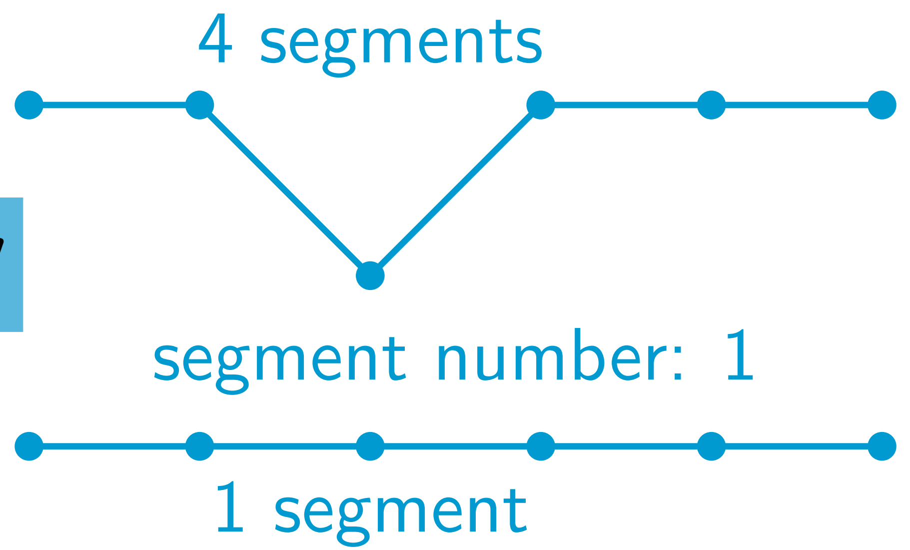
The Parametrized Complexity of the Segment Number



Sabine Cornelsen
Konstanz, Germany

Giordano Da Lozzo
Roma III, Italy

Luca Grilli
Perugia, Italy

Siddharth Gupta
Warwick, UK

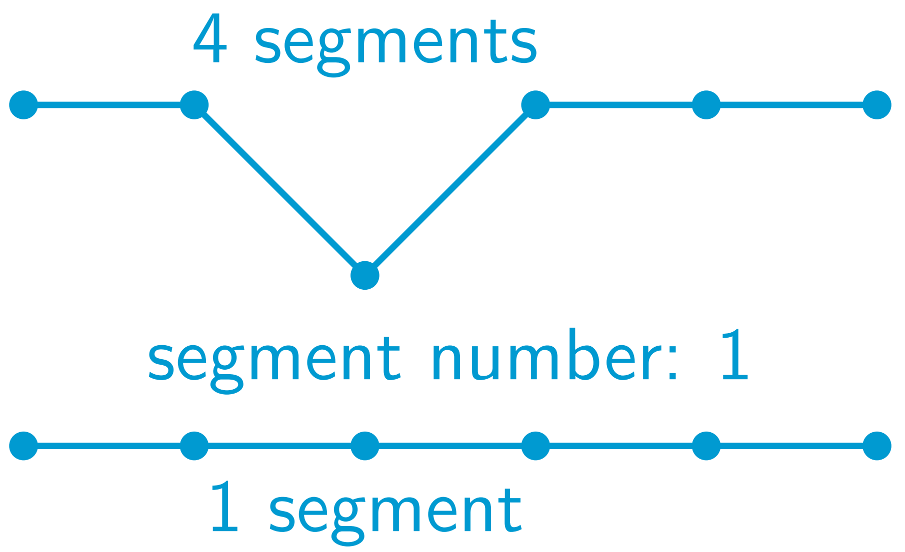
Jan Kratochvíl
Prague, Czech Republic
Charles University

Alexander Wolff
Würzburg, Germany

Symposium on Graph Drawing and Network Visualization, GD 2023, Isola delle Femmine

Basic Definitions

segment = maximal set of edges forming a line segment

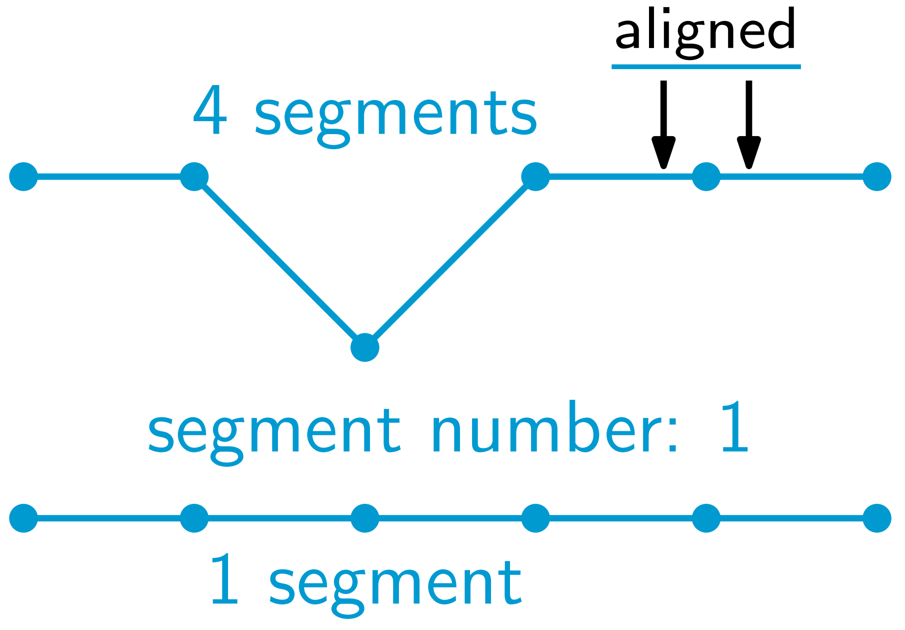


segment number seg(G) of a planar graph G:
minimum number of segments
in any planar straight-line drawing of G

Dujmović, Eppstein, Suderman, Wood (CGTA'07)

Basic Definitions

segment $=$ maximal set of edges forming a line segment



segment number seg(G) of a planar graph G:
minimum number of segments
in any planar straight-line drawing of G
Dujmović, Eppstein, Suderman, Wood (CGTA'07)

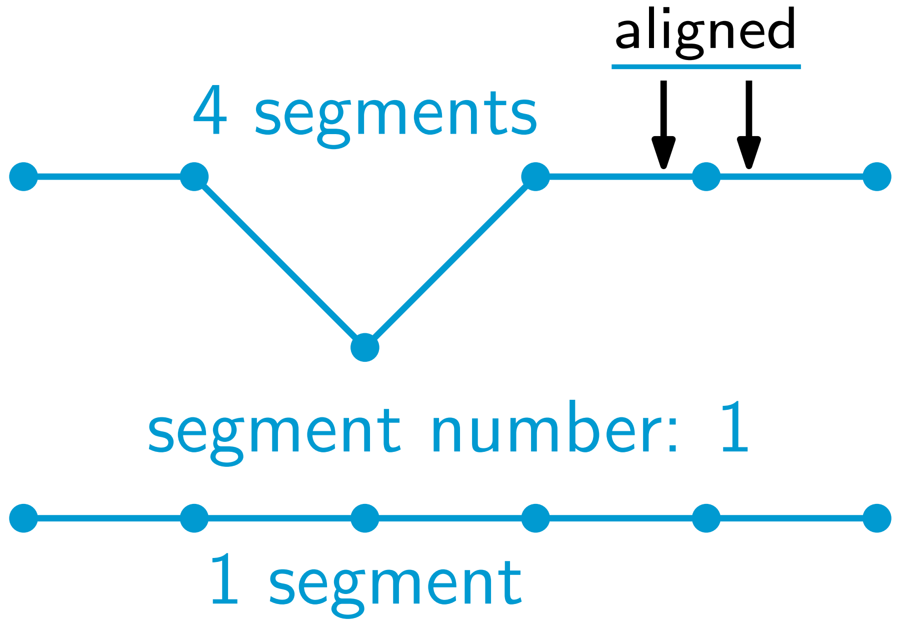
minimizing number of segments
$\hat{=}$
maximizing number of alignments

Basic Definitions

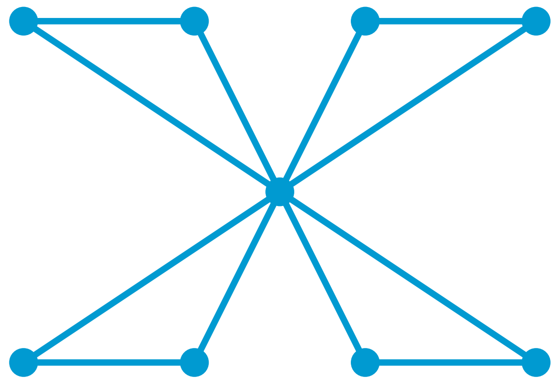
segment $=$ maximal set of edges forming a line segment

segment number seg(G) of a planar graph G:
minimum number of segments
in any planar straight-line drawing of G

Dujmović, Eppstein, Suderman, Wood (CGTA'07)



minimizing number of segments
$\hat{=}$
maximizing number of alignments



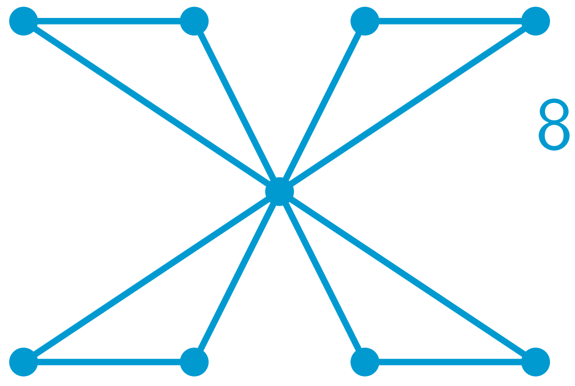
Basic Definitions

segment = maximal set of edges forming a line segment

segment number seg(G) of a planar graph G:
minimum number of segments
in any planar straight-line drawing of G

Dujmović, Eppstein, Suderman, Wood (CGTA'07)

minimizing number of segments
$\hat{=}$
maximizing number of alignments



Basic Definitions

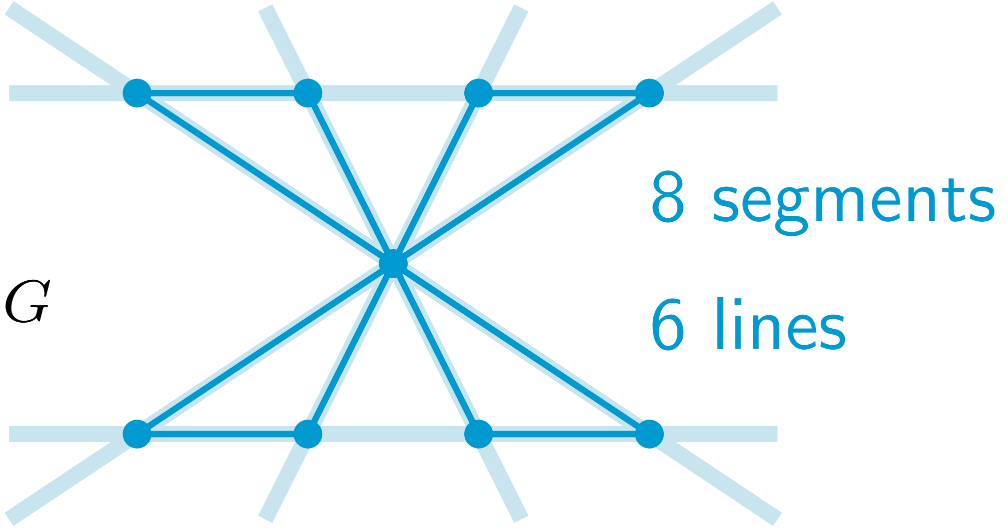
segment $=$ maximal set of edges forming a line segment

segment number $\text{seg}(G)$ of a planar graph G:
minimum number of segments
in any planar straight-line drawing of G
Dujmović, Eppstein, Suderman, Wood (CGTA'07)

segment number: 1

minimizing number of segments $\hat{=}$ maximizing number of alignments

line cover number $\text{line}(G)$ of a planar graph G:
minimum number lines supporting all the edges
in any planar straight-line drawing of G
Chaplick et al. (GD'16)

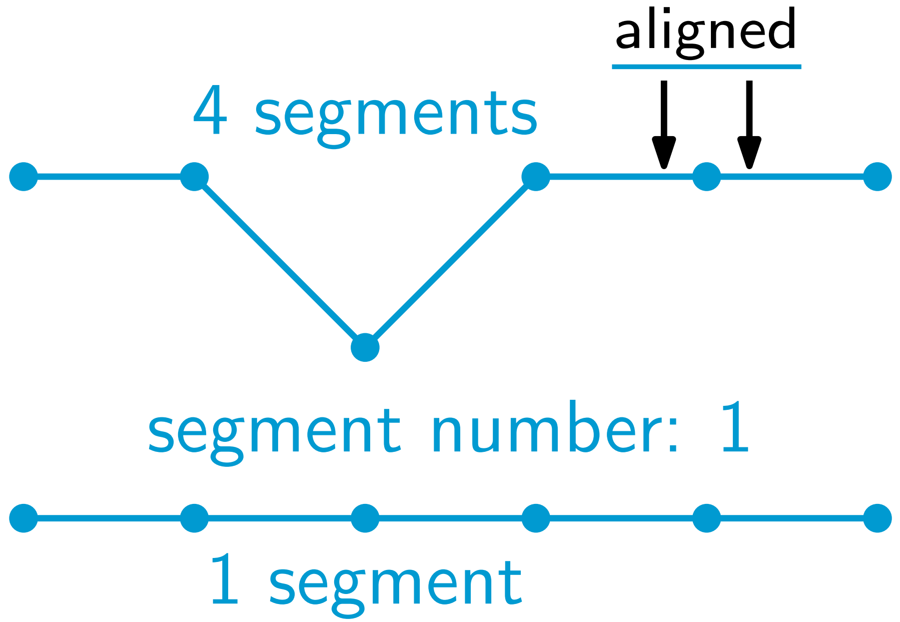


Basic Definitions

segment $=$ maximal set of edges forming a line segment

segment number $\mathrm{seg}(G)$ of a planar graph G:
minimum number of segments
in any planar straight-line drawing of G

Dujmović, Eppstein, Suderman, Wood (CGTA'07)



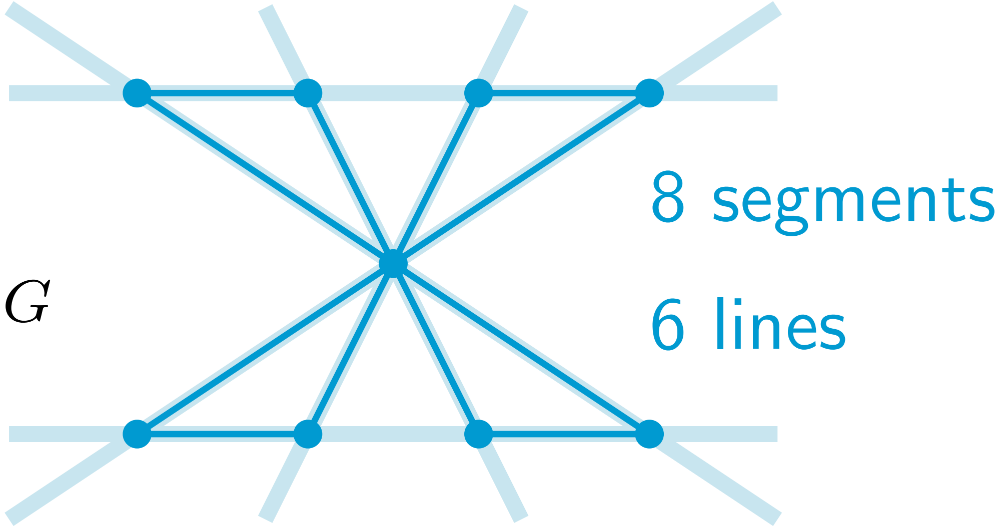
> minimizing number of segments
> $\hat{=}$
> maximizing number of alignments

respective optimization problem: Segment Number

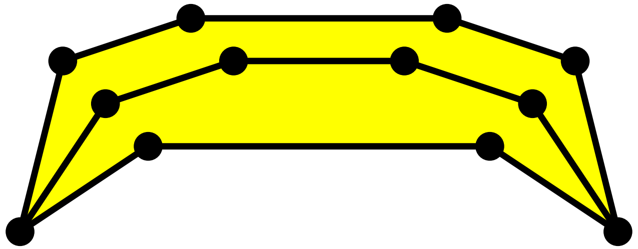
line cover number $\mathrm{line}(G)$ of a planar graph G:
minimum number lines supporting all the edges
in any planar straight-line drawing of G

Chaplick et al. (GD'16)

respective minimization problem: Line Cover Number



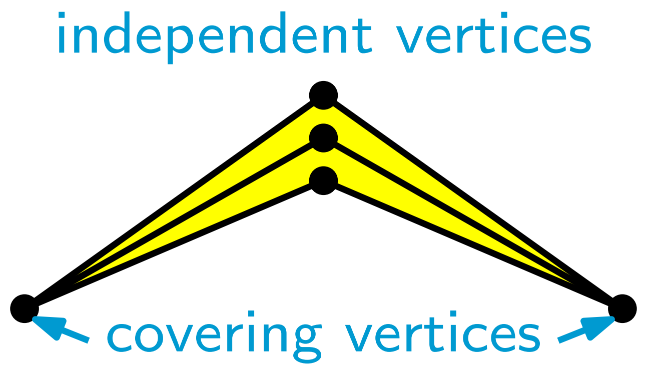
Warm-Up: Banana-Trees, and -Cycles



banana: Scott/Seymour 2020

union of internally disjoint paths with common endpoints

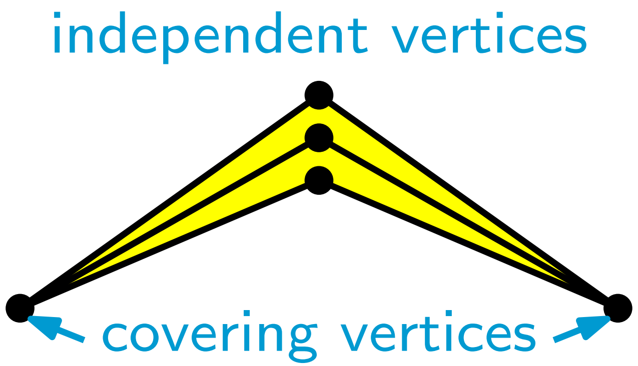
Warm-Up: Banana-Trees, and -Cycles



banana: (Scott/Seymour 2020)

union of internally disjoint paths of length two with common endpoints

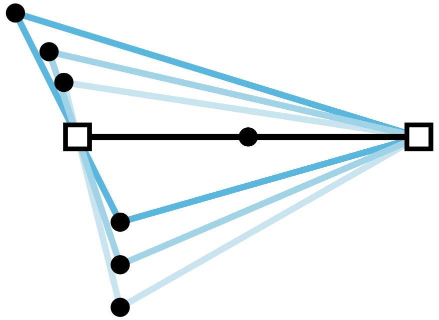
Warm-Up: Banana-Trees, and -Cycles



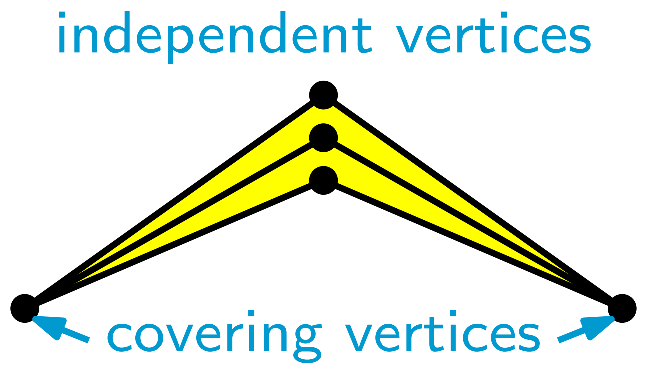
banana: (Scott/Seymour 2020)
union of internally disjoint paths
of length two with common endpoints

Observation: Dujmović, Eppstein, Suderman, Wood '07

A banana with k parallel paths of length two has segment number $\lfloor 3k/2 \rfloor$.



Warm-Up: Banana-Trees, and -Cycles

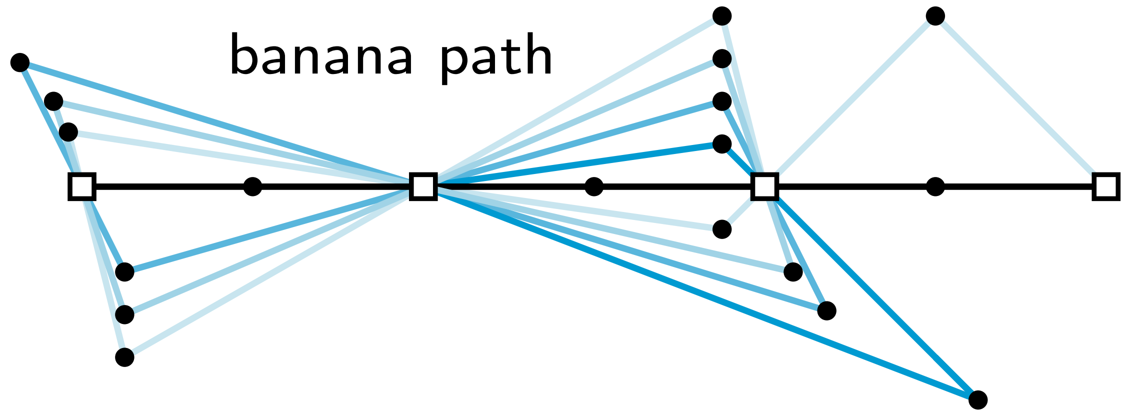


banana tree Scott/Seymour 2020
tree where each edge is replaced by a banana.

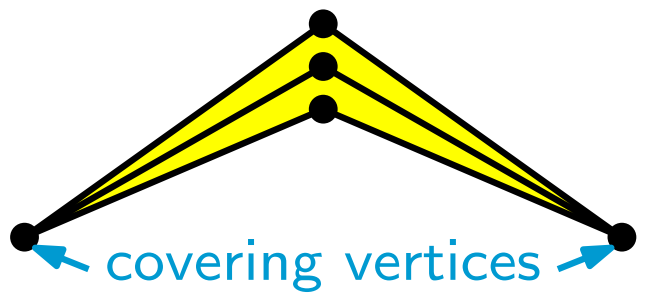
banana: (Scott/Seymour 2020)
union of internally disjoint paths
of length two with common endpoints

Observation: Dujmović, Eppstein, Suderman, Wood '07

A banana with k parallel paths of length two has segment number $\lfloor 3k/2 \rfloor$.



Warm-Up: Banana-Trees, and -Cycles

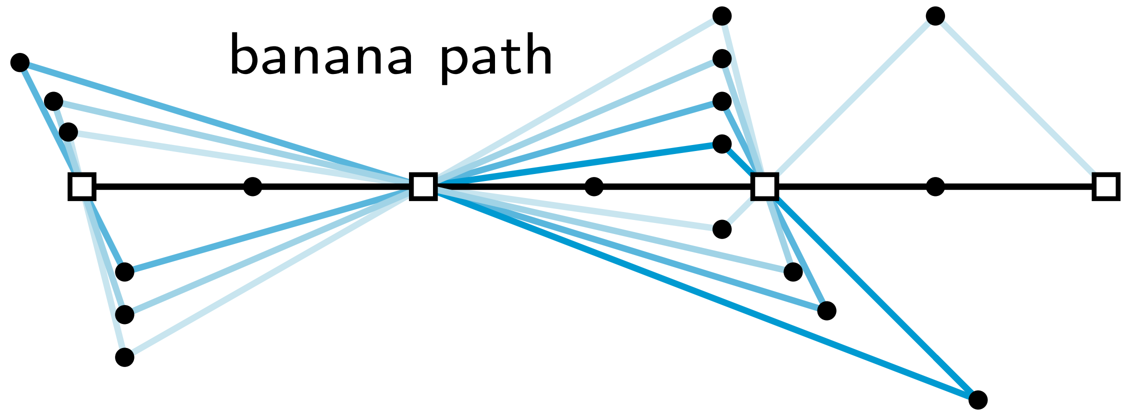


banana tree Scott/Seymour 2020
tree where each edge is replaced by a banana.

banana: (Scott/Seymour 2020)
union of internally disjoint paths
of length two with common endpoints

Observation: Dujmović, Eppstein, Suderman, Wood '07

A banana with k parallel paths of length two has segment number $\lfloor 3k/2 \rfloor$.

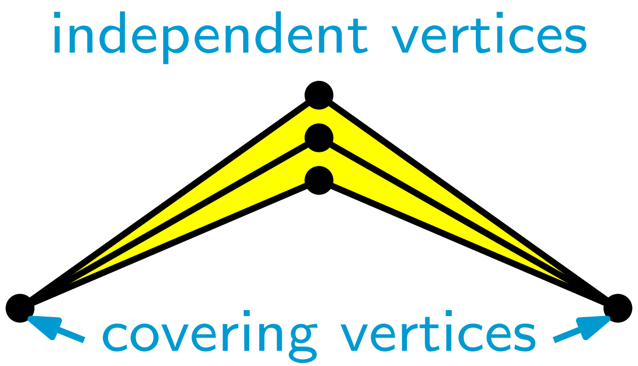


Theorem:

The segment number of a banana tree can be determined in linear time.

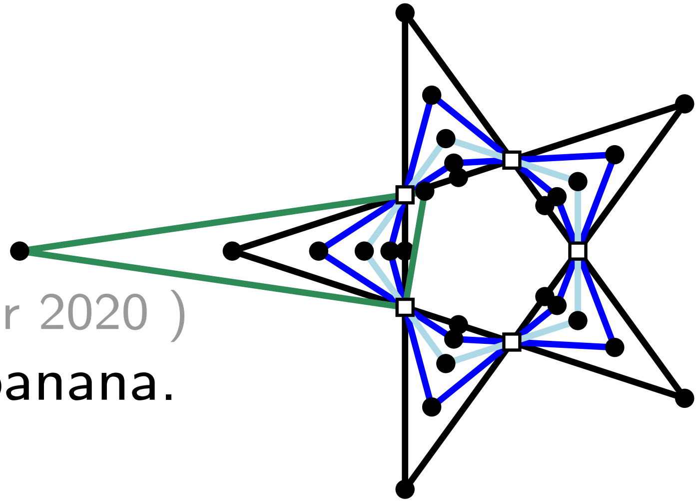
- align as many edges as possible with other bananas,
- the (larger) remainder with the same banana

Warm-Up: Banana-Trees, and -Cycles



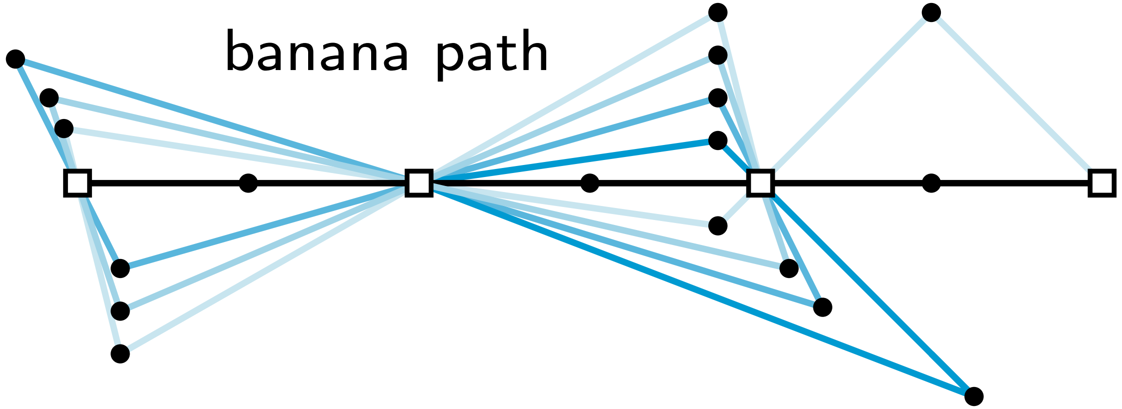
banana: (Scott/Seymour 2020)
union of internally disjoint paths
of length two with common endpoints

banana tree (cycle) (Scott/Seymour 2020)
tree where each edge is replaced by a banana.
(simple cycle)



Observation: Dujmović, Eppstein, Suderman, Wood ’07

A banana with k parallel paths of length two has segment number $\lfloor 3k/2 \rfloor$.

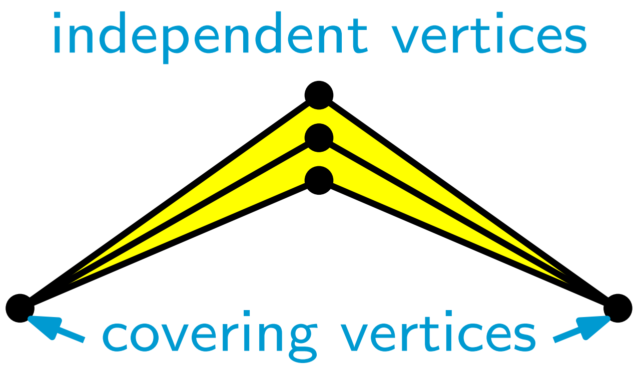


Theorem:

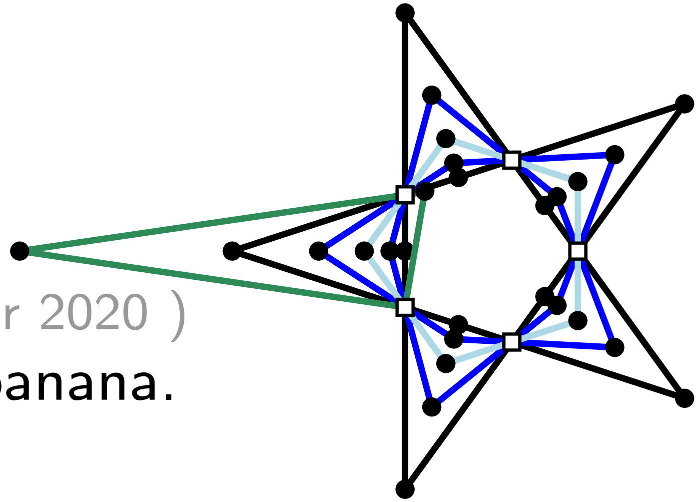
The segment number of a banana tree can be determined in linear time.

- align as many edges as possible with other bananas,
- the (larger) remainder with the same banana

Warm-Up: Banana-Trees, and -Cycles



banana tree (cycle) (Scott/Seymour 2020)
tree where each edge is replaced by a banana.
(simple cycle)



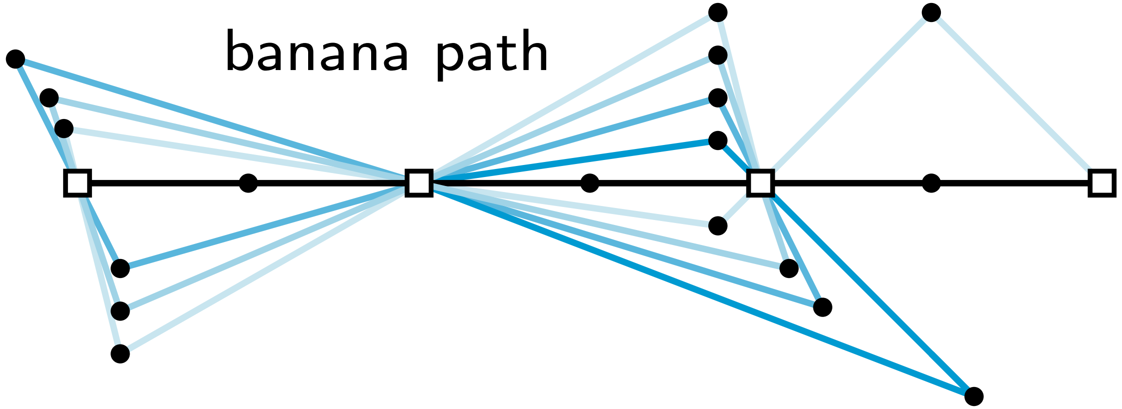
banana: (Scott/Seymour 2020)
union of internally disjoint paths
of length two with common endpoints

Theorem:

The segment number of a banana cycle
of length at least five and with at least two independent vertices per banana
can be determined in linear time.

Observation: Dujmović, Eppstein, Suderman, Wood '07

A banana with k parallel paths of length two
has segment number $\lfloor 3k/2 \rfloor$.



Theorem:

The segment number of a banana tree
can be determined in linear time.

- align as many edges as possible with other bananas,
- the (larger) remainder with the same banana

Related Work

The segment number was defined by Dujmović, Eppstein, Suderman, Wood (CGTA 2007)

Related Work

The segment number was defined by Dujmović, Eppstein, Suderman, Wood (CGTA 2007)

SEGMENT NUMBER is $\exists\mathbb{R}$-complete [ORW-GD'19], NP-hard for fixed embedding [DMNW-JGAA'17]

Related Work

The segment number was defined by Dujmović, Eppstein, Suderman, Wood (CGTA 2007)

SEGMENT NUMBER is $\exists\mathbb{R}$-complete [ORW-GD'19], NP-hard for fixed embedding [DMNW-JGAA'17]

SEGMENT NUMBER is in $\mathcal{P}$ for

- trees [DESW-CGTA'07]
- series-parallel graphs with deg ≤ 3 [SAAR-GD'08]
- subdivisions of outerplanar paths [Adnan'08]
- 3-connected cubic planar graphs [DESW-CGTA'07,IMS-JGAA'17]
- cacti [G5KWZ-WG'22]

Related Work

The segment number was defined by Dujmović, Eppstein, Suderman, Wood (CGTA 2007)

Segment Number is $\exists\mathbb{R}$-complete [ORW-GD'19], NP-hard for fixed embedding [DMNW-JGAA'17]

Segment Number is in $\mathcal{P}$ for

- trees [DESW-CGTA'07]
- series-parallel graphs with deg ≤ 3 [SAAR-GD'08]
- subdivisions of outerplanar paths [Adnan'08]
- 3-connected cubic planar graphs [DESW-CGTA'07,IMS-JGAA'17]
- cacti [G5KWZ-WG'22]

Bounds for various graph classes, e.g.,

- outerplanar graphs, 2-trees, planar 3-trees, 3-connected plane graphs [DESW-CGTA'07]
- (4-connected) triangulations [DM-CGTA'19]
- triconnected planar 4-regular graphs [G5KWZ-WG'22]

Related Work

The segment number was defined by Dujmović, Eppstein, Suderman, Wood (CGTA 2007)

SEGMENT NUMBER is $\exists\mathbb{R}$-complete [ORW-GD'19], NP-hard for fixed embedding [DMNW-JGAA'17]

SEGMENT NUMBER is in $\mathcal{P}$ for

- trees [DESW-CGTA'07]
- series-parallel graphs with deg ≤ 3 [SAAR-GD'08]
- subdivisions of outerplanar paths [Adnan'08]
- 3-connected cubic planar graphs [DESW-CGTA'07,IMS-JGAA'17]
- cacti [G5KWZ-WG'22]

Bounds for various graph classes, e.g.,

- outerplanar graphs, 2-trees, planar 3-trees, 3-connected plane graphs [DESW-CGTA'07]
- (4-connected) triangulations [DM-CGTA'19]
- triconnected planar 4-regular graphs [G5KWZ-WG'22]

grid drawings [Schulz-JGAA'15, HKMS-JGAA'18, KMSS-GD'19]

user studies [KMS-GD'17]

Related Work

The segment number was defined by Dujmović, Eppstein, Suderman, Wood (CGTA 2007)

SEGMENT NUMBER is $\exists\mathbb{R}$-complete [ORW-GD'19], NP-hard for fixed embedding [DMNW-JGAA'17]

SEGMENT NUMBER is in $\mathcal{P}$ for

- trees [DESW-CGTA'07]
- series-parallel graphs with deg ≤ 3 [SAAR-GD'08]
- subdivisions of outerplanar paths [Adnan'08]
- 3-connected cubic planar graphs [DESW-CGTA'07,IMS-JGAA'17]
- cacti [G5KWZ-WG'22]

LINE COVER NUMBER is in FPT wrt. the natural parameter [CFLRVW-JGAA'23]

Recall: decision problem with input x, parameter k is fixed-parameter tractable (FPT) if solvable with run time $\mathcal{O}(f(k)|x|^c)$, c constant, f computable

Bounds for various graph classes, e.g.,

- outerplanar graphs, 2-trees, planar 3-trees, 3-connected plane graphs [DESW-CGTA'07]
- (4-connected) triangulations [DM-CGTA'19]
- triconnected planar 4-regular graphs [G5KWZ-WG'22]

grid drawings [Schulz-JGAA'15, HKMS-JGAA'18, KMSS-GD'19]

user studies [KMS-GD'17]

Renegar's Decision Algorithm (Renegar, 1992)

Given an existential first-order formula about the reals

$$\exists x_1 \ldots x_m \; \Phi(x_1, \ldots, x_m)$$

(Φ: Boolean combination of equalities and inequalities of polynomials over $\mathbb{Q}$)
it can be decided in time exponentially in m whether the formula is realizable over the reals.

basis depends on size and degree of polynomials

Renegar's Decision Algorithm (Renegar, 1992)

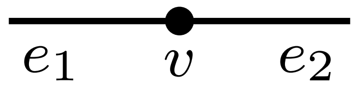
Given an existential first-order formula about the reals

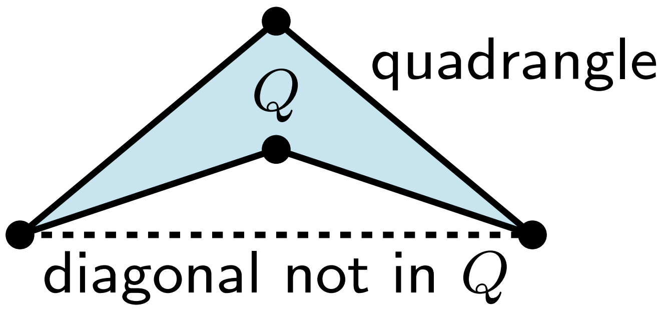
$$\exists x_1 \ldots x_m \; \Phi(x_1, \ldots, x_m)$$

(Φ: Boolean combination of equalities and inequalities of polynomials over $\mathbb{Q}$)
it can be decided in time exponentially in m whether the formula is realizable over the reals.
basis depends on size and degree of polynomials

It can be expressed as an existential first-order formula about the reals whether there is a set of points in the plane

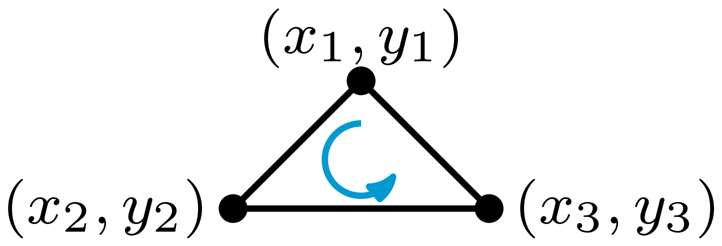
- that is a straight-line planar drawing of a plane graph, (CFLRVW-JGAA'23)
- given pairs of edges are aligned
- given quadrangles are not convex





$$\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} > 0$$

iff



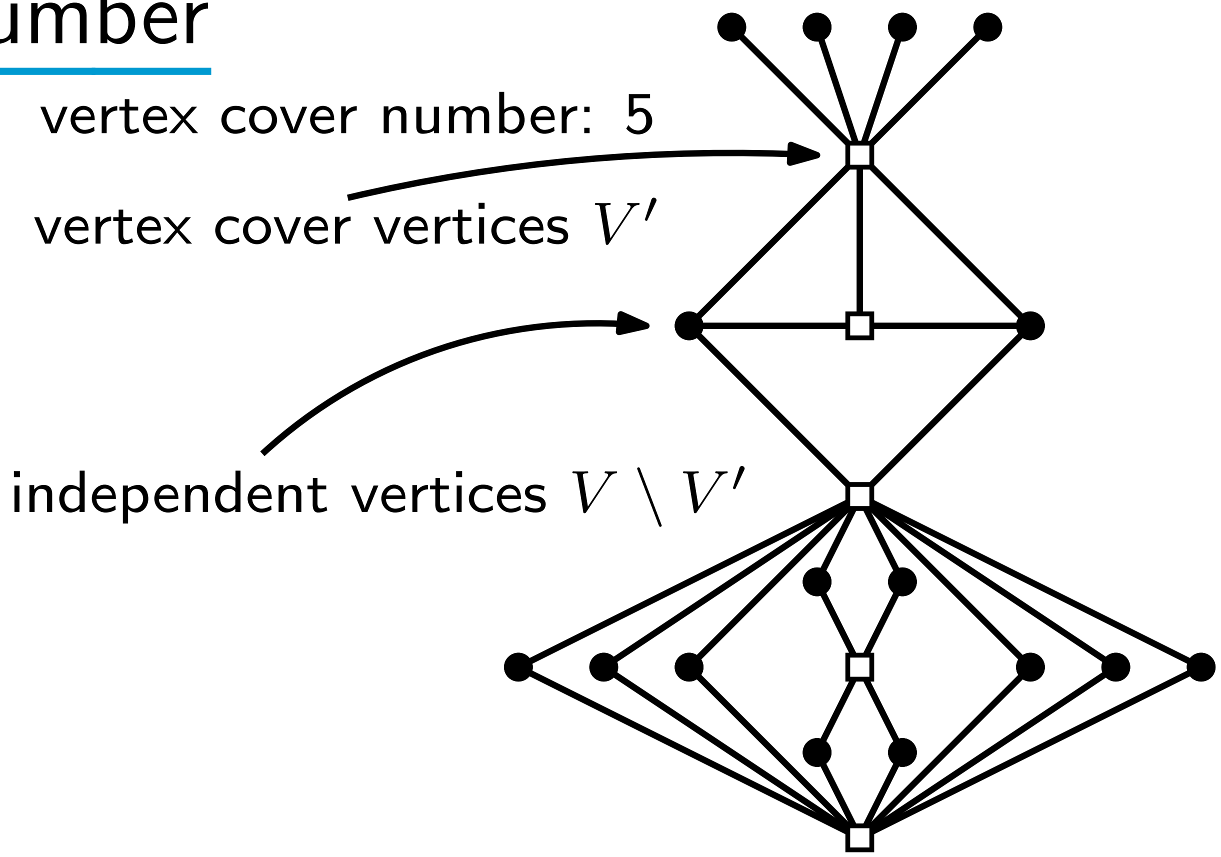
(=0 iff collinear)

$\rightsquigarrow$ $\mathcal{O}(|V|^{\mathcal{O}(|V|)})$ algorithm for Segment Number

SEGMENT NUMBER by Vertex Cover Number

vertex cover of a graph $G = (V, E)$: set $V' \subseteq V$ s.t. $e \cap V' \neq \emptyset$ for each $e \in E$

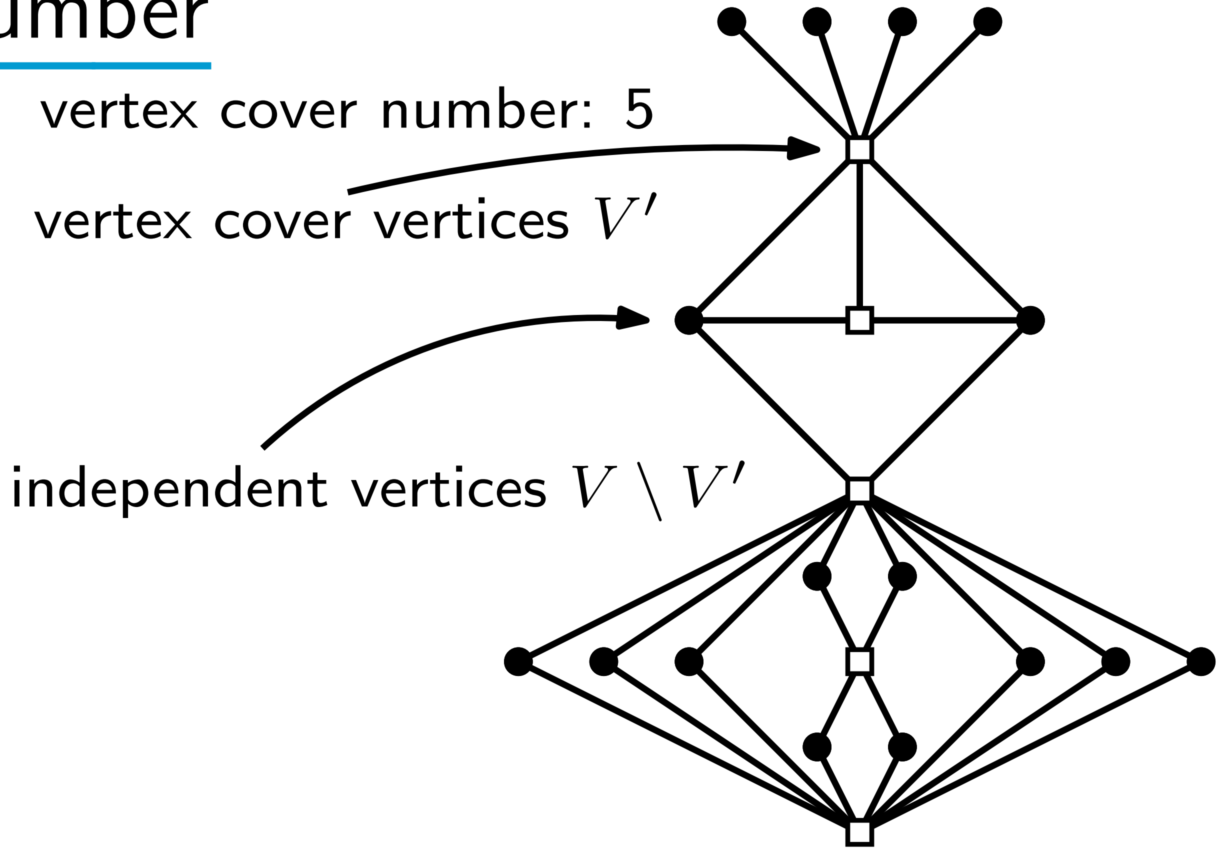
vertex cover number of a graph: size of its smallest vertex cover



Segment Number by Vertex Cover Number

vertex cover of a graph $G = (V, E)$: set $V' \subseteq V$ s.t. $e \cap V' \neq \emptyset$ for each $e \in E$

vertex cover number of a graph: size of its smallest vertex cover

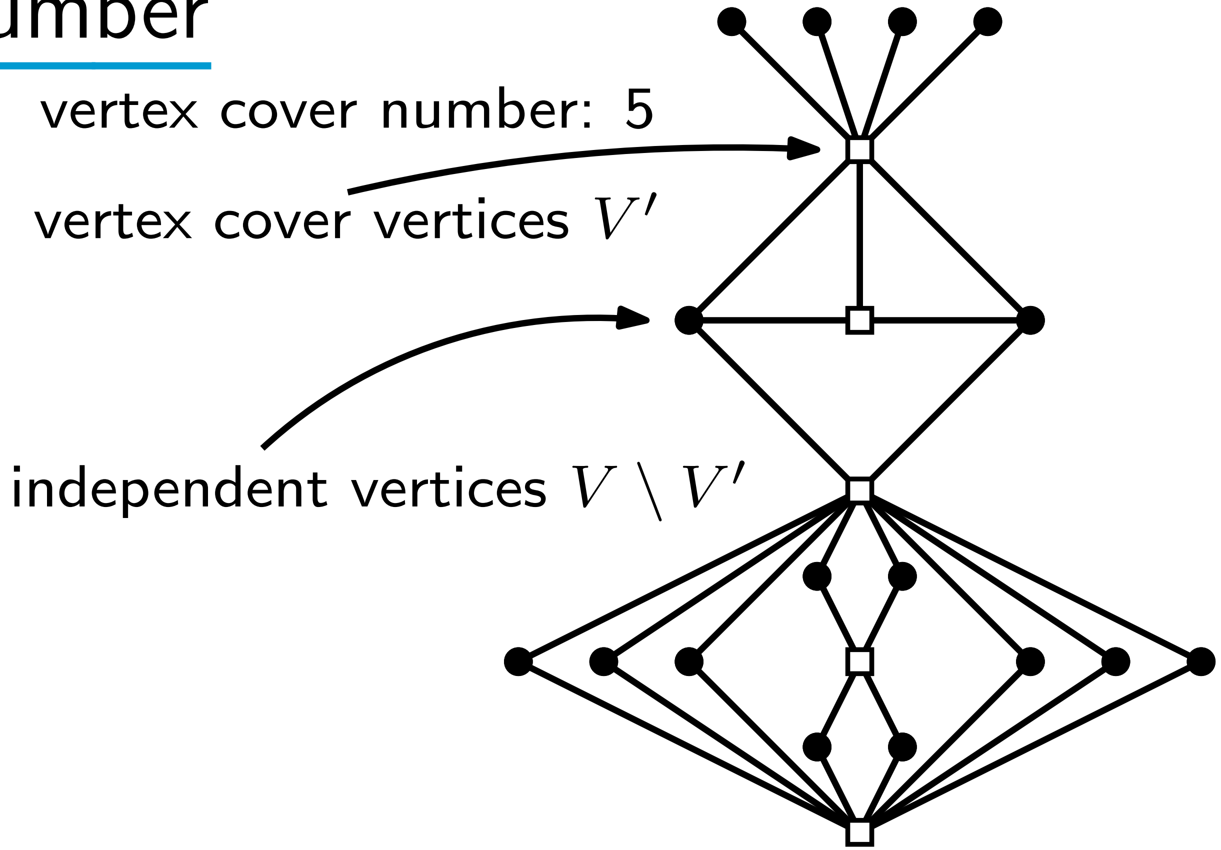


Segment Number by Vertex Cover Number
Input: planar graph $G = (V, E)$, integer s
Parameter: vertex cover number k of G
Question: Is segment number of G at most s?

Segment Number by Vertex Cover Number

vertex cover of a graph $G = (V, E)$: set $V' \subseteq V$ s.t. $e \cap V' \neq \emptyset$ for each $e \in E$

vertex cover number of a graph: size of its smallest vertex cover



Segment Number by Vertex Cover Number

Input: planar graph $G = (V, E)$, integer s
Parameter: vertex cover number k of G
Question: Is segment number of G at most s?

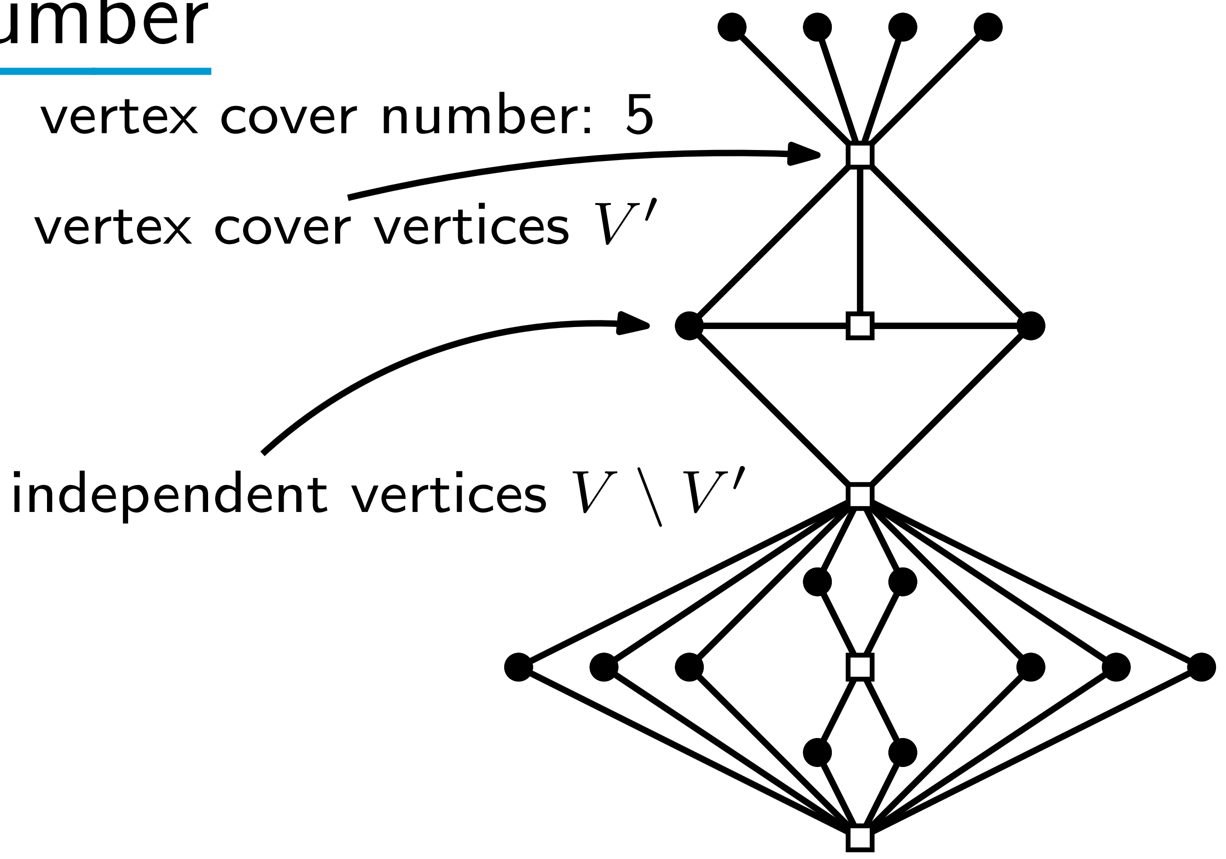
Overview of the Approach for **computing** the segment number:

1. Remove some vertices of degree one and two $\rightsquigarrow$ $\mathcal{O}(2^k)$ vertices
2. Iterate over all possible embeddings and alignments $\rightsquigarrow$ number of choices is a function in k
3. Use Renegar to test for realizability $\rightsquigarrow$ $2^{\mathcal{O}(k2^k)}$ time per choice
4. Reinsert the missing vertices optimally via an ILP $\rightsquigarrow$ $2^{\mathcal{O}(k2^k)}$ time per choice

→ Take the best

Segment Number by Vertex Cover Number

1. Remove some vertices of degree one and two



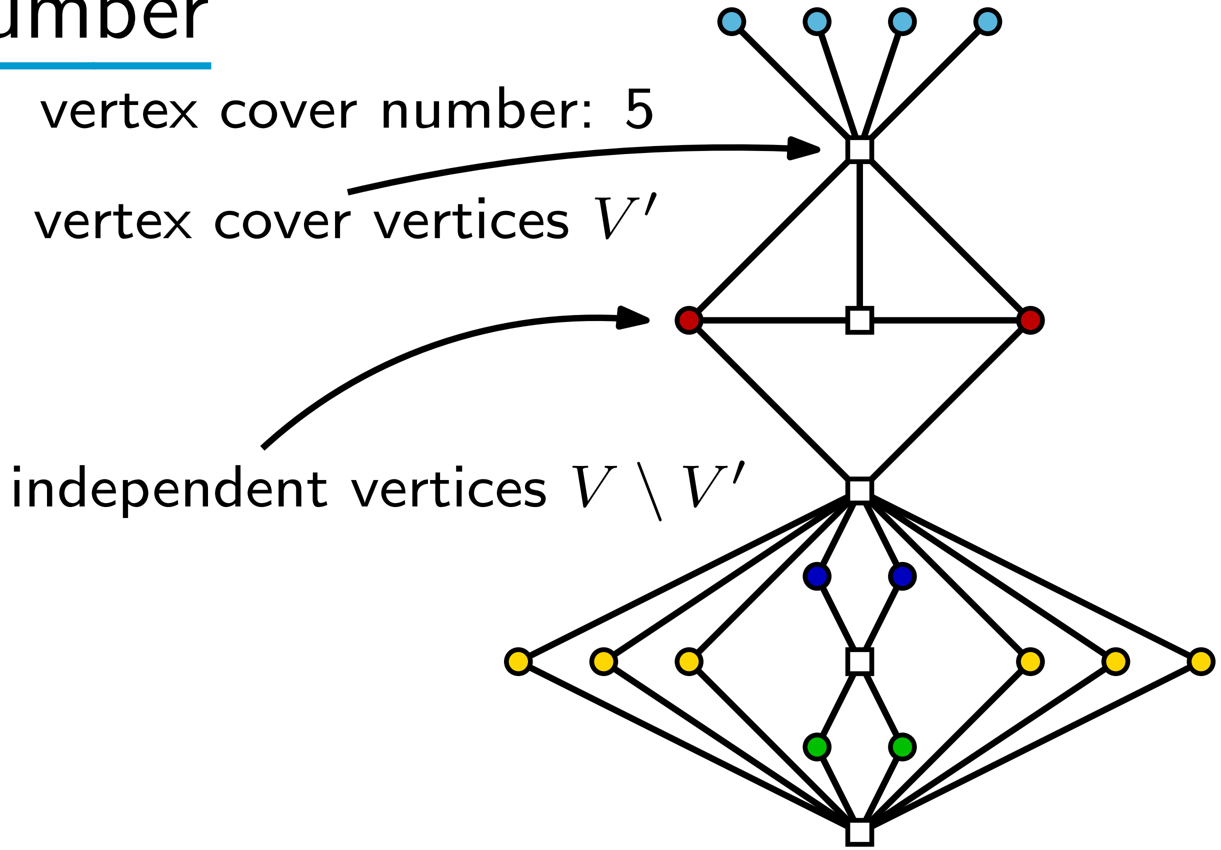
$\rightsquigarrow \mathcal{O}(2^k)$ vertices

Segment Number by Vertex Cover Number

1. Remove some vertices of degree one and two

Two independent vertices v, v' are equivalent iff adjacent to the same vertices in V'

j-class: equivalence class where each vertex is adjacent to exactly j vertices.



$\rightsquigarrow \mathcal{O}(2^k)$ vertices

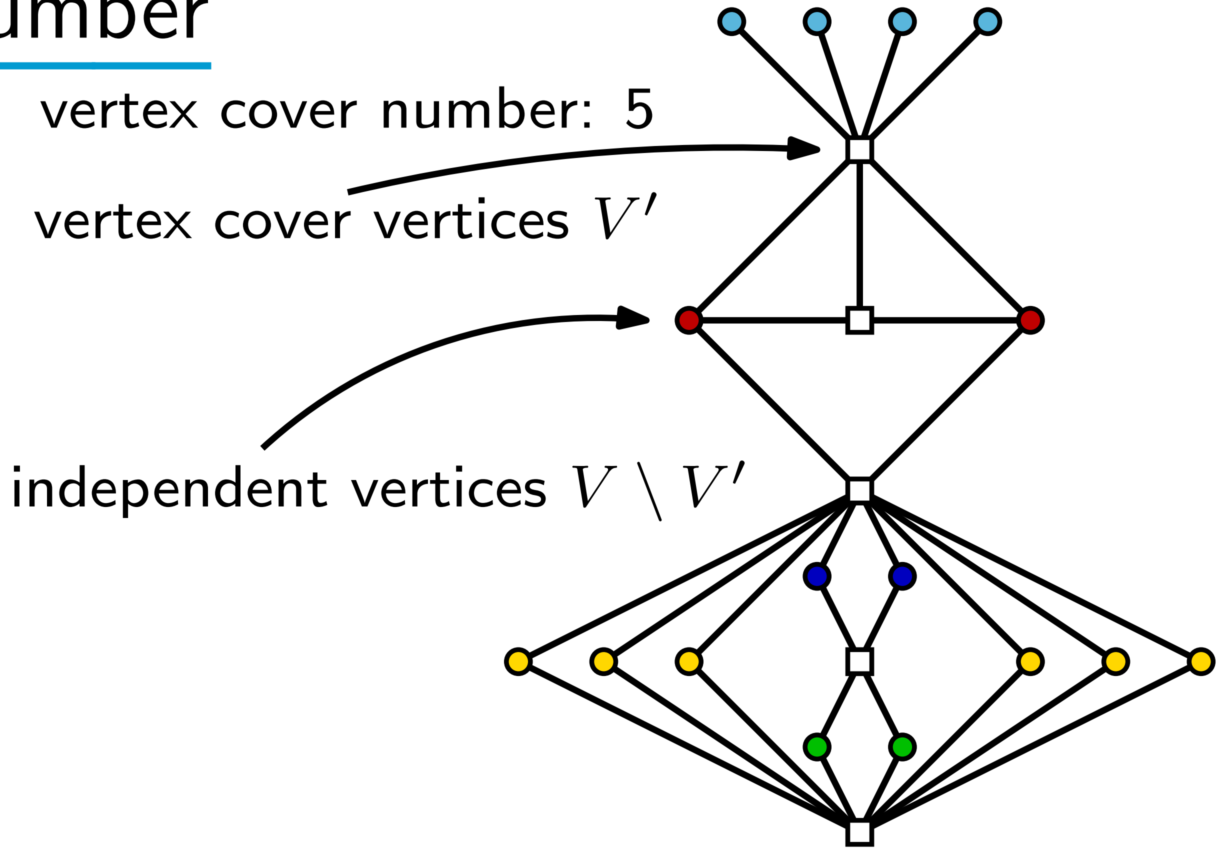
Segment Number by Vertex Cover Number

1. Remove some vertices of degree one and two

Two independent vertices v, v' are equivalent iff adjacent to the same vertices in V'

j-class: equivalence class where each vertex is adjacent to exactly j vertices.

a) Remove all vertices of degree 1 (1-classes)



$\rightsquigarrow \mathcal{O}(2^k)$ vertices

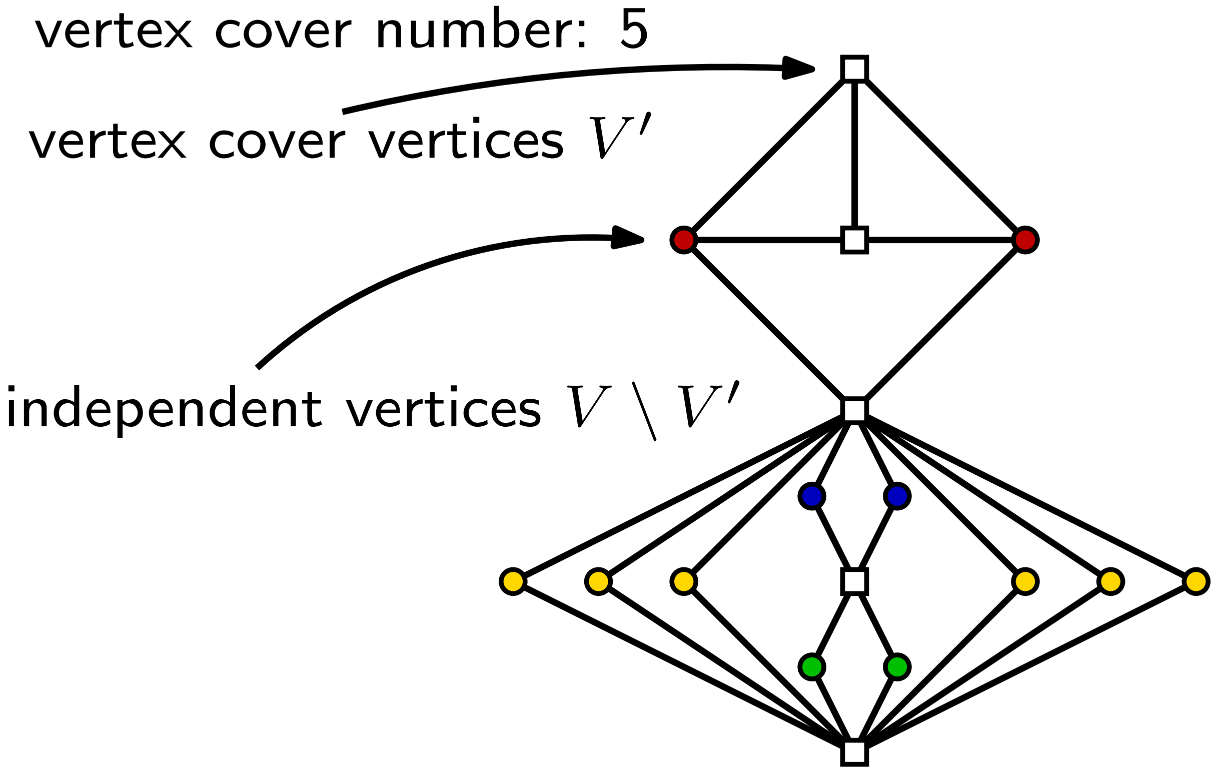
SEGMENT NUMBER by Vertex Cover Number

1. Remove some vertices of degree one and two

Two independent vertices v, v' are equivalent iff adjacent to the same vertices in V'

j-class: equivalence class where each vertex is adjacent to exactly j vertices.

a) Remove all vertices of degree 1 (1-classes)



$\rightsquigarrow \mathcal{O}(2^k)$ vertices

Segment Number by Vertex Cover Number

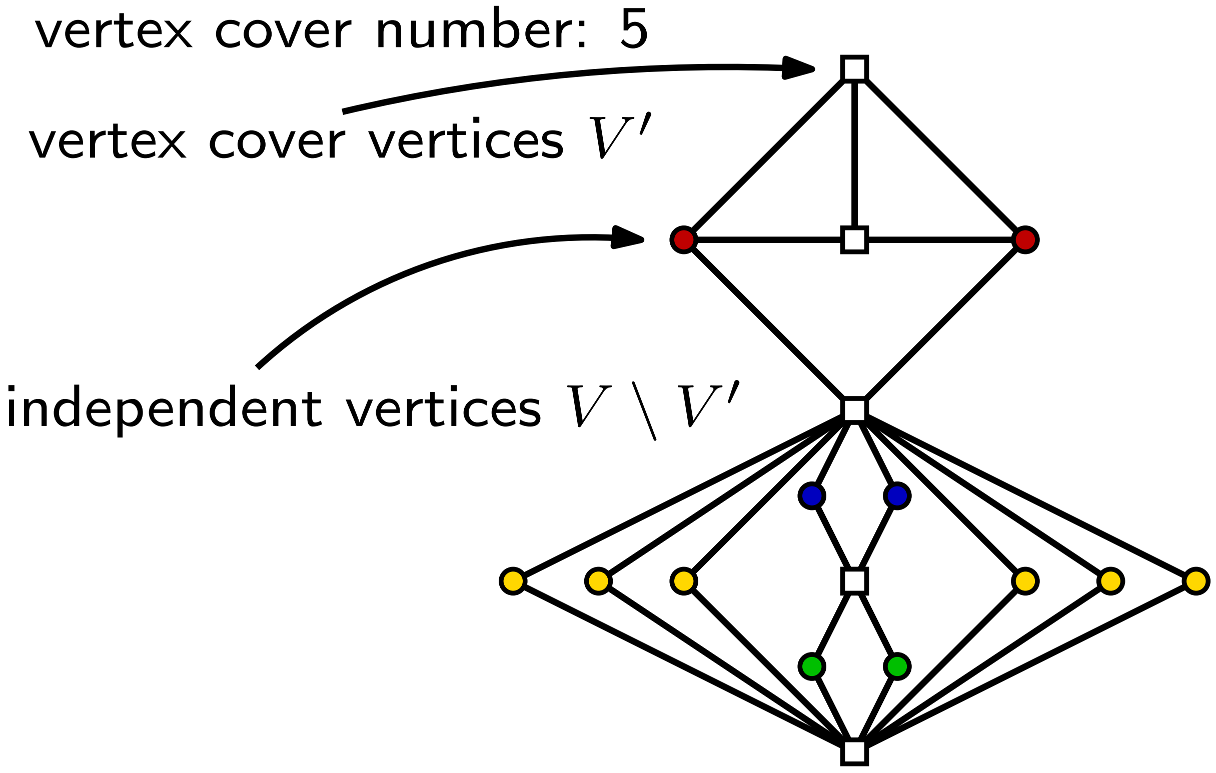
1. Remove some vertices of degree one and two

Two independent vertices v, v' are equivalent iff adjacent to the same vertices in V'

j-class: equivalence class where each vertex is adjacent to exactly j vertices.

a) Remove all vertices of degree 1 (1-classes)

b) For each 2-class, maintain at most k vertices
$\rightsquigarrow$ one per contiguous 2-class



$\rightsquigarrow \mathcal{O}(2^k)$ vertices

Segment Number by Vertex Cover Number

1. Remove some vertices of degree one and two

Two independent vertices v, v' are equivalent iff adjacent to the same vertices in V'

j-class: equivalence class where each vertex is adjacent to exactly j vertices.

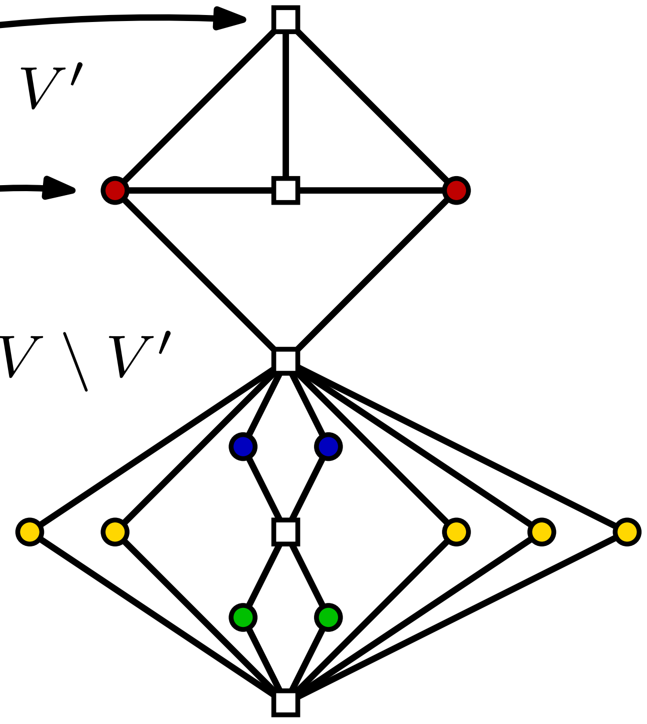
a) Remove all vertices of degree 1 (1-classes)

b) For each 2-class, maintain at most k vertices $\rightsquigarrow$ one per contiguous 2-class

vertex cover number: 5

vertex cover vertices V'

independent vertices $V \setminus V'$



$k\binom{k}{2}$

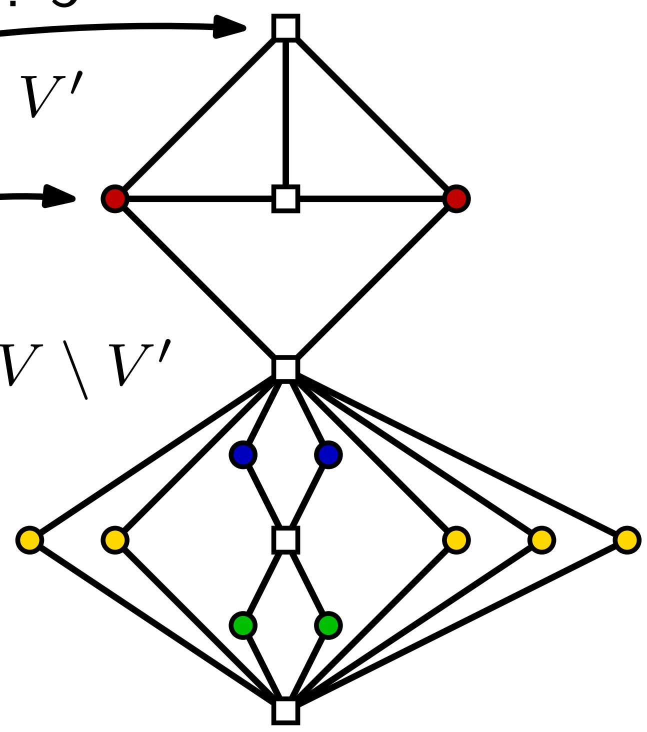
$\rightsquigarrow \mathcal{O}(2^k)$ vertices

Segment Number by Vertex Cover Number

1. Remove some vertices of degree one and two

Two independent vertices v, v' are equivalent iff adjacent to the same vertices in V'

j-class: equivalence class where each vertex is adjacent to exactly j vertices.



a) Remove all vertices of degree 1 (1-classes)

b) For each 2-class, maintain at most k vertices $\rightsquigarrow$ one per contiguous 2-class

$k\binom{k}{2}$

c) Each j-class, $j > 2$ contains at most two vertices otherwise there would be a $K_{3,3}$

$2 \cdot \sum_{j=3}^{k} \binom{k}{j} \in \mathcal{O}(2^k)$

$\rightsquigarrow \mathcal{O}(2^k)$ vertices

Segment Number by Vertex Cover Number

1. Remove some vertices of degree one and two

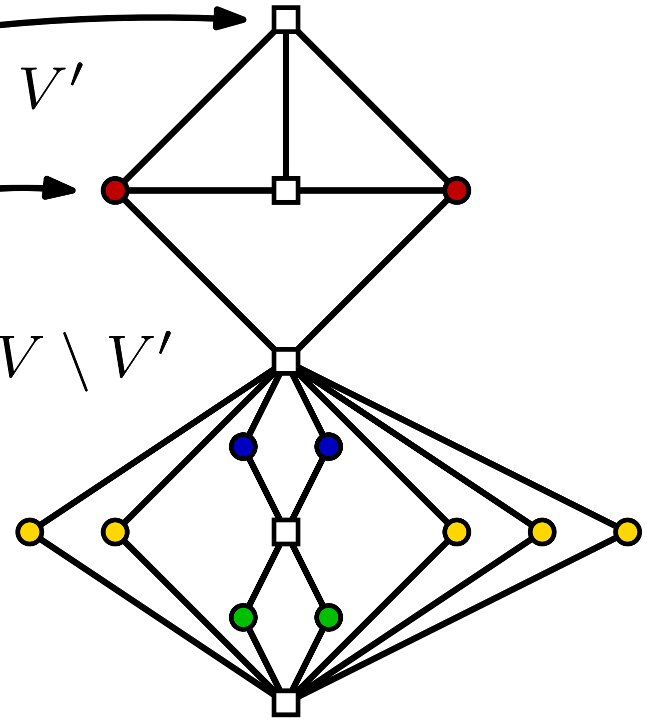
Two independent vertices v, v' are equivalent iff adjacent to the same vertices in V'

j-class: equivalence class where each vertex is adjacent to exactly j vertices.

vertex cover number: 5

vertex cover vertices V'

independent vertices $V \setminus V'$



a) Remove all vertices of degree 1 (1-classes)

b) For each 2-class, maintain at most k vertices $\rightsquigarrow$ one per contiguous 2-class — $k\binom{k}{2}$

c) Each j-class, $j > 2$ contains at most two vertices otherwise there would be a $K_{3,3}$ — $2 \cdot \sum_{j=3}^{k} \binom{k}{j} \in \mathcal{O}(2^k)$

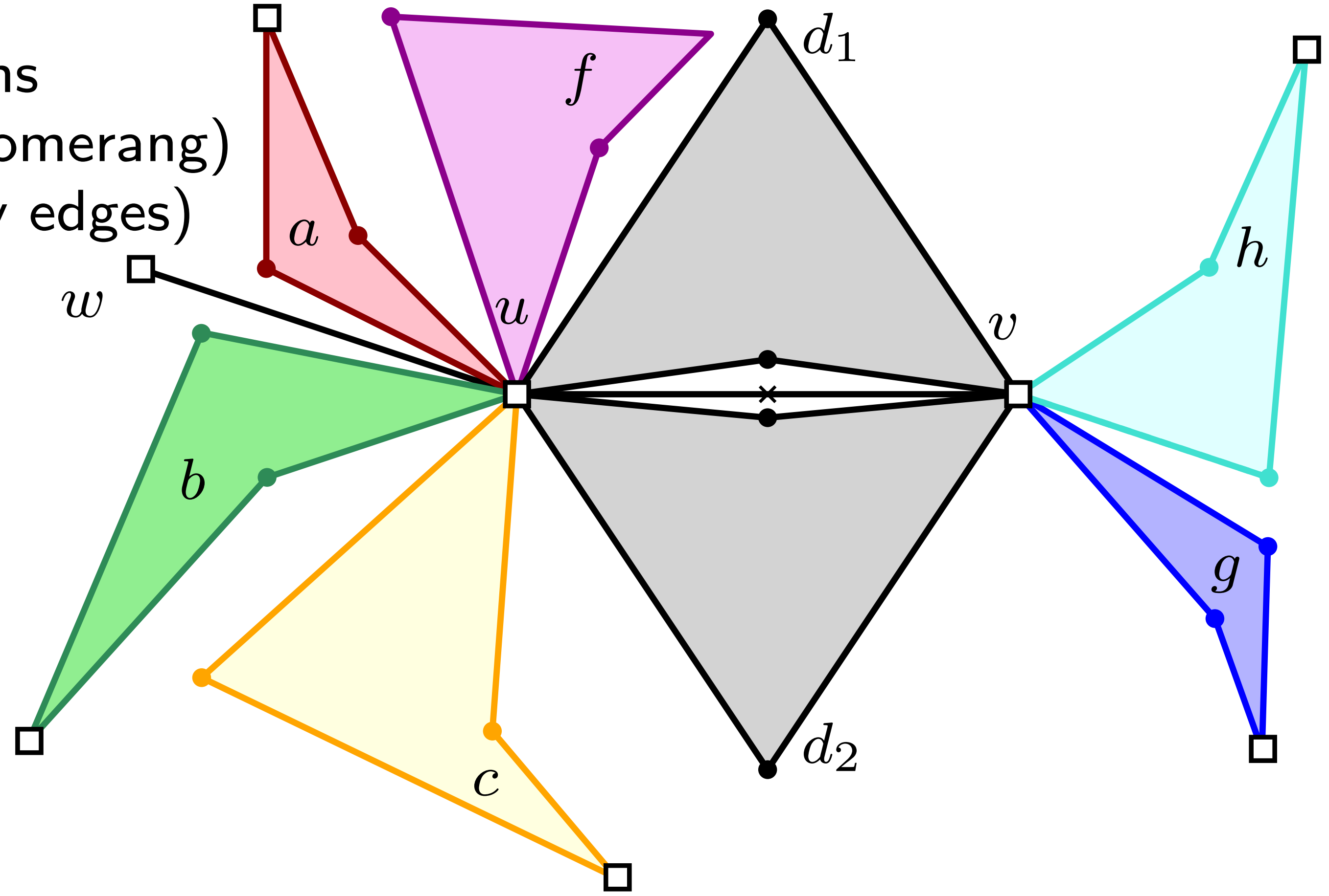
d) Vertex cover — k

$\rightsquigarrow \mathcal{O}(2^k)$ vertices

SEGMENT NUMBER by Vertex Cover Number

2. Iterate over all possible embeddings and alignments

a) each contiguous 2-class is represented by 4 paths
which must form a non-convex quadrangle (boomerang)
(alignments at independent vertices represented by edges)



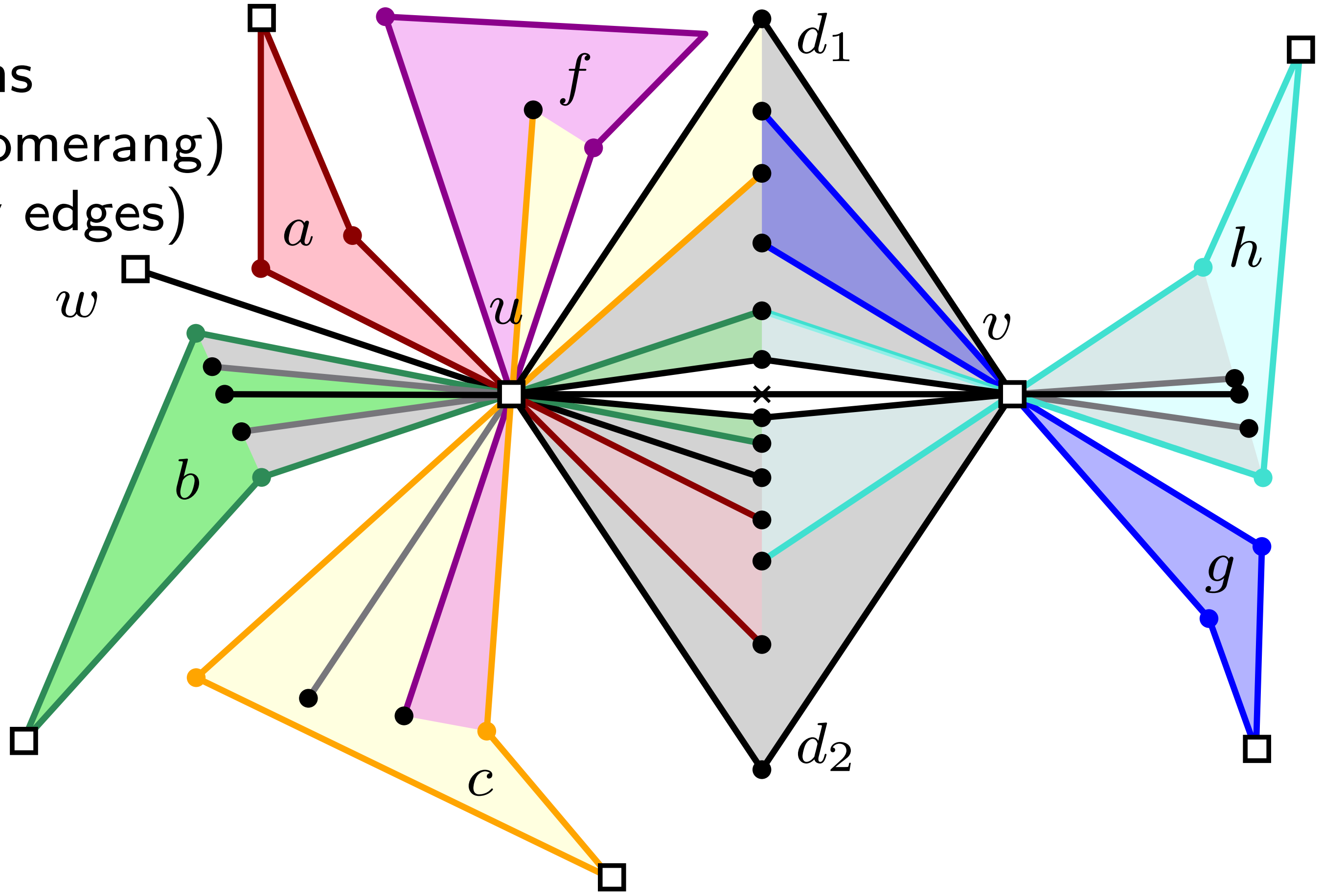
Segment Number by Vertex Cover Number

2. Iterate over all possible embeddings and alignments

a) each contiguous 2-class is represented by 4 paths
which must form a non-convex quadrangle (boomerang)
(alignments at independent vertices represented by edges)

b) further subdivide boomerangs,
according to the choice of the alignments

still $\mathcal{O}(2^k)$ vertices



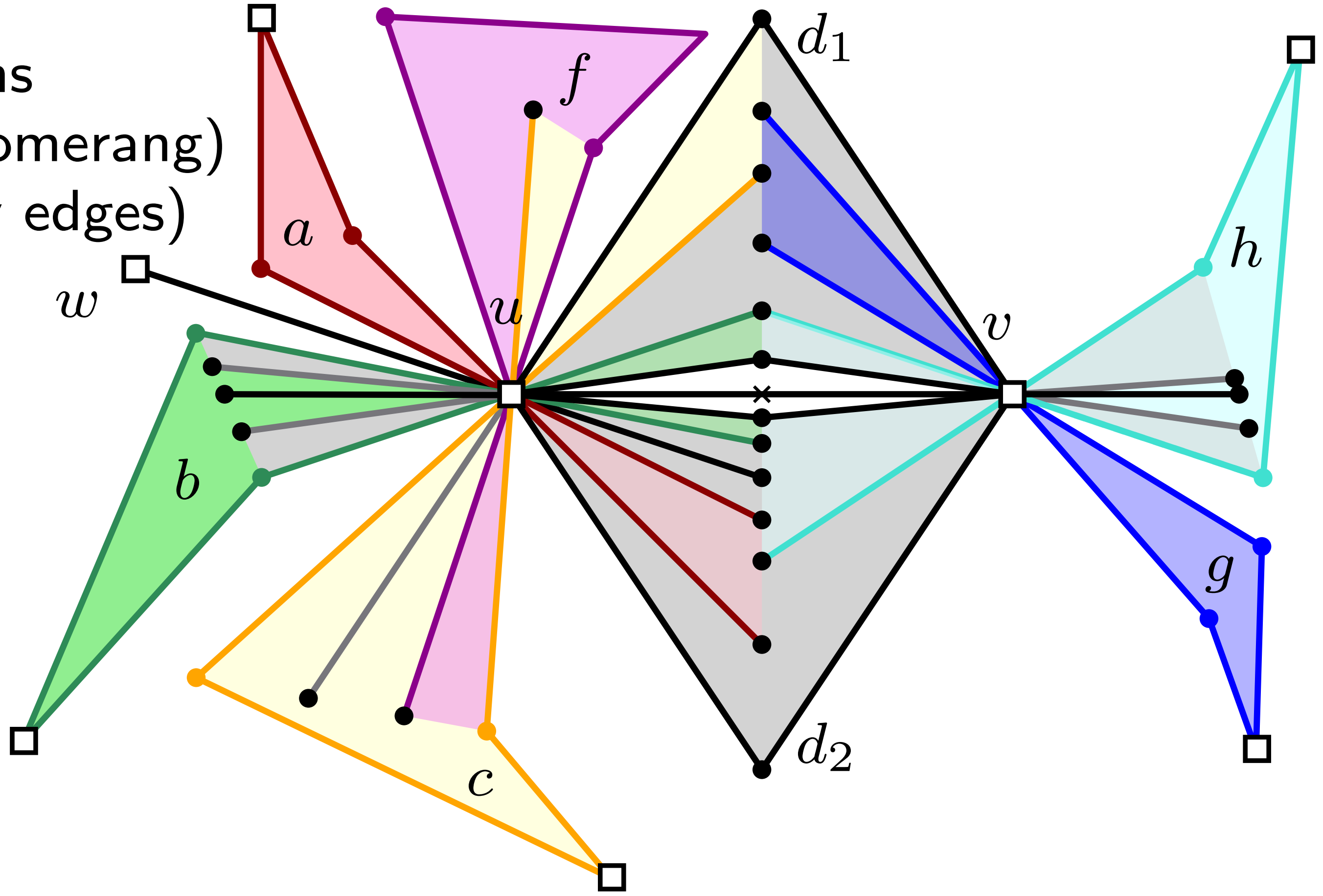
Segment Number by Vertex Cover Number

2. Iterate over all possible embeddings and alignments

a) each contiguous 2-class is represented by 4 paths
which must form a non-convex quadrangle (boomerang)
(alignments at independent vertices represented by edges)

b) further subdivide boomerangs,
according to the choice of the alignments

still $\mathcal{O}(2^k)$ vertices

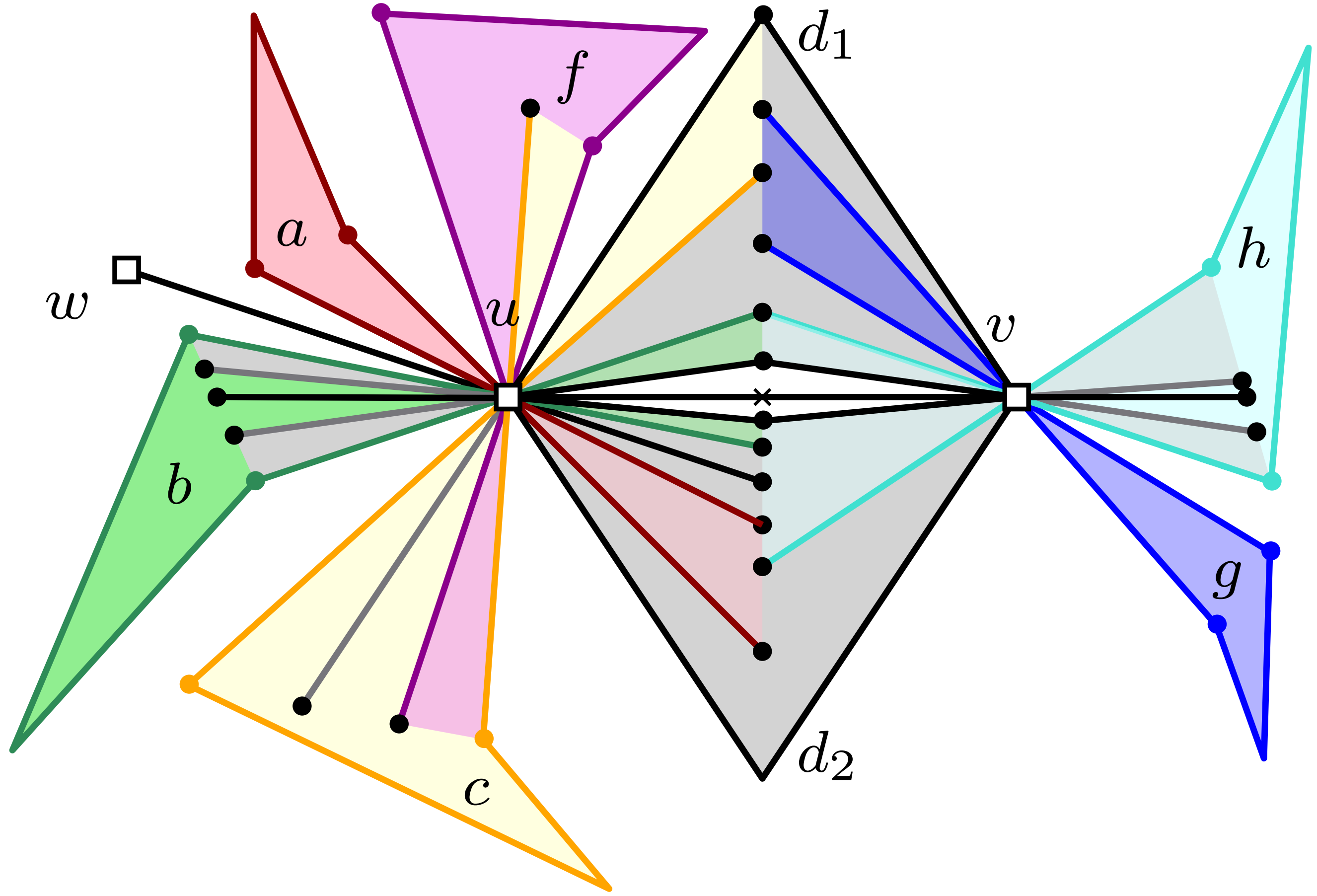


3. Use Renegar to test in $2^{\mathcal{O}(k2^k)}$ time for realizability

if the answer is yes then . . .

Segment Number by Vertex Cover Number

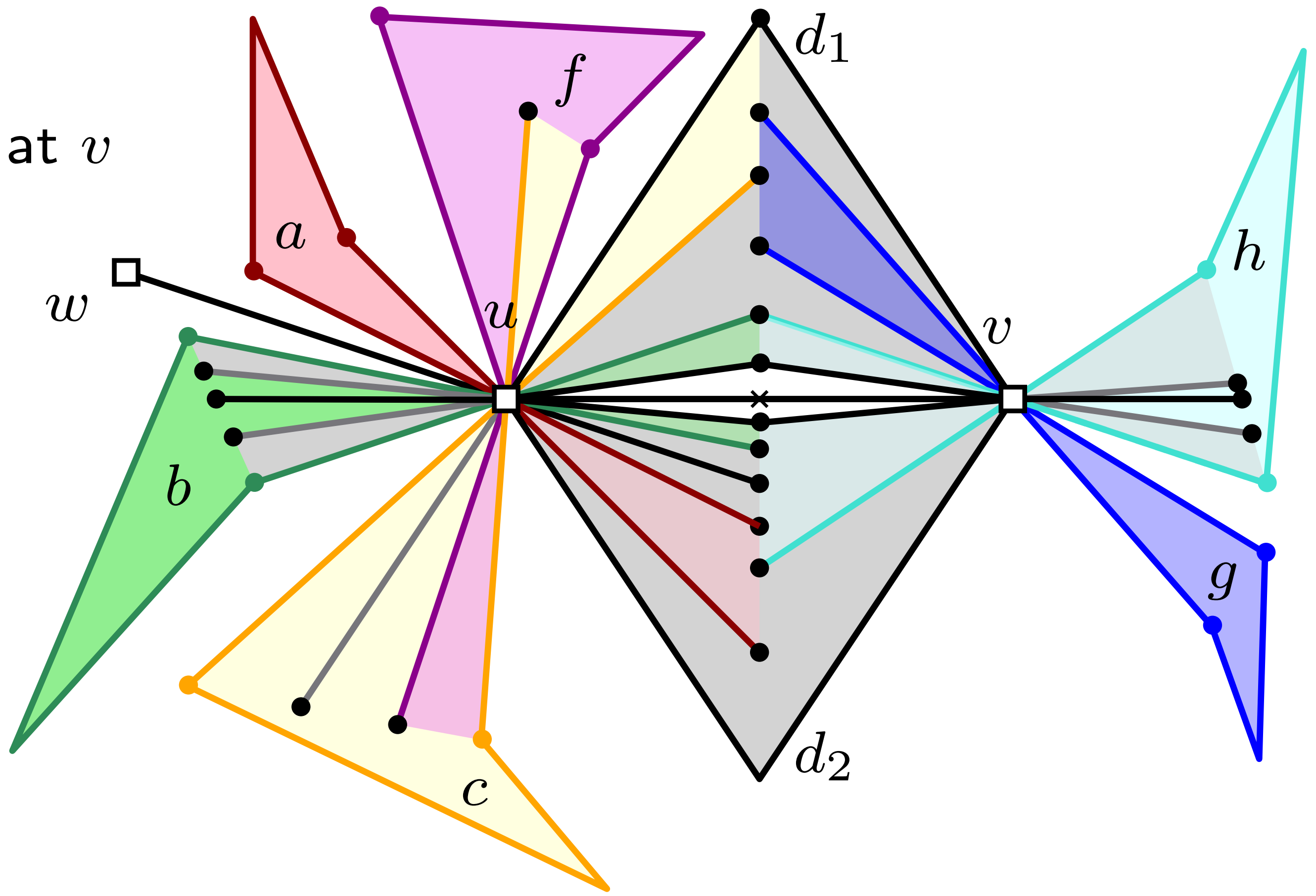
4. Reinsert the missing vertices optimally via an ILP



Segment Number by Vertex Cover Number

4. Reinsert the missing vertices optimally via an ILP

x(vertex v, boomerang b, boomerang d):
number of edges in b and d that should be aligned at v

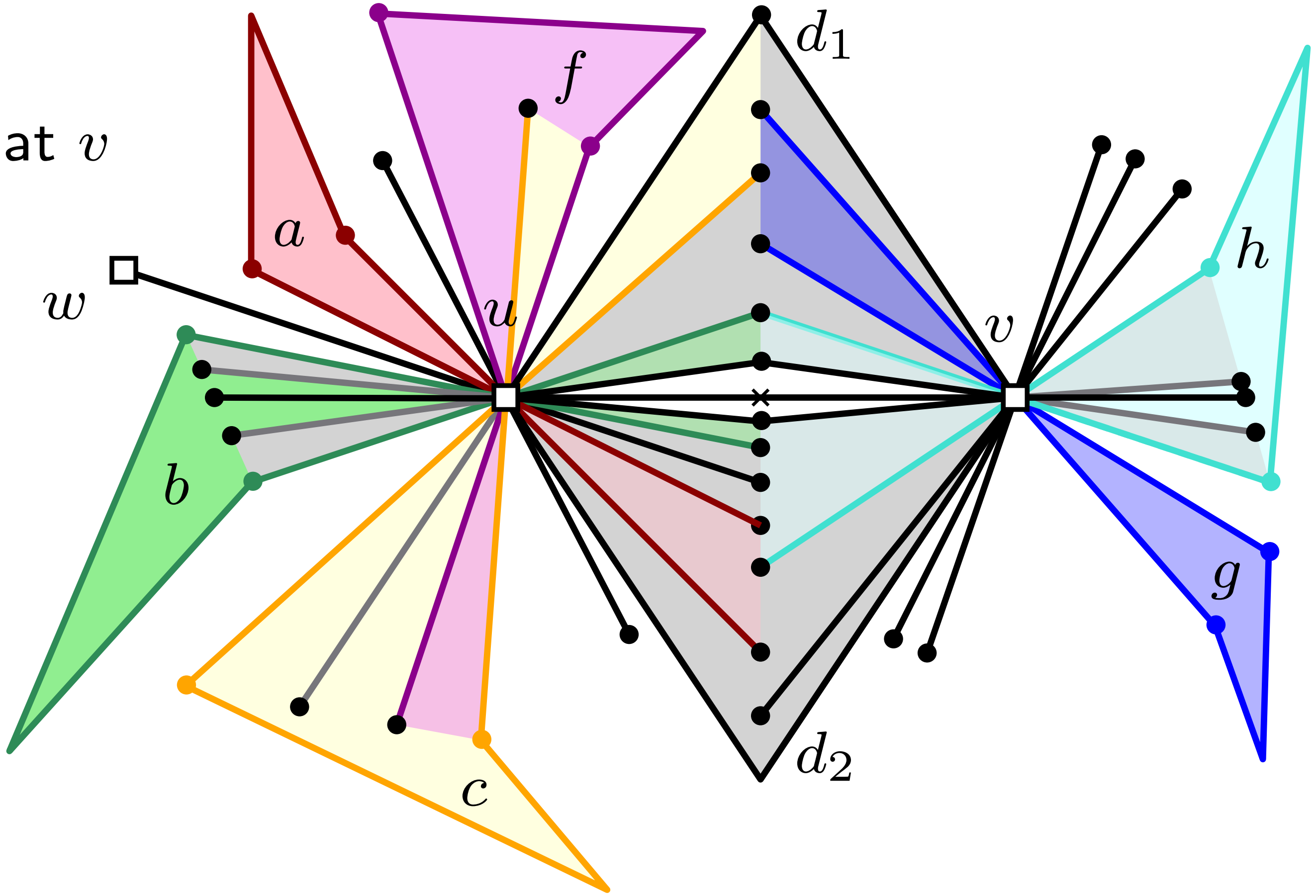


Segment Number by Vertex Cover Number

4. Reinsert the missing vertices optimally via an ILP

x(vertex v, boomerang b, boomerang d):
number of edges in b and d that should be aligned at v

y(vertex v, boomerang b):
number of edges in b that should be aligned
with leaves adjacent to v



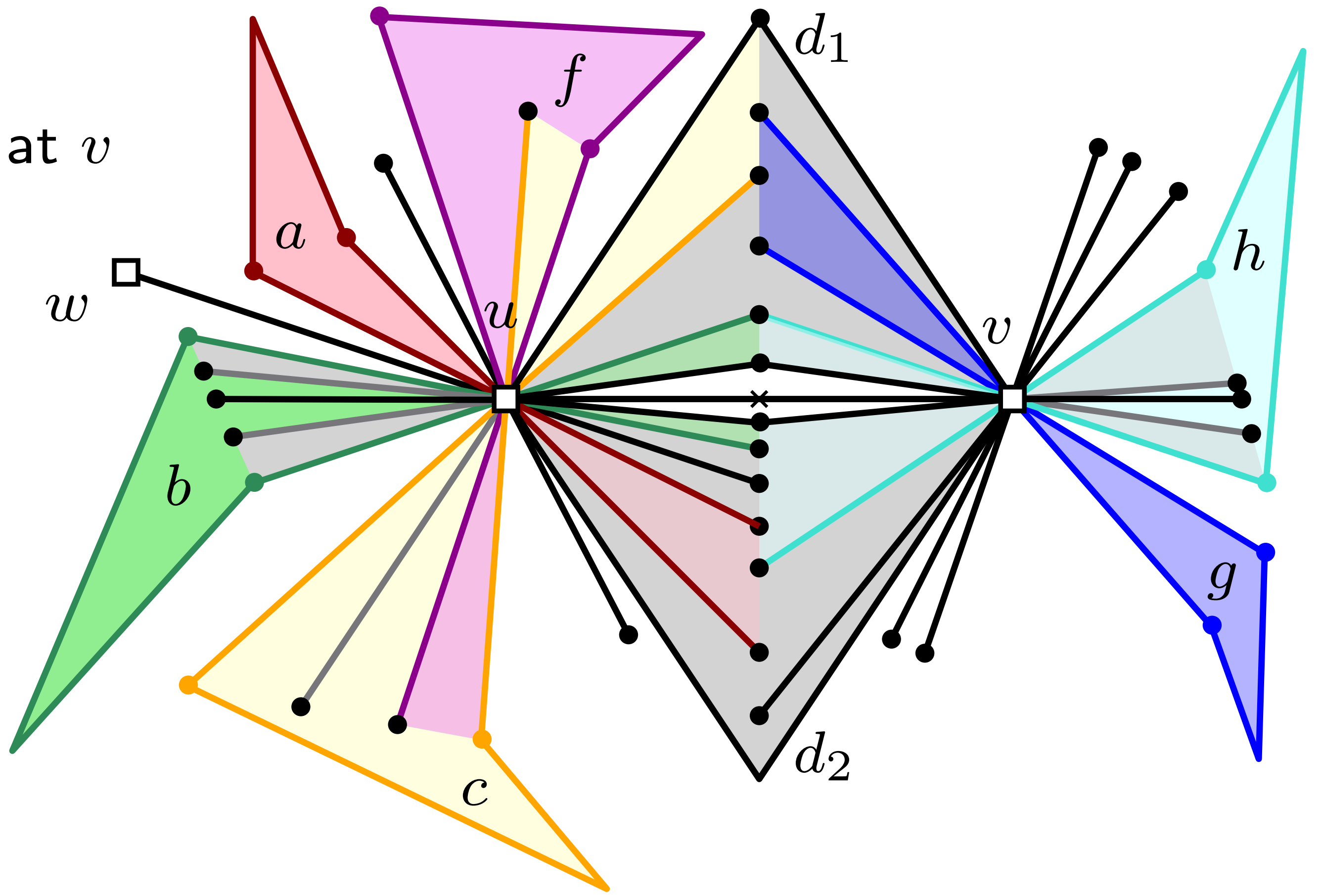
Segment Number by Vertex Cover Number

4. Reinsert the missing vertices optimally via an ILP

x(vertex v, boomerang b, boomerang d):
number of edges in b and d that should be aligned at v

y(vertex v, boomerang b):
number of edges in b that should be aligned with leaves adjacent to v

maximize $x + y$



make sure that total number of independent vertices per 1- and 2-class is not exceeded

$\mathcal{O}(2^k)$ variables and constraints
$\rightsquigarrow$ can be solved in $2^{\mathcal{O}(k2^k)}$ time

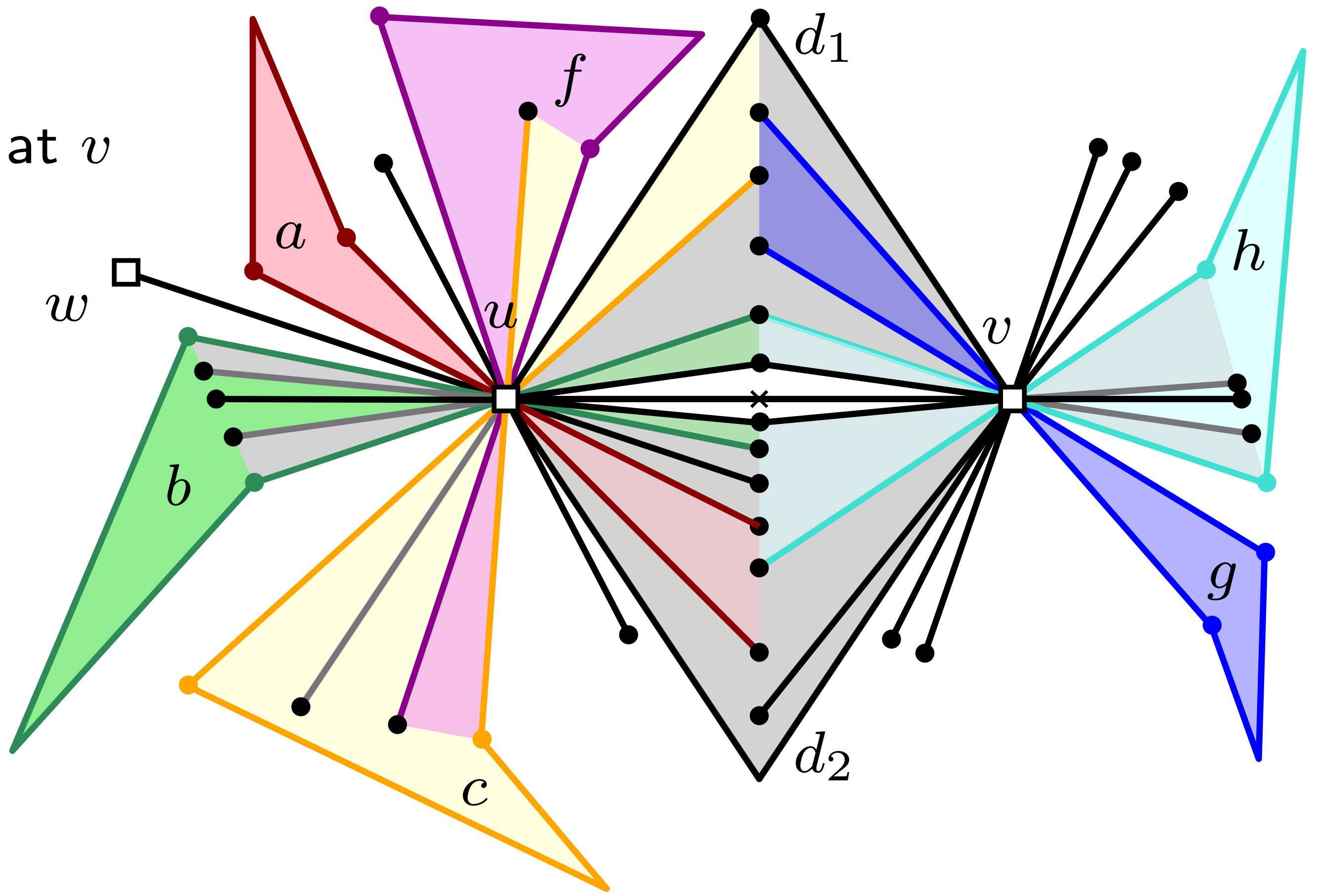
Segment Number by Vertex Cover Number

4. Reinsert the missing vertices optimally via an ILP

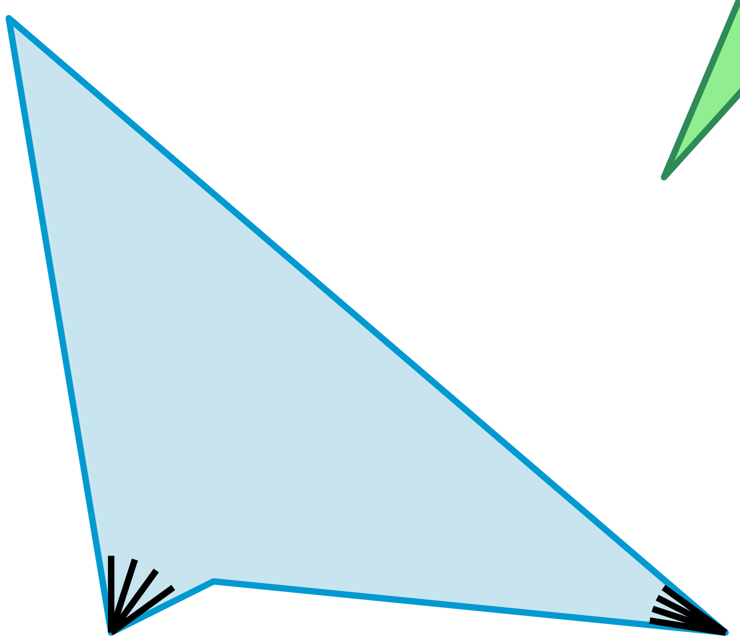
x(vertex v, boomerang b, boomerang d):
number of edges in b and d that should be aligned at v

y(vertex v, boomerang b):
number of edges in b that should be aligned with leaves adjacent to v

maximize $x + y$



Observe:
Due to the non-convex shape, any given slopes on either sides can be combined s.t. intersection point lies inside boomerang.



make sure that total number of independent vertices per 1- and 2-class is not exceeded

$\mathcal{O}(2^k)$ variables and constraints
$\rightsquigarrow$ can be solved in $2^{\mathcal{O}(k2^k)}$ time

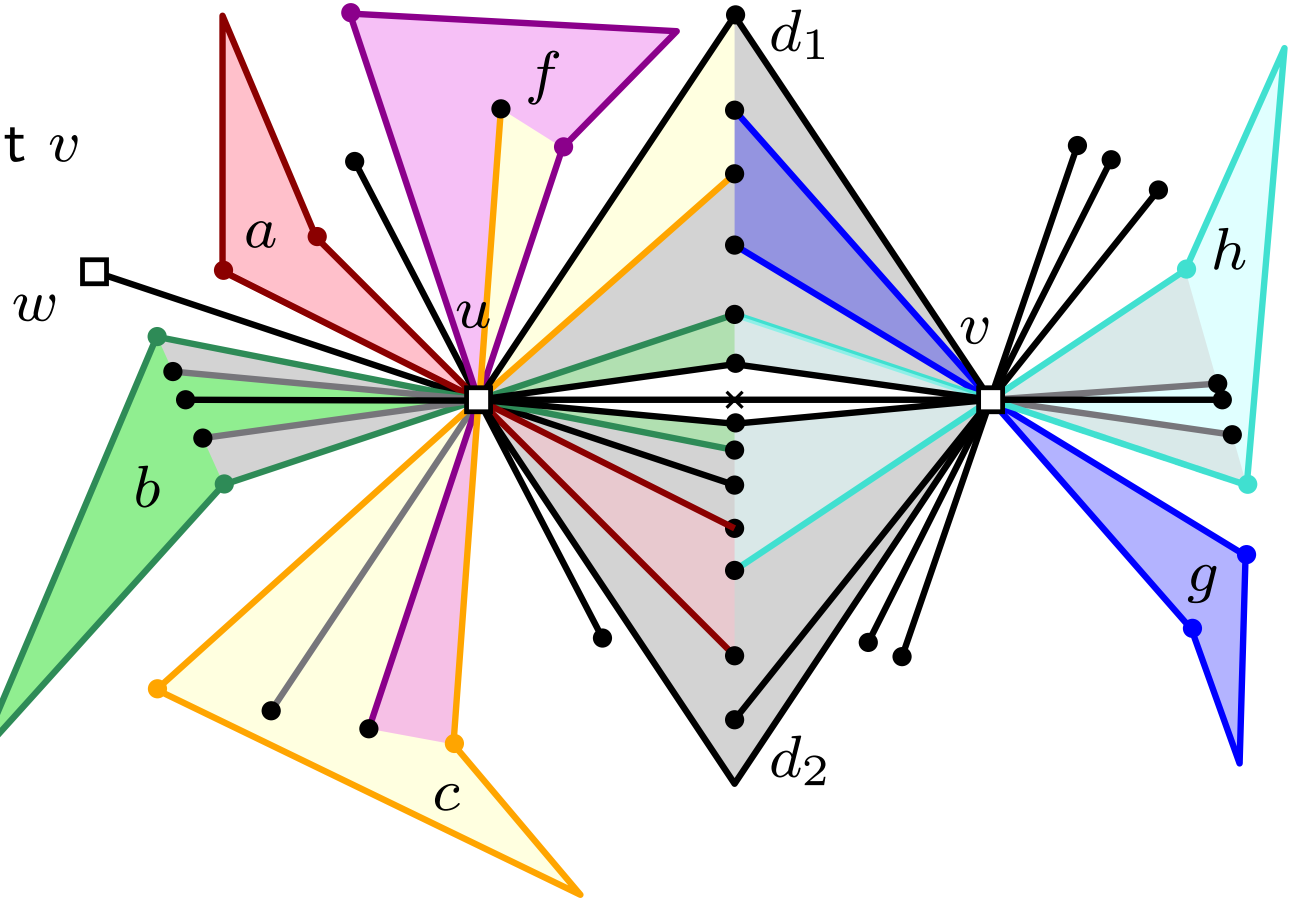
Segment Number by Vertex Cover Number

4. Reinsert the missing vertices optimally via an ILP

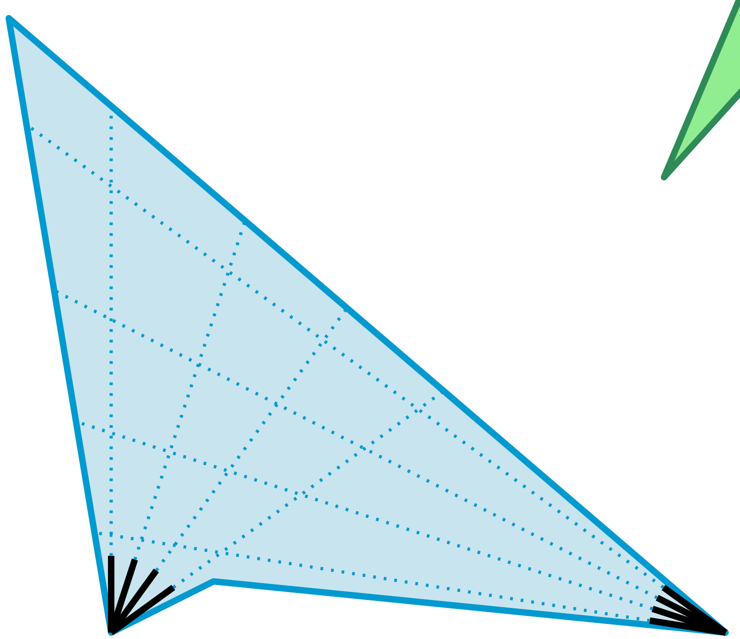
x(vertex v, boomerang b, boomerang d):
number of edges in b and d that should be aligned at v

y(vertex v, boomerang b):
number of edges in b that should be aligned with leaves adjacent to v

maximize $x + y$



Observe:
Due to the non-convex shape, any given slopes on either sides can be combined s.t. intersection point lies inside boomerang.



make sure that total number of independent vertices per 1- and 2-class is not exceeded

$\mathcal{O}(2^k)$ variables and constraints
$\rightsquigarrow$ can be solved in $2^{\mathcal{O}(k2^k)}$ time

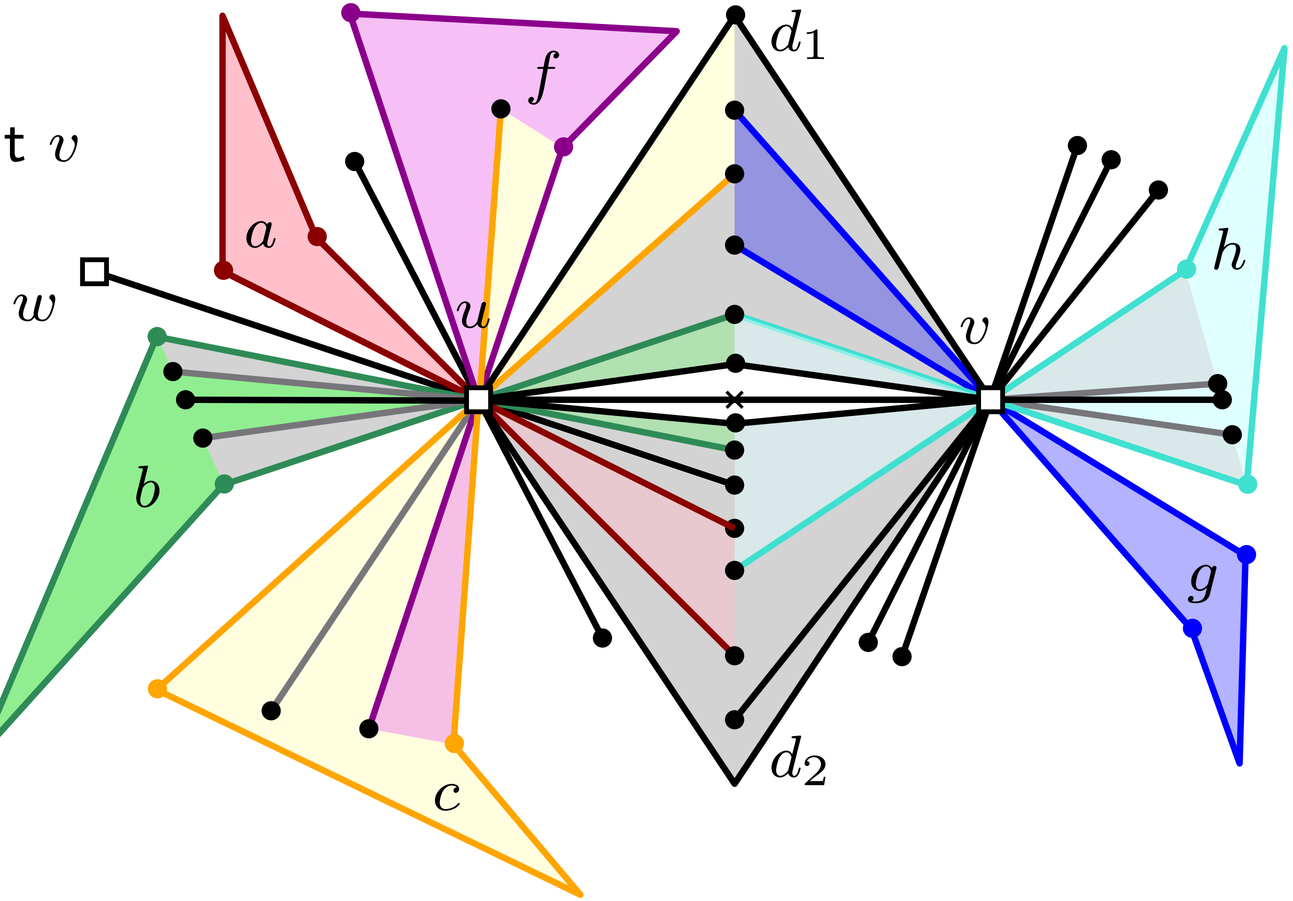
Segment Number by Vertex Cover Number

4. Reinsert the missing vertices optimally via an ILP

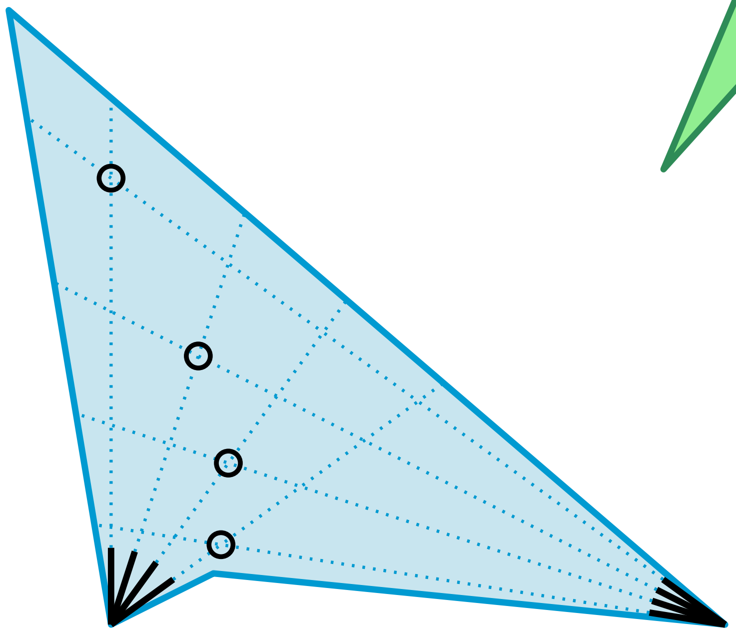
x(vertex v, boomerang b, boomerang d):
number of edges in b and d that should be aligned at v

y(vertex v, boomerang b):
number of edges in b that should be aligned with leaves adjacent to v

maximize $x + y$



Observe:
Due to the non-convex shape, any given slopes on either sides can be combined s.t. intersection point lies inside boomerang.



make sure that total number of independent vertices per 1- and 2-class is not exceeded

$\mathcal{O}(2^k)$ variables and constraints
$\rightsquigarrow$ can be solved in $2^{\mathcal{O}(k2^k)}$ time

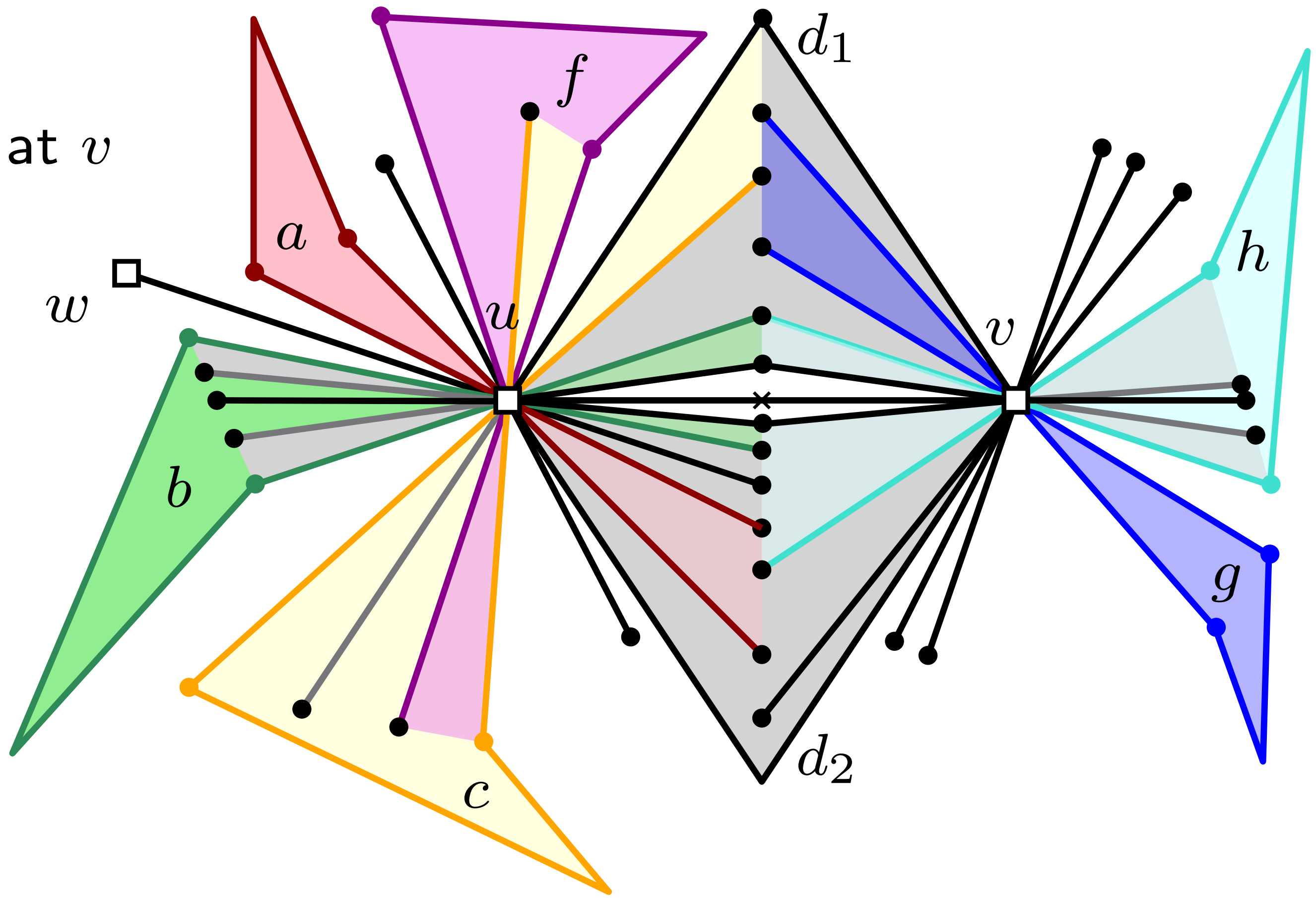
Segment Number by Vertex Cover Number

4. Reinsert the missing vertices optimally via an ILP

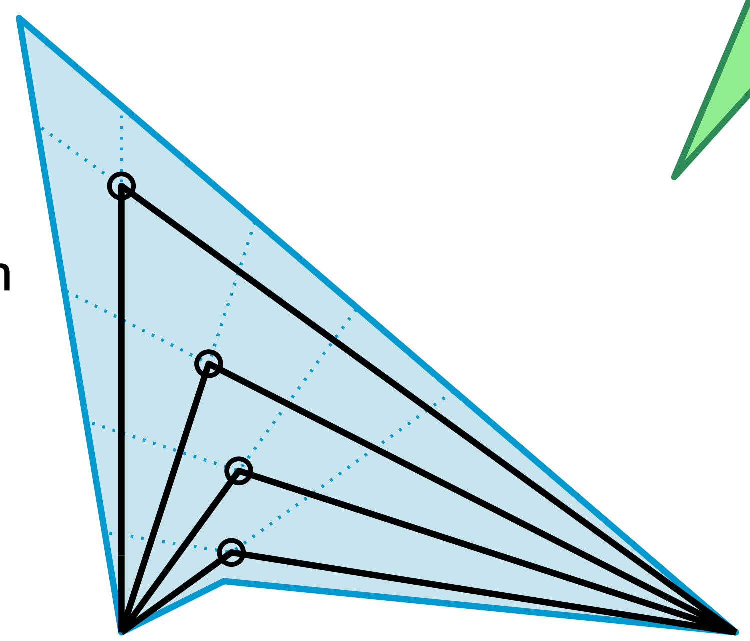
x(vertex v, boomerang b, boomerang d):
number of edges in b and d that should be aligned at v

y(vertex v, boomerang b):
number of edges in b that should be aligned with leaves adjacent to v

maximize $x + y$



Observe:
Due to the non-convex shape, any given slopes on either sides can be combined s.t. intersection point lies inside boomerang.



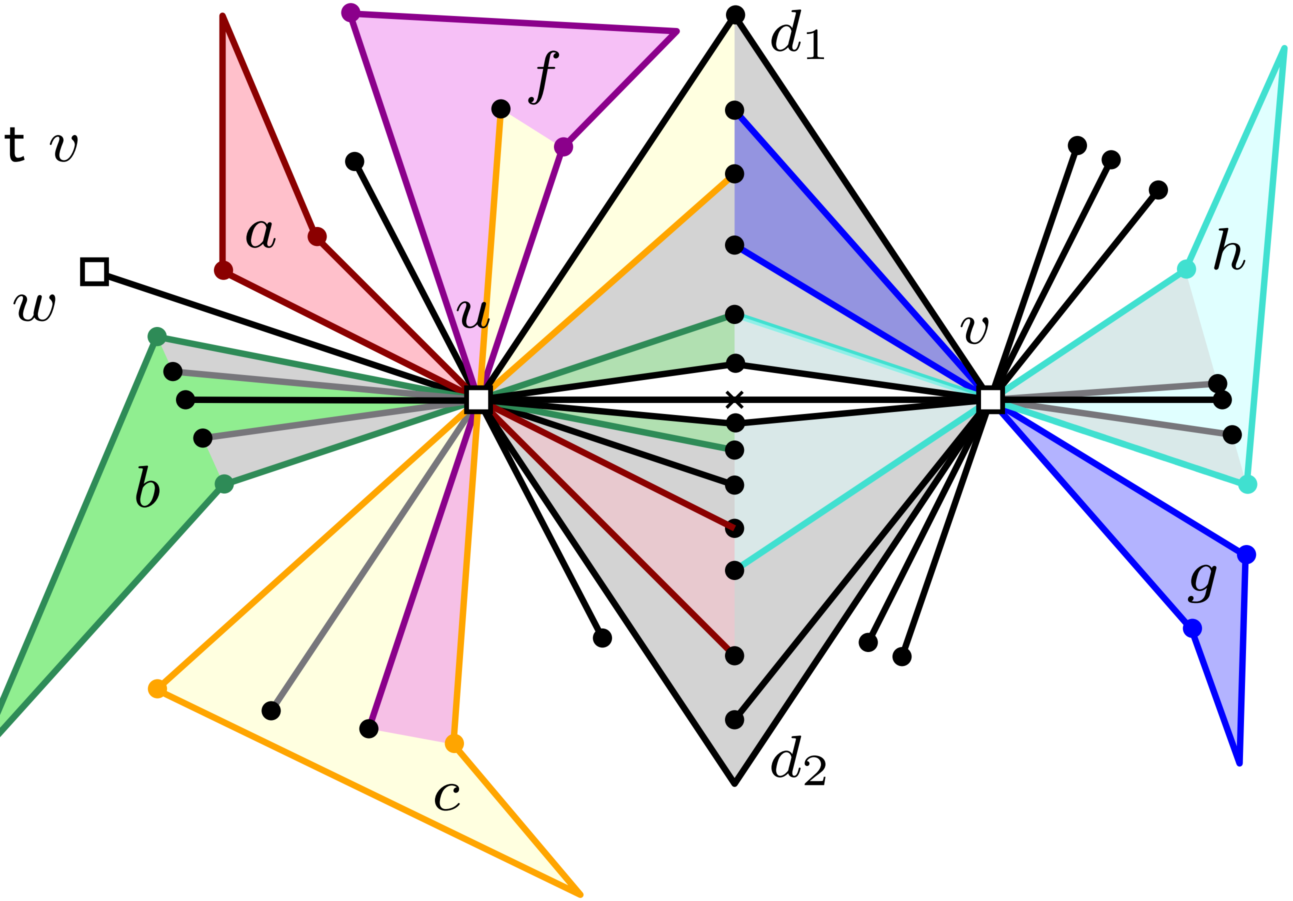
make sure that total number of independent vertices per 1- and 2-class is not exceeded

$\mathcal{O}(2^k)$ variables and constraints
$\rightsquigarrow$ can be solved in $2^{\mathcal{O}(k2^k)}$ time

Segment Number by Vertex Cover Number ← ✓

4. Reinsert the missing vertices optimally via an ILP

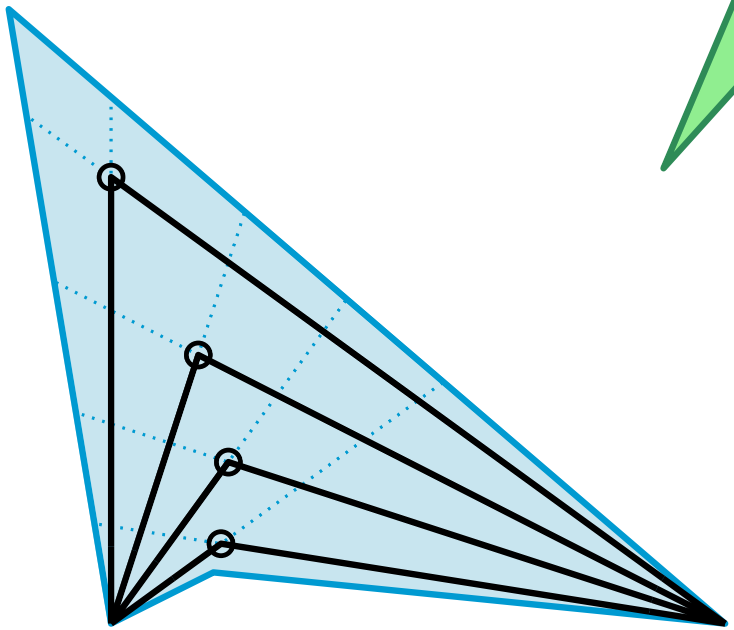
x(vertex v, boomerang b, boomerang d):
number of edges in b and d that should be aligned at v



y(vertex v, boomerang b):
number of edges in b that should be aligned with leaves adjacent to v

maximize $x + y$

Observe:
Due to the non-convex shape, any given slopes on either sides can be combined s.t. intersection point lies inside boomerang.



make sure that total number of independent vertices per 1- and 2-class is not exceeded

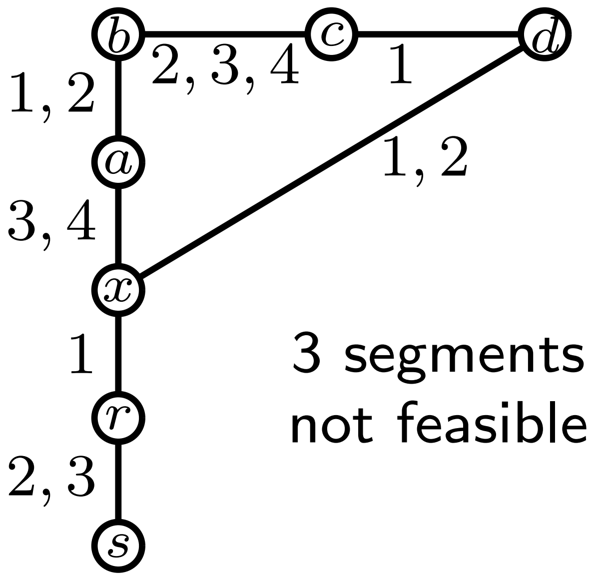
$\mathcal{O}(2^k)$ variables and constraints
$\rightsquigarrow$ can be solved in $2^{\mathcal{O}(k2^k)}$ time

List-Coloring meets Segment Number

List-Incidence Segment Number

Input: planar graph G and, for each $e \in E(G)$, a list $L(e) \subseteq [k]$.
Parameter: An integer k.
Question: Does there exist

- a planar straight-line drawing of G with $\leq k$ segments and
- a labeling $s_1, s_2, \ldots$ of its segments, s.t.
- for every $e \in E(G)$, e is drawn on a segment in $\{s_i : i \in L(e)\}$?



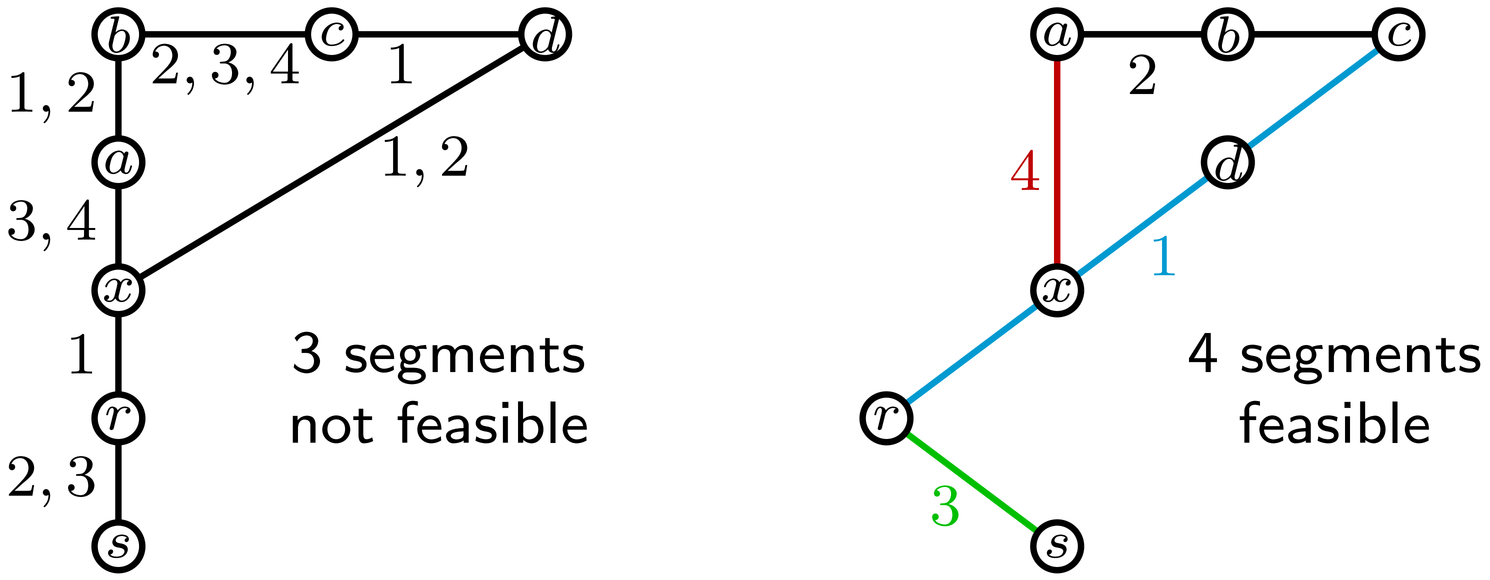
3 segments
not feasible

List-Coloring meets Segment Number

List-Incidence Segment Number

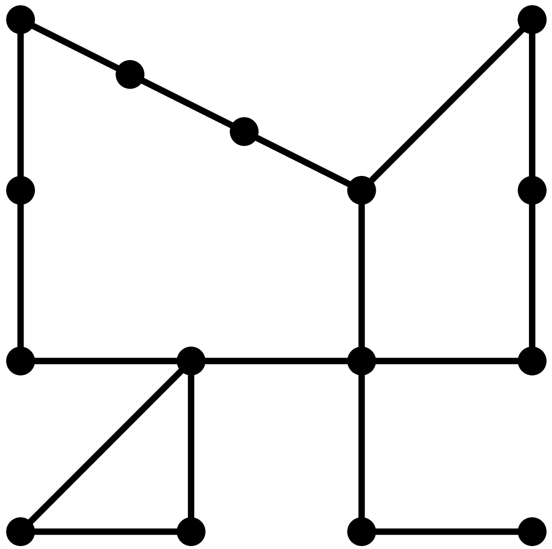
Input: planar graph G and, for each $e \in E(G)$, a list $L(e) \subseteq [k]$.
Parameter: An integer k.
Question: Does there exist

- a planar straight-line drawing of G with $\leq k$ segments and
- a labeling $s_1, s_2, \ldots$ of its segments, s.t.
- for every $e \in E(G)$, e is drawn on a segment in $\{s_i : i \in L(e)\}$?



List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

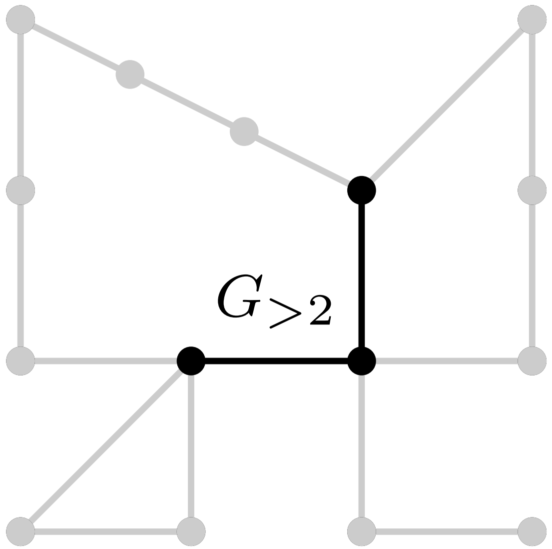


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2.

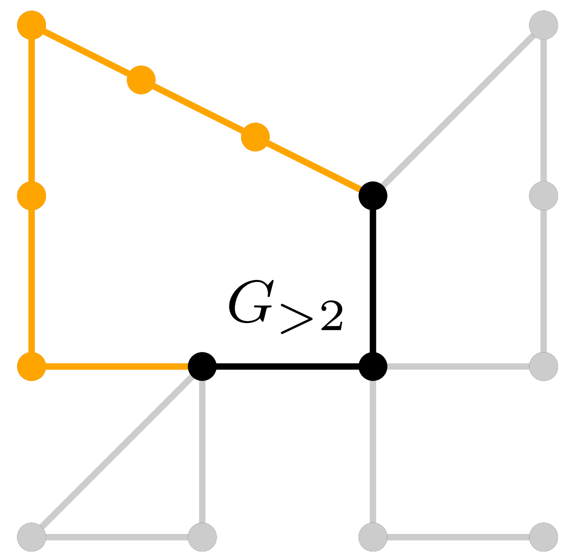


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2.

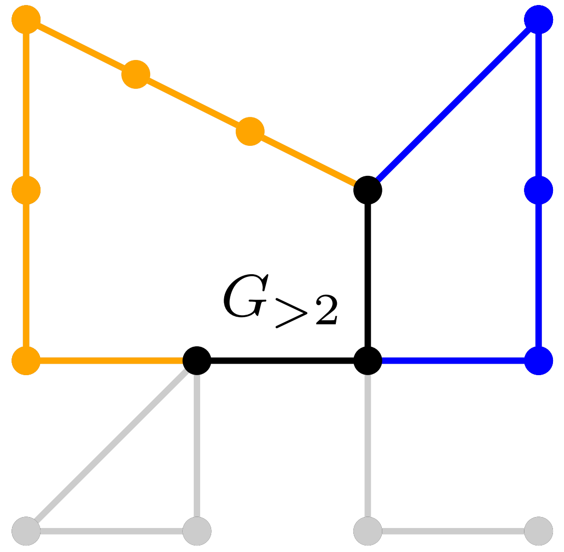


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2.

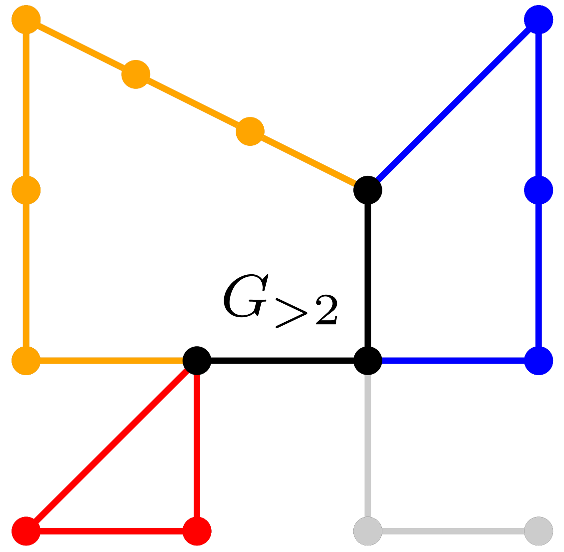


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2.

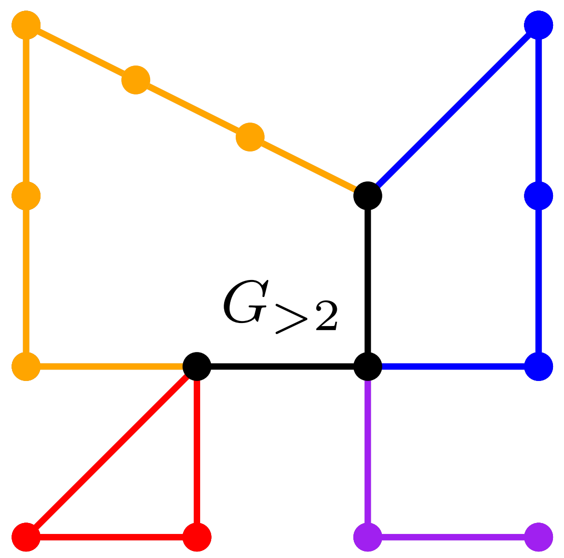


LIST-INCIDENCE SEGMENT NUMBER is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2.

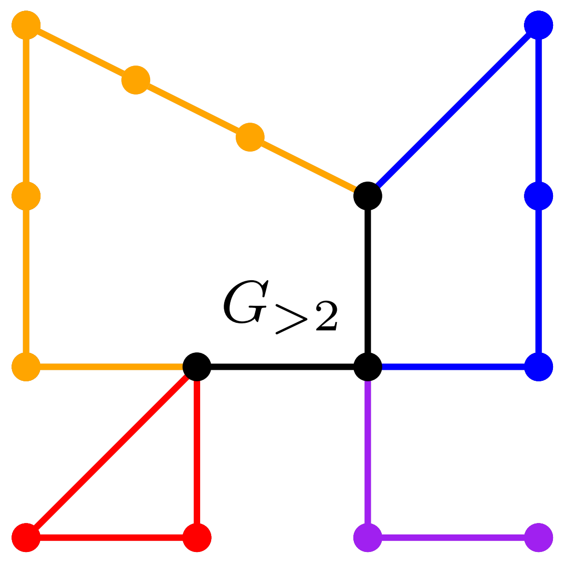


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many



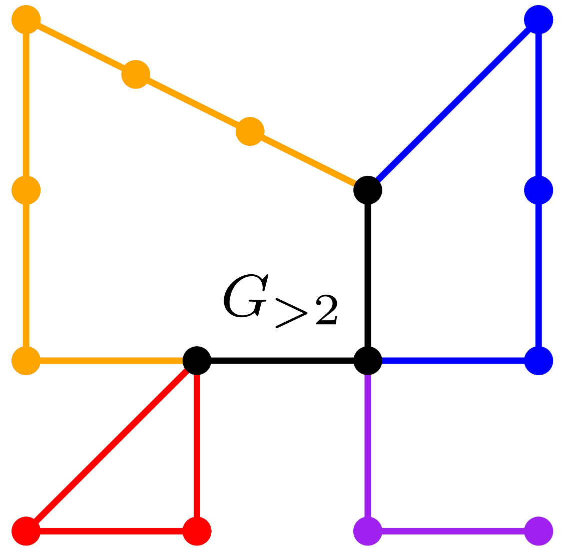
List-Incidence Segment Number is in FPT wrt. Segment Number

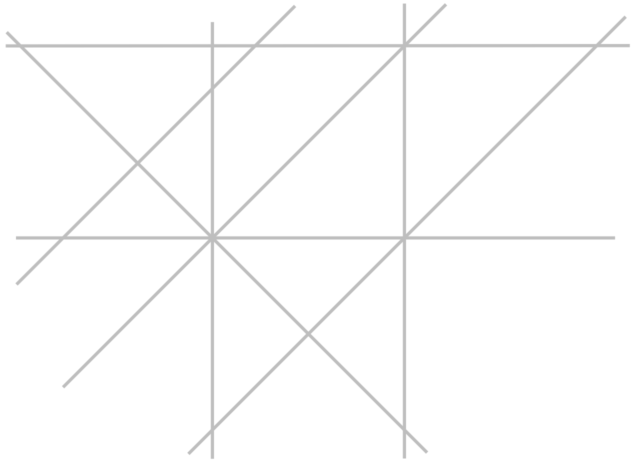
Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many

1. For each arrangement of k lines



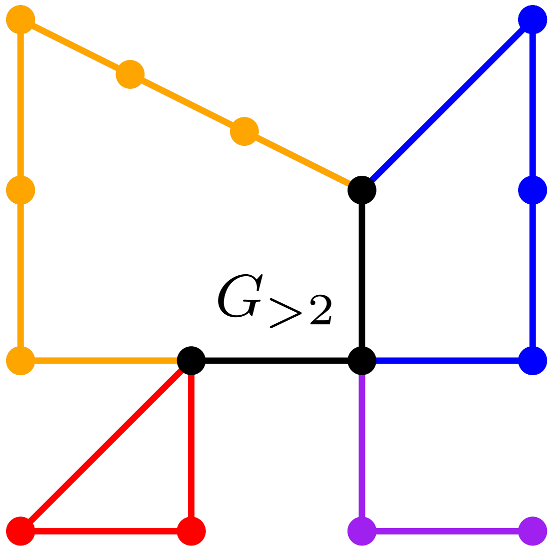


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

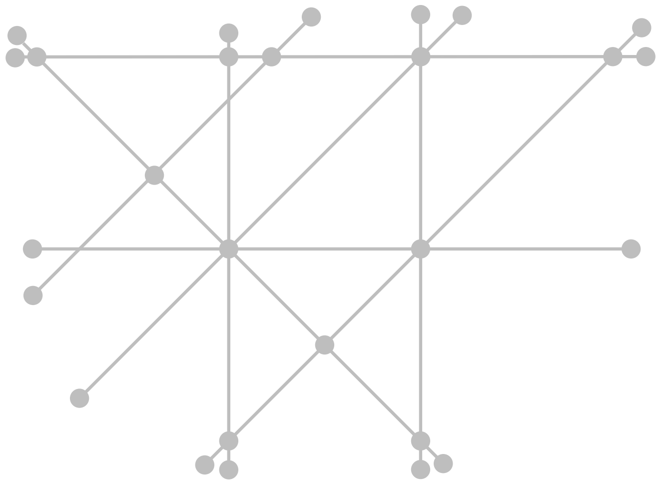
Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many



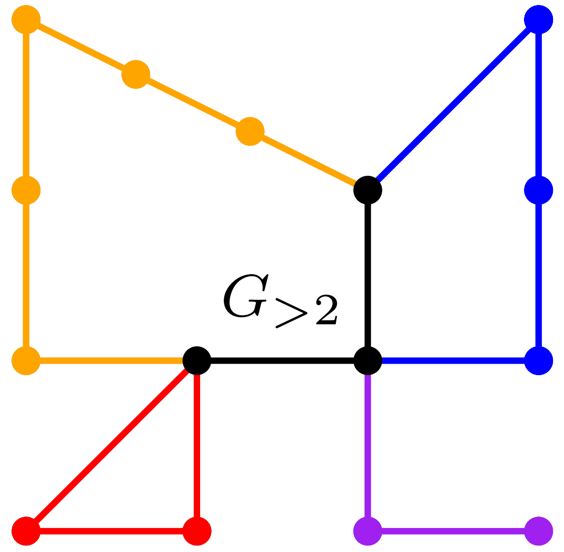
1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.



List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$



Split G into $|V_{>2}| \leq \binom{k}{2}$

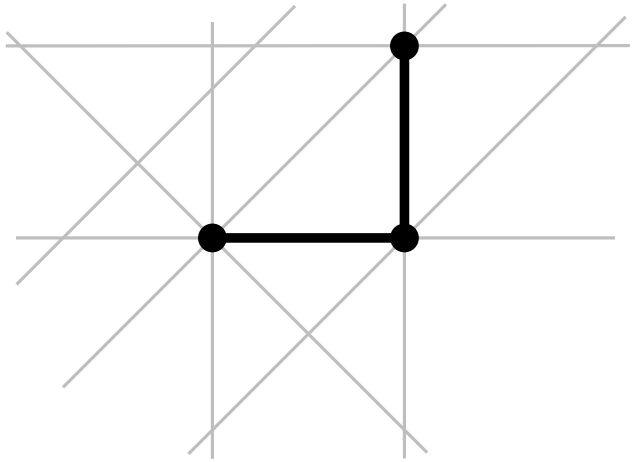
- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many

1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

   2. For each placement of $V_{>2}$ on crossings of the lines

      If this yields a planar drawing of $G_{>2}$ with edges on the lines

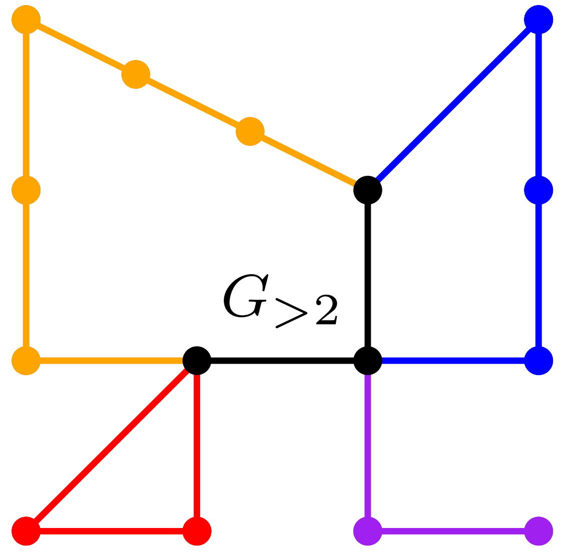


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many

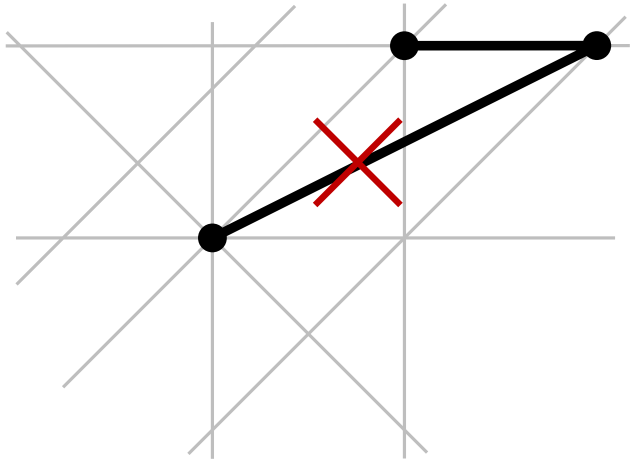


1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

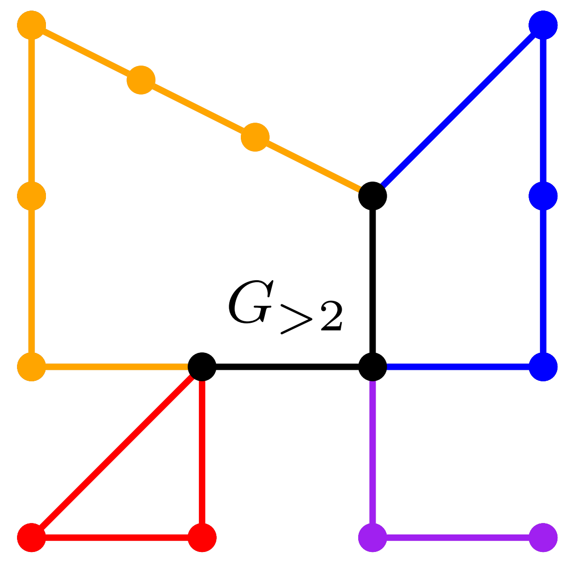
   2. For each placement of $V_{>2}$ on crossings of the lines

      If this yields a planar drawing of $G_{>2}$ with edges on the lines



List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$



Split G into $|V_{>2}| \leq \binom{k}{2}$

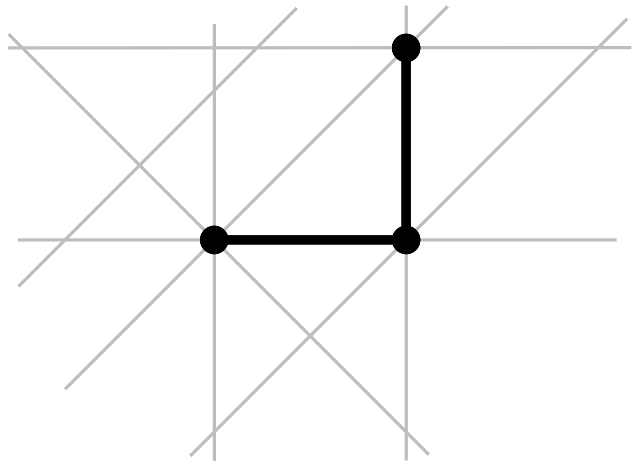
- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many

1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

   If this yields a planar drawing of $G_{>2}$ with edges on the lines

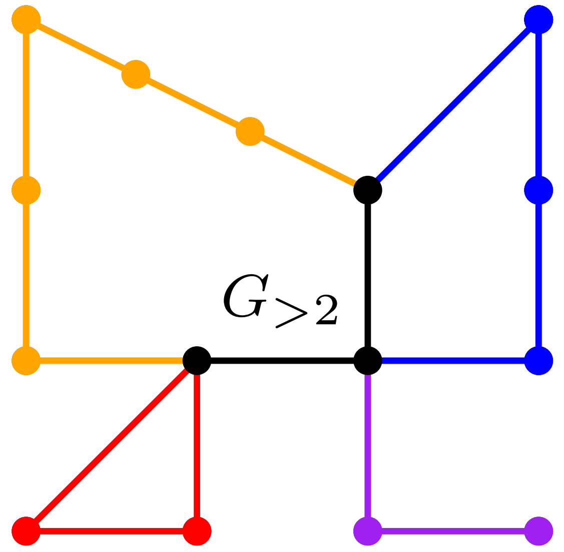


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many



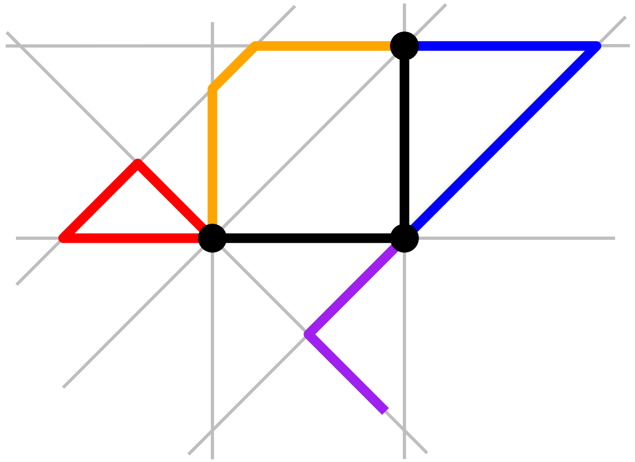
1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

   2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

      If this yields a planar drawing of $G_{>2}$ with edges on the lines

      3. for each routing of the degree-2-paths that yield a planar drawing of G with $\leq k$ segments



List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many

1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

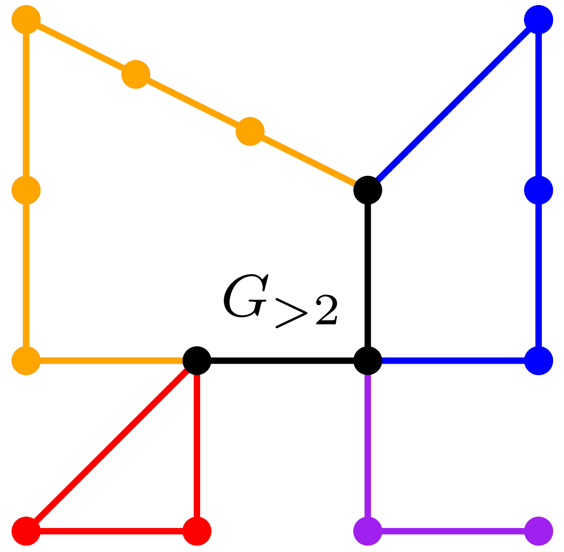
   2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

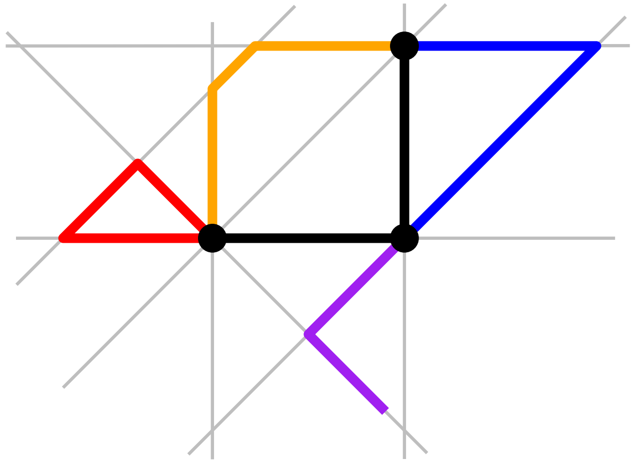
      If this yields a planar drawing of $G_{>2}$ with edges on the lines

      3. for each routing of the degree-2-paths that yield a planar drawing of G with $\leq k$ segments

$\left((2k(k-1))^k\right)^{k^3}$ many

- $2k(k-1)$: number of choices per path and crossing
- exponent k: number of segments per light path
- exponent k^3: number of light paths





List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many

1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

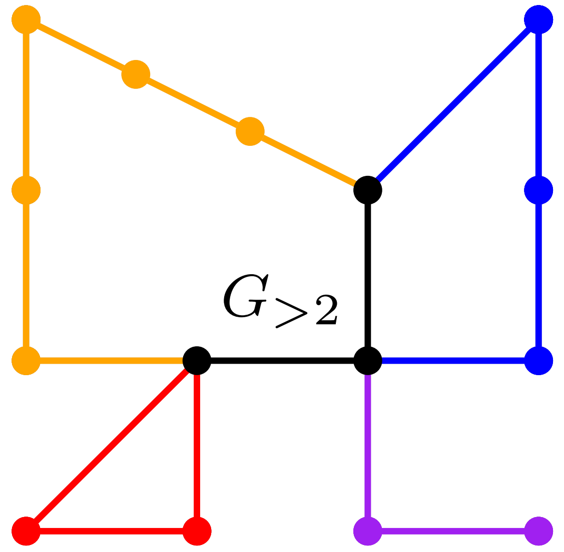
   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

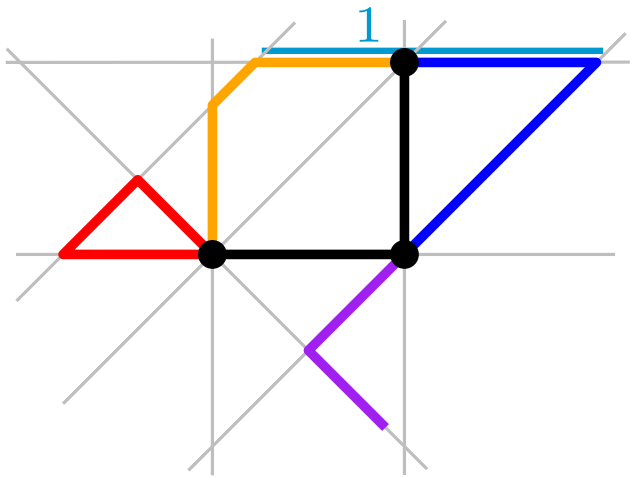
2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

   If this yields a planar drawing of $G_{>2}$ with edges on the lines

3. for each routing of the degree-2-paths that yield a planar drawing of G with $\leq k$ segments $\left((2k(k-1))^k\right)^{k^3}$ many

4. for each labeling $s_1, \ldots, s_k$ of the segments



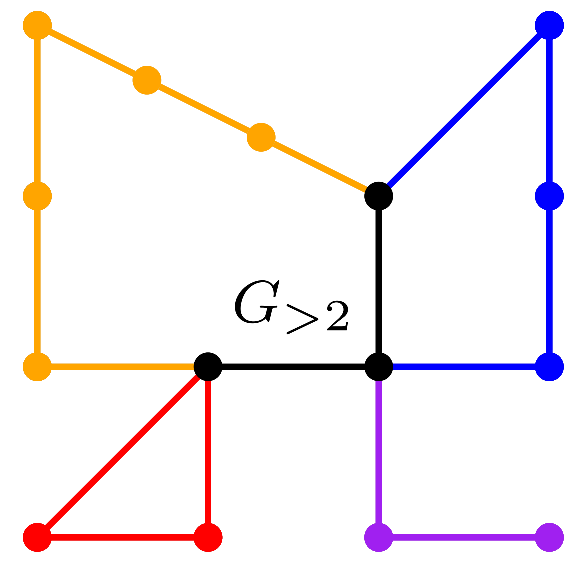


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many



1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

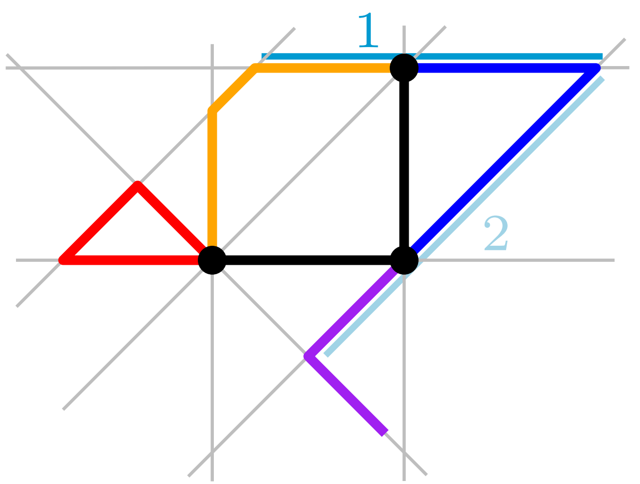
   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

   2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

      If this yields a planar drawing of $G_{>2}$ with edges on the lines

      3. for each routing of the degree-2-paths that yield a planar drawing of G with $\leq k$ segments $\left((2k(k-1))^k\right)^{k^3}$ many

         4. for each labeling $s_1, \ldots, s_k$ of the segments

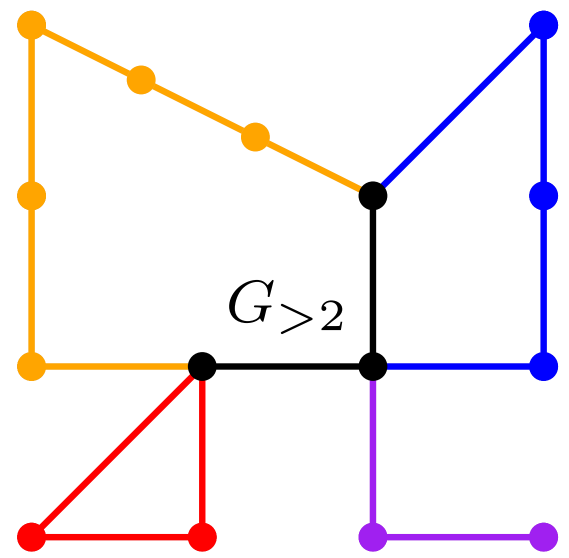


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $\quad |V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many



1. For each arrangement of k lines $\quad \mathcal{O}(2^{k^2})$ arrangements

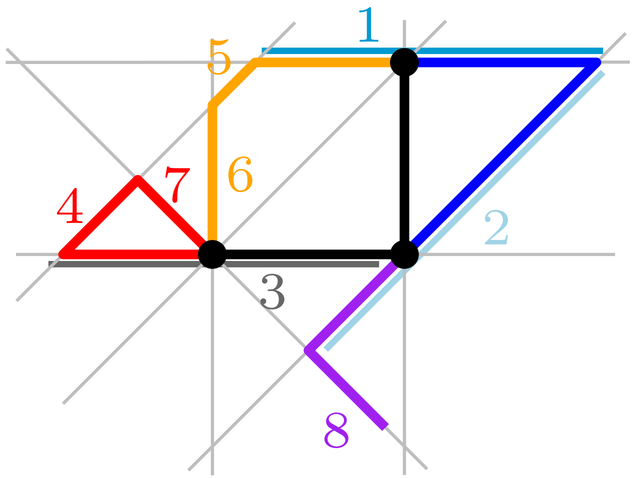
   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

   2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

      If this yields a planar drawing of $G_{>2}$ with edges on the lines

      3. for each routing of the degree-2-paths that yield a planar drawing of G with $\leq k$ segments $\quad \left((2k(k-1))^k\right)^{k^3}$ many

         4. for each labeling $s_1, \ldots, s_k$ of the segments $\quad k!$ many

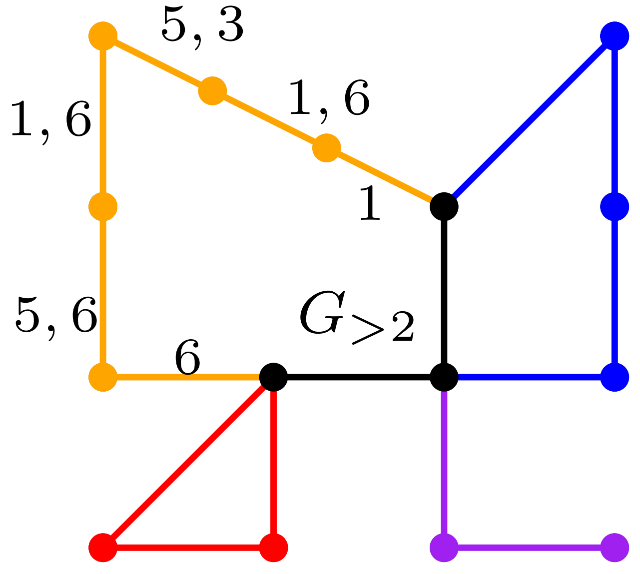


List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many



1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

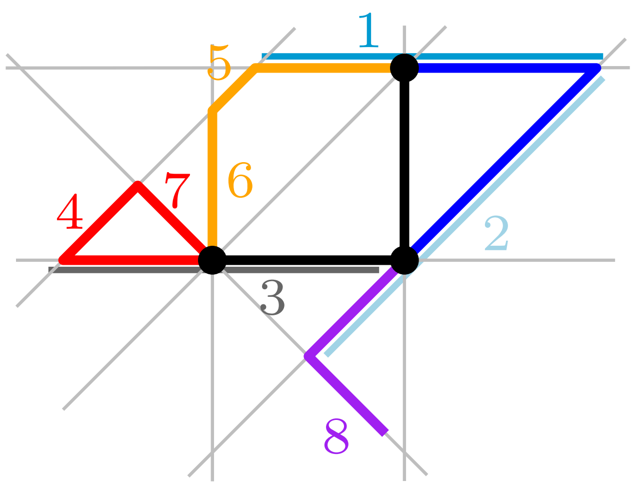
   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

   2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

      If this yields a planar drawing of $G_{>2}$ with edges on the lines

      3. for each routing of the degree-2-paths that yield a planar drawing of G with $\leq k$ segments $\left((2k(k-1))^k\right)^{k^3}$ many

         4. for each labeling $s_1, \ldots, s_k$ of the segments $k!$ many



List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many

1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves,
   and all coverings with k edge-disjoint paths.
   Use Renegar to check wether they are stretchable.

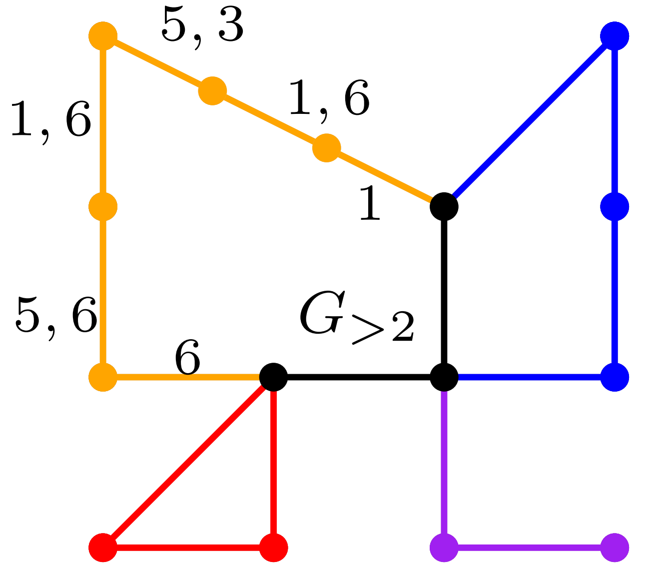
2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

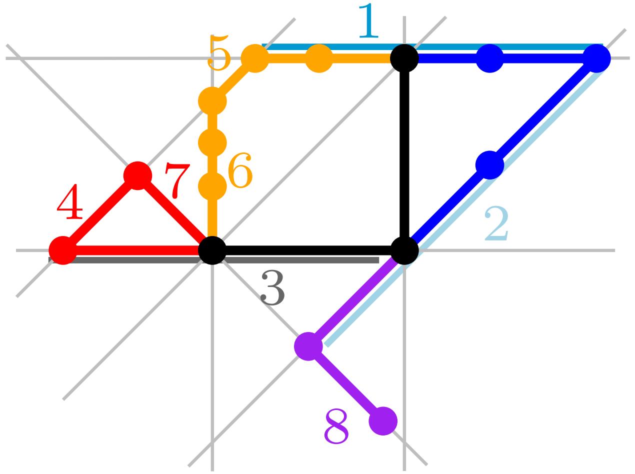
   If this yields a planar drawing of $G_{>2}$ with edges on the lines

3. for each routing of the degree-2-paths that yield a planar drawing of G with $\leq k$ segments $\left((2k(k-1))^k\right)^{k^3}$ many

4. for each labeling $s_1, \ldots, s_k$ of the segments $k!$ many

   use dynamic programming to test whether
   each light path P can be realized on route S obeying $L(e)$





List-Incidence Segment Number is in FPT wrt. Segment Number

Input: planar graph $G = (V, E)$, integer k, lists $L(e) \subseteq [k]$, $e \in E$

Split G into $|V_{>2}| \leq \binom{k}{2}$

- graph $G_{>2} = (V_{>2}, E_{>2})$ induced by vertices of degree > 2
- light paths of degree 2. at most $2k \cdot \binom{k}{2}$ many

1. For each arrangement of k lines $\mathcal{O}(2^{k^2})$ arrangements

   Construct all plane graphs on $\leq \binom{k}{2} + 2k$ vertices with $2k$ leaves, and all coverings with k edge-disjoint paths. Use Renegar to check wether they are stretchable.

2. For each placement of $V_{>2}$ on crossings of the lines $\binom{k}{2}!$ many

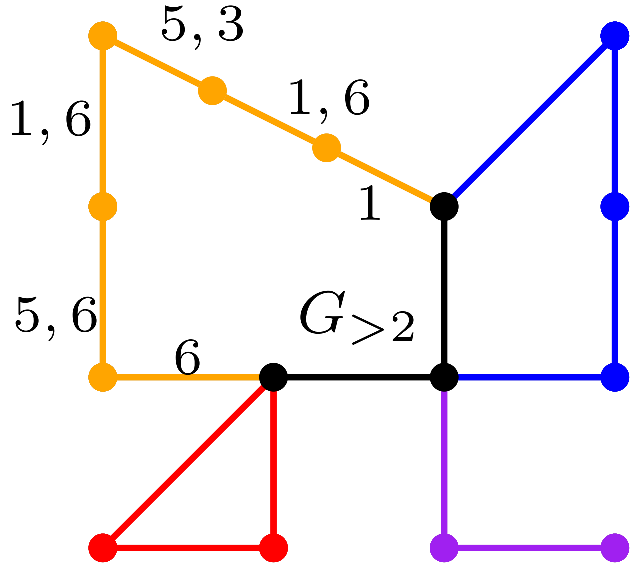
   If this yields a planar drawing of $G_{>2}$ with edges on the lines

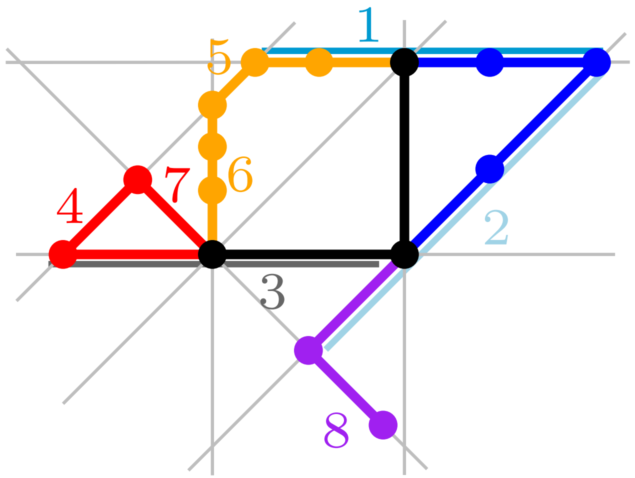
3. for each routing of the degree-2-paths that yield a planar drawing of G with $\leq k$ segments $\left((2k(k-1))^k\right)^{k^3}$ many

4. for each labeling $s_1, \ldots, s_k$ of the segments $k!$ many

   use dynamic programming to test whether each light path P can be realized on route S obeying $L(e)$

   $\mathcal{O}(|P| \cdot |S| \cdot \log k)$ per path $\rightsquigarrow \mathcal{O}(n \cdot k \cdot \log k)$ for all paths





Summary and Open Problems

- The segment number can be determined in linear time for banana trees.

Summary and Open Problems

- The segment number can be determined in linear time for banana trees.
- SEGMENT NUMBER is in FPT parameterized by
  - the vertex cover number
  - the segment number
  - the line cover number

Summary and Open Problems

- The segment number can be determined in linear time for banana trees.
- Segment Number is in FPT parameterized by
  - the vertex cover number
  - the segment number
  - the line cover number
- The list-coloring versions of Segment Number and Line Cover Number are in FPT parameterized by their natural parameters

Summary and Open Problems

- The segment number can be determined in linear time for banana trees.
- SEGMENT NUMBER is in FPT parameterized by
  - the vertex cover number
  - the segment number
  - the line cover number
- The list-coloring versions of SEGMENT NUMBER and LINE COVER NUMBER are in FPT parameterized by their natural parameters
- It would be interesting to investigate SEGMENT NUMBER w.r.t. other parameters
  - treewidth (complexity even for tree-width 2 unknown)
  - cluster deletion number
    minimum number of vertices that have to be removed s.t.
    remainder is union of disjoint cliques.

Summary and Open Problems

- The segment number can be determined in linear time for banana trees.
- Segment Number is in FPT parameterized by
  - the vertex cover number
  - the segment number
  - the line cover number
- The list-coloring versions of Segment Number and Line Cover Number are in FPT parameterized by their natural parameters
- It would be interesting to investigate Segment Number w.r.t. other parameters
  - treewidth (complexity even for tree-width 2 unknown)
  - cluster deletion number
    minimum number of vertices that have to be removed s.t. remainder is union of disjoint cliques.

Thank you