Obstructions to bounded cutwidth

Jean-Florent Raymond

LIRMM and University of Warsaw

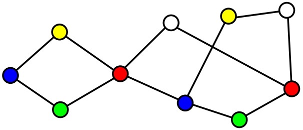
Thursday 13th April, 2017

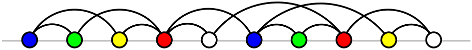
Joint work with:

- Archontia Giannopoulou (TU Berlin);
- Michał Pilipczuk (University of Warsaw);
- Dimitrios M. Thilikos (LIRMM–CNRS); and
- Marcin Wrochna (University of Warsaw).

Cutwidth

- *layout* of G: ordering of $V(G)$.

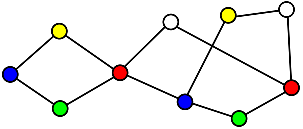


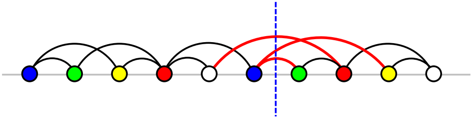


Cutwidth

- *layout* of G: ordering of $V(G)$.
- the *width* of a layout σ:

$$\max_{i=1,2,\ldots,n-1} \#(\text{edges with one endpoint} \leqslant i \text{ and second} > i).$$





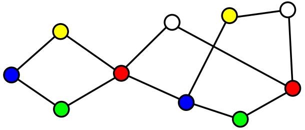
Cutwidth

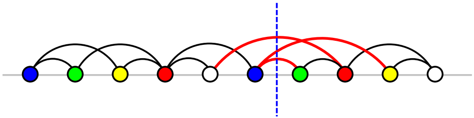
- *layout* of G: ordering of $V(G)$.
- the *width* of a layout σ:

$$\max_{i=1,2,\dots,n-1} \#(\text{edges with one endpoint} \leqslant i \text{ and second} > i).$$

- the *cutwidth* of G:

$$\mathbf{ctw}(G) = \min \{\text{width}(\sigma) : \sigma \text{ is a layout of } G\}.$$





Cutwidth

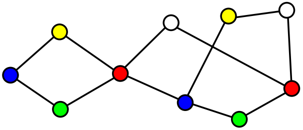
- *layout* of G: ordering of $V(G)$.
- the *width* of a layout σ:

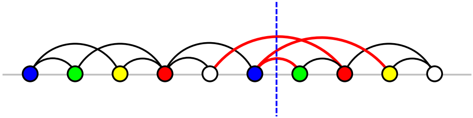
$$\max_{i=1,2,\dots,n-1} \#(\text{edges with one endpoint} \leqslant i \text{ and second} > i).$$

- the *cutwidth* of G:

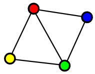
$$\mathbf{ctw}(G) = \min\{\text{width}(\sigma) : \sigma \text{ is a layout of } G\}.$$

Class considered of this talk: graphs of cutwidth $\leqslant k$.

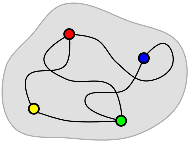




Immersions

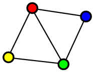


$\leqslant_{\mathrm{imm}}$

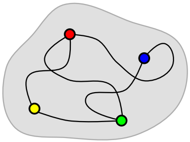


$$\begin{cases} \text{vertices} & \mapsto & \text{distinct vertices} \\ \text{edges} & \mapsto & \text{edge-disjoint paths} \end{cases}$$

Immersions



$\leqslant_{\mathrm{imm}}$

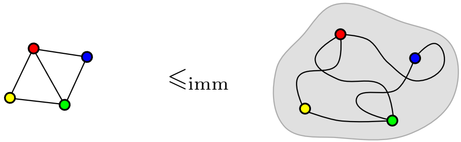


$$\begin{cases} \text{vertices} & \mapsto & \text{distinct vertices} \\ \text{edges} & \mapsto & \text{edge-disjoint paths} \end{cases}$$

“Monotonicity” of cutwidth:

$$H \leqslant_{\mathrm{imm}} G \Rightarrow \mathbf{ctw}(H) \leqslant \mathbf{ctw}(G).$$

Immersions



$$\begin{cases} \text{vertices} & \mapsto & \text{distinct vertices} \\ \text{edges} & \mapsto & \text{edge-disjoint paths} \end{cases}$$

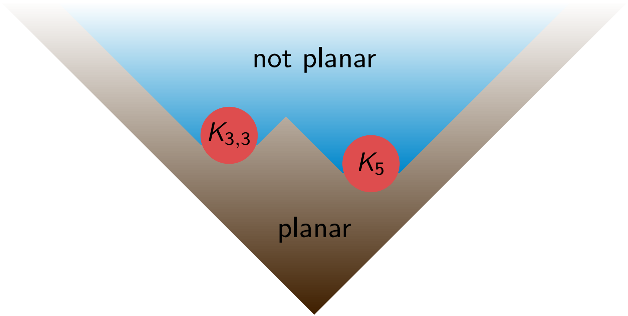
"Monotonicity" of cutwidth:

$$H \leqslant_{\text{imm}} G \Rightarrow \textbf{ctw}(H) \leqslant \textbf{ctw}(G).$$

$\{G,\ \textbf{ctw}(G) \leqslant k\}$ is immersion-closed.

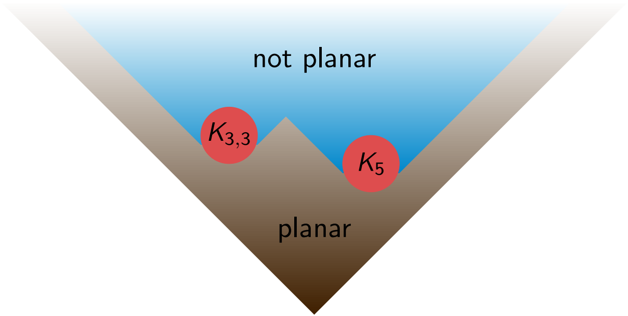
Obstructions

K_5 and $K_{3,3}$ are the *minor-obstructions* of planar graphs



Obstructions

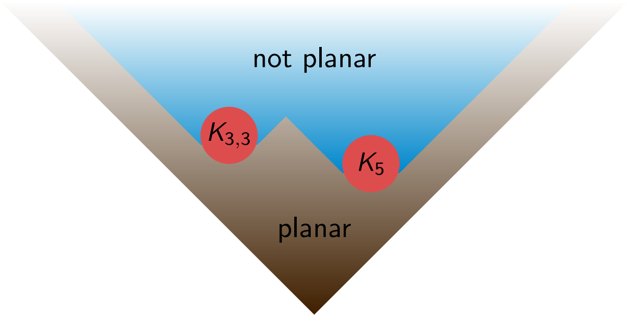
K_5 and $K_{3,3}$ are the *minor-obstructions* of planar graphs



obstruction for a class = minimal element of its complementary

Obstructions

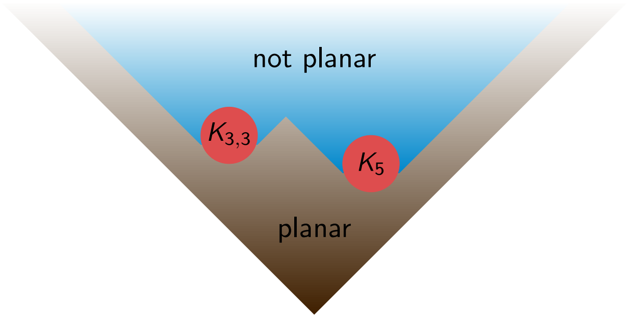
K_5 and $K_{3,3}$ are the *minor-obstructions* of planar graphs



obstruction for a class = minimal element of its complementary
(depends on the class and on the order)

Obstructions

K_5 and $K_{3,3}$ are the *minor-obstructions* of planar graphs



obstruction for a class = minimal element of its complementary
(depends on the class and on the order)

The set of obstructions may be infinite!

Obstructions for bounded cutwidth

What about obstructions of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$?

Theorem (Robertson and Seymour, consequence of GM.XXII)

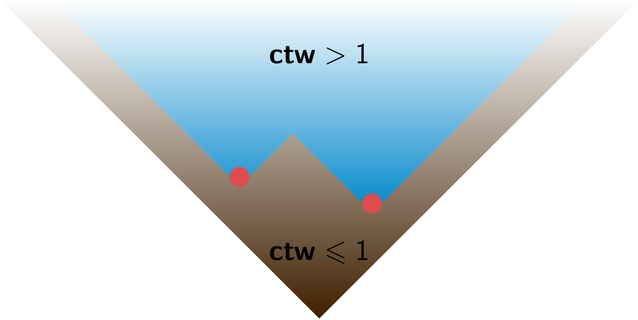
For every $k \in \mathbb{N}$, $\{G,\ \mathbf{ctw}(G) \leqslant k\}$, has finitely many obstructions.

Obstructions for bounded cutwidth

What about obstructions of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$?

Theorem (Robertson and Seymour, consequence of GM.XXII)

For every $k \in \mathbb{N}$, $\{G,\ \mathbf{ctw}(G) \leqslant k\}$, *has finitely many obstructions.*

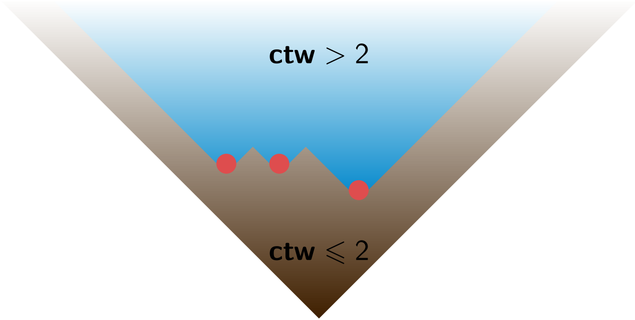


Obstructions for bounded cutwidth

What about obstructions of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$?

Theorem (Robertson and Seymour, consequence of GM.XXII)

For every $k \in \mathbb{N}$, $\{G,\ \mathbf{ctw}(G) \leqslant k\}$, has finitely many obstructions.

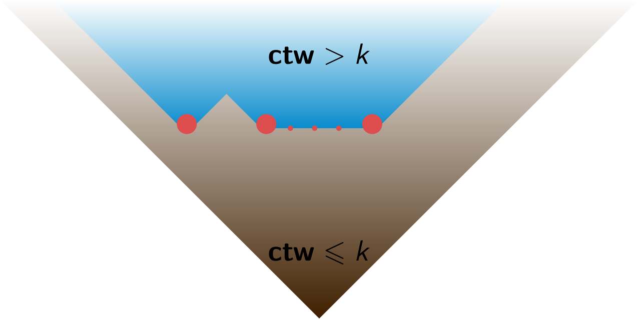


Obstructions for bounded cutwidth

What about obstructions of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$?

Theorem (Robertson and Seymour, consequence of GM.XXII)

For every $k \in \mathbb{N}$, $\{G,\ \mathbf{ctw}(G) \leqslant k\}$, has finitely many obstructions.

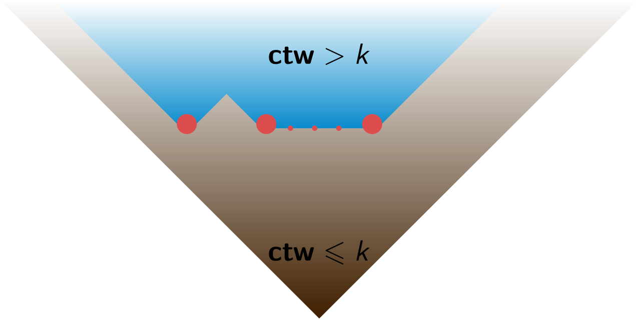


Obstructions for bounded cutwidth

What about obstructions of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$?

Theorem (Robertson and Seymour, consequence of GM.XXII)

For every $k \in \mathbb{N}$, $\{G,\ \mathbf{ctw}(G) \leqslant k\}$, has finitely many obstructions.



How many?

Bounding the size of the obstructions

$s_k :=$ max size of an obstruction for cutwidth $\leqslant k$

(immersion-min. graph with **ctw** $> k$)

Bounding the size of the obstructions

$s_k :=$ max size of an obstruction for cutwidth $\leqslant k$

(immersion-min. graph with **ctw** $> k$)

Govindan and Ramachandramurthi, 2001

$$\frac{1}{2}(3^{k-5} - 1) \quad \overset{\downarrow}{\leqslant} \quad s_k$$

Bounding the size of the obstructions

$s_k :=$ max size of an obstruction for cutwidth $\leqslant k$

(immersion-min. graph with **ctw** $> k$)

Govindan and Ramachandramurthi, 2001

$$\frac{1}{2}(3^{k-5} - 1) \quad \leqslant \quad s_k \quad = \quad 2^{O(k^3 \log k)}$$

Giannopoulou, Pilipczuk, R., Thilikos, Wrochna, 2016

Bounding the size of the obstructions

$s_k :=$ max size of an obstruction for cutwidth $\leqslant k$

(immersion-min. graph with **ctw** $> k$)

Govindan and Ramachandramurthi, 2001

$$\frac{1}{2}(3^{k-5} - 1) \quad \leqslant \quad s_k \quad = \quad 2^{O(k^3 \log k)}$$

Giannopoulou, Pilipczuk, R., Thilikos, Wrochna, 2016

Results of Lagergren (1998):

- G minor-obstruction for pathwidth $\leqslant k \Rightarrow |G| = 2^{O(k^4)}$;
- G minor-obstruction for treewidth $\leqslant k \Rightarrow |G| = 2^{2^{O(k^5)}}$.

How to show that obstructions are small?

General idea

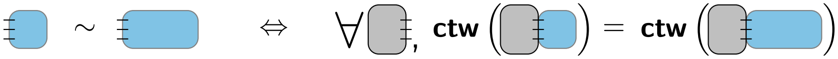
If an obstruction is too large, some part of it is *redundant*.

How to show that obstructions are small?

General idea

If an obstruction is too large, some part of it is *redundant*.

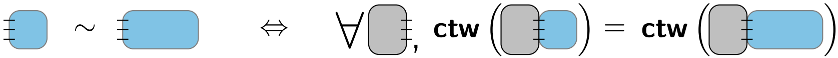
We define an equivalence relation on boundaried subgraphs:



General idea

If an obstruction is too large, some part of it is *redundant*.

We define an equivalence relation on boundaried subgraphs:



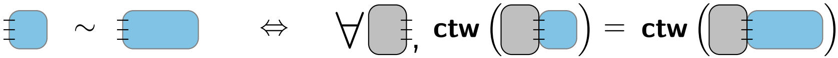
Let G be an obstruction of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$:

How to show that obstructions are small?

General idea

If an obstruction is too large, some part of it is *redundant*.

We define an equivalence relation on boundaried subgraphs:



Let G be an obstruction of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$:

- replace a subgraph with an equivalent one that is smaller;

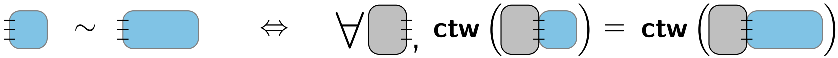
(this does not change the cutwidth)

How to show that obstructions are small?

General idea

If an obstruction is too large, some part of it is *redundant*.

We define an equivalence relation on boundaried subgraphs:



Let G be an obstruction of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$:

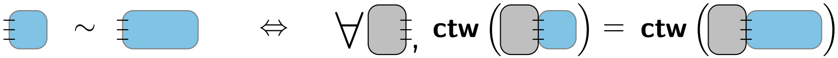
- replace a subgraph with an equivalent one that is smaller;
  (this does not change the cutwidth)
- we prove that the obtained graph is an immersion of G;

How to show that obstructions are small?

General idea

If an obstruction is too large, some part of it is *redundant*.

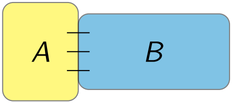
We define an equivalence relation on boundaried subgraphs:

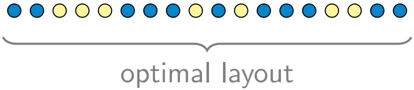


Let G be an obstruction of $\{G,\ \mathbf{ctw}(G) \leqslant k\}$:

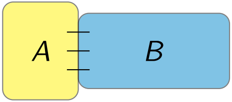
- replace a subgraph with an equivalent one that is smaller;
  (this does not change the cutwidth)
- we prove that the obtained graph is an immersion of G;
- contradicts the minimality of G!

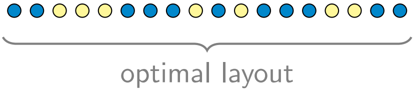
Bounding the number of equivalence classes





Bounding the number of equivalence classes

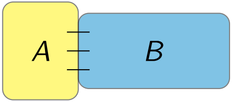


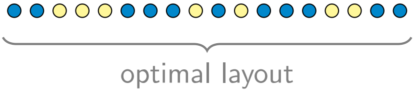


Key Lemma

If $\mathbf{ctw}(G) \leqslant k$ and $|E(A, B)| \leqslant \ell$, then there exists an optimum-width layout of G with $O(k\ell)$ blocks.

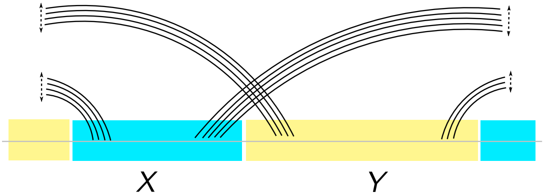
Bounding the number of equivalence classes



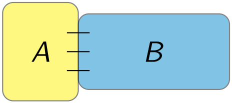


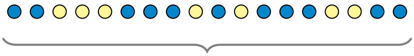
Key Lemma

If **ctw**$(G) \leqslant k$ and $|E(A, B)| \leqslant \ell$, then there exists an optimum-width layout of G with $O(k\ell)$ blocks.



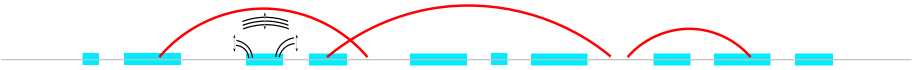
Bounding the number of equivalence classes





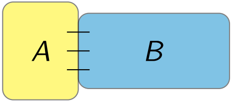
Key Lemma

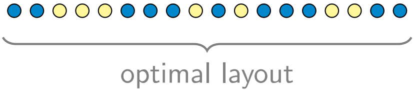
If $\mathbf{ctw}(G) \leqslant k$ and $|E(A, B)| \leqslant \ell$, then there exists an optimum-width layout of G with $O(k\ell)$ blocks.



Relevant information about B: $O(k\ell)$ numbers up to k.

Bounding the number of equivalence classes





Key Lemma

If $\mathbf{ctw}(G) \leqslant k$ and $|E(A, B)| \leqslant \ell$, then there exists an optimum-width layout of G with $O(k\ell)$ blocks.

Relevant information about B: $O(k\ell)$ numbers up to k.

$\rightarrow$ finite number of equivalence classes for fixed k, ℓ.

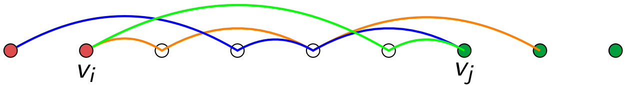
Linked orderings

An ordering $v_1 \ldots, v_n$ of is *linked* if, for every $i < j$ and every t,

Linked orderings

An ordering $v_1 \ldots, v_n$ of is *linked* if, for every $i < j$ and every t,

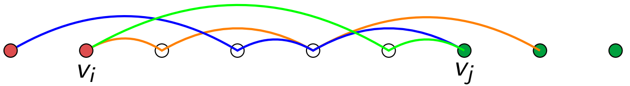
- either there are t edge-disj. paths from $v_1, \ldots, v_i$ to $v_j, \ldots, v_n$;



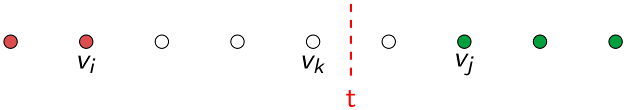
Linked orderings

An ordering $v_1 \ldots, v_n$ of is *linked* if, for every $i < j$ and every t,

- either there are t edge-disj. paths from $v_1, \ldots, v_i$ to $v_j, \ldots, v_n$;



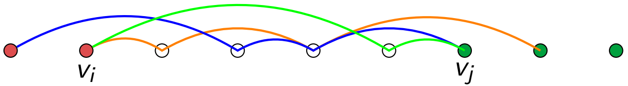
- or $\exists k, i \leqslant k < j$ s.t. $|E(\{v_1, \ldots, v_k\}, \{v_{k+1}, \ldots, v_n\})| < t$.



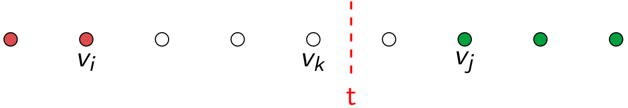
Linked orderings

An ordering $v_1 \ldots, v_n$ of is *linked* if, for every $i < j$ and every t,

- either there are t edge-disj. paths from $v_1, \ldots, v_i$ to $v_j, \ldots, v_n$;



- or $\exists k, i \leqslant k < j$ s.t. $|E(\{v_1, \ldots, v_k\}, \{v_{k+1}, \ldots, v_n\})| < t$.



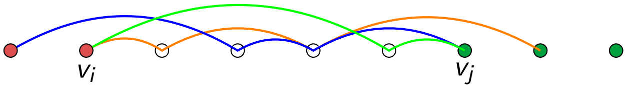
Lemma

Every graph has a linked ordering of optimal width.

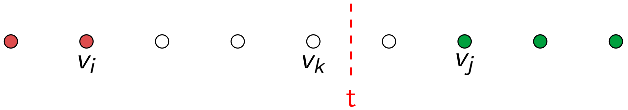
Linked orderings

An ordering $v_1 \ldots, v_n$ of is *linked* if, for every $i < j$ and every t,

- either there are t edge-disj. paths from $v_1, \ldots, v_i$ to $v_j, \ldots, v_n$;



- or $\exists k, i \leqslant k < j$ s.t. $|E(\{v_1, \ldots, v_k\}, \{v_{k+1}, \ldots, v_n\})| < t$.



Lemma

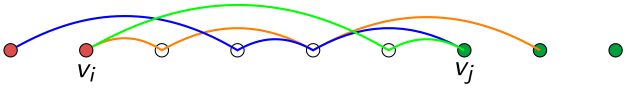
Every graph has a linked ordering of optimal width.

Proof: non-linked orderings can be *improved* without increasing the width.

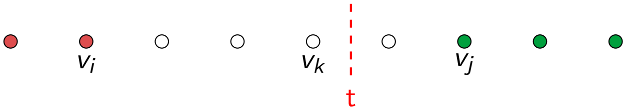
Linked orderings

An ordering $v_1 \dots, v_n$ of is *linked* if, for every $i < j$ and every t,

- either there are t edge-disj. paths from $v_1, \dots, v_i$ to $v_j, \dots, v_n$;



- or $\exists k, i \leqslant k < j$ s.t. $|E(\{v_1, \dots, v_k\}, \{v_{k+1}, \dots, v_n\})| < t$.



Lemma

Every graph has a linked ordering of optimal width.

Proof: non-linked orderings can be *improved* without increasing the width.

Similar notions: linked path decompositions, linked tree decompostions.

On words with repeating patterns

Lemma

If w is a word of length N^r over $[r]$, there is a $p \in [r]$ s.t.:

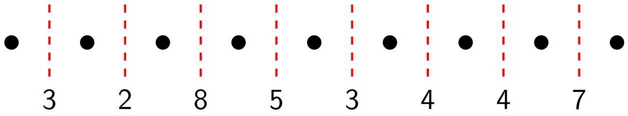
- *some subword u contains numbers $\geqslant p$;*
- *u contains p at least N times.*

On words with repeating patterns

Lemma

If w is a word of length N^r over $[r]$, there is a $p \in [r]$ s.t.:

- *some subword u contains numbers $\geqslant p$;*
- *u contains p at least N times.*

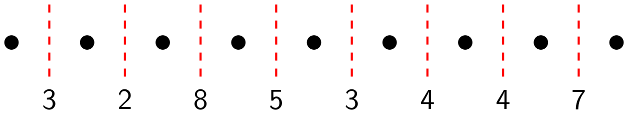


On words with repeating patterns

Lemma

If w is a word of length N^r over $[r]$, there is a $p \in [r]$ s.t.:

- *some subword u contains numbers $\geqslant p$;*
- *u contains p at least N times.*



If $|G| > N^r$, some contiguous subsequence of $v_1, \ldots, v_n$ has cuts $\geqslant p$ and $\geqslant N$ cuts of size p.

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k$

Bounding the size of obstructions

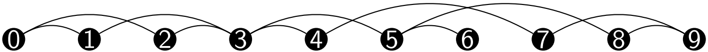
Goal: show that obstructions for $\mathbf{ctw} \leqslant k$ are small.

- G obstruction for $\mathbf{ctw} \leqslant k \Rightarrow \mathbf{ctw}(G) = k + 1$

Bounding the size of obstructions

Goal: show that obstructions for $\mathbf{ctw} \leqslant k$ are small.

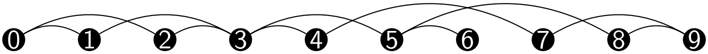
- G obstruction for $\mathbf{ctw} \leqslant k \Rightarrow \mathbf{ctw}(G) = k + 1$
- consider a linked optimal ordering of G:



Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k+1$
- consider a linked optimal ordering of G:

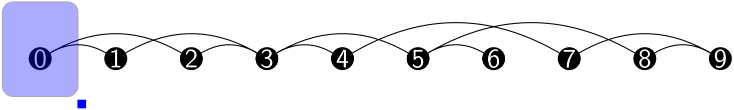


- assign a type to every prefix ("equivalence class for **ctw**")

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k+1$
- consider a linked optimal ordering of G:

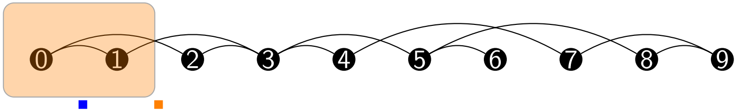


- assign a type to every prefix ("equivalence class for **ctw**")

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k+1$
- consider a linked optimal ordering of G:

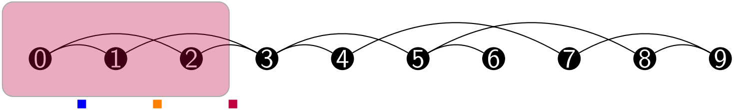


- assign a type to every prefix ("equivalence class for **ctw**")

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k + 1$
- consider a linked optimal ordering of G:

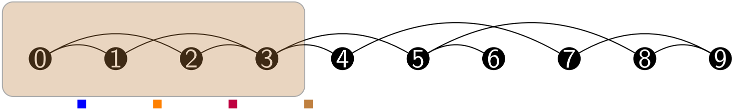


- assign a type to every prefix ("equivalence class for **ctw**")
- recall: there are finitely many different types

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k + 1$
- consider a linked optimal ordering of G:

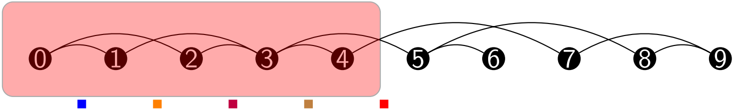


- assign a type to every prefix ("equivalence class for **ctw**")
- recall: there are finitely many different types

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k+1$
- consider a linked optimal ordering of G:

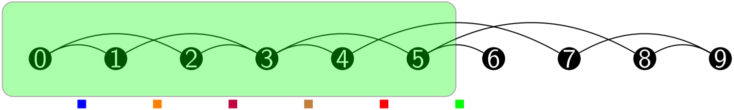


- assign a type to every prefix ("equivalence class for **ctw**")
- recall: there are finitely many different types

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k + 1$
- consider a linked optimal ordering of G:

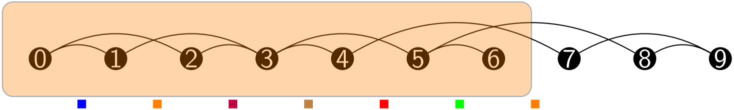


- assign a type to every prefix ("equivalence class for **ctw**")
- recall: there are finitely many different types
- if $|G|$ is large enough, types will repeat

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k + 1$
- consider a linked optimal ordering of G:

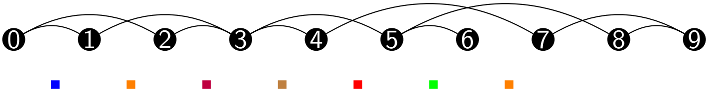


- assign a type to every prefix ("equivalence class for **ctw**")
- recall: there are finitely many different types
- if $|G|$ is large enough, types will repeat

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k+1$
- consider a linked optimal ordering of G:

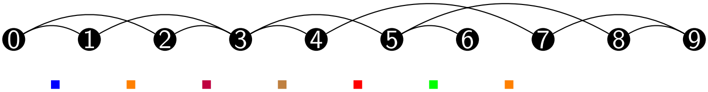


- assign a type to every prefix ("equivalence class for **ctw**")
- recall: there are finitely many different types
- if $|G|$ is large enough, types will repeat
- shrink using edge-disjoint paths:

Bounding the size of obstructions

Goal: show that obstructions for **ctw** $\leqslant k$ are small.

- G obstruction for **ctw** $\leqslant k \Rightarrow$ **ctw**$(G) = k+1$
- consider a linked optimal ordering of G:



- assign a type to every prefix ("equivalence class for **ctw**")
- recall: there are finitely many different types
- if $|G|$ is large enough, types will repeat
- shrink using edge-disjoint paths:

- this immersion of G has cutwidth $k+1$: contradiction.

Computing cutwidth

Problem: deciding given (G, k) if $\mathbf{ctw}(G) \leqslant k$.

- non-uniform, non-constructive FPT (by the finiteness of obstructions);

Computing cutwidth

Problem: deciding given (G, k) if $\mathbf{ctw}(G) \leqslant k$.

- non-uniform, non-constructive FPT (by the finiteness of obstructions);
- constructive FPT algorithm with running time $2^{O(k^2)} \cdot n$.
  (Thilikos, Bodlaender, and Serna)

Computing cutwidth

Problem: deciding given (G, k) if $\mathbf{ctw}(G) \leqslant k$.

- non-uniform, non-constructive FPT (by the finiteness of obstructions);
- constructive FPT algorithm with running time $2^{O(k^2)} \cdot n$.
  (Thilikos, Bodlaender, and Serna)

Theorem (Giannopoulou, Pilipczuk, R., Thilikos, Wrochna, 2016)

The cutwidth of a graph can be computed in time $2^{O(k^2 \log k)} \cdot n$.

Computing cutwidth

Problem: deciding given (G, k) if $\mathbf{ctw}(G) \leqslant k$.

- non-uniform, non-constructive FPT (by the finiteness of obstructions);
- constructive FPT algorithm with running time $2^{O(k^2)} \cdot n$. (Thilikos, Bodlaender, and Serna)

Theorem (Giannopoulou, Pilipczuk, R., Thilikos, Wrochna, 2016)

The cutwidth of a graph can be computed in time $2^{O(k^2 \log k)} \cdot n$.

Ingredients:

- equivalence classes of subgraphs w.r.t. cutwidth;
- DP on graphs of bounded cutwidth;
- "edge-removal" lemma.

Computing cutwidth

Problem: deciding given (G, k) if $\mathbf{ctw}(G) \leqslant k$.

- non-uniform, non-constructive FPT (by the finiteness of obstructions);
- constructive FPT algorithm with running time $2^{O(k^2)} \cdot n$.
  (Thilikos, Bodlaender, and Serna)

Theorem (Giannopoulou, Pilipczuk, R., Thilikos, Wrochna, 2016)

The cutwidth of a graph can be computed in time $2^{O(k^2 \log k)} \cdot n$.

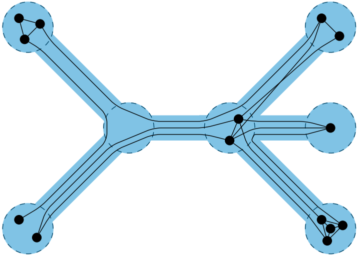
Ingredients:

- equivalence classes of subgraphs w.r.t. cutwidth;
- DP on graphs of bounded cutwidth;
- "edge-removal" lemma.

Slightly slower... but much easier!

Extension to tree-like parameters

Tree-cut decomposition:



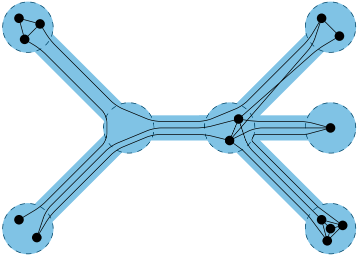
Associated parameter: **tcw**
(tree-cut width)

Small **tcw** implies:

- *small* bags;
- *thin* edges;
- small number of *thick* neighbors.

Extension to tree-like parameters

Tree-cut decomposition:



Associated parameter: **tcw**
(tree-cut width)

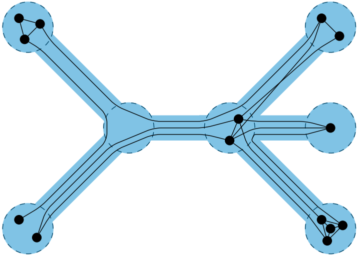
Small **tcw** implies:

- *small* bags;
- *thin* edges;
- small number of *thick* neighbors.

treewidth and minors $\sim$ tree-cut width and immersions

Extension to tree-like parameters

Tree-cut decomposition:



Associated parameter: **tcw**
(tree-cut width)

Small **tcw** implies:

- *small* bags;
- *thin* edges;
- small number of *thick* neighbors.

treewidth and minors $\sim$ tree-cut width and immersions

Hope for similar results in this context (work in progress).

The end

Our contribution:

- a single exponential upper-bound on the size of the obstructions for **ctw** $\leqslant k$;
- a simpler FPT algorithm for cutwidth.

The end

Our contribution:

- a single exponential upper-bound on the size of the obstructions for **ctw** $\leqslant k$;
- a simpler FPT algorithm for cutwidth.

Thank you!