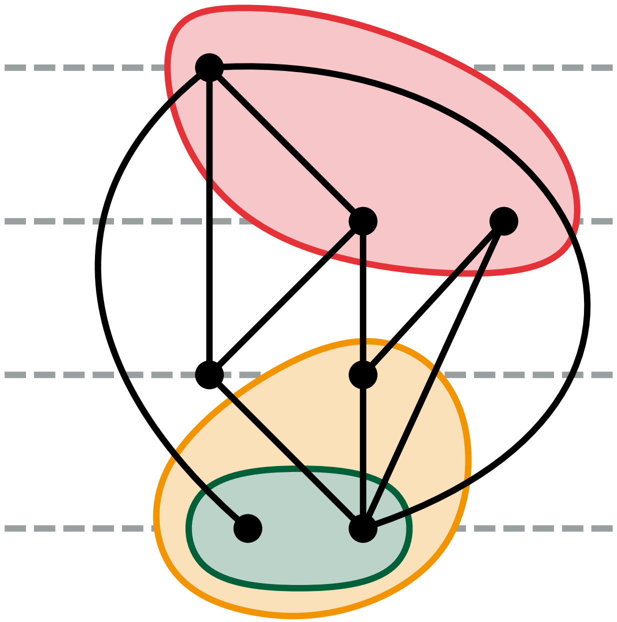
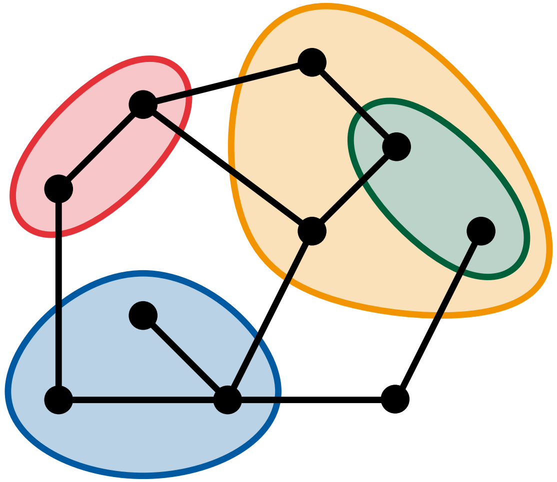


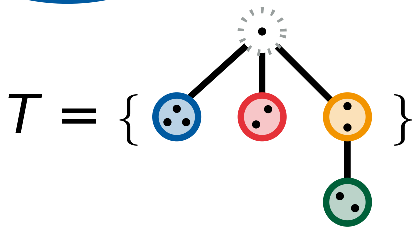
Clustered Planarity Variants for Level Graphs

Simon D. Fink, Matthias Pfretzschner, Ignaz Rutter,
Marie D. Sieper

Clustered Planarity

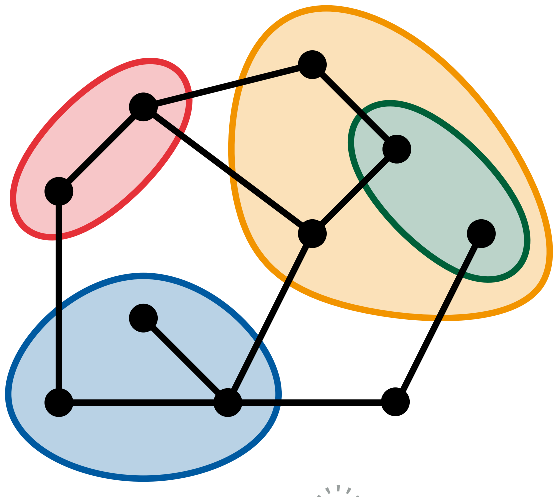
[Lengauer ’89], [Feng et al. ’95]

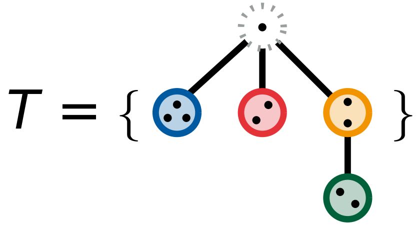




Clustered Planarity

[Lengauer ’89], [Feng et al. ’95]

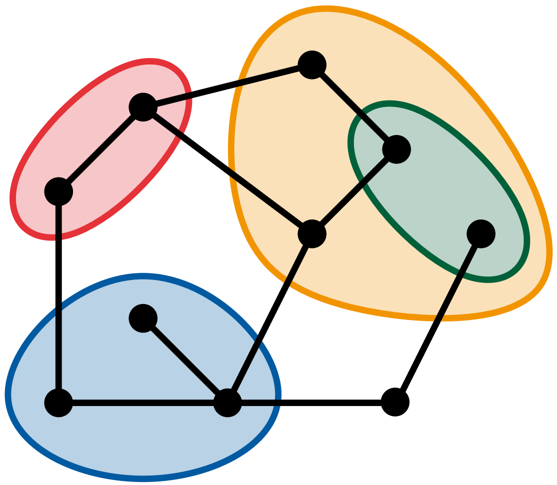


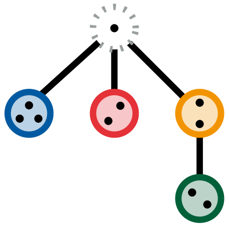


Given graph G, clustering T, we seek a planar drawing where:

Clustered Planarity

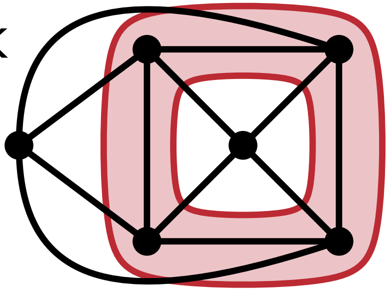
[Lengauer '89], [Feng et al. '95]



$T = \{$  $\}$

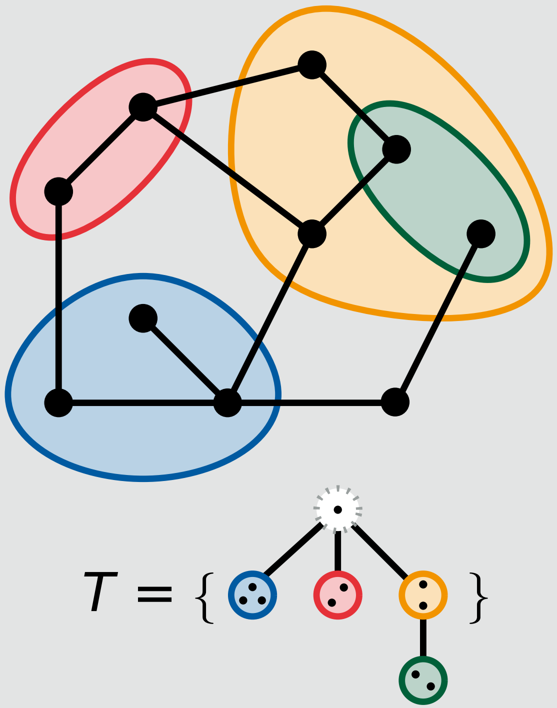
Given graph G, clustering T, we seek a planar drawing where:

- Clusters are represented by hole-free regions...



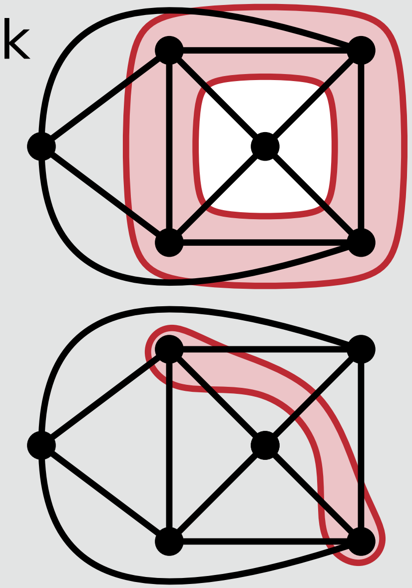
Clustered Planarity

[Lengauer '89], [Feng et al. '95]



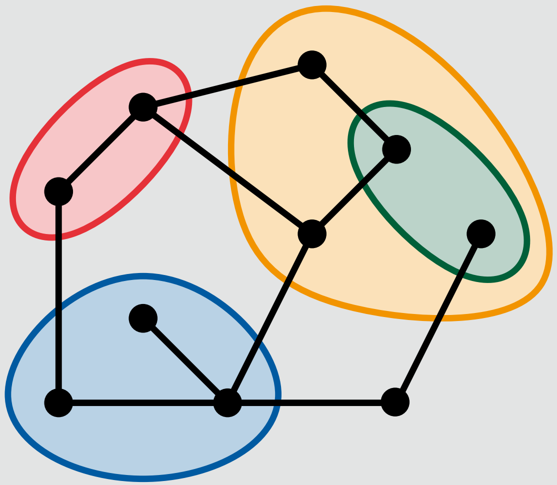
Given graph G, clustering T, we seek a planar drawing where:

- Clusters are represented by hole-free regions...
- ...that cross no unconcerned edges or other clusters



Clustered Planarity

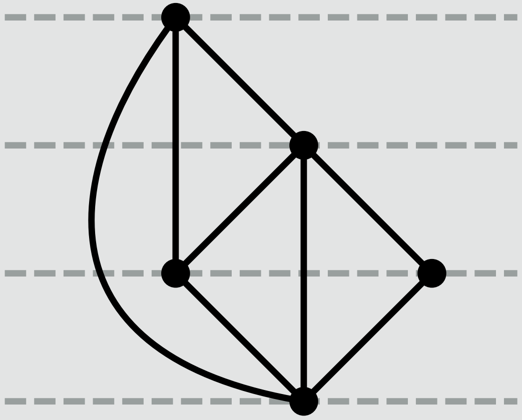
[Lengauer ’89], [Feng et al. ’95]



Given graph G,
function $\gamma\colon V \to \{1,\ldots,k\}$,
we seek a planar drawing
where:

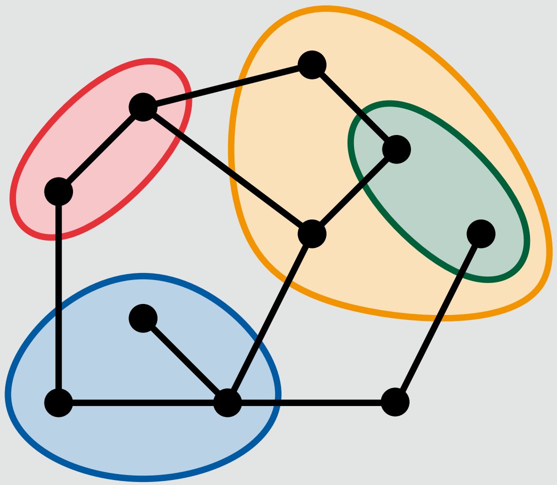
Level Planarity

[Di Battista & Nardelli ’88]



Clustered Planarity

[Lengauer ’89], [Feng et al. ’95]

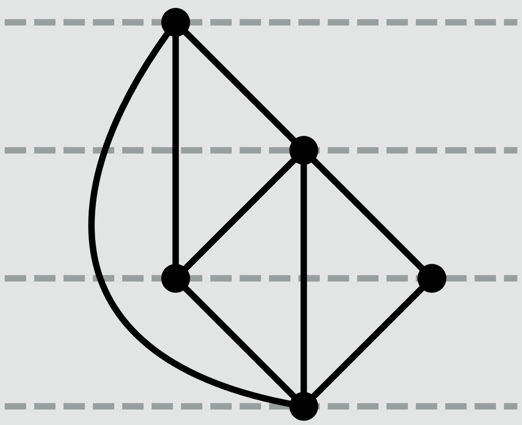


Given graph G,
function $\gamma\colon V \to \{1, \dots, k\}$,
we seek a planar drawing
where:

- every $v \in V$ has y-coordinate $\gamma(v)$

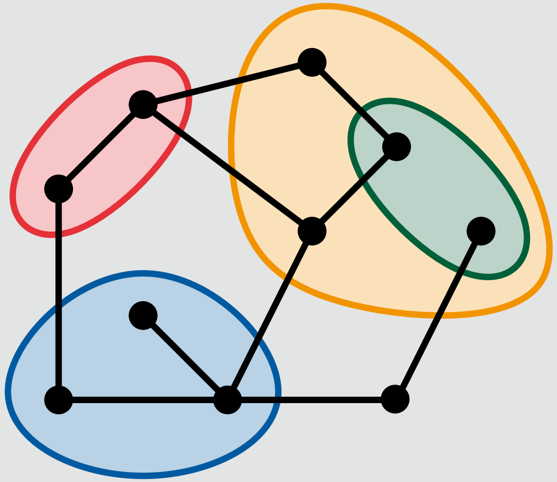
Level Planarity

[Di Battista & Nardelli ’88]



Clustered Planarity

[Lengauer '89], [Feng et al. '95]

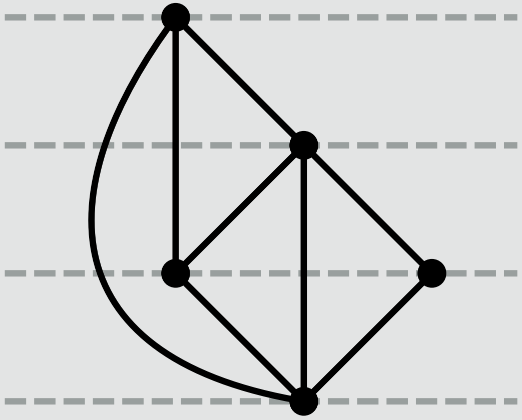


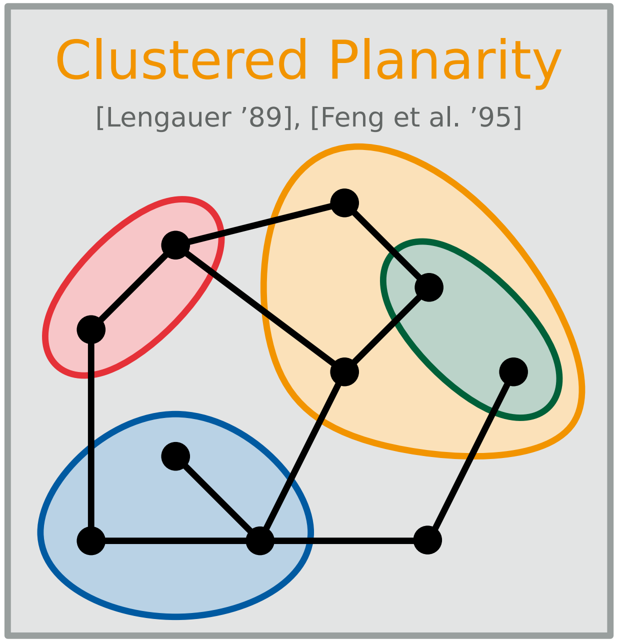
Given graph G,
function $\gamma\colon V \to \{1, \ldots, k\}$,
we seek a planar drawing
where:

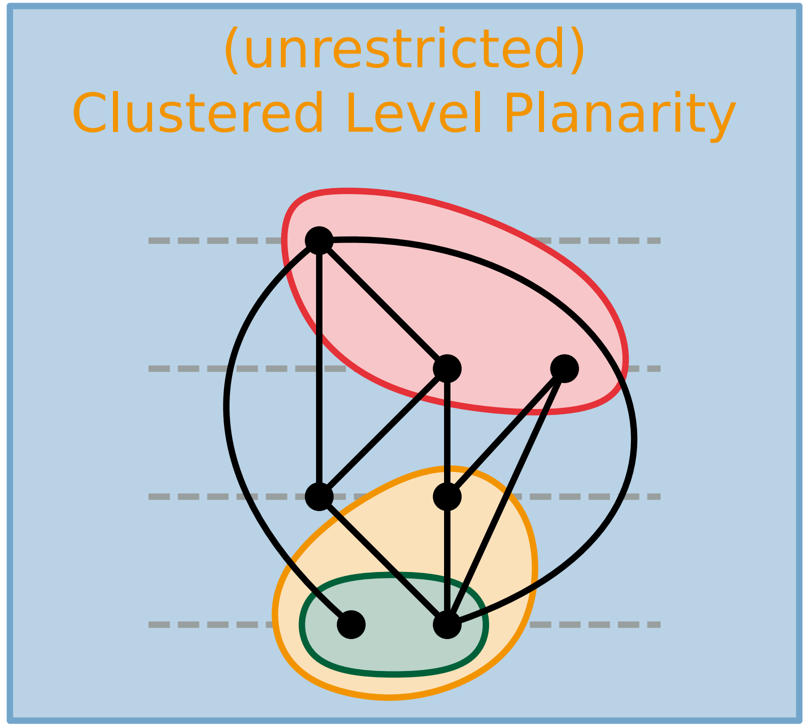
- every $v \in V$ has y-coordinate $\gamma(v)$
- edges are y-monotone

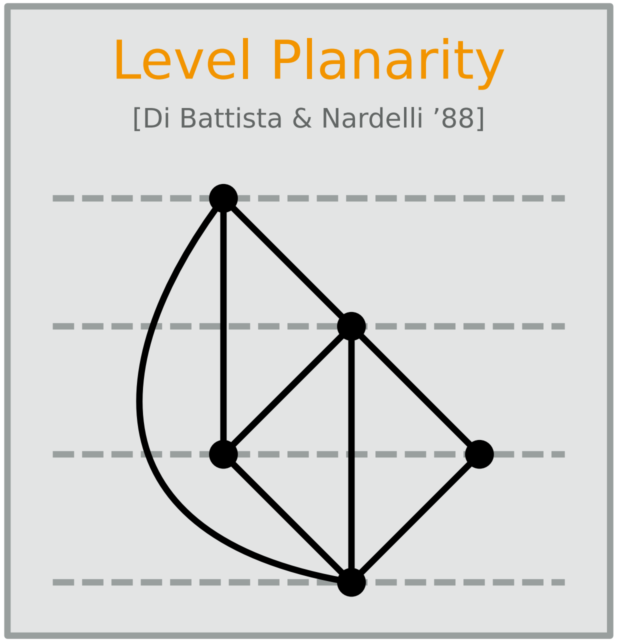
Level Planarity

[Di Battista & Nardelli '88]







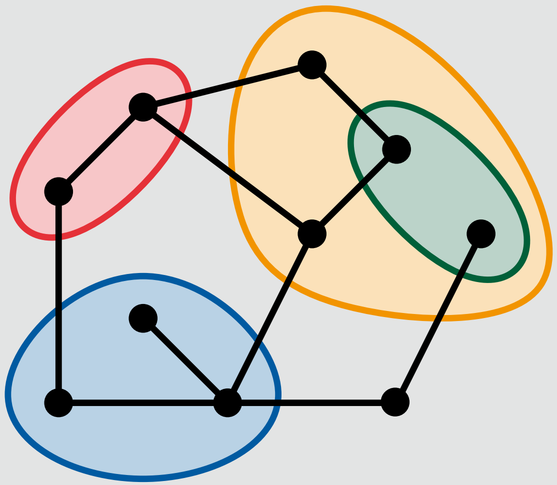


Given

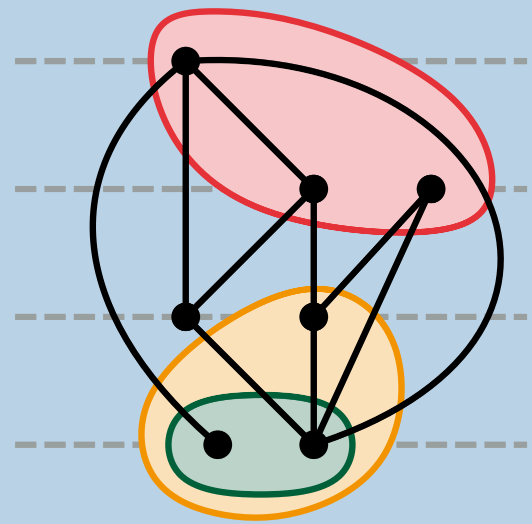
- Graph G
- Clustering T
- function $\gamma : V \to \{1, \ldots, k\}$

Clustered Planarity

[Lengauer ’89], [Feng et al. ’95]

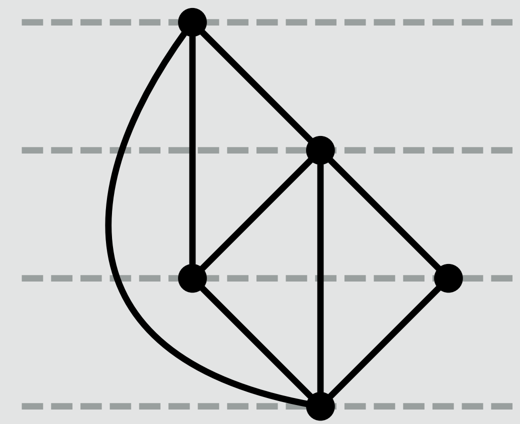


(unrestricted) Clustered Level Planarity



Level Planarity

[Di Battista & Nardelli ’88]



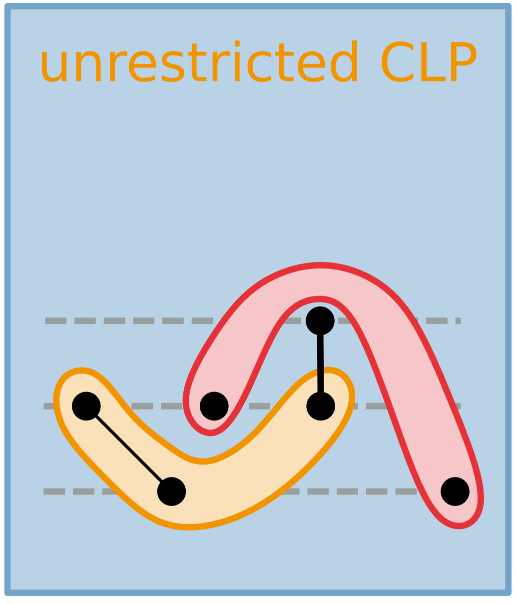
Given

- Graph G
- Clustering T
- function $\gamma : V \to \{1, \ldots, k\}$

Seek: drawing of G

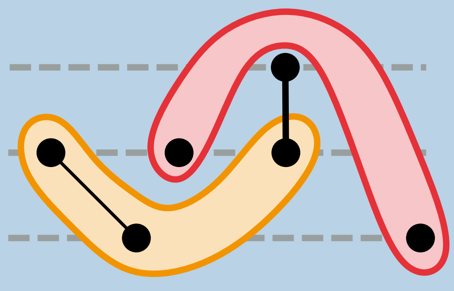
clustered planar   level planar

Problem Variants



Problem Variants

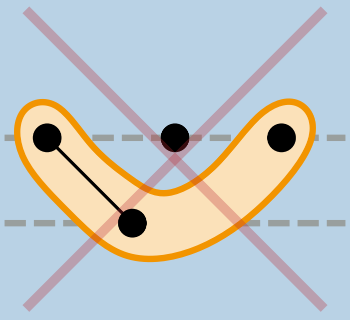
unrestricted CLP



Convex CLP

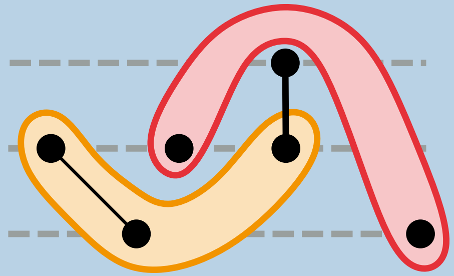
[Forster & Bachmaier ’04]

clusters convex regions



Problem Variants

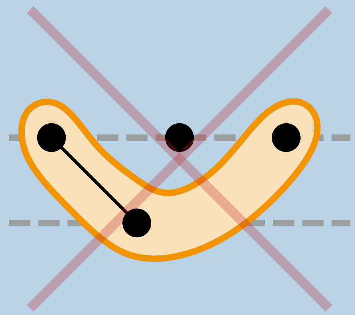
unrestricted CLP



Convex CLP

[Forster & Bachmaier '04]

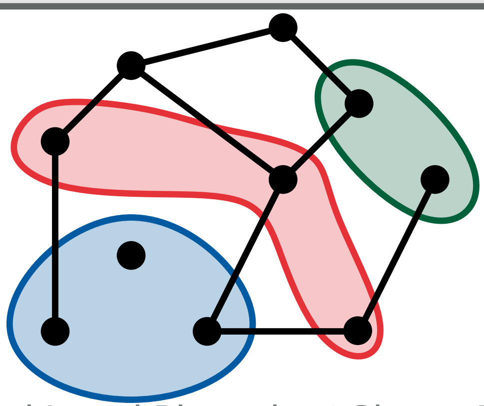
clusters convex regions



Theorem [Cornelsen & Wagner '06]

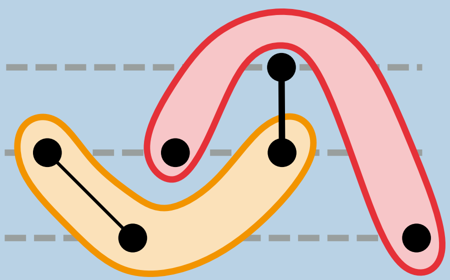
G is clustered planar iff it can be augmented with edges s.t.

- each cluster becomes connected
- *G* remains planar
- no cycle formed by a cluster encloses vertices of a different cluster



Problem Variants

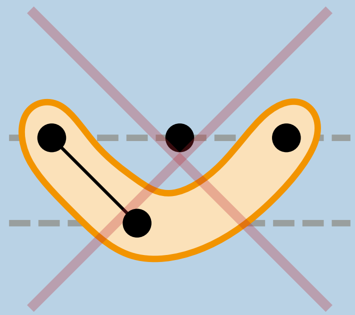
unrestricted CLP



Convex CLP

[Forster & Bachmaier ’04]

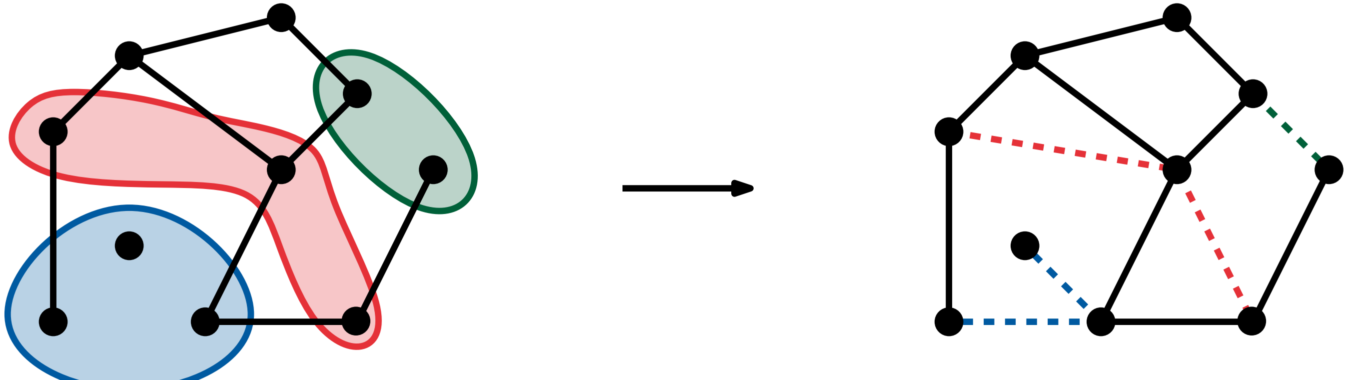
clusters convex regions



Theorem [Cornelsen & Wagner ’06]

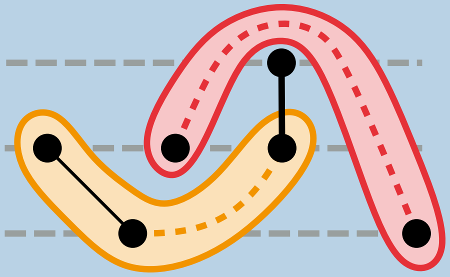
G is clustered planar iff it can be augmented with edges s.t.

- each cluster becomes connected
- G remains planar
- no cycle formed by a cluster encloses vertices of a different cluster

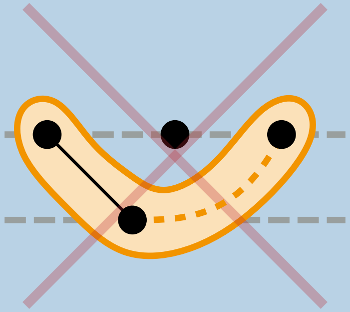


Problem Variants

unrestricted CLP

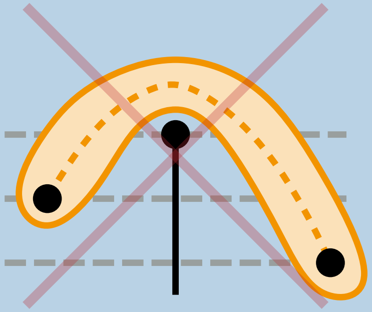


Convex CLP
[Forster & Bachmaier ’04]
clusters convex regions



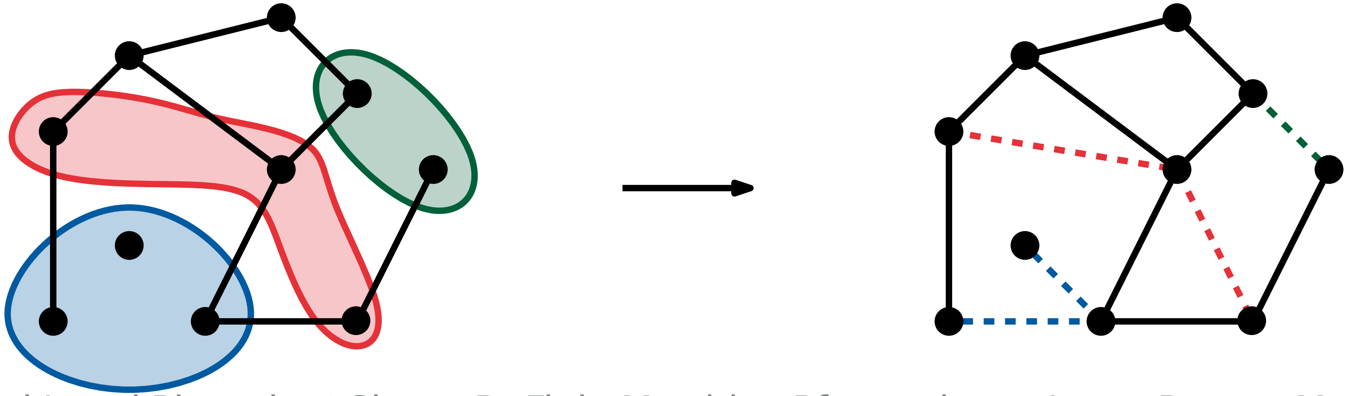
***y*-monotone CLP**

augmentation edges *y*-monotone



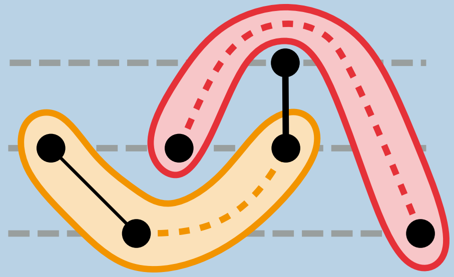
Theorem [Cornelsen & Wagner ’06]
G is clustered planar iff it can be augmented with edges s.t.

- each cluster becomes connected
- *G* remains planar
- no cycle formed by a cluster encloses vertices of a different cluster

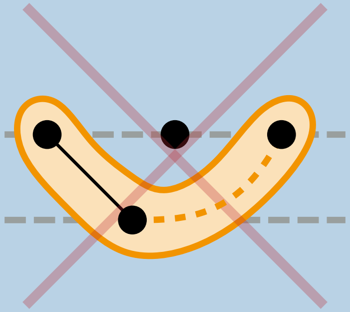


Problem Variants

unrestricted CLP

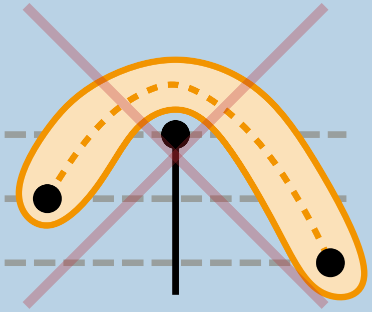


Convex CLP
[Forster & Bachmaier '04]
clusters convex regions



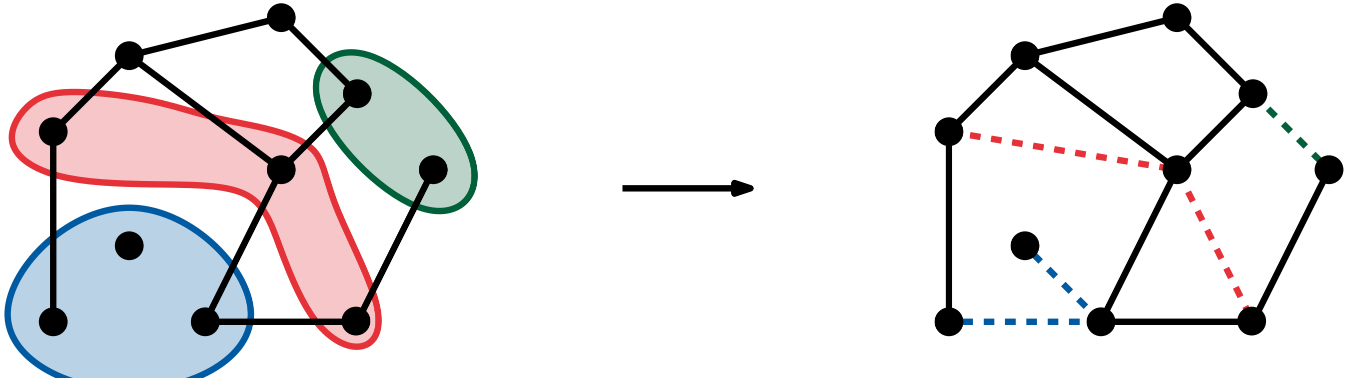
y-monotone CLP

augmentation edges *y*-monotone



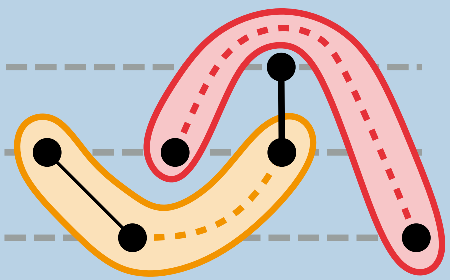
Definition
G is *y*-monotone cluster level planar iff it can be augmented with edges s.t.

- each cluster becomes connected
- *G* remains level planar
- no cycle formed by a cluster encloses vertices of a different cluster



Problem Variants

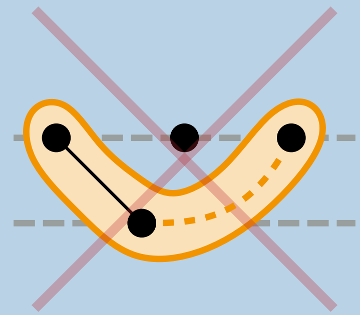
unrestricted CLP



Convex CLP

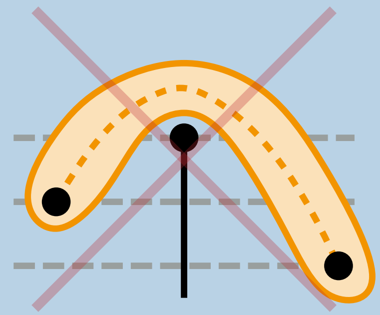
[Forster & Bachmaier ’04]

clusters convex regions



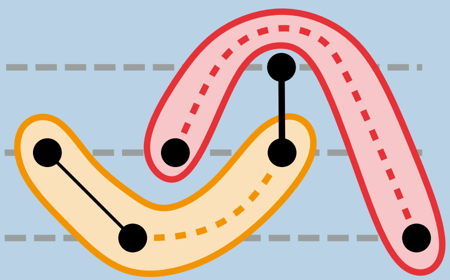
y-monotone CLP

augmentation edges *y*-monotone



Problem Variants

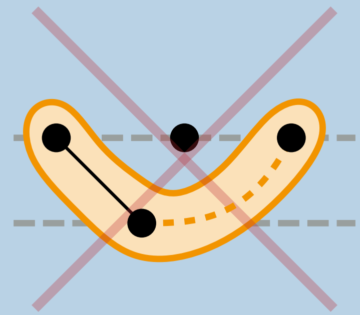
unrestricted CLP



Convex CLP

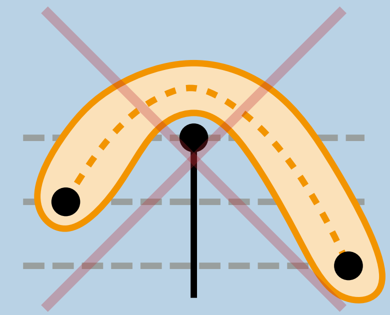
[Forster & Bachmaier ’04]

clusters convex regions



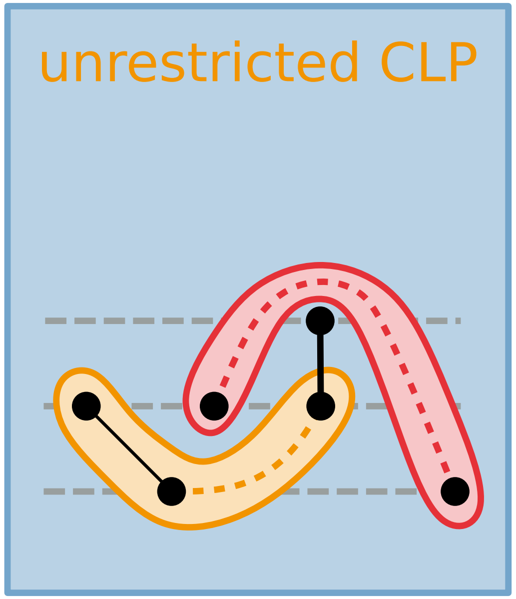
y-monotone CLP

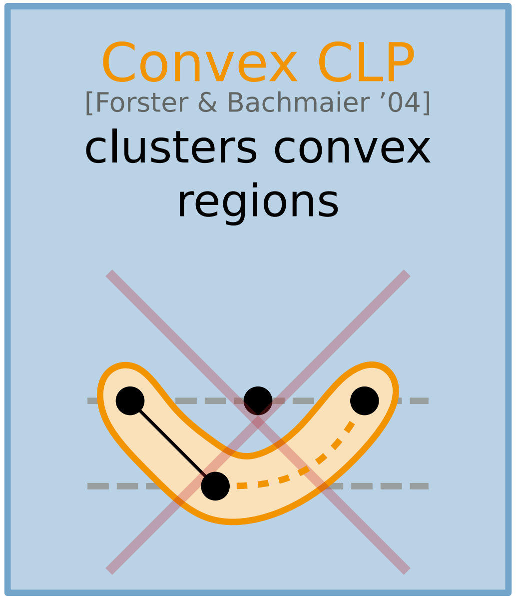
augmentation edges y-monotone

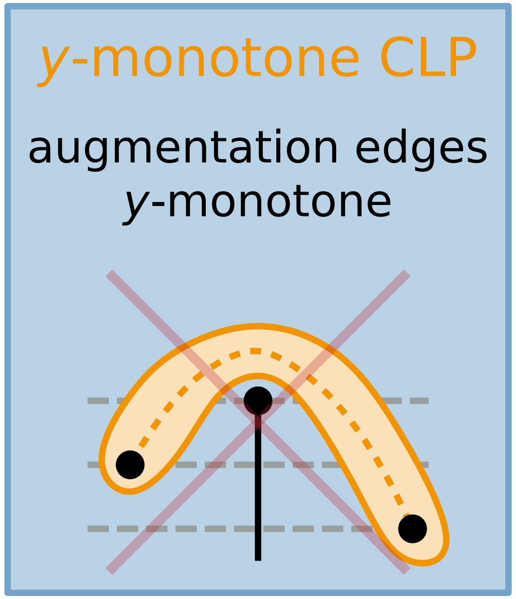


- NP-complete
  [Angelini et al. ’14]
- $\mathcal{O}(n)$-algorithm
  (proper, single-source, level-connected)
  [Forster & Bachmaier ’04]
- $\mathcal{O}(n^4)$-algorithm
  (proper)
  [Angelini et al. ’14]

Problem Variants





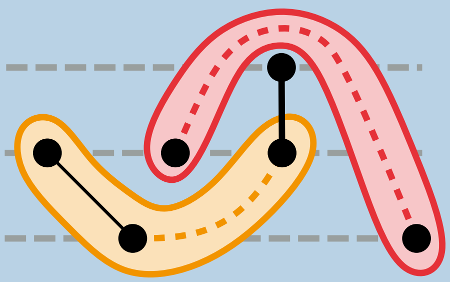


- $O(n^3)$-algorithm
  (single-source, biconnected)
  [this work]

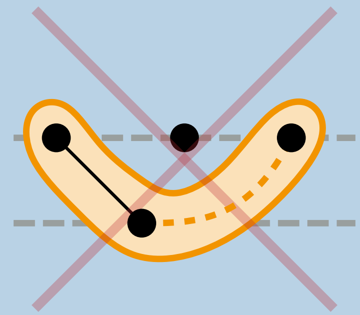
- NP-complete
  [Angelini et al. '14]
- $\mathcal{O}(n)$-algorithm
  (proper, single-source, level-connected)
  [Forster & Bachmaier '04]
- $\mathcal{O}(n^4)$-algorithm
  (proper)
  [Angelini et al. '14]

Problem Variants

unrestricted CLP

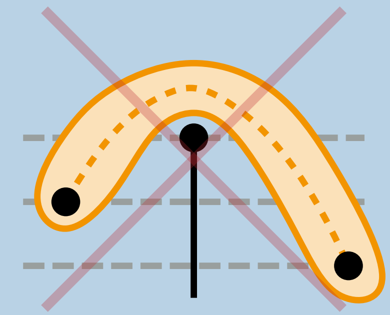


Convex CLP
[Forster & Bachmaier '04]
clusters convex regions



y-monotone CLP

augmentation edges y-monotone



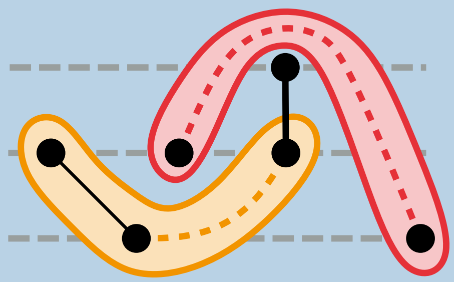
- $O(n^3)$-algorithm
  (single-source, biconnected)
  [this work]

- NP-complete
  [Angelini et al. '14]
- $\mathcal{O}(n)$-algorithm
  (proper, single-source, level-connected)
  [Forster & Bachmaier '04]
- $\mathcal{O}(n^4)$-algorithm
  (proper)
  [Angelini et al. '14]

- NP-complete
  (single-source, biconnected)
  [this work]
- NP-complete
  (constant #levels + #clusters)
  [this work]

Problem Variants

unrestricted CLP

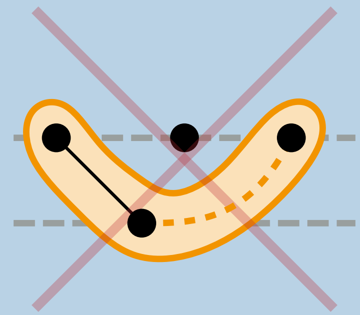


- $O(n^3)$-algorithm
  (single-source, biconnected)
  [this work]

Convex CLP

[Forster & Bachmaier ’04]

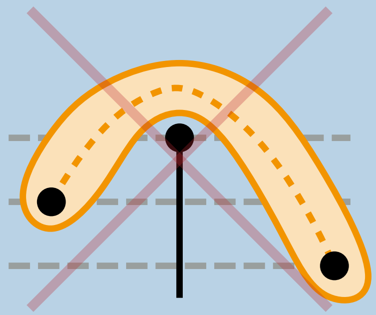
clusters convex regions



- NP-complete
  [Angelini et al. ’14]
- $\mathcal{O}(n)$-algorithm
  (proper, single-source, level-connected)
  [Forster & Bachmaier ’04]
- $\mathcal{O}(n^4)$-algorithm
  (proper)
  [Angelini et al. ’14]

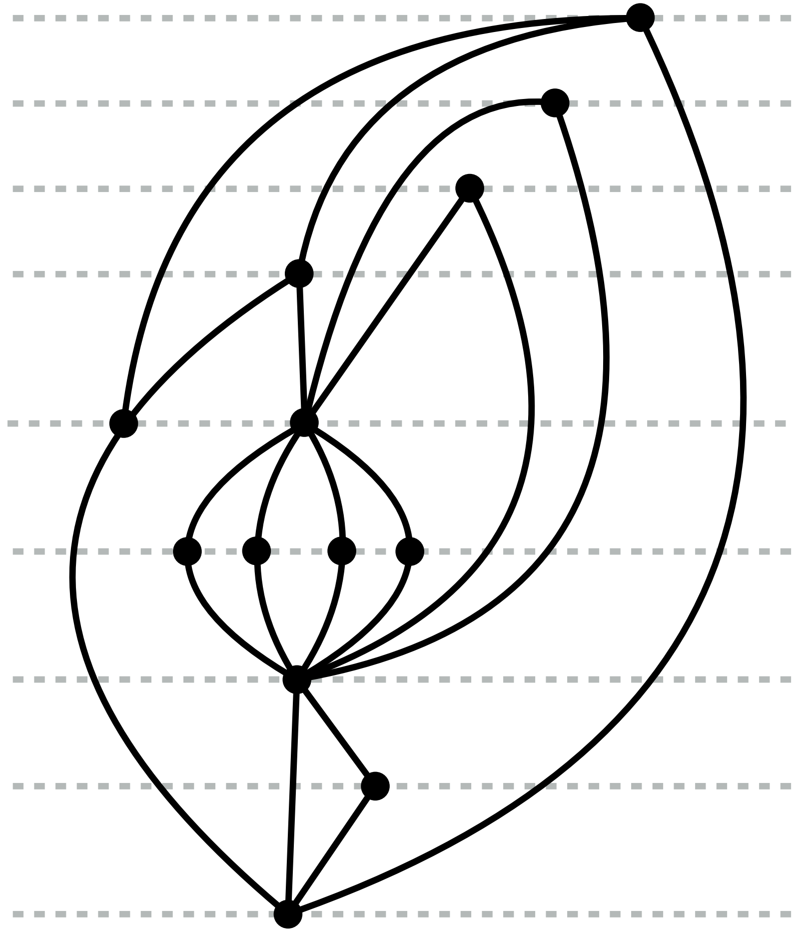
y-monotone CLP

augmentation edges *y*-monotone

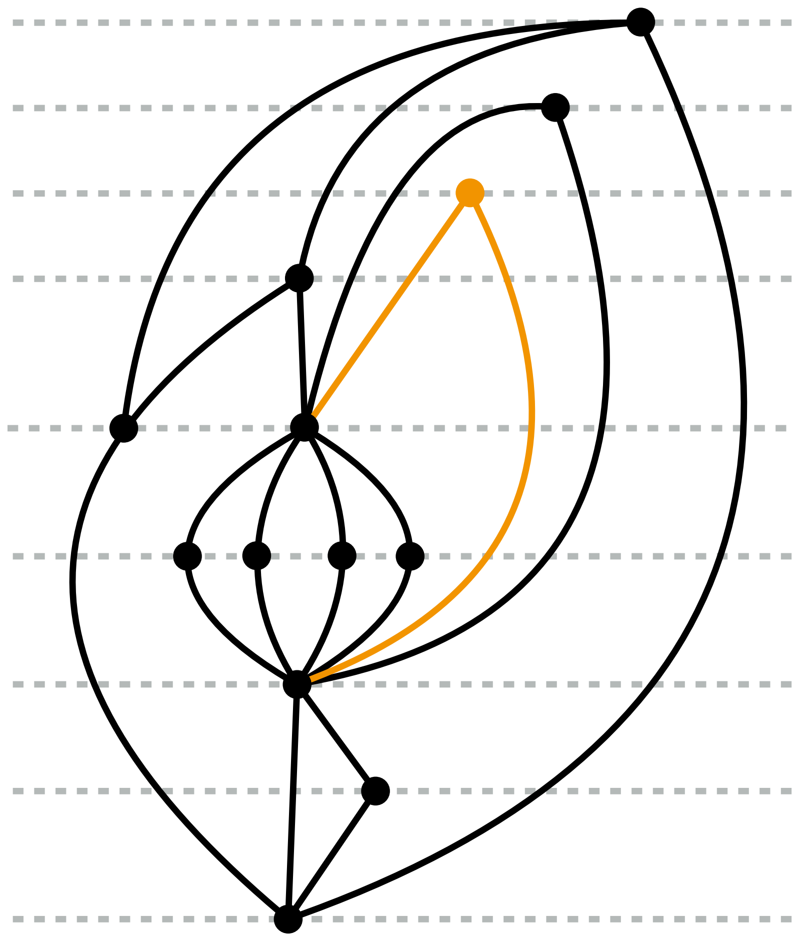


- NP-complete
  (single-source, biconnected)
  [this work]
- NP-complete
  (constant #levels + #clusters)
  [this work]

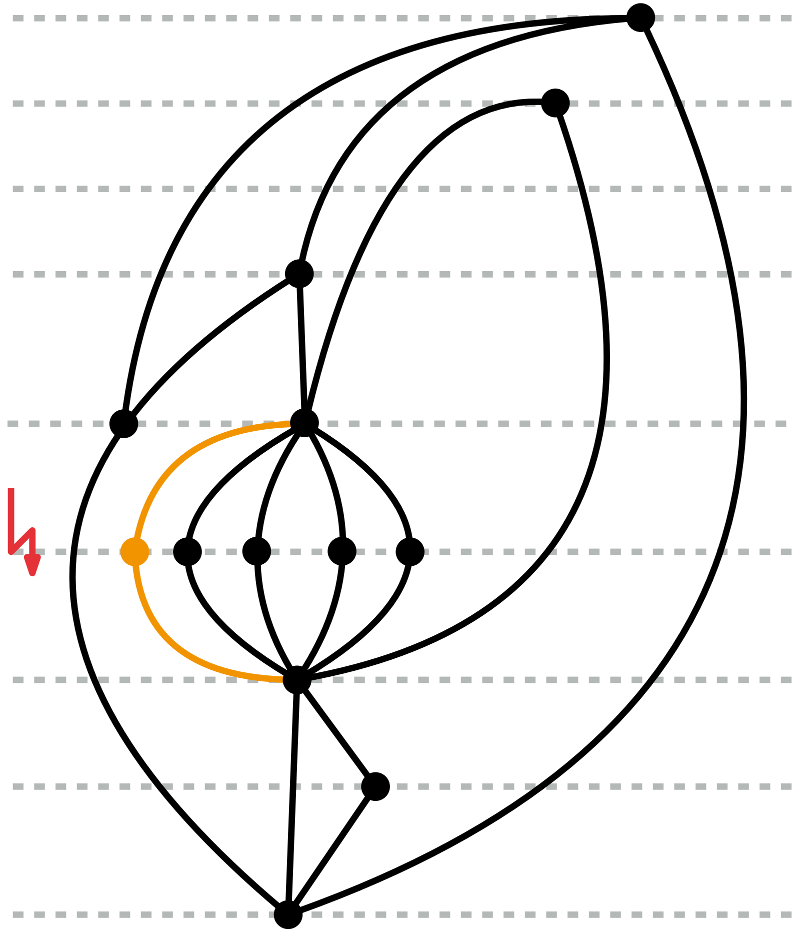
Embeddings of Biconnected Single-Source Graphs



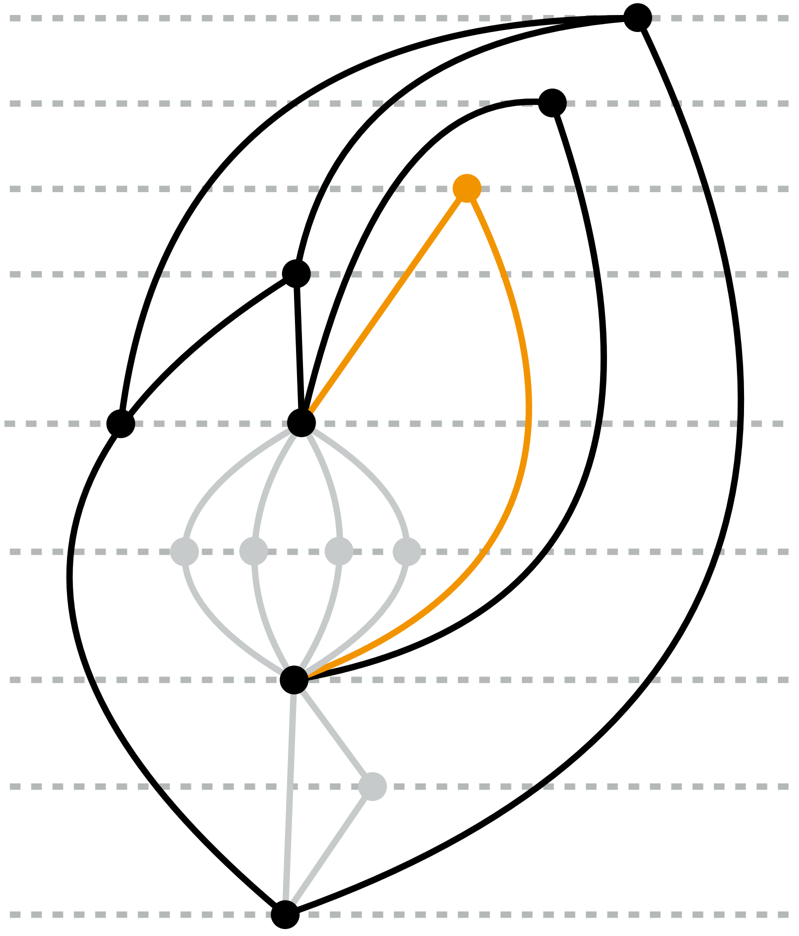
Embeddings of Biconnected Single-Source Graphs



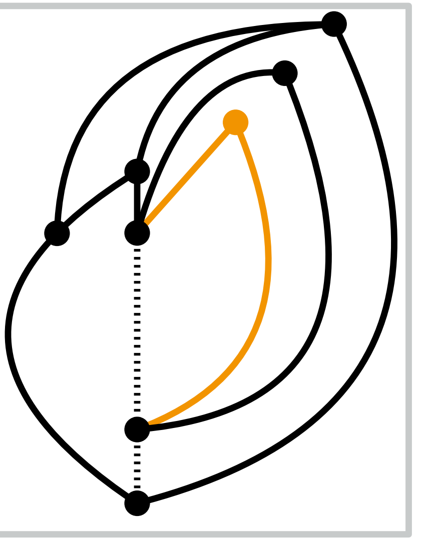
Embeddings of Biconnected Single-Source Graphs



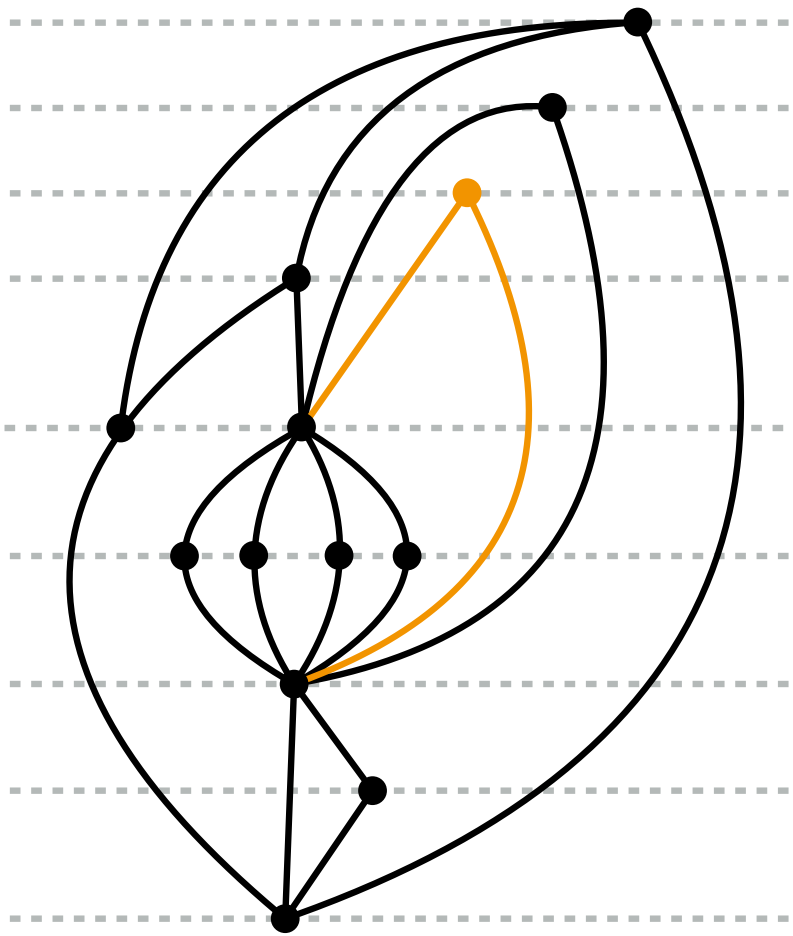
Embeddings of Biconnected Single-Source Graphs

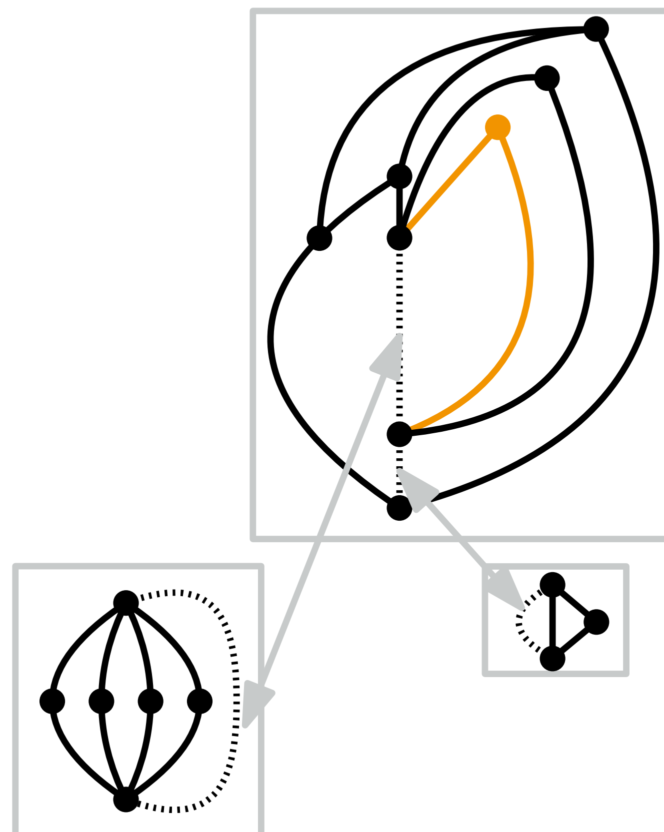


rigid subgraph

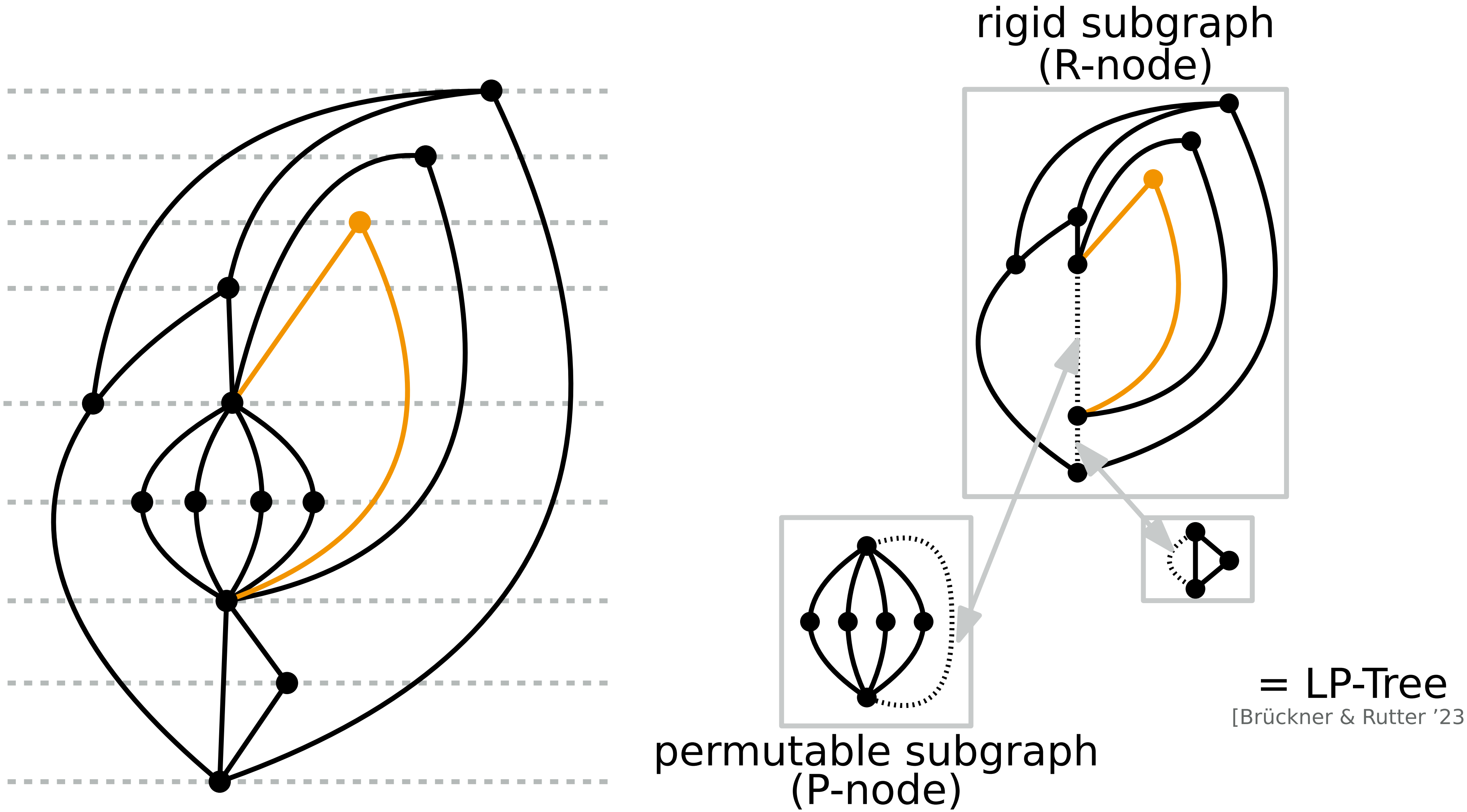


Embeddings of Biconnected Single-Source Graphs

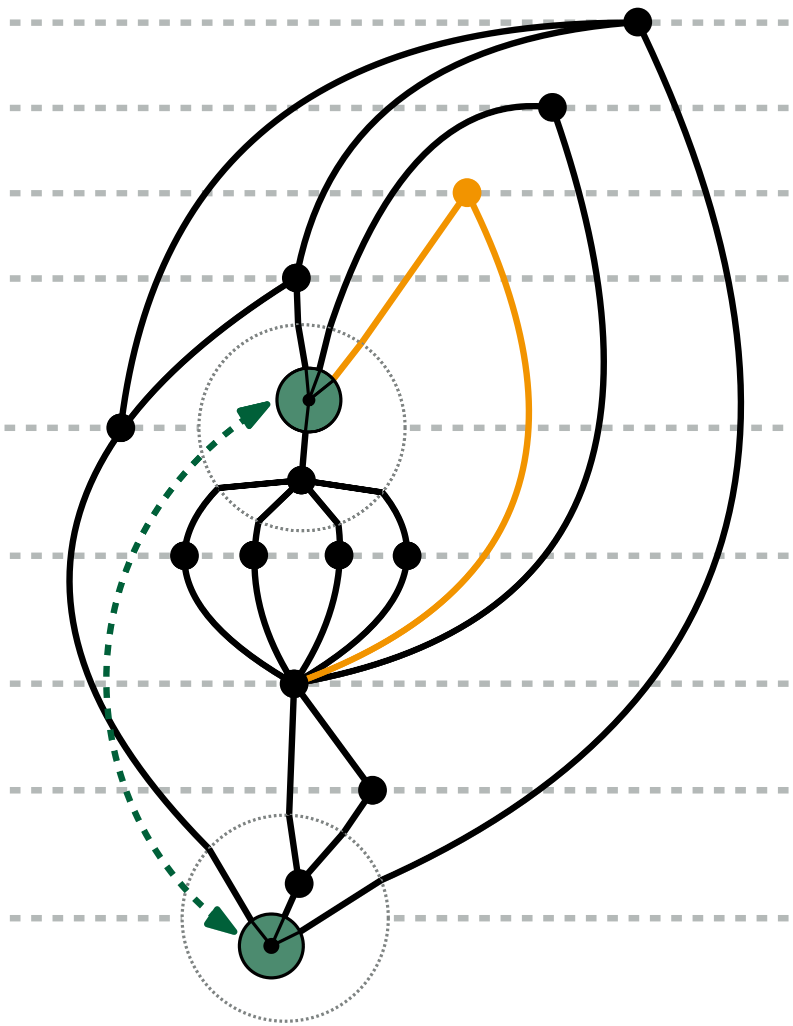




Embeddings of Biconnected Single-Source Graphs

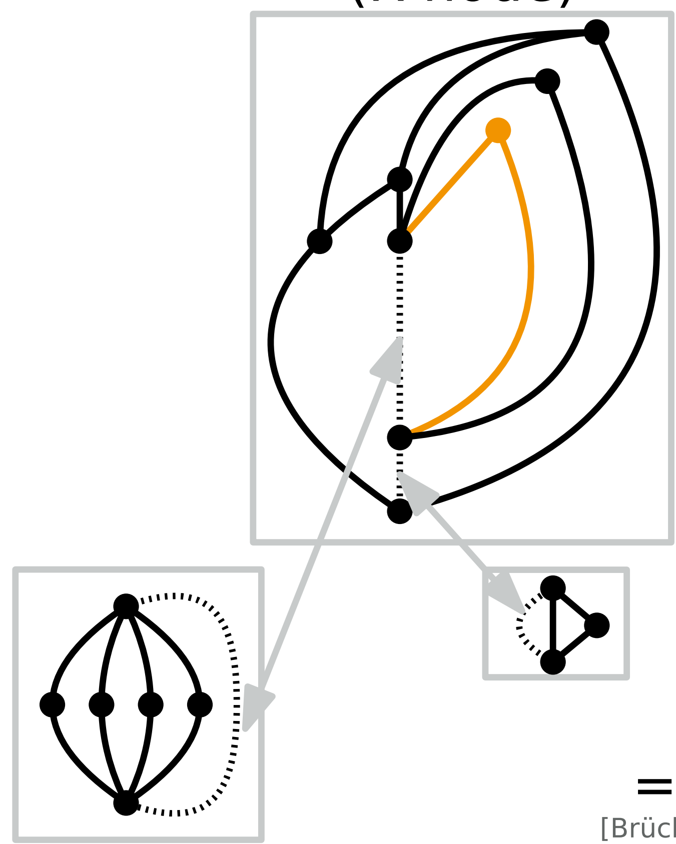


Embeddings of Biconnected Single-Source Graphs



Encode level planarity constraints
using synchronized wheels

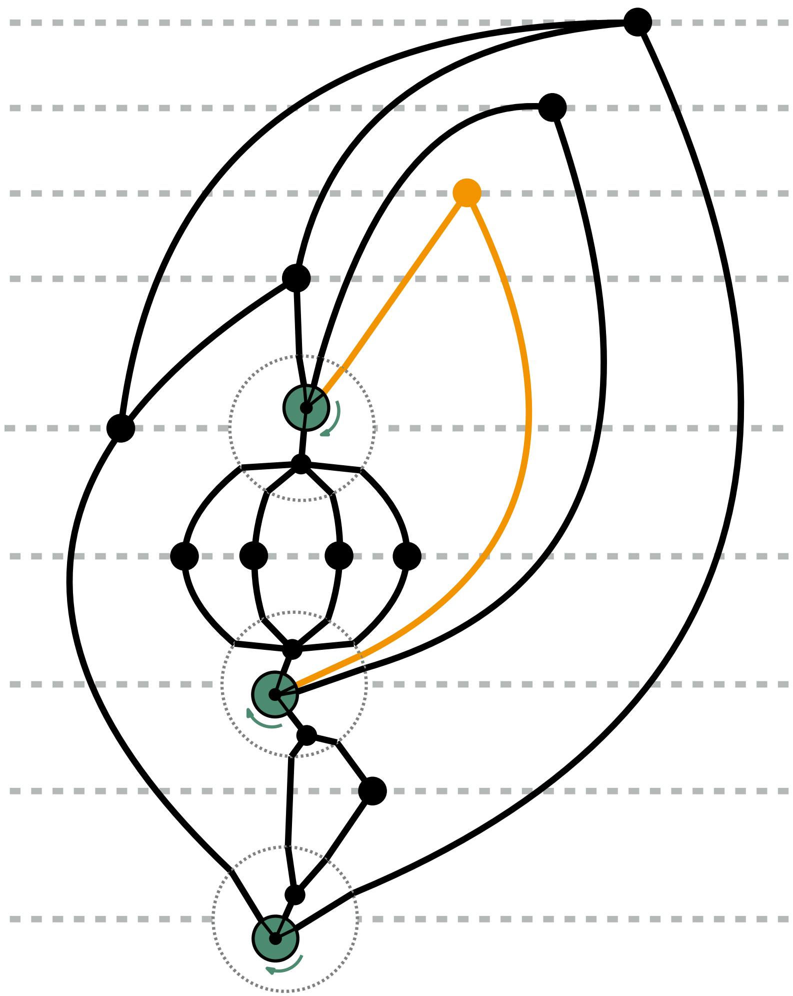
rigid subgraph
(R-node)



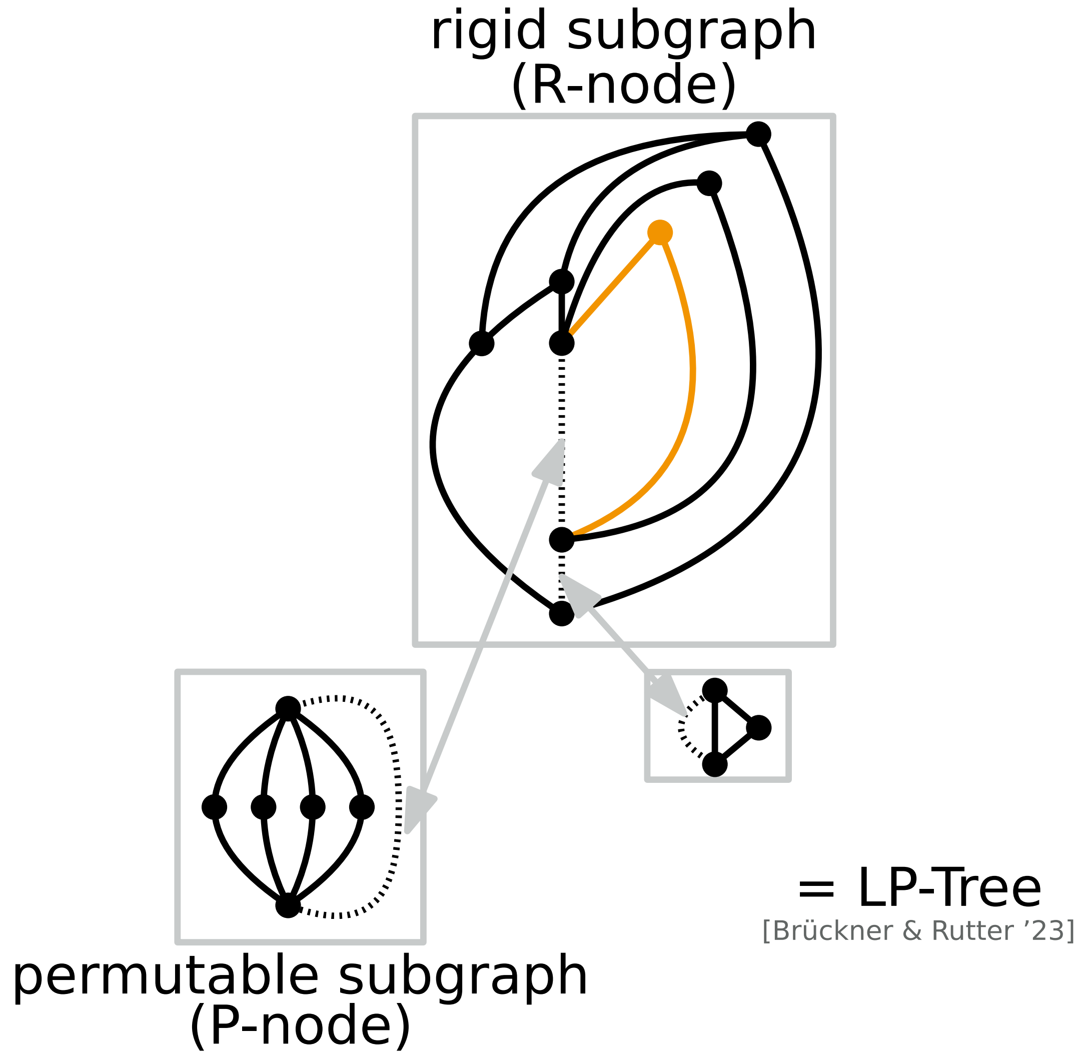
permutable subgraph
(P-node)

= LP-Tree
[Brückner & Rutter ’23]

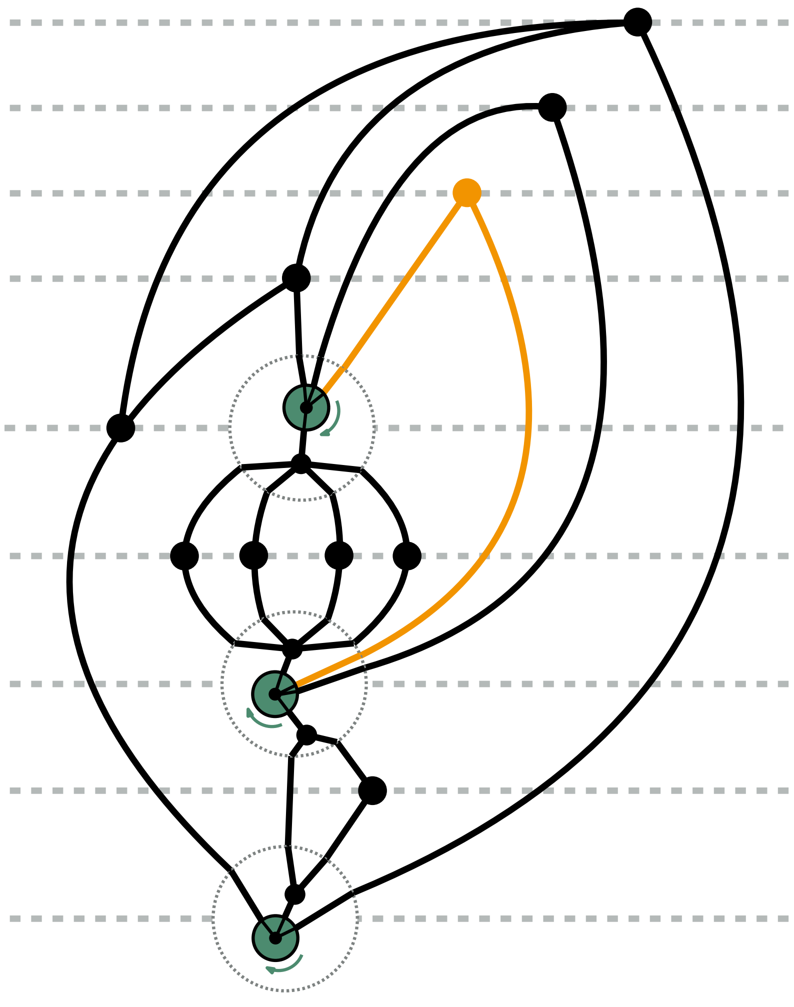
Embeddings of Biconnected Single-Source Graphs



Encode level planarity constraints
using synchronized wheels

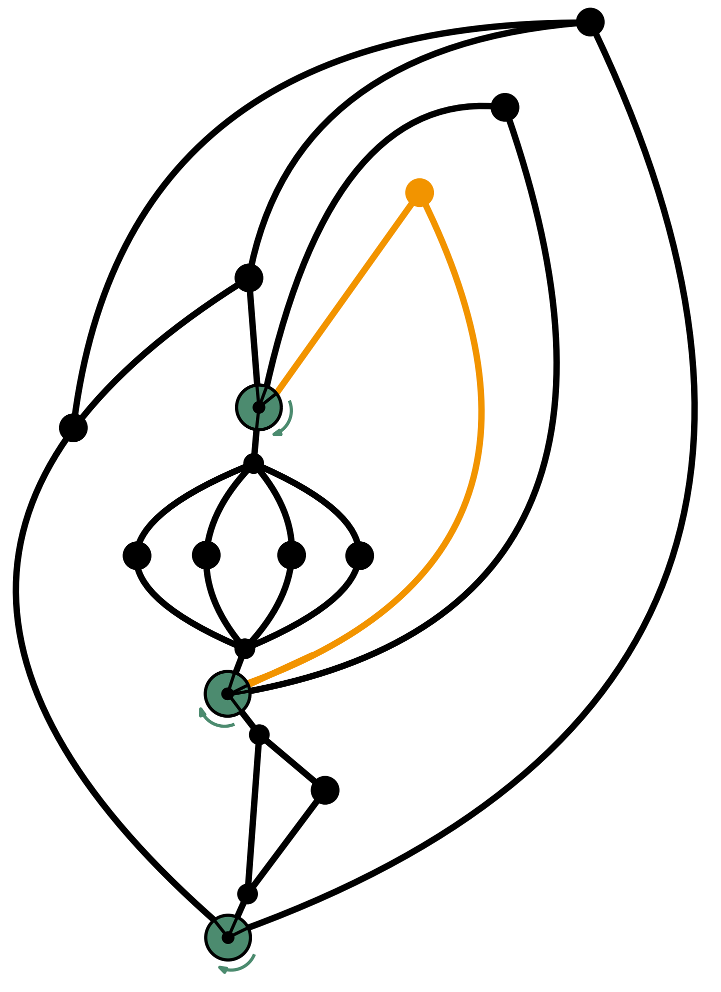


Embeddings of Biconnected Single-Source Graphs



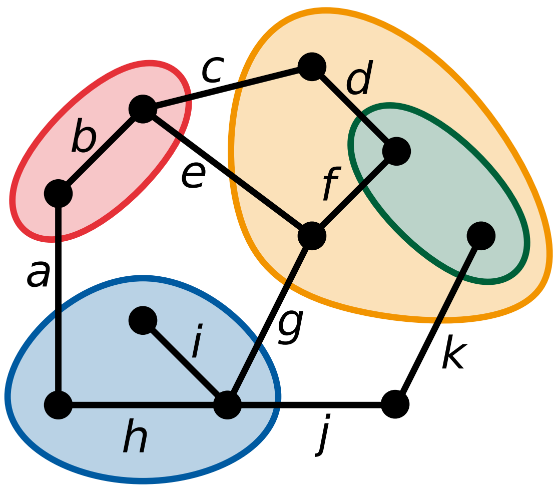
Encode level planarity constraints
using synchronized wheels

$\equiv$

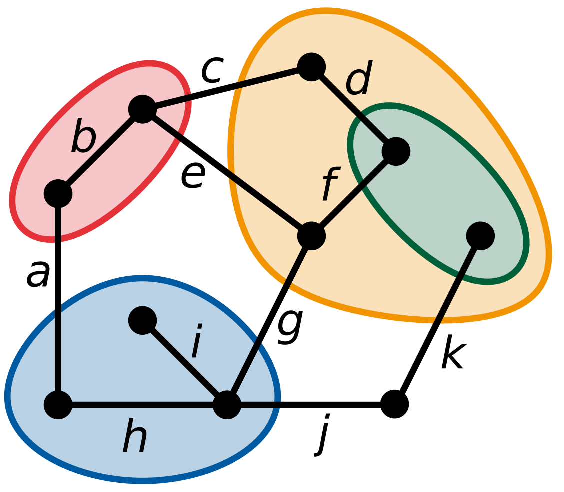


Graph with synchronized
fixed-vertex constraints

Embeddings of Clustered Graphs



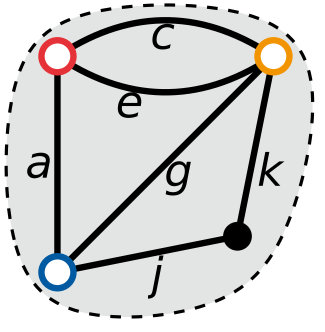
Embeddings of Clustered Graphs



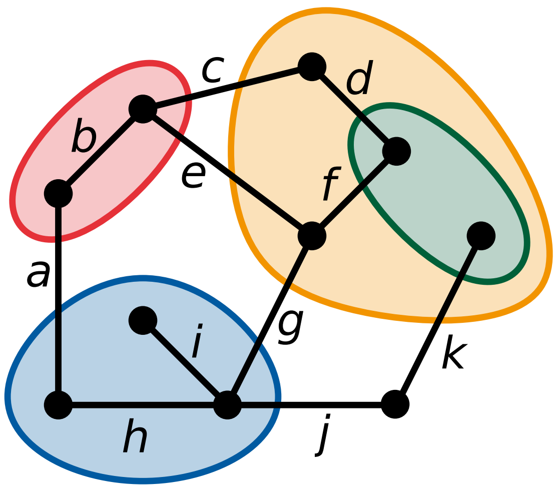
[Bläsius, Rutter ’16]

CD-Tree Representation:

for each cluster, contract neighboring clusters into single vertices



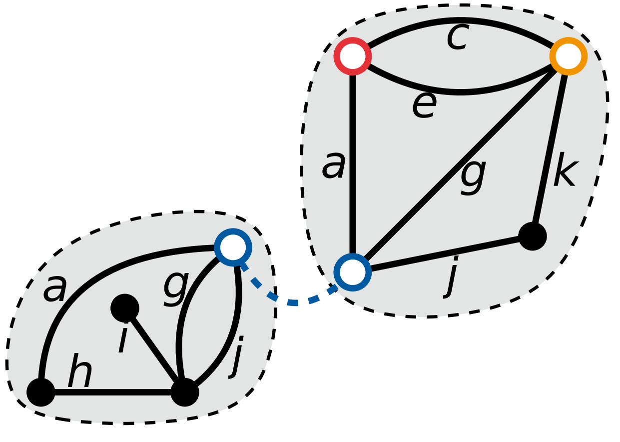
Embeddings of Clustered Graphs



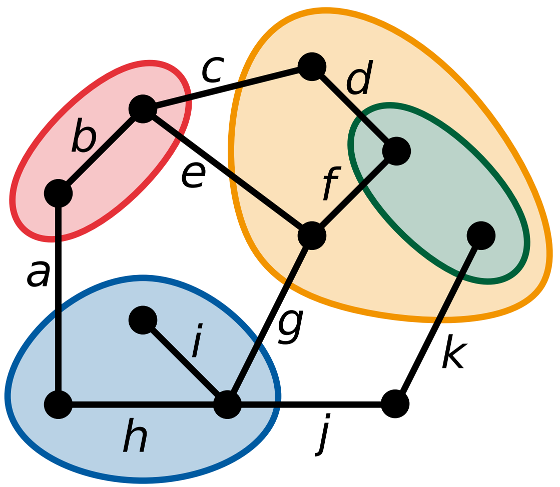
[Bläsius, Rutter '16]

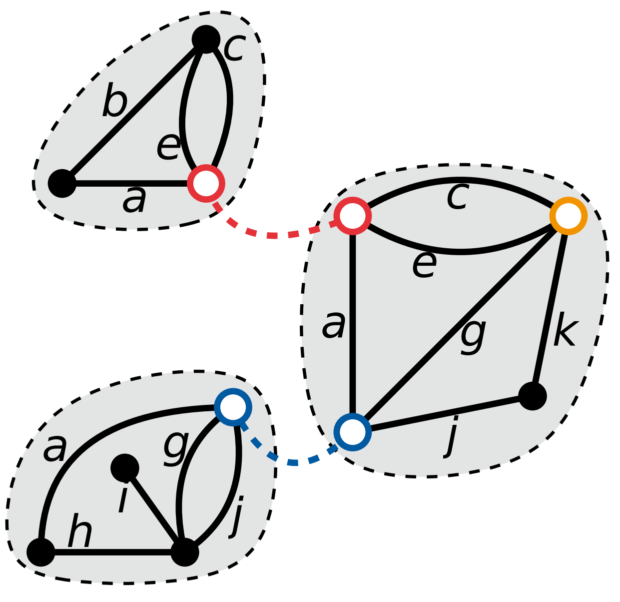
CD-Tree Representation:

for each cluster, contract neighboring clusters into single vertices



Embeddings of Clustered Graphs



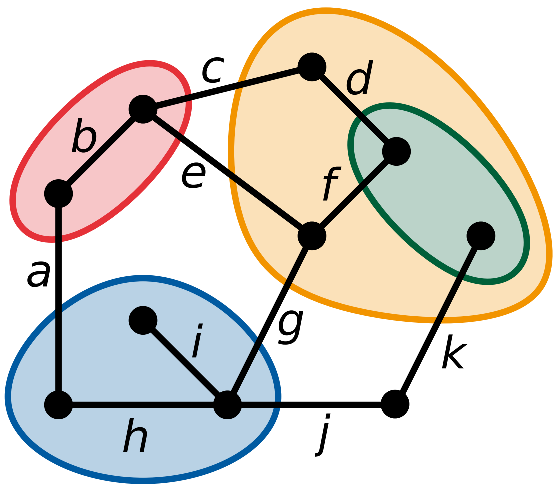


[Bläsius, Rutter ’16]

CD-Tree Representation:

for each cluster, contract neighboring clusters into single vertices

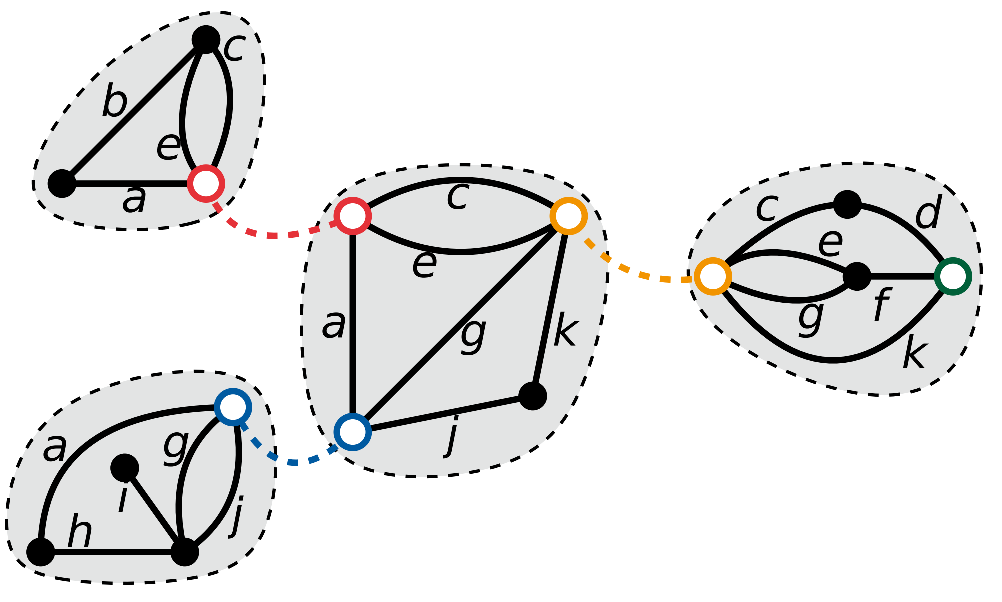
Embeddings of Clustered Graphs



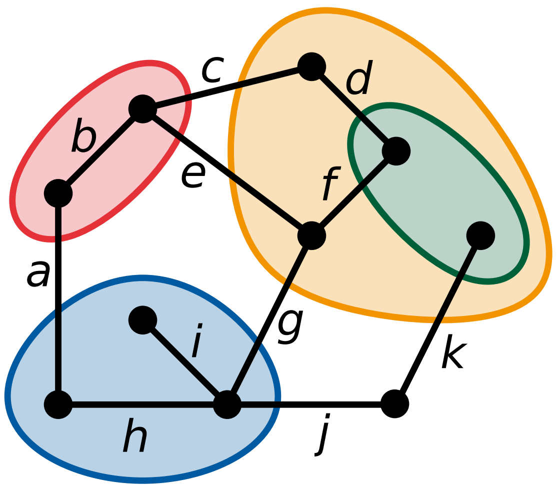
[Bläsius, Rutter '16]

CD-Tree Representation:

for each cluster, contract neighboring clusters into single vertices



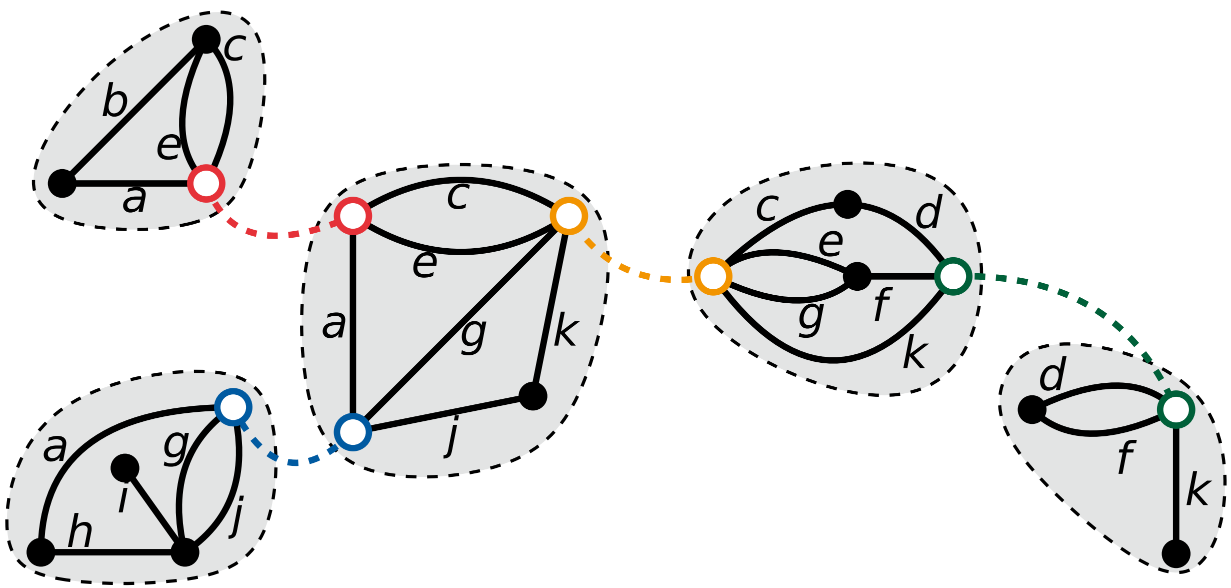
Embeddings of Clustered Graphs



[Bläsius, Rutter '16]

CD-Tree Representation:

for each cluster, contract neighboring clusters into single vertices

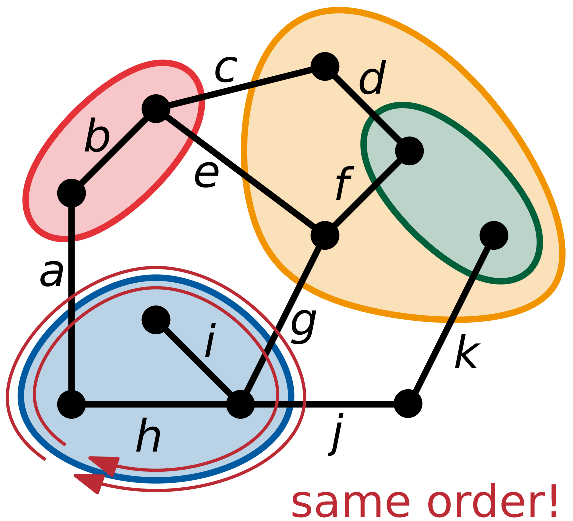


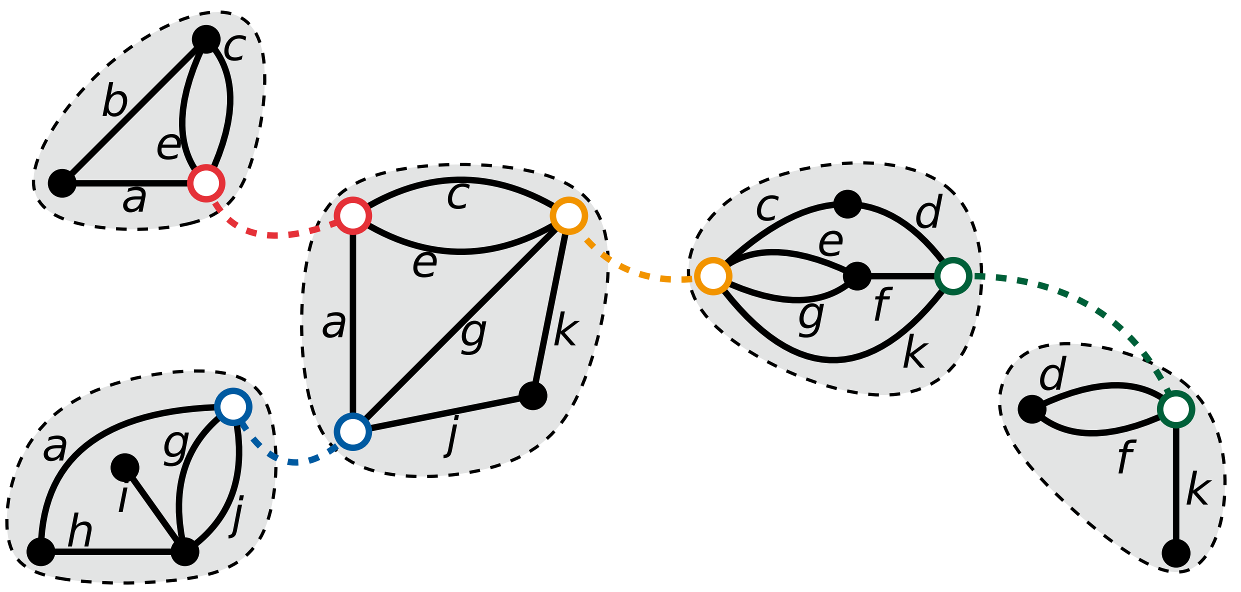
Embeddings of Clustered Graphs

[Bläsius, Rutter '16]

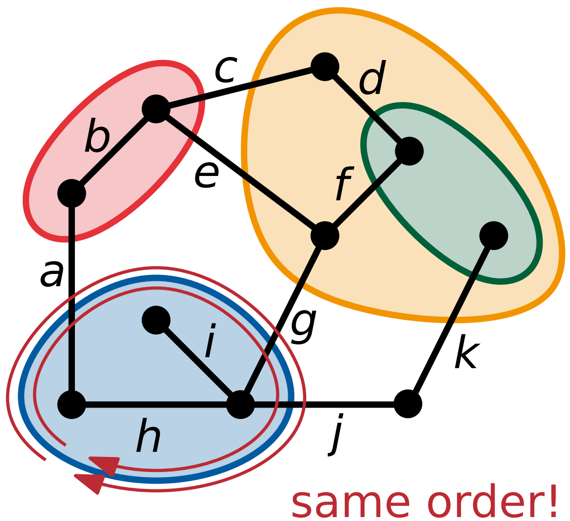
CD-Tree Representation:

for each cluster, contract neighboring clusters into single vertices



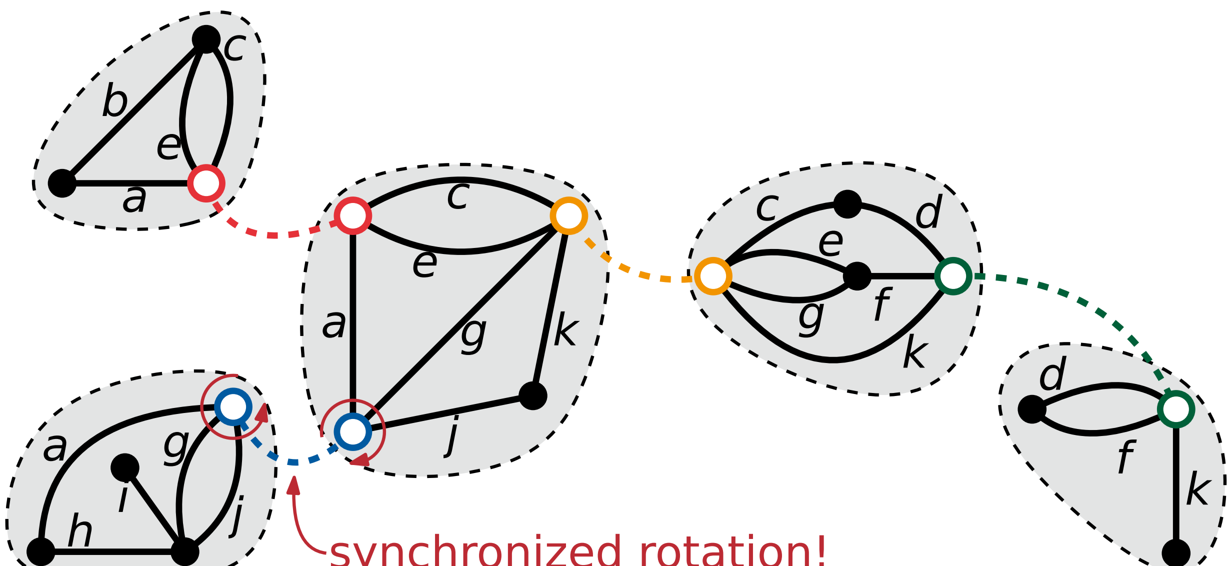


Embeddings of Clustered Graphs



[Bläsius, Rutter '16]

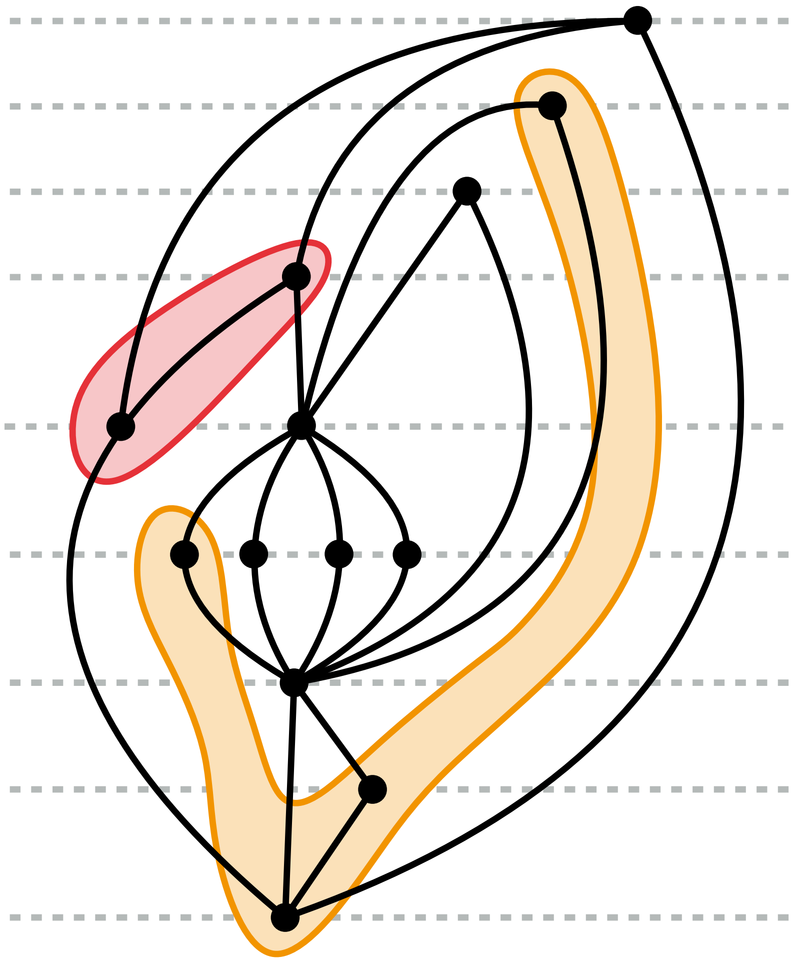
CD-Tree Representation:
for each cluster, contract neighboring clusters into single vertices



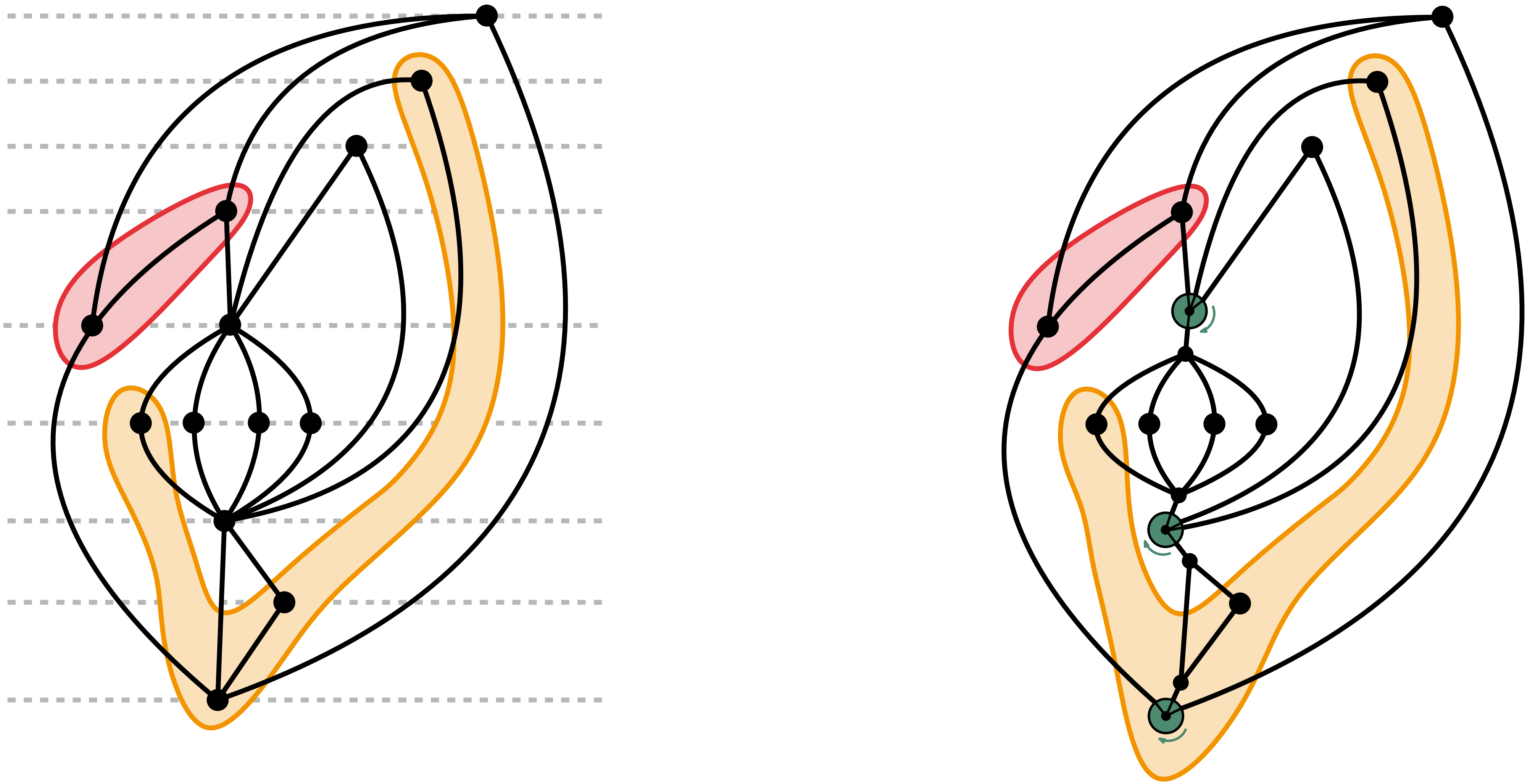
[Bläsius, Rutter '16]

planar embedding where the rotations line up ⇔ Cluster Planar ✓

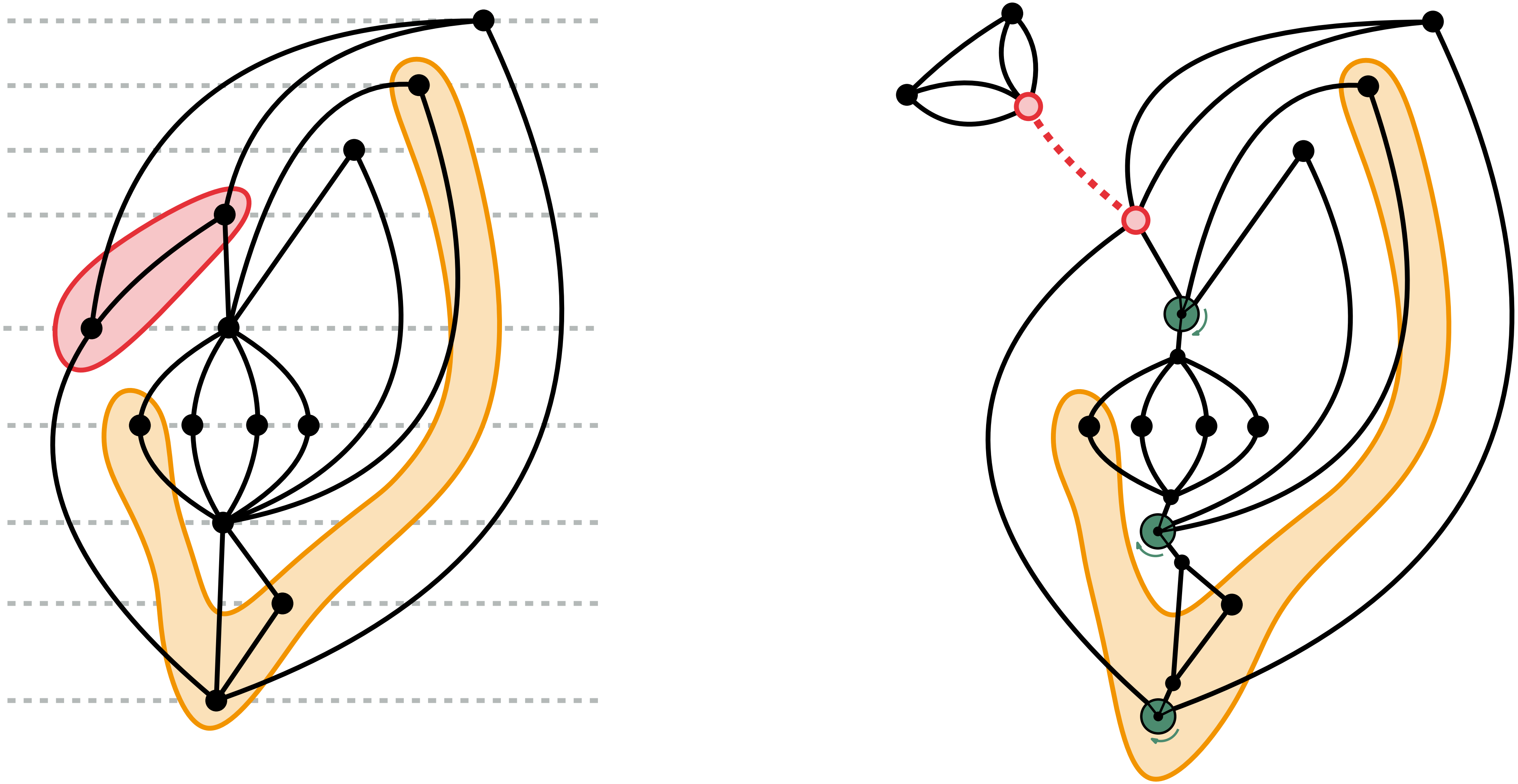
Embeddings of Clustered Level Graphs



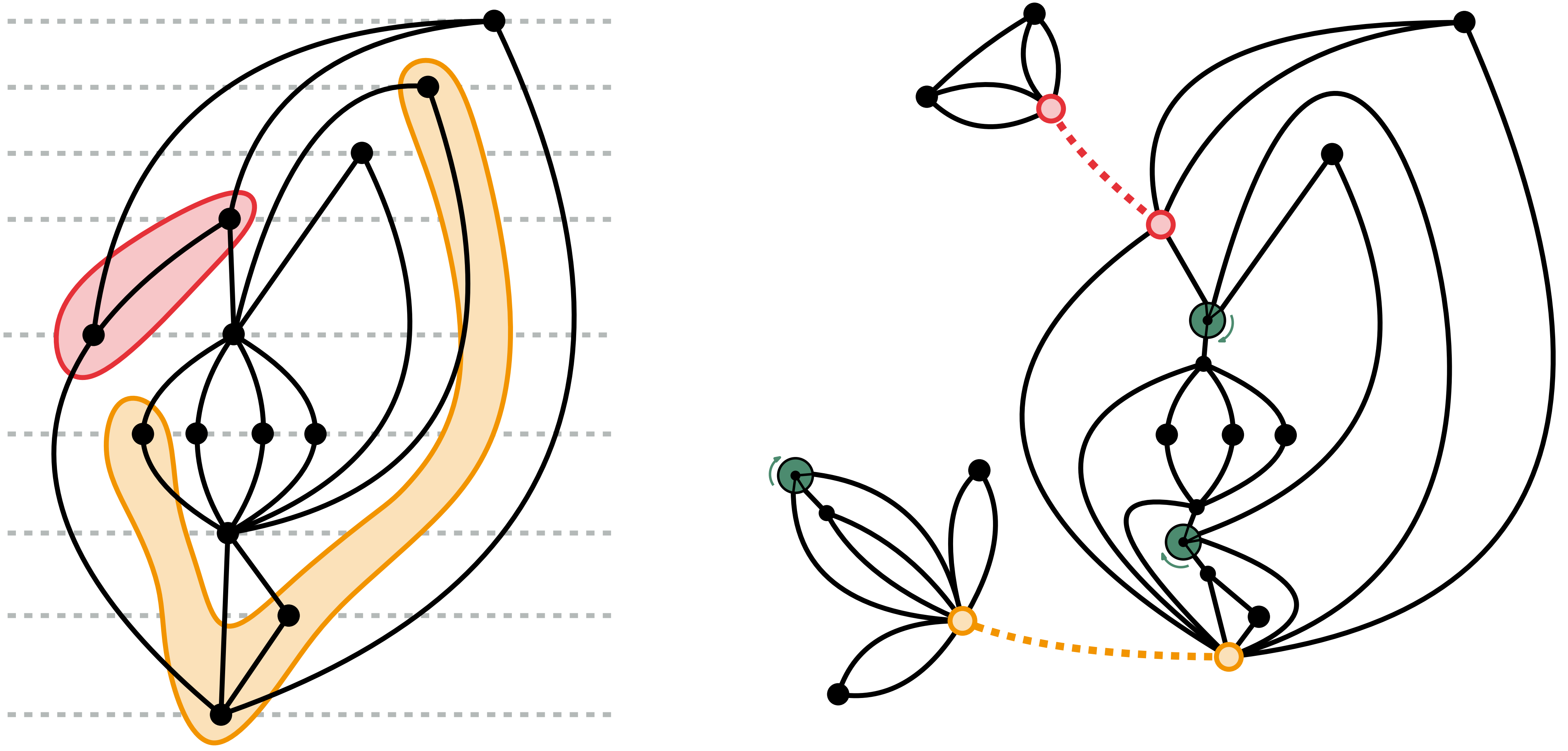
Embeddings of Clustered Level Graphs



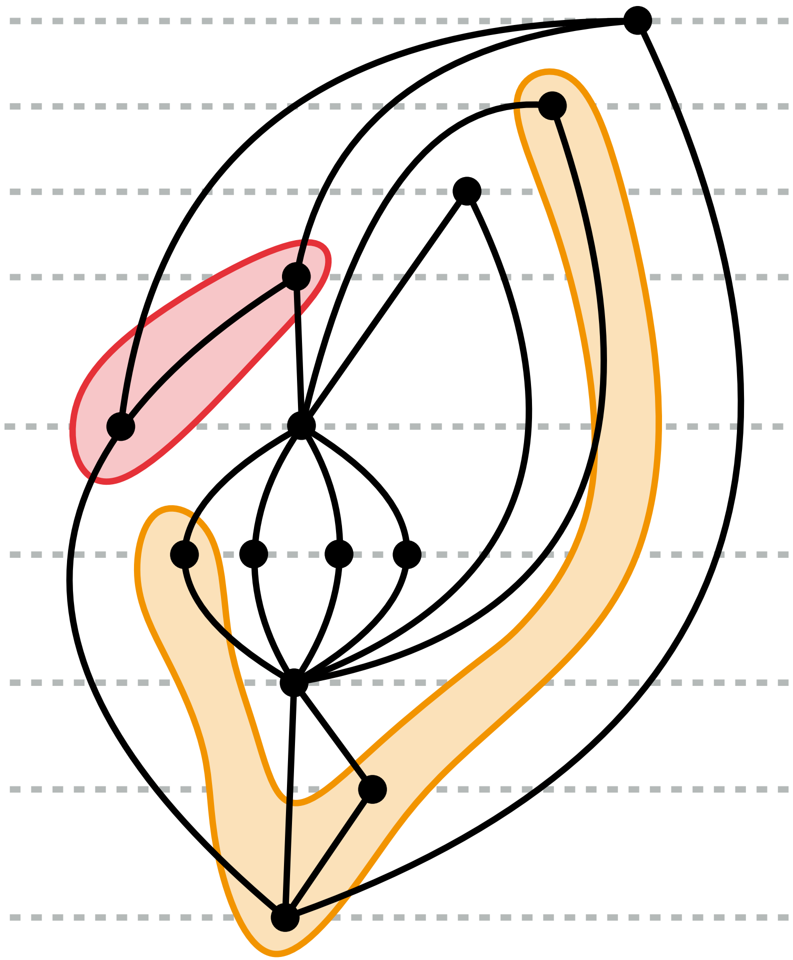
Embeddings of Clustered Level Graphs



Embeddings of Clustered Level Graphs

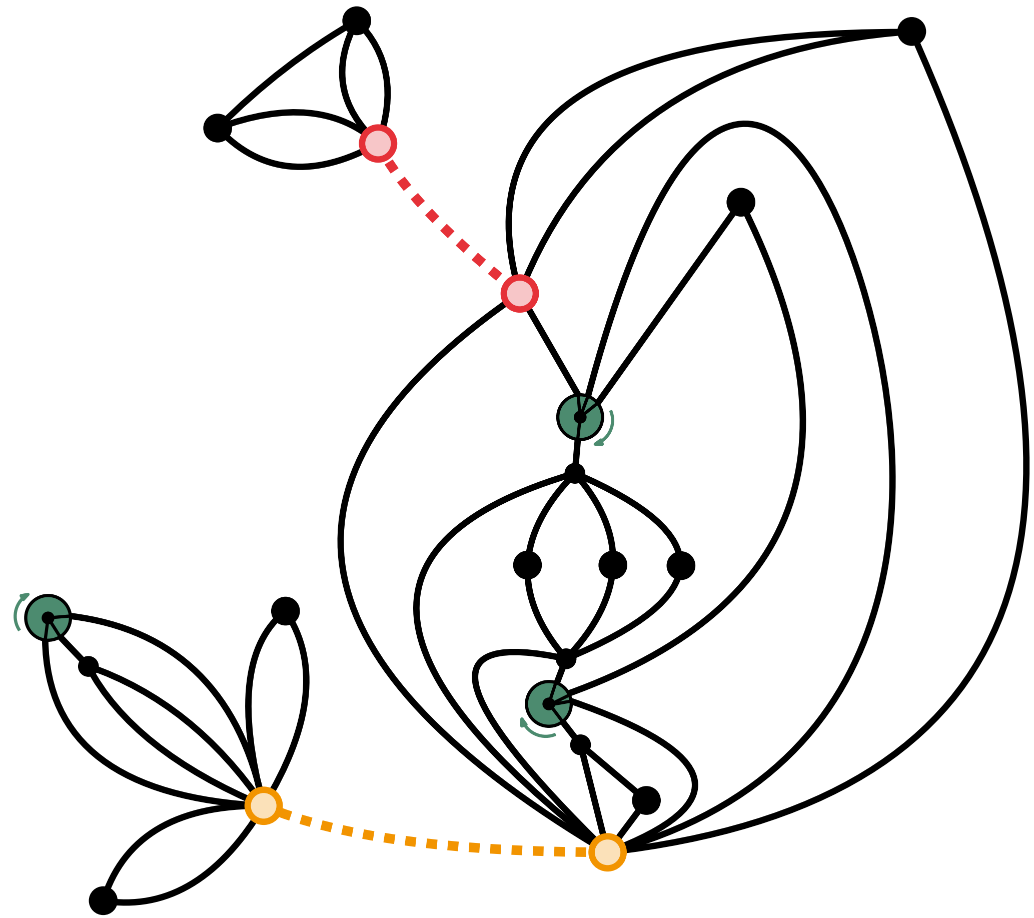


Embeddings of Clustered Level Graphs



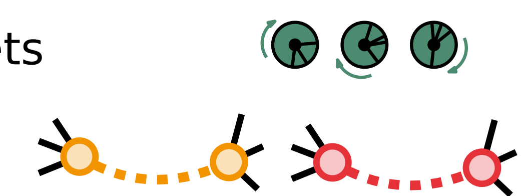
Clustered Level Graph

≡

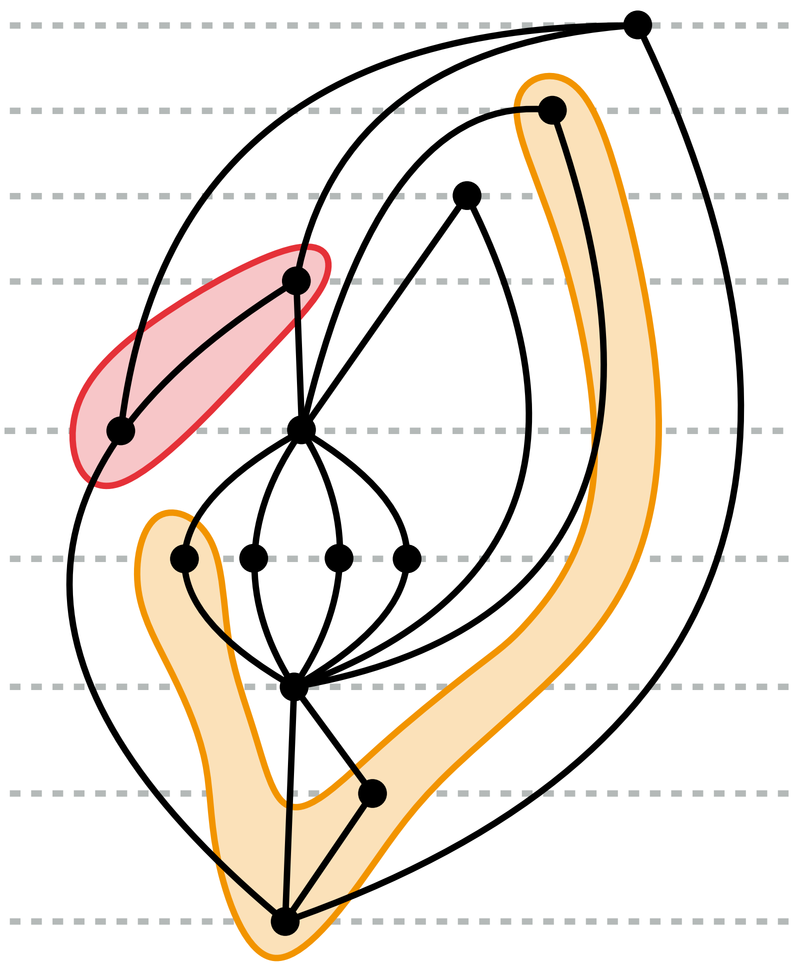


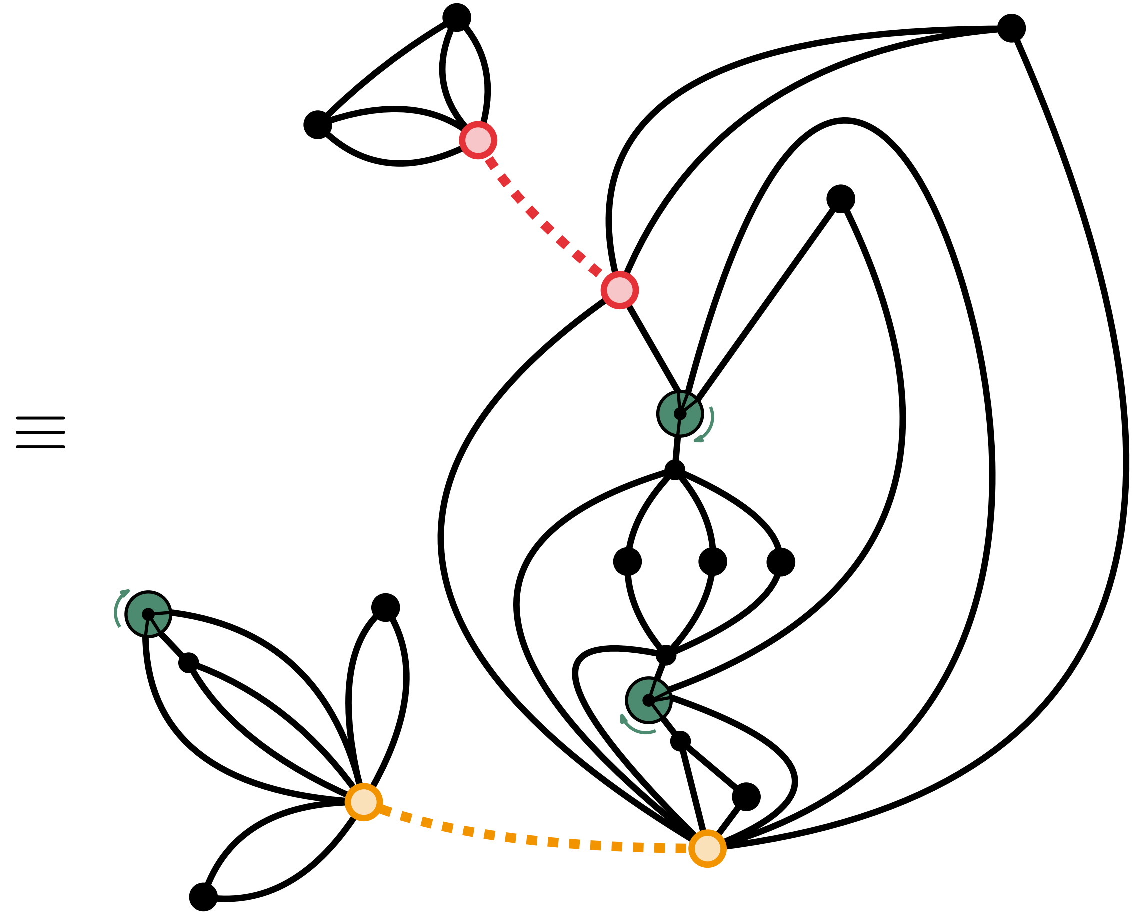
Graph with synchronized

- fixed-vertex sets
- vertex pairs



Embeddings of Clustered Level Graphs





Corresponding problem is called

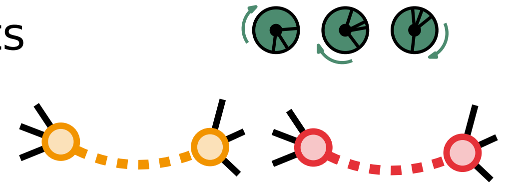
Synchronized Planarity

[Bläsius et al. '21]

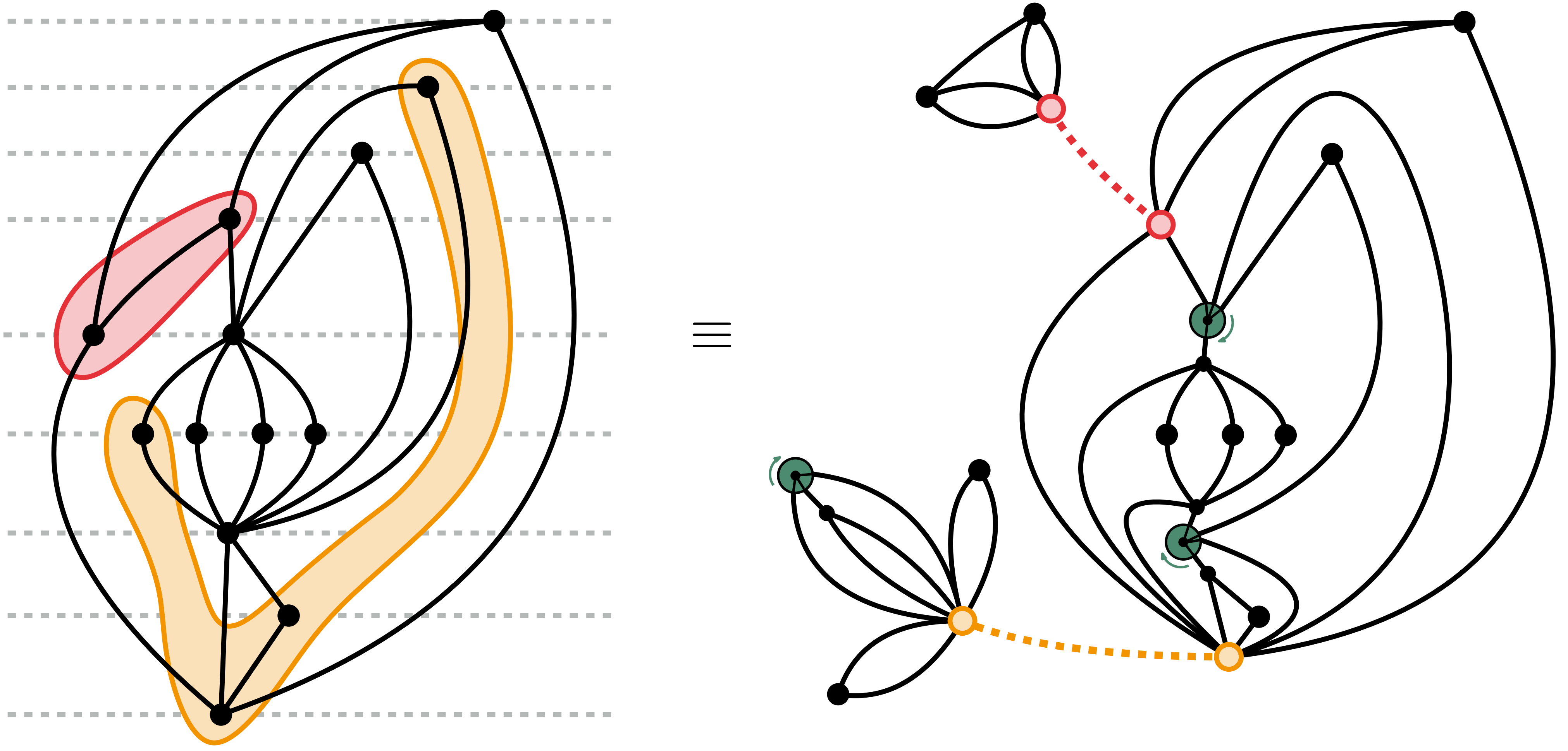
quadratic solution!

Graph with synchronized

- fixed-vertex sets
- vertex pairs



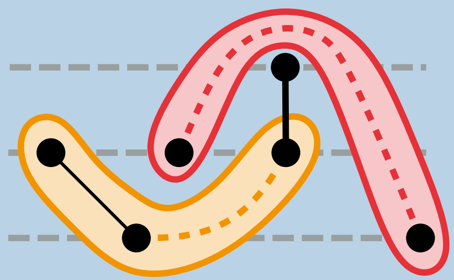
Embeddings of Clustered Level Graphs



Theorem
UNRESTRICTED CLUSTERED LEVEL PLANARITY can be solved in $O(n^3)$ time if the input graph is single-source and biconnected

Problem Variants

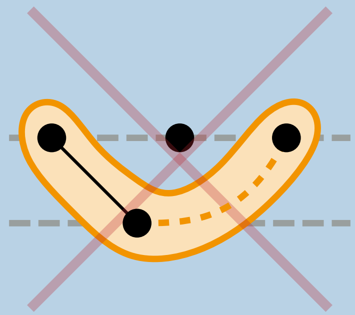
unrestricted CLP



Convex CLP

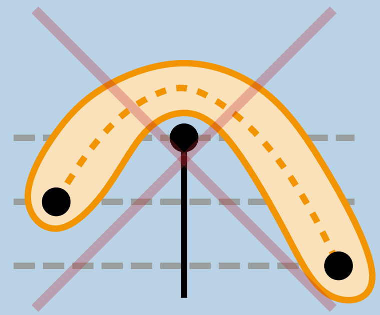
[Forster & Bachmaier '04]

clusters convex regions



y-monotone CLP

augmentation edges y-monotone



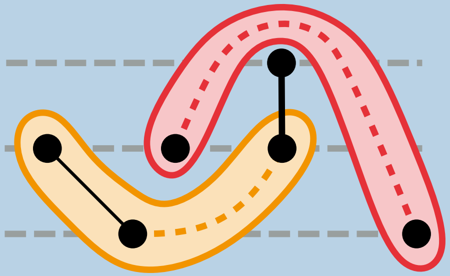
- $O(n^3)$-algorithm
  (single-source, biconnected)
  [this work]

- NP-complete
  [Angelini et al. '14]
- $\mathcal{O}(n)$-algorithm
  (proper, single-source, level-connected)
  [Forster & Bachmaier '04]
- $\mathcal{O}(n^4)$-algorithm
  (proper)
  [Angelini et al. '14]

- NP-complete
  (single-source, biconnected)
  [this work]
- NP-complete
  (constant #levels + #clusters)
  [this work]

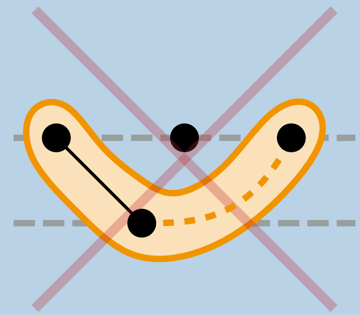
Problem Variants

unrestricted CLP



- $O(n^3)$-algorithm
  (single-source, biconnected)
  [this work]

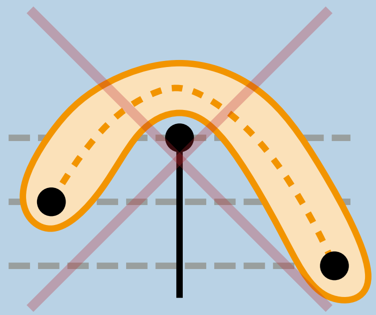
Convex CLP
[Forster & Bachmaier '04]
clusters convex regions



- NP-complete
  [Angelini et al. '14]
- $\mathcal{O}(n)$-algorithm
  (proper, single-source, level-connected)
  [Forster & Bachmaier '04]
- $\mathcal{O}(n^4)$-algorithm
  (proper)
  [Angelini et al. '14]

***y*-monotone CLP**

augmentation edges *y*-monotone



- NP-complete
  (single-source, biconnected)
  [this work]
- NP-complete
  (constant #levels + #clusters)
  [this work]

Planar Monotone 3-SAT

Input:

- Monotone 3-Sat formula Φ

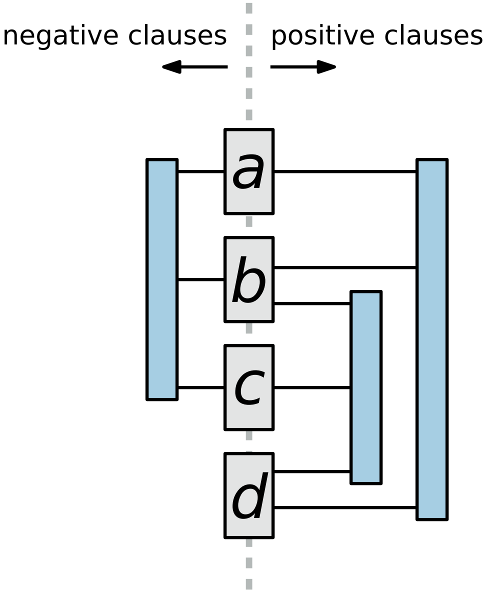
  $\Phi = (\neg a \vee \neg b \vee \neg c) \wedge (a \vee b \vee c) \wedge (b \vee c \vee d)$

PLANAR MONOTONE 3-SAT

Input:
- Monotone 3-Sat formula Φ

  $\Phi = (\neg a \vee \neg b \vee \neg c) \wedge (a \vee b \vee c) \wedge (b \vee c \vee d)$

- Planar monotone rectilinear drawing of incidence graph of Φ

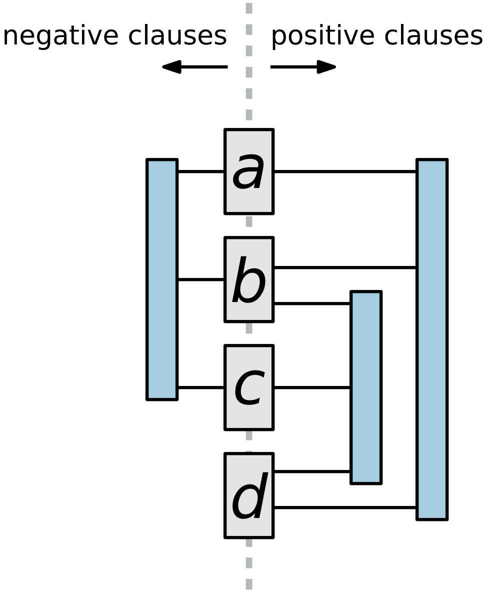


Planar Monotone 3-SAT

Input:
- Monotone 3-Sat formula Φ

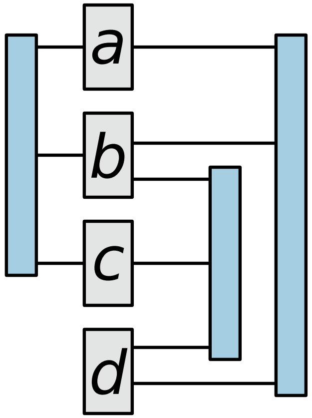
  $\Phi = (\neg a \vee \neg b \vee \neg c) \wedge (a \vee b \vee c) \wedge (b \vee c \vee d)$

- Planar monotone rectilinear drawing of incidence graph of Φ

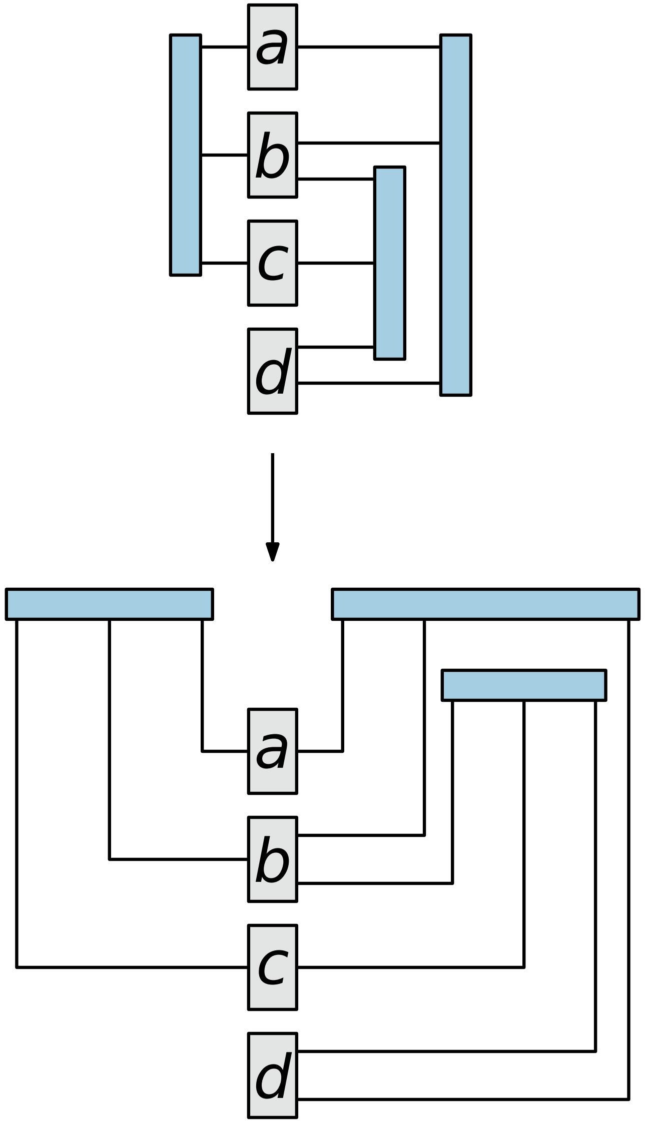


Question: Is Φ satisfiable?

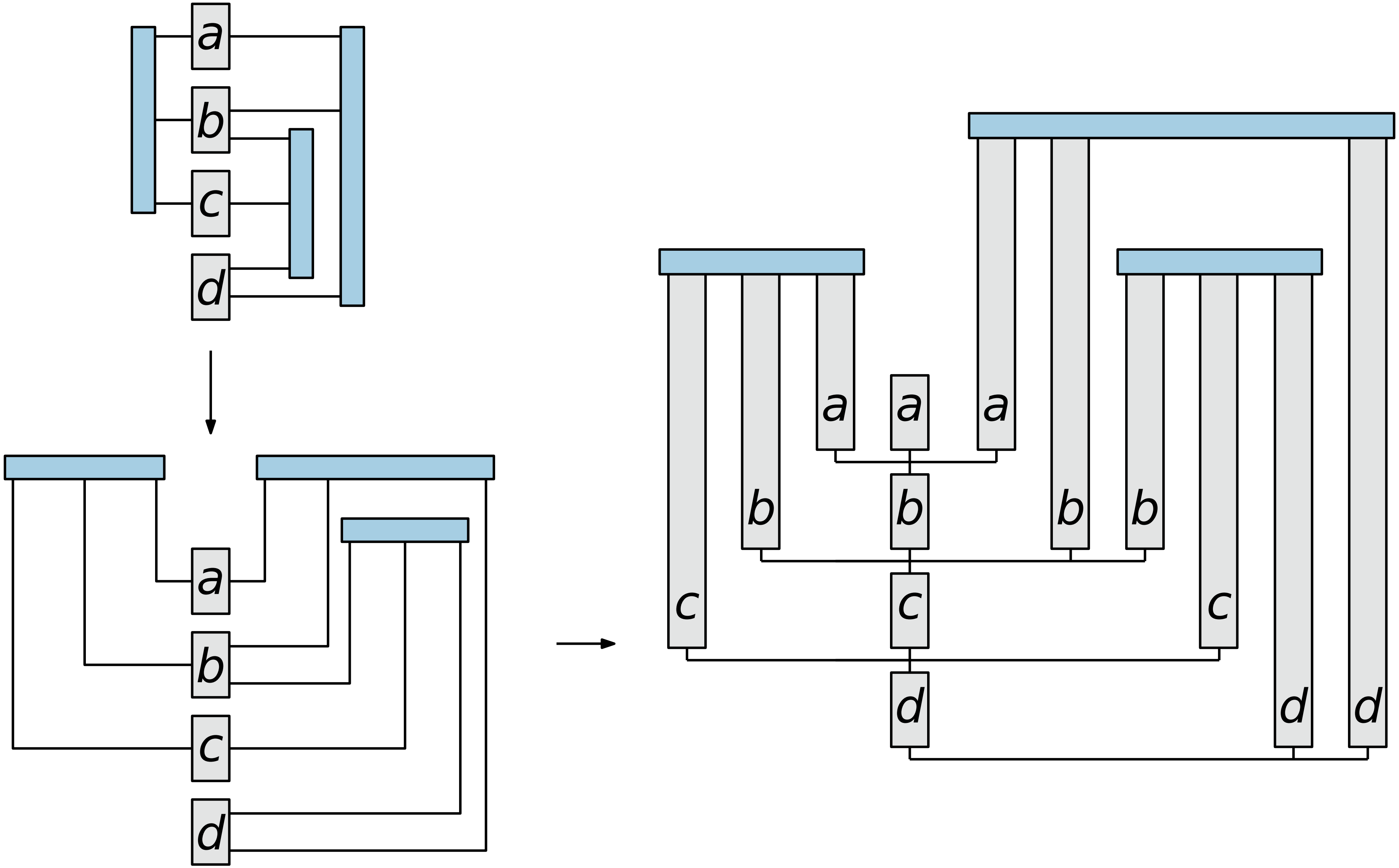
Planar Monotone 3-SAT $\leq$ y-Monotone CLP



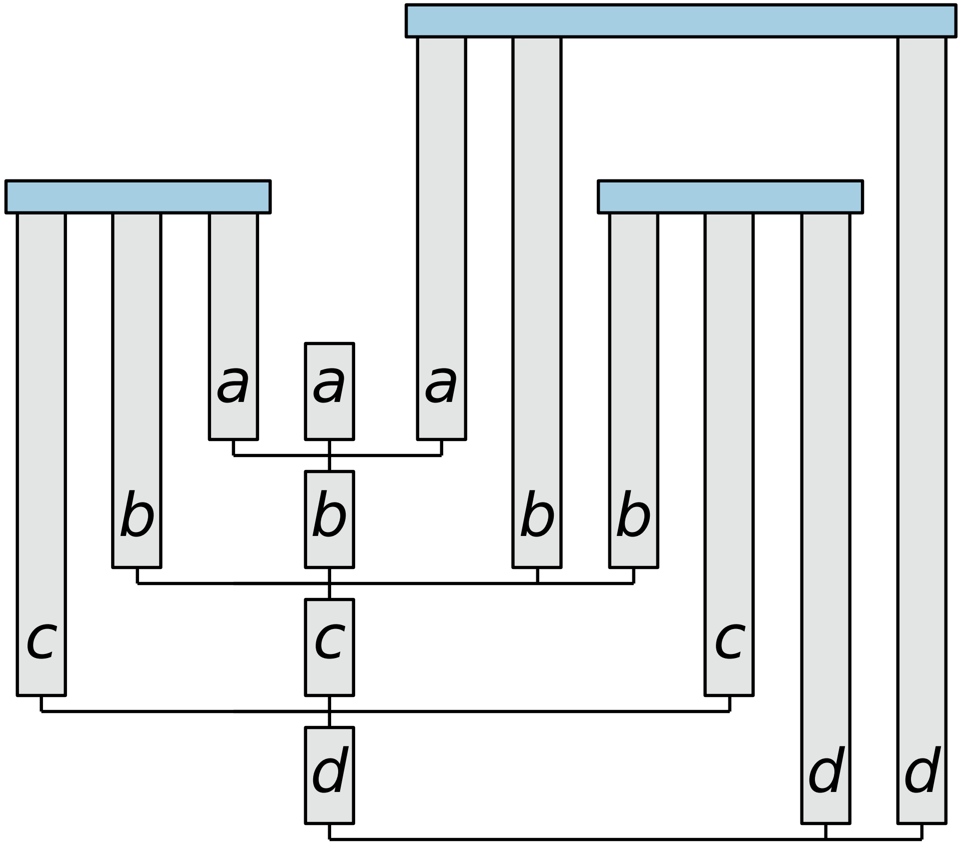
Planar Monotone 3-SAT $\leq$ y-Monotone CLP



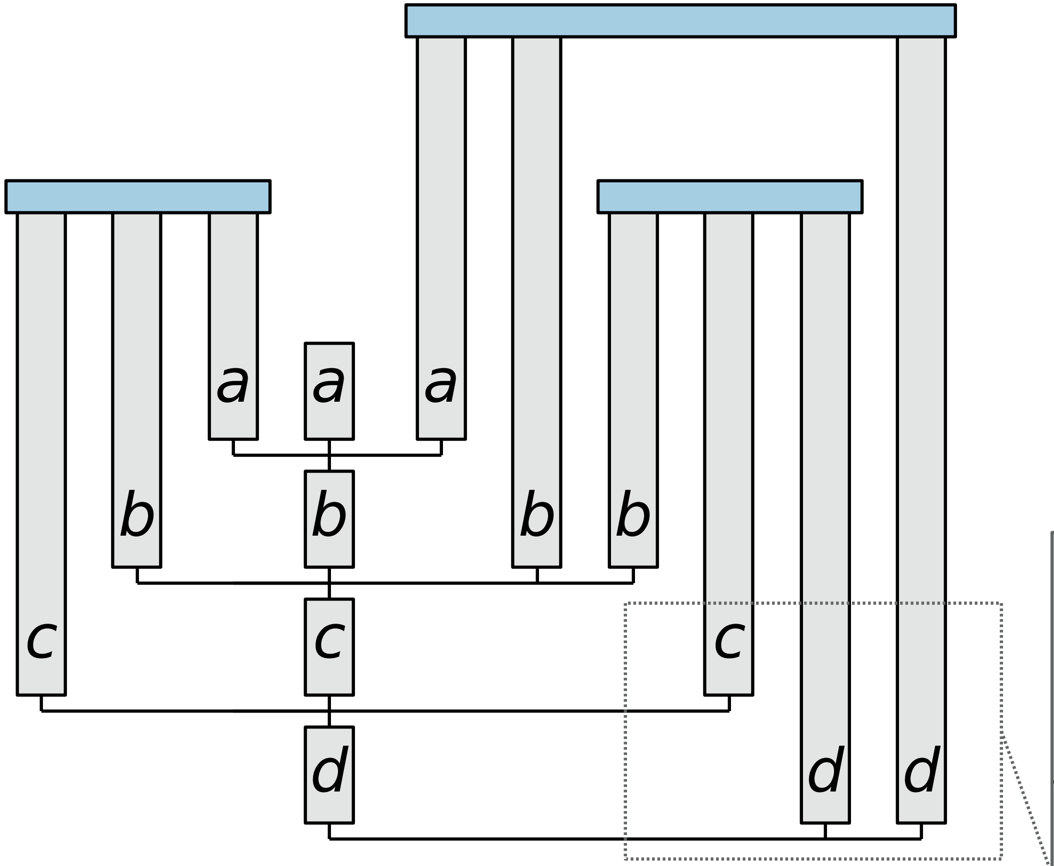
PLANAR MONOTONE 3-SAT ≤ y-MONOTONE CLP



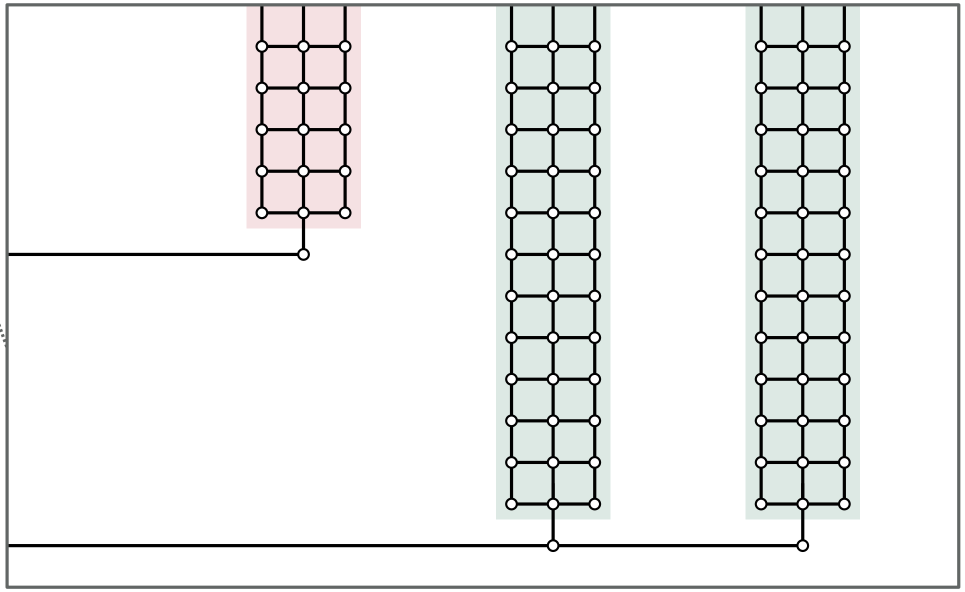
Planar Monotone 3-SAT ≤ y-Monotone CLP



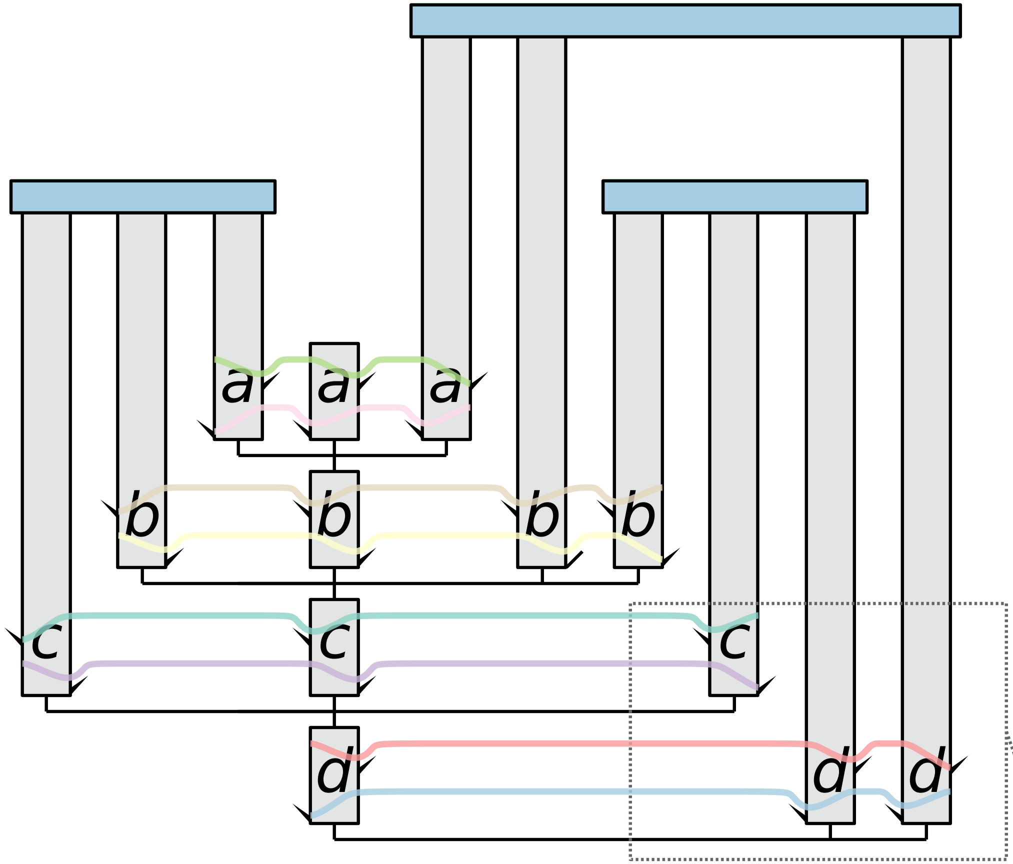
PLANAR MONOTONE 3-SAT ≤ y-MONOTONE CLP



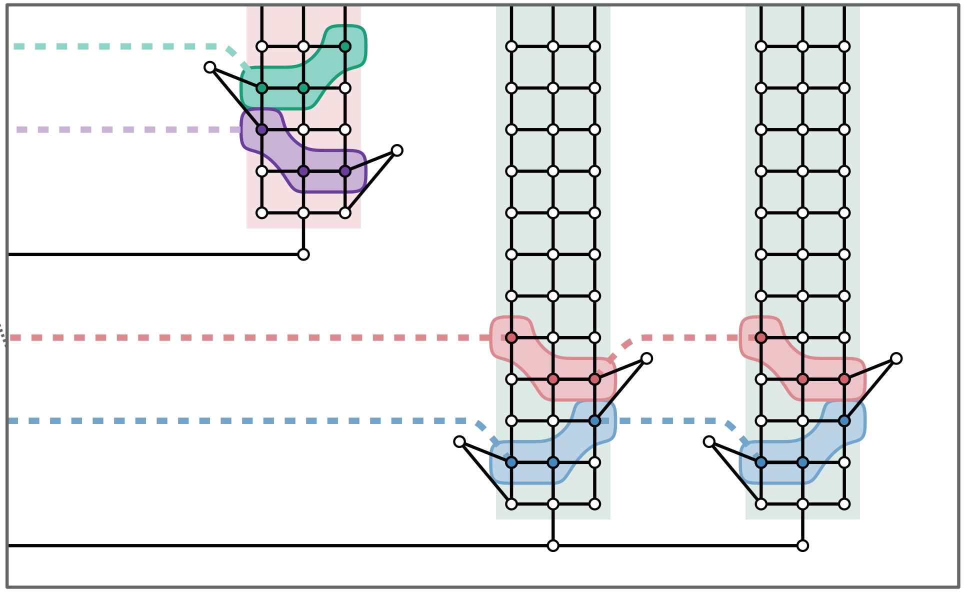
- literals are triconnected pillars (flip ≡ truth value)



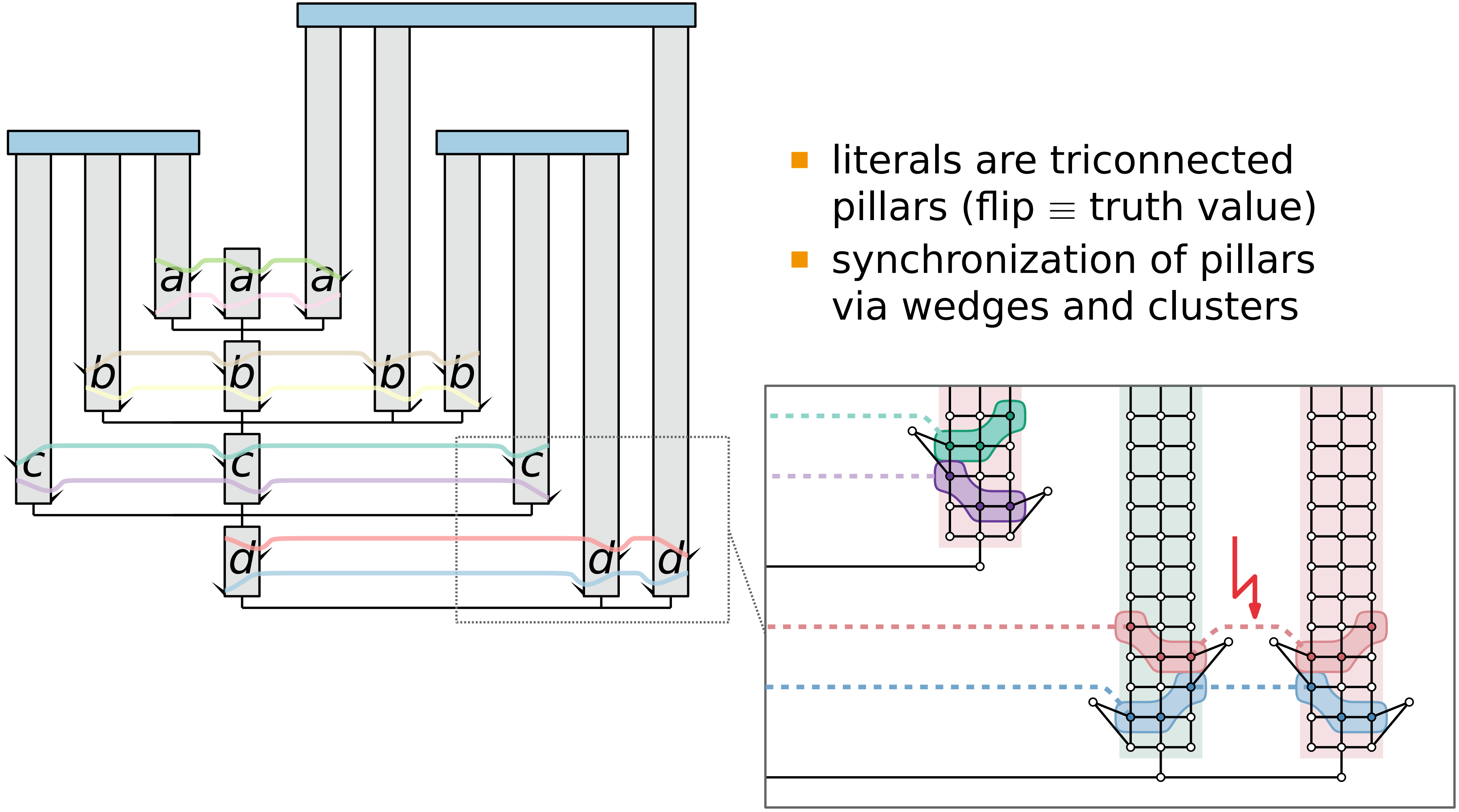
Planar Monotone 3-SAT ≤ y-Monotone CLP



- literals are triconnected pillars (flip ≡ truth value)
- synchronization of pillars via wedges and clusters



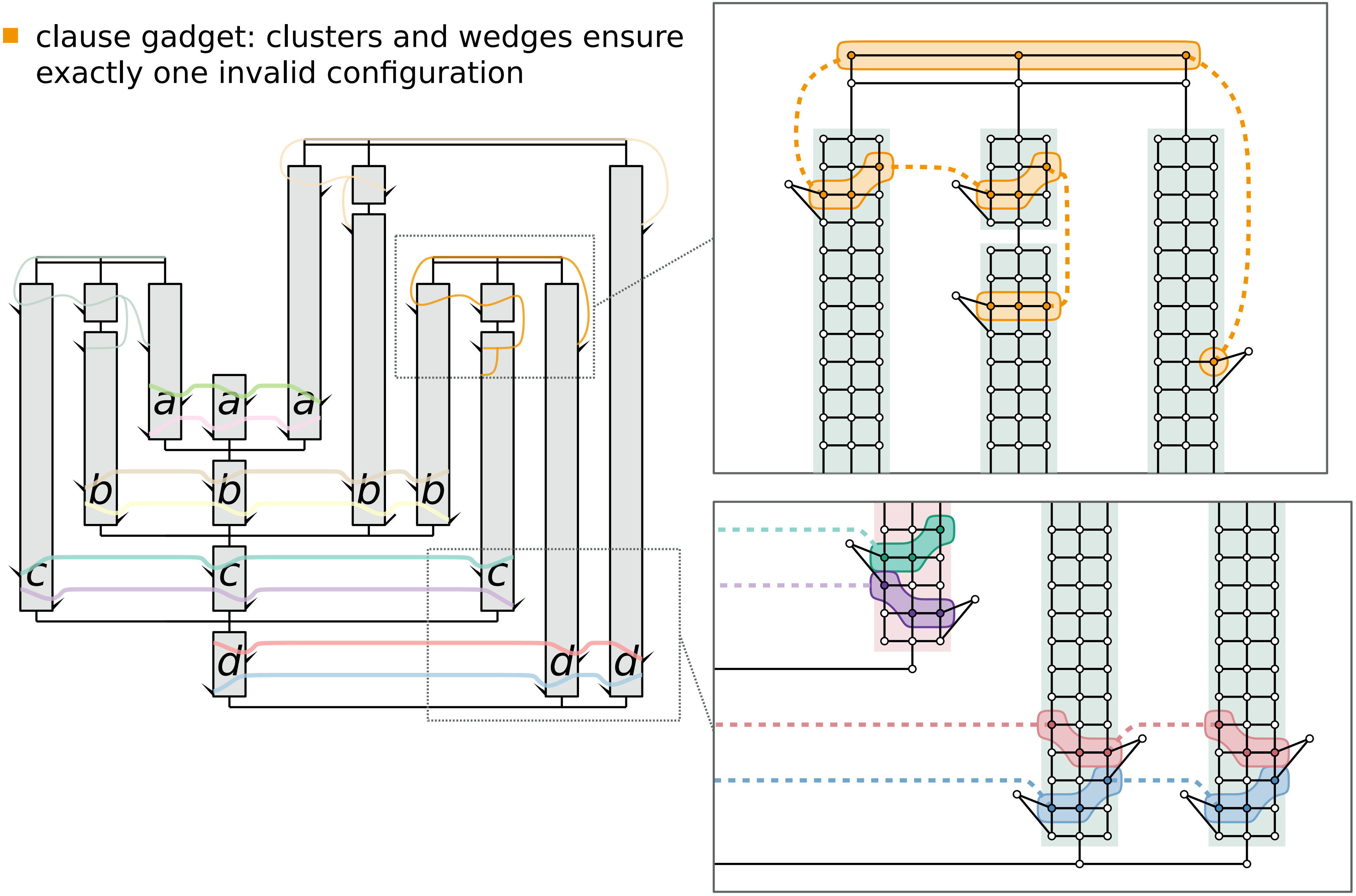
PLANAR MONOTONE 3-SAT $\leq$ y-MONOTONE CLP



- literals are triconnected pillars (flip $\equiv$ truth value)
- synchronization of pillars via wedges and clusters

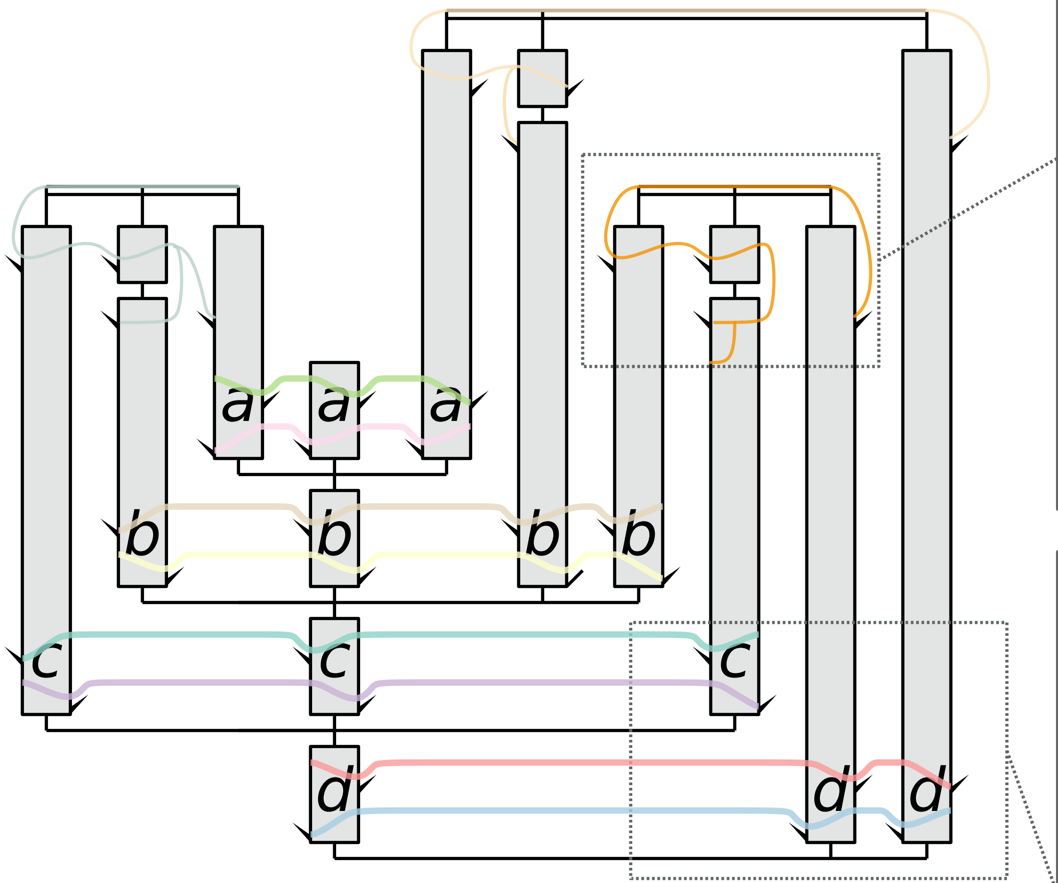
PLANAR MONOTONE 3-SAT ≤ y-MONOTONE CLP

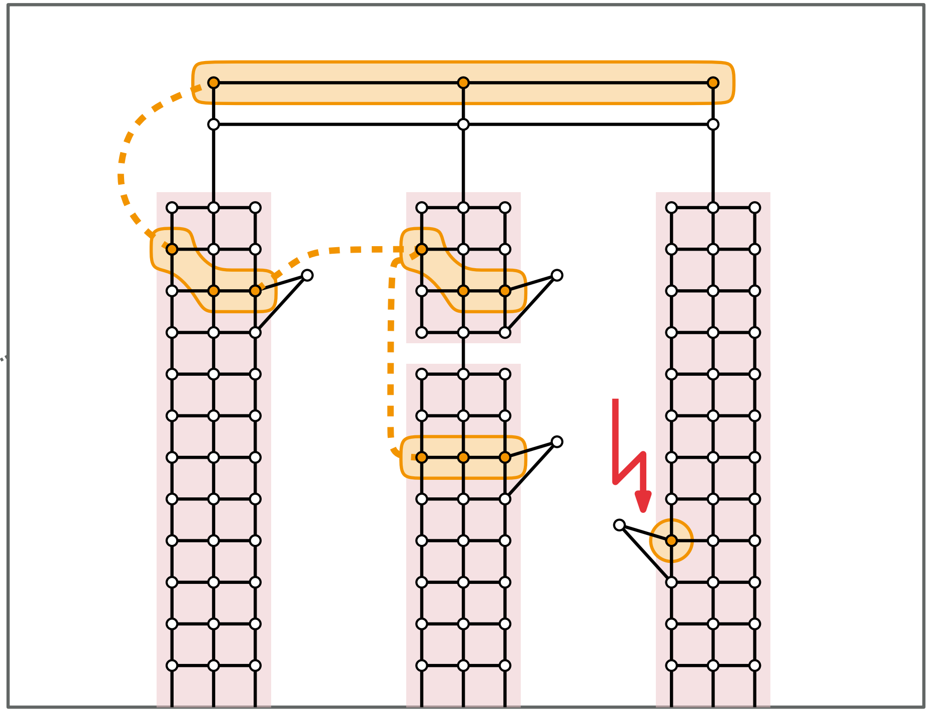
- clause gadget: clusters and wedges ensure exactly one invalid configuration

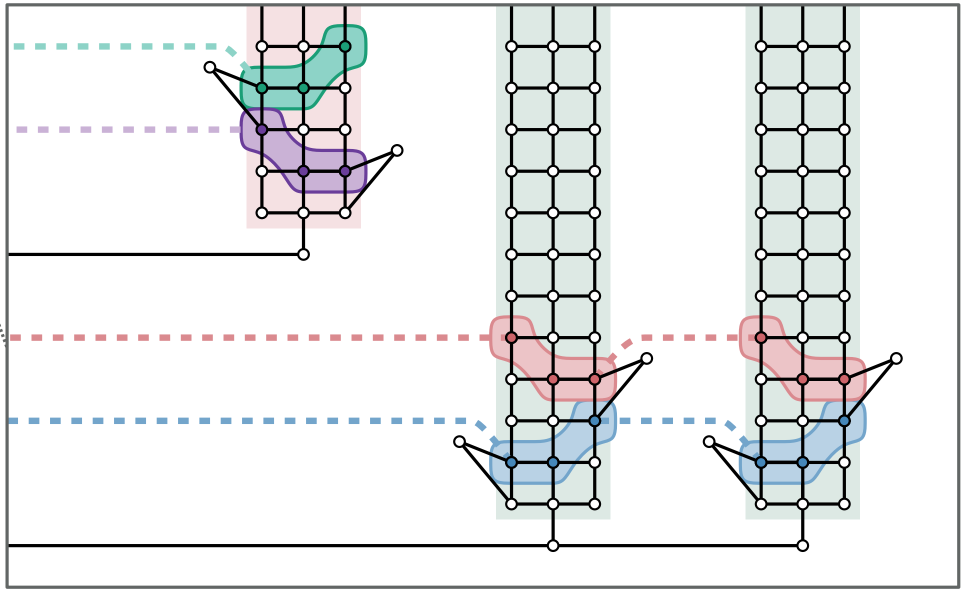


PLANAR MONOTONE 3-SAT $\leq$ y-MONOTONE CLP

- clause gadget: clusters and wedges ensure exactly one invalid configuration

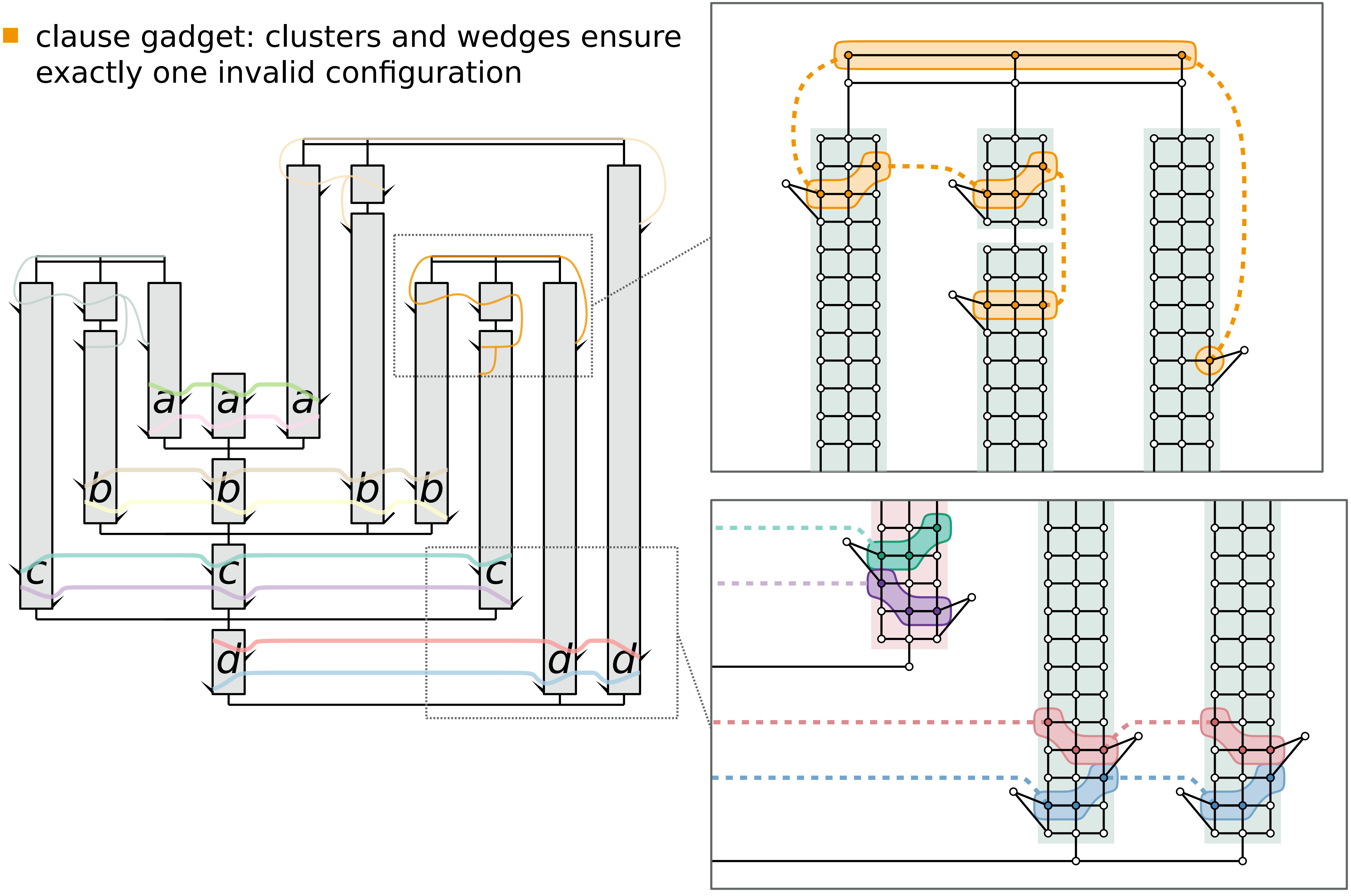






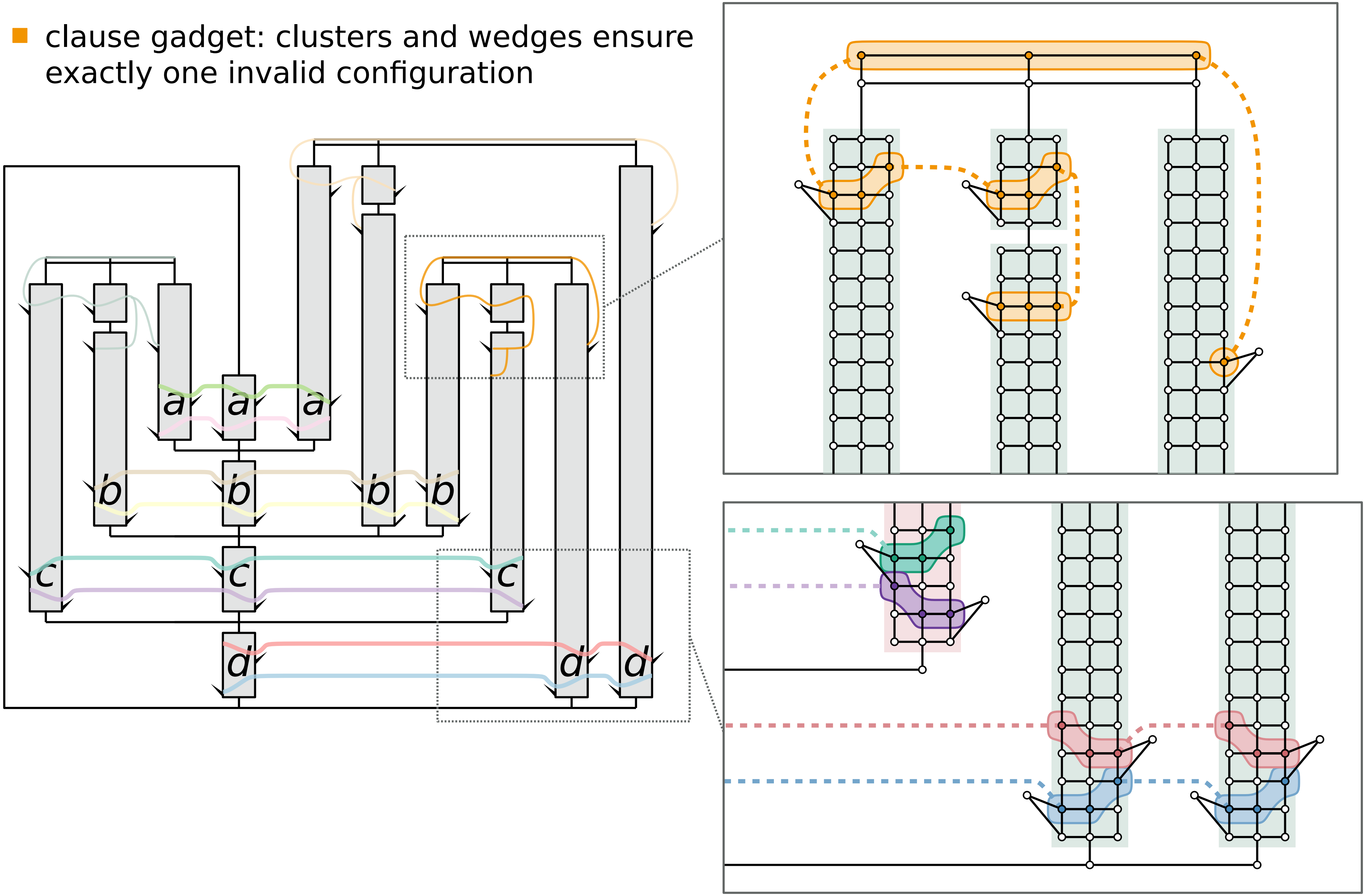
Planar Monotone 3-SAT ≤ y-Monotone CLP

- clause gadget: clusters and wedges ensure exactly one invalid configuration



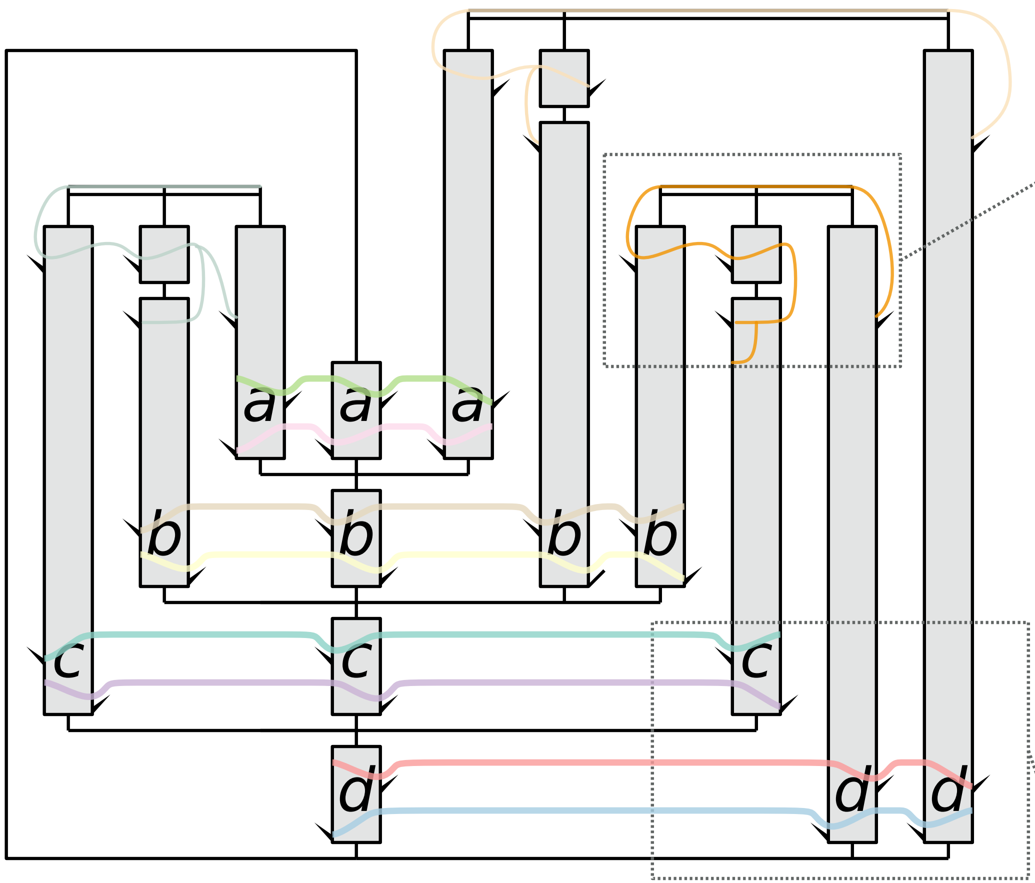
Planar Monotone 3-SAT ≤ y-Monotone CLP

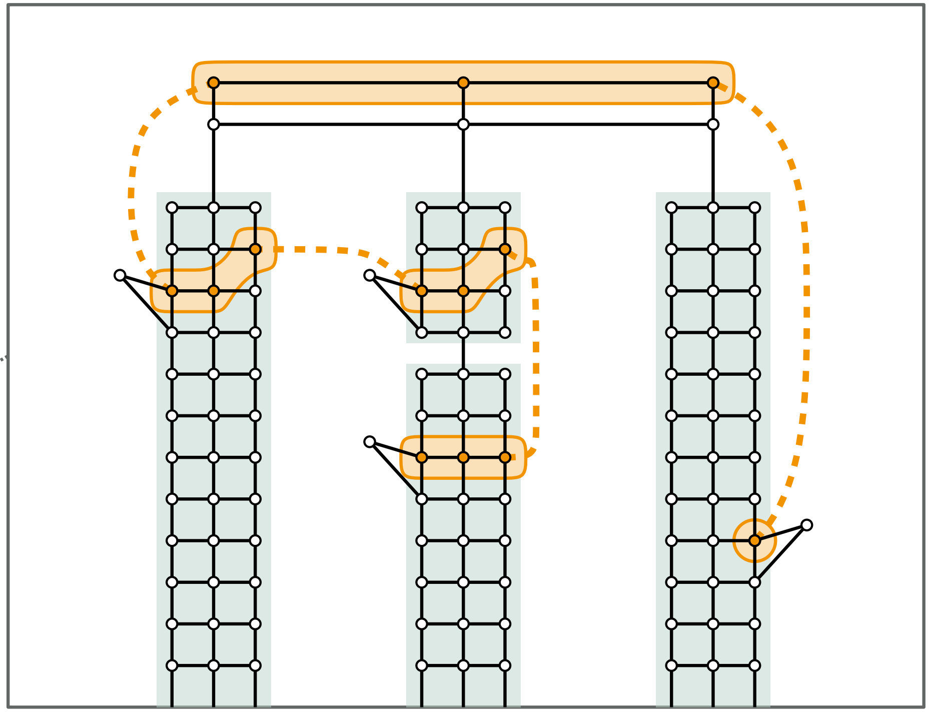
- clause gadget: clusters and wedges ensure exactly one invalid configuration

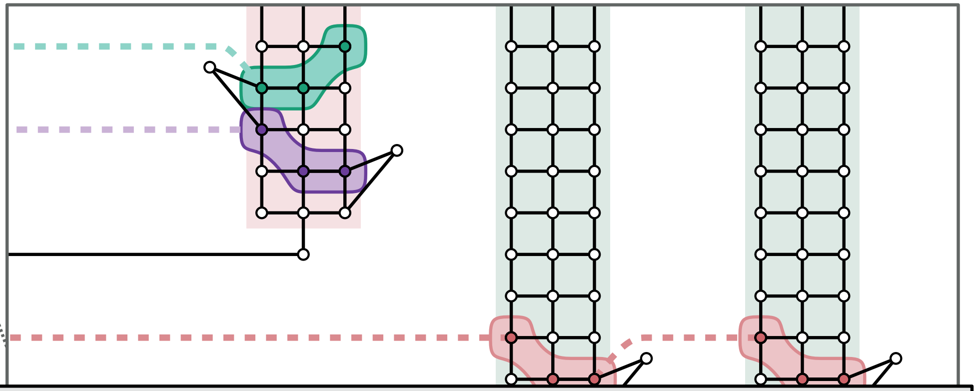


Planar Monotone 3-SAT ≤ y-Monotone CLP

- clause gadget: clusters and wedges ensure exactly one invalid configuration



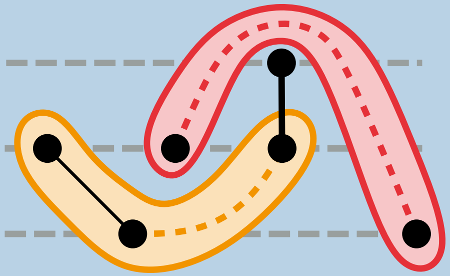




Theorem
y-Monotone Clustered Level Planarity is NP-complete, even if the input graph is biconnected and single-source

Summary

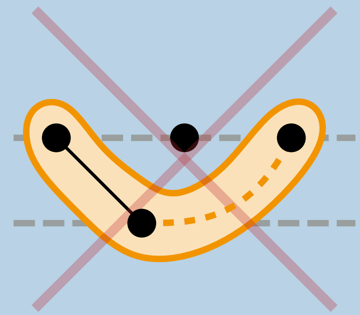
unrestricted CLP



- $O(n^3)$-algorithm
  (single-source, biconnected)
  [this work]

Convex CLP
[Forster & Bachmaier ’04]

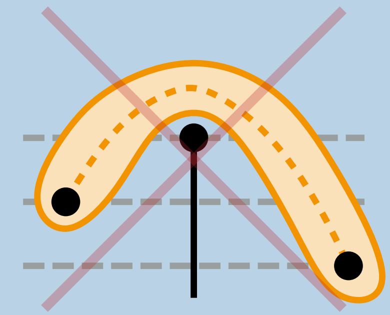
clusters convex regions



- NP-complete
  [Angelini et al. ’14]
- $\mathcal{O}(n)$-algorithm
  (proper, single-source, level-connected)
  [Forster & Bachmaier ’04]
- $\mathcal{O}(n^4)$-algorithm
  (proper)
  [Angelini et al. ’14]

y-monotone CLP

augmentation edges y-monotone



- NP-complete
  (single-source, biconnected)
  [this work]
- NP-complete
  (constant #levels + #clusters)
  [this work]