FEDOR V. FOMIN

Graph Minors, Bidimensionality and Algorithms

PART III

Warsaw, 2011

Reminder

Back to the YEAR 2010...

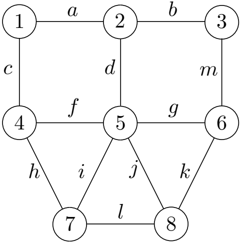
Branch Decompositions

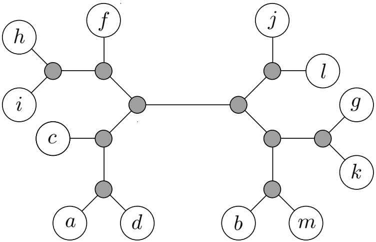
Definition

A branch decomposition of a graph $G = (V, E)$ is a tuple (T, μ) where

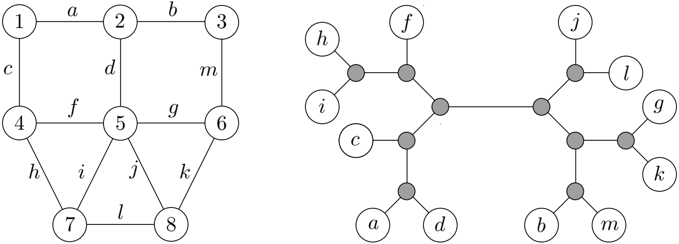
- T is a tree with degree 3 for all internal nodes.
- μ is a bijection between the leaves of T and $E(G)$.

Example of Branch Decomposition

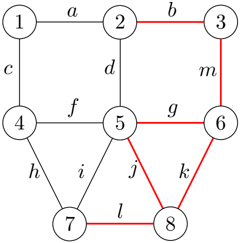


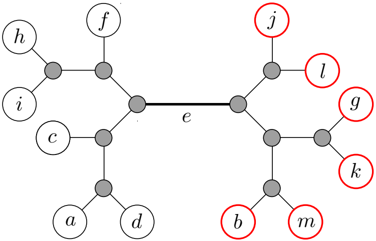


Edge $e \in T$ partitions the edge set of G in A_e and B_e

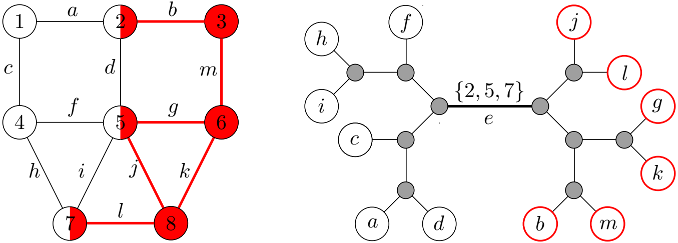


Edge $e \in T$ partitions the edge set of G in A_e and B_e





Middle set $\mathrm{mid}(e) = V(A_e) \cap V(B_e)$

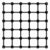


Branchwidth

- The *width* of a branch decomposition is $\max_{e \in T} |\operatorname{mid}(e)|$.
- The *branchwidth* of a graph G is the minimum width over all branch decompositions of G.

Grid Theorem

Theorem (Robertson, Seymour & Thomas, 1994)

Let $\ell \geq 1$ be an integer. Every planar graph of branchwidth $\geq 4\ell$ contains $_{\ell}$ *as a minor.*

Subexponential algorithms on planar graphs: What is the main idea?

Dynamic programming and Grid Theorem

Meta conditions

(A) For every graph $G \in \mathcal{G}$, $\mathbf{bw}(G) \leq \alpha \cdot \sqrt{P(G)} + O(1)$

(B) For every graph $G \in \mathcal{G}$ and given a branch decomposition (T, μ) of G, the value of $P(G)$ can be computed in $f(\mathbf{bw}(T, \mu)) \cdot n^{O(1)}$ steps.

Algorithm

(A) For every graph $G \in \mathcal{G}$, $\mathbf{bw}(G) \leq \alpha \cdot \sqrt{P(G)} + O(1)$

(B) For every graph $G \in \mathcal{G}$ and given a branch decomposition (T, μ) of G, the value of $P(G)$ can be computed in $f(\mathbf{bw}(T, \mu)) \cdot n^{O(1)}$ steps.

If $\mathbf{bw}(T, \mu) > \alpha \cdot \sqrt{k}$, then by (A) the answer is clear

Else, by (B), $P(G)$ can be computed in $f(\alpha \cdot \sqrt{k}) \cdot n^{O(1)}$ steps.

When $f(k) = 2^{O(k)}$, the running time is $2^{O(\sqrt{k})} \cdot n^{O(1)}$

Bidimensionality: The main idea

If the graph parameter is closed under taking minors or contractions, the only thing needed for the proof branchwidth/parameter bound is to understand how this parameter behaves on a (partially triangulated) grid.

Bidimensionality: Demaine, FF, Hajiaghayi, Thilikos, 2005

Definition

A parameter P is *minor bidimensional with density* δ if

1. P is closed under taking of minors, and
2. for the $(r \times r)$-grid R, $P(R) = (\delta r)^2 + o((\delta r)^2)$.

Bidimensionality: Demaine, FF, Hajiaghayi, Thilikos, 2005

Definition

A parameter P is called *contraction bidimensional with density* δ if

1. P is closed under contractions,
2. for any partially triangulated $(r \times r)$-grid R,

   $P(R) = (\delta_R r)^2 + o((\delta_R r)^2)$, and
3. δ is the smallest δ_R among all paritally triangulated

   $(r \times r)$-grids.

Bidimensionality

(A) For every graph $G \in \mathcal{G}$, $\mathbf{bw}(G) \leq \alpha \cdot \sqrt{P(G)} + O(1)$

Lemma

If P is a bidimensional parameter with density δ then P satisfies property (A) for $\alpha = 4/\delta$, on planar graphs.

Proof.

Let R be an $(r \times r)$-grid.

$P(R) \geq (\delta_R r)^2$.

If G contains R as a minor, then $\mathbf{bw}(G) \leq 4r \leq 4/\delta\sqrt{P(G)}$. $\square$

Examples of bidimensional problems

Vertex cover

Dominating Set

Independent Set

(k, r)-center

Feedback Vertex Set

Minimum Maximal Matching

Planar Graph TSP

Longest Path ...

Bidimensional theory

If $\mathbf{P}$ is a parameter that

(A) is minor (contraction) bidimensional

(B) can be computed in $f(\mathbf{bw}(G)) \cdot n^{O(1)}$ steps.

then there is a $f(O(\sqrt{k})) \cdot n^{O(1)}$ step algorithm for checking

whether $\mathbf{P}(G) \leq k$ for H (apex) -minor free graphs.

We now fix our attention to property (B) and function f.

PART III

Dynamic programming and Catalan structures

Dynamic programming for branch decompositions

- We root the tree T of the branch decomposition (T, τ),
- We define a partial solution for each cut-set of an edge e of T
- We compute all partial solutions bottom-up (using the partial solutions corresponding to the children edges).

This can be done in $O(f(\ell) \cdot n)$ if we have a branch decomposition of width at most ℓ.

Dynamic programming for branch decompositions

- We root the tree T of the branch decomposition (T, τ),
- We define a partial solution for each cut-set of an edge e of T
- We compute all partial solutions bottom-up (using the partial solutions corresponding to the children edges).

This can be done in $O(f(\ell) \cdot n)$ if we have a branch decomposition of width at most ℓ.

$f(\ell)$ depends on the number of partial solutions we have to compute for each edge of T.

- To find a good bound for $f(\ell)$ is important!

For many problems, $2^{O(\mathbf{bw}(G))} \cdot n^{O(1)}$ step algorithms exist.

Dynamic programming on graphs with small branchwidth gives such algorithms for problems like

Vertex Cover,

Dominating Set, or

Edge Dominating Set, (and others...)

However: There are (many) problems where no general $2^{O(\mathbf{bw}(G))} \cdot n^{O(1)}$ step algorithm is known.

However: There are (many) problems where no general $2^{O(\mathbf{bw}(G))} \cdot n^{O(1)}$ step algorithm is known.

Such problems are

Longest Path, Longest Cycle, Connected Dominating Set, Feedback Vertex Set, Hamiltonian Cycle, Max Leaf Tree and Graph Metric TSP

However: There are (many) problems where no general $2^{O(\mathbf{bw}(G))} \cdot n^{O(1)}$ step algorithm is known.

Such problems are

Longest Path, Longest Cycle, Connected Dominating Set, Feedback Vertex Set, Hamiltonian Cycle, Max Leaf Tree and Graph Metric TSP

For the natural parameterizations of these problems, no $2^{O(\sqrt{k})} \cdot n^{O(1)}$ step FPT-algorithm follows by just using bidimensionality theory and dynamic programming.

Example: k- Path

The k-Path problem is to decide, given a graph G and a positive integer k, whether G contains a path of length k.

k- Path

On general graphs k-Path can be solved Color Coding technique in time $O((2e)^k n)$ [Alon, Yuster, Zwick, 1995].

A chain of improvements up to recent $O^*(1.66^k)$ algorithm of [Björklund, Husfeldt, Kaski, Koivisto, 2010]

k-Path

On planar graphs k-Path has a

$2^{O(\sqrt{k} \cdot \log k)} \cdot n^{O(1)}$ step algorithm.

Because

k-PATH

On planar graphs k-PATH has a

$2^{O(\sqrt{k} \cdot \log k)} \cdot n^{O(1)}$ step algorithm.

Because

(A) The parameter is minor bidimensional

k-PATH

On planar graphs k-PATH has a

$2^{O(\sqrt{k} \cdot \log k)} \cdot n^{O(1)}$ step algorithm.

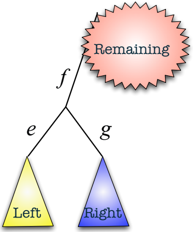
Because

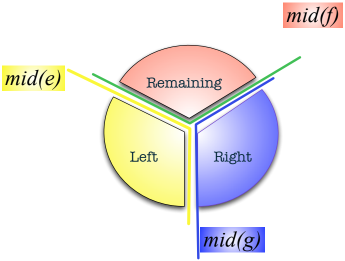
(A) The parameter is minor bidimensional

(B) to find a longest path in a graph G takes

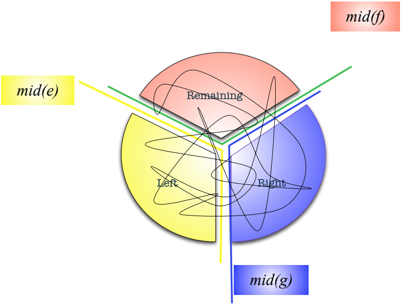
$2^{O(\mathbf{bw}(G) \cdot \log\ \mathbf{bw}(G))} \cdot n$ steps

Dynamic programming: k-Path





Dynamic programming: k-Path



k-Path

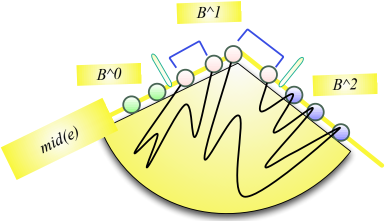
Fact: Given a branch decomposition of width w, k-PATH can be solved in time $w^{O(w)} \cdot n$.

G_e: subgraph formed by edges of the subtree rooted at e.

If P is a path, then the subgraph

$P_e = G_e \cap P$ is a set of paths with

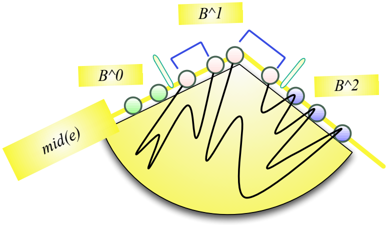
endpoints in $\mathrm{mid}(e)$.



k-Path

What are the important properties of P_e "seen from the outside world"?

- The subsets B_e^0, B_e^1, B_e^2 of $\mathrm{mid}(e)$ having degree 0, 1, and 2.
- Disjoint pairs $\texttt{pairs}(e)$ of B_e^1.



Number of subproblems $(B_e^0, B_e^1, B_e^2, \texttt{pairs}(e))$ for each edge e: at most $3^w \cdot w^w$.

The running time of dynamic programming is proportional to the amount of all possible subproblems $(B_e^0, B_e^1, B_e^2, \texttt{pairs}(e))$, which is $3^w \cdot w^w = 2^{O(w \log w)}$.

The running time of dynamic programming is proportional to the amount of all possible subproblems $(B_e^0, B_e^1, B_e^2, \texttt{pairs}(e))$, which is $3^w \cdot w^w = 2^{O(w \log w)}$.

- Issue: The same problem appears in many dynamic programming algorithms

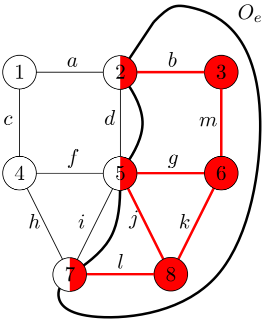
The running time of dynamic programming is proportional to the amount of all possible subproblems $(B_e^0, B_e^1, B_e^2, \texttt{pairs}(e))$, which is $3^w \cdot w^w = 2^{O(w \log w)}$.

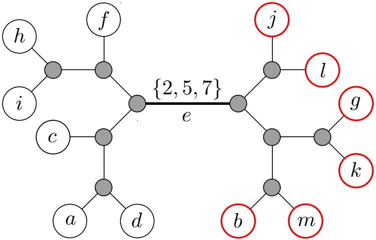
- Issue: The same problem appears in many dynamic programming algorithms
- Idea: as long as we care about sparse graph classes, we can take their structure into consideration
- We want to show that on planar graphs, the amount of pairs $\texttt{pairs}(e)$ in $\mathrm{mid}(e)$ is $2^{O(w)}$

Sphere-cut decomposition

Let G be a planar graph embedded on the sphere (or a plane) $\mathcal{S}_0$

A *sphere-cut decomposition* of G is a branch decomposition (T, τ) where for every $e \in E(T)$, the vertices in $\mathrm{mid}(e)$ are the vertices in a Jordan curve of $\mathcal{S}_0$ that meets no edges of G (a noose).





Seymour-Thomas 1994, Dorn-Penninkx-Bodlaender-FF 2005

Theorem

Every planar graph G of branchwidth ℓ has a sphere-cut decomposition of width ℓ. This decomposition can be constructed in $O(n^3)$ steps.

Proof ideas

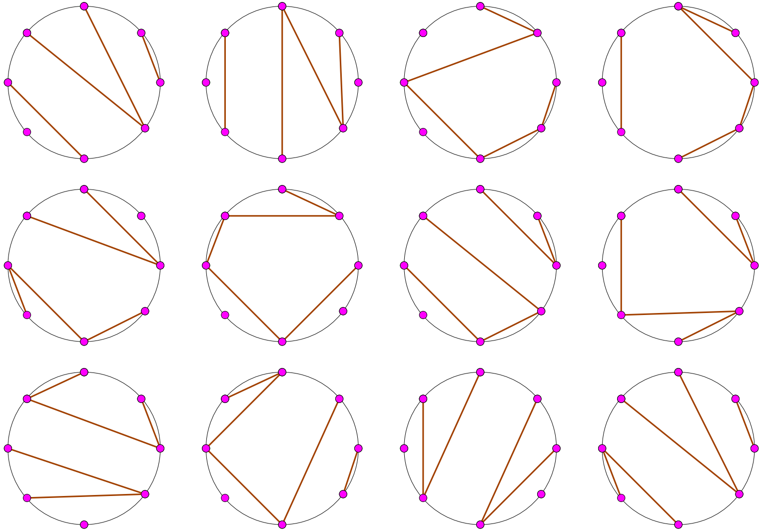
- Carving-width
- Connectivity properties of carving-width decompositions
- Branch-width as the carving width of a radial graph

Bounding $(B_e^0, B_e^1, B_e^2, \texttt{pairs}(e))$

We now have that

1: the vertices of $\mathrm{mid}(e)$ lay on the boundary of a disk and

2: the pairings $\texttt{pairs}(e))$ cannot be crossing because of planarity.



It follows that $\texttt{pairs}(e) = O(C(|\operatorname{mid}(e)|)) = O(C(w))$

Where $C(w)$ is the w-th *Catalan Number*.

It is known that $C(w) \sim \frac{4^w}{w^{3/2}\sqrt{\pi}} = 2^{O(w)}$

It follows that $\texttt{pairs}(e) = O(C(|\operatorname{mid}(e)|)) = O(C(w))$

Where $C(w)$ is the w-th *Catalan Number*.

It is known that $C(w) \sim \frac{4^w}{w^{3/2}\sqrt{\pi}} = 2^{O(w)}$

We conclude that for planar graphs, there is an optimal branch decomposition of width w such that for every edge e, the amount of tuples $(B_e^0, B_e^1, B_e^2, \texttt{pairs}(e))$ is $2^{O(w)}$

It follows that $\texttt{pairs}(e) = O(C(|\,\mathrm{mid}(e)|)) = O(C(w))$

Where $C(w)$ is the w-th *Catalan Number*.

It is known that $C(w) \sim \frac{4^w}{w^{3/2}\sqrt{\pi}} = 2^{O(w)}$

We conclude that for planar graphs, there is an optimal branch decomposition of width w such that for every edge e, the amount of tuples $(B_e^0, B_e^1, B_e^2, \texttt{pairs}(e))$ is $2^{O(w)}$

Therefore: dynamic programming for finding a k-path of a planar graph G on a sphere cut decompositions of G with width $\leq w$ takes $O(2^{O(w)} \cdot n)$ steps.

Conclusion:

Planar k-Path can be solved in $O(2^{O(\sqrt{k})} \cdot nn^{O(1)})$ steps

Algorithm:

- Compute branch-width of G in polynomial time. If $\mathbf{bw}(G) \geq 4\sqrt{k}$, then G contains $\sqrt{k} \times \sqrt{k}$ gris as a minor. Thus it also contains a path of length k
- If $\mathbf{bw}(G) \leq 4\sqrt{k}$, use sphere-cut decomposition to find in time $2^{O(\sqrt{k})}n$ if G has a path of length k.

- Similar results hold for several other problems where an analogue of $\texttt{pairs}(e)$ can be defined for controlling the size of the tables in dynamic programming.

- Similar results hold for several other problems where an analogue of `pairs`(e) can be defined for controlling the size of the tables in dynamic programming.

- Like that one can design $2^{O(\sqrt{k})} \cdot n^{O(1)}$ step algorithms for parameterized planar versions of Cycle Cover, Path Cover, Longest Cycle, Connected Dominating Set, Feedback Vertex Set, Graph Metric TSP, Max Leaf Tree, etc.

Planarity not a limit

- For H-minor free graphs, one can construct an algorithm that solves the k-$\textsc{Path}$ problem in $2^{O(\sqrt{k})} \cdot n^{O(1)}$ steps.

More grids

Grids for other problems or walls

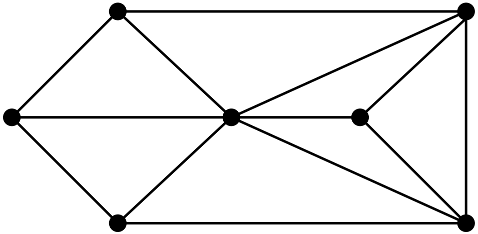
t-spanners

Definition (t-spanner)

Let t be a positive integer. A subgraph S of G, such that $V(S) = V(G)$, is called a t-spanner, if $\mathrm{dist}_S(u,v) \leq t \cdot \mathrm{dist}_G(u,v)$ for every pair of vertices u and v. The parameter t is called the *stretch factor* of S.

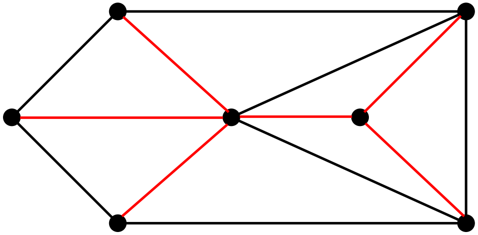
Examples of spanners

3 and 2-spanners



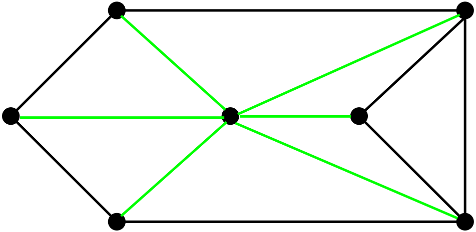
Examples of spanners

3 and 2-spanners



Examples of spanners

3 and 2-spanners



Spanners of bounded branchwidth

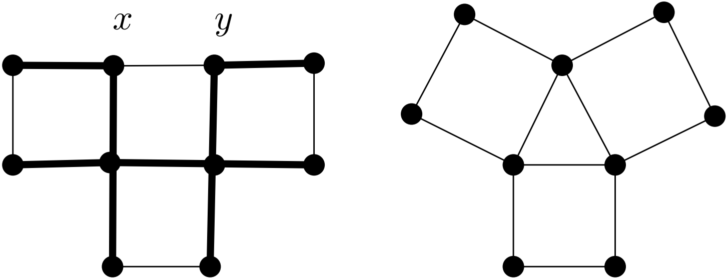
Problem (k-Branchwidth t-spanner)

Instance: A connected graph G and positive integers k and t.

Question: Is there a t-spanner of G of branchwidth at most k?

Topological minors

Contracting edge can increase t

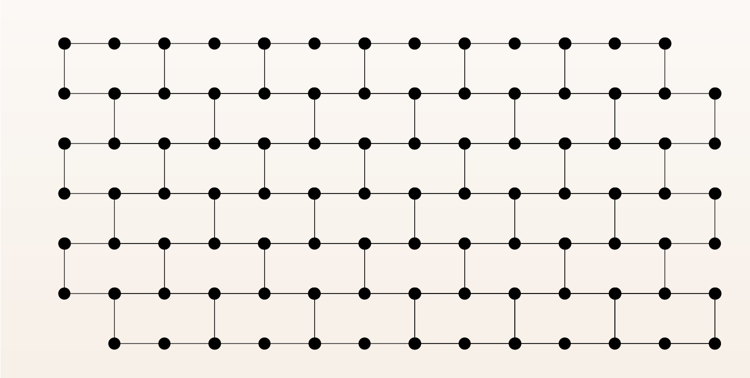


Planar graphs

Theorem (Bounds for planar graphs)

Let G be a planar graph of branchwidth k and let S be a t-spanner of G. Then the branchwidth of S is $\Omega(k/t)$.

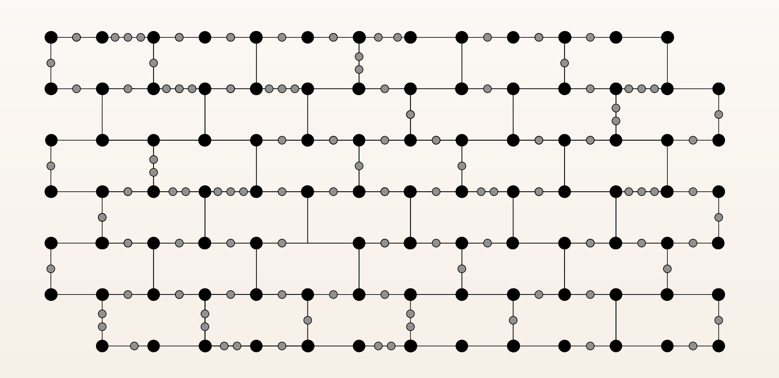
Walls



A wall of height k.

Walls

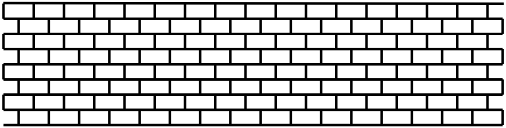
Observation: If a graph contains a $k \times k$ grid as a minor, it contains a wall of height k as a topological subgraph.



Which can be seen as a subgraph that is a subdivided wall.

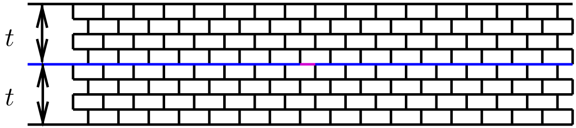
Sketch of the proof

Walls and grids



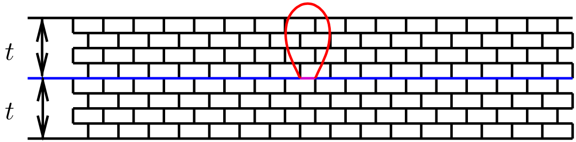
Sketch of the proof

Walls and grids



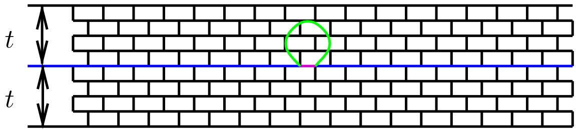
Sketch of the proof

Walls and grids



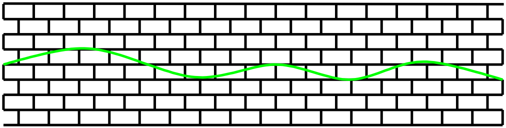
Sketch of the proof

Walls and grids



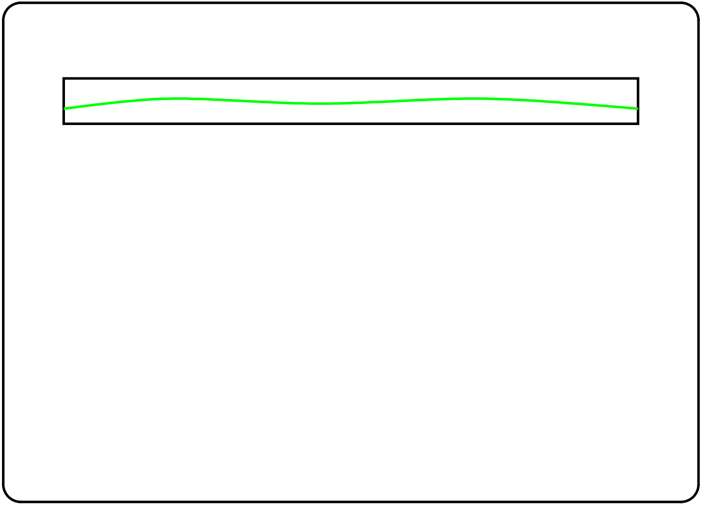
Sketch of the proof

Walls and grids



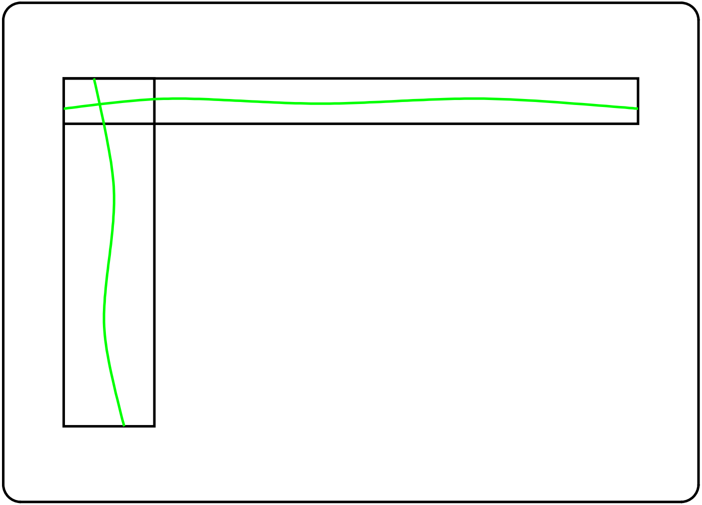
Sketch of the proof

Walls and grids



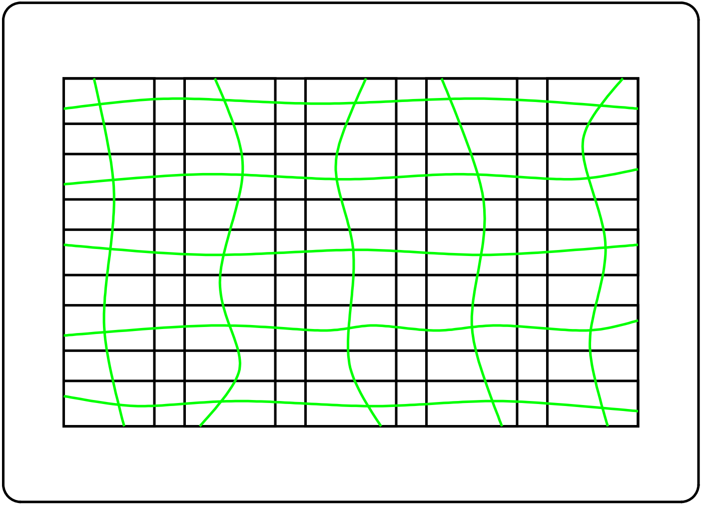
Sketch of the proof

Walls and grids



Sketch of the proof

Walls and grids



Algorithmic consequences

Theorem (Dragan, FF, Golovach, 2008)

Deciding if a planar graph G *has a* t*-spanner of treewidth at most* k *is solvable in time* $O(f(k,t) \cdot n^{O(1)})$.

Disjoint Path Problem

p-Disjoint Paths Problem

Instance: a graph G and terminals $(s_i, t_i) \in V(G)^2, i = 1, \ldots, k$.

Parameter: k

Question: Does G have k disjoint paths between s_i and t_i, $i = 1, \ldots, k$?

Disjoint Path Problem

- To solve problem on planar graphs: If the branch-width is small, do DP

Disjoint Path Problem

- To solve problem on planar graphs: If the branch-width is small, do DP
- If the branch-widht is large, there is a wall of large height

Disjoint Path Problem

- To solve problem on planar graphs: If the branch-width is small, do DP
- If the branch-widht is large, there is a wall of large height
- If there is a wall of large height, there is a wall of large height containing no terminal vertices

Disjoint Path Problem

- To solve problem on planar graphs: If the branch-width is small, do DP
- If the branch-widht is large, there is a wall of large height
- If there is a wall of large height, there is a wall of large height containing no terminal vertices
- A vertex in the center of this wall is irrelevant. i.e. its removal does not change the problem

Conclusion

- WIN/WIN approach via Excluding Grid Theorem
- Dynamic programming and Catalan structures
- Working on planar problems don't afraid to hit the wall!

Further reading. Subexponential algorithms and bidimensionality

- J. Alber, H. L. Bodlaender, H. Fernau, T. Kloks, and R. Niedermeier, *Fixed parameter algorithms for dominating set and related problems on planar graphs*, Algorithmica, 33 (2002), pp. 461–493.
- E. D. Demaine, F. V. Fomin, M. Hajiaghayi, and D. M. Thilikos, *Subexponential parameterized algorithms on graphs of bounded genus and H-minor-free graphs*, Journal of the ACM, 52 (2005), pp. 866–893.

Further reading. Catalan structures and dynamic programming

- F. Dorn, E. Penninkx, H. Bodlaender, and F. V. Fomin, *Efficient exact algorithms on planar graphs: Exploiting sphere cut branch decompositions*, Proceedings of the 13th Annual European Symposium on Algorithms (ESA 2005), vol. 3669 of LNCS, Springer, 2005, pp. 95–106.
- F. Dorn, F. V. Fomin, and D. M. Thilikos, *Catalan structures and dynamic programming on H-minor-free graphs*, in Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2008), ACM-SIAM, pp. 631–640.

Further reading. Surveys

- E. Demaine and M. Hajiaghayi, *The bidimensionality theory and its algorithmic applications*, The Computer Journal, (2007), pp. 332–337.
- F. Dorn, F. V. Fomin, and D. M. Thilikos, *Subexponential parameterized algorithms*, Computer Science Review, 2 (2008), pp. 29–39.