

On the parameterized complexity of computing tree-partitions

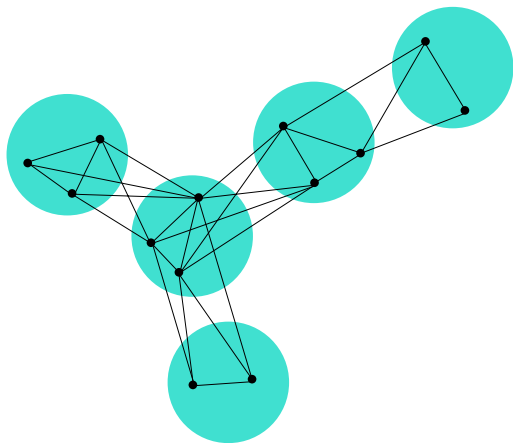
Hans L. Bodlaender, Carla Groenland, Hugo Jacob



école —————
normale —————
supérieure —————
paris-saclay —————

Tree-partition

What is a tree-partition?

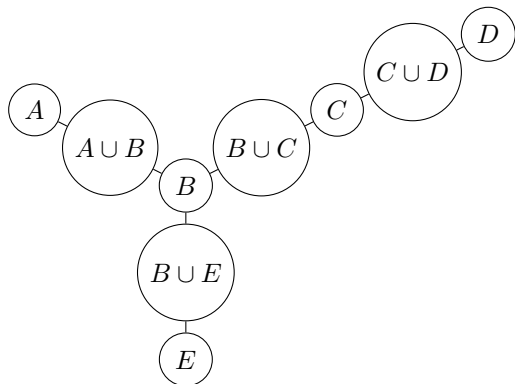


Width is largest bag size (3 here).

Some context

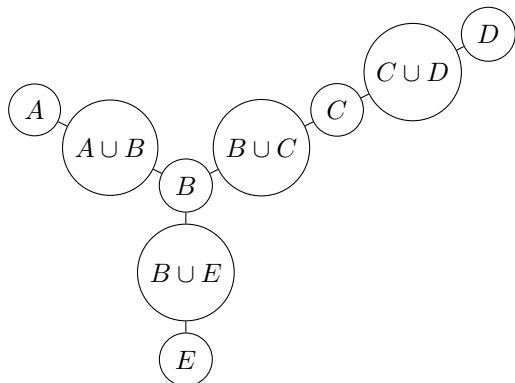
- ▶ Introduced independently by Seese ('85) and Halin ('91).
- ▶ Work of Ding and Oporowski ('95,'96) on tree-partition-width: its relationship with treewidth and a characterisation by topological minor obstructions.
- ▶ Mostly studied for 'structural' purposes (coloring, layout, ...)
- ▶ Recently Bodlaender, Cornelissen, and van der Wegen give parameterized problems that are FPT parameterized by tree-partition-width but hard for treewidth.

Relation to treewidth



$$\text{tw} \leq 2\text{tpw} - 1$$

Relation to treewidth



$$\mathbf{tw} \leq 2\mathbf{tpw} - 1$$

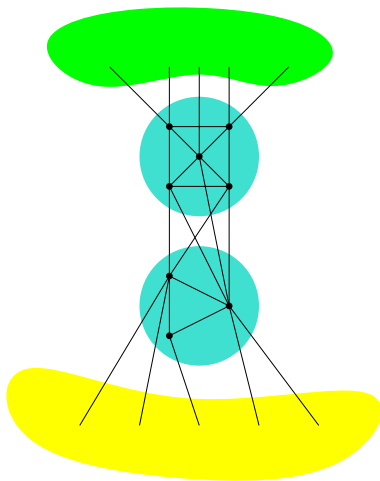
Ding and Oporowski ('95) and Wood ('09) show $\mathbf{tpw} = O(\Delta\mathbf{tw})$.

Missing piece

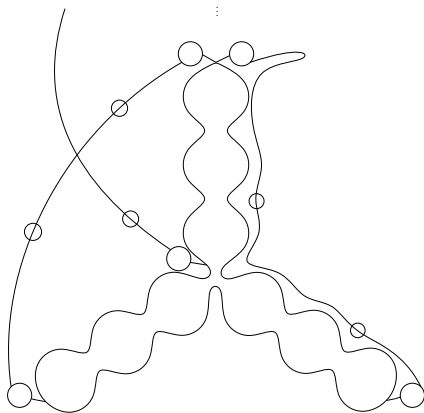
No known algorithms to efficiently compute a tree-partition of good width.

Optimal tree-partition

A naive XP algorithm



Hardness



Reduction from a variant of INDEPENDENT SET.

Theorem

TREE-PARTITION-WIDTH is *XALP-complete*.

This means we do not expect a significantly better algorithm to find an optimal tree-partition to exist.

Theorem

TREE-PARTITION-WIDTH is *XALP-complete*.

This means we do not expect a significantly better algorithm to find an optimal tree-partition to exist.

We can deduce XALP completeness for DOMINO TREEWIDTH.

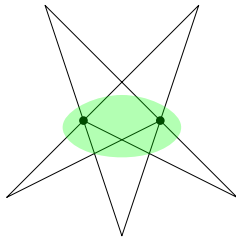
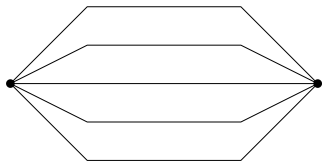
Exploiting the topological minor obstructions

- ▶ We can find topological minor obstructions in FPT time, using the algorithm of Grohe, Kawarabayashi, Marx, and Wollan (STOC 2011).
- ▶ The result of Ding and Oporowski is constructive and can be turned into an algorithm.

Scheme

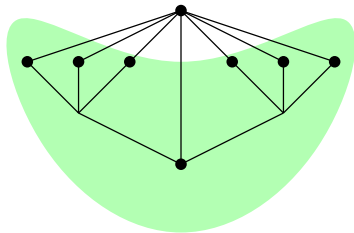
- ▶ First check that the treewidth is small and get a tree decomposition.
- ▶ Compute some clever auxiliary graph.
- ▶ Compute a tree-partition on blocks of the auxiliary graph.
- ▶ Combine them and deduce a tree-partition of the original graph.

Auxiliary graphs



Approximation

Degree in blocks of the reduced graph



Results

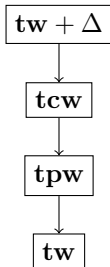
The scheme allows to find tree-partitions of width:

- ▶ $O(k^5 \log k)$ in polynomial time; or
- ▶ $O(k^5)$ in $2^{O(k^2)}n + k^{O(1)}n \log n$ time; or
- ▶ $O(k^7)$ in $k^{O(1)}n^2$.

or conclude that any tree-partition has width more than k .

Other result

Tree-partition width is polynomially bounded by tree-cut width.



Further work

- ▶ Can we find better approximations ?
- ▶ Can we improve the running time of our approximations ?
($f(k)n$ or $k^c n \log n$?)
- ▶ What are the problems that can be solved in FPT time when parameterized by tree-partition-width ?