Twin-width of Planar Graphs is at most ~~11~~ 9

Petr Hliněný

Faculty of Informatics, Masaryk University
Brno, Czech Republic

1 Twin-Width

- A structural measure of how (recursively) diverse are vertex neighbourhoods in a graph – but it is not the neighbourhood diversity.

1 Twin-Width

- A structural measure of how (recursively) diverse are vertex neighbourhoods in a graph – but it is not the neighbourhood diversity.

  Or, how similar the graph is to a cograph.

1 Twin-Width

- A structural measure of how (recursively) diverse are vertex neighbourhoods in a graph – but it is not the neighbourhood diversity.

  Or, how similar the graph is to a cograph.

Nowadays, everybody is speaking about twin-width...

1 Twin-Width

- A structural measure of how (recursively) diverse are vertex neighbourhoods in a graph – but it is not the neighbourhood diversity.

  Or, how similar the graph is to a cograph.

  Nowadays, everybody is speaking about twin-width...

- **Trigraph** – a simple graph with some edges marked red

  (we want the maximum red degree to stay low).

1 Twin-Width

- A structural measure of how (recursively) diverse are vertex neighbourhoods in a graph – but it is not the neighbourhood diversity.

  Or, how similar the graph is to a cograph.

  Nowadays, everybody is speaking about twin-width...

- **Trigraph** – a simple graph with some edges marked red

  (we want the maximum red degree to stay low).

- **Contraction sequence** – a sequence of simple contractions of vertex pairs (arbitrary pairs, unlike in graph minors!);

1 Twin-Width

- A structural measure of how (recursively) diverse are vertex neighbourhoods in a graph – but it is not the neighbourhood diversity.

  Or, how similar the graph is to a cograph.

Nowadays, everybody is speaking about twin-width...

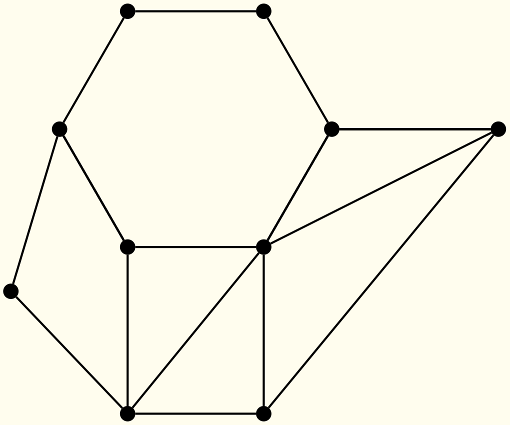
- **Trigraph** – a simple graph with some edges marked red

  (we want the maximum red degree to stay low).

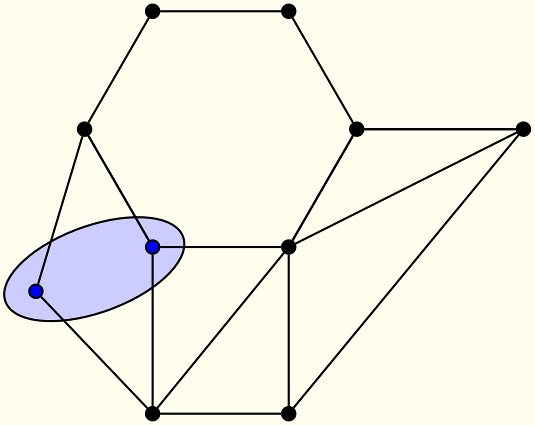
- **Contraction sequence** – a sequence of simple contractions of vertex pairs (arbitrary pairs, unlike in graph minors!);
  - a contraction of a pair makes an edge red if it existed to one of the contracted vertices but not to the other, and
  - red edge stays red till the end.

Definition. The **twin-width** of a simple graph G is the least int. d

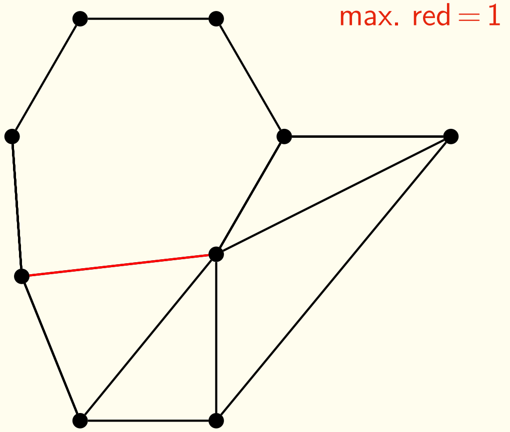
Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



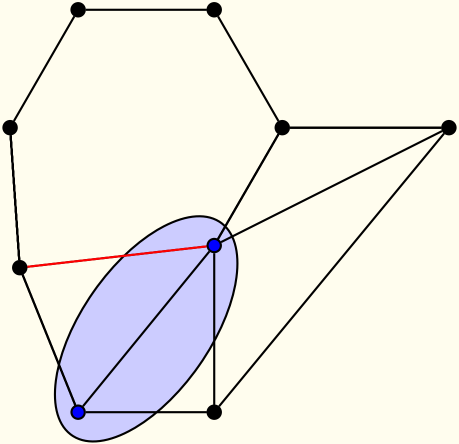
Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



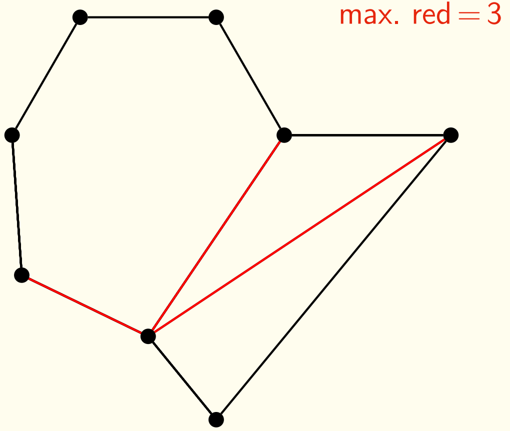
Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



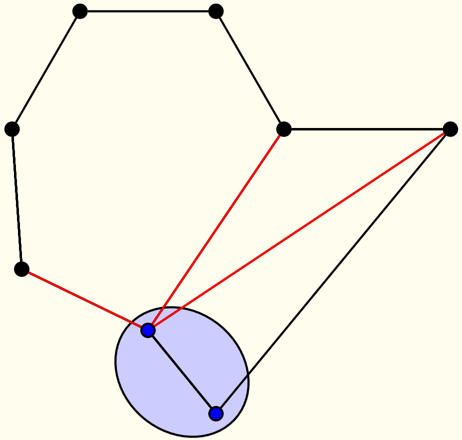
Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



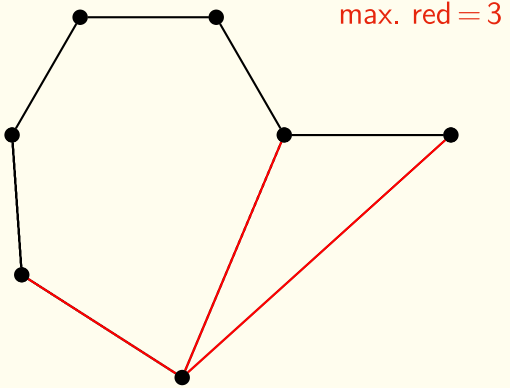
Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



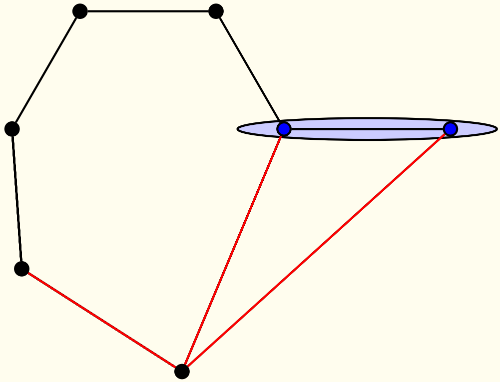
Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



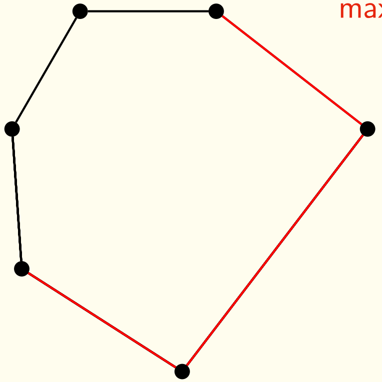
Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

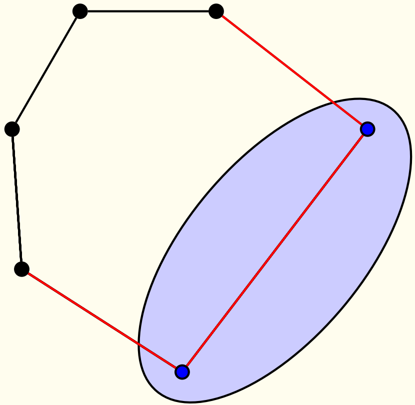


Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

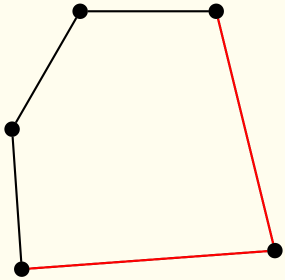


max. red $= 2$

Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

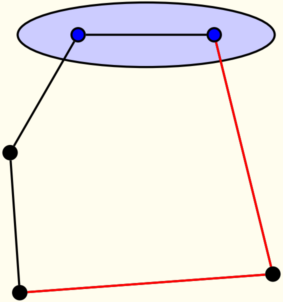


Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

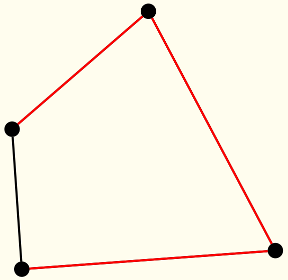


max. red $= 2$

Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

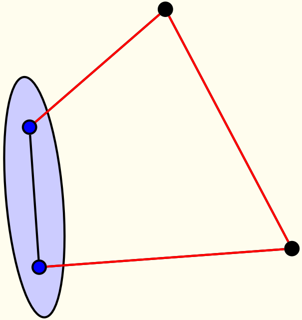


Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

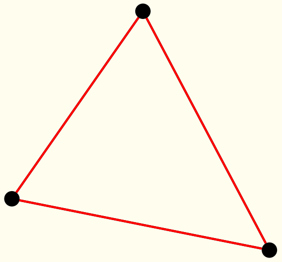


max. red $= 2$

Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

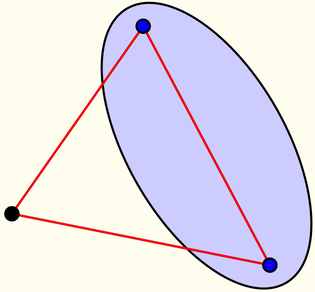


Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

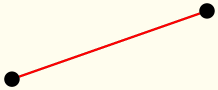


max. red $= 2$

Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

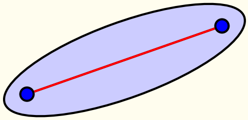


Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



max. red $= 1$

Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.



Definition. The **twin-width** of a simple graph G is the least int. d such that there exists a contraction sequence of G in which every trigraph has maximum red degree $\leq d$.

max. red $= 0$

twin-width ≤ 3

Importance of Twin-Width

- The concept introduced by **Bonnet, Kim, Thomassé and Watrigant** in 2020 $\leadsto$ [FOCS 2020, JACM 2022], and

Importance of Twin-Width

- The concept introduced by **Bonnet, Kim, Thomassé and Watrigant** in 2020 $\rightsquigarrow$ [FOCS 2020, JACM 2022], and

  nowadays we already have *tens of papers on twin-width* and the number is quickly growing.

Importance of Twin-Width

- The concept introduced by **Bonnet, Kim, Thomassé and Watrigant** in 2020 $\rightsquigarrow$ [FOCS 2020, JACM 2022], and

  nowadays we already have *tens of papers on twin-width* and the number is quickly growing.

- Among the key properties, graph classes of bounded twin-width have FO model cheking in FPT [FOCS 2020], and

Importance of Twin-Width

- The concept introduced by **Bonnet, Kim, Thomassé and Watrigant** in 2020 $\rightsquigarrow$ [FOCS 2020, JACM 2022], and

  nowadays we already have *tens of papers on twin-width* and the number is quickly growing.

- Among the key properties, graph classes of bounded twin-width have FO model cheking in FPT [FOCS 2020], and

  this new concept seems to be crucial in the ongoing quest to characterise hereditary classes with tractable FO model checking (cf. the subsequent talks. . .).

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,
- also grids (incl. multidimensional), full grids, and their subgraphs,

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,
- also grids (incl. multidimensional), full grids, and their subgraphs,
- proper interval graphs and their generalizations,
- proper hereditary subclasses of permutation or circle graphs,

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,
- also grids (incl. multidimensional), full grids, and their subgraphs,
- proper interval graphs and their generalizations,
- proper hereditary subclasses of permutation or circle graphs,
- posets of bounded width,

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,
- also grids (incl. multidimensional), full grids, and their subgraphs,
- proper interval graphs and their generalizations,
- proper hereditary subclasses of permutation or circle graphs,
- posets of bounded width,
- planar graphs and graphs embedded on surfaces,

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,
- also grids (incl. multidimensional), full grids, and their subgraphs,
- proper interval graphs and their generalizations,
- proper hereditary subclasses of permutation or circle graphs,
- posets of bounded width,
- planar graphs and graphs embedded on surfaces,
- graphs drawn with limited number of crossings per edge, map graphs.

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,
- also grids (incl. multidimensional), full grids, and their subgraphs,
- proper interval graphs and their generalizations,
- proper hereditary subclasses of permutation or circle graphs,
- posets of bounded width,
- planar graphs and graphs embedded on surfaces,
- graphs drawn with limited number of crossings per edge, map graphs.

NOT bounded twin-width

- Interval and permutation graphs in general,

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,
- also grids (incl. multidimensional), full grids, and their subgraphs,
- proper interval graphs and their generalizations,
- proper hereditary subclasses of permutation or circle graphs,
- posets of bounded width,
- planar graphs and graphs embedded on surfaces,
- graphs drawn with limited number of crossings per edge, map graphs.

NOT bounded twin-width

- Interval and permutation graphs in general,
- small subdivisions of cliques,

2 Classes of Bounded Twin-Width

Only few examples:

- Graphs of bounded tree-width or rank-width, cliques,
- also grids (incl. multidimensional), full grids, and their subgraphs,
- proper interval graphs and their generalizations,
- proper hereditary subclasses of permutation or circle graphs,
- posets of bounded width,
- planar graphs and graphs embedded on surfaces,
- graphs drawn with limited number of crossings per edge, map graphs.

NOT bounded twin-width

- Interval and permutation graphs in general,
- small subdivisions of cliques,
- cubic graphs (!!!).

Bounding Twin-Width of Planar Graphs

- Astronomical upper bounds already since the first pap. [FOCS 2020].

Bounding Twin-Width of Planar Graphs

- Astronomical upper bounds already since the first pap. [FOCS 2020].
- ArXiv Jan 2022: $\leq \mathbf{183}$ by Jacob and Pilipczuk.

Bounding Twin-Width of Planar Graphs

- Astronomical upper bounds already since the first pap. [FOCS 2020].
- ArXiv Jan 2022: $\leq \mathbf{183}$ by Jacob and Pilipczuk.
- ArXiv Feb 2022: $\leq \mathbf{583}$ by Bonnet, Kwon and Wood.

Bounding Twin-Width of Planar Graphs

- Astronomical upper bounds already since the first pap. [FOCS 2020].
- ArXiv Jan 2022: $\leq \mathbf{183}$ by Jacob and Pilipczuk.
- ArXiv Feb 2022: $\leq \mathbf{583}$ by Bonnet, Kwon and Wood.
- ArXiv Apr 2022: $\leq \mathbf{37}$ by Bekos, Da Lozzo, PH and Kaufman.

All previous concrete bounds use in some (indirect) way the *product structure* machinery of planar graphs.

Bounding Twin-Width of Planar Graphs

- Astronomical upper bounds already since the first pap. [FOCS 2020].
- ArXiv Jan 2022: $\leq \mathbf{183}$ by Jacob and Pilipczuk.
- ArXiv Feb 2022: $\leq \mathbf{583}$ by Bonnet, Kwon and Wood.
- ArXiv Apr 2022: $\leq \mathbf{37}$ by Bekos, Da Lozzo, PH and Kaufman.

All previous concrete bounds use in some (indirect) way the *product structure* machinery of planar graphs.

With a NEW approach:

- **The twin-width of any simple planar graph is at most 9.**

Bounding Twin-Width of Planar Graphs

- Astronomical upper bounds already since the first pap. [FOCS 2020].
- ArXiv Jan 2022: $\leq \mathbf{183}$ by Jacob and Pilipczuk.
- ArXiv Feb 2022: $\leq \mathbf{583}$ by Bonnet, Kwon and Wood.
- ArXiv Apr 2022: $\leq \mathbf{37}$ by Bekos, Da Lozzo, PH and Kaufman.

All previous concrete bounds use in some (indirect) way the *product structure* machinery of planar graphs.

With a NEW approach:

- **The twin-width of any simple planar graph is at most 9.**

Lower bounds?

Bounding Twin-Width of Planar Graphs

- Astronomical upper bounds already since the first pap. [FOCS 2020].
- ArXiv Jan 2022: $\leq \mathbf{183}$ by Jacob and Pilipczuk.
- ArXiv Feb 2022: $\leq \mathbf{583}$ by Bonnet, Kwon and Wood.
- ArXiv Apr 2022: $\leq \mathbf{37}$ by Bekos, Da Lozzo, PH and Kaufman.

All previous concrete bounds use in some (indirect) way the *product structure* machinery of planar graphs.

With a NEW approach:

- **The twin-width of any simple planar graph is at most 9.**

Lower bounds?
≥ 5 quite easily, but no better lower bound published so far...

3 Simplified Proof of Twin-Width $\leq$12

Preliminaries

- *BFS tree* – a spanning tree of shortest paths from the given root.

  Vertical path – a subpath of a leaf-to-root path of the BFS tree.

3 Simplified Proof of Twin-Width ≤ 12

Preliminaries

- *BFS tree* – a spanning tree of shortest paths from the given root.

  Vertical path – a subpath of a leaf-to-root path of the BFS tree.

  Sink of a vertical path – the end vertex closest to the root.

3 Simplified Proof of Twin-Width $\leq$12

Preliminaries

- *BFS tree* – a spanning tree of shortest paths from the given root.

  Vertical path – a subpath of a leaf-to-root path of the BFS tree.

  Sink of a vertical path – the end vertex closest to the root.

Setup of the proof

- Given a simple planar graph G, extend G into a plane triangulation $G^+ \supseteq G$ (but keep in mind original G regarding the twin-width).

3 Simplified Proof of Twin-Width $\leq$12

Preliminaries

- *BFS tree* – a spanning tree of shortest paths from the given root.

  Vertical path – a subpath of a leaf-to-root path of the BFS tree.

  Sink of a vertical path – the end vertex closest to the root.

Setup of the proof

- Given a simple planar graph G, extend G into a plane triangulation $G^+ \supseteq G$ (but keep in mind original G regarding the twin-width).
- Choose a root on the outer f., and a BFS tree of G^+ from this root.
  Note that all edges are only between same and successive BFS layers.

3 Simplified Proof of Twin-Width $\leq$12

Preliminaries

- *BFS tree* – a spanning tree of shortest paths from the given root.

  Vertical path – a subpath of a leaf-to-root path of the BFS tree.

  Sink of a vertical path – the end vertex closest to the root.

Setup of the proof

- Given a simple planar graph G, extend G into a plane triangulation $G^+ \supseteq G$ (but keep in mind original G regarding the twin-width).
- Choose a root on the outer f., and a BFS tree of G^+ from this root. *Note that all edges are only between same and successive BFS layers.*
- Formulate a suitable (recursive) claim about partial contractions inside a bounded region of the plane triangulation. *Prove by induction.*

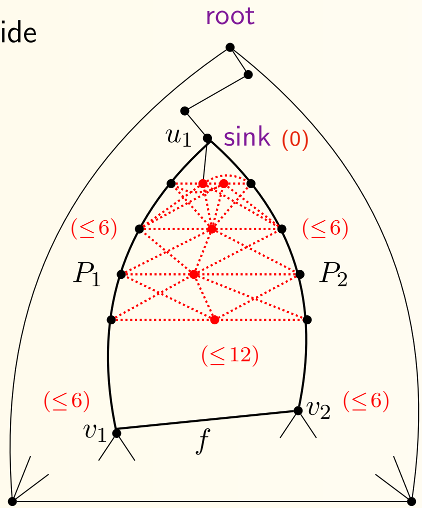
The Recursive Claim

Lemma. Given a subgraph of our G^+ bounded by a cycle which is formed by two vertical paths P_1, P_2 with a common sink and an edge f joining their far ends,

The Recursive Claim

Lemma. Given a subgraph of our G^+ bounded by a cycle which is formed by two vertical paths P_1, P_2 with a common sink and an edge f joining their far ends, there is a partial contraction sequence such that:

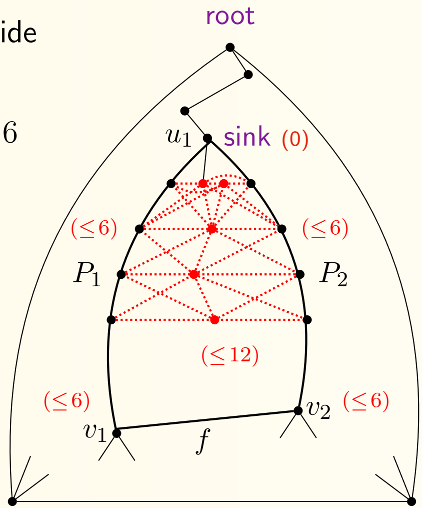
- only vert. of the same BFS layer inside are ever contracted in this Lemma,



The Recursive Claim

Lemma. Given a subgraph of our G^+ bounded by a cycle which is formed by two vertical paths P_1, P_2 with a common sink and an edge f joining their far ends, there is a partial contraction sequence such that:

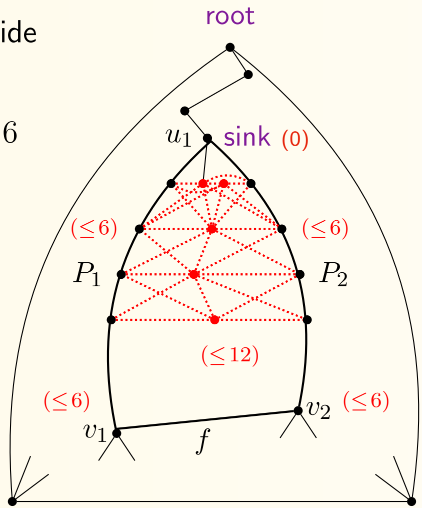
- only vert. of the same BFS layer inside are ever contracted in this Lemma,
- on the boundary, red degrees are ≤ 6 during the whole subsequence,



The Recursive Claim

Lemma. Given a subgraph of our G^+ bounded by a cycle which is formed by two vertical paths P_1, P_2 with a common sink and an edge f joining their far ends, there is a partial contraction sequence such that:

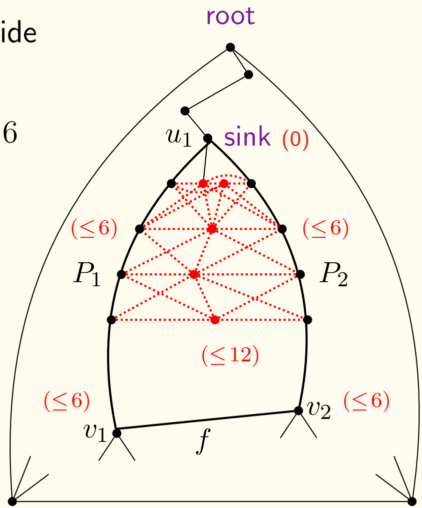
- only vert. of the same BFS layer inside are ever contracted in this Lemma,
- on the boundary, red degrees are ≤ 6 during the whole subsequence,
- the sink has red degree **0**,



The Recursive Claim

Lemma. Given a subgraph of our G^+ bounded by a cycle which is formed by two vertical paths P_1, P_2 with a common sink and an edge f joining their far ends, there is a partial contraction sequence such that:

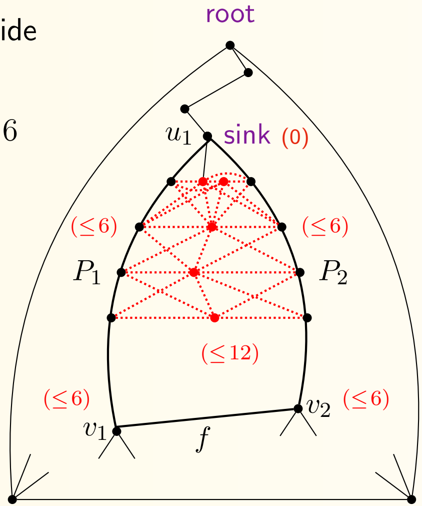
- only vert. of the same BFS layer inside are ever contracted in this Lemma,
- on the boundary, red degrees are ≤ 6 during the whole subsequence,
- the sink has red degree **0**,
- the red degrees inside are ≤ 12 during the whole subsequence,



The Recursive Claim

Lemma. Given a subgraph of our G^+ bounded by a cycle which is formed by two vertical paths P_1, P_2 with a common sink and an edge f joining their far ends, there is a partial contraction sequence such that:

- only vert. of the same BFS layer inside are ever contracted in this Lemma,
- on the boundary, red degrees are ≤ 6 during the whole subsequence,
- the sink has red degree **0**,
- the red degrees inside are ≤ 12 during the whole subsequence,
- after the contractions, each BFS layer inside has only 1 vertex, except ≤ 2 vert. next to the sink.

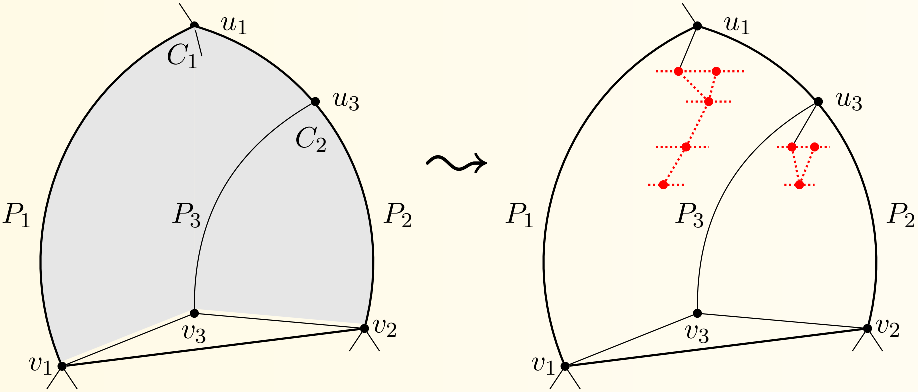


The Proof (by induction)

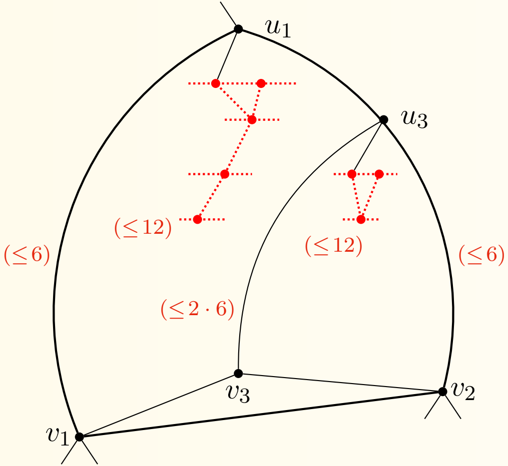
- Take the triangle incident to the “far edge” $f = v_1v_2$, and the vertical path P_3 from its tip v_3 (to the boundary at u_3).

The Proof (by induction)

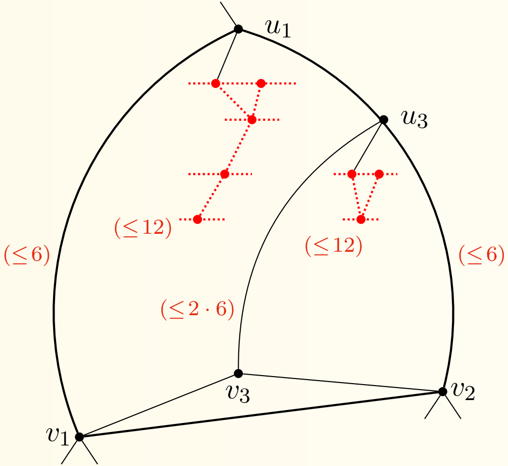
- Take the triangle incident to the "far edge" $f = v_1v_2$, and the vertical path P_3 from its tip v_3 (to the boundary at u_3).
- Apply the Lemma inductively to each of the two subregions:



- The partial contraction sequences from the inductive invocations can be put one after another, since there are no edges "across" P_3.

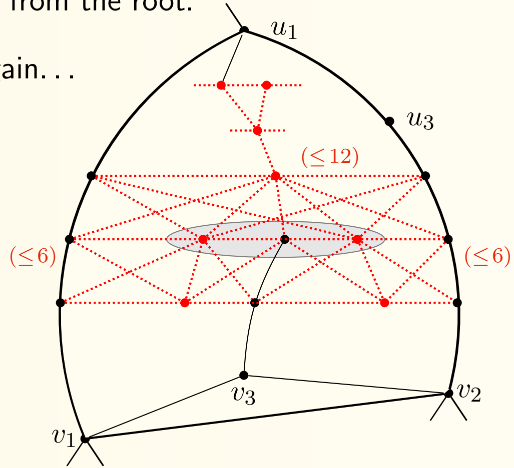


- The partial contraction sequences from the inductive invocations can be put one after another, since there are no edges "across" P_3.
- Then contract, by the BFS layers inside, the recursively contracted vertices with those of vertical "divisor" P_3 down to one or two vert.



- The partial contraction sequences from the inductive invocations can be put one after another, since there are no edges "across" P_3.
- Then contract, by the BFS layers inside, the recursively contracted vertices with those of vertical "divisor" P_3 down to one or two vert. Proceed in increasing distance from the root.
- And, check the red degrees again. . .

□



4 Towards Proving Twin-Width $\leq$9

Several adjustments at different places are necessary (which make the proof quite technical)...

4 Towards Proving Twin-Width $\leq$9

Several adjustments at different places are necessary (which make the proof quite technical)...

- The key is to consider a *left-aligned BFS tree* instead of general one. This makes the left and right bounding paths non-symmetric, and their claimed recursive red degrees are ≤ 5 and ≤ 3, respectively.

4 Towards Proving Twin-Width ≤9

Several adjustments at different places are necessary (which make the proof quite technical)...

- The key is to consider a *left-aligned BFS tree* instead of general one.

  This makes the left and right bounding paths non-symmetric, and their claimed recursive red degrees are ≤ 5 and ≤ 3, respectively.

- We do not contract the partial solutions of the subcases layer-by-layer, but first fully contract the right subcase with the dividing path P_3, and then the outcome with the left subcase.

4 Towards Proving Twin-Width ≤ 9

Several adjustments at different places are necessary (which make the proof quite technical)...

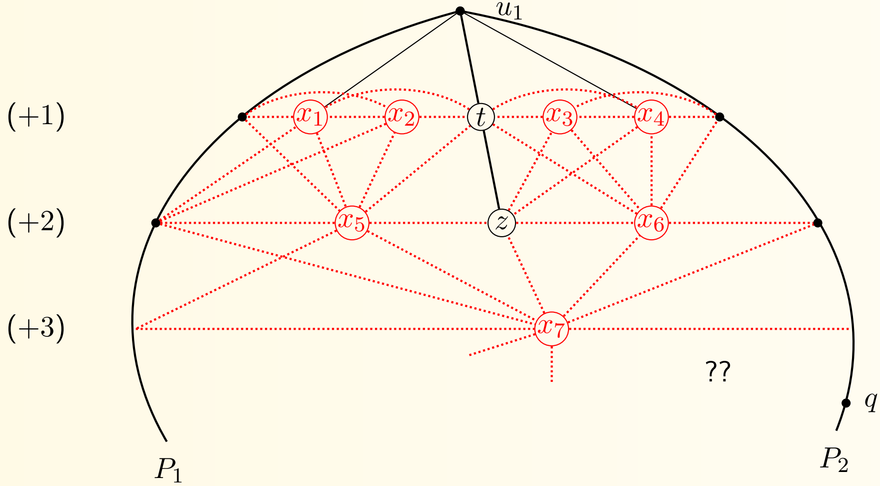
- The key is to consider a *left-aligned BFS tree* instead of general one.

  This makes the left and right bounding paths non-symmetric, and their claimed recursive red degrees are ≤ 5 and ≤ 3, respectively.

- We do not contract the partial solutions of the subcases layer-by-layer, but first fully contract the right subcase with the dividing path P_3, and then the outcome with the left subcase.

- Now we proceed from the farthest BFS layers towards the root, and a few of the layers closest to the sink are possibly handled ad-hoc.

Illustrating the Proof Adjustments...



5 Conclusions

- Our proof technique seems to be at its limit.

  Quite possibly, $\mathbf{9}$ may be the right answer...

5 Conclusions

- Our proof technique seems to be at its limit.
  Quite possibly, $\mathbf{9}$ may be the right answer...

- So, again, what about lower bounds?

5 Conclusions

- Our proof technique seems to be at its limit. Quite possibly, $\mathbf{9}$ may be the right answer…

- So, again, what about lower bounds?

  - Take the dual of the *soccer ball graph*;

5 Conclusions

- Our proof technique seems to be at its limit.
  Quite possibly, $\mathbf{9}$ may be the right answer...

- So, again, what about lower bounds?

  – Take the dual of the *soccer ball graph*;
    $\rightarrow$ already the first contraction must create ≥ 5 red edges.

5 Conclusions

- Our proof technique seems to be at its limit. Quite possibly, $\mathbf{9}$ may be the right answer...
- So, again, what about lower bounds?
  - Take the dual of the *soccer ball graph*;
    $\rightarrow$ already the first contraction must create ≥ 5 red edges.
  - Stepping further, inscribe a degree-3 vertex inside each face of the previous. The result seems to have twin-width ≥ 7, but a careful (computer asisted?) proof is needed.

5 Conclusions

- Our proof technique seems to be at its limit. Quite possibly, $\mathbf{9}$ may be the right answer...
- So, again, what about lower bounds?
  - Take the dual of the *soccer ball graph*;
    $\to$ already the first contraction must create ≥ 5 red edges.
  - Stepping further, inscribe a degree-3 vertex inside each face of the previous. The result seems to have twin-width ≥ 7, but a careful (computer asisted?) proof is needed.
  - Another, more complicated, construction may actually give a planar graph in which the lower-bound proof is easier...

5 Conclusions

- Our proof technique seems to be at its limit. Quite possibly, $\mathbf{9}$ may be the right answer...
- So, again, what about lower bounds?
  - Take the dual of the *soccer ball graph*;
    $\rightarrow$ already the first contraction must create ≥ 5 red edges.
  - Stepping further, inscribe a degree-3 vertex inside each face of the previous. The result seems to have twin-width ≥ 7, but a careful (computer asisted?) proof is needed.
  - Another, more complicated, construction may actually give a planar graph in which the lower-bound proof is easier...

Thank you for your attention.