Moving Vertices to Make Drawings Plane

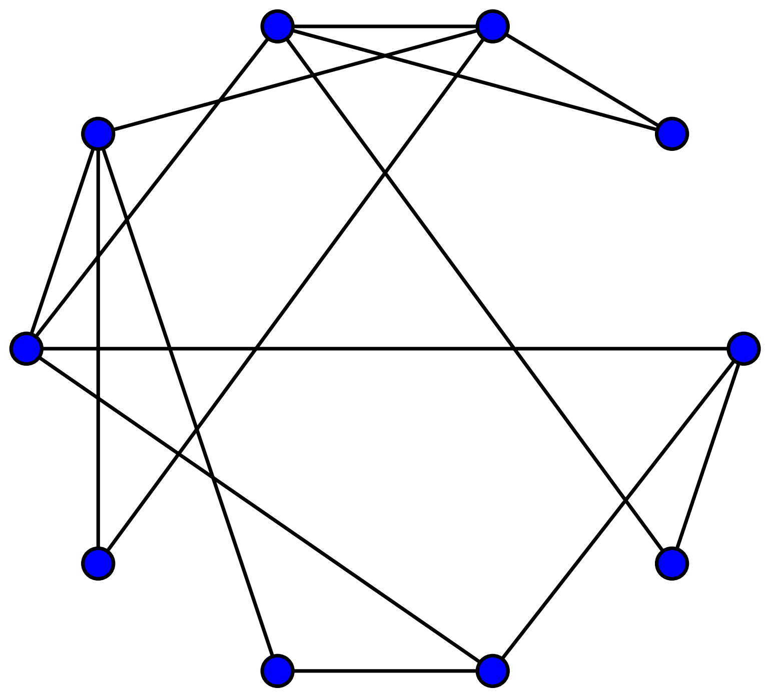
Xavier Goaoc	INRIA Lorraine	FR
Jan Kratochvíl	Charles U	CZ
Yoshio Okamoto	Toyohashi U Tech	JP
Chan-Su Shin	Hankuk U Foreign Studies	KR
Alexander Wolff	TU Eindhoven	NL

September 24, 2007 @ 15th International Conference on Graph Drawing

Swiss-Grand Resort & Spa Bondi Beach, Sydney, Australia

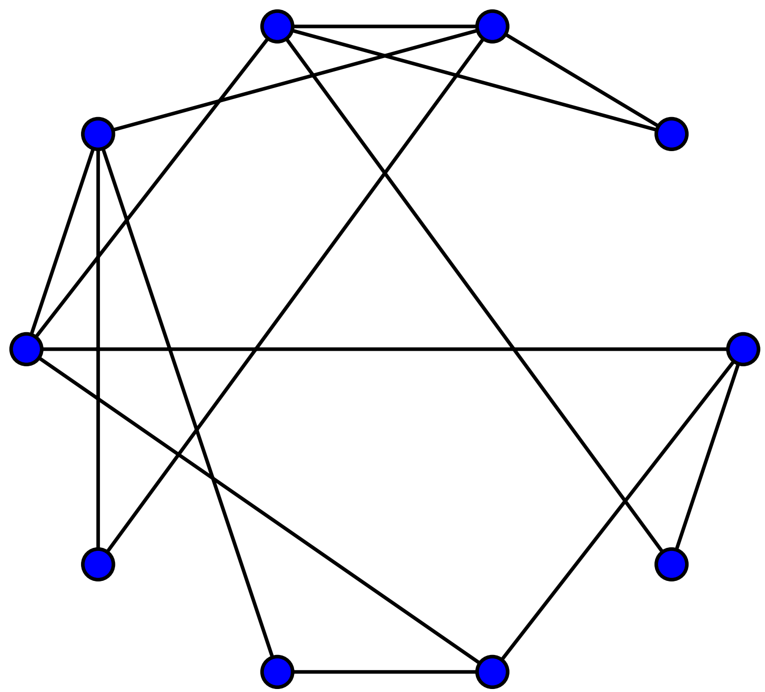
http://www.planarity.net/

Given: a straight-line drawing of a planar graph G



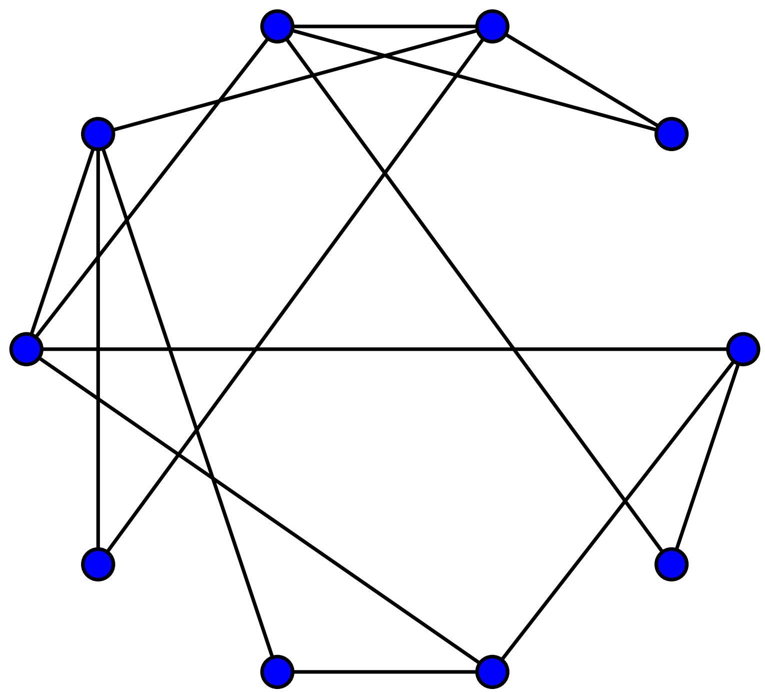
http://www.planarity.net/

Given:	a straight-line drawing of a planar graph G
Task:	to make it non-crossing (i.e., plane)



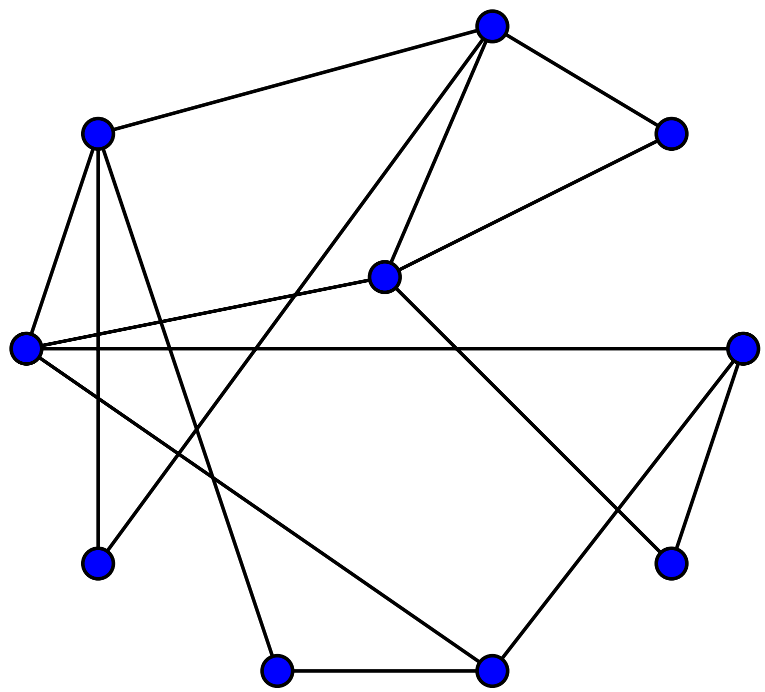
http://www.planarity.net/

Given:	a straight-line drawing of a planar graph G
Task:	to make it non-crossing (i.e., plane)
Operation:	move vertices on the plane (by dragging)



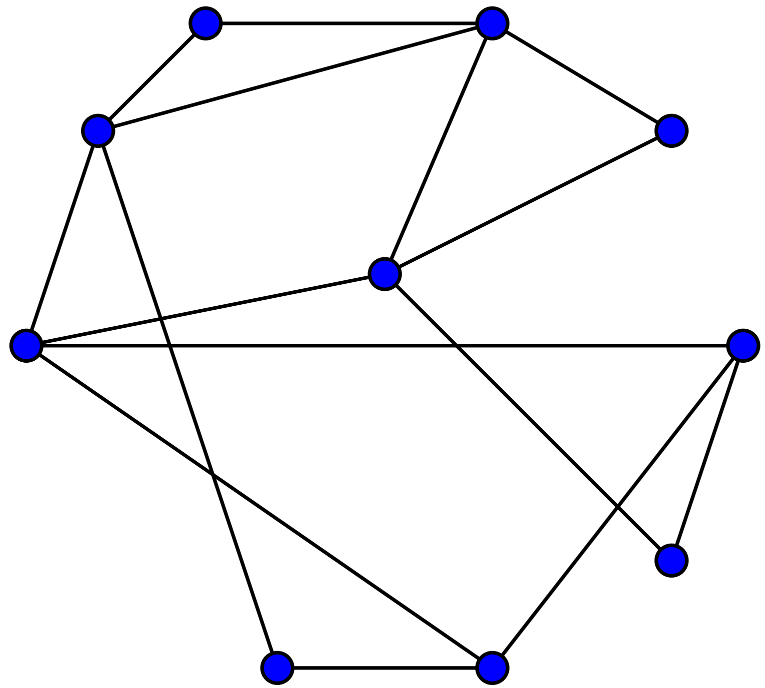
http://www.planarity.net/

Given:	a straight-line drawing of a planar graph G
Task:	to make it non-crossing (i.e., plane)
Operation:	move vertices on the plane (by dragging)



http://www.planarity.net/

Given:	a straight-line drawing of a planar graph G
Task:	to make it non-crossing (i.e., plane)
Operation:	move vertices on the plane (by dragging)

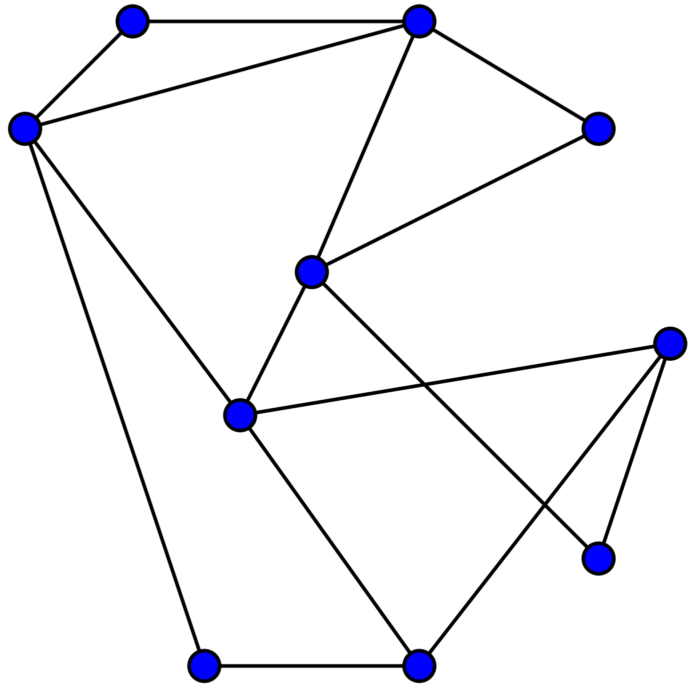


http://www.planarity.net/

Given: a straight-line drawing of a planar graph G

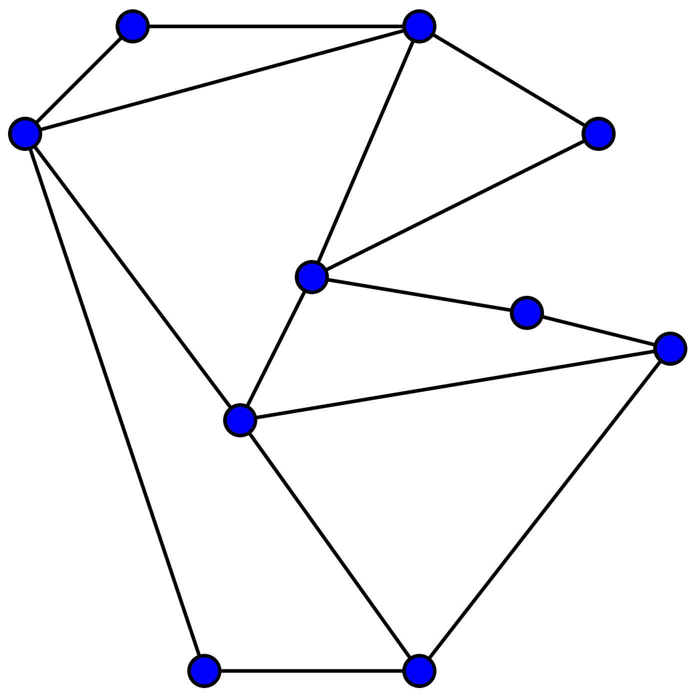
Task: to make it non-crossing (i.e., plane)

Operation: move vertices on the plane (by dragging)



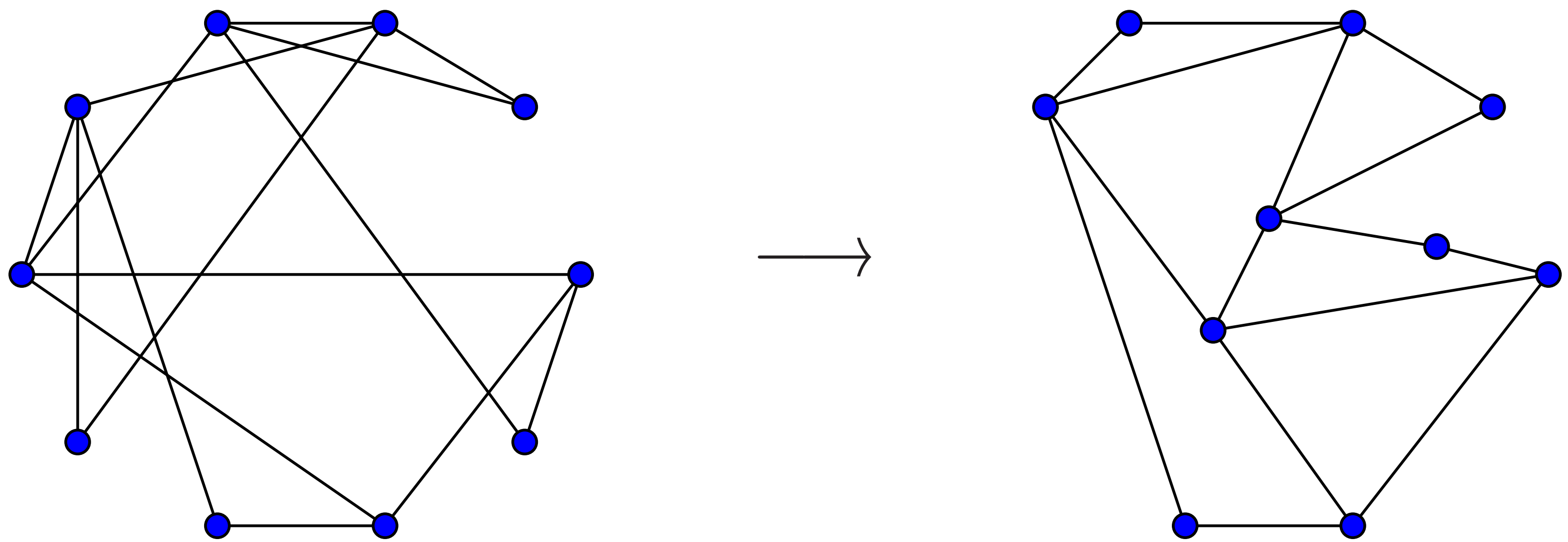
http://www.planarity.net/

Given:	a straight-line drawing of a planar graph G
Task:	to make it non-crossing (i.e., plane)
Operation:	move vertices on the plane (by dragging)



Questions

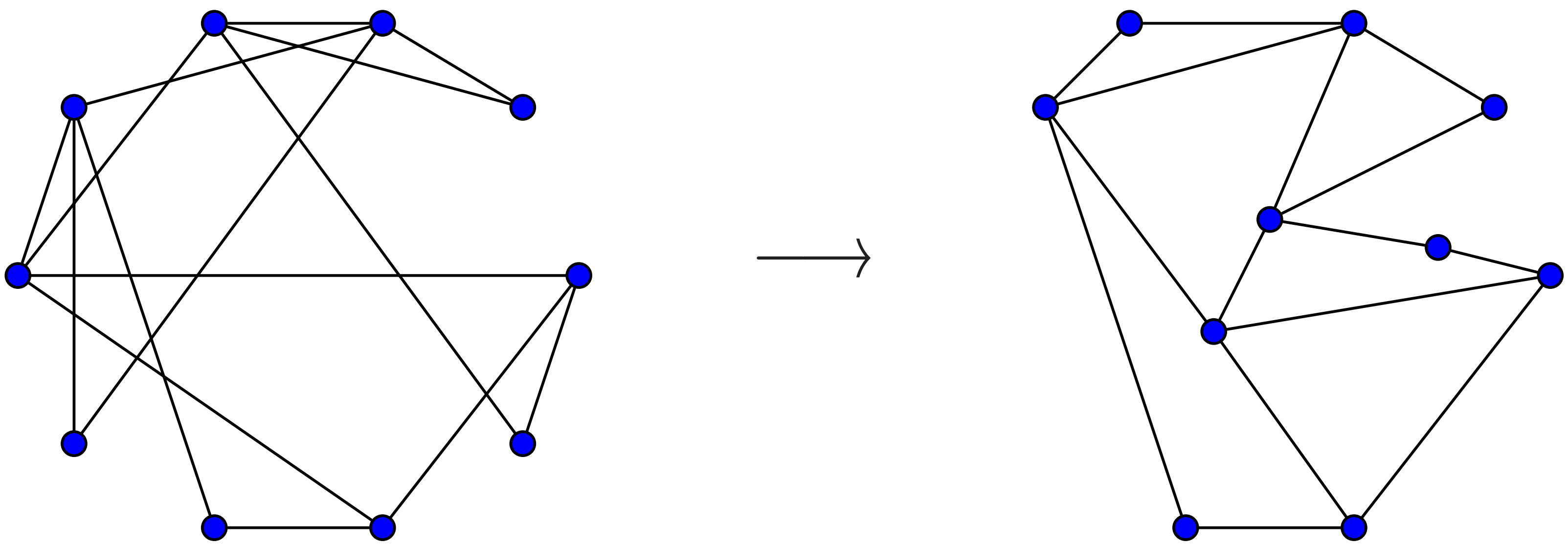
- (computational question)
- (combinatorial question)



Questions

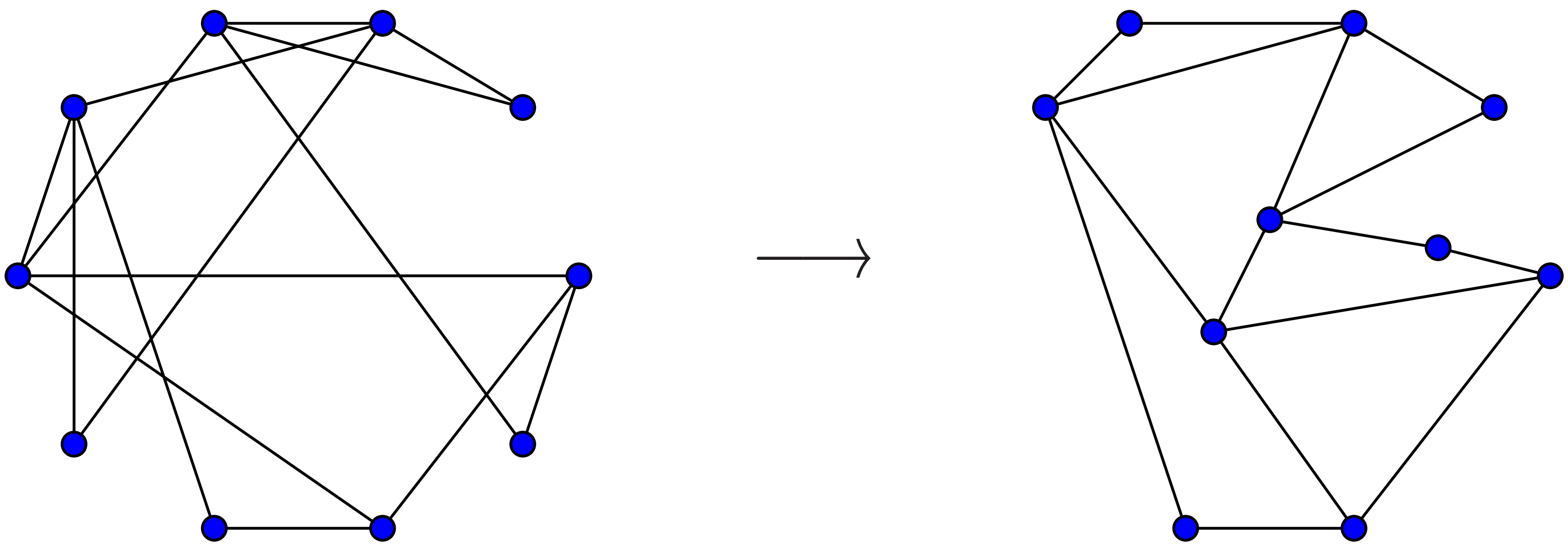
- (computational question)
  How hard to find a min # of vertices to move?

- (combinatorial question)



Questions

- (computational question)
  How hard to find a min # of vertices to move?
- (combinatorial question)
  How many vertices must be moved in worst case?



Theorem

- It is **NP-hard** to compute the min # of vertices to move in order to make a given drawing plane

Theorem

- It is **NP-hard** to compute the min # of vertices to move in order to make a given drawing plane
- It is **NP-hard to approximate** $(1 + \min \#)$ within a factor of $n^{1-\varepsilon}$ (for any fixed $\varepsilon \in (0, 1]$)

$n = \#$ of vertices

We switch to the max # of vertices that we can keep fixed

- For n-vertex **cycles** (Pach & Tardos (GD '01, DCG '02))
  we can always keep $\lfloor \sqrt{n} \rfloor$ vertices
  we can't keep $O((n \log n)^{2/3})$ vertices in some cases

Theorem

- For n-vertex **trees**

- For n-vertex **planar graphs**

We switch to the max # of vertices that we can keep fixed

- For n-vertex **cycles** (Pach & Tardos (GD '01, DCG '02))
  we can always keep $\lfloor\sqrt{n}\rfloor$ vertices
  we can't keep $O((n\log n)^{2/3})$ vertices in some cases

Theorem

- For n-vertex **trees**
  we can always keep $\lfloor\sqrt{n/3}\rfloor$ vertices
  we can't keep $\lceil n/3\rceil+4$ vertices in some cases

- For n-vertex **planar graphs**

We switch to the max # of vertices that we can keep fixed

- For n-vertex **cycles** (Pach & Tardos (GD '01, DCG '02))
  we can always keep $\lfloor\sqrt{n}\rfloor$ vertices
  we can't keep $O((n \log n)^{2/3})$ vertices in some cases

Theorem

- For n-vertex **trees**
  we can always keep $\lfloor\sqrt{n/3}\rfloor$ vertices
  we can't keep $\lceil n/3\rceil+4$ vertices in some cases

- For n-vertex **planar graphs**
  we can always keep 3 vertices
  we can't keep $\lceil\sqrt{n-2}\rceil+1$ vertices in some cases

We switch to the max # of vertices that we can keep fixed

Theorem

	Lower Bound	Upper Bound
Cycles	$\lfloor\sqrt{n}\rfloor$	$O((n\log n)^{2/3})$
Trees	$\lfloor\sqrt{n}/3\rfloor$	$\lceil n/3\rceil+4$
General	3	$\lceil\sqrt{n-2}\rceil+1$

Pach & Tardos (GD '01, DCG '02)

- Aug 06: this work started
- Jun 07: submitted to GD
- Jul 07: accepted for GD
- Today: presented at GD

- **Aug 06: this work started**
- May 07: Verbitsky @ arXiv
- **Jun 07: submitted to GD**
- **Jul 07: accepted for GD**
- **Today: presented at GD**

- **Aug 06: this work started**
- May 07: Verbitsky @ arXiv
- **Jun 07: submitted to GD**
- Jun 07: aware of Pach & Tardos (DCG '02)
- **Jul 07: accepted for GD**

- **Today: presented at GD**

- **Aug 06: this work started**
- May 07: Verbitsky @ arXiv
- **Jun 07: submitted to GD**
- Jun 07: aware of Pach & Tardos (DCG '02)
- **Jul 07: accepted for GD**
- Jul 07: Kang, Schacht & Verbitsky @ arXiv

- **Today: presented at GD**

- **Aug 06: this work started**
- May 07: Verbitsky @ arXiv
- **Jun 07: submitted to GD**
- Jun 07: aware of Pach & Tardos (DCG '02)
- **Jul 07: accepted for GD**
- Jul 07: Kang, Schacht & Verbitsky @ arXiv
- Sep 07: Spillner & Wolff @ arXiv
- **Today: presented at GD**

We concentrate on the complexity result

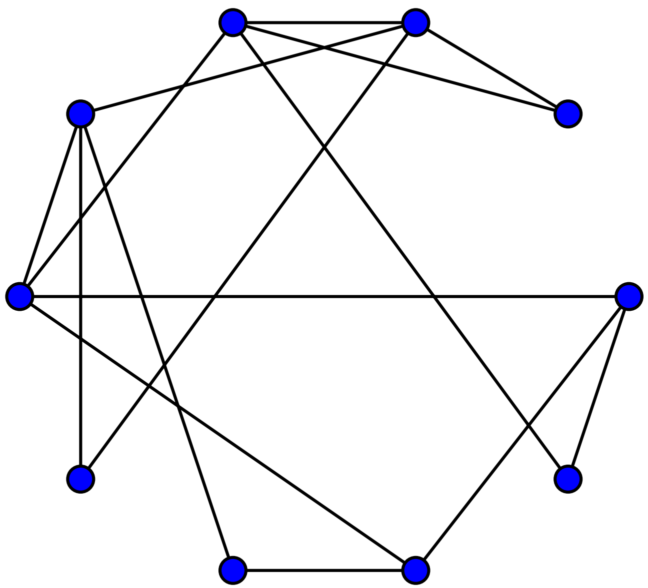
- Problem statement (more formally)
- NP-hardness proof
- Inapproximability (briefly)
- Connection to the one-bend embeddability problem

Def.: Straight-Line Drawing

Setup: $G = (V, E)$ an undirected graph (w/o loop or parallel edges)

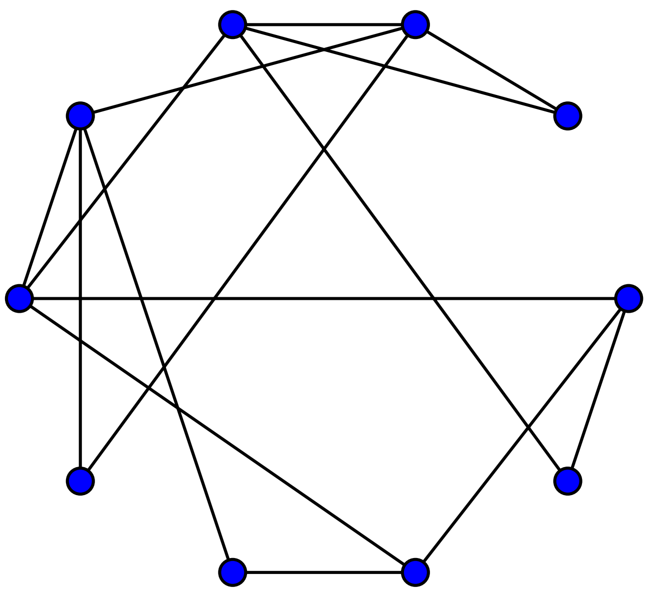
Setup: $G = (V, E)$ an undirected graph (w/o loop or parallel edges)

Def: A **(straight-line) drawing** of G is an injective map $\delta \colon V \to \mathbb{R}^2$, image of $\{u, v\} \in E$ is a line segment $\overline{\delta(u)\delta(v)}$



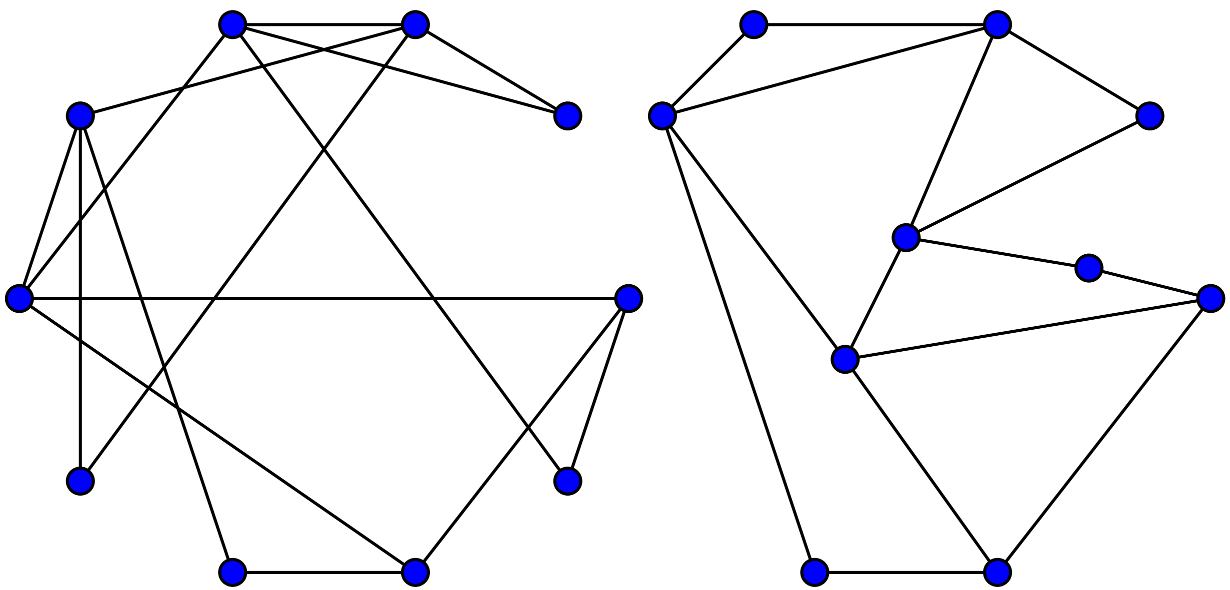
Setup: $G = (V, E)$ an undirected graph (w/o loop or parallel edges)

Def: A drawing δ of G is **plane** if (the images under δ of) two edges are only allowed to share a common endpoint



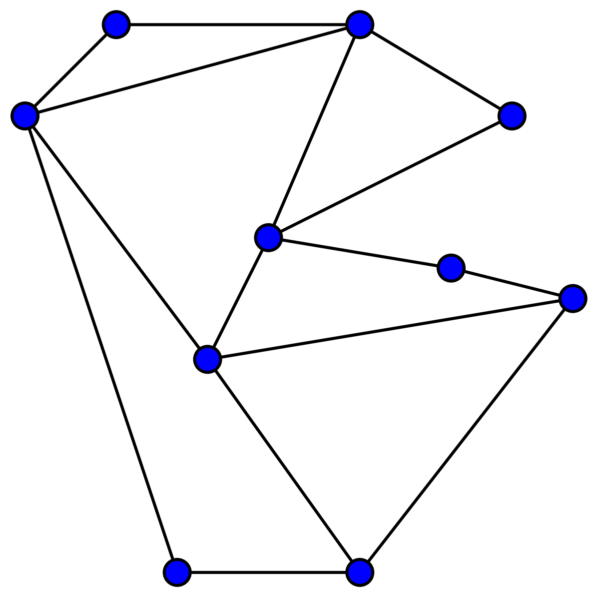
Setup: $G = (V, E)$ an undirected graph (w/o loop or parallel edges)

Def: A drawing δ of G is **plane** if (the images under δ of) two edges are only allowed to share a common endpoint



Setup: $G = (V, E)$ an undirected graph (w/o loop or parallel edges)

Def: A graph G is **planar** if $\exists$ a plane drawing of G (characterization due to Fáry '48)

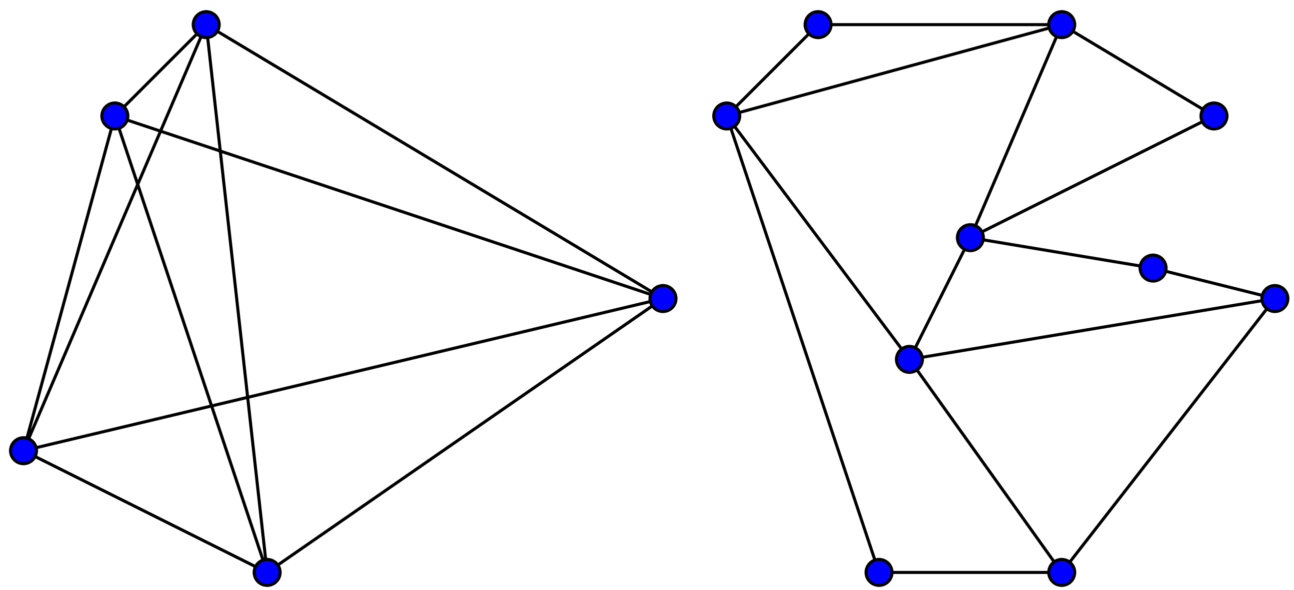


Def.: Planar Graph

Setup: $G = (V, E)$ an undirected graph (w/o loop or parallel edges)

Def: A graph G is **planar** if $\exists$ a plane drawing of G

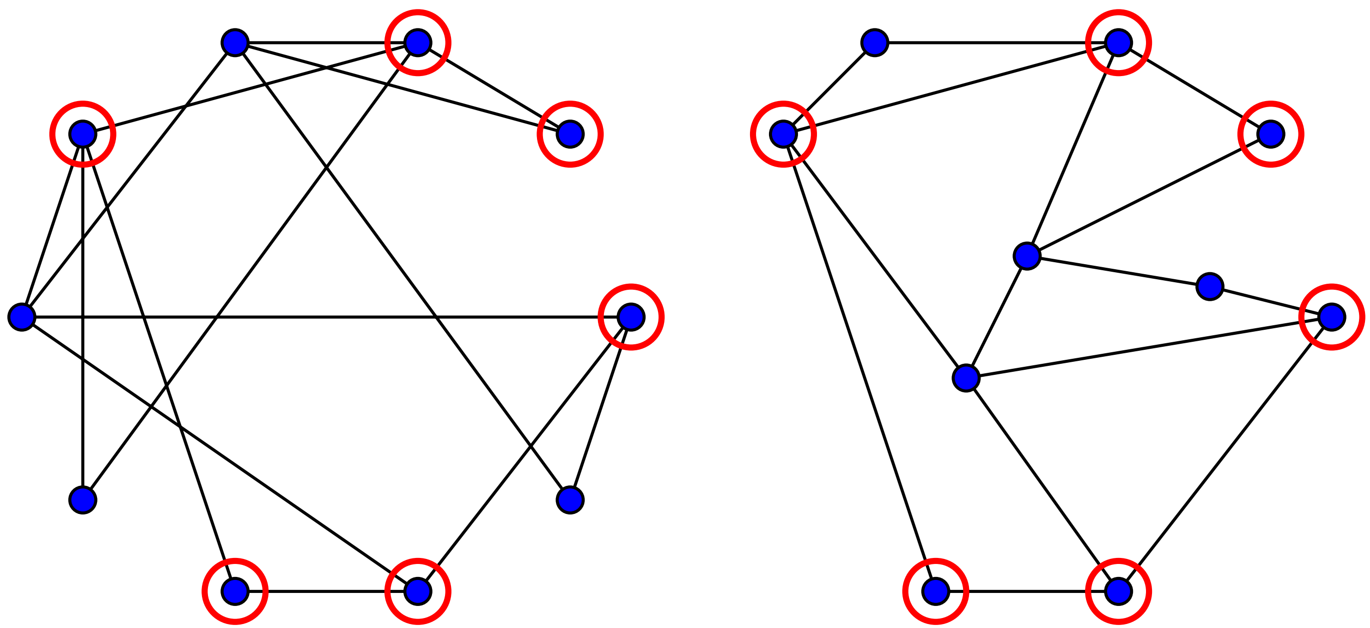
(characterization due to Fáry '48)



Def.: Distance of Drawings

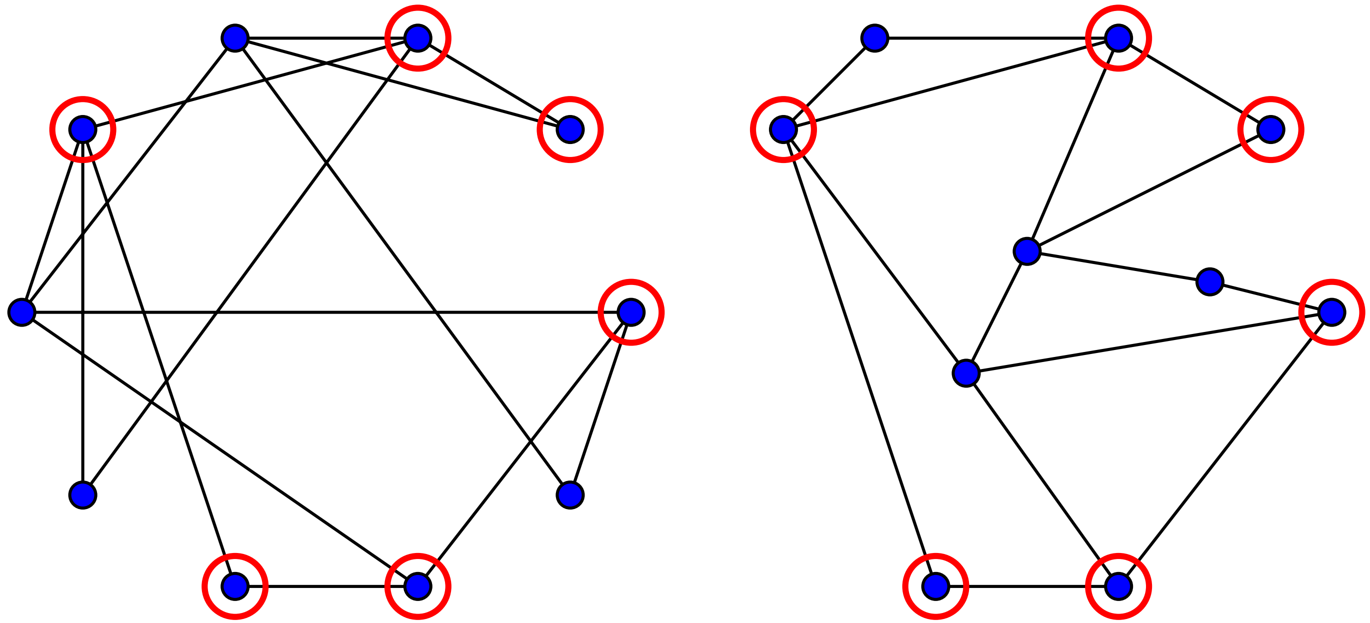
Setup: $G = (V, E)$ an undirected graph (w/o loop or parallel edges)

Def: The **distance** of two drawings δ, δ' of G is
$$d(\delta, \delta') = |\{v \in V \mid \delta(v) \neq \delta'(v)\}|$$



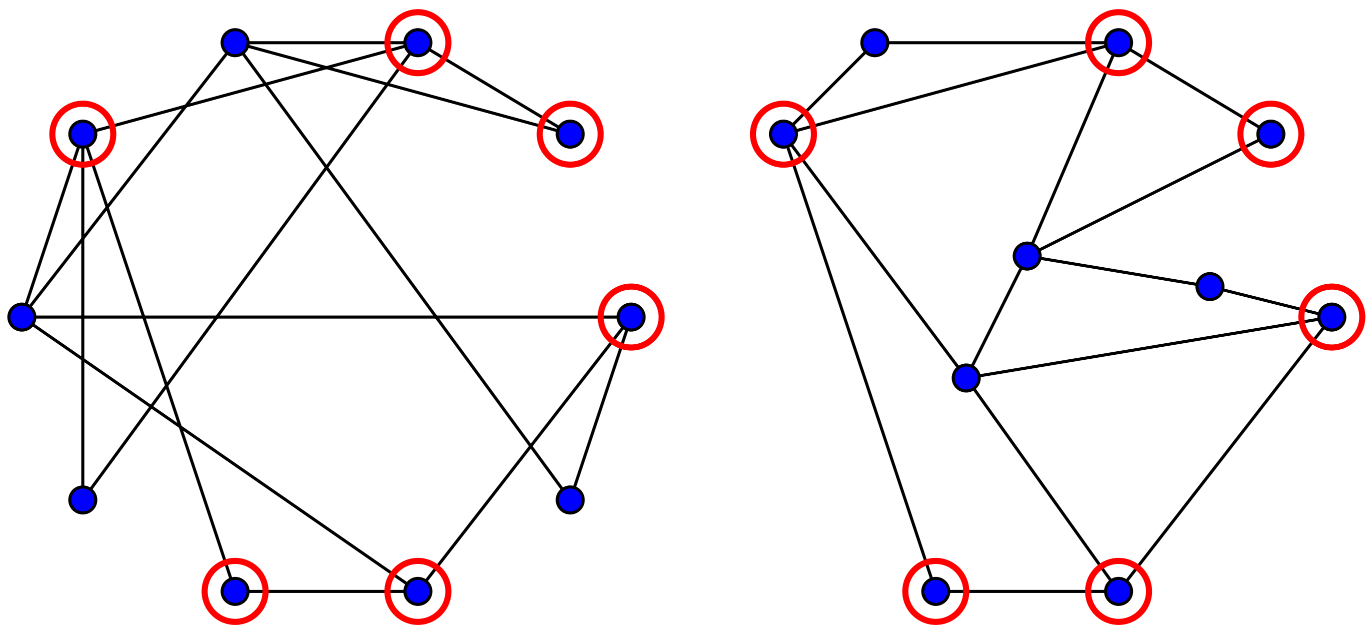
Setup: $G = (V, E)$ an undirected graph (w/o loop or parallel edges)
δ a drawing of G

Def: $\mathrm{MMV}(G, \delta) = \min_{\delta' \text{plane}} d(\delta, \delta')$



Setup: $G = (V, E)$ an n-vertex undirected graph (w/o loop or parallel edges)
δ a drawing of G

Def: $\mathrm{MKV}(G, \delta) = n - \mathrm{MMV}(G, \delta)$



Contents

- Problem statement (more formally)
- **NP-hardness proof**
- Inapproximability (briefly)
- Connection to the one-bend embeddability problem

Theorem

- For a given planar graph G and a drawing δ of G, it is NP-hard to compute $\mathrm{MMV}(G, \delta)$

Theorem

- For a given planar graph G and a drawing δ of G, it is NP-hard to compute $\mathrm{MMV}(G, \delta)$

Proof

- Reduction from Planar 3SAT

NP-complete (Lichtenstein (SICOMP '82))

Theorem

- For a given planar graph G and a drawing δ of G, it is NP-hard to compute $\mathrm{MMV}(G, \delta)$

Proof

- Reduction from Planar 3SAT

  NP-complete (Lichtenstein (SICOMP '82))

Note

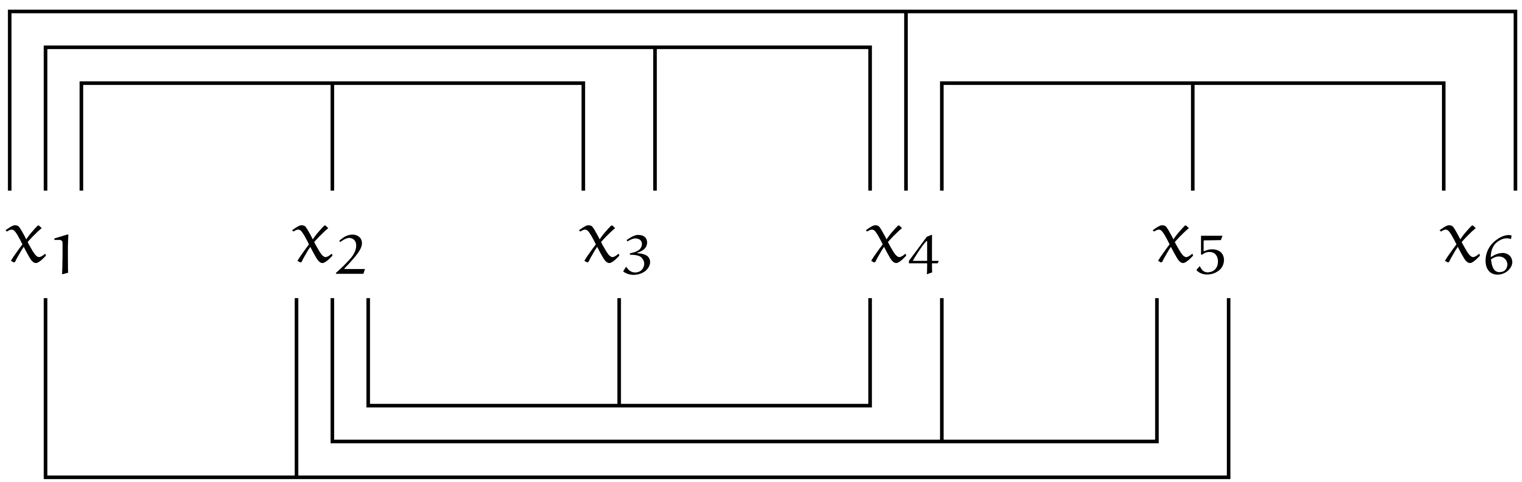
- More direct proof by Verbitsky (arXiv '07), but does not generalize to inapproximability

Planar 3SAT

Input : 3CNF formula φ with
a planar variable-clause graph

Question : Is φ satisfiable?

Note: such a graph can be embedded as below
(Knuth & Ragunathan (SIAMDM ’92))

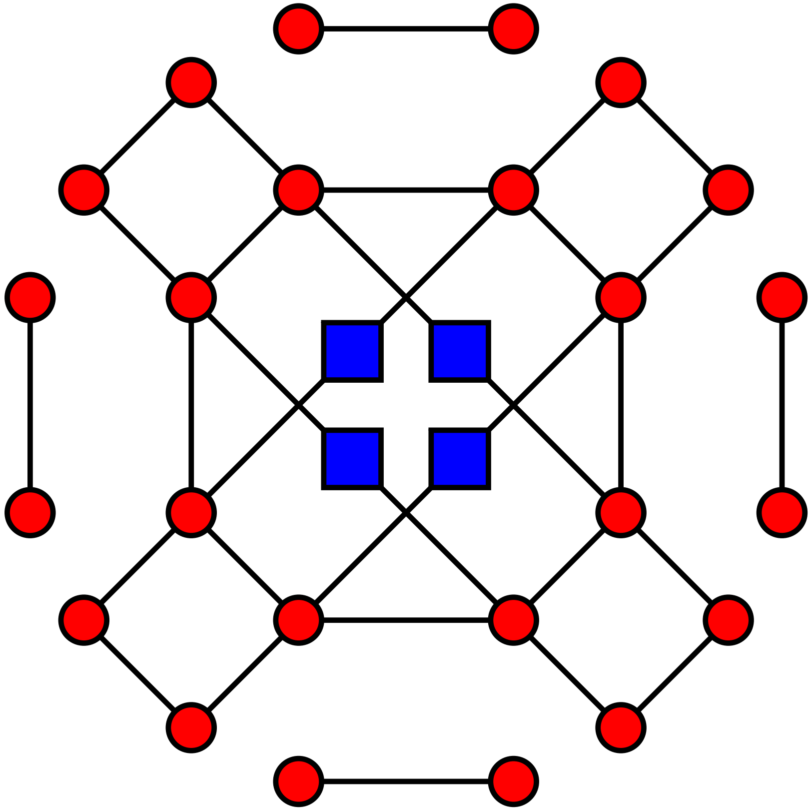


Basic strategy

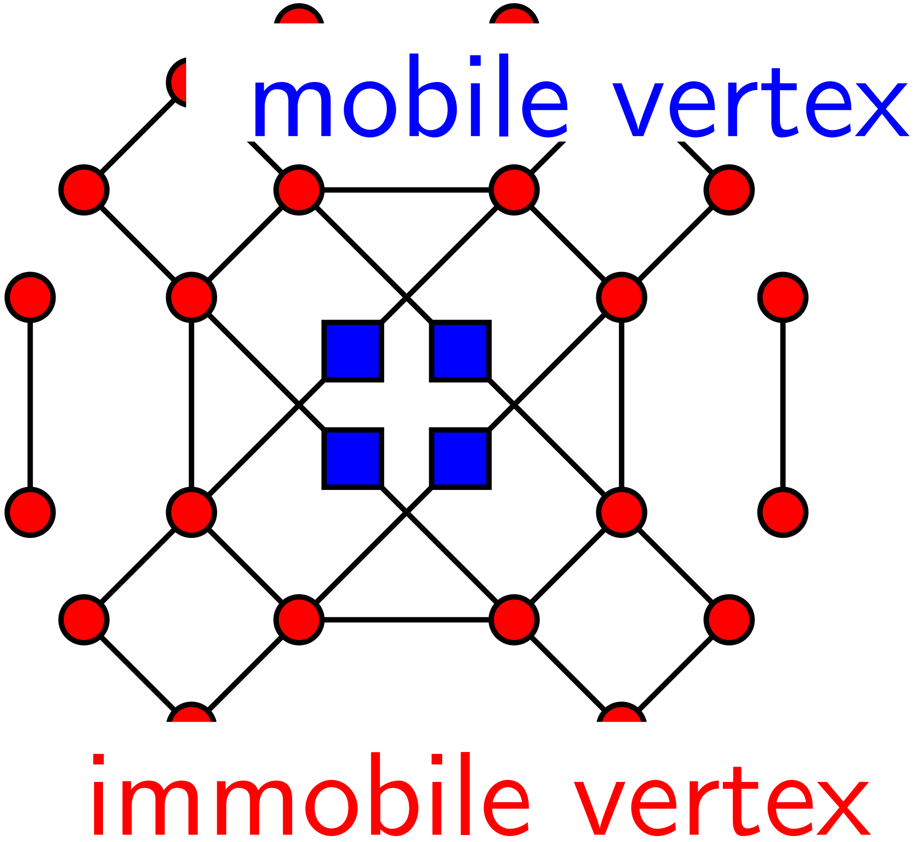
Given a planar 3CNF formula φ

- Construct a planar graph G_φ and a drawing δ_φ s.t.
- φ is satisfiable $\Leftrightarrow$
  δ_φ can be made plane by moving $\leq K$ vertices
- Vertices: two types
  - Mobile vertices (those that *may* move)
  - Immobile vertices (those that are *meant* not to move)
- Edges: each contributes to ≤ 1 crossing
- Gadgets: two types
  - Variable gadgets
  - Clause gadgets

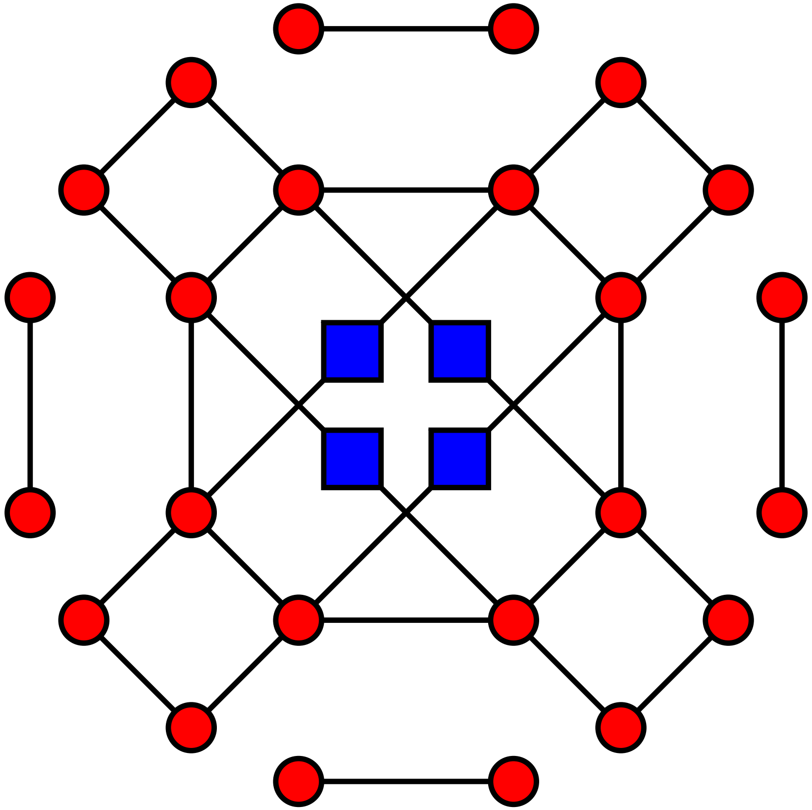
Gadget: block



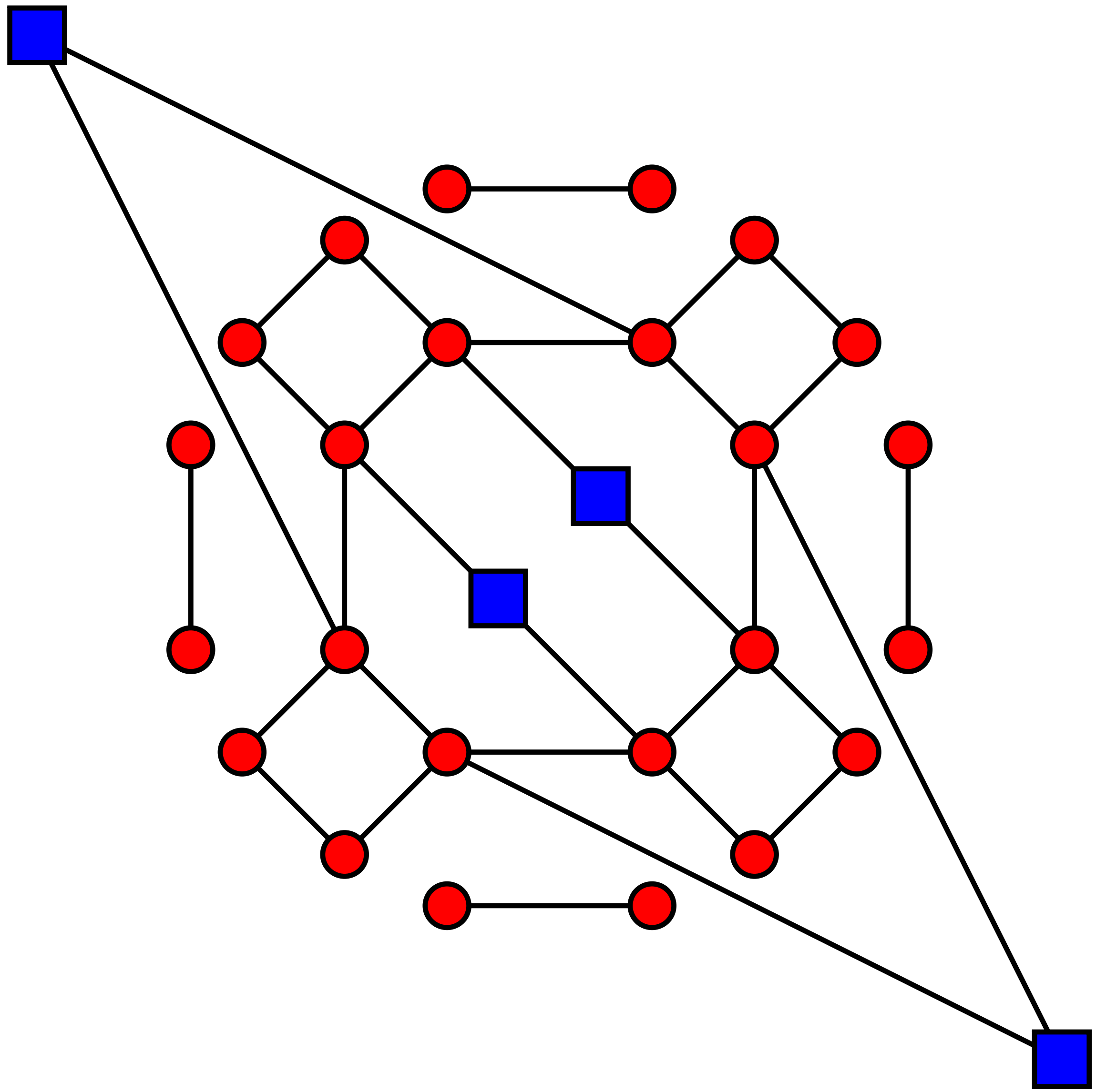
Gadget: block



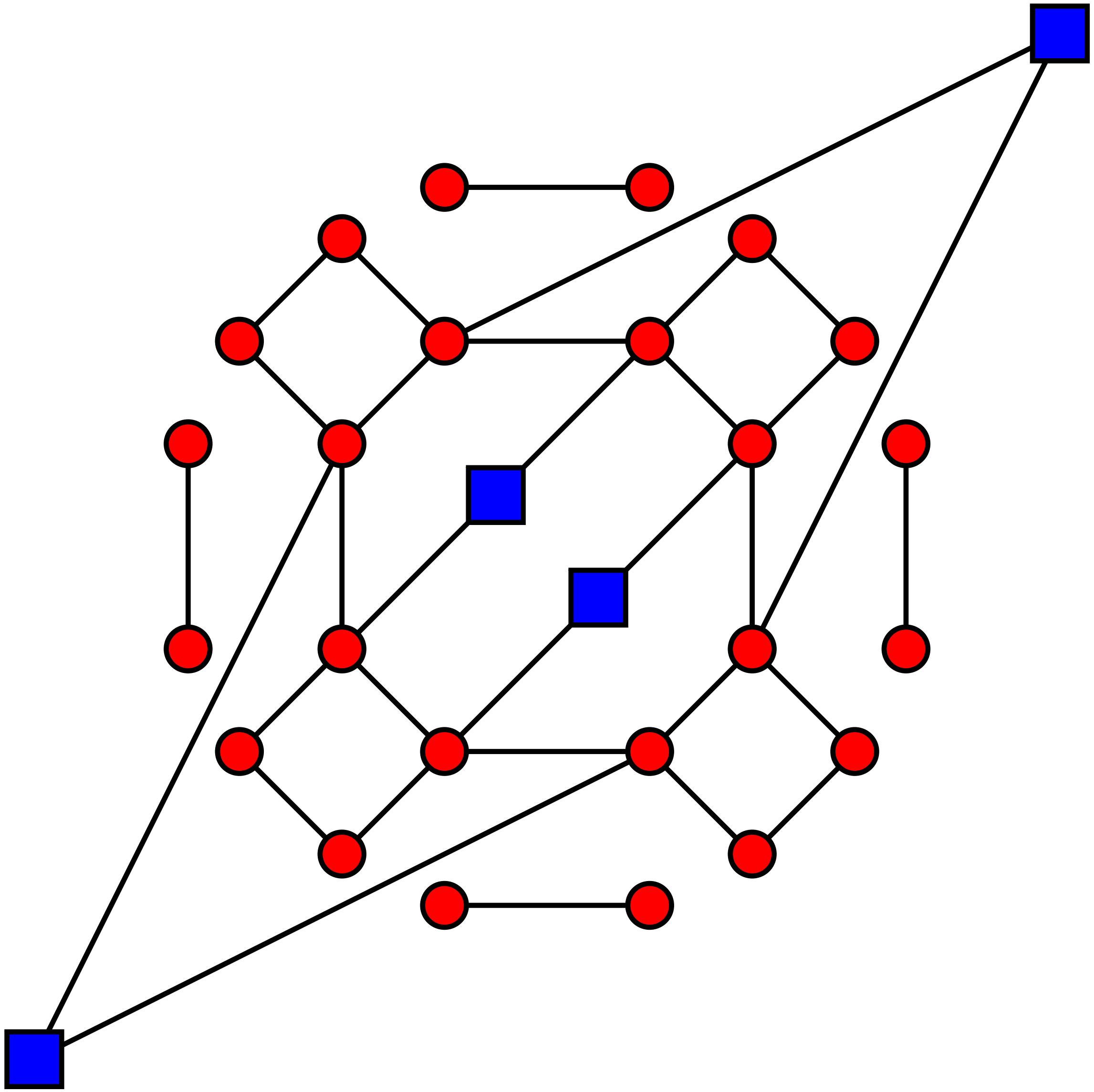
Gadget: block



Gadget: block

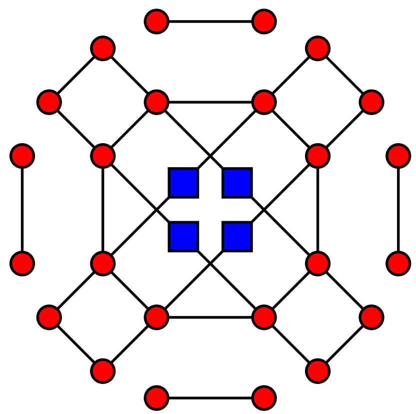


Gadget: block

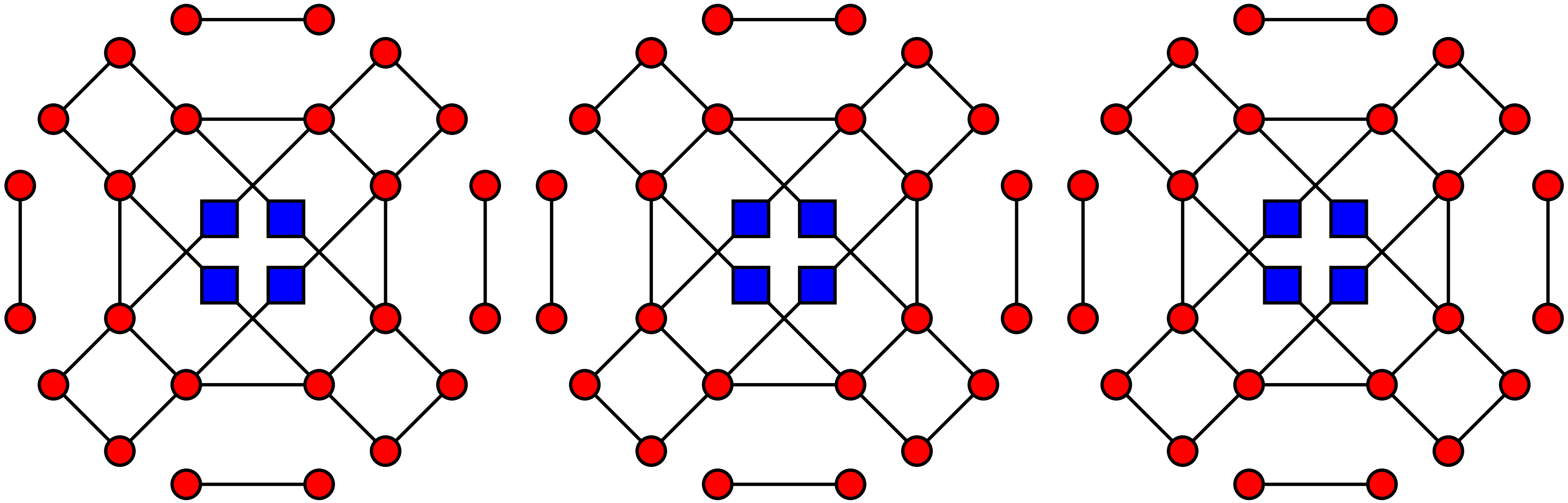


Important properties of gadgets

- Each mobile vertex has exactly two incident edges
- These two edges have crossings
- Mobile vertices are not adjacent
- Movement of a mobile vertex can get rid of two crossings

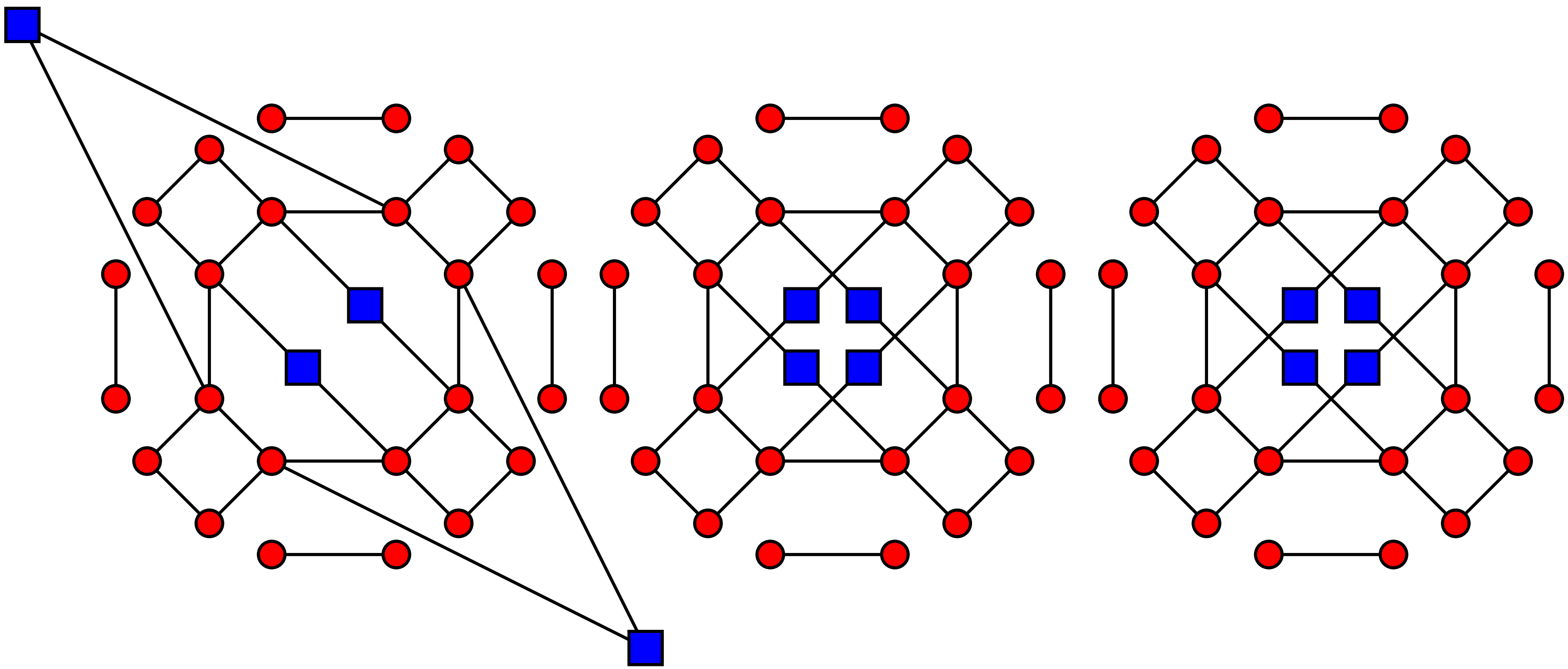


Gadget: variable



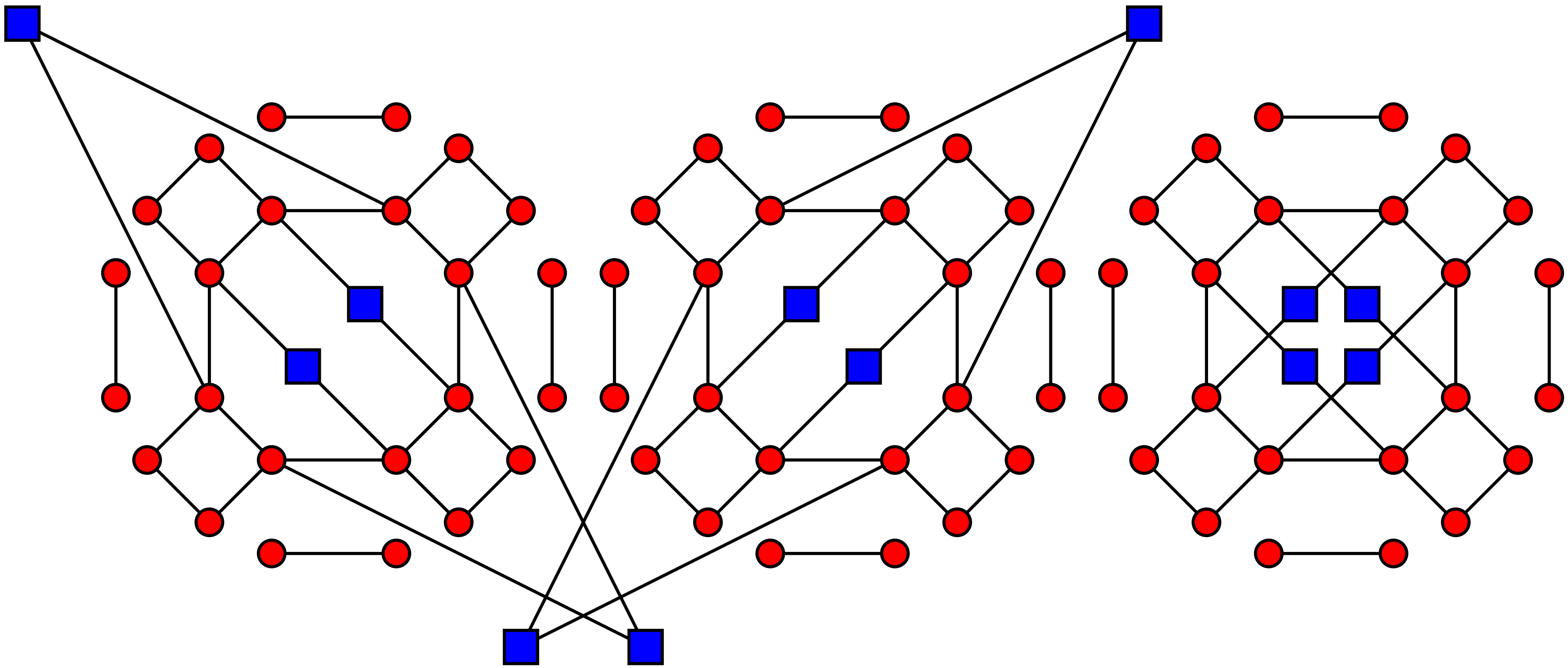
$\#$ blocks $=$ $\#$ occurences

Gadget: variable



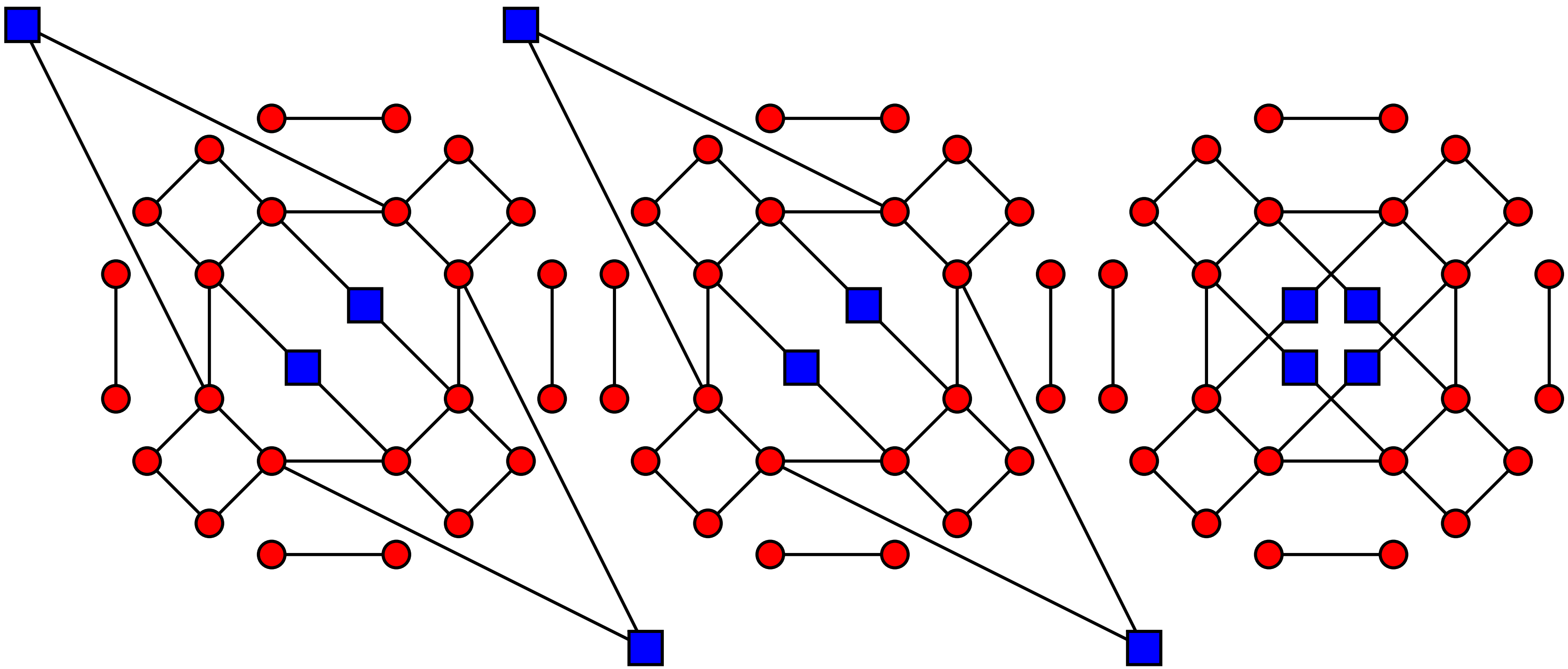
blocks = # occurences

Gadget: variable



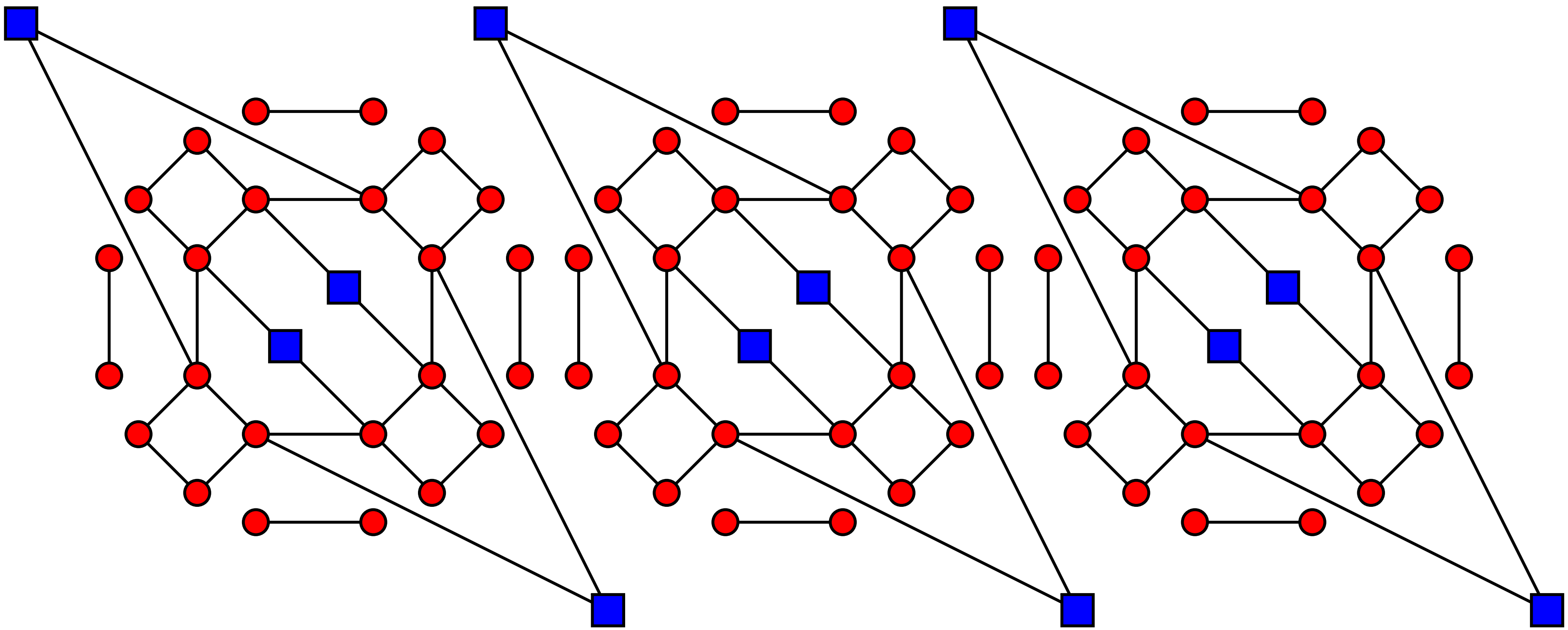
\# blocks = # occurences

Gadget: variable



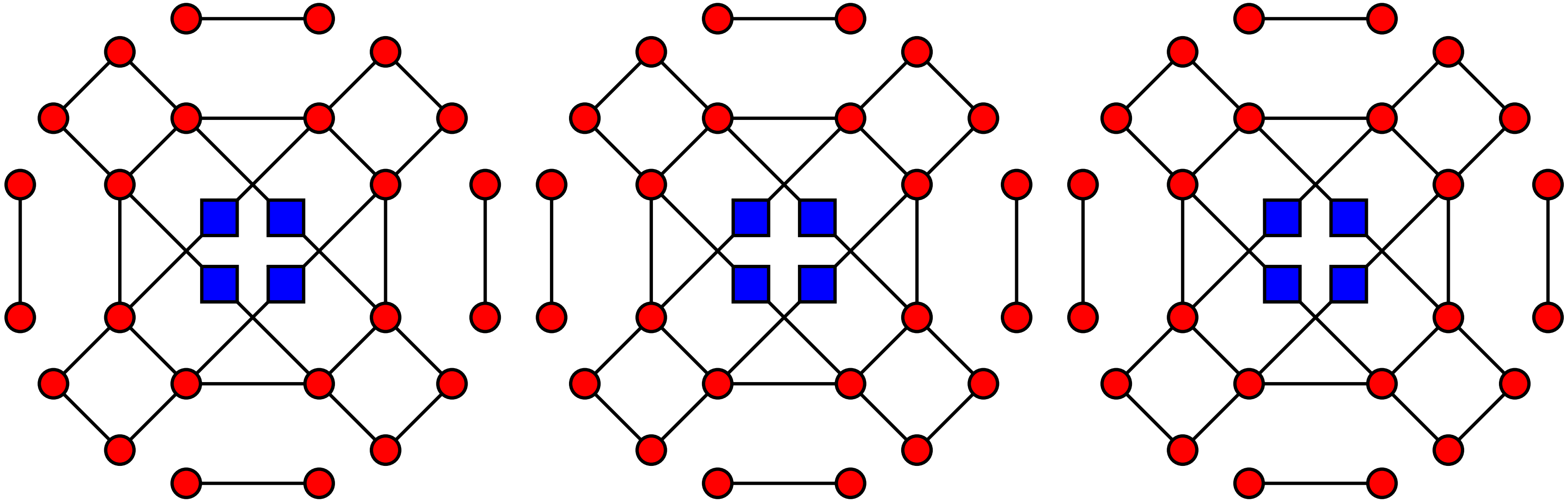
blocks = # occurences

Gadget: variable



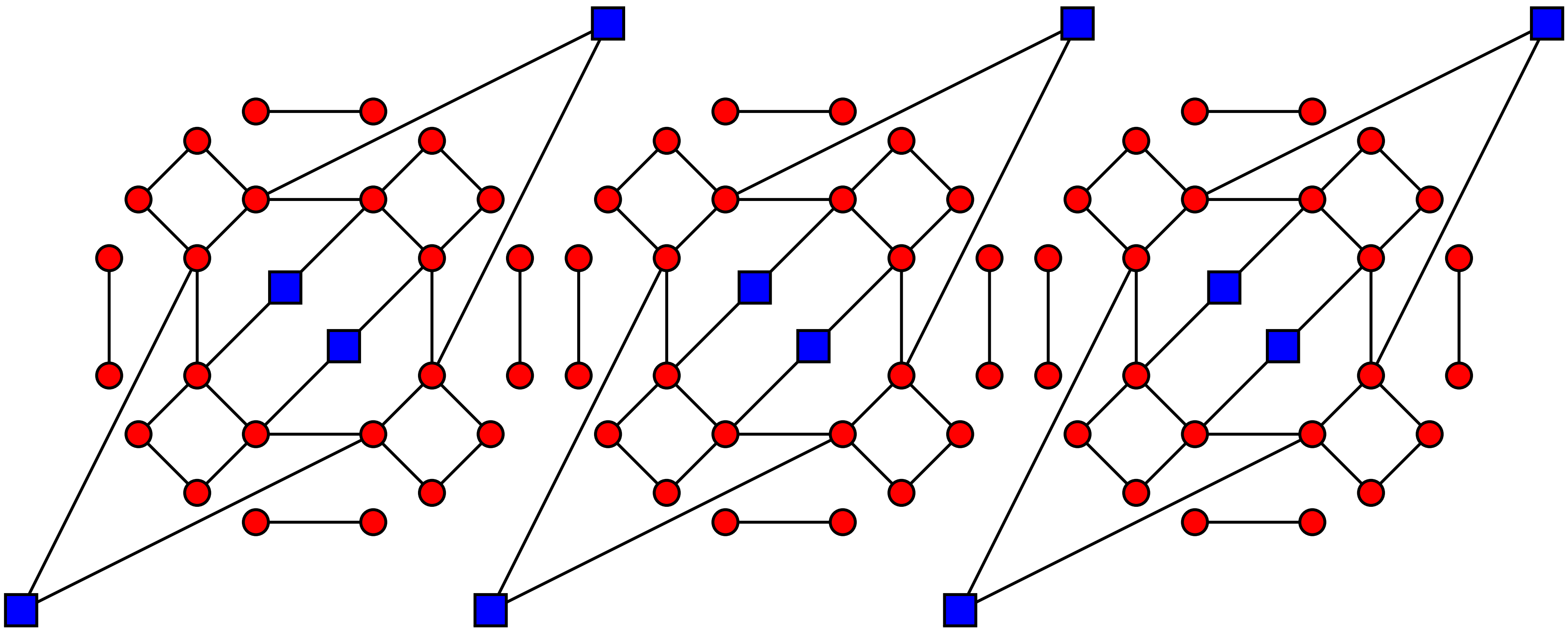
blocks = # occurences

Gadget: variable



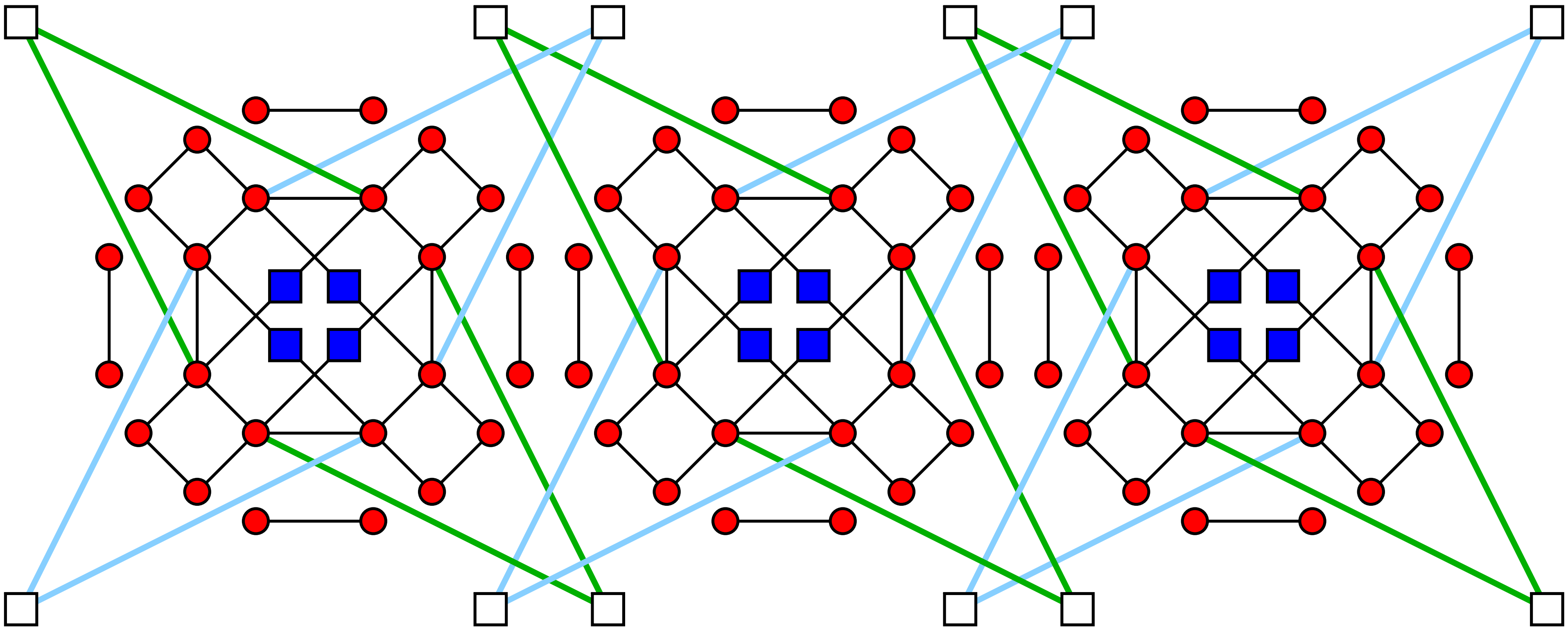
$\#$ blocks $= \#$ occurences

Gadget: variable



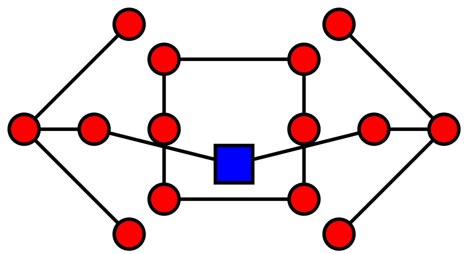
$\#$ blocks $=$ $\#$ occurences

Gadget: variable

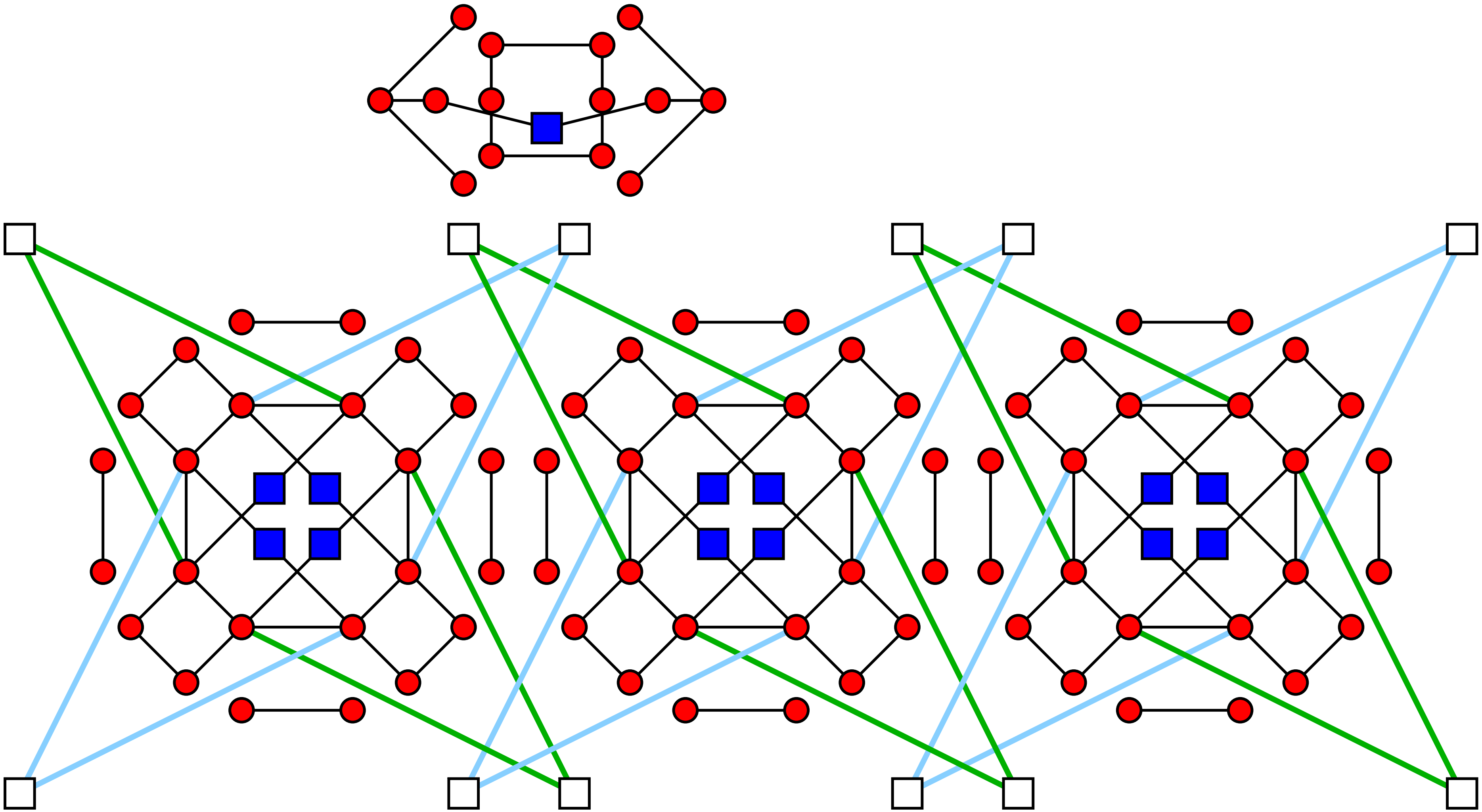


$\#$ blocks $=$ $\#$ occurences

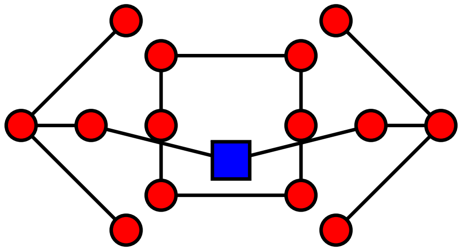
Gadgets: connection to clause gadgets



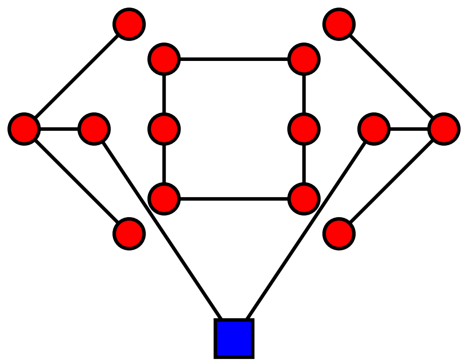
Gadgets: connection to clause gadgets



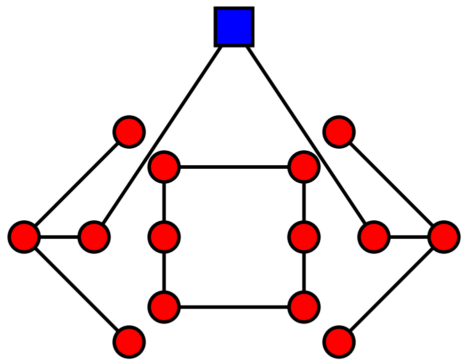
Gadgets: connection to clause gadgets



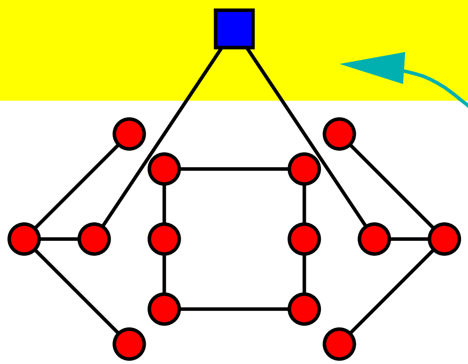
Gadgets: connection to clause gadgets



Gadgets: connection to clause gadgets



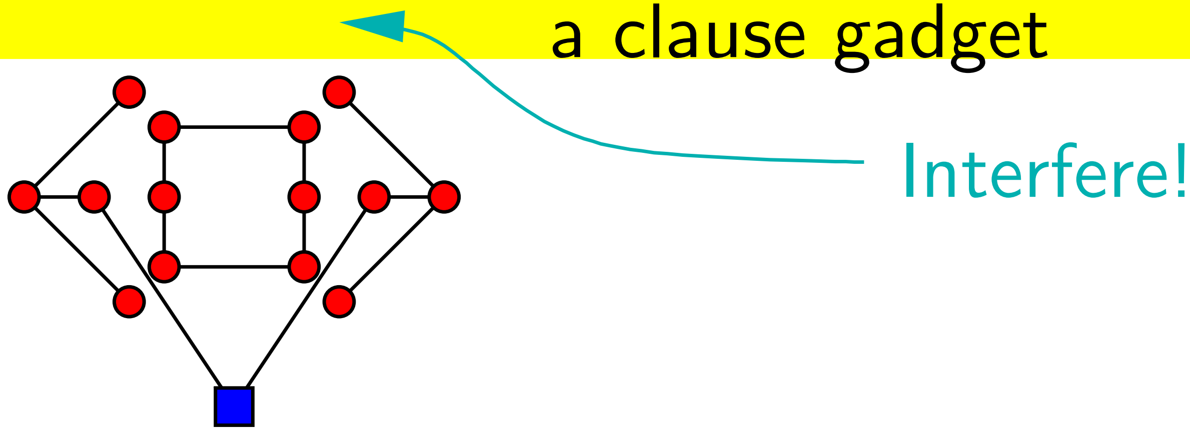
a clause gadget

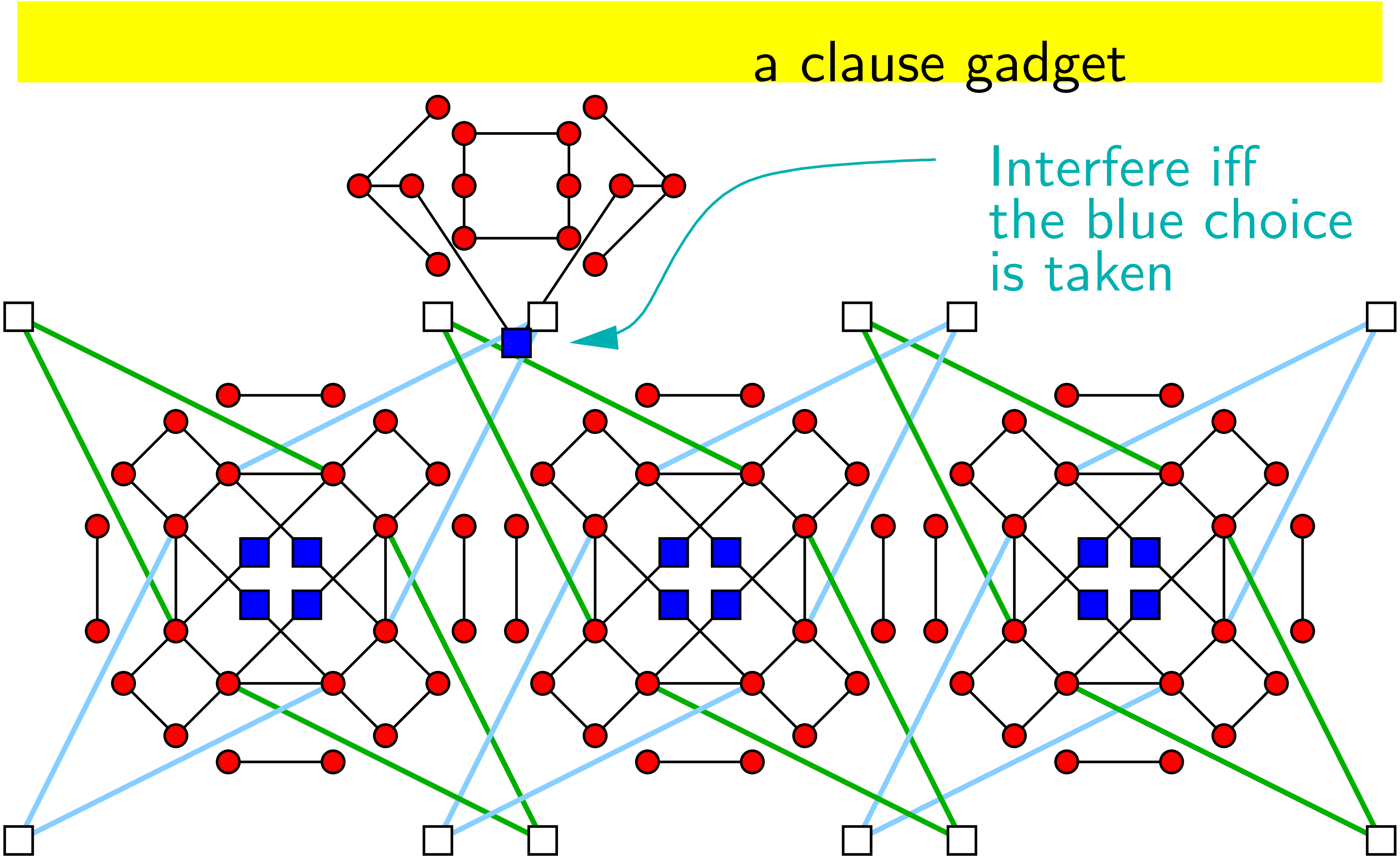


Interfere!

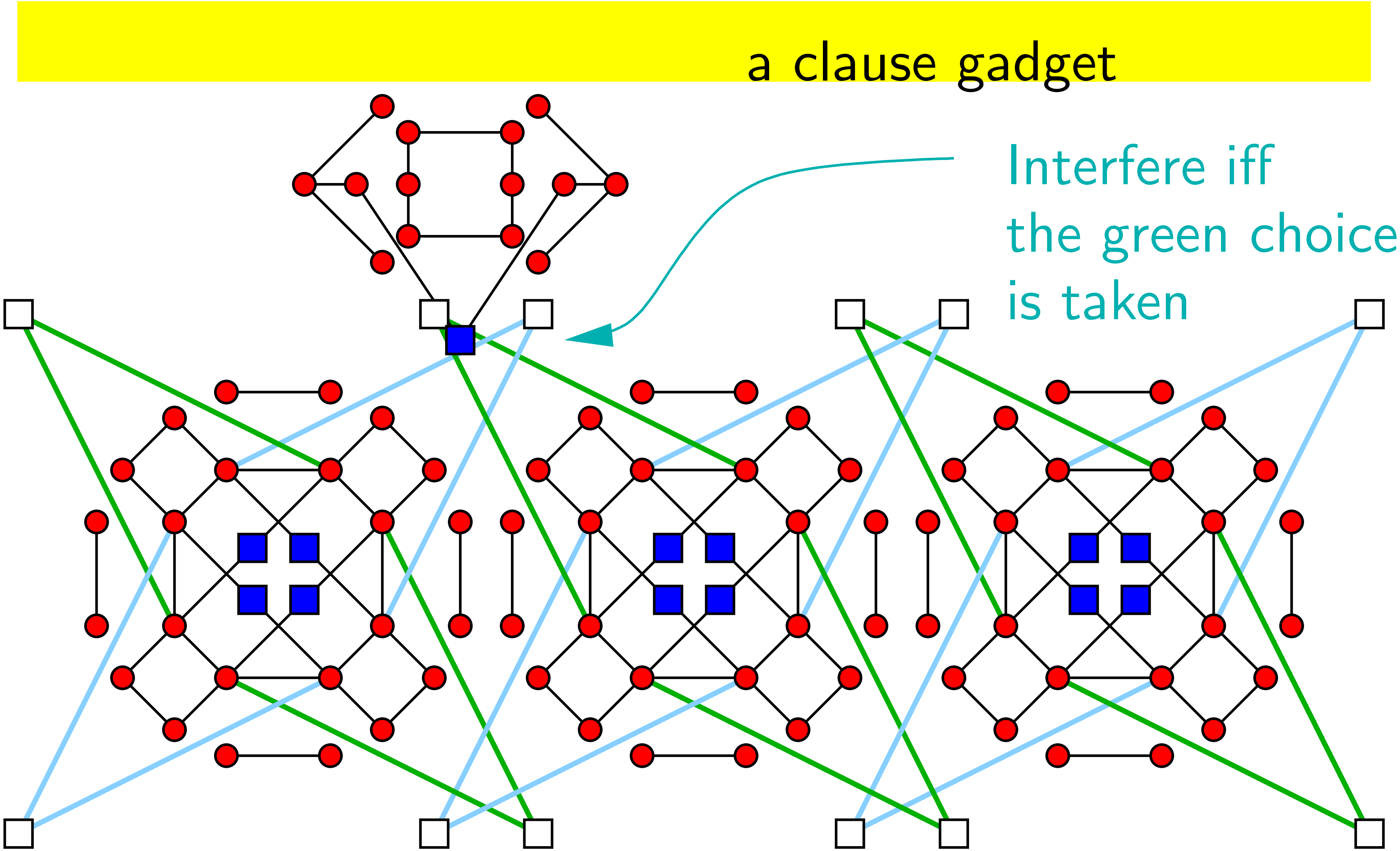
Gadgets: connection to clause gadgets

a clause gadget

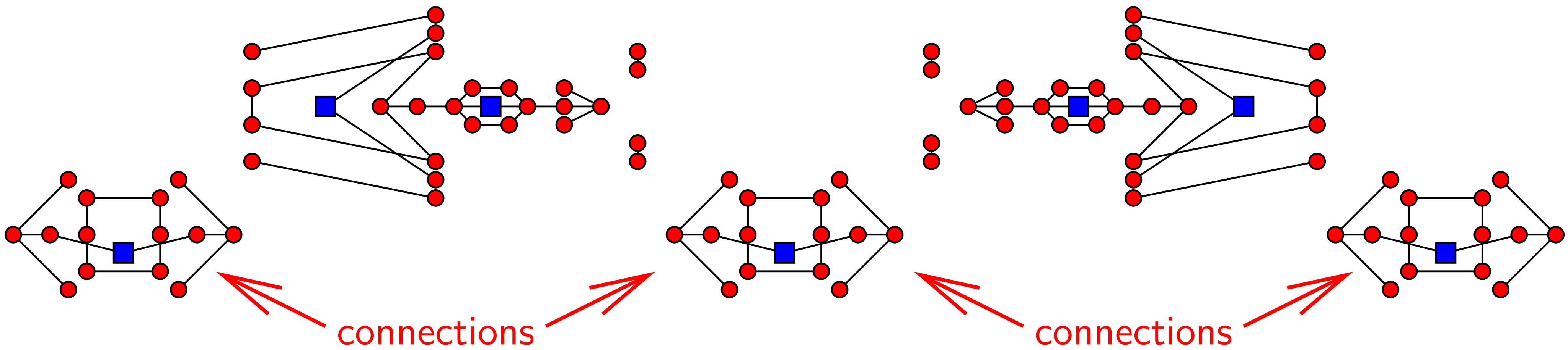



a clause gadget
Interfere iff
the blue choice
is taken

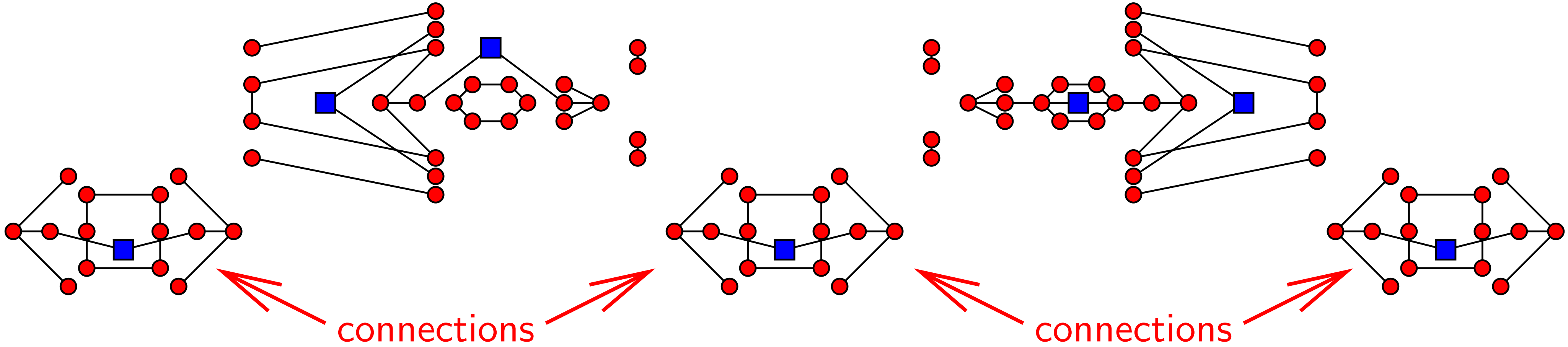
Gadgets: connection to clause gadgets



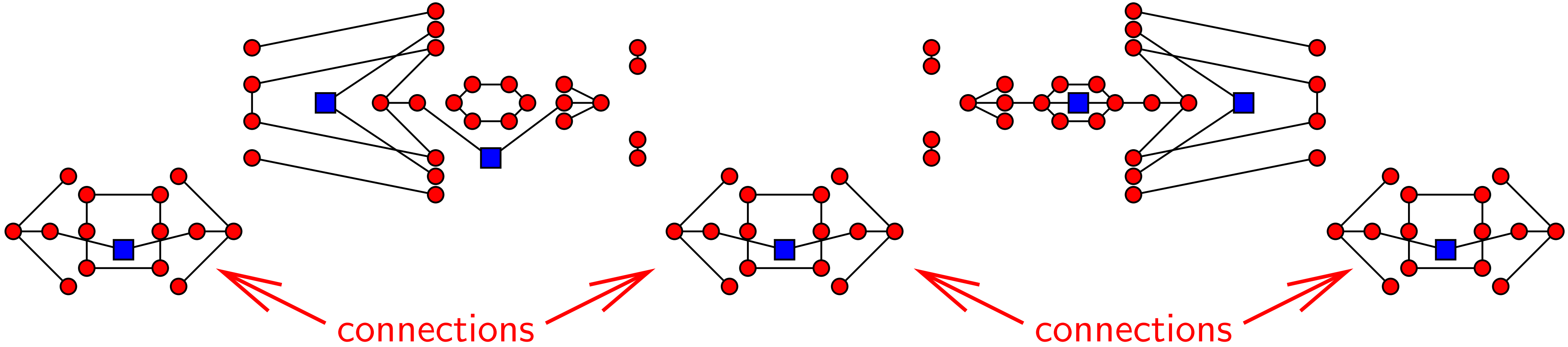
Gadgets: clause



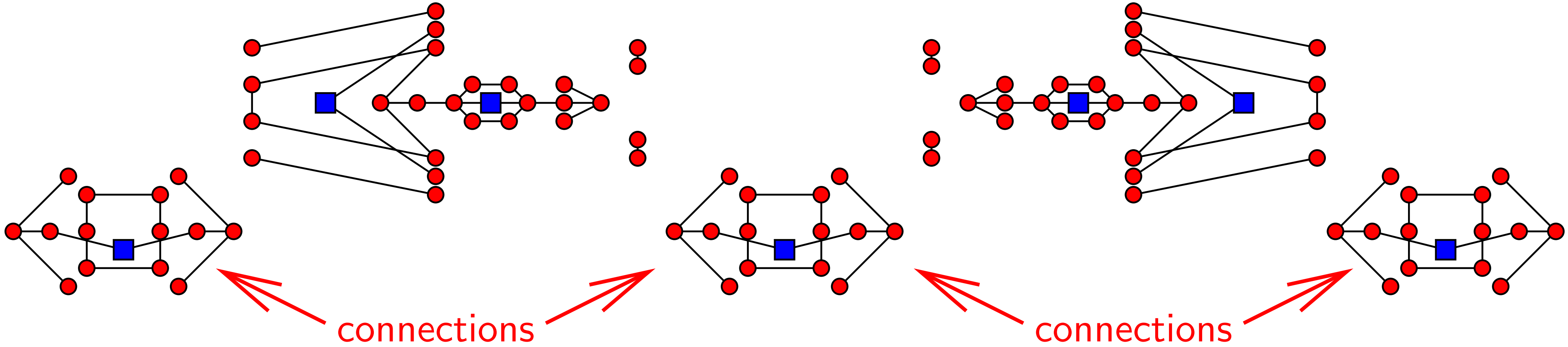
Gadgets: clause



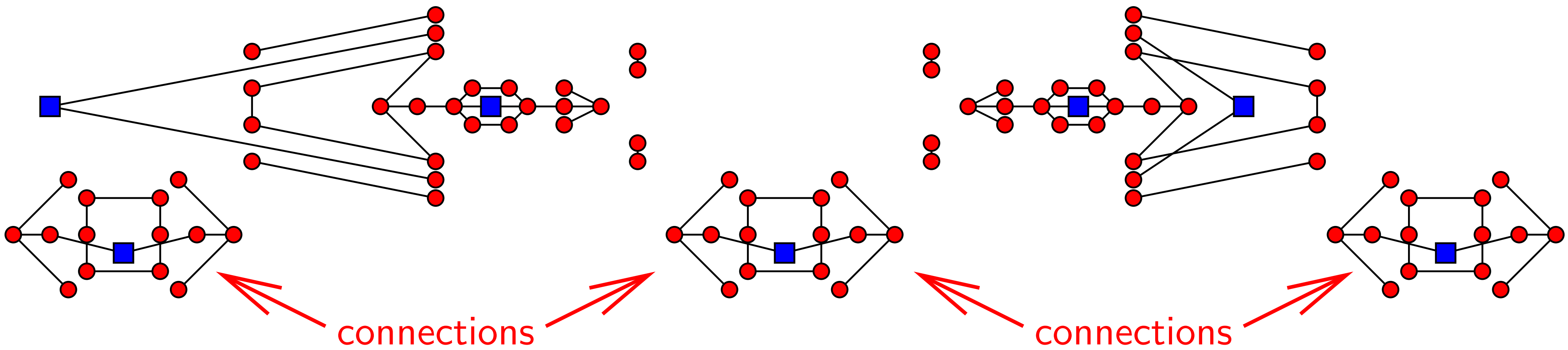
Gadgets: clause



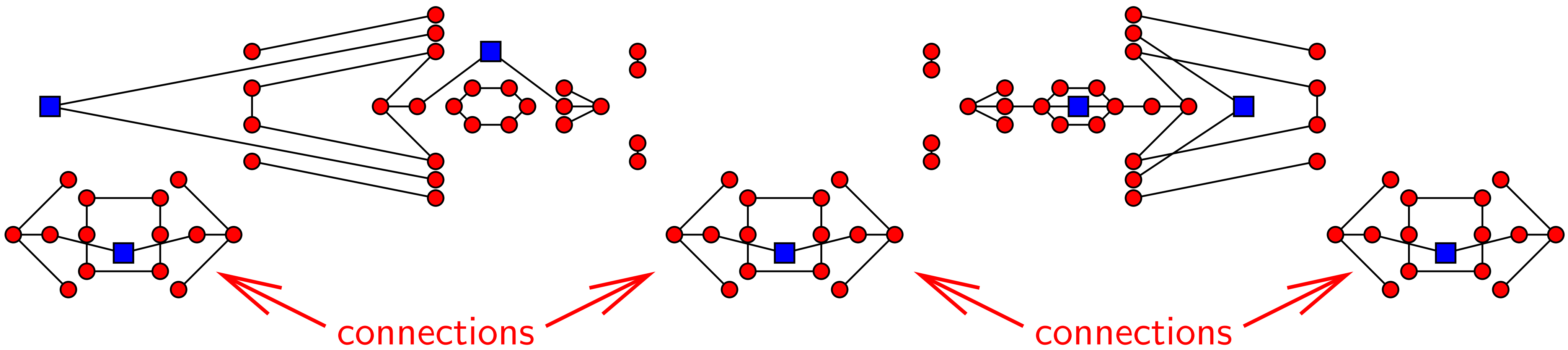
Gadgets: clause



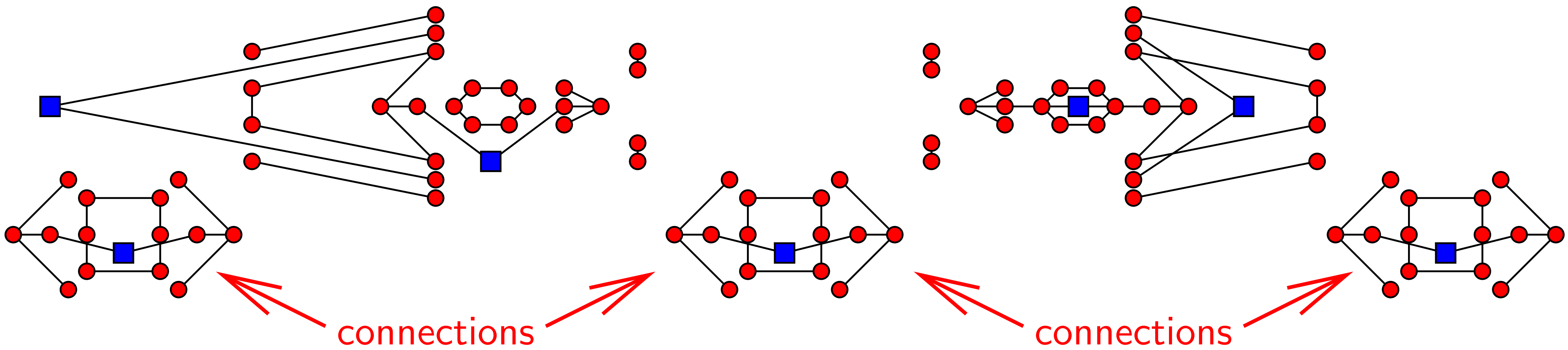
Gadgets: clause



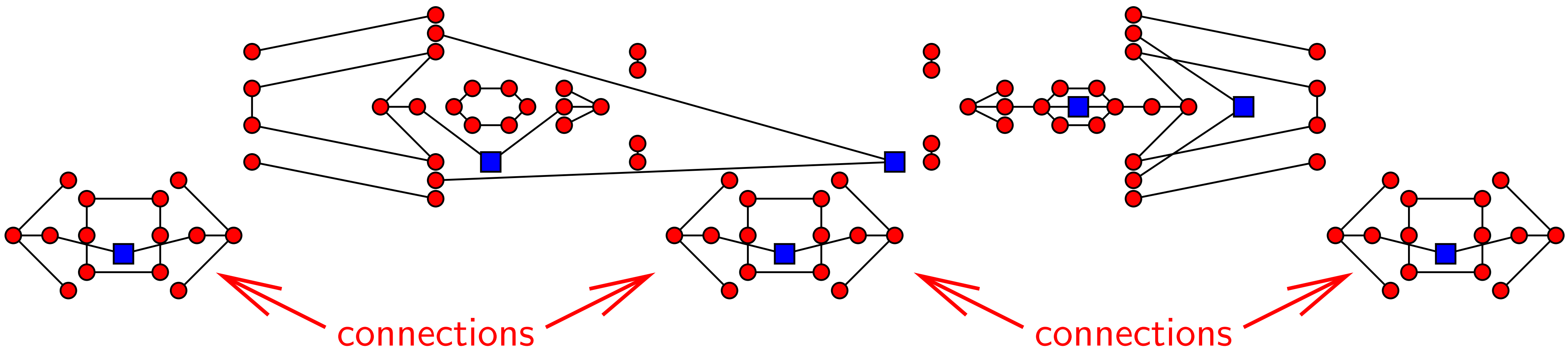
Gadgets: clause



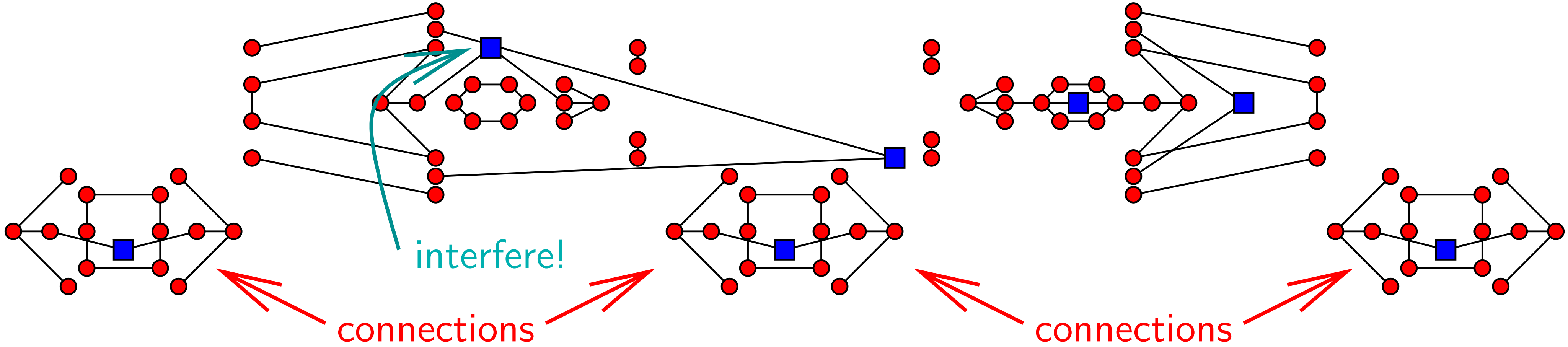
Gadgets: clause



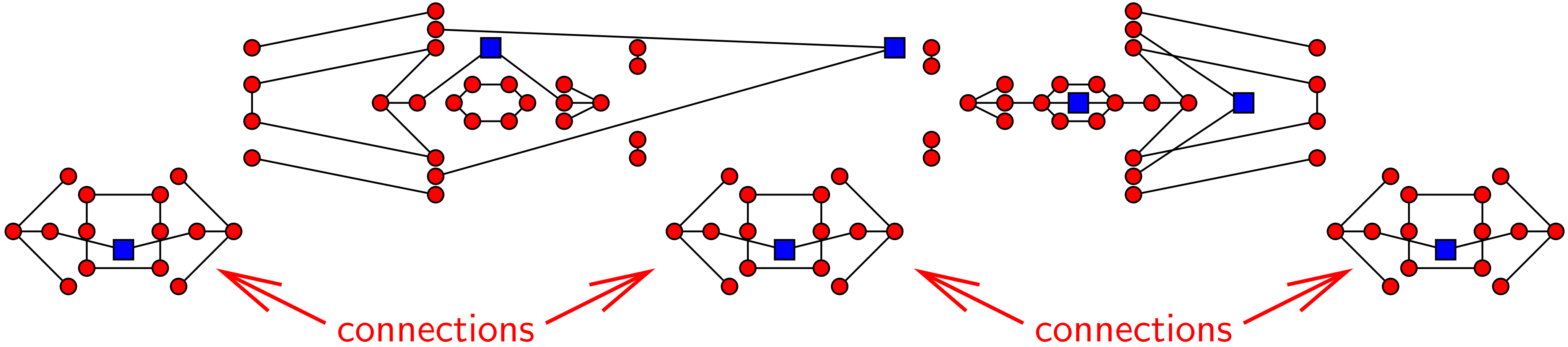
Gadgets: clause



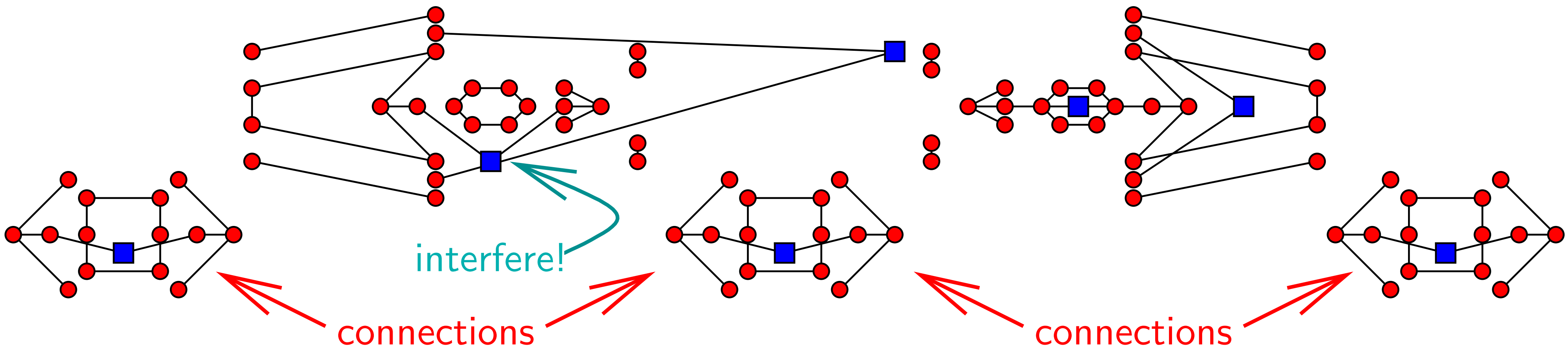
Gadgets: clause



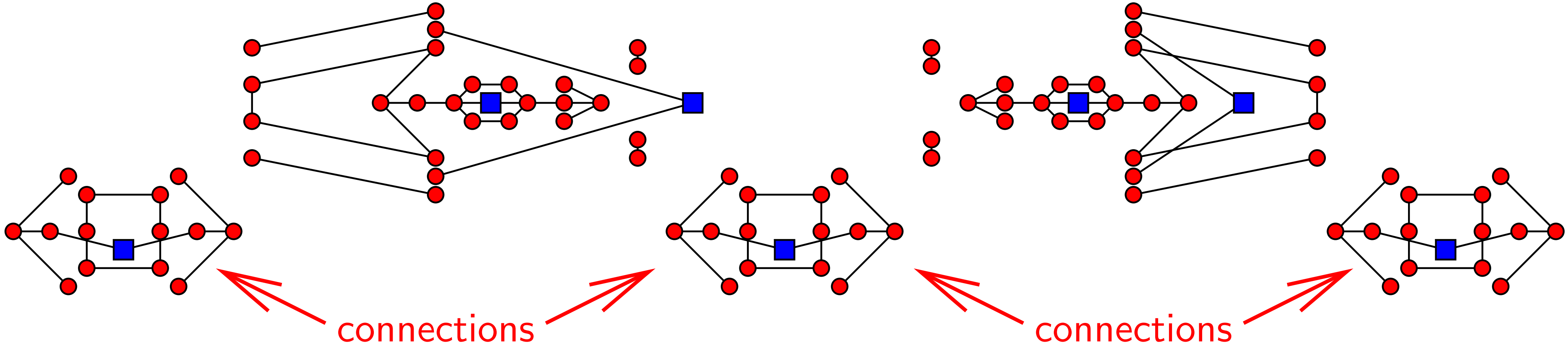
Gadgets: clause



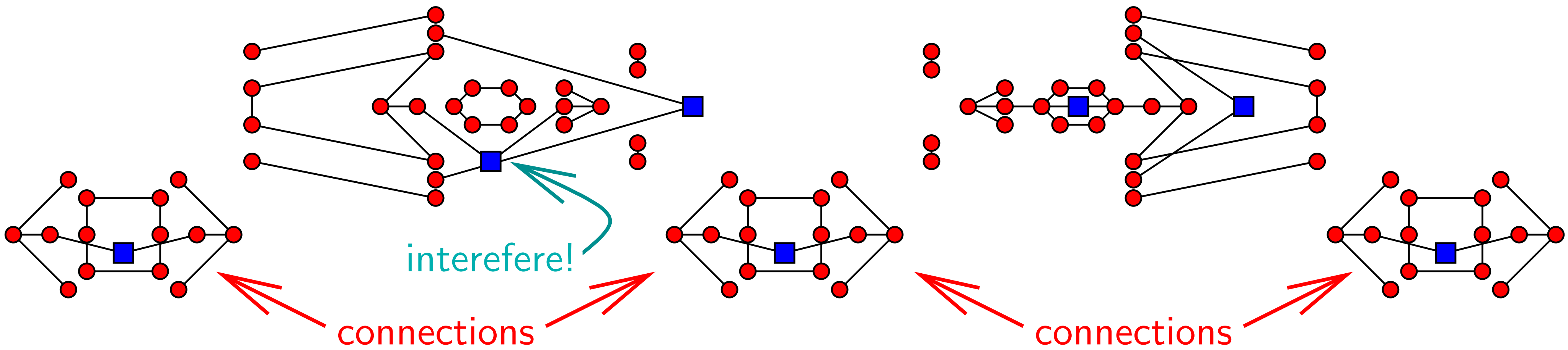
Gadgets: clause



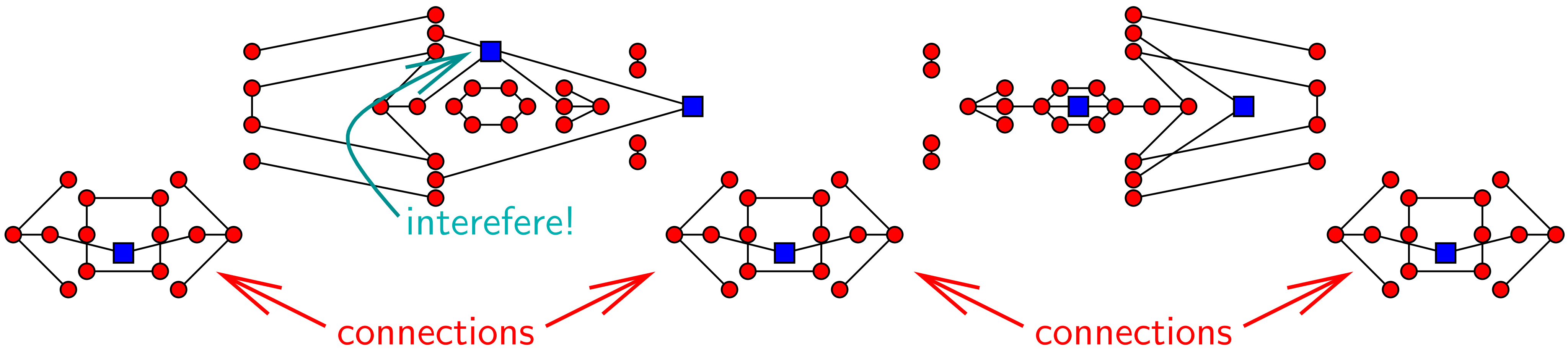
Gadgets: clause



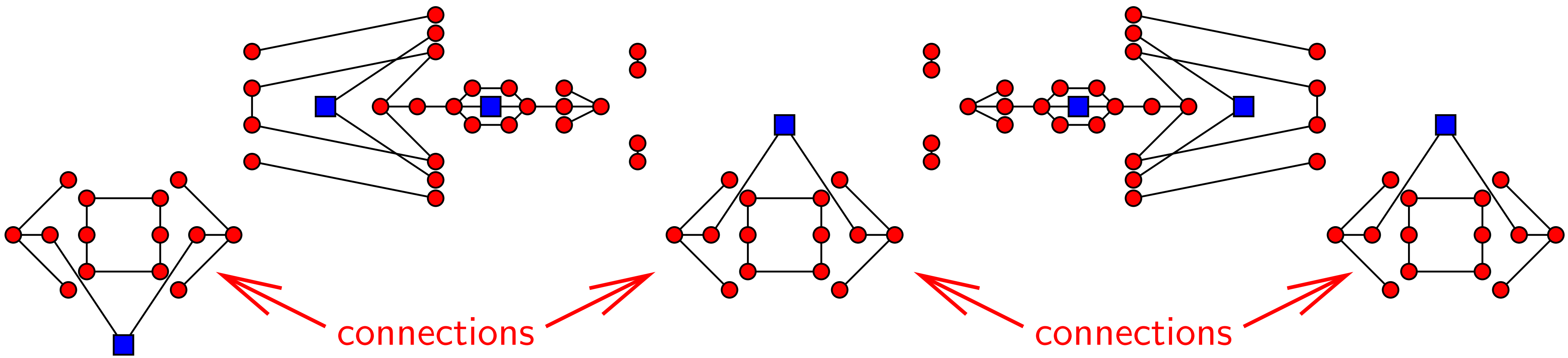
Gadgets: clause



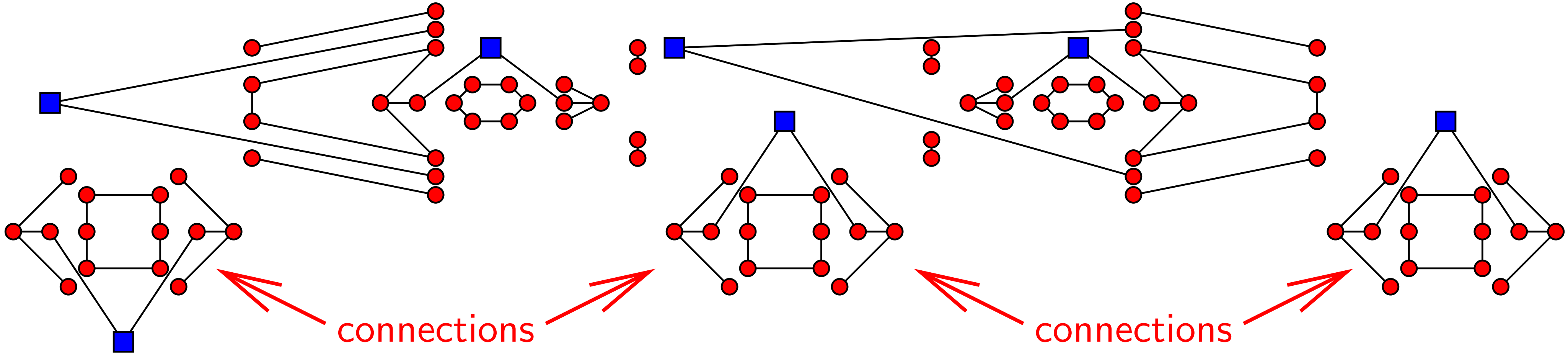
Gadgets: clause



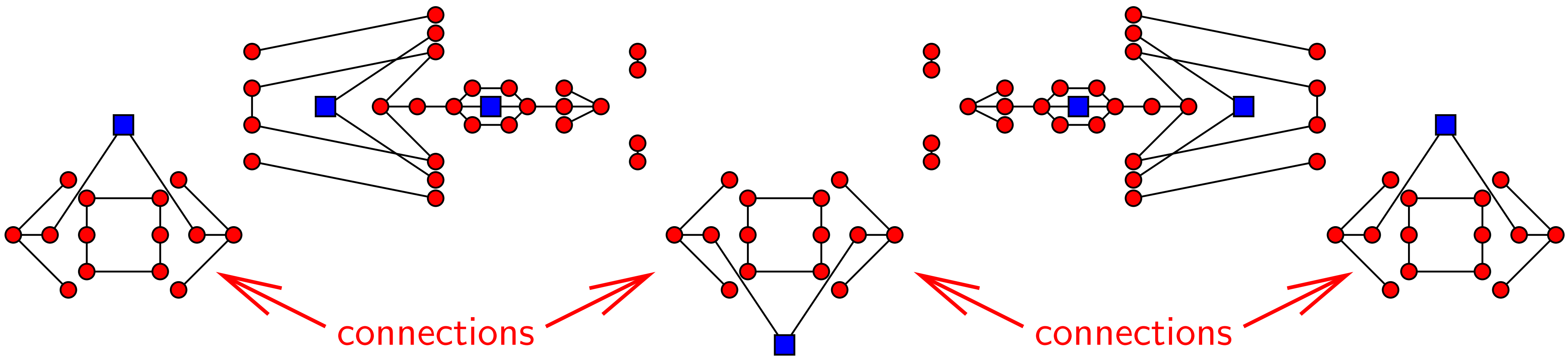
Gadgets: clause



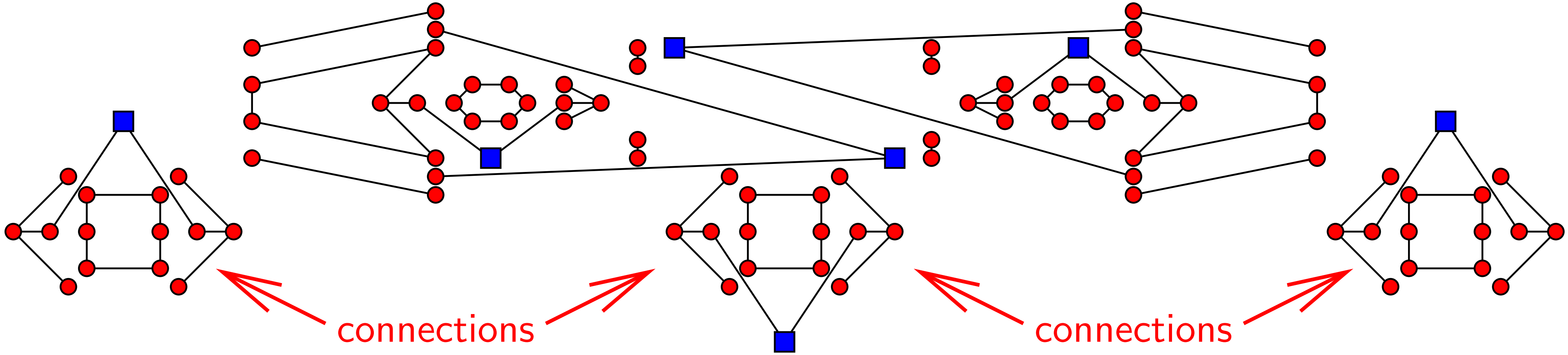
Gadgets: clause



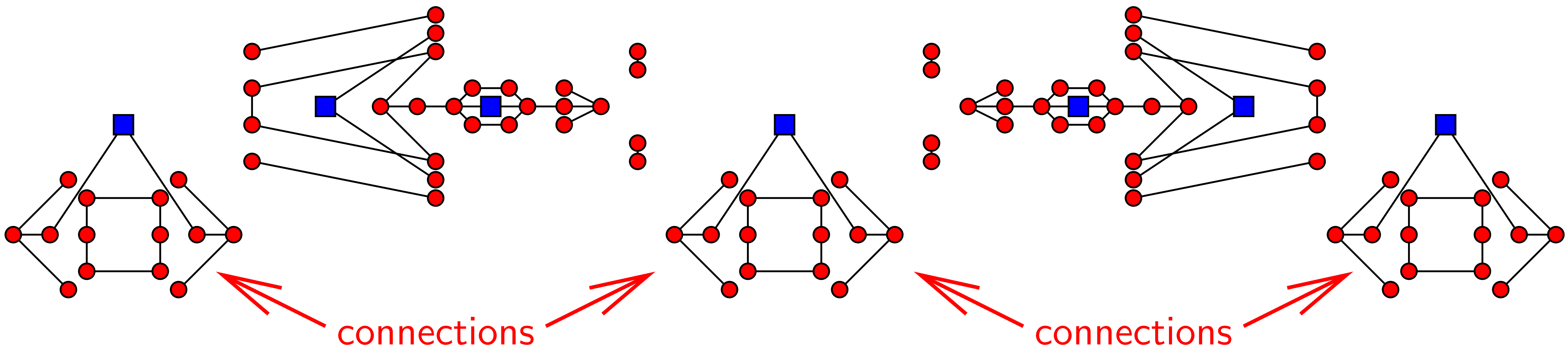
Gadgets: clause



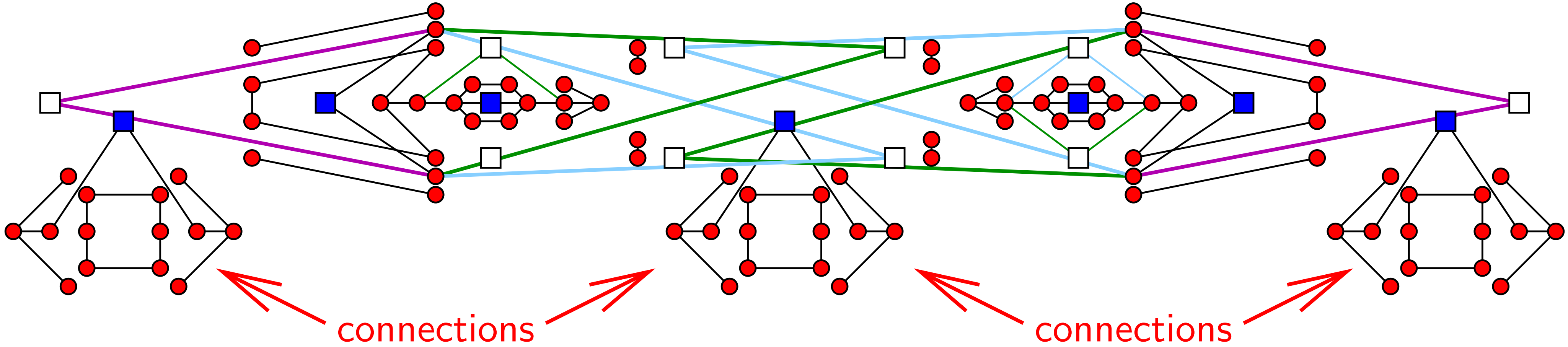
Gadgets: clause



Gadgets: clause



Gadgets: clause



Summing up...

- φ satisfiable $\Rightarrow$
  Movement of blue vertices suffices

  $\#$ moved vertices $= \#$ initial crossings$/2$

Summing up...

- φ satisfiable $\Rightarrow$
  Movement of blue vertices suffices

  $\#$ moved vertices $= \#$ initial crossings$/2$

- φ unsatisfiable $\Rightarrow$
  Movement of blue vertices doesn't suffice
  $\therefore$ at least one crossing requires both endpoints to move

  $\#$ moved vertices $\geq \#$ initial crossings$/2 + 1$

Summing up...

- φ satisfiable $\Rightarrow$
  Movement of blue vertices suffices

  $\#$ moved vertices $= \#$ initial crossings$/2$

- φ unsatisfiable $\Rightarrow$
  Movement of blue vertices doesn't suffice
  $\therefore$ at least one crossing requires both endpoints to move

  $\#$ moved vertices $\geq \#$ initial crossings$/2 + 1$

Conclude the reduction

Contents

- Problem statement (more formally)
- NP-hardness proof
- **Inapproximability (briefly)**
- Connection to the one-bend embeddability problem

Theorem

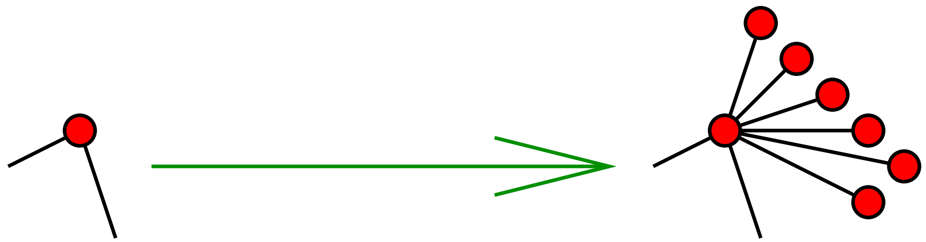
- For a given planar graph G and a drawing δ of G, it is NP-hard to approximate $1+\mathrm{MMV}(G,\delta)$ within a factor of $n^{1-\varepsilon}$ ($\forall$ fixed $\varepsilon \in (0,1]$)

Remark

- Since $\mathrm{MMV}(G,\delta)$ could be zero, we modify the objective value by adding one for the approximation to make sense.

Proof idea

- Use the same reduction as the NP-hardness proof
- Replace every immobile vertex with an immobile star



- Immobile stars give us a large gap
  ∴ Calculation shows our inapproximability

- Problem statement (more formally)
- NP-hardness proof
- Inapproximability (briefly)
- **Connection to the one-bend embeddability problem**

Def.: k-bend embedding

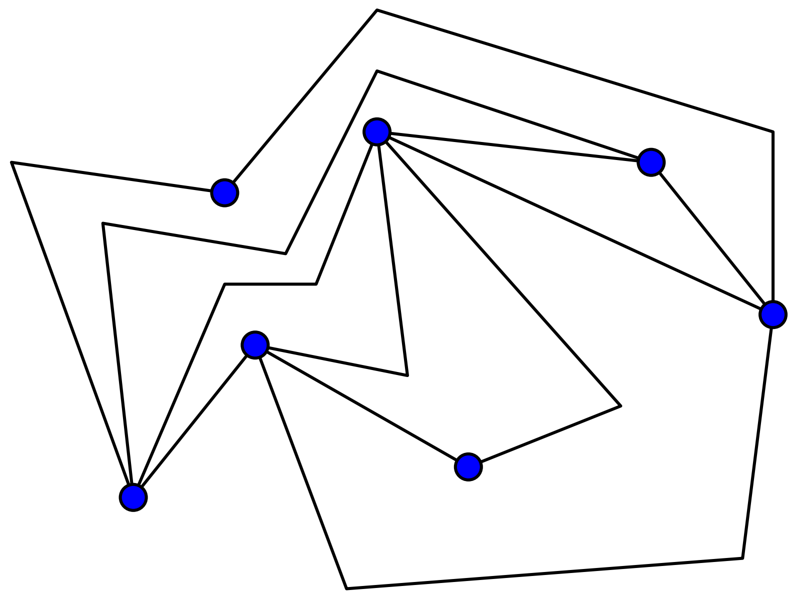
Setup: $G = (V, E)$ a planar graph

Def: A **k-bend embedding** of G is an embedding of G into a plane s.t. every edge is drawn as a non-crossing polygonal chain with k bends

Def.: k-bend embedding

Setup: $G = (V, E)$ a planar graph

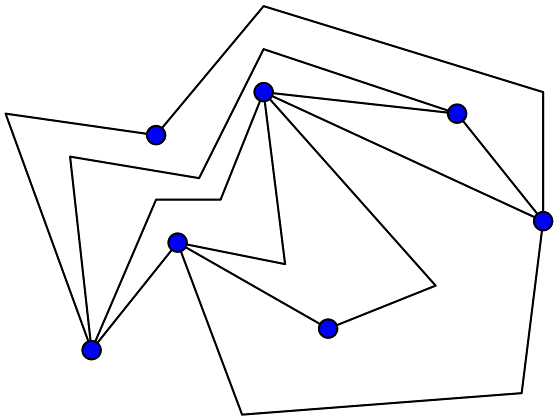
Def: A **k-bend embedding** of G is
an embedding of G into a plane s.t.
every edge is drawn as a non-crossing polygonal chain with k bends



Def.: k-bend point-set embeddability

Setup: $G = (V, E)$ a planar graph

Def: G is **k-bend (point-set) embeddable** if
$\forall\ S \subset \mathbb{R}^2$ with $|S| = |V|$
$\exists$ a bijection $\delta \colon V \to S$ s.t.
G can be k-bend embedded while
each $v \in V$ is placed at $\delta(v) \in S$



Known results

(Kaufmann & Wiese (GD ’99, JGAA ’02))

- G 4-connected planar ⇒ G 1-bend embeddable
- G planar ⇒ G 2-bend embeddable
- It is NP-complete to decide if
  for a given planar $G = (V, E)$ and a point set S
  $\exists$ a bijection $\delta \colon V \to S$
  that makes it possible to 1-bend embed G on S

New result

Theorem

- For a given planar graph $G = (V, E)$, a point set S and a bijection $\delta\colon V \to S$
  it is NP-hard to decide if
  δ makes it possible to 1-bend embed G on S

Reminder

Kaufmann–Wiese '02

- For a given planar graph $G = (V, E)$ and a point set S
  it is NP-hard to decide if
  $\exists$ a bijection $\delta\colon V \to S$
  that makes it possible to 1-bend embed G on S

Proof idea

- Use the same reduction as the NP-hardness of MMV
- But contract the mobile vertices

Proof idea

- Use the same reduction as the NP-hardness of MMV
- But contract the mobile vertices

Remark: a similar inapproximability holds

Theorem

- For a given planar graph $G = (V, E)$, a point set S and a bijection $\delta \colon V \to S$
  it is NP-hard to approximate min # total bends (+1)
  when G is embedded on S with the correspondence δ
  within a factor of $n^{1-\varepsilon}$ ($\forall$ fixed $\varepsilon \in (0, 1]$)

- Problem statement (more formally)
- NP-hardness proof
- Inapproximability (briefly)
- Connection to the one-bend embeddability problem
- **Concluding remarks**

max # of vertices that we can keep fixed

	Lower Bound	Upper Bound
Cycles	$\lfloor\sqrt{n}\rfloor$	$O((n \log n)^{2/3})$
Trees	$\lfloor\sqrt{n}/3\rfloor$	$\lceil n/3\rceil + 4$
General	3	$\lceil\sqrt{n-2}\rceil+1$

Pach & Tardos (GD '01, DCG '02)

max # of vertices that we can keep fixed

	Lower Bound	Upper Bound
Cycles	$\lfloor\sqrt{n}\rfloor$	$O((n\log n)^{2/3})$
Trees	$\lfloor\sqrt{n/3}\rfloor$	$\lceil n/3\rceil + 4$
Outerplanar	$\sqrt{n-1/3}$	$2\sqrt{n-1}+1$
General	3	$\lceil\sqrt{n-2}\rceil+1$
	$\Omega(\sqrt{\log n/\log\log n})$	

Pach & Tardos (GD '01, DCG '02)

Spillner & Wolff (arXiv Sept '07)

Summary and open problems: computational

Results

- Minimizing the number of moved vertices is
  - NP-hard to compute precisely
  - NP-hard to compute approximately with factor $n^{1-\varepsilon}$

Summary and open problems: computational

Results

- Minimizing the number of moved vertices is
  - NP-hard to compute precisely
  - NP-hard to compute approximately with factor $n^{1-\varepsilon}$

Open Problems

- Minimizing the number of moved vertices is
  - hard in the parameterized sense?
- Maximizing the number of kept vertices is
  - NP-hard to compute approximately??
  - hard in the parameterized sense?
- How about when we restrict a graph class?

Summary and open problems: computational

Results

- Minimizing the number of moved vertices is
  - NP-hard to compute precisely
  - NP-hard to compute approximately with factor $n^{1-\varepsilon}$

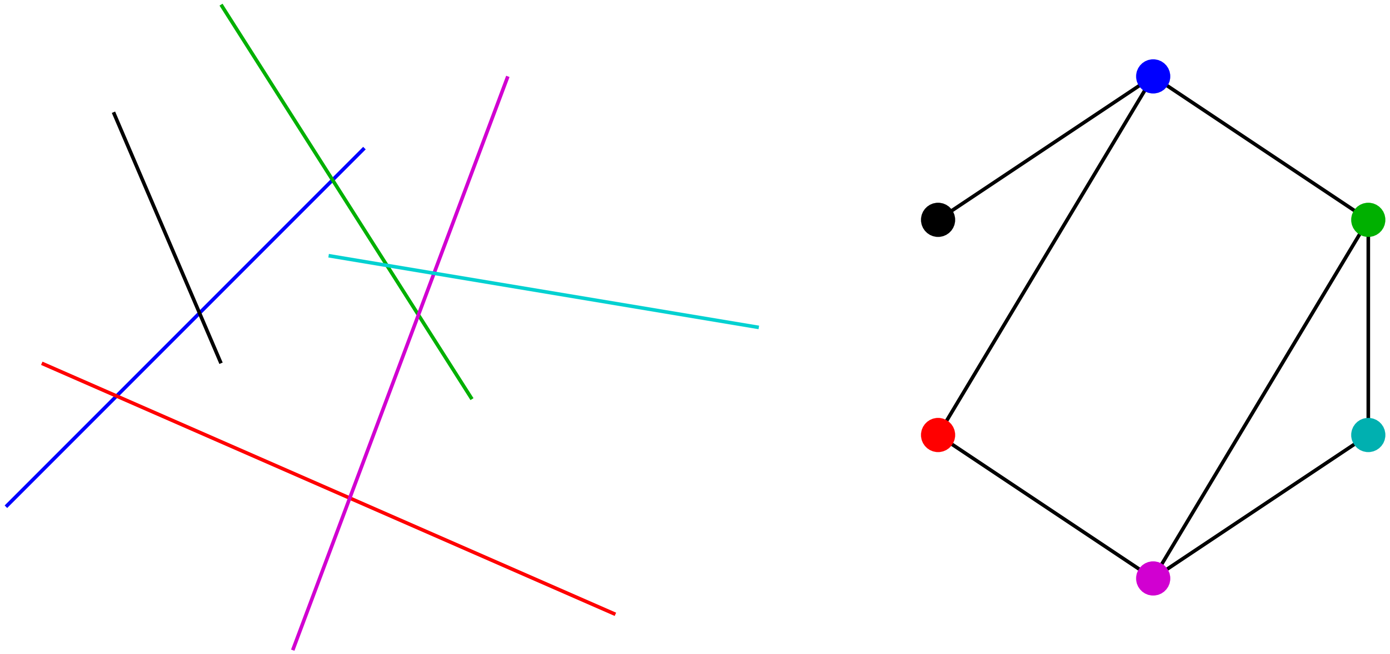
Open Problems

- Minimizing the number of moved vertices is
  - hard in the parameterized sense?
- Maximizing the number of kept vertices is
  - NP-hard to compute approximately??
  - hard in the parameterized sense?
- How about when we restrict a graph class?

[End of Talk]

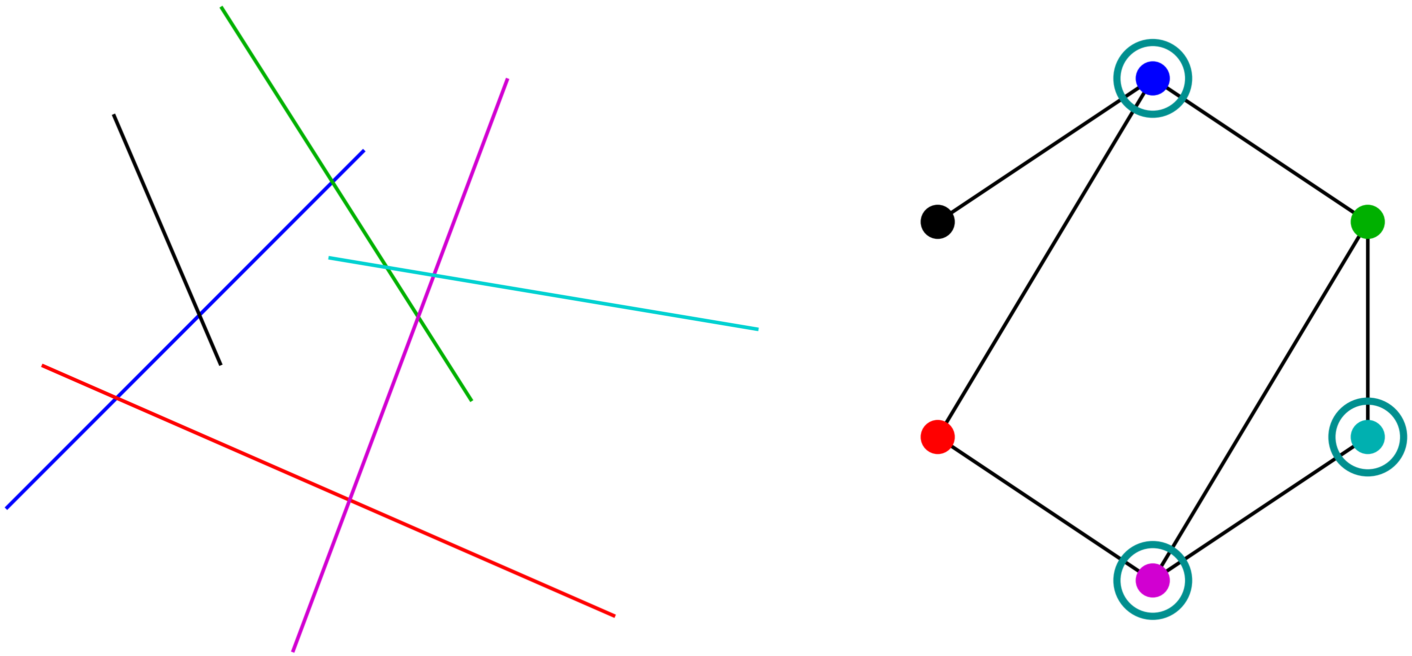
Supplementary slides

Min Vertex Cover of line segments

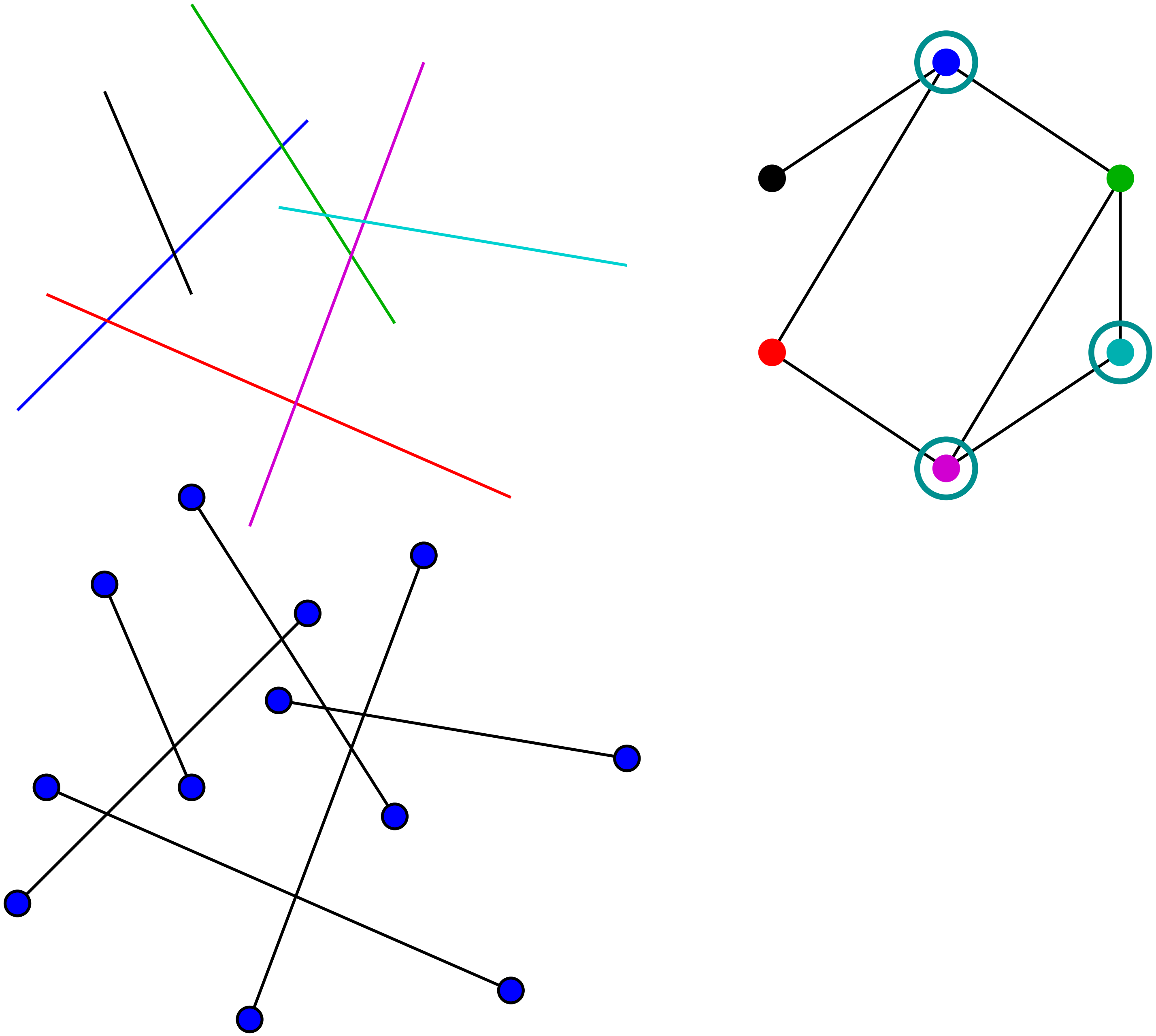


Vertex cover = set of vertices that doesn't miss any edge.

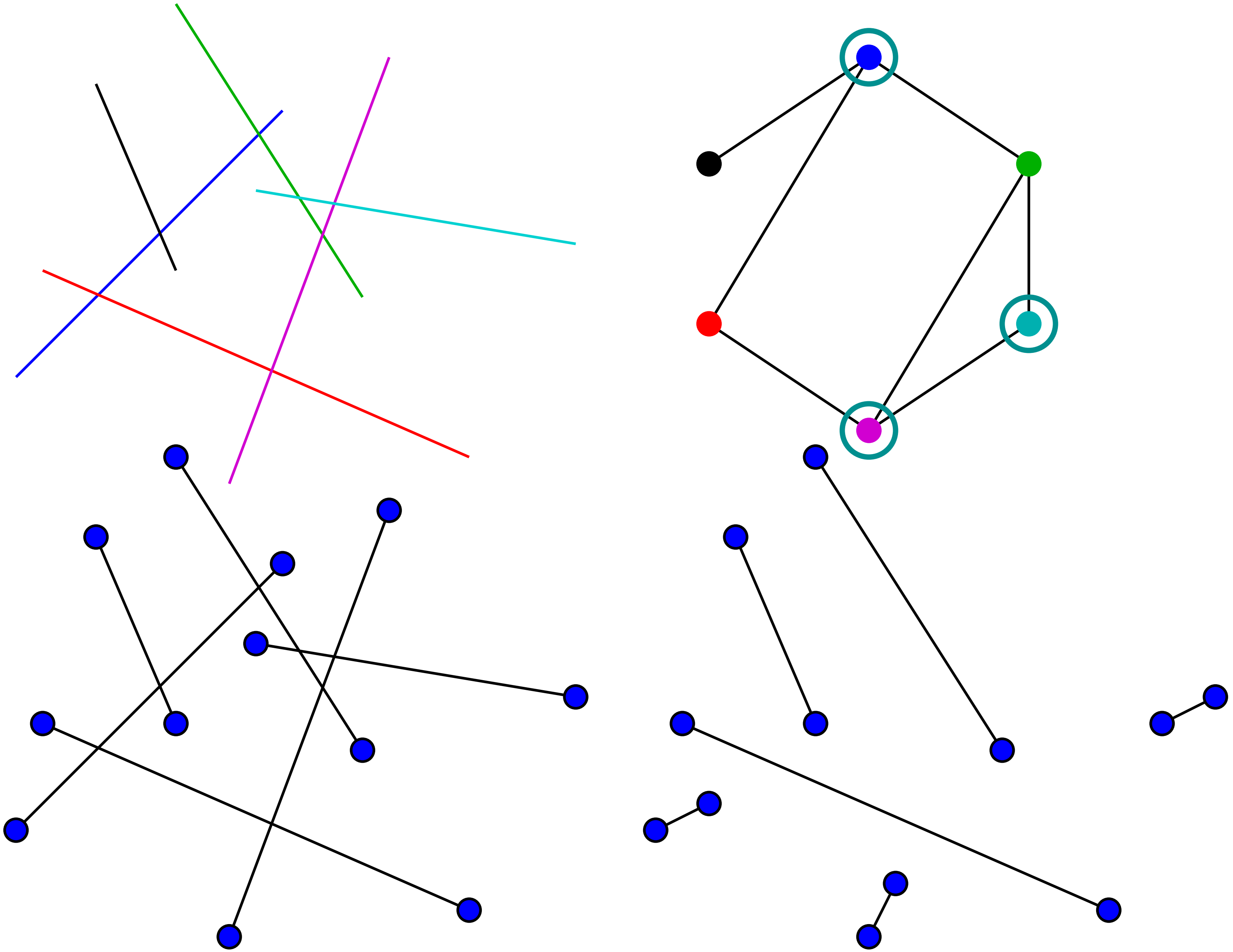
Min Vertex Cover of line segments



Vertex cover = set of vertices that doesn't miss any edge.



Reduction



Cycles: Lower Bound

Recall: $\mathrm{MKV}(G, \delta) =$ max # vertices we can keep fixed to make δ plane

Theorem (Pach & Tardos GD '01, DCG '02)

For any drawing δ of an n-cycle C_n

$\mathrm{MKV}(C_n, \delta) \geq \lfloor \sqrt{n} \rfloor$

Cycles: Lower Bound

Recall: $\mathrm{MKV}(G, \delta) =$ max # vertices we can keep fixed to make δ plane

Theorem (Pach & Tardos GD '01, DCG '02)
For any drawing δ of an n-cycle C_n
$\mathrm{MKV}(C_n, \delta) \geq \lfloor \sqrt{n} \rfloor$

Proof Use the Erdős–Szekeres theorem

Lemma (Erdős and Szekeres '35)

A sequence of n different real numbers contains a monotone subsequence of length (at least) $\lfloor\sqrt{n}\rfloor$

3 9 12 16 7 6 13 1 10 11 4 8 2 15 5 14

$n = 16$

Erdős–Szekeres for a monotone subsequence

Lemma (Erdős and Szekeres '35)

A sequence of n different real numbers contains a monotone subsequence of length (at least) $\lfloor\sqrt{n}\rfloor$

3 9 12 16 7 6 13 1 10 11 4 8 2 15 5 14

$n = 16$

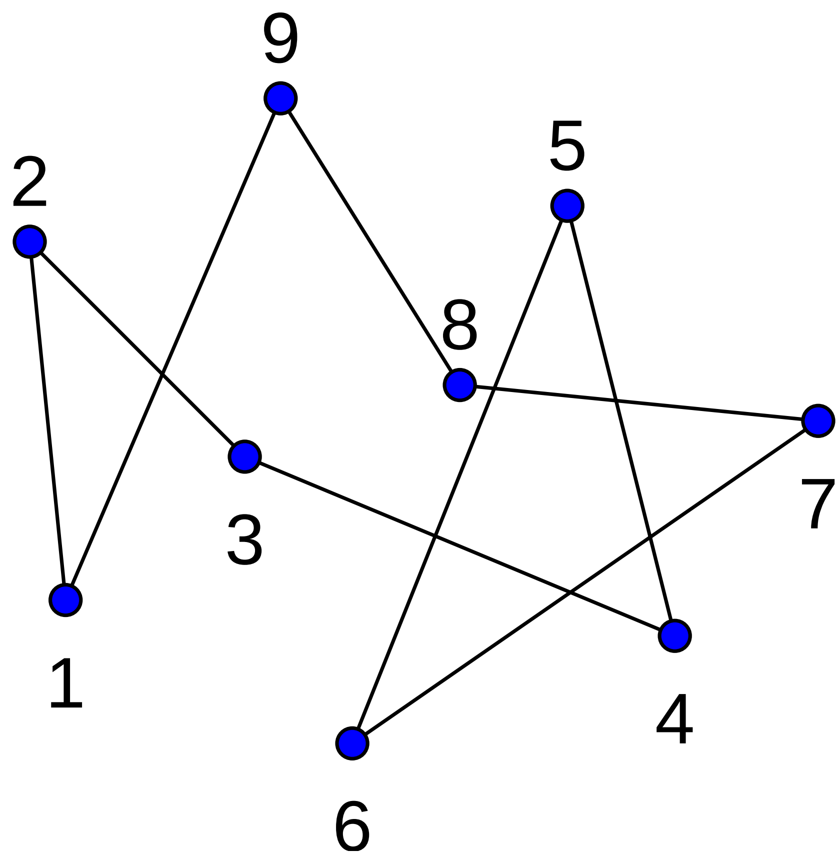
Lemma (Erdős and Szekeres ’35)

A sequence of n different real numbers contains a monotone subsequence of length (at least) $\lfloor\sqrt{n}\rfloor$

3 9 12 16 7 6 13 1 10 11 4 8 2 15 5 14

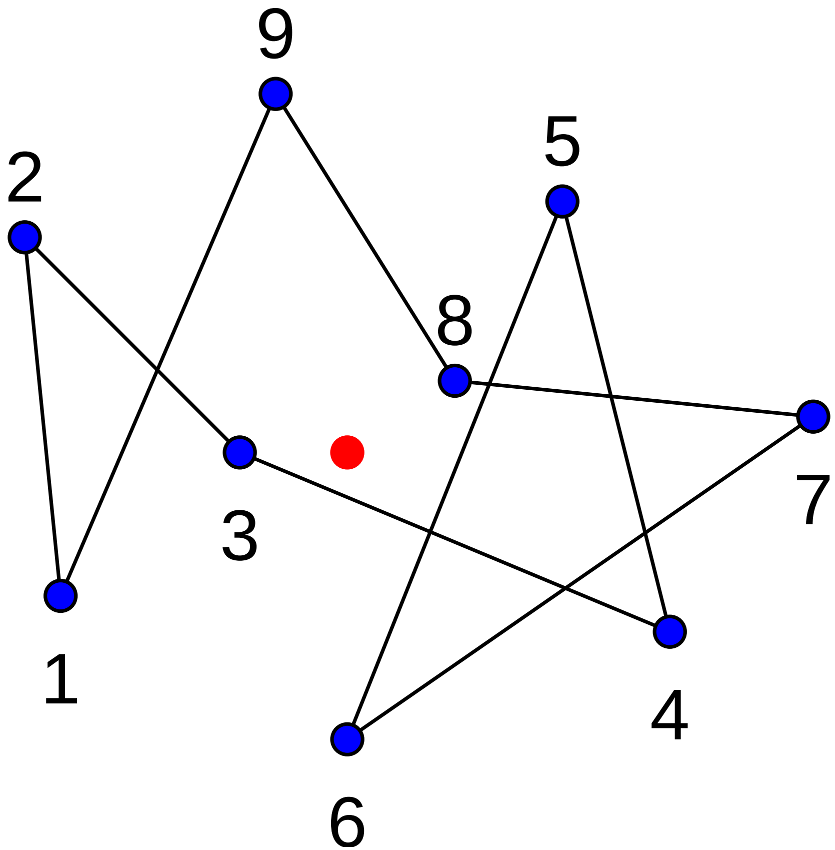
$n = 16$

From drawings to sequences

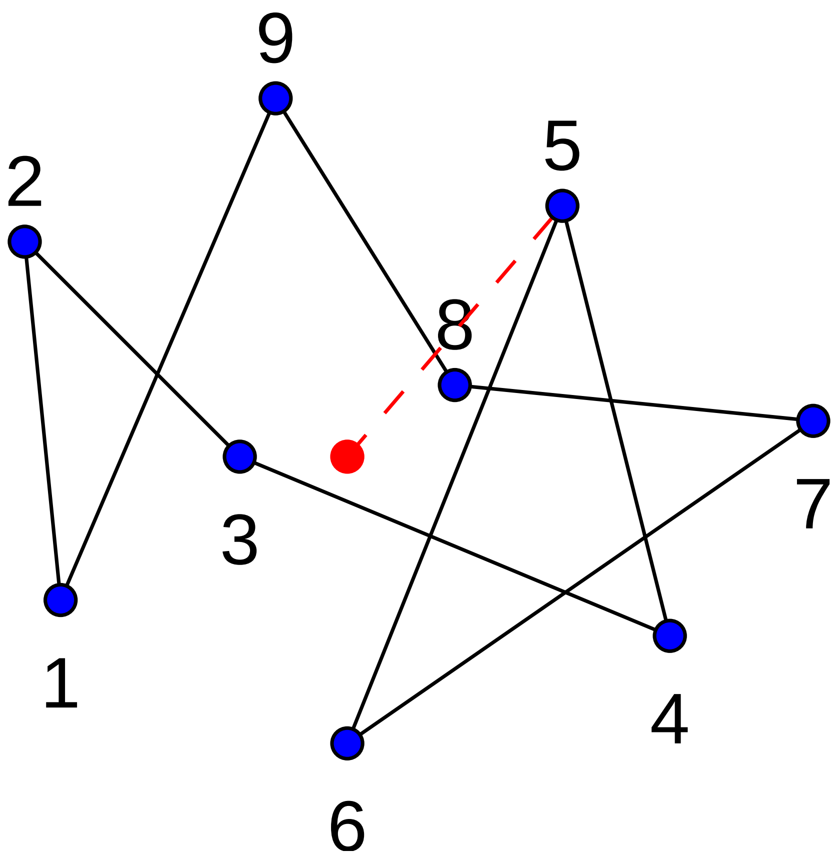


Given a drawing...

From drawings to sequences



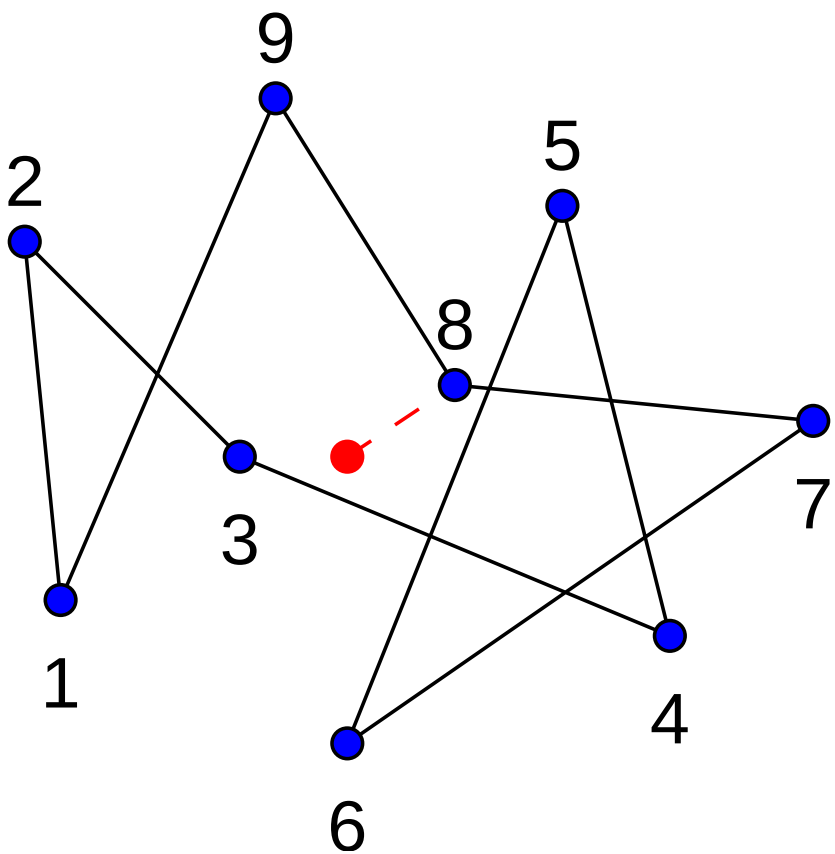
A point in general position



Obtain a sequence according to the angular order

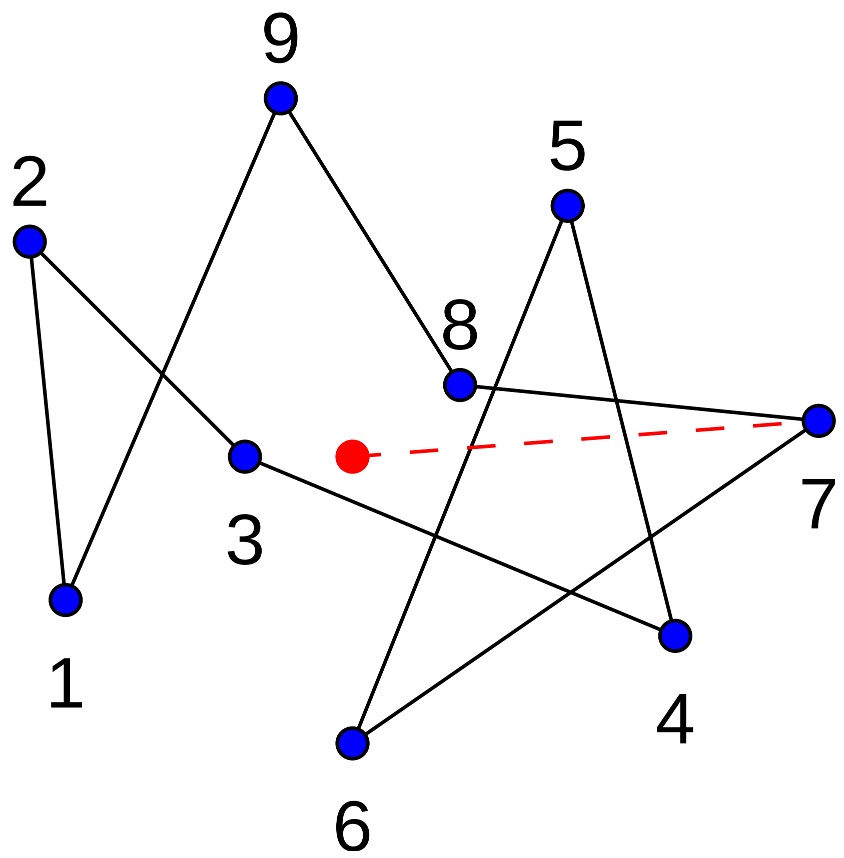
5

From drawings to sequences



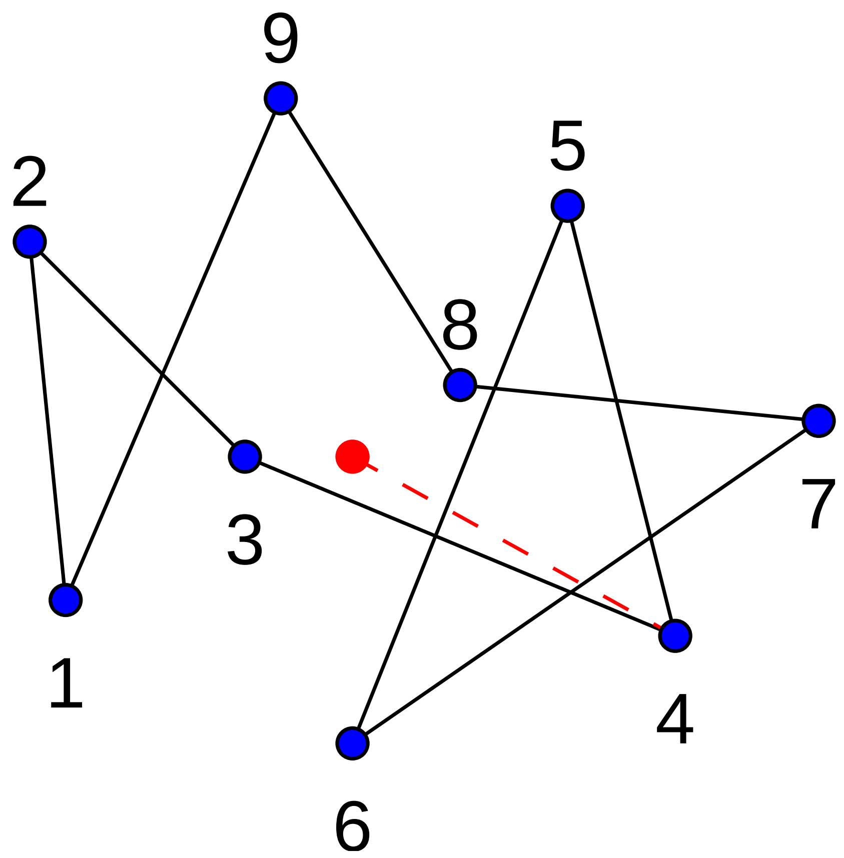
Obtain a sequence according to the angular order

5 8



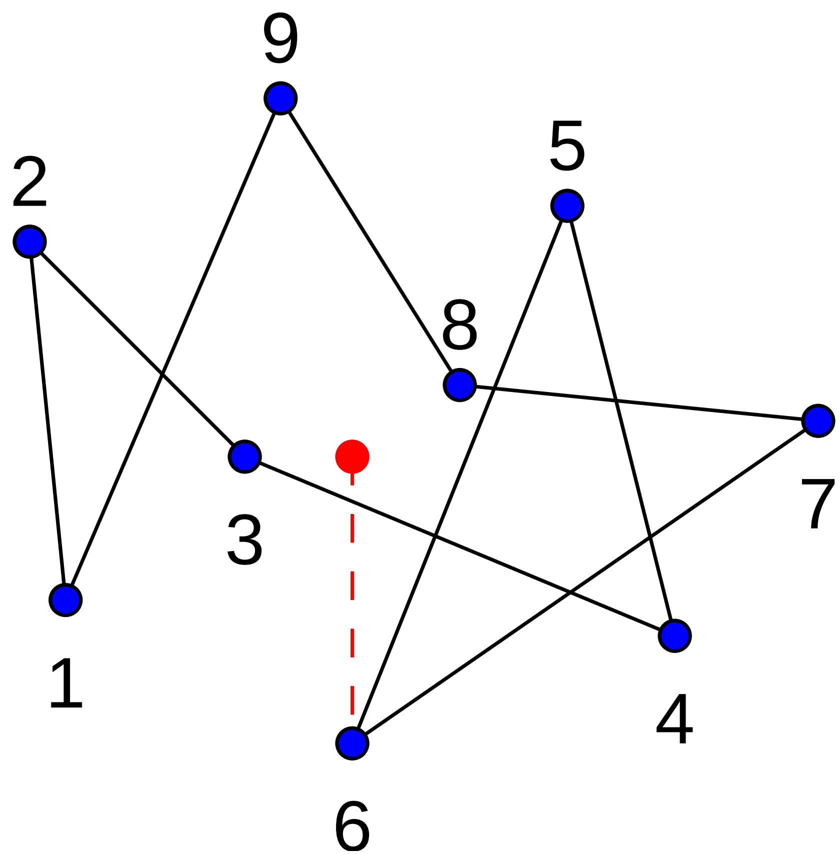
Obtain a sequence according to the angular order

5 8 7



Obtain a sequence according to the angular order

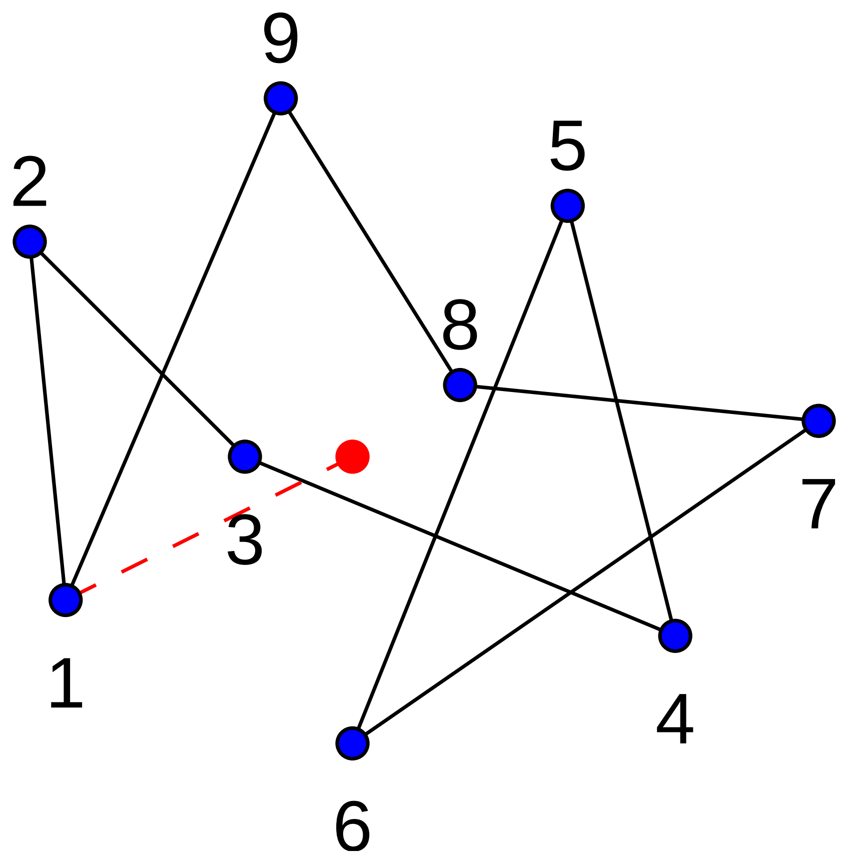
5 8 7 4



Obtain a sequence according to the angular order

5 8 7 4 6

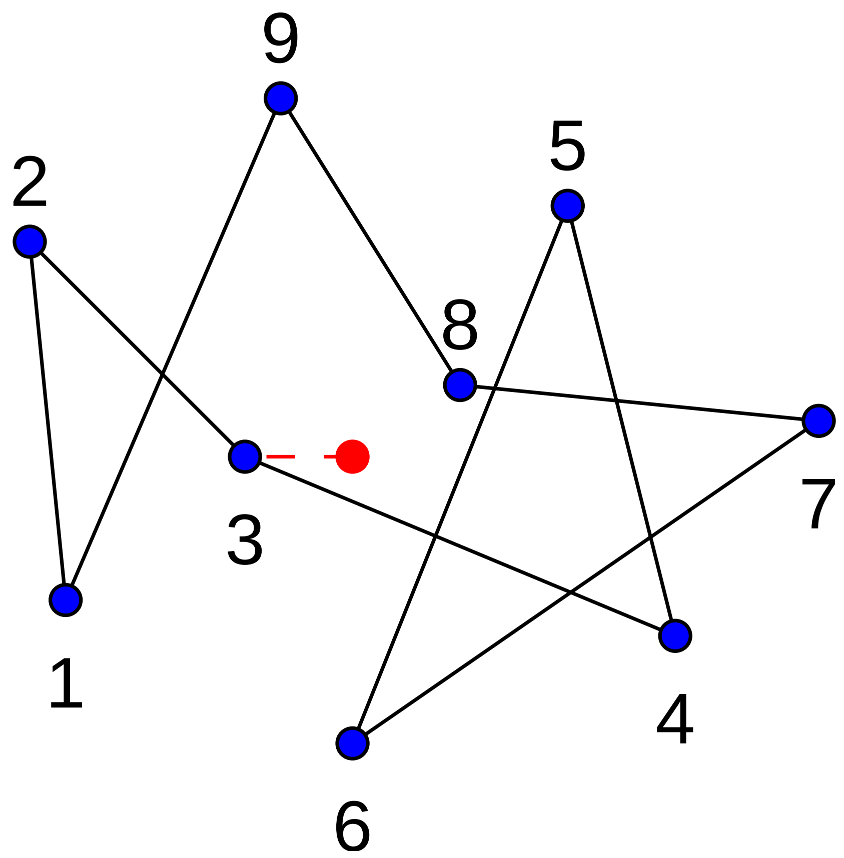
From drawings to sequences



Obtain a sequence according to the angular order

5 8 7 4 6 1

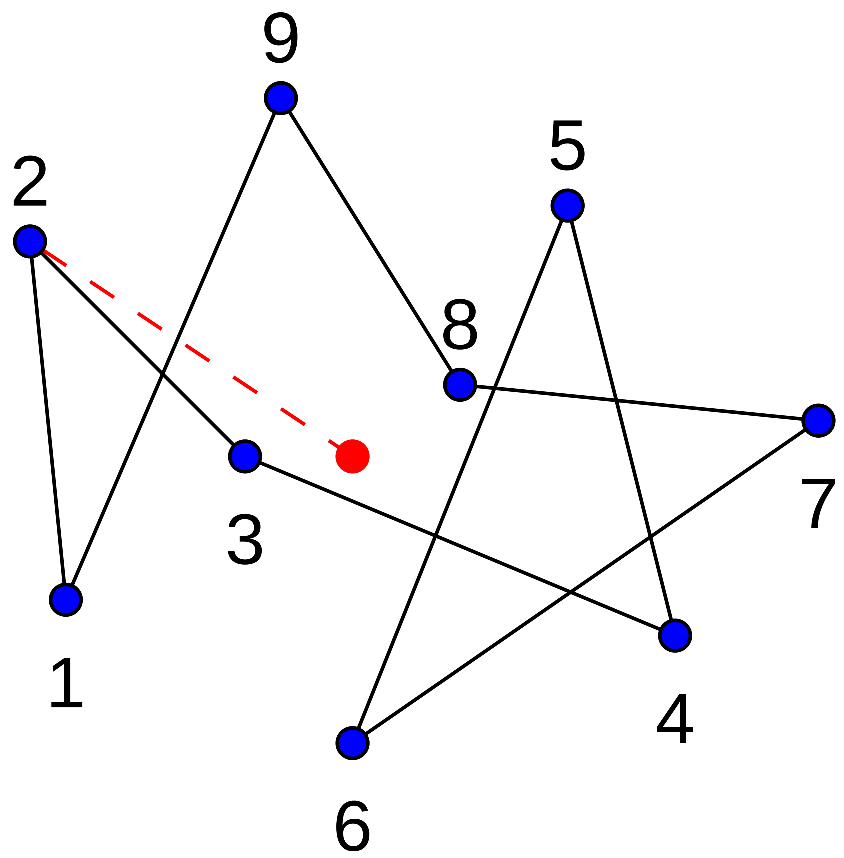
From drawings to sequences



Obtain a sequence according to the angular order

5 8 7 4 6 1 3

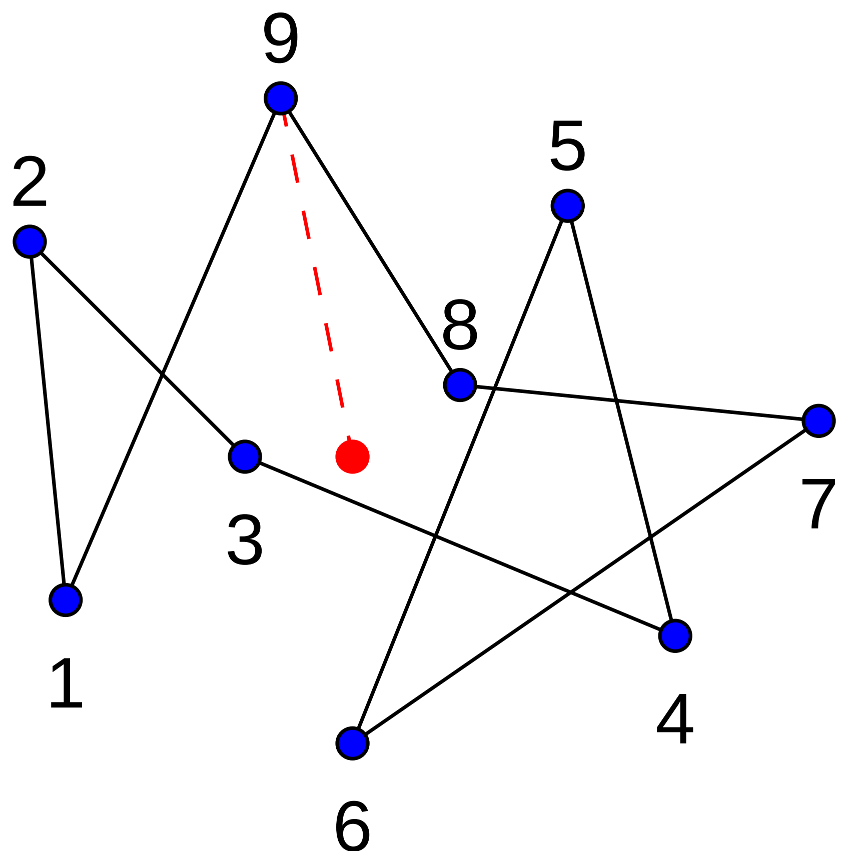
From drawings to sequences



Obtain a sequence according to the angular order

5 8 7 4 6 1 3 2

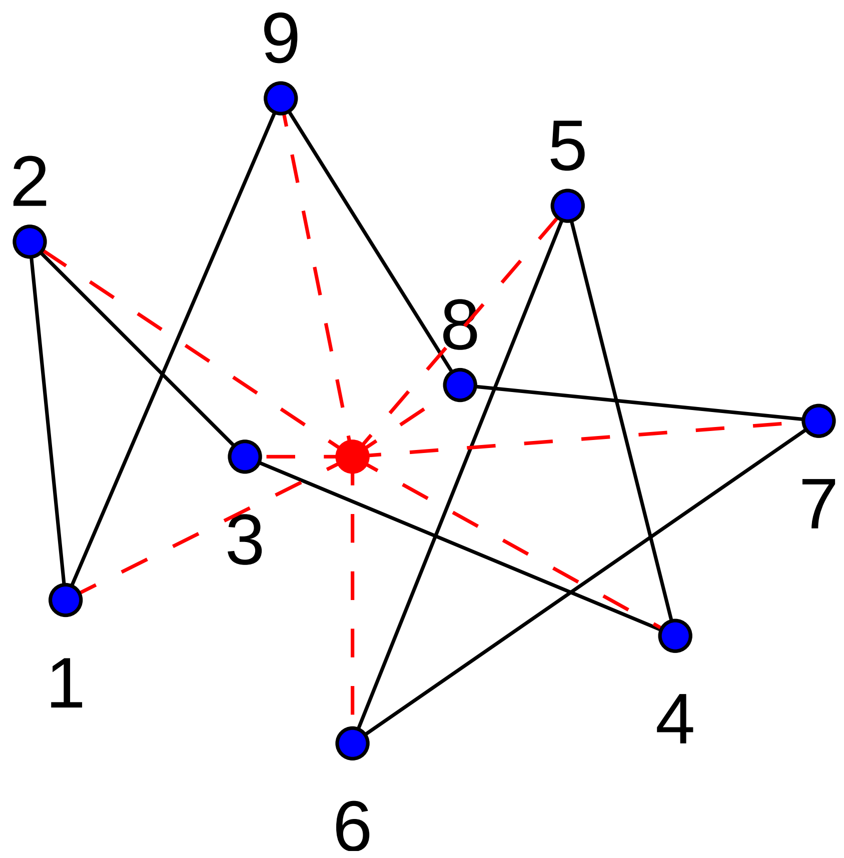
From drawings to sequences



Obtain a sequence according to the angular order

5 8 7 4 6 1 3 2 9

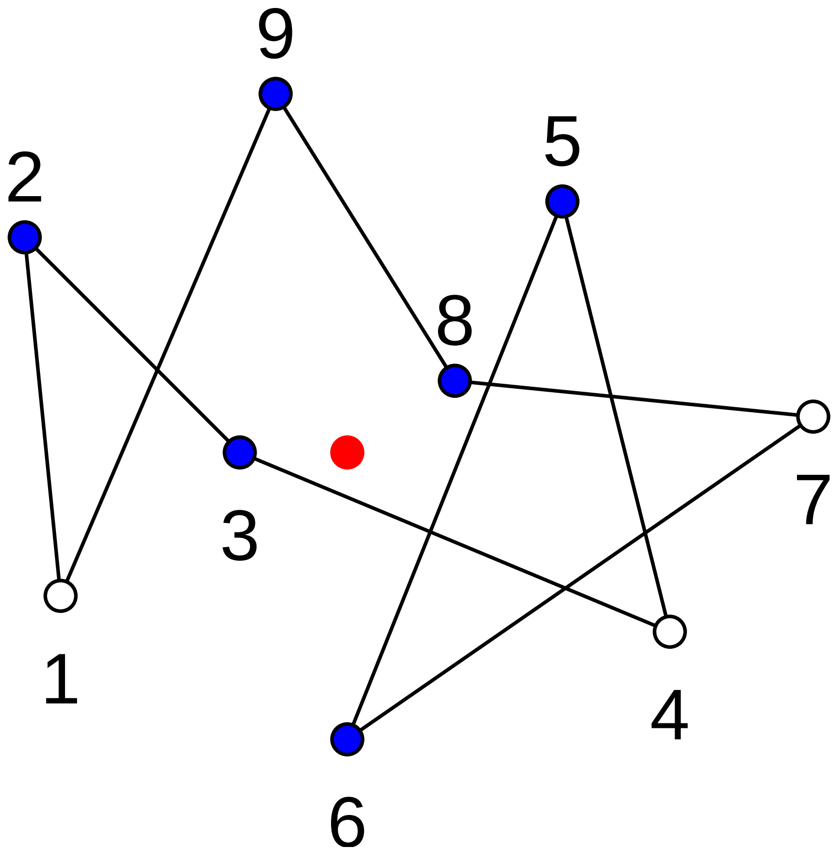
From drawings to sequences



Choose a monotone subsequence of length $\lfloor \sqrt{n} \rfloor$

5 8 7 4 6 1 3 2 9

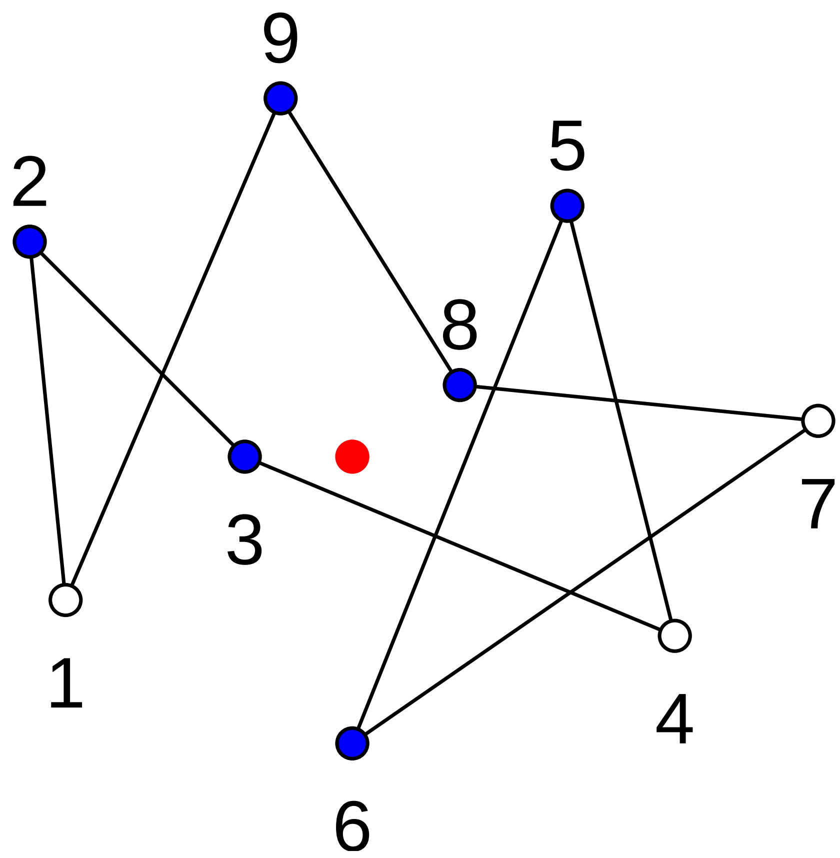
From drawings to sequences



Choose a monotone subsequence of length $\lfloor \sqrt{n} \rfloor$

5 8 7 4 6 1 3 2 9

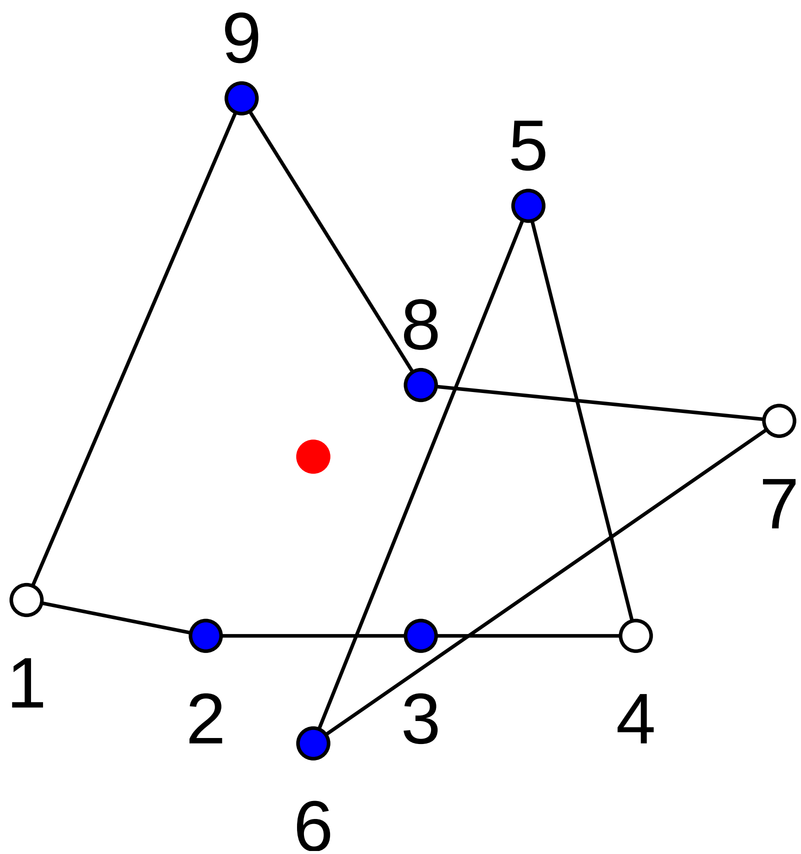
From drawings to sequences



Keep pts in the subseq fixed, and move remaining pts

5 8 7 4 6 1 3 2 9

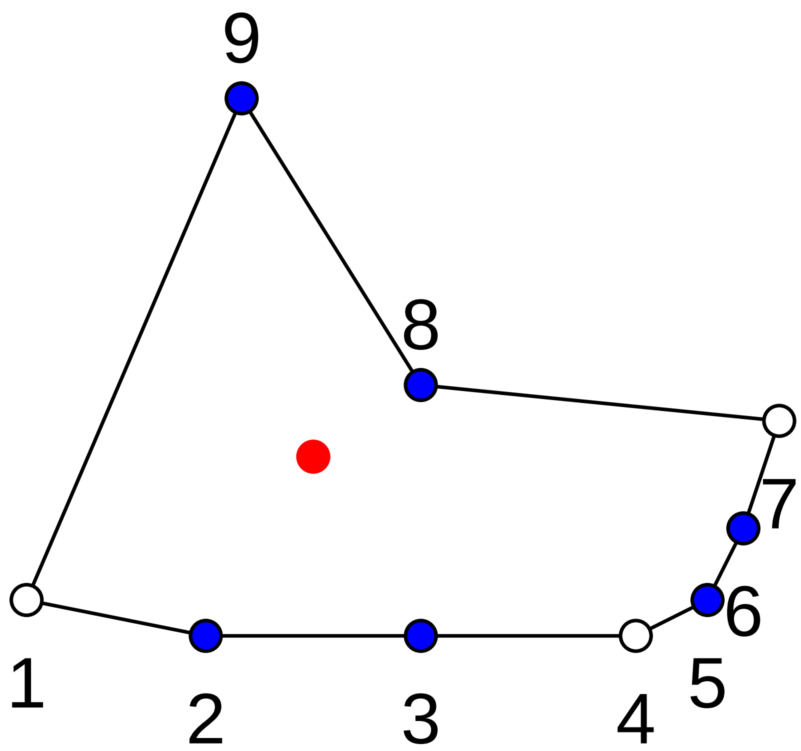
From drawings to sequences



Keep pts in the subseq fixed, and move remaining pts

5 8 7 4 6 1 3 2 9

From drawings to sequences



Keep pts in the subseq fixed, and move remaining pts

5 8 7 4 6 1 3 2 9

Cycles: Easy Upper Bound

Recall: $\mathrm{MKV}(G, \delta) =$ max # vertices we can keep fixed to make δ plane

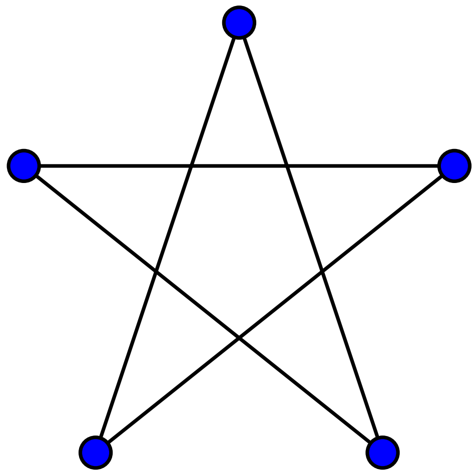
Theorem $\exists$ a drawing δ of C_n (n odd) s.t. $\mathrm{MKV}(C_n, \delta) \leq \lfloor n/2 \rfloor$

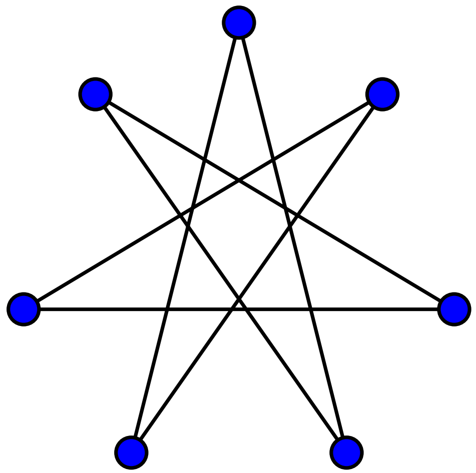
Cycles: Easy Upper Bound

Recall: $\mathrm{MKV}(G, \delta) =$ max # vertices we can keep fixed to make δ plane

Theorem $\exists$ a drawing δ of C_n (n odd) s.t. $\mathrm{MKV}(C_n, \delta) \leq \lfloor n/2 \rfloor$

Proof Use a *thrackle*:





Verbitsky ('07) independently obtained the following

- It is NP-hard to compute $\mathrm{MMV}(G, \delta)$
- For n-vertex planar graphs, $\mathrm{MKV} \geq 3$
- $\mathrm{MMV}(G, \delta) \geq$ (matching no. of $G) - 1$
  - For n-vertex planar graphs, $\delta \geq 3$ and $n \geq 10$
    we cannot keep $(2n+1)/3$ vertices in some cases
    the matching no. $\geq (n+2)/3$ (Nishizeki & Baybars '79)
  - For n-vertex planar graphs, 4-connected
    we cannot keep $(n+3)/2$ vertices in some cases
    4-conn. planar graphs are Hamiltonian (Tutte '56)
- Also investigate "obfuscation complexity of a graph"
  that might be called "max rectilinear crossing number"