Subexponential parameterized algorithms on planar graphs via low-treewidth pattern covering

Dániel Marx

(joint work with Fedor Fomin,
Daniel Lokshtanov, Marcin Pilipczuk,
Michał Pilipczuk, and Saket Saurabh)

Institute for Computer Science and Control,
Hungarian Academy of Sciences (MTA SZTAKI)
Budapest, Hungary

Dagstuhl Seminar 17041
Schloss Dagstuhl, Germany
January 26, 2017

Square root phenomenon

Most NP-hard problems (e.g., 3-Coloring, Independent Set, Hamiltonian Cycle, Steiner Tree, etc.) remain NP-hard on planar graphs,[1] but often get easier on planar graphs in the sense that the running time is still exponential, but significantly smaller:

$$\begin{aligned} 2^{O(n)} &\Rightarrow 2^{O(\sqrt{n})} \\ n^{O(k)} &\Rightarrow n^{O(\sqrt{k})} \\ 2^{O(k)} \cdot n^{O(1)} &\Rightarrow 2^{O(\sqrt{k})} \cdot n^{O(1)} \end{aligned}$$

This talk: a new technique for such algorithms.

[1] Notable exception: Max Cut is in P for planar graphs.

Subgraph Isomorphism

Subgraph Isomorphism

Input: Graphs H and G

Decide: Does G has a subgraph isomorphic to H?

Standard dynamic programming:

Fact

If connected graph H has k vertices and maximum degree Δ, G has treewidth w, then Subgraph Isomorphism can be solved

- in time $2^{O(k)} \cdot k^{O(w)} \cdot n$ or
- in time $k^{O(\Delta w)} \cdot n$.

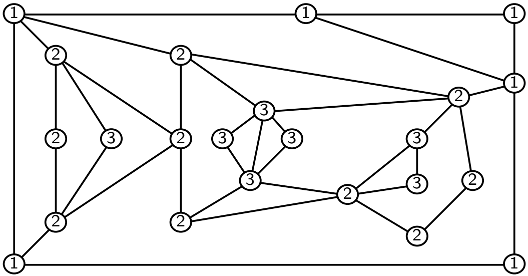
Remark: Robust algorithm, can be easily generalized to colored, directed, weighted etc. versions.

k-outerplanar graphs

Given a planar embedding, we can define **layers** by iteratively removing the vertices on the infinite face.

Definition

A planar graph is k-**outerplanar** if it has a planar embedding having at most k layers.



Fact

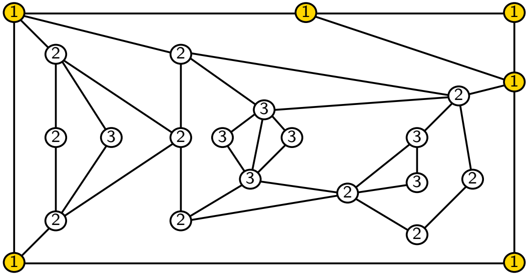
Every k-outerplanar graph has treewidth at most $3k + 1$.

k-outerplanar graphs

Given a planar embedding, we can define **layers** by iteratively removing the vertices on the infinite face.

Definition

A planar graph is k-**outerplanar** if it has a planar embedding having at most k layers.



Fact

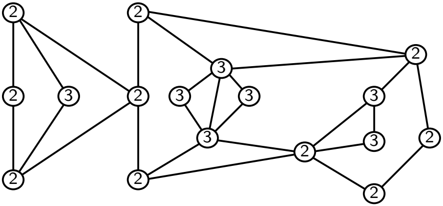
Every k-outerplanar graph has treewidth at most $3k+1$.

k-outerplanar graphs

Given a planar embedding, we can define **layers** by iteratively removing the vertices on the infinite face.

Definition

A planar graph is k-**outerplanar** if it has a planar embedding having at most k layers.



Fact

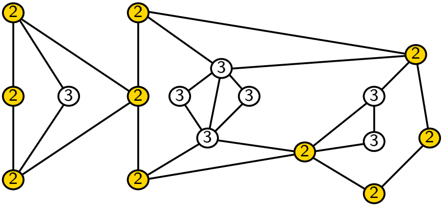
Every k-outerplanar graph has treewidth at most $3k + 1$.

k-outerplanar graphs

Given a planar embedding, we can define **layers** by iteratively removing the vertices on the infinite face.

Definition

A planar graph is k-**outerplanar** if it has a planar embedding having at most k layers.



Fact

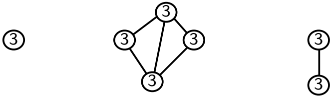
Every k-outerplanar graph has treewidth at most $3k+1$.

k-outerplanar graphs

Given a planar embedding, we can define **layers** by iteratively removing the vertices on the infinite face.

Definition

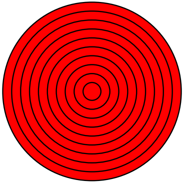
A planar graph is k-**outerplanar** if it has a planar embedding having at most k layers.



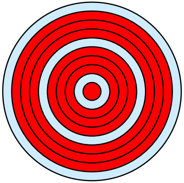
Fact

Every k-outerplanar graph has treewidth at most $3k + 1$.

Baker's shifting strategy

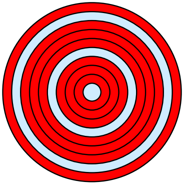


Baker's shifting strategy



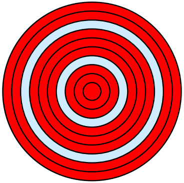
- For a fixed $0 \leq s < k+1$, delete every layer L_i with $i = s \pmod{k+1}$

Baker's shifting strategy



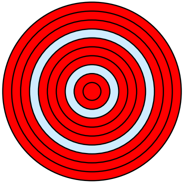
- For a fixed $0 \leq s < k+1$, delete every layer L_i with $i = s \pmod{k+1}$

Baker's shifting strategy



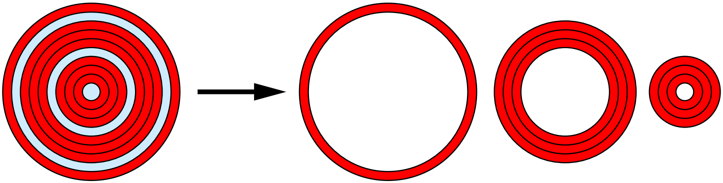
- For a fixed $0 \leq s < k+1$, delete every layer L_i with $i = s \pmod{k+1}$

Baker's shifting strategy



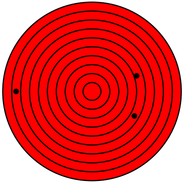
- For a fixed $0 \leq s < k+1$, delete every layer L_i with $i = s \pmod{k+1}$

Baker's shifting strategy



- For a fixed $0 \leq s < k+1$, delete every layer L_i with $i = s \pmod{k+1}$
- The resulting graph is k-outerplanar, hence it has treewidth at most $w := 3k+1$.
- Using the $2^{O(k)} \cdot k^{O(w)} \cdot n$ time algorithm for Subgraph Isomorphism, the problem can be solved in time $k^{O(k)} \cdot n = 2^{O(k \log k)} \cdot n$.

Baker's shifting strategy

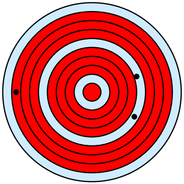


We do this for every $0 \leq s < k+1$:
for at least one value of s, we do not delete
any of the k vertices of the solution

$\Downarrow$

We find a copy of H in G if there is one.

Baker's shifting strategy

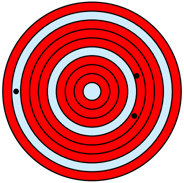


We do this for every $0 \leq s < k+1$:
for at least one value of s, we do not delete
any of the k vertices of the solution

$\Downarrow$

We find a copy of H in G if there is one.

Baker's shifting strategy

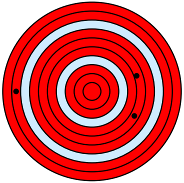


We do this for every $0 \leq s < k+1$:
for at least one value of s, we do not delete
any of the k vertices of the solution

$\Downarrow$

We find a copy of H in G if there is one.

Baker's shifting strategy

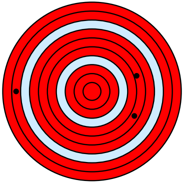


We do this for every $0 \leq s < k+1$:
for at least one value of s, we do not delete
any of the k vertices of the solution

$\Downarrow$

We find a copy of H in G if there is one.

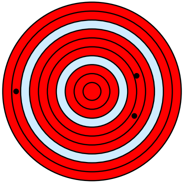
Baker's shifting strategy



Theorem

Subgraph Isomorphism for planar graphs can be solved in time $2^{O(k \log k)} \cdot n$ for $k := |V(H)|$.

Baker's shifting strategy



Theorem

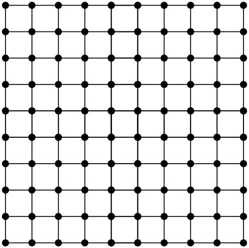
Subgraph Isomorphism for planar graphs can be solved in time $2^{O(k \log k)} \cdot n$ for $k := |V(H)|$.

Next: Improved algorithm for the special case k-Path via bidimensionality.

Planar Excluded Grid Theorem

Theorem [Robertson, Seymour, Thomas 1994]

Every planar graph with treewidth at least $5k$ has a $k \times k$ grid minor.

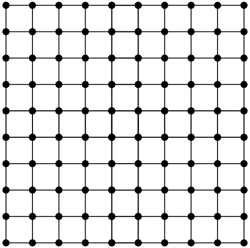


Note: for general graphs, treewidth at least k^{19} or so guarantees a $k \times k$ grid minor!

Planar Excluded Grid Theorem

Theorem [Robertson, Seymour, Thomas 1994]

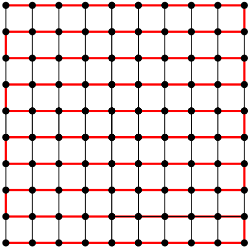
Every planar graph with treewidth at least $5k$ has a $k \times k$ grid minor.



Consequence: every n-vertex planar graph has treewidth $O(\sqrt{n})$.

Subexponential algorithm for k-Path

Observation: If the treewidth of a planar graph G is at least $5\sqrt{k}$
$\Rightarrow$ It has a $\sqrt{k} \times \sqrt{k}$ grid minor (Planar Excluded Grid Theorem)
$\Rightarrow$ The grid has a path of length at least k.
$\Rightarrow$ G has a path of length at least k.

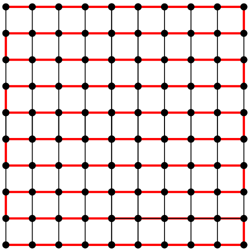


Subexponential algorithm for k-Path

Observation: If the treewidth of a planar graph G is at least $5\sqrt{k}$
$\Rightarrow$ It has a $\sqrt{k} \times \sqrt{k}$ grid minor (Planar Excluded Grid Theorem)
$\Rightarrow$ The grid has a path of length at least k.
$\Rightarrow$ G has a path of length at least k.

We use this observation to find a path of length at least k on planar graphs:

- Set $w := 5\sqrt{k}$.
- Find an $O(1)$-approximate tree decomposition.
  - If treewidth is at least w: we answer "there is a path of length at least k."
  - If we get a tree decomposition of width $O(w)$, then we can solve the problem in time $k^{O(\Delta w)} \cdot n^{O(1)} = 2^{O(\sqrt{k}\log k)} \cdot n^{O(1)}$.



Lower bounds based on ETH

Lower bound technology introduced by Impagliazzo, Paturi, and Zane:

Exponential-Time Hypothesis

There is no $2^{o(n)}$-time algorithm for n-variable $\mathrm{3SAT}$.

Lower bounds based on ETH

Lower bound technology introduced by Impagliazzo, Paturi, and Zane:

Exponential-Time Hypothesis + Sparsification Lemma

There is no $2^{o(n+m)}$-time algorithm for n-variable m-clause $\mathrm{3SAT}$.

Lower bounds based on ETH

Lower bound technology introduced by Impagliazzo, Paturi, and Zane:

Exponential-Time Hypothesis + Sparsification Lemma

There is no $2^{o(n+m)}$-time algorithm for n-variable m-clause $\mathrm{3SAT}$.

Textbook reduction from 3SAT to PLANAR HAMILTONIAN PATH:

3SAT formula ϕ		Planar graph G'
n variables	$\Rightarrow$	$O((n+m)^2)$ vertices
m clauses		$O((n+m)^2)$ edges

Lower bounds based on ETH

Lower bound technology introduced by Impagliazzo, Paturi, and Zane:

Exponential-Time Hypothesis + Sparsification Lemma

There is no $2^{o(n+m)}$-time algorithm for n-variable m-clause 3SAT.

Textbook reduction from 3SAT to Planar Hamiltonian Path:

3SAT formula ϕ		Planar graph G'
n variables	$\Rightarrow$	$O((n+m)^2)$ vertices
m clauses		$O((n+m)^2)$ edges

Corollary

Assuming ETH, there is no $2^{o(\sqrt{n})}$ algorithm for Planar Hamiltonian Path on an n-vertex planar graph G.

Lower bounds based on ETH

Lower bound technology introduced by Impagliazzo, Paturi, and Zane:

Exponential-Time Hypothesis + Sparsification Lemma

There is no $2^{o(n+m)}$-time algorithm for n-variable m-clause 3SAT.

Textbook reduction from 3SAT to Planar Hamiltonian Path:

3SAT formula ϕ		Planar graph G'
n variables	$\Rightarrow$	$O((n+m)^2)$ vertices
m clauses		$O((n+m)^2)$ edges

Corollary

Assuming ETH, there is no $2^{o(\sqrt{k})} \cdot n^{O(1)}$ algorithm for k-Path on an n-vertex planar graph G.

Our $2^{O(\sqrt{k}\log k)} \cdot n^{O(1)}$ algorithm is essentially best possible.

Other problems:

Good news:

- Same algorithm works for finding a cycle of length at least k.

Other problems:

Good news:

- Same algorithm works for finding a cycle of length at least k.

Bad news:

- Does not work for finding a cycle of length **exactly** k.
- Does not work for finding an $s - t$ path of length at least/exactly k.
- Does not work for finding a minimum weight k-path.
- Does not work for finding a directed k-path.
- ...

Main combinatorial result

Theorem

There is a randomized polynomial-time algorithm that, given a planar graph G and an integer k, computes an induced subgraph G' such that

1. G' has treewidth $O(\sqrt{k} \cdot \mathrm{polylog}(k))$ and
2. for any **connected** subgraph $H \subseteq G$ with at most k vertices, we have $H \subseteq G'$ with probability at least $(2^{O(\sqrt{k} \cdot \mathrm{polylog}(k))} \cdot n^{O(1)})^{-1}$.

Main combinatorial result

Theorem

There is a randomized polynomial-time algorithm that, given a planar graph G and an integer k, computes an induced subgraph G' such that

1. G' has treewidth $O(\sqrt{k} \cdot \text{polylog}(k))$ and
2. for any **connected** subgraph $H \subseteq G$ with at most k vertices, we have $H \subseteq G'$ with probability at least $(2^{O(\sqrt{k} \cdot \text{polylog}(k))} \cdot n^{O(1)})^{-1}$.

Thus the Subgraph Isomorphism problem for connected H can be solved by restriction to G'.

Theorem

Subgraph Isomorphism for planar graphs can be solved in time $2^{O(\Delta\sqrt{k} \cdot \text{polylog}(k))} \cdot n^{O(1)}$ if H is connected with maximum degree Δ.

Remark: Robust algorithm, can be easily generalized to colored, directed, weighted etc. versions.

Main combinatorial result

Theorem

There is a randomized polynomial-time algorithm that, given a planar graph G and an integer k, computes an induced subgraph G' such that

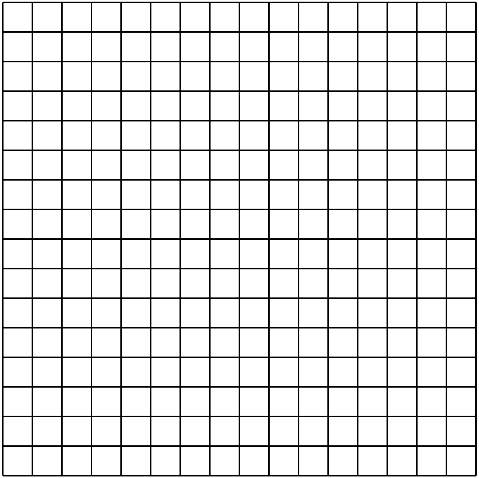
1. G' has treewidth $O(\sqrt{k} \cdot \mathrm{polylog}(k))$ and
2. for any **connected** subgraph $H \subseteq G$ with at most k vertices, we have $H \subseteq G'$ with probability at least $(2^{O(\sqrt{k} \cdot \mathrm{polylog}(k))} \cdot n^{O(1)})^{-1}$.

Theorem [Bodlaender, Nederlof, van der Zanden 2016]

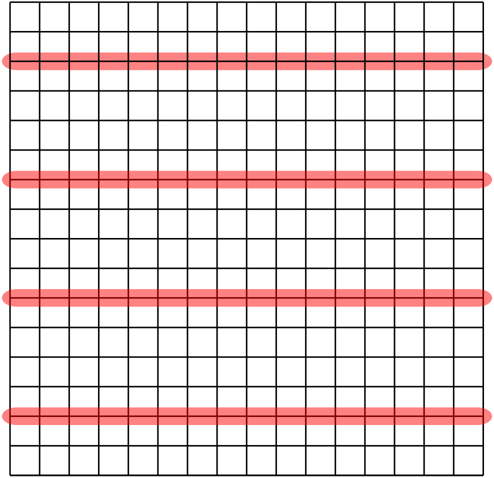
Assuming ETH, there is no $2^{o(k/\log k)} \cdot n^{O(1)}$ time algorithm for planar Subgraph Isomorphism, even when

- H is a forest of maximum degree 3, or
- H is a tree with only one vertex having degree larger than 3.

Example: grids

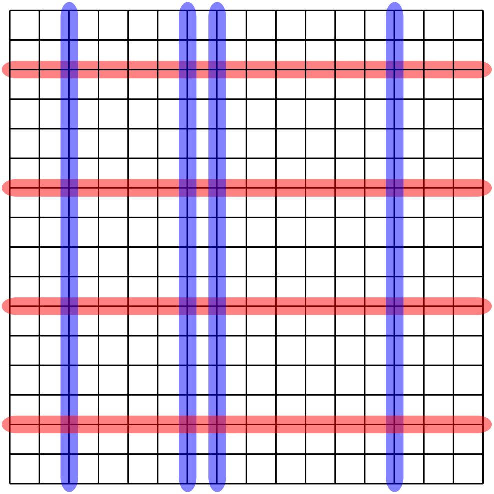


Example: grids



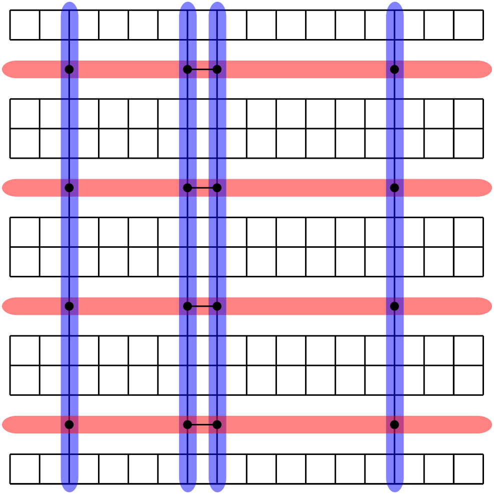
1. Guess an index $0 \leq i < \sqrt{k}$ such that rows $i \mod \sqrt{k}$ contain a total of $\sqrt{k}$ vertices of H.

Example: grids



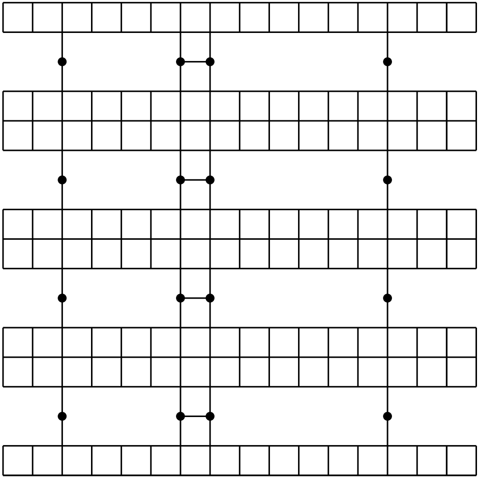
1. Guess an index $0 \leq i < \sqrt{k}$ such that rows $i \mod \sqrt{k}$ contain a total of $\sqrt{k}$ vertices of H.
2. Guess the at most $\sqrt{k}$ columns where H appears in these rows.

Example: grids



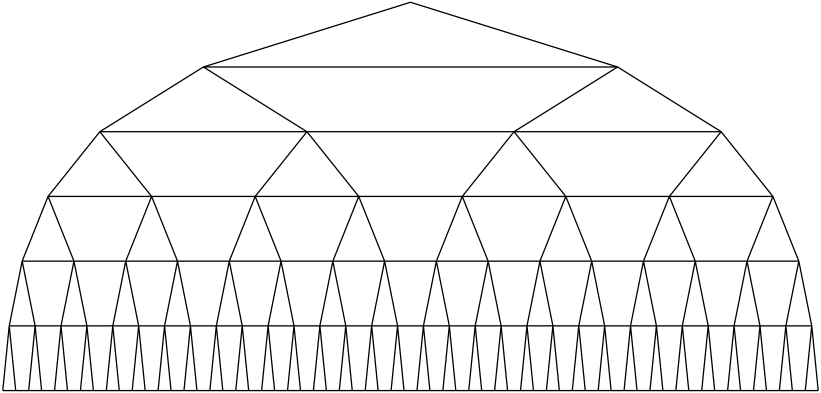
1. Guess an index $0 \le i < \sqrt{k}$ such that rows $i \mod \sqrt{k}$ contain a total of $\sqrt{k}$ vertices of H.
2. Guess the at most $\sqrt{k}$ columns where H appears in these rows.
3. Graph falls apart into $\sqrt{k}$-tall grids connected by $\sqrt{k}$ vertices.

Example: grids

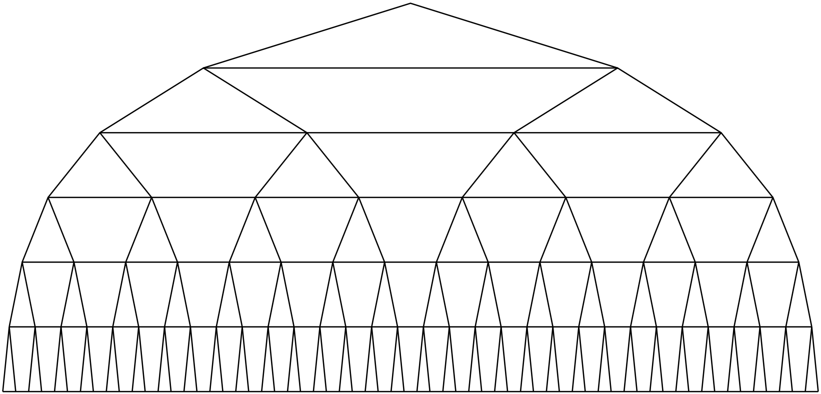


1. Guess an index $0 \leq i < \sqrt{k}$ such that rows $i \mod \sqrt{k}$ contain a total of $\sqrt{k}$ vertices of H.
2. Guess the at most $\sqrt{k}$ columns where H appears in these rows.
3. Graph falls apart into $\sqrt{k}$-tall grids connected by $\sqrt{k}$ vertices.

Example: complete binary tree



Example: complete binary tree



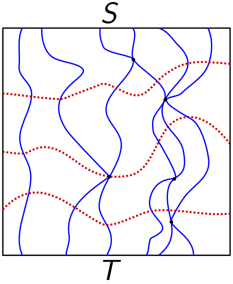
Ball-growing argument: there is an index $\sqrt{k} \leq i \leq O(\sqrt{k} \log k)$ such that the first i rows in total contain $\sqrt{k}$ times more vertices of the solution than row i.

Duality result

Theorem

Given a graph G and two sets of vertices S and T there is either

- a family $P_1, \ldots, P_C$ of "almost-disjoint" $S-T$ paths such that $\exists A_i \subseteq P_i$ with $\sum |A_i| \leq \ell C$ and $P_i \setminus A_i$'s are pairwise disjoint or
- a family $S_1, \ldots, S_\ell$ of disjoint "small" $S-T$ separators with $|S_i| \leq C$.

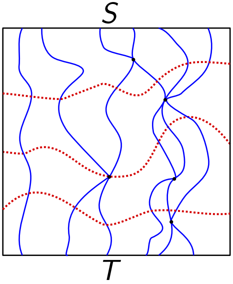


Duality result

Theorem

Given a graph G and two sets of vertices S and T there is either

- a family $P_1, \ldots, P_{\boldsymbol{C/2}}$ of "almost-disjoint" $S - T$ paths such that $\exists A_i \subseteq P_i$ with $|\boldsymbol{A_i}| \leq \boldsymbol{\ell}$ and $P_i \setminus A_i$'s are pairwise disjoint or
- a family $S_1, \ldots, S_\ell$ of disjoint "small" $S - T$ separators with $|S_i| \leq C$.

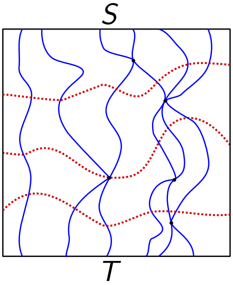


Duality result

Theorem

Given a graph G and two sets of vertices S and T there is either

- a family $P_1, \ldots, P_{\boldsymbol{C/2}}$ of "almost-disjoint" $S - T$ paths such that $\exists A_i \subseteq P_i$ with $\boldsymbol{|A_i| \leq \ell}$ and $P_i \setminus A_i$'s are pairwise disjoint or
- a family $S_1, \ldots, S_\ell$ of disjoint "small" $S - T$ separators with $|S_i| \leq C$.



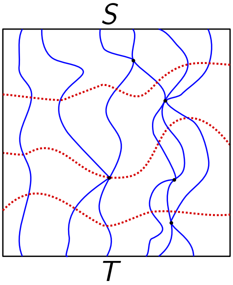
We will use the duality with $C = 2(k+1)$ and $\ell = \sqrt{k}$

Duality result

Theorem

Given a graph G and two sets of vertices S and T there is either

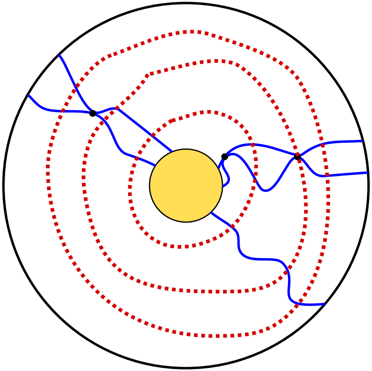
- a family $P_1, \ldots, P_{k+1}$ of "almost-disjoint" $S-T$ paths such that $\exists A_i \subseteq P_i$ with $|A_i| \leq \sqrt{k}$ and $P_i \setminus A_i$'s are pairwise disjoint or
- a family $S_1, \ldots, S_{\sqrt{k}}$ of disjoint "small" $S-T$ seps. with $|S_i| \leq 2k+2$.



We will use the duality with $C = 2(k+1)$ and $\ell = \sqrt{k}$

Using duality

The correct viewpoint:
Using the duality between the outside and the inside.



But where is this “inside”?

Conclusions

- Subexponential parameterized algorithms for finding bounded-degree connected subgraphs.
- Connectedness of the pattern H seems essential (but easy to generalize to bounded number of connected components).
- Can be generalized to bounded local treewidth, H-minor-free in progress.
- Other classes of graphs: polynomial growth property.