Zuordnung von Punkten mittels geometrischer Objekte

Nikolaus Mutsanas

Universität Karlsruhe

Problemstellung

<u>**Geg:**</u> geometrische Objekte: $Quadrate$
Punktemenge P, $|P| = 2n$.
<u>**Ges:**</u> Menge von Quadraten so dass jedes Quadrat genau zwei Punkte enthält.

Matching ist perfekt: Jedes $p \in P$ wird überdeckt.
Matching ist stark: Quadrate überschneiden sich nicht.

Objekte: $Quadrate$, $Rechtecke$, $Strecken$, $Kreise$...

Bis jetzt bekannt

[Abrego, Arkin, Fernandez-Merchant, Hurtado, Kano, Mitchell & Urrutia]

Für jedes P können mindestens 25% der Punkte stark **Kreis**-gematched werden.

Für jedes P können mindestens 40% der Punkte stark **Quadrat**-gematched werden.

$Voraussetzung:$ allgemeine Lage.

Offene Fragen

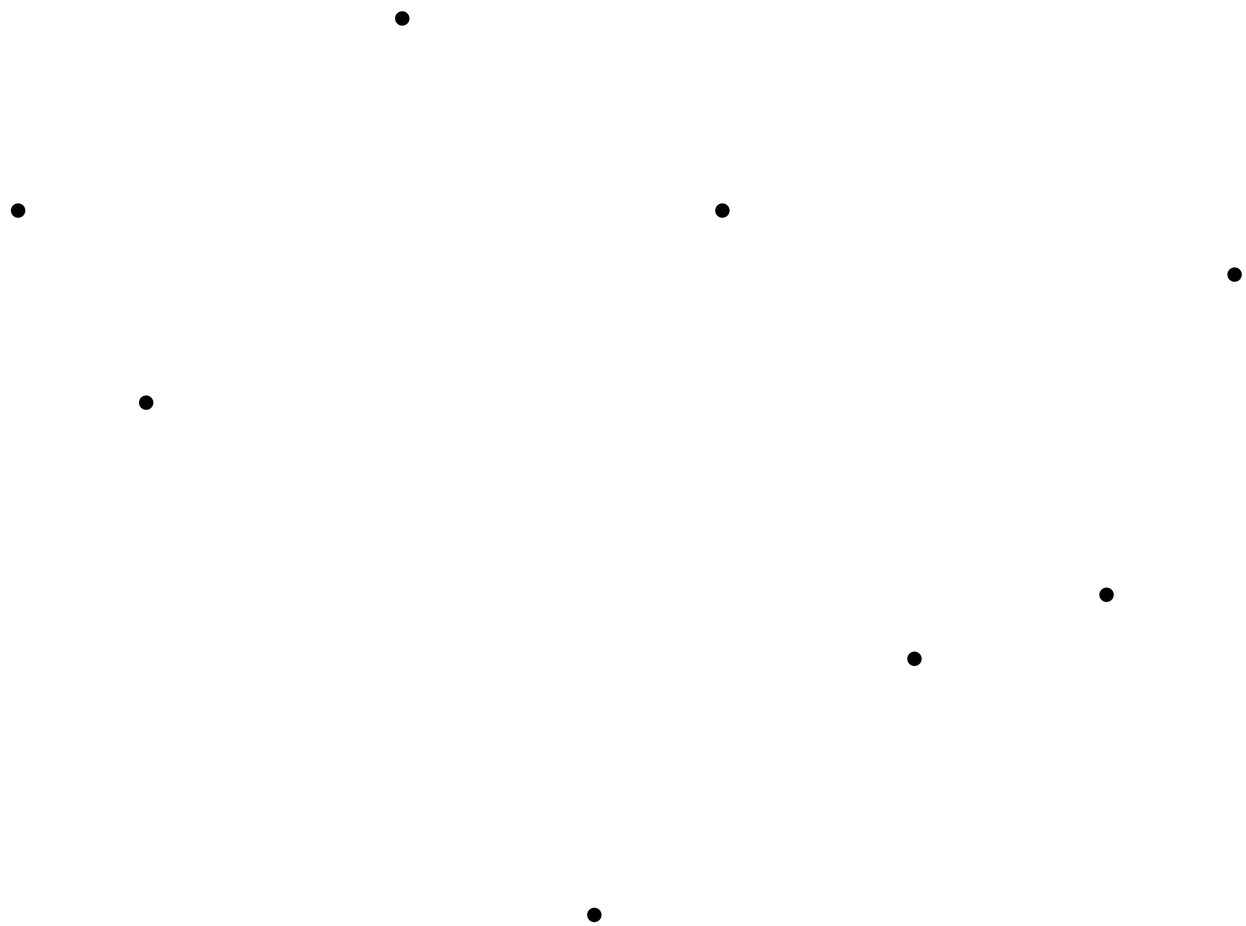
Wie schwer ist es zu entscheiden ob es ein perfektes, starkes Matching gibt?

Wie schwer ist es zu überprüfen ob ein perfektes Matching stark ist?

	ist geg. M. stark?	ex. starkes M.
Strecken:	trivial	trivial
Rechtecke:	trivial	?
Quadrate:	?	?
Kreise:	?	?

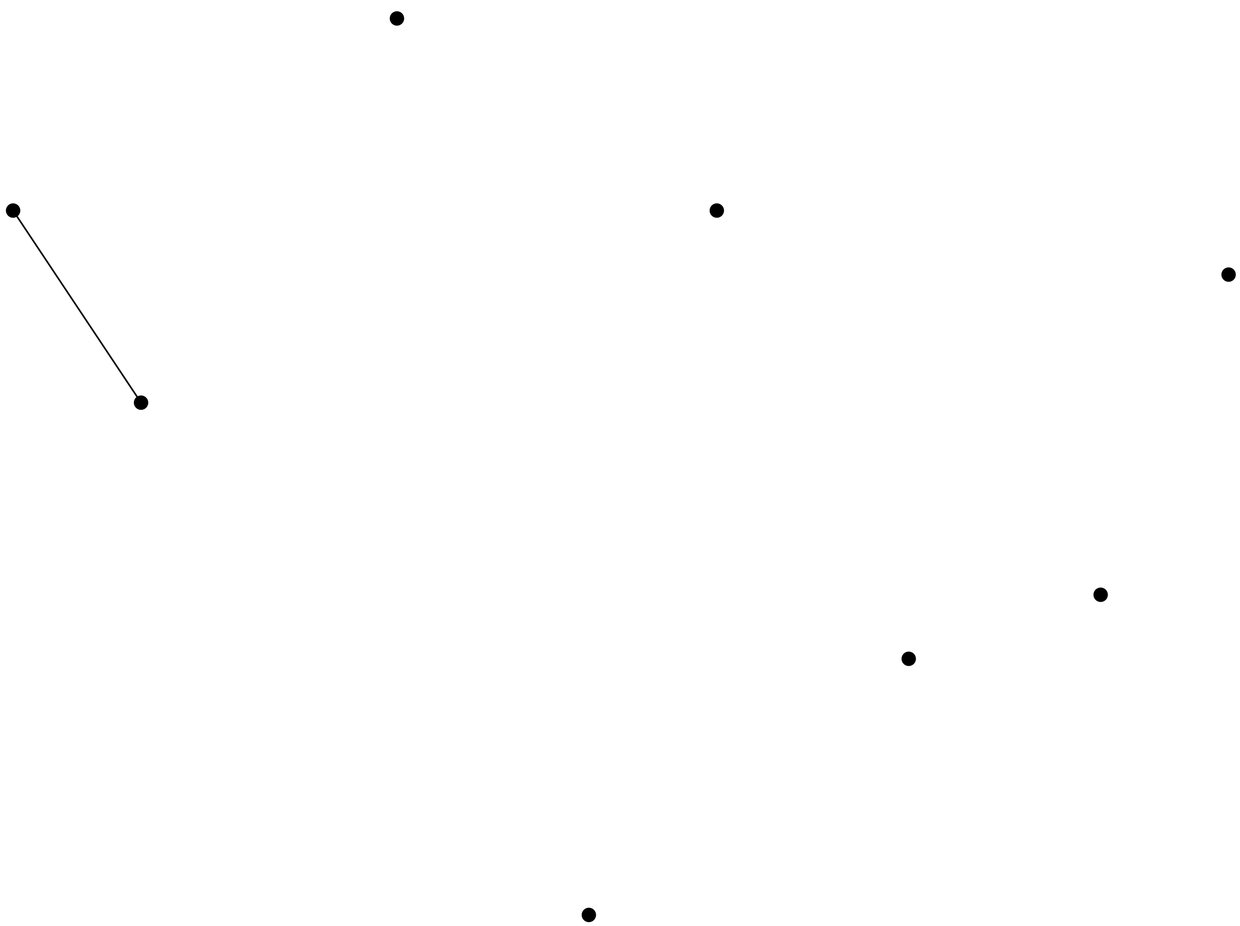
Rechtecke

Allgemeine Lage.



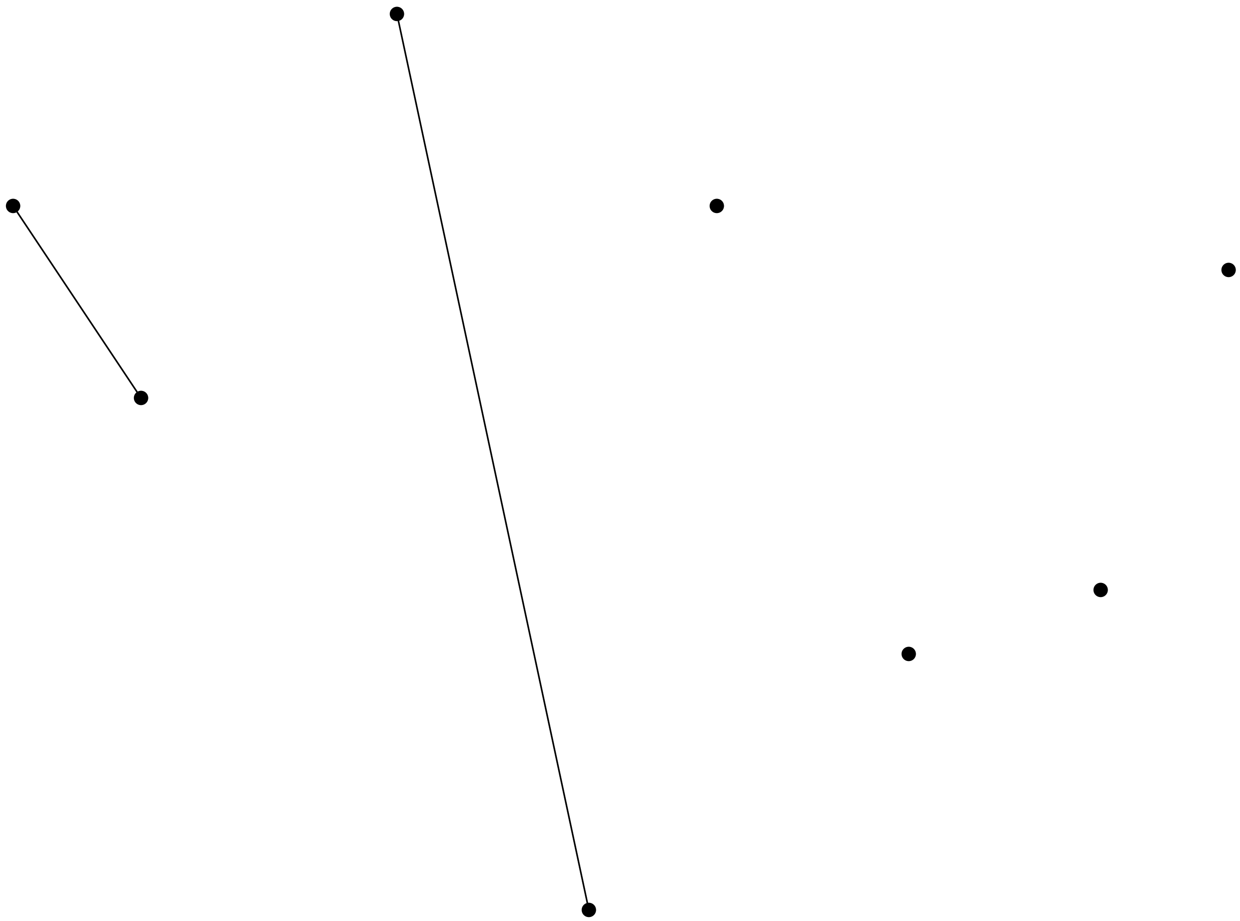
Rechtecke

Allgemeine Lage.



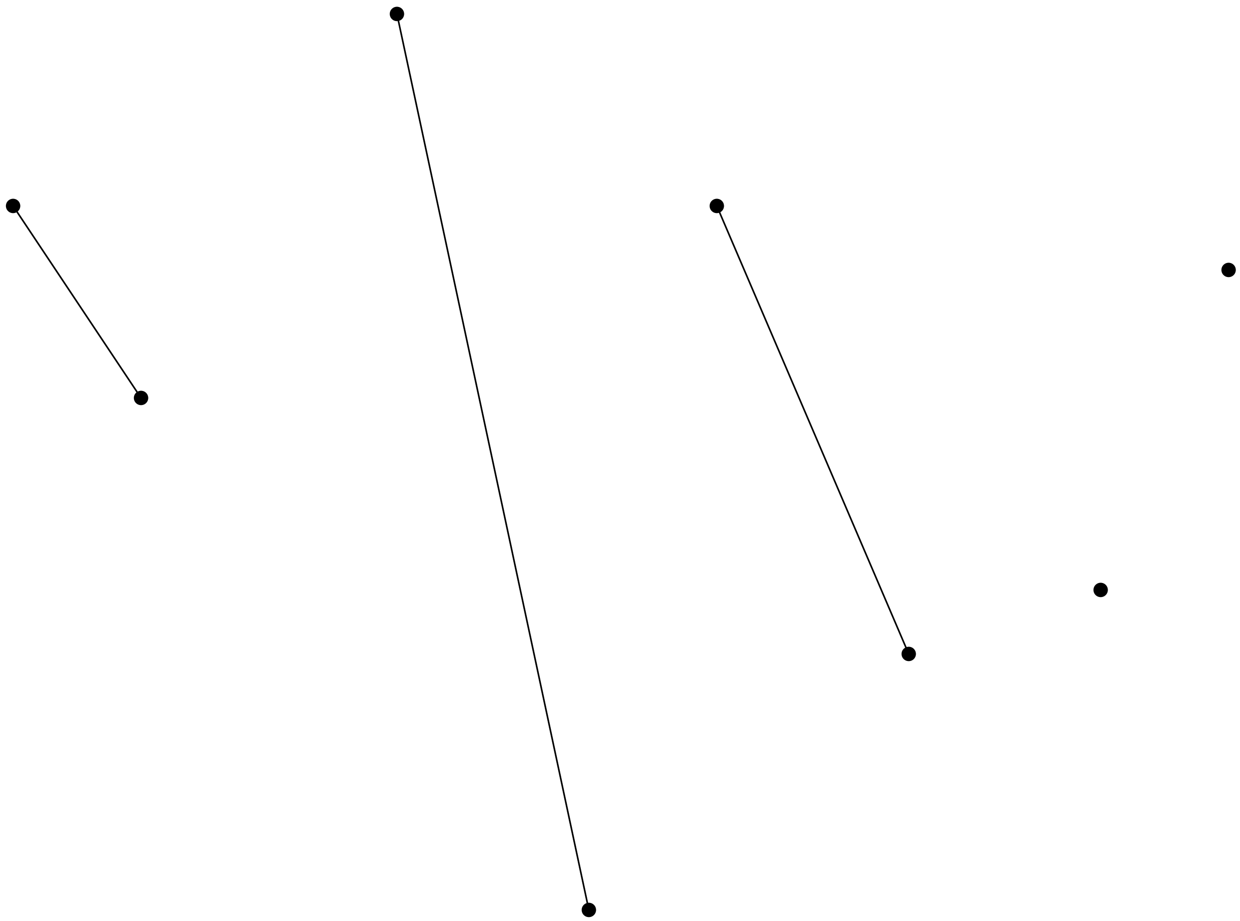
Rechtecke

Allgemeine Lage.



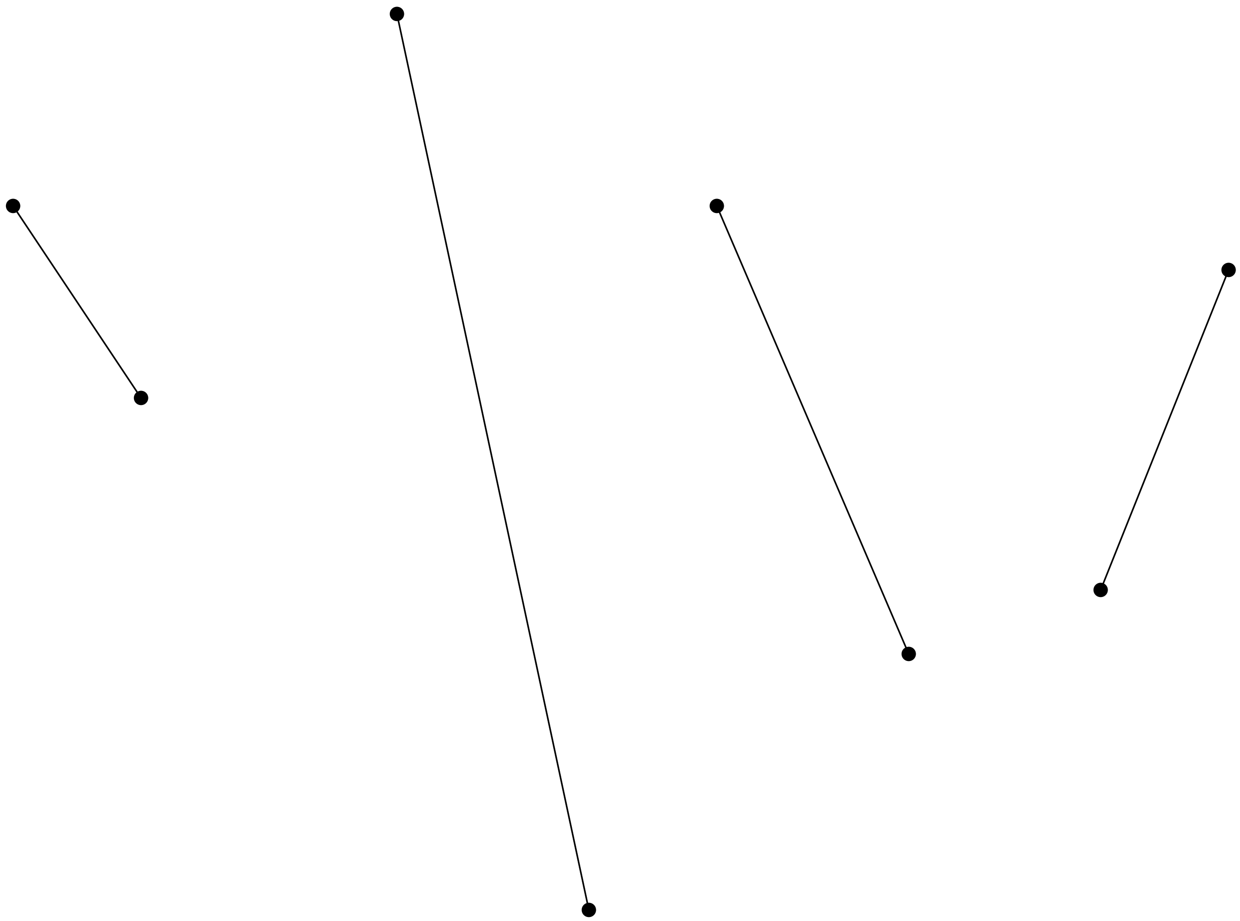
Rechtecke

Allgemeine Lage.



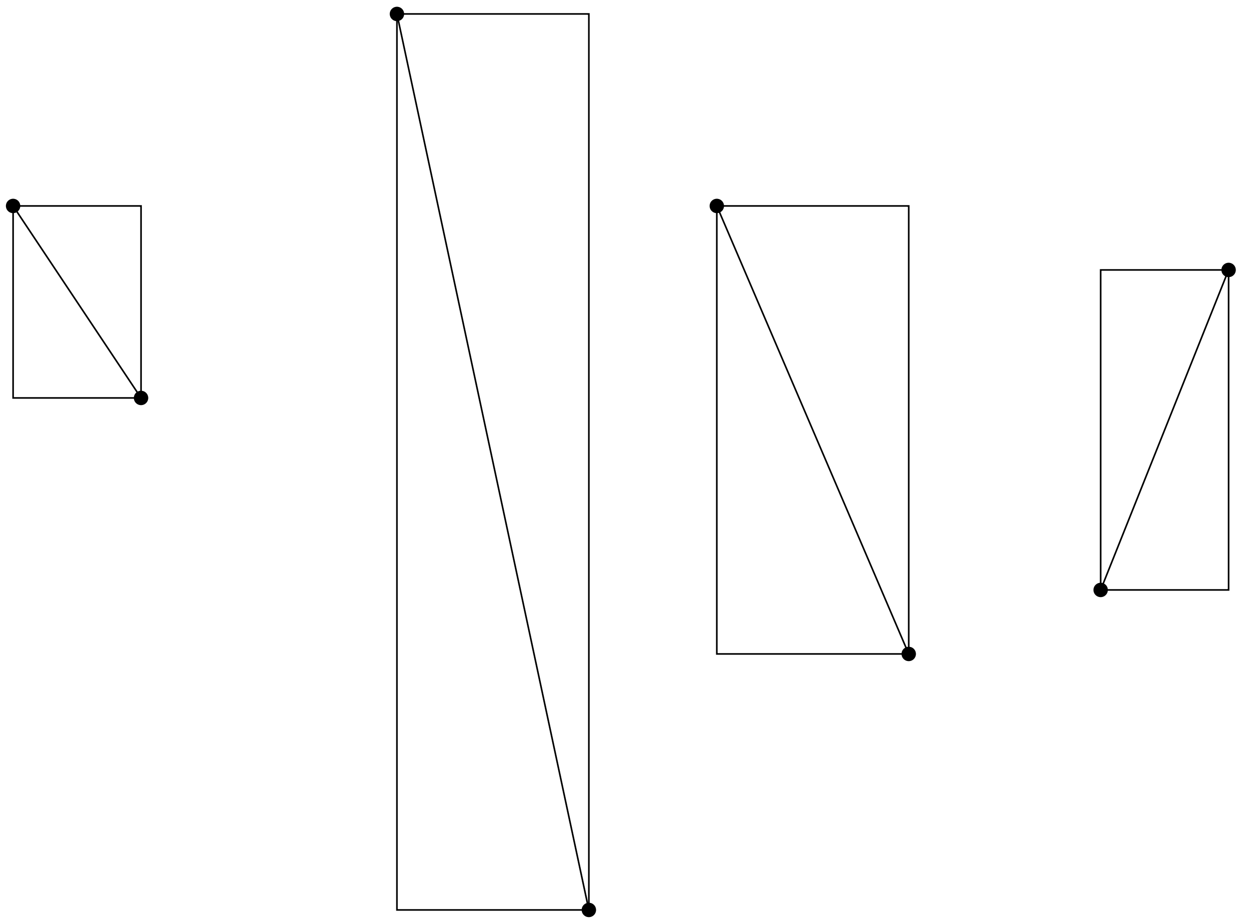
Rechtecke

Allgemeine Lage.



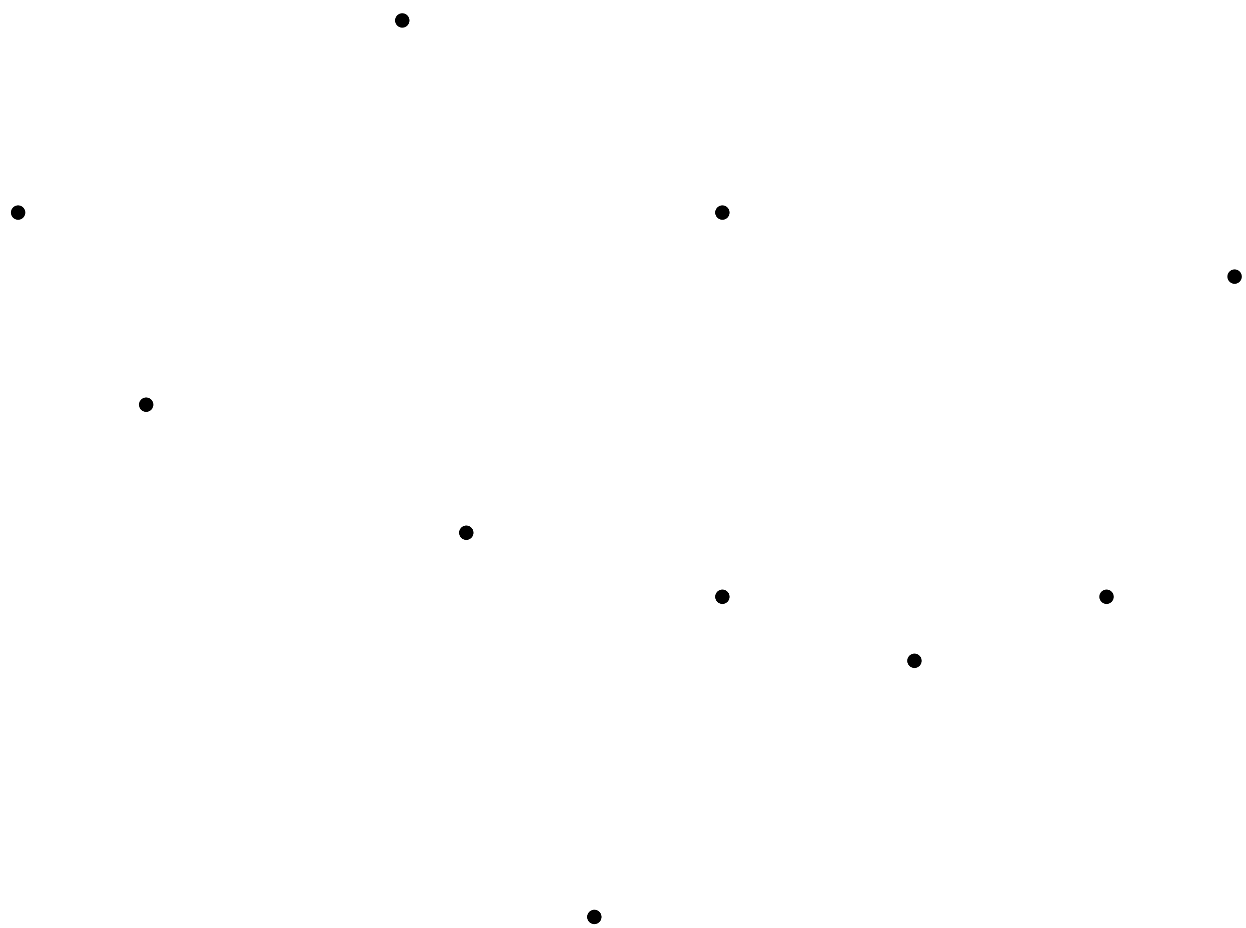
Rechtecke

Allgemeine Lage.



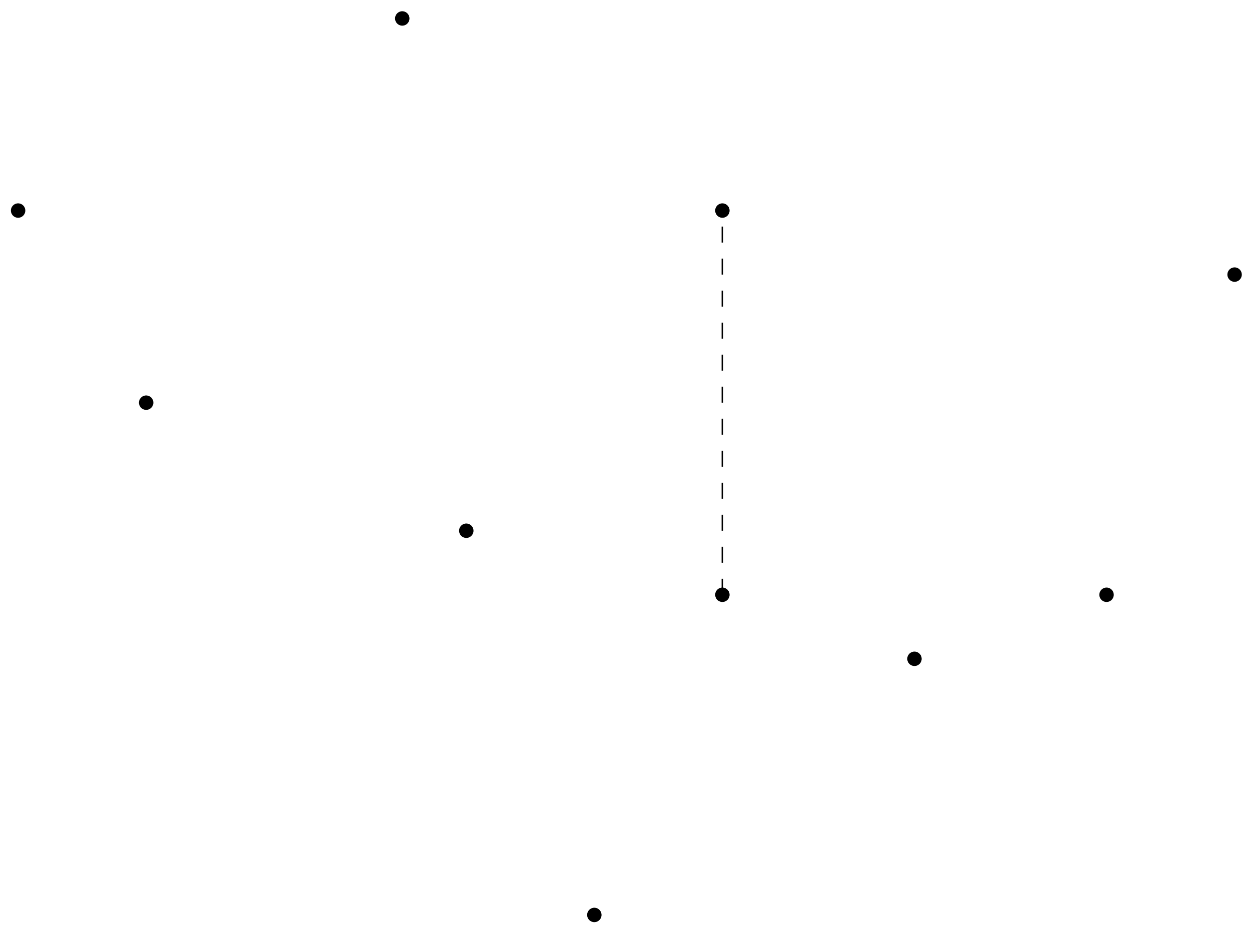
Rechtecke

keine allgemeine Lage.



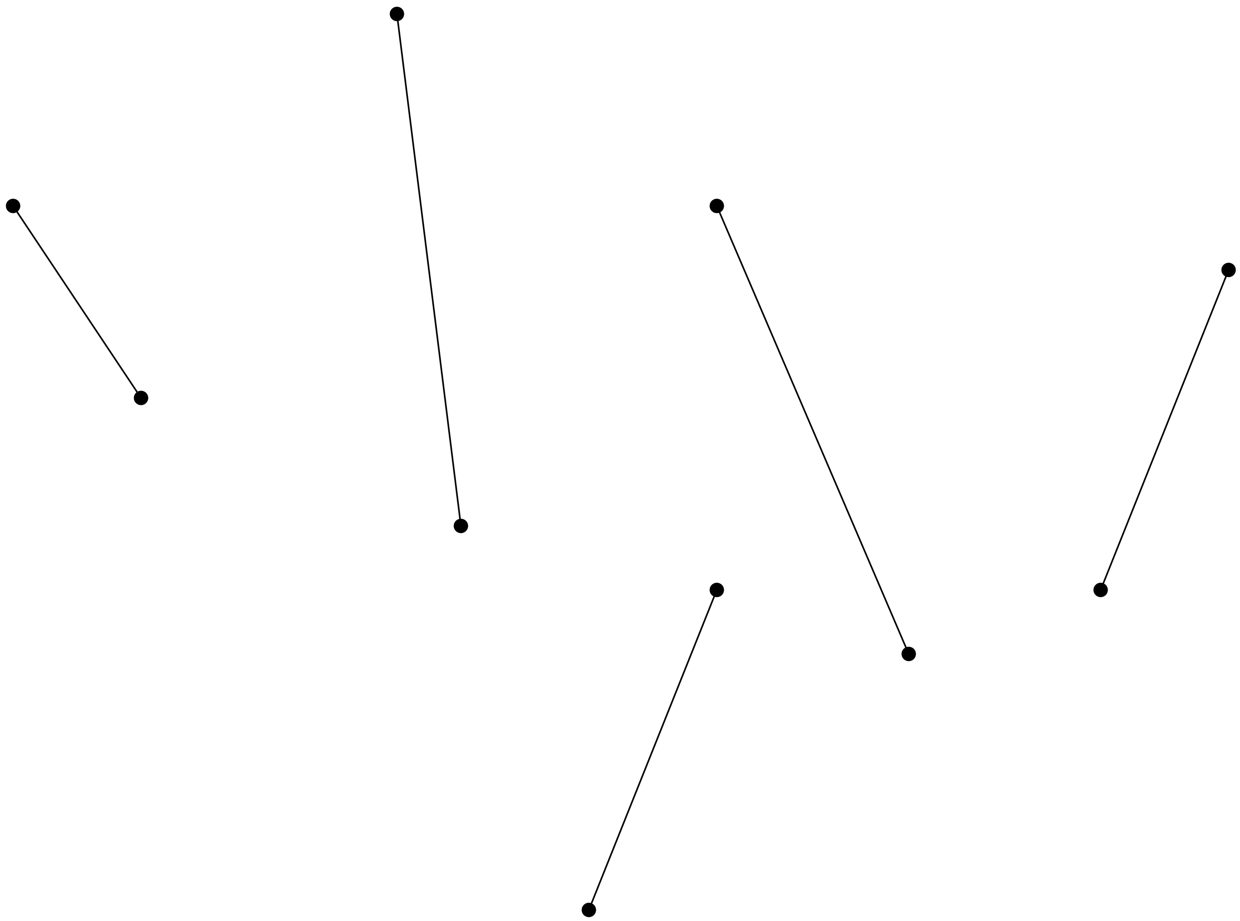
Rechtecke

keine allgemeine Lage.



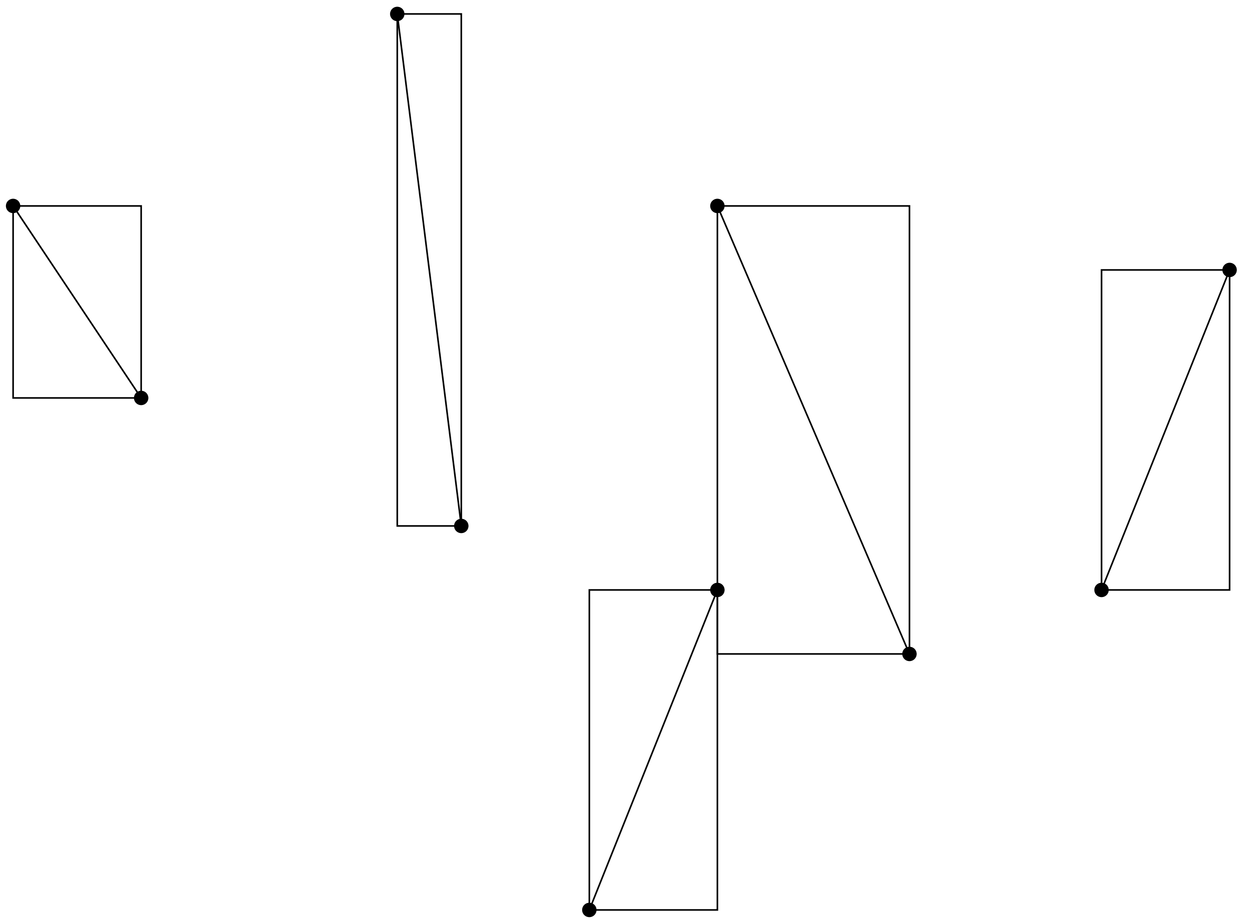
Rechtecke

$keine$ allgemeine Lage.



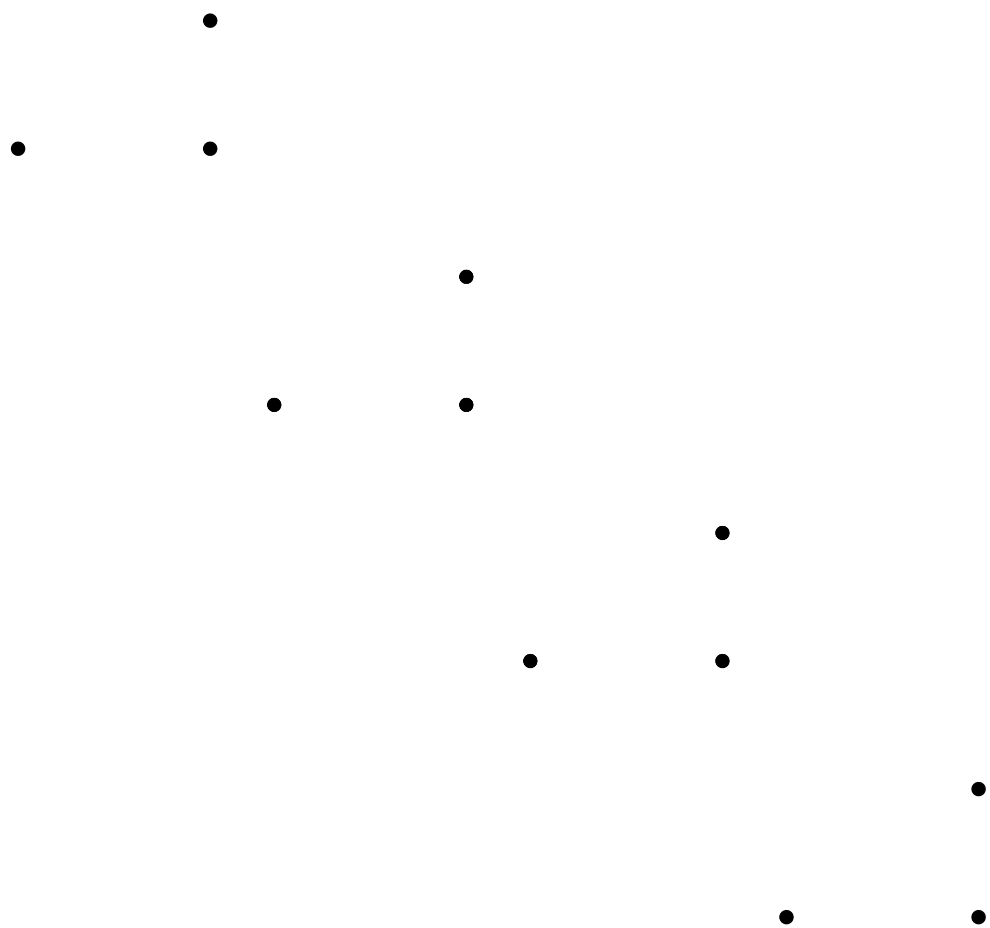
Rechtecke

keine allgemeine Lage.



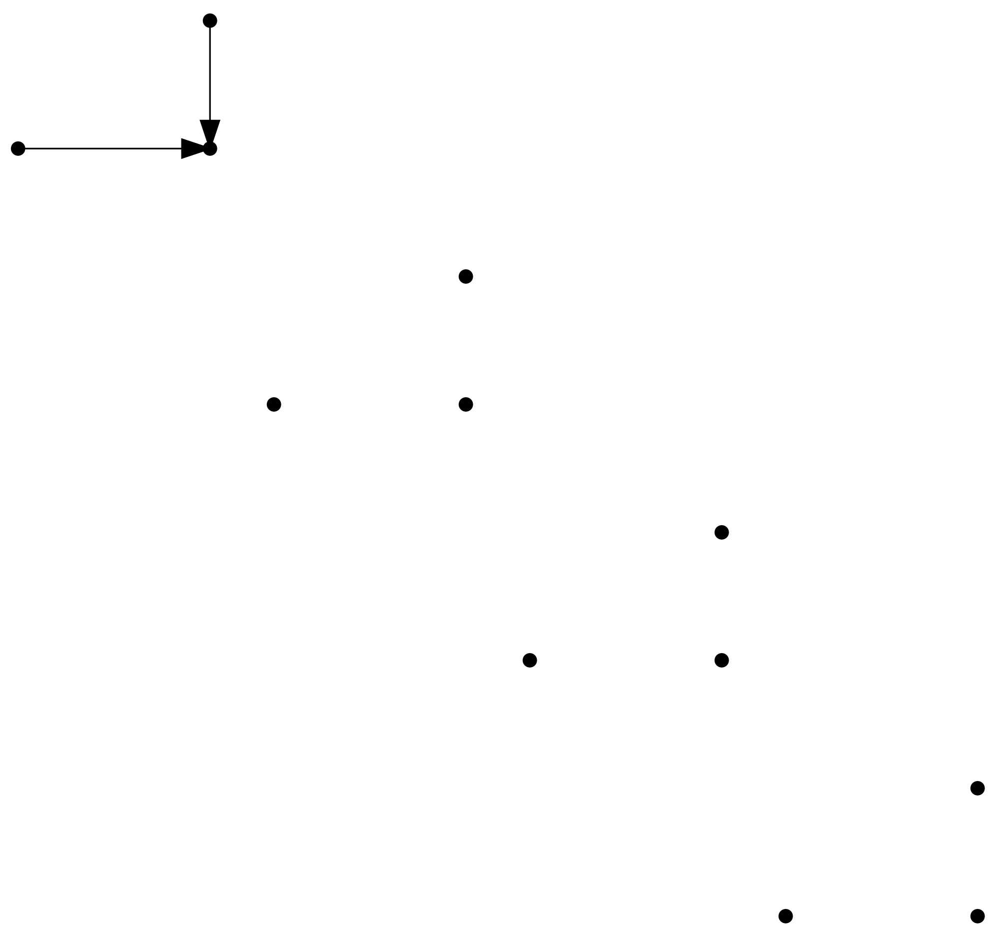
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**



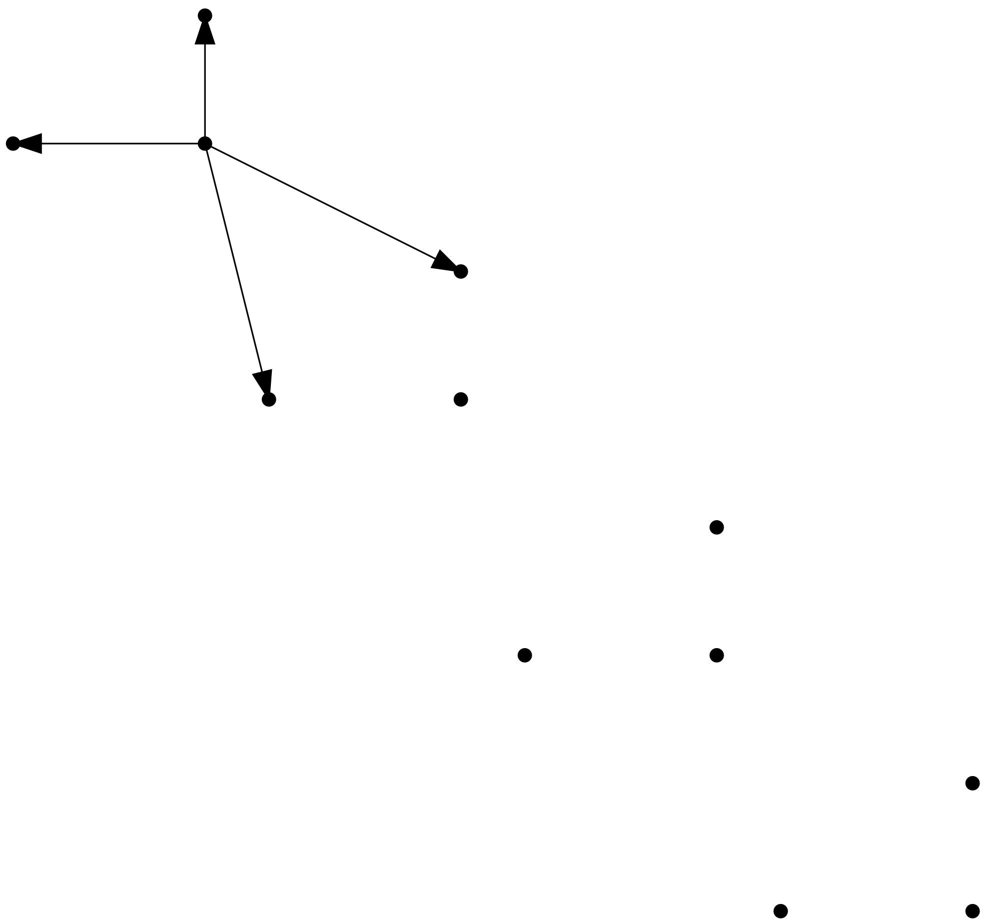
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**



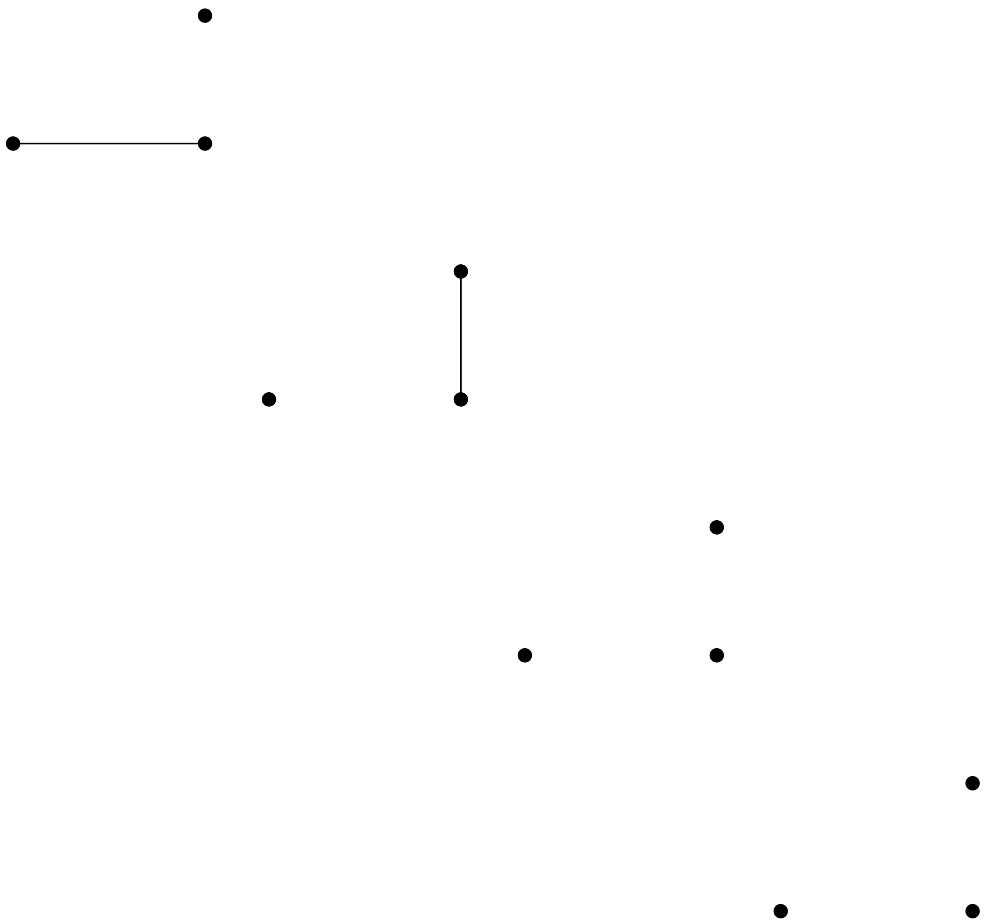
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**



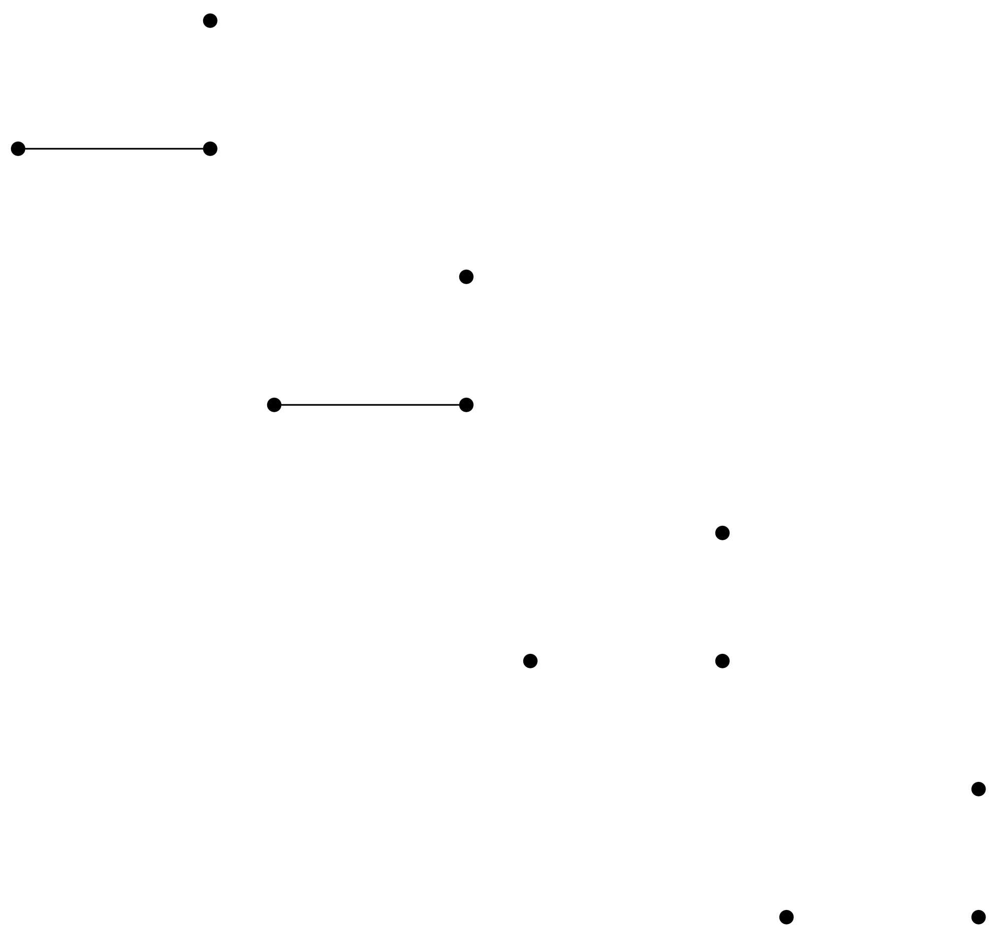
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**



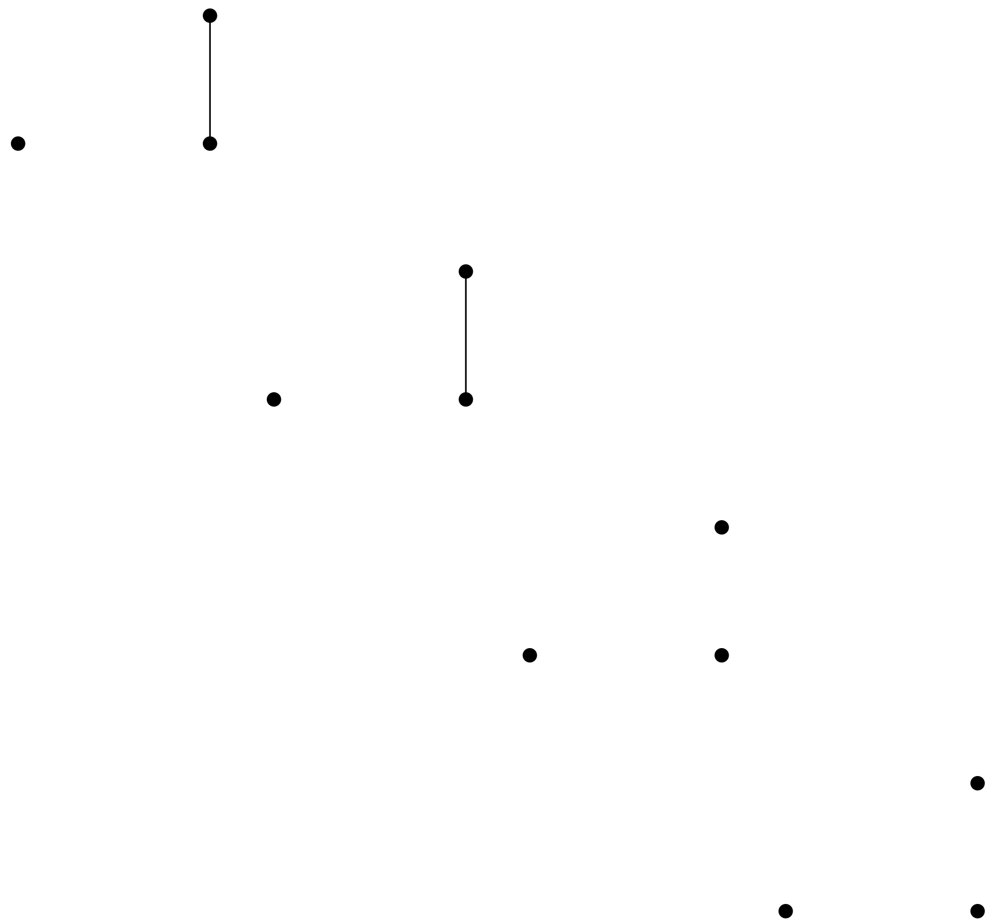
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**



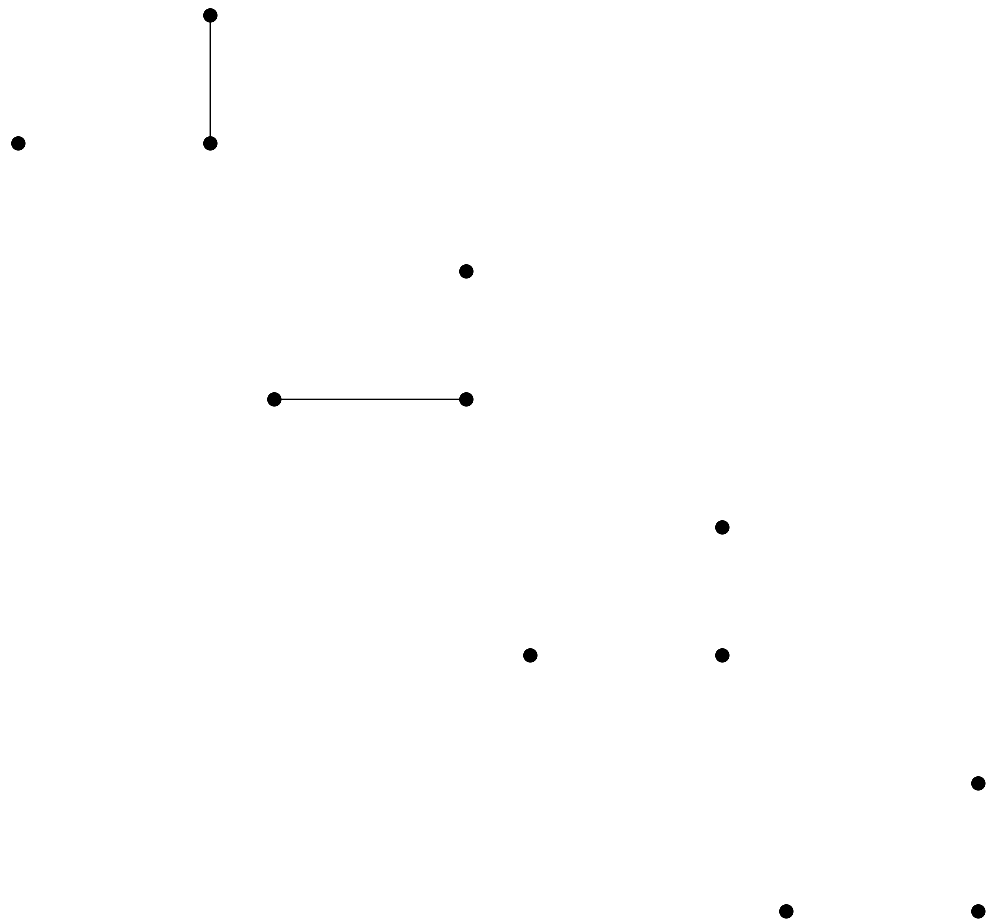
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**



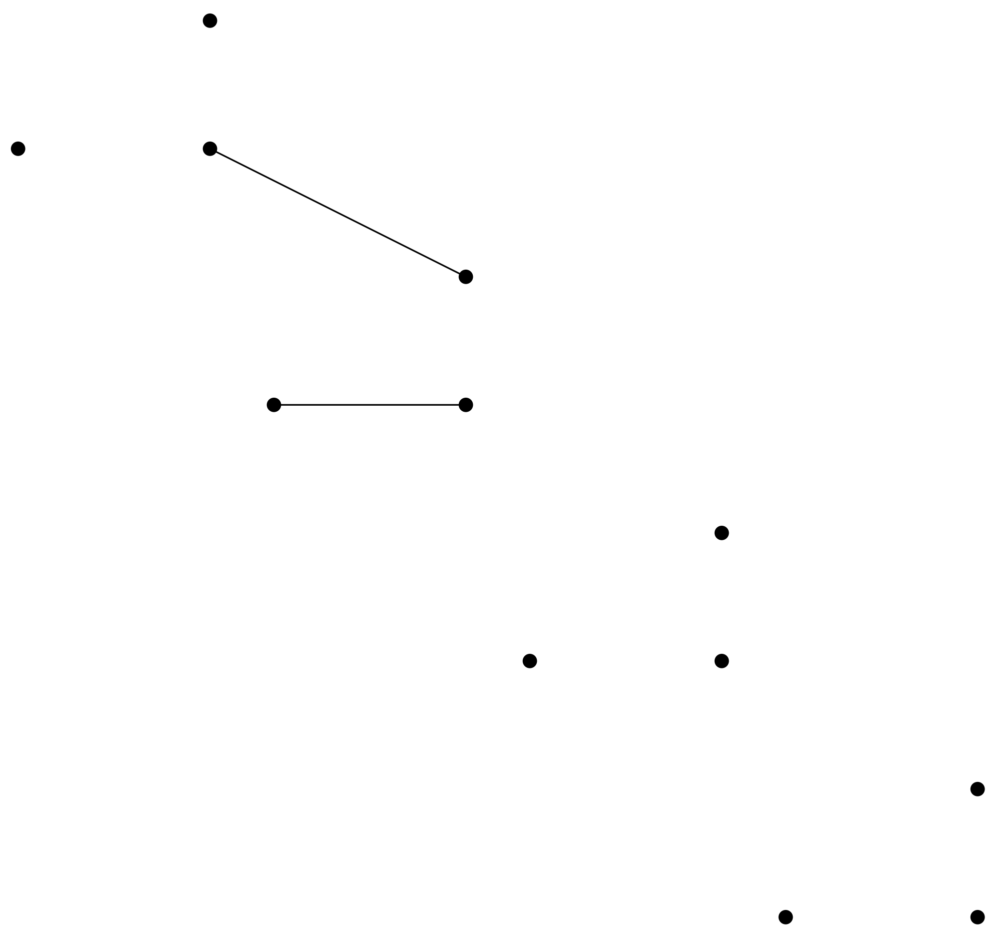
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**



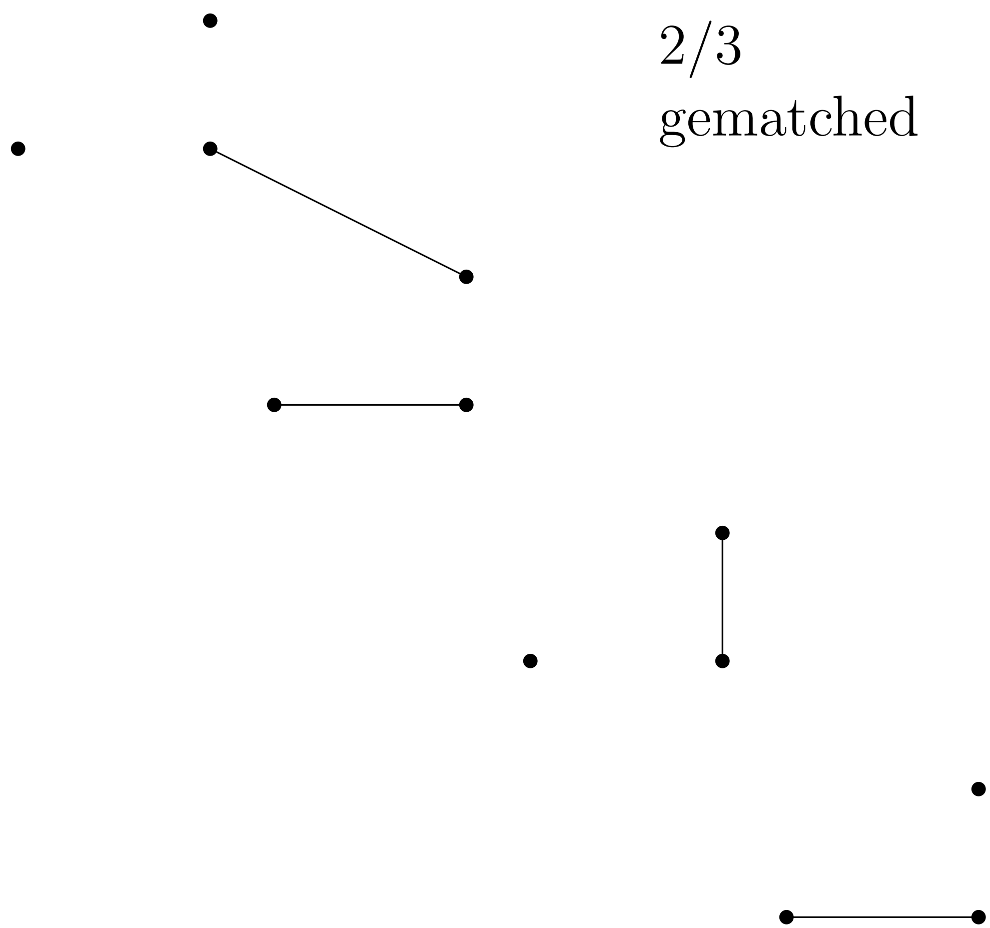
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**



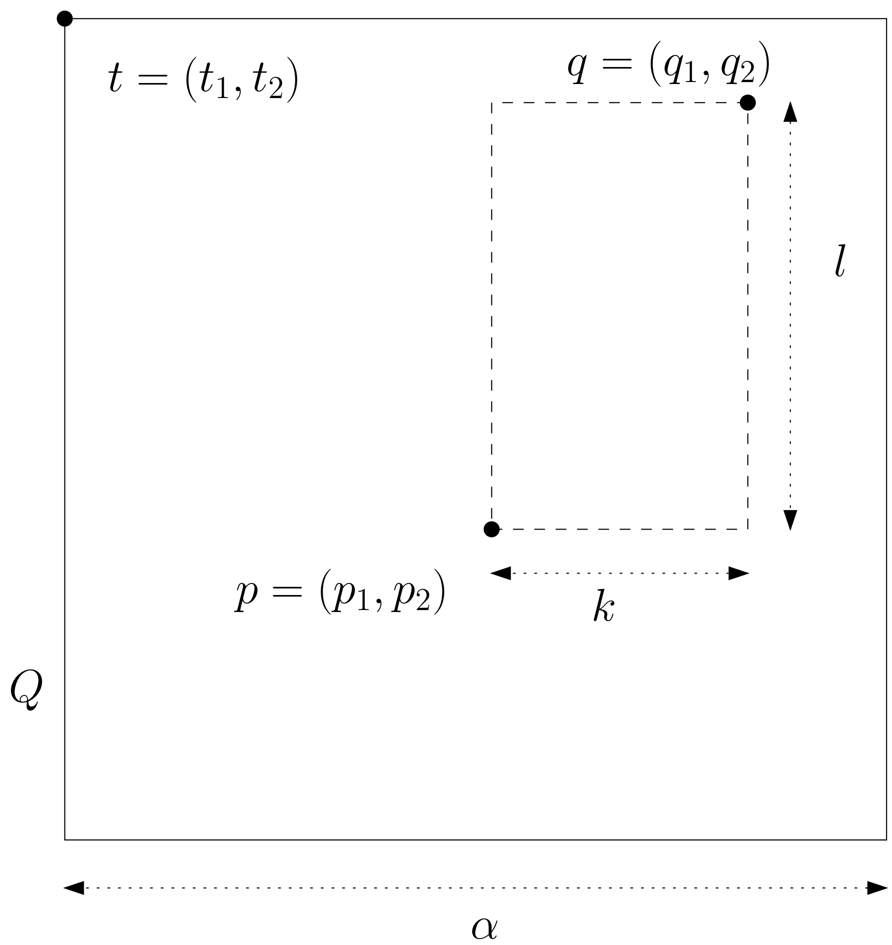
Rechtecke

Gibt es immer eine starke Rechteck-Zuordnung? **NEIN!**

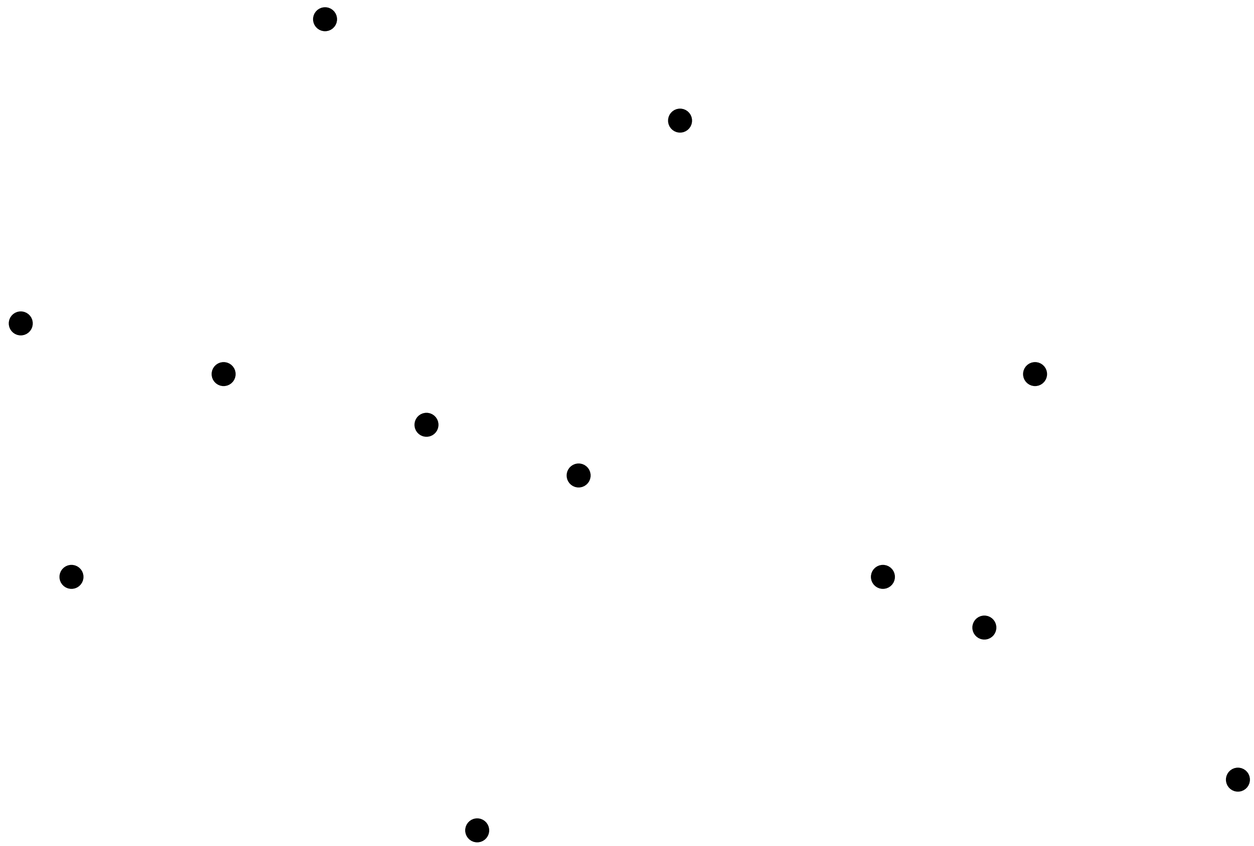


Quadrate

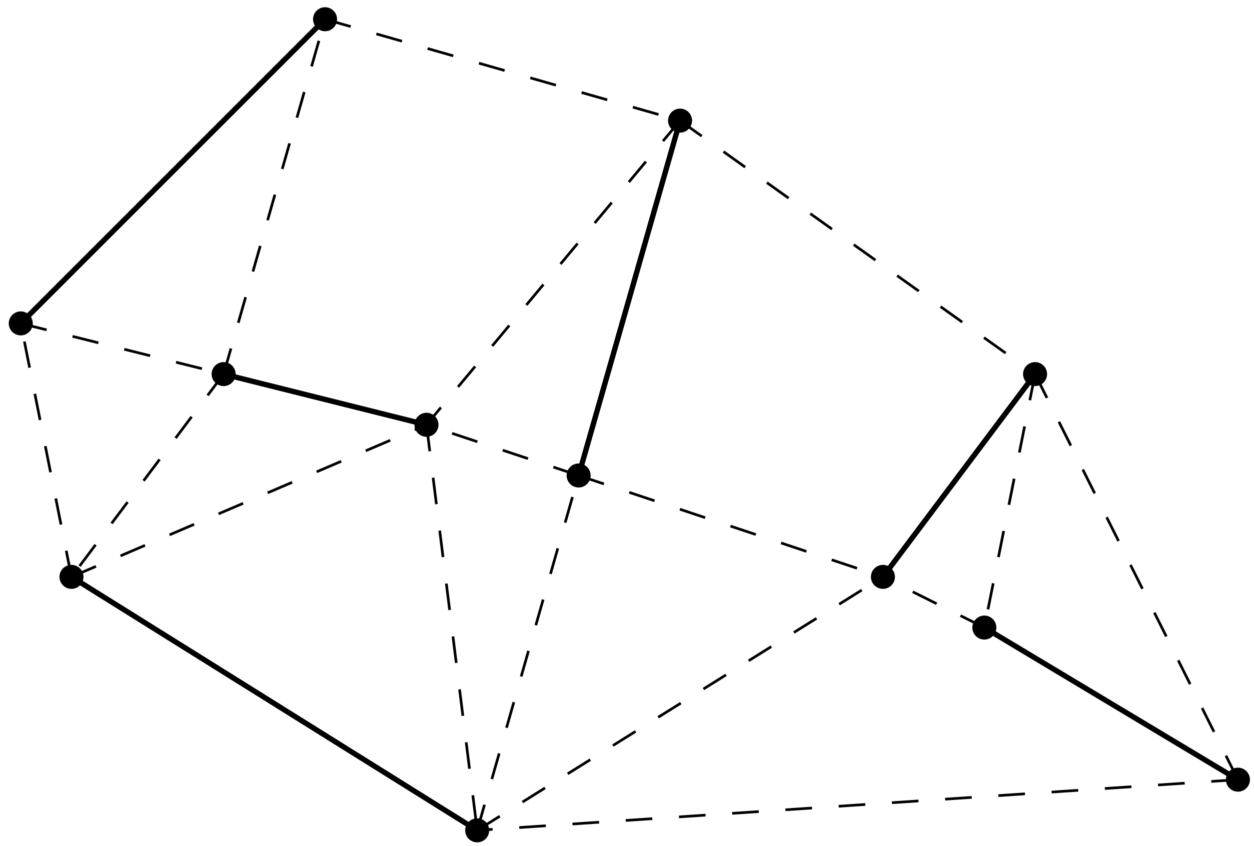
O.B.d.A.: Punkte liegen am Rand.



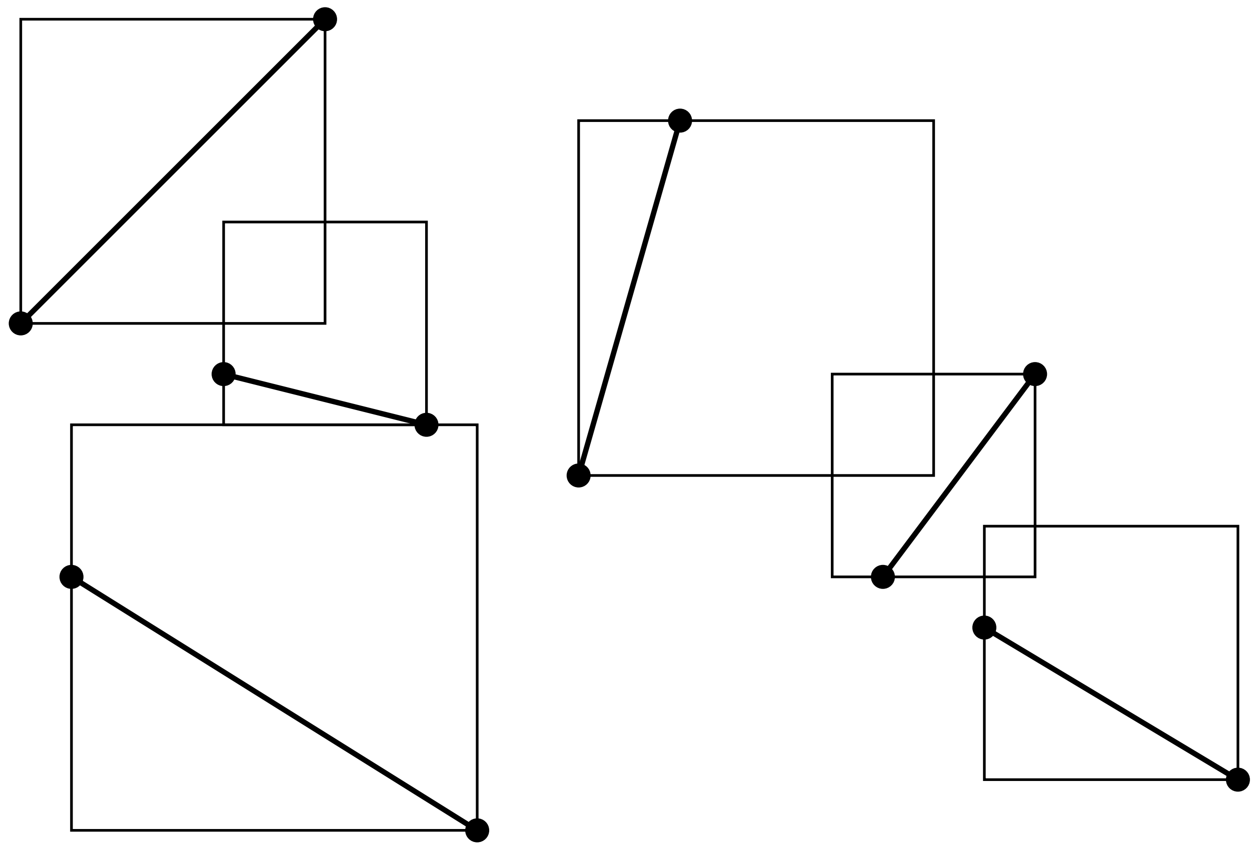
Quadrate - Ist geg. M. stark?



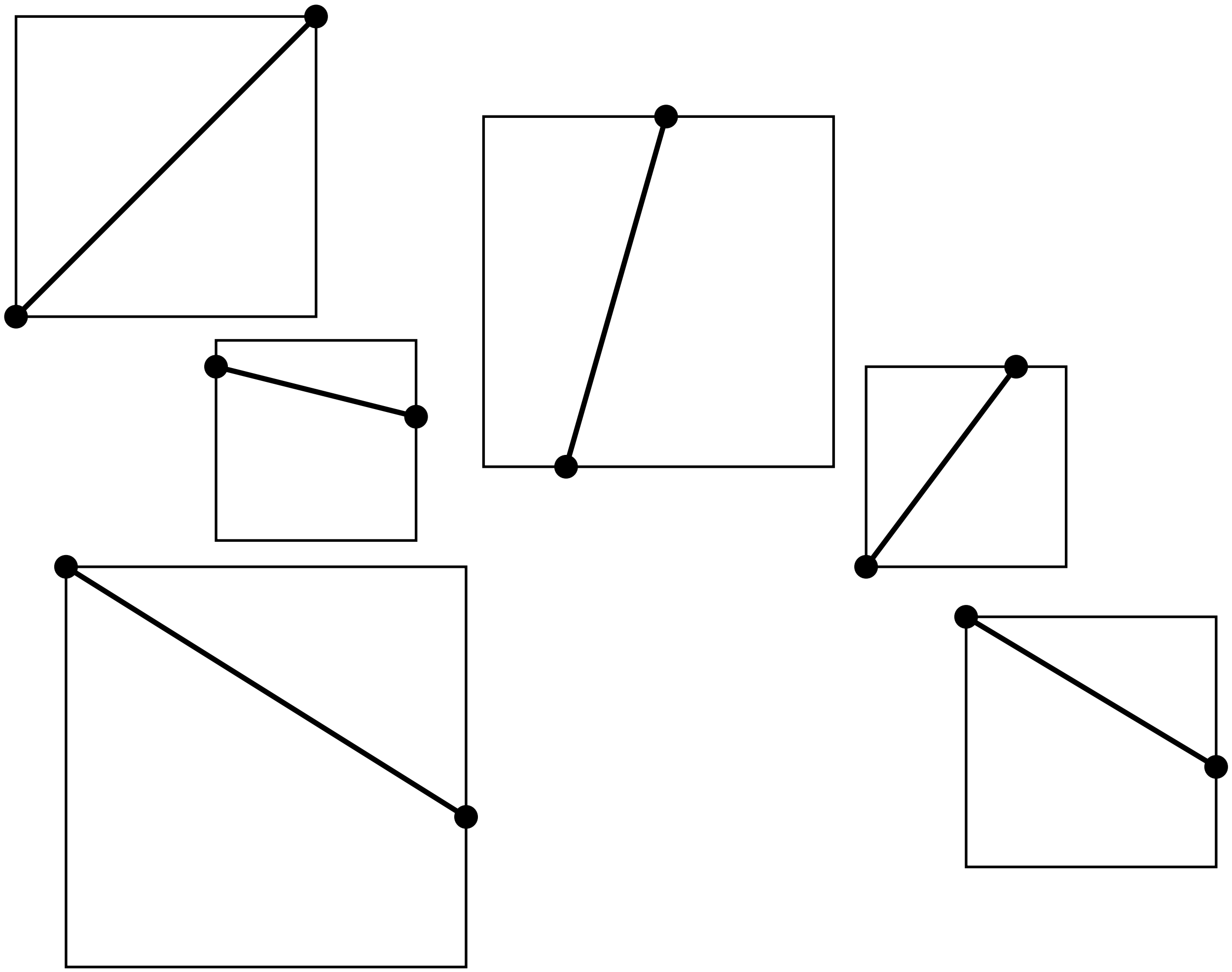
Quadrate - Ist geg. M. stark?



Quadrate - Ist geg. M. stark?

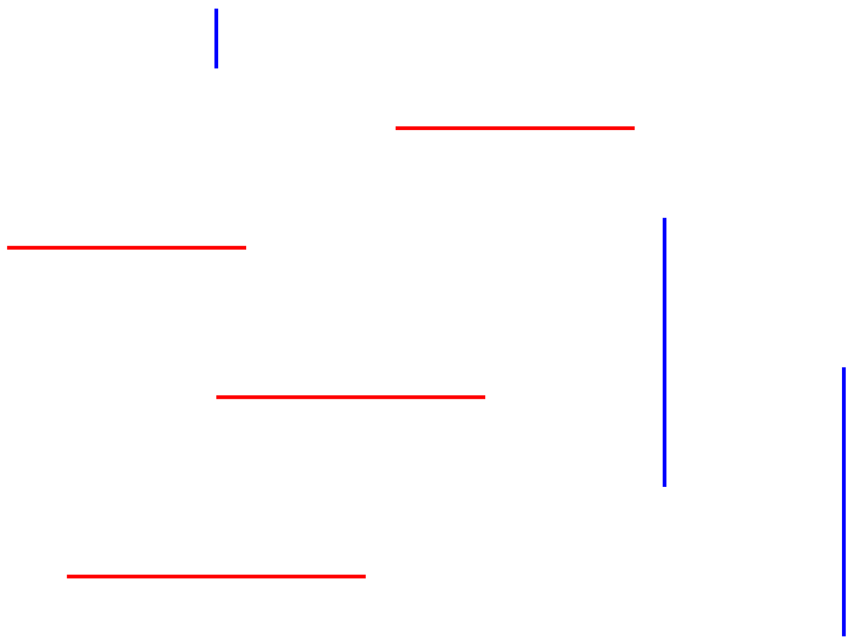


Quadrate - Ist geg. M. stark?



Hilfe aus dem Labeling-Bereich

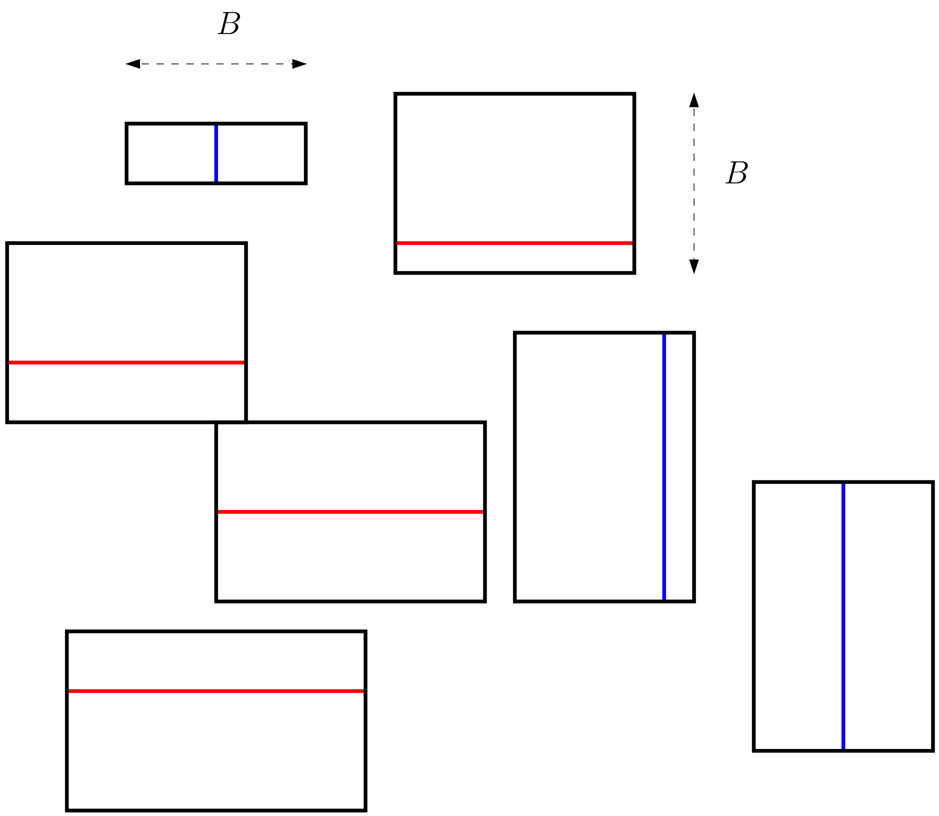
Geg: Menge von achsenparallelen Strecken, $B \in \mathbb{R}$.
Ges: Gibt es Beschriftung der Höhe B?



Hilfe aus dem Labeling-Bereich

Geg: Menge von achsenparallelen Strecken, $B \in \mathbb{R}$.

Ges: Gibt es Beschriftung der Höhe B?



Hilfe aus dem Labeling-Bereich

Geg: Menge von achsenparallelen Strecken, $B \in \mathbb{R}$.
Ges: Gibt es Beschriftung der Höhe B?
Lösungsskizze: [Kim, Shin & Yang]

- Beschriftung $\to$ kanonische Form
- Für jede Strecke berechne Positionen seines Labels bei kanonischer Beschriftung
- Prüfe alle Kombinationen mit Hilfe von 2-SAT

$k_{\max}$: Max. möglicher Positionen eines Labels.
Laufzeit: $O(k_{\max} n \log^2 n)$

Labeling-Problem - Kanonisierung

Verschiebe die Labels der

- der *vertikalen* Strecken so weit nach *links* wie möglich
- der *horizontalen* Strecken so weit nach *unten* wie möglich.

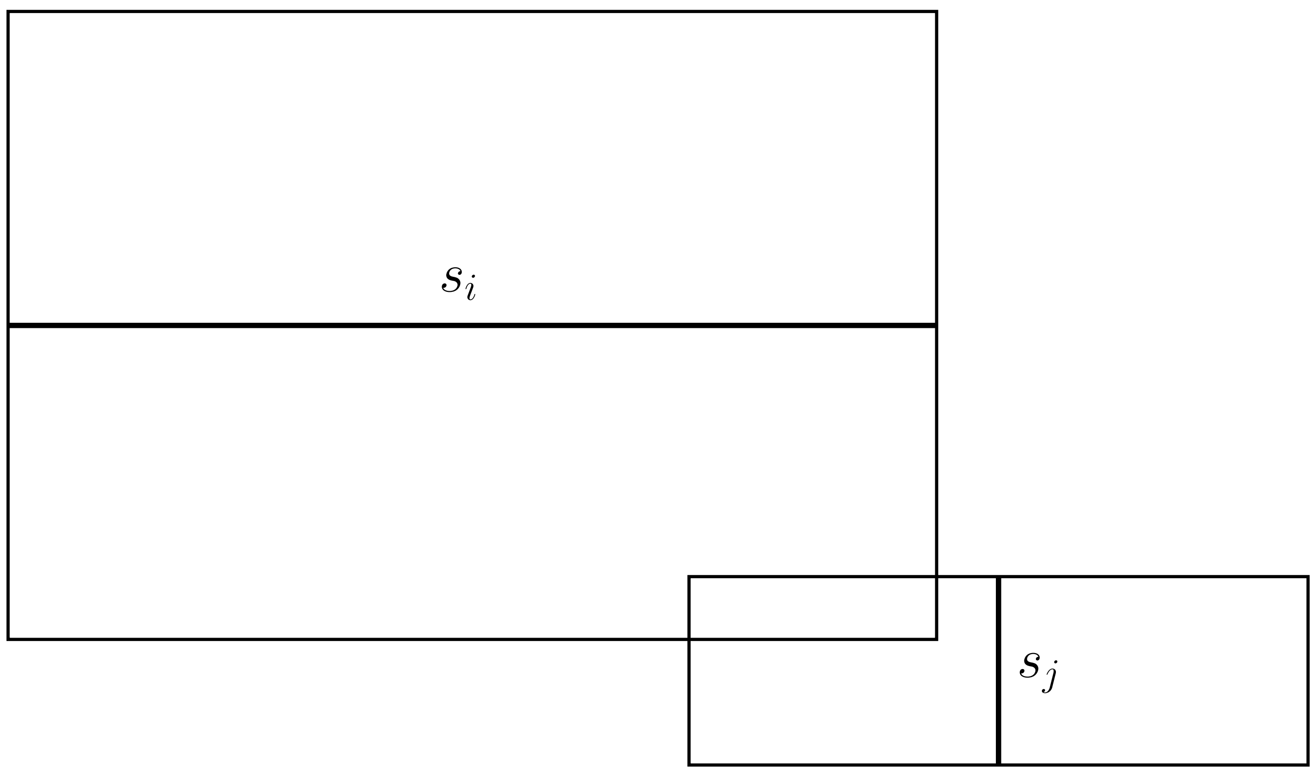
Alle dabei entstehende Positionen kann man im Voraus berechnen!

Labeling-Problem - Relevante Positionen

Für horizontale Strecken:

Sortierung nach y-Koordinate: $s_1, s_2, \ldots, s_n$

Für s_i: Betrachte $s_j \in \{s_1, \ldots, s_{i-1}\}$. Falls s_j *vertikal*:

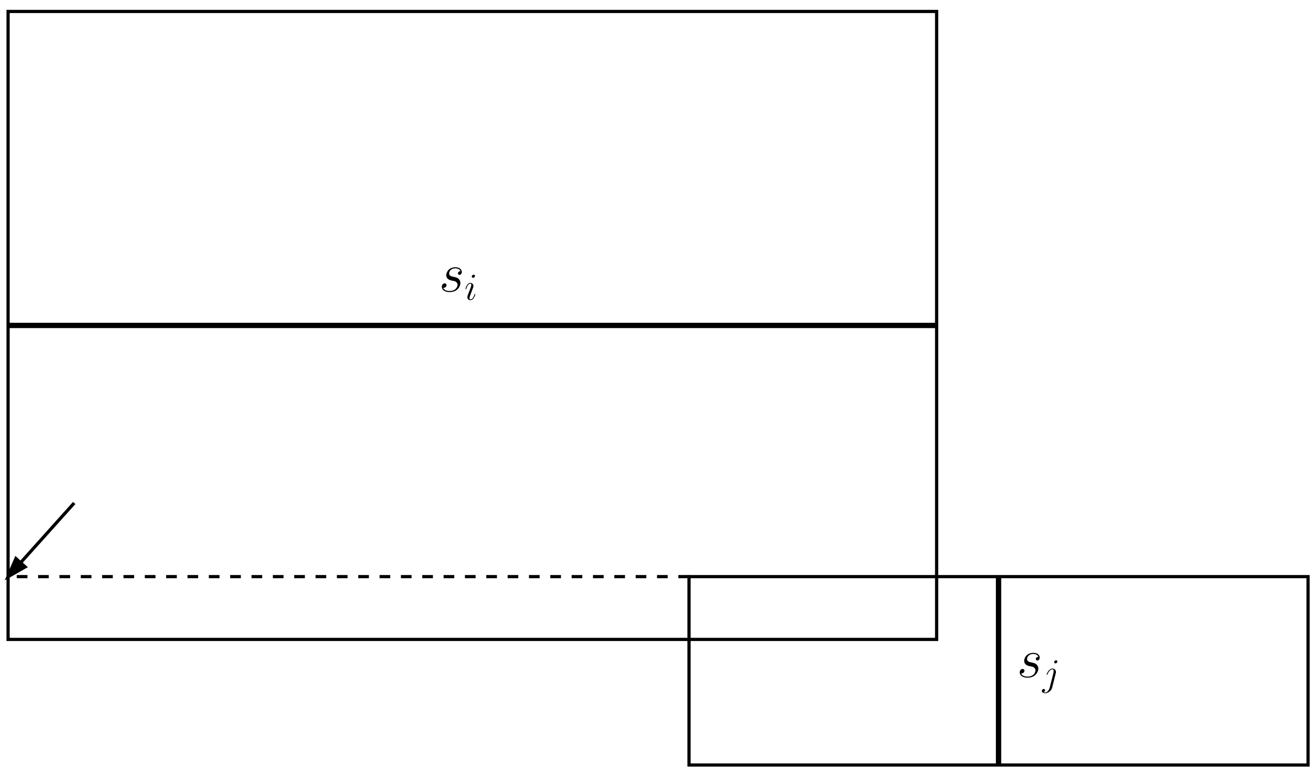


Labeling-Problem - Relevante Positionen

Für horizontale Strecken:

Sortierung nach y-Koordinate: $s_1, s_2, \ldots, s_n$

Für s_i: Betrachte $s_j \in \{s_1, \ldots, s_{i-1}\}$. Falls s_j *vertikal*:

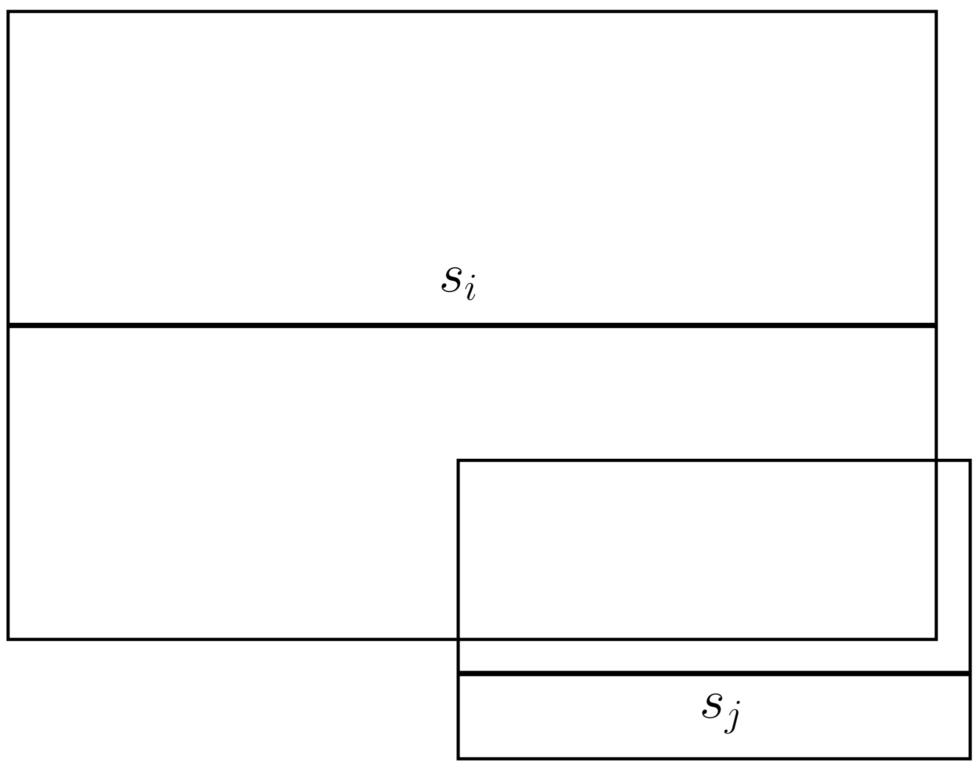


Labeling-Problem - Relevante Positionen

Für horizontale Strecken:

Sortierung nach y-Koordinate: $s_1, s_2, \ldots, s_n$

Für s_i: Betrachte $s_j \in \{s_1, \ldots, s_{i-1}\}$. Falls s_j *horizontal*:

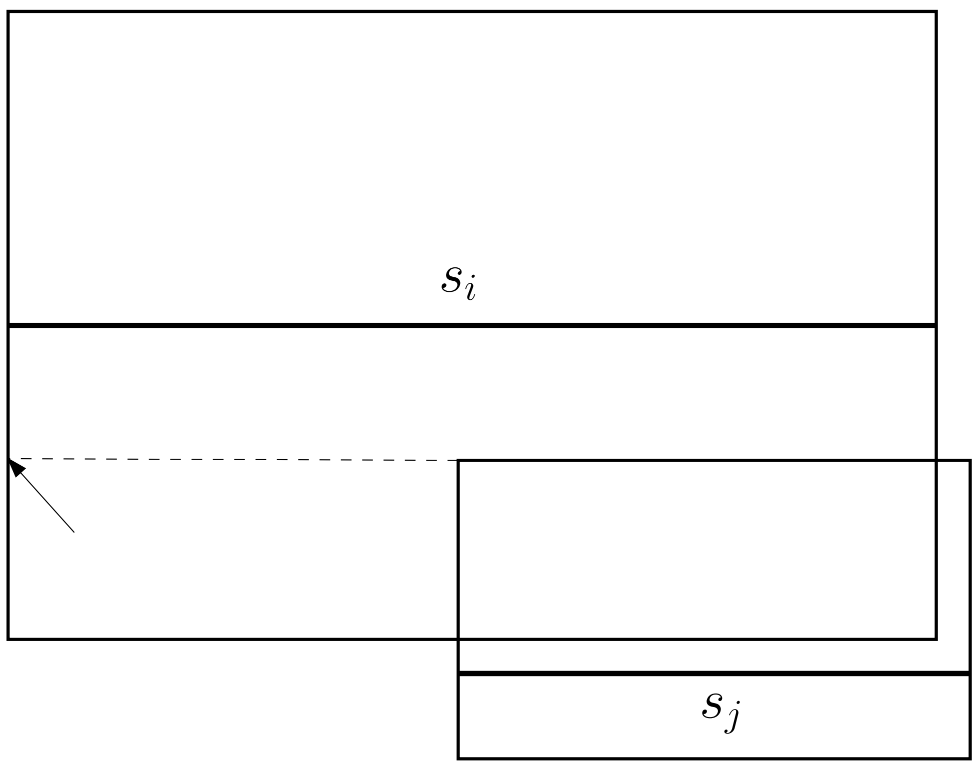


Labeling-Problem - Relevante Positionen

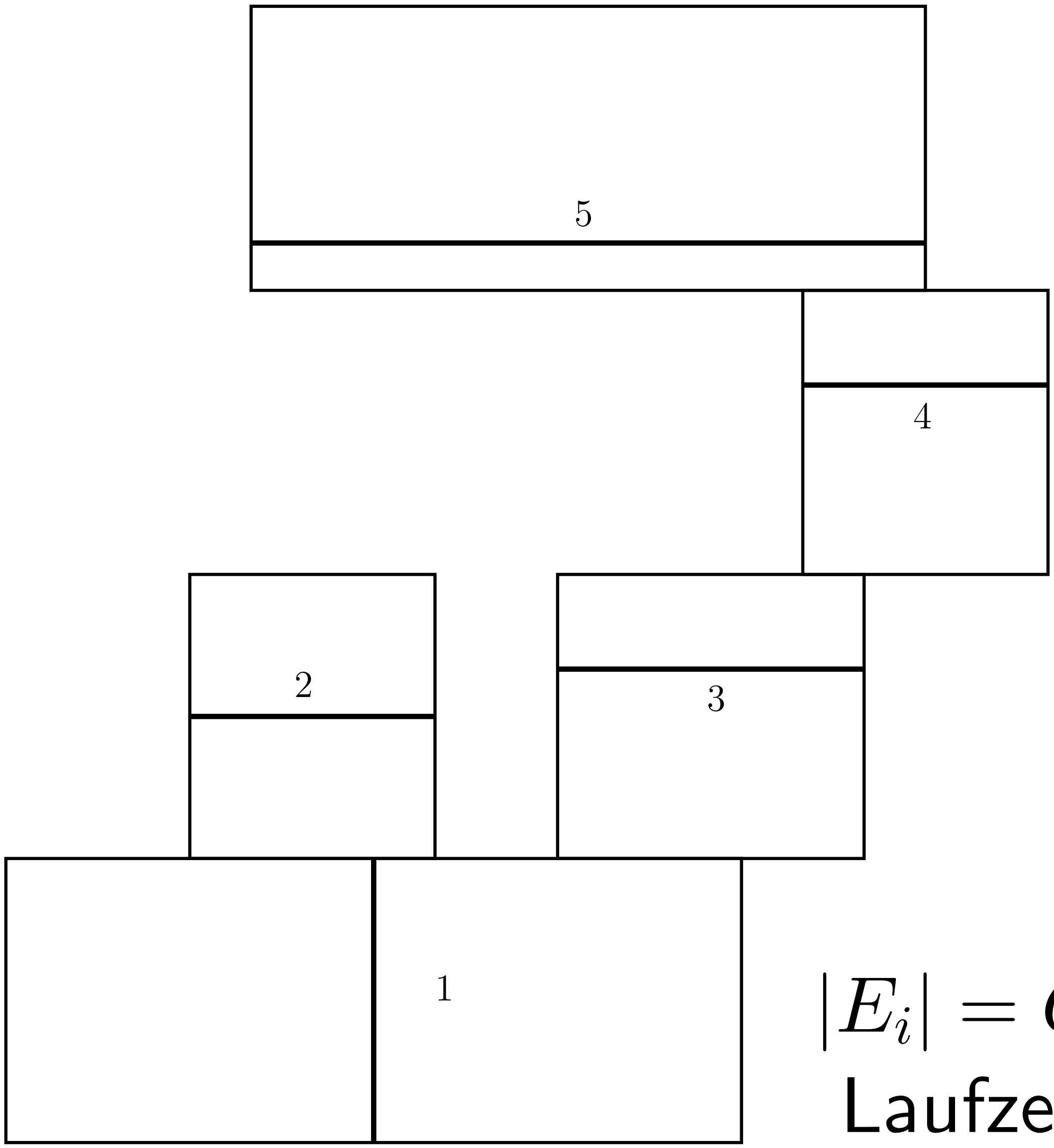
Für horizontale Strecken:

Sortierung nach y-Koordinate: $s_1, s_2, \ldots, s_n$

Für s_i: Betrachte $s_j \in \{s_1, \ldots, s_{i-1}\}$. Falls s_j *horizontal*:



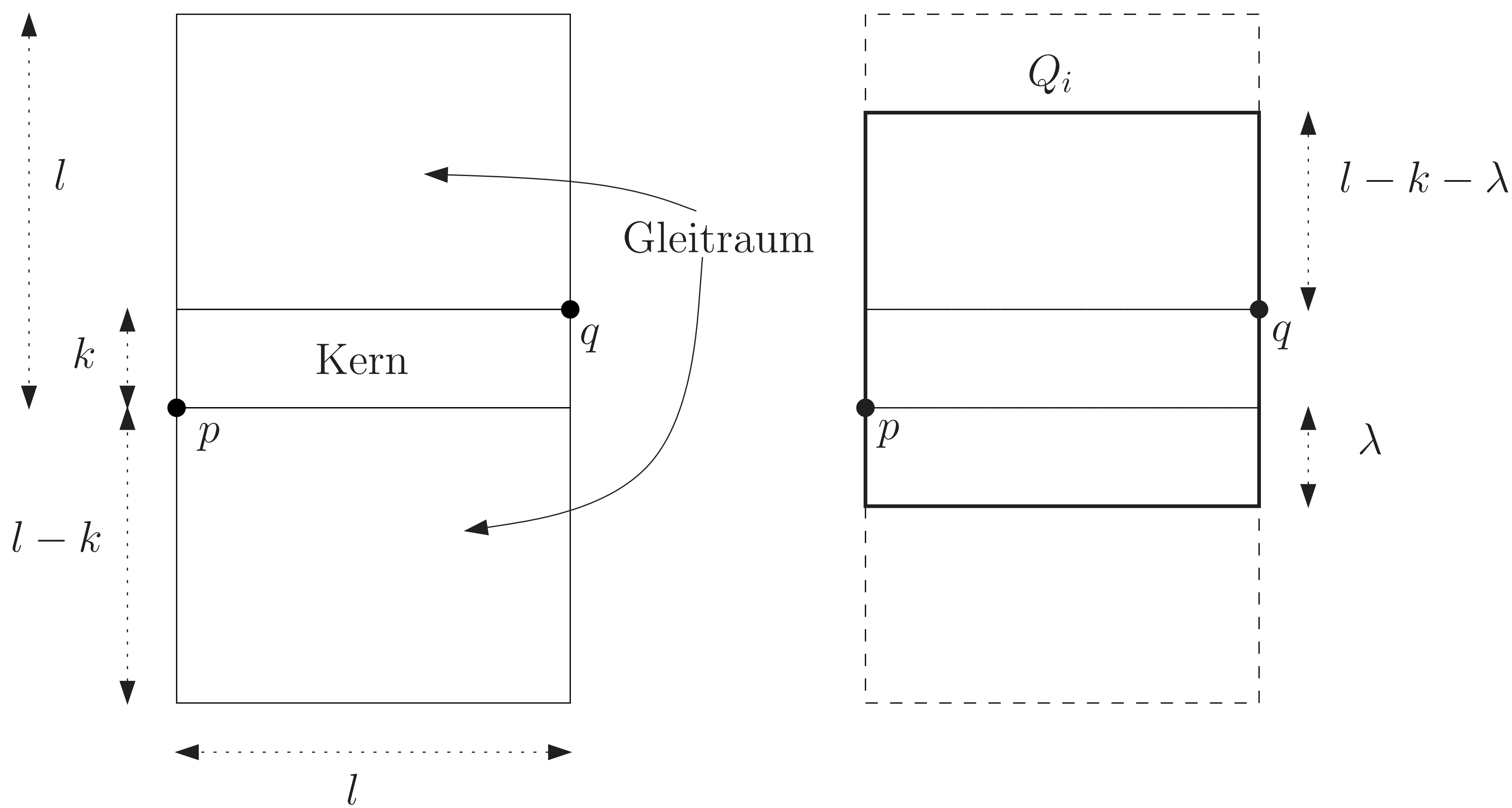
Labeling-Problem - Abschätzung



$|E_i| = O(n) \Rightarrow$

Laufzeit $O(n^2 \log^2 n)$

Quadrat-Matching



Quadrat-Matching

Unterschiede:

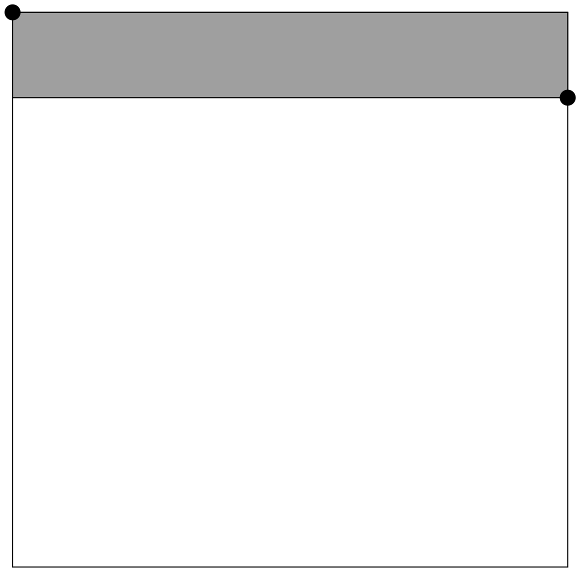
- „Dicke“ Strecken.
- Größe der Labels variiert.

Lösungsskizze:

- Schneiden sich Kerne? (evtl. Abbruch)
- Berechne relevante Positionen
- Überprüfe alle Kombinationen mit 2-SAT
- Evtl. Berührungen aufheben

Quadrat-Matching - Vollständigkeit

Sei M ein perfektes, starkes Quadrat-Matching.
Verschiebe *vertikale* Quadrate nach *links*.
Verschiebe *horizontale* Quadrate nach *unten*.

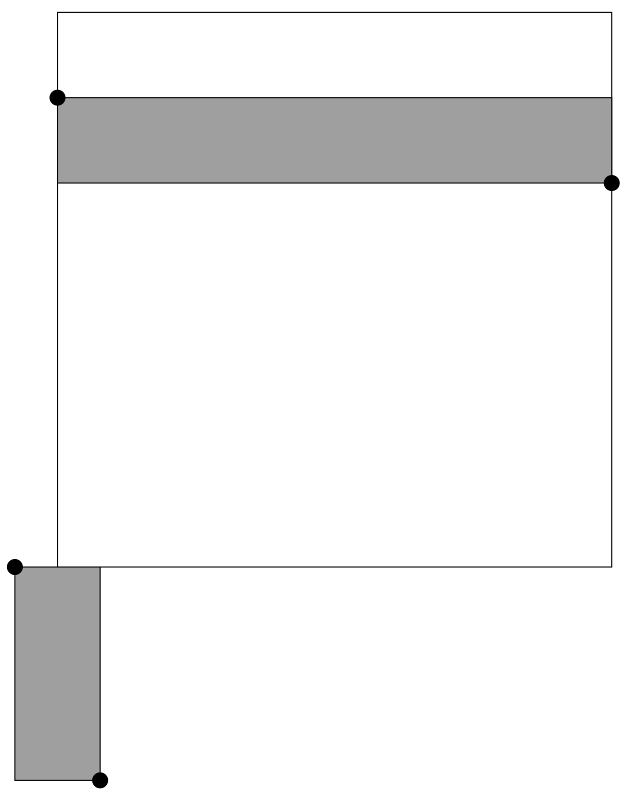


Quadrat-Matching - Vollständigkeit

Sei M ein perfektes, starkes Quadrat-Matching.

Verschiebe *vertikale* Quadrate nach *links*.

Verschiebe *horizontale* Quadrate nach *unten*.

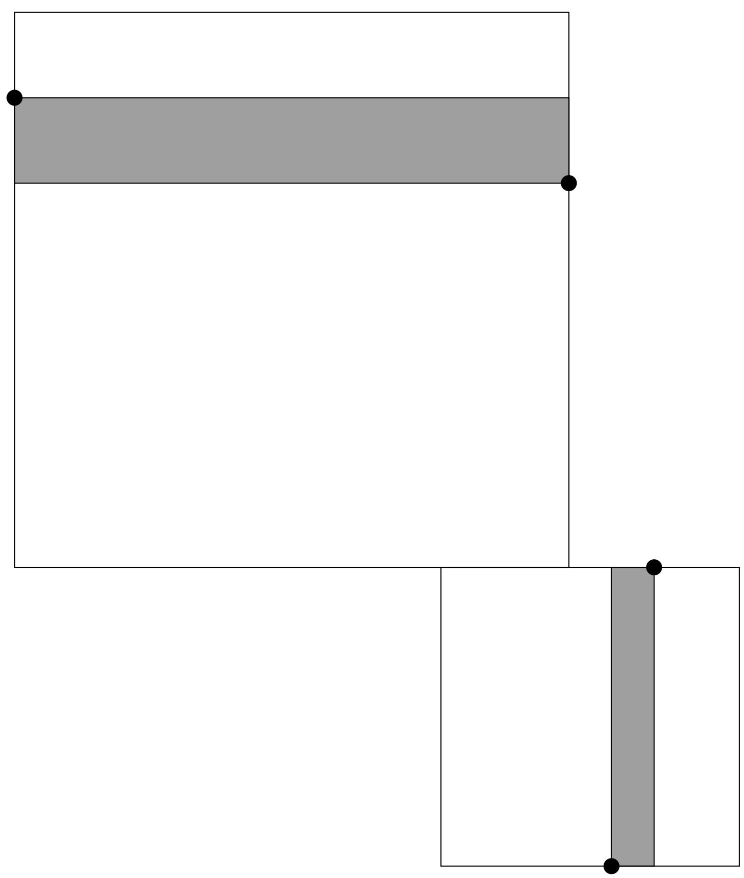


Quadrat-Matching - Vollständigkeit

Sei M ein perfektes, starkes Quadrat-Matching.

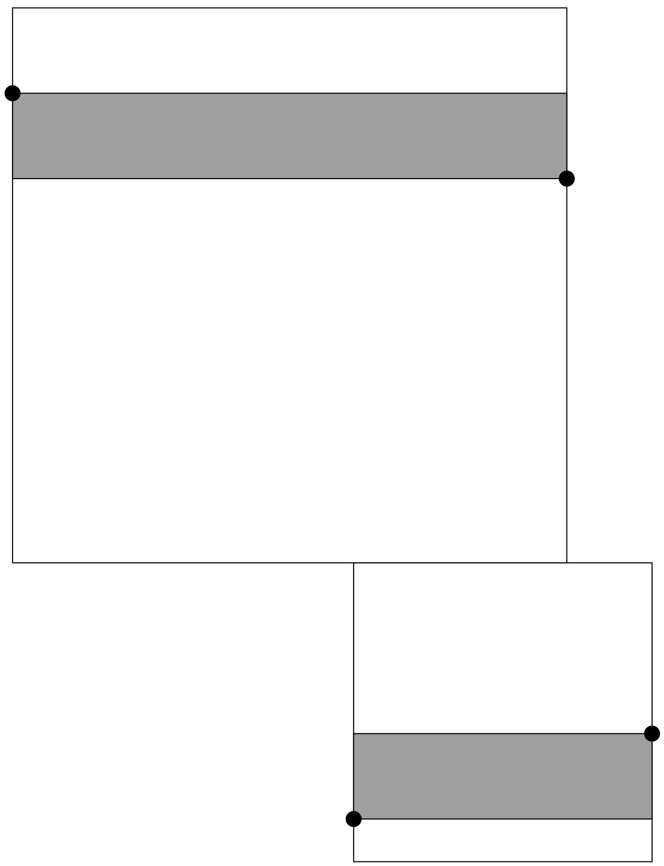
Verschiebe *vertikale* Quadrate nach *links*.

Verschiebe *horizontale* Quadrate nach *unten*.



Quadrat-Matching - Vollständigkeit

Sei M ein perfektes, starkes Quadrat-Matching.
Verschiebe *vertikale* Quadrate nach *links*.
Verschiebe *horizontale* Quadrate nach *unten*.



Quadrat-Matching - Vollständigkeit

Sei M ein perfektes, starkes Quadrat-Matching.

Verschiebe $vertikale$ Quadrate nach $links$.

Verschiebe $horizontale$ Quadrate nach $unten$.

Diese Konfiguration wurde betrachtet!

Inverse Verschiebung ergibt M.

Abschätzung

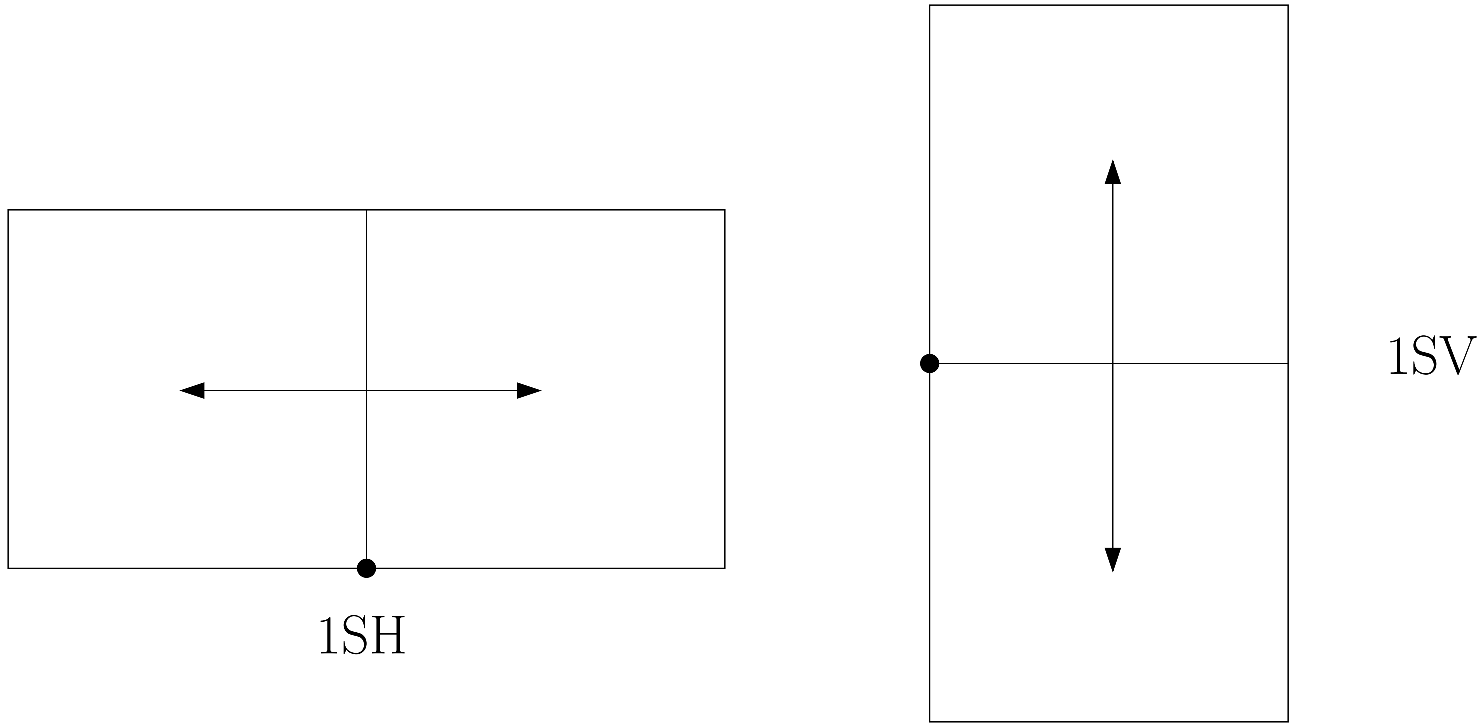
Für jede Position eines Quadrats, ist die *Extremalposition* eines gleichorientierten Quadrats verantwortlich, oder die *Existenz* eines anders orientierten Quadrats.

$\Rightarrow |E_i| = O(n)$.

$\Rightarrow$ Laufzeit $O(n^2 \log n)$.

Noch offen: *Berührungen aufheben*

Anwendung - Point Labeling



Hinzufügen von einem zweiten Punkt $\longrightarrow$ Kern.

Danke für Ihre Aufmerksamkeit!