Technische Universität Braunschweig

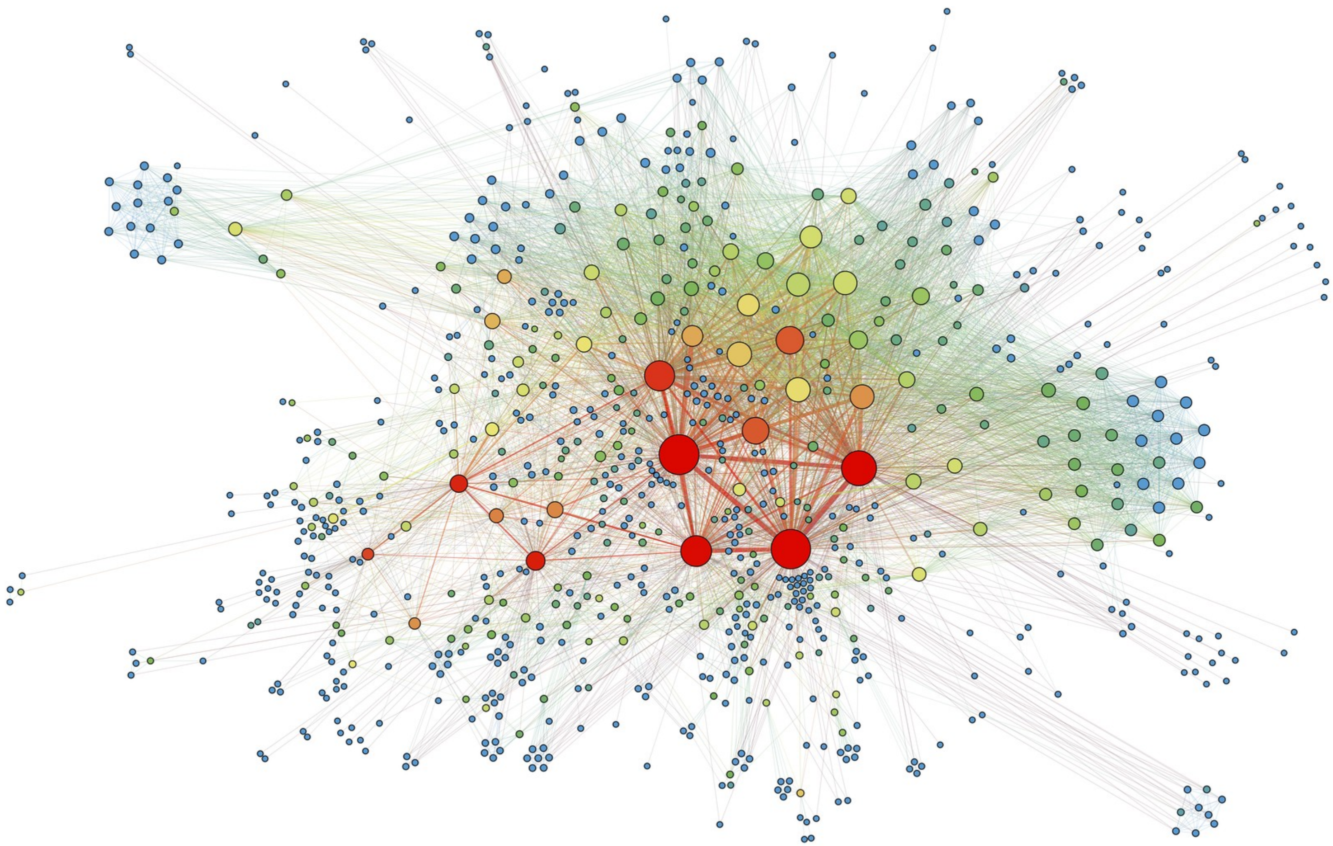
Probing a Set of Trajectories to Maximize Captured Movement

Sándor P. Fekete, **Alexander Hill**, Dominik Krupke, Tyler Mayer, Joseph S. B. Mitchell, Ojas Parekh and Cynthia A. Phillips

SEA 2020

ahill@ibr.cs.tu-bs.de

Motivation

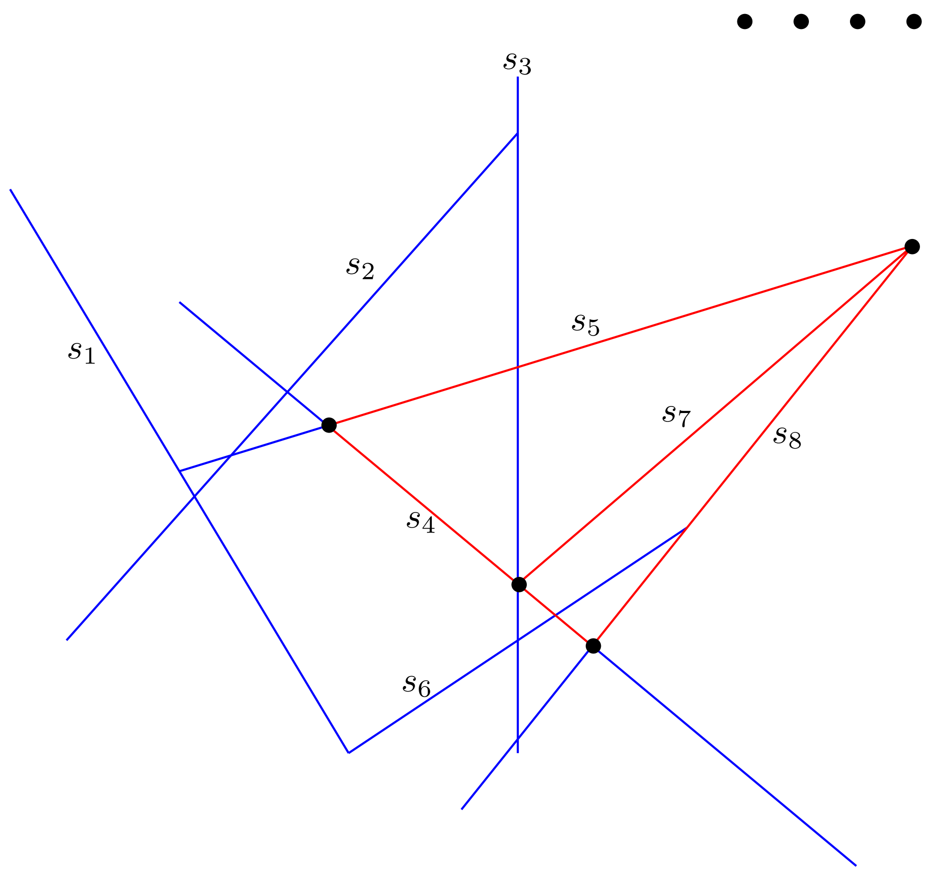


Motivation

Sándor P. Fekete, **Alexander Hill**, Dominik Krupke, Tyler Mayer, Joseph S. B. Mitchell, Ojas Parekh and Cynthia A. Phillips
ahill@ibr.cs.tu-bs.de

Overview

Trajectory Capturing Problem



Given:

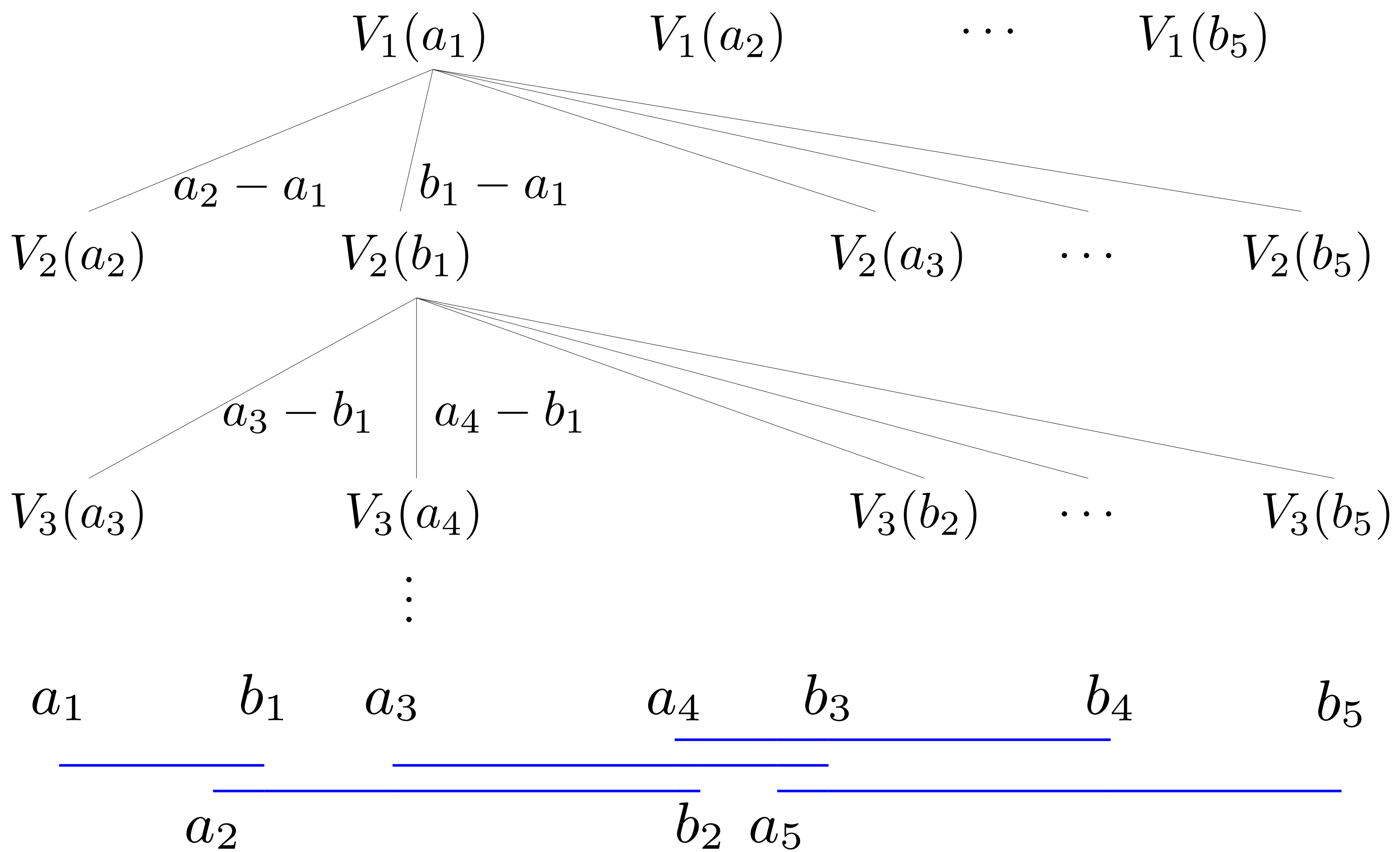
- Set $\mathcal{T}$ of trajectories
- $k \geq 2$ portals/pins

Wanted:

- Place k portals
- maximize the total length
- Measured between two portals

Overview

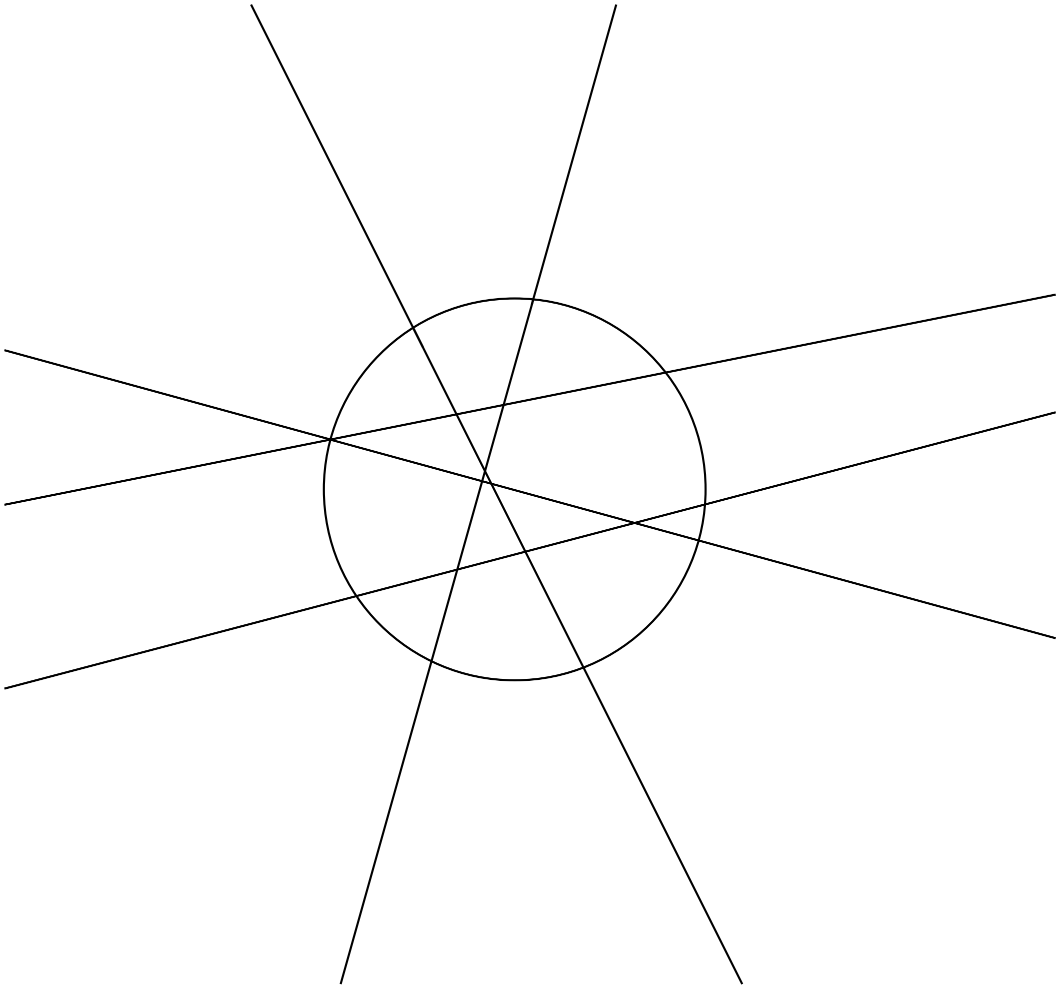
Complexity: One-Dimensional TCP



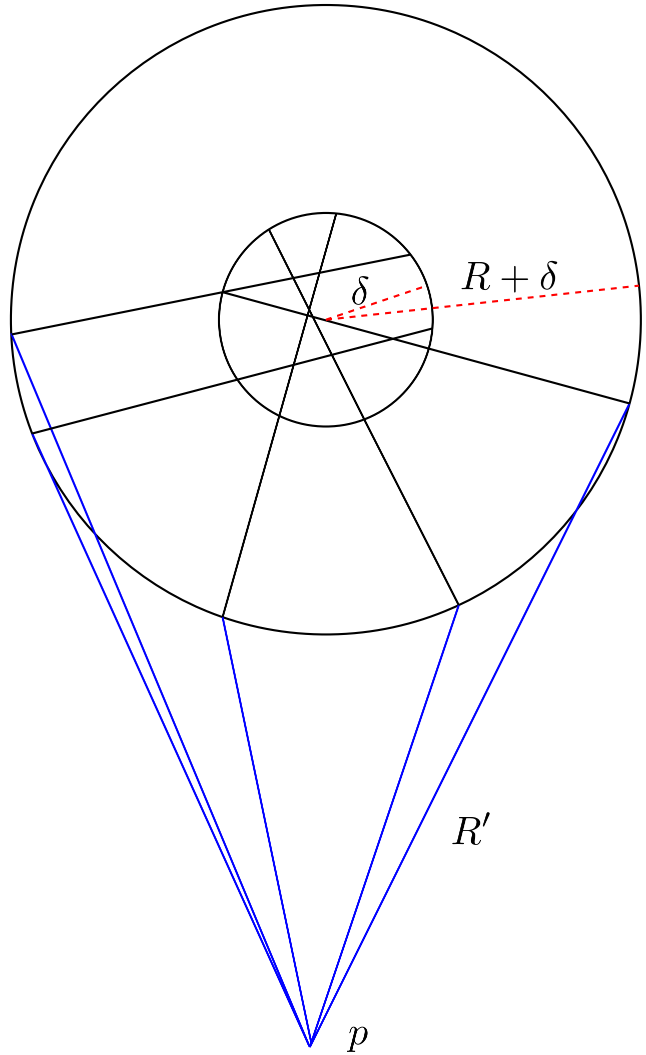
Complexity: One-Dimensional TCP

$$V_i(x) = \max_{x' \in \{a_1,\ldots,a_n,b_1,\ldots,b_n\}, x'>x} \{V_{i+1}(x') + \sum_{j:(x,x') \subseteq (a_j,b_j)} (x'-x)\}$$

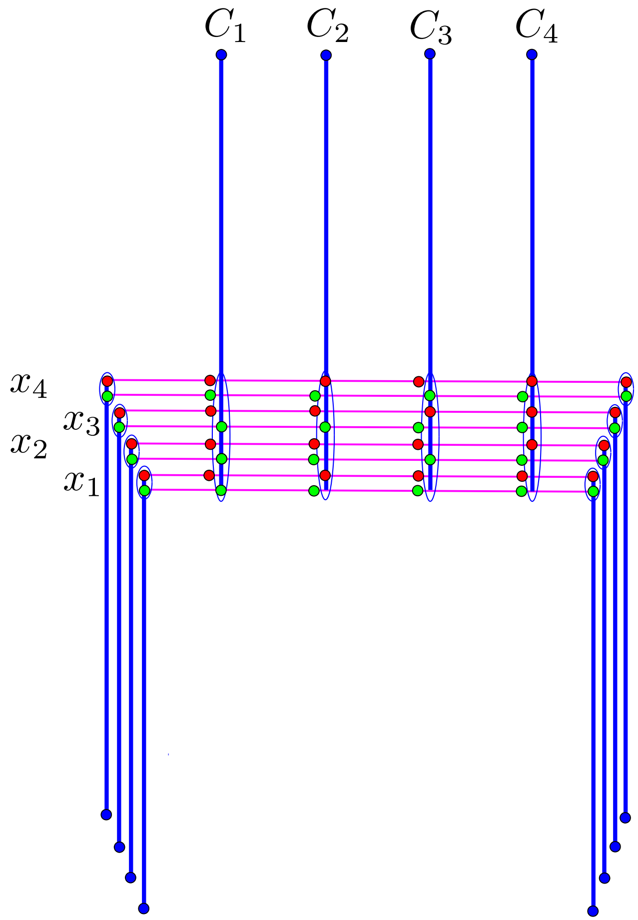
Complexity: Geometric TCP



Complexity: Geometric TCP



Complexity: Axis-parallel TCP



$$(x_1 \vee x_2 \vee x_3) \wedge (\overline{x}_1 \vee x_3 \vee \overline{x}_4) \wedge (x_2 \vee \overline{x}_3 \vee x_4) \wedge (\overline{x}_2 \vee \overline{x}_3 \vee \overline{x}_4)$$

Overview

1. Trajectory Capturing Problem
2. Complexity
3. **Approximation Schemes**
    1. **K-Approximation**
    2. **Δ-Approximation**
4. Integer Programming
5. Heuristics
6. Evaluation
7. Conclusion

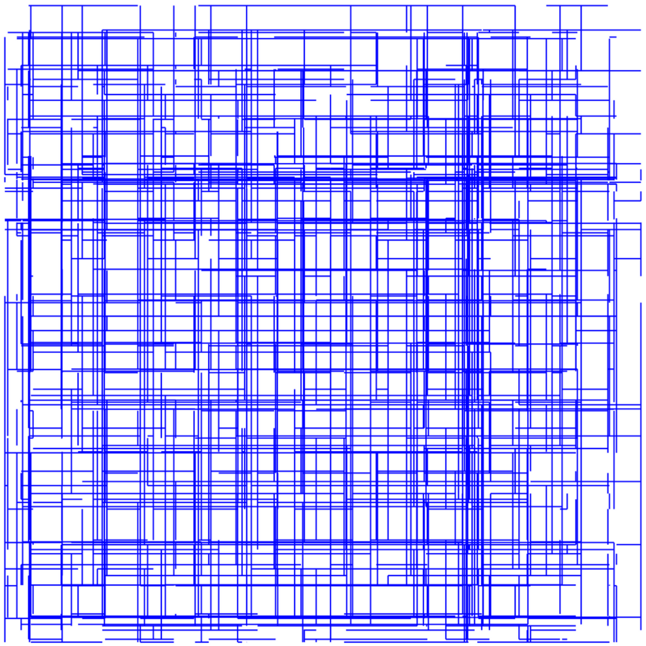
K-approximation

$$\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 \cup \cdots \cup \mathcal{T}_K$$

$$\Rightarrow \forall \mathcal{T}_i \in \mathcal{T} :$$

Solve $\mathcal{T}_i$ with 1D DP

Best solution: K-approximation



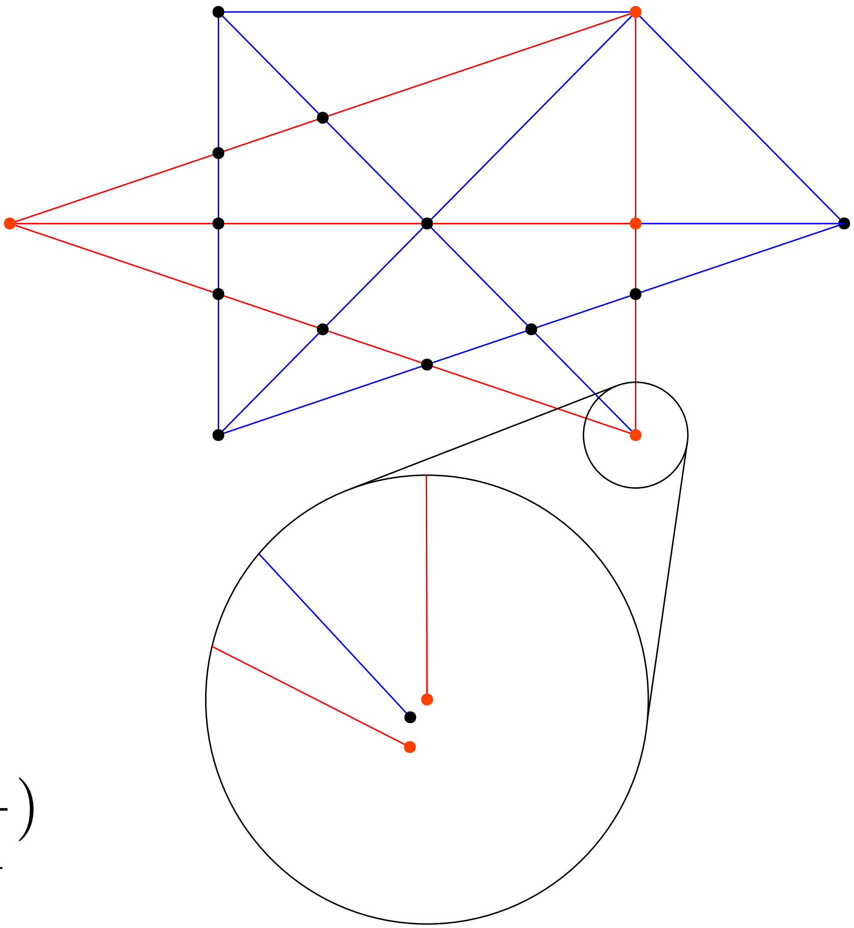
Δ-approximation

$$L^* \leq \ell_1 + \ell_2 + \cdots + \ell_{\lfloor k\Delta/2 \rfloor}$$

$$L = \ell_1 + \ell_2 + \cdots + \ell_{\lfloor k/2 \rfloor}$$

$$\frac{L^*}{L} \leq \frac{k\Delta/2}{k/2} = \Delta$$

$$\frac{L^*}{L} \leq \frac{k\Delta/2}{(k-1)/2} = \Delta(1 + \frac{1}{k-1})$$

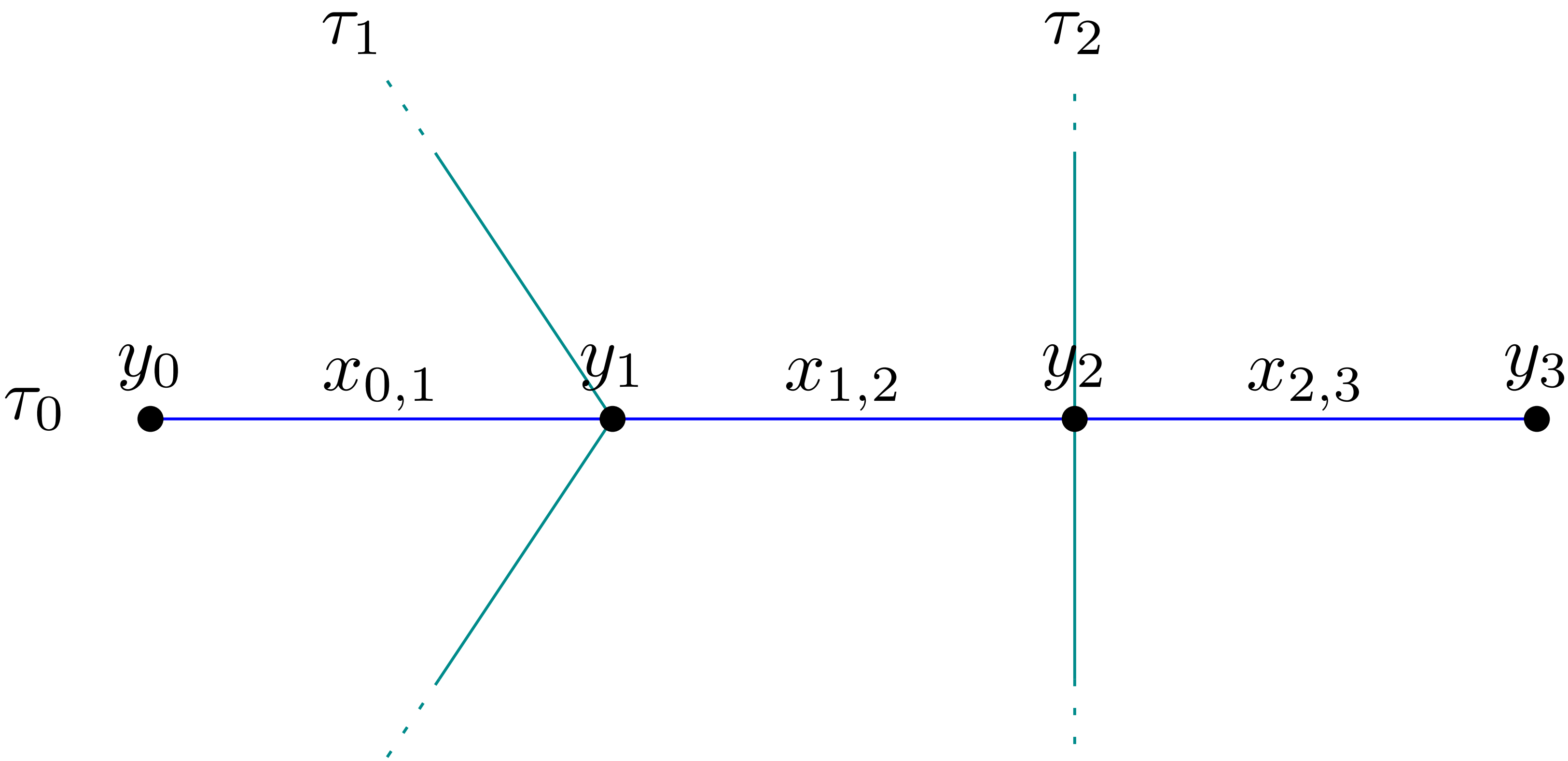


$\delta \leq \Delta$ clones

Overview

1. Trajectory Capturing Problem
2. Complexity
3. Approximation Schemes
4. **Integer Programming**
    1. **Formulation**
    2. **Integrality Gap**
5. Heuristics
6. Evaluation
7. Conclusion

Integer Program Preliminaries



Integer Program Formulation

$$\max \sum_{\tau \in \mathcal{T}, e \in E(\tau)} f(e) x_{\tau,e}$$

$$\sum_{v \in V} y_v \leq k \qquad \text{(Constraint 1)}$$

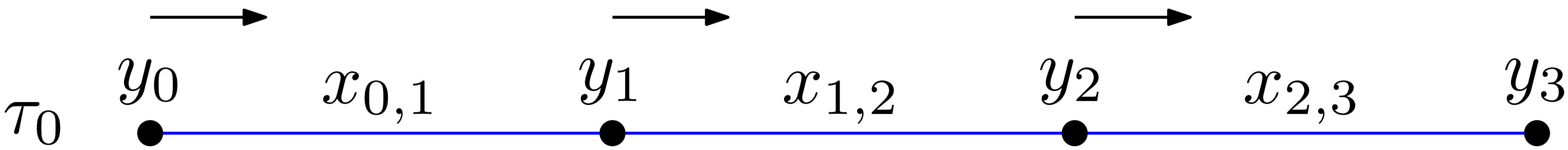
$$\forall \tau = (v_0, \ldots, v_l) \in \mathcal{T}:$$

$$\forall i \in 0, \ldots, l-1: \quad \begin{cases} x_{\tau, v_i v_{i+1}} \leq y_i & \text{if } i = 0, \\ x_{\tau, v_i v_{i+1}} \leq y_i + x_{\tau, v_{i-1} v_i} & \text{else} \end{cases} \qquad \text{(Constraint 2)}$$

$$\forall i \in 1, \ldots, l: \quad \begin{cases} x_{\tau, v_{i-1} v_i} \leq y_i & \text{if } i = l, \\ x_{\tau, v_{i-1} v_i} \leq y_i + x_{\tau, v_i v_{i+1}} & \text{else} \end{cases} \qquad \text{(Constraint 3)}$$

$$\forall v \in V, \tau \in e \in \tau: \quad x_{\tau,e}, y_v \in \{0, 1\}$$

Integer Program Formulation



$y_0 \geq x_{0,1}$ $\quad x_{0,1} + y_1 \geq x_{1,2}$ $\quad x_{1,2} + y_2 \geq x_{2,3}$

Generalization:

$$\forall \tau = (v_0, \ldots, v_l) \in \mathcal{T} :$$

$$\forall i \in 0, \ldots, l-1 : \quad \begin{cases} x_{\tau, v_i v_{i+1}} \leq y_i & \text{if } i = 0, \\ x_{\tau, v_i v_{i+1}} \leq y_i + x_{\tau, v_{i-1} v_i} & \text{else} \end{cases}$$

Integer Program Formulation

$$\max \sum_{\tau \in \mathcal{T}, e \in E(\tau)} f(e) x_{\tau,e}$$

$$\sum_{v \in V} y_v \leq k \qquad \text{(Constraint 1)}$$

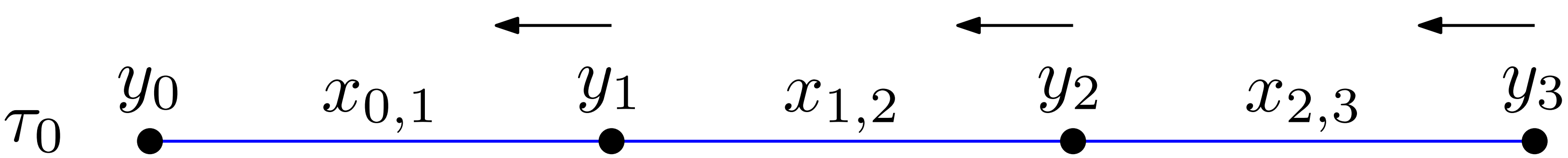
$$\forall \tau = (v_0, \dots, v_l) \in \mathcal{T}:$$

$$\forall i \in 0, \dots, l-1: \begin{cases} x_{\tau, v_i v_{i+1}} \leq y_i & \text{if } i = 0, \\ x_{\tau, v_i v_{i+1}} \leq y_i + x_{\tau, v_{i-1} v_i} & \text{else} \end{cases} \qquad \text{(Constraint 2)}$$

$$\forall i \in 1, \dots, l: \begin{cases} x_{\tau, v_{i-1} v_i} \leq y_i & \text{if } i = l, \\ x_{\tau, v_{i-1} v_i} \leq y_i + x_{\tau, v_i v_{i+1}} & \text{else} \end{cases} \qquad \text{(Constraint 3)}$$

$$\forall v \in V, \tau \in e \in \tau: \quad x_{\tau,e}, y_v \in \{0, 1\}$$

Integer Program Formulation



$x_{0,1} \leq y_1 + x_{1,2}$ $\quad$ $x_{1,2} \leq y_2 + x_{2,3}$ $\quad$ $x_{2,3} \leq y_3$

Generalization:

$$\forall \tau = (v_0, \ldots, v_l) \in \mathcal{T}:$$

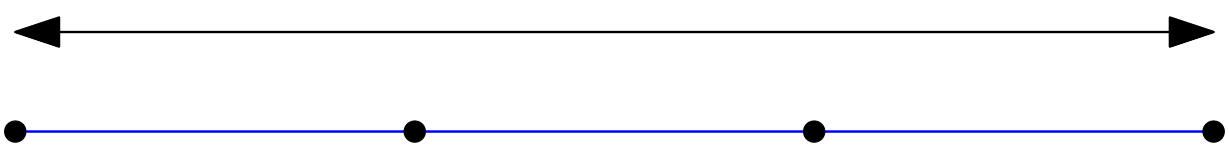
$$\forall i \in 1, \ldots, l: \qquad \begin{cases} x_{\tau, v_{i-1} v_i} \leq y_i & \text{if } i = l, \\ x_{\tau, v_{i-1} v_i} \leq y_i + x_{\tau, v_i v_{i+1}} & \text{else} \end{cases}$$

Integer Program Formulation

$$\max \sum_{\tau \in \mathcal{T}, e \in E(\tau)} f(e) x_{\tau,e}$$

$$\sum_{v \in V} y_v \leq k \qquad \text{(Constraint 1)}$$

$$\begin{array}{ll}
\forall \tau = (v_0, \ldots, v_l) \in \mathcal{T}: & \\
\quad \forall i \in 0, \ldots, l-1: & \begin{cases} x_{\tau, v_i v_{i+1}} \leq y_i & \text{if } i = 0, \\ x_{\tau, v_i v_{i+1}} \leq y_i + x_{\tau, v_{i-1} v_i} & \text{else} \end{cases} \qquad \text{(Constraint 2)} \\
\quad \forall i \in 1, \ldots, l: & \begin{cases} x_{\tau, v_{i-1} v_i} \leq y_i & \text{if } i = l, \\ x_{\tau, v_{i-1} v_i} \leq y_i + x_{\tau, v_i v_{i+1}} & \text{else} \end{cases} \qquad \text{(Constraint 3)} \\
\forall v \in V, \tau \in e \in \tau: & x_{\tau,e}, y_v \in \{0, 1\}
\end{array}$$



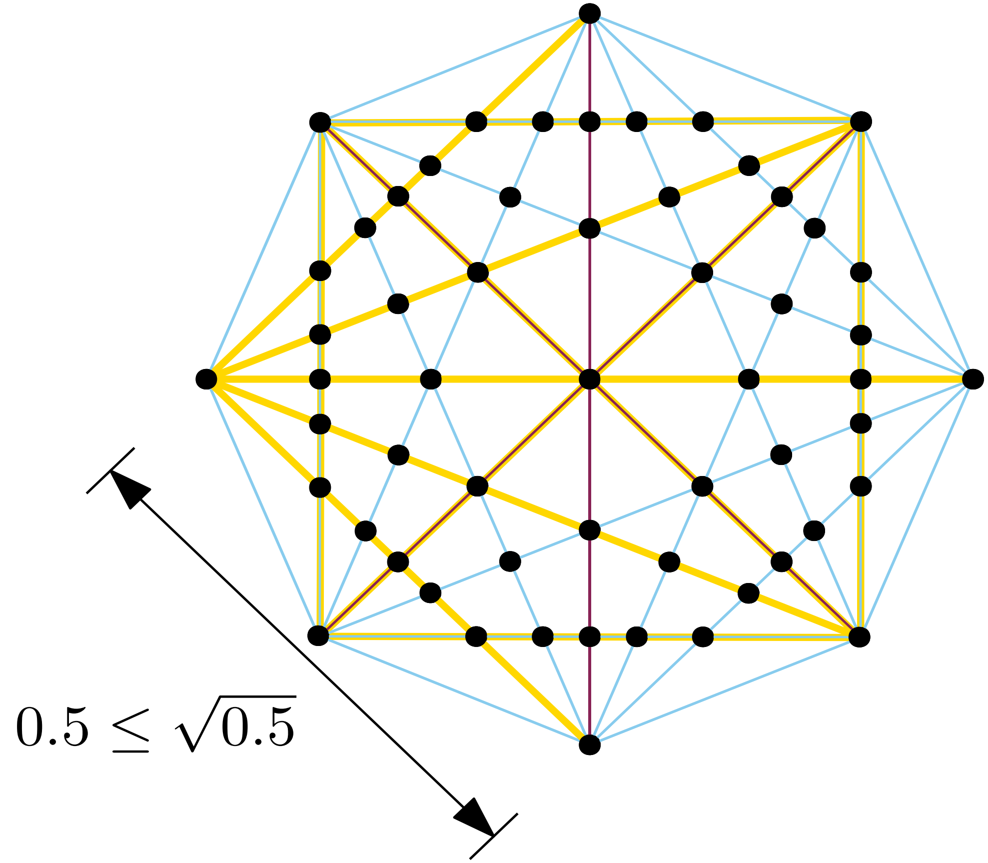
Sándor P. Fekete, **Alexander Hill**, Dominik Krupke, Tyler Mayer, Joseph S. B. Mitchell, Ojas Parekh and Cynthia A. Phillips
ahill@ibr.cs.tu-bs.de

Integrality Gap

$$\mathrm{OPT}_{INT}(k) \leq k^2$$

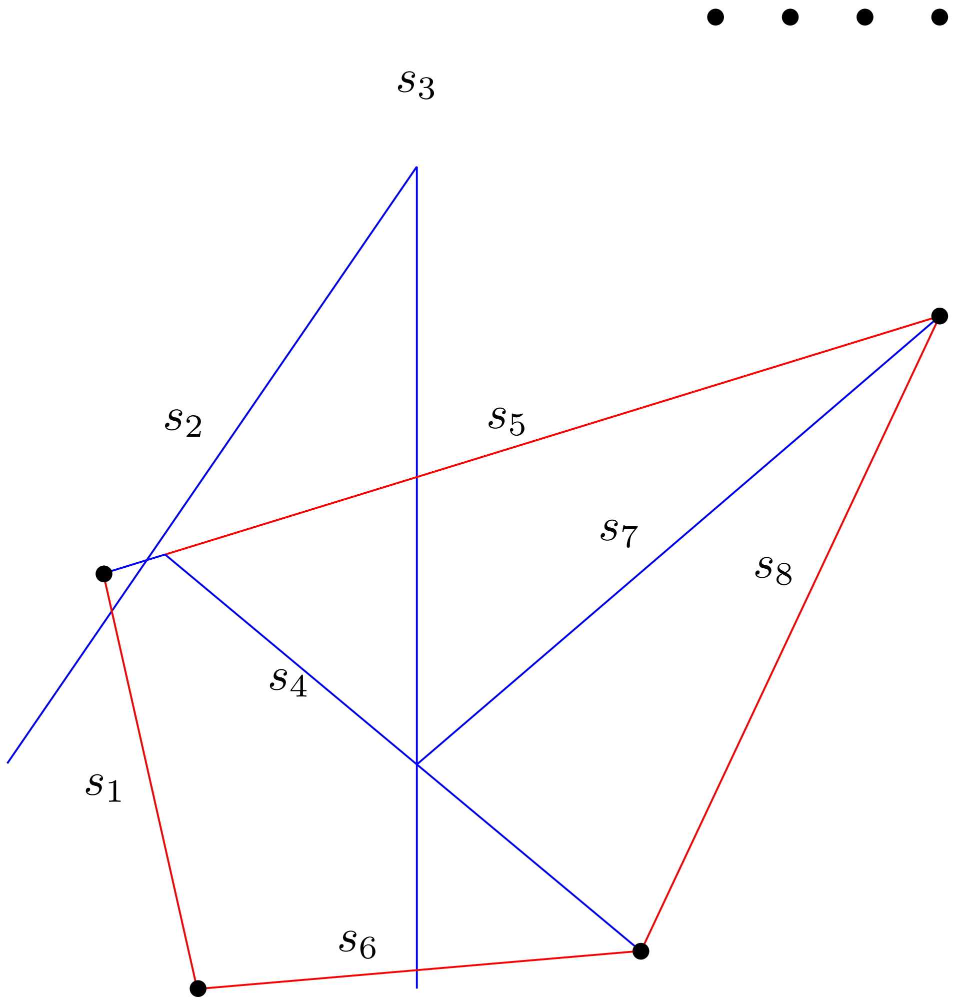
$$OPT_{FRAC}(k) \geq \frac{1}{k} \cdot \frac{n^2}{8}$$

$$\frac{OPT_{FRAC}(k)}{OPT_{INT}(k)} \geq \frac{n^2}{8k^3}$$

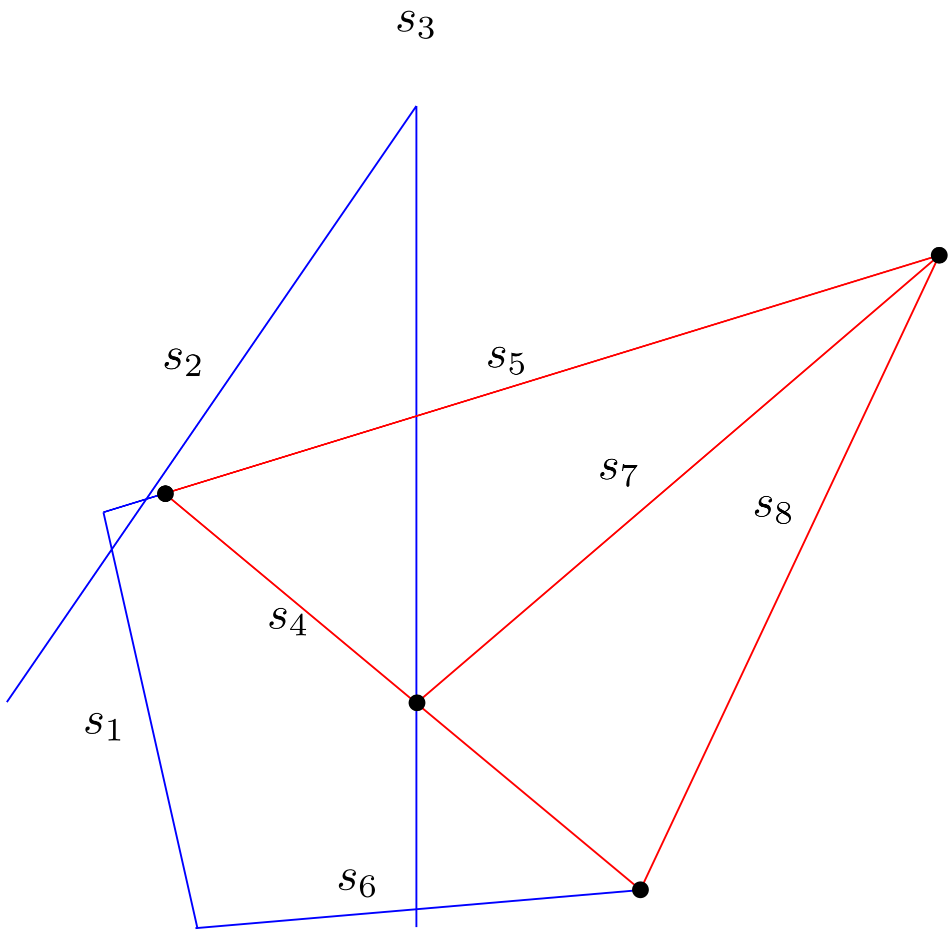


Overview

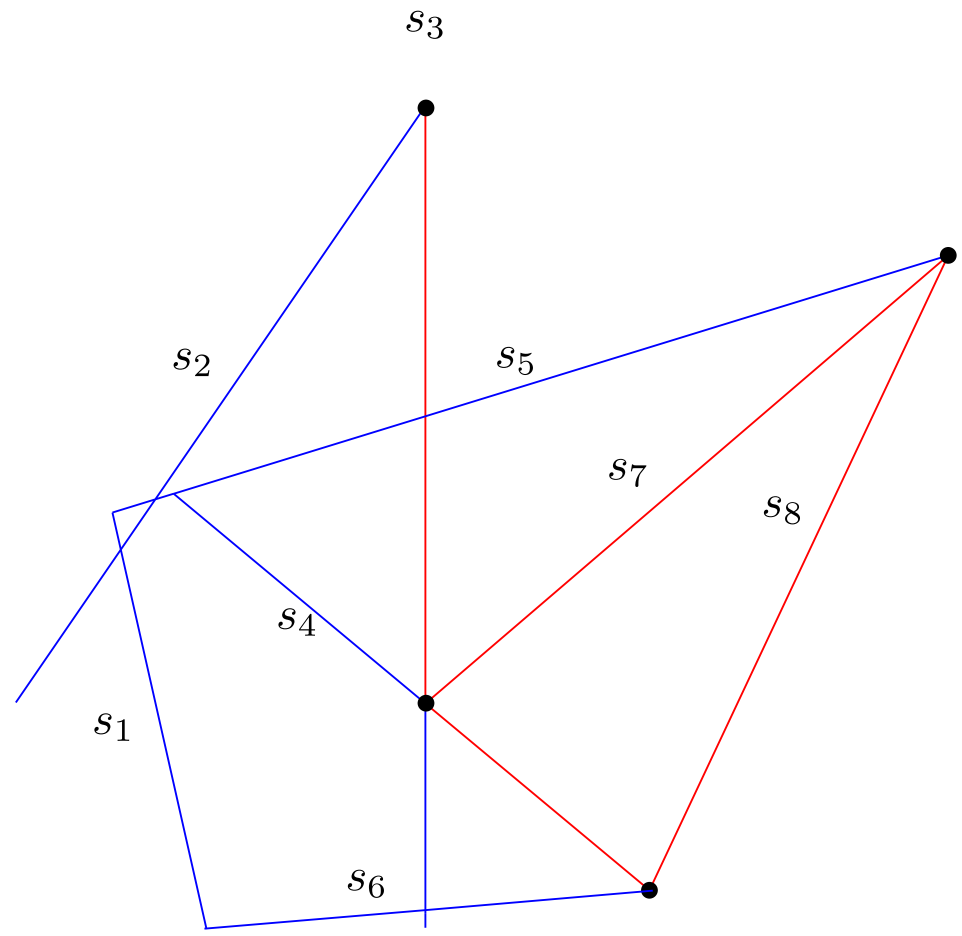
Heuristics: Greedy



Heuristics: Iterated Local Search

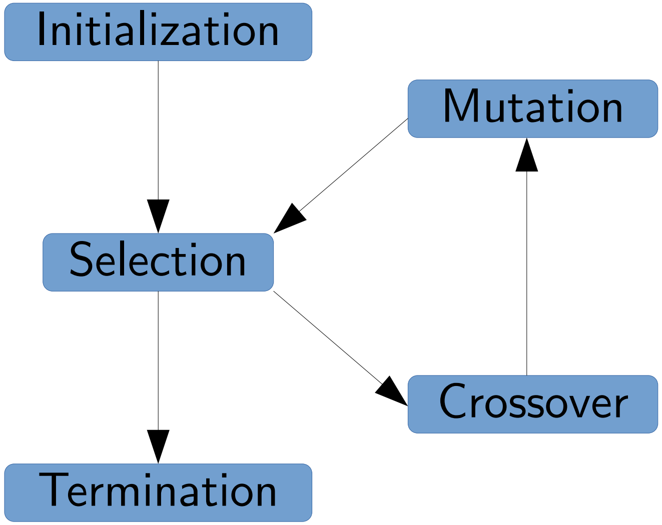


Heuristics: Simulated Annealing



Heuristics: Evolutionary Algorithms

- Initial solutions via greedy variant
- Create new solutions from previous
  - Uniform random recombination
  - Mutation: ILS or SA
- Fitness: Captured length
- Selection: Uniform random

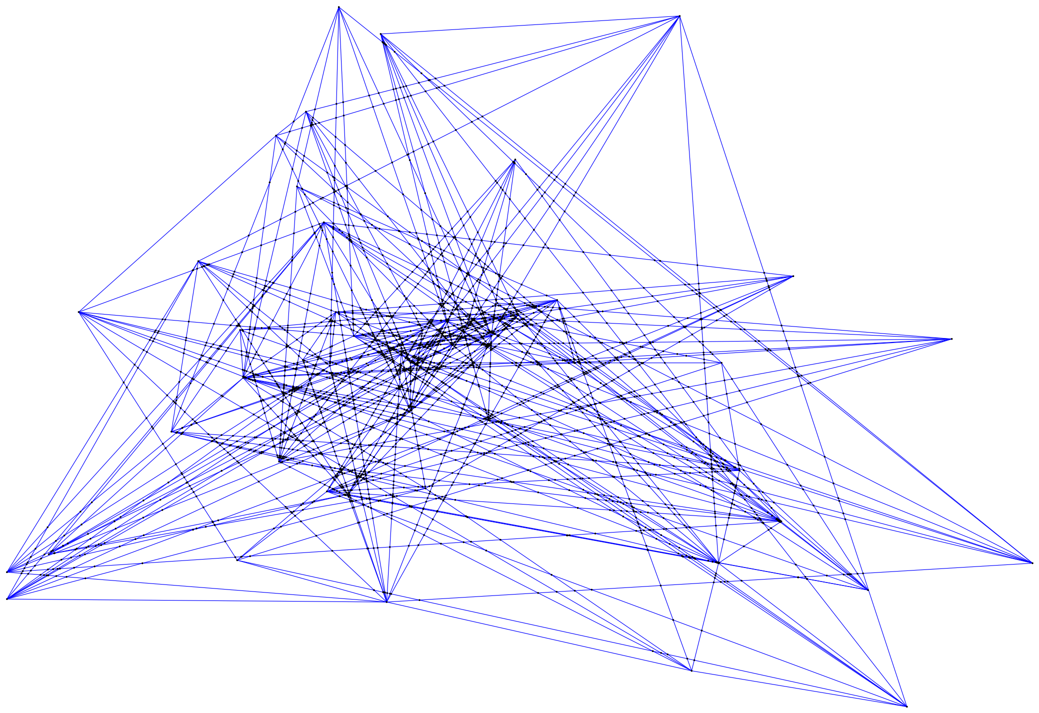


Overview

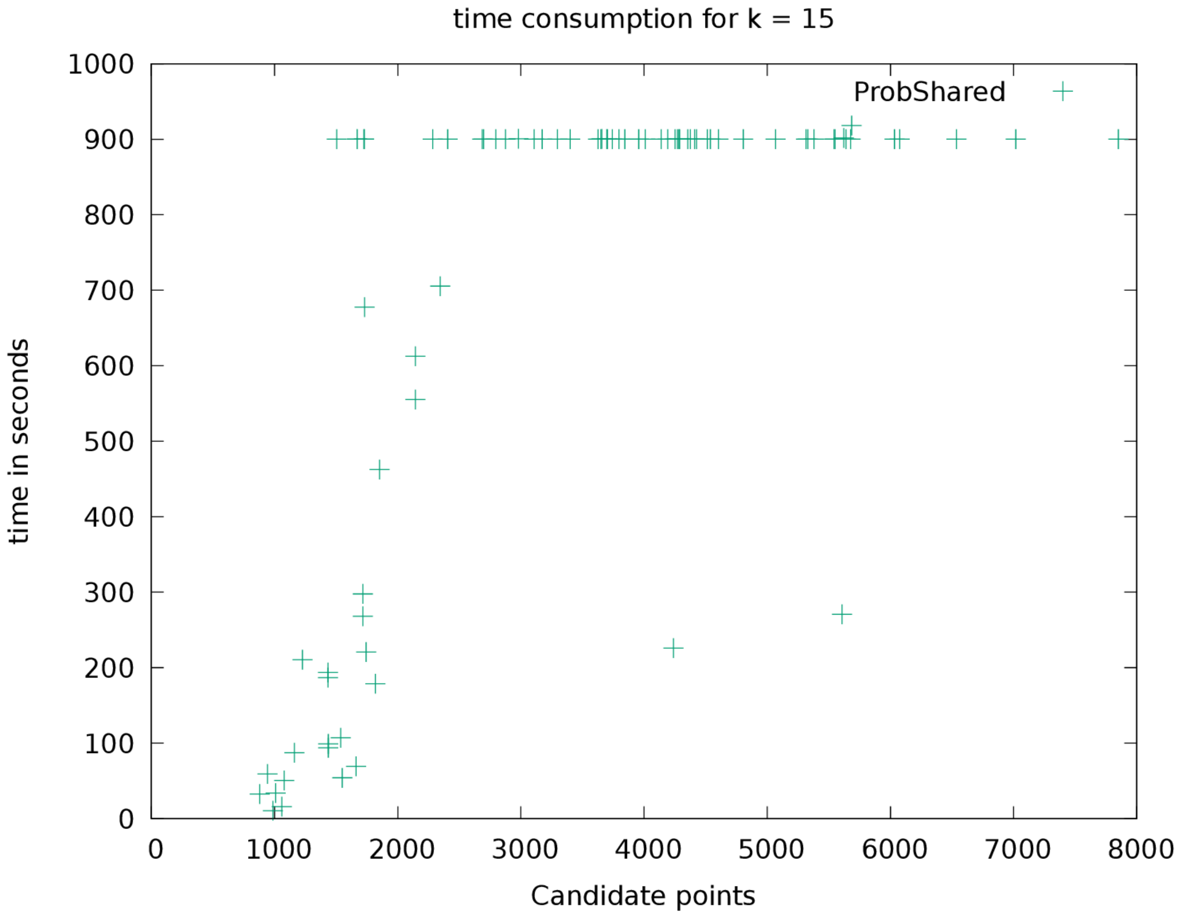
1. Trajectory Capturing Problem
2. Complexity
3. Approximation Schemes
4. Integer Programming
5. Heuristics
6. **Evaluation**
    1. **Instance Generation**
    2. **Integer Programming**
    3. **EA Mutation Strategies**
    4. **Heuristics Comparisons**
    5. **Taxi Data**
7. Conclusion

Instance generation

- Geometric instances
- Random segments easy
- Use seed points
  - Random points
  - TSP library
  - Light maps

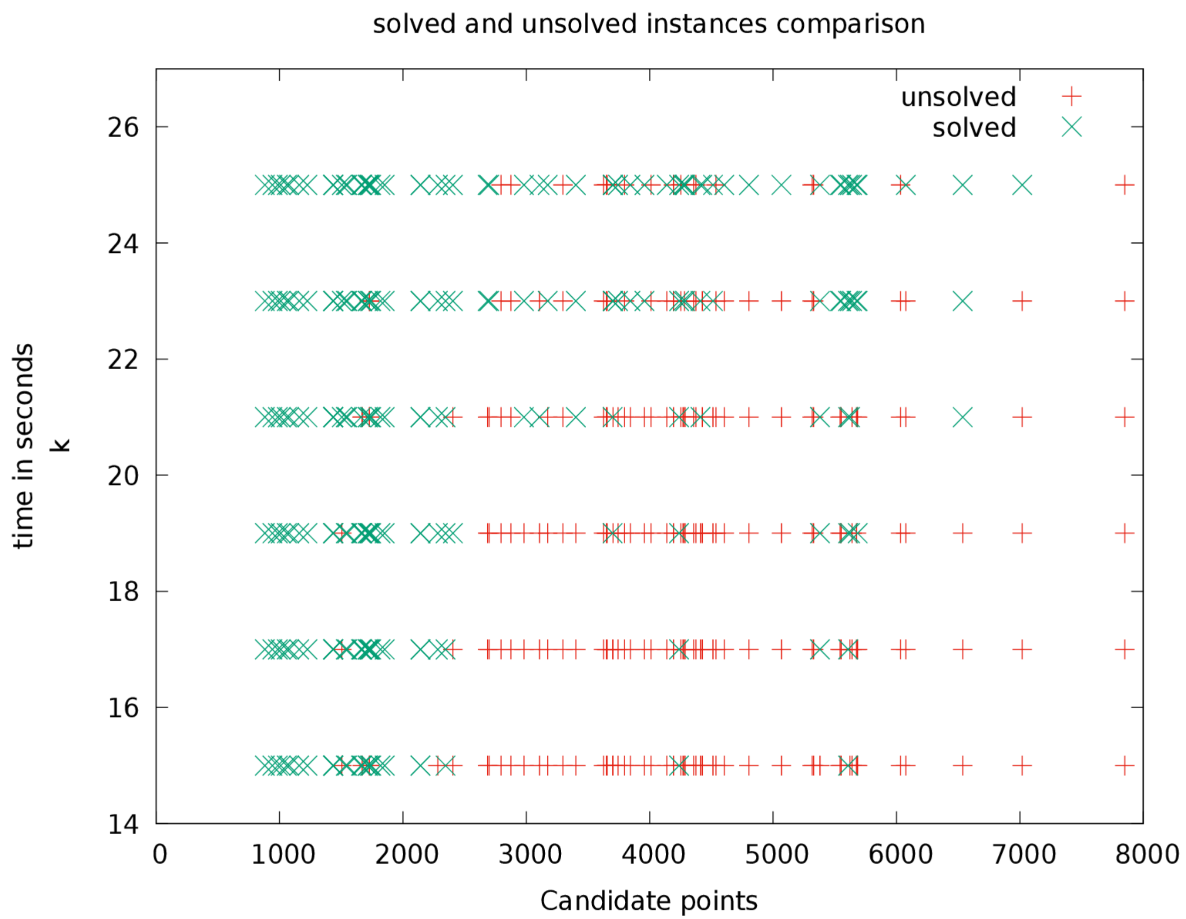


Evaluation: Integer Programming

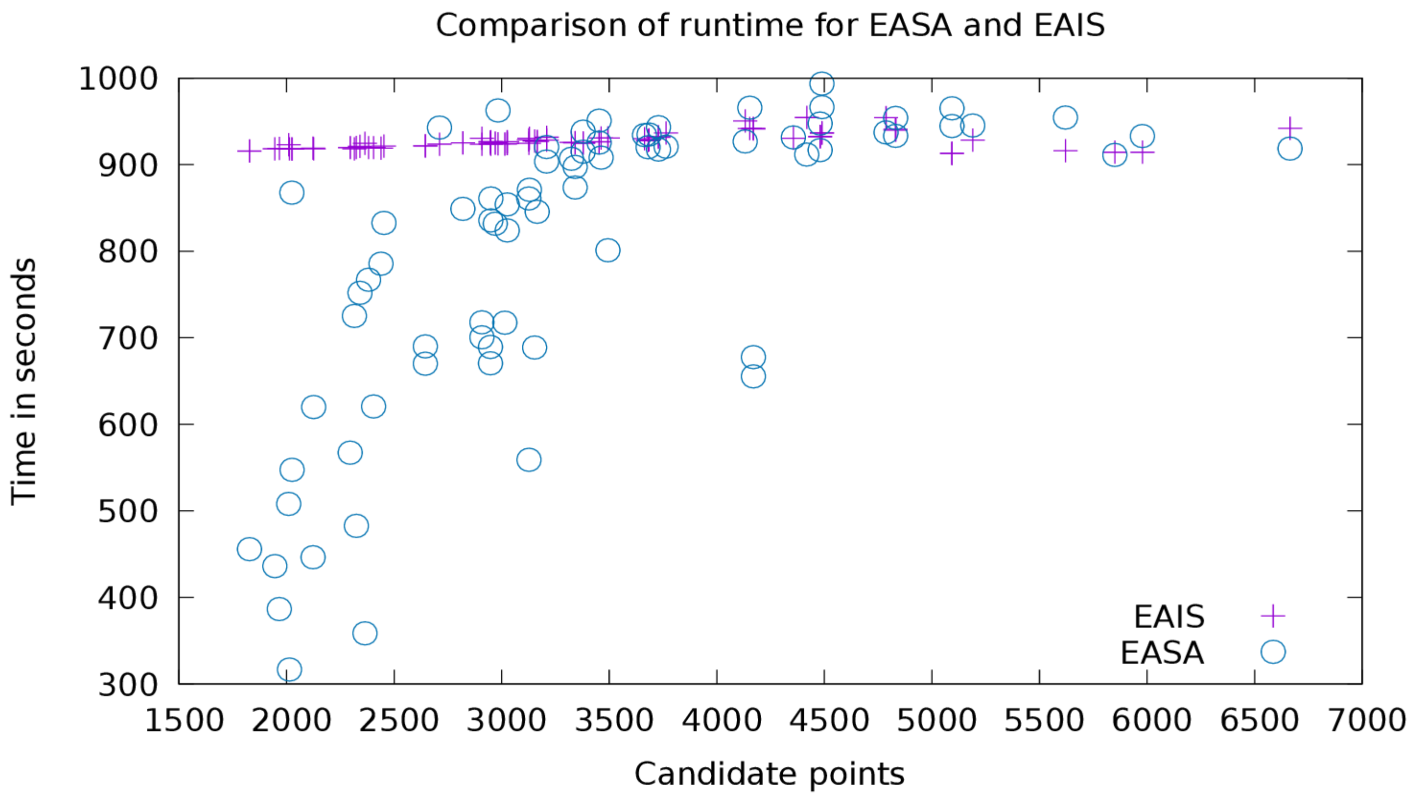


Technische Universität Braunschweig

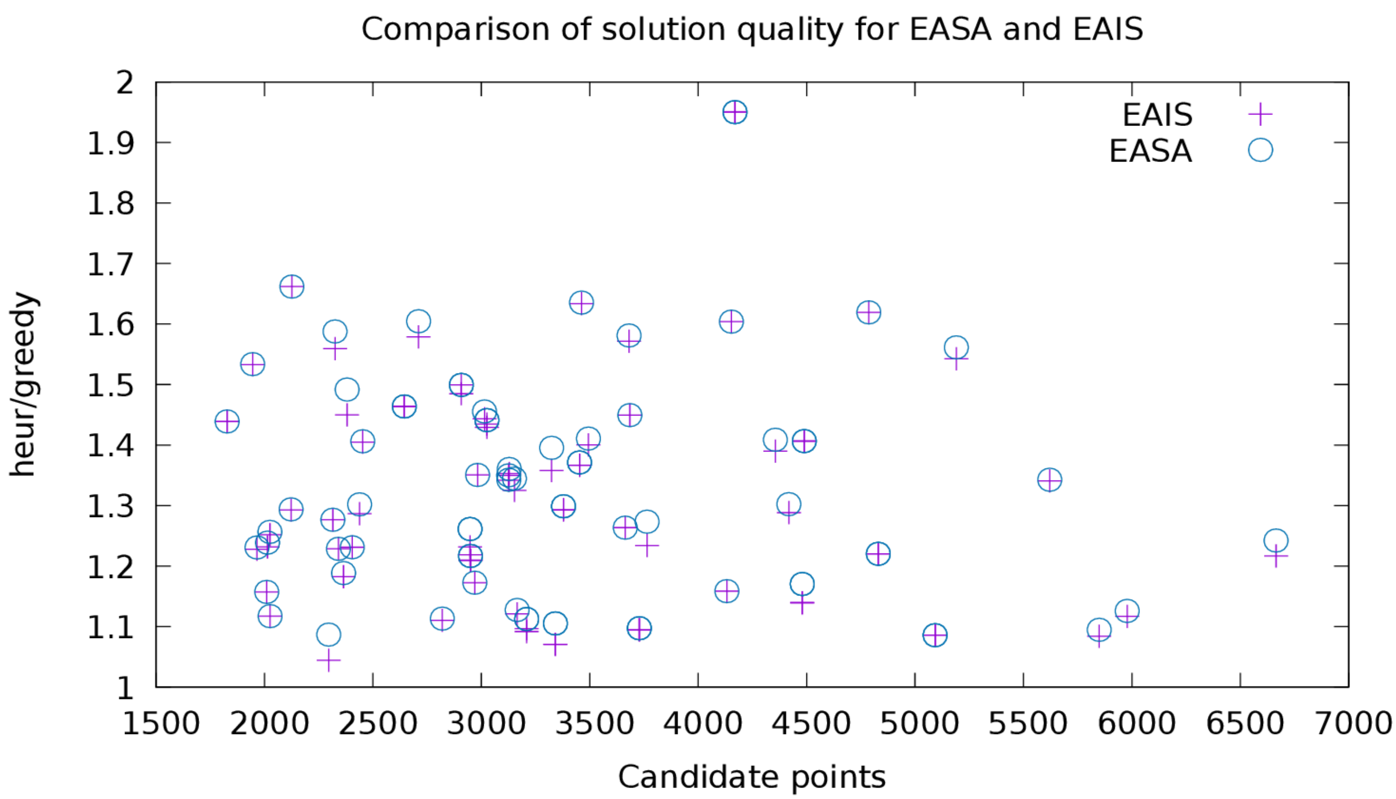
Evaluation: Integer Programming



Evaluation: Evolutionary Algorithm Mutation Strategy

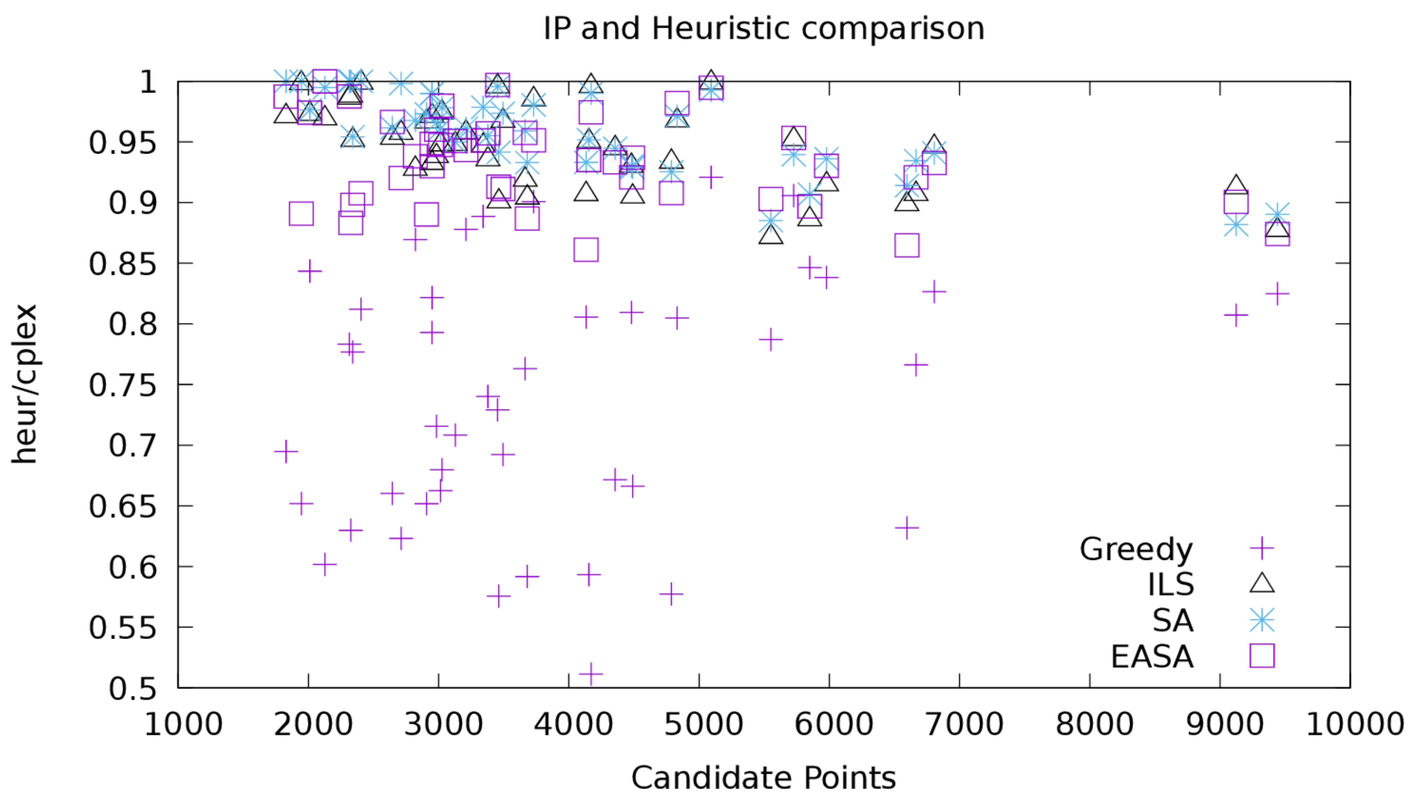


Evaluation: Evolutionary Algorithm Mutation Strategy

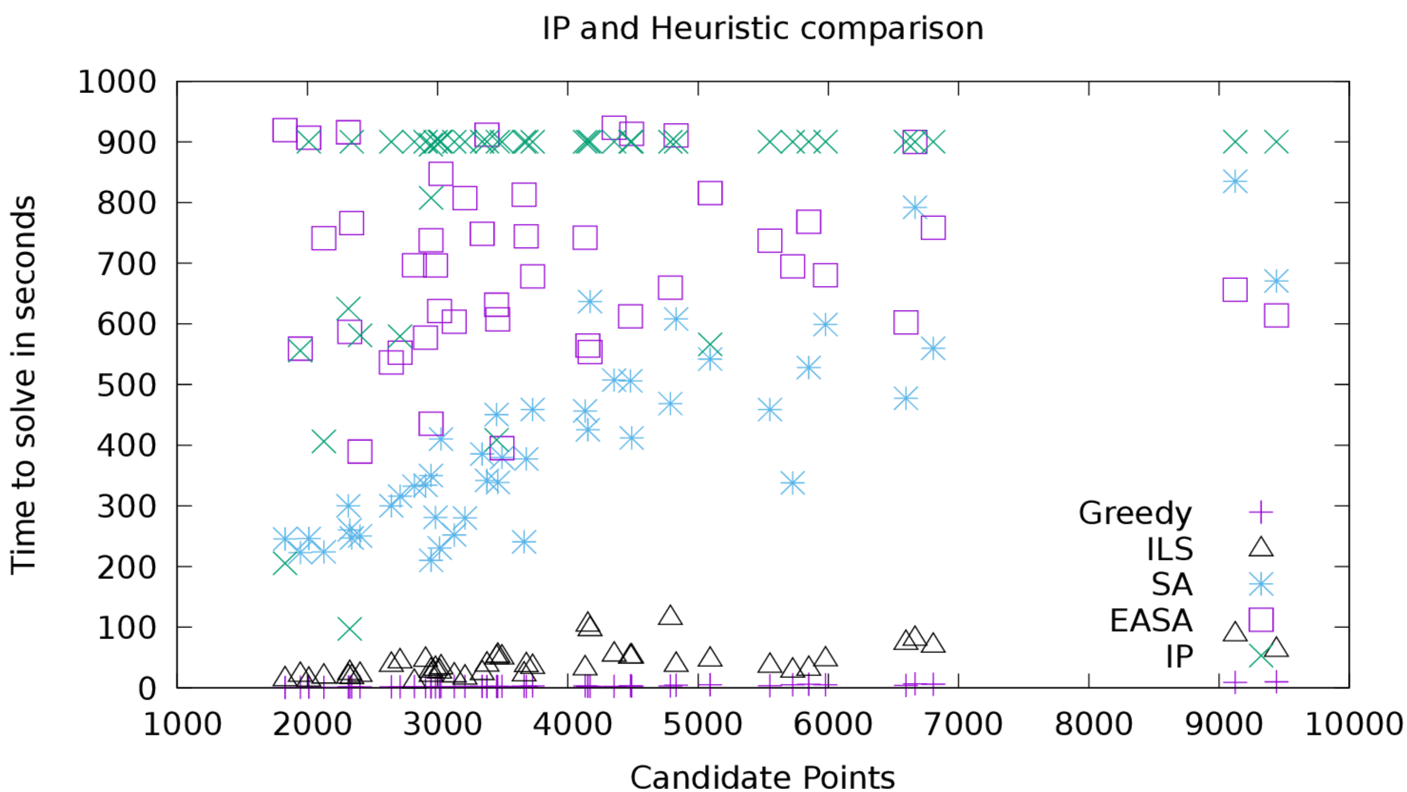


Technische Universität Braunschweig

Evaluation: Heuristics comparison



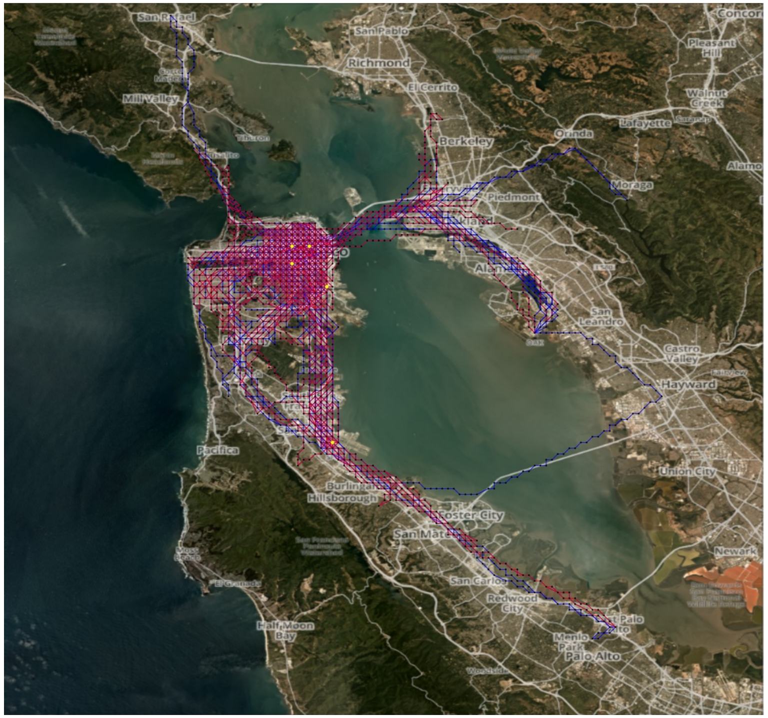
Evaluation: Heuristics comparison



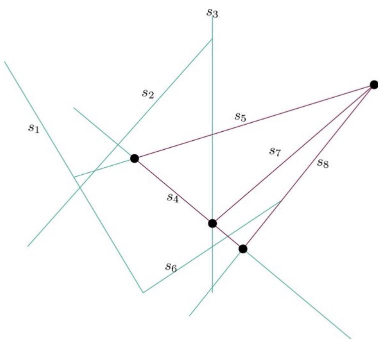
Evaluation: Real world data

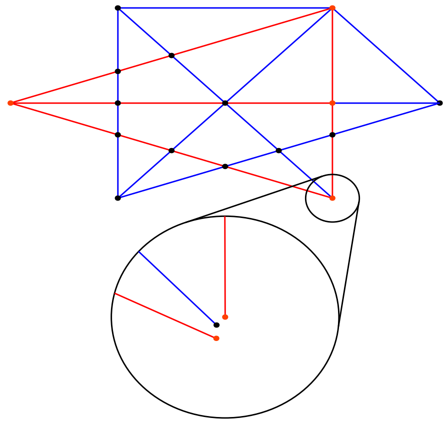
Sándor P. Fekete, **Alexander Hill**, Dominik Krupke, Tyler Mayer, Joseph S. B. Mitchell, Ojas Parekh and Cynthia A. Phillips
ahill@ibr.cs.tu-bs.de

Evaluation: Real world data

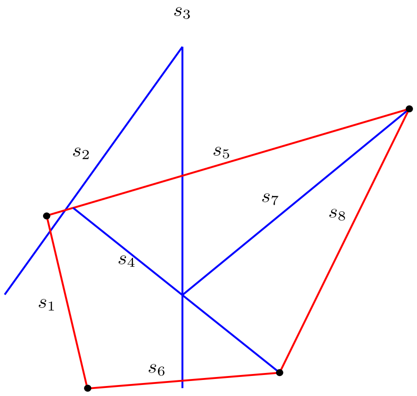


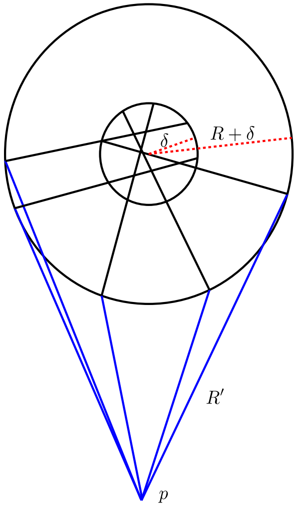
Conclusion





$\delta \leq \Delta$ clones

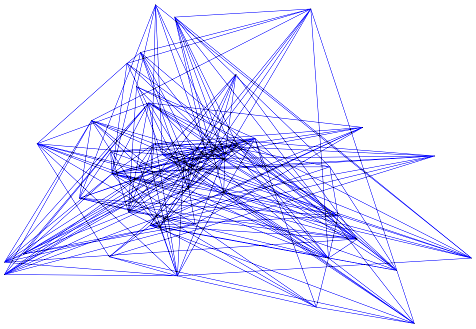




$$
\max \sum_{\tau \in \mathcal{T}, e \in E(\tau)} f(e) x_{\tau,e}
$$

$$
\sum_{v \in V} y_v \leq k
$$

$$
\begin{array}{ll}
\forall \tau = (v_0, \ldots, v_l) \in \mathcal{T}: & \\
\quad \forall i \in 0, \ldots, l-1: & \left\{ \begin{array}{lcll} x_{\tau, v_i v_{i+1}} & \leq & y_i & \text{if } i = 0, \\ x_{\tau, v_i v_{i+1}} & \leq & y_i + x_{\tau, v_{i-1} v_i} & \text{else} \end{array} \right. \\
\quad \forall i \in 1, \ldots, l: & \left\{ \begin{array}{lcll} x_{\tau, v_{i-1} v_i} & \leq & y_i & \text{if } i = l, \\ x_{\tau, v_{i-1} v_i} & \leq & y_i + x_{\tau, v_i v_{i+1}} & \text{else} \end{array} \right. \\
\forall v \in V, \tau \in e \in \tau: & \quad x_{\tau,e}, y_v \quad \in \quad \{0, 1\}
\end{array}
$$



Sándor P. Fekete, **Alexander Hill**, Dominik Krupke, Tyler Mayer, Joseph S. B. Mitchell, Ojas Parekh and Cynthia A. Phillips
ahill@ibr.cs.tu-bs.de

References

Slide 2: social network visualization by Martin Grandjean - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=48169324