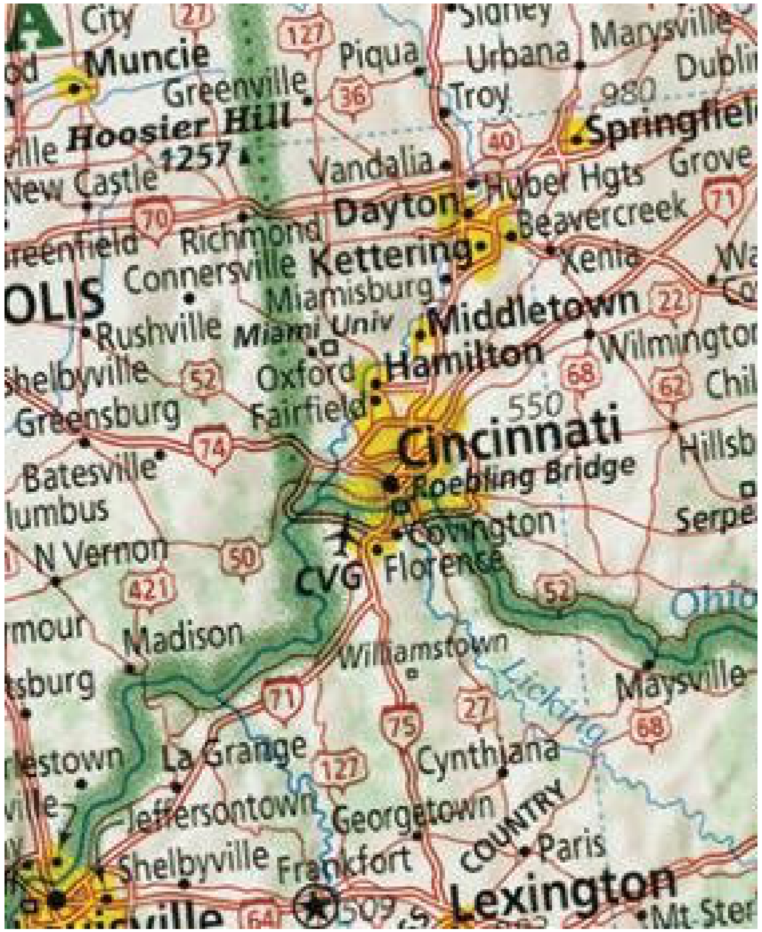
Algorithmen für Geographische Informationssysteme

9. Vorlesung
Boundary Labeling

Alexander Wolff

Folien von Filipp Kindermann

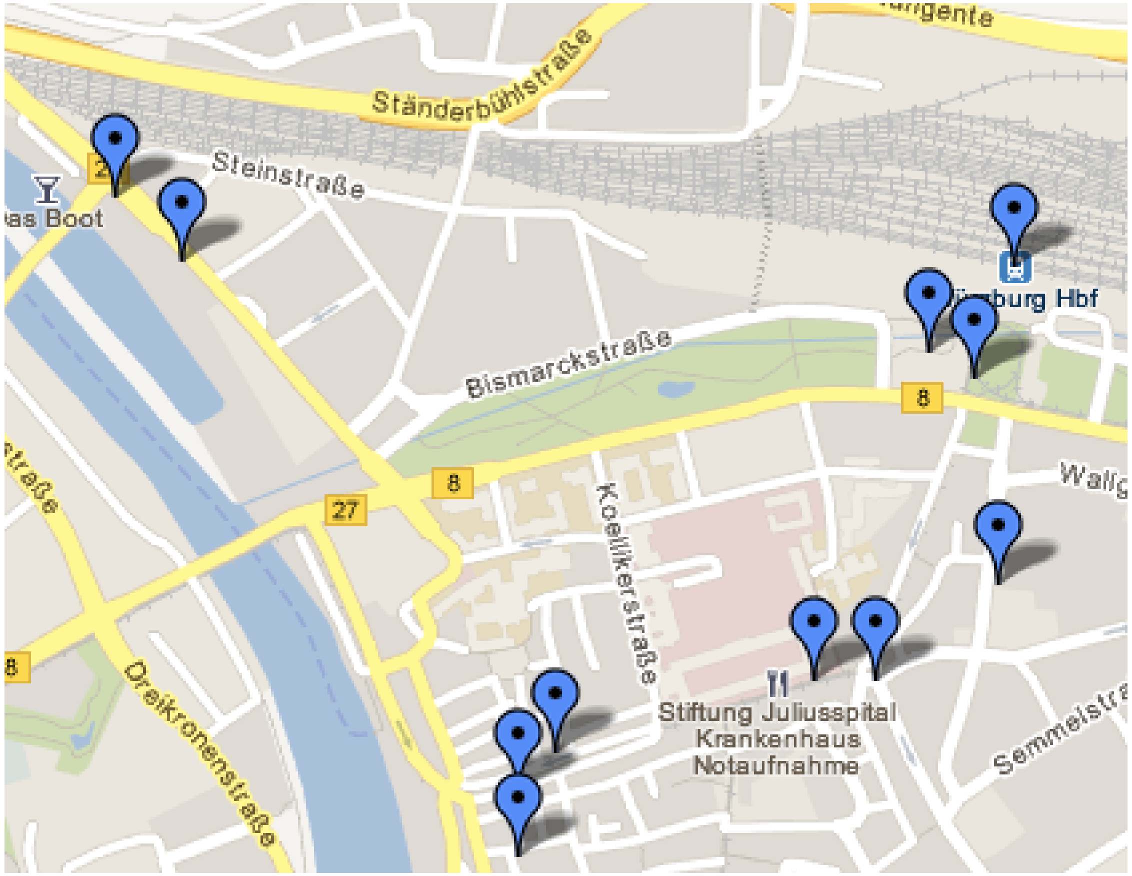
Kartenbeschriftungen



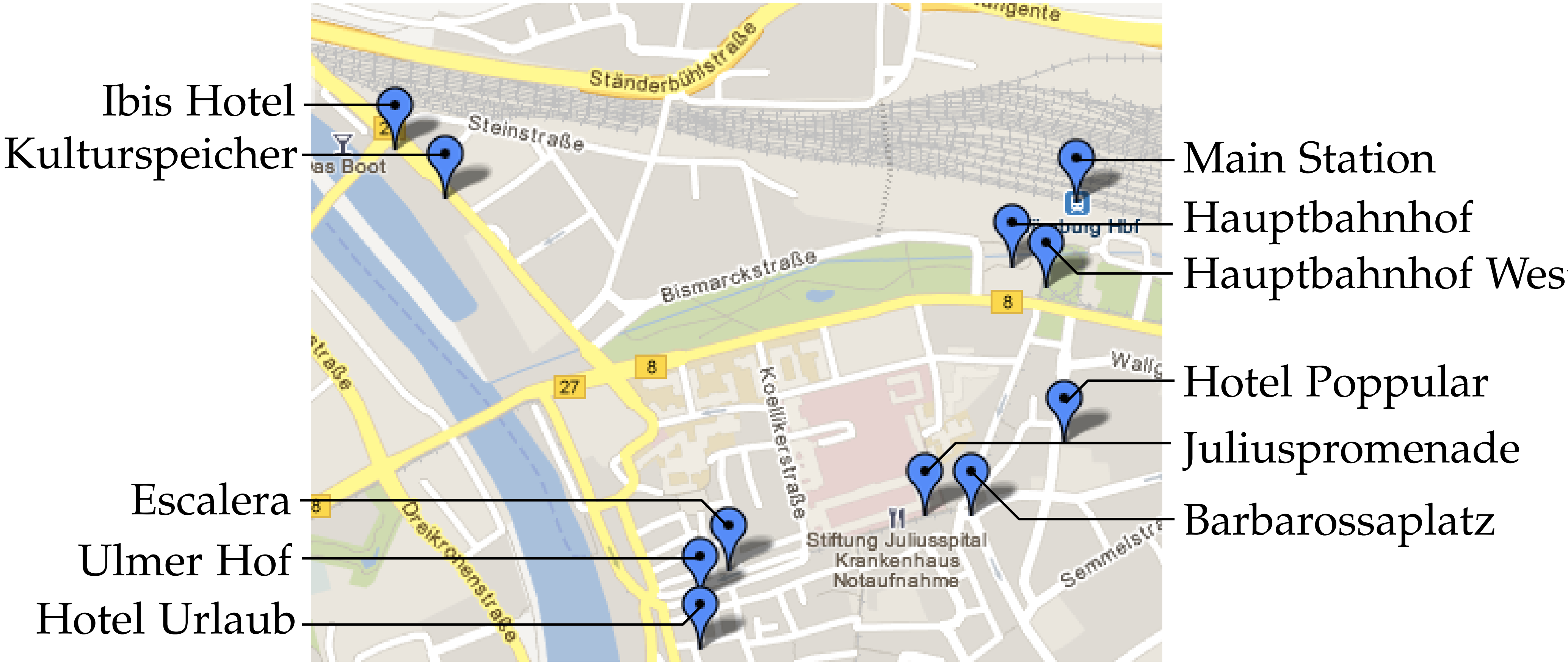
Eduard Imhof (Schiers 1895–1996 Erlenbach)

"Poor, sloppy, amateurisch type placement is irresponsible; it spoils even the best image and impedes reading."
[Eduard Imhof '75]

Randbeschriftungen



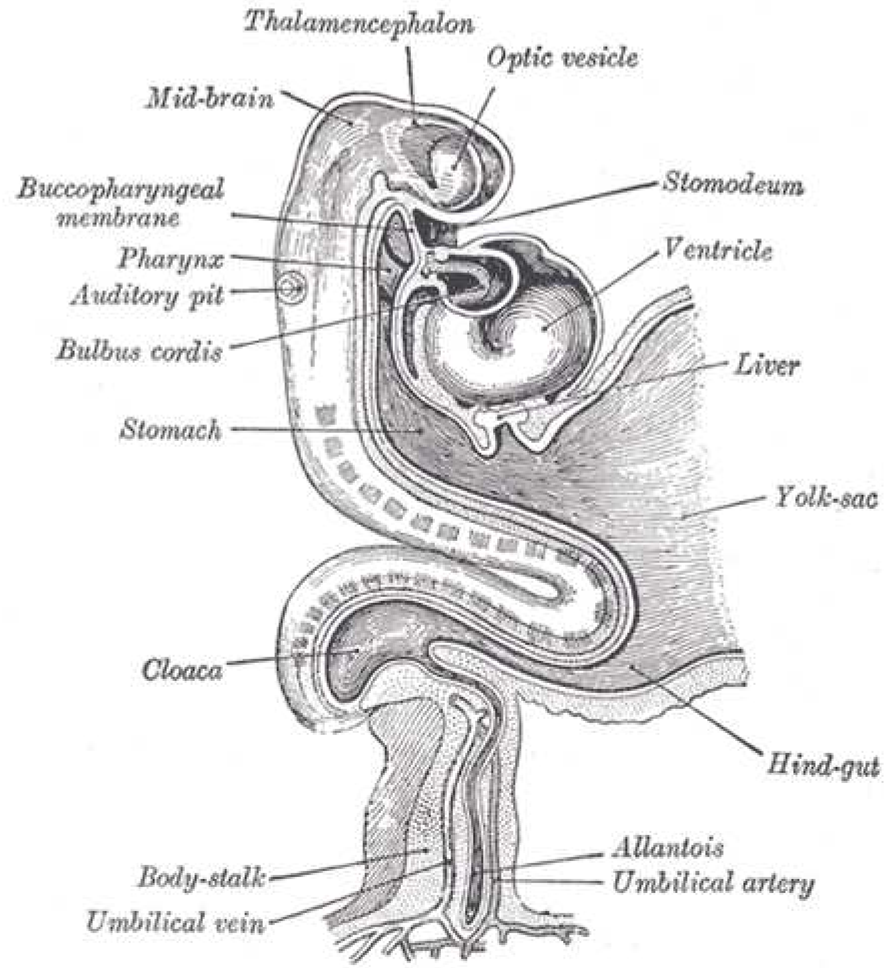
Randbeschriftungen



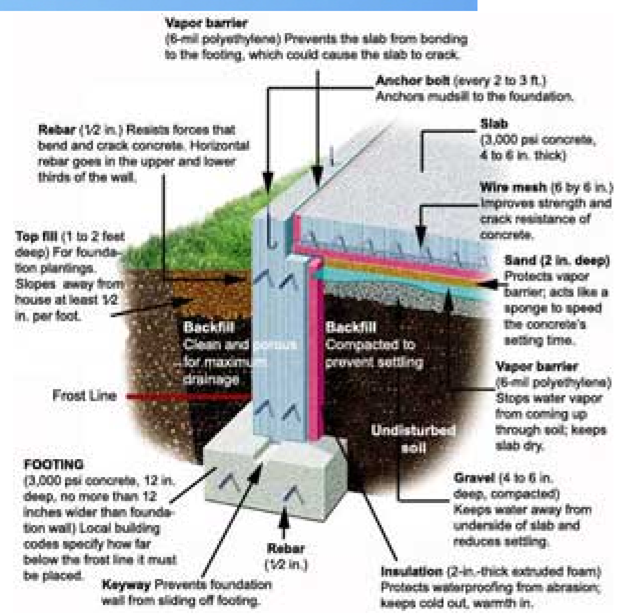
Vorteile von Randbeschriftungen

Randbeschriftungen

©DW-TV

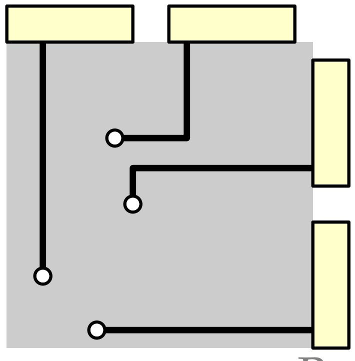


Henry Vandyke Carter, via Wikimedia Commons



©friv5games.me

Problembeschreibung

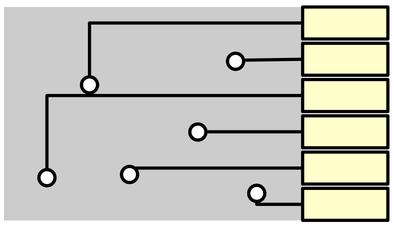


Geg.:
- Ein umschließendes Rechteck R
- n Punkte in R in allgemeiner Lage
- Je Punkt eine Beschriftung (achsen-paralleles Rechteck)

Ges.: Zulässige Platzierung der Beschriftungen am Rand von R,
leader → zulässige Verbindungen von Punkten zu Beschriftungen,
so dass Beschriftungsqualität optimiert wird.

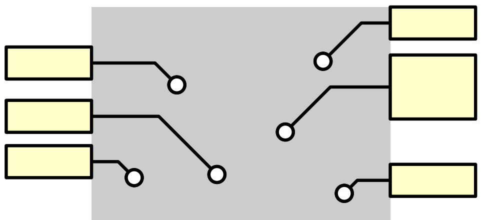
Beschriftungsqualität
- Min. Länge
- Min. Knicke
- Beliebige Funktion

1 Seite, uniform, fix



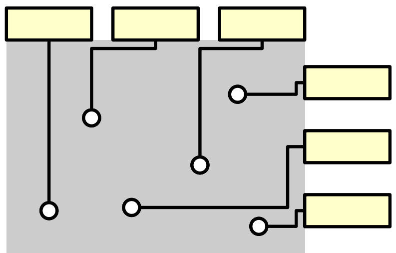
po, fix

2 Seiten (gegenüber), nicht uniform, nicht fix



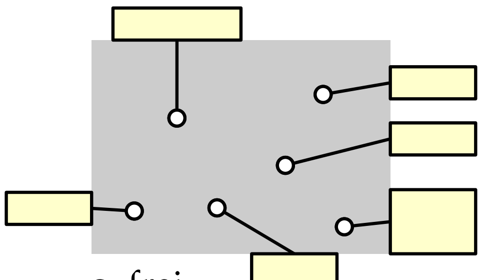
do, frei

2 Seiten (benachbart), uniform, fix



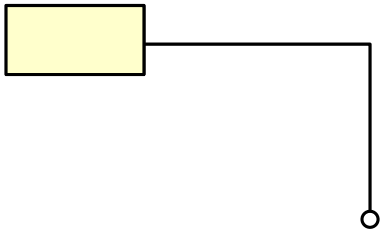
opo, fix

4 Seiten, nicht uniform, nicht fix



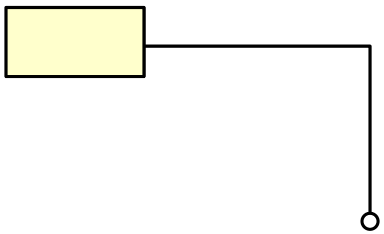
s, frei

po-Leader

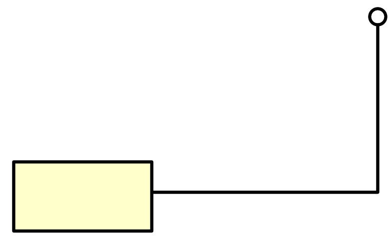


aufsteigender Leader

po-Leader

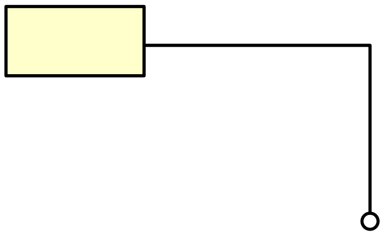


aufsteigender Leader

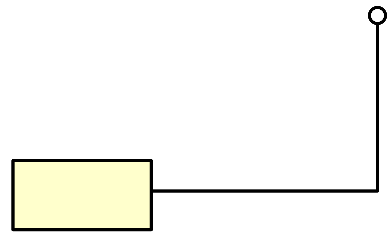


absteigender Leader

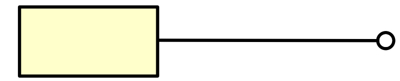
po-Leader



aufsteigender Leader



absteigender Leader



direkter Leader

Neuverdrahtung

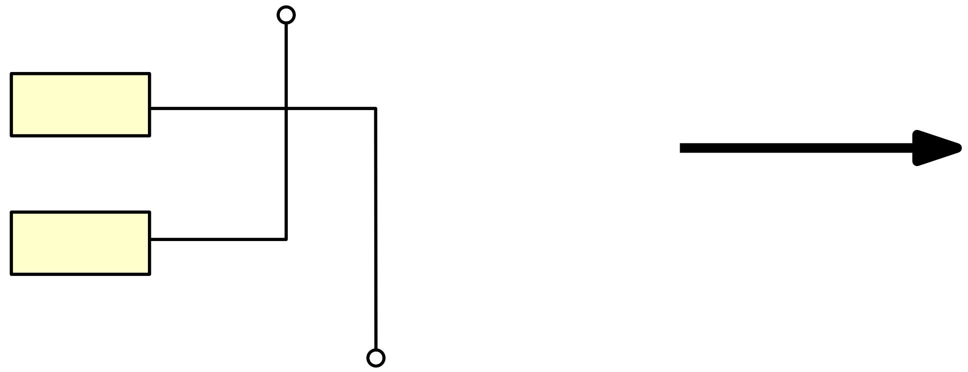
Geg.: Beschriftung minimaler Länge mit Kreuzungen.
Ges.: Kreuzungsfreie Beschriftung gleicher Länge.

Absteigende und aufsteigende Leader schneiden sich nicht.

Kein Leader schneidet absteigenden und aufsteigenden Leader.

$\Rightarrow$ Keine gemischten Kreuzungen.

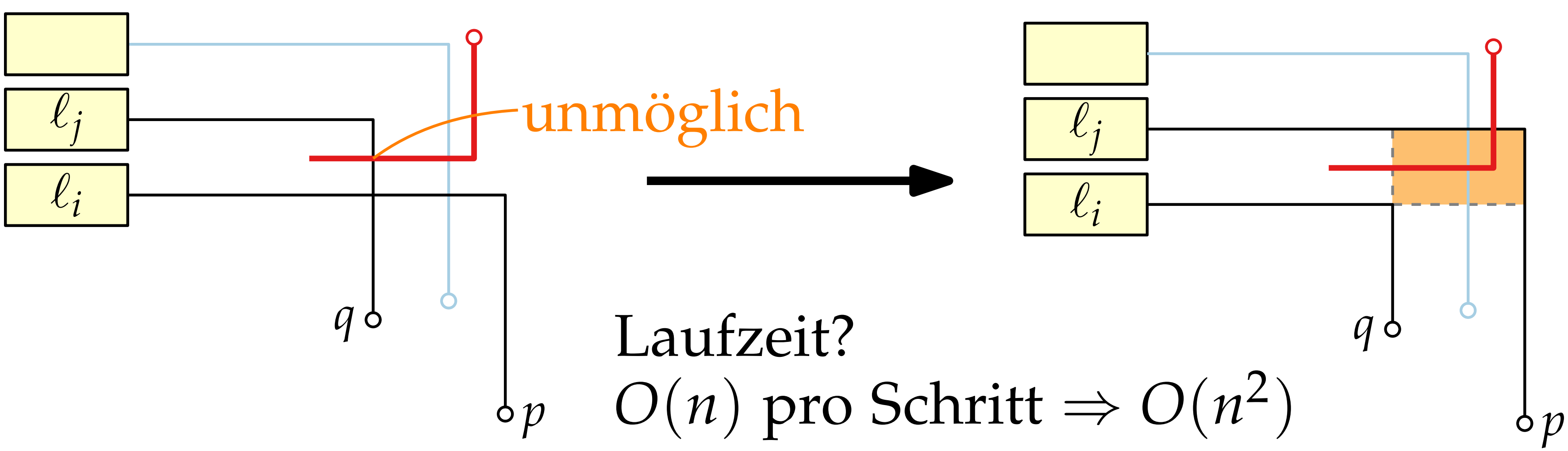
$\Rightarrow$ Absteigende und aufsteigende Kreuzungen disjunkt.



Übung!

Entfernung aufsteigender Kreuzungen

Sortiere Beschriftungen $\ell_1, \dots, \ell_n$ aufsteigend
for $i = 1$ **to** n **do**
    Sei $\lambda(p, \ell_i)$ Leader.
    if $\lambda(p, \ell_i)$ aufsteigend **then**
        Sei leader $\lambda(q, \ell_j)$ mit linkester Kreuzung mit $\lambda(p, \ell_i)$.
        Neuverdrahtung: $\lambda(q, \ell_i)$ und $\lambda(p, \ell_j)$.



Laufzeit?
$O(n)$ pro Schritt $\Rightarrow O(n^2)$

Induktion über i:
$j > i$ durch IA

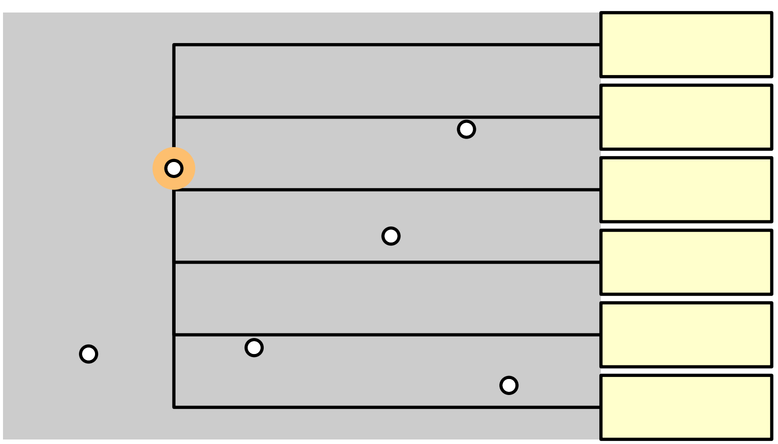
1. Gesamtlänge bleibt gleich ✓
2. Alle Kreuzungen nur oberhalb $\lambda(q, \ell_i)$ ✓
3. Keine absteigenden Kreuzungen ✓

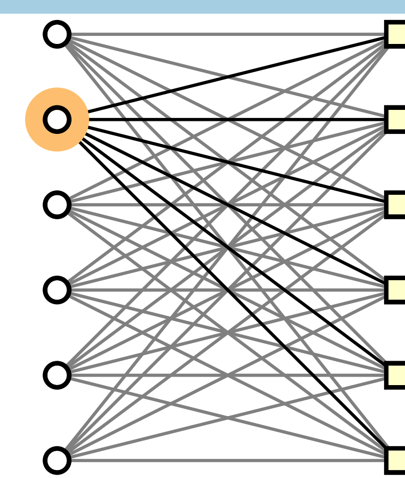
Längenminimierung

Lemma. Für jede Beschriftung minimaler Länge mit Kreuzungen im 1-seitigen po-Leader-Model mit fixen Ports kann eine kreuzungsfreie Beschriftung der gleichen Länge in $O(n^2)$ Zeit gefunden werden.

Lemma. Eine Beschriftung minimaler Länge mit Kreuzungen im 1-seitigen po-Leader-Model mit fixen Ports kann in $O(n^3)$ Zeit gefunden werden.

Satz. Eine kreuzungsfreie Beschriftung minimaler Länge im 1-seitigen po-Leader Model mit fixen Ports kann in $O(n^3)$ Zeit gefunden werden.

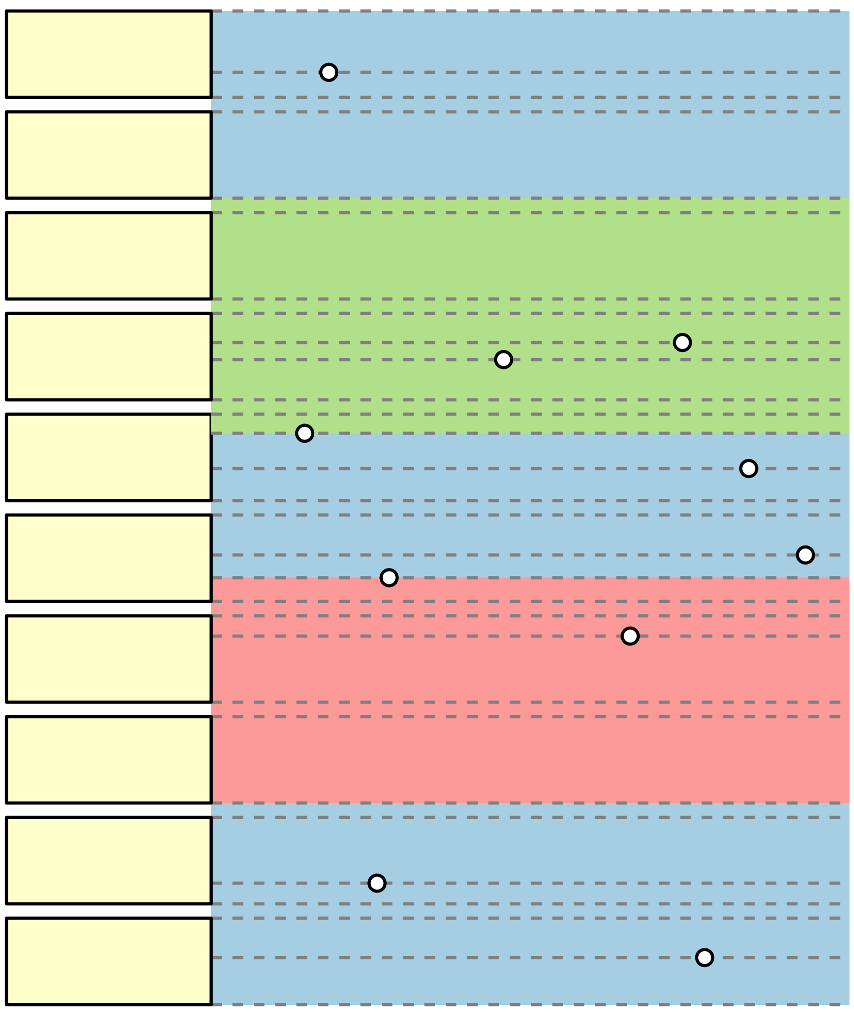




Minimum-weight bipartite matching!

Sweepline-Verfahren

Definiere $O(n)$ Streifen durch Punkte und Beschriftungen.
1. Unterteile Instanz in einfache Teilinstanzen.



Für jeden Streifen, zähle Anzahl Punkte und Beschriftungen im Streifen oder darunter. Ein Streifen ist...

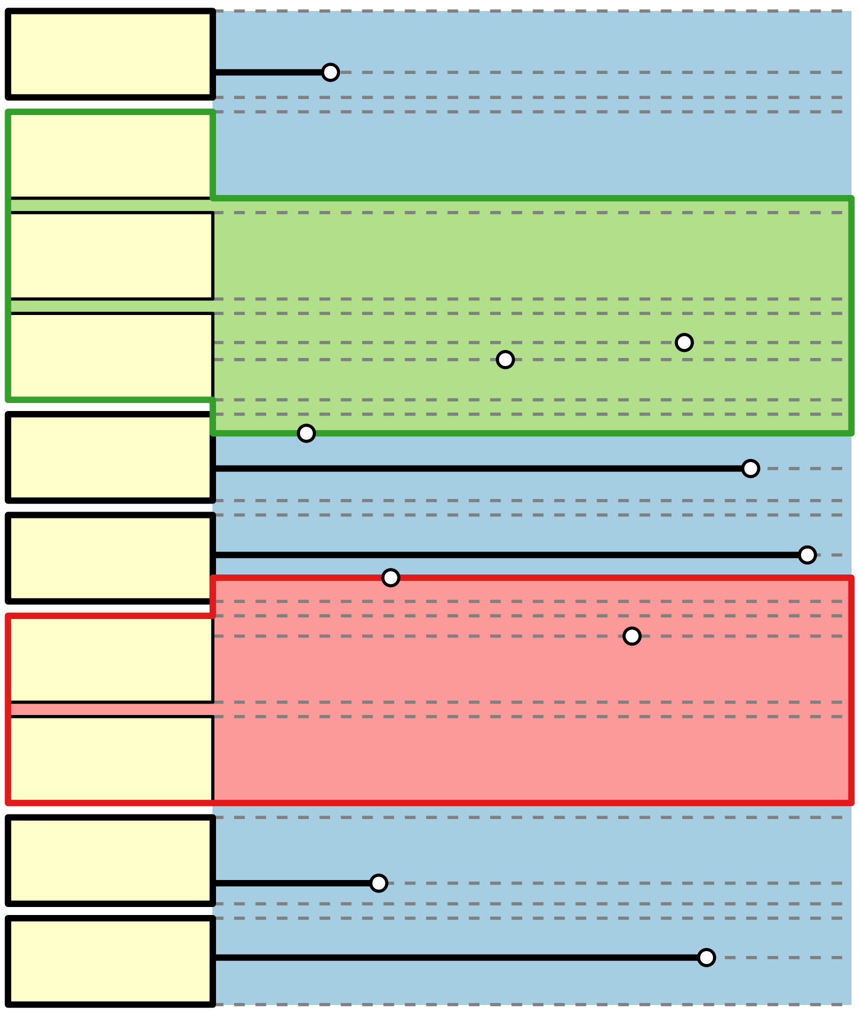
ohne Rand

- neutral, falls gleiche Anzahl.
- aufsteigend, falls mehr Punkte.
- absteigend, falls mehr Beschriftungen.

Sweepline-Verfahren

Definiere $O(n)$ Streifen durch Punkte und Beschriftungen.

1. Unterteile Instanz in einfache Teilinstanzen.
2. Löse Teilinstanzen.



Für jeden Streifen, zähle Anzahl Punkte und Beschriftungen im Streifen oder darunter. Ein Streifen ist...

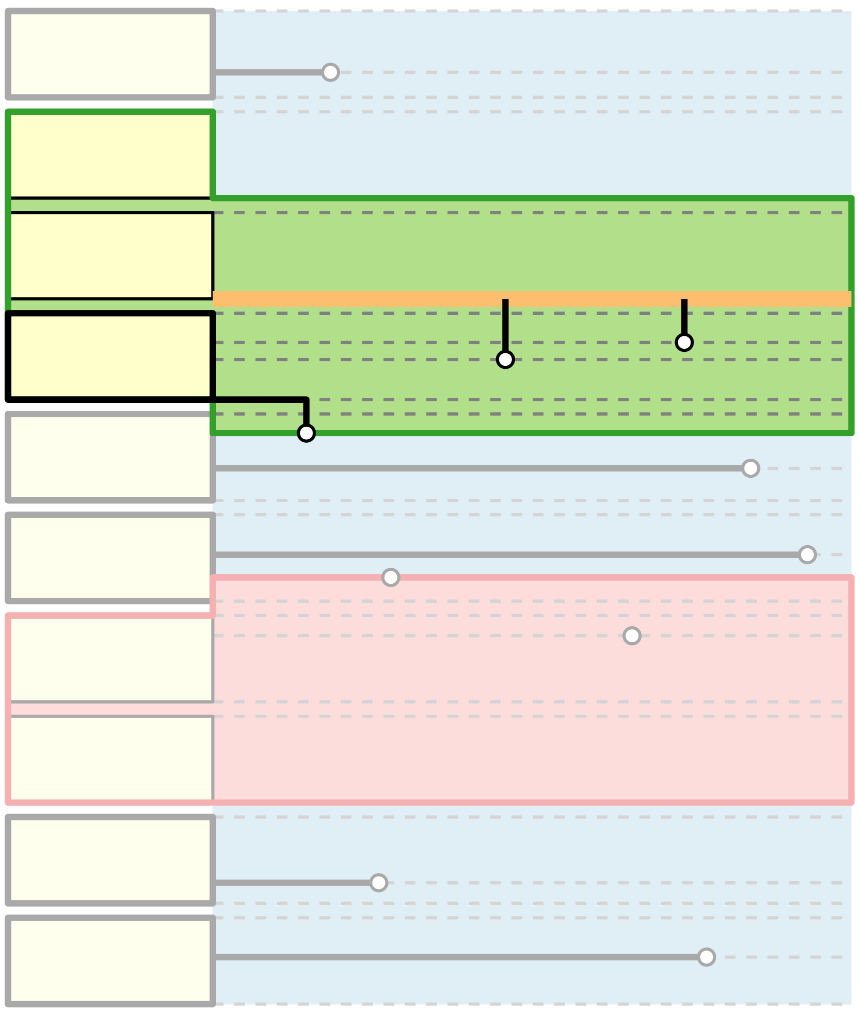
ohne Rand

- neutral, falls gleiche Anzahl.
  Verbinde Punkte horizontal.
- aufsteigend, falls mehr Punkte.
- absteigend, falls mehr Beschriftungen.

Sweepline-Verfahren

Definiere $O(n)$ Streifen durch Punkte und Beschriftungen.

1. Unterteile Instanz in einfache Teilinstanzen.
2. Löse Teilinstanzen.



Für jeden Streifen, zähle Anzahl Punkte und Beschriftungen im Streifen oder darunter. Ein Streifen ist...

ohne Rand

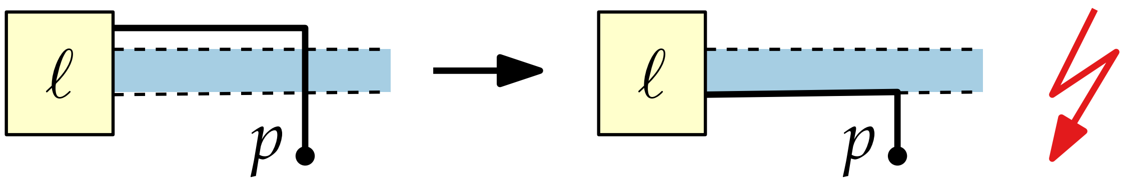
- neutral, falls gleiche Anzahl.
  Verbinde Punkte horizontal.
- aufsteigend, falls mehr Punkte.
  Sweepline: *von unten nach oben*
  Verbinde linkesten Punkt
  mit unterster Beschriftung.
- absteigend, falls mehr Beschriftungen.
  Sweepline: *von oben nach unten*
  Verbinde linkesten Punkt
  mit oberster Beschriftung.

Sweepline-Verfahren

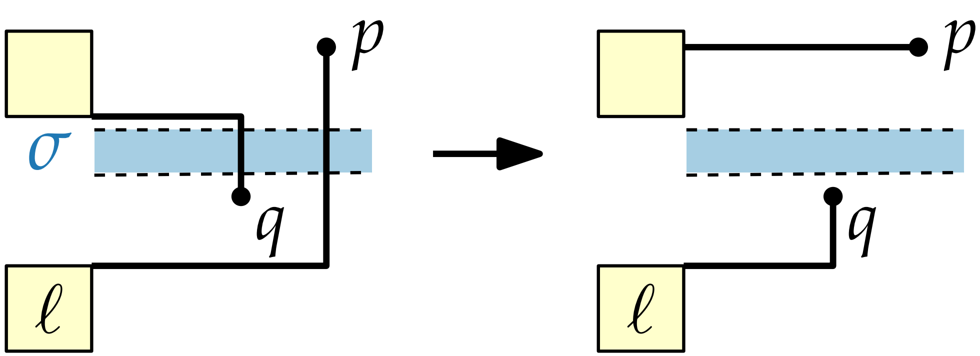
Lemma. Für jede kreuzungsfreie Zuordnung minimaler Länge gilt, dass kein Leader einen neutralen Streifen kreuzt.

Beweis. Angenommen Leader (p, ℓ) kreuzt neutralen Steifen σ.

1. Fall. ℓ umschließt σ



2. Fall. p oberhalb, ℓ unterhalb von σ.



\# Punkte = \# Beschriftungen bis σ
$\Rightarrow \exists q$ in oder unter σ
mit Beschriftung über σ ↯

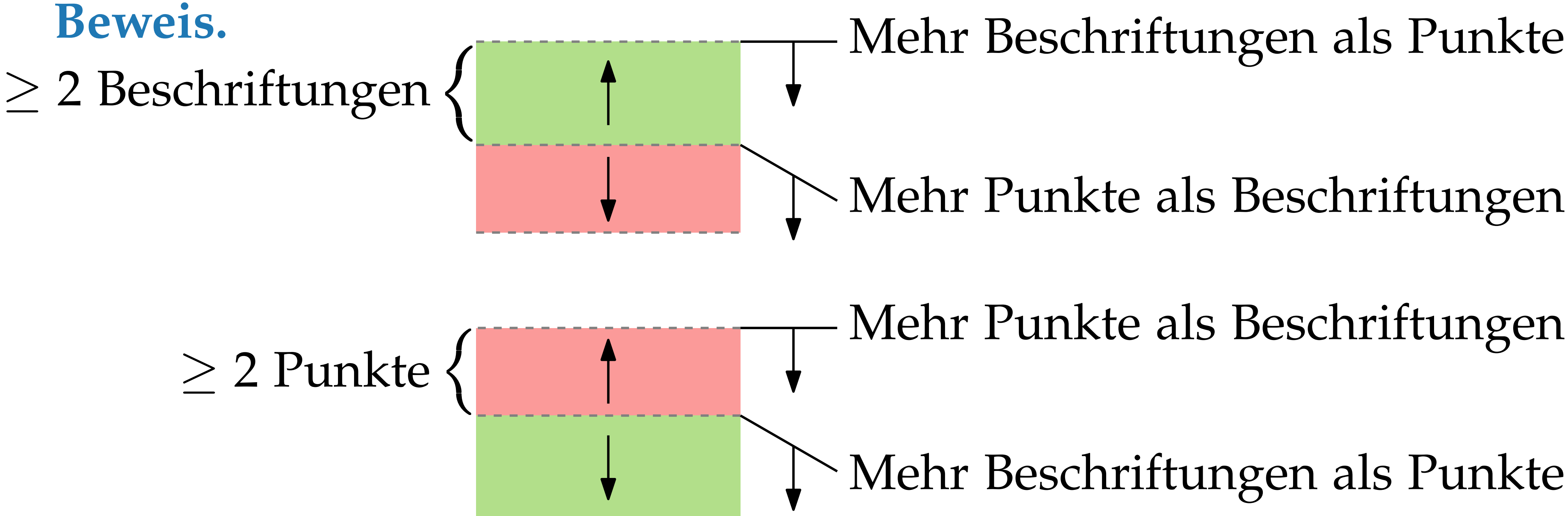
3. Fall. p unterhalb, ℓ oberhalb von σ.

Sweepline-Verfahren

Lemma. Für jede kreuzungsfreie Zuordnung minimaler Länge gilt, dass kein Leader einen neutralen Streifen kreuzt.

Lemma. Zwischen absteigendem Streifen und aufsteigendem Streifen befindet sich immer ein neutraler Streifen.

Beweis.



Sweepline-Verfahren

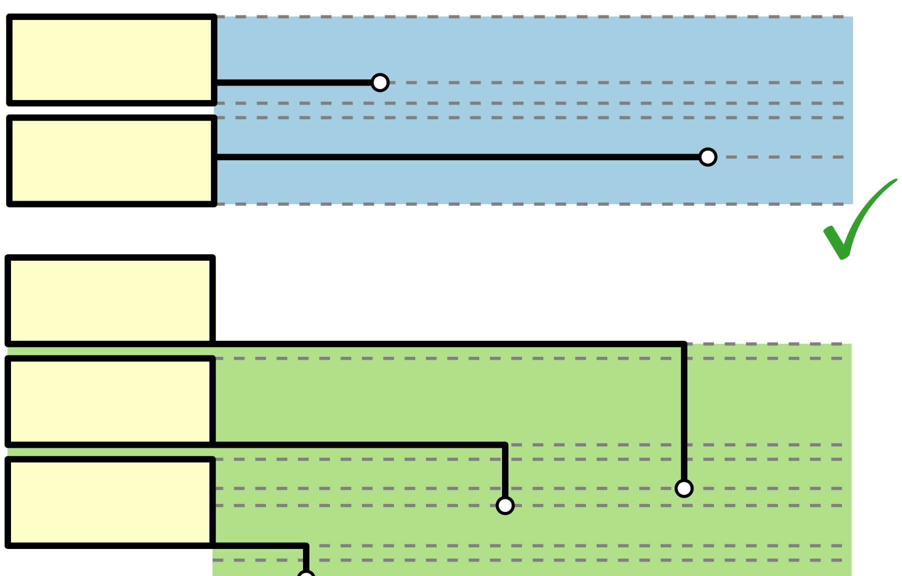
Lemma. Für jede kreuzungsfreie Zuordnung minimaler Länge gilt, dass kein Leader einen neutralen Streifen kreuzt.

Lemma. Zwischen absteigendem Streifen und aufsteigendem Streifen befindet sich immer ein neutraler Streifen.

$\Rightarrow$ Jede optimale Lösung respektiert die Aufteilung in Teilinstanzen.

Lemma. Für jede Teilinstanz berechnet der Algorithmus eine kreuzungsfreie Lösung minimaler Länge.

Beweis.



Kreuzungsfrei: ✓

Horizontale Segmente: Nur auf Sweepline, „linkester“ Punkt

Minimale Länge: ✓

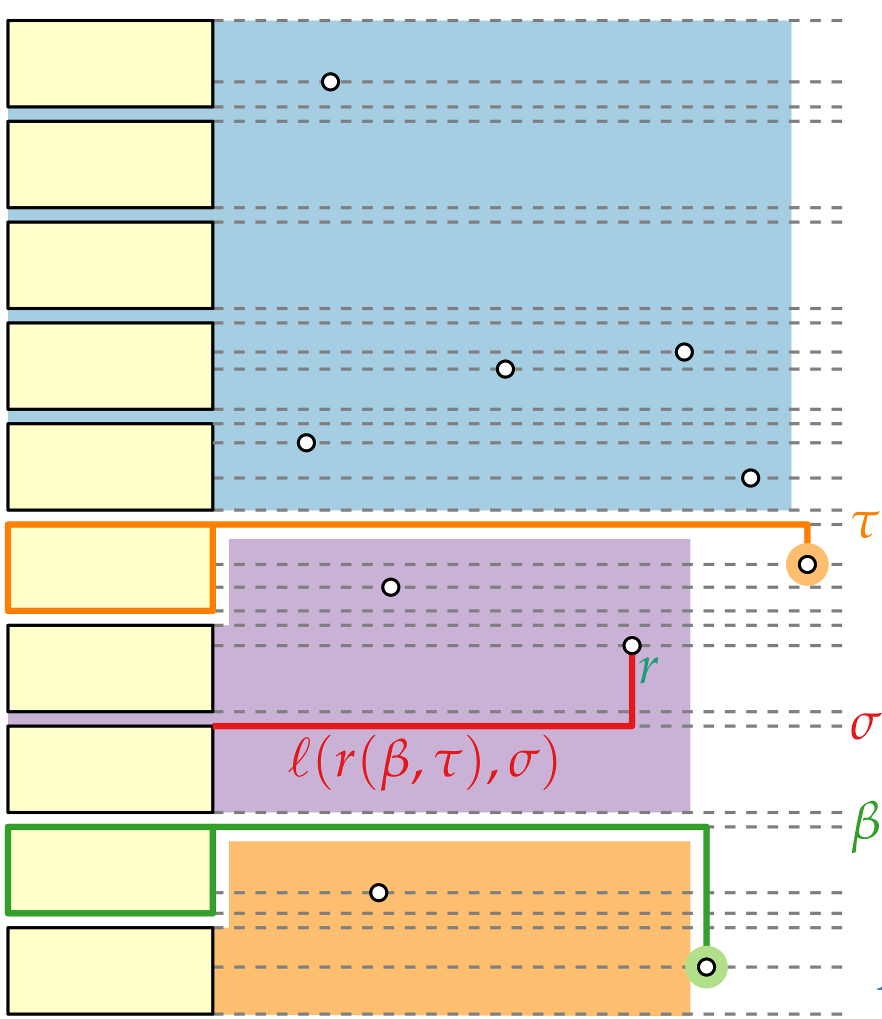
Alle aufsteigenden Lösungen haben gleiche Länge!

Allgemeines Qualitätsmaß

Definiere $O(n)$ Streifen durch Punkte und Beschriftungen.

1. Unterteile Instanz in Teilinstanzen.
2. Löse Teilinstanzen rekursiv.

$\}$ Dynamisches Programm



Betrachte rechtesten Punkt.
Wähle Leader aus einem Streifenrand.
Teilinstanzen müssen *balanciert* sein
(#Punkte= #Beschriftungen)

Teilinstanz definiert durch:

τ: oberer Rand
β: unterer Rand
~~r: rechtester Punkt~~
$r(\beta,\tau)$ eindeutig!

$$T[\beta,\tau] = \min_{\beta<\sigma<\tau,\ \ell(r(\beta,\tau),\sigma)\text{ balanciert}} \{ T[\beta,\sigma] + T[\sigma,\tau] + \mathrm{qual}(\ell(r(\beta,\tau),\sigma)) \}$$

Allgem. Qualitätsmaß:

Allgemeines Qualitätsmaß

$T[\beta, \tau] = \min_{\beta < \sigma < \tau, \text{ balanciert}} \{ T[\beta, \sigma] + T[\sigma, \tau] + \mathrm{qual}(\ell(r(\beta, \tau), \sigma)) \}$

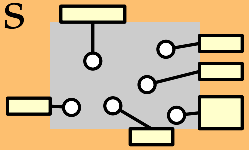
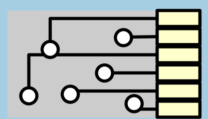
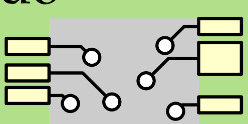
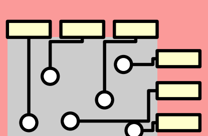
Annahme: Funktion $\mathrm{qual}(\ell(r(\beta, \tau), \sigma))$ berechenbar in $f(n)$ Zeit.

Laufzeit:

- $O(n)$ Streifen, 2 Dimensionen $\Rightarrow$ DP-Tabelle hat Größe $O(n^2)$.
- Berechne $\mathrm{qual}(\ell(r(\beta, \tau), \sigma))$ vor $\Rightarrow O(n^3 \cdot f(n))$ Zeit.
- Sortiere Punkte nach x- und y-Koordinate $\Rightarrow O(n \log n)$ Zeit.
- Berechne einen Eintrag $T[\beta, \tau]$ der DP-Tabelle:
  - Finde $r(\beta, \tau)$ in $O(n)$ Zeit.
  - Alle σ können durch Sortierung in $O(n)$ Zeit gefunden werden. $\Rightarrow O(n)$ Zeit

Satz. Eine kreuzungsfreie Beschriftung im 1-seitigen po-Leader Model mit freien Ports, die eine beliebiges Qualitätsmaß für Leader minimiert, kann in $O(n^3 \cdot f(n))$ Zeit gefunden werden.

Ergebnisse

Leader	Seiten	Ports	Zielfkt.	Laufzeit	
s 	1, 2, 3, 4	fix	Länge	$O(n^{2+\varepsilon})$	[2]
	1, 2, 3, 4	frei	Länge	$O(n^3)$	[2]
po 	1	frei	Länge	$O(n \log n)$	[1]
	1	frei	beliebig	$O(n^3)$	[1]
	2 (geg.)	frei	Länge	$O(n^2)$	[2]
	2 (adj.)	fix	–	$O(n^2)$	[3]
	2 (adj.)	fix	Länge	$O(n^3 \log n)$	[4]
	3	fix	–	$O(n^4)$	[3]
	3, 4	fix	Länge	$O(n^6)$	[4]
	1, 2, 3, 4	beides	Knicke	$O(n^6)$	[4]
do 	1	frei	Länge	$O(n^2)$	[1]
	1	frei	beliebig	$O(n^5)$	[1]
opo 	1	fix	Länge	$O(n \log n)$	[2]
	2	beides	Länge	$O(n^2)$	[2]
	4	fix	Länge	$O(n^2 \log^3 n)$	[2]
	1, 2, 3, 4	beides	Länge	$O(n^{12})$	[4]

[1] Benkert et al. ’09

[2] Bekos et al. ’07

[3] Kindermann et al. ’16

[4] Bose et al. ’18

Für beliebige Zielfunktion:
Annahme $f(n) \in O(1)$