Computing Twin-Width Parameterized by the Feedback Edge Number

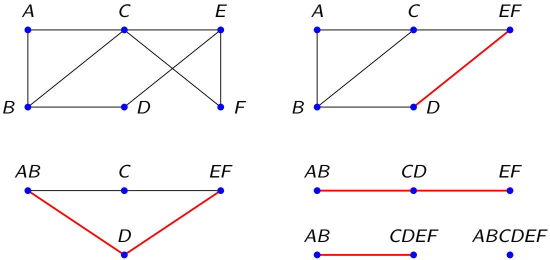
Jakub Balabán [1] Robert Ganian [2] Mathis Rocton [2]

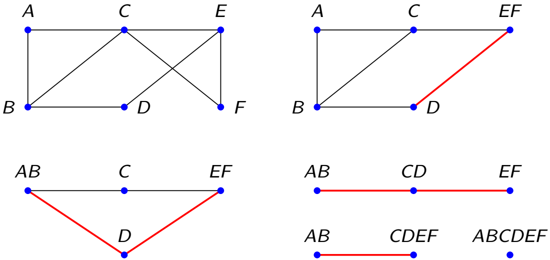
[1] Masaryk University, Brno

[2] Technische Universität Wien

November 24, 2023

Twinwidth





The **twinwidth** $tww(G)$ of G is the *minimum over all contraction sequences* of the contraction's width.

Twinwidth

[Bergé, Bonnet, Déprés (2022)]

Deciding if the twin-width is at most 4 is NP-hard.

[Bergé, Bonnet, Déprés (2022)]

Deciding if the twin-width is at most 4 is NP-hard.

Why do we care?

[Bonnet, Kim, Thomassé, Watrigant (2020)]

Provided a contraction sequence of width t as part of the input, FO model checking can be solved in FPT time, parameterized by t.

Twinwidth

[Bergé, Bonnet, Déprés (2022)]

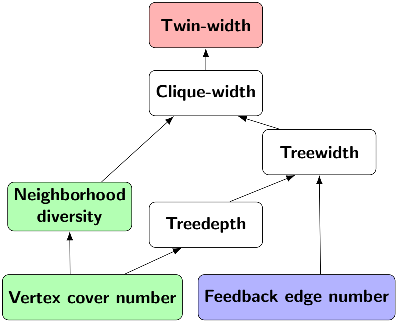
Deciding if the twin-width is at most 4 is NP-hard.

Why do we care?

[Bonnet, Kim, Thomassé, Watrigant (2020)]

Provided a contraction sequence of width t as part of the input, FO model checking can be solved in FPT time, parameterized by t.

Parameterized computation of twinwidth



Feedback edge number $fen(G)$**:** minimum size of an edge set S, such that $G - S$ has no cycles (*i.e.* is a forest).

Twinwidth and Feedback Edge Number

Easy observation:

For any graph G, $tww(G) \leq fen(G) + 2$.

Twinwidth and Feedback Edge Number

Easy observation:

For any graph G, $tww(G) \leq fen(G) + 2$.

We want to compute an approximate value for $tww(G)$ **independent of** $fen(G)$**!**

Our results

Theorem

Deciding tww$(G) \leq 2$ admits a linear bikernel when parameterized by the feedback edge number k of G. Moreover, a 2-contraction sequence for G (if one exists) can be computed in time $2^{\mathcal{O}(k \cdot \log k)} + n^{\mathcal{O}(1)}$.

Our results

Theorem

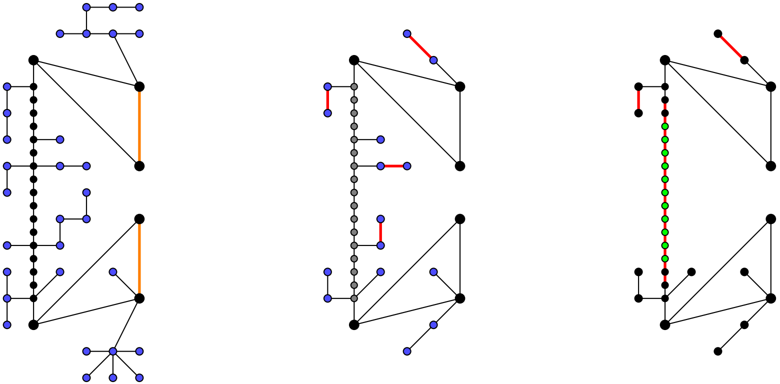
Deciding $tww(G) \leq 2$ admits a linear bikernel when parameterized by the feedback edge number k of G. Moreover, a 2-contraction sequence for G (if one exists) can be computed in time $2^{\mathcal{O}(k \cdot \log k)} + n^{\mathcal{O}(1)}$.

Theorem

For an n-vertex graph G with feedback edge number k, there is an algorithm running in time $f(k) \cdot n^{\mathcal{O}(1)}$ for a computable function f which outputs a contraction sequence for G of width at most $tww(G) + 1$.

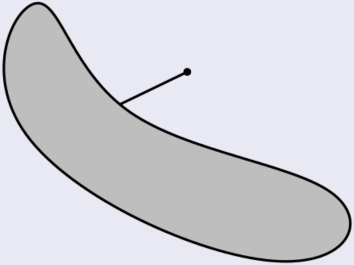
General idea

- Replace pendant trees by simpler structures of small size
- Replace long paths by paths of bounded size
- Combine these two rules as in classic kernelizations with feedback edge number

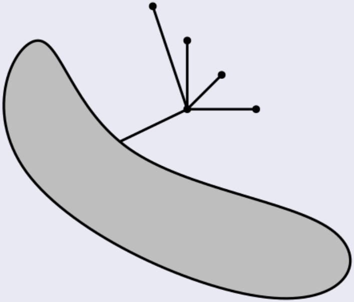


Reducing pendant trees

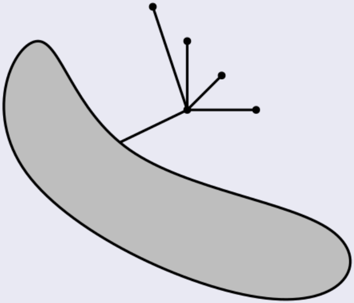
We replace pendant trees by **stumps**:



We replace pendant trees by **stumps**:

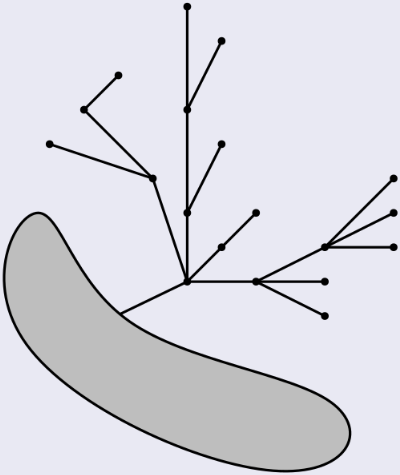


We replace pendant trees by **stumps**:

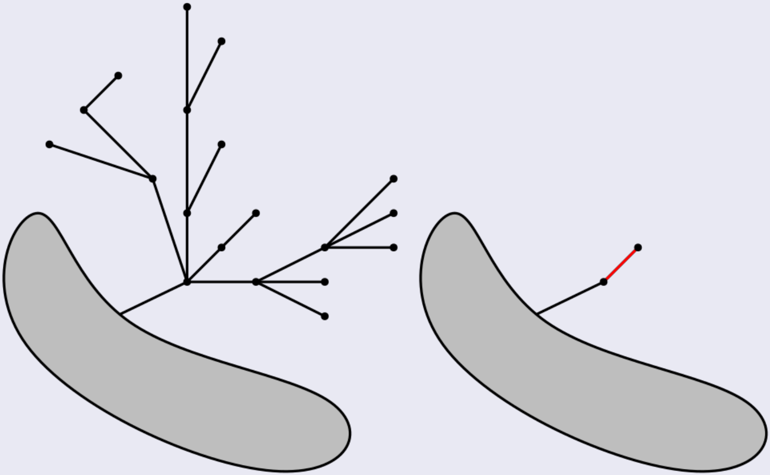


Reducing pendant trees

We replace pendant trees by **stumps**:



We replace pendant trees by **stumps**:



Reducing dangling paths

When $tww(G) = 2 \ldots$

… we can shorten paths to a bounded length!

Reducing dangling paths

When $tww(G) = 2\ldots$

$\ldots$ we can shorten paths to a bounded length!

It does not generalize!

There cannot be any constant c such that shortening paths to length c preserves twinwidth in general.

Make it simpler, take bigger steps!

Decisive or not?

Given a set of vertices $S \subseteq V(G)$, a contraction is **decisive** if:

- two vertices of S are contracted together, **or;**
- a new red edge is created between two vertices of S.

Make it simpler, take bigger steps!

Decisive or not?

Given a set of vertices $S \subseteq V(G)$, a contraction is **decisive** if:

- two vertices of S are contracted together, **or;**
- a new red edge is created between two vertices of S.

Big-step contraction sequence

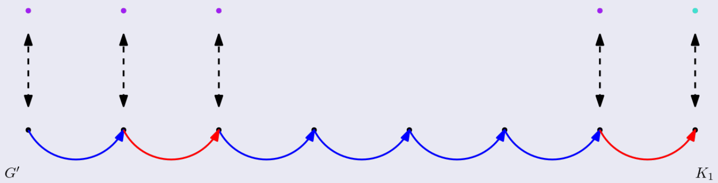
Make it simpler, take bigger steps!

Decisive or not?

Given a set of vertices $S \subseteq V(G)$, a contraction is **decisive** if:

- two vertices of S are contracted together, **or;**
- a new red edge is created between two vertices of S.

Big-step contraction sequence



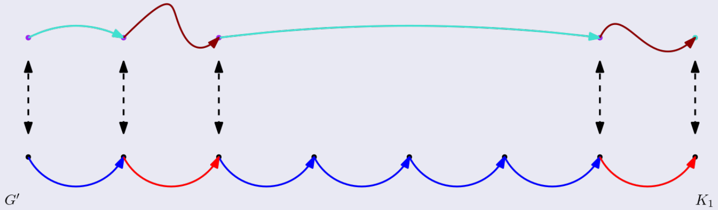
Make it simpler, take bigger steps!

Decisive or not?

Given a set of vertices $S \subseteq V(G)$, a contraction is **decisive** if:

- two vertices of S are contracted together, **or;**
- a new red edge is created between two vertices of S.

Big-step contraction sequence



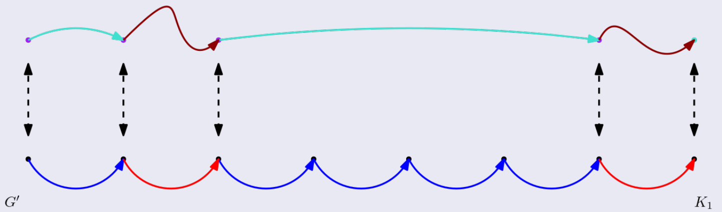
Make it simpler, take bigger steps!

Decisive or not?

Given a set of vertices $S \subseteq V(G)$, a contraction is **decisive** if:

- two vertices of S are contracted together, **or;**
- a new red edge is created between two vertices of S.

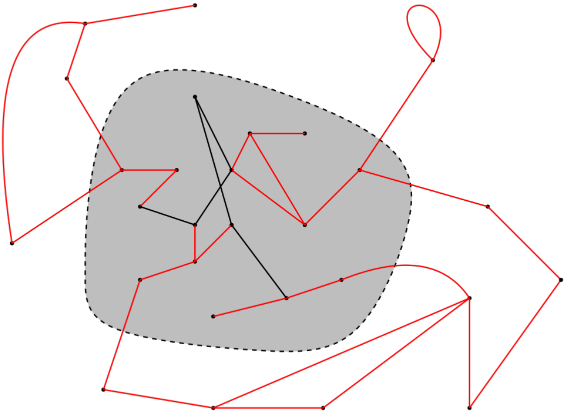
Big-step contraction sequence



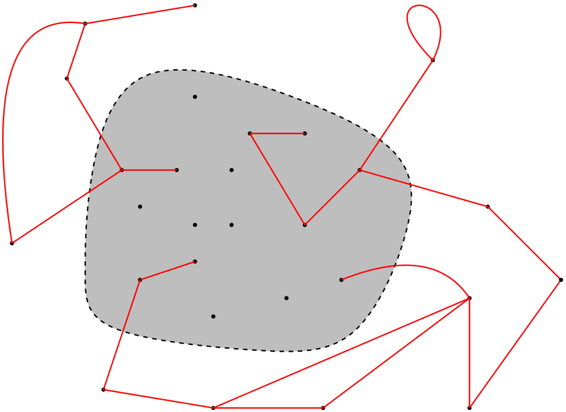
The number of big steps is $\mathcal{O}(|S|^2)$:

→ For us $|S|$ will be function of the **feedback edge number**.

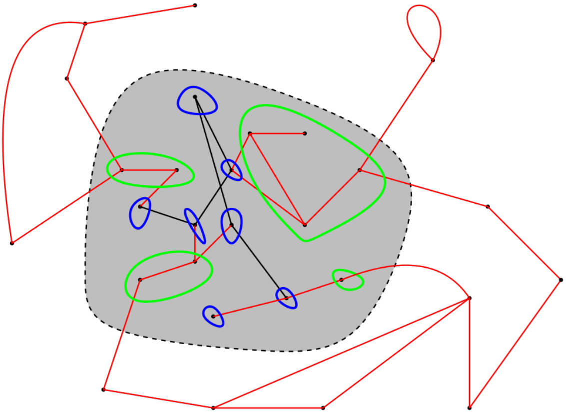
Blueprints, Representatives and Centipedes



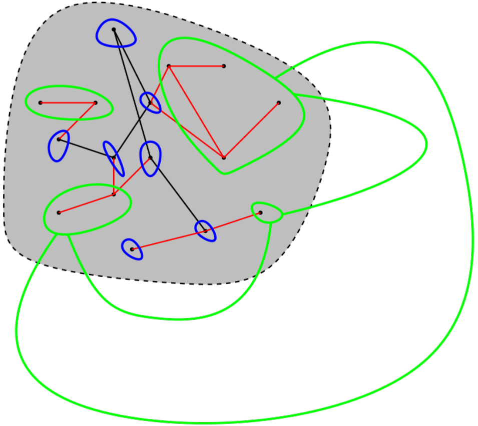
Blueprints, Representatives and Centipedes

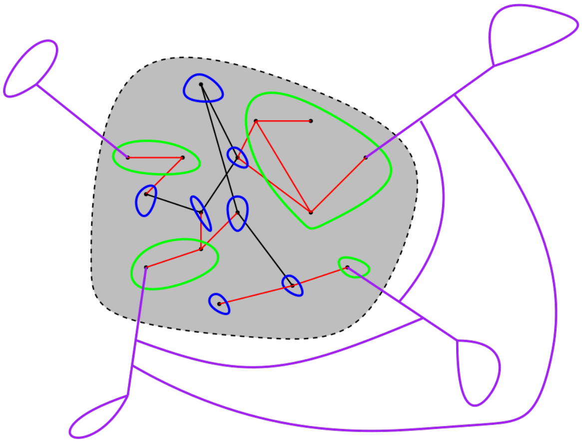


Blueprints, Representatives and Centipedes

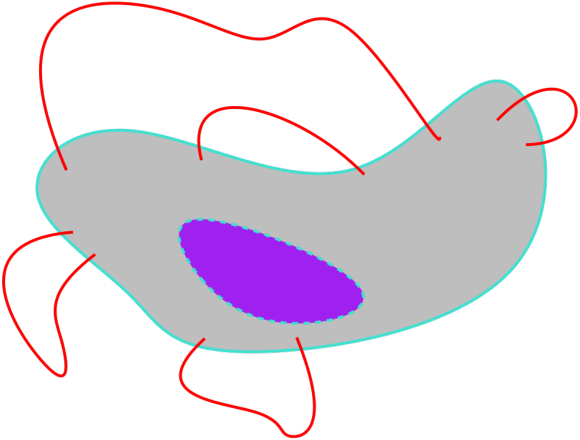


Blueprints, Representatives and Centipedes





Bikernelization



Future work: near-optimal contraction sequences in FPT-time by *treedepth*? By *treewidth*?

Thanks!

Questions?