Polynomial Kernels for Hitting Forbidden Minors under Structural Parameterizations*

Technische Universiteit Eindhoven University of Technology

Bart M.P. Jansen and Astrid Pieterse

NETWORKS

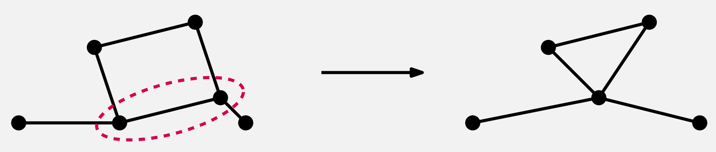
Background and preliminaries

The goal of this research is to investigate polynomial-time preprocessing algorithms (also known as kernels). Such an algorithm takes an input instance X, and outputs an equivalent input instance X' that is hopefully small. We want to bound the size of X' in terms of some complexity measure for X, also known as the parameter.

The aim of this research is to find a good preprocessing algorithm that is applicable in as many situations as possible. Therefore, we want to study a very general problem. A possible candidate problem would be the set of problems expressible in some sort of logic. Unfortunately, the Dominating Set problem is often easy to express, but does not allow for good preprocessing even when using Vertex Cover as the parameter [1]. Therefore, we study a more restricted class of problems, called $\mathcal{F}$-minor free deletion.

Graph minors

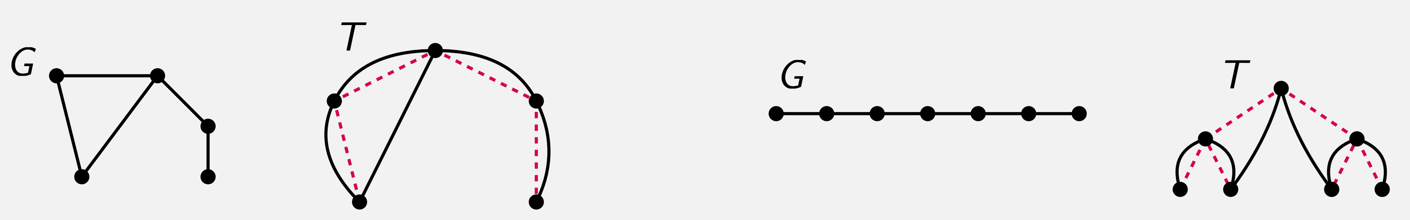
H is a minor of G, if H can be obtained from a subgraph of G by a sequence of edge contractions.



Treedepth

The treedepth td(G) of a graph G is defined as the minimum depth of a treedepth decomposition. A treedepth decomposition of G is a tree T on the vertex set of G, that satisfies

- For each edge $\{u, v\} \in V(G)$, u and v are in ancestor/child relation in T.



Graph of treedepth 3 — Paths have treedepth $O(\log n)$

Problem

We will study the $\mathcal{F}$-minor free deletion problem, where $\mathcal{F}$ is a set of connected graphs.

Input A graph G and an integer k.

Question Is it possible to remove at most k vertices from G, such that G no longer has any graph in $\mathcal{F}$ as a minor?

$\mathcal{F}$-minor free deletion generalizes many problems:

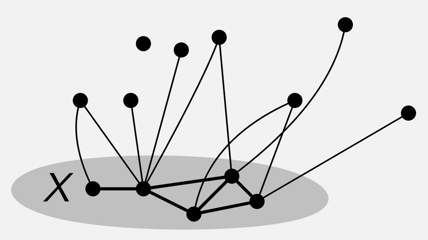
Problem	Choice for $\mathcal{F}$
Vertex Cover	$\{K_2\}$
Feedback Vertex Set	$\{K_3\}$
Graph planarizition (by vertex deletions)	$\{K_{3,3}, K_5\}$

Parameter

We want to give a preprocessing algorithm, with a guarantee on the size of the reduced instance. To bound the size, an additional parameter is used to measure the complexity of the problem. Often, the solution size is used as the parameter, c.f. [2]. However, the solution size can be very large. We want to obtain usable preprocessing algorithms even in this situation. Therefore, we use the following structural parameter:

Modulator to treedepth η

- Set of vertices X such that td$(G - X) = \eta$, the parameter is $|X|$.



Treedepth-1 modulator

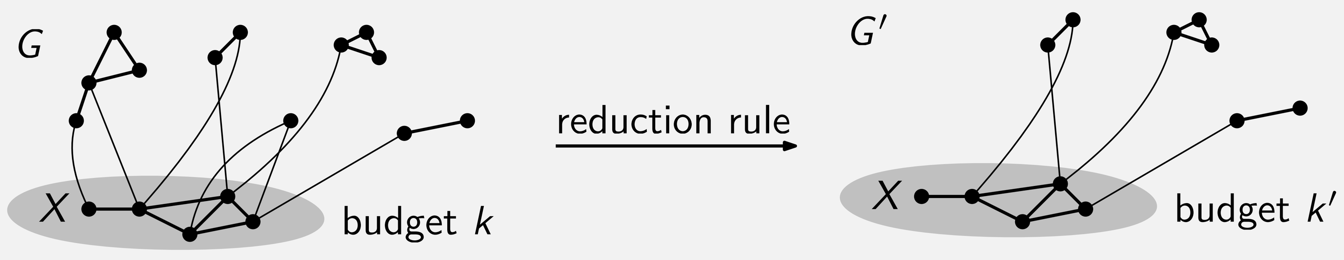
Main result

$\mathcal{F}$-minor free deletion parameterized by a treedepth-η modulator has a polynomial kernel: There exists a polynomial-time algorithm that is given an instance G with budget k and modulator X, and outputs an equivalent instance G', k' such that $|G'| = \text{poly}(|X|)$.

The kernelization procedure we obtain is constructive, and does not rely on techniques like protrusion replacement and well-quasi ordering.

Proof strategy

We will give a reduction rule to reduce the number of connected components of $G - X$ to polynomially many. Then, we use the next lemma.

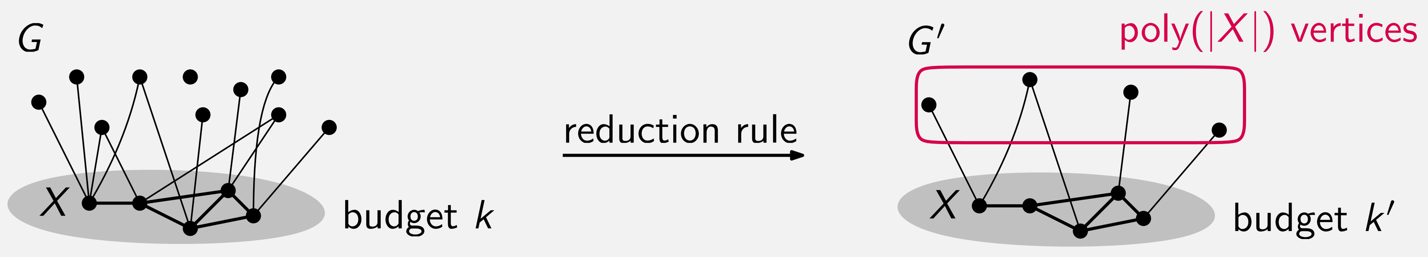


Lemma

Given such a reduction rule, that reduces the number of connected components outside X to a polynomial in $|X|$, we can show that $\mathcal{F}$-minor free deletion has a polynomial kernel. We can prove this by induction on η (the treedepth of $G - X$).

$\eta = 1$

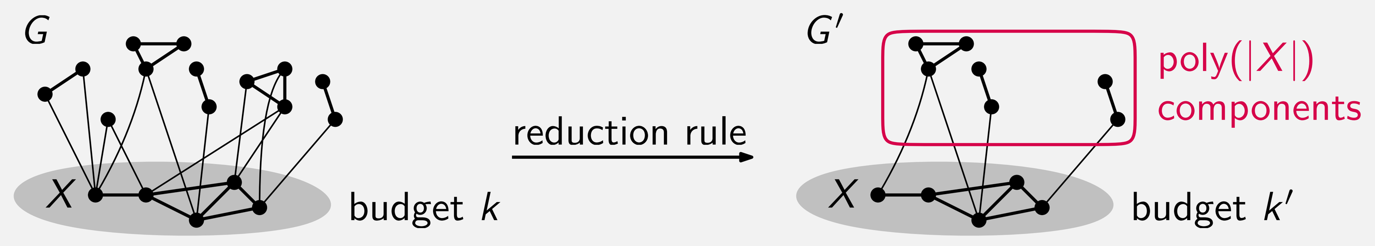
In this case, $G - X$ has treedepth 1. Thereby, every component of $G - X$ has size one by the definition of treedepth.



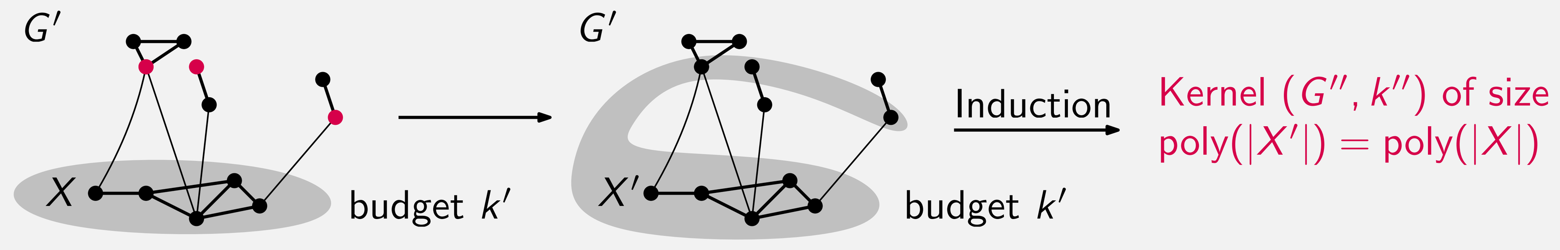
(G', k') is a kernel of size poly$(|X|)$

$\eta > 1$

Start by applying the reduction rule.



Then, add the roots of the treedepth decomposition of each component of $G' - X$, to X.



X' is now a treedepth-$(\eta - 1)$ modulator. Obtain a kernel of size poly$(|X'|)$ by induction. Since $|X'|$ is polynomial in $|X|$, this gives the desired kernel.

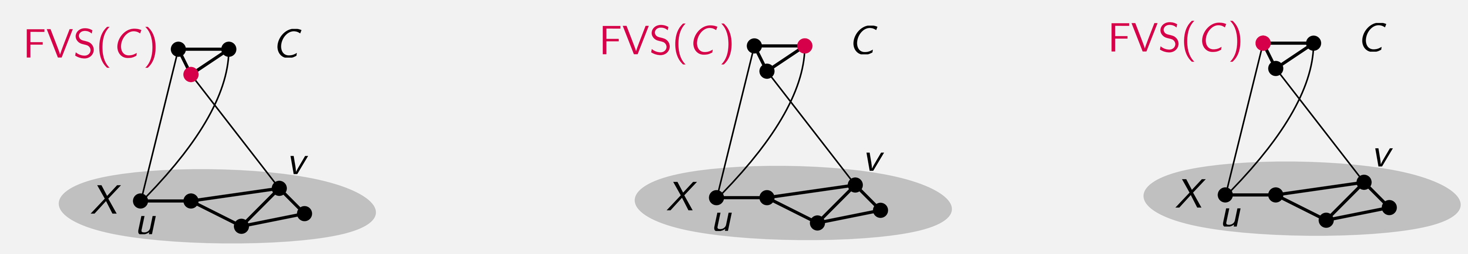
Removing connected components of G − X

We will reduce the number of connected components in $G - X$. Let us use Feedback Vertex Set (FVS) as an example. It is important to note that any optimal FVS in G, is in fact locally optimal in all but $\leq |X|$ components of $G - X$. Otherwise, a better solution can be obtained by removing X and removing an optimal FVS from each component of $G - X$.

Using this observation, we analyze the behavior of optimal solutions in each component of $G - X$. We will show that there are not too many different "types" of behavior. For each such type, we mark a (small) number of representative components. Then, all unmarked components can be safely removed.

Examples of interesting behavior for FVS

We study what happens when removing a local optimum from a component of $G - X$, and whether this could form a cycle in combination with the rest of the graph.



Connects $u \in X$ to itself — Connects u to v — Connects u to v

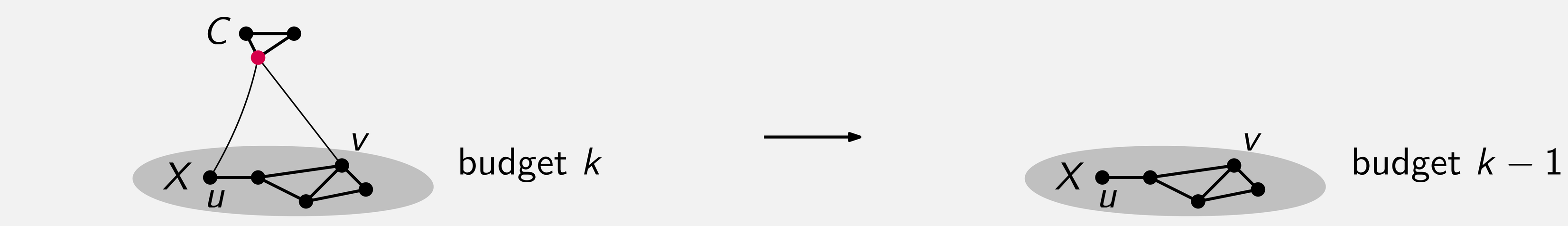
Main lemma for FVS (sketch)

Let L be a set of u, v-pairs for $u, v \in X$ and let C be a component of $G - X$. If there is no optimal FVS in C that separates all u, v-pairs in L, then there is a constant-size set $L' \subseteq L$, such that no optimal FVS in C separates all u, v-pairs in L'.

Usage of main lemma

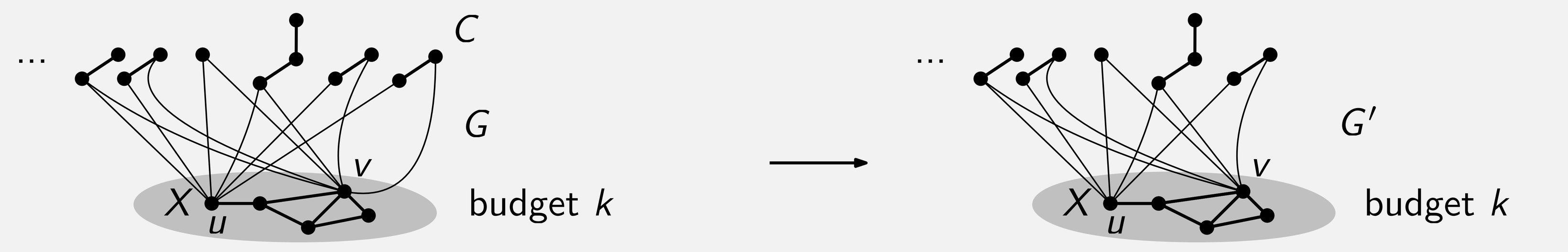
The main lemma shows that the number of types of behavior that a component may have is limited. In particular, it allows us to only mark components for lists L of constant size.

Example of a safe reduction rule (1)



Some optimal FVS in C disconnects C from X — Remove C, decrease budget by FVS(C)

Example of a safe reduction rule (2)

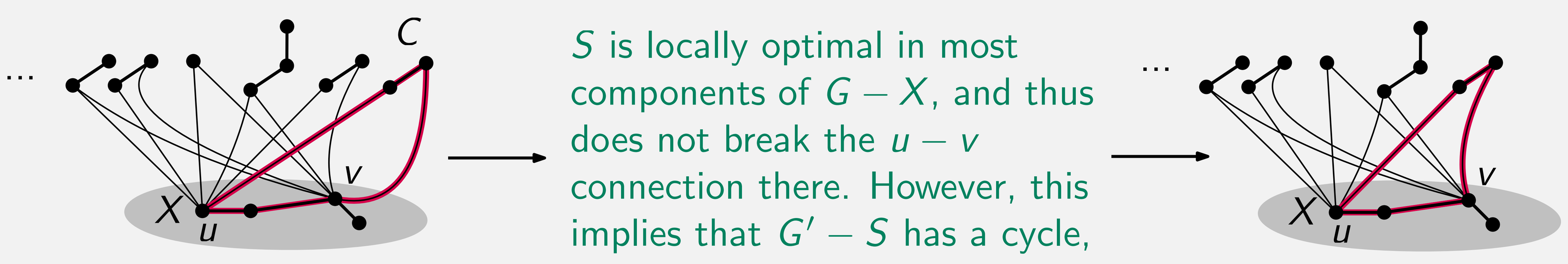


There are many components of $G - X$ where no optimal FVS removes the $u - v$ connection via the component, and C only connects to u and v.

Remove C, decrease budget by FVS(C)

Correctness

Let S be a FVS in G', we show that $S \cup FVS(C)$ is a FVS in G. Suppose not, then the connection between u and v introduced by $C - FVS(C)$ creates a cycle:



$G - (S \cup FVS(C))$ — $G' - S$

References

[1] Michael Dom, Daniel Lokshtanov, and Saket Saurab. Incompressibility through Colors and IDs. In *Proc. 36th ICALP*, 2009.

[2] Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, and Saket Saurabh. Planar F-Deletion: Approximation, Kernelization and Optimal FPT Algorithms. In *Proc. 53rd FOCS*, 2012.

* See https://arxiv.org/pdf/1804.08885.pdf