Bounding the Treewidth of Outer k-Planar Graphs via Triangulations

Oksana Firman
(Universität Würzburg)

Grzegorz Gutowski
(Jagiellonian University)

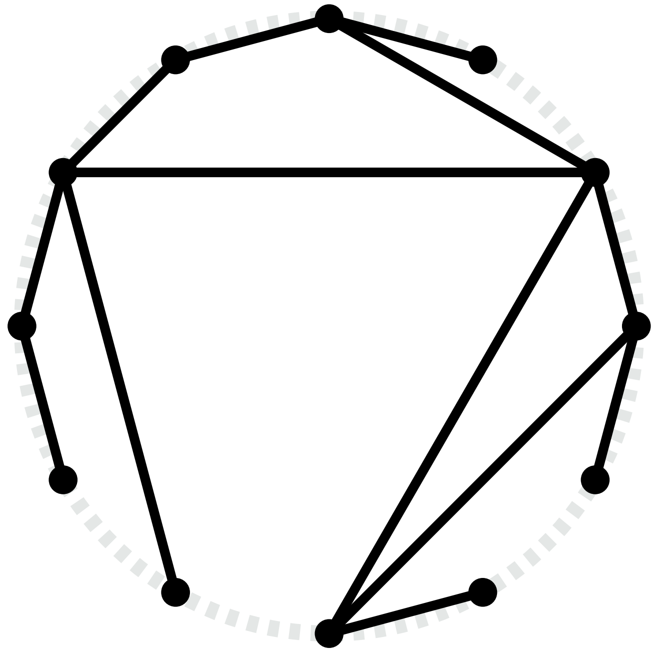
Myroslav Kryven
(University of Manitoba)

Yuto Okada
(Nagoya University)

Alexander Wolff
(Universität Würzburg)

2024-09-19 @ GD 2024

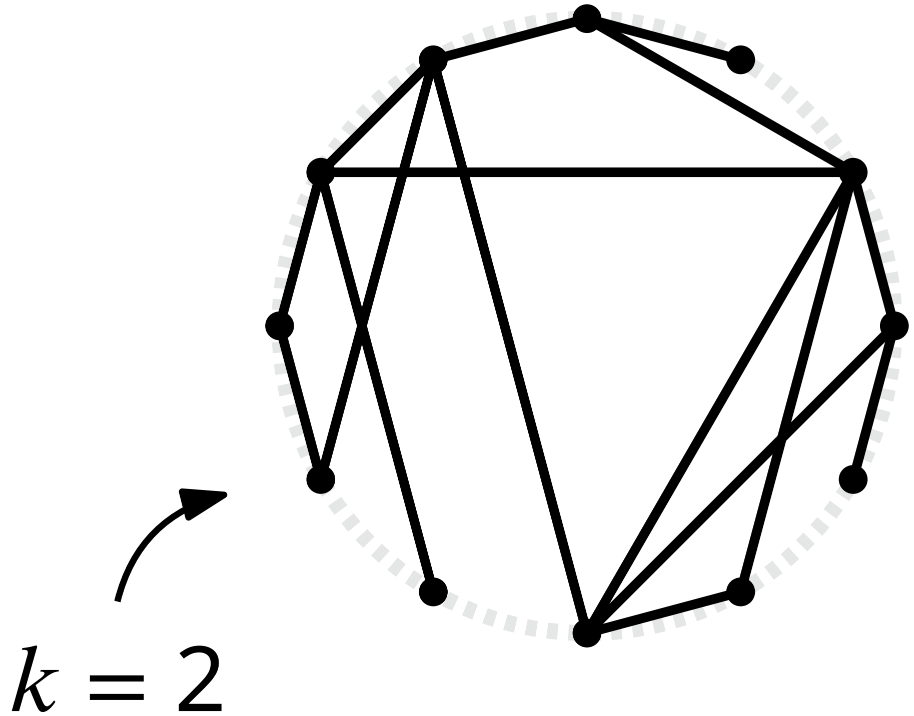
Outer k-Planar Graphs



Outerplanar graph

admits a drawing s.t.

- vertices on a circle
- straight-line
- no crossing



Outer k-planar graph

admits a drawing s.t.

- vertices on a circle
- straight-line
- k-planar

Abstract

Main Contribution

Any outer k-planar graph admits a good triangulation.

Improved Upper Bounds ← gives

- Separation Number

$$2k + 3 \rightarrow k + 2$$

[Chaplick et al., GD 2017] (almost) tight

- Treewidth

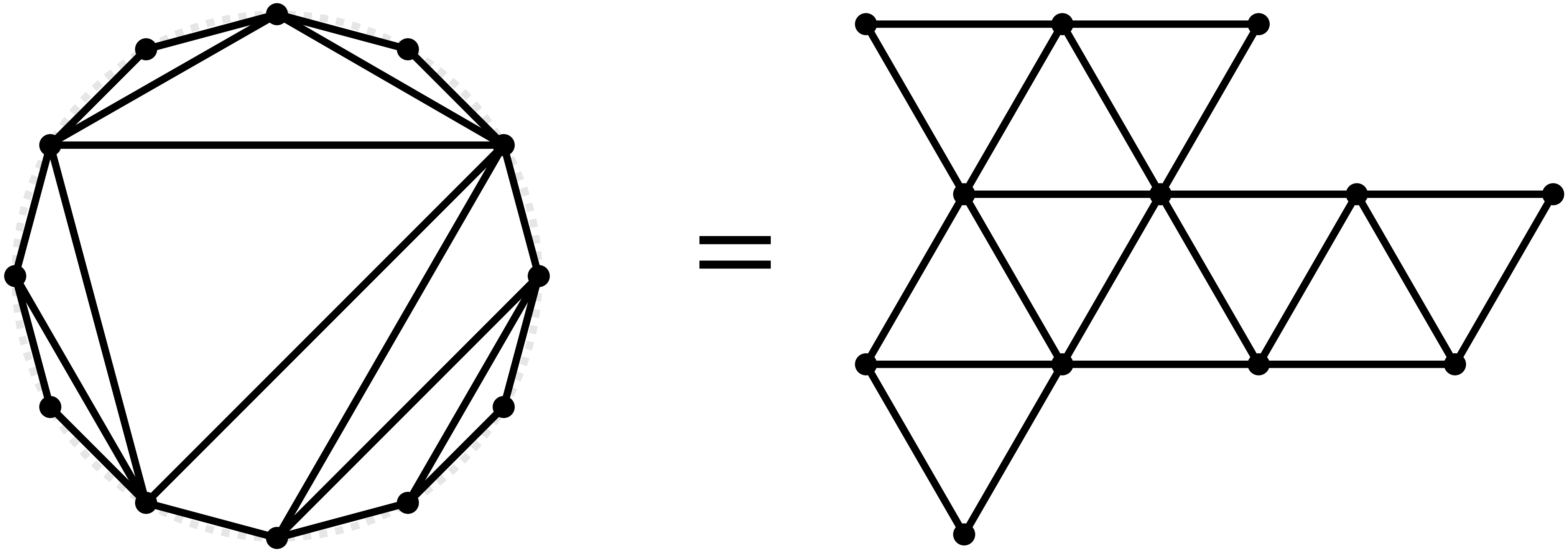
$$3k + 11 \rightarrow 1.5k + 2$$

[Wood and Telle, GD 2006] (lowerbound: $k + 2$)

... and some other results.

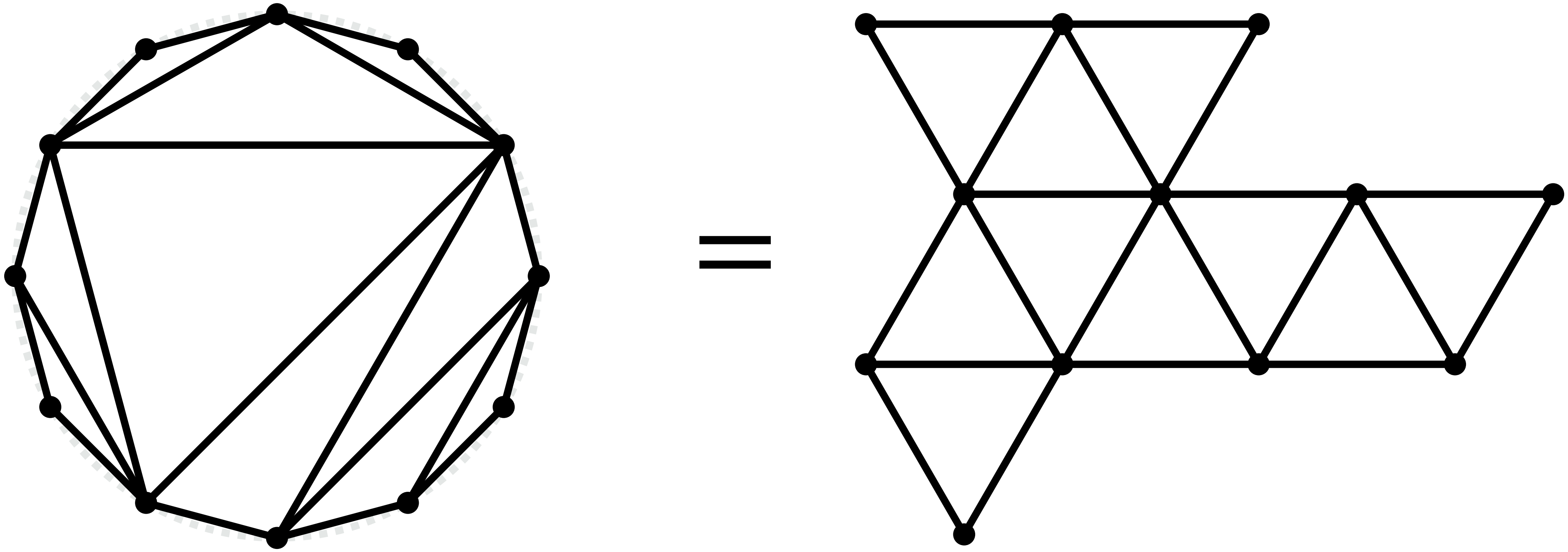
Triangulation on Outerplanar Graphs

After adding some edges until it becomes maximal, every inner face will be a triangle.



Triangulation on Outerplanar Graphs

After adding some edges until it becomes maximal, every inner face will be a triangle.

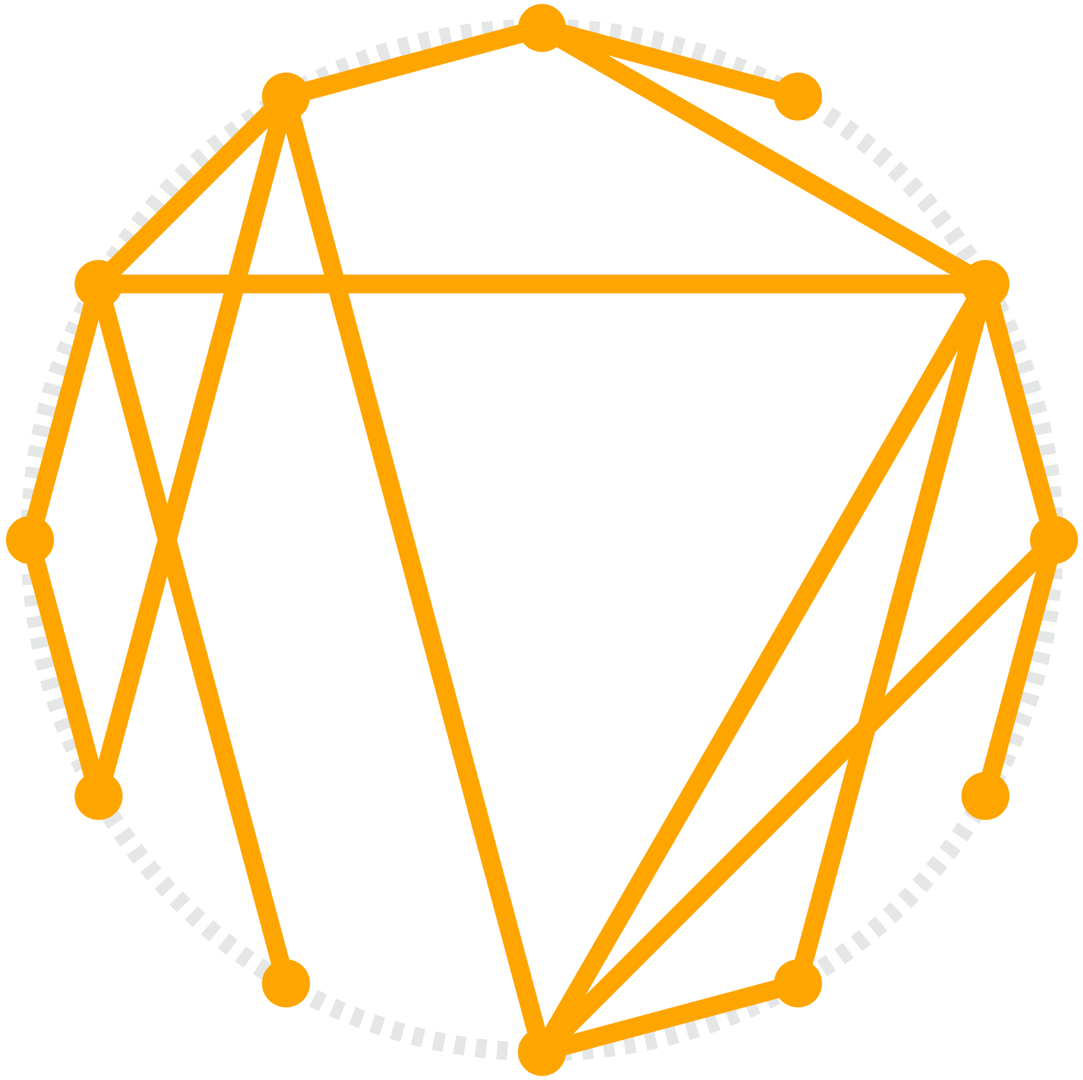


But outer k-planar graphs may have a crossing...?

Triangulation on Outer k-Planar Graphs

Lemma 6

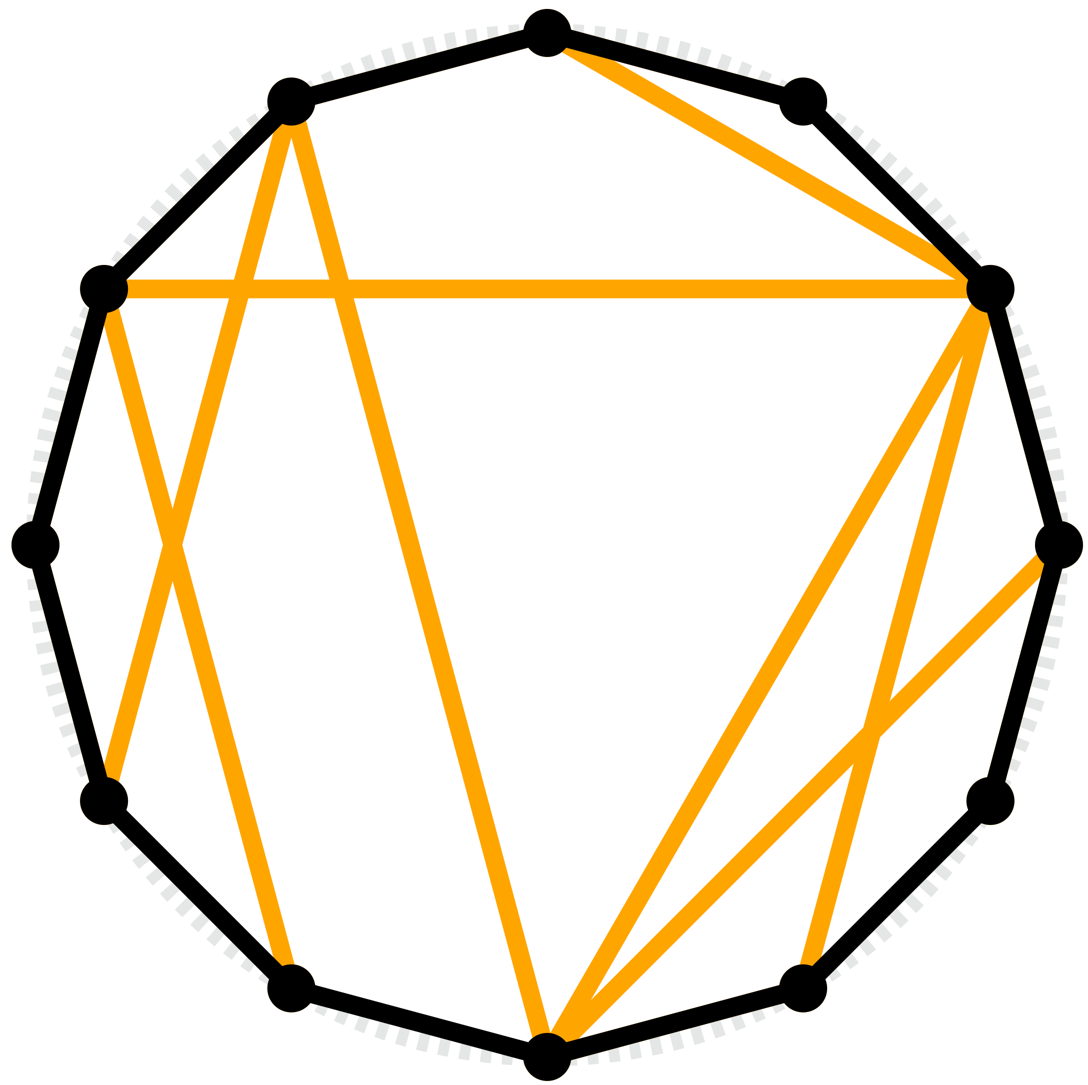
For every $k \geq 0$, given an outer k-planar drawing,



Triangulation on Outer k-Planar Graphs

Lemma 6

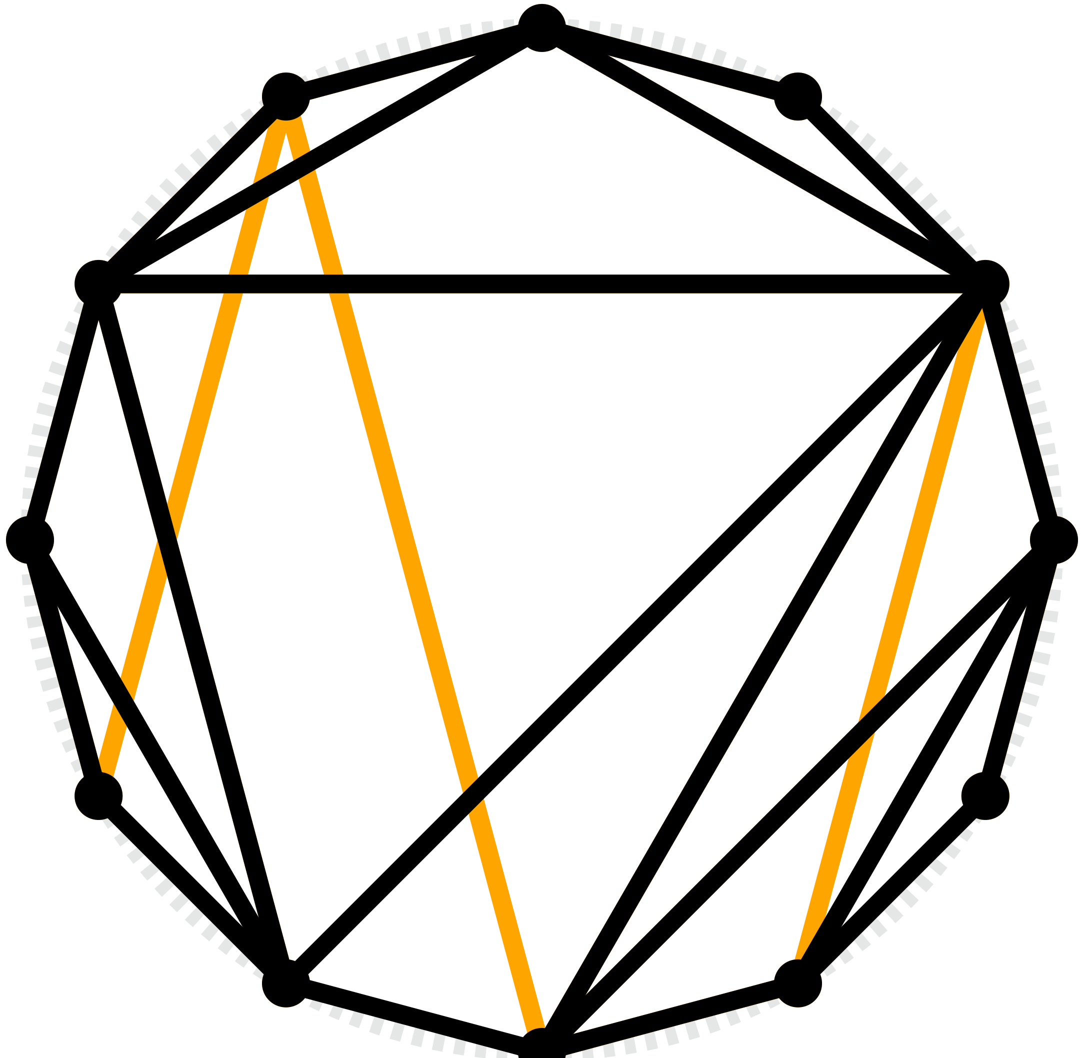
For every $k \geq 0$, given an outer k-planar drawing,
The outer cycle admits a triangulation s.t.



Triangulation on Outer k-Planar Graphs

Lemma 6

For every $k \geq 0$, given an outer k-planar drawing,
The outer cycle admits a triangulation s.t.
each edge crosses the original graph at most k times.

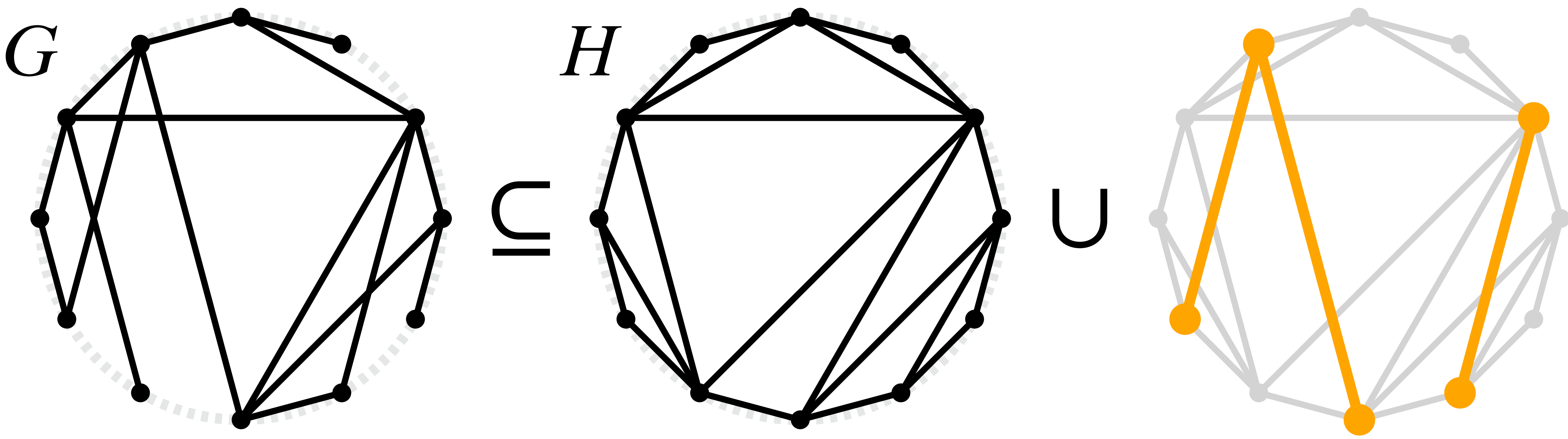


The Advantage of Triangulation

Key Point

The triangulation separates the given graph G into:

- (a subgraph of) maximal outerplane graph H
- crossing edges (sparsely distributed!!)

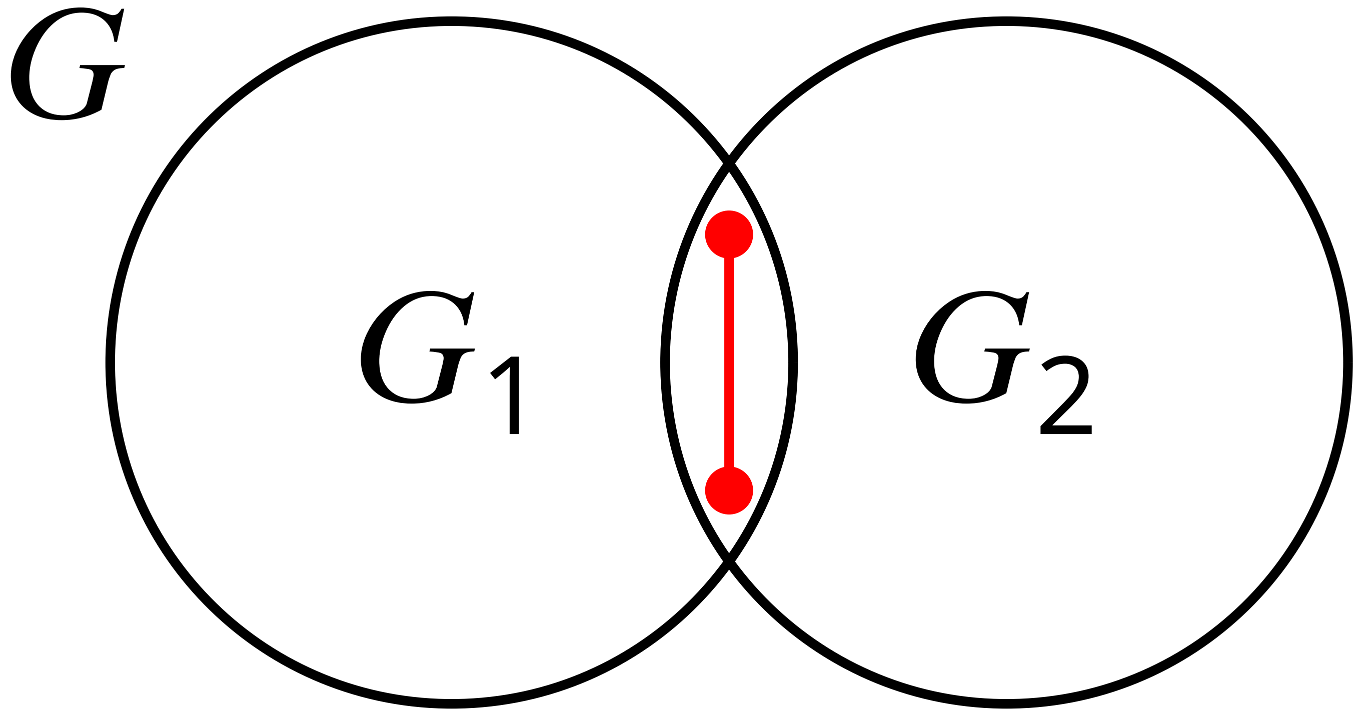


Every edge of H has at most k crossing edges.
Every triangle of H has at most $3k$ crossing edges.
$\rightarrow$ We can re-use properties of outerplanar graphs.

Upper Bound on Separation Number

On outerplanar graphs...

Every maximal outerplanar graph
is known to have an edge that is a balanced separator.



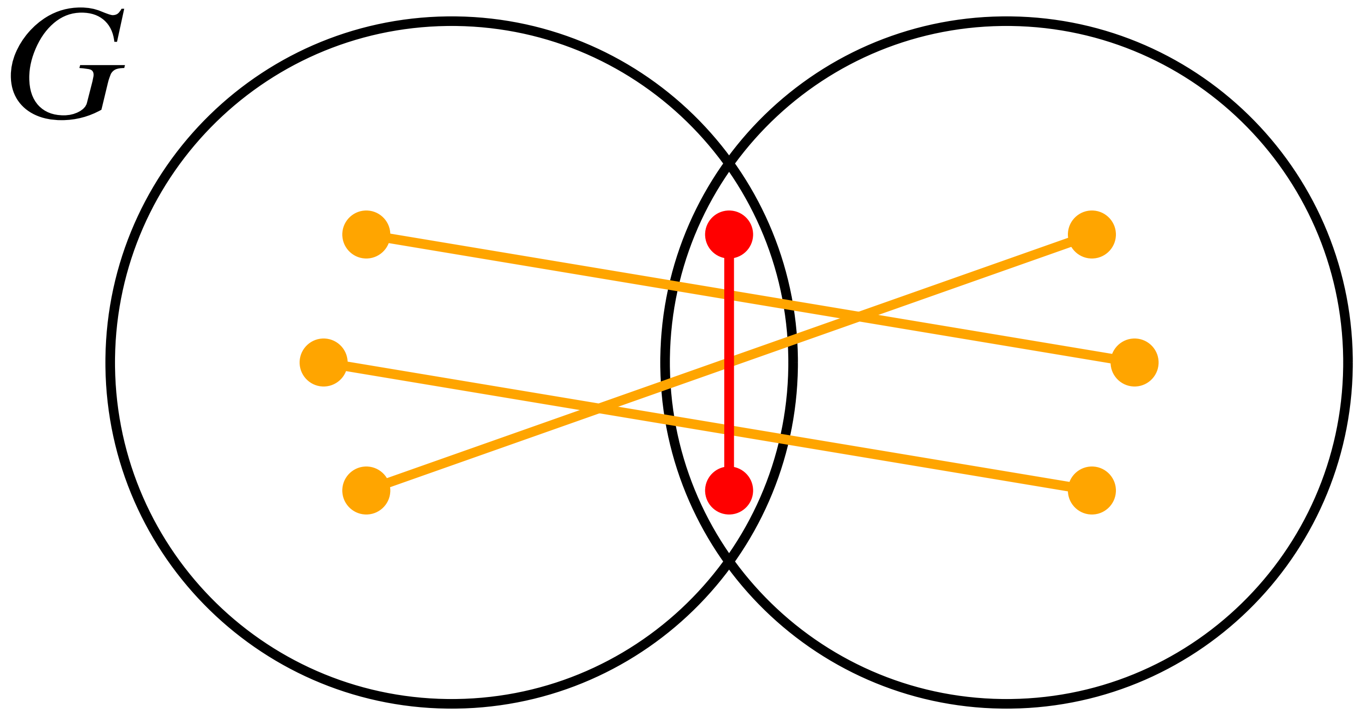
$|G_1|, |G_2| \leq \frac{2}{3}|G| + 2$

Note: A balanced separator of G is a vertex set
whose removal yields components of size at most $\frac{2}{3}|G|$.

Upper Bound on Separation Number

On outer k-planar graphs...

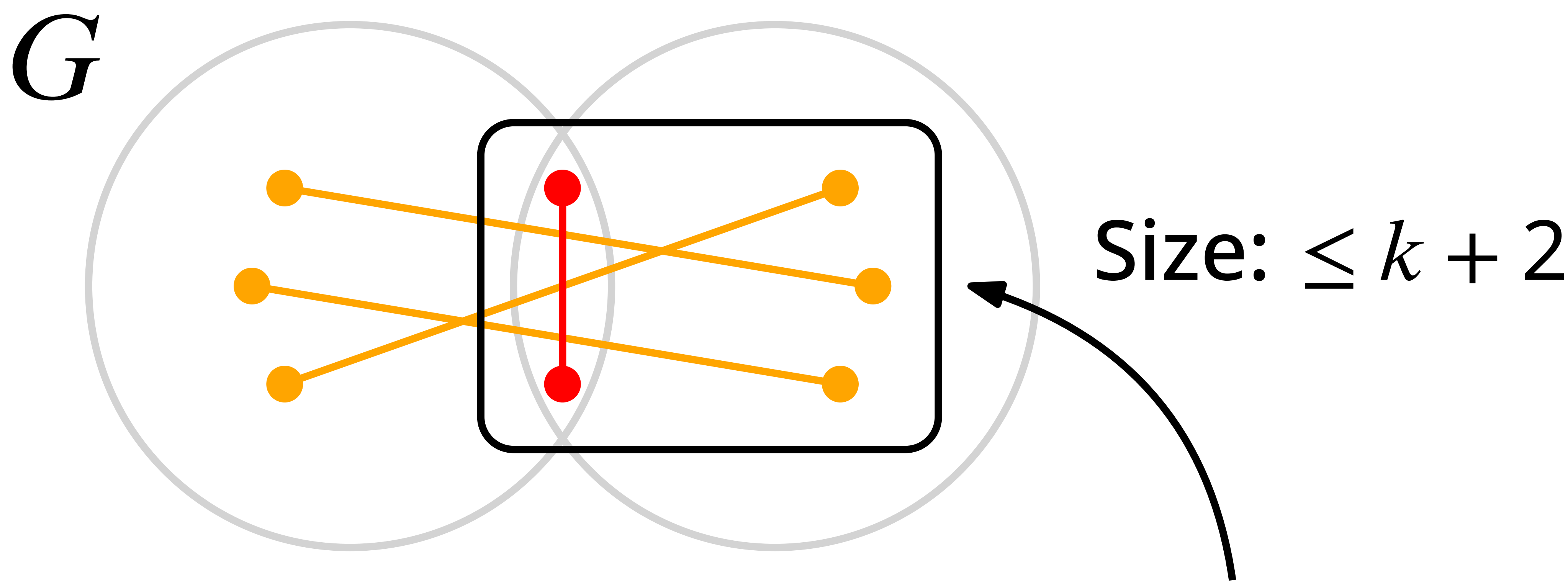
The maximal outerplane graph H also has
a balanced separator edge with $\leq k$ crossing edges.



Upper Bound on Separation Number

On outer k-planar graphs...

The maximal outerplane graph H also has
a balanced separator edge with $\leq k$ crossing edges.

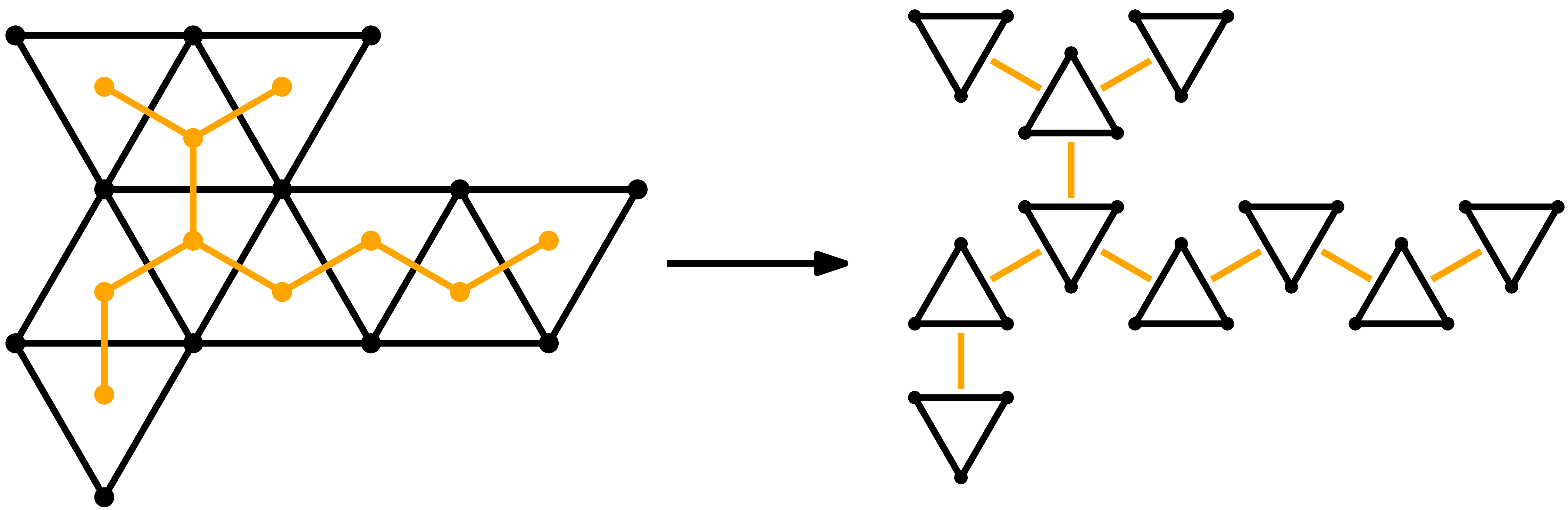


The edge and k endpoints form a balanced separator.
$\rightarrow$ separation number at most $k + 2$.

Upper Bound on Treewidth

On outerplanar graphs...

The weak dual of a maximal outerplane graph is a tree.

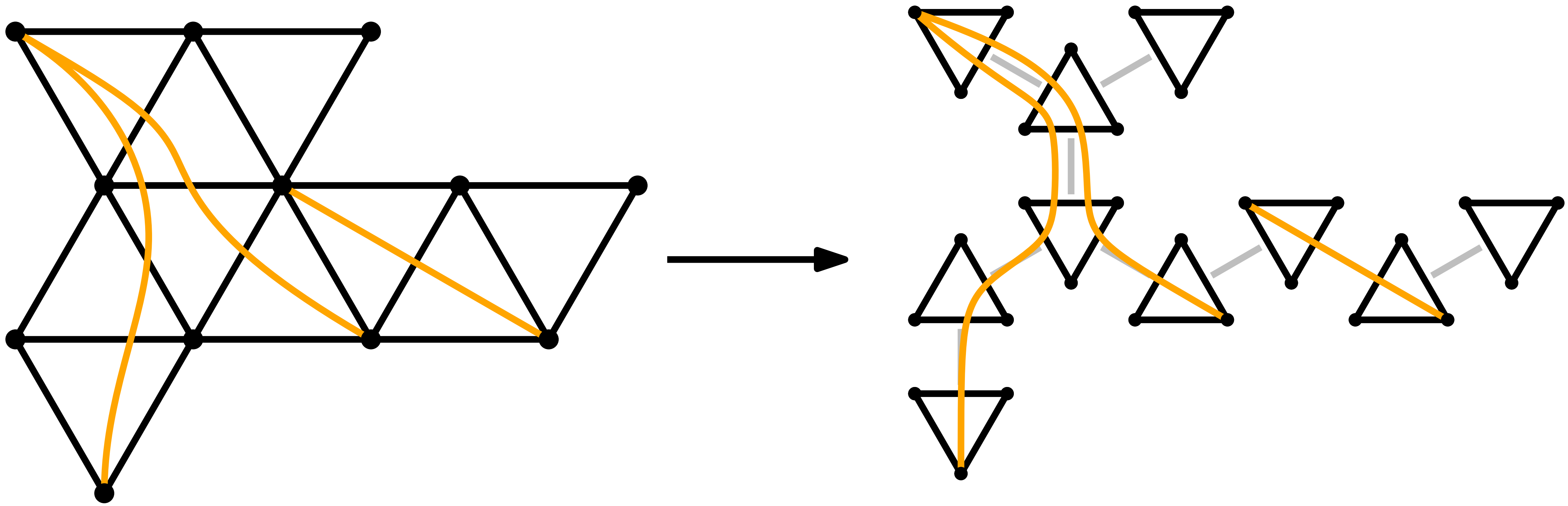


Replacing each vertex with the corresponding triangle yields a tree decomposition of width 2.

Upper Bound on Treewidth

On outer k-planar graphs...

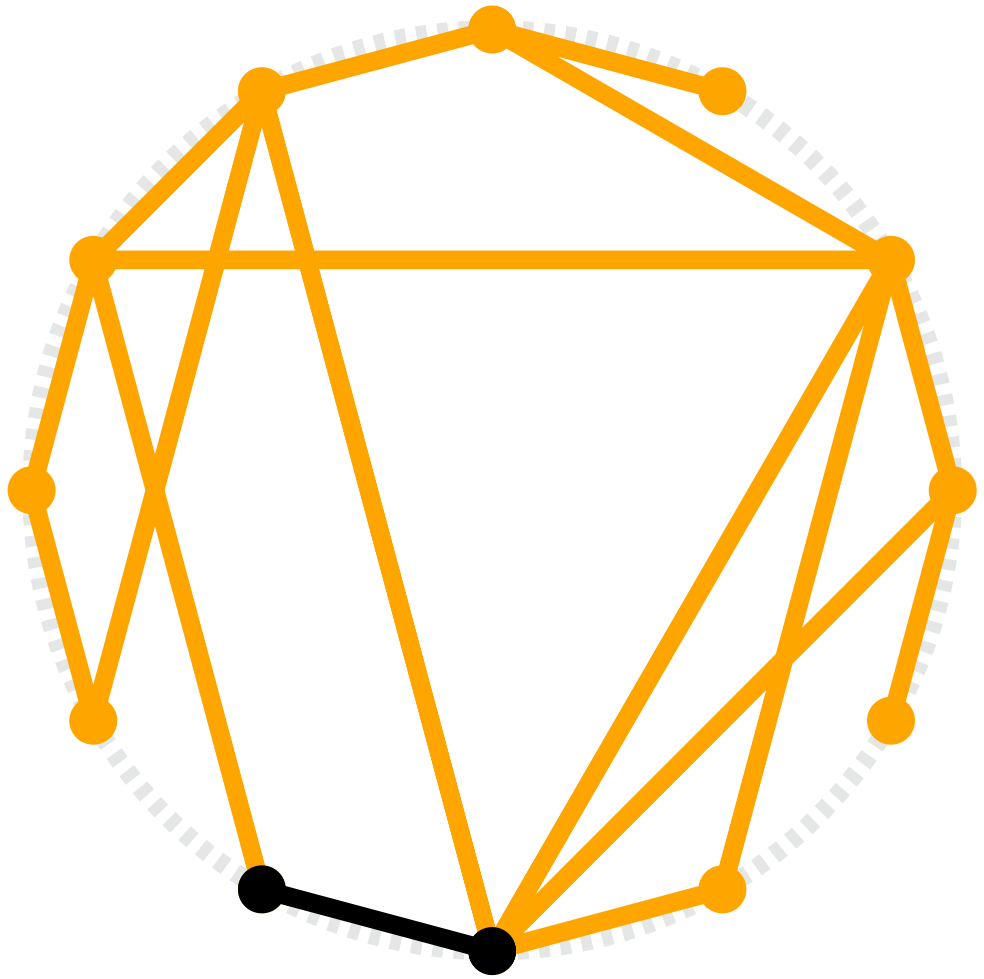
We use a tree decomposition of the graph H.
Then we only need to deal with the crossing edges.



Each triangle (= bag) has at most $3k$ crossing edges.
$\rightarrow$ treewidth $3k + 2$ (naïve), can be improved to $1.5k + 2$.

Construction of the Triangulation

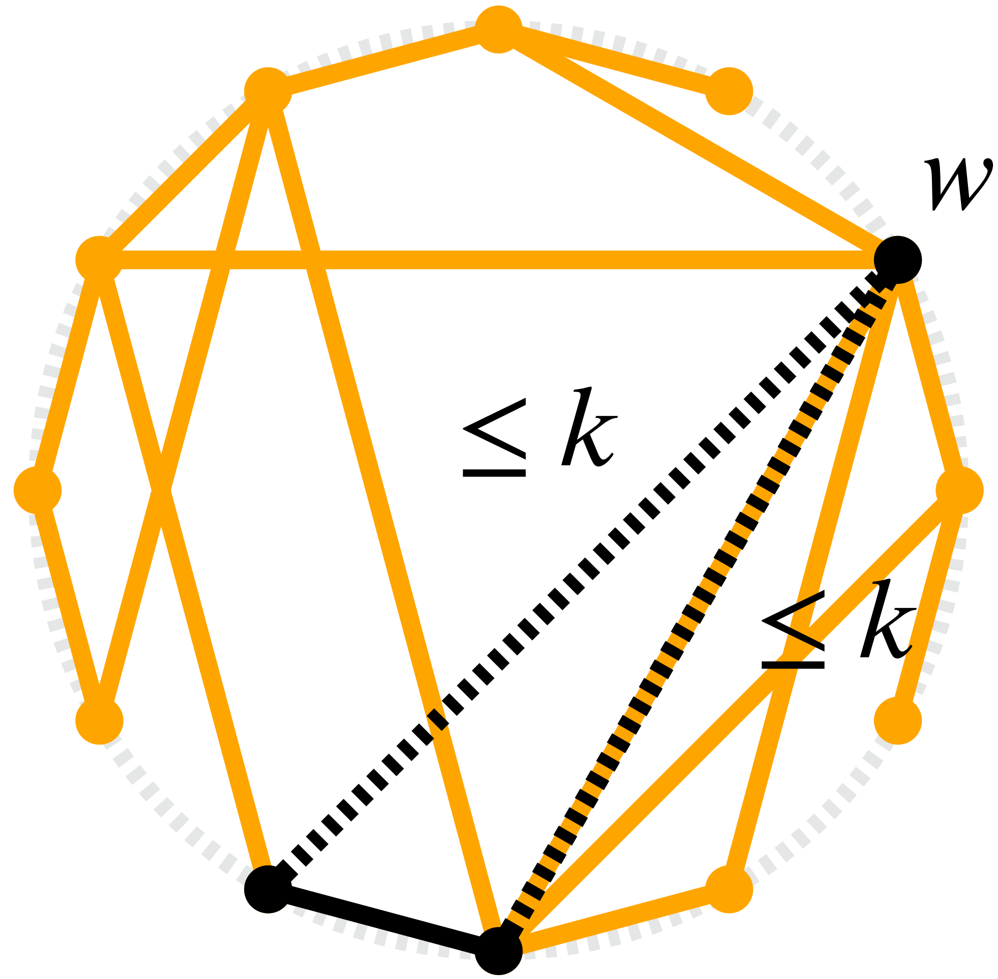
A greedy algorithm yields the triangulation.



1. Take any edge from the outer cycle.
2. There always exists a suitable vertex w.
3. Create a triangle with the vertex, and recurse.

Construction of the Triangulation

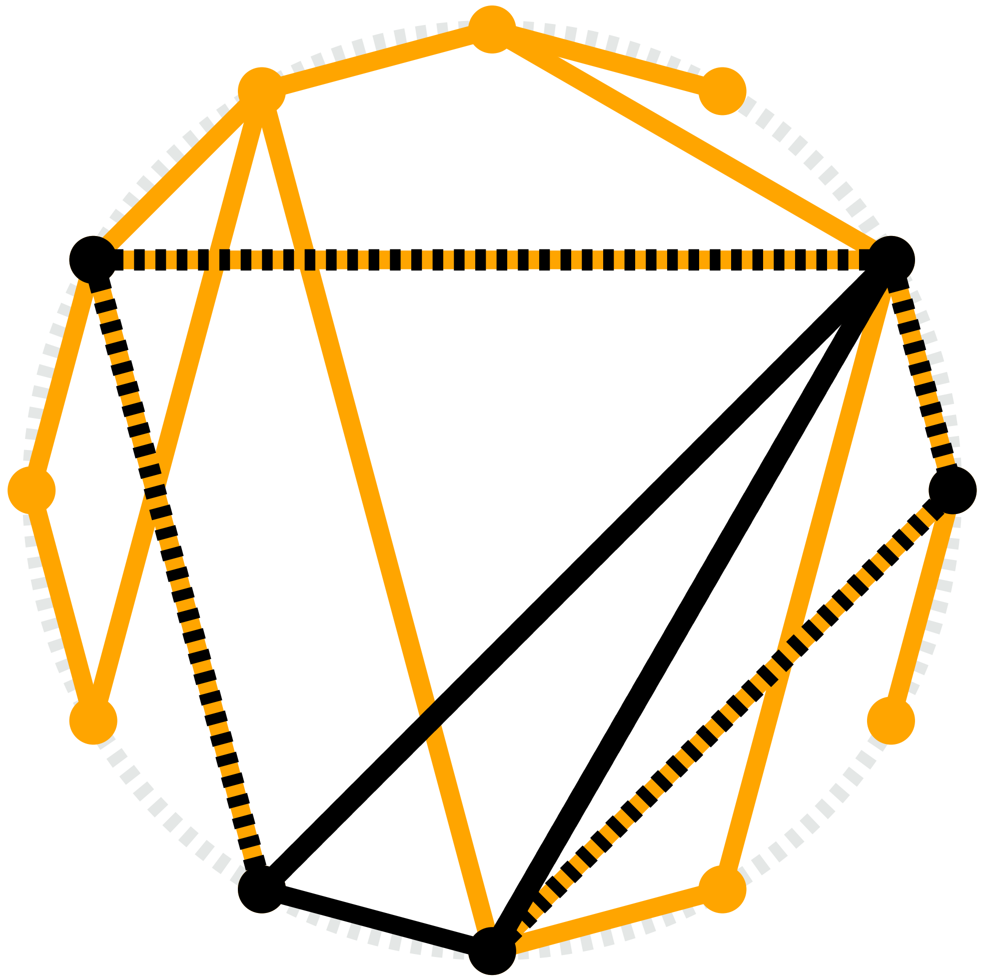
A greedy algorithm yields the triangulation.



1. Take any edge from the outer cycle.
2. There always exists a suitable vertex w.
3. Create a triangle with the vertex, and recurse.

Construction of the Triangulation

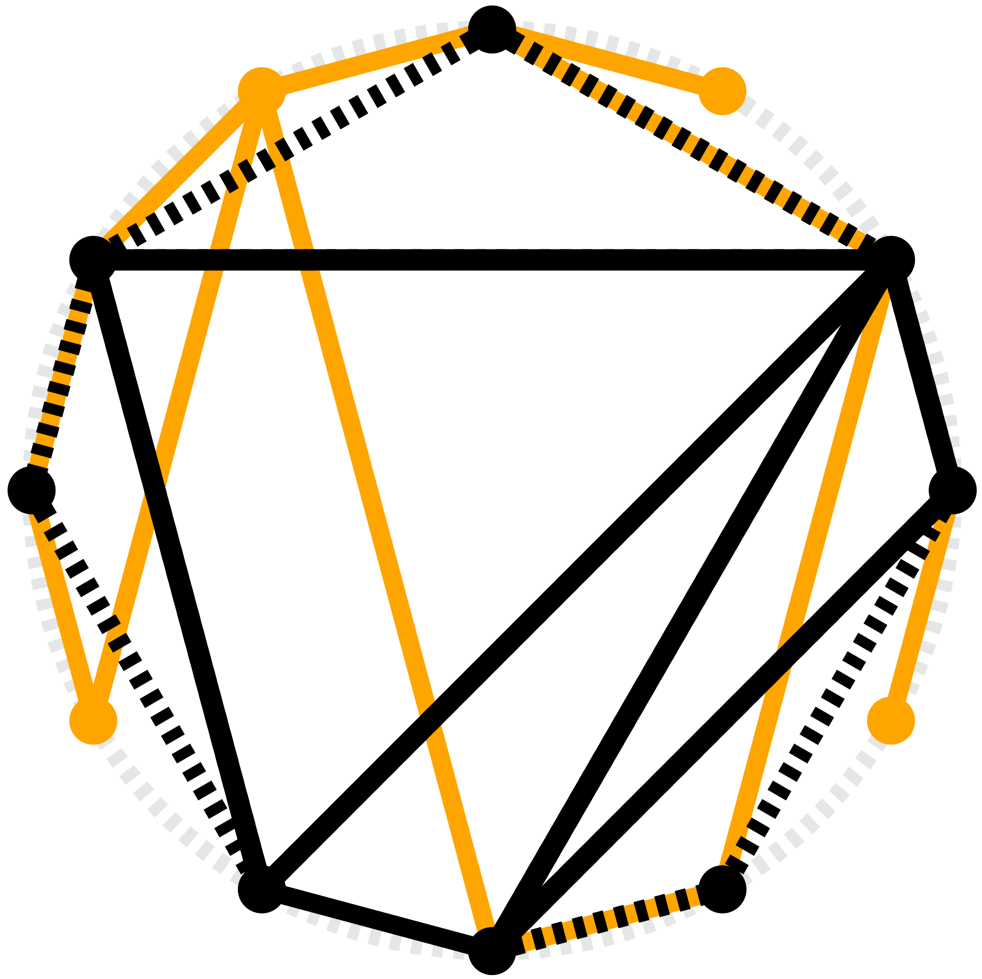
A greedy algorithm yields the triangulation.



1. Take any edge from the outer cycle.
2. There always exists a suitable vertex w.
3. Create a triangle with the vertex, and recurse.

Construction of the Triangulation

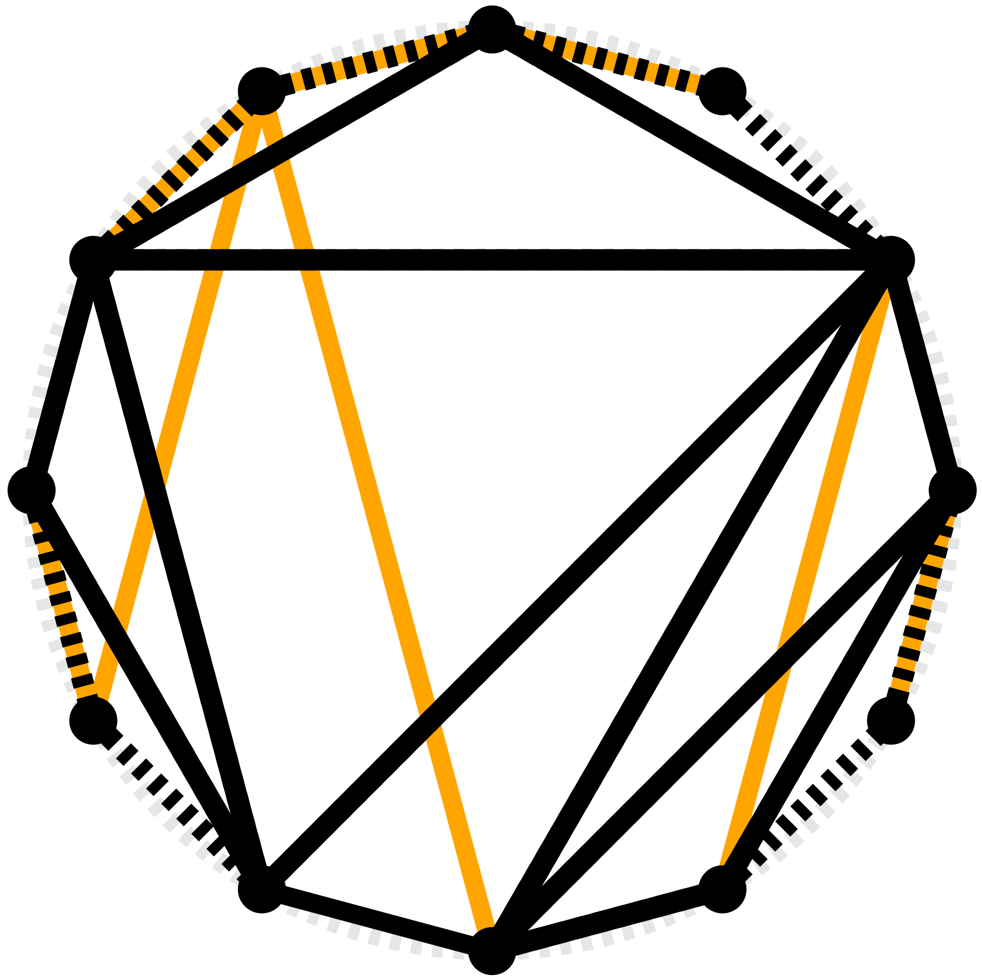
A greedy algorithm yields the triangulation.



1. Take any edge from the outer cycle.
2. There always exists a suitable vertex w.
3. Create a triangle with the vertex, and recurse.

Construction of the Triangulation

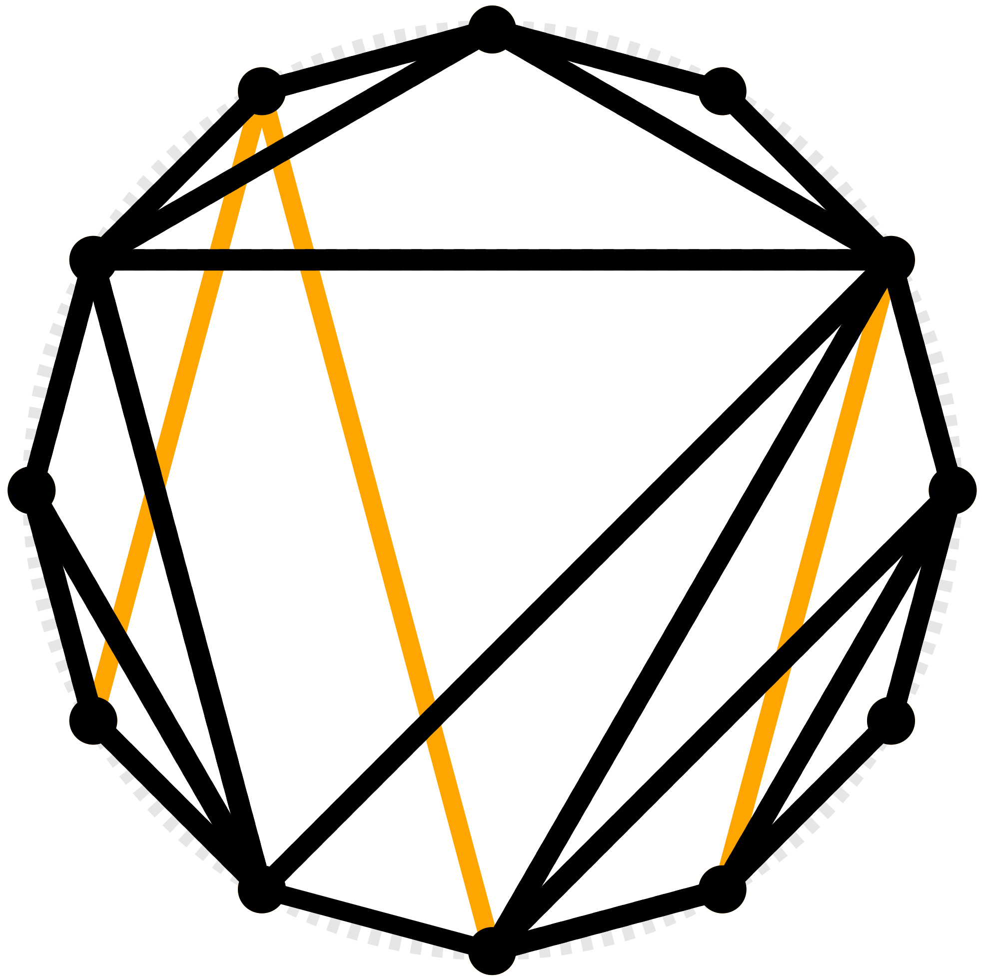
A greedy algorithm yields the triangulation.



1. Take any edge from the outer cycle.
2. There always exists a suitable vertex w.
3. Create a triangle with the vertex, and recurse.

Construction of the Triangulation

A greedy algorithm yields the triangulation.



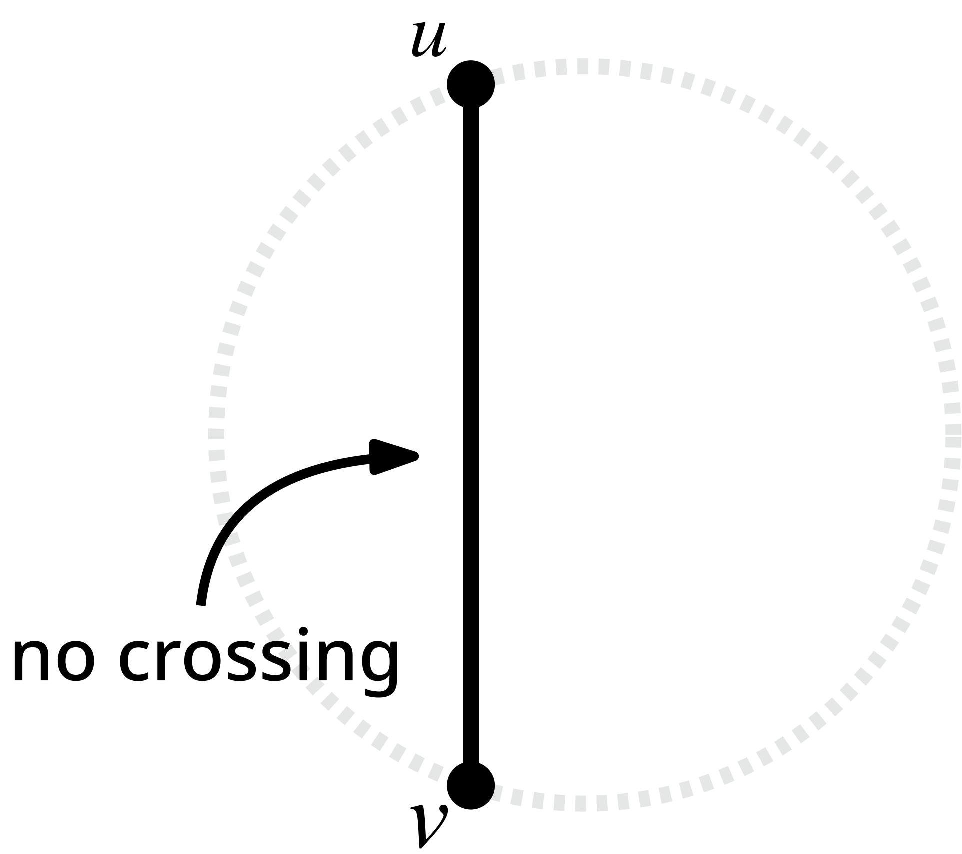
1. Take any edge from the outer cycle.
2. There always exists a suitable vertex w.
3. Create a triangle with the vertex, and recurse.

Correctness of the Greedy Algorithm

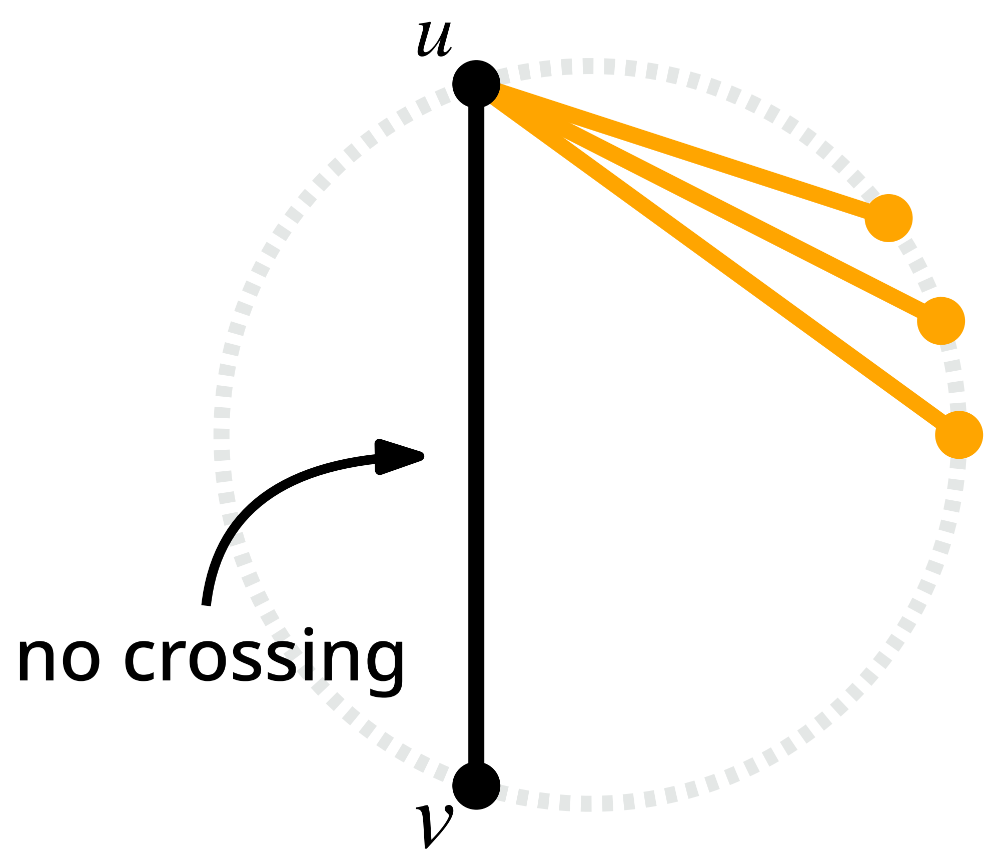
(in Lemma 6)

If $\{u, v\}$ crosses the original graph at most k times, there always exists a suitable vertex w.

Proved by case analysis. If $\{u, v\}$ has no crossing,



(i) u has no neighbor



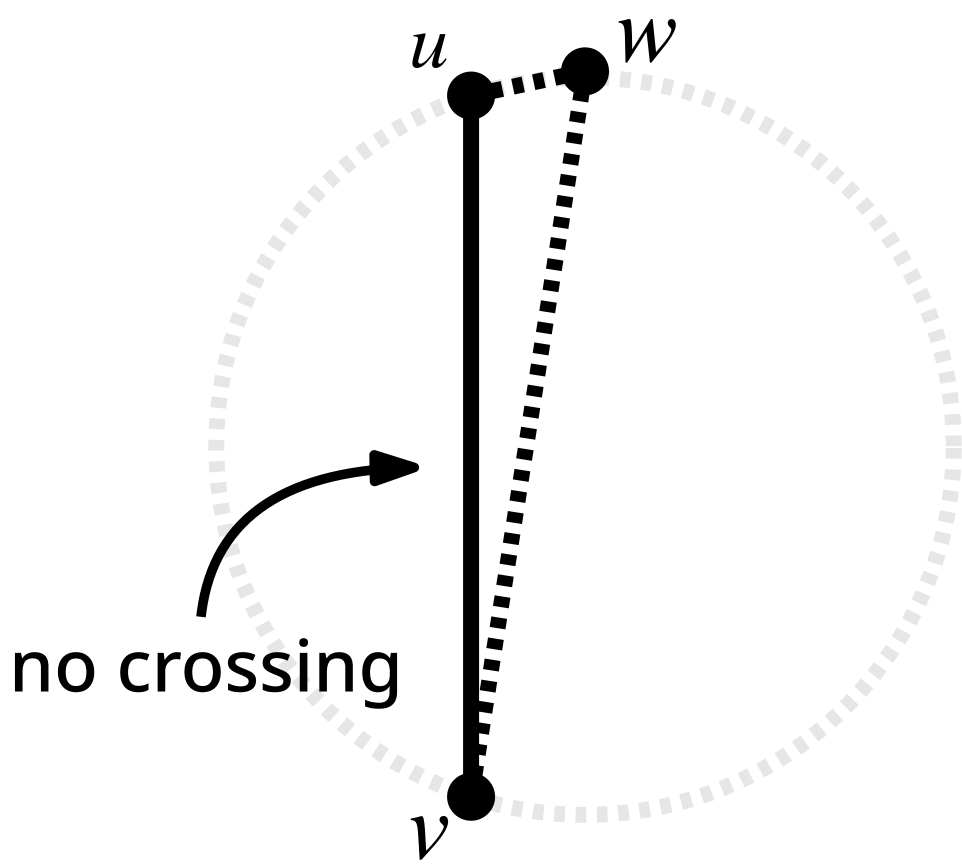
(ii) u has some neighbors

Correctness of the Greedy Algorithm

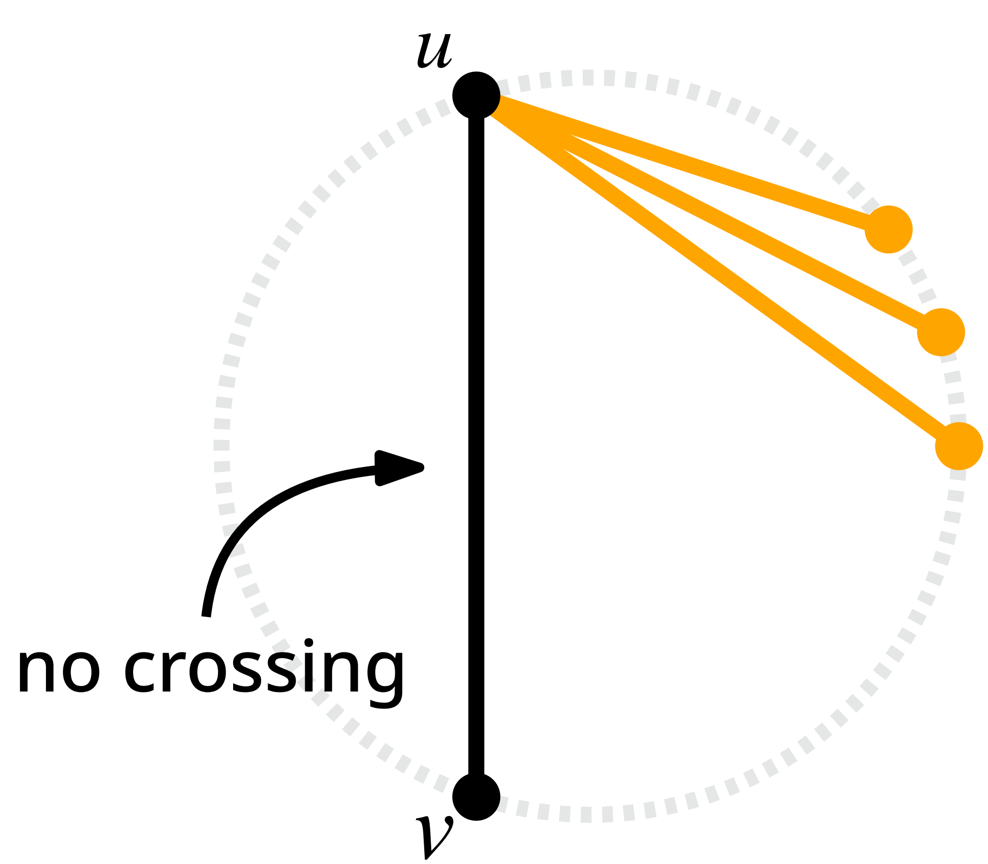
(in Lemma 6)

If $\{u, v\}$ crosses the original graph at most k times, there always exists a suitable vertex w.

Proved by case analysis. If $\{u, v\}$ has no crossing,



(i) u has no neighbor



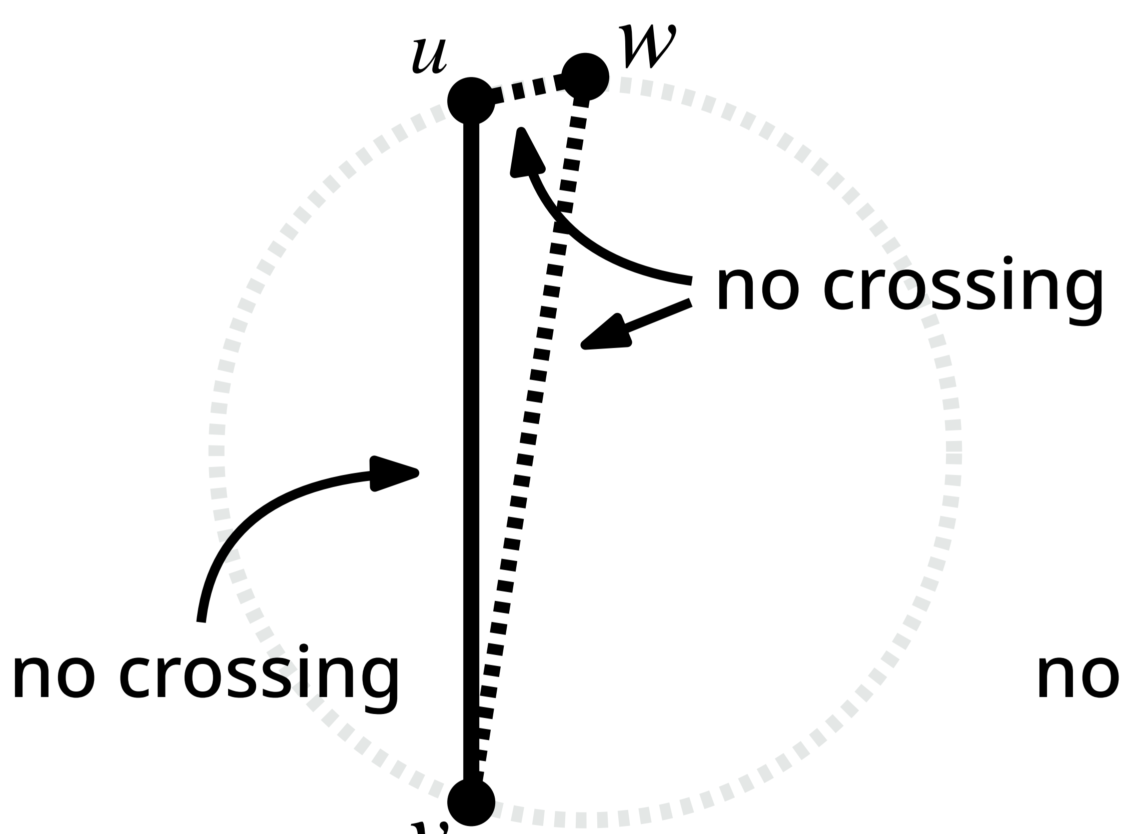
(ii) u has some neighbors

Correctness of the Greedy Algorithm

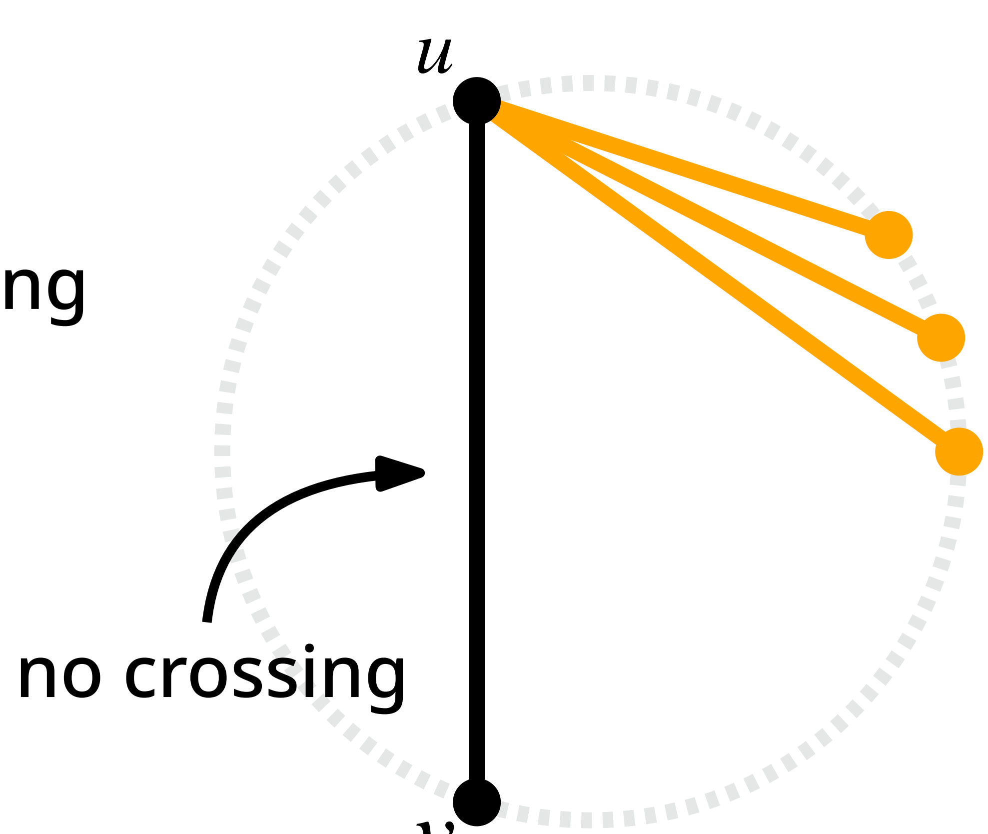
(in Lemma 6)

If $\{u, v\}$ crosses the original graph at most k times, there always exists a suitable vertex w.

Proved by case analysis. If $\{u, v\}$ has no crossing,



(i) u has no neighbor



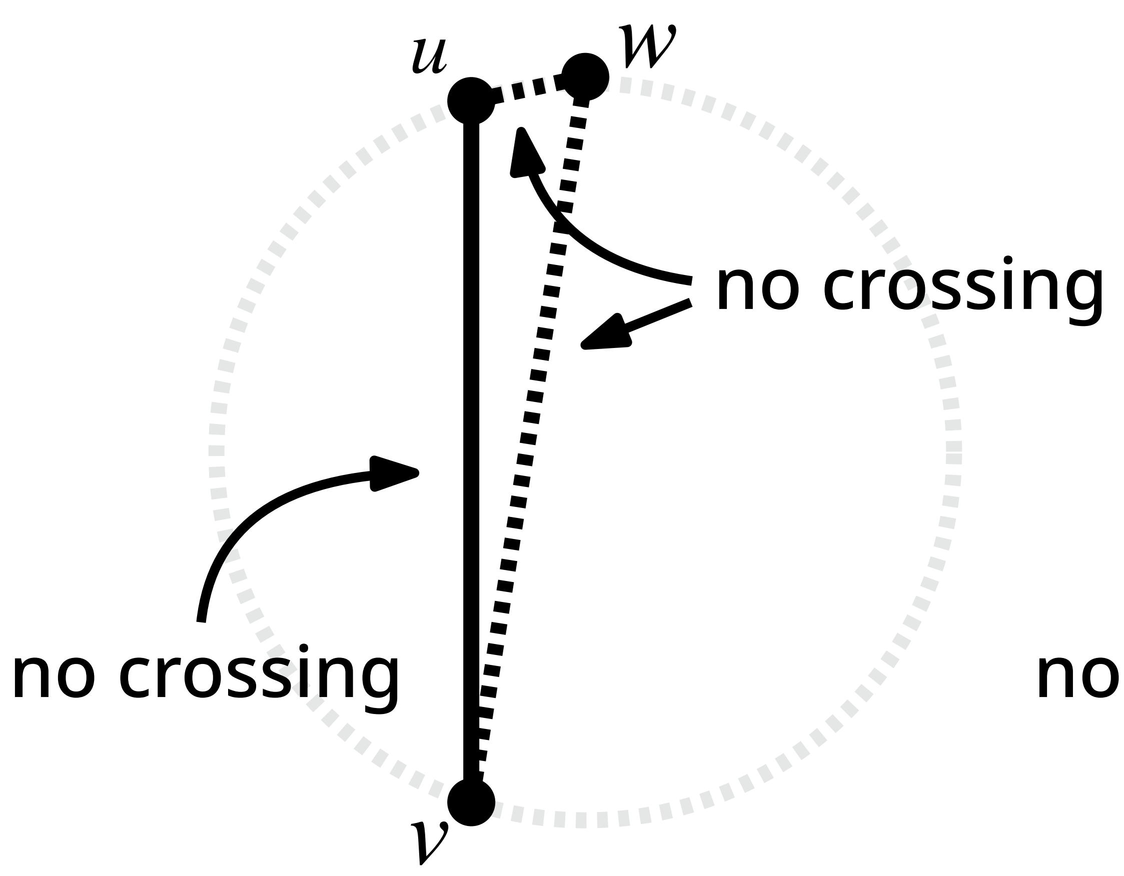
(ii) u has some neighbors

Correctness of the Greedy Algorithm

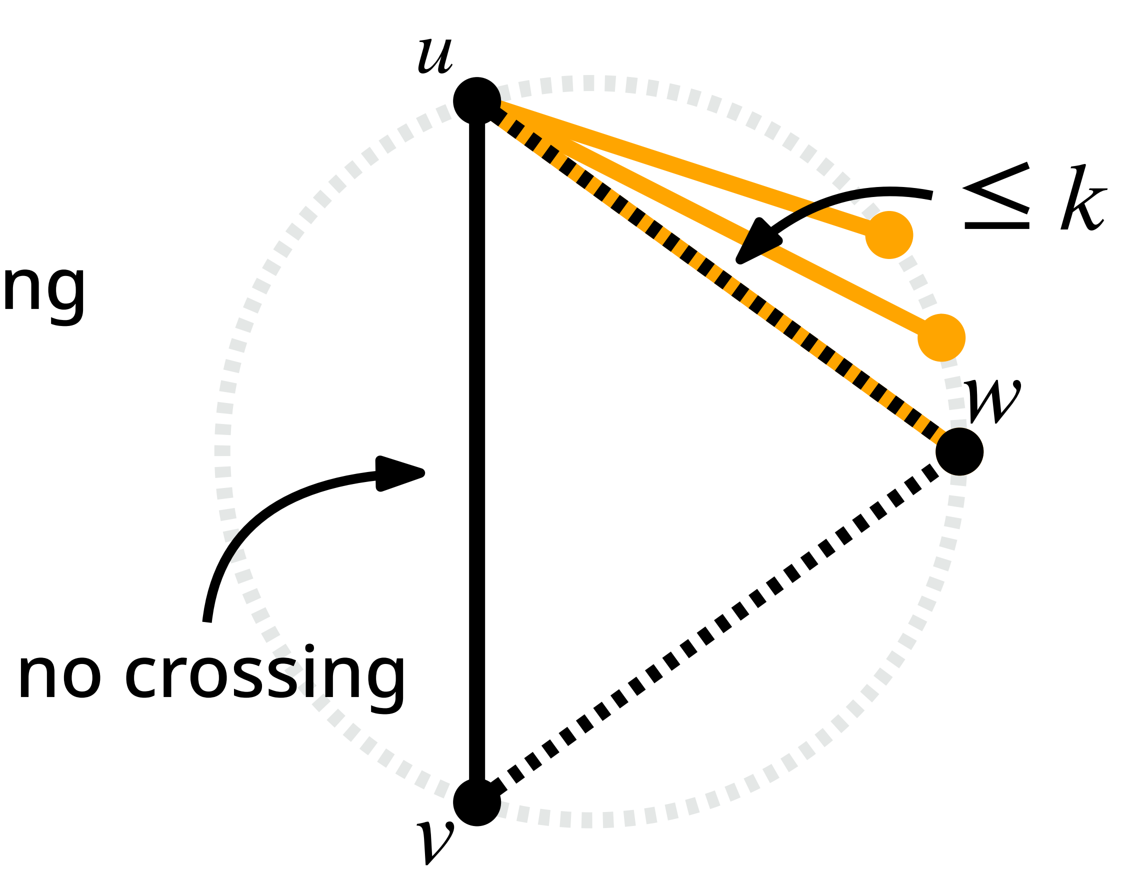
(in Lemma 6)

If $\{u, v\}$ crosses the original graph at most k times, there always exists a suitable vertex w.

Proved by case analysis. If $\{u, v\}$ has no crossing,



(i) u has no neighbor



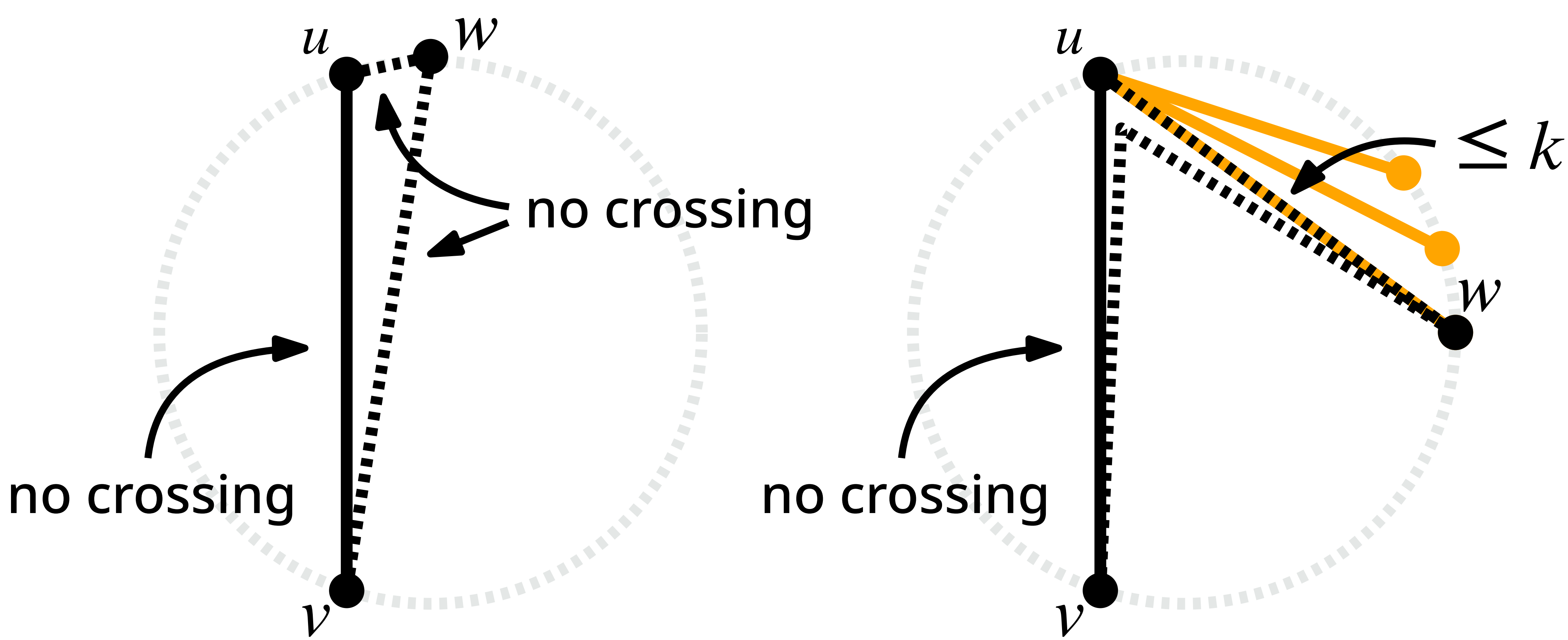
(ii) u has some neighbors

Correctness of the Greedy Algorithm

(in Lemma 6)

If $\{u, v\}$ crosses the original graph at most k times, there always exists a suitable vertex w.

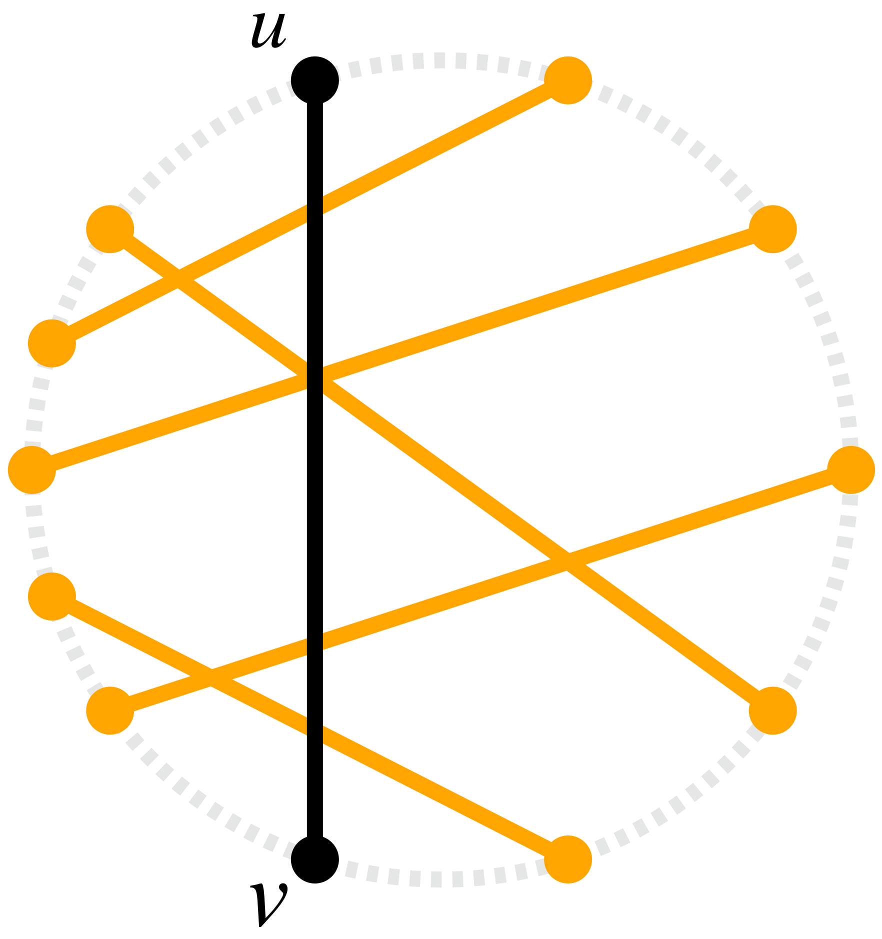
Proved by case analysis. If $\{u, v\}$ has no crossing,



(i) u has no neighbor (ii) u has some neighbors

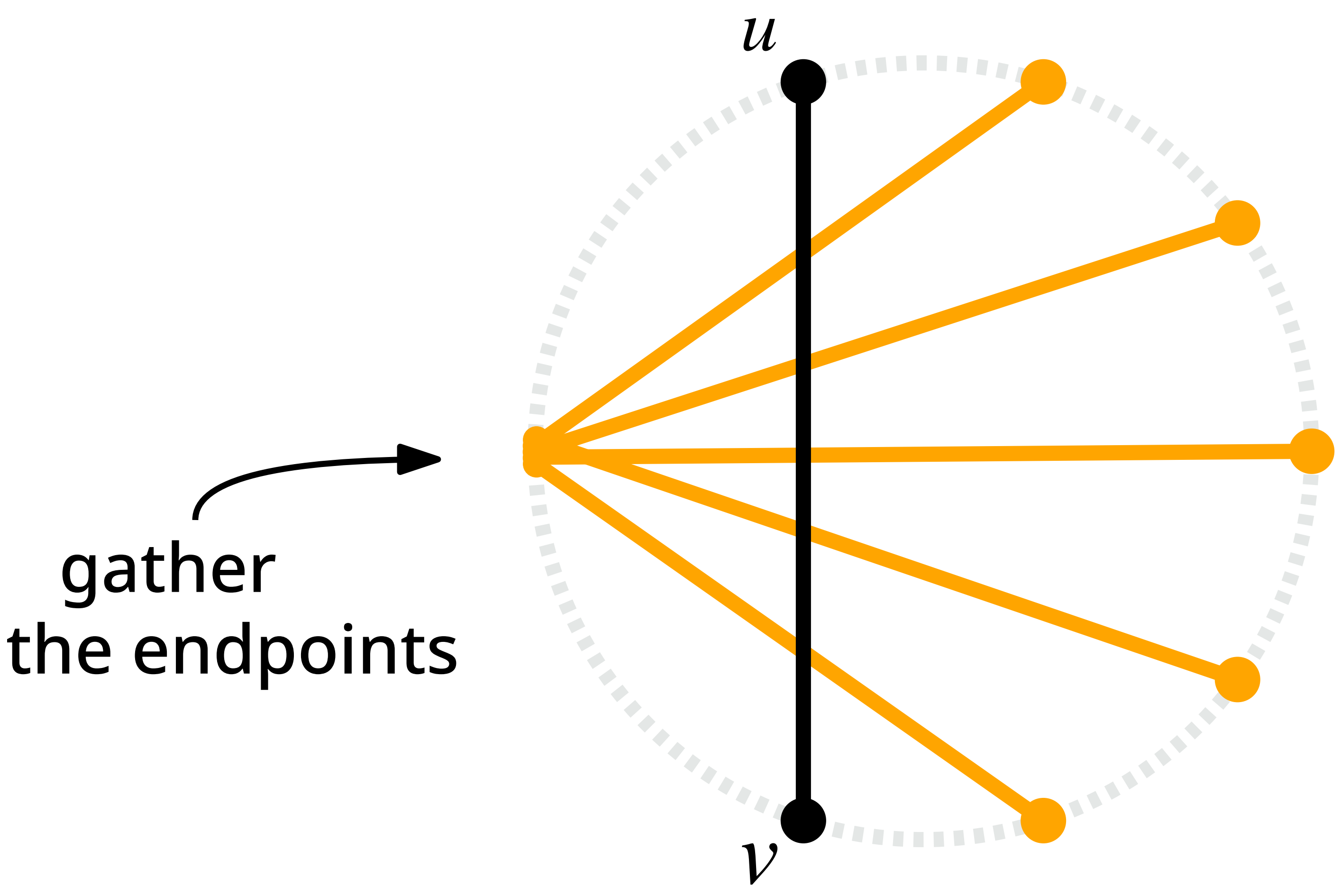
Correctness of the Greedy Algorithm

If $\{u, v\}$ has some crossings,



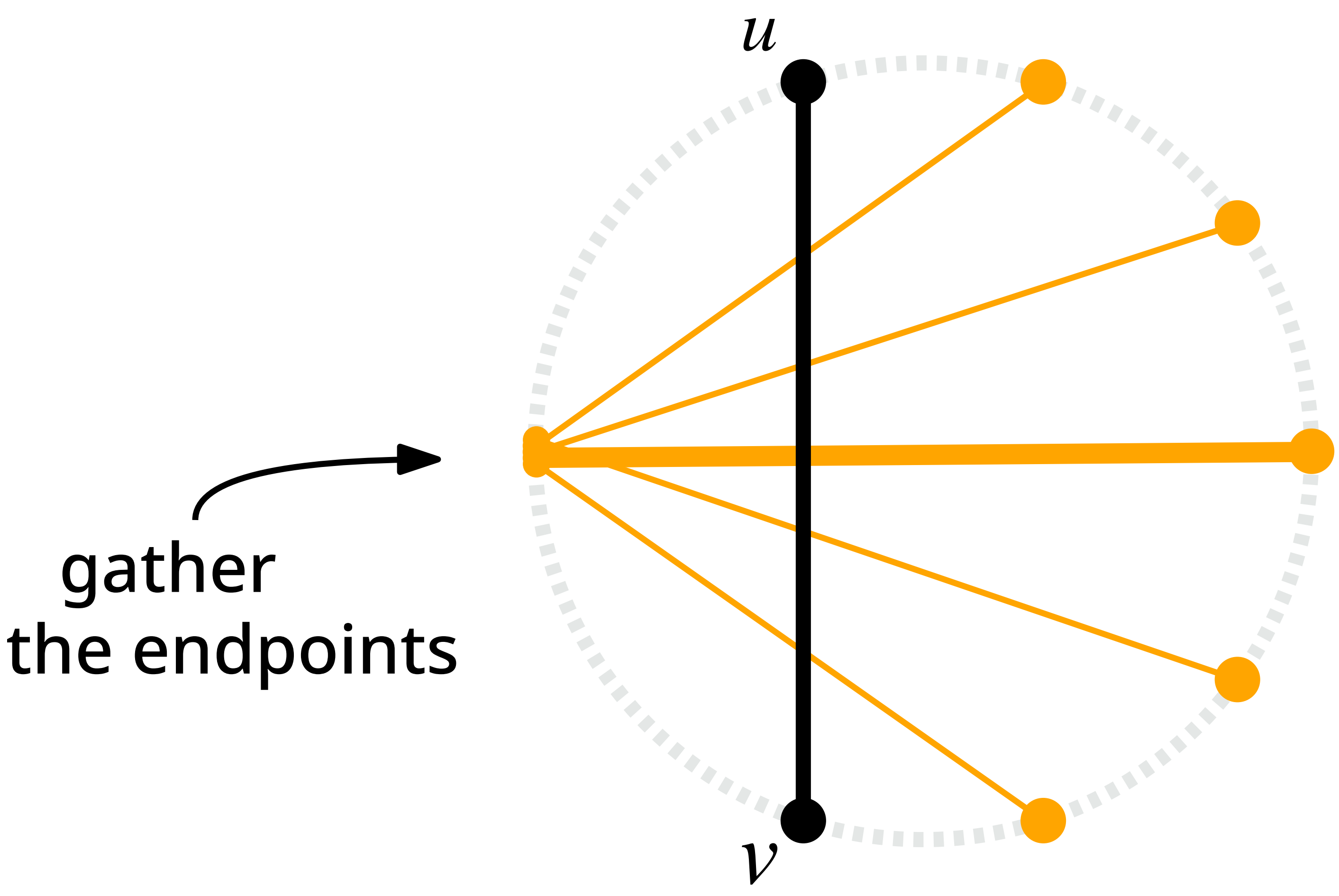
Correctness of the Greedy Algorithm

If $\{u, v\}$ has some crossings,



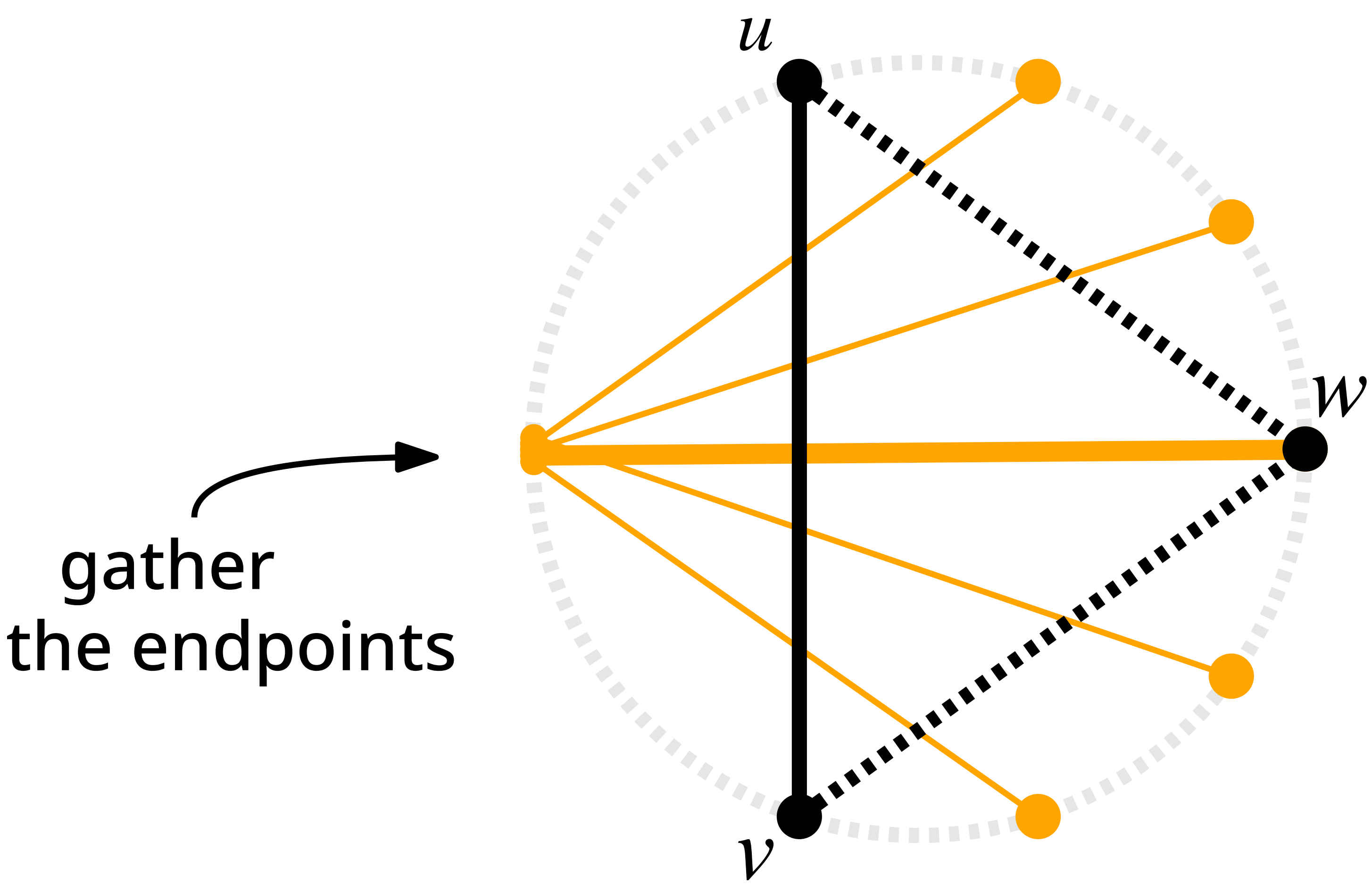
Correctness of the Greedy Algorithm

If $\{u, v\}$ has some crossings,



Correctness of the Greedy Algorithm

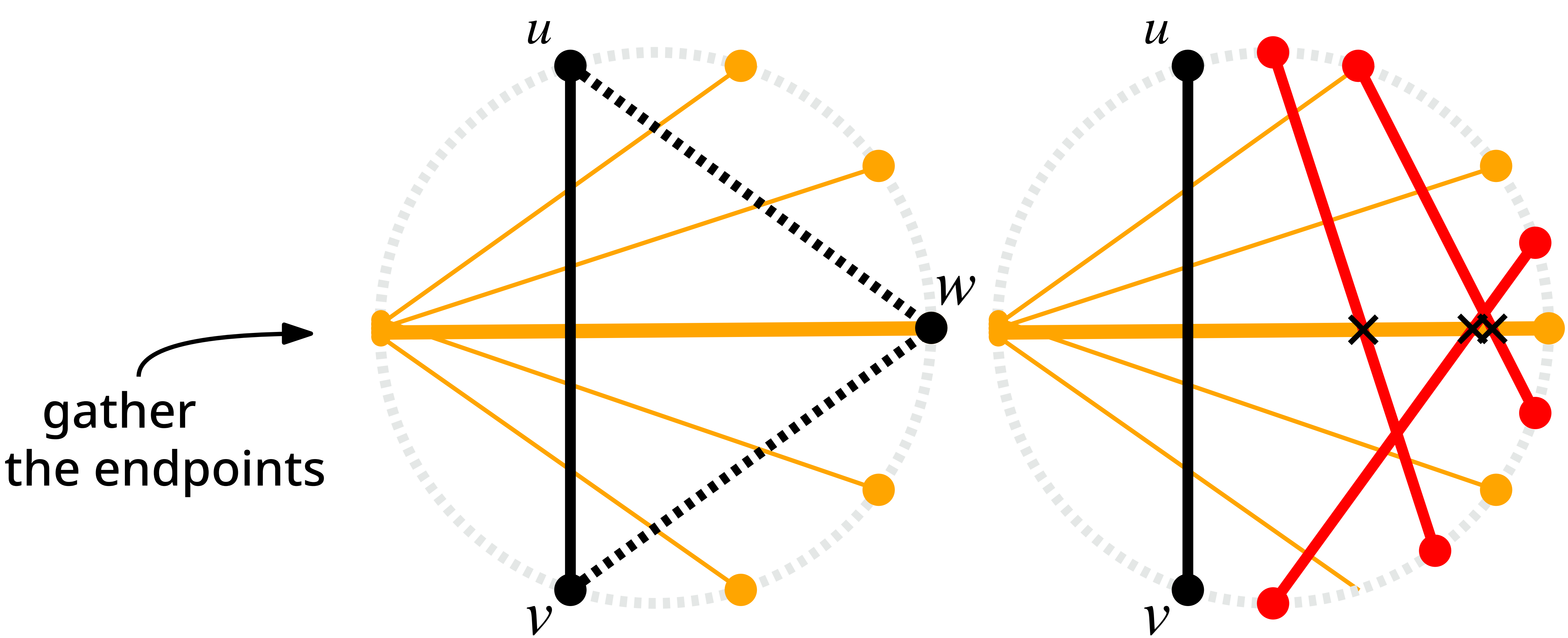
If $\{u, v\}$ has some crossings,



The middle edge has (iii) no crossing

Correctness of the Greedy Algorithm

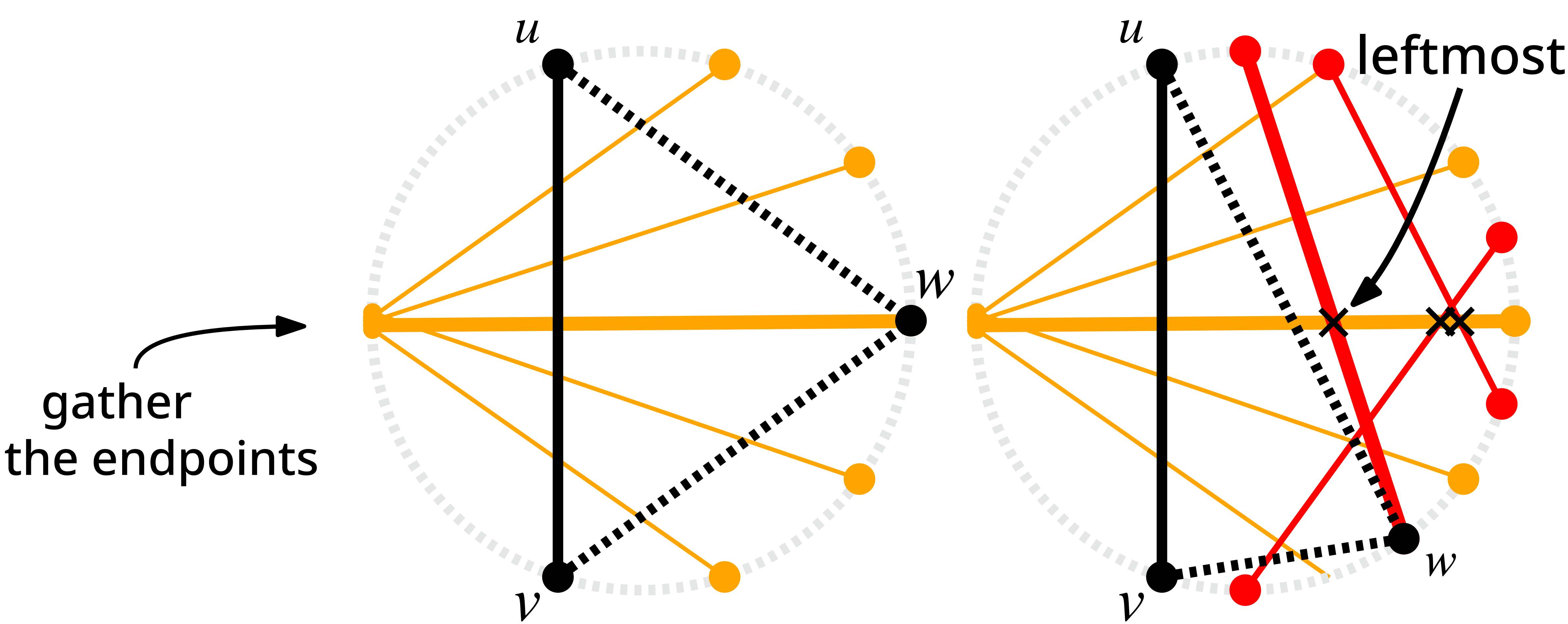
If $\{u, v\}$ has some crossings,



The middle edge has (iii) no crossing (iv) some crossings

Correctness of the Greedy Algorithm

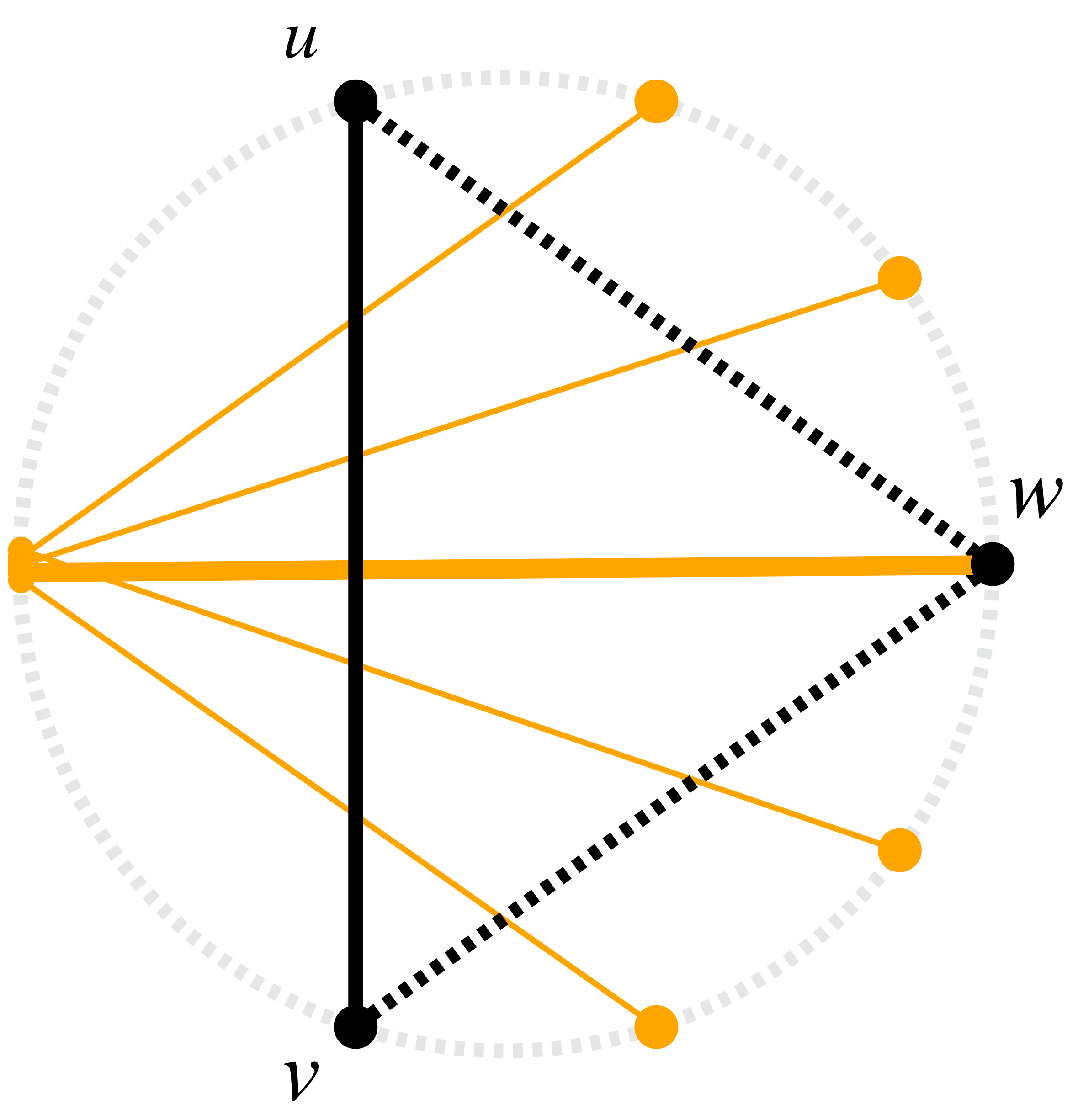
If $\{u, v\}$ has some crossings,



The middle edge has (iii) no crossing (iv) some crossings

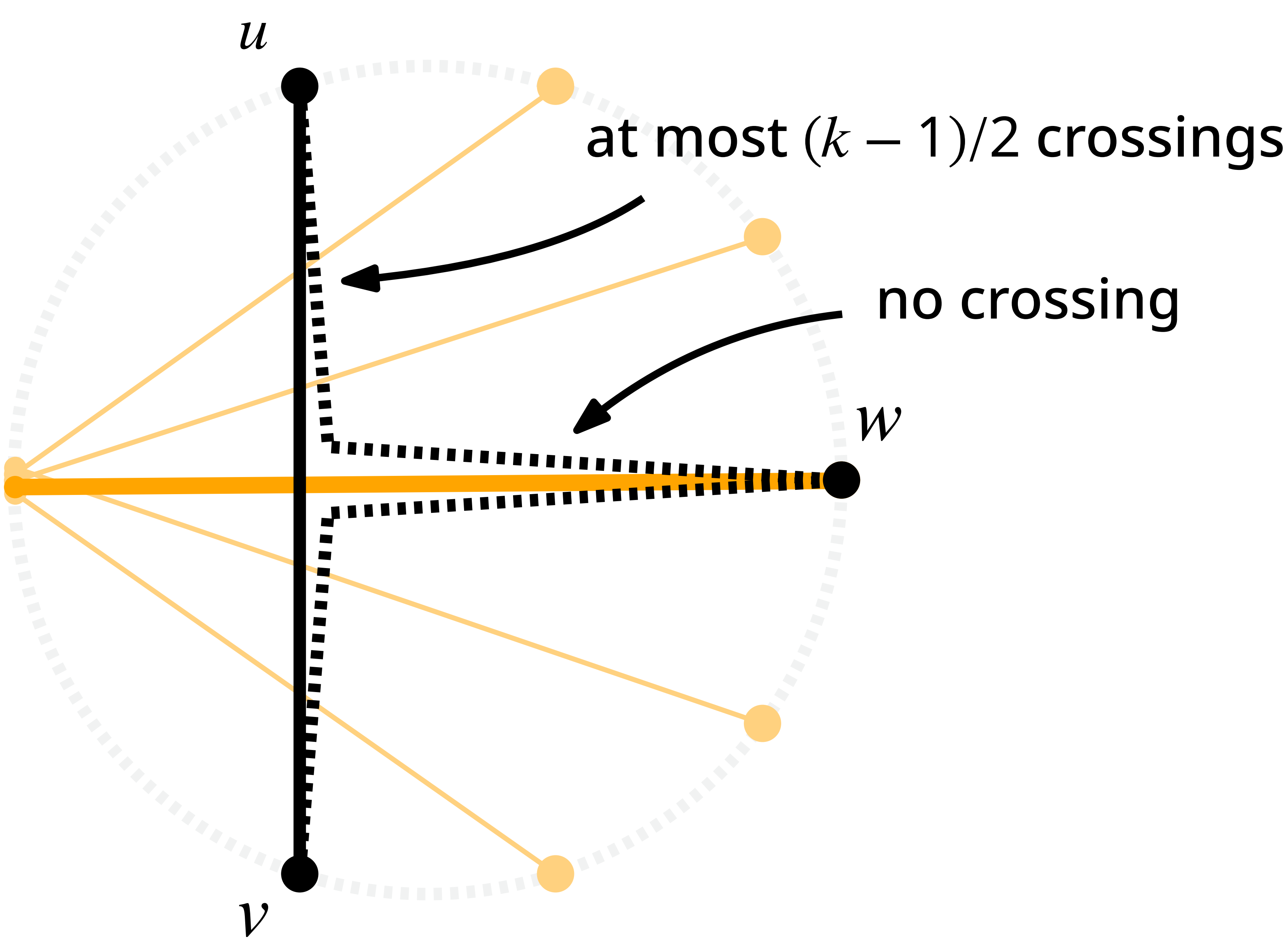
Correctness of the Greedy Algorithm

(iii) The middle edge has no crossing.



Correctness of the Greedy Algorithm

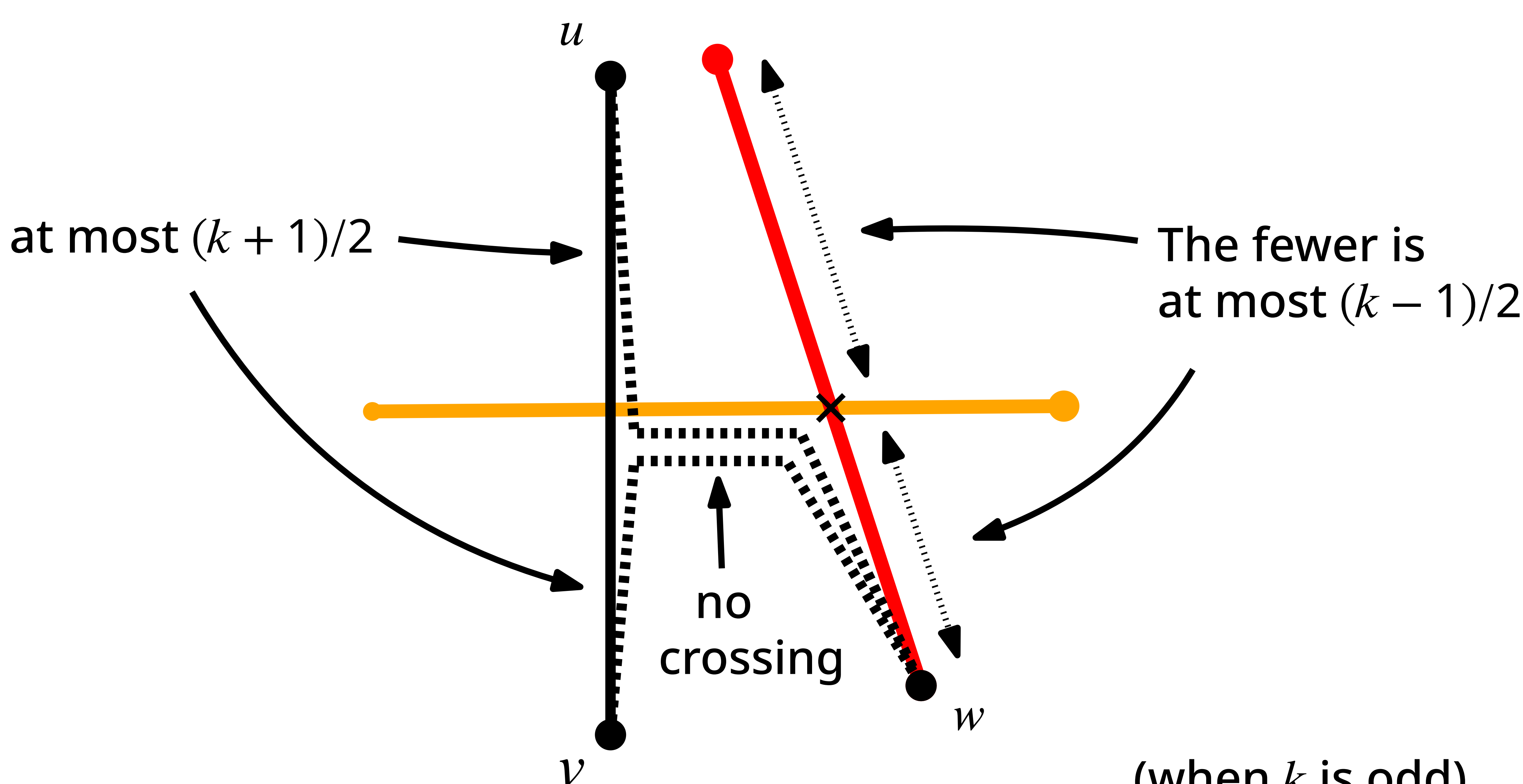
(iii) The middle edge has no crossing.



Correctness of the Greedy Algorithm

(iv) The middle edge has some crossings.

Choose the endpoint of fewer side as w.



(when k is odd)

Summary & Other Results

Main Results

- Improved upper bounds via a good triangulation.

	Upper	Lower
k	$k+2$	$k+2$

separation number
(Orange results are ours.)

	Upper	Lower
0	2	2 (K_3)
1	3	3 (K_4)
2	4	4 (K_5)
k	$1.5k+2$	$k+2$

treewidth

Other Results

- Lower bounds with stacked prisms for even k.
- Treewidth 4 for $k=2$ by a specialized triangulation.
- Similar results on outer min-k-planar graphs.

Future Work & Open Problems

Other application of the triangulation?

Fill the gap between bounds of the treewidth.

Poly-time recognition of outer k-planar graphs.

Linear-time algorithm known for $k = 1$ [Auer et al., '16].

For fixed k, $O(2^{\text{poly}(\log(n))})$-time [Chaplick et al., '17].

Triangulation of k-planar graphs?

Our triangulation can be stated as this:

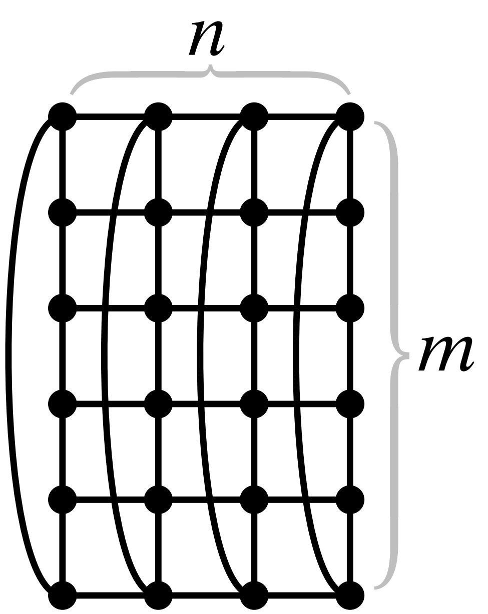
> On every outer k-planar drawing, we can draw
> a maximal outerplane graph with the same vertices
> s.t. each edge crosses the drawing at most k times.

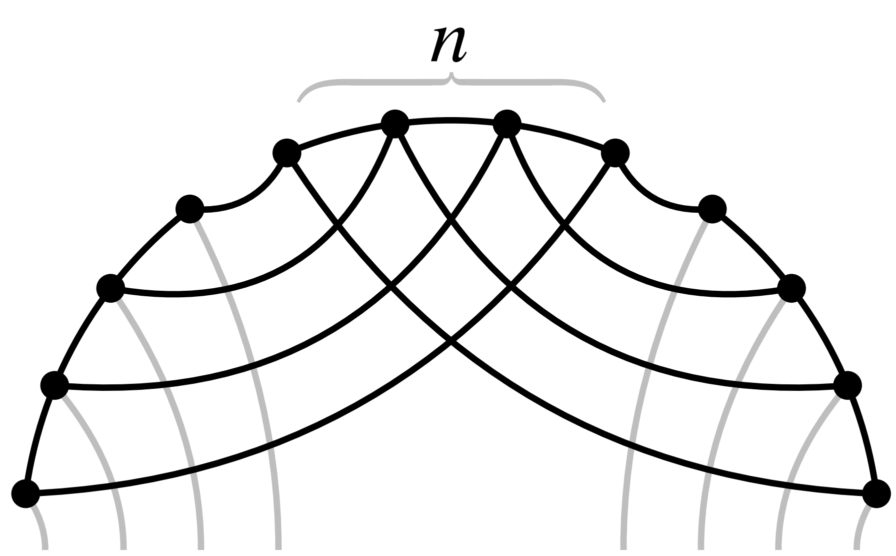
Does this also hold for k-planar graphs?

Appendix: Lower Bounds

Stacked Prisms

are obtained by connecting the top & bottom of grids.





The $m \times n$ grid satisfies: (for large enough m)

- Outer $(2n-2)$-planar graph
- Treewidth $2n$
- Separation number $2n$

$\rightarrow$ lower bounds $k+2$ on both for every even $k > 0$.

Appendix: Treewidth 4 for $k = 2$

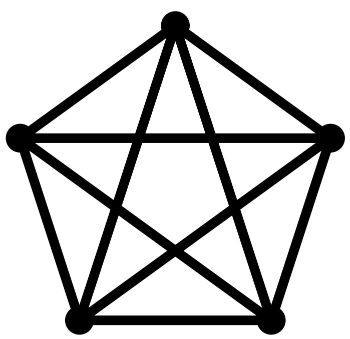
Lemma 8

Every outer 2-planar graph admits a triangulation s.t. each triangle has at most 4 crossing edges.

$+$

Lemma 11

If the outer cycle of G admits a triangulation s.t. each triangle has at most c crossings, then tw$(G) \leq (c + 5)/2$.



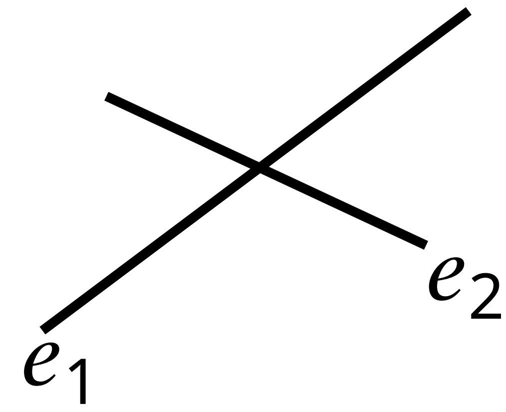
Theorem 12

Every outer 2-planar graphs have treewidth at most 4, which is tight because of K_5.

Appendix: Results on Outer Min-k-Planar

Min-k-Planar

If edges e_1 and e_2 cross, then either e_1 or e_2 have at most k crossings.



Lemma 9

Every outer min-k-planar graph admits a triangulation s.t. each edge has at most $2k - 1$ crossing edges.

↓

Theorem 14 (Treewidth)

Every outer min-k-planar graph has tw at most $3k + 1$.

Theorem 16 (Separation Number)

Every outer min-k-planar graph has sn at most $2k + 1$.