# Structure Theorem and Isomorphism Test for Graphs with Excluded Topological Subgraphs

Martin Grohe[1] Dániel Marx[1]

[1]Computer and Automation Research Institute,
Hungarian Academy of Sciences (MTA SZTAKI)
Budapest, Hungary,

Graph Theory @ Georgia Tech
Atlanta, GA
May 8, 2012

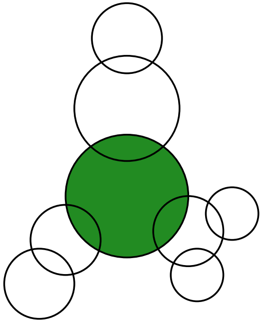
# Overview

- Decomposition theorem for graphs excluding a topological minor (subdivision) of a fixed graph $H$.
- Algorithmic applications
  - Example: Partial Dominating Set
  - Isomorphism test.
- Warning: technical details and definitions are omitted.

# Tree decompositions

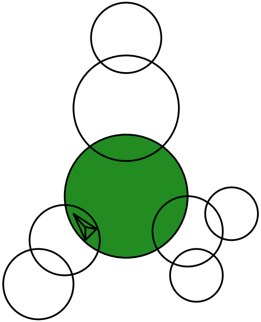
Torso of a bag: we make the intersections with the adjacent bags cliques.

Adhesion: maximum intersection with adjacent bags.



# Tree decompositions

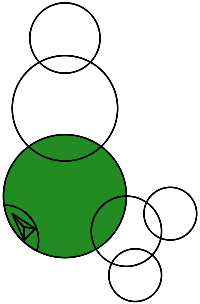
Torso of a bag: we make the intersections with the adjacent bags cliques.
Adhesion: maximum intersection with adjacent bags.



# Tree decompositions

Torso of a bag: we make the intersections with the adjacent bags cliques.

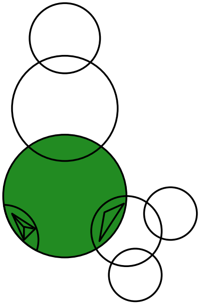
Adhesion: maximum intersection with adjacent bags.



# Tree decompositions

Torso of a bag: we make the intersections with the adjacent bags cliques.

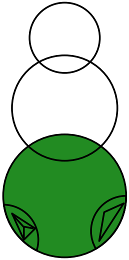
Adhesion: maximum intersection with adjacent bags.



# Tree decompositions

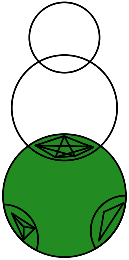
Torso of a bag: we make the intersections with the adjacent bags cliques.

Adhesion: maximum intersection with adjacent bags.



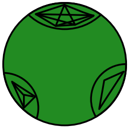
# Tree decompositions

Torso of a bag: we make the intersections with the adjacent bags cliques.
Adhesion: maximum intersection with adjacent bags.



# Tree decompositions

Torso of a bag: we make the intersections with the adjacent bags cliques.
Adhesion: maximum intersection with adjacent bags.



# Structure theorems

**Theorem [Robertson and Seymour]**

Every $H$-minor free graph has a tree decomposition where the torso of every bag is "$c_H$-almost-embeddable."

Note: There is an $f(H) \cdot n^{O(1)}$ time algorithm for computing such a decomposition [Kawarabayashi-Wollan 2011].

Can we prove a similar result for the more general class of $H$-subdivision free graphs?

These classes are significantly more general: e.g., every 3-regular graph is $K_5$-subdivision free.

# Structure theorems

**Theorem [Robertson and Seymour]**

Every $H$-minor free graph has a tree decomposition where the torso of every bag is "$c_H$-almost-embeddable."

Note: There is an $f(H) \cdot n^{O(1)}$ time algorithm for computing such a decomposition [Kawarabayashi-Wollan 2011].

Can we prove a similar result for the more general class of $H$-subdivision free graphs?

These classes are significantly more general: e.g., every 3-regular graph is $K_5$-subdivision free.

# Structure theorems

## New result

Every $H$-subdivision free graph has a tree decomposition where the torso of every bag is either

- $K_{c_H}$-minor free or
- has degree at most $c_H$ with the exception of at most $c_H$ vertices ("almost bounded degree").

Note: there is an $f(H) \cdot n^{O(1)}$ time algorithm for computing such a decomposition.

# Structure theorems

## New result

Every $H$-subdivision free graph has a tree decomposition where the torso of every bag is either

- "$c_H$-almost-embeddable" or
- has degree at most $c_H$ with the exception of at most $c_H$ vertices ("almost bounded degree").

Note: there is an $f(H) \cdot n^{O(1)}$ time algorithm for computing such a decomposition.

# Proof overview

Star decomposition: tree decomposition where the tree is a star.

## Local decomposition theorem

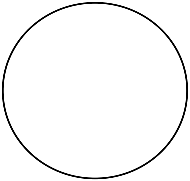
Given an $H$-subdivision free graph and a set $S$ of at most $a_H$ vertices, there is star decomposition with adhesion at most $a_H$ where $S$ is in the center bag and the torso of the center + (clique on $S$) either

(i) has bounded size.

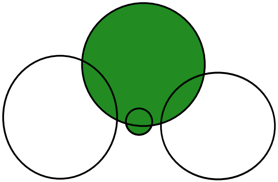
(ii) excludes a clique minor.

(iii) has almost-bounded degree.

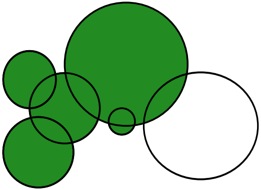
# Iterating local decompositions



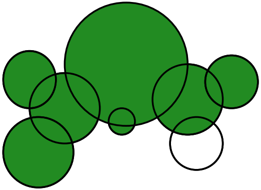
# Iterating local decompositions



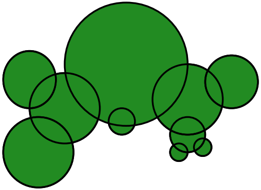
# Iterating local decompositions

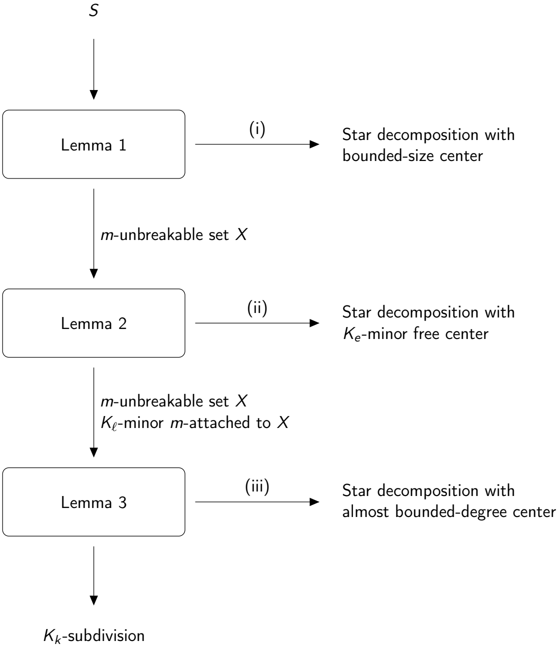


# Iterating local decompositions



# Iterating local decompositions




$S$
Lemma 1
(i)
Star decomposition with bounded-size center
$m$-unbreakable set $X$
Lemma 2
(ii)
Star decomposition with $K_e$-minor free center
$m$-unbreakable set $X$
$K_\ell$-minor $m$-attached to $X$
Lemma 3
(iii)
Star decomposition with almost bounded-degree center
$K_k$-subdivision

# Local decomposition

Idea behind (i) is standard (approximating treewidth).

Same general idea for (ii) and (iii):

- Locate the objects that violate the property (clique minors, high degree vertices).
- Argue that they can be removed with small separators.
- Uncrossing arguments show that these separators do not interfere much.
- Removing something introduces cliques in the torsos. Show that they don't cause problems.

# Algorithmic applications

**New result**

Every $H$-subdivision free graph has a tree decomposition where the torso of every bag is either

- "$c_H$-almost-embeddable" or
- has degree at most $c_H$ with the exception of at most $c_H$ vertices ("almost bounded degree").

General message:

If a problem can be solved both

- on (almost-) bounded degree graphs and
- on (almost-) embeddable graphs,

then these results can be raised to

- $H$-subdivision free graphs

without too much extra effort.

# Partial Dominating Set

### Partial Dominating Set

Input: graph $G$, integer $k$
Find: a set $S$ of at most $k$ vertices whose closed neighborhood has maximum size

### Theorem

Partial Dominating Set can be solved in time $f(H,k) \cdot n^{O(1)}$ on $H$-subdivision free graphs.

# Partial Dominating Set

Sketch:

- Partial Dominating Set can be solved in linear-time on bounded-degree graphs (the closed neighborhood has bounded size).
- Partial Dominating Set can be solved in linear-time on planar graphs (standard layering/treewidth arguments).
- With some extra work, we can generalize this to almost-bounded degree and almost-embeddable graphs.
- The structure theorem together with bottom-up dynamic programming gives an algorithm for $H$-subdivision free graphs.

# Graph Isomorphism

**Graph Isomorphism**

Input: graph $G_1$ and $G_2$
Decide: are $G_1$ and $G_2$ isomorphic?

Not known to be polynomial-time solvable, not believed to be NP-hard.

Related problems:

- Decide if two graphs are isomorphic.
- Compute a canonical label for the graph.
- Compute a canonical labeling of the vertices.

# Graph Isomorphism

## Graph Isomorphism

Input: graph $G_1$ and $G_2$
Decide: are $G_1$ and $G_2$ isomorphic?

Not known to be polynomial-time solvable, not believed to be NP-hard.

Related problems:

- Decide if two graphs are isomorphic.
- Compute a canonical label for the graph.
- Compute a canonical labeling of the vertices.

# Graph Isomorphism

**Theorem [Luks 1982] [Babai, Luks 1983]**

For every fixed $d$, Graph Isomorphism can be solved in polynomial time on graphs with maximum degree $d$.

**Theorem [Ponomarenko 1988]**

For every fixed $H$, Graph Isomorphism can be solved in polynomial time on $H$-minor free graphs.

**New result**

For every fixed $H$, Graph Isomorphism can be solved in polynomial-time on $H$-subdivision free graphs.

Note: running time is $n^{f(H)}$, not FPT parameterized by $H$.

# Graph Isomorphism

**New result**

For every fixed $H$, Graph Isomorphism can be solved in polynomial-time on $H$-subdivision free graphs.

Proof idea:

- Use bottom up dynamic programing to compute a canonical label for every subtree.
- We can compute a canonical label for each torso using the bounded-degree or the excluded minor algorithm.
- Incorporate the labels of the children as annotation.

# Graph Isomorphism

**Huge problem**

Even if $G_1$ and $G_2$ are isomorphic, we are not guaranteed to obtain isomorphic tree decompositions.

Idea 1:
Try to make the algorithm invariant (avoid arbitrary choices in the algorithms). Not known how to do this already for bounded-treewidth graphs.

Idea 2:
Use the more general notion of treelike decompositions and try to find such decompositions in an invariant way.

# Graph Isomorphism

## Huge problem

Even if $G_1$ and $G_2$ are isomorphic, we are not guaranteed to obtain isomorphic tree decompositions.

Idea 1:

Try to make the algorithm invariant (avoid arbitrary choices in the algorithms). Not known how to do this already for bounded-treewidth graphs.

Idea 2:

Use the more general notion of treelike decompositions and try to find such decompositions in an invariant way.

# Graph Isomorphism

## Huge problem

Even if $G_1$ and $G_2$ are isomorphic, we are not guaranteed to obtain isomorphic tree decompositions.

Idea 1:

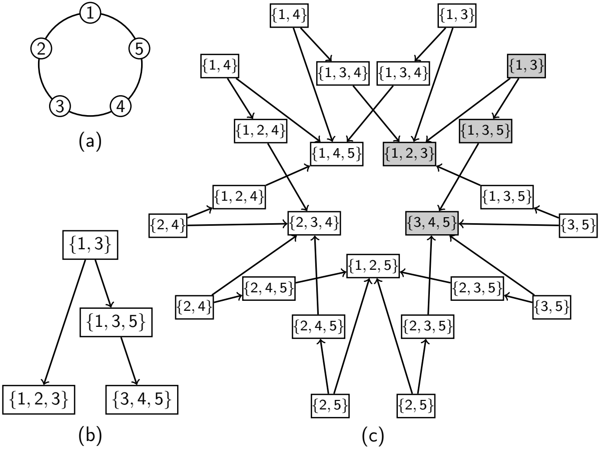
Try to make the algorithm invariant (avoid arbitrary choices in the algorithms). Not known how to do this already for bounded-treewidth graphs.

Idea 2:

Use the more general notion of treelike decompositions and try to find such decompositions in an invariant way.

# Treelike decompositions

[Grohe 2008] generalized the notion of tree decompositions to acyclic treelike decompositions:



# Graph Isomorphism

**New result**

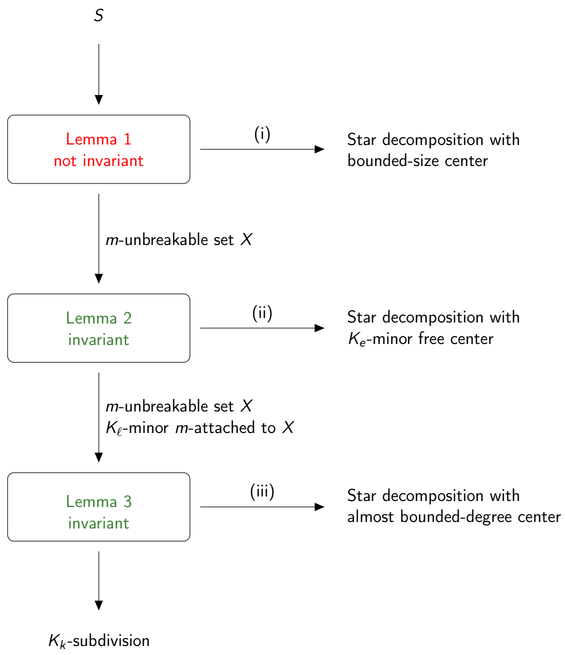
Every $H$-subdivision free graph has a tree decomposition where the torso of every bag is either

- "$c_H$-almost-embeddable" or
- has degree at most $c_H$ with the exception of at most $c_H$ vertices ("almost bounded degree").

**Theorem**

We can compute such a treelike decomposition in time $n^{f(H)}$ such that for isomorphic graphs we create isomorphic decompositions.

Now the difficulty disappears: we can compute a canonical label with a bottom-up dynamic programming approach.


$S$
Lemma 1
not invariant
(i)
Star decomposition with
bounded-size center
$m$-unbreakable set $X$
Lemma 2
invariant
(ii)
Star decomposition with
$K_e$-minor free center
$m$-unbreakable set $X$
$K_\ell$-minor $m$-attached to $X$
Lemma 3
invariant
(iii)
Star decomposition with
almost bounded-degree center
$K_k$-subdivision

# Summary

- Structure theorem for decomposing $H$-subdivision free graphs into almost-embeddable and almost bounded-degree graphs.
- Algorithmic applications on $H$-subdivision free graphs:
  - $f(k, H) \cdot n^{O(1)}$ time algorithm for Partial Dominating Set.
  - $n^{f(H)}$ time algorithm for Graph Isomorphism.