Technische Universität Braunschweig

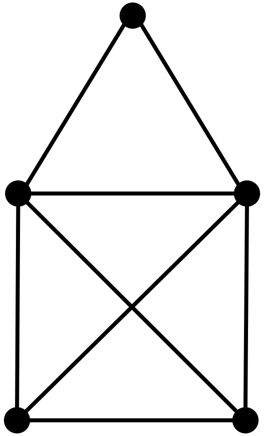
Algorithmen und Datenstrukturen
Detour: Polygonalisierungen

Matthias Konitzny, Arne Schmidt

18.11.2021

Euler, Hamilton und ein Handelsreisender

Euler und Hamilton



Eulerweg: Weg, der jede Kante genau einmal benutzt.

Eulertour: Geschlossener Weg, der jede Kante genau einmal benutzt.

Hamiltonpfad: Pfad, der jeden Knoten genau einmal benutzt.

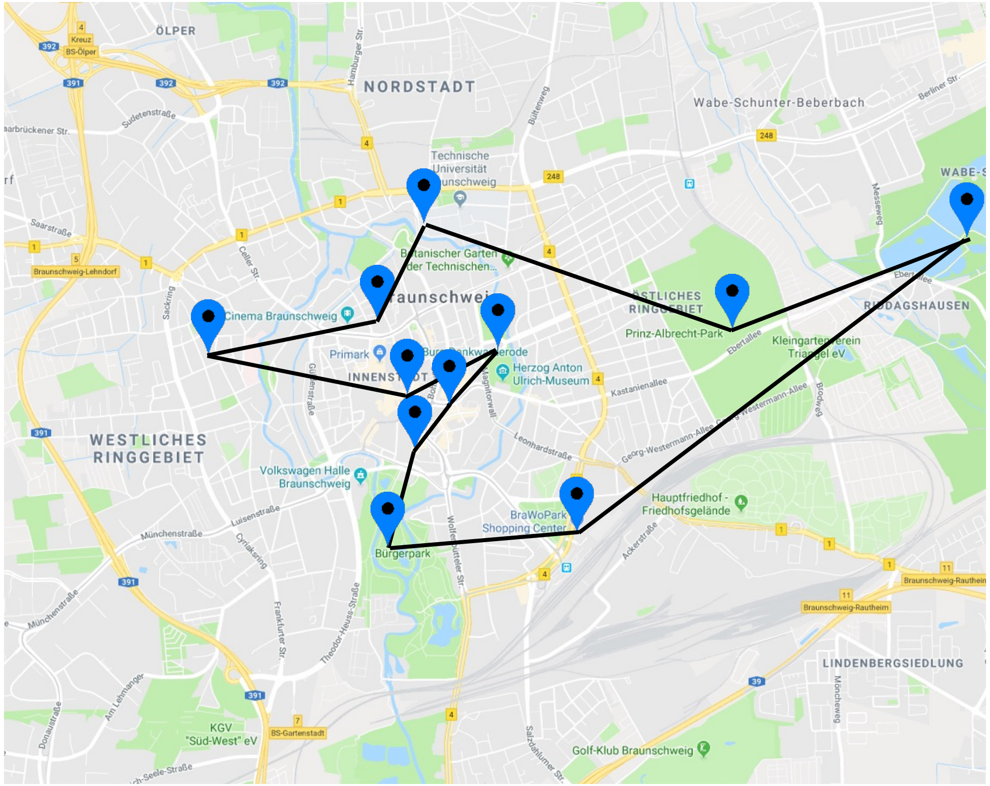
Hamiltonkreis: Geschlossener Pfad, der jeden Knoten genau einmal benutzt.

Ein Handelsreisender

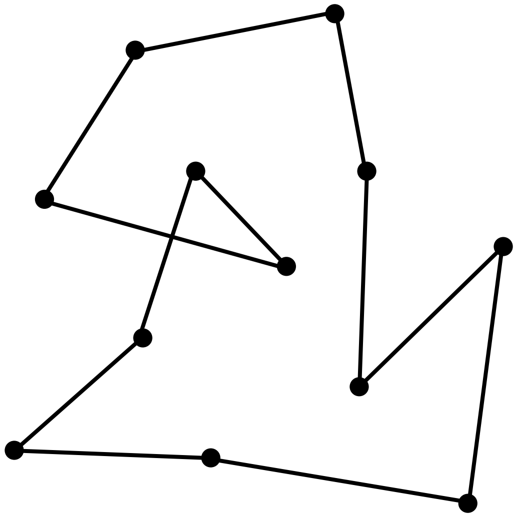
Gegeben:
Punkte $p_1, \ldots, p_n \in \mathbb{R}^2$

Gesucht:
Eine **kürzeste** Rundreise, die alle Punkte besucht.

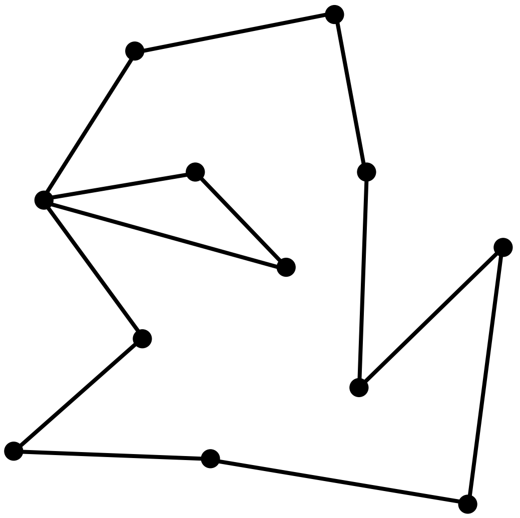
Wichtig: Euklidische Distanzen



Ein Handelsreisender



Ist diese Tour kleinstmöglich?

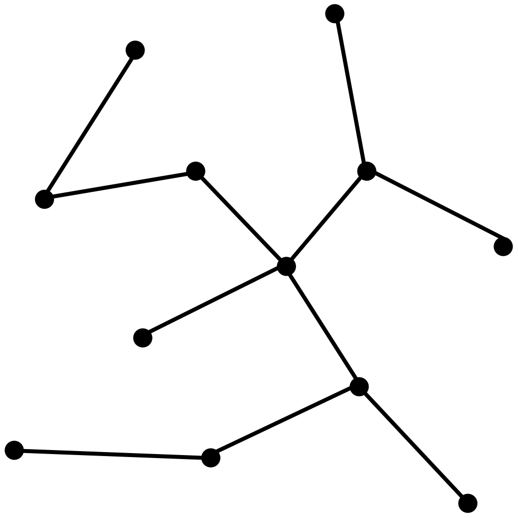


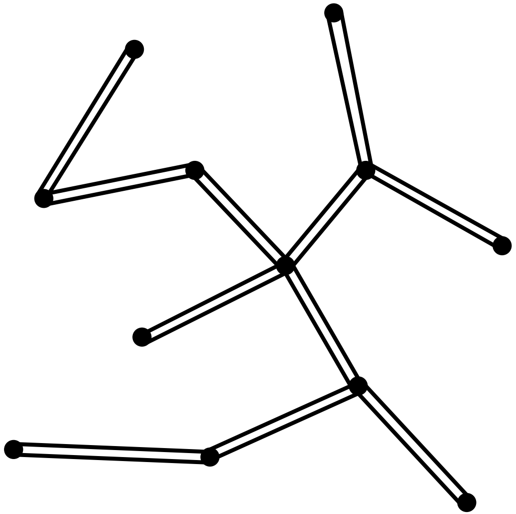
Kann man mit mehr Kanten kürzer werden?

Für beide Fragen: Nein! (Dreiecksungleichung)

Ein Handelsreisender

Gute Lösungen finden



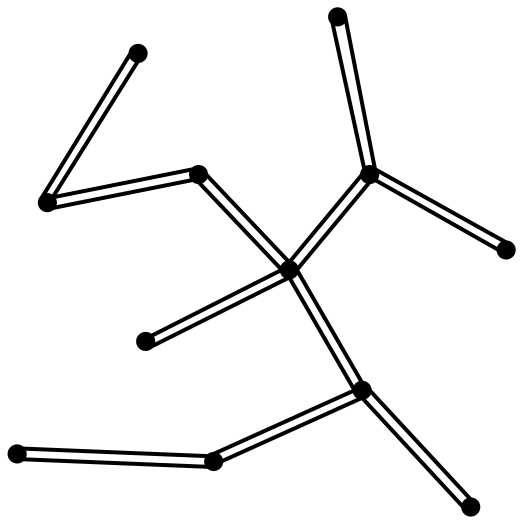


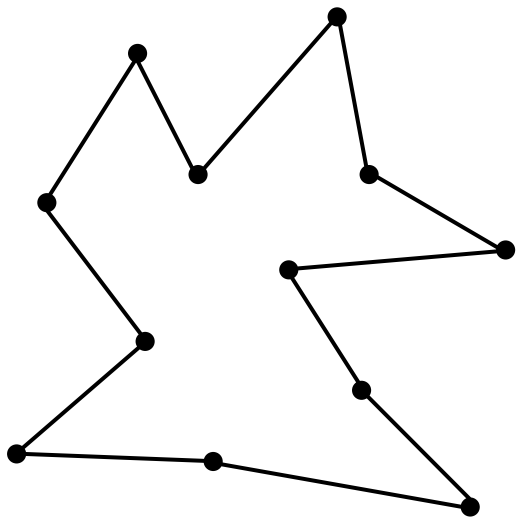
Verbinden mit möglichst kurzen Kanten.

Kantenkosten höchsten so hoch wie
die beste Rundreise!

Verdoppeln der Kanten liefert eine Rundreise, die
höchsten doppelt so lang wie die kürzeste ist.

Ein Handelsreisender



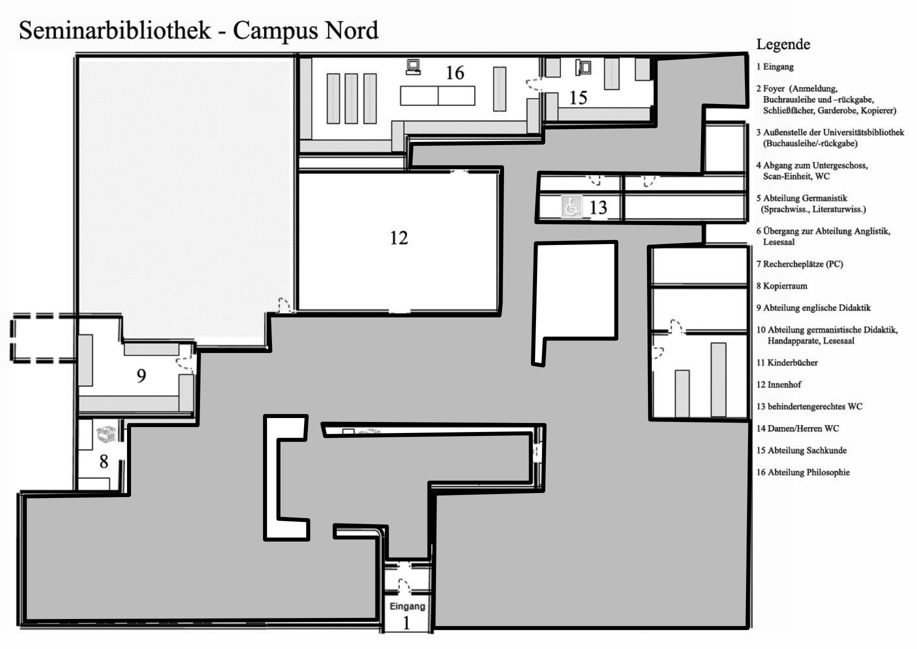


Verdoppeln der Kanten liefert eine Rundreise, die höchsten doppelt so lang wie die kürzeste ist.

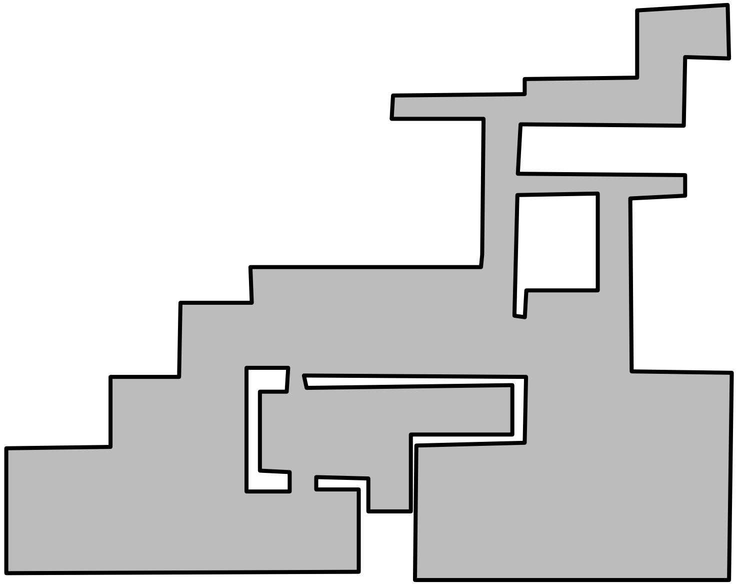
Auflösen der Doppelkanten über Dreiecksungleichung

Polygone und Polygonalisierungen

Polygone


Seminarbibliothek - Campus Nord
16
15
13
12
9
8
Eingang
1
Legende
1 Eingang
2 Foyer (Anmeldung, Buchrausleihe und –rückgabe, Schließfächer, Garderobe, Kopierer)
3 Außenstelle der Universitätsbibliothek (Buchausleihe/-rückgabe)
4 Abgang zum Untergeschoss, Scan-Einheit, WC
5 Abteilung Germanistik (Sprachwiss., Literaturwiss.)
6 Übergang zur Abteilung Anglistik, Lesesaal
7 Rechercheplätze (PC)
8 Kopierraum
9 Abteilung englische Didaktik
10 Abteilung germanistische Didaktik, Handapparate, Lesesaal
11 Kinderbücher
12 Innenhof
13 behindertengerechtes WC
14 Damen/Herren WC
15 Abteilung Sachkunde
16 Abteilung Philosophie

Polygone

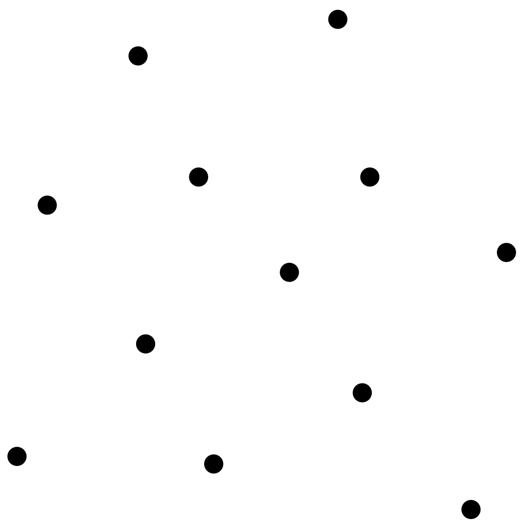


Ein Polygon P ist ein kreuzungsfreier Zug von n Segmenten mit möglichen weiteren Segmentzügen im Inneren.

Segmentzüge im Inneren heißen Löcher.

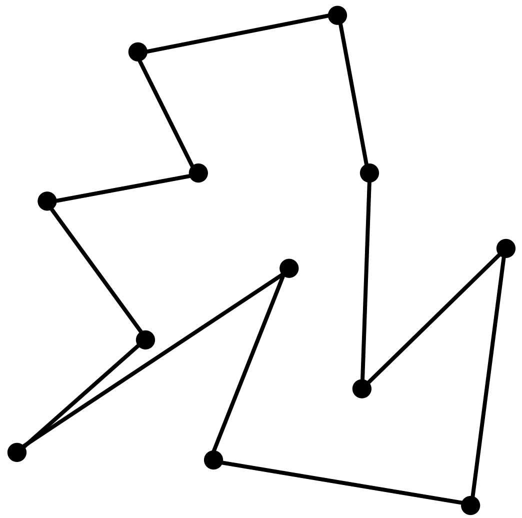
Gib es kein Loch, heißt P *einfach* (engl. simple).

Polygonalisierung



Gegeben: Eine Punktmenge V.

Polygonalisierung



Gegeben: Eine Punktmenge V.

Gesucht: (Einfaches) Polygon, dessen Eckpunkte die Punktmenge V ist.

Dabei: Optimiere bestimmte Eigenschaften

Minimaler Umfang

Minimaler Umfang

Computing Nonsimple Polygons of Minimum Perimeter

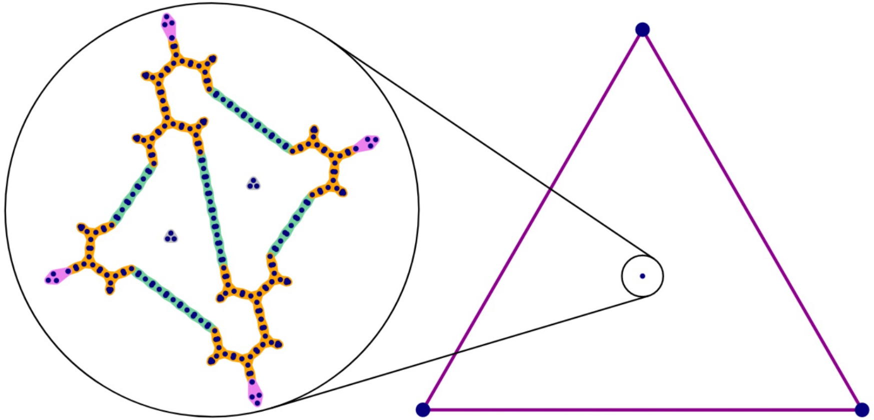
Sándor P. Fekete[1], Andreas Haas[1], Michael Hemmer[1], Michael Hoffmann[2], Irina Kostitsyna[3], Dominik Krupke[1], Florian Maurer[1], Joseph S. B. Mitchell[4], Arne Schmidt[1], Christiane Schmidt[5], and Julian Troegel[1]

[1] TU Braunschweig, Germany.
[2] ETH Zurich, Switzerland.
[3] TU Eindhoven, the Netherlands.
[4] Stony Brook University, USA.
[5] Linköping University, Sweden.

Minimaler Umfang

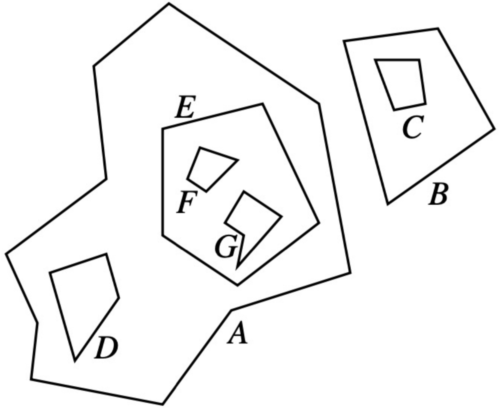
Minimaler Umfang

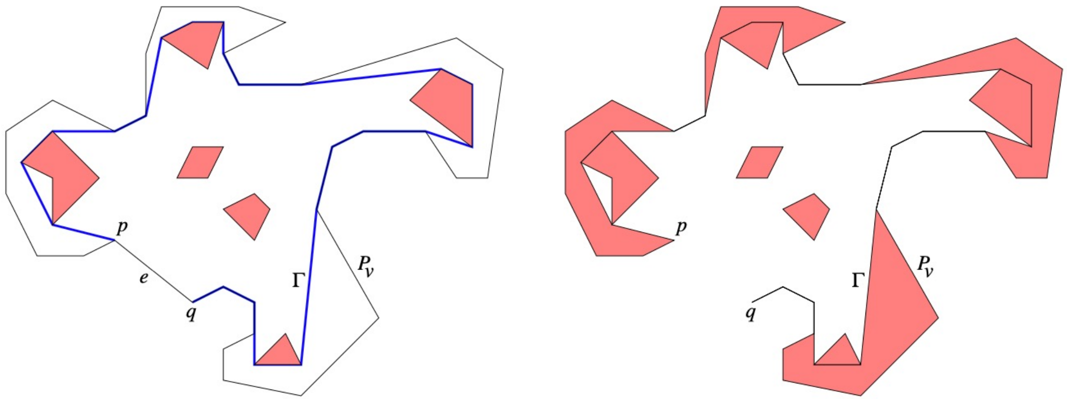
Problem:
Vermutlich kein effizienter Algorithmus möglich.



Minimaler Umfang

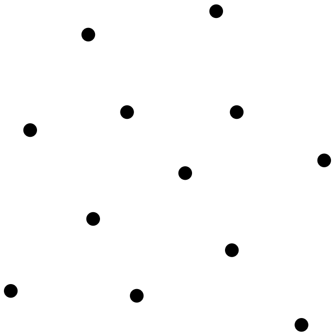
Aber:
Gute Lösungen können effizient berechnet werden.





Minimaler Umfang

Trotzdem:
Wie können optimale Lösungen berechnet werden?



Instanz

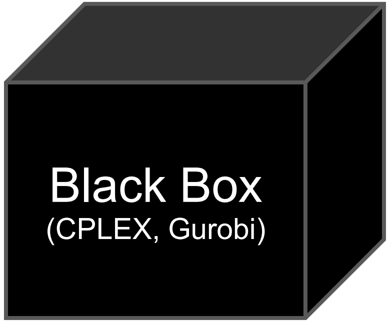
$$\min \sum_{e \in E} x_e c_e \,.$$

$$\forall v \in P: \sum_{e \in \delta(v)} x_e = 2\,,$$

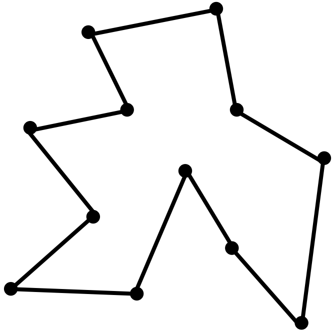
$$\forall C \in \mathcal{C}: \sum_{e \in C} x_e \leq |C| - 1\,,$$

$$x_e \in \{0,1\}\,.$$

Modell



Lösungsverfahren



Lösung

Minimaler Umfang

Die Menge $\mathcal{C}$

$$\min \sum_{e \in E} x_e c_e \,.$$

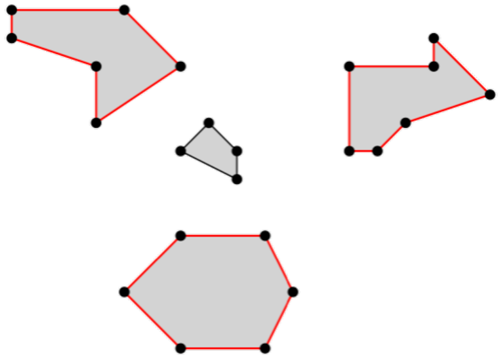
$$\forall v \in P: \sum_{e \in \delta(v)} x_e = 2\,,$$

$$\forall C \in \mathcal{C}: \quad \sum_{e \in C} x_e \leq |C| - 1\,,$$

$$x_e \in \{0, 1\}\,.$$

Minimaler Umfang

Die Menge $\mathcal{C}$



$$\min \sum_{e \in E} x_e c_e \,.$$

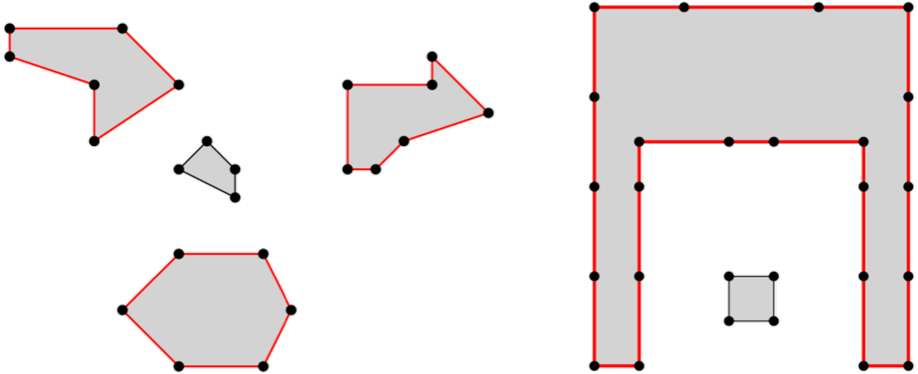
$$\forall v \in P: \sum_{e \in \delta(v)} x_e = 2\,,$$

$$\forall C \in \mathcal{C}: \quad \sum_{e \in C} x_e \leq |C| - 1\,,$$

$$x_e \in \{0,1\}\,.$$

Minimaler Umfang

Die Menge $\mathcal{C}$



$$\min \sum_{e \in E} x_e c_e \,.$$

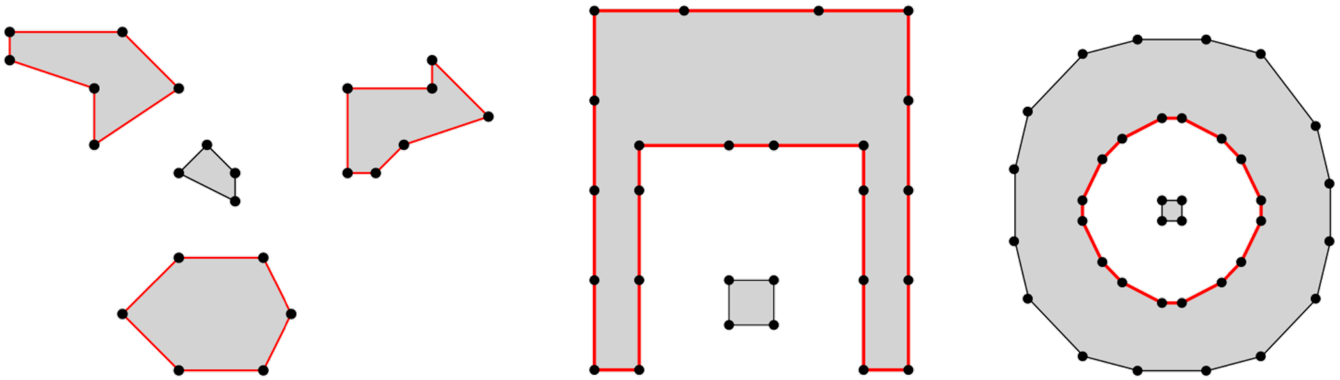
$$\forall v \in P: \sum_{e \in \delta(v)} x_e = 2\,,$$

$$\forall C \in \mathcal{C}: \quad \sum_{e \in C} x_e \leq |C| - 1\,,$$

$$x_e \in \{0,1\}\,.$$

Minimaler Umfang

Die Menge $\mathcal{C}$



$$\min \sum_{e \in E} x_e c_e \,.$$

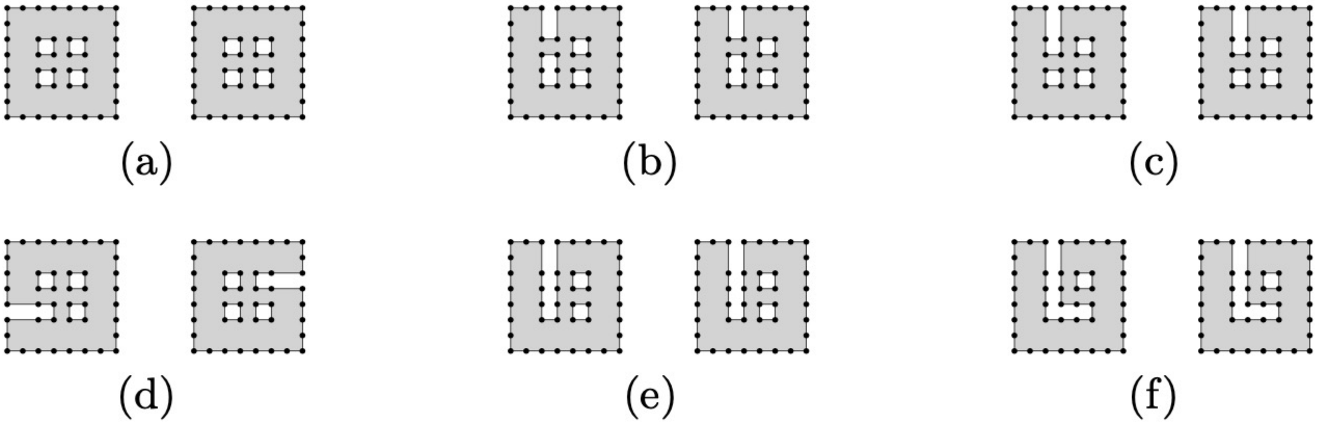
$$\forall v \in P: \sum_{e \in \delta(v)} x_e = 2\,,$$

$$\forall C \in \mathcal{C}: \quad \sum_{e \in C} x_e \leq |C| - 1\,,$$

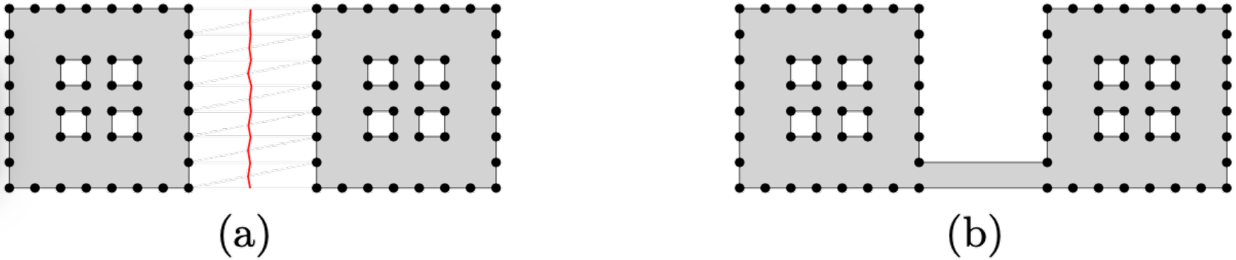
$$x_e \in \{0,1\}\,.$$

→ Sehr viele (exponentiell) Bedingungen!
Füge Bedingungen nach und nach hinzu.

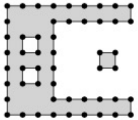
Minimaler Umfang



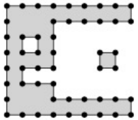
$$\sum_{e \in \mathcal{X}(R_D)} x_e \geq 2$$



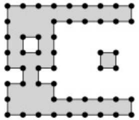
Minimaler Umfang



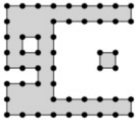
(a)



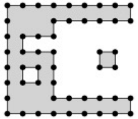
(b)



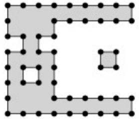
(c)



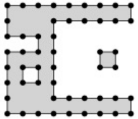
(d)



(e)

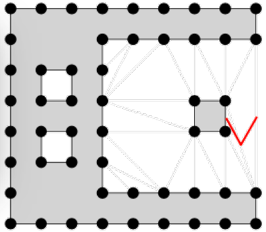


(f)

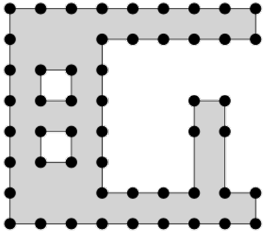


(g)

$$\underbrace{\sum_{e \in \mathcal{X}(R_T) \setminus \delta(C)} x_e}_{\text{C gets surrounded}} + \underbrace{\sum_{e \in \delta(C)} x_e}_{\text{C dissolves}} \geq 1$$

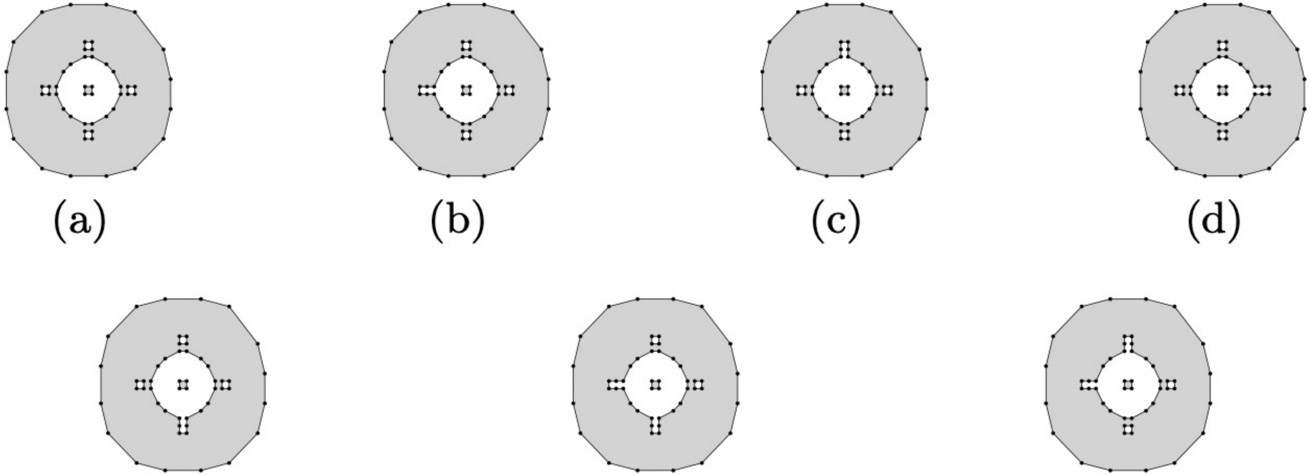


(a)



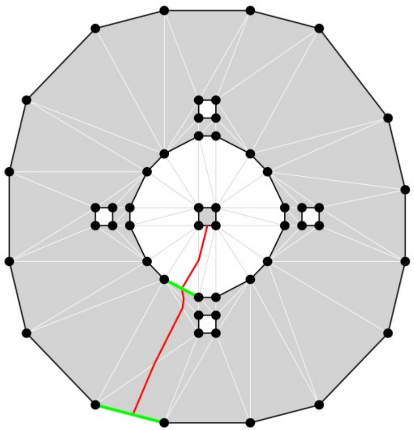
(b)

Minimaler Umfang

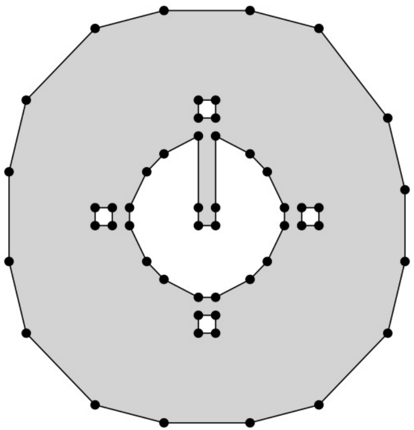


(a) (b) (c) (d) (e) (f) (g)

Minimaler Umfang



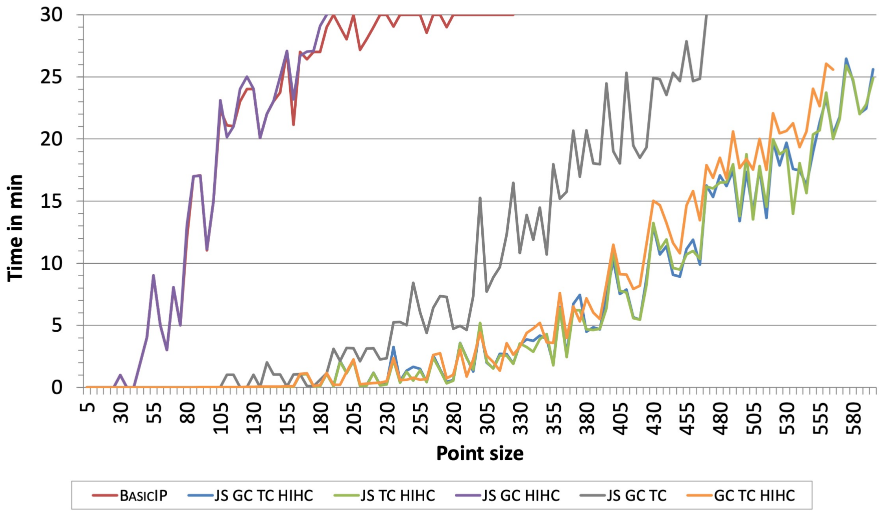
(a)

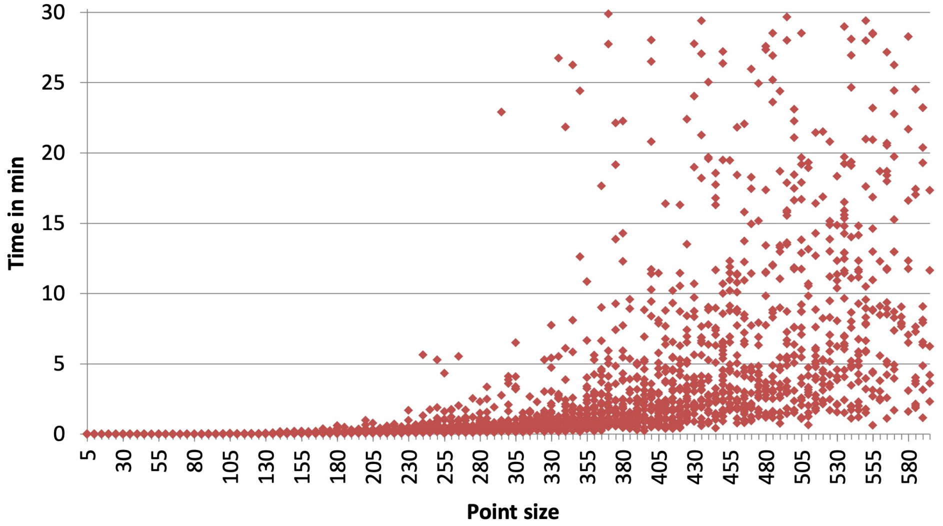


(b)

$$\underbrace{\sum_{e \in \delta(V_H, V \setminus V_H)} x_e}_{H \text{ dissolves}} + \underbrace{\sum_{e \in \mathcal{X}(P) \setminus \{e_1, e_2\}} x_e + -x_{e_1} - x_{e_2}}_{\text{Crossing of } P \text{ changes}} \geq -1$$

Minimaler Umfang





Minimale/Maximale Fläche

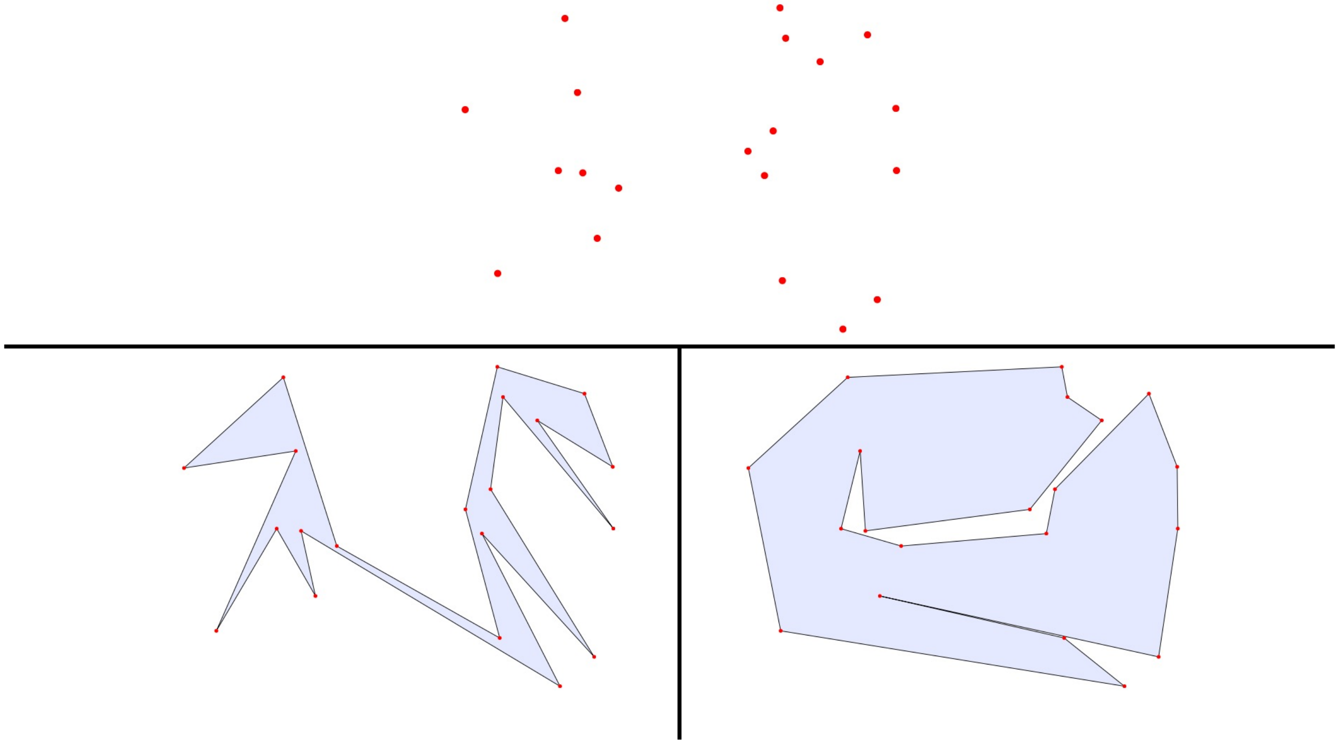
Minimaler Umfang

Computing Area-Optimal Simple Polygonizations

SÁNDOR P. FEKETE, Department of Computer Science, TU Braunschweig, Germany
ANDREAS HAAS, Department of Computer Science, TU Braunschweig, Germany
PHILLIP KELDENICH, Department of Computer Science, TU Braunschweig, Germany
MICHAEL PERK, Department of Computer Science, TU Braunschweig, Germany
ARNE SCHMIDT, Department of Computer Science, TU Braunschweig, Germany

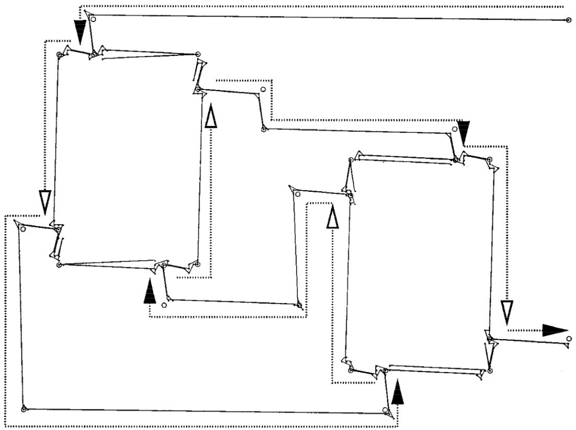
We consider methods for finding a simple polygon of minimum (MIN-AREA) or maximum (MAX-AREA) possible area for a given set of points in the plane. Both problems are known to be NP-hard; at the center of the recent CG Challenge, practical methods have received considerable attention. However, previous methods focused on heuristic methods, with no proof of optimality. We develop exact methods, based on a combination of geometry and integer programming. As a result, we are able to solve instances of up to $n = 25$ points to provable optimality. While this extends the range of solvable instances by a considerable amount, it also illustrates the practical difficulty of both problem variants.

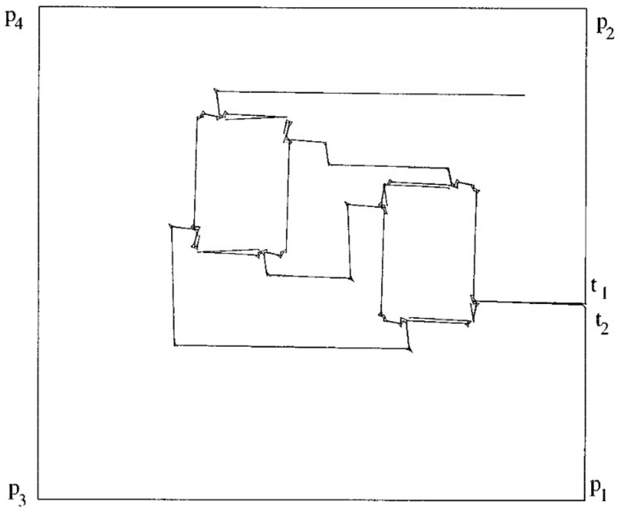
Minimale/Maximale Fläche



Minimale/Maximale Fläche

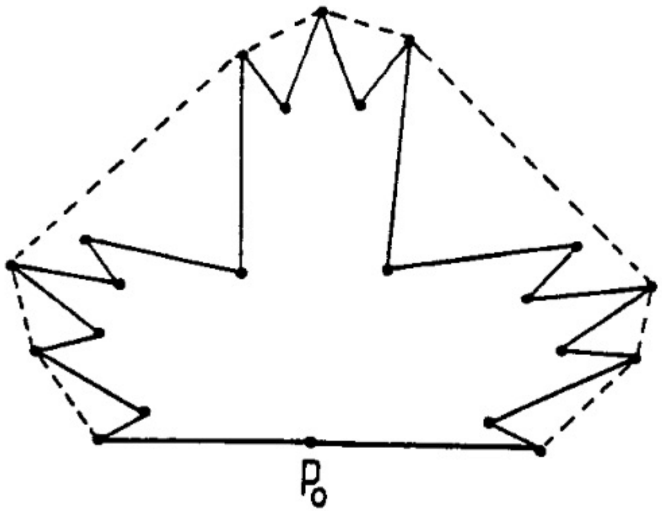
Problem:
Vermutlich kein effizienter Algorithmus
für beide Probleme möglich.

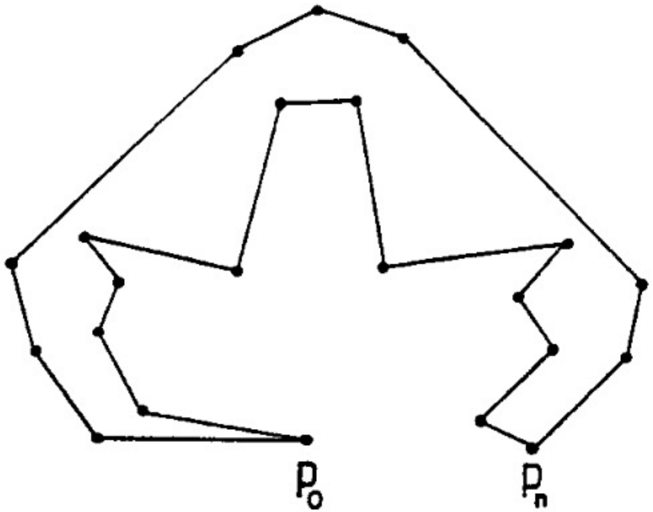




Minimale/Maximale Fläche

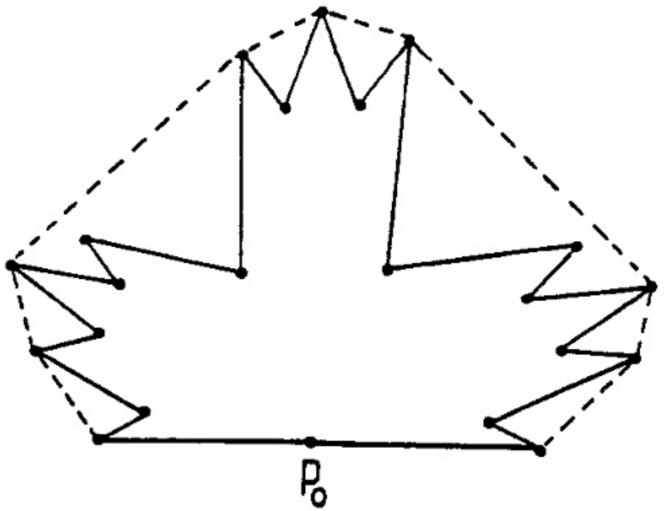
Aber:
Gute Lösungen für MaxArea können effizient berechnet werden.

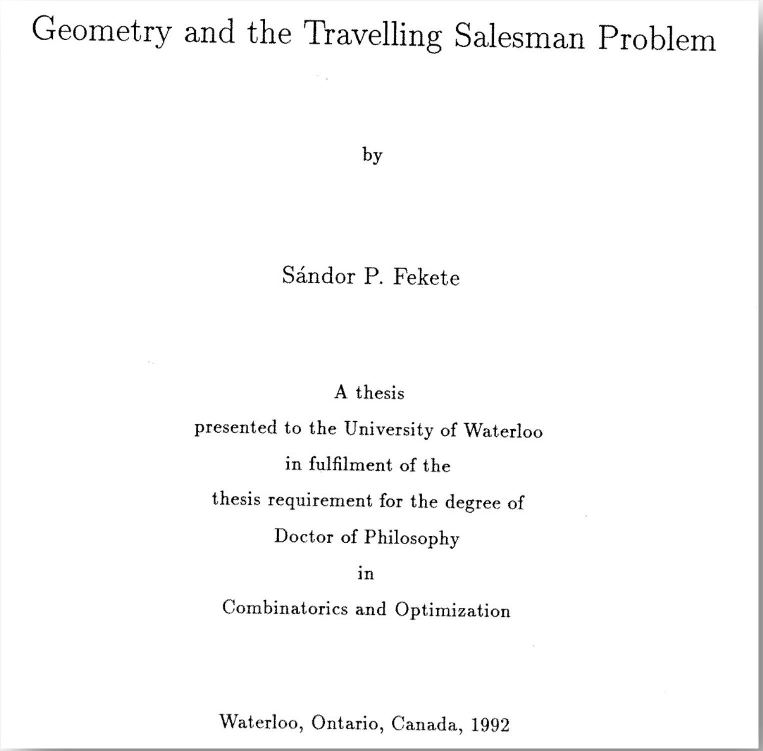




Minimale/Maximale Fläche

Aber:
Gute Lösungen für MaxArea können effizient berechnet werden.




Geometry and the Travelling Salesman Problem

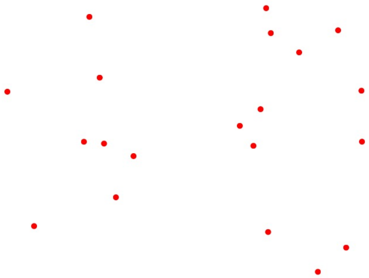
by

Sándor P. Fekete

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 1992

Trotzdem:
Wie können optimale Lösungen berechnet werden?



$$\{\min, \max\} \sum_{e \in E^r} z_e \cdot f_e$$

$$\forall s_i \in S: \quad \sum_{(j,i) \in \delta^+(s_i)} z_{ji} = 1$$

$$\forall s_i \in S: \quad \sum_{(i,j) \in \delta^-(s_i)} z_{ij} = 1$$

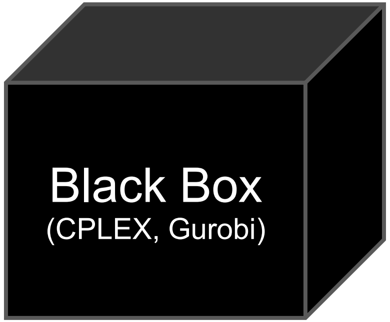
$$\forall e = \{i,j\} \in E: \quad z_{ij} + z_{ji} \leq 1$$

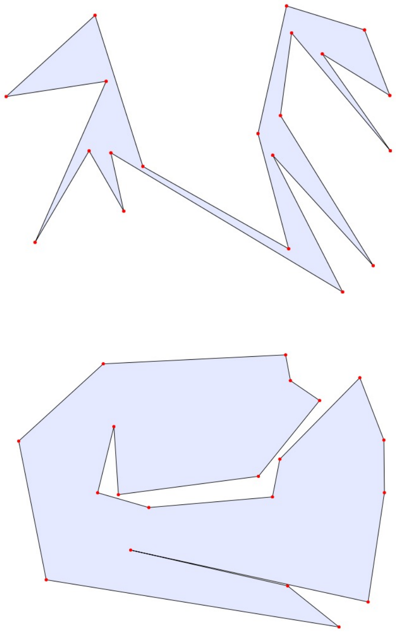
$$\forall \text{intersecting} \quad \{i,j\},\{k,l\} \in E: \quad z_{ij} + z_{ji} + z_{kl} + z_{lk} \leq 1$$

$$(\forall \text{slabs} \quad D)(\forall m = 1, \ldots, |D|): \quad \sum_{i=1}^{m} z_{e^{lr}_{i_D}} - z_{e^{rl}_{i_D}} \begin{array}{l} \leq 1 \\ \geq 0 \end{array}$$

$$\forall \emptyset \neq D \subsetneq S: \quad \begin{array}{l} \sum_{(k,l) \in \delta^-(D)} z_{kl} \geq 1 \\ \sum_{(k,l) \in \delta^+(D)} z_{kl} \geq 1 \end{array}$$

$$\forall i,j \in E: \quad z_{ij}, z_{ji} \in \{0,1\}$$



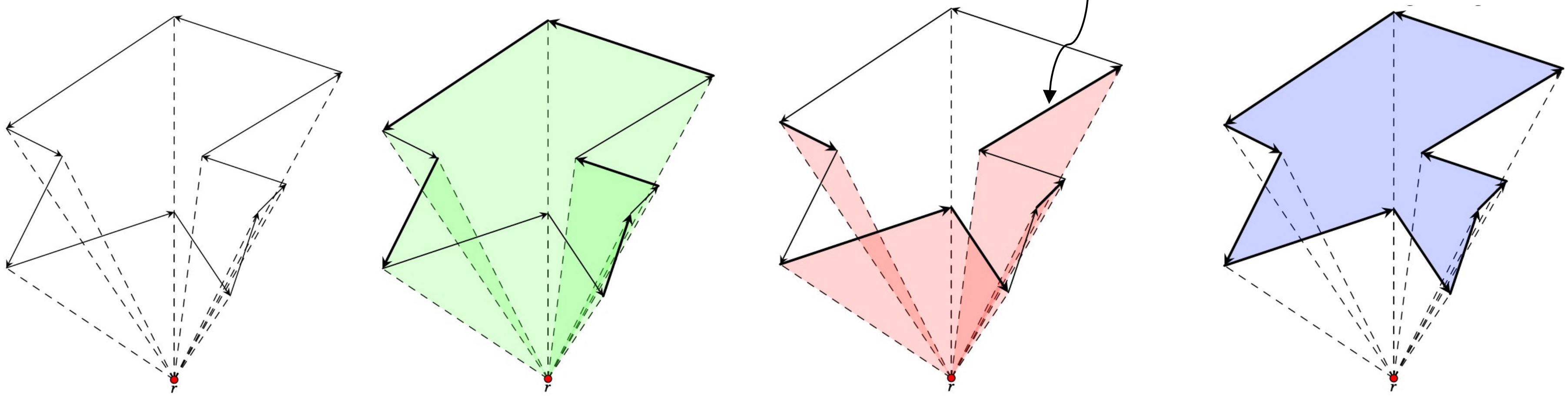


Instanz | Modell | Lösungsverfahren | Lösung

Minimale/Maximale Fläche

Bestimmung des Flächeninhaltes.

Fläche des Dreiecks $\widehat{=}$ Gewicht der Kante
Orientierung $\widehat{=}$ Vorzeichen



Kanten gegen den Uhrzeigesinn richten.

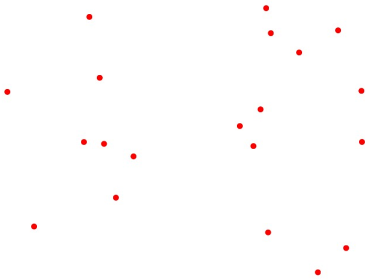
Kante bzgl. r von rechts nach links: Dreieck positiv.

Kante bzgl. r von links nach rechts: Dreieck negativ.

Differenz

Minimale/Maximale Fläche

Alternativ:
Modell über Dreiecke



$$\{\min, \max\} \sum_{\triangle \in T(P)} f_\triangle \cdot x_\triangle$$

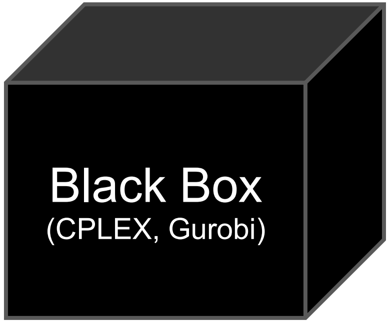
$$\sum_{\triangle \in T} x_\triangle = n - 2$$

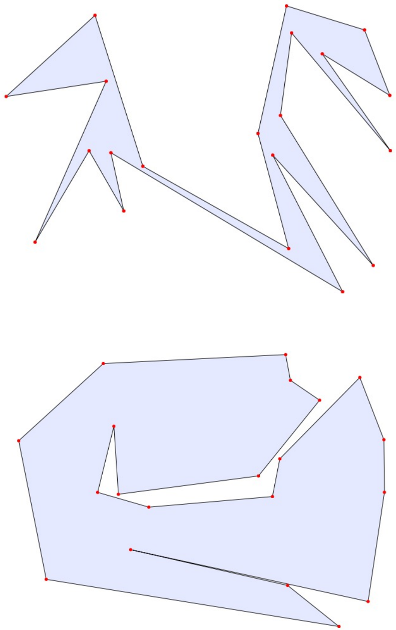
$$\forall s_i \in S: \quad \sum_{\triangle \in \delta(s_i)} x_\triangle \geq 1$$

$$\forall \text{intersecting} \quad \triangle_i, \triangle_j \in T(P): \quad x_{\triangle_i} + x_{\triangle_j} \leq 1$$

$$\forall \emptyset \neq D \subsetneq T(P), |D| \leq n-3: \quad \sum_{\triangle \in D} x_\triangle - \sum_{\triangle \in \delta(D)} x_\triangle \leq |D| - 1$$

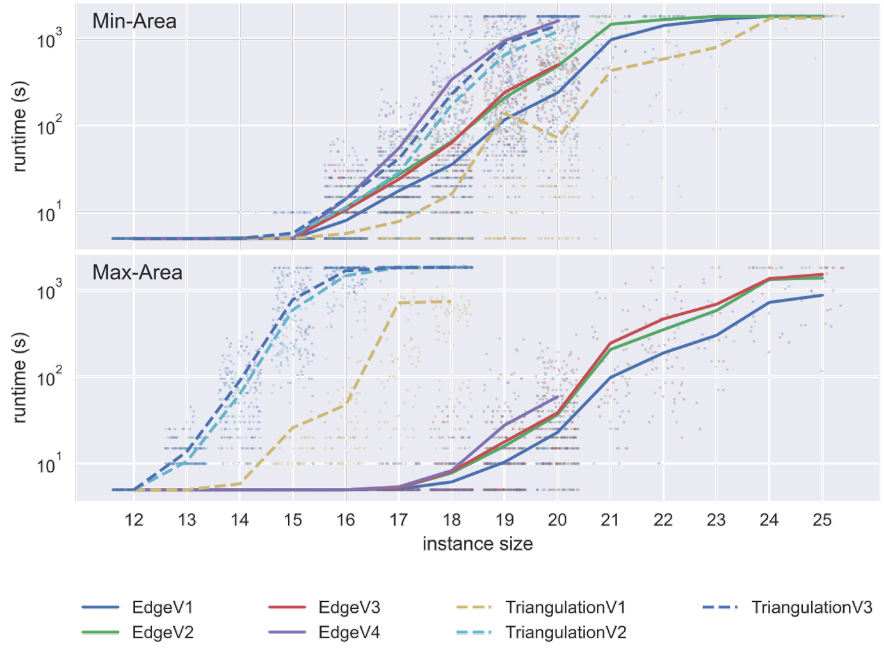
$$\forall \triangle \in T(P): \quad x_\triangle \in \{0, 1\}$$

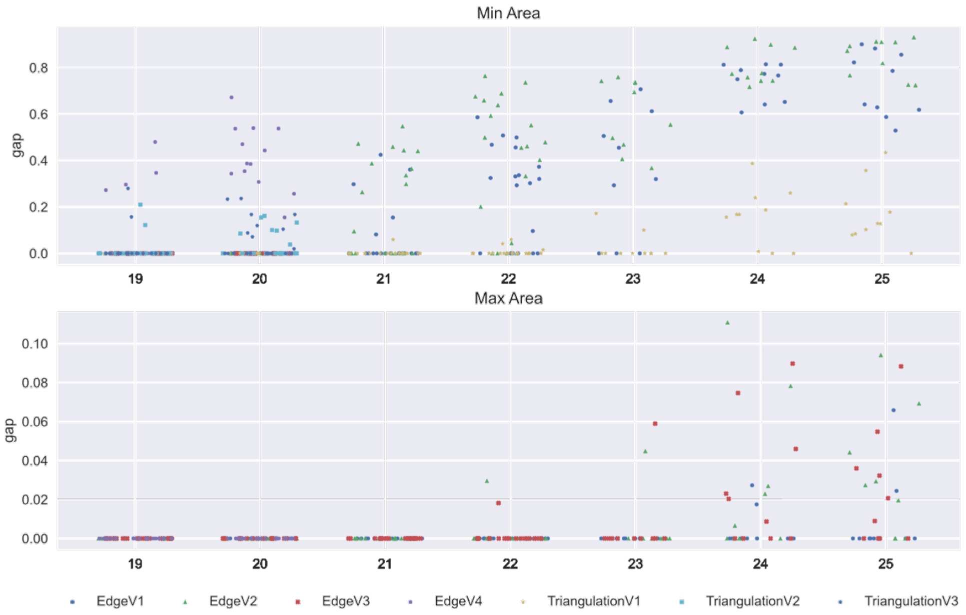




Instanz | Modell | Lösungsverfahren | Lösung

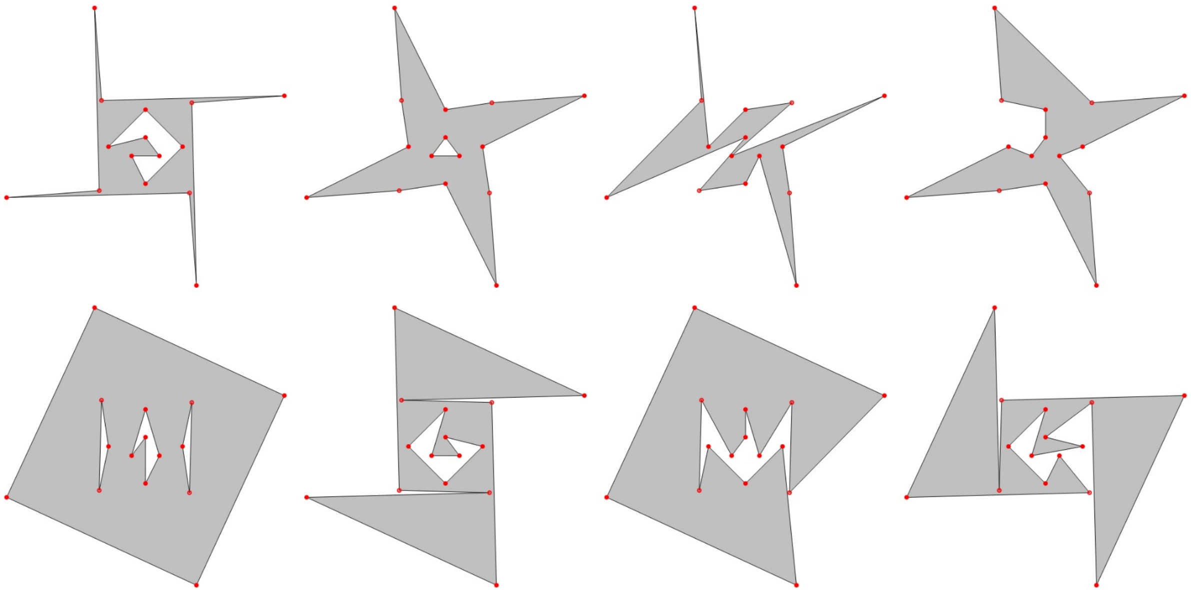
Minimale/Maximale Fläche





Ausblick

Verschiedene Polygonalisierungen



Top: Minimizing; Bottom: Maximizing

From left to right: Area; Boundary; Simple Area; Simple Boundary

Weitere Polygonalisierungen

Eigenschaften
Kantenlänge, Fläche, Innenwinkel,
Anzahl konvexer Ecken, …

Polygonart
Einfache/Nicht-einfache Polygone
Orthogonal, selbstschneidend, …

Optimierungen
Maximieren, Minimieren
Min-Max, Max-Min, …

Budget Variante
Für eine gegebene Maximallänge
optimiere eine Eigenschaft.

Weitere Polygonalisierungen

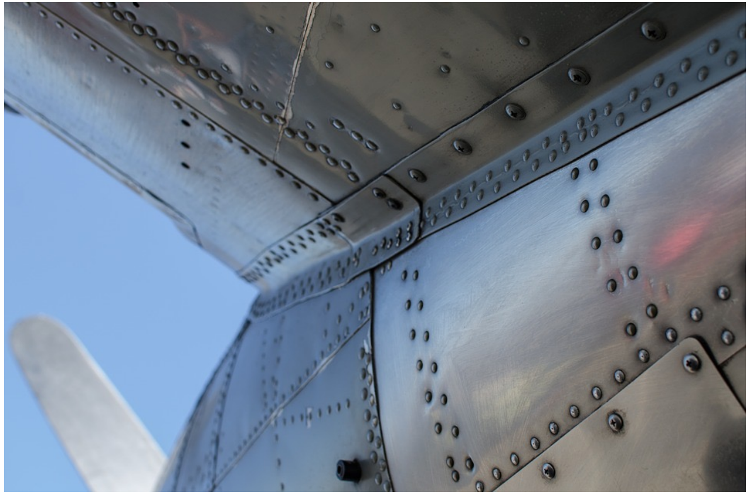
Eigenschaften
Kantenlänge, Fläche, Innenwinkel,
Anzahl konvexer Ecken, …

Polygonart
Einfache/Nicht-einfache Polygone
Orthogonal, selbstschneidend, …

Optimierungen
Maximieren, Minimieren
Min-Max, Max-Min, …

Budget Variante
Für eine gegebene Maximallänge
optimiere eine Eigenschaft.

Maximum Scatter TSP



On the Maximum Scatter TSP*

Esther M. Arkin† Yi-Jen Chiang‡ Joseph S. B. Mitchell§
Steven S. Skiena¶ Tae-Cheon Yang‖

December 27, 1996

Motivation.

This *maximum scatter TSP* problem arises in some manufacturing processes where it is important to have substantial separation (distance) between consecutive (or *nearly* consecutive) operations on a workpiece. We first encountered the problem in discussions with Boeing, where the problem was that of sequencing the riveting operations when fastening sheets of metal together [16, 17, 18, 19, 20].