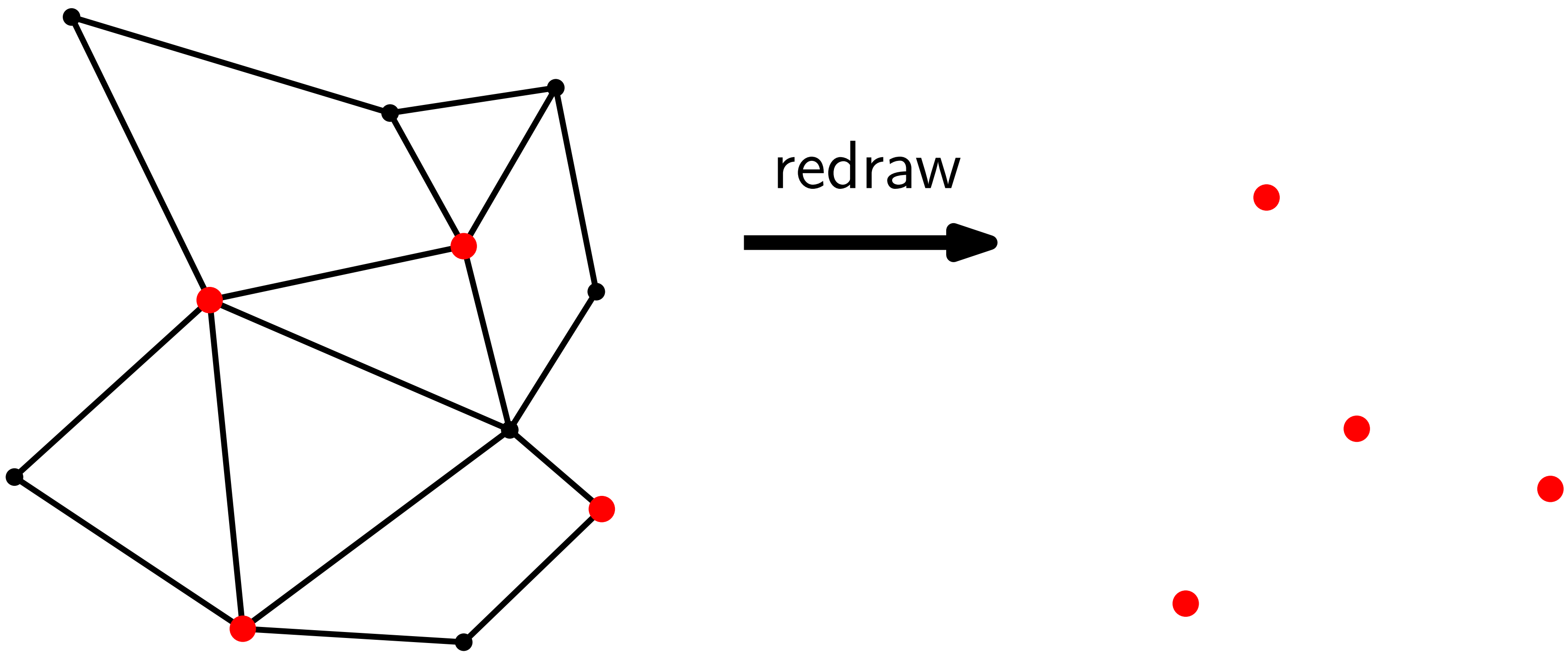
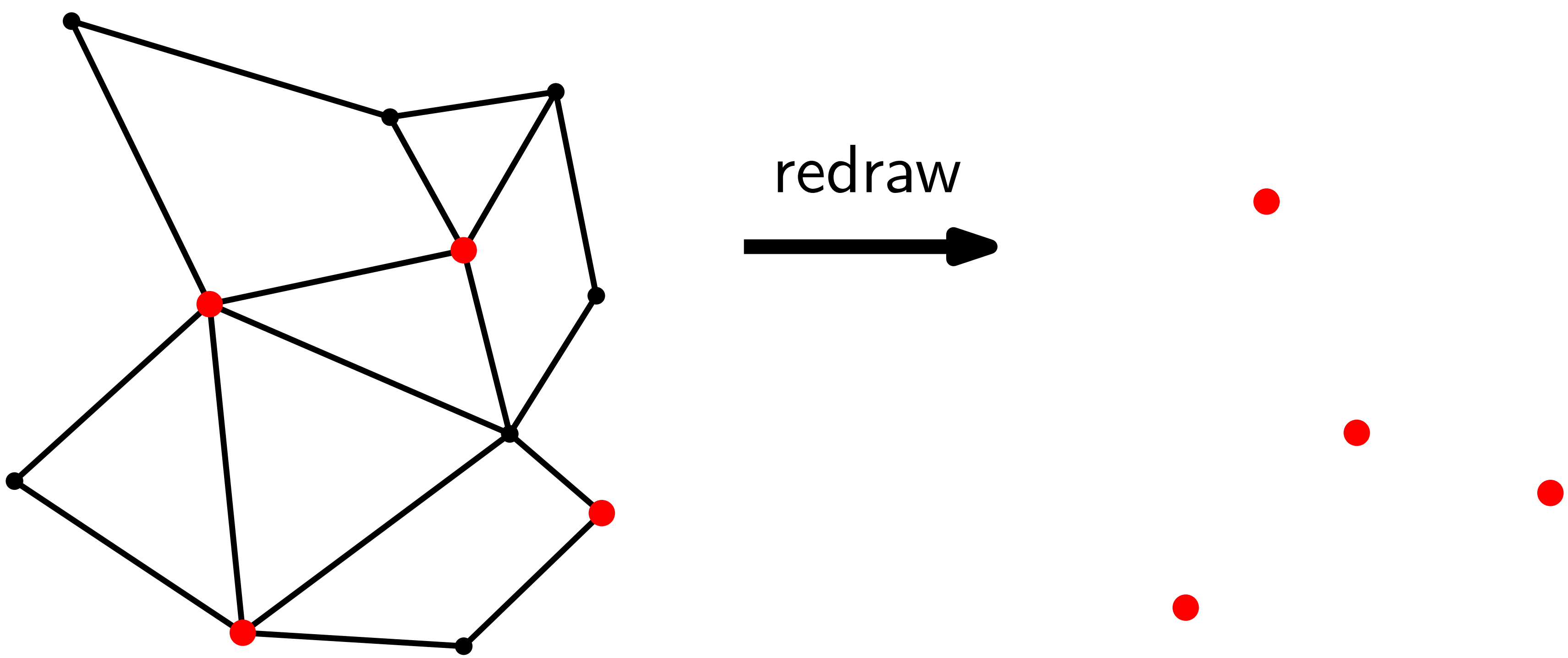
Every Collinear Vertex Set in a Planar Graph is Free

Vida Dujmović, Fabrizio Frati, Daniel Gonçalves, Pat Morin, Günter Rote



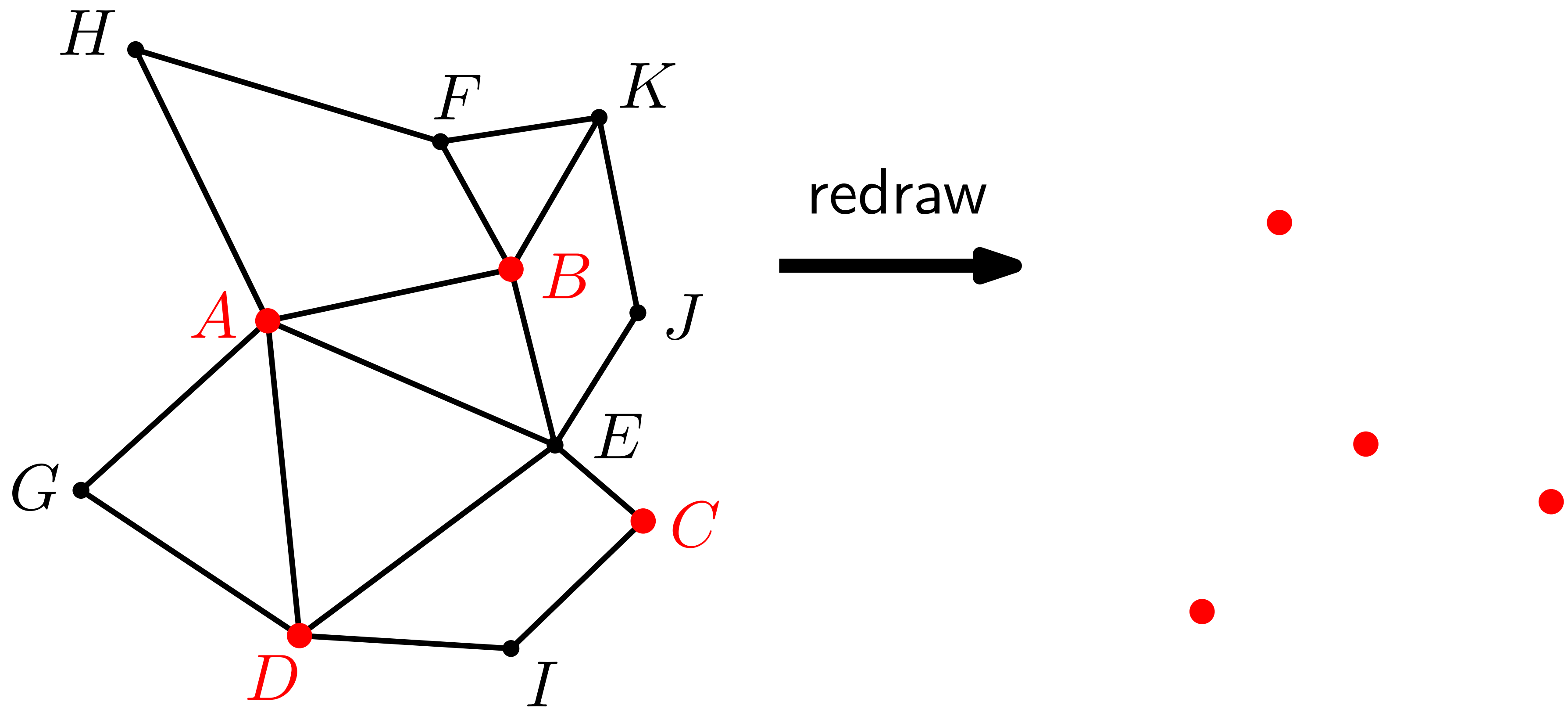
Every Collinear Vertex Set in a Planar Graph is Free

Vida Dujmović, Fabrizio Frati, Daniel Gonçalves, Pat Morin, Günter Rote



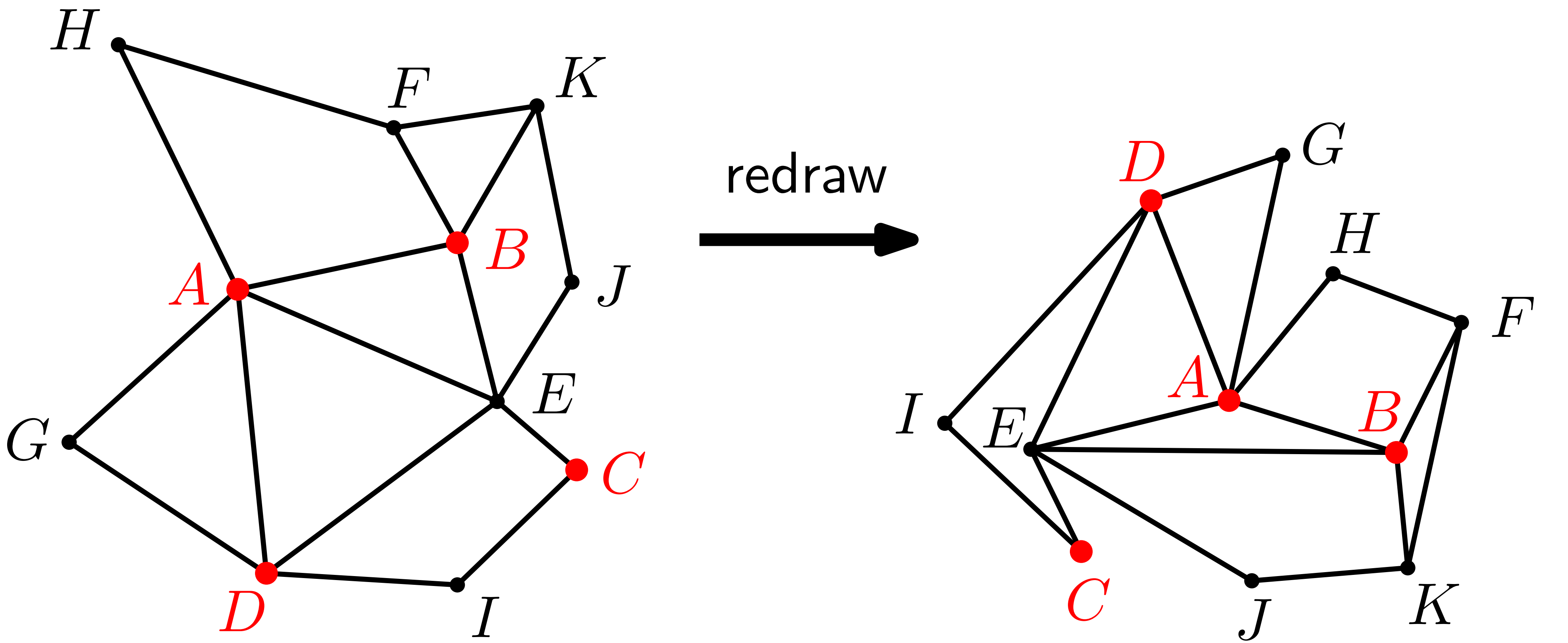
Every Collinear Vertex Set in a Planar Graph is Free

Vida Dujmović, Fabrizio Frati, Daniel Gonçalves, Pat Morin, Günter Rote



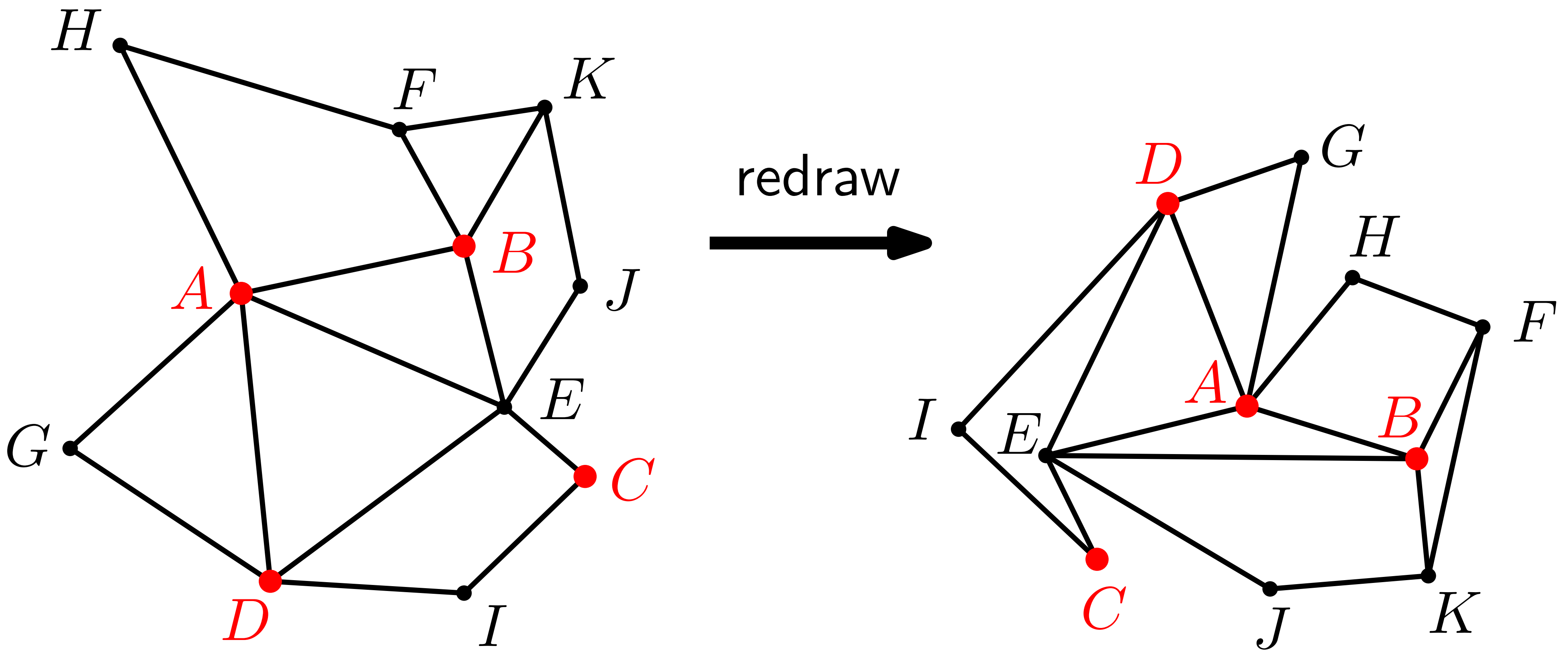
Every Collinear Vertex Set in a Planar Graph is Free

Vida Dujmović, Fabrizio Frati, Daniel Gonçalves, Pat Morin, Günter Rote



Every Collinear Vertex Set in a Planar Graph is Free

Vida Dujmović, Fabrizio Frati, Daniel Gonçalves, Pat Morin, Günter Rote



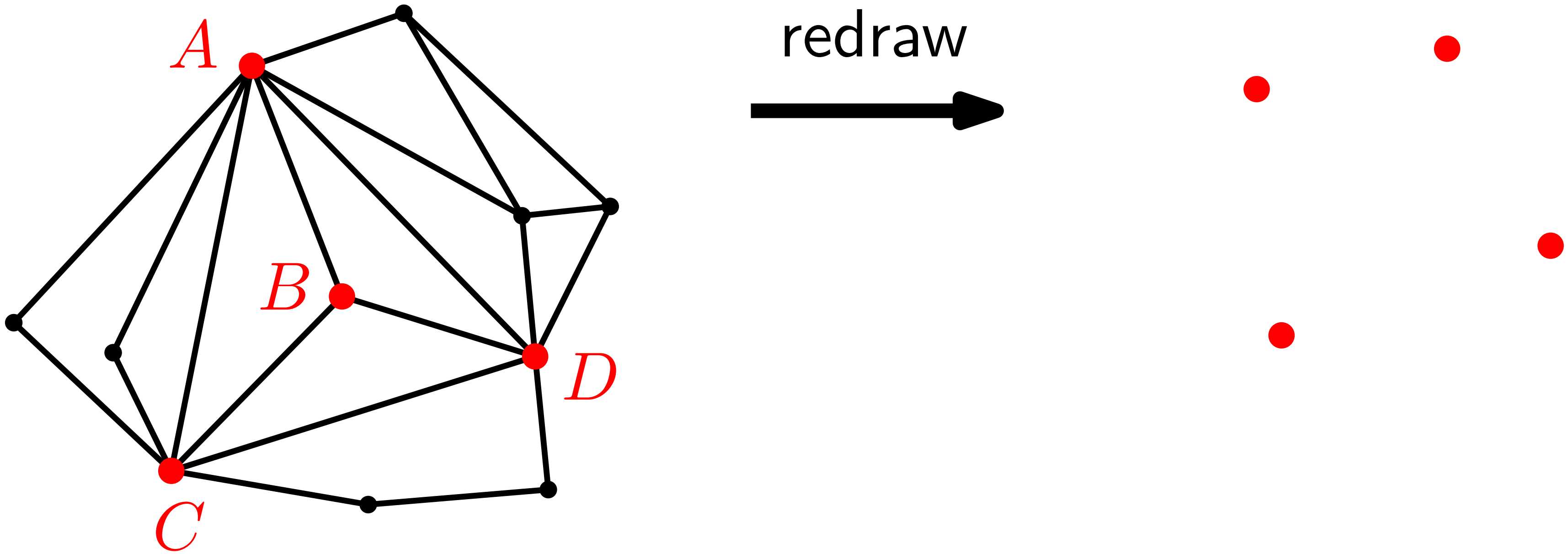
$\{A, B, C, D\}$ is *free* if it can be redrawn on *every* 4-point set.

Freie Universität Berlin

Every Collinear Vertex Set in a Planar Graph is Free

Vida Dujmović, Fabrizio Frati, Daniel Gonçalves, Pat Morin, Günter Rote

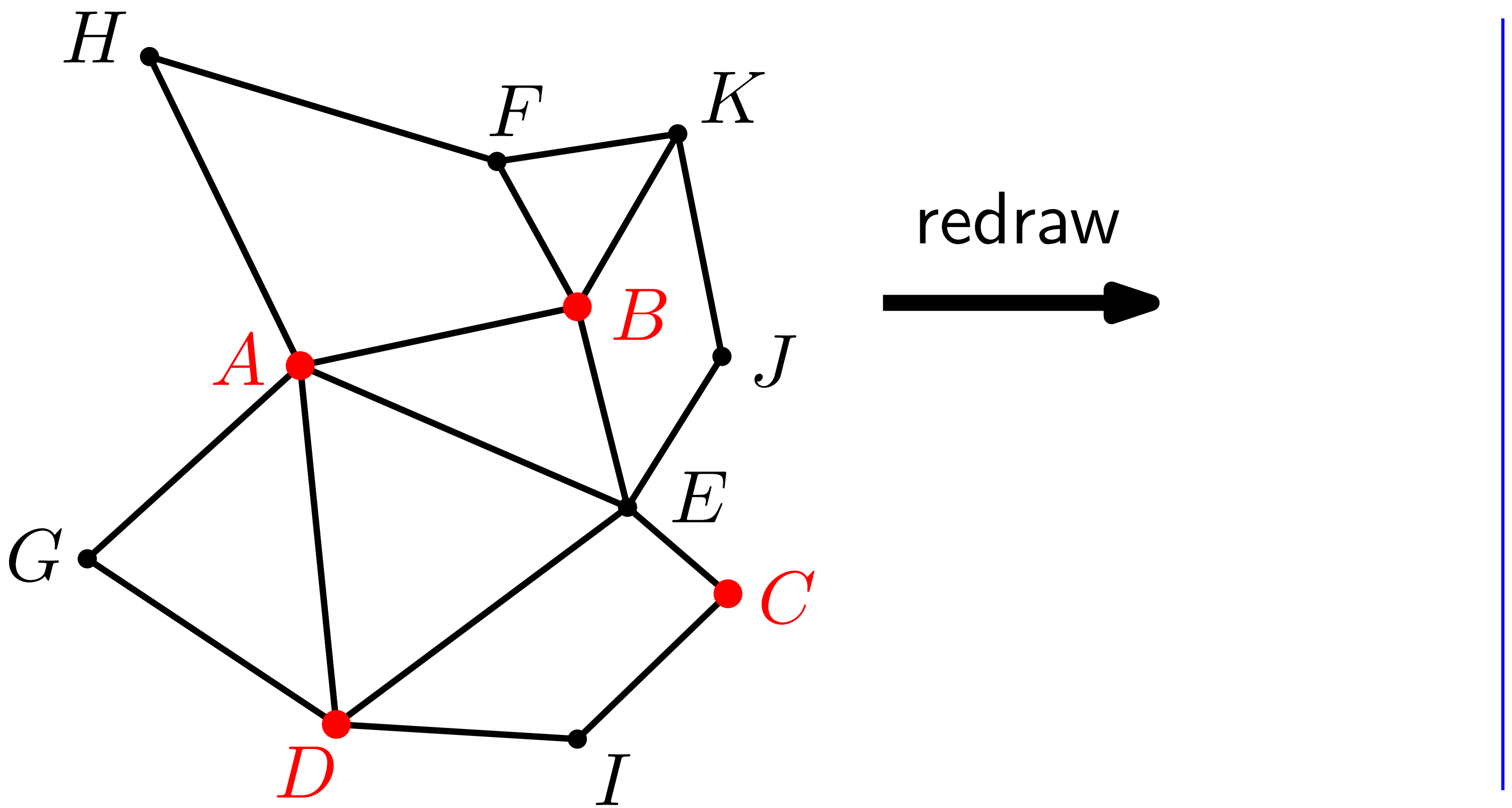
NOT a free set



$\{A, B, C, D\}$ is *free* if it can be redrawn on *every* 4-point set.

Every Collinear Vertex Set in a Planar Graph is Free

Vida Dujmović, Fabrizio Frati, Daniel Gonçalves, Pat Morin, Günter Rote

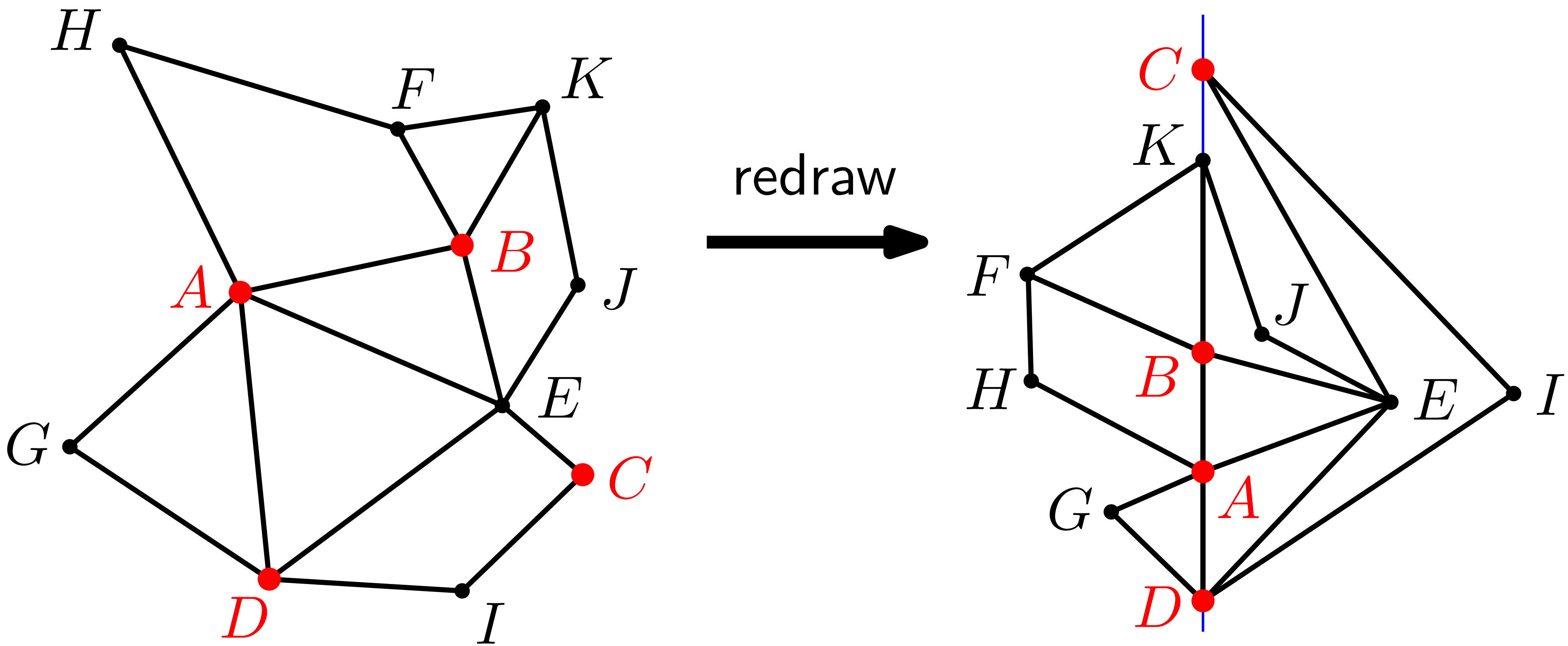


$\{A, B, C, D\}$ is *free* if it can be redrawn on *every* 4-point set.

$\{A, B, C, D\}$ is *collinear* if it can be redrawn on a line.

Every Collinear Vertex Set in a Planar Graph is Free

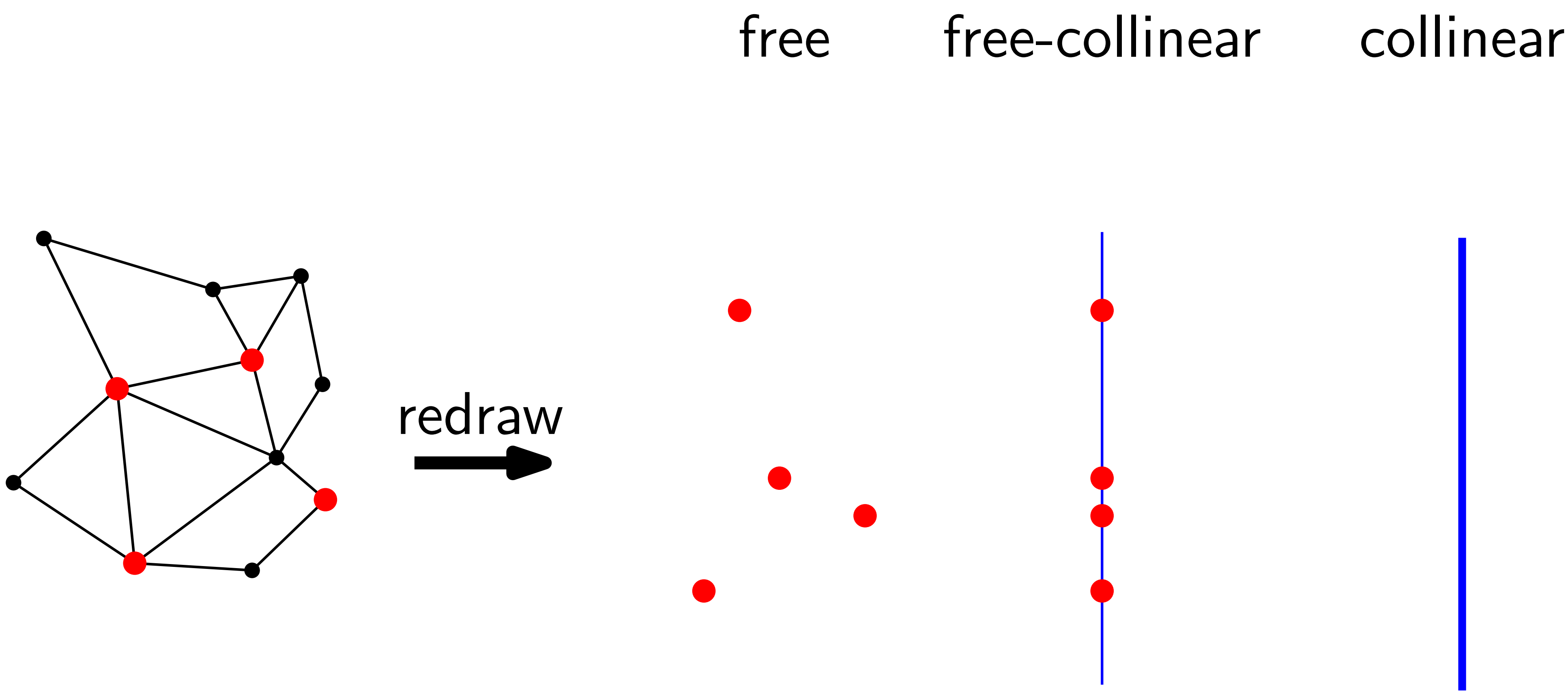
Vida Dujmović, Fabrizio Frati, Daniel Gonçalves, Pat Morin, Günter Rote



$\{A, B, C, D\}$ is *free* if it can be redrawn on *every* 4-point set.

$\{A, B, C, D\}$ is *collinear* if it can be redrawn on a line.

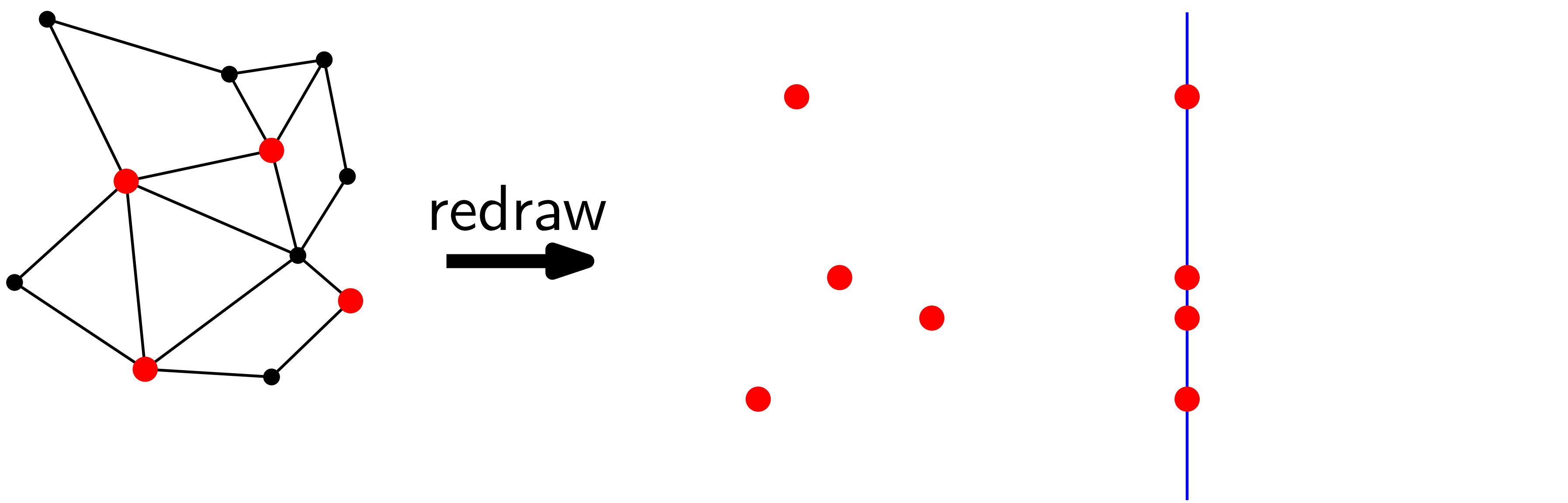
Free, Free-Collinear, Collinear



Free, Free-Collinear, Collinear

(trivial) (trivial)

free $\Rightarrow$ free-collinear $\Rightarrow$ collinear



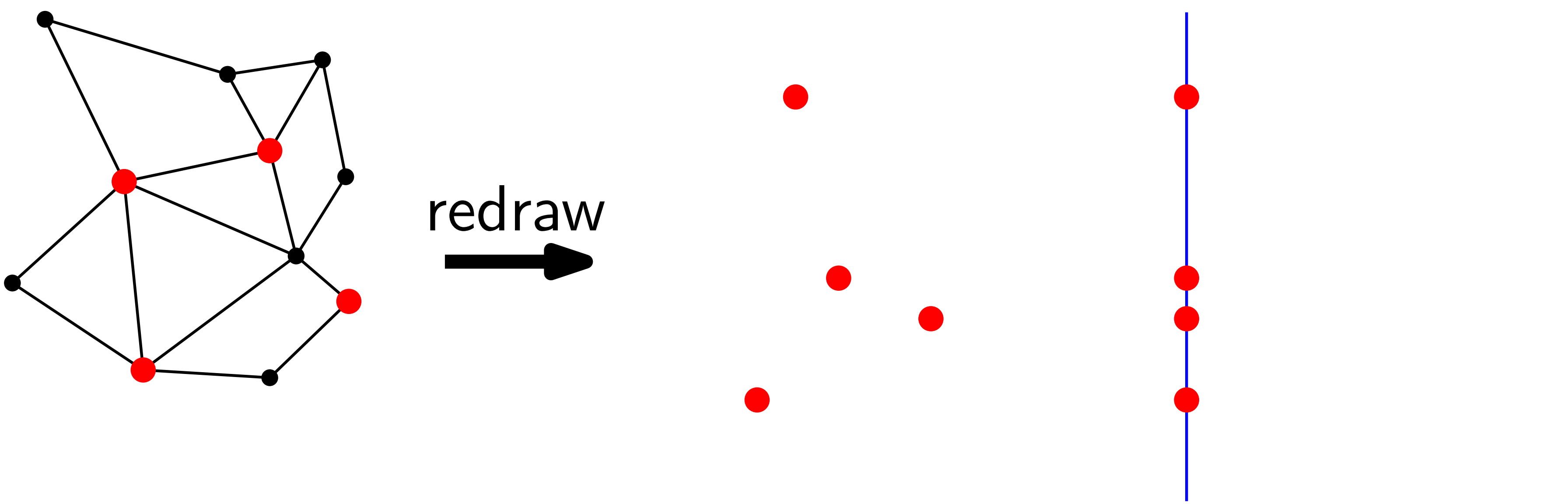
Free, Free-Collinear, Collinear

(trivial) (trivial)

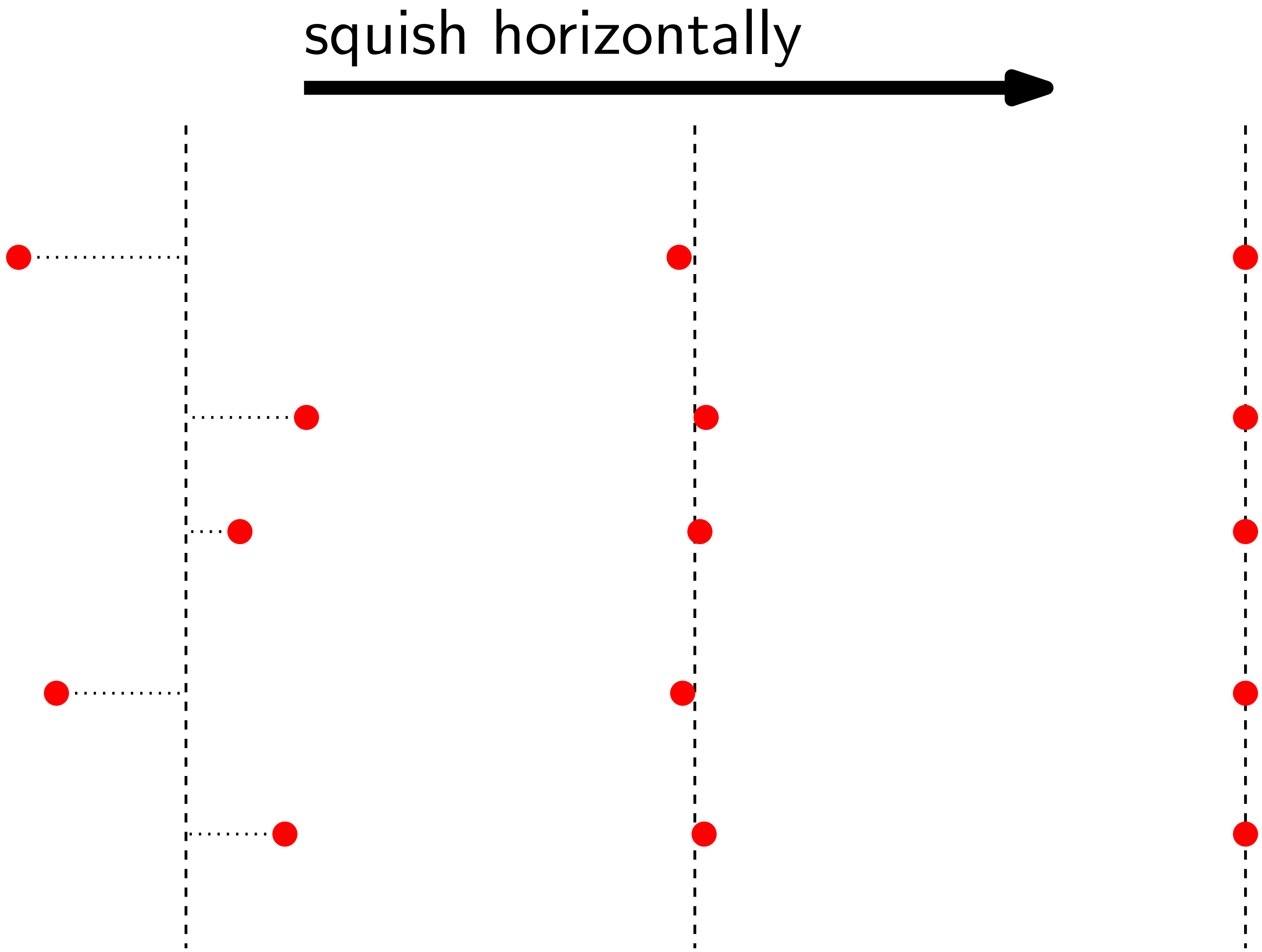
free $\Rightarrow$ free-collinear $\Rightarrow$ collinear

(easy) this talk

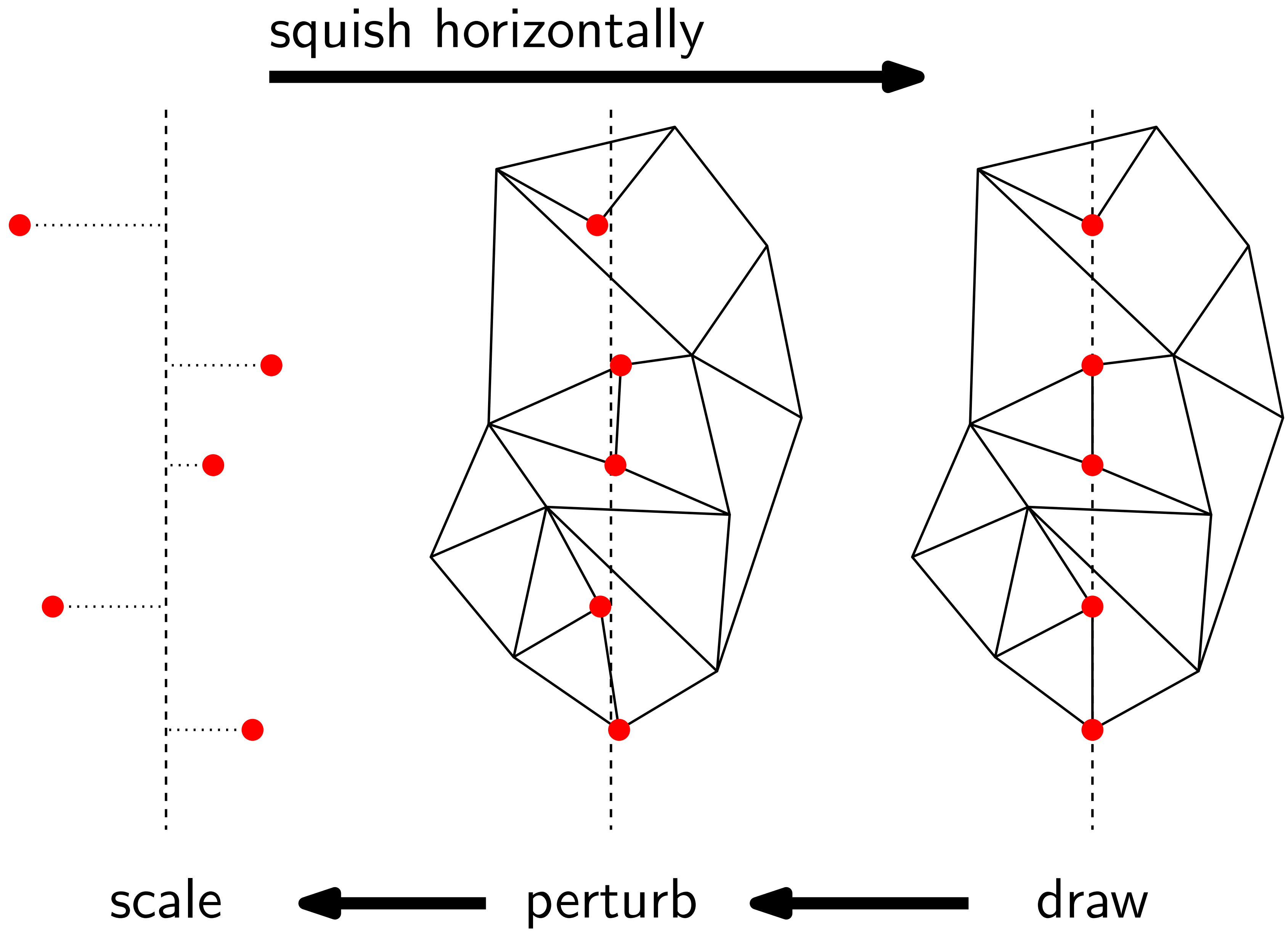
$\Leftarrow$ $\Leftarrow$



Free-Collinear $\Longrightarrow$ Free

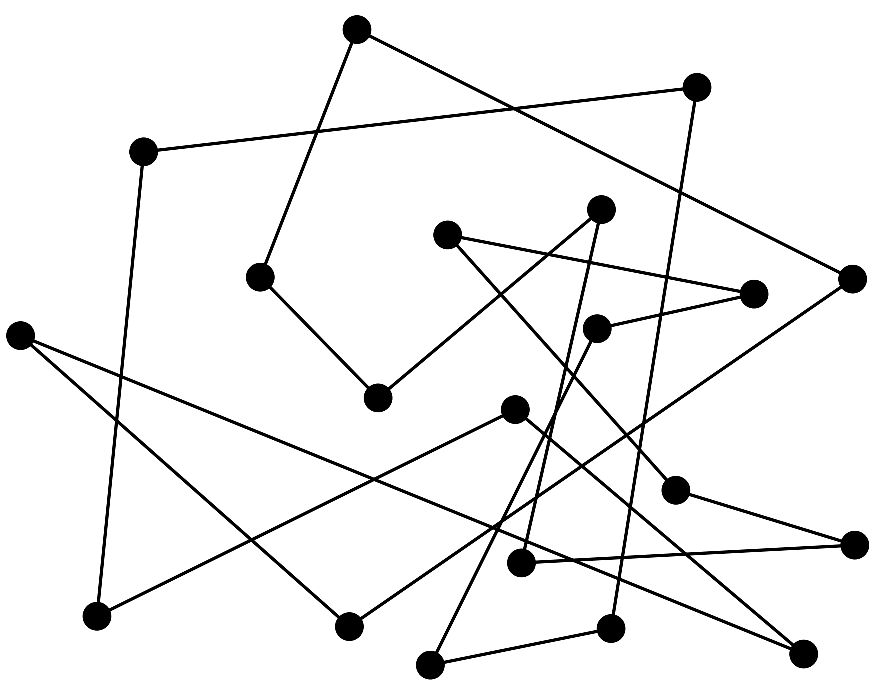


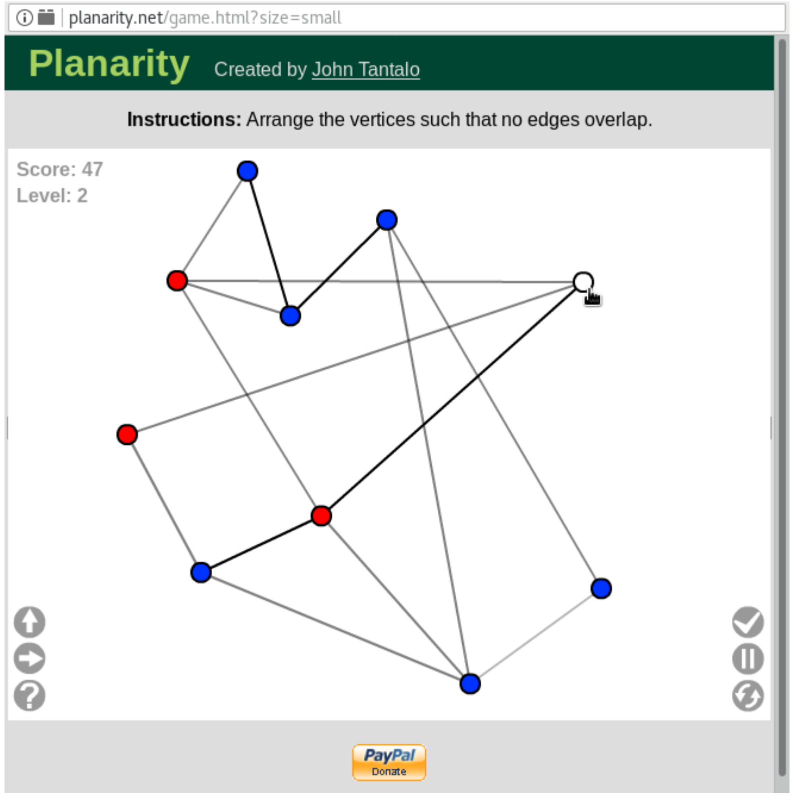
Free-Collinear $\Longrightarrow$ Free



Applications

- Partial simultaneous geometric embeddings
- Universal point subsets
- Column planarity
- Untangling:
  Given a drawing of a planar graph with crossings:
  Find a *plane* redrawing while keeping *many vertices fixed*.



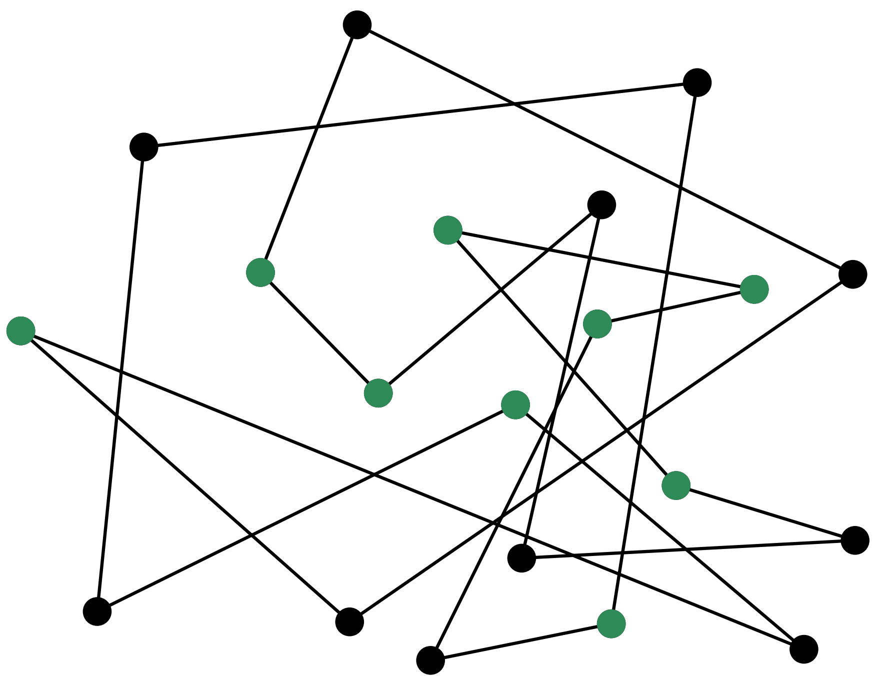


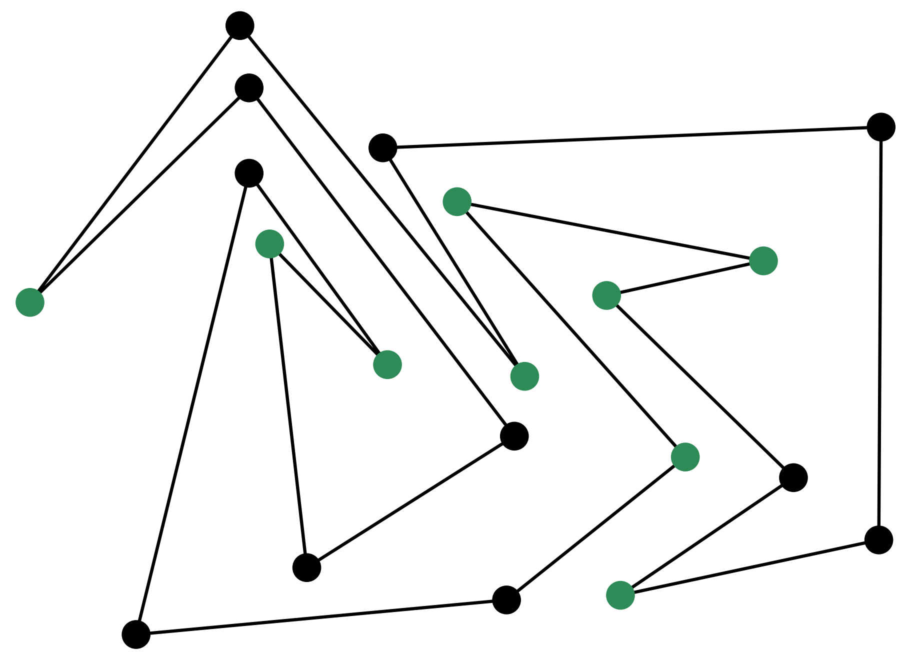
Applications

Freie Universität Berlin

- Partial simultaneous geometric embeddings
- Universal point subsets
- Column planarity
- Untangling:

  Given a drawing of a planar graph with crossings:
  Find a *plane* redrawing while keeping *many vertices fixed*.





Untangling

Freie Universität Berlin

Given a drawing of a planar graph with crossings:
Find a *plane* redrawing while keeping *many vertices fixed*.

THEOREM. If G has a free (*now:* collinear) set S, then we can leave $\Omega(\sqrt{|S|})$ vertices fixed. [Ravsky and Verbitsky 2011]
[Bose, Dujmović, Hurtado, Langerman, Morin, Wood 2009]

COROLLARY: If G is a 3-connected triangulation, then $\Omega(\sqrt{n})$ vertices (*previously:* $\Omega(n^{1/4})$ vertices) can remain fixed.

(Upper bound: $O(\sqrt{n \log^3 n})$ [Cibulka 2010])

Largest Free Sets

Every n-vertex planar graph has a collinear/free set of size $\Omega(\sqrt{n}\,)$. [Bose, Dujmović, Hurtado, Langerman, Morin, Wood 2009]

Proof by canonical order or by Schnyder decomposition.

Upper bound: $O(n^{0.986})$ [Ravsky and Verbitsky 2011]

follows from 3-regular planar graphs without long cycles [Grünbaum and Walther 1973]

Every planar graph of *bounded degree* has a collinear/free set of size $\Omega(n^{0.8})$. [Dujmović and Morin 2019]

Testing whether a set is collinear (*now also:* free) is NP-hard. [Mchedlidze, Radermacher, Rutter 2017]

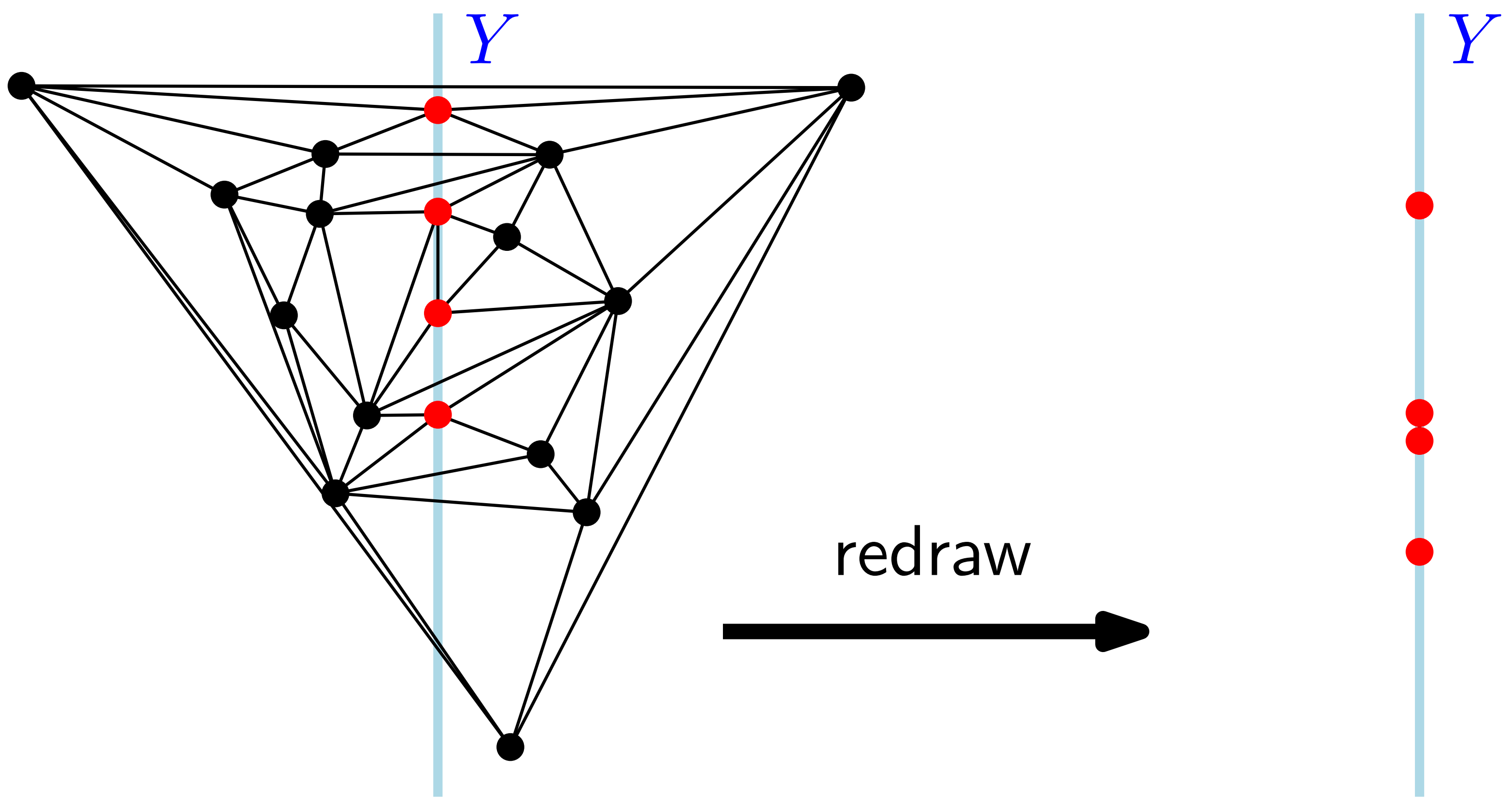
Proof Outline

Given:

- a drawing with some vertices $v_1, \ldots, v_k$ on a vertical line Y
- new y-coordinates $b_1 < \cdots < b_k$ for these vertices

Wanted:

- a redrawing with these y-coordinates



Proof Outline

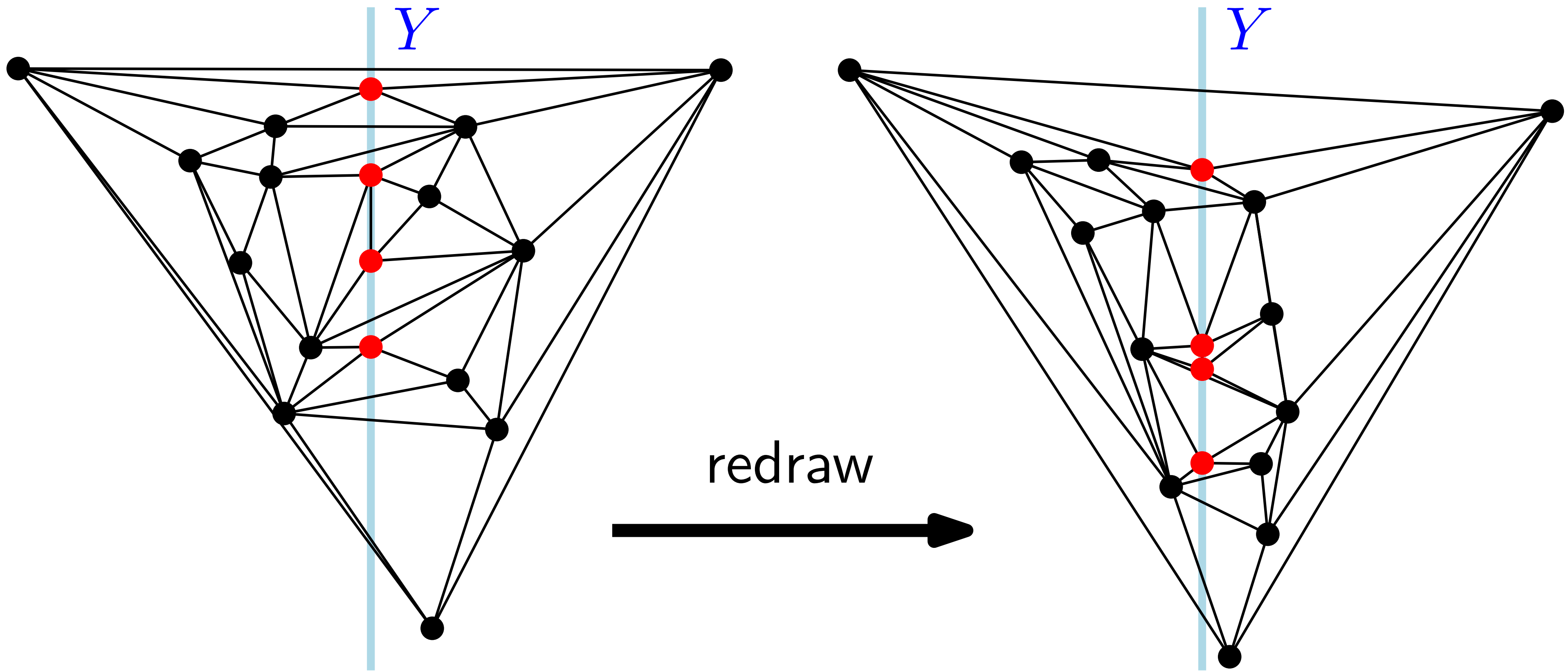
Freie Universität Berlin

Given:

- a drawing with some vertices $v_1, \ldots, v_k$ on a vertical line Y
- new y-coordinates $b_1 < \cdots < b_k$ for these vertices

Wanted:

- a redrawing with these y-coordinates



Proof Outline

Given:

- a drawing with some vertices $v_1, \ldots, v_k$ on a vertical line Y
- new y-coordinates $b_1 < \cdots < b_k$ for these vertices

Wanted:

- a redrawing with these y-coordinates

(i) Special case:
Maximal graph such that all edges intersect Y.

(ii) Treat edges e that do not intersect Y:
Use several types of reductions

A. a) *contract* e into a single vertex v
   b) draw the resulting graph
   c) perturb v into two vertices with an edge between

B. *flip* the edge

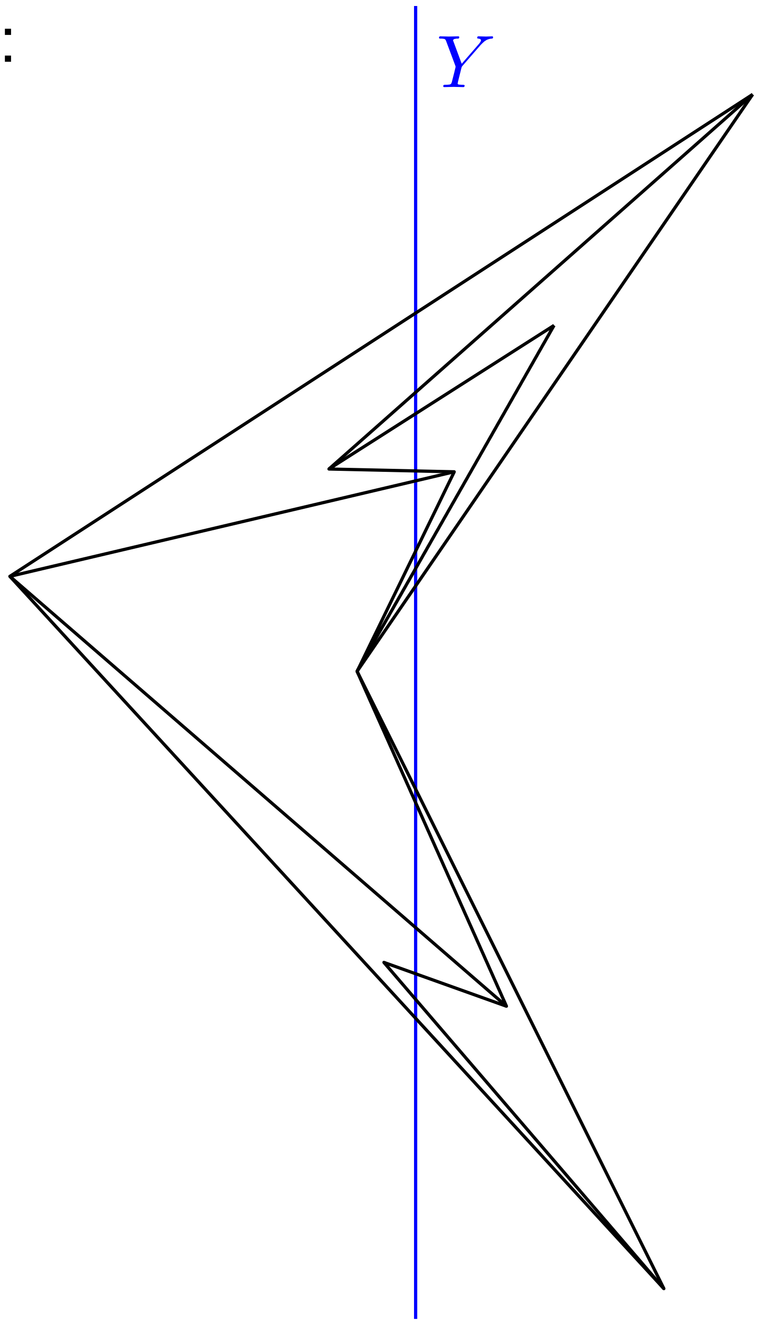
C. Induction on separating triangles

Quadrilateralizations with no point on Y

Freie Universität Berlin

(i.0) An even more special special case:

- 1. Every edge crosses Y.
- 2. No vertex lies on Y.
- *Edge* e_i should intersect Y at b_i.

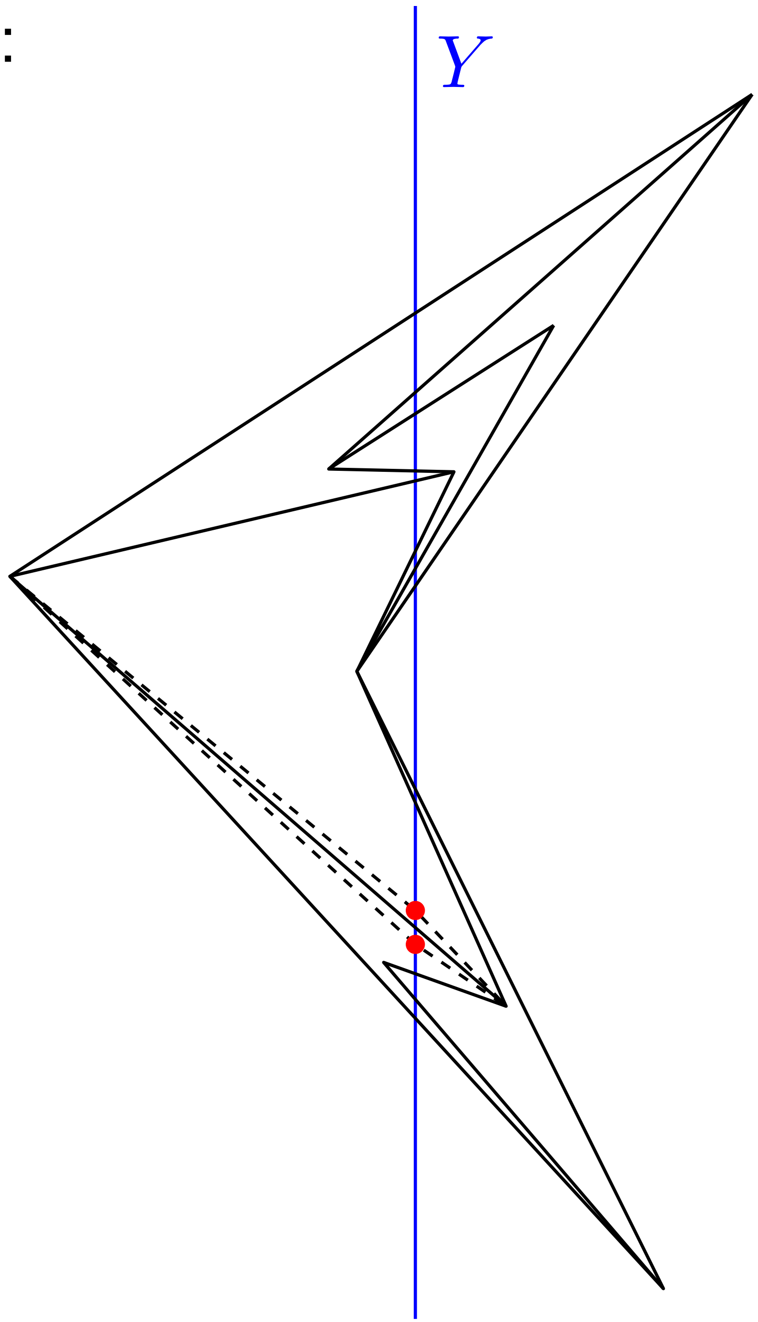


Quadrilateralizations with no point on Y

(i.0) An even more special special case:

- 1. Every edge crosses Y.
- 2. No vertex lies on Y.
- *Edge* e_i should intersect Y at b_i.

  Think of two close points on Y straddling e_i.



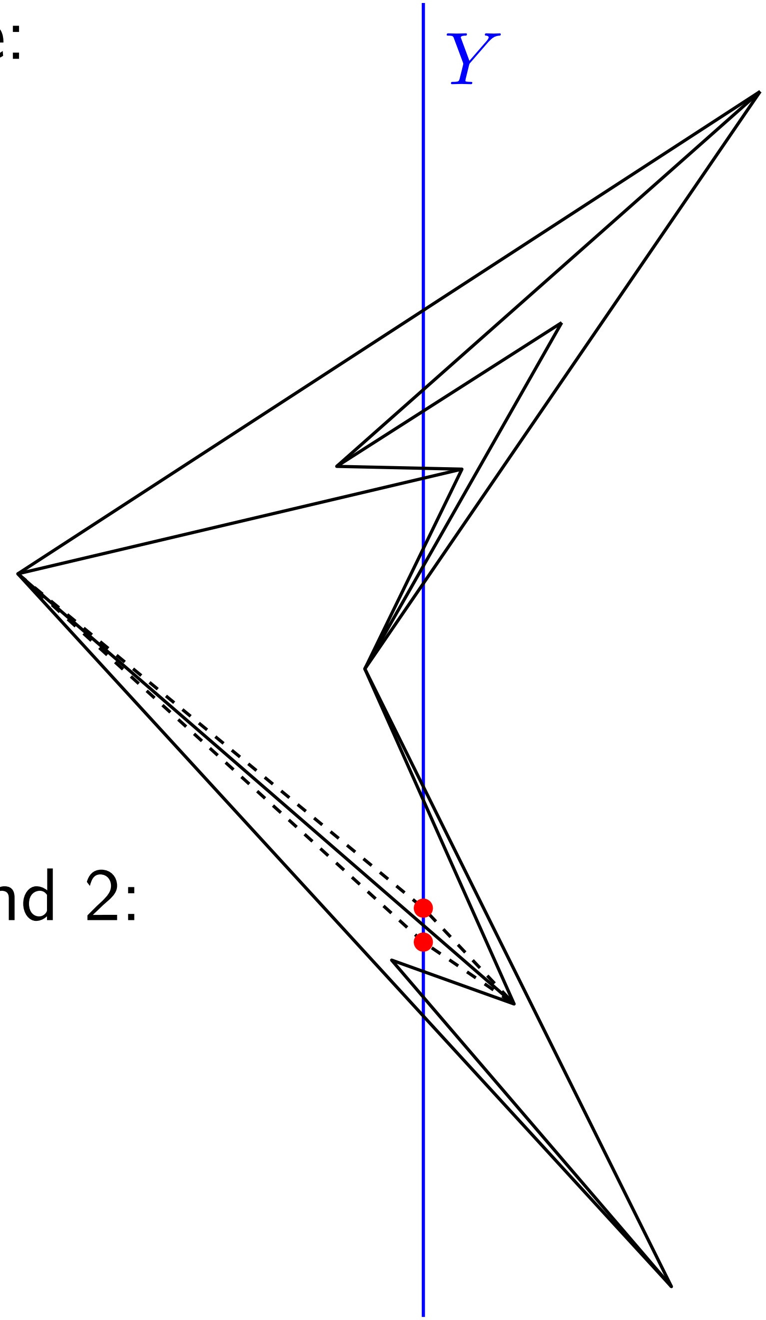
Quadrilateralizations with no point on Y

(i.0) An even more special special case:

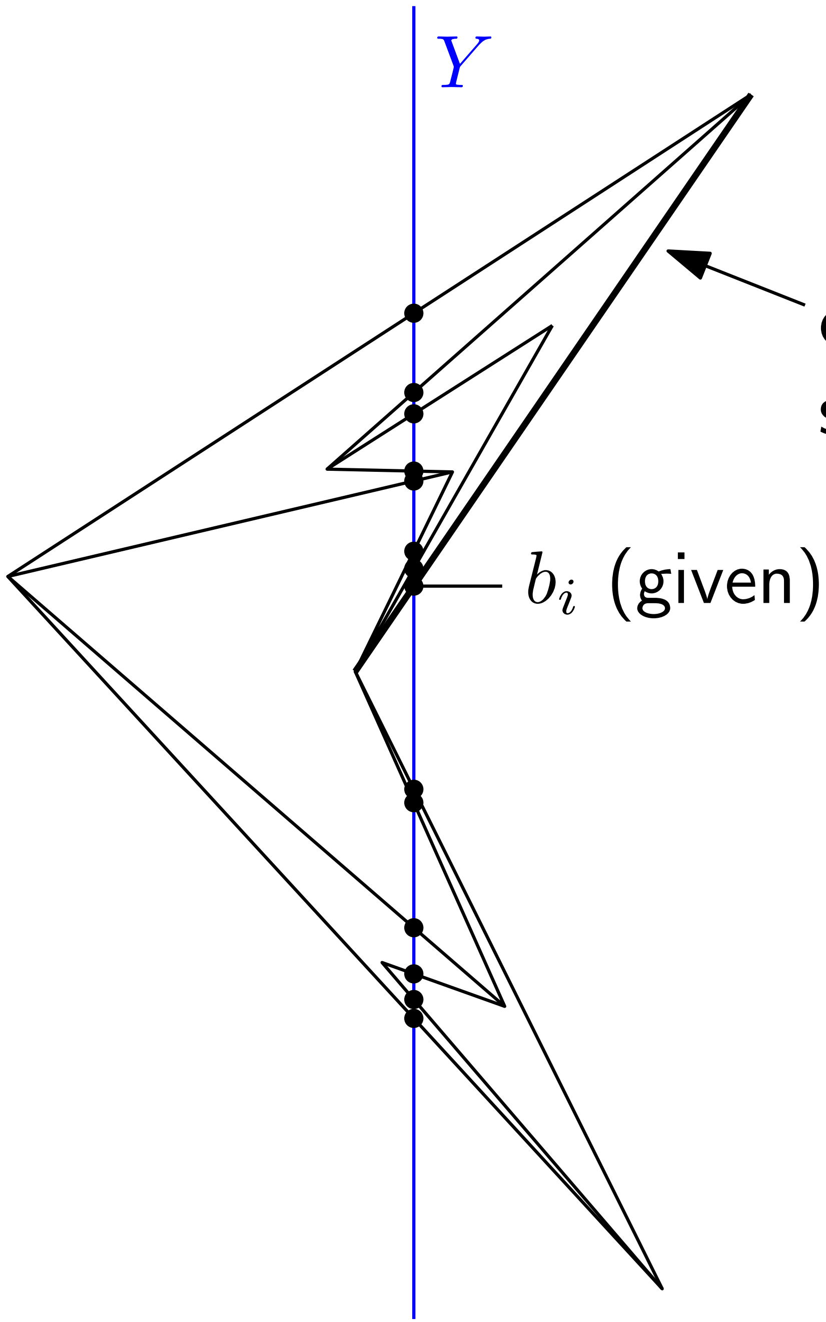
- 1. Every edge crosses Y.
- 2. No vertex lies on Y.
- *Edge* e_i should intersect Y at b_i.

  Think of two close points on Y straddling e_i.

- maximal graph with properties 1 and 2:
  $\rightarrow$ quadrilateralization.



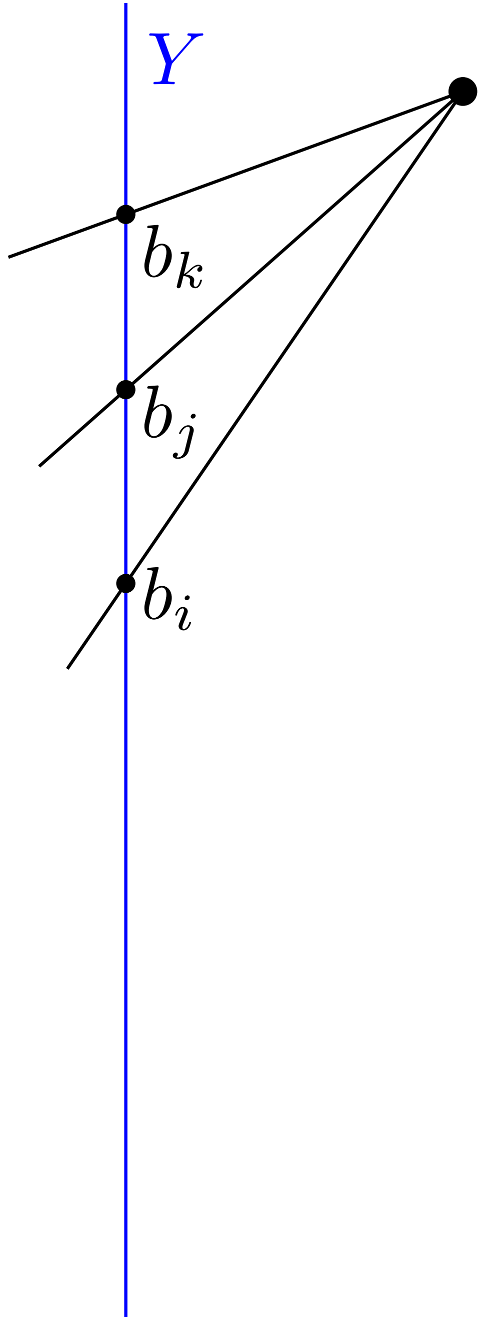
Parameterization by slope



edge e_i: $y = b_i + s_i x$
slope $= s_i$ (unknown)

m edges $e_1, \ldots, e_m$
Given: $b_1 < b_2 < \cdots < b_m$
m variables: $s_1, \ldots, s_m$

Concurrency constraints



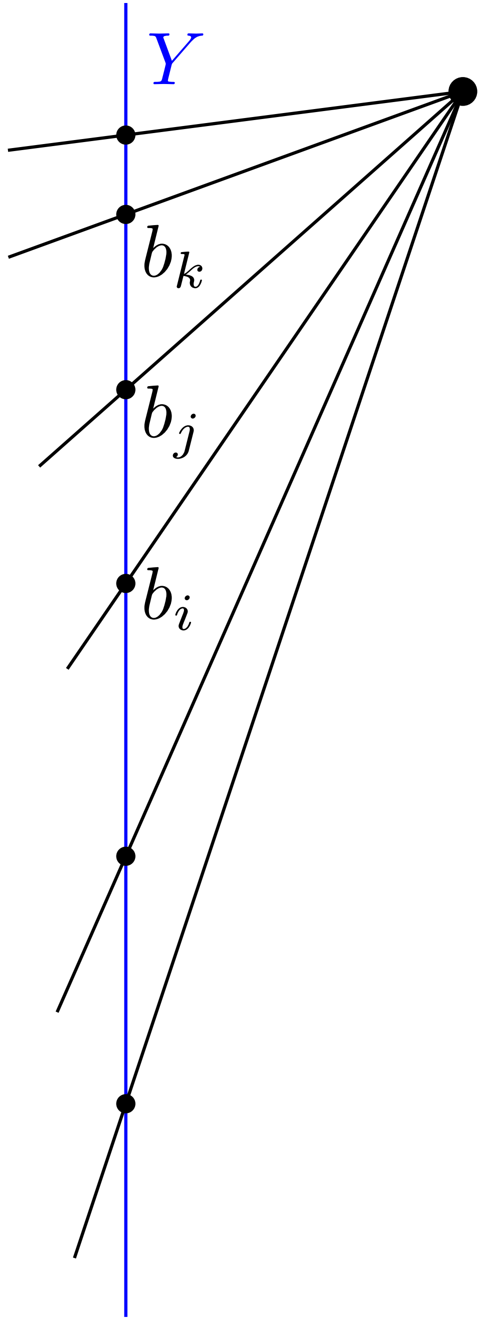
$$\begin{vmatrix} 1 & 1 & 1 \\ b_i & b_j & b_k \\ s_i & s_j & s_k \end{vmatrix} = 0$$

$$s_i(b_k - b_j) + s_j(b_i - b_k) + s_k(b_j - b_i) = 0$$

a *linear* equation in s_i, s_j, s_k

Concurrency constraints

Freie Universität Berlin



$$\begin{vmatrix} 1 & 1 & 1 \\ b_i & b_j & b_k \\ s_i & s_j & s_k \end{vmatrix} = 0$$

$$s_i(b_k - b_j) + s_j(b_i - b_k) + s_k(b_j - b_i) = 0$$

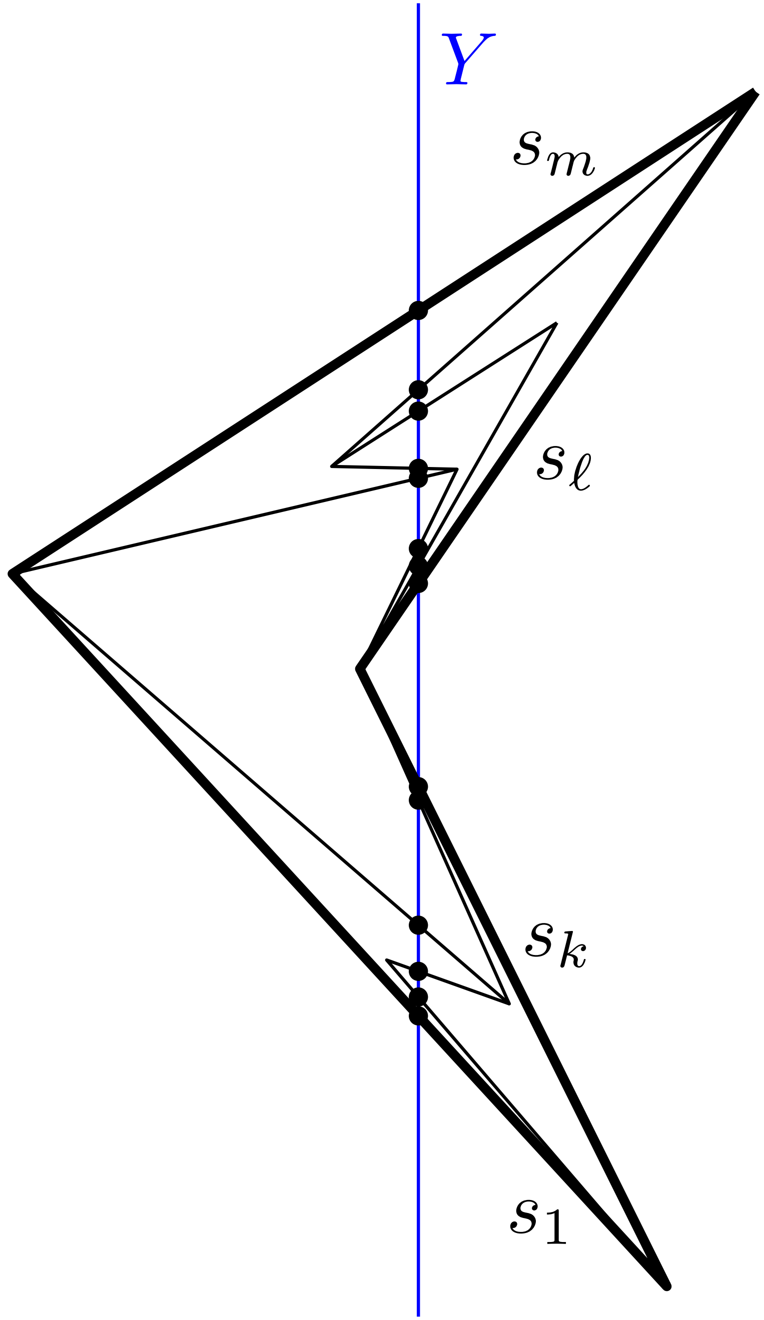
a *linear* equation in s_i, s_j, s_k

A vertex of degree d gives rise to $d - 2$ concurrency constraints.

Euler's formula
$\rightarrow m - 4$ equations in m variables

4 boundary equations to the rescue!

Boundary constraints

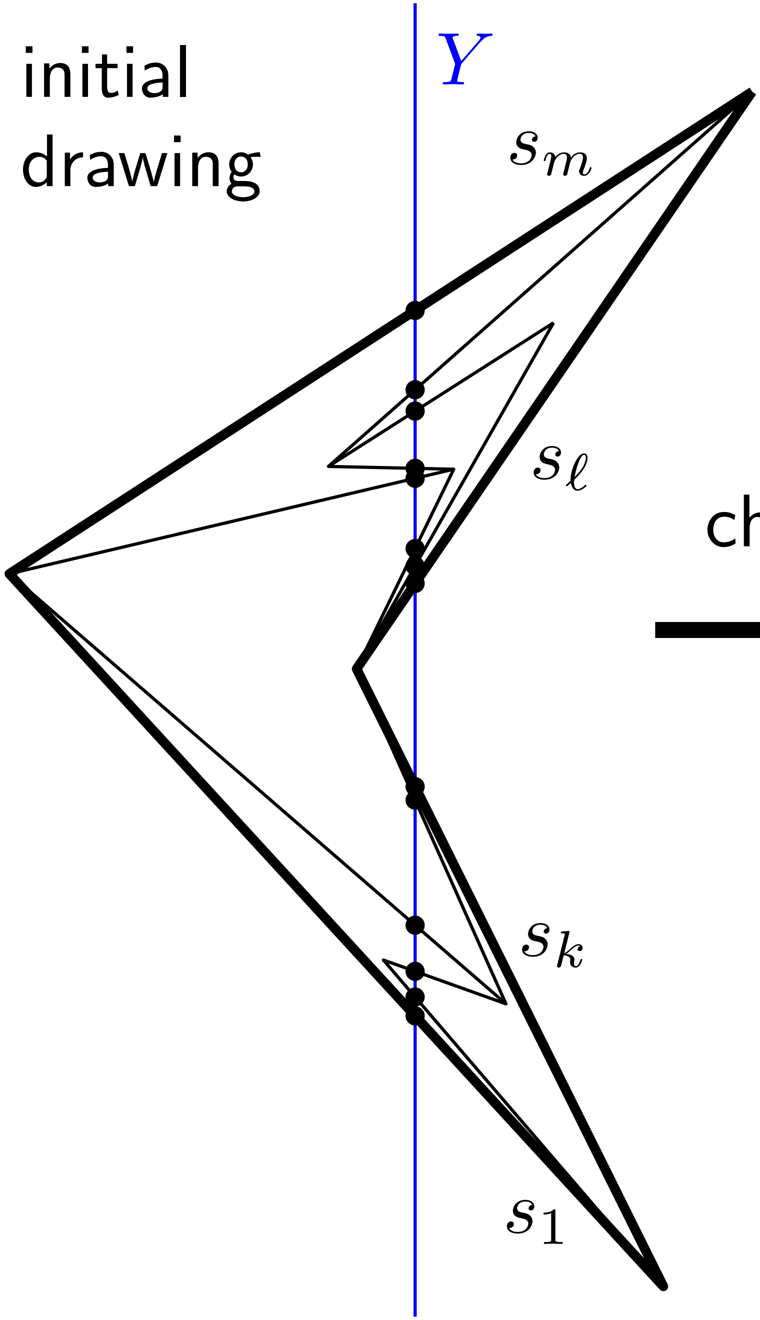


Set the 4 slopes s_1, s_k, s_ℓ, s_m of the outer boundary to fixed values.

$\rightarrow$ m equations in m variables

Morphing to the target solution

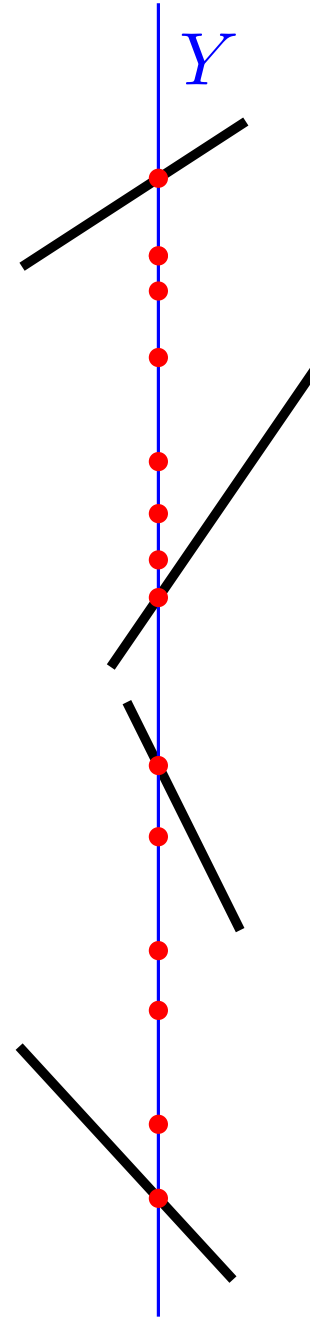
initial
drawing



change b_i continuously

follow the solution
$(s_1, \ldots, s_m)$ of the
linear system

target
drawing

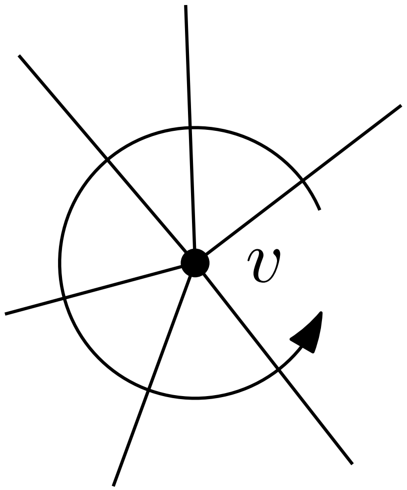


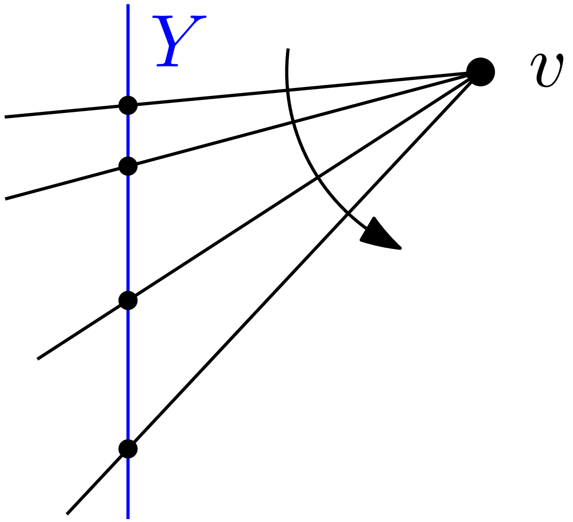
Solution of the linear system

- The solution $(s_1, \ldots, s_m)$ might not exist.
- The solution $(s_1, \ldots, s_m)$ might not be unique.
- The drawing might have crossings. ←

LEMMA: In a *straight-line drawing*, if
- the cyclic order of edges around each vertex agrees with a given planar drawing,
- and every face is a non-crossing polygon,

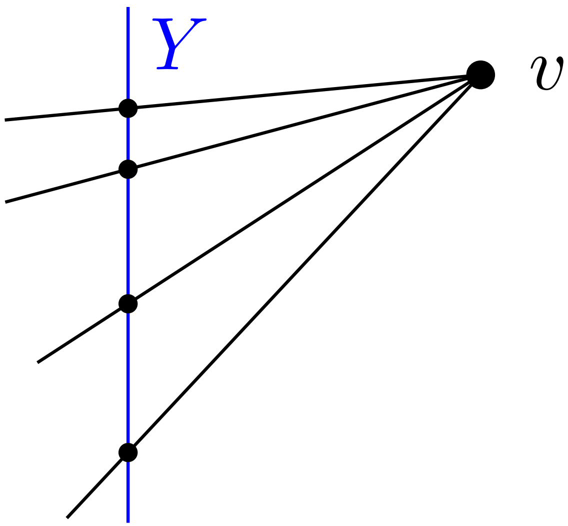
then the edges do not cross.





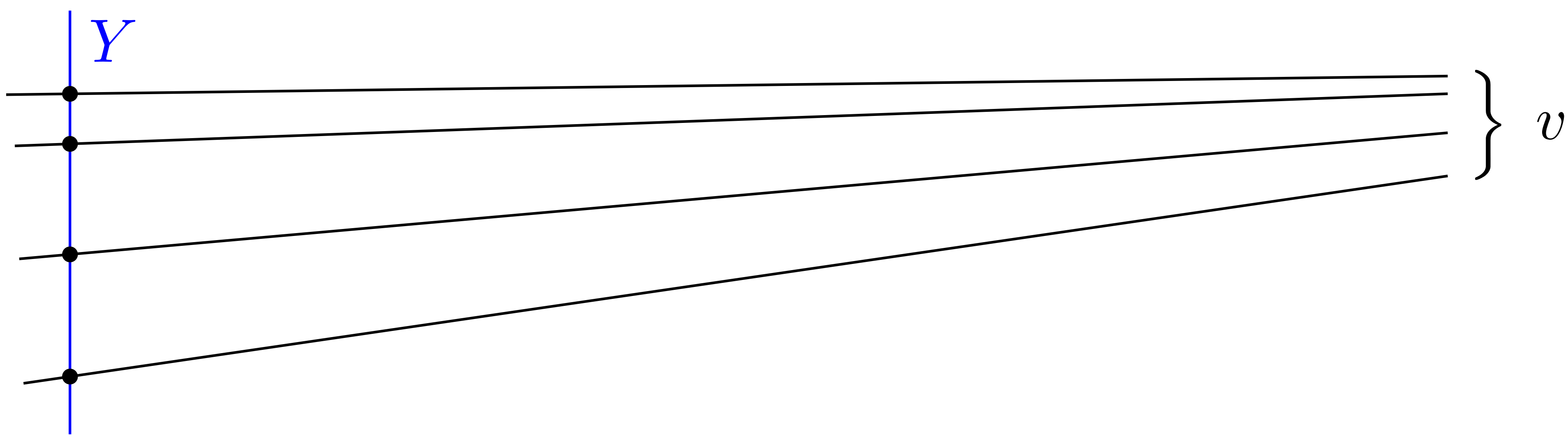
The cyclic order is correct because the order of b_i is fixed,
EXCEPT if the edges intersect on the *other side* of Y.

Flipping to the other side?

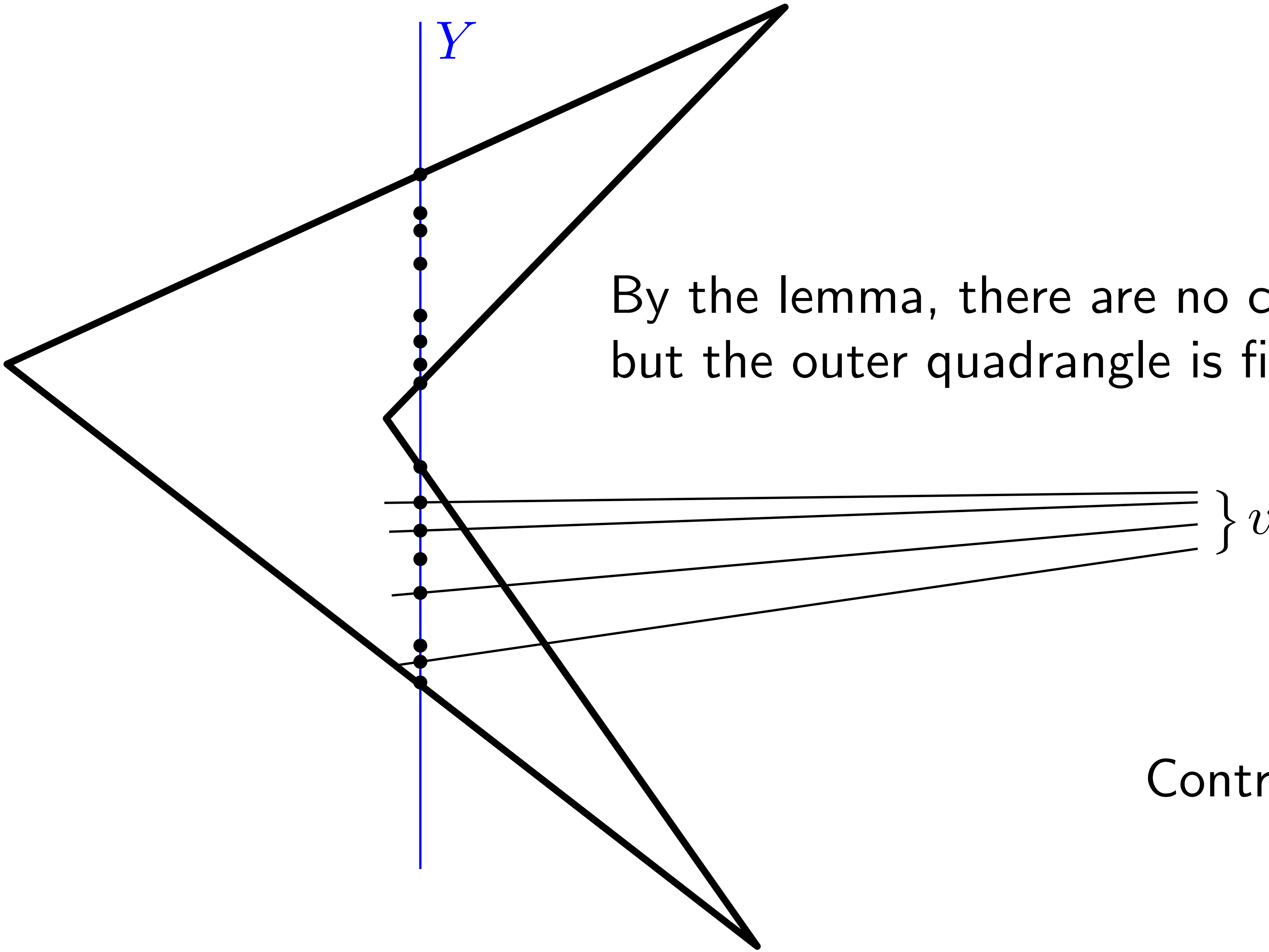


Continuity argument:

A vertex v that flips to the other side must first run off to infinity.



Flipping to the other side?



By the lemma, there are no crossings, but the outer quadrangle is fixed.

Contradiction.

Solution of the linear system

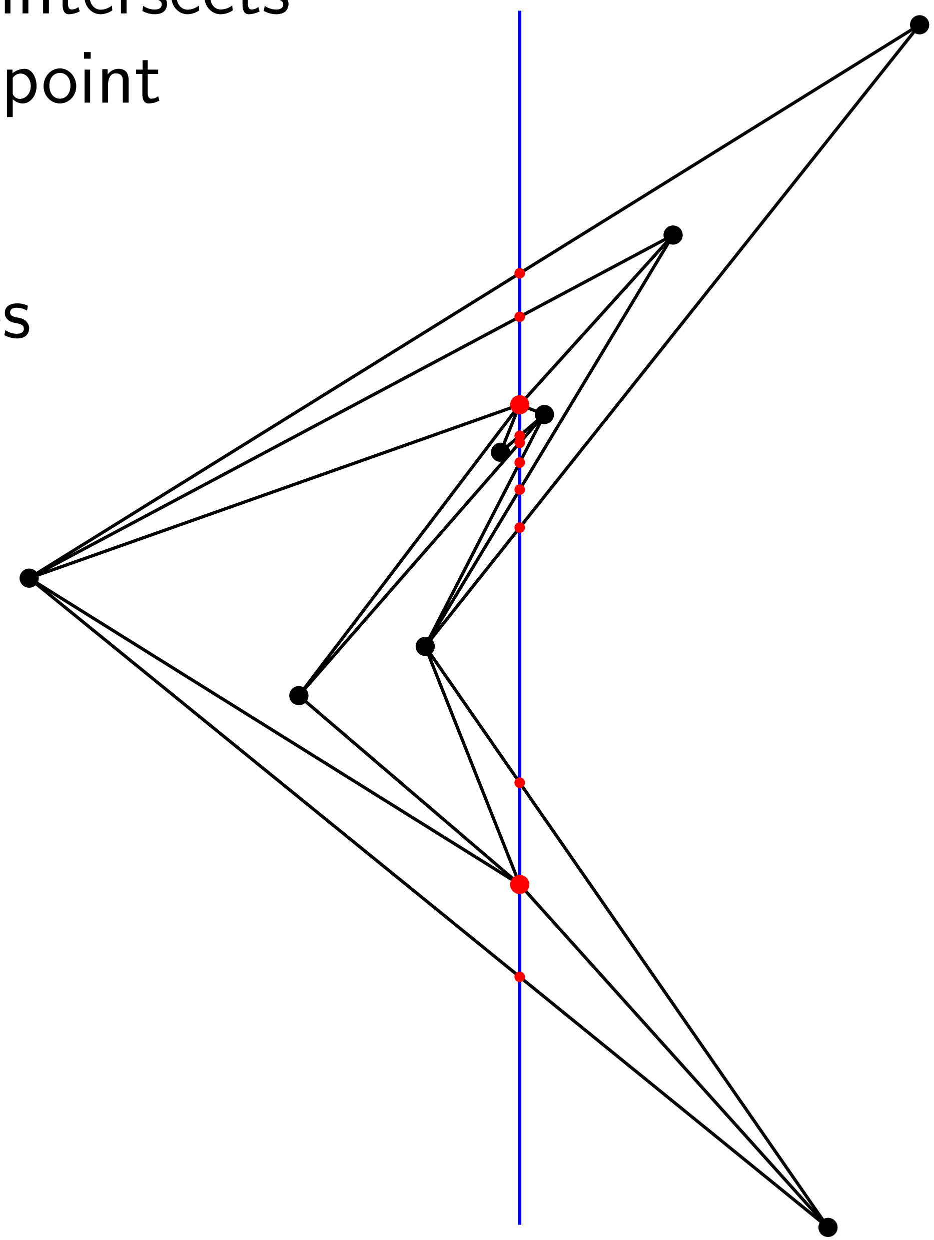
Existence and uniqueness are established with similar arguments.

- The solution $(s_1, \ldots, s_m)$ might not exist.
- The solution $(s_1, \ldots, s_m)$ might not be unique.
- The drawing might have crossings. ✓

(i) Extension to points on Y

- maximal graphs such that Y intersects every edge, at least in an endpoint (so-called "A-graphs")

$\rightarrow$ triangle and quadrilateral faces



Add some artificial *proportionality constraints*, in order to get m equations in m variables.

(Ensure at the same time the correct cyclic order)

(ii) Edges that don't intersect Y

Freie Universität Berlin

Use several types of reductions

A. a) *contract* xy into a single vertex v

b) draw the resulting graph

c) perturb v into two vertices with an edge between

B. *flip* the edge xy

C. Induction on separating triangles