Hitting minors on bounded treewidth graphs

Julien Baste[1] Ignasi Sau[2] Dimitrios M. Thilikos[2,3]

Dagstuhl 19041
January 2019

[1] Universität Ulm, Ulm, Germany

[2] CNRS, LIRMM, Université de Montpellier, France

[3] Dept. of Maths, National and Kapodistrian University of Athens, Greece

[arXiv 1704.07284]

Treewidth behaves very well algorithmically

Treewidth behaves very well algorithmically

Monadic Second Order Logic (MSOL):
Graph logic that allows quantification over sets of vertices and edges.

Example: `DomSet`(S) : $[\ \forall v \in V(G) \setminus S, \exists u \in S : \{u, v\} \in E(G)\ ]$

Treewidth behaves very well algorithmically

Monadic Second Order Logic (MSOL):
Graph logic that allows quantification over sets of vertices and edges.

Example: `DomSet`(S) : $[\ \forall v \in V(G) \setminus S, \exists u \in S : \{u, v\} \in E(G)\ ]$

Theorem (Courcelle, 1990)

Every problem expressible in MSOL can be solved in time $f(\text{tw}) \cdot n$ on graphs on n vertices and treewidth at most tw.

In parameterized complexity: FPT parameterized by treewidth.

Examples: Vertex Cover, Dominating Set, Hamiltonian Cycle, Clique, Independent Set, k-Coloring for fixed k, ...

Is it enough to prove that a problem is FPT?

Typically, Courcelle's theorem allows to prove that a problem is FPT...

$$f(\text{tw}) \cdot n^{\mathcal{O}(1)}$$

Is it enough to prove that a problem is FPT?

Typically, Courcelle's theorem allows to prove that a problem is FPT...

... but the running time can (and must) be huge!

$$f(\text{tw}) \cdot n^{\mathcal{O}(1)} = 2^{3^{4^{5^{6^{7^{8^{\text{tw}}}}}}}} \cdot n^{\mathcal{O}(1)}$$

Is it enough to prove that a problem is FPT?

Typically, Courcelle's theorem allows to prove that a problem is FPT...
... but the running time can (and must) be huge!

$$f(\text{tw}) \cdot n^{\mathcal{O}(1)} = 2^{3^{4^{5^{6^{7^{8^{\text{tw}}}}}}}} \cdot n^{\mathcal{O}(1)}$$

Major goal find the smallest possible function $f(\text{tw})$.

This is a very active area in parameterized complexity.

Remark: Algorithms parameterized by treewidth appear very often as a "black box" in all kinds of parameterized algorithms.

Two behaviors for problems parameterized by treewidth

Typically, FPT algorithms parameterized by treewidth are based on dynamic programming (DP) over a tree decomposition.

Two behaviors for problems parameterized by treewidth

Typically, FPT algorithms parameterized by treewidth are based on dynamic programming (DP) over a tree decomposition.

For many problems, like VERTEX COVER or DOMINATING SET, the "natural" DP algorithms lead to (optimal) single-exponential algorithms:

$$2^{\mathcal{O}(\text{tw})} \cdot n^{\mathcal{O}(1)}.$$

Two behaviors for problems parameterized by treewidth

Typically, FPT algorithms parameterized by treewidth are based on dynamic programming (DP) over a tree decomposition.

For many problems, like Vertex Cover or Dominating Set, the "natural" DP algorithms lead to (optimal) single-exponential algorithms:

$$2^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}.$$

But for the so-called connectivity problems, like Longest Path or Steiner Tree, the "natural" DP algorithms provide only time

$$2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}.$$

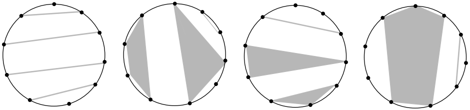
(Single-exponential algorithms on sparse graphs)

On topologically structured graphs (planar, surfaces, minor-free), it is possible to solve connectivity problems in time $2^{\mathcal{O}(\text{tw})} \cdot n^{\mathcal{O}(1)}$:

- Planar graphs: [Dorn, Penninkx, Bodlaender, Fomin. 2005]
- Graphs on surfaces: [Dorn, Fomin, Thilikos. 2006]
  [Rué, S., Thilikos. 2010]
- Minor-free graphs: [Dorn, Fomin, Thilikos. 2008]
  [Rué, S., Thilikos. 2012]

Main idea special type of decomposition with nice topological properties:

partial solutions $\Longleftrightarrow$ non-crossing partitions



$$\text{CN}(k) = \frac{1}{k+1}\binom{2k}{k} \sim \frac{4^k}{\sqrt{\pi}k^{3/2}} \leq 4^k.$$

The revolution of single-exponential algorithms

It was believed that, except on sparse graphs (planar, surfaces), algorithms in time $2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$ were optimal for connectivity problems.

The revolution of single-exponential algorithms

It was believed that, except on sparse graphs (planar, surfaces), algorithms in time $2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$ were optimal for connectivity problems.

This was false!!

Cut&Count technique: [Cygan, Nederlof, Pilipczuk^2, van Rooij, Wojtaszczyk. 2011]
Randomized single-exponential algorithms for connectivity problems.

The revolution of single-exponential algorithms

It was believed that, except on sparse graphs (planar, surfaces), algorithms in time $2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$ were optimal for connectivity problems.

This was false!!

Cut&Count technique: [Cygan, Nederlof, Pilipczuk^2, van Rooij, Wojtaszczyk. 2011]

Randomized single-exponential algorithms for connectivity problems.

Deterministic algorithms with algebraic tricks: [Bodlaender, Cygan, Kratsch, Nederlof. 2013]

Representative sets in matroids: [Fomin, Lokshtanov, Saurabh. 2014]

End of the story?

Do all connectivity problems admit single-exponential algorithms (on general graphs) parameterized by treewidth?

End of the story?

Do all connectivity problems admit single-exponential algorithms (on general graphs) parameterized by treewidth?

No!

Cycle Packing: find the maximum number of vertex-disjoint cycles.

An algorithm in time $2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$ is optimal under the ETH.

[Cygan, Nederlof, Pilipczuk, Pilipczuk, van Rooij, Wojtaszczyk. 2011]

ETH: The 3-SAT problem on n variables cannot be solved in time $2^{o(n)}$

[Impagliazzo, Paturi. 1999]

End of the story?

Do all connectivity problems admit single-exponential algorithms (on general graphs) parameterized by treewidth?

No!

Cycle Packing: find the maximum number of vertex-disjoint cycles.

An algorithm in time $2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$ is optimal under the ETH.

[Cygan, Nederlof, Pilipczuk, Pilipczuk, van Rooij, Wojtaszczyk. 2011]

ETH: The 3-SAT problem on n variables cannot be solved in time $2^{o(n)}$

[Impagliazzo, Paturi. 1999]

There are other examples of such problems...

The $\mathcal{F}$-M-DELETION problem

Let $\mathcal{F}$ be a fixed finite collection of graphs.

The $\mathcal{F}$-M-Deletion problem

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion
Input: A graph G and an integer k.
Parameter: The treewidth tw of G.
Question: Does G contain a set $S \subseteq V(G)$ with $|S| \leq k$ such that $G - S$ does not contain any of the graphs in $\mathcal{F}$ as a minor?

The $\mathcal{F}$-M-Deletion problem

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion	
Input:	A graph G and an integer k.
Parameter:	The treewidth tw of G.
Question:	Does G contain a set $S \subseteq V(G)$ with $\lvert S \rvert \leq k$ such that $G - S$ does not contain any of the graphs in $\mathcal{F}$ as a minor?

- $\mathcal{F} = \{K_2\}$: Vertex Cover.

The $\mathcal{F}$-M-Deletion problem

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion	
Input:	A graph G and an integer k.
Parameter:	The treewidth tw of G.
Question:	Does G contain a set $S \subseteq V(G)$ with $\|S\| \leq k$ such that $G - S$ does not contain any of the graphs in $\mathcal{F}$ as a minor?

- $\mathcal{F} = \{K_2\}$: Vertex Cover.
  Easily solvable in time $2^{\Theta(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.

The $\mathcal{F}$-M-DELETION problem

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-DELETION	
Input:	A graph G and an integer k.
Parameter:	The treewidth tw of G.
Question:	Does G contain a set $S \subseteq V(G)$ with $\|S\| \leq k$ such that $G - S$ does not contain any of the graphs in $\mathcal{F}$ as a minor?

- $\mathcal{F} = \{K_2\}$: VERTEX COVER.
  Easily solvable in time $2^{\Theta(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F} = \{C_3\}$: FEEDBACK VERTEX SET.

The $\mathcal{F}$-M-Deletion problem

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion	
Input:	A graph G and an integer k.
Parameter:	The treewidth tw of G.
Question:	Does G contain a set $S \subseteq V(G)$ with $\vert S\vert \leq k$ such that $G - S$ does not contain any of the graphs in $\mathcal{F}$ as a minor?

- $\mathcal{F} = \{K_2\}$: Vertex Cover.
  Easily solvable in time $2^{\Theta(\text{tw})} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F} = \{C_3\}$: Feedback Vertex Set.
  "Hardly" solvable in time $2^{\Theta(\text{tw})} \cdot n^{\mathcal{O}(1)}$. [Cut&Count. 2011]

The $\mathcal{F}$-M-Deletion problem

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion	
Input:	A graph G and an integer k.
Parameter:	The treewidth tw of G.
Question:	Does G contain a set $S \subseteq V(G)$ with $\|S\| \leq k$ such that $G - S$ does not contain any of the graphs in $\mathcal{F}$ as a minor?

- $\mathcal{F} = \{K_2\}$: Vertex Cover.
  Easily solvable in time $2^{\Theta(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F} = \{C_3\}$: Feedback Vertex Set.
  "Hardly" solvable in time $2^{\Theta(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$. [Cut&Count. 2011]
- $\mathcal{F} = \{K_5, K_{3,3}\}$: Vertex Planarization.

The $\mathcal{F}$-M-DELETION problem

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-DELETION	
Input:	A graph G and an integer k.
Parameter:	The treewidth tw of G.
Question:	Does G contain a set $S \subseteq V(G)$ with $\vert S\vert \leq k$ such that $G - S$ does not contain any of the graphs in $\mathcal{F}$ as a minor?

- $\mathcal{F} = \{K_2\}$: VERTEX COVER.
  Easily solvable in time $2^{\Theta(\text{tw})} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F} = \{C_3\}$: FEEDBACK VERTEX SET.
  "Hardly" solvable in time $2^{\Theta(\text{tw})} \cdot n^{\mathcal{O}(1)}$. [Cut&Count. 2011]
- $\mathcal{F} = \{K_5, K_{3,3}\}$: VERTEX PLANARIZATION.
  Solvable in time $2^{\Theta(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$. [Jansen, Lokshtanov, Saurabh. 2014 + Pilipczuk. 2017]

Covering topological minors

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion	
Input:	A graph G and an integer k.
Parameter:	The treewidth tw of G.
Question:	Does G contain a set $S \subseteq V(G)$ with $\lvert S\rvert \leq k$ such that $G - S$ does not contain any graph in $\mathcal{F}$ as a minor?

Covering topological minors

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion
Input: A graph G and an integer k.
Parameter: The treewidth tw of G.
Question: Does G contain a set $S \subseteq V(G)$ with $|S| \leq k$ such that $G - S$ does not contain any graph in $\mathcal{F}$ as a minor?

$\mathcal{F}$-TM-Deletion
Input: A graph G and an integer k.
Parameter: The treewidth tw of G.
Question: Does G contain a set $S \subseteq V(G)$ with $|S| \leq k$ such that $G - S$ does not contain any graph in $\mathcal{F}$ as a topol. minor?

Covering topological minors

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion
Input: A graph G and an integer k.
Parameter: The treewidth tw of G.
Question: Does G contain a set $S \subseteq V(G)$ with $|S| \leq k$ such that $G - S$ does not contain any graph in $\mathcal{F}$ as a minor?

$\mathcal{F}$-TM-Deletion
Input: A graph G and an integer k.
Parameter: The treewidth tw of G.
Question: Does G contain a set $S \subseteq V(G)$ with $|S| \leq k$ such that $G - S$ does not contain any graph in $\mathcal{F}$ as a topol. minor?

Both problems are NP-hard if $\mathcal{F}$ contains some edge. [Lewis, Yannakakis. 1980]

Covering topological minors

Let $\mathcal{F}$ be a fixed finite collection of graphs.

$\mathcal{F}$-M-Deletion
Input: A graph G and an integer k.
Parameter: The treewidth tw of G.
Question: Does G contain a set $S \subseteq V(G)$ with $|S| \leq k$ such that $G - S$ does not contain any graph in $\mathcal{F}$ as a minor?

$\mathcal{F}$-TM-Deletion
Input: A graph G and an integer k.
Parameter: The treewidth tw of G.
Question: Does G contain a set $S \subseteq V(G)$ with $|S| \leq k$ such that $G - S$ does not contain any graph in $\mathcal{F}$ as a topol. minor?

Both problems are NP-hard if $\mathcal{F}$ contains some edge. [Lewis, Yannakakis. 1980]

FPT by Courcelle, or by Graph Minors theory.

Goal of this project

Objective

Determine, for every fixed $\mathcal{F}$, the (asymptotically) smallest function $f_{\mathcal{F}}$ such that $\mathcal{F}$-M-DELETION/$\mathcal{F}$-TM-DELETION can be solved in time

$$f_{\mathcal{F}}(\mathsf{tw}) \cdot n^{\mathcal{O}(1)}$$

on n-vertex graphs.

Goal of this project

Objective

Determine, for every fixed $\mathcal{F}$, the (asymptotically) smallest function $f_{\mathcal{F}}$ such that $\mathcal{F}$-M-Deletion/$\mathcal{F}$-TM-Deletion can be solved in time

$$f_{\mathcal{F}}(\text{tw}) \cdot n^{\mathcal{O}(1)}$$

on n-vertex graphs.

- We do not want to optimize the degree of the polynomial factor.
- We do not want to optimize the constants.
- Our hardness results hold under the ETH.

Summary of our results

[1] Connected collection $\mathcal{F}$: all the graphs are connected.
[2] Planar collection $\mathcal{F}$: contains at least one planar graph.

Summary of our results

- For every $\mathcal{F}$: $\mathcal{F}$-M/TM-Deletion in time $2^{2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})}} \cdot n^{\mathcal{O}(1)}$.

[1] Connected collection $\mathcal{F}$: all the graphs are connected.

[2] Planar collection $\mathcal{F}$: contains at least one planar graph.

Summary of our results

- For every $\mathcal{F}$: $\mathcal{F}$-M/TM-Deletion in time $2^{2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})}} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F}$ connected[1] + planar[2]: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.

[1] Connected collection $\mathcal{F}$: all the graphs are connected.

[2] Planar collection $\mathcal{F}$: contains at least one planar graph.

Summary of our results

- For every $\mathcal{F}$: $\mathcal{F}$-M/TM-Deletion in time $2^{2^{\mathcal{O}(\mathsf{tw}\cdot\log \mathsf{tw})}} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F}$ connected[1] + planar[2]: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\mathsf{tw}\cdot\log \mathsf{tw})} \cdot n^{\mathcal{O}(1)}$.
- G planar + $\mathcal{F}$ connected: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\mathsf{tw})} \cdot n^{\mathcal{O}(1)}$.

[1] Connected collection $\mathcal{F}$: all the graphs are connected.

[2] Planar collection $\mathcal{F}$: contains at least one planar graph.

Summary of our results

- For every $\mathcal{F}$: $\mathcal{F}$-M/TM-Deletion in time $2^{2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})}} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F}$ connected[1] + planar[2]: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$.
- G planar + $\mathcal{F}$ connected: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\text{tw})} \cdot n^{\mathcal{O}(1)}$.

  (For $\mathcal{F}$-TM-Deletion we need: $\mathcal{F}$ contains a subcubic planar graph.)

[1] Connected collection $\mathcal{F}$: all the graphs are connected.

[2] Planar collection $\mathcal{F}$: contains at least one planar graph.

Summary of our results

- For every $\mathcal{F}$: $\mathcal{F}$-M/TM-Deletion in time $2^{2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})}} \cdot n^{\mathcal{O}(1)}$.

- $\mathcal{F}$ connected[1] + planar[2]: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.

- G planar + $\mathcal{F}$ connected: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.

  (For $\mathcal{F}$-TM-Deletion we need: $\mathcal{F}$ contains a subcubic planar graph.)

- $\mathcal{F}$ (connected): $\mathcal{F}$-M/TM-Deletion not in time $2^{o(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$ unless the ETH fails, even if G planar.

[1] Connected collection $\mathcal{F}$: all the graphs are connected.

[2] Planar collection $\mathcal{F}$: contains at least one planar graph.

Summary of our results

- For every $\mathcal{F}$: $\mathcal{F}$-M/TM-Deletion in time $2^{2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})}} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F}$ connected[1] + planar[2]: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.
- G planar + $\mathcal{F}$ connected: $\mathcal{F}$-M-Deletion in time $2^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.

  (For $\mathcal{F}$-TM-Deletion we need: $\mathcal{F}$ contains a subcubic planar graph.)
- $\mathcal{F}$ (connected): $\mathcal{F}$-M/TM-Deletion not in time $2^{o(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$ unless the ETH fails, even if G planar.
- $\mathcal{F} = \{H\}$, H planar + connected:

[1] Connected collection $\mathcal{F}$: all the graphs are connected.

[2] Planar collection $\mathcal{F}$: contains at least one planar graph.

Summary of our results

- For every $\mathcal{F}$: $\mathcal{F}$-M/TM-DELETION in time $2^{2^{\mathcal{O}(\mathrm{tw}\cdot\log\mathrm{tw})}} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F}$ connected[1] + planar[2]: $\mathcal{F}$-M-DELETION in time $2^{\mathcal{O}(\mathrm{tw}\cdot\log\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.
- G planar + $\mathcal{F}$ connected: $\mathcal{F}$-M-DELETION in time $2^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.

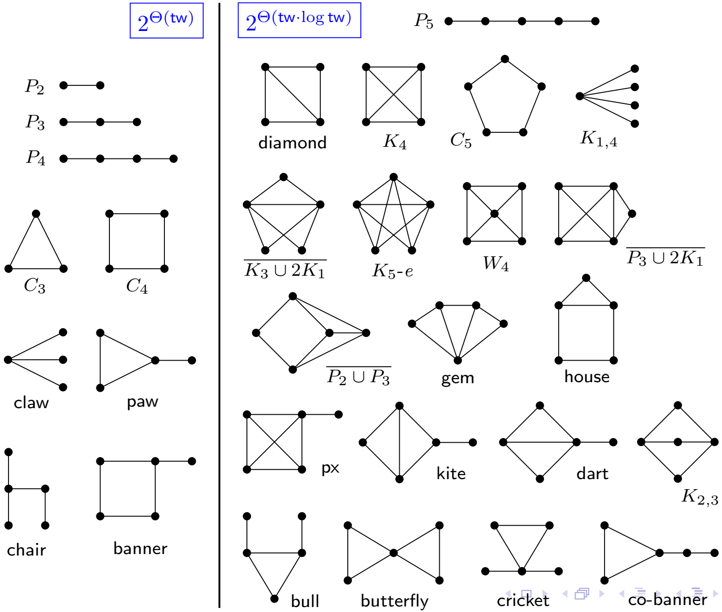
  (For $\mathcal{F}$-TM-DELETION we need: $\mathcal{F}$ contains a subcubic planar graph.)
- $\mathcal{F}$ (connected): $\mathcal{F}$-M/TM-DELETION not in time $2^{o(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$ unless the ETH fails, even if G planar.
- $\mathcal{F} = \{H\}$, H planar + connected: complete tight dichotomy.

---

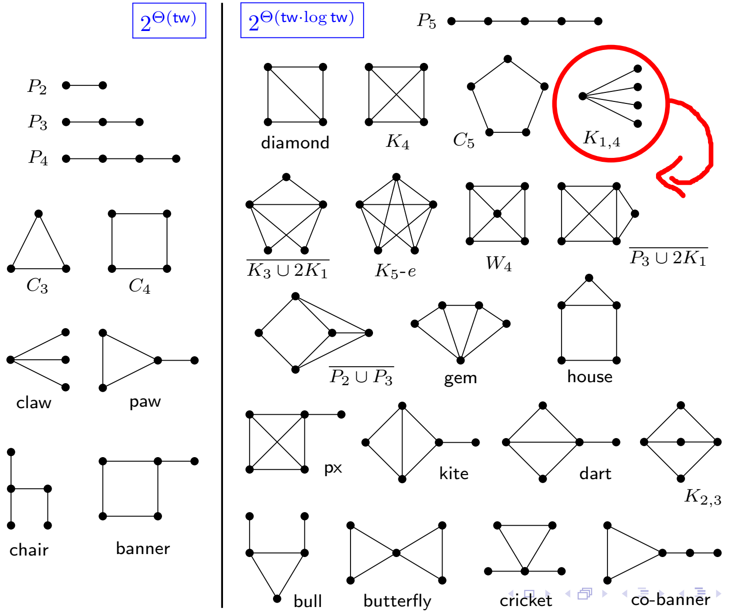
[1] Connected collection $\mathcal{F}$: all the graphs are connected.

[2] Planar collection $\mathcal{F}$: contains at least one planar graph.

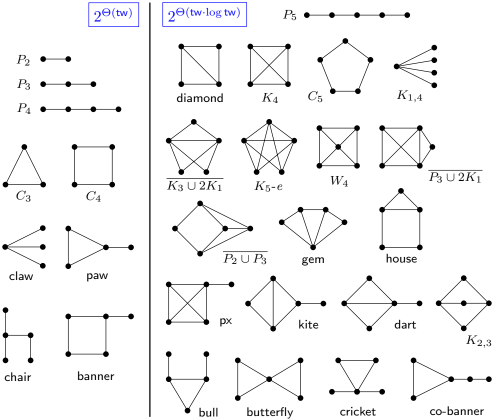
Complexity of hitting small planar minors H



For topological minors, there (at least) one change

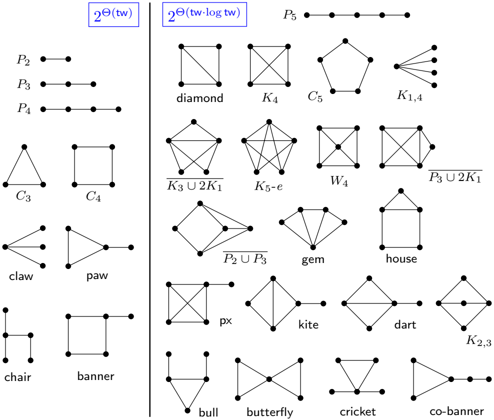


A compact statement for small planar minors



All these cases can be succinctly described as follows:

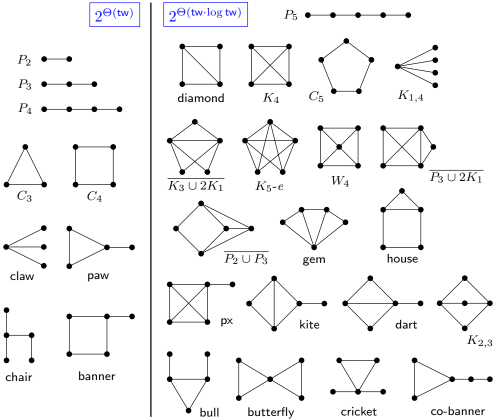
A compact statement for small planar minors



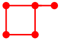
All these cases can be succinctly described as follows:

- All the graphs on the left are minors of  (called the banner)

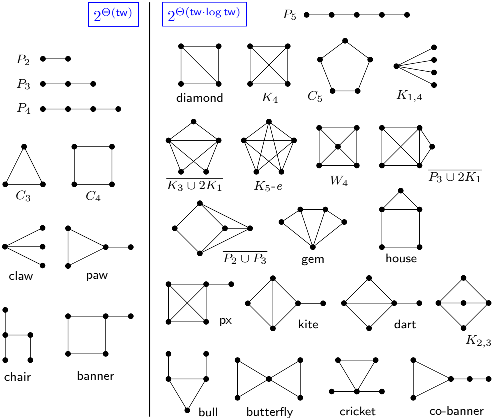
A compact statement for small planar minors



All these cases can be succinctly described as follows:

- All the graphs on the left are minors of  (called the banner)
- All the graphs on the right are not minors of 

A compact statement for small planar minors



All these cases can be succinctly described as follows:

- All the graphs on the left are minors of  (called the banner)
- All the graphs on the right are not minors of  ... except P_5.

A dichotomy for hitting connected planar minors

We can prove that any connected (planar) H with $|V(H)| \geq 6$ is "hard".

A dichotomy for hitting connected planar minors

We can prove that any connected (planar) H with $|V(H)| \geq 6$ is "hard".

Theorem

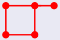
Let H be a connected planar graph.

A dichotomy for hitting connected planar minors

We can prove that any connected (planar) H with $|V(H)| \geq 6$ is "hard".

Theorem

Let H be a connected planar graph.
The $\{H\}$-M-DELETION problem is solvable in time

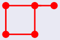
- $2^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$, *if* $H \preceq_m$  *and* $H \neq P_5$.

A dichotomy for hitting connected planar minors

We can prove that any connected (planar) H with $|V(H)| \geq 6$ is “hard”.

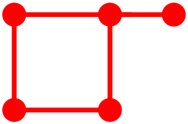
Theorem

Let H be a connected planar graph.
The $\{H\}$-M-Deletion problem is solvable in time

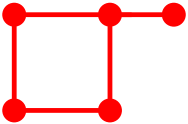
- $2^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$, *if* $H \preceq_{\mathrm{m}}$  *and* $H \neq P_5$.
- $2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$, *otherwise.*

In both cases, the running time is asymptotically optimal under the ETH.

Why the banner??

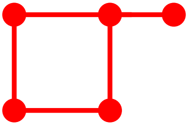


Why the banner??



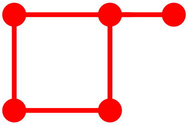
- Every connected component (with at least 5 vertices) of a graph that excludes the banner as a (topological) minor is either:

Why the banner??



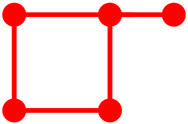
- Every connected component (with at least 5 vertices) of a graph that excludes the banner as a (topological) minor is either:
  - a cycle (of any length),
  - or a tree in which some vertices have been replaced by triangles.

Why the banner??



- Every connected component (with at least 5 vertices) of a graph that excludes the banner as a (topological) minor is either:
  - a cycle (of any length),
  - or a tree in which some vertices have been replaced by triangles.
- Both such types of components can be maintained by a dynamic programming algorithm in single-exponential time.

Why the banner??



- Every connected component (with at least 5 vertices) of a graph that excludes the banner as a (topological) minor is either:
  - a cycle (of any length),
  - or a tree in which some vertices have been replaced by triangles.
- Both such types of components can be maintained by a dynamic programming algorithm in single-exponential time.
- If the characterization of the allowed connected components is enriched in some way, such as restricting the length of the allowed cycles or forbidding certain degrees, the problem becomes harder.

We have three types of results

We have three types of results

1. General algorithms

- For every $\mathcal{F}$: time $2^{2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})}} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F}$ connected + planar: time $2^{\mathcal{O}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.
- G planar + $\mathcal{F}$ connected: time $2^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.

We have three types of results

1. General algorithms
   - For every $\mathcal{F}$: time $2^{2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})}} \cdot n^{\mathcal{O}(1)}$.
   - $\mathcal{F}$ connected + planar: time $2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$.
   - G planar + $\mathcal{F}$ connected: time $2^{\mathcal{O}(\text{tw})} \cdot n^{\mathcal{O}(1)}$.

2. Ad-hoc single-exponential algorithms
   - Some use "typical" dynamic programming.
   - Some use the rank-based approach. [Bodlaender, Cygan, Kratsch, Nederlof. 2013]

We have three types of results

1. General algorithms
   - For every $\mathcal{F}$: time $2^{2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})}} \cdot n^{\mathcal{O}(1)}$.
   - $\mathcal{F}$ connected + planar: time $2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$.
   - G planar + $\mathcal{F}$ connected: time $2^{\mathcal{O}(\text{tw})} \cdot n^{\mathcal{O}(1)}$.
2. Ad-hoc single-exponential algorithms
   - Some use "typical" dynamic programming.
   - Some use the rank-based approach. [Bodlaender, Cygan, Kratsch, Nederlof. 2013]
3. Lower bounds under the ETH
   - $2^{o(\text{tw})}$ is "easy".
   - $2^{o(\text{tw} \cdot \log \text{tw})}$ is much more involved and we get ideas from:

     [Lokshtanov, Marx, Saurabh. 2011] [Marcin Pilipczuk. 2017] [Bonnet, Brettell, Kwon, Marx. 2017]

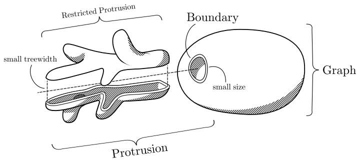
Some ideas of the general algorithms

- For every $\mathcal{F}$: time $2^{2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})}} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F}$ connected + planar: time $2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$.
- G planar + $\mathcal{F}$ connected: time $2^{\mathcal{O}(\text{tw})} \cdot n^{\mathcal{O}(1)}$.

Some ideas of the general algorithms

- For every $\mathcal{F}$: time $2^{2^{\mathcal{O}(\text{tw}\cdot\log \text{tw})}} \cdot n^{\mathcal{O}(1)}$.
- $\mathcal{F}$ connected + planar: time $2^{\mathcal{O}(\text{tw}\cdot\log \text{tw})} \cdot n^{\mathcal{O}(1)}$.
- G planar + $\mathcal{F}$ connected: time $2^{\mathcal{O}(\text{tw})} \cdot n^{\mathcal{O}(1)}$.

We build on the machinery of boundaried graphs and representatives:



[Bodlaender, Fomin, Lokshtanov, Penninkx, Saurabh, Thilikos. 2009]

[Fomin, Lokshtanov, Saurabh, Thilikos. 2010]

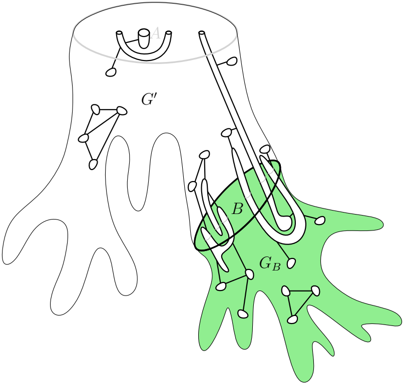
[Kim, Langer, Paul, Reidl, Rossmanith, S., Sikdar. 2013]

[Garnero, Paul, S., Thilikos. 2014]

skip

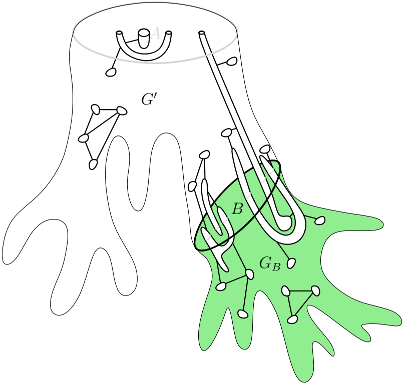
Algorithm for a general collection $\mathcal{F}$

- We see G as a t-boundaried graph.



Algorithm for a general collection $\mathcal{F}$

- We see G as a t-boundaried graph.
- folio of G: set of all its $\mathcal{F}$-minor-free minors, up to size $\mathcal{O}_{\mathcal{F}}(t)$.

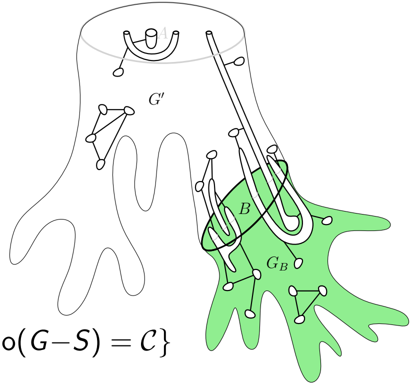


Algorithm for a general collection $\mathcal{F}$

- We see G as a t-boundaried graph.

- folio of G: set of all its $\mathcal{F}$-minor-free minors, up to size $\mathcal{O}_{\mathcal{F}}(t)$.

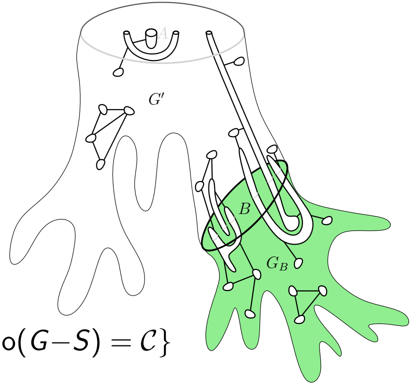
- We compute, using DP over a tree decomposition of G, the following parameter for every folio $\mathcal{C}$:



$$\mathbf{p}(G,\mathcal{C}) = \min\{|S| \ : \ S \subseteq V(G) \wedge \text{folio}(G-S) = \mathcal{C}\}$$

Algorithm for a general collection $\mathcal{F}$

- We see G as a t-boundaried graph.
- folio of G: set of all its $\mathcal{F}$-minor-free minors, up to size $\mathcal{O}_{\mathcal{F}}(t)$.
- We compute, using DP over a tree decomposition of G, the following parameter for every folio $\mathcal{C}$:



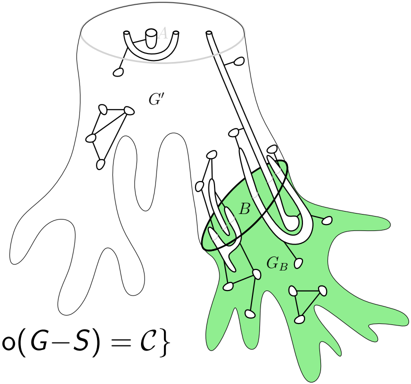
$$\mathbf{p}(G, \mathcal{C}) = \min\{|S| \ : \ S \subseteq V(G) \wedge \text{folio}(G - S) = \mathcal{C}\}$$

- For every t-boundaried graph G, $|\text{folio}(G)| = 2^{\mathcal{O}_{\mathcal{F}}(t \log t)}$.

Algorithm for a general collection $\mathcal{F}$

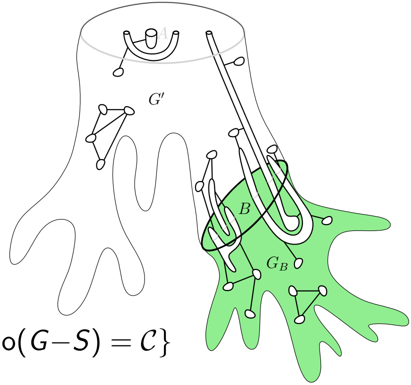
- We see G as a t-boundaried graph.
- folio of G: set of all its $\mathcal{F}$-minor-free minors, up to size $\mathcal{O}_{\mathcal{F}}(t)$.
- We compute, using DP over a tree decomposition of G, the following parameter for every folio $\mathcal{C}$:

$$\mathbf{p}(G, \mathcal{C}) = \min\{|S| \ : \ S \subseteq V(G) \wedge \text{folio}(G - S) = \mathcal{C}\}$$



- For every t-boundaried graph G, $|\text{folio}(G)| = 2^{\mathcal{O}_{\mathcal{F}}(t \log t)}$.
- The number of distinct folios is $2^{2^{\mathcal{O}_{\mathcal{F}}(t \log t)}}$.

Algorithm for a general collection $\mathcal{F}$



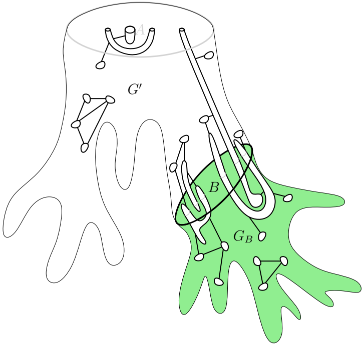
- We see G as a t-boundaried graph.
- folio of G: set of all its $\mathcal{F}$-minor-free minors, up to size $\mathcal{O}_{\mathcal{F}}(t)$.
- We compute, using DP over a tree decomposition of G, the following parameter for every folio $\mathcal{C}$:

$$\mathbf{p}(G, \mathcal{C}) = \min\{|S| \ : \ S \subseteq V(G) \land \text{folio}(G - S) = \mathcal{C}\}$$

- For every t-boundaried graph G, $|\text{folio}(G)| = 2^{\mathcal{O}_{\mathcal{F}}(t \log t)}$.
- The number of distinct folios is $2^{2^{\mathcal{O}_{\mathcal{F}}(t \log t)}}$.
- This gives an algorithm running in time $2^{2^{\mathcal{O}_{\mathcal{F}}(\text{tw} \cdot \log \text{tw})}} \cdot n^{\mathcal{O}(1)}$.

skip

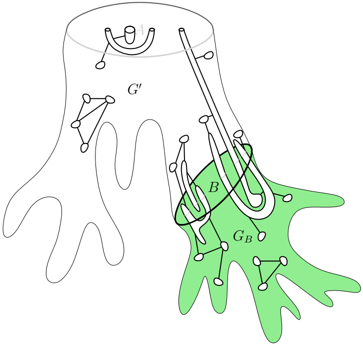
Algorithm for a connected and planar collection $\mathcal{F}$



Algorithm for a connected and planar collection $\mathcal{F}$

- For a fixed $\mathcal{F}$, we define an equivalence relation $\equiv^{(\mathcal{F},t)}$ on t-boundaried graphs:

$$G_1 \equiv^{(\mathcal{F},t)} G_2 \quad \text{if } \forall G' \in \mathcal{B}^t,$$
$$\mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_1 \iff \mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_2.$$

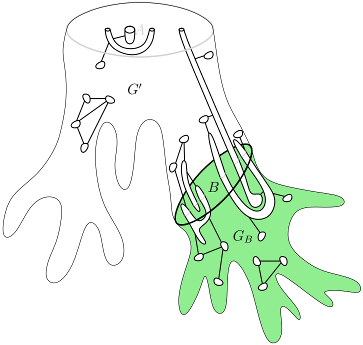


Algorithm for a connected and planar collection $\mathcal{F}$

- For a fixed $\mathcal{F}$, we define an equivalence relation $\equiv^{(\mathcal{F},t)}$ on t-boundaried graphs:

  $$G_1 \equiv^{(\mathcal{F},t)} G_2 \quad \text{if } \forall G' \in \mathcal{B}^t,$$
  $$\mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_1 \iff \mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_2.$$

- $\mathcal{R}^{(\mathcal{F},t)}$: set of minimum-size representatives of $\equiv^{(\mathcal{F},t)}$.

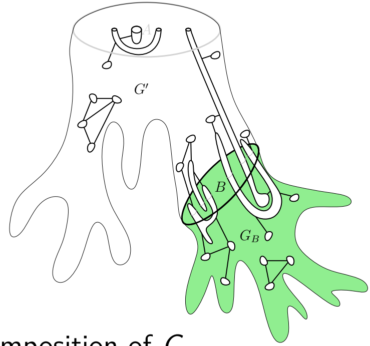


Algorithm for a connected and planar collection $\mathcal{F}$

- For a fixed $\mathcal{F}$, we define an equivalence relation $\equiv^{(\mathcal{F},t)}$ on t-boundaried graphs:

$$G_1 \equiv^{(\mathcal{F},t)} G_2 \quad \text{if } \forall G' \in \mathcal{B}^t,$$
$$\mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_1 \iff \mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_2.$$

- $\mathcal{R}^{(\mathcal{F},t)}$: set of minimum-size representatives of $\equiv^{(\mathcal{F},t)}$.



- We compute, using DP over a tree decomposition of G, the following parameter for every representative R:

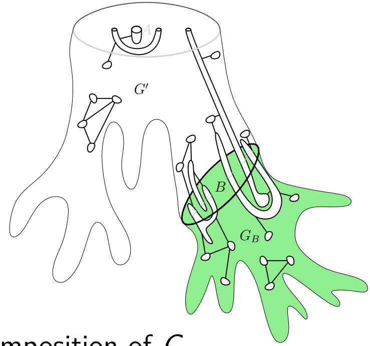
$$\mathbf{p}(G,R) = \min\{|S| \ : \ S \subseteq V(G) \ \wedge \ \mathrm{rep}_{\mathcal{F},t}(G-S) = R\}$$

Algorithm for a connected and planar collection $\mathcal{F}$

- For a fixed $\mathcal{F}$, we define an equivalence relation $\equiv^{(\mathcal{F},t)}$ on t-boundaried graphs:

$$G_1 \equiv^{(\mathcal{F},t)} G_2 \quad \text{if } \forall G' \in \mathcal{B}^t,$$
$$\mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_1 \iff \mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_2.$$

- $\mathcal{R}^{(\mathcal{F},t)}$: set of minimum-size representatives of $\equiv^{(\mathcal{F},t)}$.

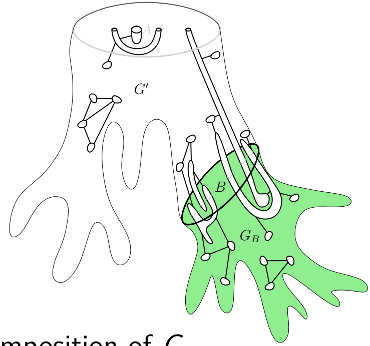


- We compute, using DP over a tree decomposition of G, the following parameter for every representative R:

$$\mathbf{p}(G,R) = \min\{|S| \ : \ S \subseteq V(G) \ \wedge \ \mathrm{rep}_{\mathcal{F},t}(G-S) = R\}$$

- The number of representatives is $|\mathcal{R}^{(\mathcal{F},t)}| = 2^{\mathcal{O}_{\mathcal{F}}(t \cdot \log t)}$.

Algorithm for a connected and planar collection $\mathcal{F}$



- For a fixed $\mathcal{F}$, we define an equivalence relation $\equiv^{(\mathcal{F},t)}$ on t-boundaried graphs:

$$G_1 \equiv^{(\mathcal{F},t)} G_2 \quad \text{if } \forall G' \in \mathcal{B}^t,$$
$$\mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_1 \iff \mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_2.$$

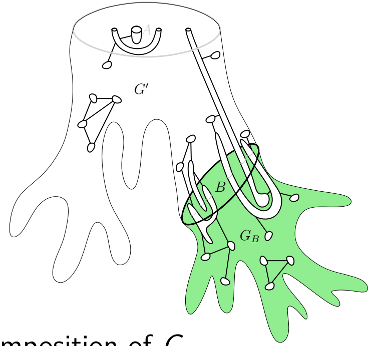
- $\mathcal{R}^{(\mathcal{F},t)}$: set of minimum-size representatives of $\equiv^{(\mathcal{F},t)}$.

- We compute, using DP over a tree decomposition of G, the following parameter for every representative R:

$$\mathbf{p}(G,R) = \min\{|S| \ : \ S \subseteq V(G) \ \wedge \ \mathrm{rep}_{\mathcal{F},t}(G-S) = R\}$$

- The number of representatives is $|\mathcal{R}^{(\mathcal{F},t)}| = 2^{\mathcal{O}_{\mathcal{F}}(t \cdot \log t)}$.
  $\#$ labeled graphs of size $\leq t$ and tw $\leq h$ is $2^{\mathcal{O}_h(t \cdot \log t)}$. [Baste, Noy, S. 2017]

Algorithm for a connected and planar collection $\mathcal{F}$



- For a fixed $\mathcal{F}$, we define an equivalence relation $\equiv^{(\mathcal{F},t)}$ on t-boundaried graphs:

  $G_1 \equiv^{(\mathcal{F},t)} G_2$ if $\forall G' \in \mathcal{B}^t$,
  $\mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_1 \iff \mathcal{F} \preceq_{\mathrm{m}} G' \oplus G_2$.

- $\mathcal{R}^{(\mathcal{F},t)}$: set of minimum-size representatives of $\equiv^{(\mathcal{F},t)}$.

- We compute, using DP over a tree decomposition of G, the following parameter for every representative R:

$$\mathbf{p}(G,R) = \min\{|S| \ : \ S \subseteq V(G) \ \wedge \ \mathrm{rep}_{\mathcal{F},t}(G-S) = R\}$$

- The number of representatives is $|\mathcal{R}^{(\mathcal{F},t)}| = 2^{\mathcal{O}_{\mathcal{F}}(t \cdot \log t)}$.
  \# labeled graphs of size $\leq t$ and tw $\leq h$ is $2^{\mathcal{O}_h(t \cdot \log t)}$. [Baste, Noy, S. 2017]

- This gives an algorithm running in time $2^{\mathcal{O}_{\mathcal{F}}(\mathrm{tw} \cdot \log \mathrm{tw})} \cdot n^{\mathcal{O}(1)}$.

skip

Algorithm when the input graph G is planar

- Idea get an improved bound on $|\mathcal{R}^{(\mathcal{F},t)}|$.

Algorithm when the input graph G is planar

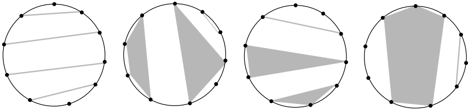
- Idea get an improved bound on $|\mathcal{R}^{(\mathcal{F},t)}|$.
- We use a sphere-cut decomposition of the input planar graph G.

  [Seymour, Thomas. 1994] [Dorn, Penninkx, Bodlaender, Fomin. 2010]

Algorithm when the input graph G is planar

- Idea get an improved bound on $|\mathcal{R}^{(\mathcal{F},t)}|$.
- We use a sphere-cut decomposition of the input planar graph G.

  [Seymour, Thomas. 1994] [Dorn, Penninkx, Bodlaender, Fomin. 2010]
- Nice topological properties: each separator corresponds to a noose.

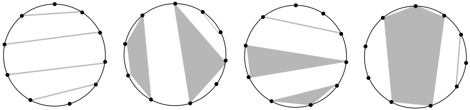


Algorithm when the input graph G is planar

- Idea get an improved bound on $|\mathcal{R}^{(\mathcal{F},t)}|$.
- We use a sphere-cut decomposition of the input planar graph G.

  [Seymour, Thomas. 1994] [Dorn, Penninkx, Bodlaender, Fomin. 2010]

- Nice topological properties: each separator corresponds to a noose.



- The number of representatives is $|\mathcal{R}^{(\mathcal{F},t)}| = 2^{\mathcal{O}_{\mathcal{F}}(t)}$.

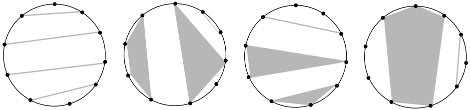
  Number of planar triangulations on t vertices is $2^{\mathcal{O}(t)}$. [Tutte. 1962]

Algorithm when the input graph G is planar

- Idea get an improved bound on $|\mathcal{R}^{(\mathcal{F},t)}|$.
- We use a sphere-cut decomposition of the input planar graph G.

  [Seymour, Thomas. 1994] [Dorn, Penninkx, Bodlaender, Fomin. 2010]

- Nice topological properties: each separator corresponds to a noose.

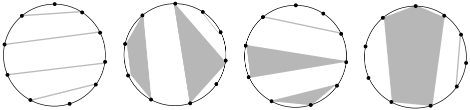


- The number of representatives is $|\mathcal{R}^{(\mathcal{F},t)}| = 2^{\mathcal{O}_{\mathcal{F}}(t)}$.
  Number of planar triangulations on t vertices is $2^{\mathcal{O}(t)}$. [Tutte. 1962]
- This gives an algorithm running in time $2^{\mathcal{O}_{\mathcal{F}}(\text{tw})} \cdot n^{\mathcal{O}(1)}$.

Algorithm when the input graph G is planar

- Idea get an improved bound on $|\mathcal{R}^{(\mathcal{F},t)}|$.
- We use a sphere-cut decomposition of the input planar graph G.

  [Seymour, Thomas. 1994] [Dorn, Penninkx, Bodlaender, Fomin. 2010]
- Nice topological properties: each separator corresponds to a noose.



- The number of representatives is $|\mathcal{R}^{(\mathcal{F},t)}| = 2^{\mathcal{O}_{\mathcal{F}}(t)}$.
  Number of planar triangulations on t vertices is $2^{\mathcal{O}(t)}$. [Tutte. 1962]
- This gives an algorithm running in time $2^{\mathcal{O}_{\mathcal{F}}(\text{tw})} \cdot n^{\mathcal{O}(1)}$.
- We can extend this algorithm to input graphs G embedded in arbitrary surfaces by using surface-cut decompositions. [Rué, S., Thilikos. 2014]

What's next about $\mathcal{F}$-Deletion?

What's next about $\mathcal{F}$-Deletion?

- Ultimate goal: classify the (asymptotically) tight complexity of $\mathcal{F}$-Deletion for every family $\mathcal{F}$.

What's next about $\mathcal{F}$-Deletion?

- Ultimate goal: classify the (asymptotically) tight complexity of $\mathcal{F}$-Deletion for every family $\mathcal{F}$.
- Dichotomy for $\{H\}$-TM-Deletion when H planar + connected.

What's next about $\mathcal{F}$-Deletion?

- Ultimate goal: classify the (asymptotically) tight complexity of $\mathcal{F}$-Deletion for every family $\mathcal{F}$.
- Dichotomy for $\{H\}$-TM-Deletion when H planar + connected.
- Only "missing" connected graph on at most 5 vertices: K_5.
  We think that $\{K_5\}$-Deletion is solvable in time $2^{\Theta(\mathrm{tw}\cdot\log\mathrm{tw})}\cdot n^{\mathcal{O}(1)}$.

What's next about $\mathcal{F}$-Deletion?

- Ultimate goal: classify the (asymptotically) tight complexity of $\mathcal{F}$-Deletion for every family $\mathcal{F}$.
- Dichotomy for $\{H\}$-TM-Deletion when H planar + connected.
- Only "missing" connected graph on at most 5 vertices: K_5.
  We think that $\{K_5\}$-Deletion is solvable in time $2^{\Theta(\mathsf{tw} \cdot \log \mathsf{tw})} \cdot n^{\mathcal{O}(1)}$.
- We do not even know if there exists some $\mathcal{F}$ such that $\mathcal{F}$-Deletion cannot be solved in time $2^{o(\mathsf{tw}^2)} \cdot n^{\mathcal{O}(1)}$ under the ETH.

What's next about $\mathcal{F}$-Deletion?

- Ultimate goal: classify the (asymptotically) tight complexity of $\mathcal{F}$-Deletion for every family $\mathcal{F}$.
- Dichotomy for $\{H\}$-TM-Deletion when H planar + connected.
- Only "missing" connected graph on at most 5 vertices: K_5.
  We think that $\{K_5\}$-Deletion is solvable in time $2^{\Theta(\mathsf{tw} \cdot \log \mathsf{tw})} \cdot n^{\mathcal{O}(1)}$.
- We do not even know if there exists some $\mathcal{F}$ such that $\mathcal{F}$-Deletion cannot be solved in time $2^{o(\mathsf{tw}^2)} \cdot n^{\mathcal{O}(1)}$ under the ETH.

  Deletion to genus at most g: $2^{\mathcal{O}_g(\mathsf{tw} \cdot \log \mathsf{tw})} \cdot n^{\mathcal{O}(1)}$. [Kociumaka, Pilipczuk. 2017]

What's next about $\mathcal{F}$-Deletion?

- Ultimate goal: classify the (asymptotically) tight complexity of $\mathcal{F}$-Deletion for every family $\mathcal{F}$.
- Dichotomy for $\{H\}$-TM-Deletion when H planar + connected.
- Only "missing" connected graph on at most 5 vertices: K_5.
  We think that $\{K_5\}$-Deletion is solvable in time $2^{\Theta(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$.
- We do not even know if there exists some $\mathcal{F}$ such that $\mathcal{F}$-Deletion cannot be solved in time $2^{o(\text{tw}^2)} \cdot n^{\mathcal{O}(1)}$ under the ETH.

  Deletion to genus at most g: $2^{\mathcal{O}_g(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$. [Kociumaka, Pilipczuk. 2017]
- Conjecture: For every connected family $\mathcal{F}$, the $\mathcal{F}$-Deletion problem is solvable in time $2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$.

What's next about $\mathcal{F}$-Deletion?

- Ultimate goal: classify the (asymptotically) tight complexity of $\mathcal{F}$-Deletion for every family $\mathcal{F}$.
- Dichotomy for $\{H\}$-TM-Deletion when H planar + connected.
- Only "missing" connected graph on at most 5 vertices: K_5.
  We think that $\{K_5\}$-Deletion is solvable in time $2^{\Theta(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$.
- We do not even know if there exists some $\mathcal{F}$ such that $\mathcal{F}$-Deletion cannot be solved in time $2^{o(\text{tw}^2)} \cdot n^{\mathcal{O}(1)}$ under the ETH.

  Deletion to genus at most g: $2^{\mathcal{O}_g(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$. [Kociumaka, Pilipczuk. 2017]
- Conjecture For every connected family $\mathcal{F}$, the $\mathcal{F}$-Deletion problem is solvable in time $2^{\mathcal{O}(\text{tw} \cdot \log \text{tw})} \cdot n^{\mathcal{O}(1)}$.
- Consider families $\mathcal{F}$ containing disconnected graphs.

Gràcies!