Alg. 1.7: Spaltenreduktion

Alg. 1.7: Spaltenreduktion

Invariante: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \geq a_{1,4} \quad a_{1,5}$
 $a_{2,2} \geq a_{2,3} \geq a_{2,4} \quad a_{2,5}$
 $a_{3,3} \geq a_{3,4} \quad a_{3,5}$
 $a_{4,4} \stackrel{?}{<} a_{4,5}$

Alg. 1.7: Spaltenreduktion

Invariante: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \geq a_{1,4} \geq a_{1,5}$
 $a_{2,2} \geq a_{2,3} \geq a_{2,4} \geq a_{2,5}$
 $a_{3,3} \geq a_{3,4} \geq a_{3,5}$
 $a_{4,4} \stackrel{?}{<} a_{4,5}$

Falls ja, dann:

Alg. 1.7: Spaltenreduktion

Invariante: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \geq a_{1,4} \quad a_{1,5}$
 $a_{2,2} \geq a_{2,3} \geq a_{2,4} \quad a_{2,5}$
 $a_{3,3} \geq a_{3,4} \quad a_{3,5}$
 $a_{4,4} \overset{?}{<} a_{4,5}$

Falls ja, dann: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \quad a_{1,5}$
 $a_{2,2} \geq a_{2,3} \quad a_{2,5}$
 $a_{3,3} \overset{?}{<} a_{3,5}$
 $a_{4,5}$

Alg. 1.7: Spaltenreduktion

Invariante: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \geq a_{1,4} \quad a_{1,5}$
 $a_{2,2} \geq a_{2,3} \geq a_{2,4} \quad a_{2,5}$
 $a_{3,3} \geq a_{3,4} \quad a_{3,5}$
 $a_{4,4} \overset{?}{<} a_{4,5}$

Falls ja, dann: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \quad a_{1,5}$
 $a_{2,2} \geq a_{2,3} \quad a_{2,5}$
 $a_{3,3} \overset{?}{<} a_{3,5}$
 $a_{4,5}$

Streiche Spalte 4, weil:

Alg. 1.7: Spaltenreduktion

Invariante: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \geq a_{1,4} \quad a_{1,5}$
 $a_{2,2} \geq a_{2,3} \geq a_{2,4} \quad a_{2,5}$
 $a_{3,3} \geq a_{3,4} \quad a_{3,5}$
 $a_{4,4} \overset{?}{<} a_{4,5}$

Falls ja, dann: $a_{1,1} \geq a_{1,2} \geq a_{1,3} \quad a_{1,5}$
 $a_{2,2} \geq a_{2,3} \quad a_{2,5}$
 $a_{3,3} \overset{?}{<} a_{3,5}$
 $a_{4,5}$

Streiche Spalte 4, weil:

$$a_{i,4} \leq a_{i,3} \text{ für } i = 1, 2, 3 \text{ und } a_{i,4} < a_{i,5} \text{ für } i = 4, \dots, n$$

Alg. 1.7: Spaltenreduktion

Alg. 1.7: Spaltenreduktion

Letzte Zeile erreicht!

Alg. 1.7: Spaltenreduktion

Letzte Zeile erreicht!

$$\begin{array}{ccccccccccc} a_{1,1} & \geq & a_{1,2} & \geq & a_{1,3} & \dots & a_{1,n} & \geq & a_{1,n+1} & \dots & a_{1,m} \\ & & a_{2,2} & \geq & a_{2,3} & \dots & a_{2,n} & \geq & a_{2,n+1} & \dots & a_{2,m} \\ & & & & \dots & & & & & & \\ & & & & & & & & & & \\ & & & & & & a_{n,n} & \geq & a_{n,n+1} & \dots & a_{n,m} \end{array}$$

Alg. 1.7: Spaltenreduktion

Letzte Zeile erreicht!

$$\begin{array}{ccccccccccc} a_{1,1} & \geq & a_{1,2} & \geq & a_{1,3} & \cdots & a_{1,n} & \geq & a_{1,n+1} & \cdots & a_{1,m} \\ & & a_{2,2} & \geq & a_{2,3} & \cdots & a_{2,n} & \geq & a_{2,n+1} & \cdots & a_{2,m} \\ & & & & \cdots & \cdots & & & & & \\ & & & & & & & & & & \\ & & & & & & a_{n,n} & \geq & a_{n,n+1} & \cdots & a_{n,m} \end{array}$$

Komplette Spalte $n + 1$ Streichen!!

Alg. 1.7: Spaltenreduktion

Letzte Zeile erreicht!

$$\begin{array}{cccccccc} a_{1,1} & \geq & a_{1,2} & \geq & a_{1,3} & \dots & a_{1,n} & \geq & a_{1,n+1} & \dots & a_{1,m} \\ & & a_{2,2} & \geq & a_{2,3} & \dots & a_{2,n} & \geq & a_{2,n+1} & \dots & a_{2,m} \\ & & & & \dots & & & & & & \\ & & & & & & & & & & \\ & & & & & & a_{n,n} & \geq & a_{n,n+1} & \dots & a_{n,m} \end{array}$$

Komplette Spalte $n + 1$ Streichen!!

Weiter mit Vergleich: $a_{n,n} \stackrel{?}{<} a_{n,n+2}$

Alg. 1.7: Spaltenreduktion

Letzte Zeile erreicht!

$$\begin{array}{cccccccc} a_{1,1} & \geq & a_{1,2} & \geq & a_{1,3} & \dots & a_{1,n} & \geq & a_{1,n+1} & \dots & a_{1,m} \\ & & a_{2,2} & \geq & a_{2,3} & \dots & a_{2,n} & \geq & a_{2,n+1} & \dots & a_{2,m} \\ & & & & \dots & & & & & & \\ & & & & & & & & & & \\ & & & & & & a_{n,n} & \geq & a_{n,n+1} & \dots & a_{n,m} \end{array}$$

Komplette Spalte $n + 1$ Streichen!!

Weiter mit Vergleich: $a_{n,n} < a_{n,n+2}$

Beispiel!!!

Alg. 1.7: Spaltenreduktion

Alg. 1.7: Spaltenreduktion

- Input: monotone $n \times m$ Matrix A , $m \geq n$

Alg. 1.7: Spaltenreduktion

- Input: monotone $n \times m$ Matrix A , $m \geq n$
- Output: monotone $n \times n$ Matrix A'

Alg. 1.7: Spaltenreduktion

- Input: monotone $n \times m$ Matrix A , $m \geq n$
- Output: monotone $n \times n$ Matrix A'
- Zeilenmaxima von A' und A identisch

Alg. 1.7: Spaltenreduktion

- Input: monotone $n \times m$ Matrix A , $m \geq n$
- Output: monotone $n \times n$ Matrix A'
- Zeilenmaxima von A' und A identisch
- Analyse:

Alg. 1.7: Spaltenreduktion

- Input: monotone $n \times m$ Matrix A , $m \geq n$
- Output: monotone $n \times n$ Matrix A'
- Zeilenmaxima von A' und A identisch
- Analyse:
 - $O(m)$ Vergleiche

Alg. 1.7: Spaltenreduktion

- Input: monotone $n \times m$ Matrix A , $m \geq n$
- Output: monotone $n \times n$ Matrix A'
- Zeilenmaxima von A' und A identisch
- Analyse:
 - $O(m)$ Vergleiche
 - $O(m)$ Zeiger für Rekonstruktion

Alg. 1.7: Spaltenreduktion

- Input: monotone $n \times m$ Matrix A , $m \geq n$
- Output: monotone $n \times n$ Matrix A'
- Zeilenmaxima von A' und A identisch
- Analyse:
 - $O(m)$ Vergleiche
 - $O(m)$ Zeiger für Rekonstruktion

Alg. 1.8: Zeilenmaxima

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index $O(n)$
 - C' durch Spaltenreduktion(C)

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index $O(n)$
 - C' durch Spaltenreduktion(C) $O(m)$

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index $O(n)$
 - C' durch Spaltenreduktion(C) $O(m)$
 - Rekursiv: Zeilenmaxima(C'), $\frac{n}{2} \times \frac{n}{2}$ Matrix

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index $O(n)$
 - C' durch Spaltenreduktion(C) $O(m)$
 - Rekursiv: Zeilenmaxima(C'), $\frac{n}{2} \times \frac{n}{2}$ Matrix $T \left(\frac{n}{2}\right)$

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index $O(n)$
 - C' durch Spaltenreduktion(C) $O(m)$
 - Rekursiv: Zeilenmaxima(C'), $\frac{n}{2} \times \frac{n}{2}$ Matrix $T \left(\frac{n}{2}\right)$
 - Rekonstruktion (Zeiger) Zeilenmaxima von C

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index $O(n)$
 - C' durch Spaltenreduktion(C) $O(m)$
 - Rekursiv: Zeilenmaxima(C'), $\frac{n}{2} \times \frac{n}{2}$ Matrix $T \left(\frac{n}{2}\right)$
 - Rekonstruktion (Zeiger) Zeilenmaxima von C $O(m)$

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index $O(n)$
 - C' durch Spaltenreduktion(C) $O(m)$
 - Rekursiv: Zeilenmaxima(C'), $\frac{n}{2} \times \frac{n}{2}$ Matrix $T \left(\frac{n}{2}\right)$
 - Rekonstruktion (Zeiger) Zeilenmaxima von C $O(m)$
 - **Berechnung** Zeilenmaxima von B

Alg. 1.8: Zeilenmaxima

- Input: monotone $n \times m$ Matrix B
- Output: alle linkesten Zeilenmaxima $\max(i)$, $1 \leq i \leq n$
- Algorithmus:
 - C Zeilen von B mit geradem Index $O(n)$
 - C' durch Spaltenreduktion(C) $O(m)$
 - Rekursiv: Zeilenmaxima(C'), $\frac{n}{2} \times \frac{n}{2}$ Matrix $T \left(\frac{n}{2}\right)$
 - Rekonstruktion (Zeiger) Zeilenmaxima von C $O(m)$
 - **Berechnung** Zeilenmaxima von B $O(m)$

$$\begin{pmatrix} X & X & X & X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X & X & X & X \end{pmatrix}$$

$O(m)$ Schritte !!

Berechnung Zeilenmaxima von B

$$\begin{pmatrix} x & x & x & x & x & x & x & x & x & x & x \\ x & x & x & x & x & x & x & x & x & x & x \\ x & x & x & x & x & x & x & x & x & x & x \\ x & x & x & x & x & x & x & x & x & x & x \\ x & x & x & x & x & x & x & x & x & x & x \end{pmatrix}$$

$O(m)$ Schritte !!

Berechnung Zeilenmaxima von B

$$\begin{pmatrix} X & X & X & X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X & X & X & X \end{pmatrix}$$

$O(m)$ Schritte !!

Berechnung Zeilenmaxima von B

$$\begin{pmatrix} \color{blue}{x} & \color{blue}{x} & x & x & x & x & x & x & x & x & x \\ x & \color{red}{x} & x & x & x & x & x & x & x & x & x \\ x & x & x & x & x & x & x & x & x & x & x \\ x & x & x & x & x & \color{red}{x} & x & x & x & x & x \\ x & x & x & x & x & x & x & x & x & x & x \end{pmatrix}$$

$O(m)$ Schritte !!

Berechnung Zeilenmaxima von B

$$\begin{pmatrix} \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \end{pmatrix}$$

$O(m)$ Schritte !!

Berechnung Zeilenmaxima von B

$$\begin{pmatrix} \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \end{pmatrix}$$

$O(m)$ Schritte !!

Berechnung Zeilenmaxima von B

$$\begin{pmatrix} \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \\ \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} & \text{X} \end{pmatrix}$$

$O(m)$ Schritte !!

Gesamtbeispiel

Gesamtbeispiel

$$\begin{pmatrix} 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & 9 & 4 & 3 \\ 4 & 6 & 5 & 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Gesamtbeispiel

$$\begin{pmatrix} 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

Gesamtbeispiel

Zeilenreduktion:

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} 9 & 3 \\ 6 & 8 \end{pmatrix}$$

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} 9 & 3 \\ 6 & \Rightarrow 8 \end{pmatrix}$$

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} \Rightarrow 9 & 3 \\ 6 & \Rightarrow 8 \end{pmatrix}$$

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} \Rightarrow 9 & 3 \\ 6 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} \Rightarrow 9 & 3 \\ 6 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$(6 \ 8)$$

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} \Rightarrow 9 & 3 \\ 6 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$(6 \ 8)$$

Spaltenreduktion:

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} \Rightarrow 9 & 3 \\ 6 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 6 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} 8 \end{pmatrix}$$

Gesamtbeispiel

$$\begin{pmatrix} \Rightarrow 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 7 & 3 & \Rightarrow 9 & 4 & 3 \\ 4 & 6 & 5 & \Rightarrow 8 & 6 & 7 \\ 1 & 4 & 2 & 6 & 7 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$\begin{pmatrix} 1 & 7 & 3 & 9 & 4 & 3 \\ 1 & 4 & 2 & 6 & 7 & 8 \end{pmatrix}$$

Spaltenreduktion:

$$\begin{pmatrix} \Rightarrow 9 & 3 \\ 6 & \Rightarrow 8 \end{pmatrix}$$

Zeilenreduktion:

$$(6 \ 8)$$

Spaltenreduktion:

$$(\Rightarrow 8)$$

Theorem 1.20: Maxima für monotone Matrizen

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

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

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

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

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

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

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

$$T(m) \leq T\left(\frac{n}{2}\right) \text{ (Rek.)} + C \times m \text{ (Sp.Red.+ Rekonstr. + Ber.)}$$

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

$$\begin{aligned} T(m) &\leq T\left(\frac{n}{2}\right) \text{ (Rek.)} + C \times m \text{ (Sp.Red.+ Rekonstr. + Ber.)} \\ &\leq T\left(\frac{n}{4}\right) + C\left(m + \frac{n}{2}\right) \end{aligned}$$

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

$$\begin{aligned} T(m) &\leq T\left(\frac{n}{2}\right) \text{ (Rek.)} + C \times m \text{ (Sp.Red.+ Rekonstr. + Ber.)} \\ &\leq T\left(\frac{n}{4}\right) + C\left(m + \frac{n}{2}\right) \\ &\vdots \\ &\vdots \end{aligned}$$

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

$$\begin{aligned} T(m) &\leq T\left(\frac{n}{2}\right) \text{ (Rek.)} + C \times m \text{ (Sp.Red.+ Rekonstr. + Ber.)} \\ &\leq T\left(\frac{n}{4}\right) + C\left(m + \frac{n}{2}\right) \\ &\vdots \\ &\leq T(1) + C\left(m + n \sum_{i=1}^{\log n} \frac{1}{2^i}\right) \end{aligned}$$

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

$$\begin{aligned} T(m) &\leq T\left(\frac{n}{2}\right) \text{ (Rek.)} + C \times m \text{ (Sp.Red.+ Rekonstr. + Ber.)} \\ &\leq T\left(\frac{n}{4}\right) + C\left(m + \frac{n}{2}\right) \\ &\vdots \\ &\leq T(1) + C\left(m + n \sum_{i=1}^{\log n} \frac{1}{2^i}\right) \in O(m) \end{aligned}$$

Theorem 1.20: Maxima für monotone Matrizen

Laufzeitanalyse:

$$\begin{aligned} T(m) &\leq T\left(\frac{n}{2}\right) \text{ (Rek.)} + C \times m \text{ (Sp.Red.+ Rekonstr. + Ber.)} \\ &\leq T\left(\frac{n}{4}\right) + C\left(m + \frac{n}{2}\right) \\ &\vdots \\ &\leq T(1) + C\left(m + n \sum_{i=1}^{\log n} \frac{1}{2^i}\right) \in O(m) \end{aligned}$$

Nur $O(m)$ viele Vergleiche!!!

Theorem 1.21: Durchmesser eines Polygons

Theorem 1.21: Durchmesser eines Polygons

- $n \times n$ Matrix A (symbolisch)

Theorem 1.21: Durchmesser eines Polygones

- $n \times n$ Matrix A (symbolisch)
- Preprocessing Guibas/Hershberger $O(n)$

Theorem 1.21: Durchmesser eines Polygones

- $n \times n$ Matrix A (symbolisch)
- Preprocessing Guibas/Hershberger $O(n)$
- Zeilenmaxima von A mit $O(n)$ Vergleichen

Theorem 1.21: Durchmesser eines Polygons

- $n \times n$ Matrix A (symbolisch)
- Preprocessing Guibas/Hershberger $O(n)$
- Zeilenmaxima von A mit $O(n)$ Vergleichen
- Je Vergleich $O(\log n)$ Aufwand für Wert

Theorem 1.21: Durchmesser eines Polygones

- $n \times n$ Matrix A (symbolisch)
- Preprocessing Guibas/Hershberger $O(n)$
- Zeilenmaxima von A mit $O(n)$ Vergleichen
- Je Vergleich $O(\log n)$ Aufwand für Wert
- Gesamtlaufzeit: $O(n \log n)$

1.2.4 Shortest Watchman Routes

1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem,

1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem, Startpunkt s

1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem, Startpunkt s
- Route: Einmal alles *sehen* und zurück

1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem, Startpunkt s
- Route: Einmal alles *sehen* und zurück
- Tour: Einmal alles *sehen* (offen)

1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem, Startpunkt s
- Route: Einmal alles *sehen* und zurück
- Tour: Einmal alles *sehen* (offen)
- Polygonale Szene: NP-hart **Theorem 1.22**

1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem, Startpunkt s
- Route: Einmal alles *sehen* und zurück
- Tour: Einmal alles *sehen* (offen)
- Polygonale Szene: NP-hart **Theorem 1.22**
- Beweis: TSP reduzieren auf SWR

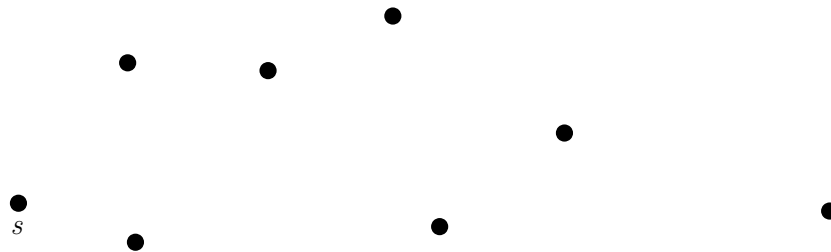
1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem, Startpunkt s
- Route: Einmal alles *sehen* und zurück
- Tour: Einmal alles *sehen* (offen)
- Polygonale Szene: NP-hart **Theorem 1.22**
- Beweis: TSP reduzieren auf SWR

●
 s

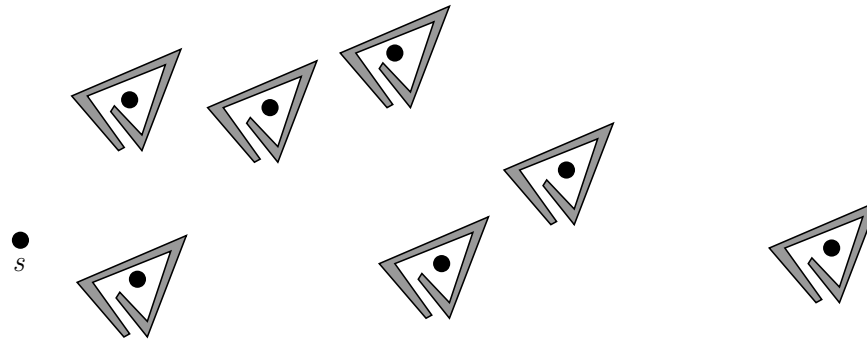
1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem, Startpunkt s
- Route: Einmal alles *sehen* und zurück
- Tour: Einmal alles *sehen* (offen)
- Polygonale Szene: NP-hart **Theorem 1.22**
- Beweis: TSP reduzieren auf SWR



1.2.4 Shortest Watchman Routes

- Agent mit Sichtsystem, Startpunkt s
- Route: Einmal alles *sehen* und zurück
- Tour: Einmal alles *sehen* (offen)
- Polygonale Szene: NP-hart **Theorem 1.22**
- Beweis: TSP reduzieren auf SWR



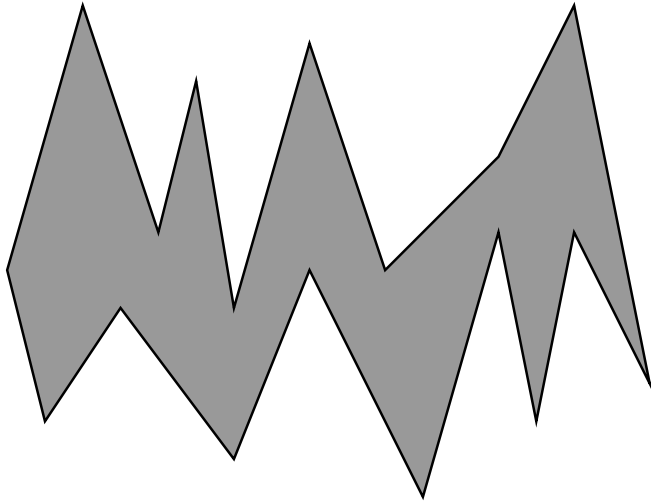
Polynomiell innerhalb einfacher Polygonen

Polynomiell innerhalb einfacher Polygonen

- Standardmittel (noch einfacher): Def. 1.23

Polynomiell innerhalb einfacher Polygone

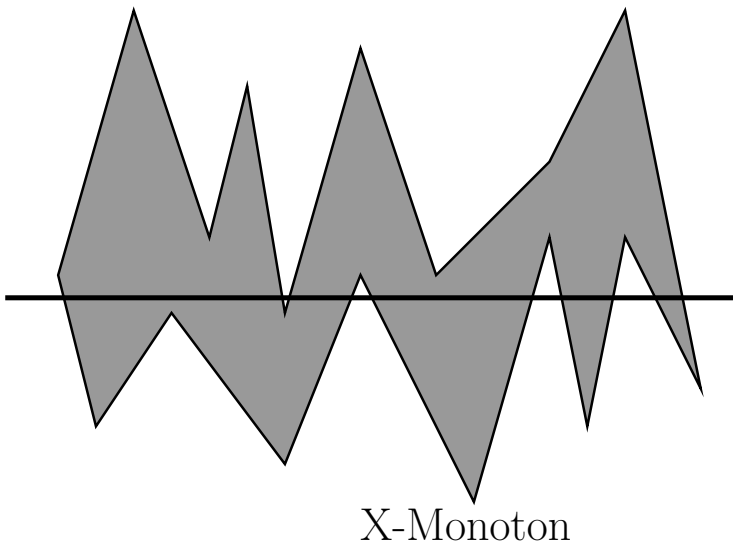
- Standardmittel (noch einfacher): **Def. 1.23**
- Monotone Polygone: Monoton bezüglich l



X-Monoton

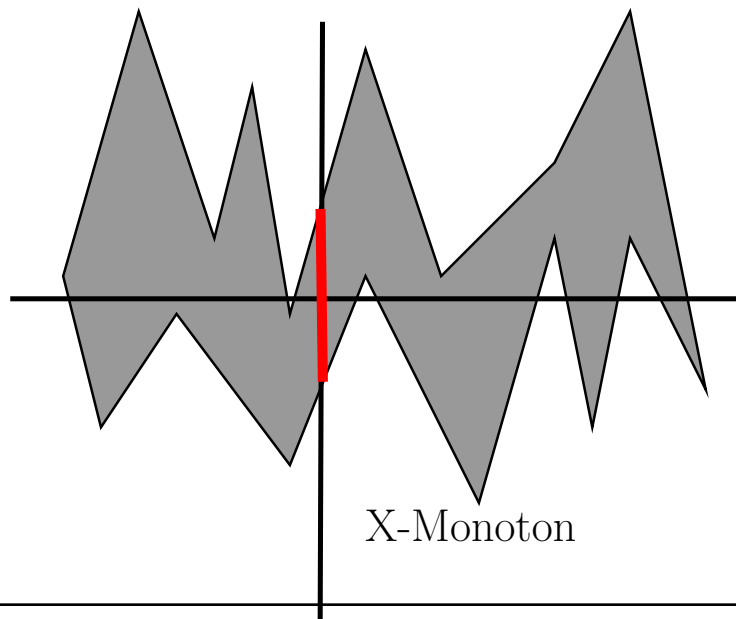
Polynomiell innerhalb einfacher Polygone

- Standardmittel (noch einfacher): **Def. 1.23**
- Monotone Polygone: Monoton bezüglich l



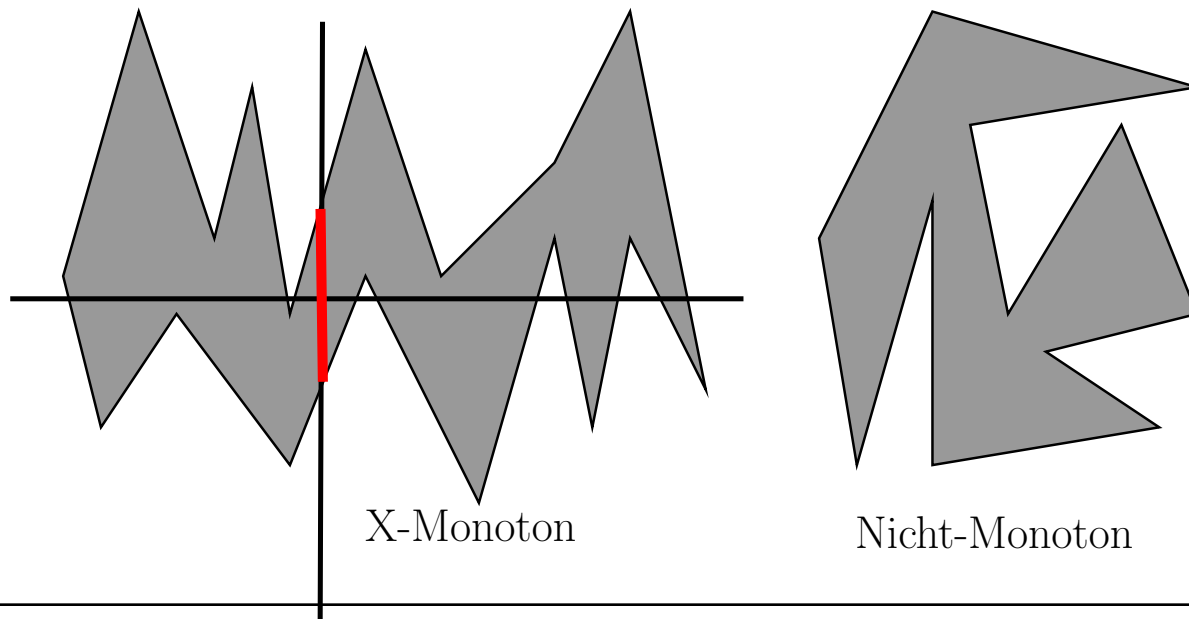
Polynomiell innerhalb einfacher Polygone

- Standardmittel (noch einfacher): **Def. 1.23**
- Monotone Polygone: Monoton bezüglich l



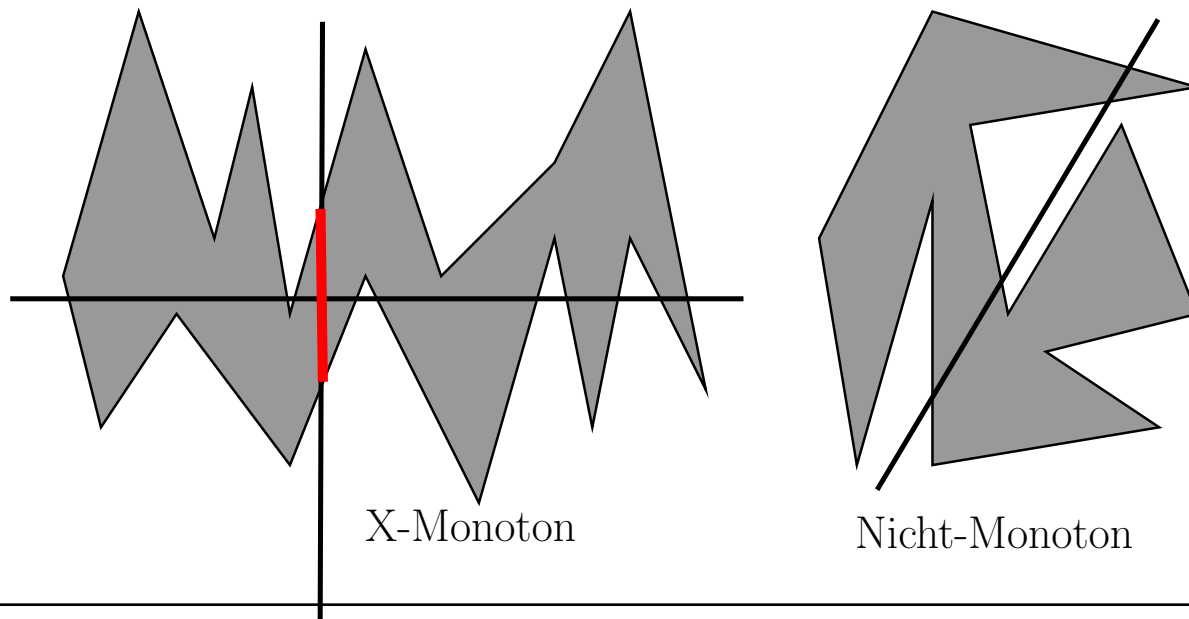
Polynomiell innerhalb einfacher Polygonen

- Standardmittel (noch einfacher): **Def. 1.23**
- Monotone Polygone: Monoton bezüglich l



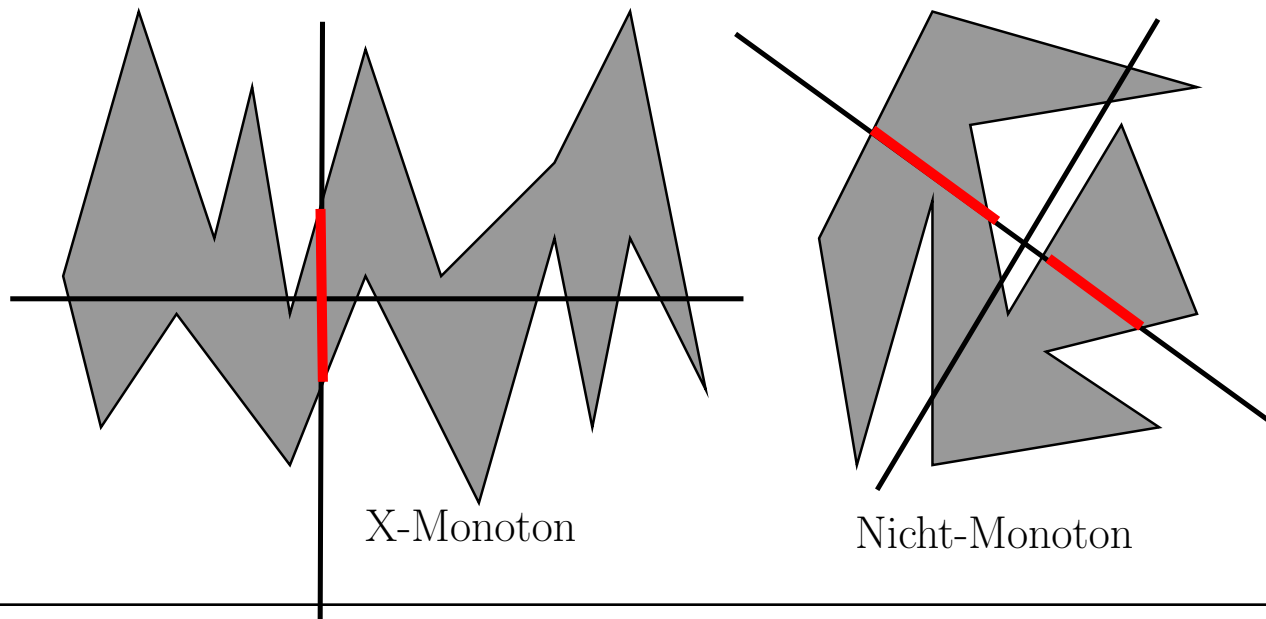
Polynomiell innerhalb einfacher Polygonen

- Standardmittel (noch einfacher): **Def. 1.23**
- Monotone Polygone: Monoton bezüglich l



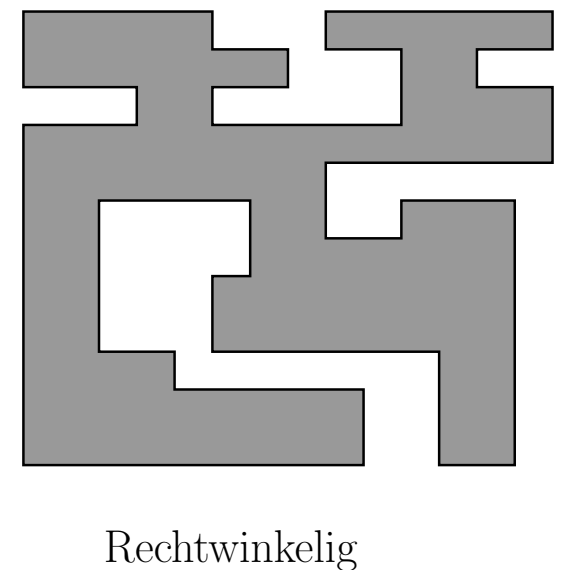
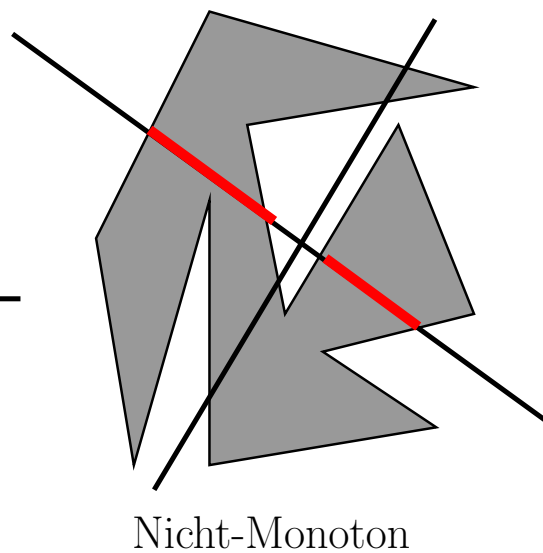
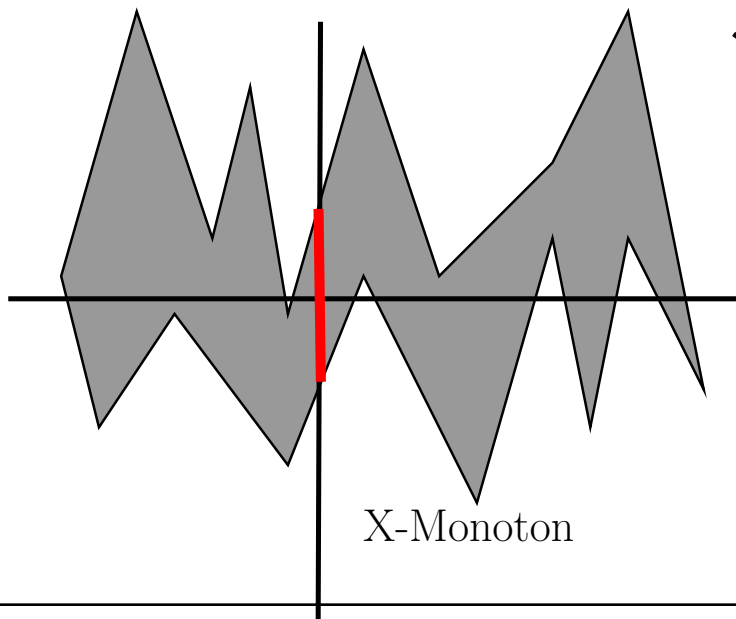
Polynomiell innerhalb einfacher Polygonen

- Standardmittel (noch einfacher): **Def. 1.23**
- Monotone Polygone: Monoton bezüglich l



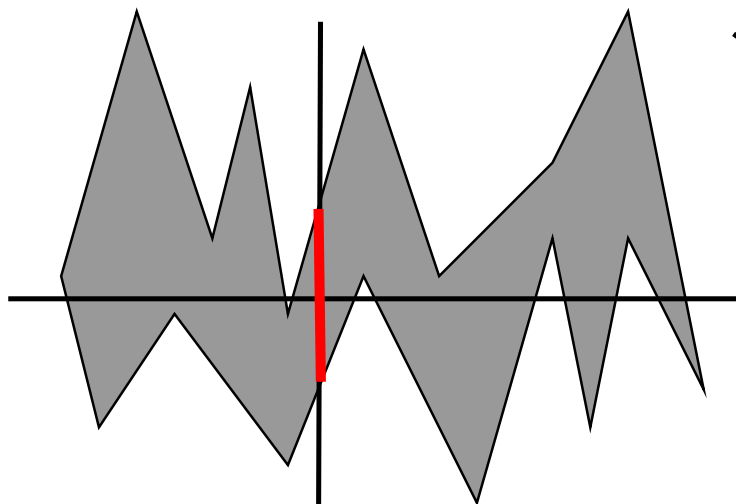
Polynomiell innerhalb einfacher Polygone

- Standardmittel (noch einfacher): **Def. 1.23**
- Monotone Polygone: Monoton bezüglich l
- Rechtwinkelige Polygone

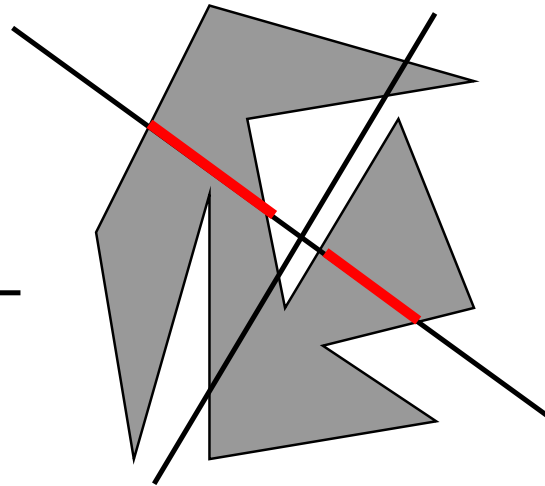


Polynomiell innerhalb einfacher Polygone

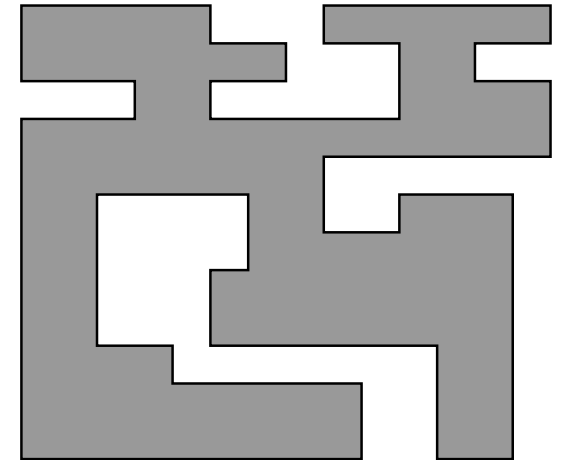
- Standardmittel (noch einfacher): **Def. 1.23**
- Monotone Polygone: Monoton bezüglich l
- Rechtwinkelige Polygone
- Rechtwinkelige und monotone Polygone: SWR in $O(n)$ **Theorem 1.24**



X-Monoton



Nicht-Monoton

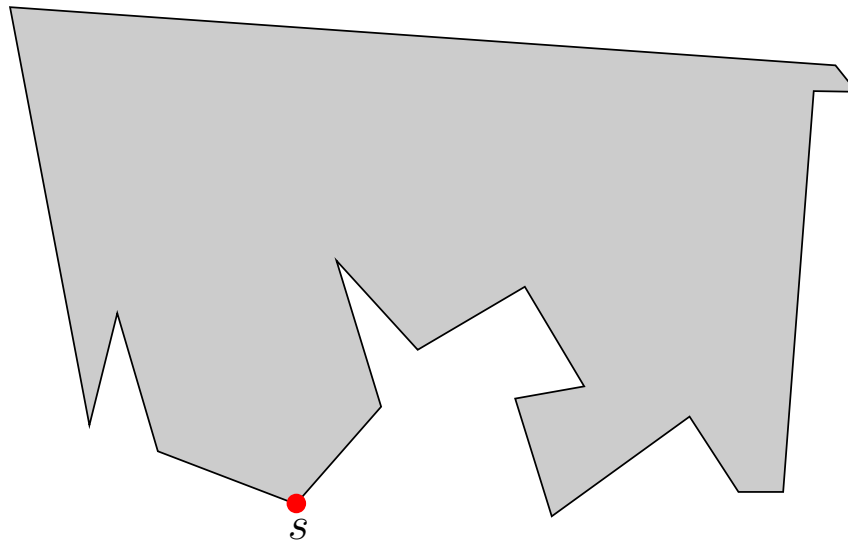


Rechtwinkelig

Was ist wichtig? Def. 1.25

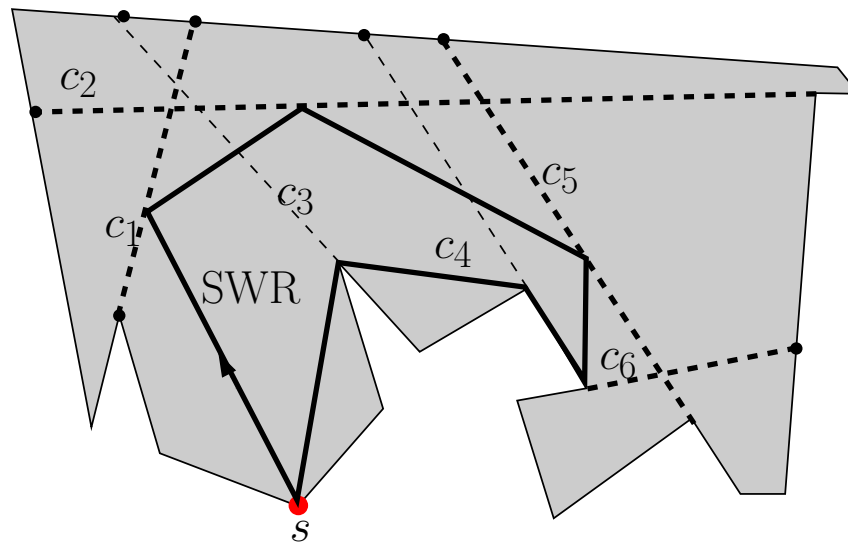
Was ist wichtig? Def. 1.25

a) (Cuts) Extensionen der reflexen Ecken



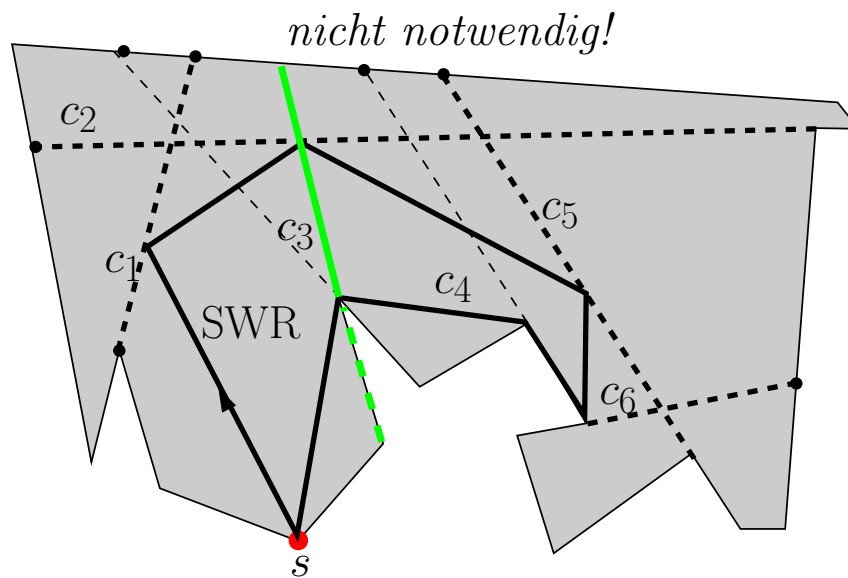
Was ist wichtig? Def. 1.25

- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)



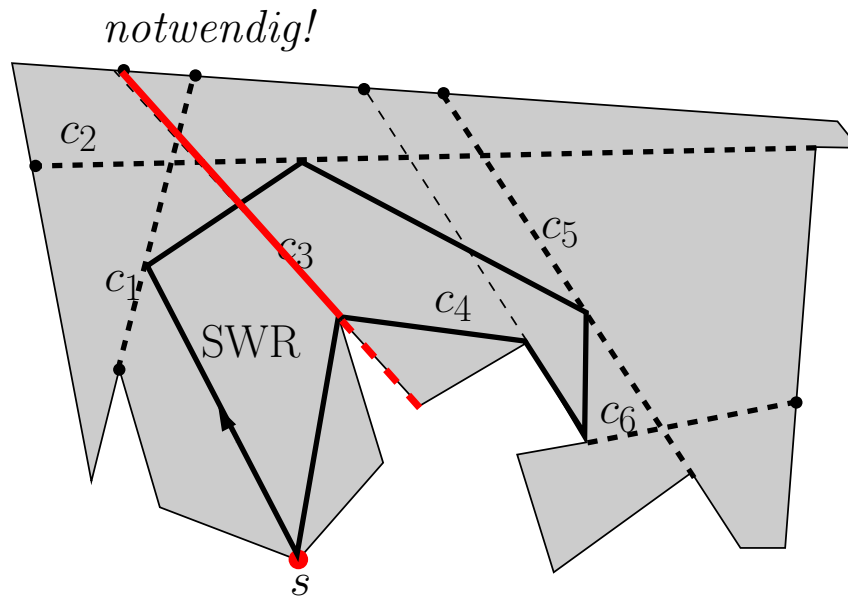
Was ist wichtig? Def. 1.25

- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)



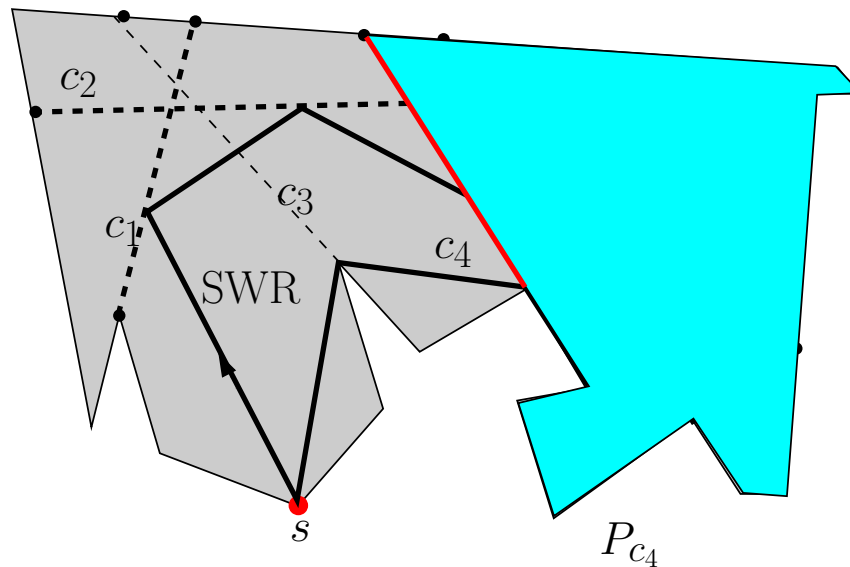
Was ist wichtig? Def. 1.25

- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)



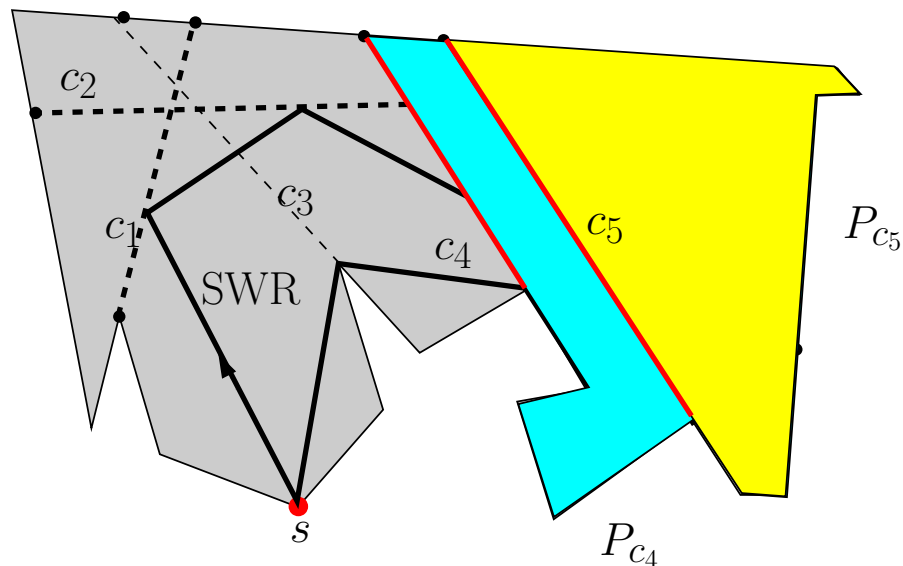
Was ist wichtig? Def. 1.25

- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)
- c) Dominanz-Beziehung $P_{c_i} \subseteq P_{c_j}$



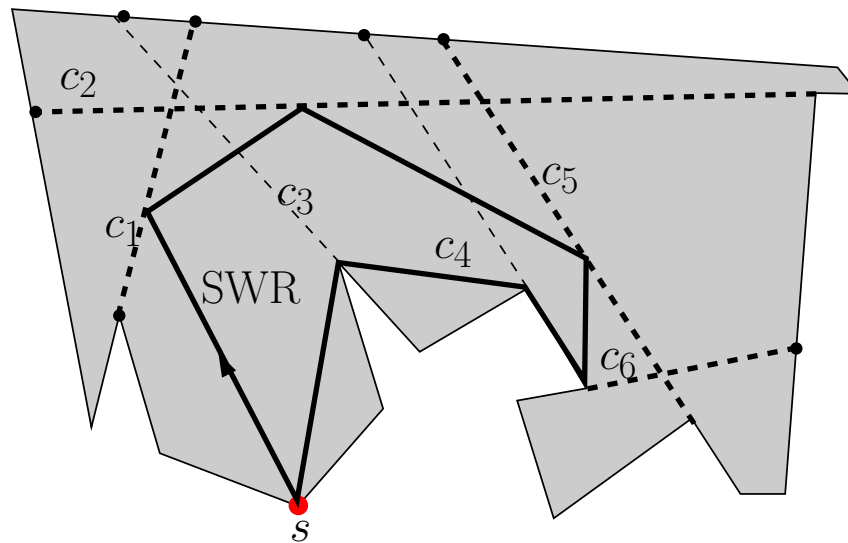
Was ist wichtig? Def. 1.25

- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)
- c) Dominanz-Beziehung $P_{c_i} \subseteq P_{c_j}$



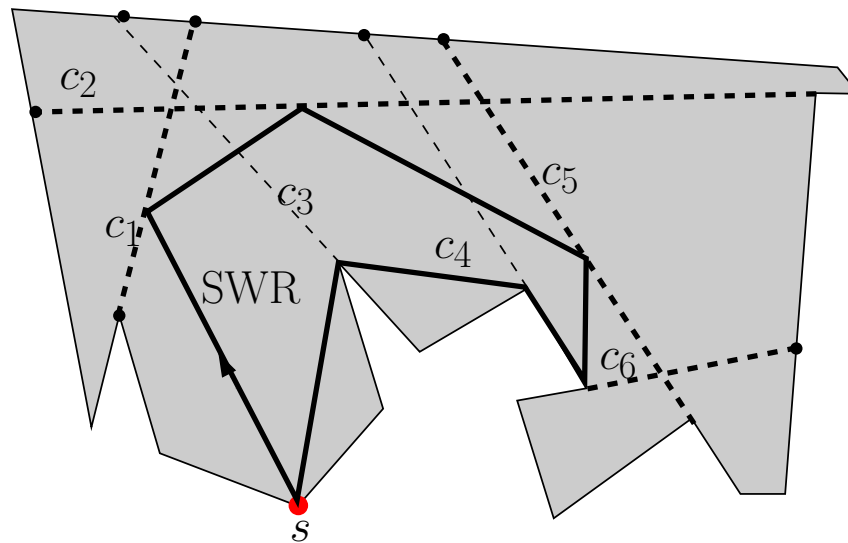
Was ist wichtig? Def. 1.25

- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)
- c) Dominanz-Beziehung $P_{c_i} \subseteq P_{c_j}$



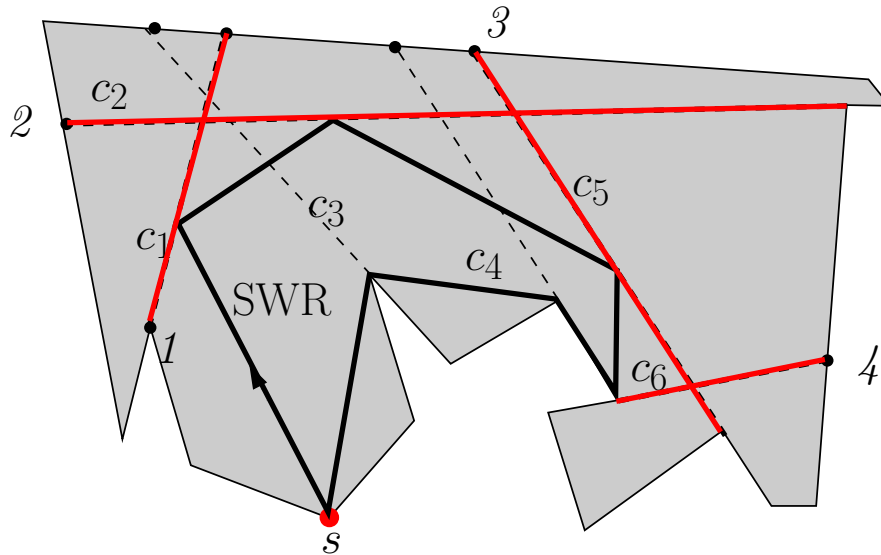
Was ist wichtig? Def. 1.25

- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)
- c) Dominanz-Beziehung $P_{c_i} \subseteq P_{c_j}$
- d) Wesentliche Cuts



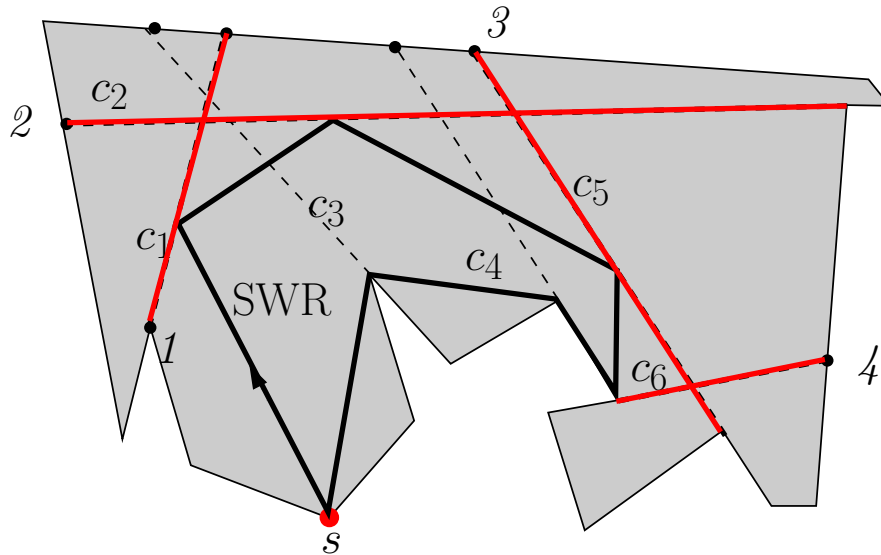
Was ist wichtig? Def. 1.25

- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)
- c) Dominanz-Beziehung $P_{c_i} \subseteq P_{c_j}$
- d) Wesentliche Cuts
- e) Ordnung der wesentlichen Cuts



Was ist wichtig? Def. 1.25

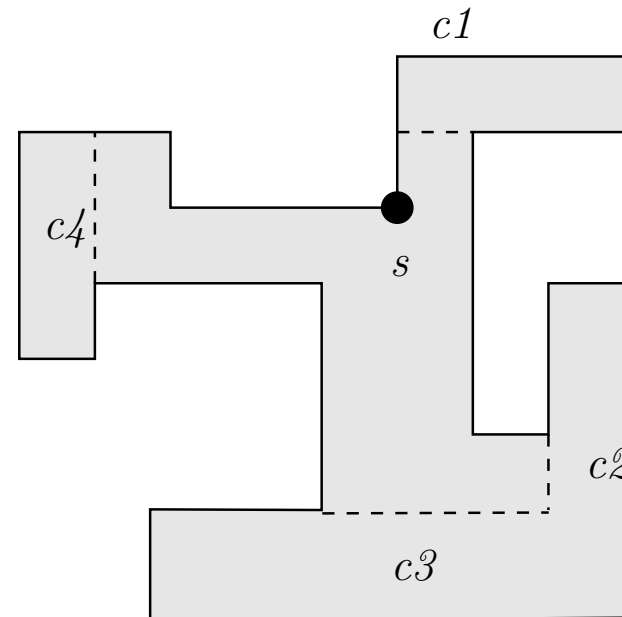
- a) (Cuts) Extensionen der reflexen Ecken
- b) Notwendige Cuts (bezüglich s)
- c) Dominanz-Beziehung $P_{c_i} \subseteq P_{c_j}$
- d) Wesentliche Cuts
- e) Ordnung der wesentlichen Cuts



Lemma 1.26 Ordnung entlang des Randes

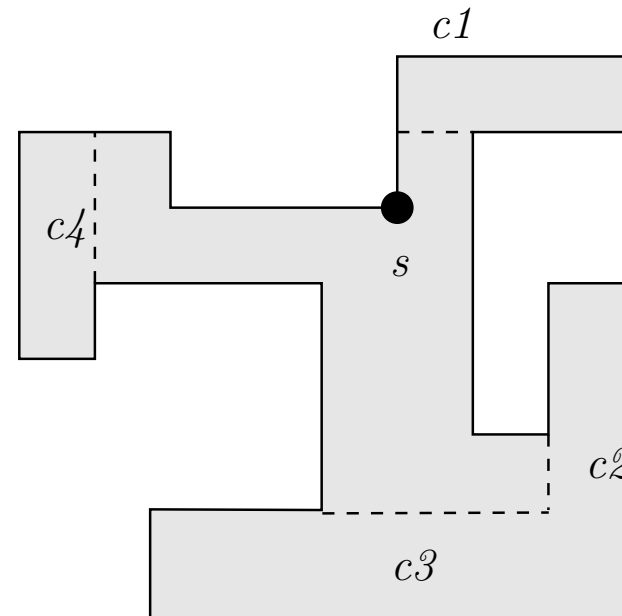
Lemma 1.26 Ordnung entlang des Randes

- Rechtwinkeliges Polygon



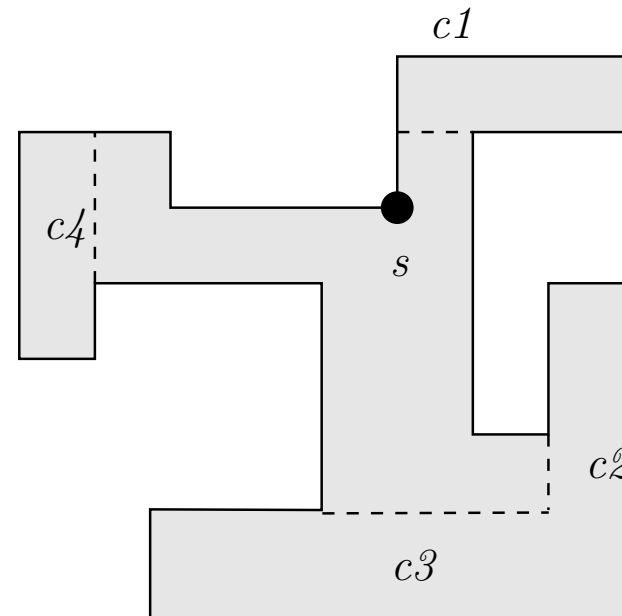
Lemma 1.26 Ordnung entlang des Randes

- Rechtwinkeliges Polygon
- Wesentliche Cuts werden nie überschritten!



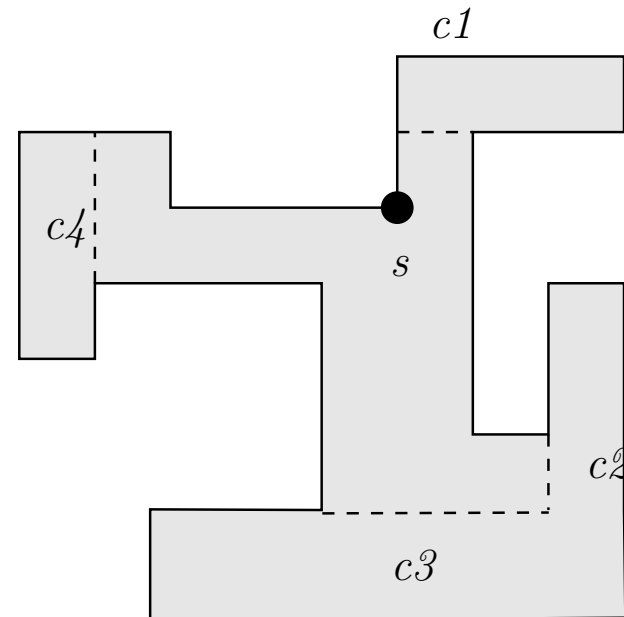
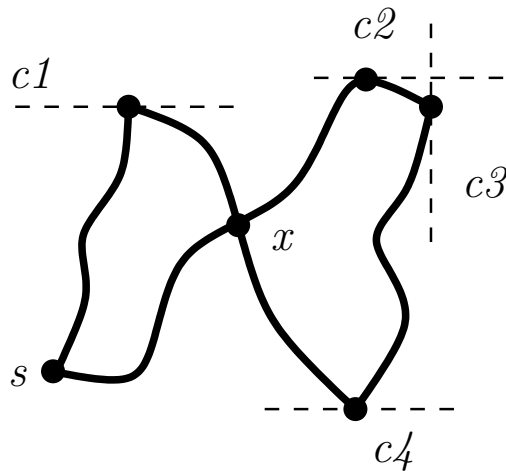
Lemma 1.26 Ordnung entlang des Randes

- Rechtwinkeliges Polygon
- Wesentliche Cuts werden nie überschritten!
- SWR besucht Cuts gemäß Ordnung



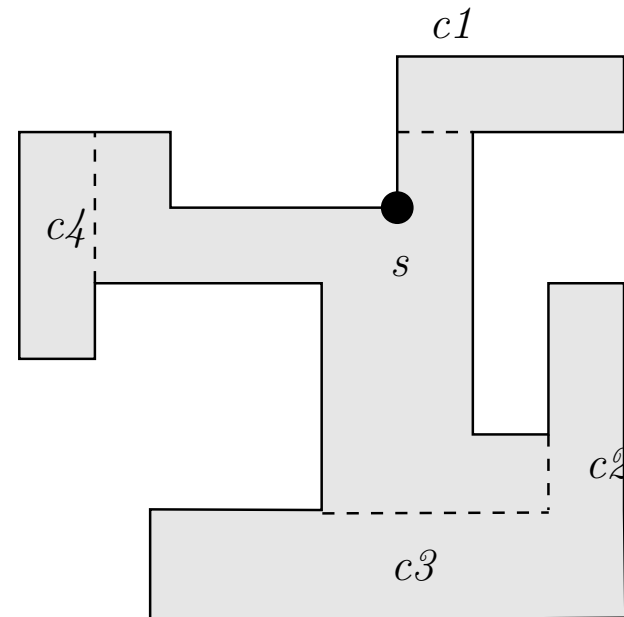
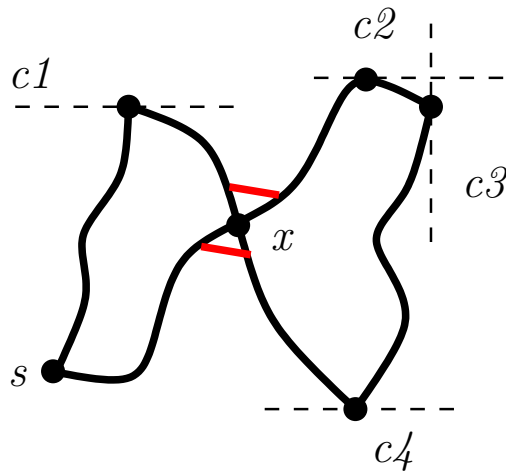
Lemma 1.26 Ordnung entlang des Randes

- Rechtwinkeliges Polygon
- Wesentliche Cuts werden nie überschritten!
- SWR besucht Cuts gemäß Ordnung
- Widerspruch! Abkürzung!



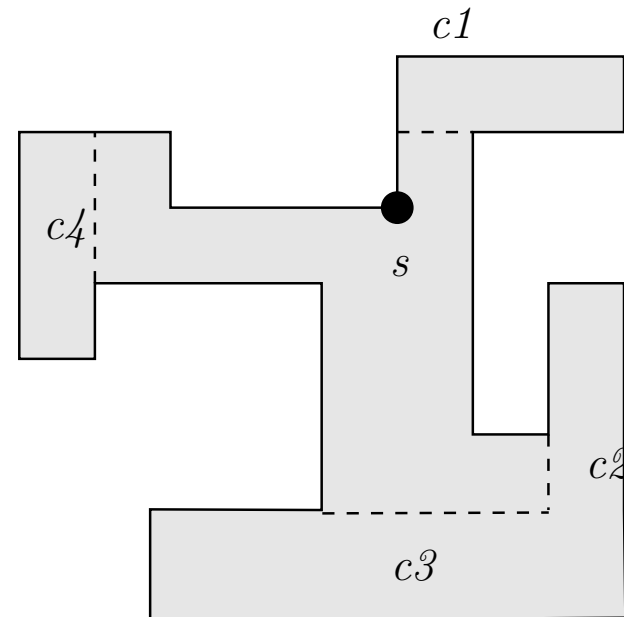
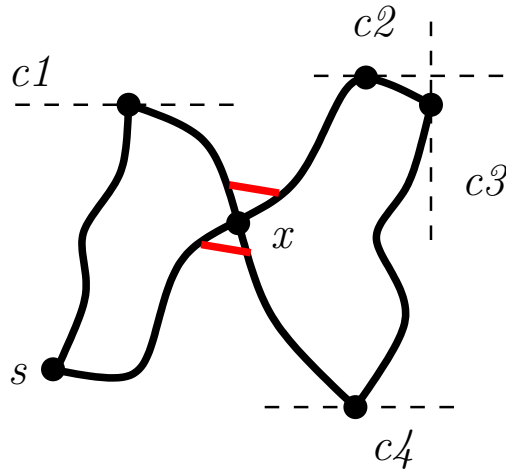
Lemma 1.26 Ordnung entlang des Randes

- Rechtwinkeliges Polygon
- Wesentliche Cuts werden nie überschritten!
- SWR besucht Cuts gemäß Ordnung
- Widerspruch! Abkürzung!



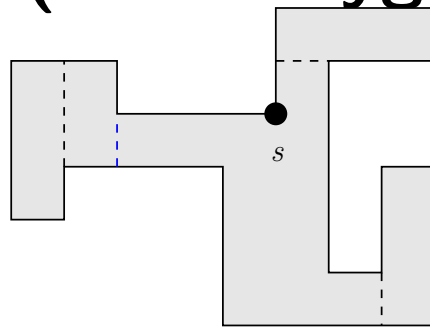
Lemma 1.26 Ordnung entlang des Randes

- Rechtwinkeliges Polygon
- Wesentliche Cuts werden nie überschritten!
- SWR besucht Cuts gemäß Ordnung
- Widerspruch! Abkürzung!
- $O(n)$ Algorithmus!!

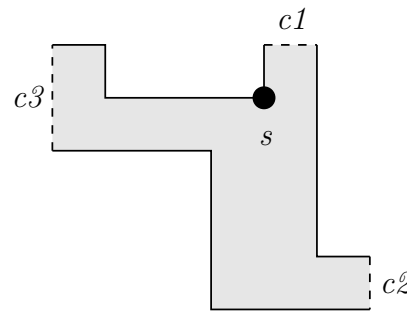


SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$

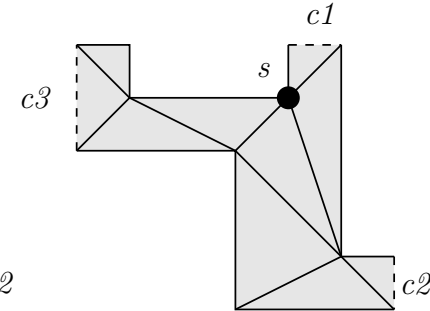
SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$



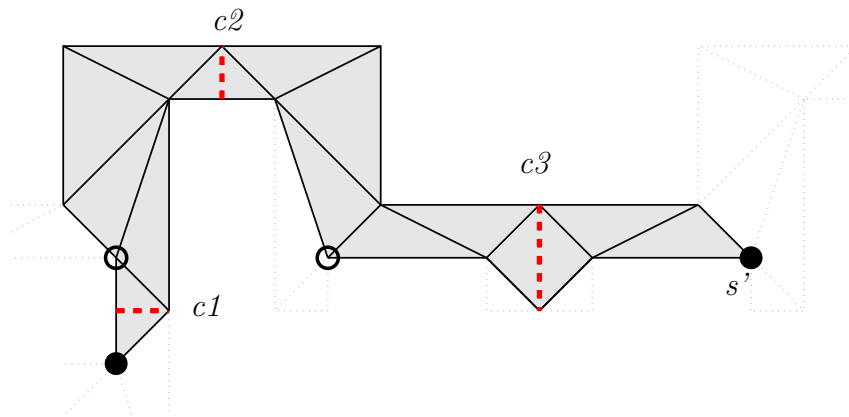
(i) Wesentliche Cuts



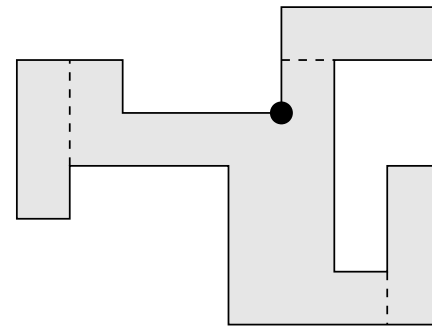
(ii) Abschneiden!



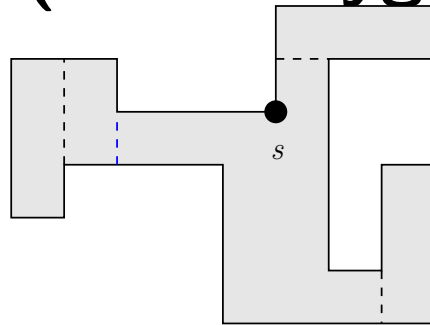
(iii) Triangulation



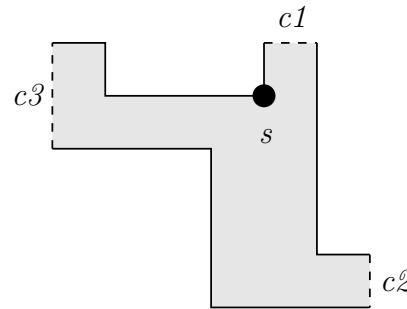
(iv) Spiegeln und Ausrollen!!



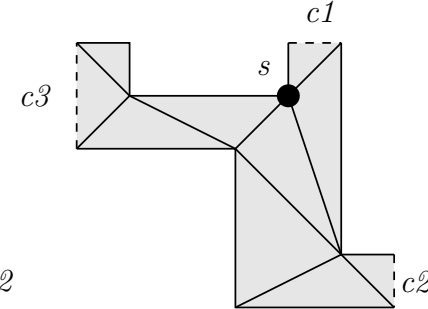
SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$



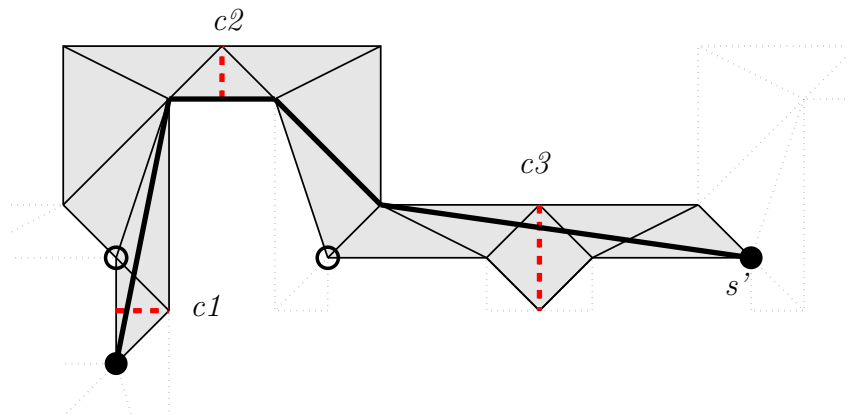
(i) Wesentliche Cuts



(ii) Abschneiden!

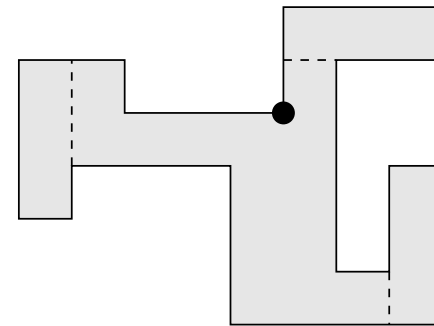


(iii) Triangulation

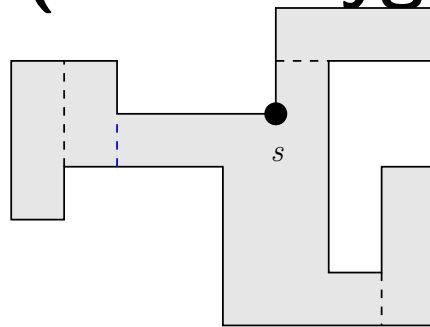


(iv) Spiegeln und Ausrollen!!

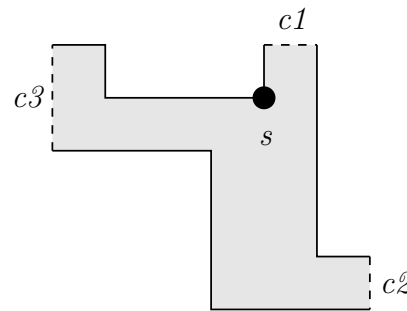
(v) Weg berechnen



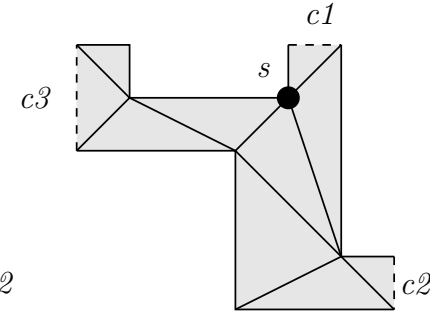
SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$



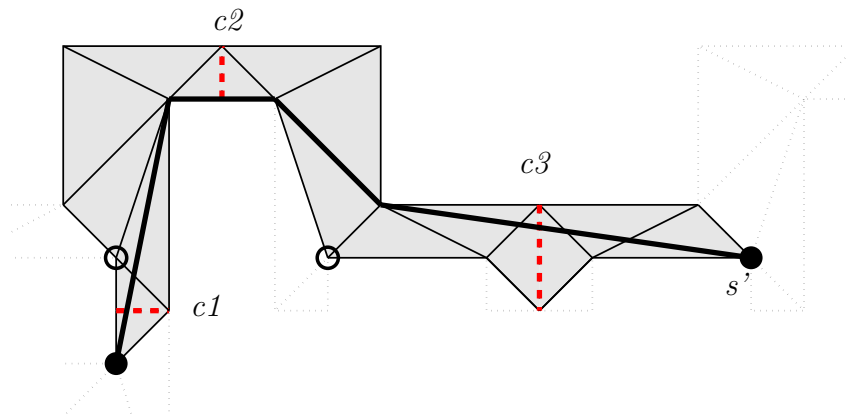
(i) Wesentliche Cuts



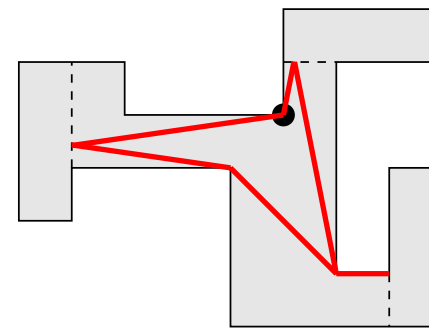
(ii) Abschneiden!



(iii) Triangulation

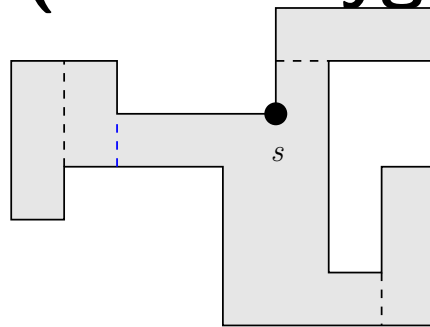


(iv) Spiegeln und Ausrollen!!
(v) Weg berechnen

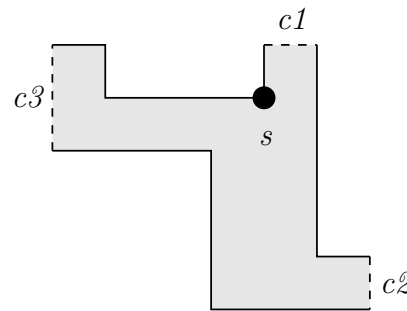


(vi) Zusammenklappen!

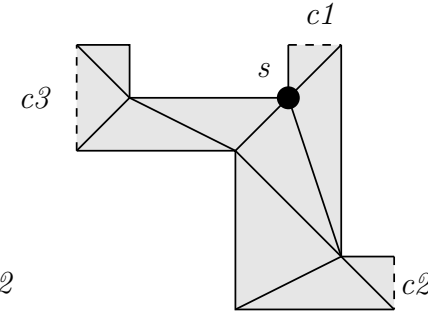
SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$



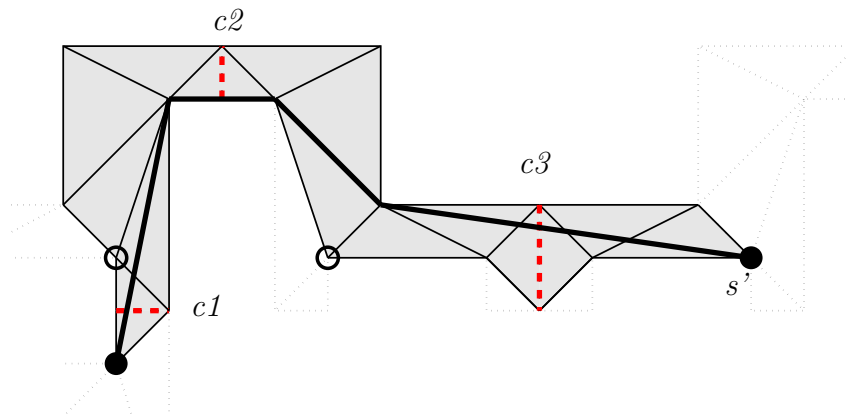
(i) Wesentliche Cuts
 $O(n)$



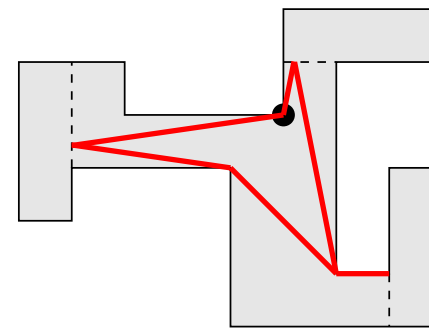
(ii) Abschneiden!



(iii) Triangulation

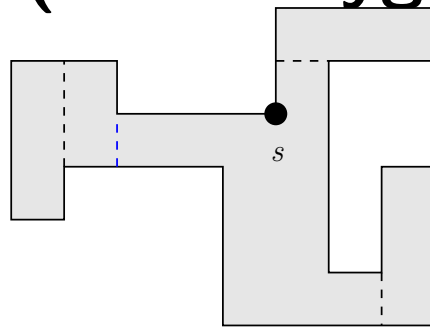


(iv) Spiegeln und Ausrollen!!
(v) Weg berechnen

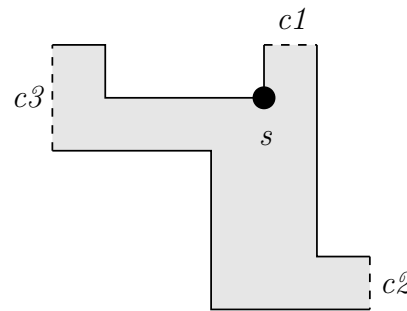


(vi) Zusammenklappen!

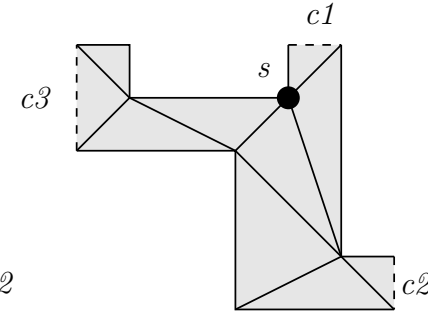
SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$



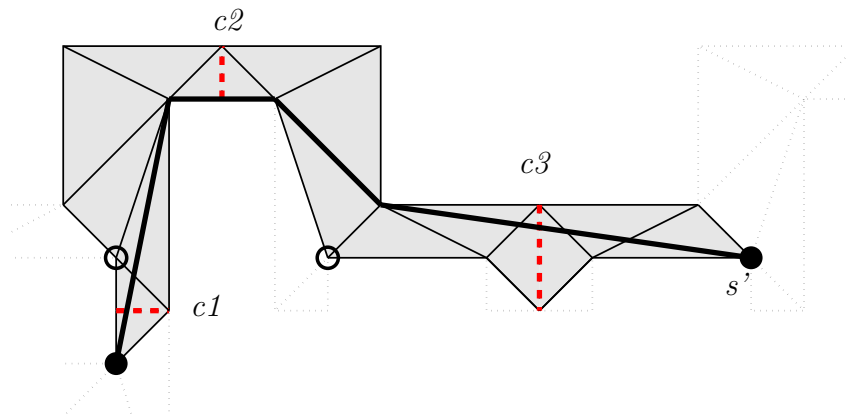
(i) Wesentliche Cuts
 $O(n)$



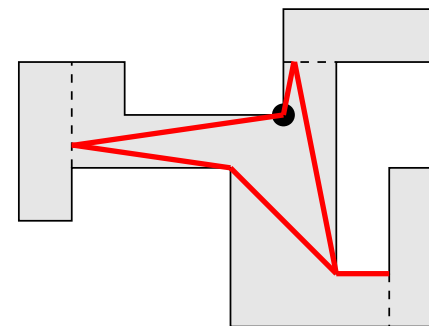
(ii) Abschneiden!
 $O(n)$



(iii) Triangulation

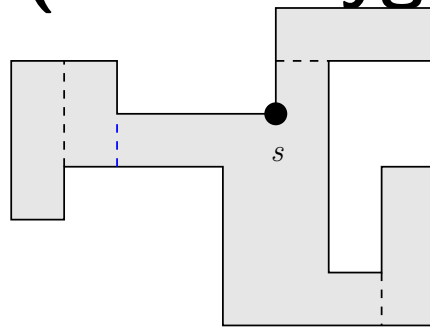


(iv) Spiegeln und Ausrollen!!
(v) Weg berechnen

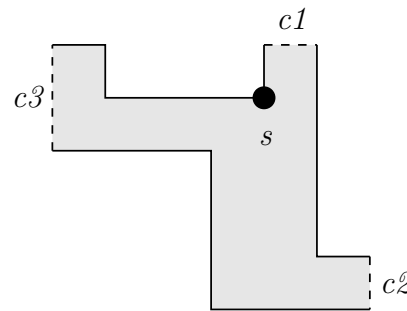


(vi) Zusammenklappen!

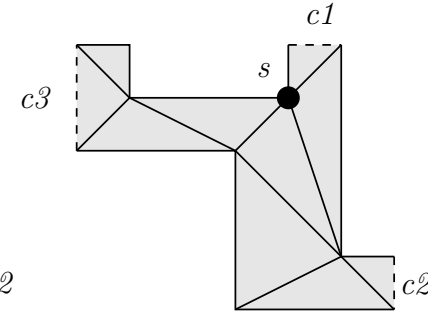
SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$



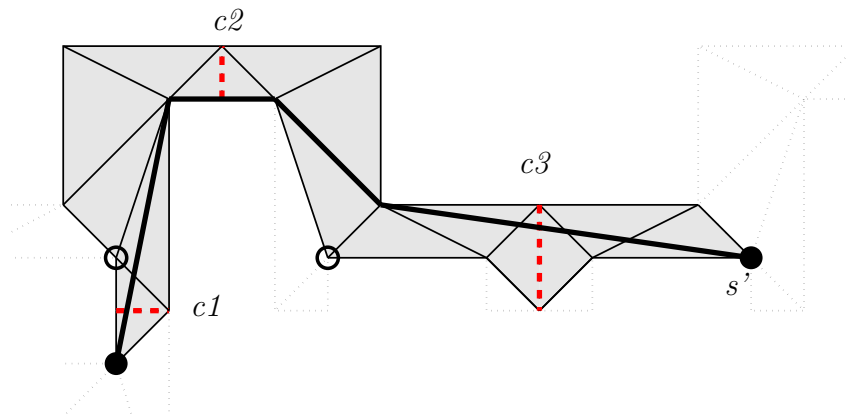
(i) Wesentliche Cuts
 $O(n)$



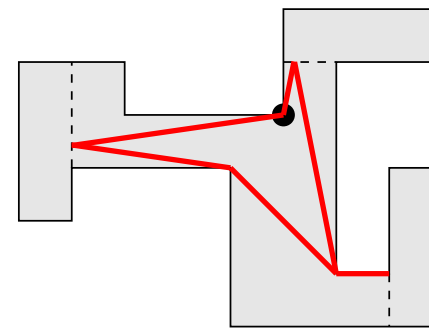
(ii) Abschneiden!
 $O(n)$



(iii) Triangulation
 $O(n)$

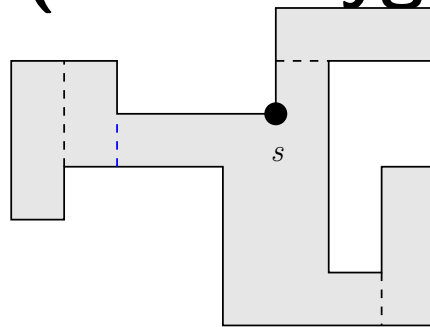


(iv) Spiegeln und Ausrollen!!
(v) Weg berechnen

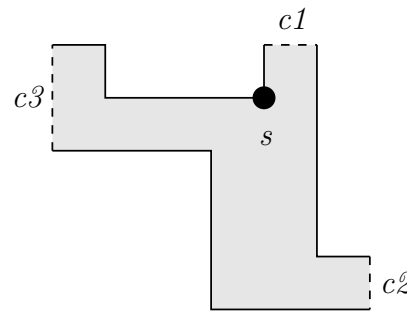


(vi) Zusammenklappen!

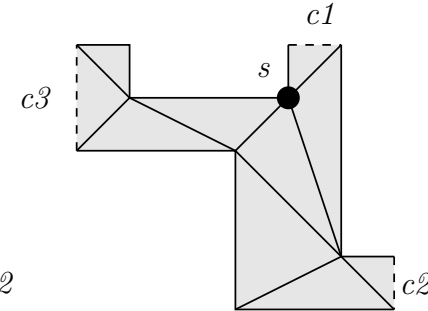
SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$



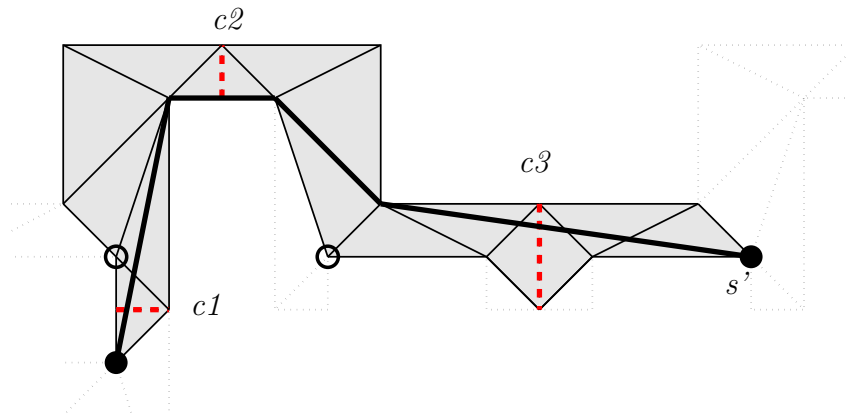
(i) Wesentliche Cuts
 $O(n)$



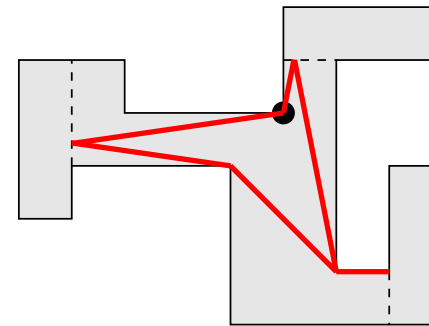
(ii) Abschneiden!
 $O(n)$



(iii) Triangulation
 $O(n)$

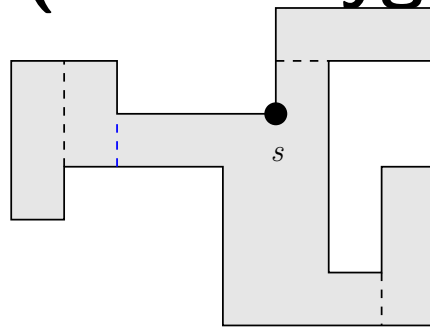


(iv) Spiegeln und Ausrollen!!
(v) Weg berechnen
 $O(n)$

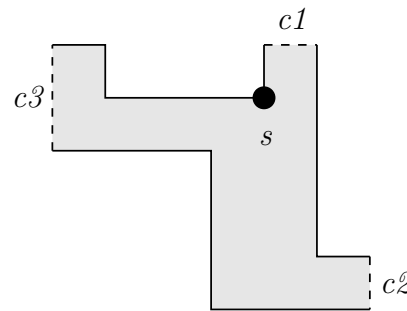


(vi) Zusammenklappen!

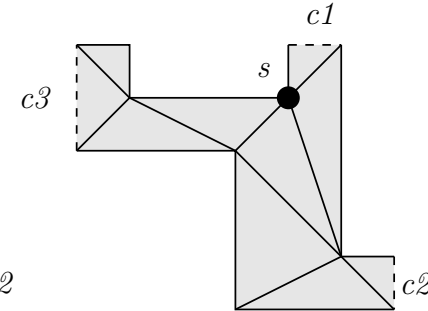
SWR (RW Polygon) Alg. 1.9/Th. 1.27: $O(n)$



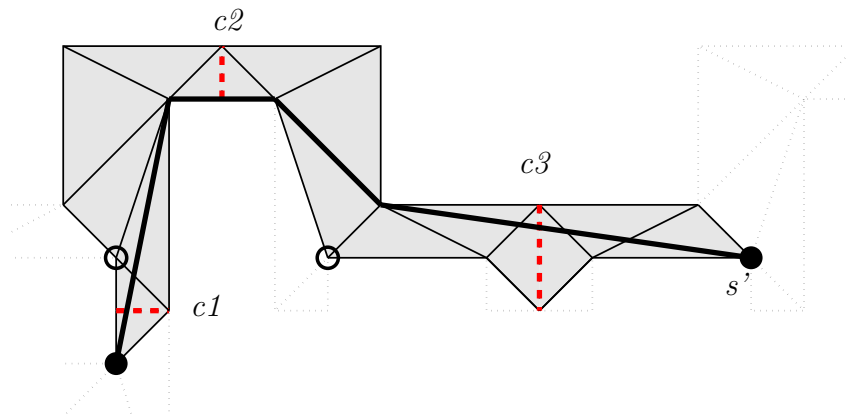
(i) Wesentliche Cuts
 $O(n)$



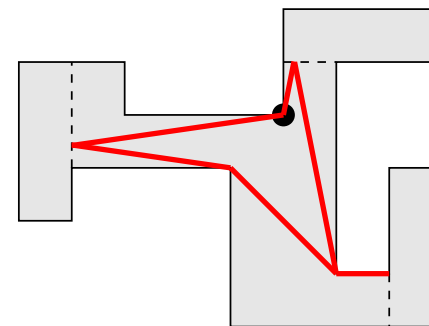
(ii) Abschneiden!
 $O(n)$



(iii) Triangulation
 $O(n)$



(iv) Spiegeln und Ausrollen!!
(v) Weg berechnen
 $O(n)$

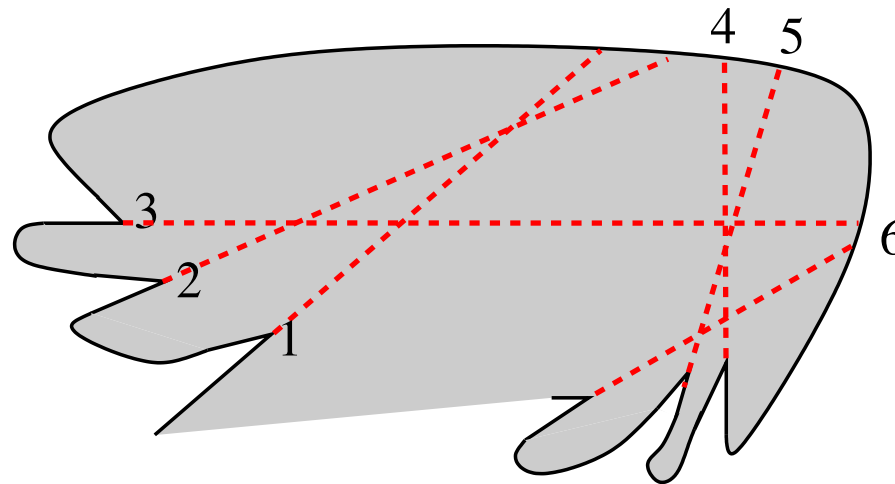


(vi) Zusammenklappen!
 $O(n)$

SWR (Allgemeiner Fall)

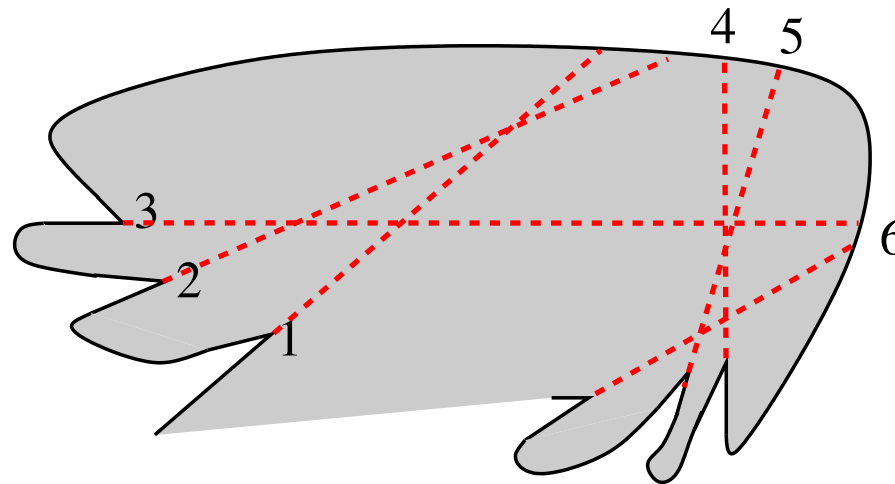
SWR (Allgemeiner Fall)

- Corner Problem!!



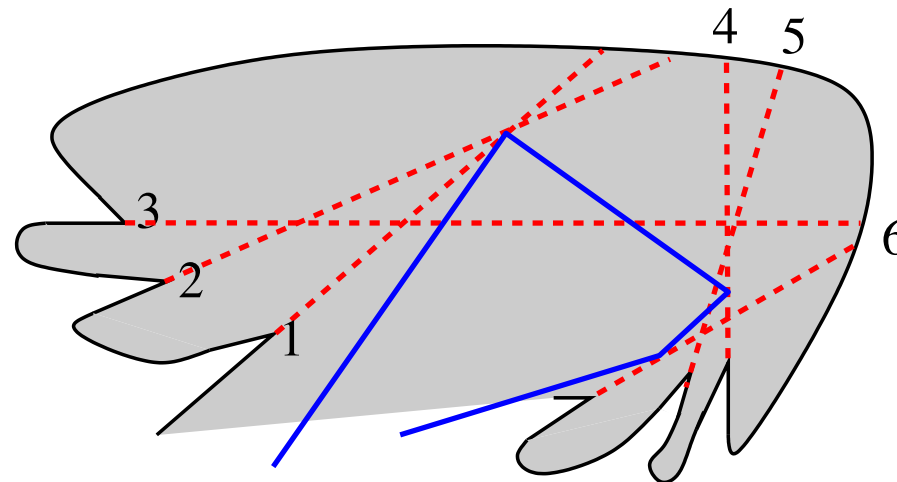
SWR (Allgemeiner Fall)

- Corner Problem!!
- **Def. 1.28** Folge wes. Cuts, die sich sukzessive schneiden



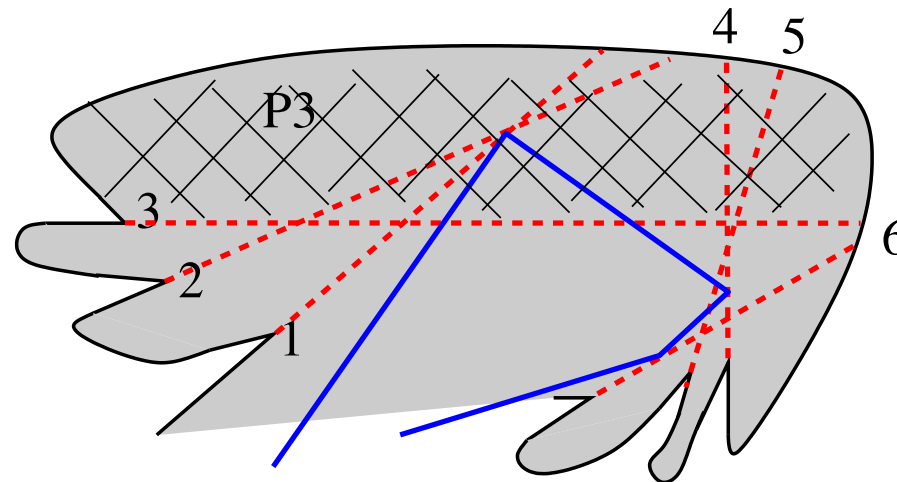
SWR (Allgemeiner Fall)

- Corner Problem!!
- **Def. 1.28** Folge wes. Cuts, die sich sukzessive schneiden
- Müssen nicht gemäß Ordnung besucht werden



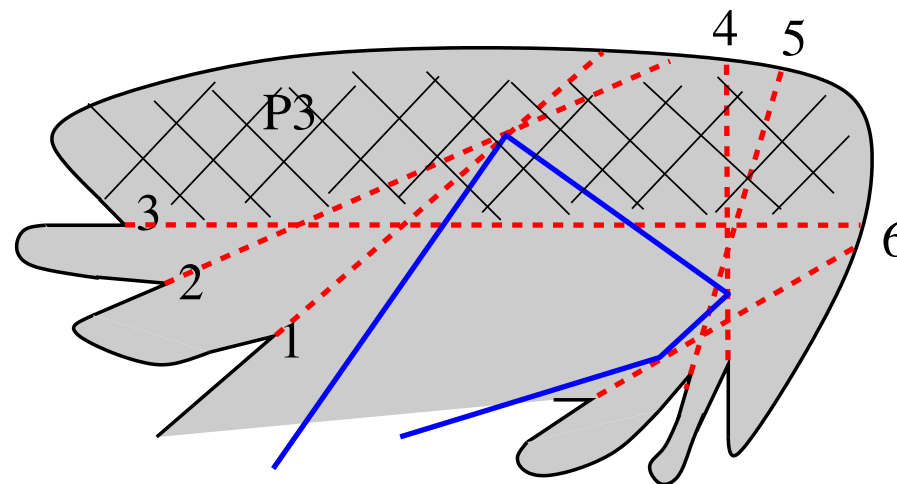
SWR (Allgemeiner Fall)

- Corner Problem!!
- **Def. 1.28** Folge wes. Cuts, die sich sukzessive schneiden
- Müssen nicht gemäß Ordnung besucht werden
- Aber die zugehörigen P_{c_i} !!!



SWR (Allgemeiner Fall)

- Corner Problem!!
- **Def. 1.28** Folge wes. Cuts, die sich sukzessive schneiden
- Müssen nicht gemäß Ordnung besucht werden
- Aber die zugehörigen P_{c_i} !!!



Lemma 1.29

Lemma 1.29

Die SWR besucht die Corner gemäß der Ordnung entlang des Randes!

Lemma 1.29

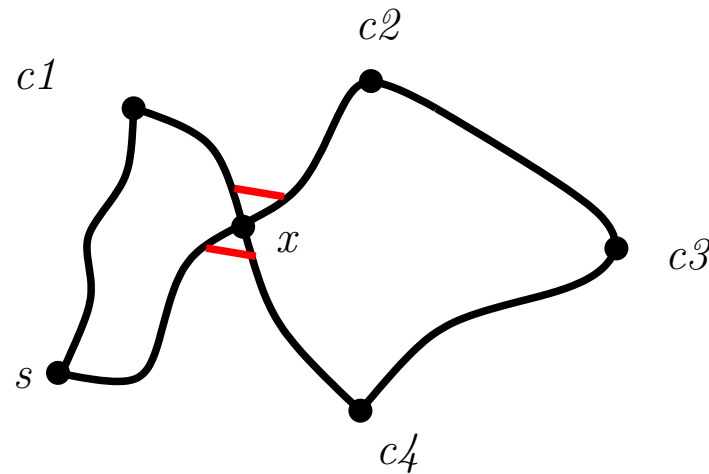
Die SWR besucht die Corner gemäß der Ordnung entlang des Randes!

Beweis: Genauso wie Lemma 1.26!

Lemma 1.29

Die SWR besucht die Corner gemäß der Ordnung entlang des Randes!

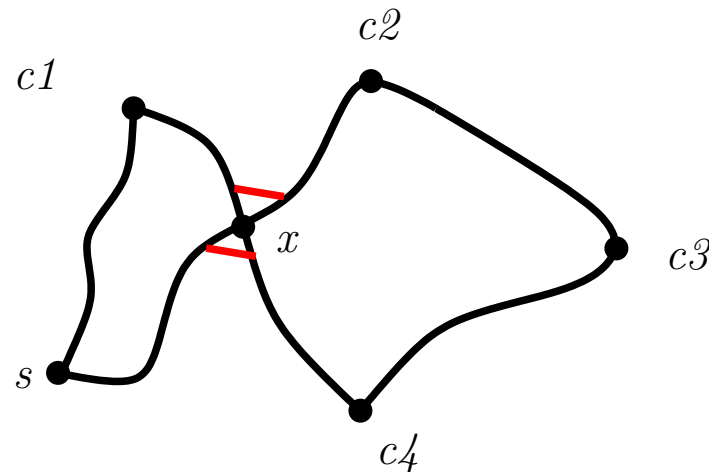
Beweis: Genauso wie Lemma 1.26! Anpassungen im Corner!



Lemma 1.29

Die SWR besucht die Corner gemäß der Ordnung entlang des Randes!

Beweis: Genauso wie Lemma 1.26! Anpassungen im Corner!



Anpassungen innerhalb der Corner: Fehleranfällig!!

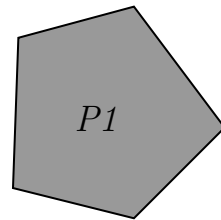
Touring a sequence of polygons (TPP)

Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone

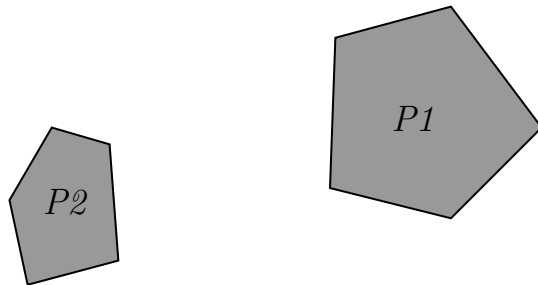
Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone



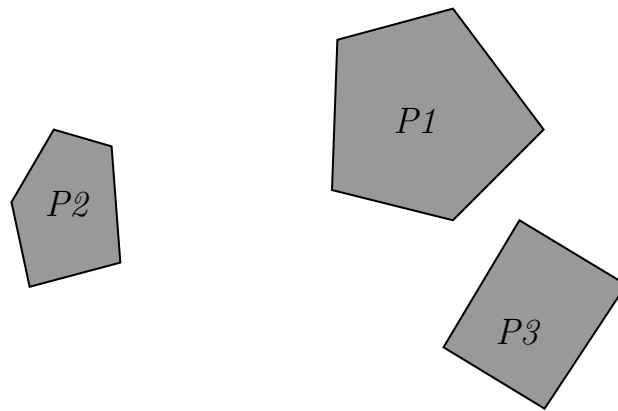
Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone



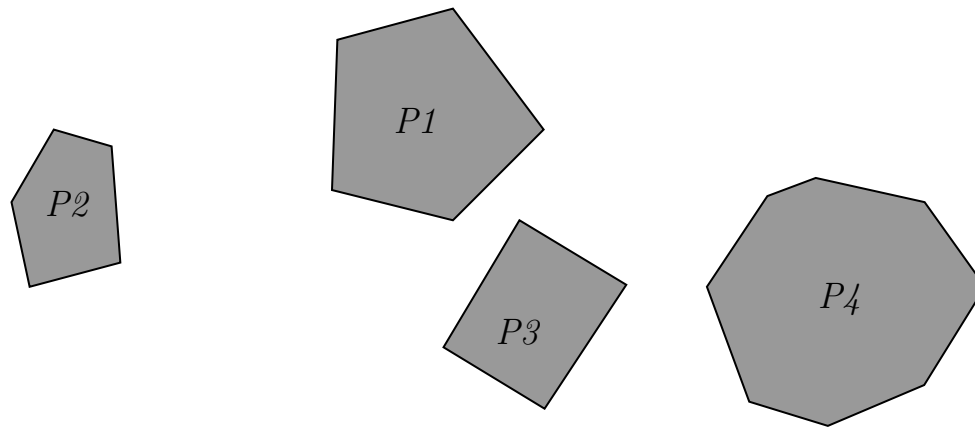
Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone



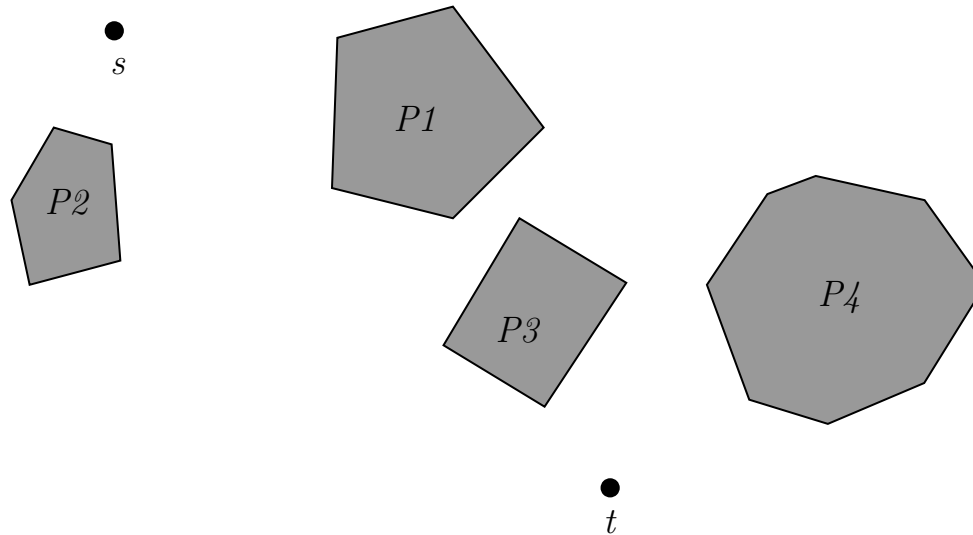
Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone



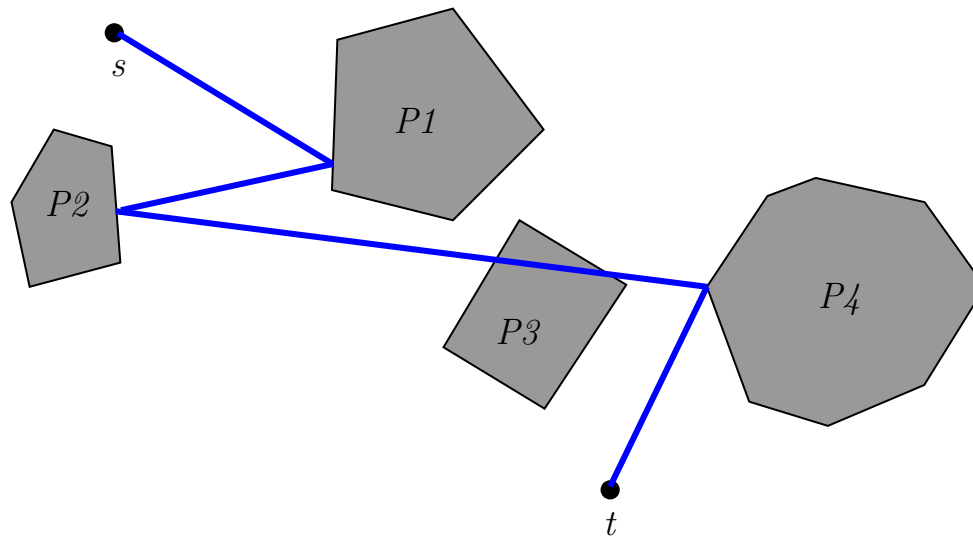
Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone
- Start s und Ziel t



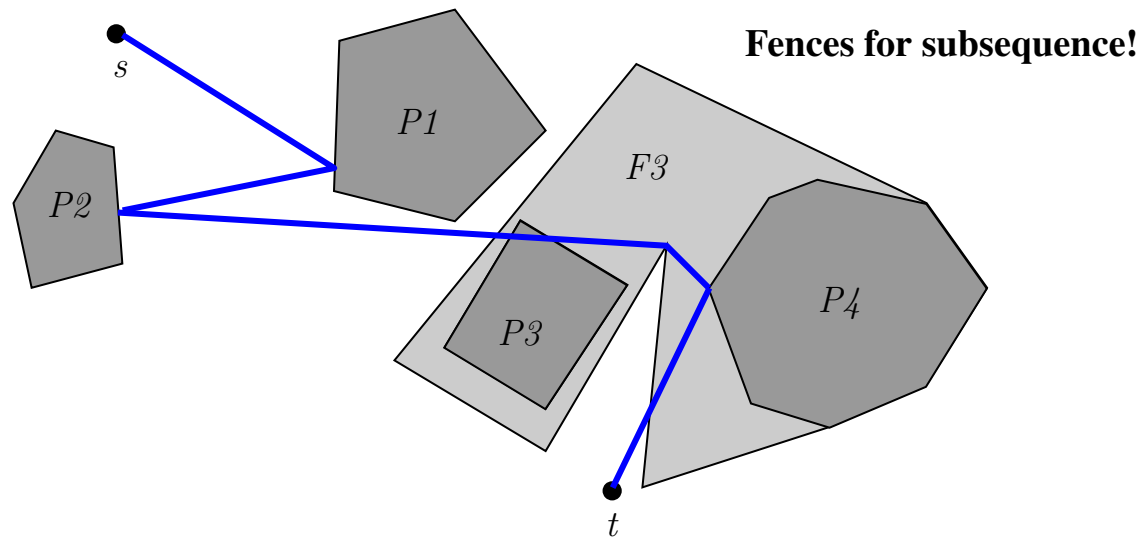
Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone
- Start s und Ziel t
- Besuche Polygone nach geg. Reihenfolge, Shortest Path



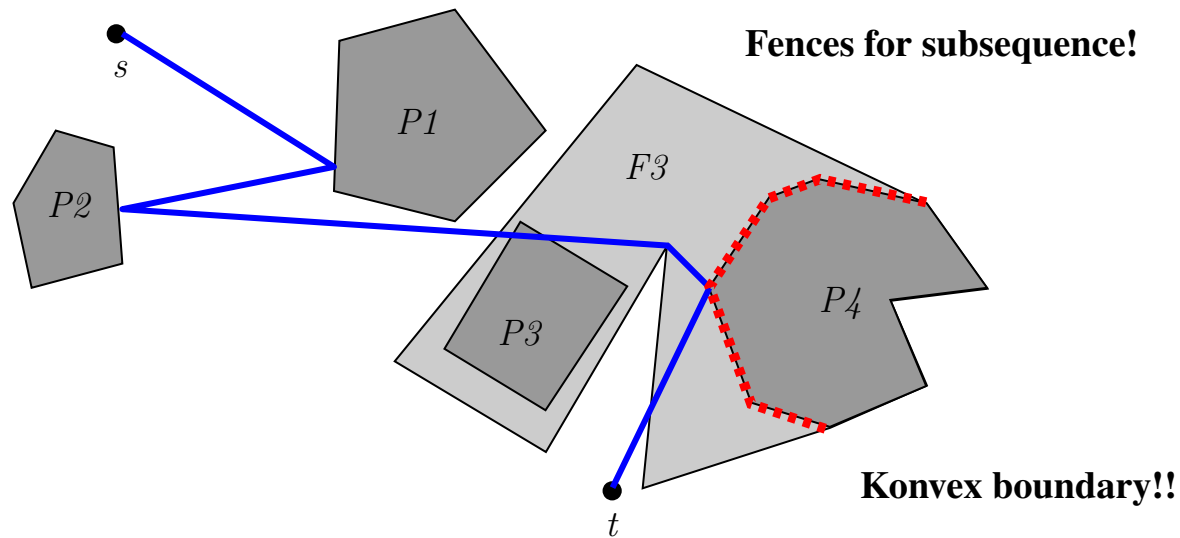
Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone
- Start s und Ziel t
- Besuche Polygone nach geg. Reihenfolge, Shortest Path



Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone
- Start s und Ziel t
- Besuche Polygone nach geg. Reihenfolge, Shortest Path



Touring a sequence of polygons (TPP)

- Sequenz konvexer Polygone
- Start s und Ziel t
- Besuche Polygone nach geg. Reihenfolge, Shortest Path

