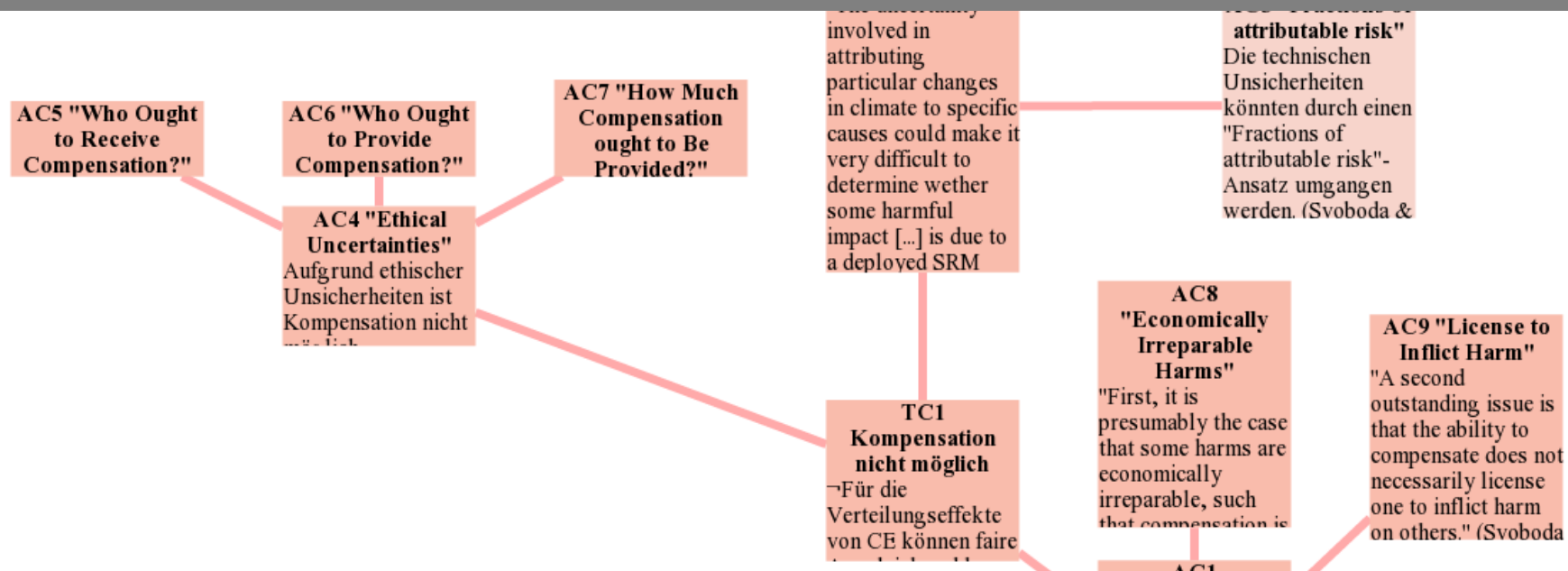


Visualisierung von Argumentkarten Inkrementelle Layouts

Seminar · February 17, 2015

Stefan Altmayer, Lukas Barth, Eric Braun, David Hopf, Frederike Neuber

INSTITUTE OF THEORETICAL INFORMATICS · ALGORITHMICS GROUP



- Das Problem
- Angedachte Lösungen
 - Force-Directed
 - Seam Carving
- Mixed Integer Program
 - Ideen zur Verbesserung
- Demo

- Das Problem
- Angedachte Lösungen
 - Force-Directed
 - Seam Carving
- Mixed Integer Program
 - Ideen zur Verbesserung
- Demo

Gegeben

- Graph vorher
- Neuer Knoten
- Evtl. Zusatzanforderungen?

Gesucht

- "Ähnliches" Layout
- ..., das Zusatzanforderungen erfüllt

Das Problem

- Was ist "Ähnlichkeit"?

- Was ist "Ähnlichkeit"?
 - Absolute Positionen gleich
 - Kanten gleich geroutet

- Was ist "Ähnlichkeit"?
 - Absolute Positionen gleich
 - Kanten gleich geroutet

- Relative Positionen gleich

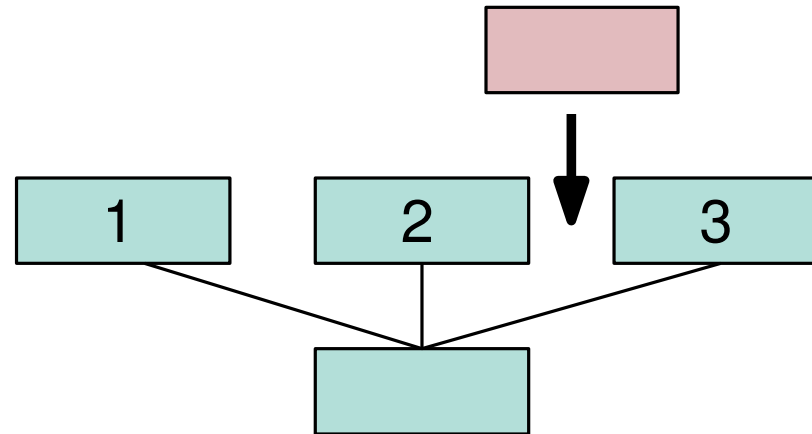
- Was ist "Ähnlichkeit"?
 - Absolute Positionen gleich
 - Kanten gleich geroutet

- Relative Positionen gleich

- Relative Positionen ähnlich
- Reihenfolge gleich

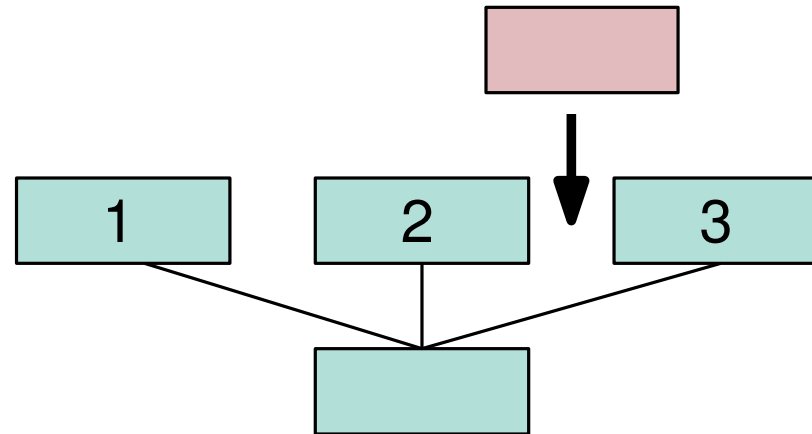
Wunschliste...

- Versucht nicht, zu klug zu sein!
- Viele Dinge sind schon in der richtigen Reihenfolge!



Wunschliste...

- Versucht nicht, zu klug zu sein!
- Viele Dinge sind schon in der richtigen Reihenfolge!



- Knoten aneinander ausrichten können
- "Starre" Gebilde

Prinzipiell: Wo Platz hernehmen?

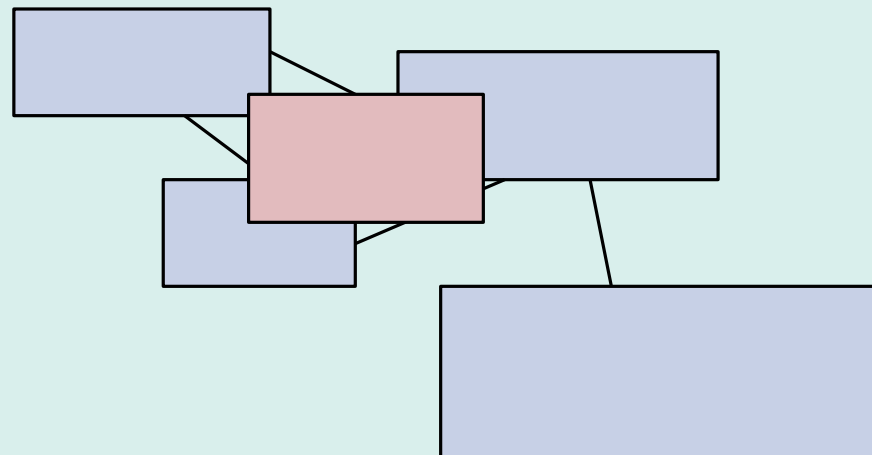
Prinzipiell: Wo Platz hernehmen?

Verschieben

- Geringe Platzverschwendung
- Nicht klar definiert
- Überlappungen?
- Relative Positionen ändern sich

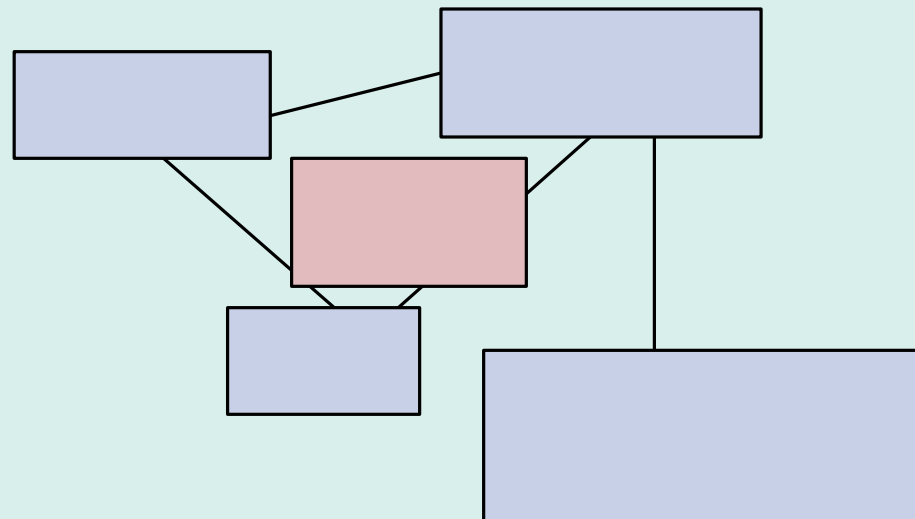
Prinzipiell: Wo Platz hernehmen?

Vergleichen



Prinzipiell: Wo Platz hernehmen?

Vorgehensweise



Prinzipiell: Wo Platz hernehmen?

Verschieben

- Geringe Platzverschwendung
- Nicht klar definiert
- Überlappungen?
- Relative Positionen ändern sich

Prinzipiell: Wo Platz hernehmen?

Verschieben

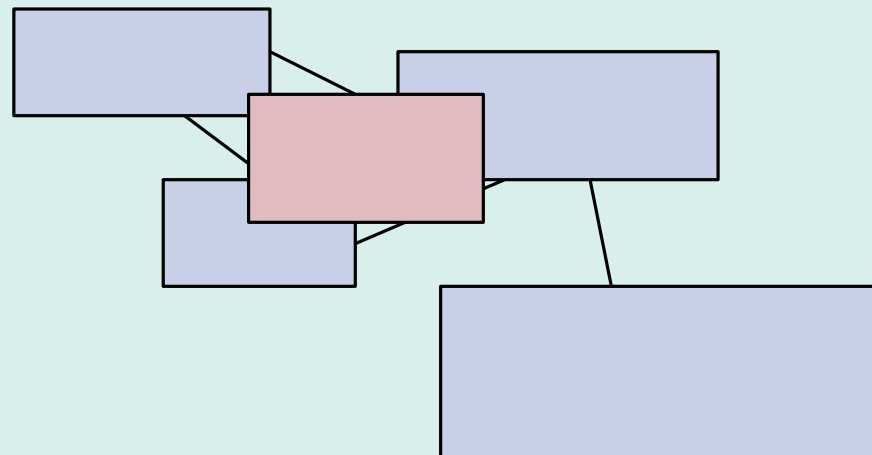
- Geringe Platzverschwendung
- Nicht klar definiert
- Überlappungen?
- Relative Positionen ändern sich

Skalieren

- Super-Einfach
- Keine Überlappungen
- Viel verschwendeter Platz
- Alles wird kleiner

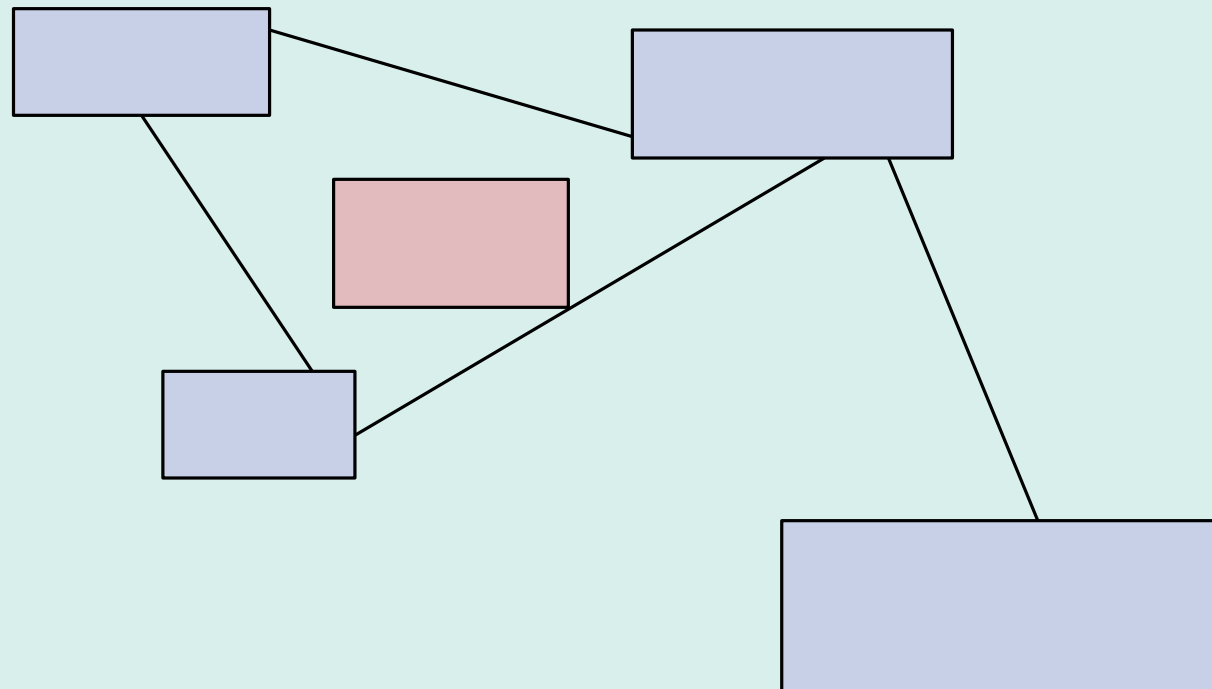
Prinzipiell: Wo Platz hernehmen?

Vorgehensweise



Prinzipiell: Wo Platz hernehmen?

Vorgehensweise



Prinzipiell: Wo Platz hernehmen?

Verschieben

- Geringe Platzverschwendung
- Nicht klar definiert
- Überlappungen?
- Relative Positionen ändern sich

Skalieren

- Super-Einfach
- Keine Überlappungen
- Viel verschwendeter Platz
- Alles wird kleiner

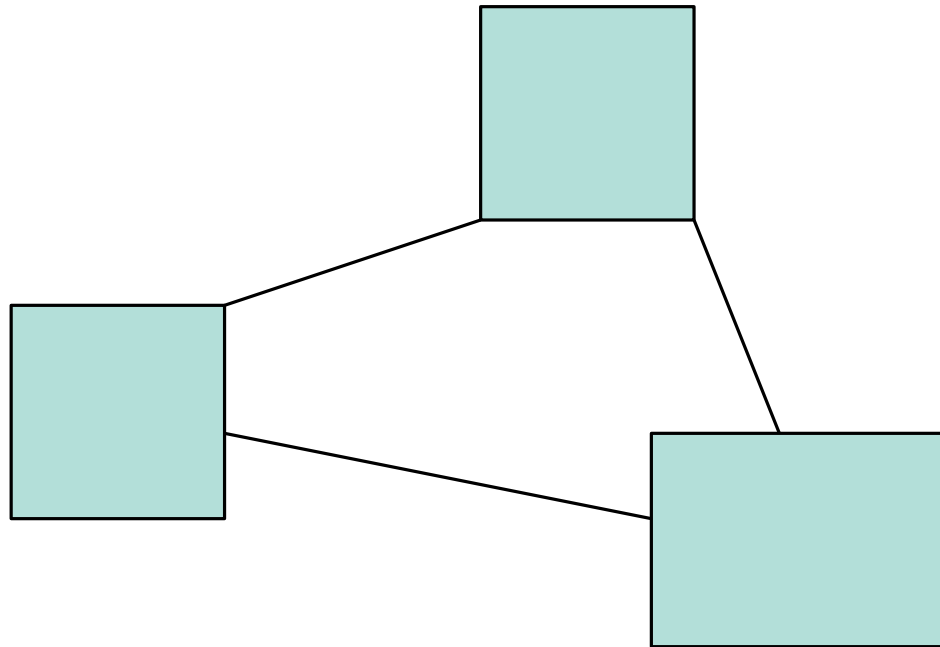
- Das Problem
- Angedachte Lösungen
 - Force-Directed
 - Seam Carving
- Mixed Integer Program
 - Ideen zur Verbesserung
- Demo

Idee

- Anker an vorherigen Positionen platzieren
- Sonst Fruchterman-Reingold o.ä.

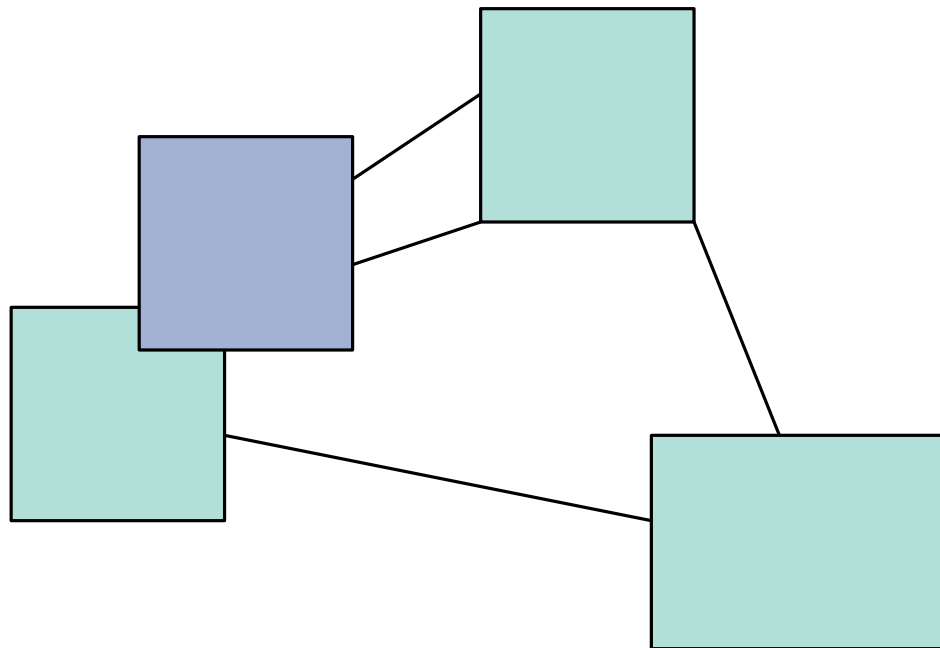
Idee

- Anker an vorherigen Positionen platzieren
- Sonst Fruchterman-Reingold o.ä.



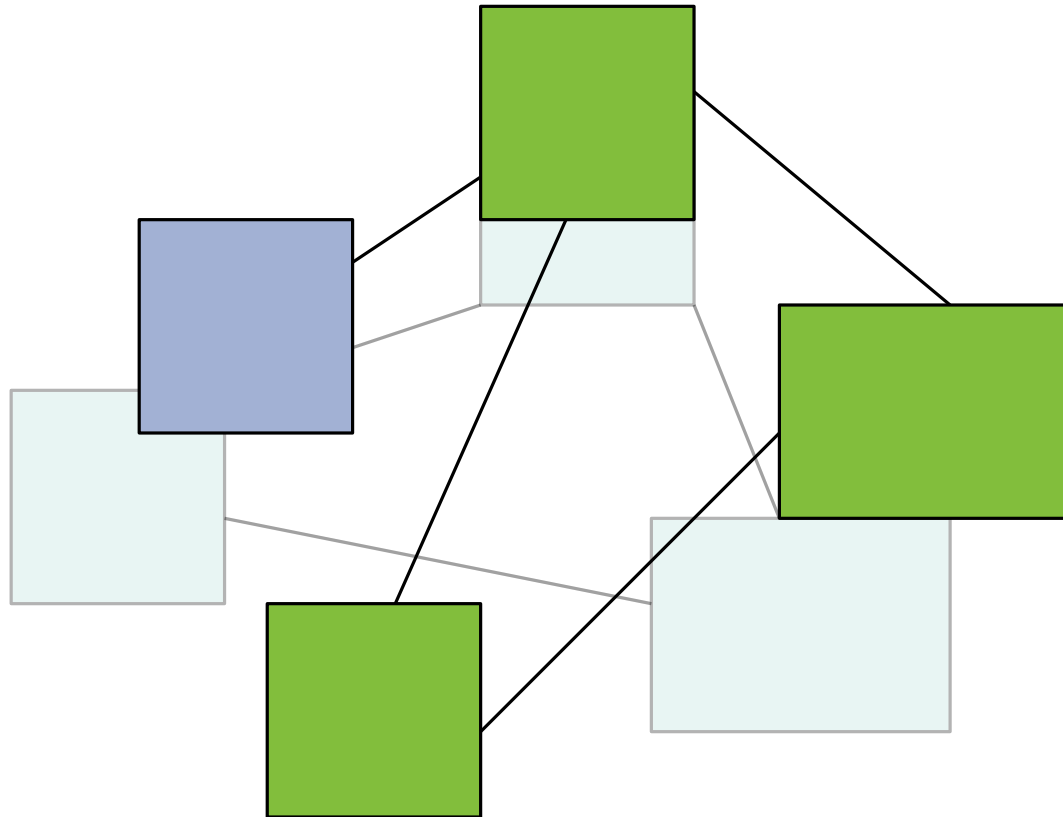
Idee

- Anker an vorherigen Positionen platzieren
- Sonst Fruchterman-Reingold o.ä.



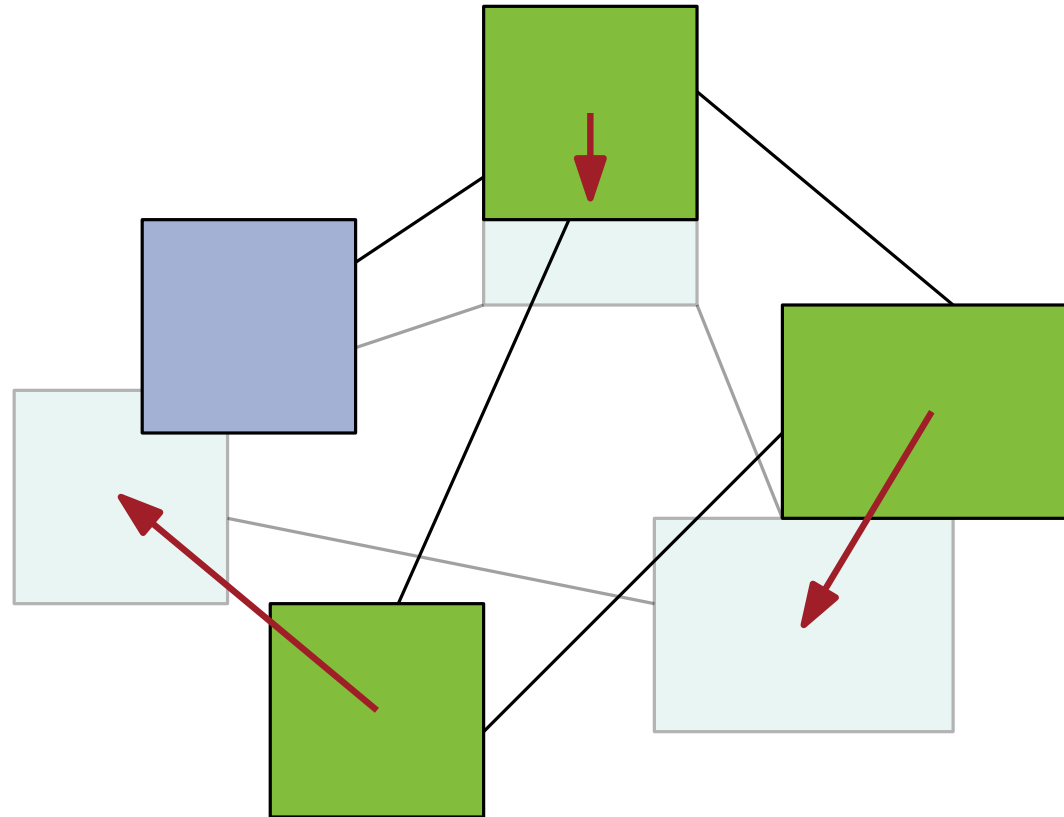
Idee

- Anker an vorherigen Positionen platzieren
- Sonst Fruchterman-Reingold o.ä.



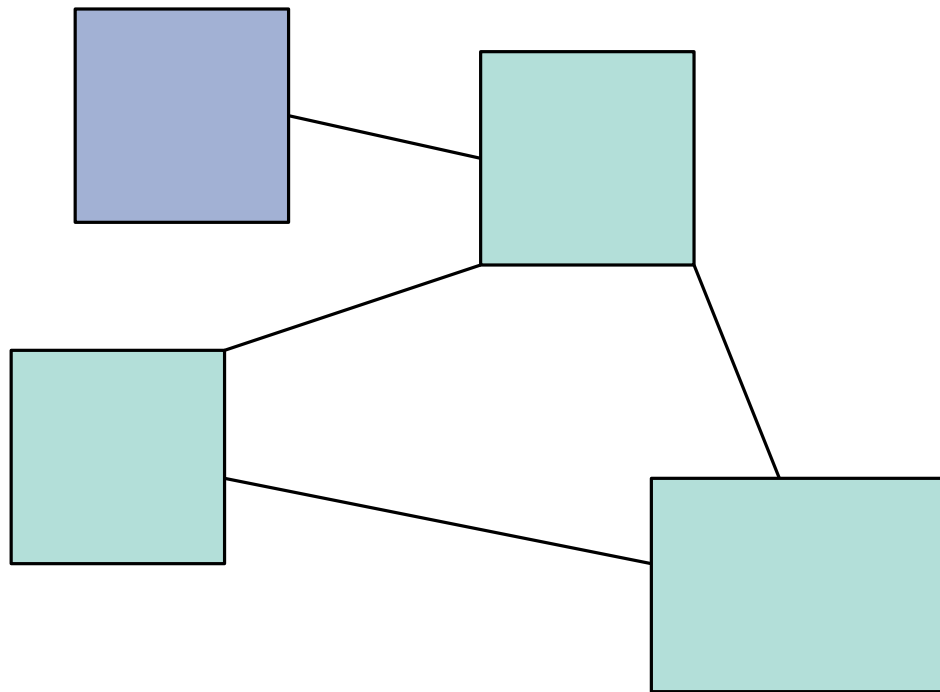
Idee

- Anker an vorherigen Positionen platzieren
- Sonst Fruchterman-Reingold o.ä.



Idee

- Anker an vorherigen Positionen platzieren
- Sonst Fruchterman-Reingold o.ä.

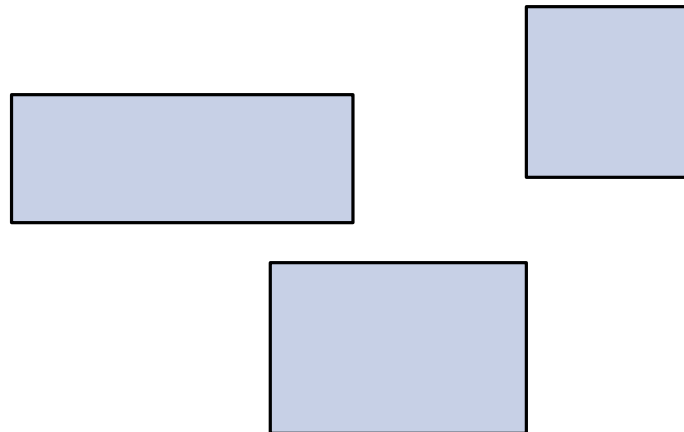


Die Idee

1. Platz schaffen durch Skalieren
2. Knoten einfügen
3. Überflüssigen Platz entfernen

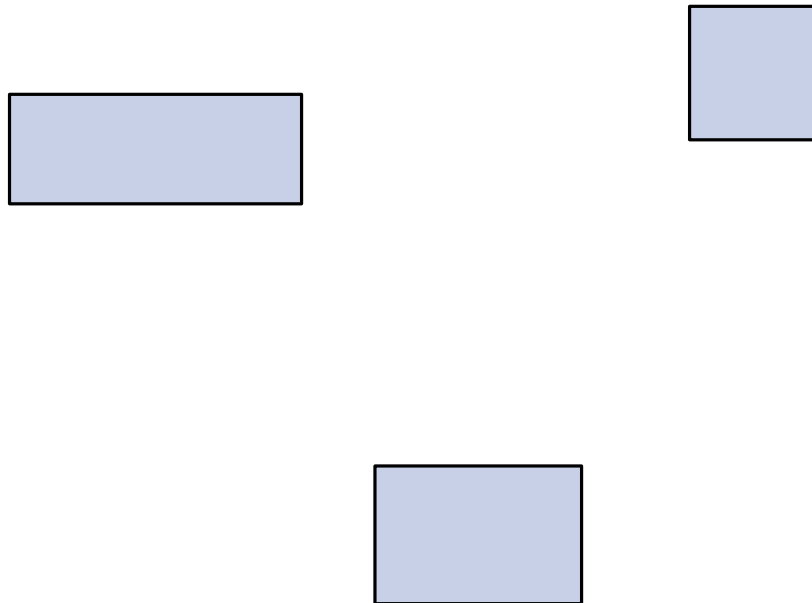
Die Idee

1. Platz schaffen durch Skalieren
2. Knoten einfügen
3. Überflüssigen Platz entfernen



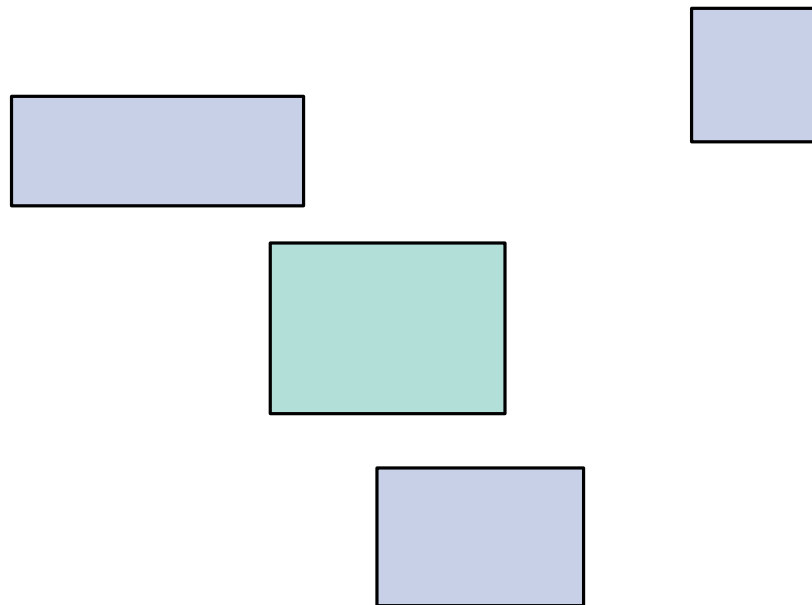
Die Idee

1. Platz schaffen durch Skalieren
2. Knoten einfügen
3. Überflüssigen Platz entfernen



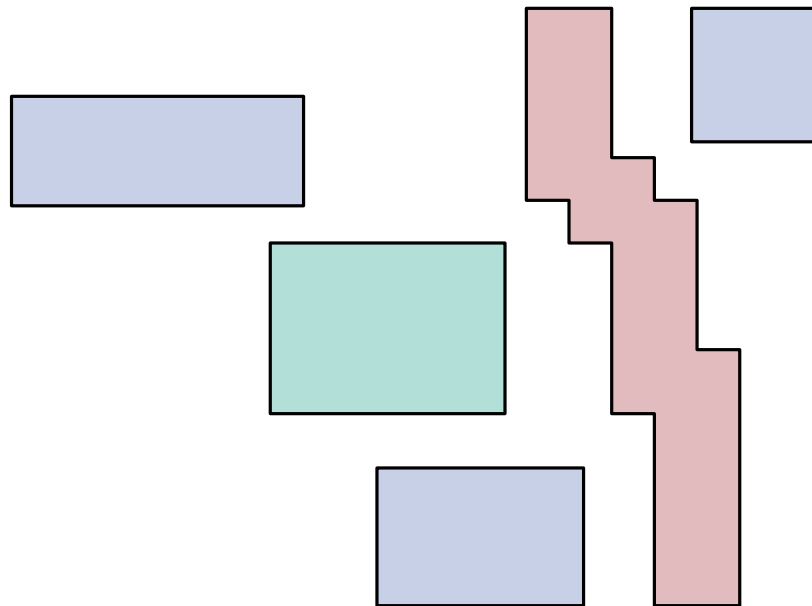
Die Idee

1. Platz schaffen durch Skalieren
2. Knoten einfügen
3. Überflüssigen Platz entfernen



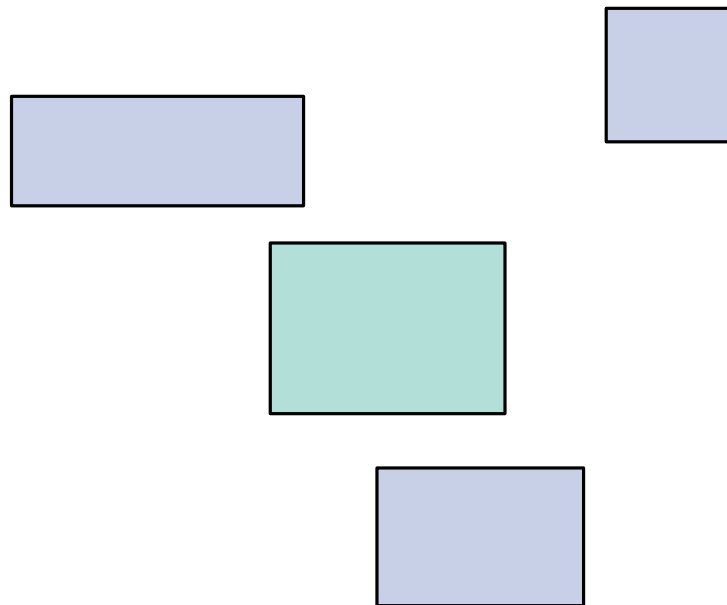
Die Idee

1. Platz schaffen durch Skalieren
2. Knoten einfügen
3. Überflüssigen Platz entfernen



Die Idee

1. Platz schaffen durch Skalieren
2. Knoten einfügen
3. Überflüssigen Platz entfernen



- Das Problem
- Angedachte Lösungen
 - Force-Directed
 - Seam Carving
- Mixed Integer Program
 - Ideen zur Verbesserung
- Demo

Einfügen

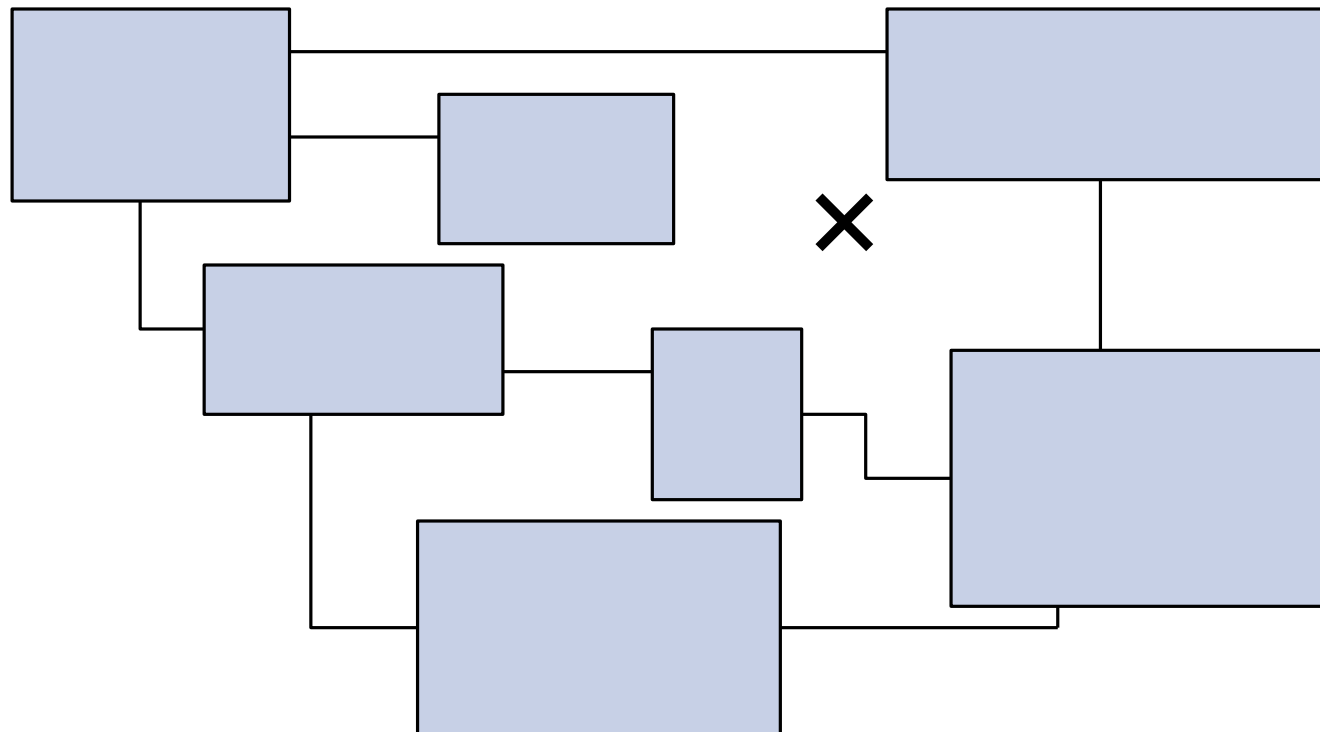
- Erhält relative Positionen
- Verhindert Überlappungen
- Skaliert nicht

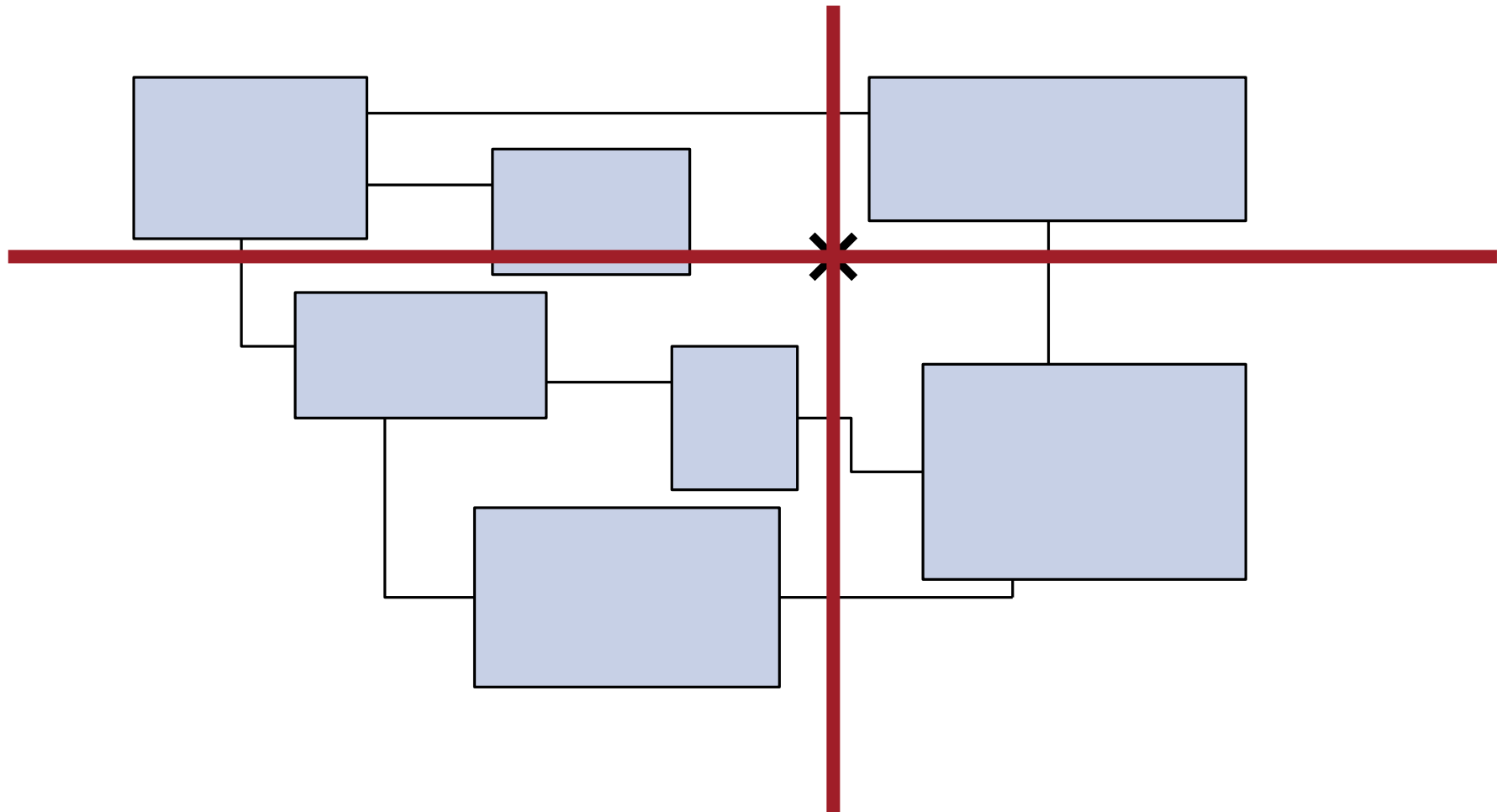
Einfügen

- Erhält relative Positionen
- Verhindert Überlappungen
- Skaliert nicht

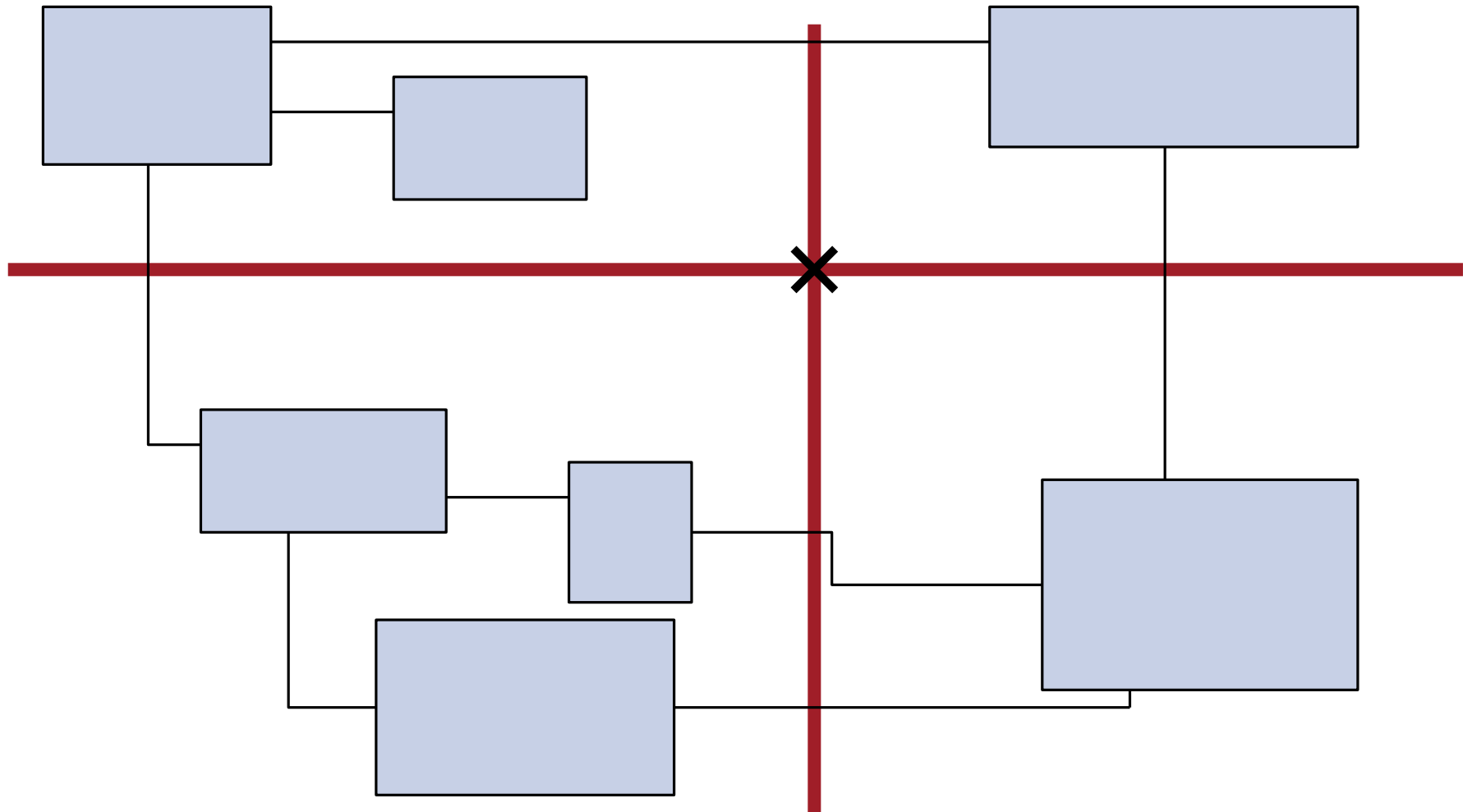
„Zurechtrücken“

- minimiert Platzverschwendung
- flexible Anpassung an Wünsche

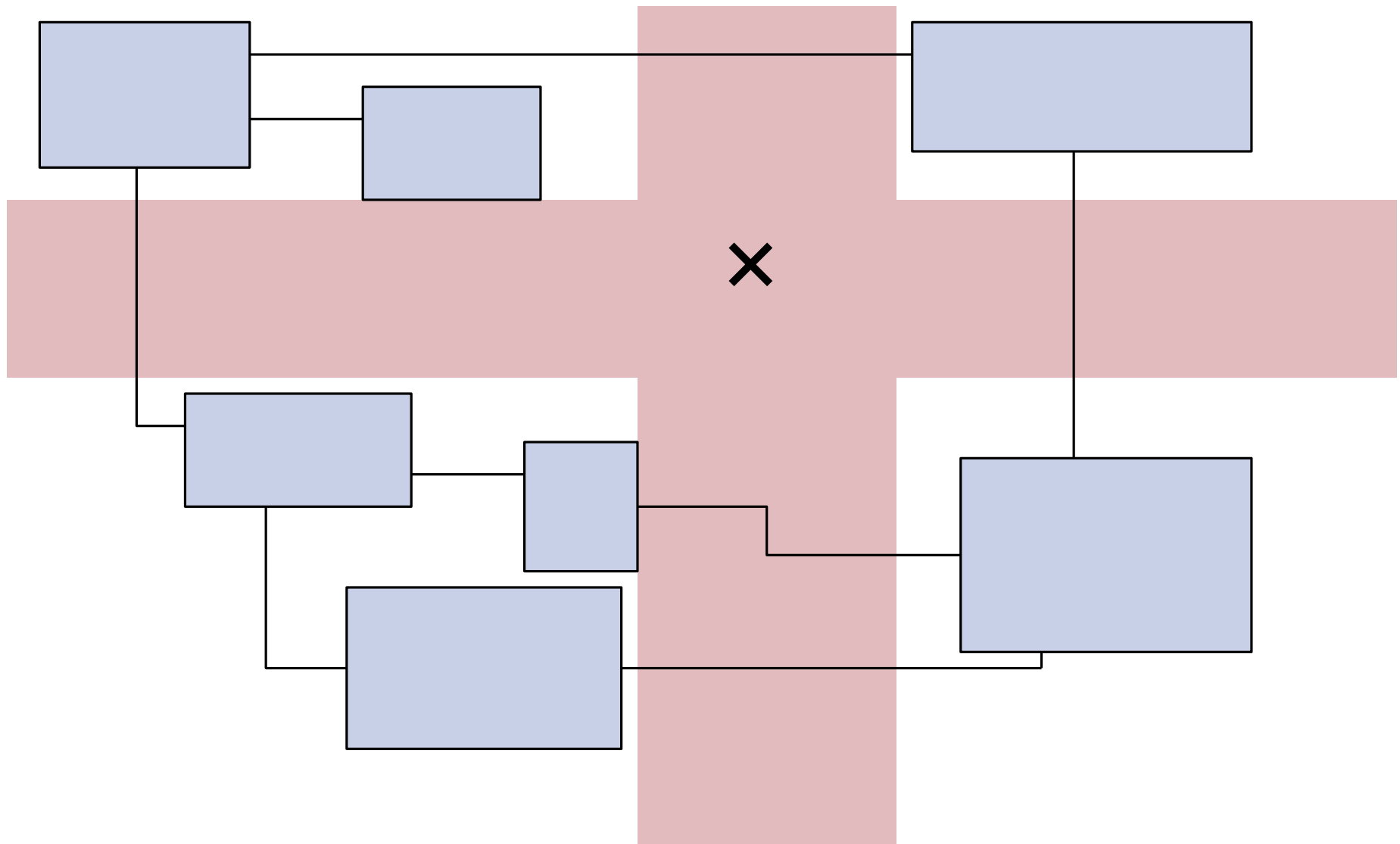




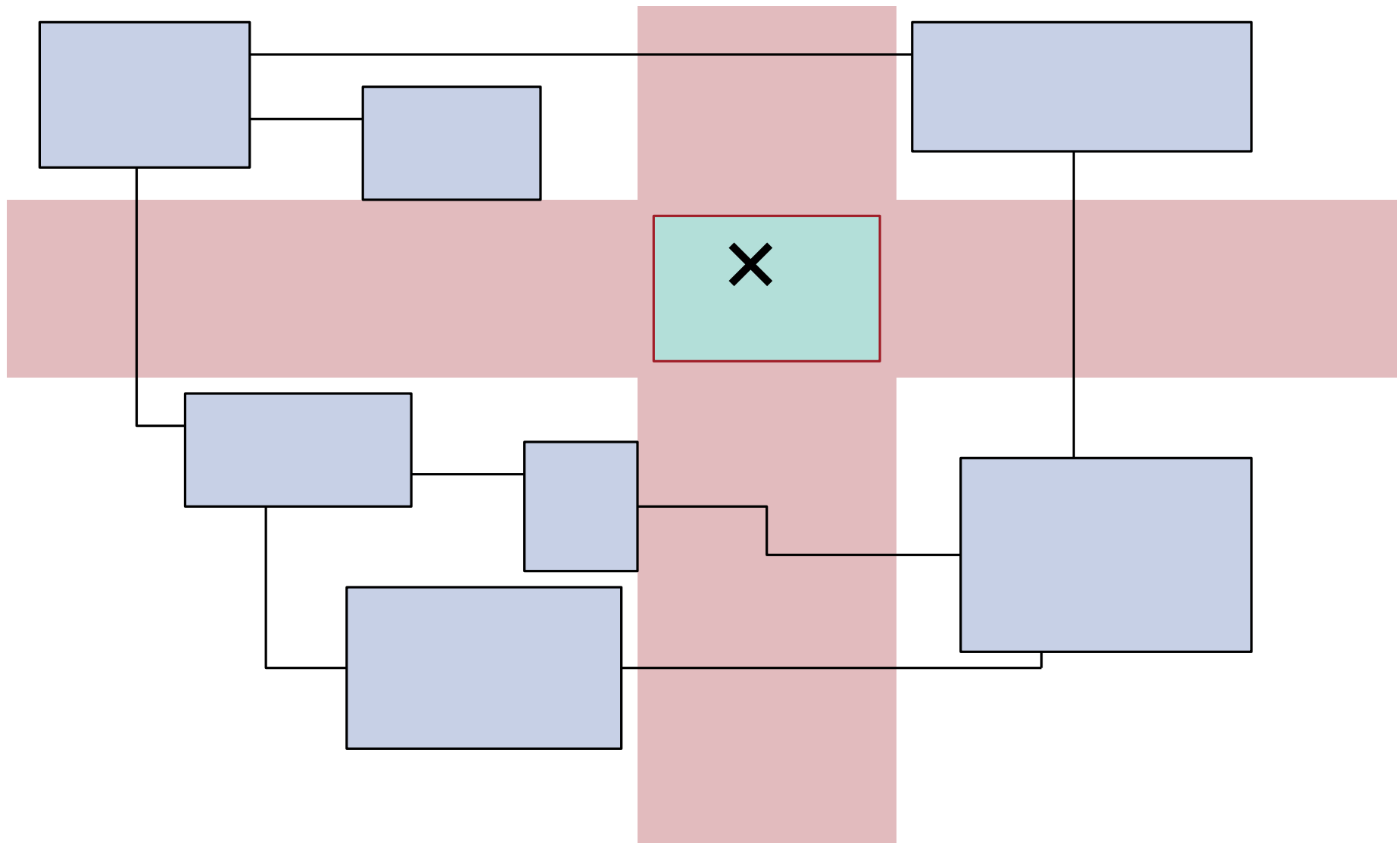
Einfügen



Einfügen



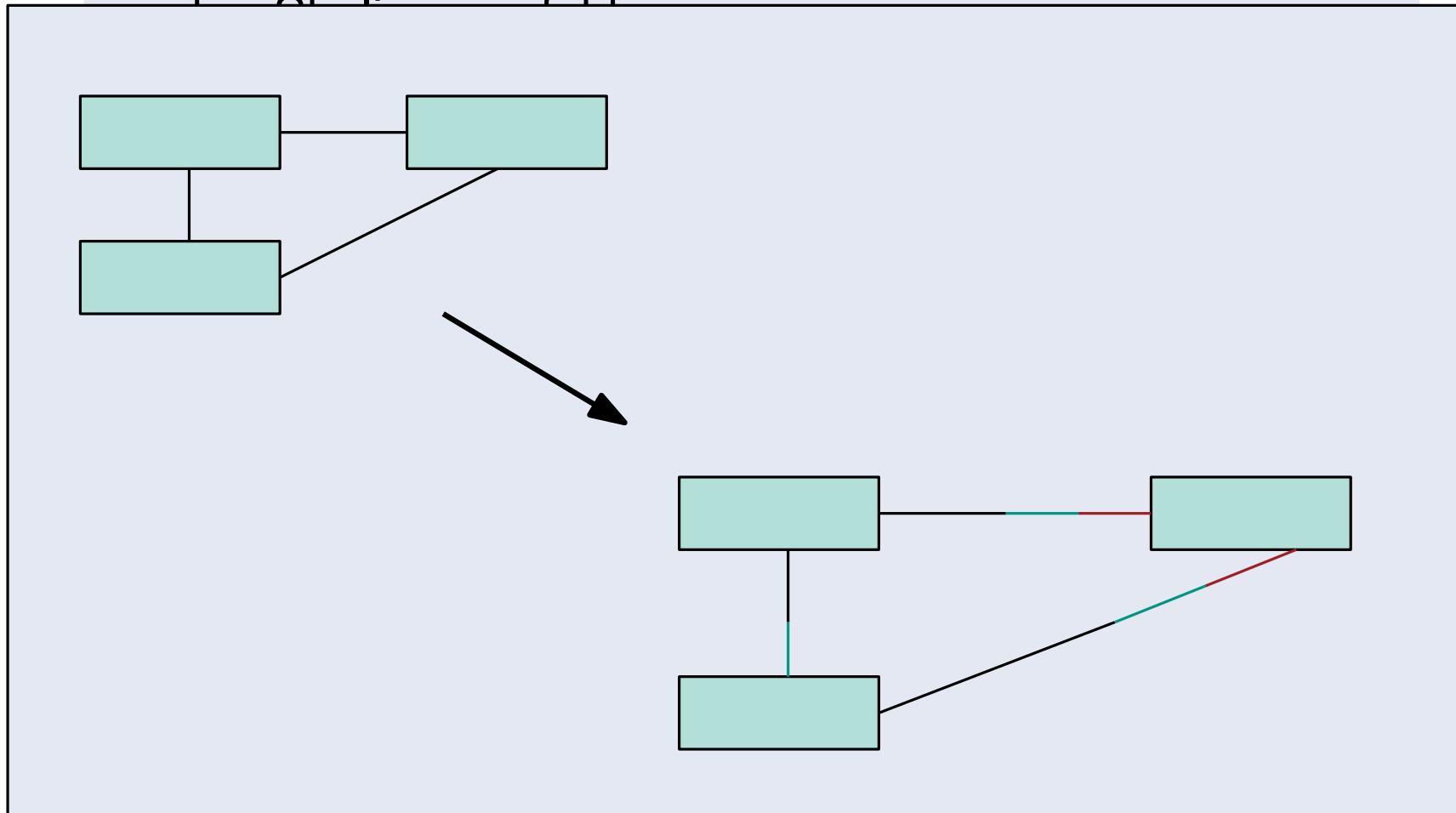
Einfügen



Minimiert

- den Skalierungsfaktor
- die Abweichung von den skalierten Wunschpositionen

Minimiert



Minimiert

- den Skalierungsfaktor
- die Abweichung von den skalierten Wunschpositionen

Verhindert

- (Knoten-)Überlappungen
- Vertauschungen von Knoten

Minimiert

- den Skalierungsfaktor
- die Abweichung von den skalierten Wunschpositionen
- Die Anzahl der Vertauschungen

Verhindert

- (Knoten-)Überlappungen

Minimiert

- den Skalierungsfaktor
- die Abweichung von den skalierten Wunschpositionen
- Die Anzahl der Vertauschungen

Verhindert

- (Knoten-)Überlappungen

Aber: Nur Vertauschungen in der Ordnung benachbarter Knoten!

Instanz mit 15 Knoten:

Constraints	411
Variablen	311 (258 Binär)
Größter Koeffizient	8141
Laufzeit	0.05 s

Mehr Vertauschungen

- Indikatorvariablen für beliebige Paare
- Transitivität erzwingen (Kubisch!)

Mehr Vertauschungen

- Indikatorvariablen für beliebige Paare
- Transitivität erzwingen (Kubisch!)

Beliebige Bedingungen

- Knoten aneinander fixieren

Mehr Vertauschungen

- Indikatorvariablen für beliebige Paare
- Transitivität erzwingen (Kubisch!)

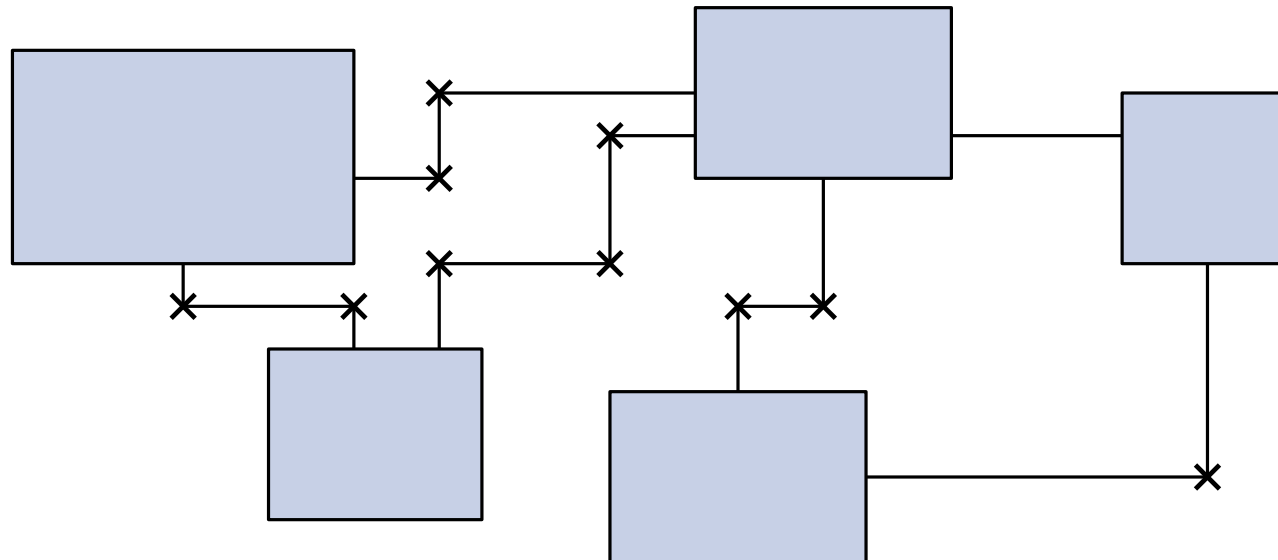
Beliebige Bedingungen

- Knoten aneinander fixieren

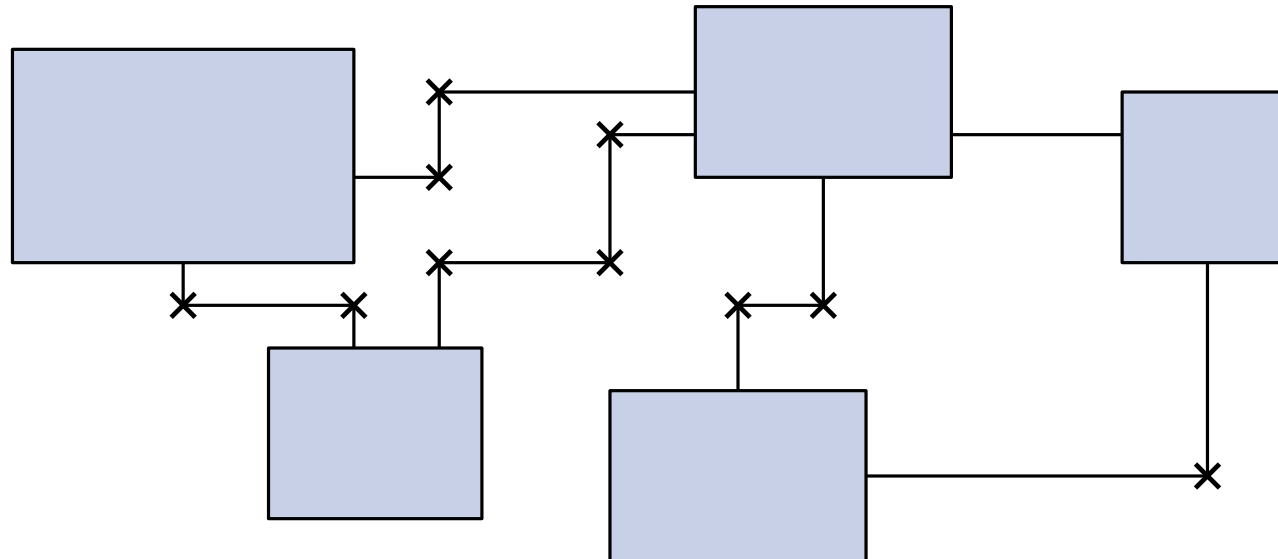
Komplexere ILPs...

- könnten live ge-updatet werden

Kantenrouting!



Kantenrouting!



- Seitenvertauschungen?
- Neue Knicke?

Trommelwirbel

Trommelwirbel

Ausbaufähig

- Positionierung des neuen Knoten ist schwierig
- Aber: Es wird "zurechtgerückt"
- Abwägung der Konstanten?
 - Skalierung vs. Verschiebung
 - Wie schlimm sind Vertauschungen?
 - Verschiebung des eingefügten Knotens vs. alte Knoten?